

The background of the cover is decorated with numerous circular images of various satellite-derived maps and data visualizations. These include topographic maps, land use maps, vegetation indices, and global maps, all arranged in a scattered pattern around the central text.

РОБОЧИЙ ЗОШИТ З ОСНОВ ДИСТАНЦІЙНОГО ЗОНДУВАННЯ ЗЕМЛІ

Частина 3

ОБРОБКА ТА АНАЛІЗ
СУПУТНИКОВИХ ЗНІМКІВ
НА ПЛАТФОРМІ
GOOGLE EARTH ENGINE

Київ
2023

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
НАЦІОНАЛЬНА АКАДЕМІЯ НАУК УКРАЇНИ
НАЦІОНАЛЬНИЙ ЦЕНТР «МАЛА АКАДЕМІЯ НАУК УКРАЇНИ»

С. М. Бабійчук, О. В. Гордієнко, О. В. Томченко, Н. С. Коблюк,
В. І. Голод, Т. Л. Кучма, Л. Я. Юрків, С. Т. Пікуль

РОБОЧИЙ ЗОШИТ З ОСНОВ ДИСТАНЦІЙНОГО ЗОНДУВАННЯ ЗЕМЛІ

Частина 3

**Обробка та аналіз супутникових знімків
на платформі Google Earth Engine**

За редакцією
академіка НАН України
С. Довгого

Київ
Національний центр
«Мала академія наук України»
2023

Авторський колектив:

С. М. Бабійчук – завідувачка лабораторії «Геоінформаційні системи та дистанційне зондування Землі» НЦ «МАНУ», кандидатка педагогічних наук;

О. В. Гордієнко – молодший науковий співробітник відділу природних ресурсів Інституту телекомунікацій та глобального інформаційного простору Національної академії наук України, спеціаліст із геоінформаційних систем «Sensilize», методист II категорії лабораторії «ГІС та ДЗЗ» НЦ «МАНУ»;

О. В. Томченко – старша наукова співробітниця Державної установи «Науковий центр аеро-космічних досліджень Землі Інституту геологічних наук Національної академії наук України», методистка II категорії лабораторії «ГІС та ДЗЗ» НЦ «МАНУ», кандидатка технічних наук;

Н. С. Коблюк – методистка «ГІС та ДЗЗ» НЦ «МАНУ»;

В. І. Голод – офіцер відділу розробки прикладного програмного забезпечення та інновацій у сфері інформаційних технологій Центру інновацій та розвитку оборонних технологій Міністерства оборони України;

Т. Л. Кучма – старша викладачка Національного університету «Києво-Могилянська академія», старша наукова співробітниця Інституту агроекології і природокористування НААНУ, кандидатка сільськогосподарських наук;

Л. Я. Юрків – спеціалістка ГІС у міжнародній неурядовій громадській організації «IMPACT Initiatives»;

С. Т. Пікуль – методист II категорії лабораторії «ГІС та ДЗЗ» НЦ «МАНУ»

Рецензенти:

Є. С. Аннілова – старша наукова співробітниця відділу природних ресурсів Інституту телекомунікацій та глобального інформаційного простору Національної академії наук України, кандидатка технічних наук;

А. П. Біатов – член громадської організації «Українська природоохоронна група», голова громадської організації «Природоохоронні ГІС України»

*Рекомендовано науково-методичною радою
Національного центру «Мала академія наук України»
(протокол № 3 від 26 жовтня 2022 р.)*

P58 **Робочий** зошит з основ дистанційного зондування Землі. Частина 3. Обробка та аналіз супутникових знімків на платформі Google Earth Engine / С. М. Бабійчук, О. В. Гордієнко, О. В. Томченко та ін.; за ред. С. О. Довгого. – Київ : Національний центр «Мала академія наук України», 2023. – 200 с.

ISBN 978-617-7945-09-2

Робочий зошит є третьою частиною комплексного збірника практичних робіт із курсу «Основи дистанційного зондування Землі», він присвячений формуванню компетентностей з обробки та аналізу супутникових знімків на платформі Google Earth Engine.

Практичні роботи, що містяться в робочому зошиті, укладені з метою ознайомити читача з можливостями аналізу супутникових знімків за допомогою потужностей хмарного сервісу Google Earth Engine, який надає вільний доступ до більш ніж 40 петабайтів даних супутникового моніторингу Землі.

Робочий зошит може використовуватися науковцями, освітянами, зокрема керівниками секції «ГІС та ДЗЗ» системи Малої академії наук України, а також усіма, хто прагне самостійно опанувати основи дистанційного зондування Землі та геоінформаційних систем.

УДК 528.8

© Бабійчук С. М., Гордієнко О. В.,
Томченко О. В., Коблюк Н. С., Голод В. І.,
Кучма Т. Л., Юрків Л. Я., Пікуль С. Т., 2023
© Національний центр
«Мала академія наук України», 2023

ЗМІСТ

Вступне слово.....	4
Вступ до Google Earth Engine. Синтаксис JavaScript для визначення площі території України.....	5
Робота з растровими даними на прикладі вибору супутникового знімка, комбінація каналів та завантаження на території пожежі в Дарницькому лісі	18
Завантаження векторних даних та топографічних мап у робоче середовище, їх аналіз і оцифрування на прикладі дослідження зміни лісу в Чорнобильській зоні	32
Створення графіків на основі тематичних даних для дослідження розподілу кількості опадів та вологості в Олешківських пісках	51
Аналіз рельєфу із застосуванням моделювання підтоплених територій України внаслідок підняття рівня океану. Створення профілю і відмивки рельєфу г. Говерла	63
Візуальне порівняння радарних і спектральних супутникових зображень, створення водної маски для дослідження паводка на р. Дністер	80
Вивантаження узагальнених середньостатистичних даних у форматі таблиці в результаті аналізу зміни температурних показників на території України	92
Розрахунки спектральних індексів NDVI та NDWI для моніторингу змін стану рослинного покриву на Кримському півострові.....	104
Розрахунок втрати лісу з візуалізацією у користувацькому інтерфейсі на основі даних Global Forest Change для дослідження негативних змін лісового покриву України.....	116
Класифікування водойм за допомогою методу машинного навчання Random Forest на території Шацьких озер.....	125
Візуалізація растрових та векторних даних з відображенням легенди на мапі для дослідження впливу промислових об'єктів на стан атмосферного повітря.....	137
Створення композитних анімаційних зображень із різних супутникових даних для відображення карстових процесів у районі Солотвино	152
Створення анімаційних зображень часових рядів вегетаційного індексу на прикладі аналізу динаміки лісового покриву	165
Створення інтерактивного застосунку з метою дослідження верхньої межі лісу в Карпатах	177

Вступне слово

Один з напрямів ІТ, який активно розвивається останнім часом, це збереження, обробка й аналіз великих даних у хмарних сервісах. Сприятливим підґрунтям для цього є майже повсюдний доступ до швидкісного інтернету і запит компаній на додаткові обчислювальні потужності, які значно дешевше й екологічніше покрити завдяки роботі в хмарі, аніж придбавати десятки чи сотні «потужних» персональних комп'ютерів.

Ми уклали цей робочий зошит для ознайомлення наукової та освітянської спільноти України з найпотужнішим на сьогодні хмарним сервісом обробки даних дистанційного зондування Землі (далі – ДЗЗ) – Google Earth Engine. Кількість даних, до яких нині надає доступ цей хмарний сервіс, становить понад 40 петабайтів (це близько 40 000 терабайтів) інформації, і ця кількість росте щосекунди. Супутникові знімки місії NASA, Європейського космічного агентства, десятки комерційних місій – доступ до всіх цих даних можна отримати через Google Earth Engine (далі – GEE).

Важливою особливістю такого сервісу є те, що ви ніяк не обмежені операційним функціоналом, оскільки в ньому є мінімальний набір вмонтованих інструментів аналітики, а весь об'єм роботи в сервісі ви виконуєте за допомогою мови програмування. У цьому робочому зошиті ми використовуємо JavaScript (також є можливість використовувати Python, мову програмування R тощо, працюючи зі сторонніми застосунками).

Практичні завдання в робочому зошиті супроводжуються великою кількістю ілюстрацій – скринів екрана персонального комп'ютера, таблицями, де ми детально описуємо кожен рядок коду, а також є відеосупровід за QR-кодом наприкінці практичної роботи.

Ми рекомендуємо виконувати практичні роботи із цього робочого зошита після ознайомлення з двома попередніми частинами курсу «Основи дистанційного зондування Землі», а саме: «Основи ДЗЗ: історія та практичне застосування», «Основи ДЗЗ: аналіз космічних знімків у ГС». Це дасть змогу більш якісно підготуватися до опанування матеріалу цього видання.

Як і в попередніх робочих зошитах, ми розробили для вас кольорові підказки. Зокрема, ви зможете візуально розпізнати рубрики, як-от: **Визначення**, **Блок «Цікаво»**, **Завдання для перевірки**, **Додаткове завдання для перевірки**, **Важливо!**

Визначення
Блок «Цікаво»
? Завдання для перевірки
? Додаткове завдання для перевірки
✓ Важливо!	<i>Важливий текст виділено червоним курсивом</i>

Вступ до Google Earth Engine. Синтаксис JavaScript для визначення площі території України

Ситуація

Інформаційно-комунікаційна компетентність на сьогодні є невід'ємною складовою спеціаліста в більшості галузей життя. Однією з технологій, які впливають на її формування, є геоінформаційні системи та обробка просторової інформації. Нині такої інформації існує дуже багато, і щодня її об'єм збільшується на десятки петабайтів. Для спеціаліста, який працює з інформацією, важливо знати, де вона зберігається, а таких ресурсів може бути якщо не десятки і сотні, то тисячі. Компанія «Google» створює своє хмарне середовище, де розміщені дані з більш ніж 30-річними історичними зображеннями Землі й набори даних, які оновлюються щодня.¹ GEE – це на сьогодні найпотужніший інструмент для аналізу просторової інформації. API Earth Engine доступний на Python і JavaScript, що спрощує роботу в цьому хмарному середовищі Google. Використовувати GEE можна різноманітним чином – від розрахунку вегетаційних індексів до прогнозування паводків, від моніторингу метеорологічних аномалій до прогнозування руху айсбергів. Сьогодні ми з вами розглянемо деякі можливості GEE, ознайомимося з основними типами даних для JavaScript, напишемо свою першу «Привіт»-функцію і подивимося, хто краще порахував кордон України – Управління географічної служби США чи Продовольча та сільськогосподарська організація ООН (ФАО).

Державний кордон – лінія (і вертикальна площина, що проходить по цій лінії), що визначає межі території держави (суші, вод, надр і повітряного простору) і відповідно межі дії державного суверенітету. Дослідженням кордонів займається лімологія – один із напрямів політичної географії. Державні кордони поділяються на сухопутні, річкові, озерні, морські та повітряні. На сьогодні питання внесення та відображення відомостей про державний кордон України є дуже актуальним і викликає високий ступінь суспільного інтересу в Україні.² В наш час існує кілька датасетів, що містять відомості про державні кордони всього світу. У цій практичній роботі ми розглянемо різницю даних у двох датасетах.

LSIB 2017: Large Scale International Boundary Polygons – Управління географічної служби США надає набір даних про великомасштабні міжнародні кордони (LSIB). Детальна версія від 2013 р. отримана з двох інших наборів даних: лінійного векторного файлу LSIB та World Vector Shorelines (WVS) Національного агентства геопросторової розвідки. Внутрішні кордони відображають політику уряду США щодо кордонів, прикордонних суперечок та суверенітету. Дані берегової лінії WVS застаріли, і зазвичай вони зміщені від кількох сотень метрів до кілометра. Кожен об'єкт є багатокутною областю, оточеною внутрішніми кордонами і зовнішніми береговими лініями³.

FAO GAUL: Global Administrative Unit Layers 2015, Country Boundaries – рівні глобальних адміністративних одиниць (GAUL) – збирають і зберігають інформацію про адміністративні одиниці для всіх країн світу, роблячи свій внесок у стандартизацію набору геопросторових даних. GAUL підтримує глобальні рівні з уніфікованою системою кодування на рівні країни, першого (наприклад, області) і другого адміністративного рівня (наприклад, райони)⁴. Для України в середовищі GEE доступні дані до рівня областей.

Під час обробки геопросторової інформації в даних можуть з'являтися неточності, інформація може застаріти. Тому сьогодні ми порівнюємо два наведених вище датасети. За офіційними даними площа України становить 603,7 тисяч кілометрів квадратних, але ми використовуємо глобальну систему координат, де площі можуть спотворюватися – все залежить від проєкції та системи координат, які ми вибираємо.

¹ URL: <https://earthengine.google.com/>

² URL: <https://web.archive.org/web/20160304214810/http://wdc.org.ua/atlas/1060000.html>

³ URL: https://developers.google.com/earth-engine/datasets/catalog/USDOS_LSIB_2017

⁴ URL: https://developers.google.com/earth-engine/datasets/catalog/FAO_GAUL_2015_level0

Завдання

Ознайомитися з базовим синтаксисом JavaScript. У цьому завданні ми розглянемо такі типи даних, які властиві мові програмування, зокрема JavaScript, тобто числа (1, 42, 3.14), текст (Привіт МАНівці), дати (1990,08,24), змінні, що називаються булевими змінними (можуть набувати двох значень: **True** (Істинне), **False** (Хибне), нульовий тип даних (NULL не є числом, це ніщо), масив (використовуються для того, щоб звернутися до об'єкта, що перебуває всередині об'єкта. Наприклад: Україна – Київська обл., Океани – Тихий, Антарктика – станція «Академік Вернадський»). Потім ми змінимо вигляд мапи на **Hybrid** (Гібрид). Останнім кроком є відобразити два датасети: LSIB 2017 і FAO GAUL та порівняти їх площі.

Алгоритм виконання завдання:

1. Зареєструватися й увійти на сервіс Google Earth Engine.
2. Ознайомитись з інтерфейсом Google Earth Engine.
3. Запустити покроково частини коду; подивитися що відбувається; задати деякі змінні, назви тощо.
 - 3.1. Запустити частину коду із числами.
 - 3.2. Запустити частину коду з текстом.
 - 3.3. Запустити частину коду з датами, змінити дату на дату виконання практичного завдання.
 - 3.4. Запустити частину коду з булевими операторами та оператором NULL.
 - 3.5. Запустити частину коду з пов'язаним словником ключ-значення та масивом, звернутися до іншого ключ-значення.
4. Написати свою першу «Привіт»-функцію.
5. Змінити вид відображення мапи.
6. Скопіювати код відображення кордонів країни з датасетів FAO GAUL і запустити його, дослідити зміни на мапі.
7. Наблизитися до території інтересу на мапі.
8. Скопіювати та запустити частину коду, яка порахує площу для LSIB 2017. Змінити частину коду, яка розрахує площу території за допомогою даних LSIB 2017 на іншу змінну, що порахує площу для FAO GAUL.

Покрокова інструкція

1. Зареєструватися й увійти на сервіс Google Earth Engine.

Для початку роботи зареєструйтеся за покликанням: <https://signup.earthengine.google.com/#/>.

✔ **Важливо!** Усі дані необхідно вводити латинськими літерами. Рекомендуємо використовувати англійську мову і вказувати контактну пошту з доменом @gmail.com.

Після реєстрації вам надійде повідомлення від компанії «Google», в якому буде покликання на **Code Editor** (Редактор коду): <https://code.earthengine.google.com/> – це основне середовище розробки для Google Earth Engine; також редактор коду можна знайти на головній сторінці сайту Google Earth Engine, а саме в закладці **Platform** (Платформа) → **Code Editor** (Редактор коду).

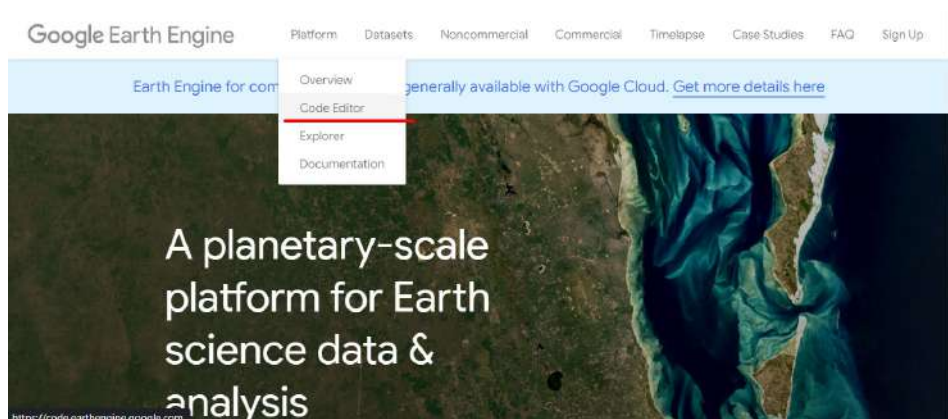


Рис. 1. Редактор коду на основній платформі

Натиснувши на лінк в пошті чи на сайті, ви перейдете до редактора коду, в якому ми будемо працювати.

2. Ознайомитися з інтерфейсом Google Earth Engine.

Відкривається нове вікно, де ви побачите **Code Editor** (Редактор коду), в якому ми будемо працювати увесь час. Це основне робоче середовище GEE.

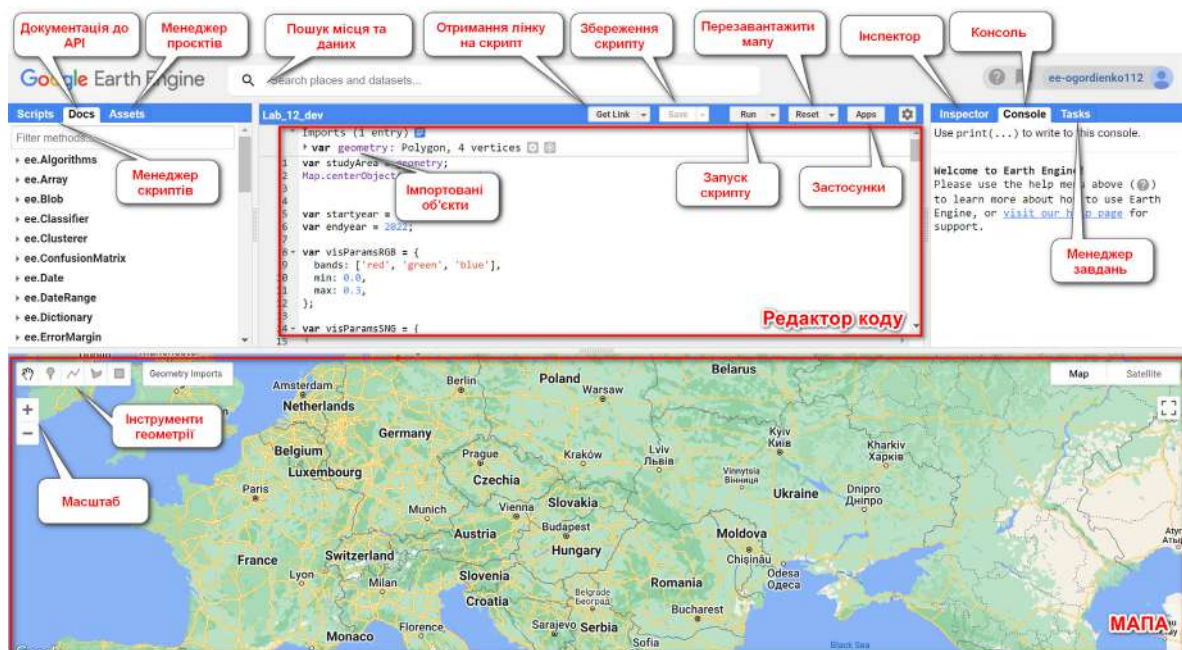


Рис. 2. Редактор коду. Зовнішній вигляд з поясненням

Map (Мапа) – займає найбільшу площу на сторінці. Саме на ній буде візуалізуватися результат.

Code Editor (Редактор коду) – головне вікно сервісу, в якому ви пишете свій скрипт.

Console (Консоль) – служить для виведення текстової інформації, результатів розрахунку, графіків тощо.

Tasks (Менеджер завдань) – у ньому можна бачити результати завантаження і вивантаження з GEE.

Inspector (Інспектор) – служить для отримання інформації за кліком, координати, значення в точці тощо.

Help (Допомога) – служить для зв'язку з компанією «Google» стосовно помилок у роботі програми.

Scripts manager (Менеджер скриптів) – у ньому зберігаються скрипти, які ви створювали, також приклади скриптів від Google та доступ до папок інших користувачів.

Docs or API Documentation (Документація до API) – повна документація з описом стосовно всіх команд у GEE.

Reset (Перезавантажити) – перезавантажує вид на мапі і результати в консолі, повністю очищає їх від результатів розрахунку, не видаляючи скрипт.

Apps (Застосунки) – дає змогу створити вебзастосунок на основі вашого скрипту, який буде працювати як сторінка в інтернеті.

Assets manager (Менеджер об'єктів) – служить для завантаження ваших власних даних до середовища GEE.

Geometry tools (Інструменти геометрії) – служать для навігації по мапі, а також для створення ваших власних векторних даних.

Scale tools or Zoom (Масштаб або зум) – служить для наближення до території інтересу на мапі.

Search for data (Пошук місця та даних) – служить для пошуку даних у середовищі GEE та місця на мапі за назвою чи координатами.

Imports (Імпорт) – тут зберігаються об'єкти, які імпортувалися з вашого скрипту.

Get link (URL) to the script (Отримання лінку на скрипт) – дає змогу ділитися своїм скриптом із користувачами GEE.

Save the script (Збереження скрипту) – дає змогу зберігати скрипт.

Run the script (Запуск скрипту) – кнопка, що виконує скрипт, написаний у редакторі коду.

3. Запустити покроково частини коду; подивитися, що відбувається; задати деякі змінні, назви тощо.

Для створення змінної, в яку можна покласти будь-які типи даних, у Google Earth Engine використовується оператор **var**. Створимо змінну з ім'ям **variable_int** і надамо їй значення **42**, також створимо змінну з ім'ям **variable_float** та значенням **2.718281828**, виведемо результат на екран. Для цього нам знадобиться команда **print** (друк), в якій перелічимо імена двох попередніх змінних. Скористаємося кодом:

3.1. Запустити частину коду із числами.

<pre>var variable_int = 42;</pre>	Змінна зі значенням 42 та ім'ям variable_int , тип даних – ціле число;
<pre>var variable_float = 2.718281828;</pre>	Змінна зі значенням 2.718281828 та ім'ям variable_float , тип даних – число з плаваючою крапкою;
<pre>print(variable_int, variable_float);</pre>	Вивести результат у консоль, через кому перелічені попередні змінні;

У Google Earth Engine команда **print** (друк) виводить результат у консоль.

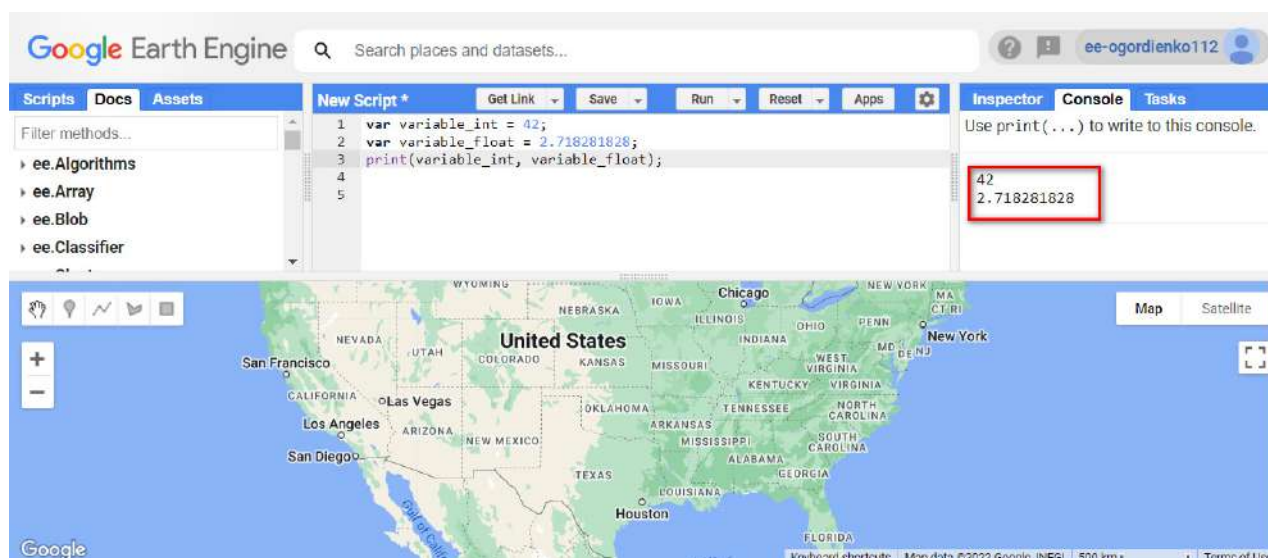


Рис. 3. Виведення змінної з числом

У консолі ви можете бачити, як відображаються цифри в змінних, які ми щойно ввели в редактор коду.

Наступним типом даних, що можуть бути використані в Google Earth Engine, є **текст**, він береться в одинарні лапки (' **ваш текст** ') або подвійні (" **ваш текст** "). Але вибирайте лише один тип лапок, не змішуйте їх.

3.2. Запустити частину коду з текстом.

<pre>var my_variable = 'Це текст'; var my_other_variable = "Це також текст"; print(my_other_variable);</pre>	Змінна my_variable з ТЕКСТОМ ; Змінна my_other_variable з ТЕКСТОМ ; Результат у консоль змінної my_other_variable ;
--	--

Команда **print** (друк) виводить результат у консоль.

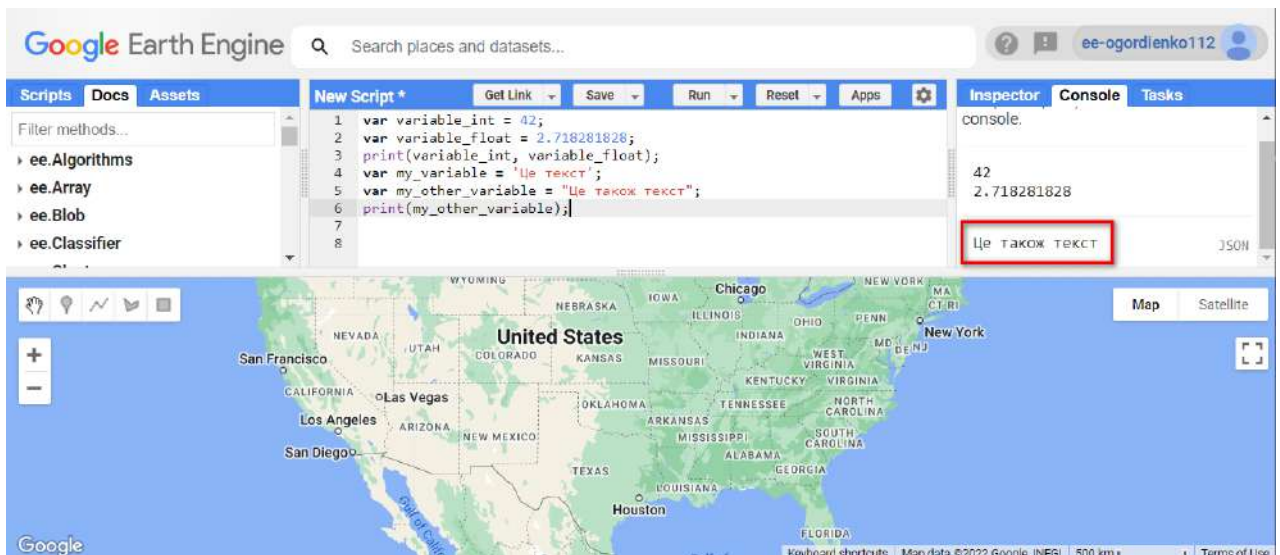


Рис. 4. Виведення змінної з текстом

Також можна використовувати такий тип даних, як дати. Вони можуть задаватися різними способами. У цій практичній роботі ми поглянемо, як працює один із них.

3.3. Запустити частину коду з датами.

<pre>var date = ee.Date.fromYMD(2021,2,19); print (date);</pre>	<p><code>ee.Date.fromYMD</code> стандартна конструкція в GEE, де в дужках задається рік, місяць, день;</p> <p>Результат у консоль;</p>
--	--

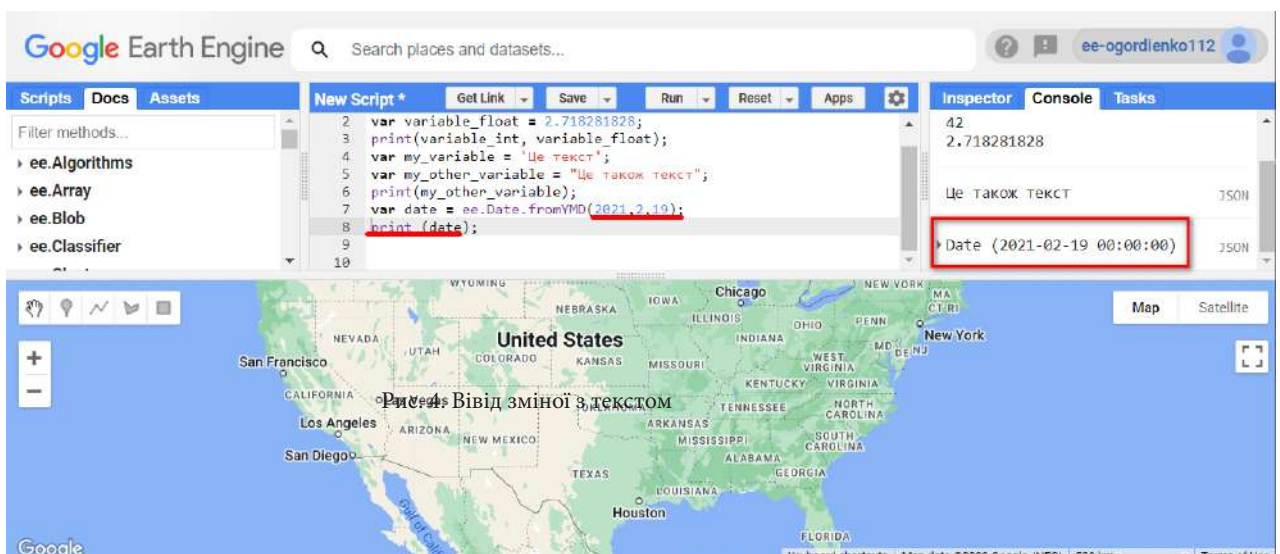


Рис. 5. Виведення змінної з датою

? Завдання для перевірки

Задайте дату виконання практичного завдання

Також стандартними структурами даних є булеві оператори, які можуть набувати лише двох значень: **True** (Істинне) і **False** (Хибне). Окремим від цих двох операторів є оператор **null**, що набуває значення **null** – тобто нічого. Виведемо їх у консоль за допомогою команди **print** (друк).

3.4. Запустити частину коду з булевими операторами та оператором NULL.

<pre>var bool_variable = true; //or false print (bool_variable); var null_variable = null; print (null_variable);</pre>	<p>Цей булевий оператор може набувати двох значень: True (Істинне), False (Хибне);</p> <p>Результат у консоль;</p> <p>Цей оператор набуває значення null, тобто нічого; не слід плутати із числовим значенням 0;</p> <p>Результат у консоль;</p>
---	---

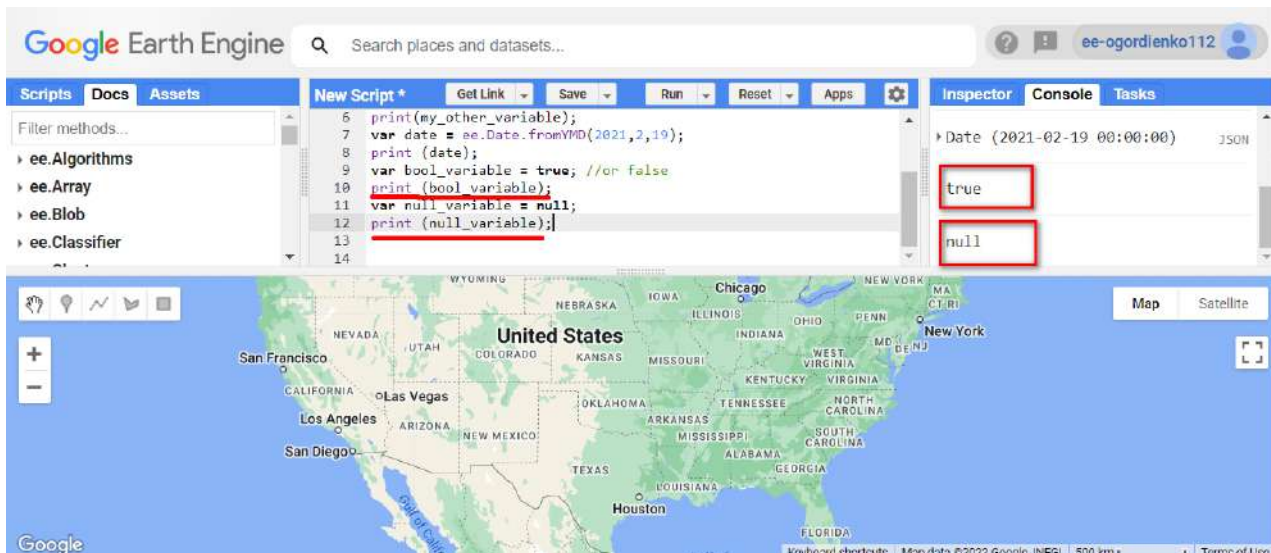


Рис. 6. Виведення змінної з булевим оператором

Ще одним стандартним об'єктом є масив і пов'язаний словник. У них можуть розміщуватися значення, і за допомогою ключ-значення можна звертатися до пов'язаних словників і об'єктів, які в них розміщені. За допомогою номера – до об'єкта, що перебуває в масиві.

Масив – це сукупність однотипних даних, розташованих безперервно в пам'яті. Доступ до елемента здійснюється за індексом $O(1)$ – ми звертаємося безпосередньо до потрібної ділянки пам'яті.

Пов'язаний словник. Доступ до елемента в пов'язаному списку в середньому здійснюється $O(N)$ шляхом перебору елементів у пошуках потрібного.

Створимо масив *my_list*, де будуть зберігатися три значення, і пов'язаний словник *my_dict*, де будуть також три значення за типом ключ-значення.

3.4. Запустити частину коду з пов'язаним словником ключ-значення і масивом, звернутися до іншого ключ-значення.

✓ **Важливо!** У програмуванні нумерація починається з 0.

<pre>var my_list = ['Шевченко', 'Taras', 123]; print(my_list[1]); var my_dict = {"food": "bread", "color": "white", "number": false}; print(my_dict['color']); print(my_dict.color);</pre>	<p>Масив, у якому три значення, нумерація починається з 0;</p> <p>Результат у консоль;</p> <p>Пов'язаний словник – це також три значення, але вони являють собою тип ключ-значення;</p> <p>Результат у консоль</p>
--	--

У GEE більшість даних, тобто їх структуру, влаштовано так: ключ-значення.

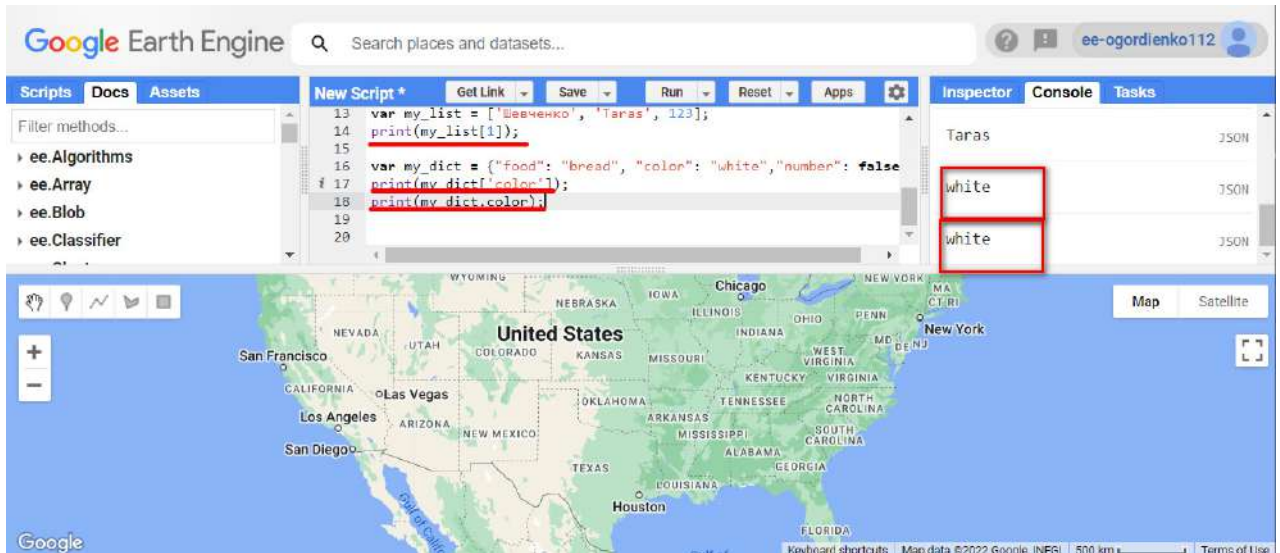


Рис. 7. Виведення змінної з масиву

Звернутися до масиву можна за допомогою квадратних дужок і знаючи номер об'єкта в списку `print(my_list[1])`.

Звернутися до пов'язаного словника можна за допомогою квадратних дужок `print(my_dict['color'])` і точкової нотації `print(my_dict.color)`.

? Завдання для перевірки

Виведені значення ідентичні. Тут ви можете спробувати звернутися до іншого значення: виведіть на екран значення для `food` за допомогою точкової нотації. Точкова нотація – це спосіб звернутися до значення в пов'язаному словнику, знаючи його ім'я. Виконується вона через знак крапки

Для цього змініть значення `print(my_dict.color)`; на `print(my_dict.food)`;

Наступним важливим об'єктом є функція. Створімо просту функцію, за допомогою якої виведемо привітання.

4. Написати свою першу «Привіт»-функцію.

<pre> var my_hello_function = function(a) { return 'Привіт' + a + '!'; }; print(my_hello_function('МАНівці!')); </pre>	<p>Функція <code>my_hello_function</code>, повертає значення <code>'Привіт' + a + '!'</code>; де <code>'a'</code> – довільне значення;</p> <p>Результат у консоль зі своїм значенням <code>'МАНівці!'</code>;</p>
--	---

Функція `my_hello_function` від `(a)` має статичне значення `'Привіт'`, до якого додає змінну, яку ви задаєте вже під час виведення команди на екран. На вхід функція бере ваше значення, а на вихід подає статичне значення, плюс ваше значення.

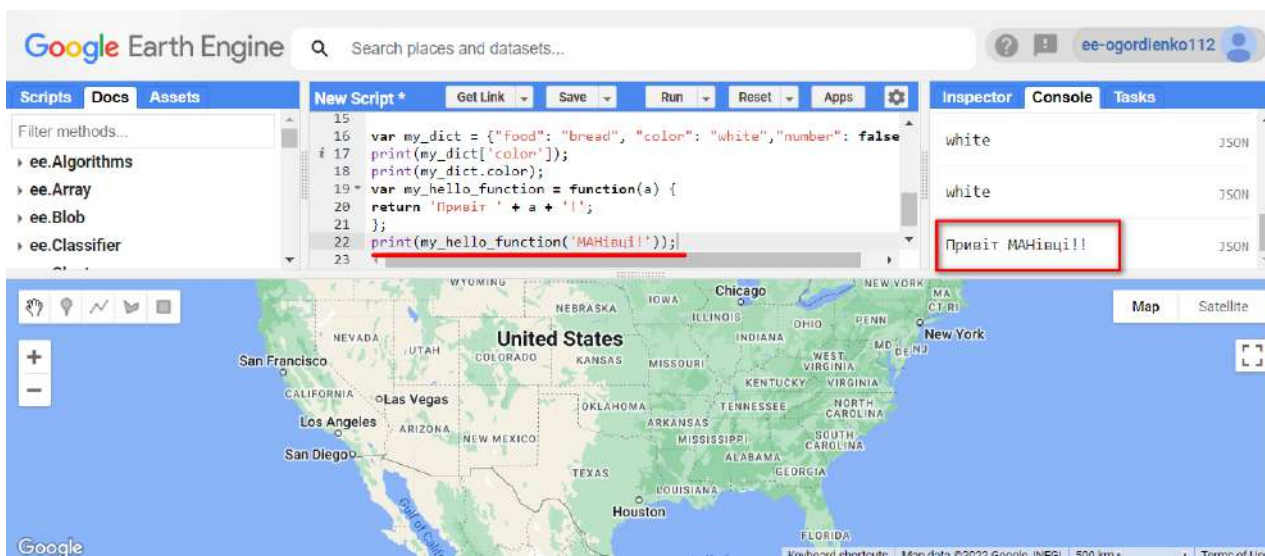


Рис. 8. Виведення привітання

GEE також дає змогу змінювати відображення мапи за замовчуванням, для цього змінимо відображення на гібрид.

Гібрид ('HYBRID') – це знімок земної поверхні, зроблений з космосу з накладанням інформації про назви населених пунктів, вулиць, адреси та ін.

5. Змінити вид відображення мапи.

<code>Map.setOptions('HYBRID');</code>	Відображення зміниться на 'HYBRID';
--	-------------------------------------

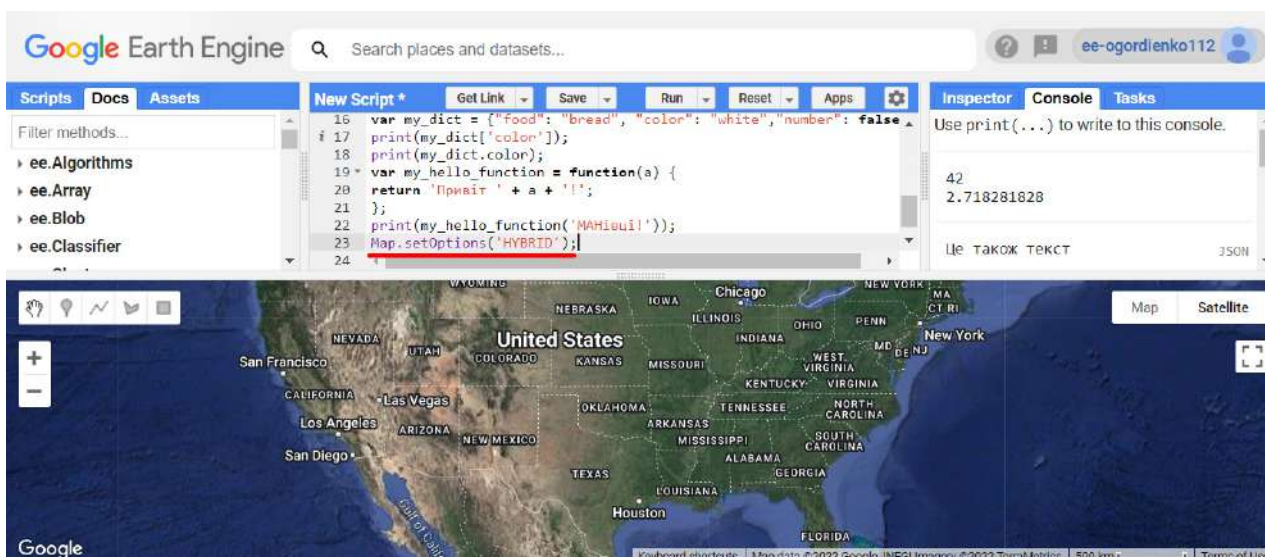


Рис. 9. Зміна вигляду мапи

Це робиться за допомогою команди `Map.setOption` у дужках `()`, і далі в лапках `"` подається вид мапи, який ми хочемо використовувати. Вона може набувати таких значень: HYBRID, ROADMAP, SATELLITE, TERRAIN.

🔍 Завдання для перевірки

Змініть вид мапи на HYBRID, ROADMAP, SATELLITE, TERRAIN. Як змінилася мапа?

До цього ми працювали зі стандартними даними, які є в більшості мов програмування. Розгляньмо дані, які є в GEE. Підвантажимо кордон України з різних джерел даних і порівняймо їх.

У цьому рядку коду ми звертаємося до датасету, а саме до нового типу даних, що притаманний саме GEE **Feature Collection** (Колекція об'єктів), це колекція векторних даних.

Для початку використаємо датасет **LSIB 2017: Large Scale International Boundary Polygons, Simplified**. Цей датасет розроблений географічною службою США, і в ньому містяться кордони країн станом на 2017 р.

6. Скопіювати код відображення кордонів країни з датасетів FAO GAUL і запустити його, дослідити зміни на мапі.

<pre> var countriesUSDOS = ee.FeatureCollection("USDOS/LSIB_SIMPLE/2017"); var UkraineUSDOS = countriesUSDOS. filter(ee.Filter.eq('country_na', 'Ukraine')); var styleParams_1 = { fillColor: 'FF000000', color: 'FF0000', width: 1.0, }; var coloredUSDOS = UkraineUSDOS. style(styleParams_1); Map.addLayer(coloredUSDOS, {}, 'Ukraine Red', true); </pre>	<p>Змінна countriesUSDOS звертається до колекції за шляхом USDOS/LSIB_SIMPLE/2017;</p> <p>Змінна UkraineUSDOS фільтрує зображення, які беруться за типом ключ-значення ('country na - 'Ukraine');</p> <p>Візуалізація даних, де styleParams_1 використовує fillColor: 'FF000000', як без кольору, а color: 'FF0000', - червоний, width: 1.0, товщина лінії;</p> <p>Застосування параметрів візуалізації до відфільтрованого датасету;</p> <p>Додавання на мапу шару з візуалізацією за замовчуванням {} та ім'ям 'Ukraine Red' і відображенням true;</p>
---	--

Задаймо параметри візуалізації. Один зі способів вибору кольорів – це задавати їх у системі HEX.

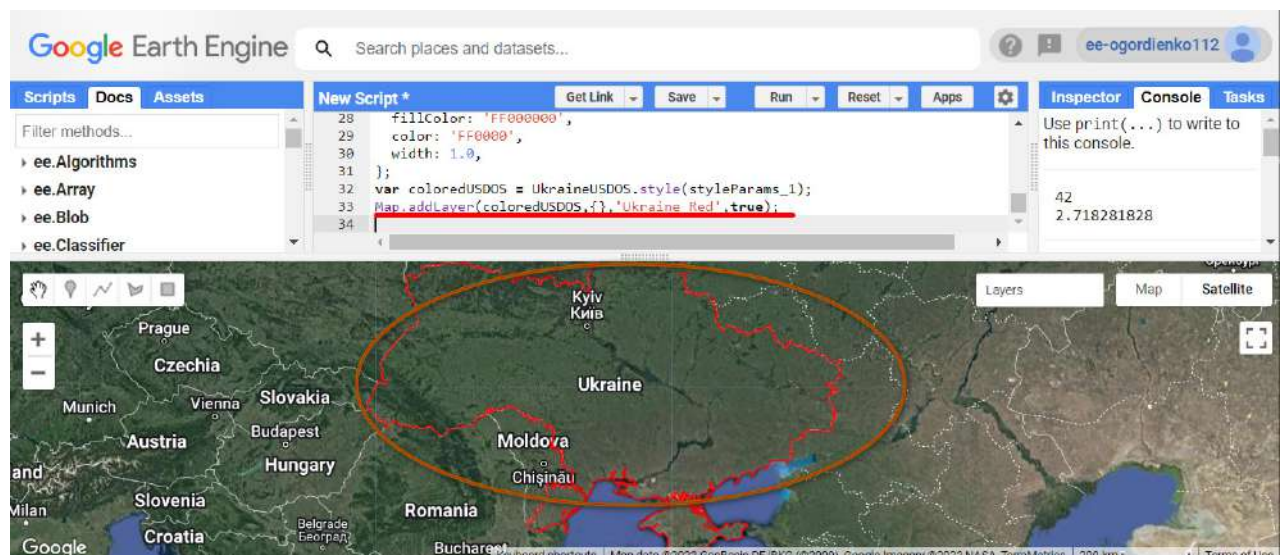


Рис. 10. Зміна кольору кордону

7. Наблизитися до території інтересу на мапі.

Також на цьому етапі можна призумитися до нашого шару. Зробити це можна так:

<pre>Map.centerObject(UkraineUSDOS,5);</pre>	<p>Де перше значення – шар, до якого ми хочемо призумитися, друге значення – рівень екстенції мапи;</p>
--	---

На мапі має з'явитися Україна з її кордоном.

Так само використаємо подібний код, але змінимо там джерело даних, візьмемо кордони від датасету **FAO GAUL** (Food and Agriculture Organization та Global Administrative Unit Layers), створеного в рамках проекту Продовольчої та сільськогосподарської організації ООН (FAO). Цей датасет розроблений на різних рівнях: від глобальних (кордони країн) до локальних кордонів (області) й оновлюється раз на рік.

Візуалізуємо їх зеленим кольором, просто додавши код.

8. Скопіювати і запустити частину коду, яка порахує площу для LSIB 2017.

<pre> var countriesFAO = ee.FeatureCollection("FAO/GAUL/2015/level0"); var UkraineFAO = countriesFAO.filter(ee.Filter.eq('ADM0_NAME','Ukraine')); var styleParams_2 = { fillColor: 'FF000000', color: '00FF00', width: 1.0, }; var coloredFAO = UkraineFAO.style(styleParams_2); Map.addLayer(coloredFAO, {}, 'Ukraine Green'); </pre>	<p>Змінна countriesFAO звертається до колекції за шляхом FAO/GAUL/2015/level0;</p> <p>Змінна UkraineFAO фільтрує зображення, які беруться за типом ключ-значення ('ADM0_NAME' - 'Ukraine');</p> <p>Візуалізація даних, де styleParams_1 використовує fillColor: 'FF000000', як без кольору, а color: '00FF00', - зелений, width: 1.0, товщина лінії;</p> <p>Застосування параметрів візуалізації до відфільтрованого датасету;</p> <p>Додавання на карту шару з візуалізацією за замовчуванням {}, та ім'ям 'Ukraine Green', і відображенням true;</p>
---	---

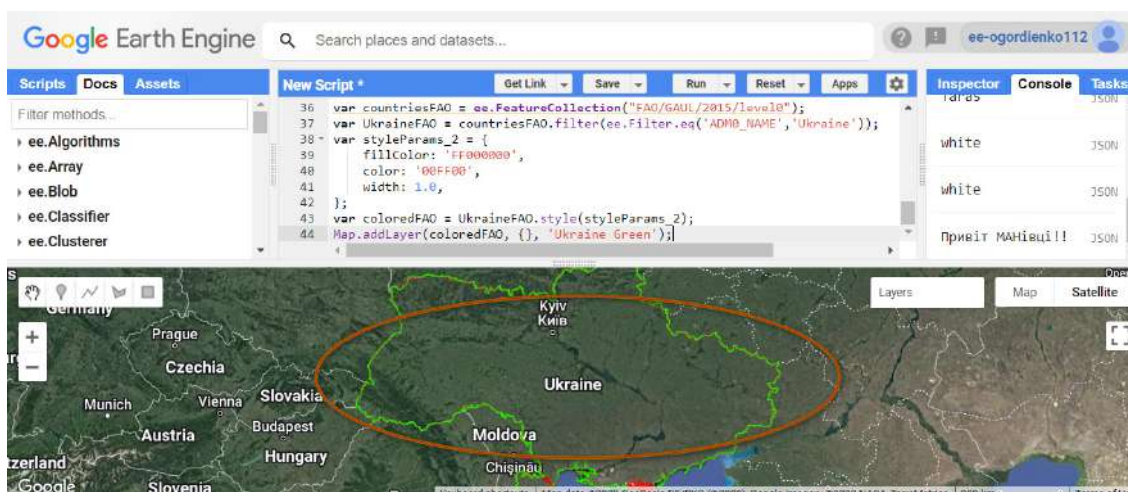


Рис. 11. Виведення даних кордону

Тепер у нас є два кордони України, і вони різні – це можна дослідити, призумившись за допомогою коліщатка миші, наприклад до Криму. Один з них відмальовує кордон по береговій лінії, інший має більш традиційний вигляд. Порівняймо їх площу.

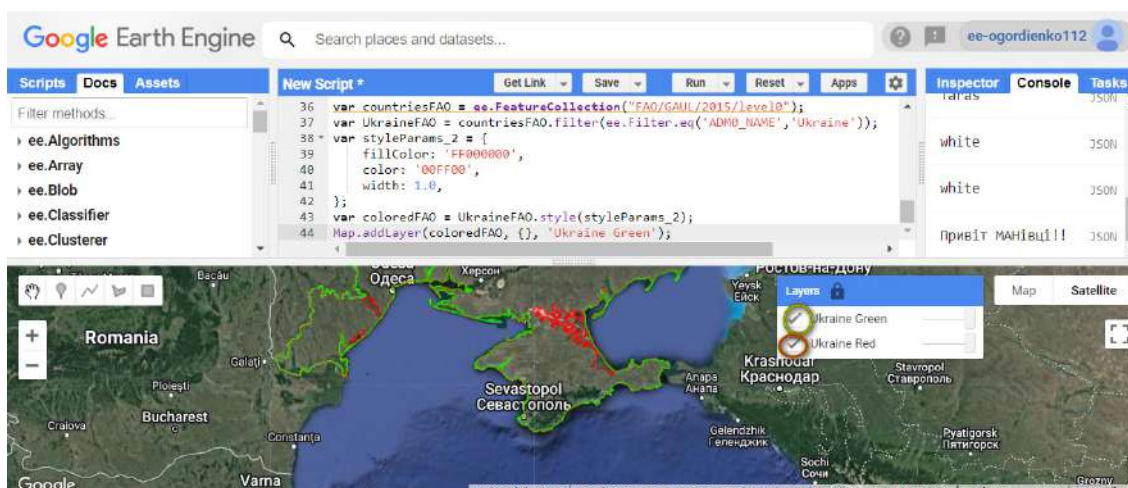


Рис. 12. Відображення даних

9. Змінити частину коду, яка розрахує площу території за допомогою даних LSIB 2017, на іншу змінну, що порахує площу для FAO GAUL.

Зараз ми порахуємо площу для одного з датасетів, а саме для змінної countriesUSDOS, тобто датасету LSIB 2017. Це можна зробити за допомогою такого коду:

<pre>var stateArea = UkraineUSDOS.geometry().area() var stateAreaSqKm = ee.Number(stateArea).divide(1e6).round() print(stateAreaSqKm, 'area')</pre>	<p>var stateArea – використовує відфільтроване зображення, потім бере з нього геометрію geometry(), та за допомогою area() рахує площу; stateAreaSqKm – переводить значення у квадратні кілометри за допомогою множення на 1 000 000 divide(1e6) та спрощує число до цілого за допомогою round();</p> <p>Виводить результат на екран;</p>
--	---

Результат підрахунку площі трохи не збігається з офіційними даними, бо системи координат різні.

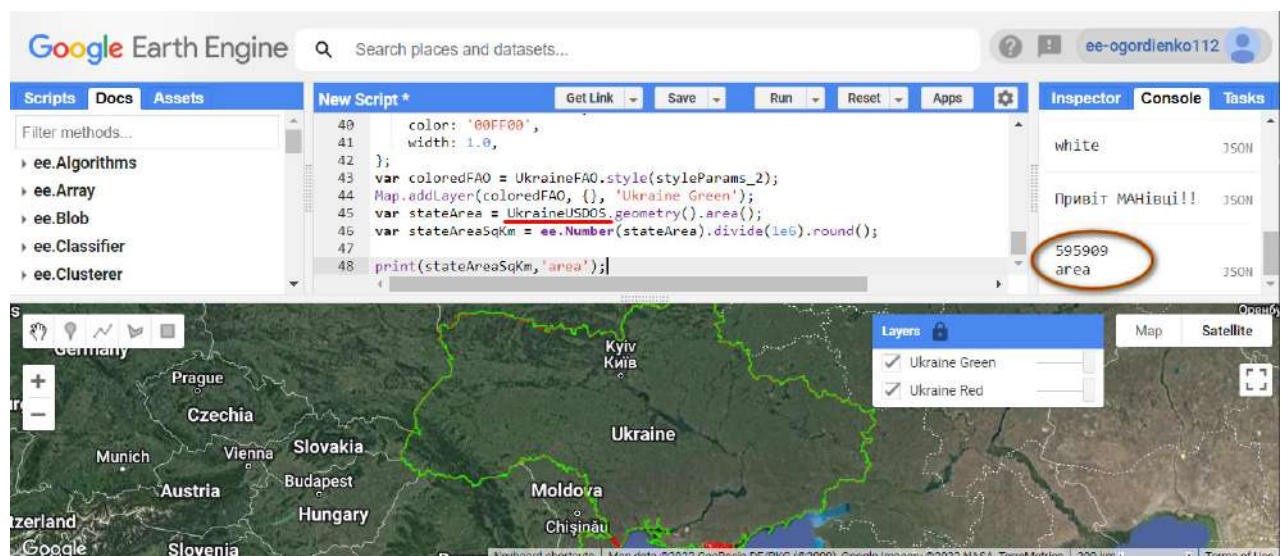


Рис. 13. Розрахунок площі для першої змінної

Тепер спробуйте порахувати площу для іншого датасету. Вам потрібно в коді вище змінити дані на інший відфільтрований датасет.

? Завдання для перевірки

<p>Напишіть, яку площу має кордон України в км² за цими датасетами</p>	<p>LSIB 2017</p>	<p>FAO GAUL</p>
<p>Чому, на вашу думку, змінилася площа? Який з наборів даних має точніший результат?</p>		

Також у GEE є можливість зберігати свої скрипти в папках та репозитаріях. Створіть репозитарій, до якого буде доступ у користувачів лабораторії «ГІС та ДЗЗ». Для цього перейдіть до вкладки **Scripts** (Скрипти) і натисніть на кнопку **NEW**, у випадному списку виберіть **Repository** (Репозитарій).

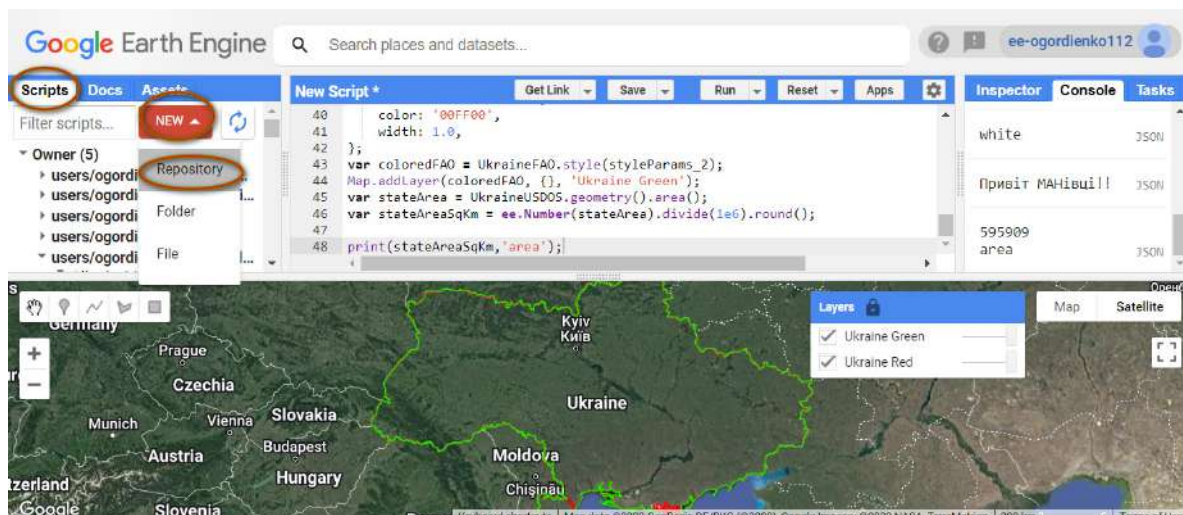


Рис. 14. Збереження даних у репозитарій

Після цього відкриється нове вікно. Введіть у ньому назву репозитарію MAN_lab.

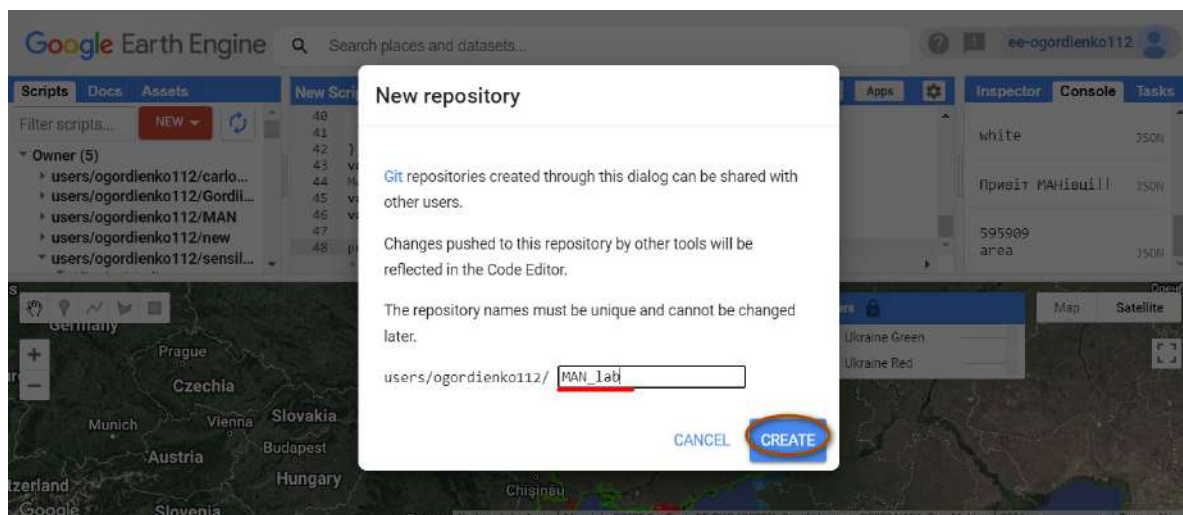


Рис. 15. Надавання назви репозитарію

Натисніть **Create** (Створити).

Цей репозитарій з'явиться в **Scripts** (Скрипті). Там вам потрібно навести на нього й вибрати іконку із шестернею.

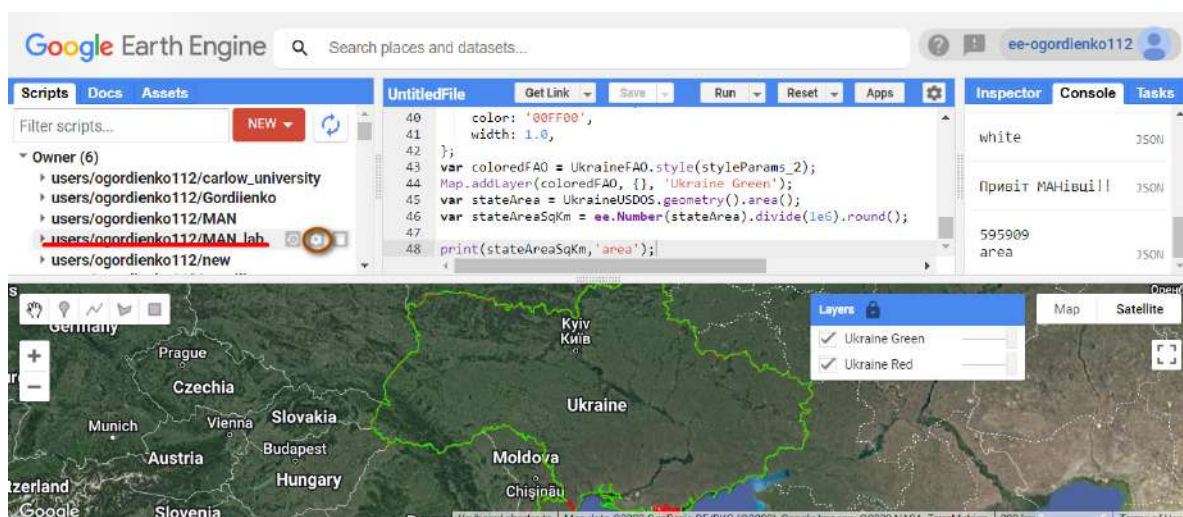


Рис. 16. Налаштування репозитарію

Натиснувши на неї, ви перейдете в налаштування доступу до цього репозитарію. Скопіюйте, будь ласка, поштову адресу: gis_rs@man.gov.ua, щоб ми мали доступ до ваших скриптів.

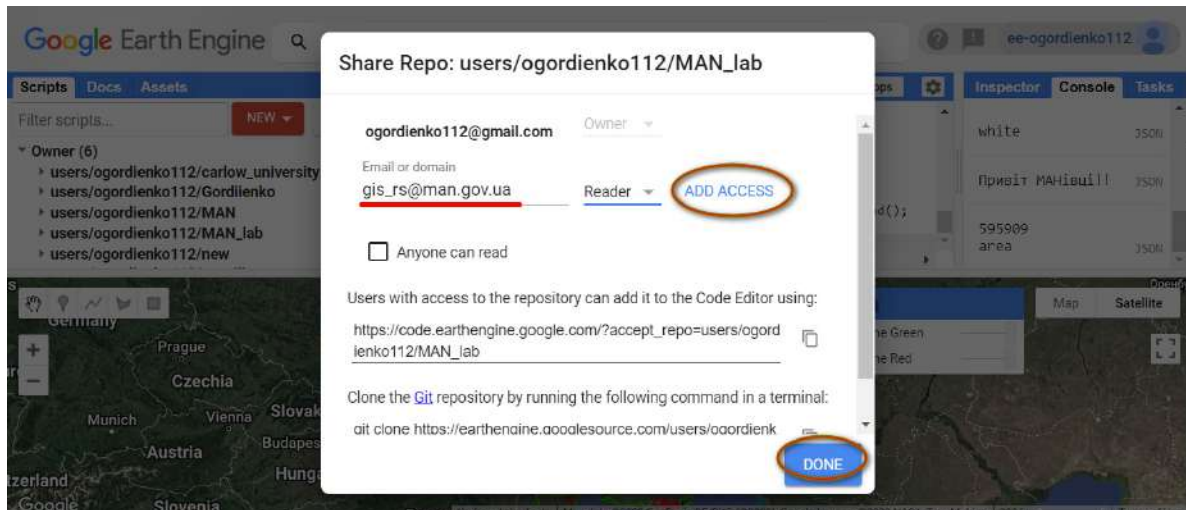


Рис. 17. Надання доступу до репозитарію

Натисніть **Done** (Виконано).

Тепер збережіть скрипт, якщо ви не робили цього на попередніх етапах. Для цього потрібно натиснути кнопку **Save** (Зберегти). З'явиться нове вікно зі шляхом, куди ви хочете зберегти файл, і з полем, де можна ввести ім'я файлу. Введіть назву **Lab1** і натисніть кнопку **OK**.

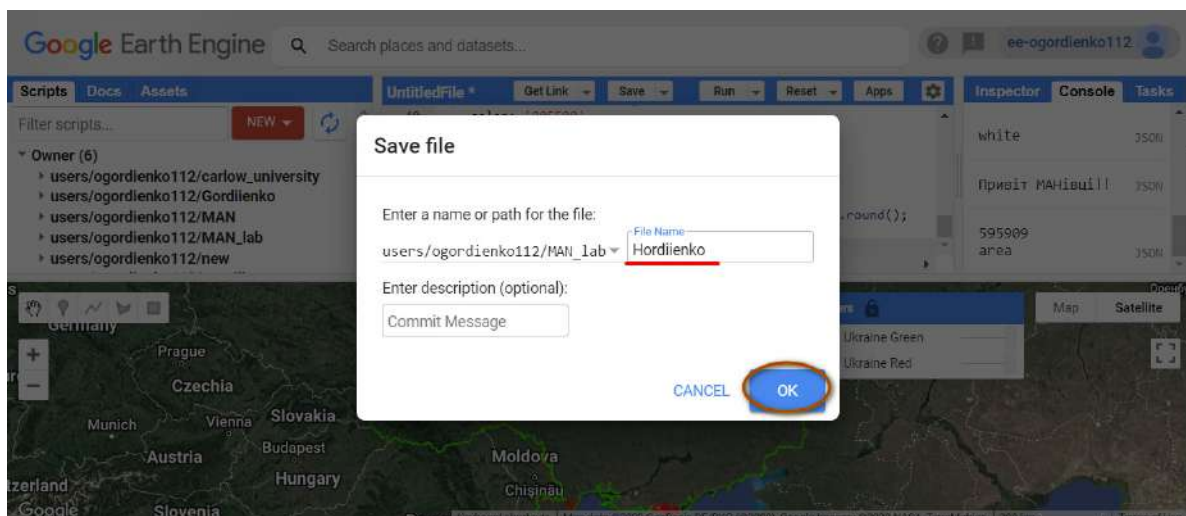


Рис. 18. Збереження скрипту

За наведеним нижче QR-кодом ви можете переглянути відео виконання практичної роботи.



Рис. 19. Відео виконання практичної роботи

Готовий код можна відкрити за лінком:

https://code.earthengine.google.com/?scriptPath=users%2Fgis_rs%2Fcourse_MAN%3ALab_1

Робота з растровими даними на прикладі вибору супутникового знімка, комбінація каналів та завантаження на території пожежі в Дарницькому лісі

Ситуація

У нашій країні щорічно в пожежах згорає 5–7 тисяч гектарів лісу і від 500 тисяч до одного мільйона гектарів сільськогосподарських земель. У 2020 р. статистика зросла як мінімум до 120 тисяч гектарів. За останні 30 років на території України середня температура повітря зросла на 1,2 градуса, а це збільшує ризики появи пожеж, посух і навіть опустелення. Проте, на жаль, перша причина вогню – це людський фактор. Кількість займань, які виникають природним шляхом, є сталою, а от кількість пожеж, які розвиваються внаслідок антропогенного фактору, значно збільшується¹. З огляду на це в нас виникла ідея знайти на супутниковому знімку один з осередків пожежі, що сталася в жовтні 2015 р. поблизу Києва в Дарницькому лісі.

Завдання

У цьому завданні ми створимо кольорове супутникове зображення м. Києва і знайдемо в Дарницькому лісі осередок пожежі на знімках, зроблених супутником (Sentinel-2) 4 жовтня 2015 р., обведемо осередок пожежі й завантажимо супутникові дані на свій Google-диск.

Алгоритм виконання завдання:

1. Відкрити головну сторінку Google Earth Engine.
 - 1.1. Знайти в Datasets знімки із супутника Sentinel-2, які пройшли атмосферну корекцію – *Top-of-Atmosphere Reflectance*.
 - 1.2. Скопіювати код, який представлений у датасеті *Top-of-Atmosphere Reflectance* для Sentinel-2 розробниками компанії «Google».
2. Змінити в скопійованому коді дати відбору знімків на дату пожежі в Дарницькому лісі, а саме із 4 жовтня по 5 жовтня 2015 р.
3. Змінити відображення космічного знімка, що за замовчуванням у коді від компанії «Google» був із природних кольорів: *Red, Green Blue*, на штучні кольори: *SWIR, NIR, Blue*.
4. Скористатися **Inspector** (Інспектор), щоб знайти координати Києва.
 - 4.1. Змінити наявні в коді координати на координати Києва, знайдені за допомогою **Inspector** (Інспектор).
5. Створити рамку (геометричний об'єкт) за допомогою інструментів геометрії для обрізання зображення.
6. Експортувати зображення знімка Sentinel-2 за 04.10.2015 р. на свій особистий Google-диск.
 - 6.1. Завантажити космічне зображення на свій особистий Google-диск.
7. Створити коментар зі своїм прізвищем.

Покрокова інструкція

1. Відкрити головну сторінку Google Earth Engine.

Для початку роботи знайдімо потрібні нам дані. Відкрийте головну сторінку Google Earth Engine і перейдіть до вкладки Datasets.

¹ URL: <https://uacrisis.org/uk/ukrayina-u-vogni-chomu-traplyayutsya-lisovi-pozhezhi-i-yak-yim-protydiyaty>

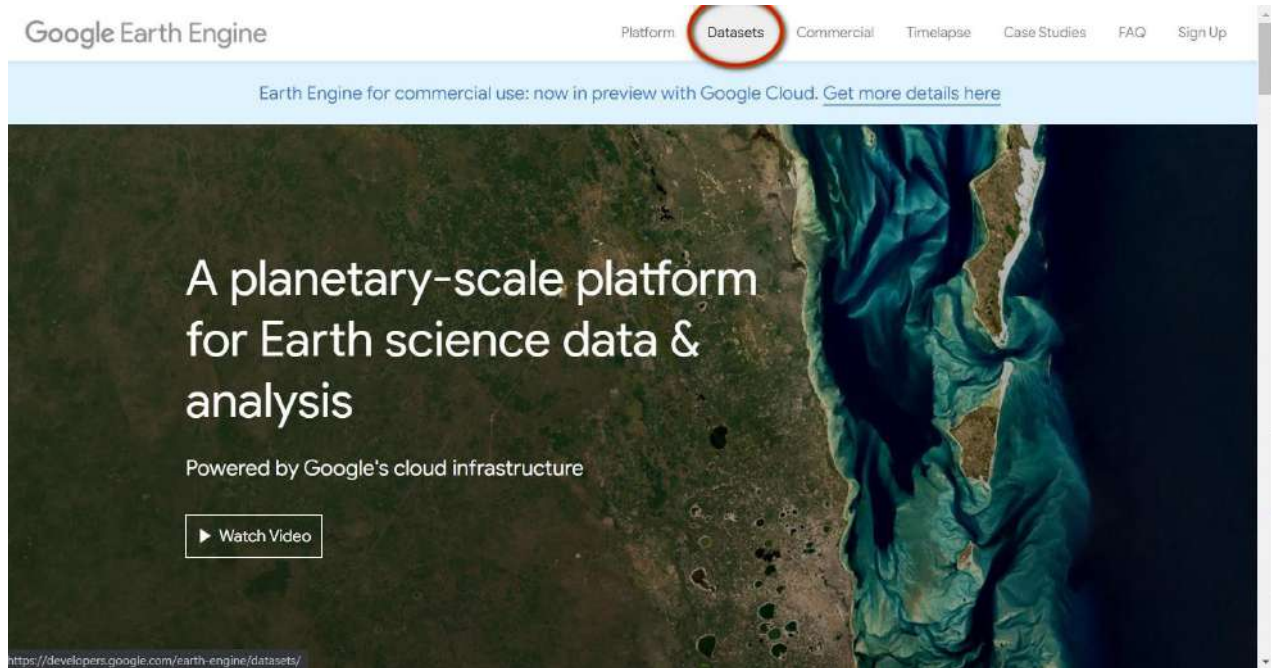


Рис. 20. Вкладка пошуку датасетів

1.1. Тепер знайдіть у Datasets знімки із супутника Sentinel-2, що пройшли атмосферну корекцію – Top-of-Atmosphere Reflectance.

У новому вікні, що відкрилося, прогорніть сторінку до датасету з ім'ям Sentinel і натисніть на нього.

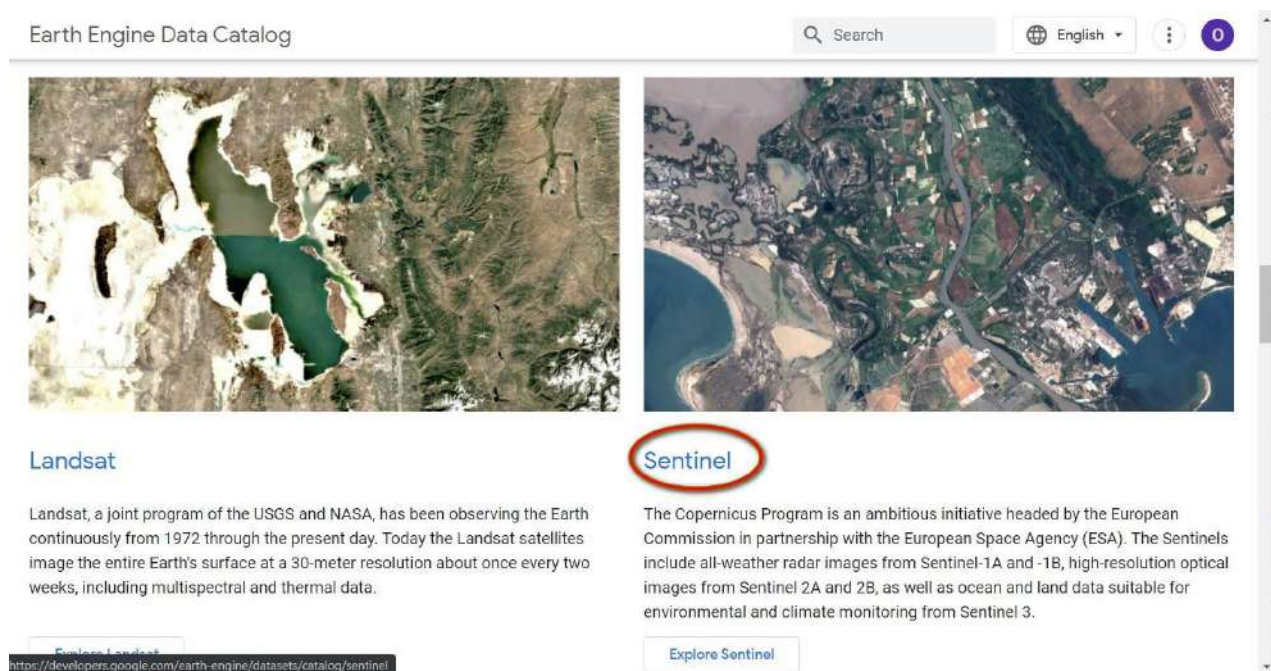


Рис. 21. Датасети Sentinel

Тут є чотири датасети:

- Sentinel-1 SAR GRD: C-band Synthetic Aperture Radar – з радарними даними;
- Sentinel-2 MSI: Multispectral Instrument – з мультиспектаральними даними;
- Sentinel-3 OLCI EFR: Ocean and Land Color Instrument – теж мультиспектральний датасет для дослідження океанів;
- Sentinel-5P TROPOMI: TROPospheric Monitoring Instrument – для дослідження повітря.

Виберіть Sentinel-2 MSI: Multispectral Instrument і натисніть на нього. Знову з'являється вибір між Top-of-Atmosphere Reflectance – «сирими» даними із супутника Sentinel-2 і Surface Reflectance із даними, що пройшли атмосферну корекцію. Виберіть Top-of-Atmosphere Reflectance. Це дані рівня Level-1C, які доступні з 2015 р. Вони ортотрансформовані і пройшли атмосферну корекцію.

Інший датасет Surface Reflectance рівня Level-2A ортотрансформований й атмосферно скоригований до відбиваючої здатності поверхні.

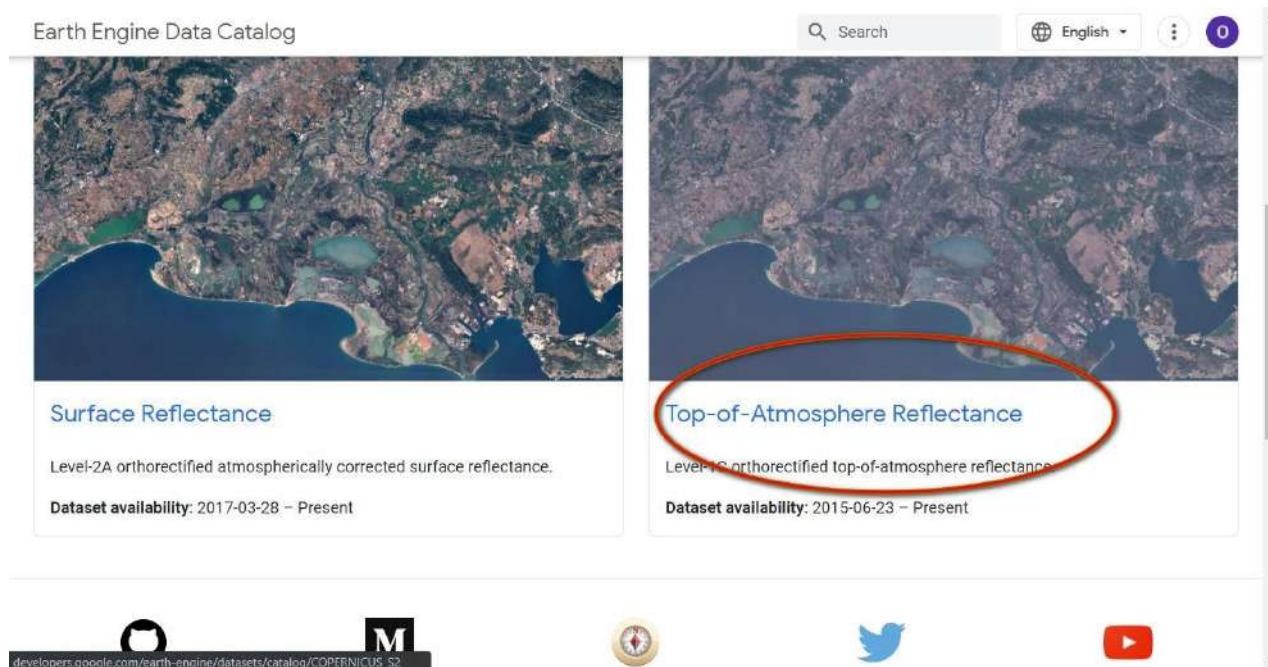


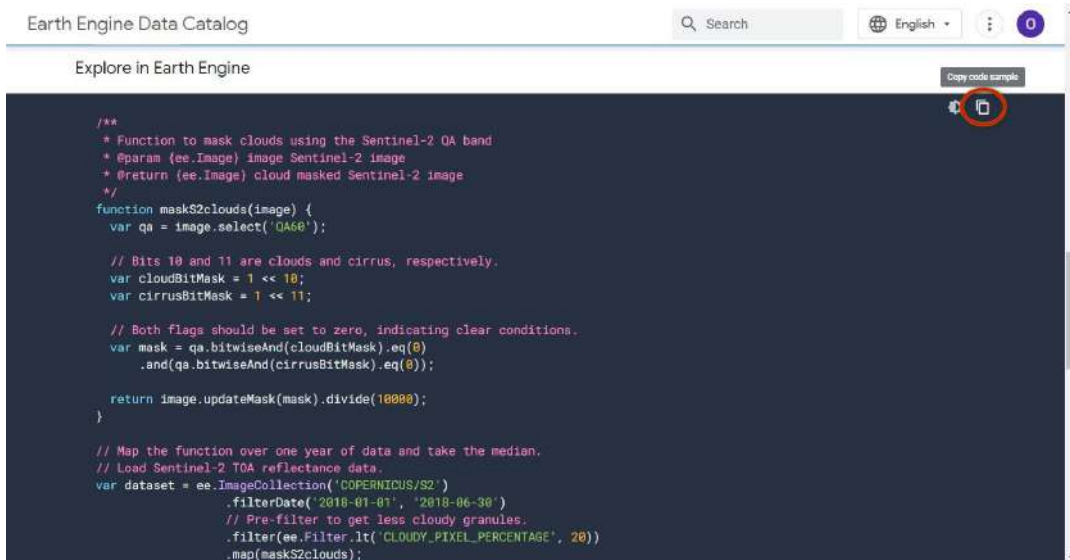
Рис. 22. Рівень обробки 2 для Sentinel

У результаті відкриється сторінка з описом знімків Sentinel-2. Розгорнувши вкладку Bands, можна побачити повний перелік каналів супутника, їх просторове та спектральне розрізнення. Таку інформацію можна знайти для усіх супутників, які є в каталозі Google Earth Engine.

1.2. Скопіювати код, який представлений у датасеті Top-of-Atmosphere Reflectance для Sentinel-2 розробниками компанії «Google».

У нижній частині сторінки є код для відображення даних Sentinel. Спочатку ми його скопіюємо, а далі почнемо редагувати цей код.

Тут зберігаються метадані для цього супутника, його характеристики, і є приклад коду, який ми будемо використовувати. Для цього прогорніть сторінку вниз і знайдіть приклад коду. Щоб його скопіювати, скористайтеся кнопкою «копіювати» в правому верхньому кутку з прикладом коду.



```
Earth Engine Data Catalog Search English

Explore in Earth Engine Copy code sample

/**
 * Function to mask clouds using the Sentinel-2 QA band
 * @param {ee.Image} image Sentinel-2 image
 * @return {ee.Image} cloud masked Sentinel-2 image
 */
function maskS2clouds(image) {
  var qa = image.select('QA60');

  // Bits 10 and 11 are clouds and cirrus, respectively.
  var cloudBitMask = 1 << 10;
  var cirrusBitMask = 1 << 11;

  // Both flags should be set to zero, indicating clear conditions.
  var mask = qa.bitwiseAnd(cloudBitMask).eq(0)
    .and(qa.bitwiseAnd(cirrusBitMask).eq(0));

  return image.updateMask(mask).divide(10000);
}

// Map the function over one year of data and take the median.
// Load Sentinel-2 TOA reflectance data.
var dataset = ee.ImageCollection('COPERNICUS/S2')
  .filterDate('2018-01-01', '2018-06-30')
  // Pre-filter to get less cloudy granules.
  .filter(ee.Filter.lt('CLOUDY_PIXEL_PERCENTAGE', 20))
  .map(maskS2clouds);
```

Рис. 23. Функція накладання маскування хмар

Після цього код скопіювався до вашого буфера обміну.

Тепер відкрийте **Code Editor (Редактор коду)** і вставте цей код у робоче середовище. Для цього поверніться на головну сторінку Google Earth Engine, наведіть курсор миші на **Platform** і виберіть **Code Editor (Редактор коду)**.

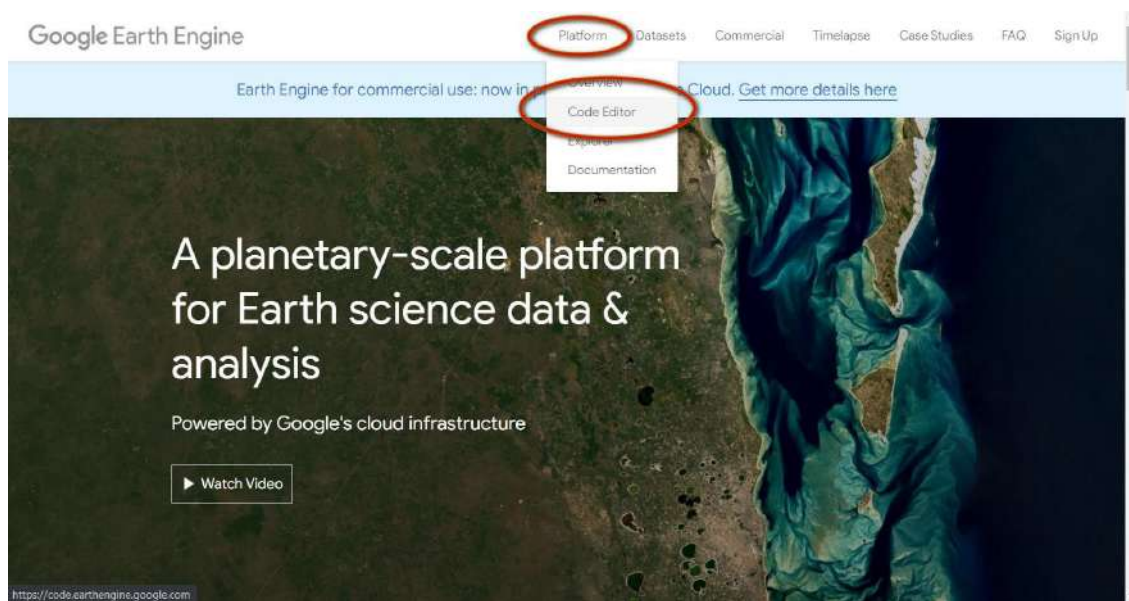


Рис. 24. Вхід до Редактора коду

Або відразу натисніть **Open in Code Editor (Відкрити в Редакторі коду)** внизу сторінки https://developers.google.com/earth-engine/datasets/catalog/COPERNICUS_S2



Рис. 25. Відкрити код у Редакторі коду

Вставте код і натисніть **Run (Запуск скрипту)**.

На мапі в результаті виконання коду завантажується кольоровий супутниковий знімок з комбінацією каналів природних кольорів. Використовуючи інструменти навігації по мапі і мишку, ви можете переміщатися мапою, збільшувати або зменшувати масштаб.

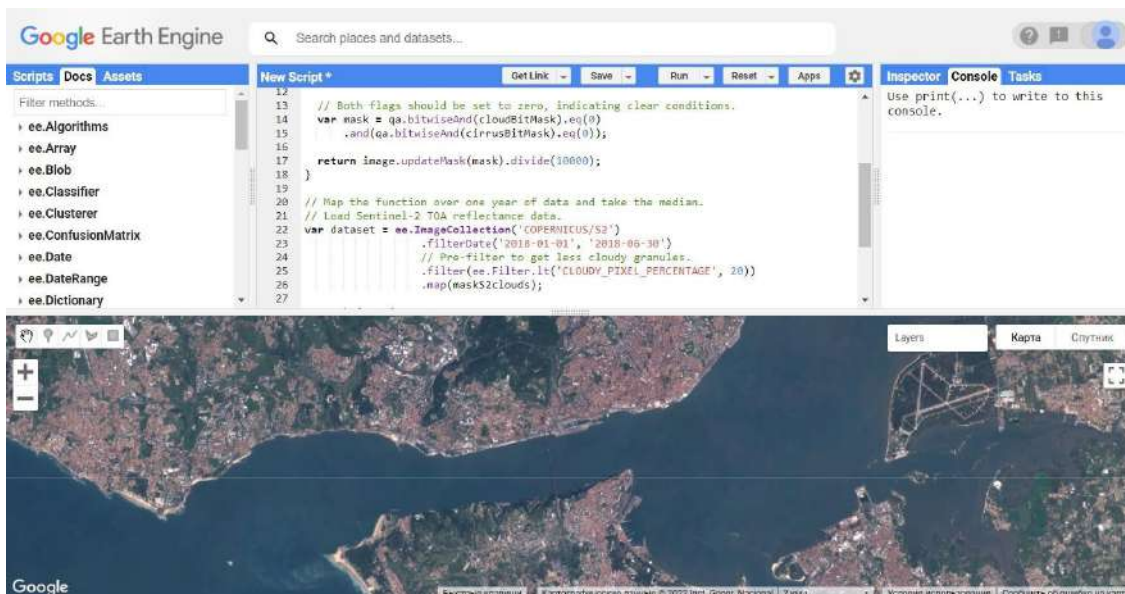


Рис. 26. Редактор коду з функцією маскуванн хмар

Цей код фільтрує зображення із супутникового знімка Sentinel-2 і накладає маску для хмар. Потім отриманий медіанний результат від першого січня по тридцять червня буде відображатися у робочому вікні сервісу.

Тепер цей код потрібно трохи модифікувати відповідно до нашого дослідження. Етап з накладанням маски залишається без змін. Ми будемо міняти лише дати і порядок відображення каналів.

Однак спочатку потрібно трохи більше ознайомитися з функцією накладання маски в GEE. Ця функція представлена розробниками компанії «Google» і є стандартним способом накладання маски для хмар в GEE. Вона бере на вхід зображення і повертає замасковане зображення. В датасеті, який ми використовуємо, є спеціальний окремий канал QA60, в якому зберігається інформація для маскуванн хмар. У цьому каналі інформація зберігається у двобітній системі і може набувати по два значення для кожного класу:

- Bit 10: Opaque clouds (*Непрозорі хмари*) – перший біт
 - 0: No opaque clouds (*Відсутні непрозорі хмари*) – значення = 0
 - 1: Opaque clouds present (*Присутні непрозорі хмари*) – значення = 1024
- Bit 11: Cirrus clouds (*Перисті хмари*) – другий біт
 - 0: No cirrus clouds (*Відсутні перисті хмари*) – значення = 0
 - 1: Cirrus clouds present (*Присутні перисті хмари*) – значення = 2048

Ця функція використовує оператор << (**Left shift**), що побітово зсуває число вліво, наприклад у синтаксисі JavaScript (https://developer.mozilla.org/en-US/docs/Web/JavaScript/Reference/Operators/Left_shift). Тобто для 1 значення зсунуте на 10 знаків дорівнюватиме 1024, а якщо зсунути значення на 11 знаків, то воно дорівнюватиме 2048. Далі, якщо немає значень 1024 і 2048, тобто вони дорівнюють 0, це вказується явно, функція повертає зображення, яке брала на вхід, і застосовує до нього метод `updateMask`, що оновлює маску зображення у всіх позиціях, де наявна маска не дорівнює нулю.

✔ **Важливо!** У лівій колонці зеленим кольором подані англomовні коментарі від розробників скрипту, які виділяються через подвійний slash // або через /* "будь-який текст" */. Вони не зчитуються програмою, а потрібні для пояснення того, як саме працює код.

✔ **Важливо!** Не потрібно копіювати і змінювати наступний код, він уже є у вас у **Code Editor** (Редактор коду).

Розшифруємо нижче, що саме виконує цей код.

<pre>/* * Function to mask clouds using the Sentinel-2 QA band * @param {ee.Image} image Sentinel-2 image * @return {ee.Image} cloud masked Sentinel-2 image */ function maskS2clouds(image) { var qa = image.select('QA60'); // Bits 10 and 11 are clouds and cirrus, respectively. var cloudBitMask = 1 << 10; var cirrusBitMask = 1 << 11; // Both flags should be set to zero, indicating clear conditions. var mask = qa.bitwiseAnd(cloudBitMask).eq(0) .and(qa.bitwiseAnd(cirrusBitMask).eq(0)); return image.updateMask(mask).divide(10000); }</pre>	<p>Багаторядкові коментарі починаються з комбінацією символів "slash+*" і закінчуються зворотною комбінацією символів */.</p> <p>Їх пишуть, щоби пояснити, що саме відбувається в коді. GEE ці рядки не зчитує.</p> <p>У цьому коментарі сказано, що функція бере на вхід зображення із супутника Sentinel-2 і повертає зображення із замаскованими хмарами.</p> <p>Це стандартна функція накладання маски на хмари, представлена розробниками GEE.</p> <p>Функція maskS2clouds задає змінну qa, де вибирає канал 'QA60', який може набувати значень, представлених вище.</p> <p>cloudBitMask зсуває значення побітово на 10 знаків</p> <p>cirrusBitMask зсуває значення побітово на 11 знаків.</p> <p>Змінна mask на елементній основі обчислює бітове значення та вхідне значення, які дорівнюють 0 для cloudBitMask та cirrusBitMask.</p> <p>Вся функція встановлює маску для зображення і ділить її на 10 тисяч</p>
---	---

Цей код далі звертається до колекції зображень, що містяться за шляхом **COPERNICUS/S2**, і фільтрує цю колекцію за датами з 1 січня 2018 р. по 30 червня 2018 р.

✔ **Важливо!** Вам потрібно змінити ці дати на "із 4 жовтня 2015 по 5 жовтня 2015".

Потім цей код фільтрує колекцію і вибирає всі зображення, показник хмар яких менший, ніж 20 відсотків, і застосовує функцію маскування хмар.

2. Змінити в скопійованому коді дати відбору знімків на дату пожежі в Дарницькому лісі, а саме: із 4 жовтня по 5 жовтня 2015 р.

У наступному вікні необхідно зробити такі заміни: в змінній `.filterDate('2018-01-01', '2018-06-30')` змінити дати на `.filterDate('2015-10-04', '2015-10-05')`, також додати рядок `.select('B12', 'B8', 'B2')`, щоби під час майбутнього аналізу експортувалися лише три канали.

Все інше залишити без змін.

<pre>// Map the function over one year of data and take the median. // Load Sentinel-2 TOA reflectance data. var dataset = ee.ImageCollection('COPERNICUS/S2') .filterDate('2015-10-04', '2015-10-05') // Pre-filter to get less cloudy granules. .filter(ee.Filter.lt('CLOUDY_PIXEL_P RCENTAGE', 20)) .map(maskS2clouds) .select('B12', 'B8', 'B2');</pre>	<p>dataset ця змінна, що звертається до дасету, який знаходиться за адресою COPERNICUS/S2 та фільтрує за двома датами зображення.</p> <p>Тут відбувається фільтрація за хмарністю, відбираються знімки зі значенням менше 20%</p> <p>Застосування функції маскування хмар</p>
---	--

3. Змінити відображення супутникового знімка, що був за замовчуванням у кодї від компанії «Google», з природних кольорів: Red, Green Blue на штучні кольори: SWIR, NIR, Blue.

За замовчуванням код, який ми скопіювали, показує зображення в природних кольорах. Для супутника Sentinel-2 це комбінація каналів 'B4', 'B3', 'B2'. Для кращого ідентифікування пожежі необхідно відобразити зображення за допомогою SWIR, NIR, Blue, а саме: 'B12', 'B8', 'B2'.

Наступним кроком буде змінити параметри відображення так, щоб краще було видно ліс. Необхідно змінити bands: ['B4', 'B3', 'B2'] на bands: ['B12', 'B8', 'B2']:

<pre>var rgbVis = { min: 0.0, max: 0.3, bands: ['B12', 'B8', 'B2'], };</pre>	<p>У візуальних параметрах створюємо композит SWIR – 'B12', NIR – 'B8', Blue – 'B2'</p>
--	---

4. Скористатися *Inspector* (Інспектор), щоб знайти координати Києва.

Тепер знайдемо Київ на мапі і визначимо його координати. Для цього можна скористатися рядком пошуку *Search places and datasets* (Пошук місць та наборів даних). Відкрийте вкладку *Inspector* (Інспектор) і натисніть мишкою на місце розташування Києва на мапі. У вікні *Inspector* (Інспектор) відобразяться координати – скопіюйте їх.

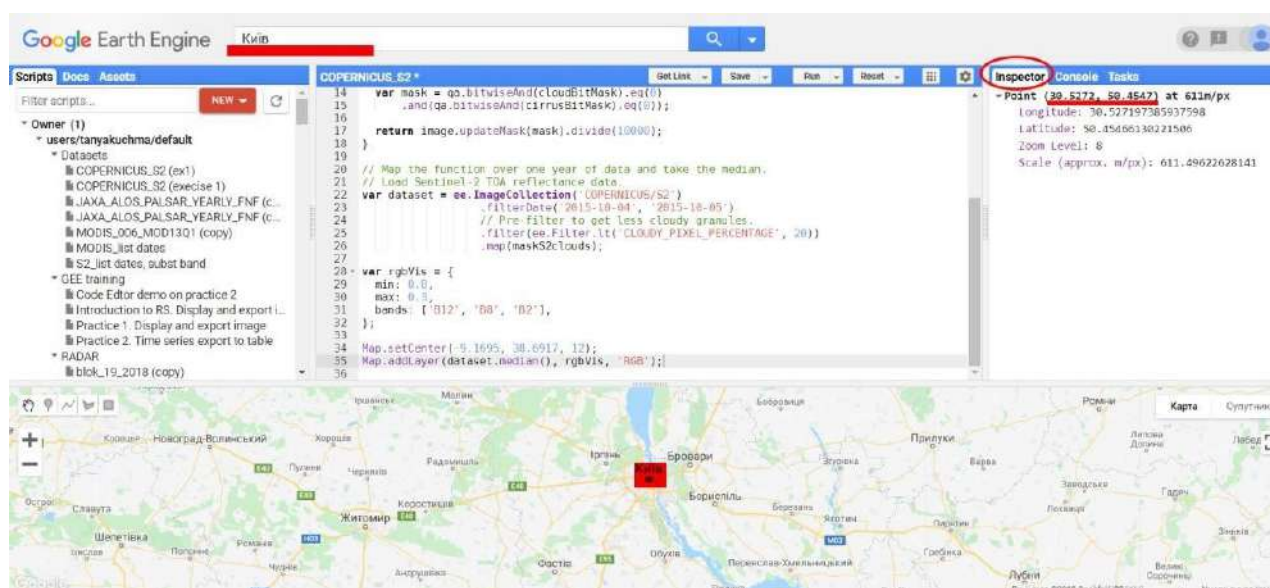


Рис. 27. Виведення координат

1.1. Змінити наявні в кодї координати на координати Києва, знайдені за допомогою *Inspector* (Інспектор).

Скопіюйте координати Києва і вставте їх у `Map.setCenter(-9.1695, 38.6917, 12)`

Також виконайте наступну команду, що звернеться до нашого датасету і порахує для нього медіанне значення, застосує параметри візуалізації, задасть назву та додасть до мапи.

<pre>Map.setCenter(30.5272, 50.4547, 11); Map.addLayer(dataset.median(), rgbVis, 'RGB');</pre>	<p>Наближається за координатами з 11-кратним зумом</p> <p>Додає медіанне зображення на мапу з параметрами відображення rgbVis та назвою 'RGB'</p>
--	---

Натисніть *Run* (Запуск скрипту).

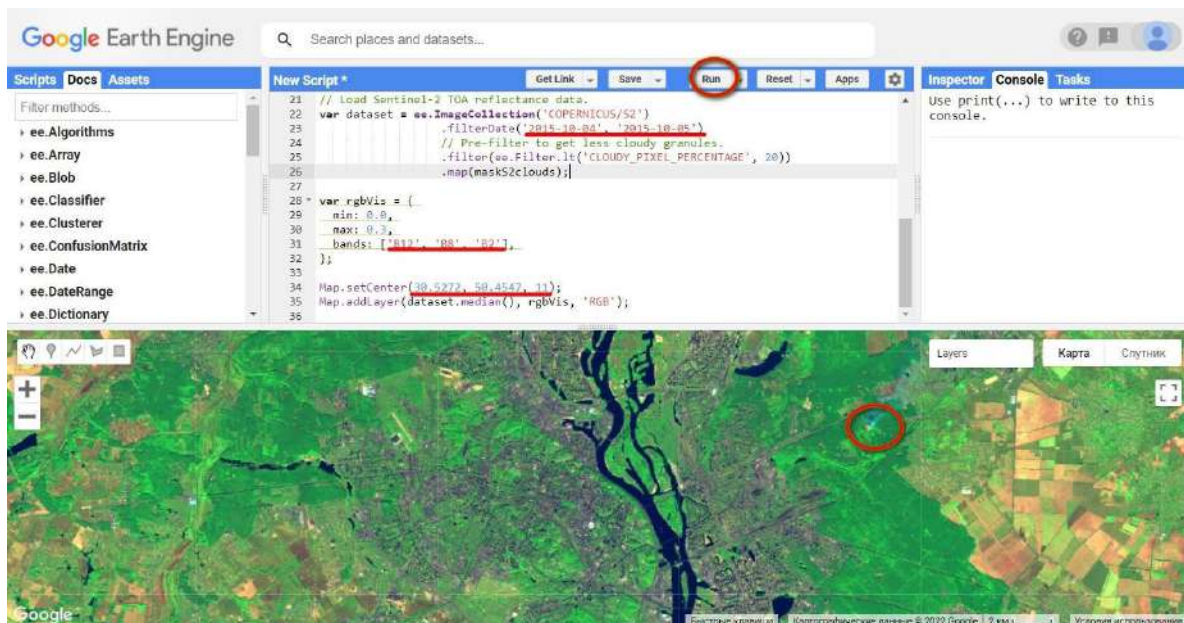



Рис. 28. Зміна відображення растра

Тепер натисніть **Run** (*Запуск скрипту*) і подивіться на результат. Кольоровий знімок Києва за 04.10.2015 р. має завантажитися в картографічному вікні.

Ви можете бачити, що знімок, який відобразився у картографічному вікні, охоплює велику територію, значно більшу, ніж м. Київ, тому далі ми окреслимо територію Києва й обріжемо зображення.

5. Створити рамку (геометричний об'єкт) за допомогою інструментів геометрії для обрізання зображення.

Для того щоб окреслити область для обрізки зображення, використаємо інструмент, , який розміщений у лівому куті картографічного вікна, і нарисуємо чотирикутник у такий спосіб, аби окреслити пожежу та невелику територію поряд з нею.

Це зображення можна завантажити собі на комп'ютер, для цього в GEE скористайтеся експортом до свого особистого Google-диска. Нам необхідно обвести ділянку, де сталася пожежа, за допомогою інструментів геометрії.


Виберіть інструмент геометрії  (*Накреслити прямокутник*) й обведіть невелику ділянку навколо пожежі в Дарницькому лісі.



Рис. 29. Пожежа в Дарницькому лісі

Поглянувши знову на код, ви можете переконатися, що в його верхній частині автоматично з'явився новий рядок, який визначає нову змінну (полігон) із назвою «*geometry*». Це буде зона, за допомогою якої ми виріжемо супутникове зображення.

Що більшу територію ми окреслимо, то довше відбуватиметься операція експорту фінального супутникового зображення.

6. Експортувати зображення супутникового знімка Sentinel-2 за 04.10.2015 р. на свій особистий Google-диск.

Відкрийте вкладку **Docs** (Документація до API Docs) і в рядку пошуку інструментів напишіть «export».



Рис. 30. Документація до експорту

У результатах пошуку натисніть на **Export.image.toDrive** – саме ця команда дає змогу зберегти зображення на Google-диск (сховище даних, поєднане з вашим Google-акаунтом), звідки ви можете завантажити опрацьований знімок на свій комп'ютер. Відкриється інформація про команду. Скопіюйте текст на синьому фоні в заголовку вікна (це точна назва команди (функції)) і додаткові параметри, які можуть бути вказані у коді.

Не всі параметри, зазначені в дужках, потрібно прописувати в коді. Залишимо з них лише найважливіші.

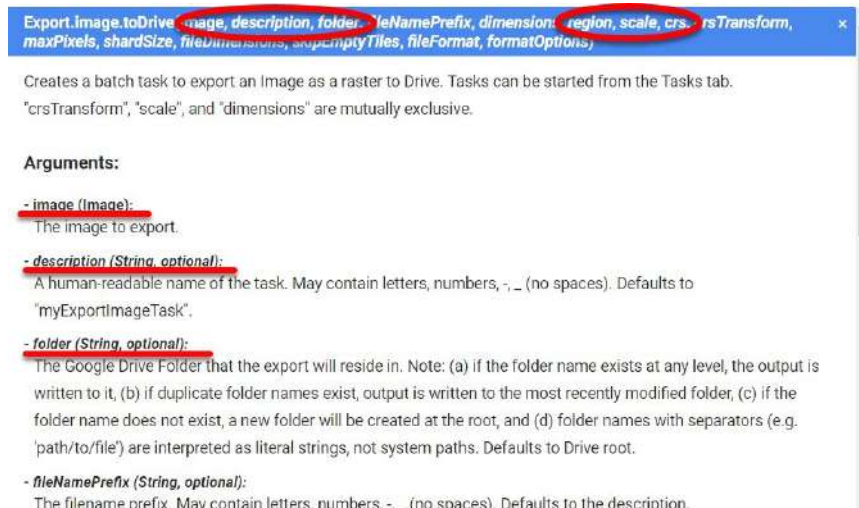


Рис. 31. Розгорнутий опис документації

✓ **Важливо!** *Обов'язкові параметри пишуться рівним шрифтом, тобто Regular, необов'язкові – Italic.*

Найперше, це **image** (знімок) – тут треба вказати змінну, тобто знімок, який потрібно експортувати. У нашому випадку це **dataset** (відфільтровані за датою, хмарністю і територією знімки Sentinel-2). Проте зазначити **dataset** буде неправильно, бо GEE намагатиметься завантажити всю колекцію зображень за вказану вами дату, а завантажувати можна лише один знімок, тобто об'єкт **Image** (Зображення), а не **ImageCollection** (Колекція зображень). Тому варто вказати не просто **dataset** (набір даних), а **dataset.median** (усереднене зображення з набору знімків).

Далі ми також хочемо, щоб кольорове зображення було створене з каналів B12, B8, B2, тому запишемо не просто мозаїку, а `dataset.median.select(['B12', 'B8', 'B2'])`. І наостанок обріжемо зображення по геометричній фігурі, яку окреслили на мапі. Отже, параметр – `image: dataset.median.select(['B12', 'B8', 'B2']).clip(geometry)`

Наступний важливий параметр – це **description** – назва вихідного файлу, наприклад: 'Kyiv_0410_2015_Sentinel2'

Ще один параметр – це назва папки, в яку буде зберігатися зображення. Потрібно вказати: **folder**: 'MAN_lab2'

Далі важливо вказати **scale** (*масштаб*) – у цьому випадку це просторове розрізнення вихідного зображення в метрах, тобто для знімків Sentinel-2 – 10 метрів.

Також зазначимо **region** (*регіон*): **geometry** (тобто окреслена полігональна область).

І останній параметр, який ми вкажемо, це **crs** (система координат). Зручною для Києва є проєктована система координат WGS 84 / UTM зона 36N або її кодова назва: EPSG:32636. Скопіюйте код:

<pre>Export.image.toDrive({ image: dataset.mean().clip(geometry), description: 'Kyiv_0410_2015_Sentinel2', folder: 'MAN_lab2', scale: 10, region: geometry, crs : 'EPSG:32636' });</pre>	<p>Export.image.toDrive функція, що експортує зображення на Google-диск, у цьому випадку має такі параметри:</p> <p>image – зображення, що буде маскуватися, зведе-не до медіанного значення та обрізане по межах geometry</p> <p>description – параметр, який вказує, як буде називатися файл</p> <p>folder – назва папки</p> <p>scale – масштаб зображення, для Sentinel-2 це 10 метрів</p> <p>region – місце, в яке буде експортуватися зображення</p> <p>crs – система координат за 'EPSG'</p>
---	--

Тепер код готовий. Ви можете зберегти його, натиснувши **Save** (*Зберегти*) угорі редактора коду, і натиснути **Run** (*Запуск скрипту*), щоб виконати код.

Далі необхідно перейти до вкладки **Tasks** (*Менеджер завдань*), де з'явиться нове завдання із завантаженням. Тут натисніть **RUN**.

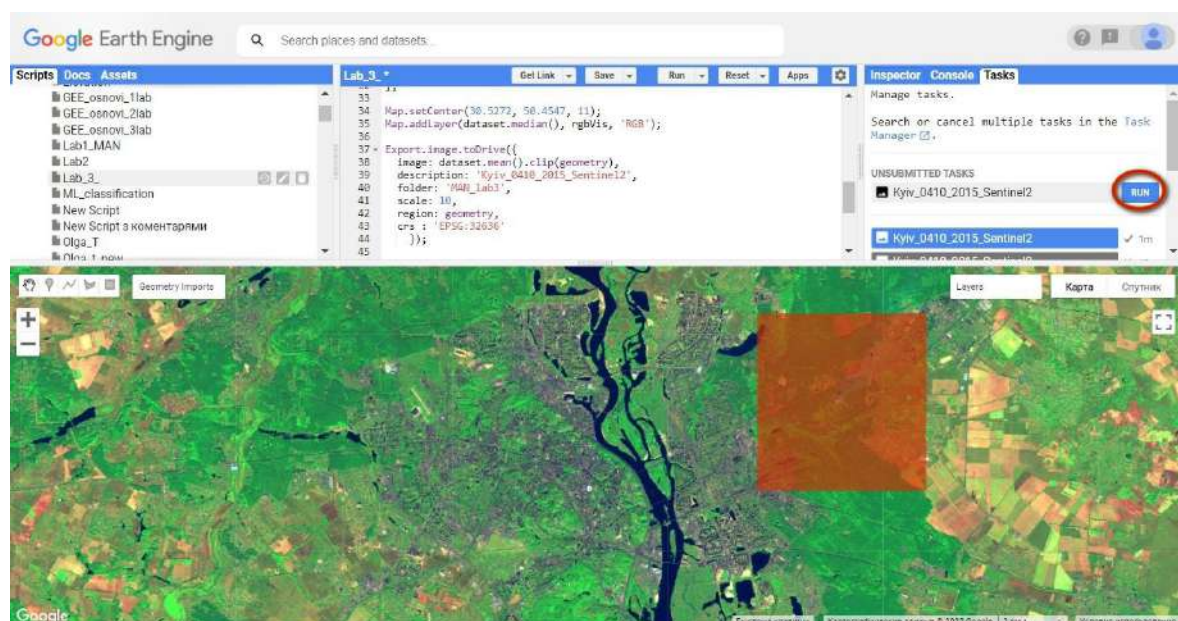


Рис. 32. Завантаження даних у Менеджер завдань

Після цього з'явиться нове вікно, в якому будуть налаштування вашого завантаження. Тут можна залишити все за замовчуванням і натиснути **RUN**.

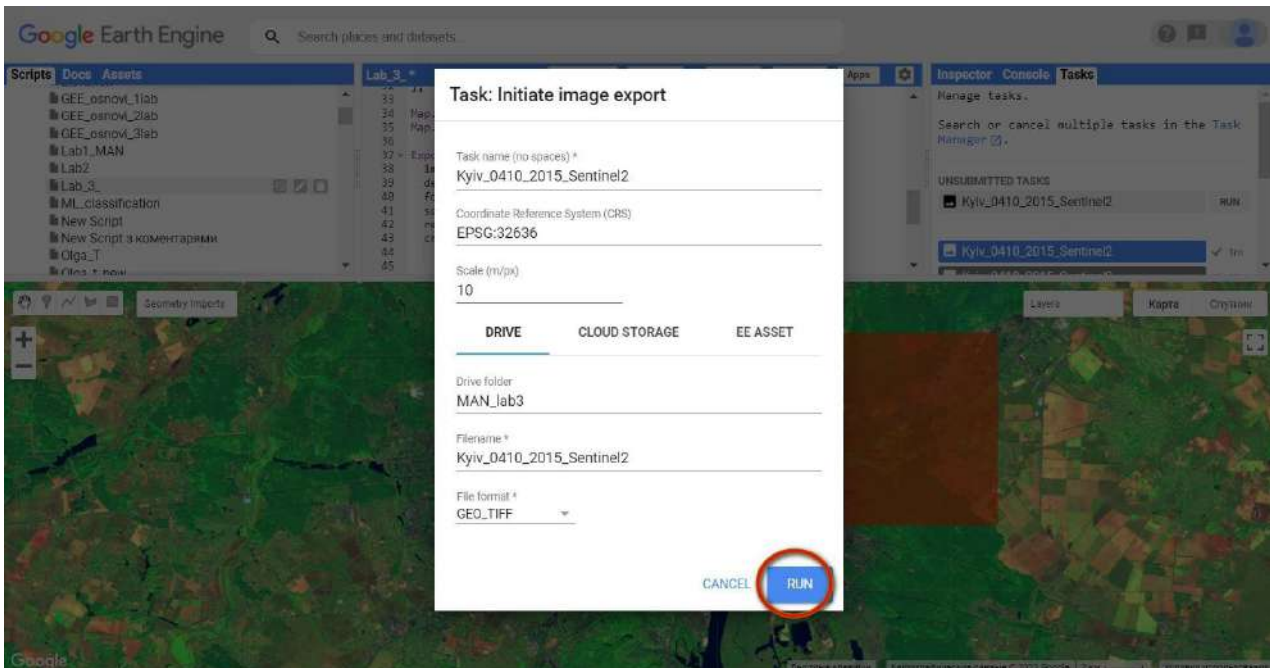


Рис. 33. Завантаження даних

Ця дія розпочне нове завантаження на ваш Google-диск. У папці 'MAN_lab2' має з'явитися файл 'Kyiv_0410_2015_Sentinel2'.

✔ **Важливо!** *Завантаження може зайняти кілька хвилин.*

6.1. Завантажити космічне зображення на свій особистий Google-диск.

Коли знімок завантажиться, рядок з назвою знімка в меню **Tasks** (*Менеджер завдань*) підсвітиться синім кольором. Далі потрібно навести на нього курсор мишки, щоб з'явився знак питання, і натиснути на нього – у вікні, що відкриється, натиснути на **Open in Drive** (*Відкрити на Google-диску*).

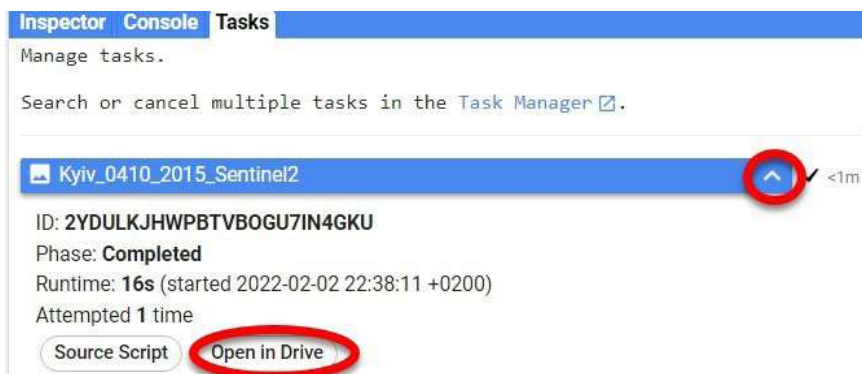


Рис. 34. Відкриття завантажених даних на Google-диск

Після цього відкриється Google-диск, де потрібно натиснути правою клавшею миші на збережений знімок, вибрати **Download** (*Завантажити*) і зберегти зображення у свою робочу папку на комп'ютері.

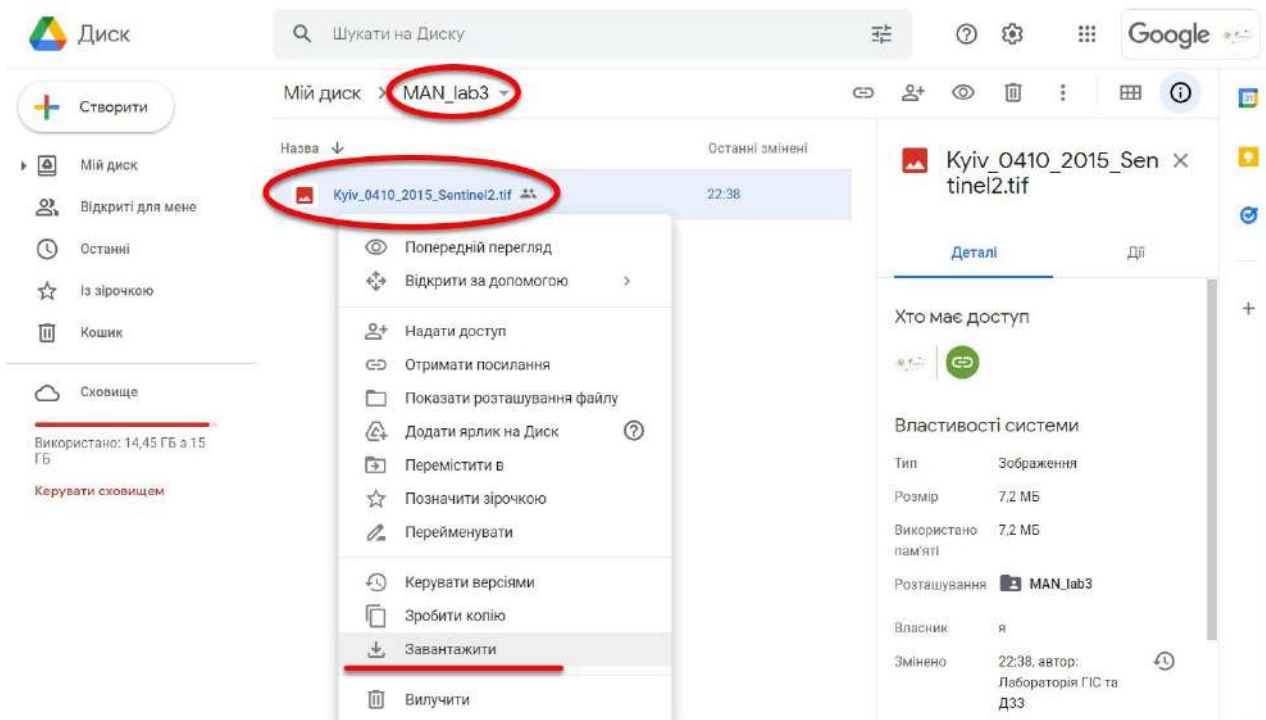


Рис. 35. Вивантаження даних із диска

У результаті ви отримаєте кольорове супутникове зображення пожежі в Дарницькому лісі з комбінацією каналів B12, B8, B2 з просторовою прив'язкою, яке можна далі використовувати в різних програмах, зокрема у QGIS.

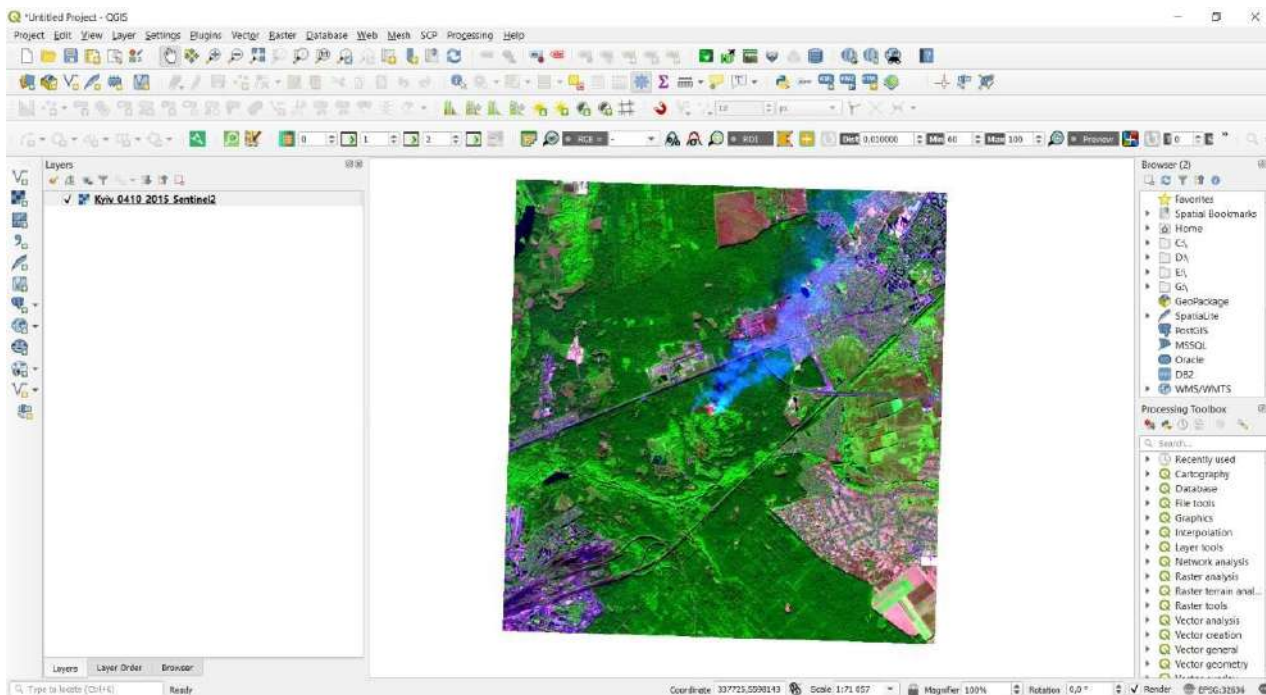


Рис. 36. Вигляд знімка в QGIS

? Завдання для перевірки

Завантажити супутникове зображення Дарницького лісу з пожежею на свій Google-диск

7. Створити коментар зі своїм прізвищем.

Як ви вже помітили, в GEE можна користуватися чужими скриптами, тобто тими, які розробили інші користувачі, а також розробники компанії «Google». Під час поширення скрипту хорошим тоном є написання коментарів до нього. Коментарі – це рядки коду, які GEE виконує під час запуску скрипту, вони створюються для більш зручного користування скриптом і виділяються зеленим кольором. Коментарі бувають однорядкові і багаторядкові.

// Для створення однорядкових коментарів використовується подвійний slash

/* Для створення багаторядкових коментарів почніть перший рядок комбінацією символів "slash+*" і закінчте останній рядок зворотною комбінацією символів */

? Завдання для перевірки

Створіть однорядковий коментар зі своїм прізвищем

? Додаткове завдання для перевірки

Якщо ви повністю зрозуміли і пройшли попередні кроки в практичній роботі, спробуйте виконати додаткове завдання.

2020 р. у Чорнобильському лісі відбулася пожежа, яка тривала майже місяць. Для пошуку знімка на момент цієї події можете використовувати той самий скрипт. Вам потрібно змінити координати або видалити (закоментувати) рядок з координатами, щоби після натискання **Run** (Запуск скрипту) вас не переміщувало знову до Києва. Також потрібно змінити дати із 2020.04.12 по 2020.04.13, інші параметри залишити за замовчуванням.



Рис. 37. Остаточний знімок

? Завдання для перевірки

Змініть код так, щоб можна було побачити пожежу в Чорнобильському лісі поблизу водойми-охолоджувача ЧАЕС

Зберегти відредагований скрипт можна за допомогою кнопки **Save**, після цього можна отримати лінк за допомогою кнопки **Get link** (Отримання лінку) і скопіювати його.

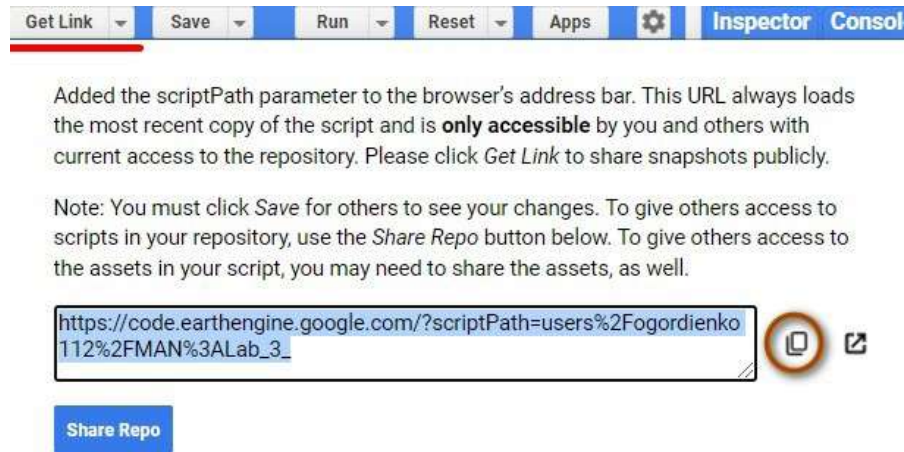


Рис. 38. Копіювання лінку з метою поділитися кодом

За наведеним нижче QR-кодом ви можете переглянути відео виконання практичної роботи.



Рис. 39. Відео виконання практичної роботи

Готовий код можна відкрити за лінком:

https://code.earthengine.google.com/?scriptPath=users%2Fogordienko%2Fcourse%2FMAN%3ALab_2

Завантаження векторних даних та топографічних мап у робоче середовище, їх аналіз і оцифрування на прикладі дослідження зміни лісу в Чорнобильській зоні

Ситуація

У минулому лісистість Полісся була значно більшою: всю територію Чорнобильської зони відчуження вкривали ліси. Проте внаслідок діяльності людини і лісокористування на початку ХХ ст. вона значно зменшилася і в центральній частині зони почала становити всього 11–12%. Планомірні роботи з лісовідновлення розпочалися у 20-х, а масові – в 50–60 рр. ХХ ст. На сьогодні території, вкриті лісом, займають половину площі Чорнобильської зони відчуження. Після аналізу історичних досліджень лісоведення на цій території в 1913 і 1986 рр., які виконувалися М. Кучмою та С. Бідною,¹ виникла ідея картографічно проілюструвати і перевірити, якою ж саме була площа лісів на початку ХХ ст. порівняно з її сучасним станом.

Завдання

Навчитися додавати до GEE зовнішні векторні та растрові дані, оцифрувати контур лісу з топографічної мапи, розраховувати площу лісового масиву станом на 1937 р. й отриману із сучасних векторних даних OSM. Проаналізувати зміну загальної лісистості Чорнобильської зони за майже 100 років.

Алгоритм виконання завдання:

1. Відкрити **Code Editor** (Редактор коду) під власним акаунтом Google Earth Engine.
2. Завантажити з Google-диска матеріали, необхідні для виконання практичної роботи (топокарти і вектор OSM).
3. Завантажити зовнішні дані в GEE.
 - 3.1. Додати топографічні мапи 1937 р.
 - 3.2. Додати вектор сучасного лісового покриву з OSM.
 - 3.3. Імпортувати завантажені дані в робоче середовище GEE.
 - 3.3.1. Імпортувати векторні дані.
 - 3.3.2. Імпортувати растрові дані.
4. Наблизитися до території дослідження й додати імпортовані дані на оглядову мапу.
5. Змінити параметри візуального відображення топокарти.
6. Створити векторний шар *OldForest* і оцифрувати контури лісу з топографічних мап.
7. Візуально порівняти сучасну лісистість території з 1937 р. й розрахувати загальну площу лісів.
 - 7.1. Отримати площу виділеного лісу з архівних топографічних мап 1937 р.
 - 7.2. Розрахувати площу сучасного лісу на основі OSM-даних.

Покрокова інструкція

1. Для того щоби почати працювати з **Code Editor** (Редактор коду), необхідно знайти його на головній сторінці сайту Google Earth Engine <https://earthengine.google.com/> у закладці **Platform** (Платформа) → **Code Editor** (Редактор коду) й увійти під своїм акаунтом.



Рис. 40. Вхід до Редактора коду

¹ URL: <http://chornobyl.in.ua/uk/priroda-chernobyl.html>

2. Завантажити з Google-диска матеріали, необхідні для виконання практичної роботи (топокарти і вектор OSM).

Для виконання цієї практичної роботи ми попередньо знайшли і завантажили з онлайн-ресурсу <http://igrek.amzp.pl/maplist.php?cat=USSR050> дві історичні топографічні карти на територію Чорнобильських лісів 1937 р. масштабом 1:50 000, а саме номенклатурні листи M-36-13-C і M-36-25-D. Далі ми їх попередньо прив'язали – «орієнтували» за чотирма кутовими точками у ГІС-програмі (QGIS або ArcGIS). І вже прив'язані карти виклали на Google-диск за таким покликанням: https://drive.google.com/drive/u/1/folders/1hvrZEtXlZrX9NNT_WpQq43m7wFP0ifpj.

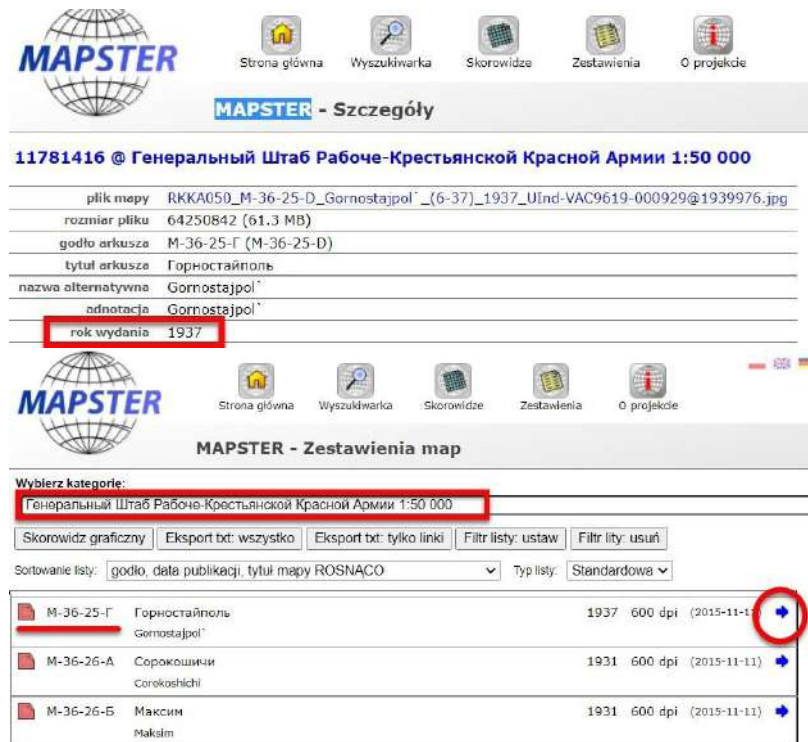


Рис. 41. Завантаження даних

Також у цій роботі ми будемо використовувати векторний шар сучасного лісового покриття з ресурсу OpenStreetMap (OSM) <https://www.openstreetmap.org/>, який був попередньо вивантажений нами й обрізаний у межах рамки території дослідження (тобто дві топографічні карти) в середовищі програми QGIS із застосуванням плагіна QuickOSM. Цей плагін дає змогу завантажувати дані з OSM. Своєю чергою, ми використали два параметри для завантаження Чорнобильського лісу: `natural = wood` і `landuse = forest`.

✔ **Важливо!** Пам'ятайте, що векторні дані мають бути прив'язані в координатній системі WGS 84 (EPSG:4326). Інакше можуть бути проблеми з відображенням на базовій мапі GEE.

3. Завантажити зовнішні дані в GEE.

Спочатку збережіть усі завантажені файли в одну папку на робочому столі комп'ютера, назвіть її `lab3_GEE`. Файли в папці повинні мати такий вигляд:

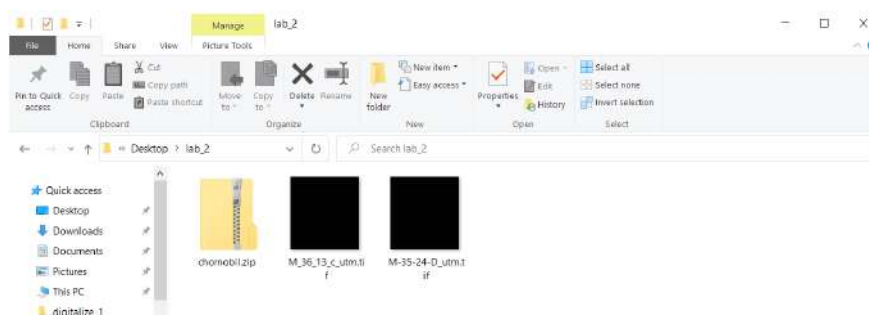


Рис. 42. Завантажені дані

Для того щоб завантажувати свої дані до GEE, необхідно відкрити редактор коду і в лівій частині вікна перейти до вкладки **Assets** (*Менеджер об'єктів*), там натиснути на кнопку **NEW** (*Новий*).

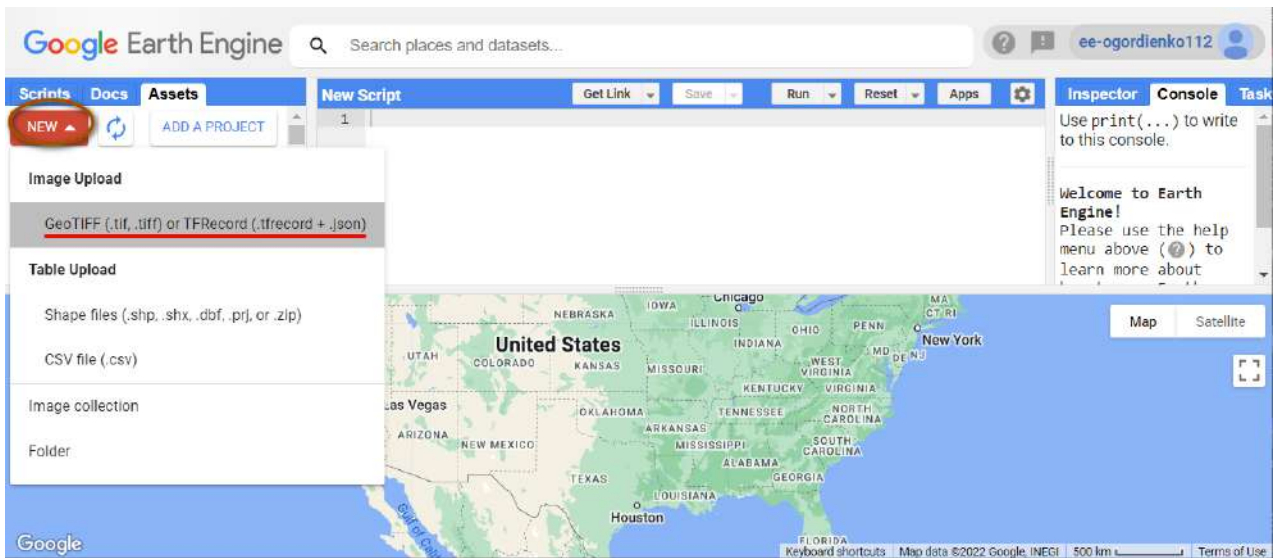


Рис. 42. Додавання даних

3.1. Додати топографічні мапи 1937 р.

У відкритому меню **NEW** (*Новий*) необхідно обрати тип завантажених даних, а саме **Image Upload** (*Завантаження зображення*) → **GeoTIFF (.tif, tiff) or TFRecord (.tfrecord + .json)**. Цей інструмент дає змогу завантажувати до GEE свої зображення, після натискання на нього відкривається нове вікно, в якому необхідно вказати шлях до файлу растрової топокарти, яку ми будемо завантажувати. Не змінюючи налаштувань, завантажте через кнопку **SELECT** (*Вибір*) першу мапу *M_36_13_c_utm.tif* з папки на вашому робочому столі і натисніть кнопку **UPLOAD** (*Завантажити*).

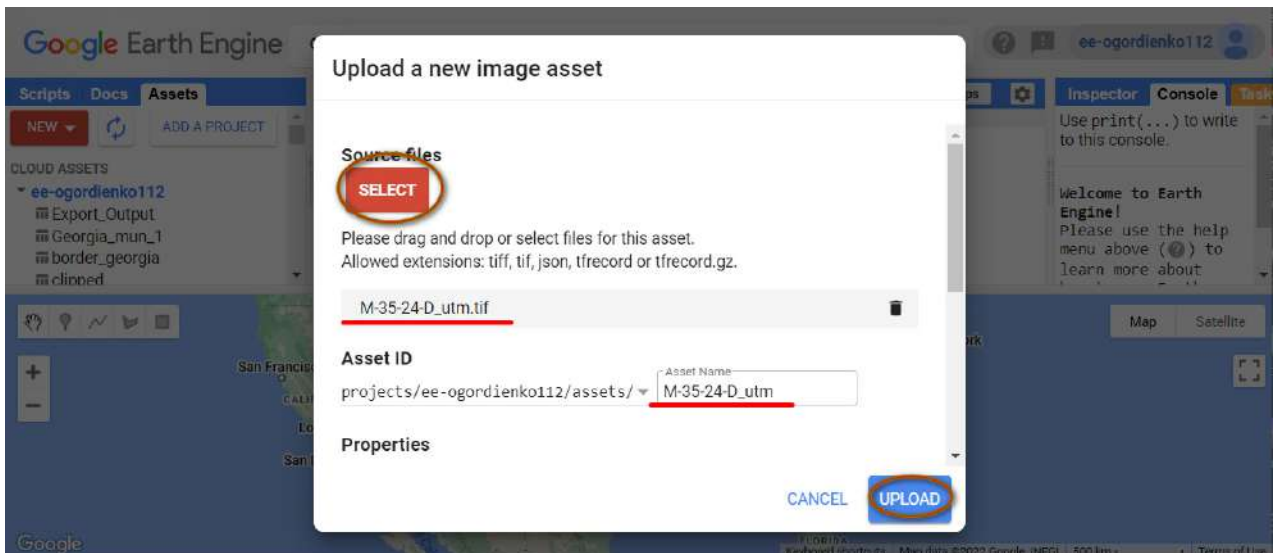


Рис. 44. Вибір зображення

Після цього в меню **Tasks** (*Менеджер завдань*) з'явиться процес, що показує завантаження зображення на сервер GEE.

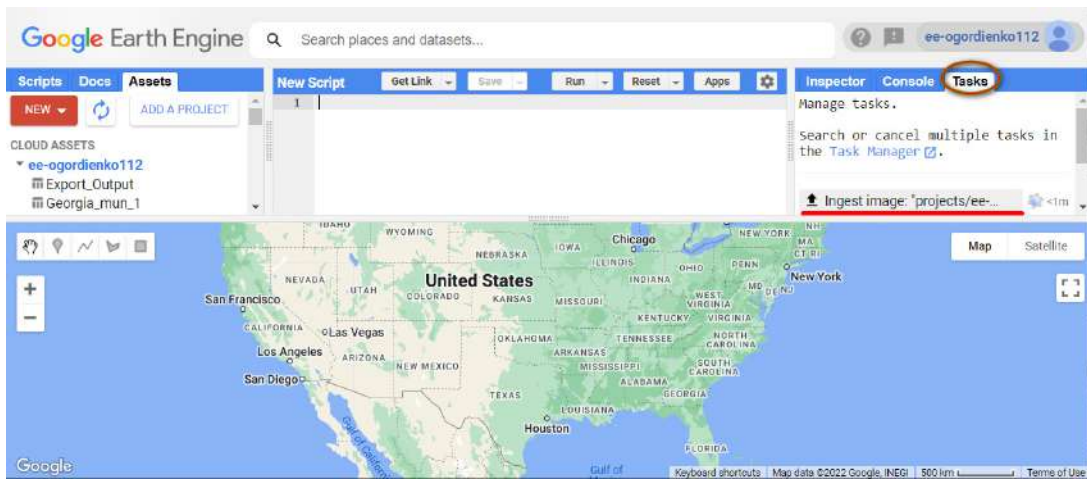


Рис. 45. Завантаження зображення

Завантаження зображень може зайняти деякий час. Можна не чекати, поки завершиться процес завантаження першої мапи, а почати завантажувати другий файл із топографічною мапою. Для цього повторіть аналогічні дії, але виберіть інший файл – *M-35-24-D_utm.tif*.

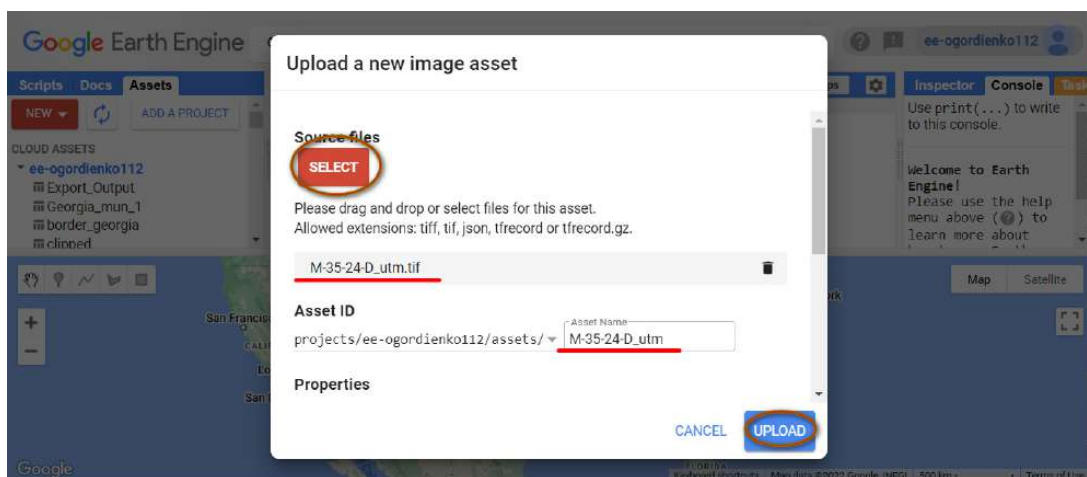


Рис. 46. Вибір другого зображення

Не чекаючи, поки завантажуються друге зображення топокарти, ми почнемо завантажувати векторні дані.

3.2. Додати вектор сучасного лісового покриву з OSM.

Знову зайдіть у меню **NEW** (Новий), але тепер виберіть тип завантажених даних **Table Upload** (Завантаження таблиці) → **Shape files (.shp, .shx, .dbf, .prj or .zip)**.



Рис. 47. Завантаження векторних даних

Після натискання на цю кнопку відкриється нове вікно, в якому через опцію **SELECT** (*Вибір*) вкажіть шлях до файлу *chornobil.zip* з папки на вашому робочому столі і натисніть **OK**. При цьому аналогічно мапам розпочнеться завантаження на сервери GEE вектора з OSM.

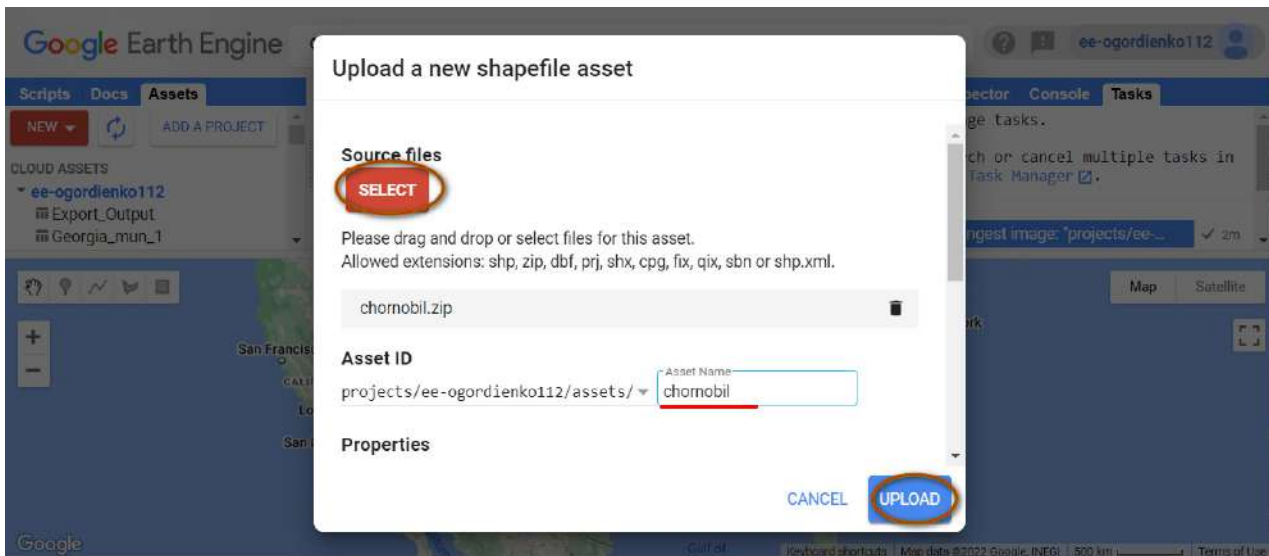


Рис. 48. Вибір векторних даних

Після цих дій потрібен час, щоб файли могли повністю завантажитися до GEE. Стежте за процесом у вкладці **Tasks** (*Менеджер завдань*): підтвердженням того, що все пройшло успішно, є синє виділення завантажених файлів.

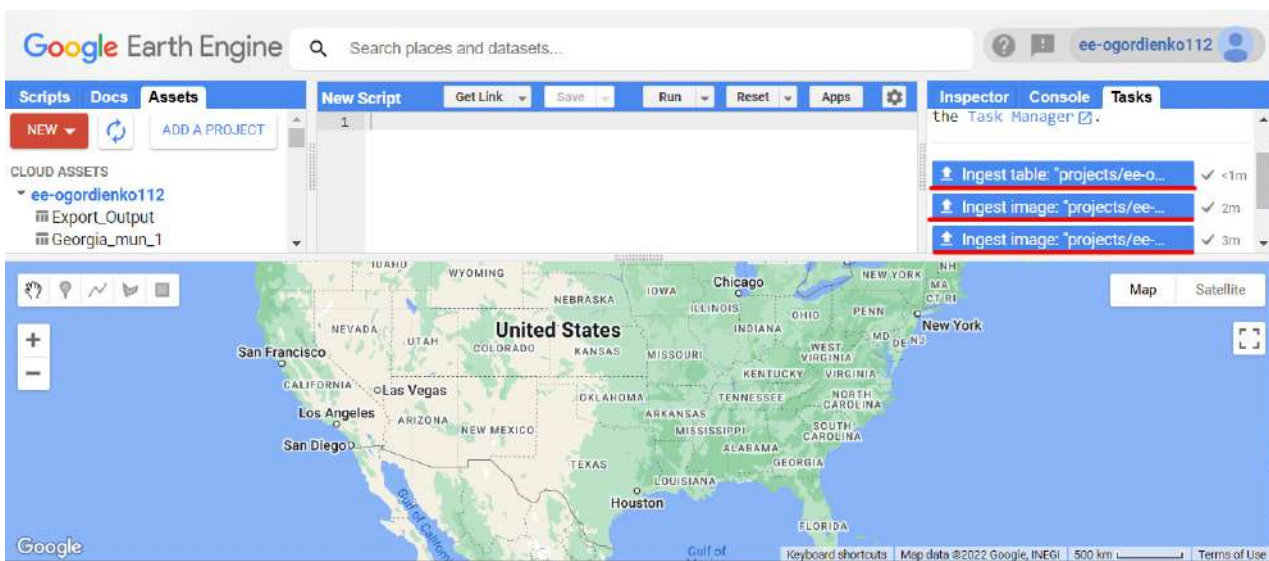


Рис. 49. Видяг екрана із завантаженими даними

Як можна бачити зі скриншоту вище, у нас завантаження зайняло по чотири хвилини для зображень і менше однієї хвилини для векторного файлу, але цей час може змінюватися залежно від швидкості вашого інтернет-з'єднання.

3.3. Імпортувати завантажені дані в робоче середовище GEE.

Далі ми будемо підвантажувати ці файли в наше робоче середовище. Файли, які ви завантажили, мають з'явитися у вкладці **Assets** (*Менеджер об'єктів*). Якщо вони там не з'явилися, натисніть кнопку у вигляді стрілок по колу **Refresh repository cache** (*Оновити кеш репозитарію*).

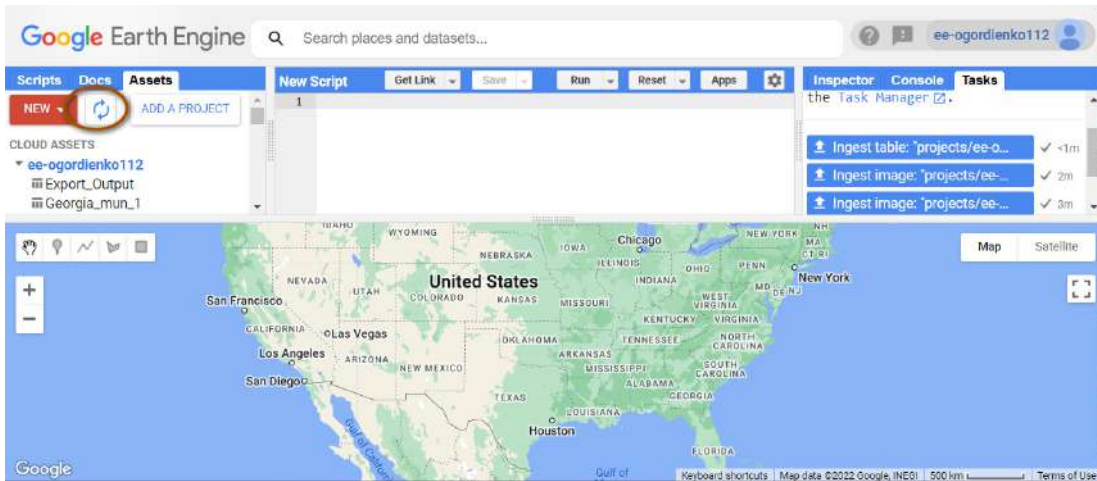



Рис. 50. Оновлення даних

Тепер файли мають з'явитися у вкладці *Assets* (Менеджер об'єктів).

3.3.1. Імпортувати векторні дані.

Почнемо із шейп-файлу, який називається *chornobil*. Для того щоб завантажити його до робочого середовища, знайдіть його в списку доступних об'єктів *Legacy assets* (Доступні об'єкти) і натисніть  на кнопку зі стрілкою вправо *Import into script* (Імпорт у скрипт).

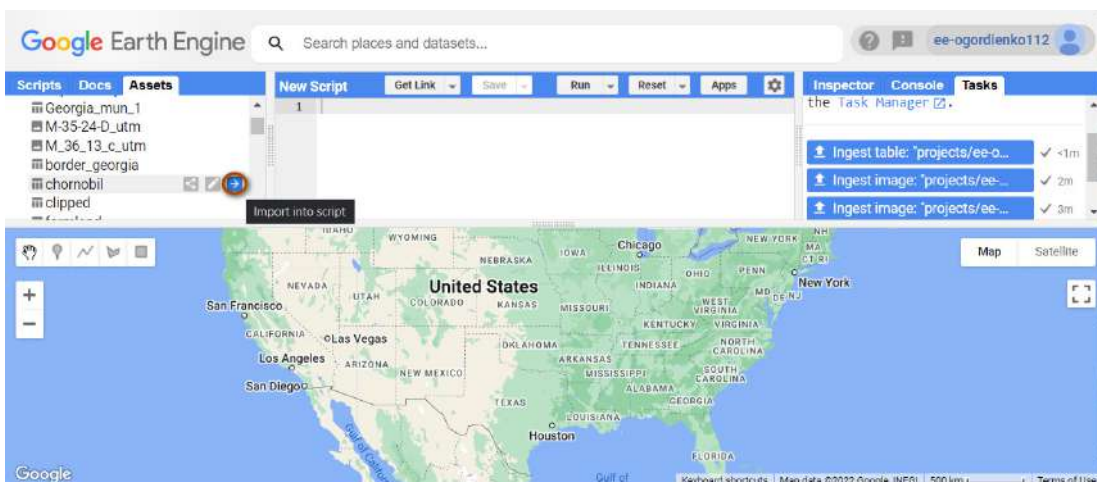


Рис. 51. Імпорт даних до робочого простору

Після цього він з'явиться у вкладці *Imports* (Імпорт) з назвою за замовчуванням *table*. Цю назву необхідно змінити на *OSM_forest*. Також буде видно шлях, де зараз перебуває файл.

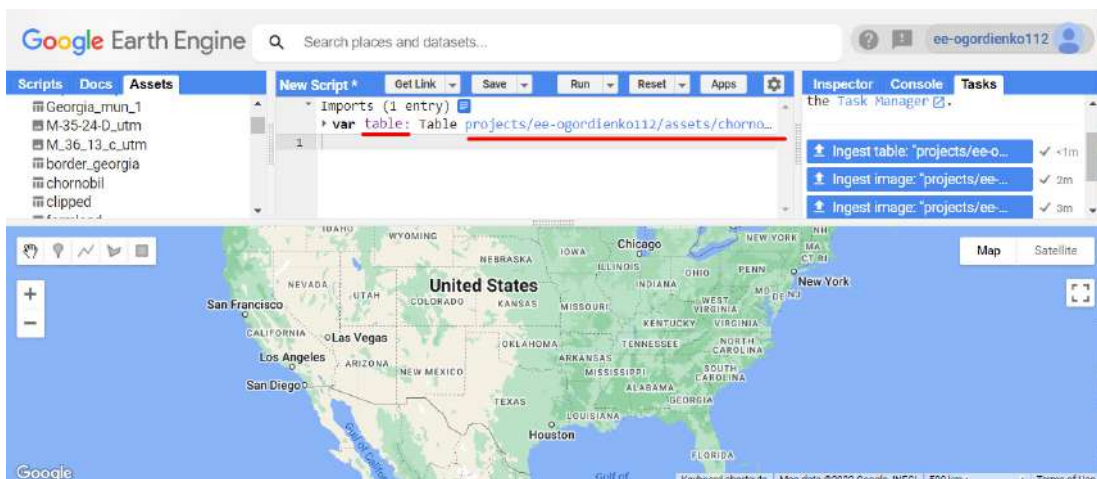


Рис. 52. Імпортовані дані

Змінімо **table** на **OSM_forest**. Зробити це можна так: натиснути один раз на назву, вона підсвітиться синім; увести нову назву – **OSM_forest**.

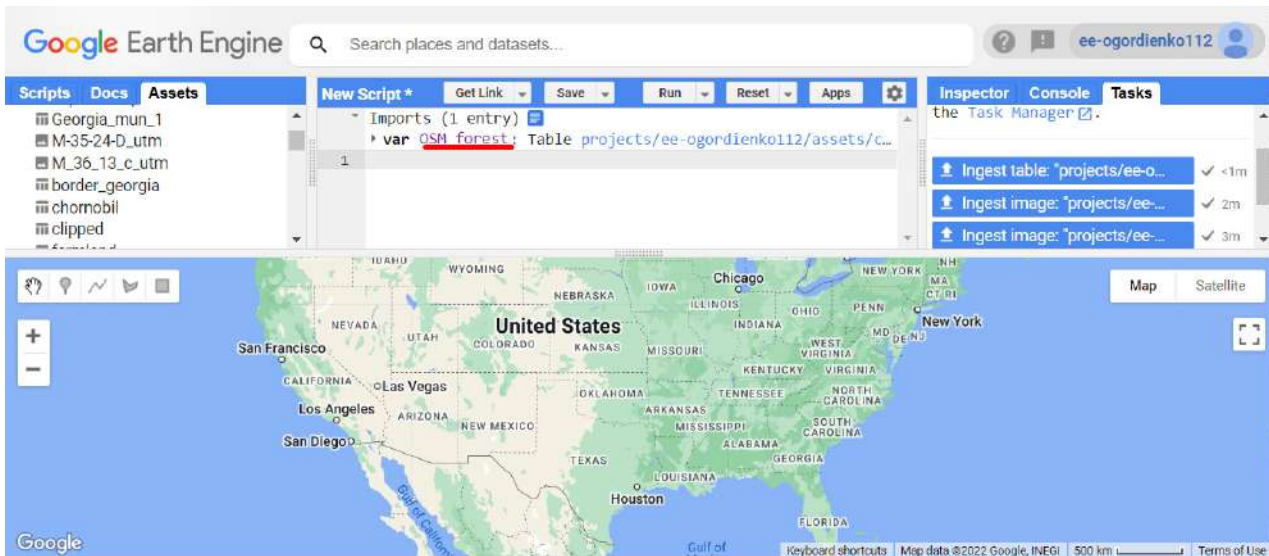


Рис. 53. Зміна назви даних

Тепер векторний файл з потрібною назвою є у вас у робочому середовищі. Далі потрібно завантажити файли з мапами. Робиться це схожим способом.

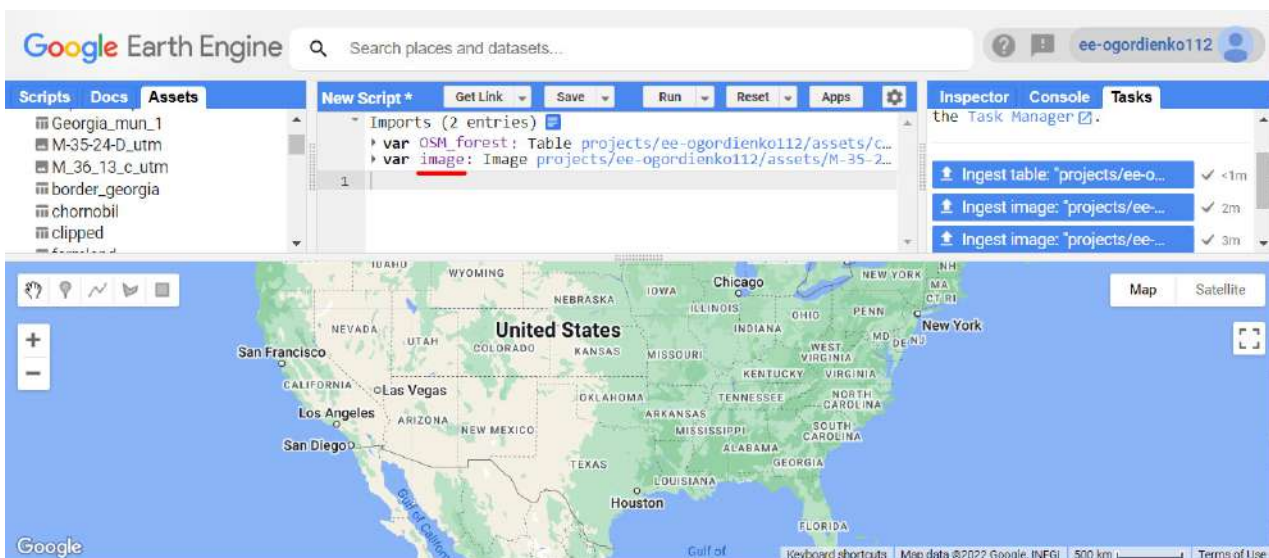


Рис. 54. Завантаження растра до робочого простору

3.3.2. Імпортувати растрові дані.

Знайдіть у списку файл із першою мапою і натисніть на кнопку зі стрілкою вправо **Import into script** (Імпорт у скрипт). Після цього файл з'явиться у вкладці **Imports** (Імпорт) і буде мати назву **image**. Його теж необхідно перейменувати на **M_35_24_D_utm**, натиснувши на назву файлу.

✔ **Важливо!** GEE в назвах файлів не сприймає знак тире (–) і знак пробілу (), тому всі ці знаки необхідно змінити на нижнє підкреслення (_). Назви обов'язково давати латинськими літерами.

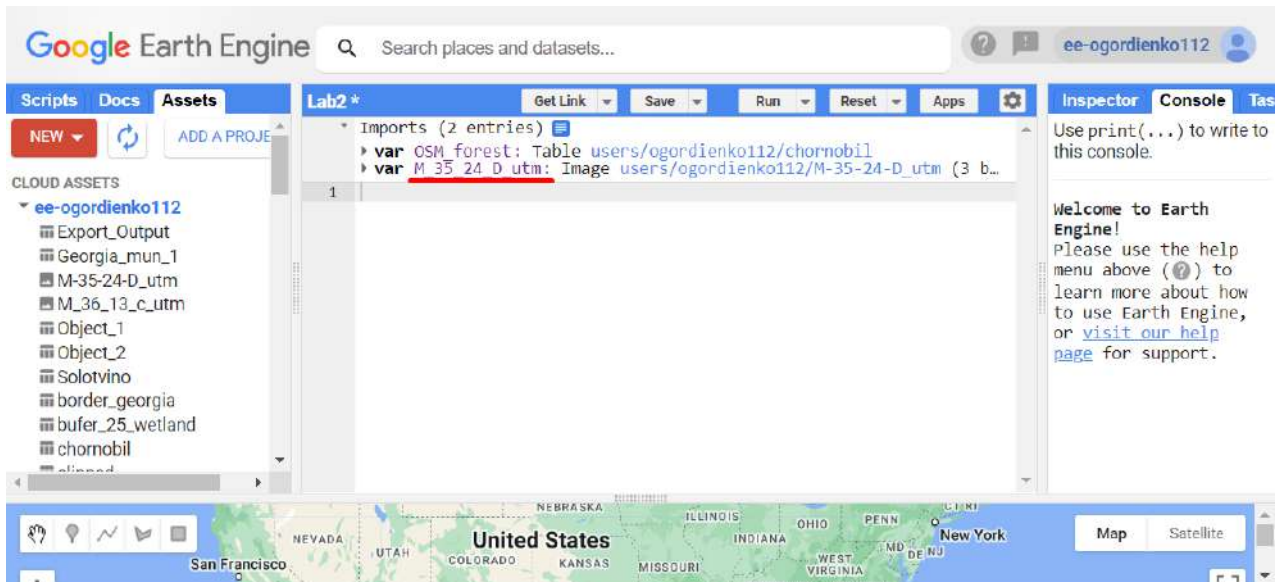


Рис. 55. Зміна назви растрових даних

Так само завантажуюмо і наступну мапу. Знаходите в **Assets** (Менеджер об'єктів) файл з назвою *M_36_13_c_utm*. Він з'явиться у вкладці **Imports** (Імпорт) і буде мати назву *image*. Його теж необхідно перейменувати на *M_36_13_C_utm*, натиснувши на назву файлу.

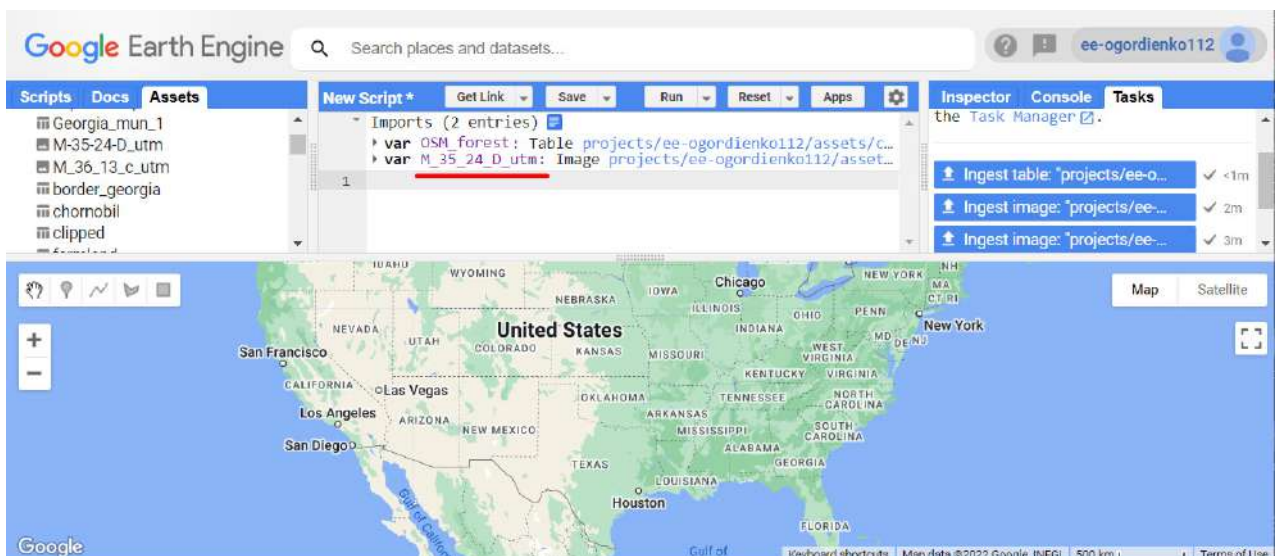


Рис. 56. Змінена назва растрових даних

4. Наблизитися до території дослідження й додати імпортовані дані на оглядову мапу. Для початку змінімо вигляд мапи на гібрид, додавши рядок коду.

<code>Map.setOptions('Hybrid');</code>	Map.setOptions змінює вид мапи
--	--------------------------------

Ви установили вид мапи HYBRID у Google Earth Engine. Нагадуємо, що можна встановити й інші параметри для відображення мапи, як-от: ROADMAP, SATELLITE, TERRAIN.

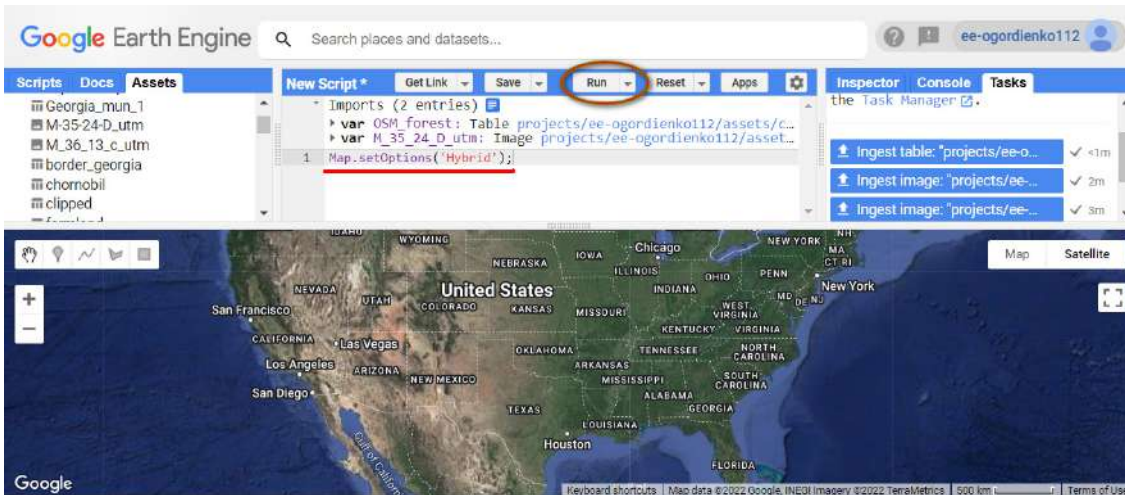


Рис. 57. Запуск скрипту

Тепер призумимося до підвантаженого нами вектора.

<code>Map.centerObject(OSM_forest,10);</code>	<code>Map.centerObject</code> центрує об'єкт на мапі по шару <code>OSM_forest</code> з 10-кратним зумом
---	---

Натисніть **Run** (Запуск скрипту).

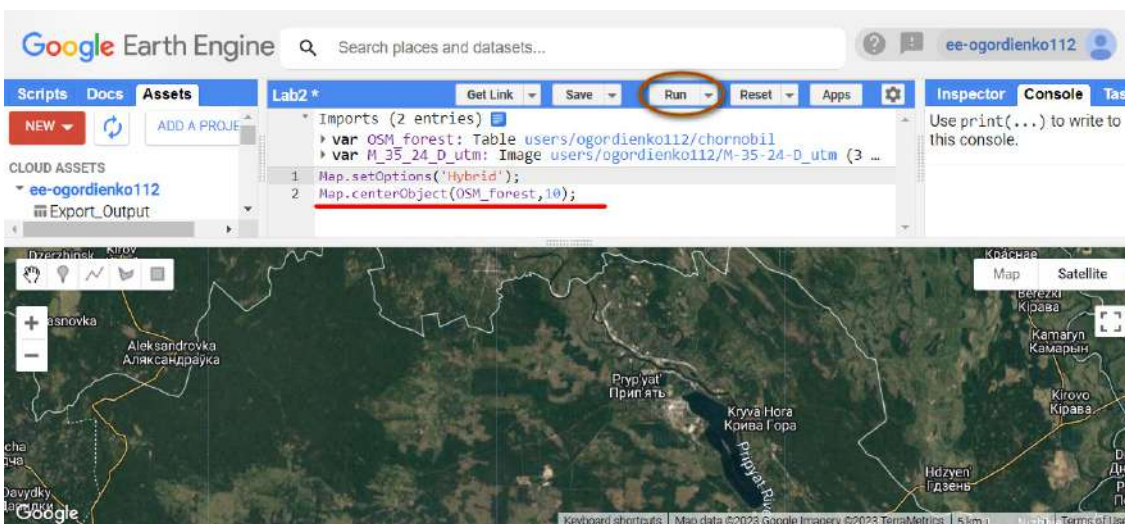


Рис. 58. Центрування мапи

Зум працює, навіть коли шари не відображаються на мапі. Пропонуємо візуалізувати перший векторний шар із лісами з OSM.

Векторні дані можна додавати на мапу безпосередньо за допомогою `Map.addLayer()`. Візуалізація за замовчуванням відобразить вектори із суцільними чорними лініями і напівпрозорою чорною заливкою. Щоб відобразити вектори в кольорі, треба вказати параметри кольору.

<pre>var styleParams = { fillColor: 'FF00000', color: '00ff00', width: 1.0, }; var OSM_forest_vis = OSM_forest.style(styleParams); Map.addLayer(OSM_forest_vis, {}, 'OSM forest');</pre>	<p><code>styleParams</code> відповідає за візуальні параметри. У нашому випадку <code>fillColor</code> (тобто колір заливки) має значення <code>'FF00000'</code> це означає, що колір заливки полігонів прозорий, а <code>color</code> (колір контуру) зі значенням <code>'00ff00'</code> вказує на салатовий колір вектора, тобто за цифрами <code>'00ff00'</code> ховається конкретний колір, яких у кольоровій гамі багато*.</p> <p>Застосуємо візуальні параметри за допомогою <code>style</code> до <code>OSM_forest</code> шару, який ми підвантажили до GEE; <code>Map.addLayer</code> додає шар до мапи;</p>
--	--

Кольори в GEE задаються різноманітними способами. Спосіб кодування кольорів, який ми використовуємо, називається HEX, тобто шістнадцяткові значення.

Спробуйте змінити (color: '00ff00') на (color: 'Lime'). Це кодування кольорів у більш зручному для людини читанні. Більшості кольорів відповідають такі текстові значення. Для вибору інших кольорів і розуміння їх назв можете скористатися покликанням <https://www.mathsisfun.com/numbers/hexadecimal-color-names.html>, де наведені базові кодування кольорів.



Рис. 59. Кодування HEX

<pre>var styleParams = { fillColor: 'FF000000', color: 'Lime', width: 1.0, };</pre>	<p>styleParams у цьому випадку не змінюються візуально, але тепер значення color: 'Lime' більш зручно читається людиною;</p>
---	--

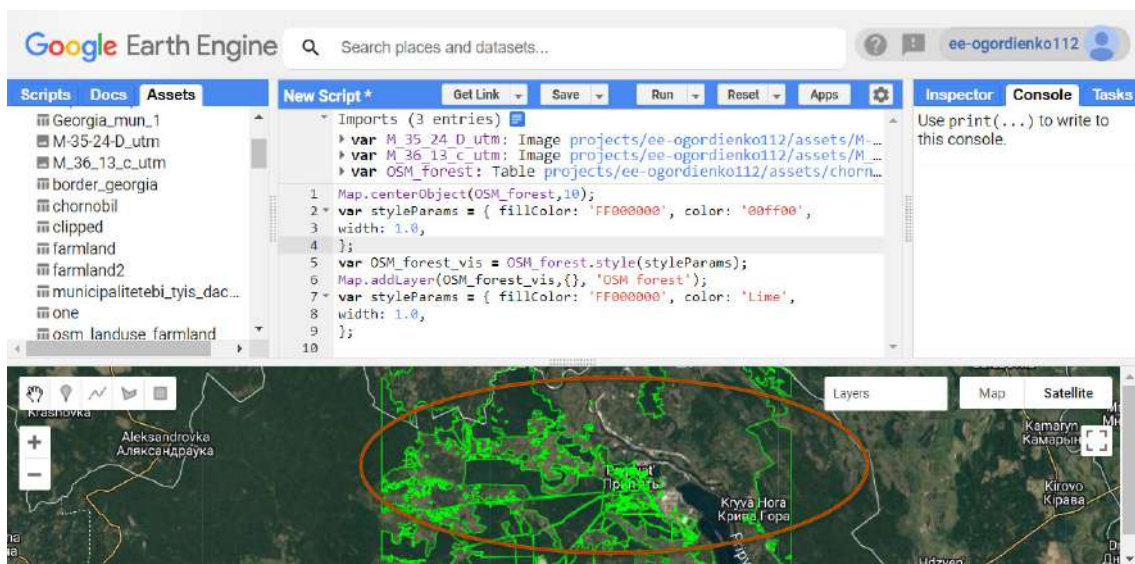


Рис. 60. Зміна відображення шару

Шар відобразиться на мапі, і з'явиться додатковий інструмент **Layers (Шару)** на панелі.

Тепер необхідно додати на мапу історичні мапи, які ми завантажували. Для цього скористайтеся цим кодом і натисніть **Run (Запуск скрипту)**.

<pre>var imageVisParam_1 = { bands: ["b1", "b2", "b3"], gamma: 3.563, max: 203.46603306820376, min: 161.15789855572788, opacity: 1 }; Map.addLayer(M_36_13_c_utm, imageVisParam_1, 'M_36_13_c_utm', false);</pre>	<p>Візуальні параметри – вибираються три канали ["b1", "b2", "b3"] до них застосовується гама максимальне та мінімальне значення для розтягування гістограми та непрозорість; Opacity – прозорість = 1, а може набувати значення від 0 до 1</p> <p>Map.addLayer додає шар з назвою M_36_13_c_utm із вбудованими візуальними параметрами з назвою M_36_13_c_utm та false значення відображення;</p>
--	--

5. Змінити параметри візуального відображення топокарти.

Існує декілька методів `ee.Image`, які створюють візуальне зображення даних знімка в RGB, наприклад: `visualize()`, `getThumbURL()`, `getMap()`, `getMapId()` та `Map.addLayer()`. За замовчуванням ці параметри призначають перші три шари червоному, зеленому та синьому каналу відповідно. Розтягнення за замовчуванням ґрунтується на типі даних у діапазонах (наприклад, числа з плаваючою здатністю розтягуються в $[0, 1]$, 16-бітові дані розтягуються до повного діапазону можливих значень), що може підійти або не підійти. Щоб досягти бажаних ефектів візуалізації, ви можете надати параметри візуалізації. Це ми опишемо далі.

Тут візуалізуємо одну з топографічних карт `M_36_13_c_utm` і додаємо її на оглядову мапу. Мапа має з'явитися у вкладці **Layers (Шари)**, але через встановлений параметр `false` вона не відображається. Тому зайдіть у **Layers (Шари)** і поставте галочку навпроти шару з назвою `M_36_13_c_utm`. Шар відобразиться правильно.

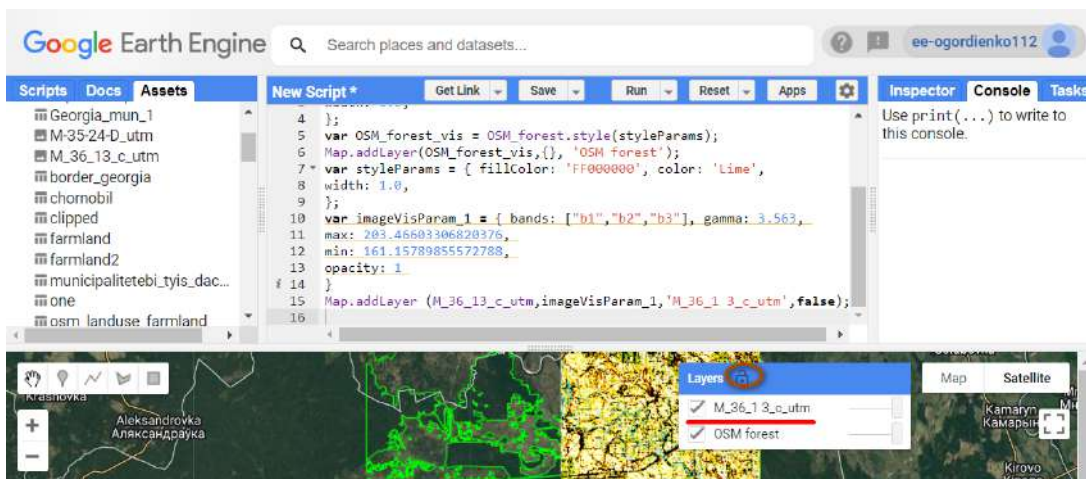


Рис. 61. Відображення даних

Наступним нашим кроком буде відображення в оглядовому вікні мапи другої історичної топомапи з назвою `M_35_24_D_utm`.

```
Map.addLayer (M_35_24_D_utm, {}, 'M_35_24_D_utm', false);
```

Фігурні дужки вказують на відображення мапи за замовчуванням без додаткових налаштувань;

У ньому також стоїть параметр `false`, тому зайдіть знову до **Layers (Шари)** і поставте галочку навпроти нового шару. Після натискання **Run (Запуск скрипту)** та відмітки галочкою шару він з'явиться, але через те що параметри візуалізації не були задані, буде відображатися чорним за замовчуванням.

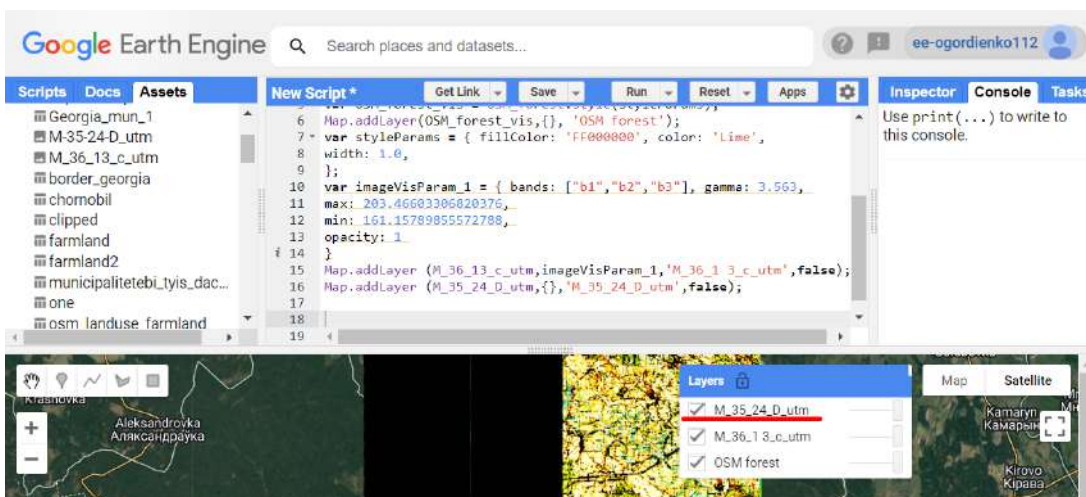



Рис. 62. Відображення іншого растра

Потрібно змінити параметри візуалізації для мапи *M_35_24_D_utm* і, в цьому випадку ми будемо задавати їх вручну. Для цього потрібно зайти в параметри відображення візуалізованого шару. Там, де ви ставили галочку в *Layers (Шари)*, є значок із символом шестерні . Після того як ви натиснете на нього, відкриється нове вікно.

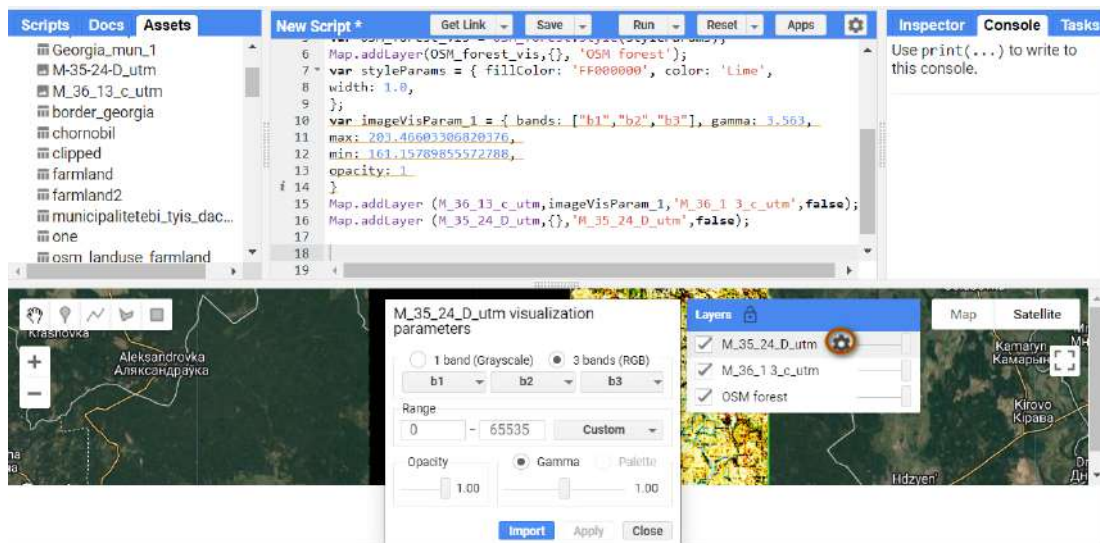


Рис. 63. Налаштування растра

У вікні, де задаються параметри візуалізації, необхідно виконати розтягування гістограми, тобто змінити *Custom (За замовчуванням)* на *Stretch: 3 σ (Розтягування: 3 σ)* та значення **Sigma** (сигма) за допомогою повзунка, що міститься під **Sigma** (Сигма), змінити значення 1.00 на 3.56, тобто менше 4 і більше 3.5 та натиснути *Apply (Виконати)*. Шар візуалізується по-новому, і тепер необхідно натиснути на *Import (Імпортувати)*, після чого шар з'явився в *Imports (Імпорт)*, а ви можете закрити це вікно, натиснувши *Close (Закрити)*.

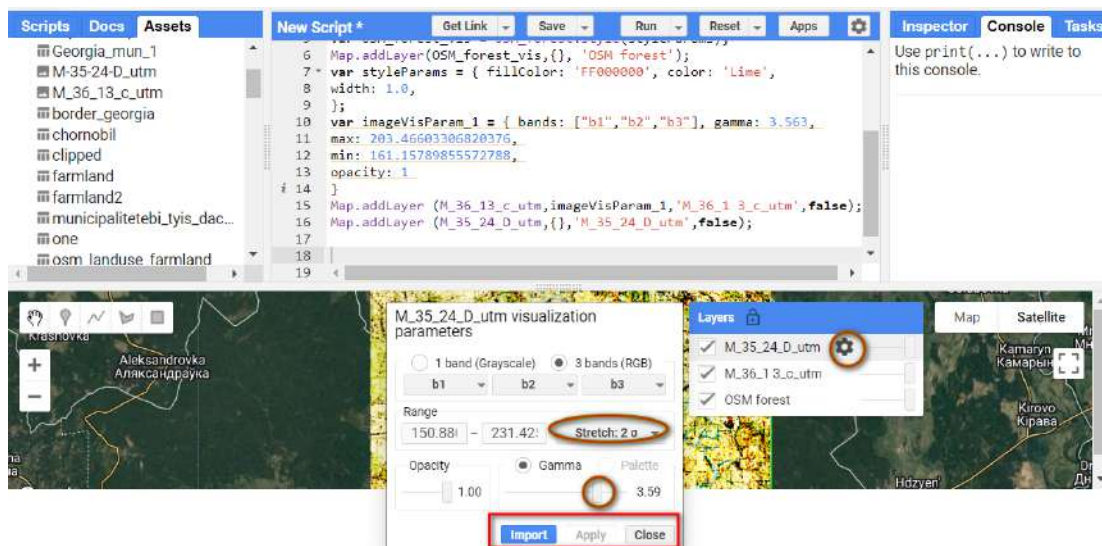


Рис. 64. Імпорт налаштувань

Друга історична мапа візуалізується, але параметри візуалізації задані лише на цей момент. Натисніть *Run (Запуск скрипту)*. Після цього наші налаштування шару зникнуть, через те що за замовчуванням для шару *M_35_24_D_utm* візуалізація відсутня. Додаймо параметри візуалізації до нашого шару. Після того як ви натиснули *Import (Імпортувати)*, ці налаштування з'явилися в *Imports (Імпорт)*, і зараз ми їх додамо.

Скопіюйте *imageVisParam* і вставте замість `{}` у `Map.addLayer(M_35_24_D_utm, {}, 'M_35_24_D_utm', false);` замість галочок потрібно, щоби було так: `Map.addLayer(M_35_24_D_utm, imageVisParam, 'M_35_24_D_utm', false);`

```
Map.addLayer (M_35_24_D_utm,imageVisParam,'M_35_24_D_utm',true);
```

Ці шари вже додані в попередніх кроках, просто змініть значення `false` на `true` та додайте `imageVisParam`, який ви імпортували замість фігурних дужок

Також для полегшення подальшої роботи зі скриптом потрібно змінити обидва значення `false` на значення `true`.

? Завдання для перевірки

Налаштувати однакове візуальне відображення обох топографічних карт із розтяжкою 3 σ і відключити їх автоматичне відображення на оглядовій мапі, змінивши `false` на `true`

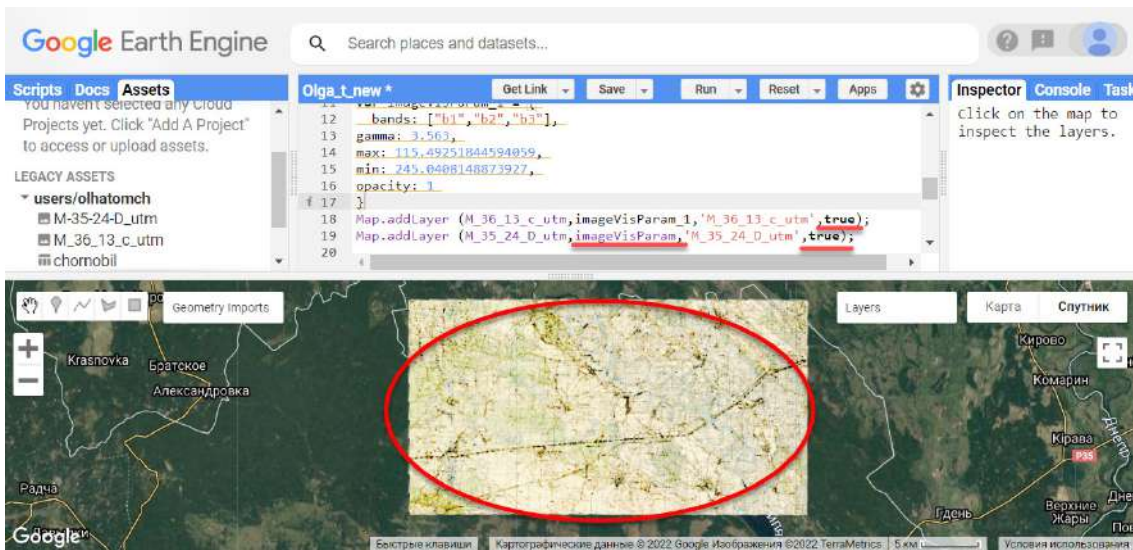


Рис. 65. Відображення растра

Тепер обидві історичні карти коректно візуалізуються.

6. Створити векторний шар *OldForest* та оцифрувати контури лісу з топографічних карт.

Для подальшого аналізу нам необхідно окреслити ліс, який є на історичній мапі, та порівняти з лісом, який є в базі даних OSM.

Щоб рисувати свої власні геометричні об'єкти в GEE, є інструменти *Geometry instruments*.

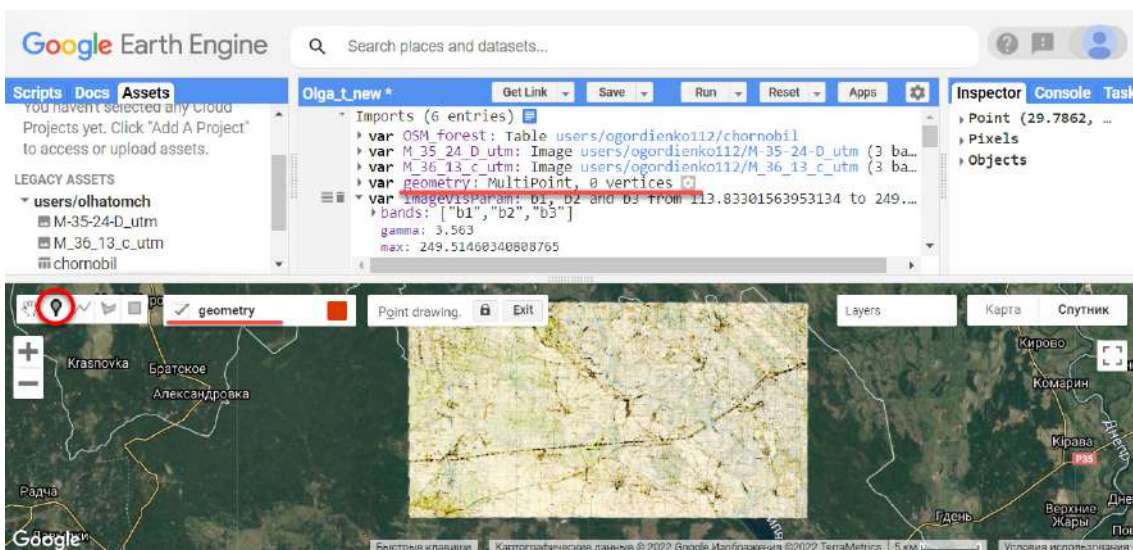


Рис. 66. Імпорт геометрії

У *Geometry imports* (Імпорт геометрії) з'явиться новий шар із назвою за замовчуванням *geometry*.

Тепер змініть його назву на *oldForest_Feature*.

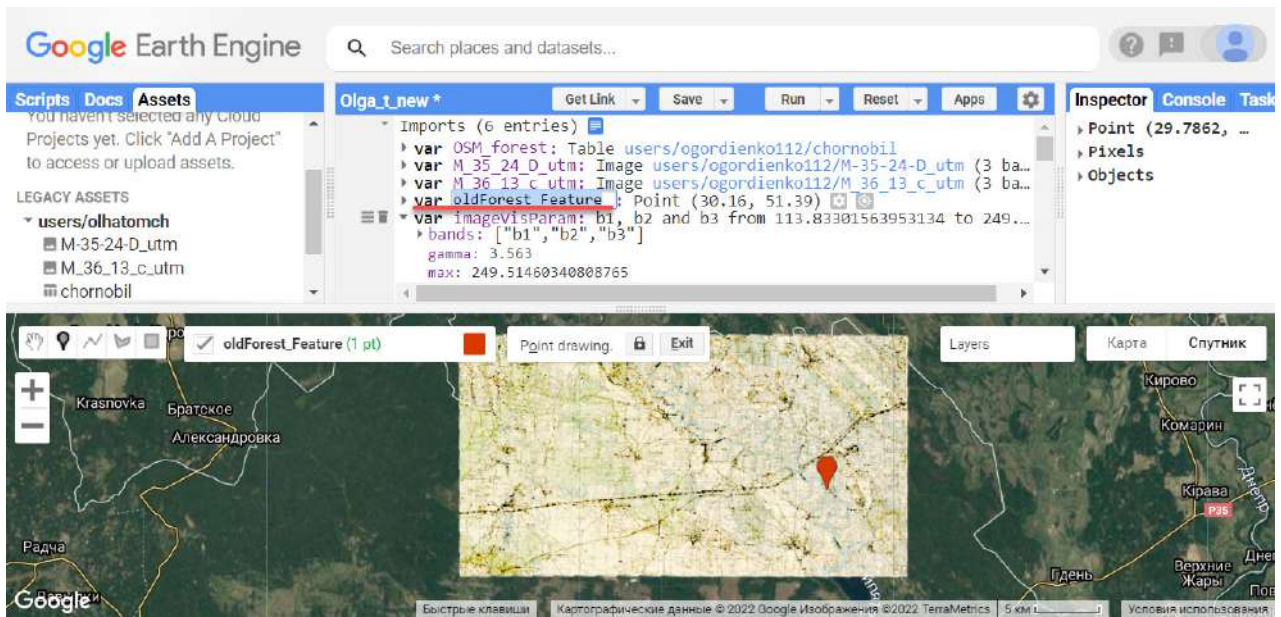


Рис. 67. Вид мапи

За допомогою коліщата миші наблизьтеся до мапи й обведіть ліс.

Умовні позначення!

Який вигляд має дерев'янисто-чагарникова рослинність на топокартах масштабу 1:25 000?

<p>Суцільний ліс – виділений на мапі зеленим кольором, однотонним або на фоні із символічними деревами (залежно від типу лісу – хвойний або листяний)</p>		
<p>Рідколісся – це нещільний (рідкий) ліс, де дерева віддалені одне від одного на помітну відстань. Відображається на мапі символом без кольорового зеленого фону</p>		
<p>Кущі – це дерев'яниста рослинність з висотою до 2 метрів, відображається на мапі символом ; якщо суцільні зарості кущів, то на зеленому фоні</p>		

Рис. 68. Види поверхні на топографічній карті

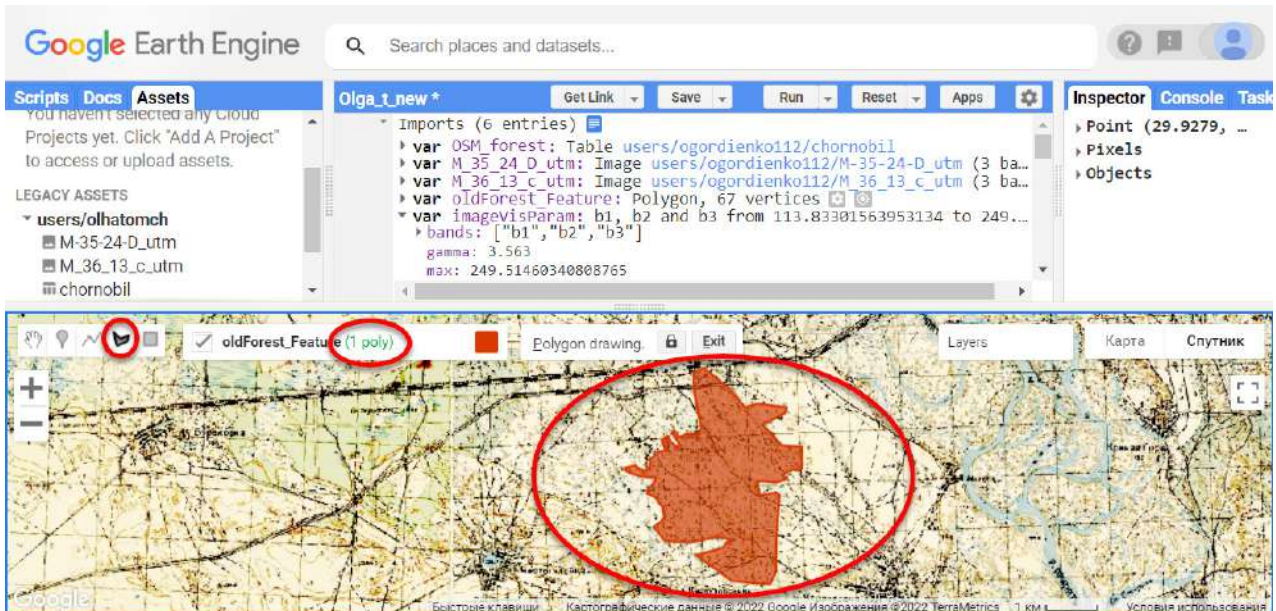



Рис. 69. Вигляд мапи

На топокартах є декілька ділянок з лісом. Обведіть найбільші ділянки. Робити це можна доволі грубо під час навчання, а для повноцінного дослідження пам'ятайте: детальність оцифрування буде безпосередньо впливати на точність підрахунку площі. Під час оцифрування завершити скетч полігональної фігури можна подвійним клацанням лівої клавіші миші. Щоб видалити непотрібний полігон, виберіть інструмент  у *Geometry instruments* (Інструменти геометрії).

Для видалення натисніть *Delete* (Видалити).

Завдання для перевірки

Доцифруйте декілька нових ділянок лісу в правому верхньому кутку, щоб вигляд був, як на зображенні нижче

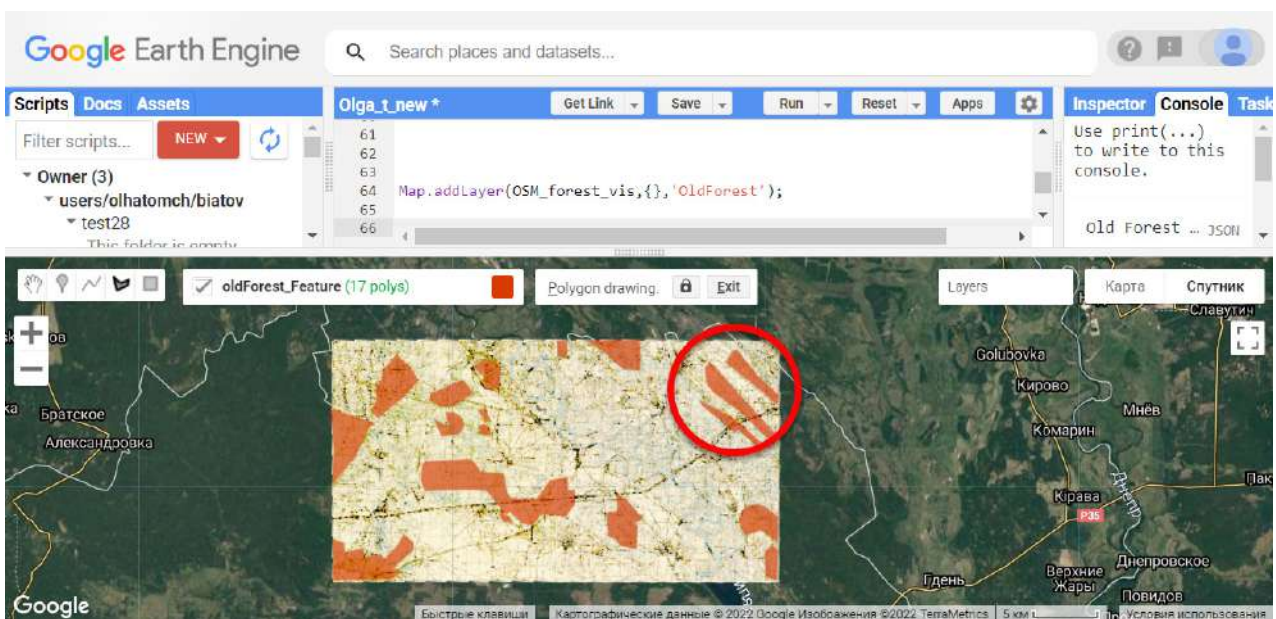


Рис. 70. Інший вигляд оцифрованої ділянки

Зараз на мапі додано двічі шар з оцифрованим лісом. Для зручності використання скрипту необхідно вимкнути шар у *Geometry imports* (Імпорт геометрії) і зняти галочку навпроти *oldForest_Feature* в *Geometry imports* (Імпорт геометрії).

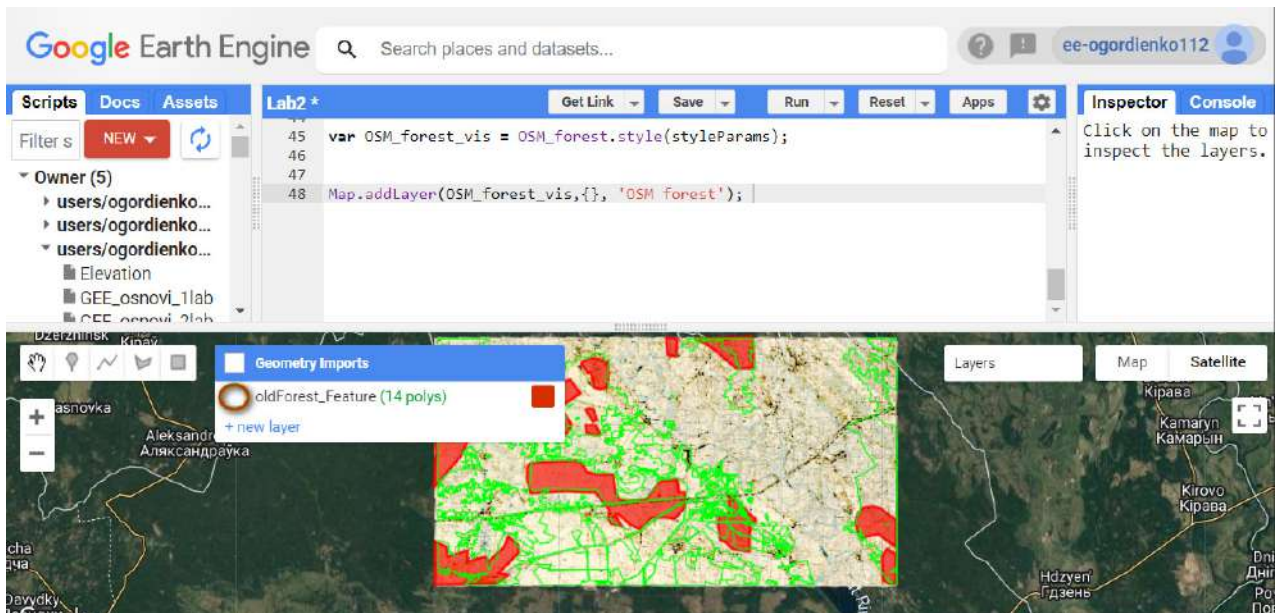


Рис. 71. Накладний шар з OSM

Тепер необхідно додати оцифровані шари до мапи. Зробити це можна за допомогою коду:

```
Map.addLayer(oldForest_Feature,{color:'red'},
OldForest');
```

Додається до мапи шар з назвою `oldForest_Feature` з кольором `'red'` та назвою `'OldForest'`

Натисніть **Run** (Запуск скрипту).

7. Візуально порівняти сучасну лісистість території з 1937 р. і розрахувати загальну площу лісів. Для візуального порівняння відключіть шари з історичними мапами в **Layers** (Шари).

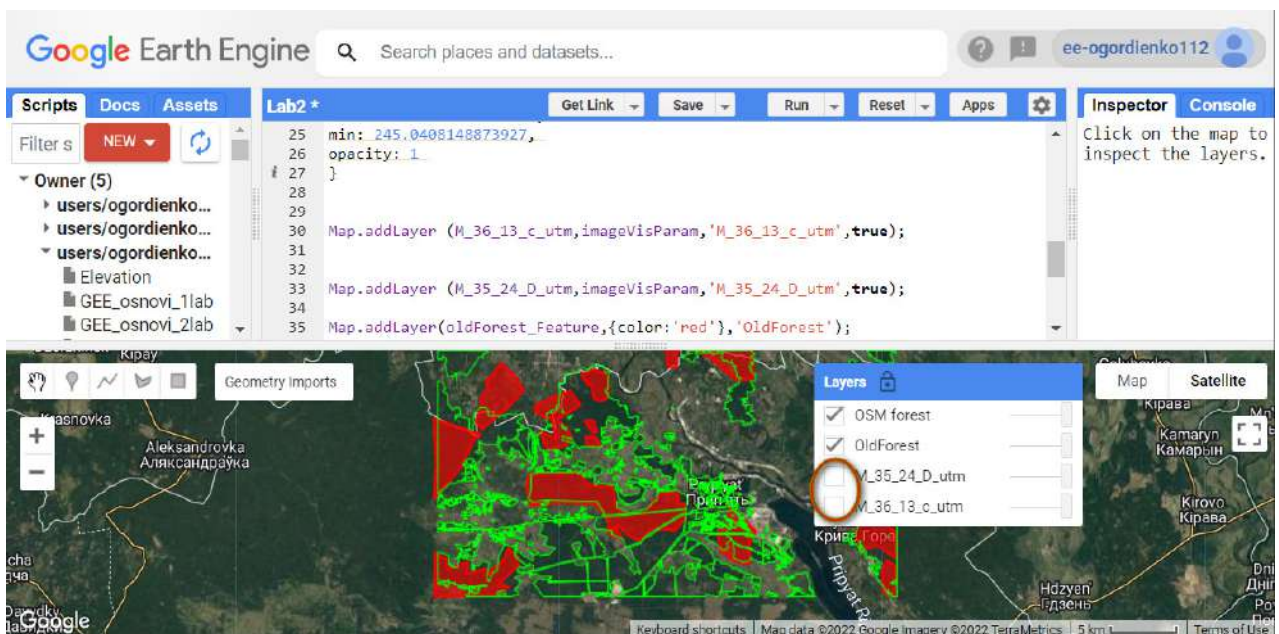


Рис. 72. Відключення растрових шарів

? Завдання для перевірки

Візуально порівняйте ці два шари векторів лісистості Чорнобильської зони

7.1. Отримати площу виділеного лісу з архівних топографічних карт 1937 р.
Тепер можна провести порівняння площ лісів. Для цього скористайтесь кодом:

<pre>var oldForestArea = oldForest_Feature.area(); var oldForestAreaSqKm = ee.Number(oldForestArea).divide(1e6).round(); print('Old Forest size (km²)',oldForestAreaSqKm);</pre>	<p>Для oldForestArea рахується площа за допомогою area();</p> <p>Тут порахована площа береться як число, множиться на 1 000 000 та скорочується до цілого;</p> <p>Результат у консоль з підписом Old Forest size (km²)'</p>
---	--

Натисніть **Run** (Запуск скрипту).

Для створених нами нових об'єктів oldForest_Feature рахується площа за допомогою area(). Після цього oldForestAreaSqKm приймає пораховане значення за число, множить його на 1 000 000 і зводить до цілого. Тепер площа для старого лісу, який ми оцифрували з топокарт, порахована.

Також на цьому етапі для зручності можна відключити відображення за замовчуванням шарів з історичними мапами.

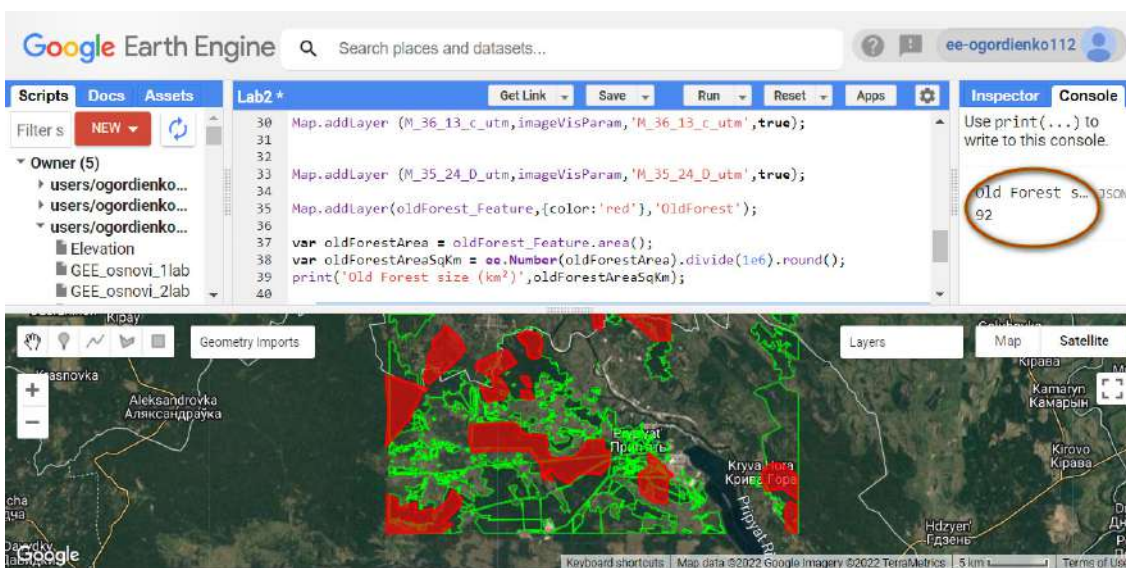


Рис. 73. Відображення площі лісів

7.2. Розрахувати площу сучасного лісу на основі OSM даних.

Тепер необхідно поррахувати площу сучасного лісу, який створили користувачі OSM. Це можна зробити за допомогою коду:

<pre>var addareaOSM = function(feature){ return feature.set({areaGEE: feature.geometry().area()}); }; var areaadedOSM = OSM_ forest.map(addareaOSM); var sumAreaOsm = areaadedOSM.aggregate_ sum('areaGEE').divide(1e6).round(); print (sumAreaOsm,'OSM Forest size (km²)');</pre>	<p>Функція для розрахунку площ, яка бере на вхід об'єкт, а на вихід подає пораховану геометрію geometry() цього об'єкта за допомогою area() та створює нову колонку з назвою areaGEE, куди зберігає ці дані;</p> <p>map застосовує попередню функцію addareaOSM до датасету OSM_forest</p> <p>До areaadedOSM застосовується aggregate_sum, який рахує суму всіх об'єктів, що знаходяться в 'areaGEE', множить на 1 000 000 та зводить до цілого</p> <p>Результат у консоль з назвою 'OSM Forest size (km²)'</p>
--	--

Ми створили функцію з назвою `addareaOSM`, яка бере на вхід об'єкт, а повертає порохвану площу, що відображається в колонці з назвою `areaGEE`. Ця функція рахує площу за геометрією об'єкта. Після цього за допомогою `map` ми застосовуємо функцію до датасету `OSM_forest`.

Потім нова змінна з назвою `sumAreaOsm` за допомогою `aggregate_sum` рахує всі значення, що перебувають в `'areaGEE'`, множить їх на 1 000 000 і зводить до цілого.

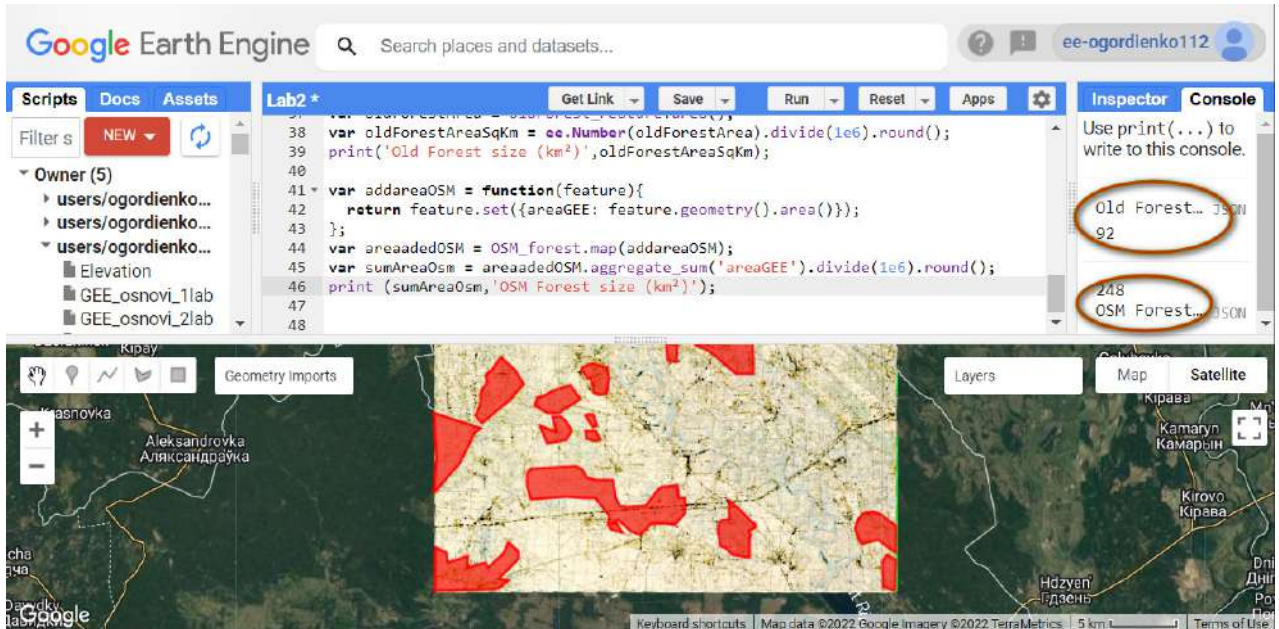


Рис. 74. Відображення розрахованої площі

? Завдання для перевірки

Вкажіть, наскільки змінилася площа лісу за майже 100 років? Напишіть отримані площі за 1937 р. та із сучасних векторних даних OSM

На завершення збережіть ваш скрипт. Це можна зробити, натиснувши кнопку **Save** (Зберегти).

З'явиться нове вікно зі шляхом, куди ви хочете зберегти файл, і з полем, де можна ввести ім'я файлу. Введіть назву **Lab3** і натисніть кнопку **OK**.

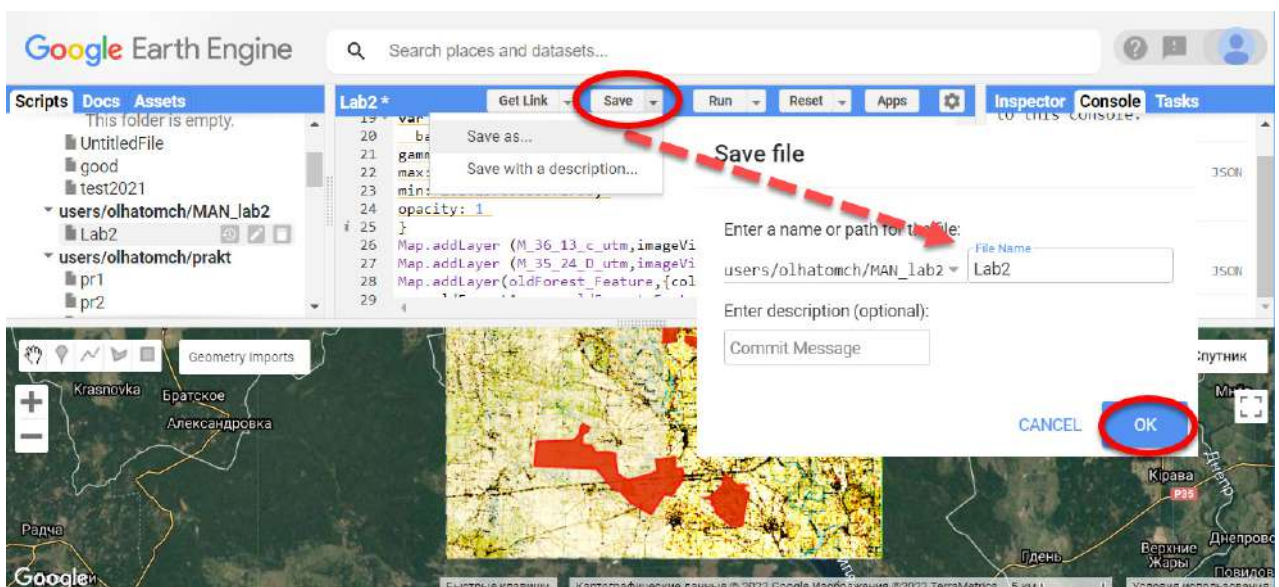


Рис. 75. Збереження скрипту

Скопіюйте покликання на ваш відредагований скрипт через **Get link** (Отримання лінку).

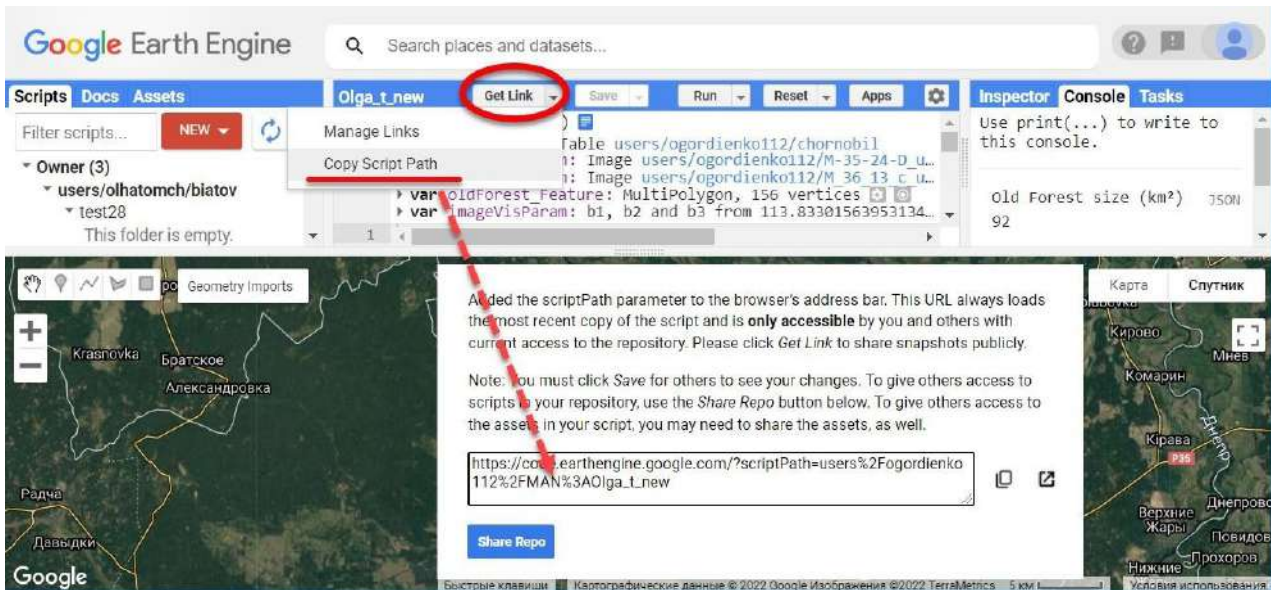


Рис. 76. Копіювання лінку для поширення

За наведеним нижче QR-кодом ви можете переглянути відео виконання практичної роботи.



Рис. 77. Відео виконання практичної роботи

Готовий код можна відкрити за лінком:

https://code.earthengine.google.com/?scriptPath=users%2Fgis_rs%2Fcourse_MAN%3ALab_3

Створення графіків на основі тематичних даних для дослідження розподілу кількості опадів та вологості в Олешківських пісках



Ситуація

Величезний піщаний масив (понад 160 тисяч гектарів) у пониззі Дніпра називають пустелею. Деякі туроператори відмовлятимуть вас їхати в Сахару, адже справжнісінька українська пустеля тут неподалік, на Херсонщині. З одного боку, вони мають рацію, бо тут можна побачити дюни й бархани, специфічну пустельну флору і навіть зустріти представників тваринного світу, але завжди є «але» :) Помічали таке?

Отже, якщо пригадати собі визначення пустелі, то це специфічний тип ландшафту з розрідженістю або абсолютною відсутністю флори, а з фауни присутні лише специфічні та притаманні лише їй види¹. Ще однією і дуже важливою ознакою пустель є посушливі (аридні) умови, а також те, що річна сума опадів тут становить менше 200 міліметрів на рік.

Завдання

Перевіримо розподіл опадів та вологість ґрунту на Олешківських пісках (ОП) в період із 2010 р. по 2020 р. і спробуємо підтвердити або спростувати гіпотезу про пустелю в Україні.

Алгоритм виконання завдання:

1. Відкрити Code Editor і визначити територію Олешківських пісків.
2. Підвантажити датасет CHIRPS для дослідження розподілу опадів.
3. Вибрати часовий період із 2010 р. по 2020 р. для аналізу даних і датасету CHIRPS.
4. Побудувати графік для розподілу опадів на Олешківських пісках.
5. Підвантажити датасет GLDAS для виміру вологості ґрунту на глибині.
6. Вибрати часовий період із 2010 р. по 2020 р. для аналізу даних із датасету GLDAS.
7. Побудувати графік для вологості ґрунту на Олешківських пісках.
8. Зберегти отримані графіки опадів і вологості у форматі PNG.
9. Створити новий вектор території дослідження для пустелі Сахара.
10. Побудувати аналогічні графіки опадів та вологості ґрунту для пустелі Сахара.

Покрокова інструкція

1. Відкрити Code Editor і визначити територію Олешківських пісків.

Відкрийте **Code Editor** (Редактор коду): <https://code.earthengine.google.com/>. У рядку **Search** (Пошук місця та даних) напишіть «Олешківський». Зі списку, що відкрився, виберіть Олешківський район, Херсонська область, Україна. На жаль, пошук за назвою географічного об'єкта не доступний, тому радимо шукати так або за назвою найближчого населеного пункту, наприклад «Раденськ».

¹ URL: <https://www.wikiwand.com/uk/Пустеля>

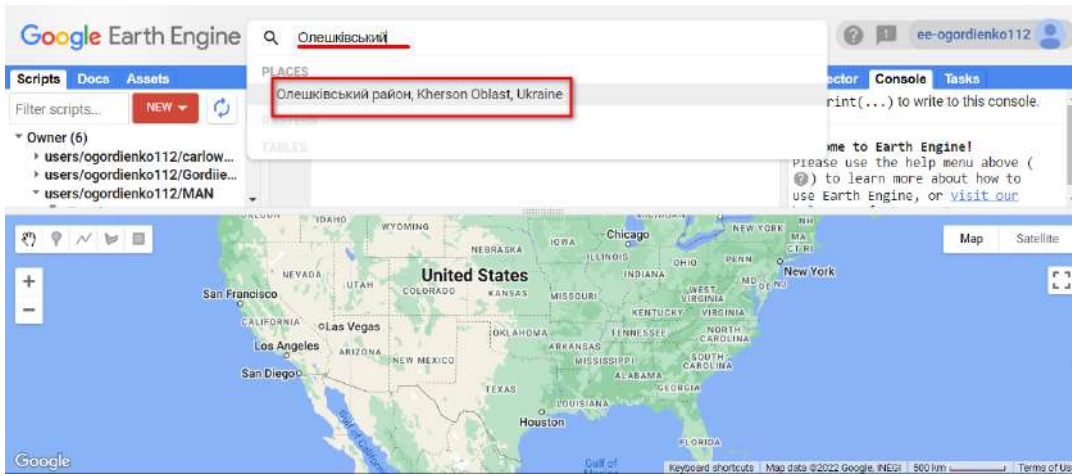


Рис. 78. Пошук території цікавості

Вікно мапи швидко перенесе вас зі США в Україну, на Херсонщину. Тоді, користуючись або коліщатком миші, або кнопками навігації, наблизьтеся до Олешківських пісків (далі будемо їх називати ОП, щоб менше писати). Якщо ви не знаєте, де точно розташовані ОП, тоді підказкою для вас буде рисунок нижче або те, що вони округлої форми, розташовані між Херсоном і Новою Каховкою, поблизу Раденська.

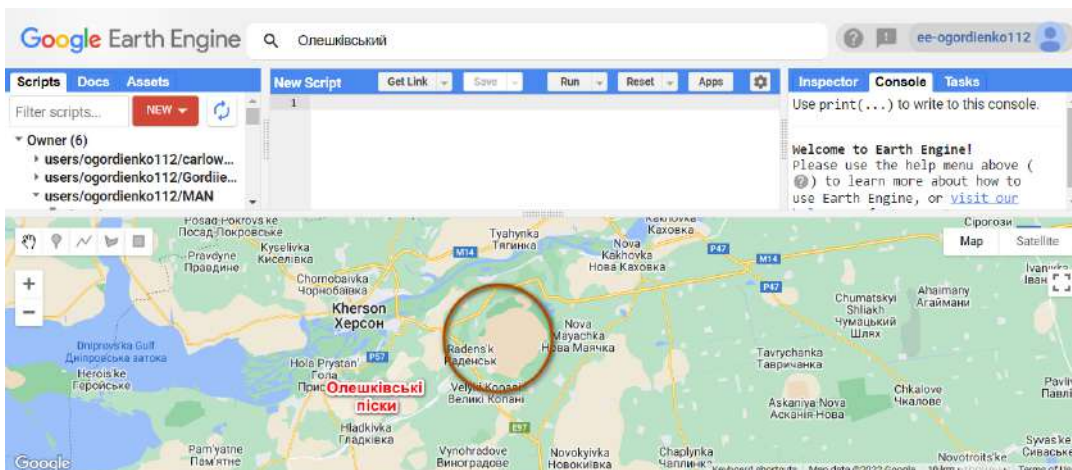


Рис. 79. Центрування до зони цікавості

Змініть відображення мапи з **Map** (Мапа) на **Satellite** (Супутник), перемкнувши кнопки у верхньому правому кутку вікна для того, щоб переконатися, що це точно масив пісків, оскільки на мапах Google він відображається зеленим, як і ліс навколо нього.

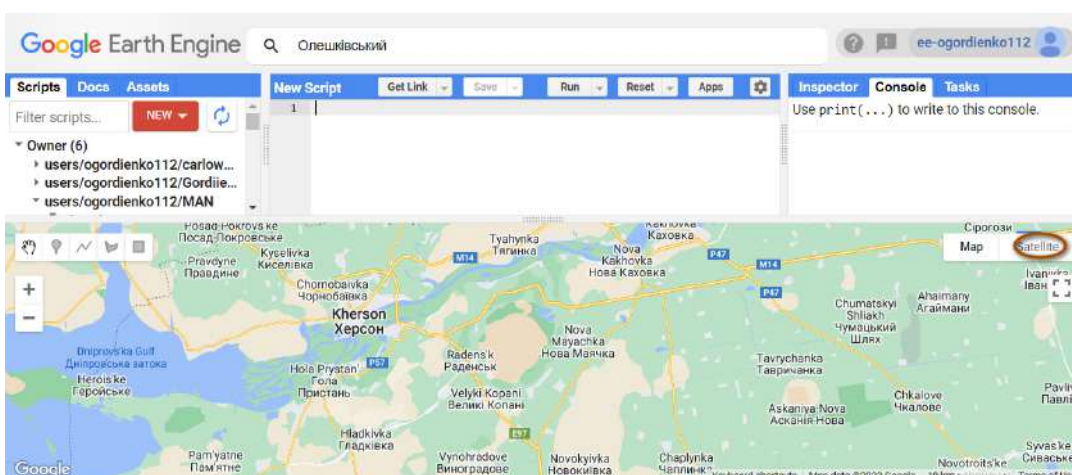


Рис. 80. Зміна виду мапи

? Завдання для перевірки

Який код можна використати, щоб змінити вид мапи?

У цій практичній роботі ми працюємо з метеорологічними даними. Для того щоб отримати дані про кількість опадів саме для території ОП, потрібно обмалювати їх, вибравши кнопку **Draw a shape** (Накреслити фігуру) на панелі інструментів **Geometry instruments** (Інструменти геометрії). Довільно окресліть територію ОП полігоном.

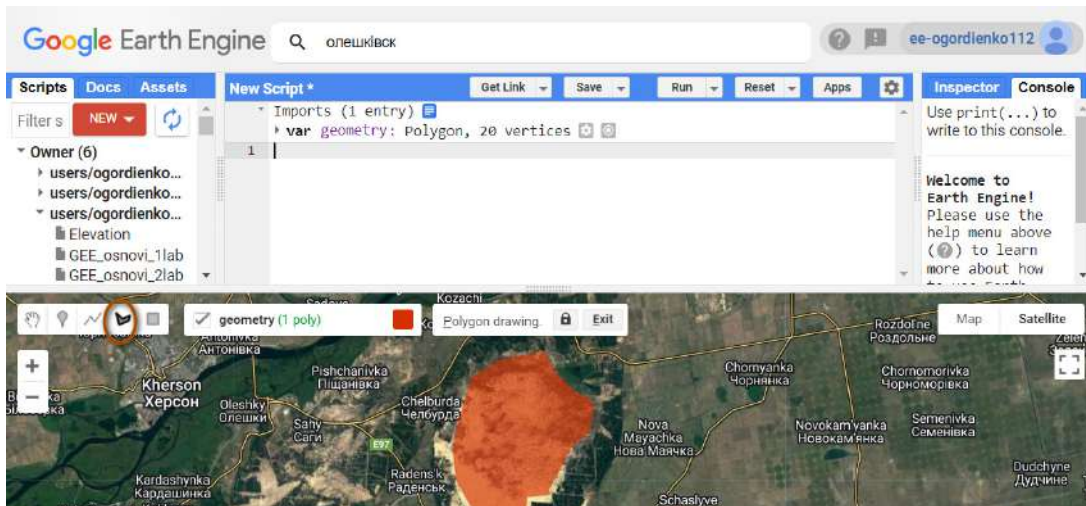


Рис. 81. Обмальована територія цікавості

Після цього в вкладці **Imports** (Імпорт) з'явиться нова змінна, яка називається **geometry**, з типом «**polygon** (полігон)».

2. Підвантажити датасет CHIRPS для дослідження розподілу опадів.

Тепер необхідно знайти датасет, який ми будемо використовувати для розрахунку опадів у пустелі. Скористайтеся пошуком за тегом CHIRPS.

Виберіть **CHIRPS Pentad: Climate Hazards Group InfraRed Precipitation With Station Data (Version 2.0 Final)** (Група кліматичних небезпек в інфрачервоному спектрі, опади з даними зі станцій) і натисніть на нього. Цей датасет – це 30-річний набір даних про опади. CHIRPS охоплює супутникові зображення з роздільною здатністю 0,05° з даними стаціонарних пунктів, щоб створити часові ряди для опадів, аналізу тенденцій та моніторингу сезонної посухи.

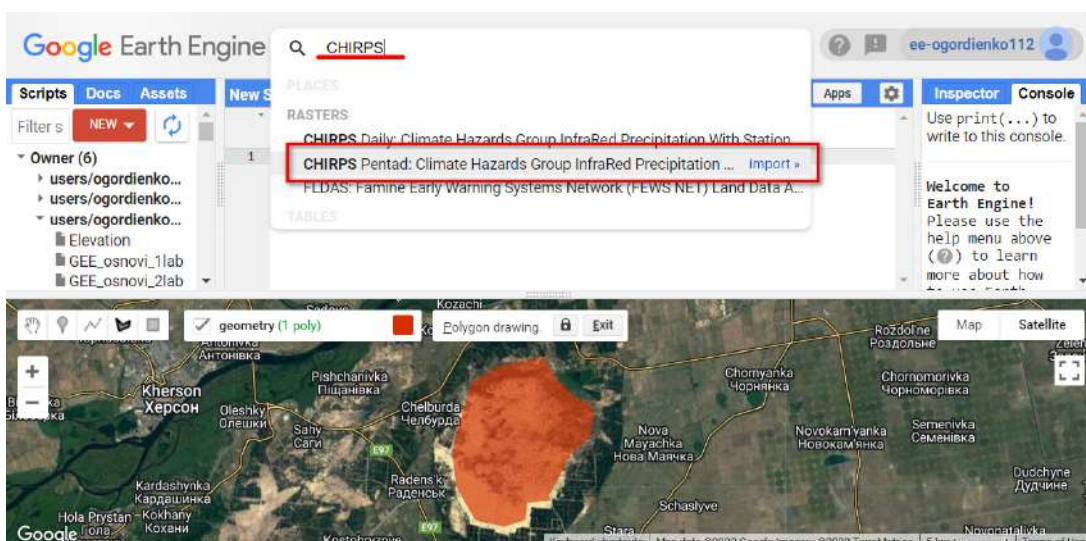



Рис. 82. Пошук датасету з кліматом

У цій практичній роботі ми скористаємося ще одним способом підвантаження датасетів до редактора коду, тому у відкритому вікні скопіюйте шлях до датасету `ee.ImageCollection("UCSB-CHG/CHIRPS/PENTAD")`, натиснувши .

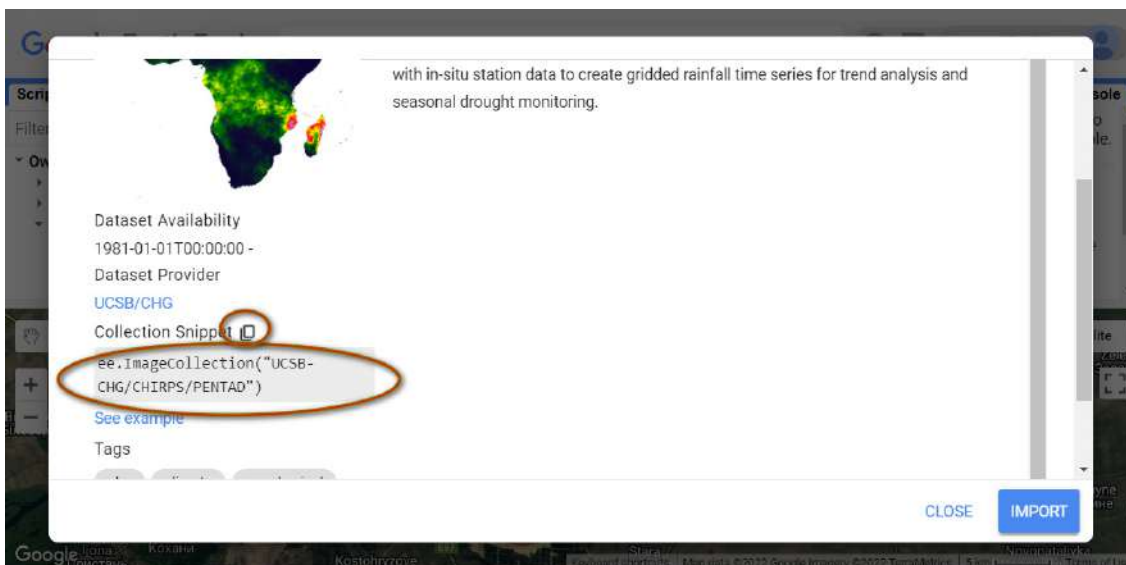




Рис. 83. Копіювання датасету до робочого простору

 **Важливо!** Не натискайте . Цього разу ми скористаємося іншим способом відвантаження даних.

Далі вам необхідно створити нову змінну **var** з ім'ям **chirps** та скопійованим значенням `ee.ImageCollection("UCSB-CHG/CHIRPS/PENTAD")`. Ця змінна тепер буде відображати датасет CHIRPS.

<pre>var chirps = ee.ImageCollection("UCSB-CHG/CHIRPS/PENTAD")</pre>	<p><code>ee.ImageCollection</code> додає джерело растрових даних</p>
--	--

Це ще один спосіб підвантажувати дані з бази даних GEE в *Code Editor* (Редактор коду).

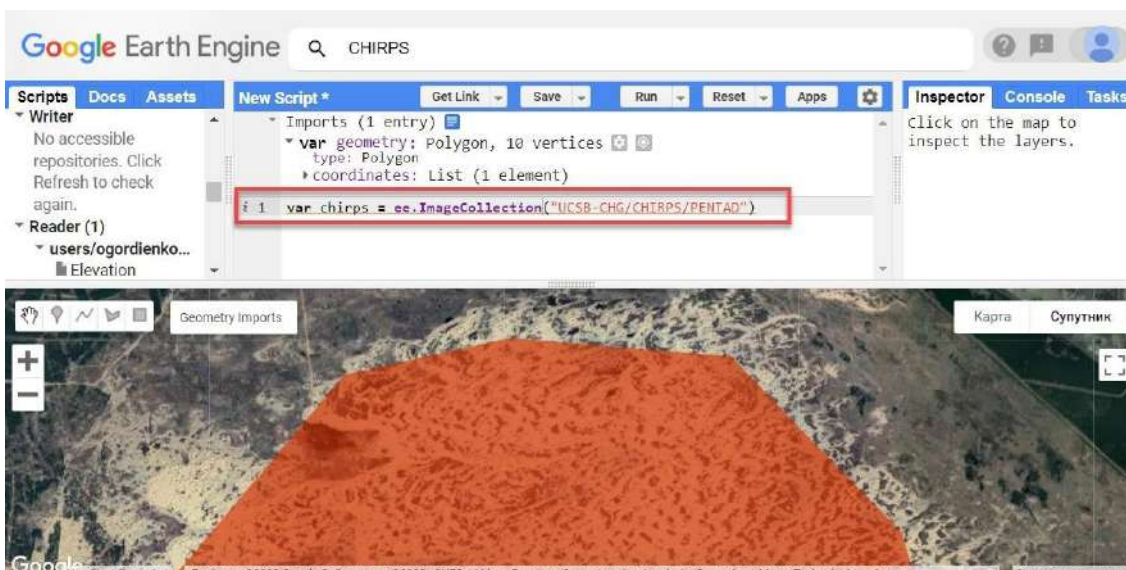


Рис. 84. Зміна назви датасету

Тепер цей датасет доступний за ім'ям **chirps**.

3. Вибрати часовий період з 2010 р. по 2020 р. для аналізу даних із датасету CHIRPS. Наступні рядки коду створять дві змінні для початкового року і для кінцевого року, які потім будуть переведені в числові значення з місяцем та днем. Далі створюємо список зі значеннями. Якщо вивести його в **Console** (Консоль), можна побачити, які роки увійшли до змінної. Скопіюйте наведений нижче код і натисніть **Run** (Запуск скрипту).

<pre>var startyear = 2010; var endyear = 2020; var startdate = ee.Date.fromYMD(startyear,1,1); var enddate = ee.Date.fromYMD(endyear + 1,1,1); var years = ee.List.sequence(startyear, endyear); print(years);</pre>	<p>startyear початкова дата; endyear кінцева дата; startdate переводиться у формат початкової дати і виводиться значення днів та місяців; enddate переводиться у формат кінцевої дати і виводиться значення днів та місяців; years створює список з датами; Виводимо результат у консоль за всі роки</p>
--	--

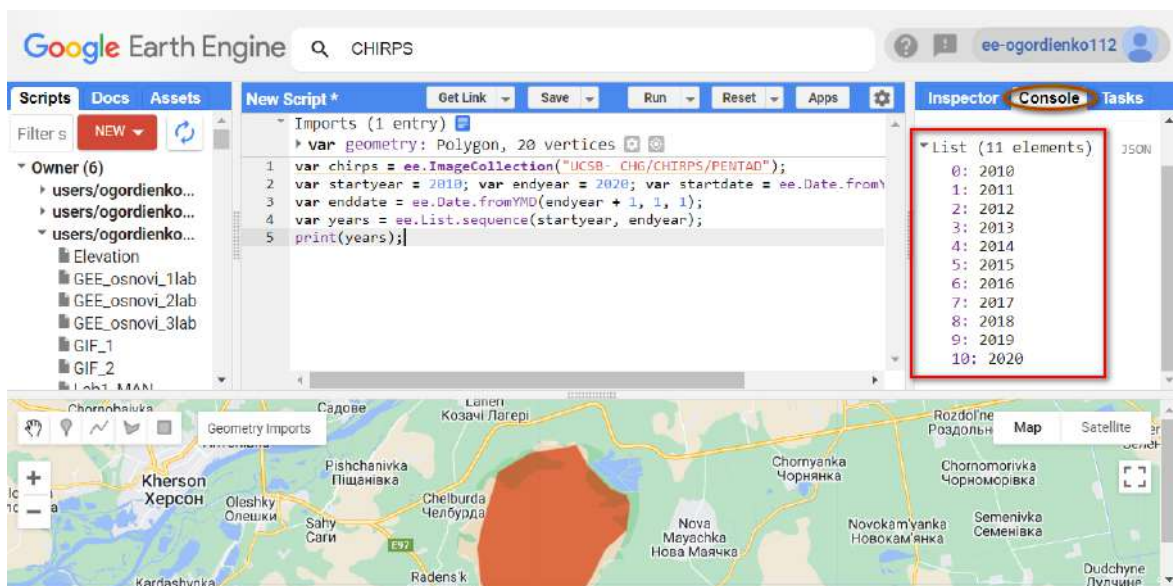


Рис. 85. Виведення даних у консоль

Після цього з'явиться список зі значеннями років. Наступним кроком буде створення функції, яка відфільтрує колекцію за заданими датами та списком, тобто порахується сума значень опадів для кожного року, що є в списку. Ця функція бере інформацію за допомогою `ee.ImageCollection.fromImages` і підвантажує всі значення, які відповідають заданим параметрам. У нашому випадку це значення дат, тобто значення року, яке ми створили в попередньому кроці.

У цій функції `map` одразу додає нове значення до колекції, у нашому випадку – річну суму опадів.

Для побудови графіків великих часових проміжків використовується `.set('system:time_start')`. Це значення – мітка часу Earth Engine у мілісекундах з епохи UNIX. Мітка часу встановлюється на номінальний час отримання зображення для окремих сцен.

<pre>var annualPrecip = ee.ImageCollection.fromImages(years.map(function (year) { var annual = chirps .filter(ee.Filter.calendarRange(year, year, 'year')) .sum(); return annual .set('year', year) .set('system:time_start', ee.Date.fromYMD(year, 1, 1)); }));</pre>	<p>annualPrecip підвантажує колекцію зображень, що відповідають наступним параметрам; До years використовується функція, що приєднує значення за роками до колекції chirps та називає значення 'year'; .sum(); рахує суму у вибраному діапазоні;</p> <p>Функція підвантажує значення для року 'system:time_start', використовується для коректного відображення графіків за великий діапазон дат;</p>
--	---

4. Побудувати графік для розподілу опадів на Олешківських пісках.

Побудова графіків у GEE відбувається за допомогою команди **Chart** та параметрів відображення графіка. Тут використовується колекція зображень `annualPrecip`, до якої щойно були додані суми значення за рік. Потім відбувається фільтрація за геометрією. Знову використовується мітка часу `'system:time_start'` і задаються параметри відображення графіка.

Скористайтеся кодом нижче і натисніть **Run** (Запуск скрипту).

<pre>var chart = ui.Chart.image .series({ imageCollection: annualPrecip, region: geometry, reducer: ee.Reducer.sum(), scale: 500, xProperty: 'system:time_start' }) .setSeriesNames(['precipitation']) .setOptions({ title: 'Sum Precipitation (mm)', hAxis: {title: 'Time', titleTextStyle: {italic: false, bold: true}}, vAxis: { title: 'Precipitation (mm)', titleTextStyle: {italic: false, bold: true} }, lineWidth: 5, colors: ['F08080'], curveType: 'function' }); print(chart);</pre>	<p><code>ui.Chart.image.series</code> використовується для створення графіків, де</p> <p><code>annualPrecip</code> колекція зображень, що буде відображатися, за <code>geometry</code> відбувається фільтрація колекції з масштабом 500 та часовою позначкою <code>'system:time_start'</code></p> <p>назва геометрії на графіку <code>'precipitation'</code> <code>setOptions</code> параметри візуалізації графіка <code>title</code> назва <code>title</code>, що відноситься до <code>hAxis</code> назва осі X <code>titleTextStyle</code> стиль тексту</p> <p><code>title</code>, що відноситься до <code>vAxis</code> назва осі Y <code>titleTextStyle</code> стиль тексту</p> <p><code>lineWidth</code> вага лінії на графіку <code>colors</code> колір графіка <code>curveType</code> тип лінії в графіку</p> <p><code>print(chart)</code>; результат у консоль</p>
---	--

? Завдання для перевірки

Змінити англійські назви підписів у графіку кількості опадів на українські

Для виконання завдання зі зміни назв необхідно замінити частини / назви коду, виділені червоним:

<code>.setSeriesNames(['precipitation'])</code>	<code>'Опади'</code>
<code>title: 'Sum Precipitation (mm)',</code>	<code>'Розподіл загальної кількості опадів (мм)',</code>
<code>hAxis: {title: 'Time', titleTextStyle:</code>	<code>'Роки'</code>
<code>title: 'Precipitation (mm)',</code>	<code>'Сума річних опадів (мм)'</code>

Після запуску цієї частини коду ви побачите значення з даних `"precipitation"` (опади) в **Console** (Консоль) у вигляді графіка за заданими датами.

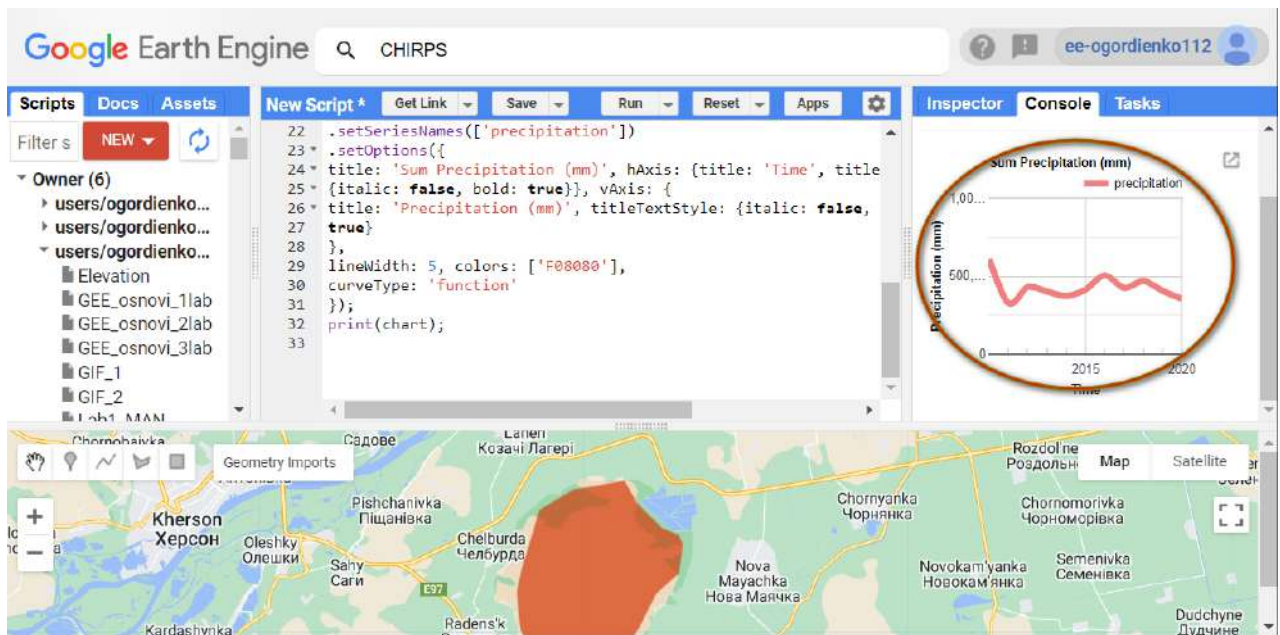


Рис. 86. Виведення графіка в консоль

5. Підвантажити датасет GLDAS для виміру вологості ґрунту на глибині.

Це можна зробити за допомогою датасету GLDAS. Для цього знову в рядку пошуку введіть GLDAS і виберіть версію датасету 2.1, тобто **GLDAS-2.1: Global Land Data Assimilation System** (Глобальна система асиміляції земельних даних). Цей датасет визначає різноманітні показники вологості, температури тощо за допомогою супутникових і наземних спостережень, використовуючи передові методи моделювання даних. Він розроблений Американським космічним агентством NASA у співпраці з NOAA – *National Oceanic and Atmospheric Administration* (Національне управління океанічних і атмосферних досліджень).

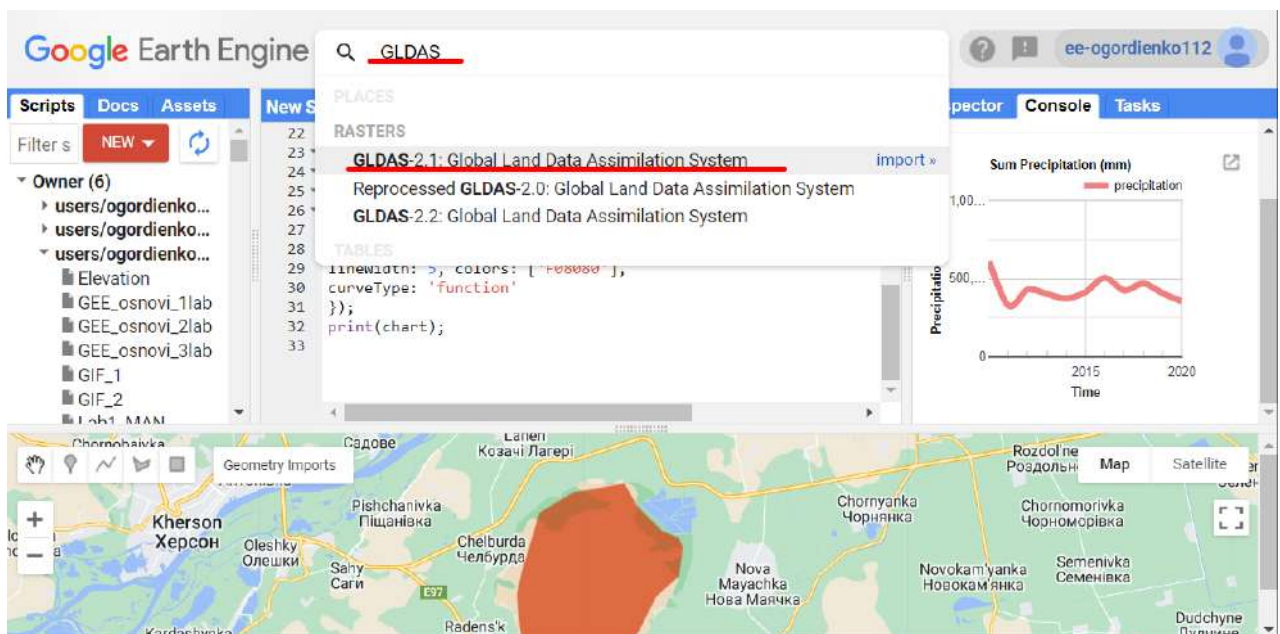


Рис. 87. Завантаження іншого датасету до робочого простору

Знову скопіюйте шлях до датасету GLDAS.

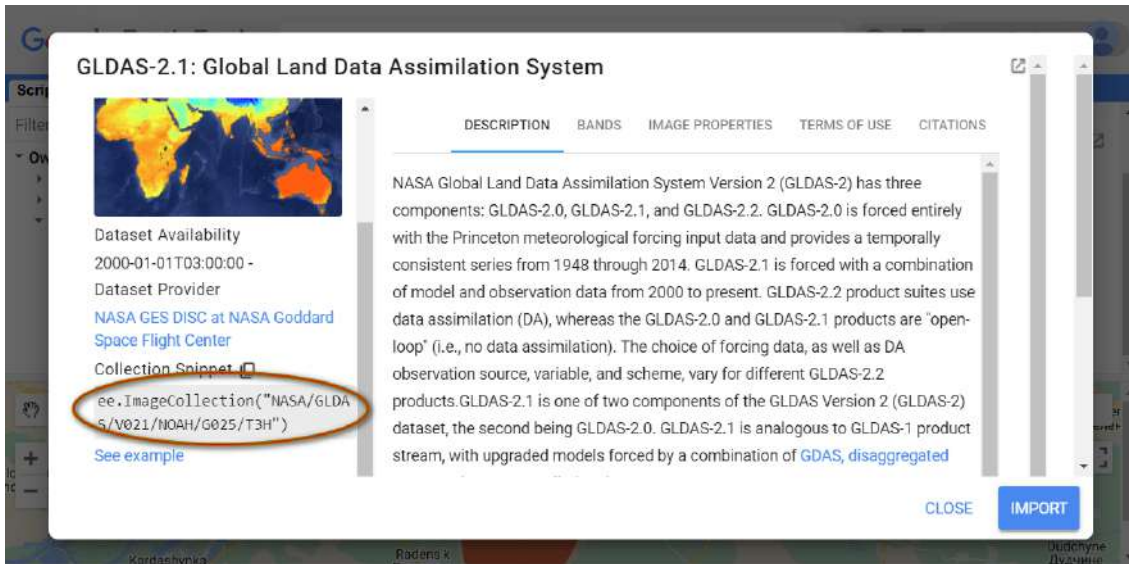


Рис. 88. Імпорт датасету до робочого простору

А тепер створіть іншу змінну з назвою *“gldas”* і відфільтруйте її за трьома показниками вологості ґрунту.

<pre>var gldas = ee.ImageCollection('NASA/GLDAS/V021/NOAH/G025/T3H') .select(['SoilMoi0_10cm_inst','SoilMoi10_40cm_inst','SoilMoi40_100cm_inst']);</pre>	<p>змінна з назвою <i>gldas</i> звертається до датасету <i>GLDAS</i></p> <p>фільтрує датасет за трьома параметрами вологості на трьох різних глибинах</p>
--	---

6. Вибрати часовий період із 2010 р. по 2020 р. для аналізу даних із датасету *GLDAS*. Скопіюйте наведений нижче код і натисніть **Run** (Запуск скрипту).

<pre>var startyear = 2010; var endyear = 2020; var startdate = ee.Date.fromYMD(startyear,1, 1); var enddate = ee.Date.fromYMD(endyear + 1, 1, 1); var years = ee.List.sequence(startyear, endyear); print(years);</pre>	<p><i>startyear</i> початкова дата</p> <p><i>endyear</i> кінцева дата</p> <p><i>startdate</i> переводиться у формат початкові дати і виводиться значення днів та місяців</p> <p><i>enddate</i> переводиться у формат кінцеві дати і виводиться значення днів та місяців</p> <p><i>years</i> створює список з датами</p> <p>Виводимо результат у консоль за всі роки</p>
---	---

Необхідно створити функцію, яка звертається до відфільтрованого датасету й застосовує до нього параметри дат, які ми створювали раніше.

<pre>var moisture = ee.ImageCollection.fromImages(years.map(function (year) { var annual = gldas .filter(ee.Filter.calendarRange(year, year, 'year')) .sum(); return annual .set('year', year) .set('system:time_start', ee.Date.fromYMD(year, 1, 1)); }));</pre>	<p><i>moisture</i> підвантажує колекцію зображень, що відповідають наступним параметрам до <i>years</i> використовується функція, що приєднує значення опадів за роками до колекції <i>chirps</i> та називає значення <i>'year'</i></p> <p><i>.sum()</i>; рахує суму значень у діапазоні.</p> <p>Функція підвантажує значення для року <i>'system:time_start'</i>, використовується для коректного відображення графіків за великий діапазон дат</p>
--	--

7. Побудувати графік для вологості ґрунту на Олешківських пісках.

Тепер усю цю інформацію можна вивести в **Console** (Консоль) за допомогою графіків. Для цього скористайтеся кодом:

<pre>var chart = ui.Chart.image .series({ imageCollection: moisture, region: geometry, scale: 500, xProperty: 'system:time_start' }) .setSeriesNames(['SoilMoi10_40cm_ inst','SoilMoi40_100cm_inst','SoilMoi0_10cm_ inst']) .setOptions({ title: 'Sum moisture', hAxis: {title: 'Time', titleTextStyle: {italic: false, bold: true}}, vAxis: { title: 'moisture (kg/m^2)', titleTextStyle: {italic: false, bold: true} }, lineWidth: 5, colors: ['e37d05', '1d6b99', '003b00'], curveType: 'function' }); print(chart);</pre>	<p><code>ui.Chart.image.series</code> використовується для створення графіків, де</p> <p><code>annualPrecip</code> колекція зображень, яка буде відображатися, за <code>geometry</code> відбувається фільтрація колекції з масштабом 500 та часовою позначкою <code>'system:time_start'</code></p> <p>Назви геометрії на графіку <code>'SoilMoi10_40cm_inst','SoilMoi40_100cm_inst','SoilMoi0_10cm_inst'</code></p> <p><code>setOptions</code> параметри візуалізації графіка <code>title</code> назва <code>title</code>, що відноситься до <code>hAxis</code> назва осі X <code>titleTextStyle</code> стиль тексту <code>title</code>, що відноситься до <code>vAxis</code> назва осі Y <code>titleTextStyle</code> стиль тексту</p> <p><code>lineWidth</code> вага лінії на графіку <code>colors</code> колір графіка <code>curveType</code> тип графіка</p> <p><code>print(chart);</code> результат у консоль</p>
---	---

Натисніть **Run** (Запуск скрипту).

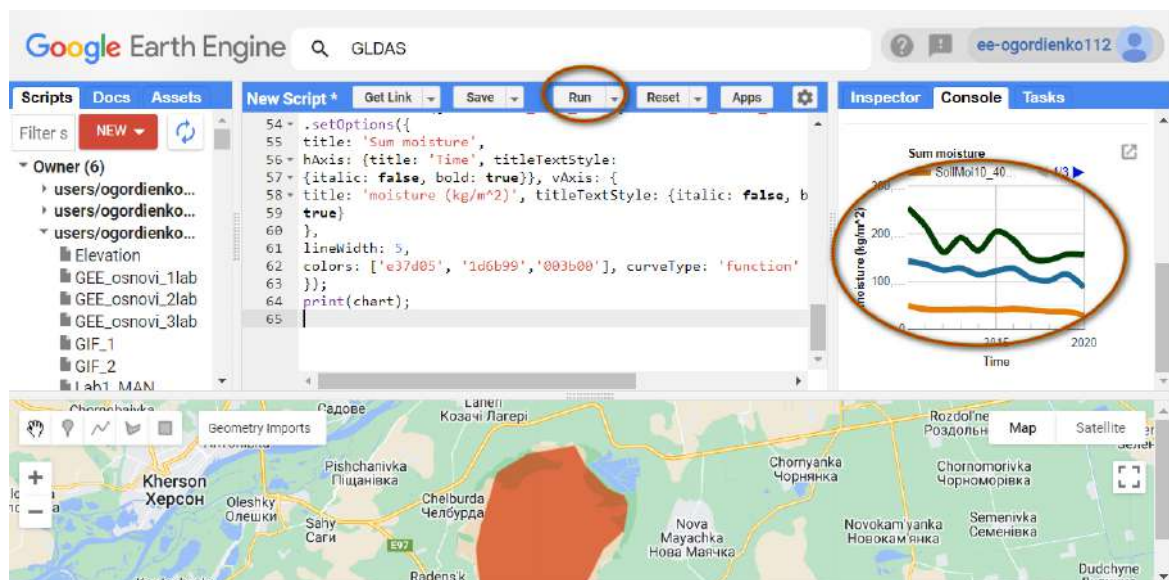


Рис. 89. Виведення графіка в консоль

? Завдання для перевірки

Змінити англійські назви підписів у графіку вологості ґрунту на різних глибинах на українськомовні

Для виконання завдання зі зміни назв необхідно замінити частини / назви коду, виділені червоним:

<code>.setSeriesNames(['SoilMoi40_100cm_inst', SoilMoi10_40cm_inst','SoilMoi0_10cm_inst'])</code>	'Вологість_ґрунту40_100см','Вологість_ґрунту10_40см','Вологість_ґрунту0_10см'
<code>title: 'Sum moisture',</code>	'Розподіл вологості'
<code>hAxis: {title: 'Time',</code>	'Роки'
<code>title: 'moisture (kg/m^2)',</code>	'Вологість (кг/м^2)'

Так ми отримаємо два показники для пустелі ОП: показник вологості ґрунту на різних глибинах і показник кількості опадів.

8. Зберегти отримані графіки опадів і вологості у форматі PNG.

У **Console** (Консоль) з'явилися два графіки, які потрібно експортувати у форматі PNG. Для цього натисніть на відмічену кнопку, графік відкриється в новому вікні.

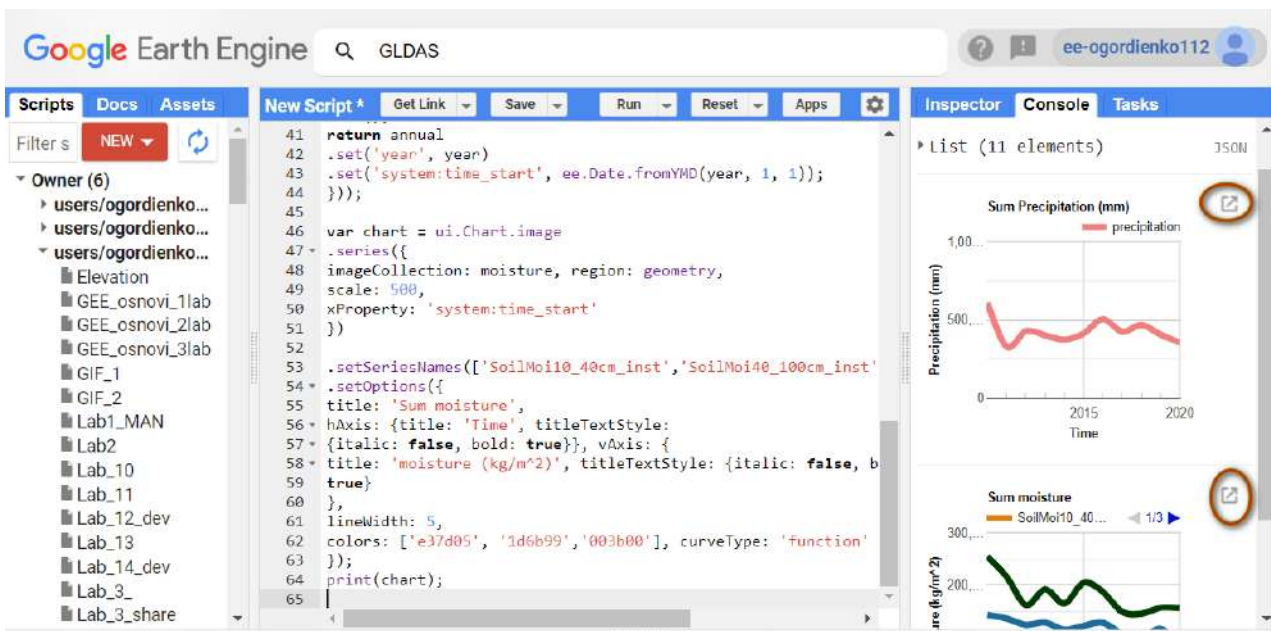


Рис. 90. Можливість збереження в різних форматах

Тут є можливість зберігати результуючі графіки в трьох форматах: CSV, SVG, PNG. Натисніть **Download PNG** (Завантажити PNG), щоб завантажити графік у вигляді кольорового растрового зображення. Якщо ж вам потрібні дані в табличному вигляді, зберігайте результати у форматі CSV.

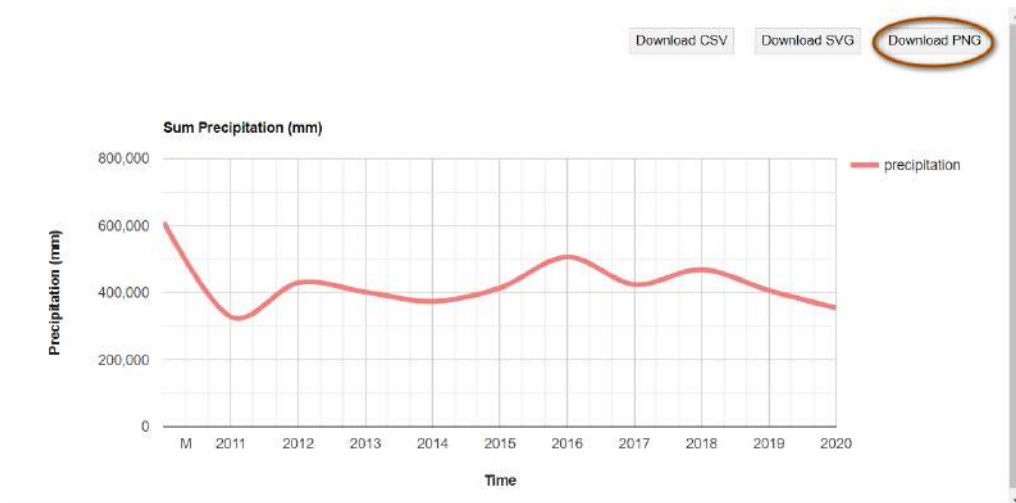


Рис. 91. Вивантаження даних на локальний комп'ютер

? Завдання для перевірки

Скопіюйте собі на комп'ютер два графіки ОП у форматі PNG. Англomовні підписи на графіках мають бути українською мовою

9. Створити новий вектор території дослідження для пустелі Сахара.

Тепер можна порівняти ОП зі справжньою пустелею Сахара. Для цього видаліть старі параметри геометрії і створіть нові в пустелі Сахара.

✓ **Важливо!** Не створюйте надто великий вектор, бо швидкість розрахунку може затягнутися.

Щоби простіше було знайти Сахару на півночі африканського континенту, перемкніть відображення мапи на супутник, наблизьтеся до країни Лівія, наприклад, і за допомогою **Draw a rectangle** (Накреслити прямокутник) з панелі інструментів **Geometry instruments** (Інструменти геометрії) обмалюйте невелику ділянку пустелі Сахара.

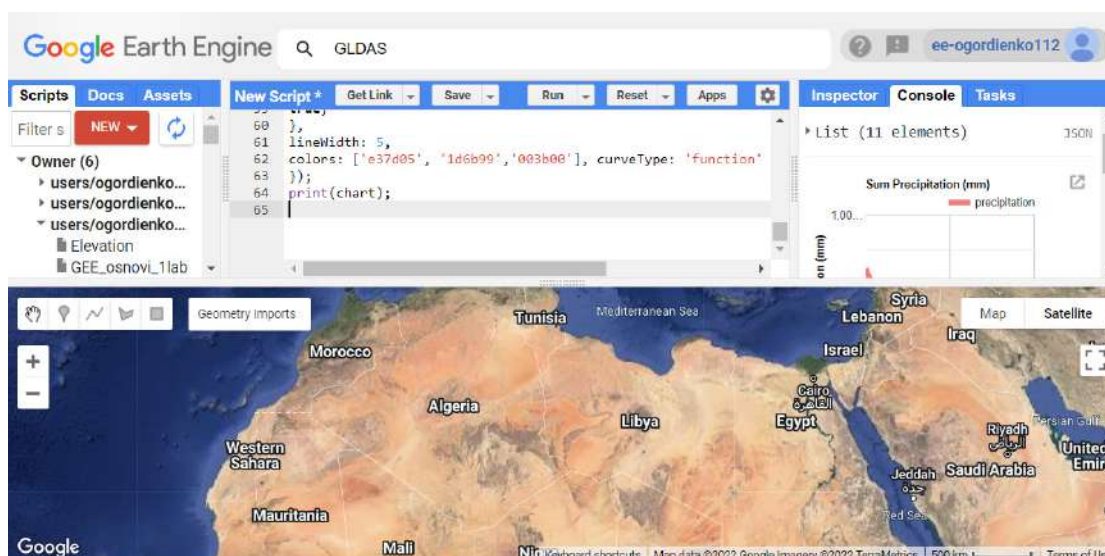


Рис. 92. Дані для Сахари

10. Побудувати аналогічні графіки опадів та вологості ґрунту для пустелі Сахара.

Скористайтеся вже створеним кодом і натисніть **Run** (Запуск скрипту), щоб отримати графіки для Сахари.

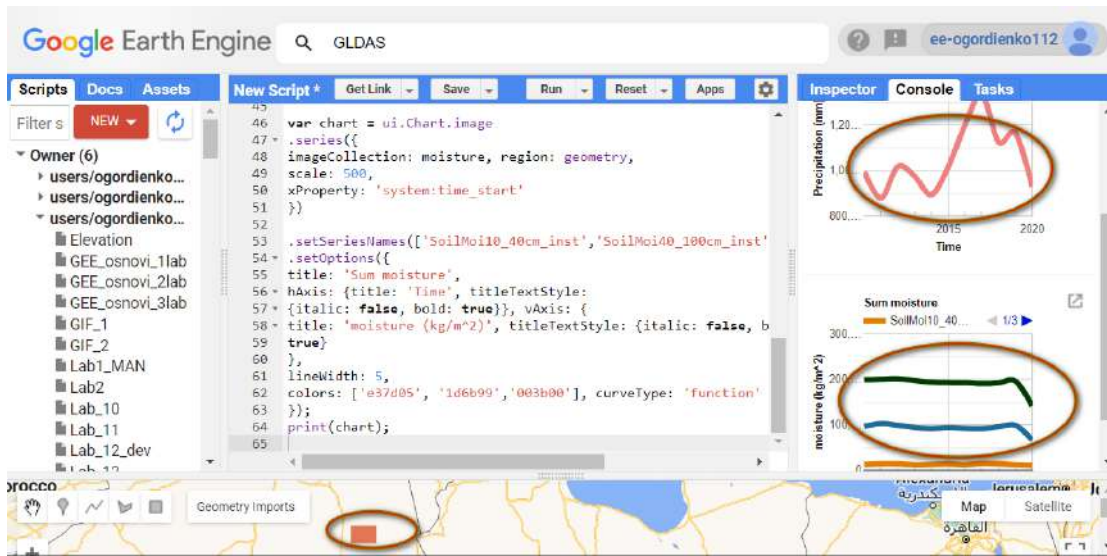


Рис. 93. Зображення графіків для Сахари

? Завдання для перевірки

Скопіюйте собі на комп'ютер два графіки – вологості й опадів – частини пустелі Сахара

Порівняйте завантажені графіки для Сахари й Олешків, напишіть спільні і відмінні ознаки розподілу опадів та вологості.

Скільки, на вашу думку, років потрібно, щоб Олешківські піски стали пустелею, зважаючи на сучасні кліматичні зміни?

За наведеним нижче QR-кодом ви можете переглянути відео виконання практичної роботи.



Рис. 94. Відео виконання практичної роботи

Готовий код можна відкрити за лінком:

https://code.earthengine.google.com/?scriptPath=users%2Fgis_rs%2Fcourse_MAN%3ALab_4

Аналіз рельєфу із застосуванням моделювання підтоплених територій України внаслідок підняття рівня океану. Створення профілю і відмивки рельєфу г. Говерла

Ситуація

Зміни клімату призводять до того, що рівень Світового океану підіймається швидше, ніж передбачалося в найпесимістичніших припущеннях учених. Це може спричинити значно швидше затоплення прибережних міст, аніж прогнозували раніше¹. У цій практичній роботі ми проведемо дослідження, яке в цілому покаже, які саме території України будуть затоплені внаслідок підвищення рівня Світового океану.

Слід зазначити, що цей дослід доволі простий з технічного погляду і не враховує особливості території, а саме: тиск води, який спричиняють річки через свою силу течії на моря, різницю висоти Чорного та Азовського морів відносно висоти Світового океану тощо.

Простіше кажучи, запропонована нами модель працює так: до Цифрової моделі рельєфу (ЦМР) додається значення в метрах й отримується результат щодо підтоплених територій.

Цифрова модель рельєфу – ЦМР (digital elevation model – DEM) – цифрове представлення рельєфу земної поверхні, створене на основі даних про рельєф та топології місцевості.

SRTM (Shuttle radar topographic mission) – міжнародна місія з отримання даних цифрової моделі рельєфу (ЦМР) Землі.

Також у цій практичній роботі буде використаний новий спосіб завантаження зображень, який дає змогу завантажувати інформацію через покликання. Слід зазначити, що такий спосіб, на відміну від завантаження на Google-диск, має свої обмеження: кількість пікселів не має перевищувати значення 10 000. Цей спосіб зручно використовувати для завантаження невеликих ділянок.

Також буде використана відмивка рельєфу – це спосіб зображення інформації про рельєф за допомогою тіней та освітлення. Джерело освітлення в нашому випадку має значення азимуту 270°, тобто воно перебуває на заході.

Буде завантажена ЦМР з відмивкою рельєфу для г. Петрос та г. Говерла.

Профіль рельєфу – лінія, проведена за сукупністю точок на земній поверхні, яка демонструє загальний вертикальний розріз рельєфу цієї території.

Також буде створено профіль рельєфу для вищезазначеної території, який може працювати на території всієї України.

Завдання

Завантажити датасети державних кордонів країн **LSIB 2017** і цифрову модель рельєфу **NASA SRTM Digital Elevation 30m**. Вирізати зі світової ЦМР фрагмент для території України і побудувати просту модель затоплення земної поверхні внаслідок підвищення рівня Світового океану. Порахувати площу підтоплення. Побудувати профіль рельєфу і створити відмивку рельєфу г. Говерла та г. Петрос.

Алгоритм виконання завдання:

1. Відкрити Code Editor і завантажити датасет **LSIB 2017**.
2. Завантажити датасет **NASA SRTM Digital Elevation 30m**.
3. Виділити кордони України з-поміж інших держав світу в датасеті **LSIB 2017**.
4. Вирізати з усього рельєфу Землі в датасеті **NASA SRTM** фрагмент для території України.
5. Створити мапу підтоплення території України в разі збільшення рівня океану на 50 метрів і розрахувати її площу.

¹ URL: <https://hromadske.ua/posts/riven-svitovogo-okeanu-pidvishuyetsya-shvidshe-za-najpesimistichnishi-pronozhennya>

6. Налаштувати візуальне представлення цифрової моделі рельєфу на територію України.
7. Створити відмивку рельєфу на виділений фрагмент території України, а саме: г. Говерла та г. Петрос.
8. Налаштувати візуальне представлення відмивки рельєфу на г. Говерла та г. Петрос і завантажити її на комп'ютер.
9. Створити лінійний об'єкт геометрії для обрахунку профілю рельєфу.
10. Отримати графік профілю рельєфу г. Говерла та г. Петрос і завантажити його на комп'ютер.

Покрокова інструкція

1. Відкрити Code Editor і завантажити датасет **LSIB 2017**.

Відкрийте **Code Editor** (Редактор коду): <https://code.earthengine.google.com/>. Для початку потрібно завантажити датасет **LSIB 2017: Large Scale International Boundary Polygons, Simplified** (*LSIB 2017: Великомасштабні міжнародні кордони країн у полігонах, спрощені*) з кордонами країн, для того щоб у майбутньому скористатися цим датасетом для вирізання потрібних нам даних.

Наберіть у пошуку тег LSIB і натисніть на **Import** (Імпортувати), щоб датасет з'явився в **Imports** (Імпорт).

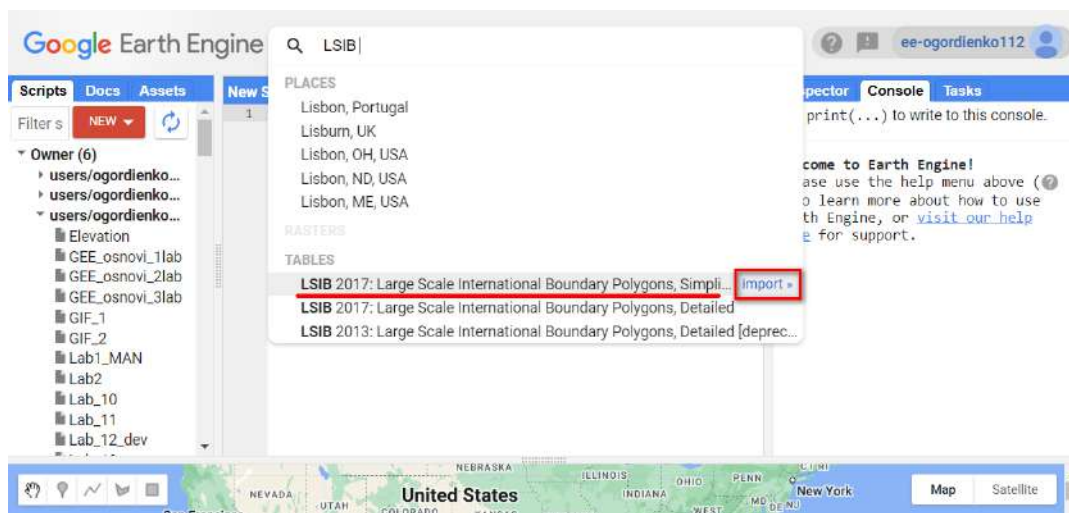


Рис. 95. Імпорт датасету з кордонами країн

Тепер потрібно перейменувати датасет. Задайте йому ім'я "LSIB", натиснувши на назву **table** в **Imports** (Імпорт).

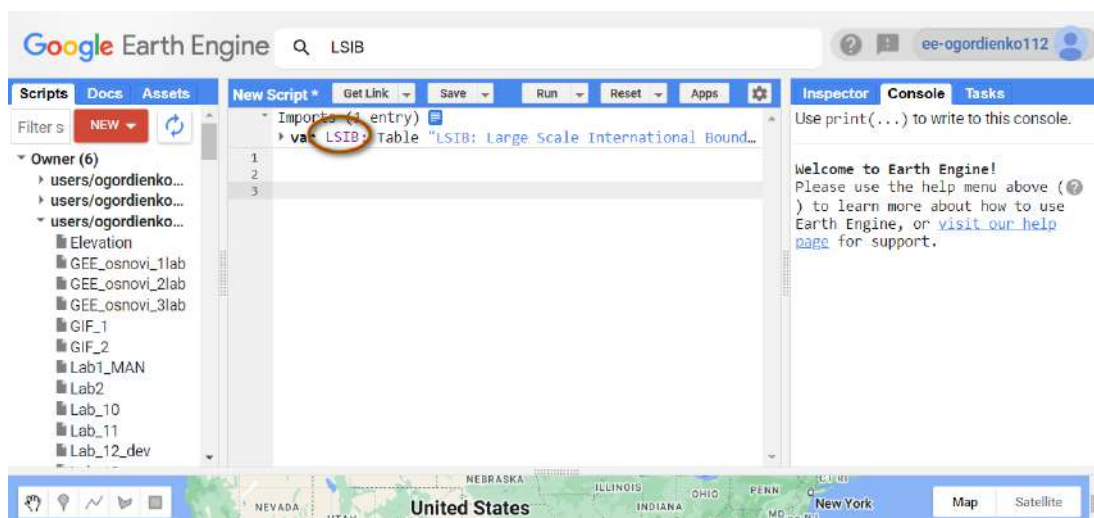


Рис. 96. Зміна назви датасету

2. Завантажити датасет **NASA SRTM Digital Elevation 30m**.

Наступний датасет, який ми будемо експортувати, це **NASA SRTM Digital Elevation 30m** (*NASA SRTM Цифрова модель рельєфу 30 м*). У ньому зберігаються дані про рельєф з детальною 30 метрів у пікселі. Для цього в рядку пошуку введіть тег **SRTM** і виберіть потрібний датасет.

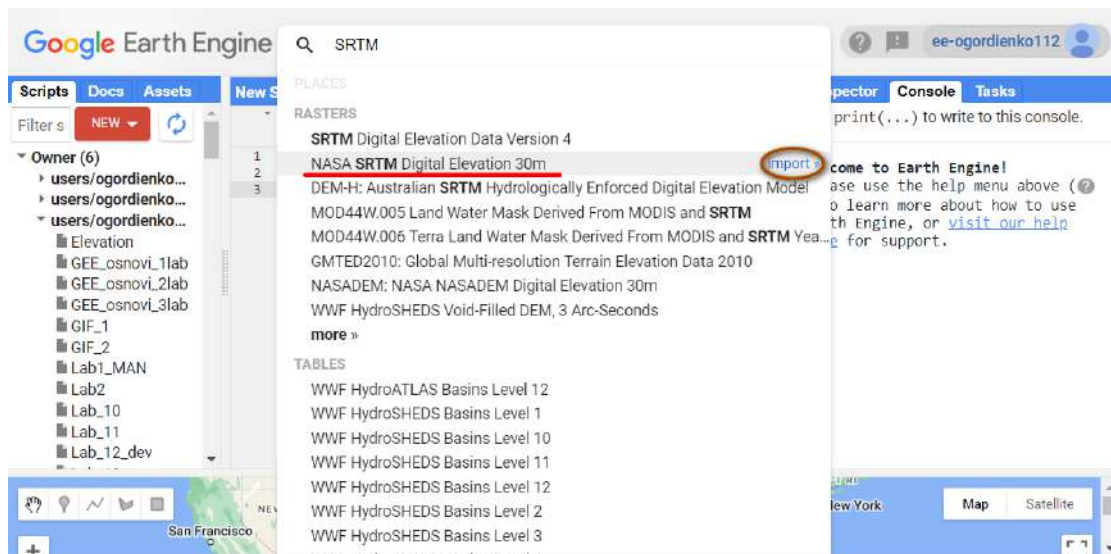


Рис. 97. Імпорт даних поверхні

Здайте цьому датасету назву “**SRTM**”, натиснувши на назву **table** в **Imports** (*Імпорт*).

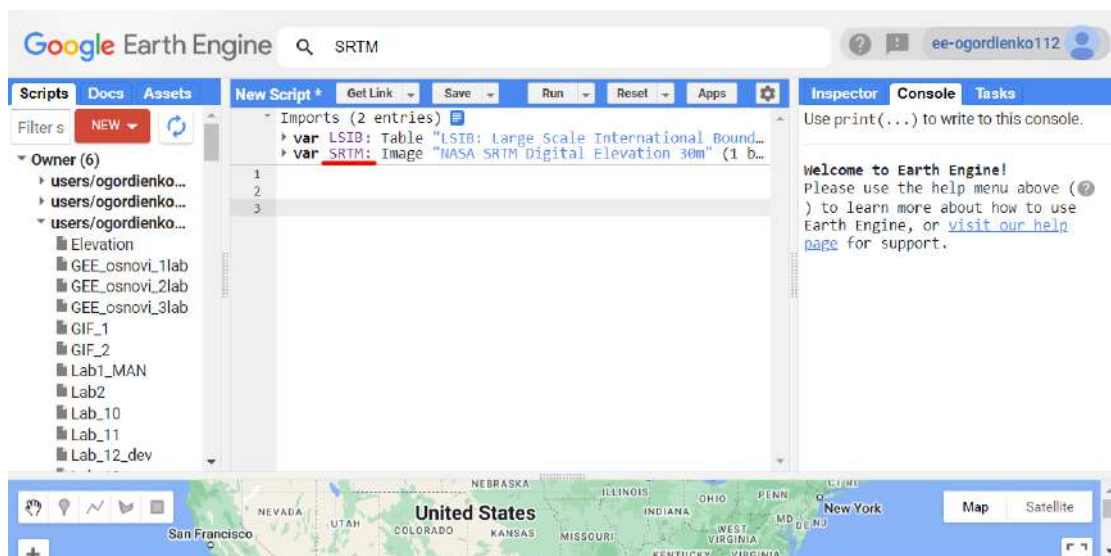


Рис. 98. Зміна назви датасету

3. Виділити кордони України з-поміж інших держав у датасеті **LSIB 2017**.

Наступний крок – це відфільтрувати перший датасет, так щоб вибрати кордони України. Для цього скористайтеся кодом:

```
var Ukraine = LSIB.filter(ee.Filter.eq('country_  
na', 'Ukraine'));
```

Ця змінна звертається до датасету з кордонами країн та вибирає зі списку Україну

4. Вирізати з усього рельєфу Землі в датасеті **NASA SRTM** фрагмент для території України.

Маючи кордон України та інформацію про рельєф, наступним кодом можна вирізати ЦМР, яка перебуває в межах України.

<code>var SRTM_PK = SRTM.clip(Ukraine);</code>	Ця змінна вирізає за межами попередньої змінної дані з ЦМР;
--	---

Наступна змінна буде числова: в ній можна буде змінювати значення так, щоби бачити, які території України стали підтопленими через підвищення рівня океану.

Створимо наступну змінну і виведемо її в **Console** (Консоль).

<code>var change_number = 50;</code> <code>print(change_number)</code>	Змінна, що вказує величину для ЦМР, а значення 50 вказує, на яку кількість метрів піднімається Світовий океан; Результат у консоль
---	---

Натисніть **Run** (Запуск скрипту).

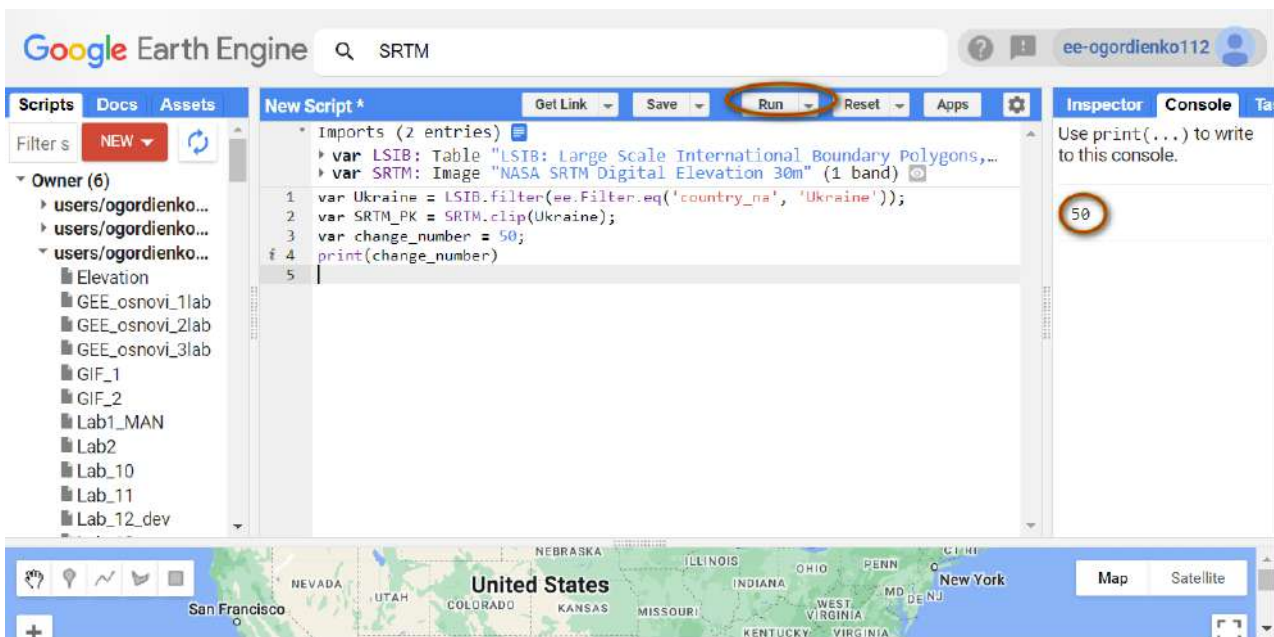


Рис. 99. Створення змінної для обрахунку

5. Створити мапу підтоплення території України в разі збільшення рівня океану на 50 метрів і розрахувати її площу.

Тепер можна створити змінну, яка буде вказувати порогове значення.

<code>var threshold = SRTM_PK.gte(change_number);</code>	Значення <code>gte</code> (greater than equal), тобто більше ніж рівне. <code>gte</code> фільтрує метадані, що перевищують або дорівнюють вказаному значенню, тобто <code>change_number</code> ;
--	--

Додамо цей шар на мапу і відцентруємося по кордону України.

<code>Map.centerObject(Ukraine,5);</code> <code>Map.addLayer(threshold, {palette:['06768F', '8F4809']}, 'threshold');</code>	Центрування мапи за параметрами; Шар на мапу з параметрами візуалізації '06768F', '8F4809' та назвою 'threshold';
---	--

Натисніть **Run** (Запуск скрипту).

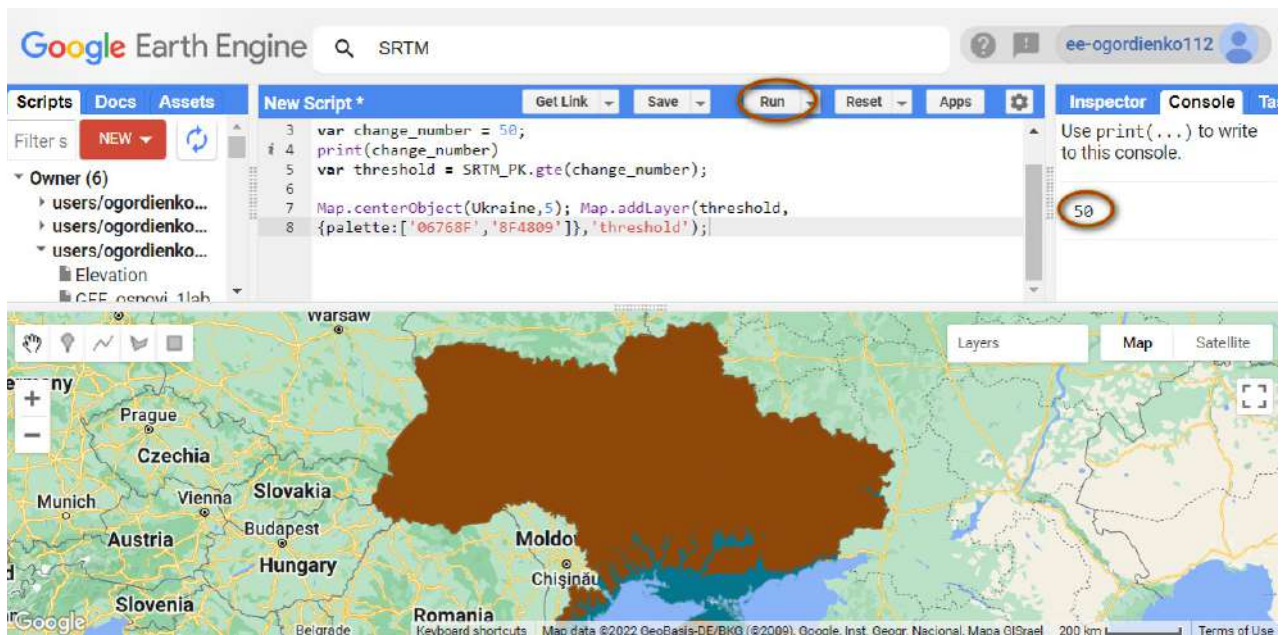


Рис. 100. Додавання даних поверхні на мапу

Приблизно такі території зазнають підтоплення через підвищення рівня океану на 50 метрів. Синім кольором показані саме ті території, які зникнуть під водою. Коричневим кольором відображаються території суші, які вищі за рівень океану, якщо він підніметься на 50 метрів.

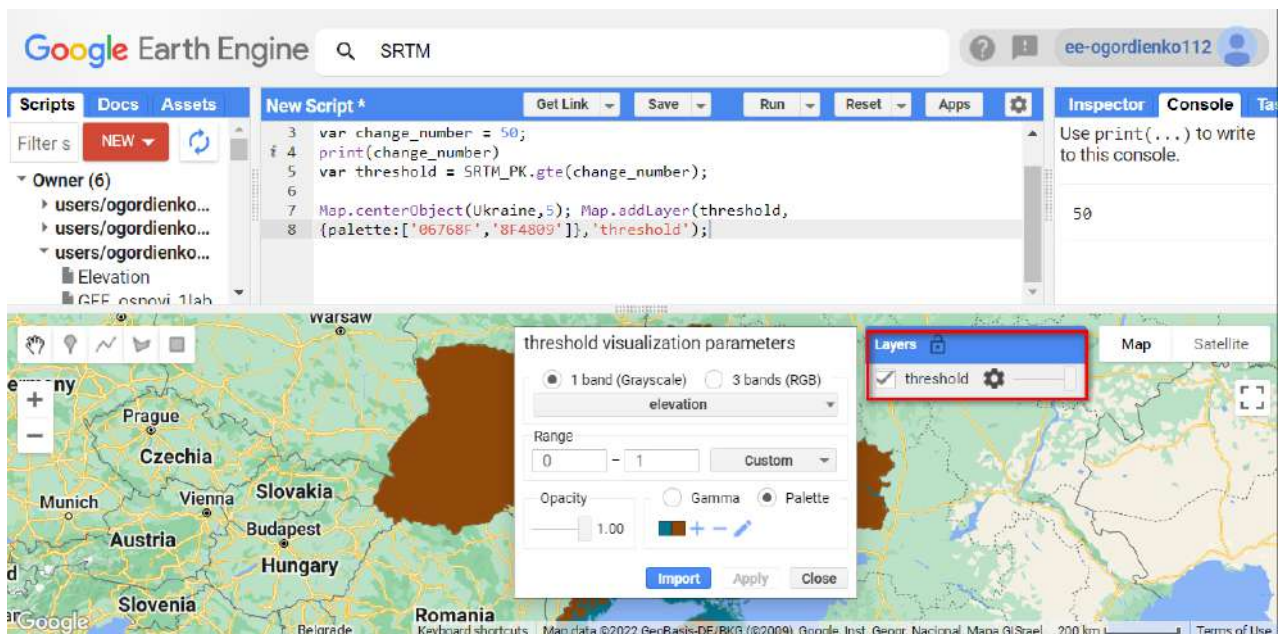


Рис. 101. Зміна даних поверхні

Наступний крок: можна отримати числові значення, тобто значення площі території, яка буде під водою. Для цього необхідно створити маску із шару *threshold* (*noris*), що відображений коричневим кольором. Це шар, який вищий за ЦМР, ніж значення, яке подається в змінній *change_number*.

Для цього потрібно скористатися кодом:

<pre>var mask = threshold.updateMask(threshold);</pre>	<p>Накладання маски на всі значення, що менші, ніж у попередній змінній;</p>
--	--

Для того щоб отримати числові значення площі території України, яка залишається сушею після підтоплення, скористайтеся кодом:

<pre>var pix_area = ee.Image.pixelArea(). addBands(mask.select('elevation')).divide(1e6). reduceRegion({ reducer: ee.Reducer.sum().group(1), geometry: Ukraine, scale: 30, bestEffort: true });</pre>	<p>Розрахунок площі для растрових даних за допомогою <code>ee.Image.pixelArea</code>, по каналу <code>mask</code> зі значенням <code>'elevation'</code> поділений на <code>1 000 000</code>.</p> <p><code>ee.Reducer.sum</code> рахує суму та поміщає її в <code>group 1</code></p> <p><code>Ukraine</code> вказує, де буде проходити обрахунок <code>30</code> розмір пікселя, для цієї ЦМР це <code>30</code> метрів</p> <p><code>bestEffort</code> Застосовує редуктор, тобто з'єднання всіх даних в одні дані, до всіх пікселів у певній області;</p>
---	---

Результат розрахунку можна вивести в **Console** (Консоль).

<pre>print(pix_area,'no Flood area');</pre>	<p>Результат у консоль</p>
---	----------------------------

Натисніть **Run** (Запуск скрипту).

Значення площі суші перебуває в `groups`. Розгорніть його і відкрийте наступний об'єкт з назвою `0`. Тут будуть два об'єкти: `group` і `sum`. Інформація, яку ми шукаємо, міститься в другому об'єкті.

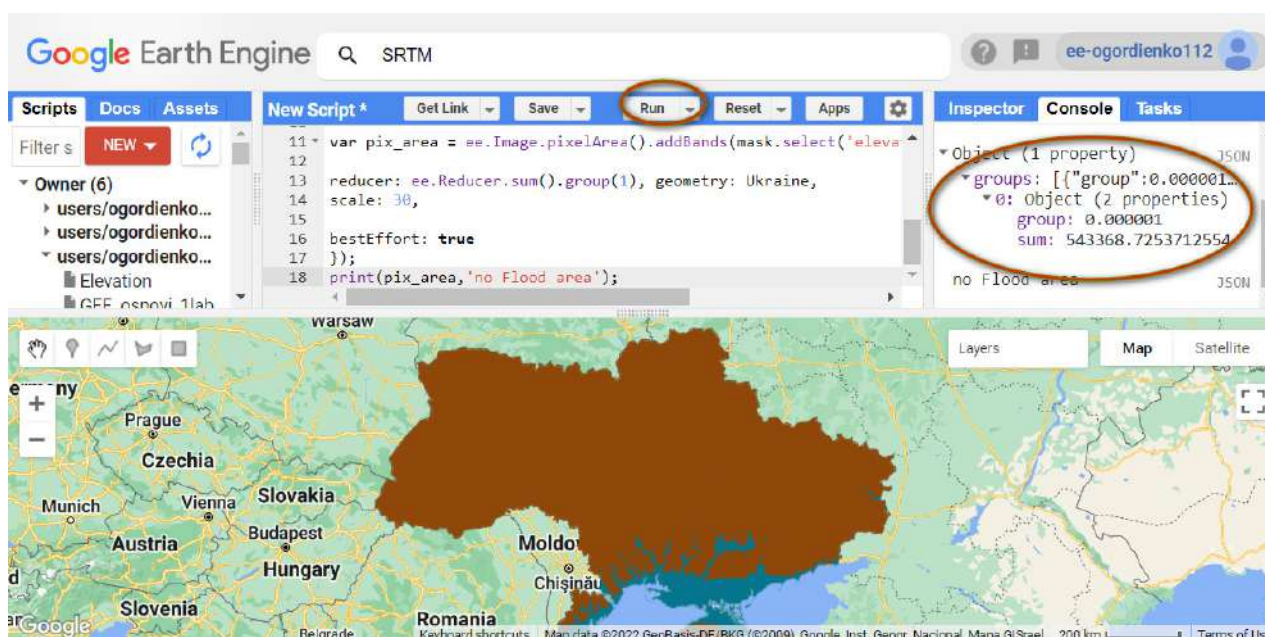


Рис. 102. Обрахунок площі території

Для порівняння масштабу затоплення із загальною площею України потрібно створити змінну, в якій буде зберігатися інформація про площу України. Потім треба вивести це значення в **Console** (Консоль).

<pre>var areaUkraine = ee.Number(603628); print (areaUkraine,'area Ukraine');</pre>	<p>Значення, що виводить у консоль площу України;</p> <p>Результат у консоль;</p>
--	---

Натисніть **Run** (Запуск скрипту).

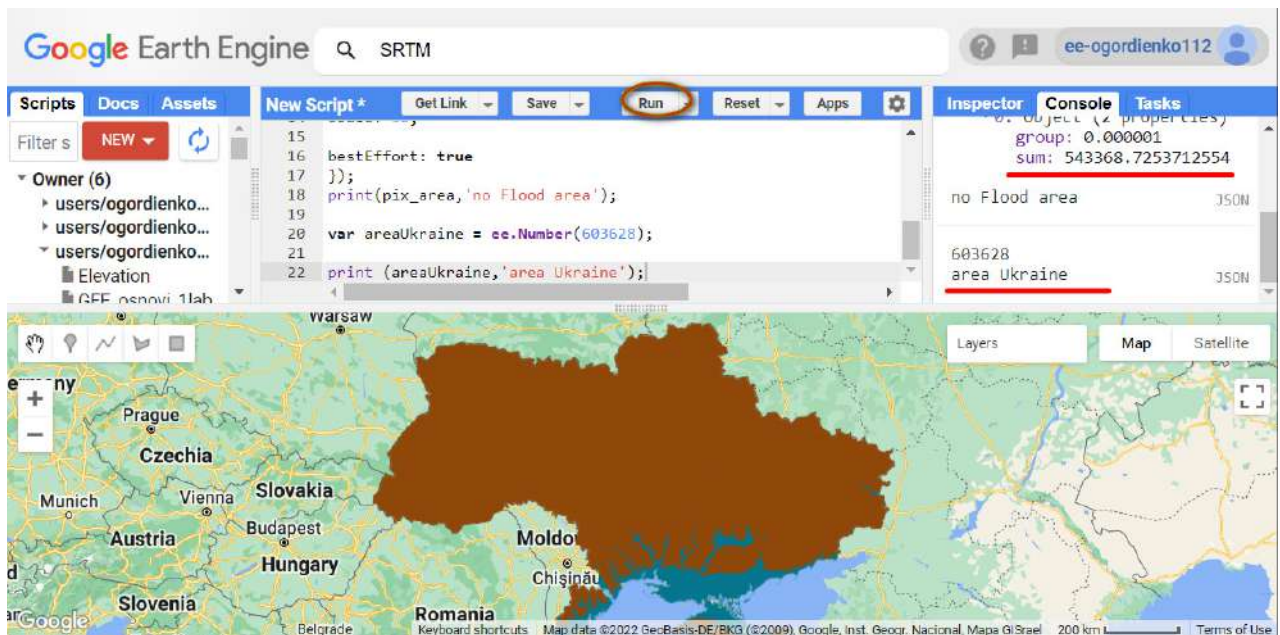


Рис. 103. Змінна для площі України

? Завдання для перевірки

Яка площа України буде затопленою, якщо рівень океану підвищиться на 42 м?

6. Налаштувати візуальне представлення цифрової моделі рельєфу на територію України. GEE дає змогу ділитися скриптами і використовувати їх іншим користувачам. Датасет за адресою: [users/gena/packages/palettes](https://code.earthengine.google.com/users/gena/packages/palettes) містить палітри візуалізації для більш зручного відображення.

Почитати про датасет можна за лінком <https://github.com/gee-community/ee-palettes> або просто за його адресою, яку слід набрати в пошуковому рядку.

Для візуалізації в GEE можна користуватися пакетами сторонніх розробників. Для цього потрібно скористатися кодом:

<pre>var palettes = require('users/gena/packages:palettes'); var palette = palettes.colorbrewer.BrBG[11].reverse(); var visParams = {min: 1, max: 2061, palette: palette};</pre>	<p>Змінна, що звертається до стороннього користувача;</p> <p>Із цього датасету береться <code>colorbrewer.BrBG[11]</code> та візуалізується в обернених кольорах за допомогою <code>reverse()</code>;</p> <p>Параметри візуалізації</p>
--	---

Тепер можна вивести на екран мапу рельєфу з параметрами візуалізації.

<pre>Map.addLayer(SRTM.clip(Ukraine), visParams, 'Elevation')</pre>	<p>Результат на мапу</p>
---	--------------------------

Натисніть **Run** (Запуск скрипту).

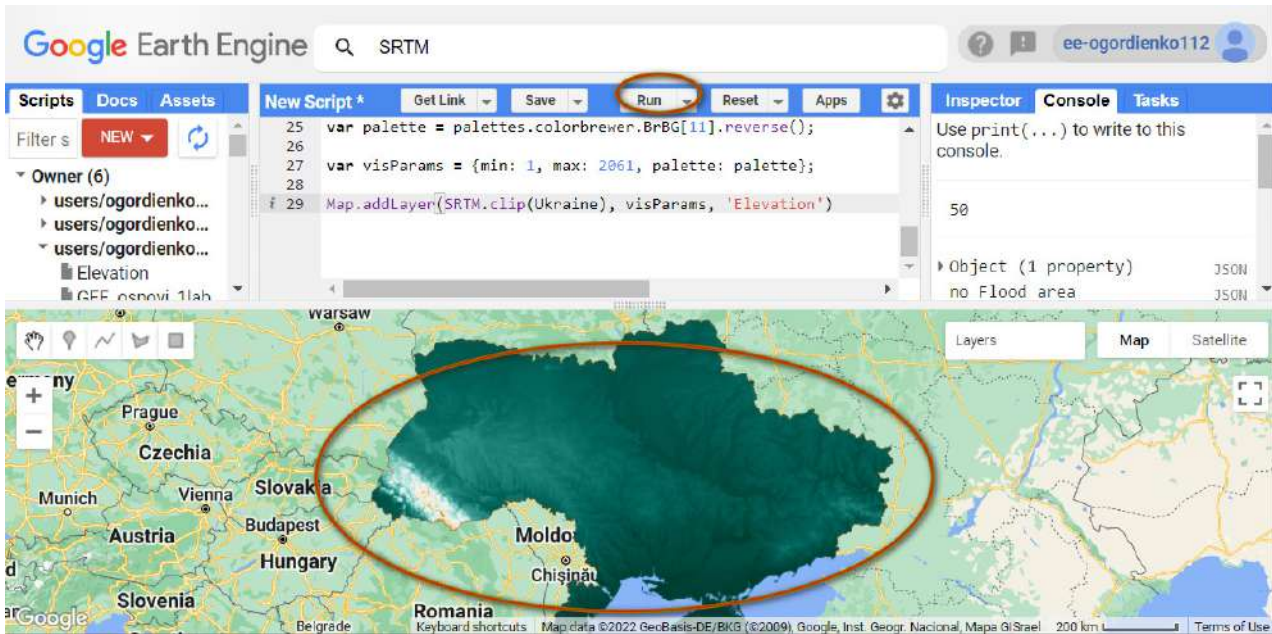


Рис. 104. Зміна відображення поверхні

7. Створити відмивку рельєфу на виділений фрагмент території України, а саме: г. Говерла та г. Петрос.

Наступний крок – побудова відмивки рельєфу на всю територію України. Для цього можна скористатися командою:

```
var shade = ee.Terrain.hillshade(SRTM.  
clip(aoi));
```

Створення відмивки рельєфу

Тепер потрібно вирізати невеликий фрагмент, а саме: ділянку г. Говерла та г. Петрос, створивши рамку території дослідження за допомогою нової геометрії через інструменти *Geometry imports* (Імпорт геометрії), і назвати її *aoi*.

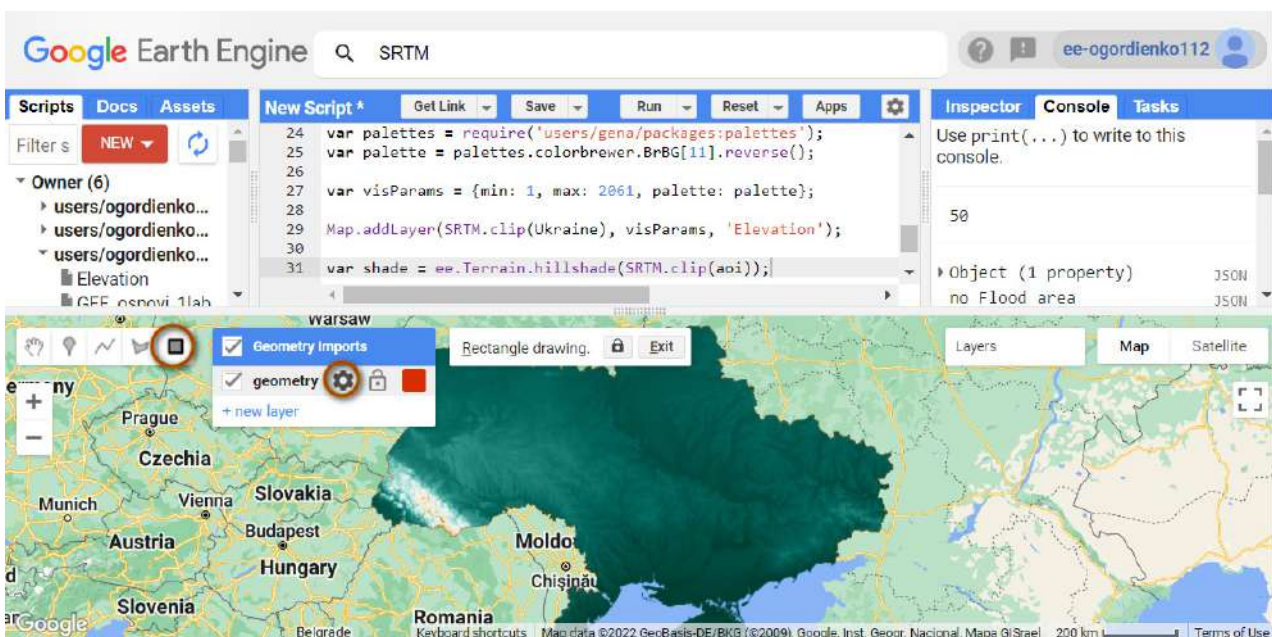




Рис. 105. Налаштування зони цікавості

Для цього скористайтесь інструментом *Draw a rectangle* (Накреслити прямокутник)  й у меню, що відкриється, натисніть на , щоби перейти до налаштувань шару.

Змінивши назву, натисніть на:

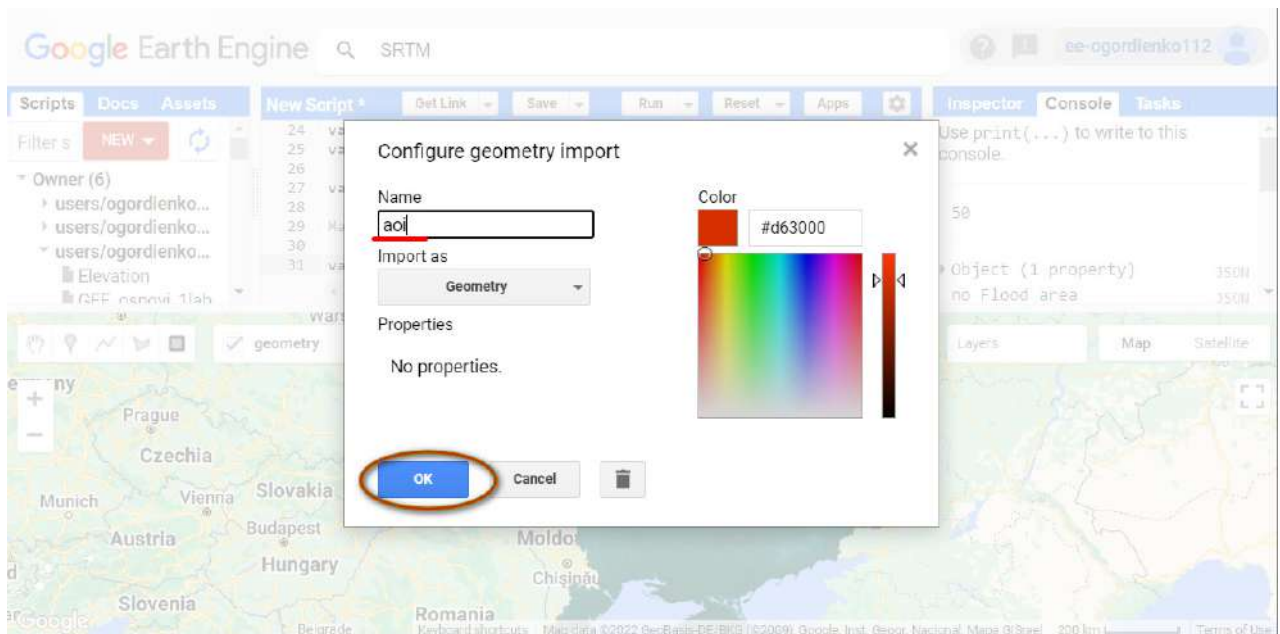


Рис. 106. Налаштування відображення зони цікавості

Для зручності вимкніть два шари в *Layers (Шари): Elevation (Висота, тобто ЦМР) і threshold (Порогове значення)* та змініть вид мапи на Супутник.

Тепер у рядку пошуку вам потрібно знайти г. Говерла. На жаль, пошук не працює за географічними назвами, тому скористаємося найближчим населеним пунктом. У рядку пошуку наберіть *Кваси, Закарпатська область, Україна*.

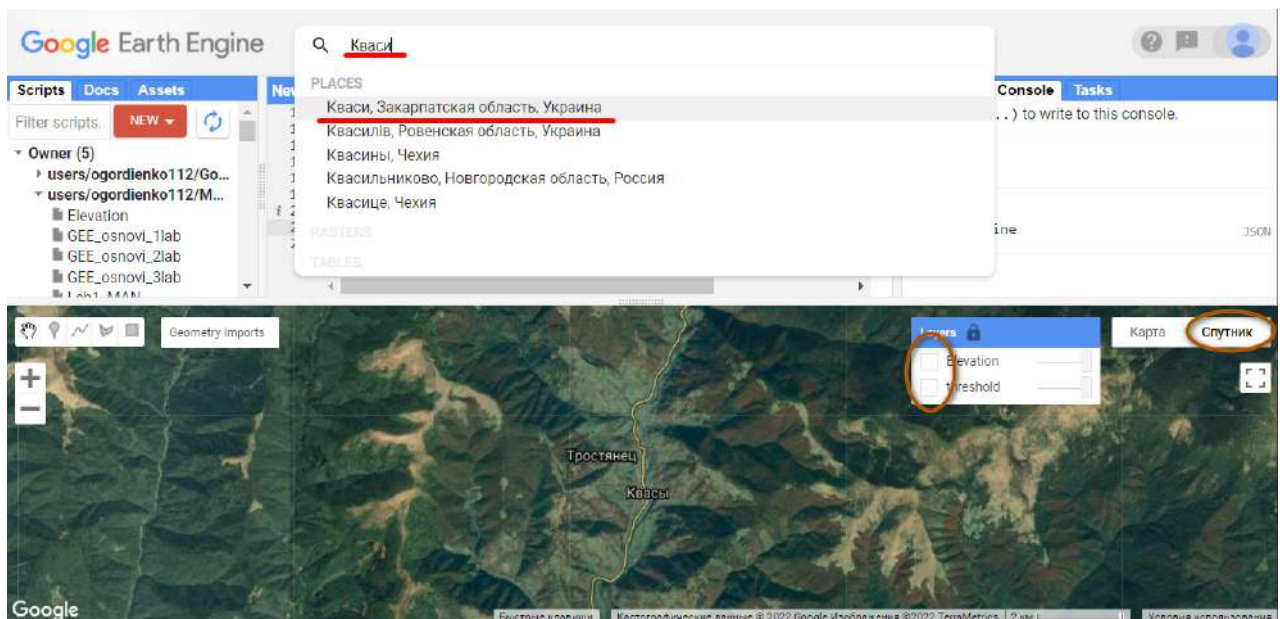


Рис. 107. Відображення шарів

Говерла і сусідня з нею г. Петрос розташовані трохи правіше від с. Кваси. Ідентифікувати їх на супутниковому зображенні можна за допомогою інформації, що ці вершини не мають рослинності. Обведіть за допомогою інструменту **Draw a rectangle (Накреслити прямокутник)**, як показано на рисунку нижче.

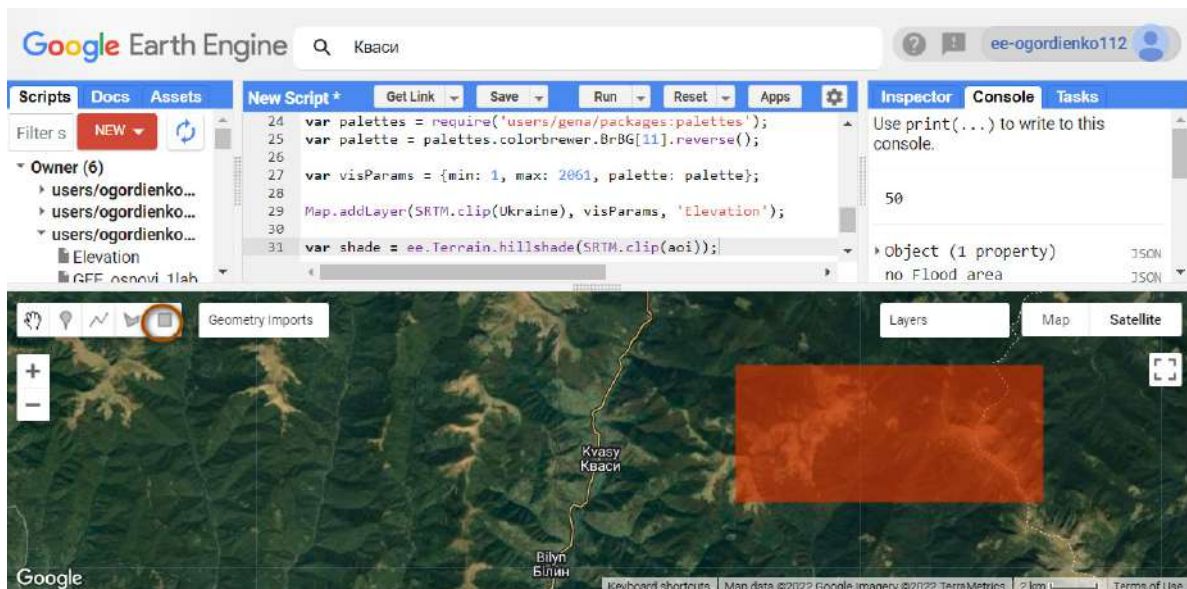


Рис. 108. Створення зони цікавості

Додайте шар відмивки рельєфу на карту. Зробити це можна за допомогою коду:

<pre>Map.addLayer (shade. clip(Ukraine), {}, 'Hillshade')</pre>	<p>Результат з відмивкою рельєфу на карту</p>
--	---

На цьому етапі закоментуйте рядок, який центрує карту.

? Завдання для перевірки

Який рядок потрібно закоментувати, щоб вид карти не змінювався?

8. Налаштувати візуальне представлення відмивки рельєфу на г. Говерла та г. Петрос і завантажити її на комп'ютер.

Щоб краще бачити отриманий шар відмивки рельєфу на Говерлу, потрібно відключити шар зі створеною геометрією в *Geometry imports* (Імпорт геометрії).

✔ **Важливо!** Відключіть шар aoi в *Geometry imports* (Імпорт геометрії).

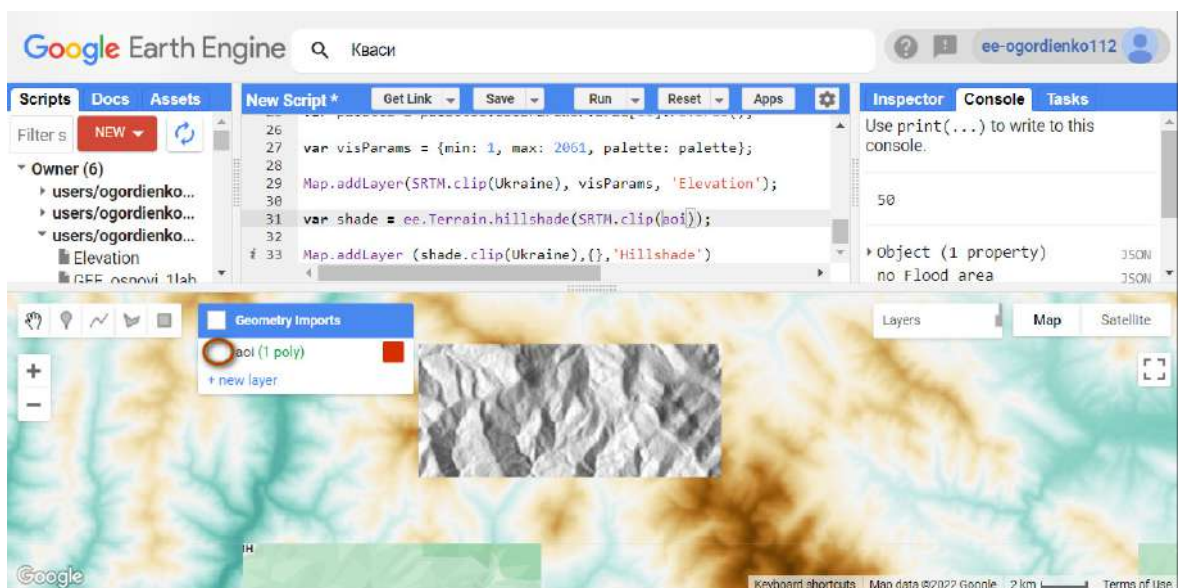




Рис. 109. Додавання відмивки рельєфу

Поки що шар має не надто інформативний вигляд (він однотонного сірого кольору), тому потрібно задати йому візуальні параметри і застосувати їх, а також виконати розтягування гістограми. Для цього в **Layers (Шари)** знайдіть шар із назвою **Hillshade (Відмивка)** і зайдіть у його налаштування  та змініть налаштування шару на таке значення: **Stretch: 1 σ**. Натисніть **Apply (Виконати)** й **Import (Імпортувати)**.

 **Важливо!** Для кожної території значення в **Range (Діапазон)** будуть різні.

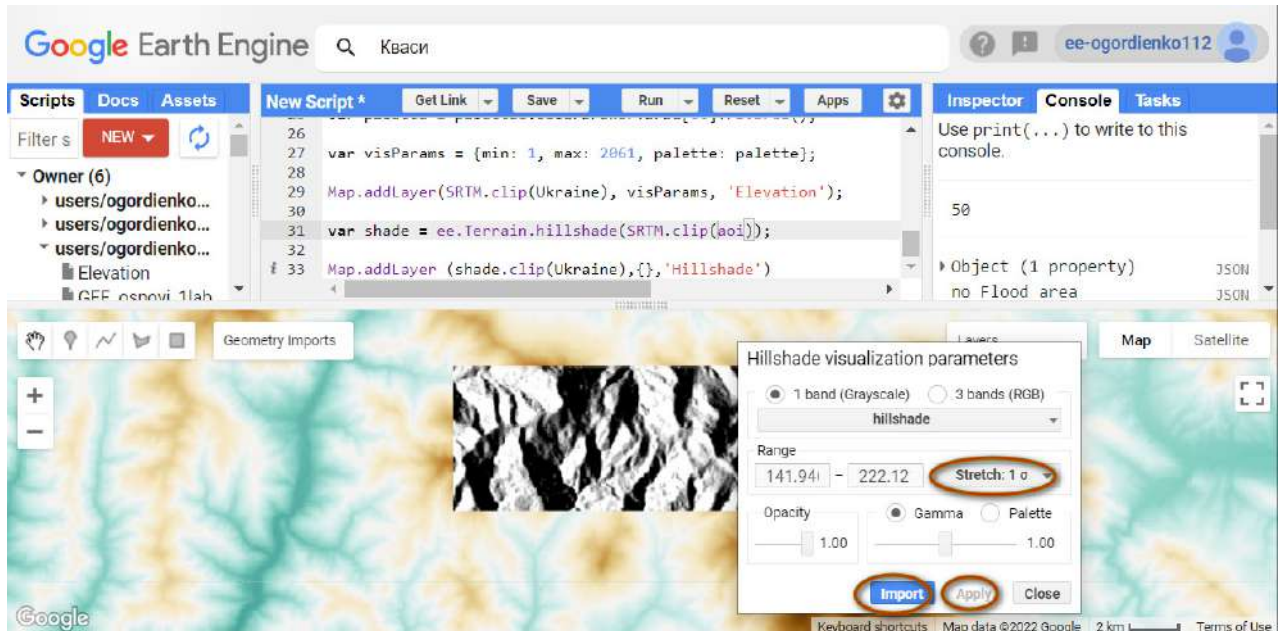



Рис. 110. Налаштування відмивки рельєфу

Тепер, аби налаштування шару зберігалися щоразу, коли ви натискаєте **Run (Запуск скрипту)**, потрібно в **Imports (Імпорт)** знайти нову змінну з назвою **imageVisParam** і змінити останній рядок так:

 **Важливо!** Наступний код не потрібно копіювати, варто лише змінити параметри візуалізації, які ви щойно імпортували.

<pre>Map.addLayer (shade.clip(Ukraine),imageVisParam,'Hillshade')</pre>	<p>Результат з відмивкою рельєфу на мапу з параметрами візуалізації</p>
---	---

Наступним кроком буде створення лінку, який буде експортувати зображення з відмивкою рельєфу.

Це можна зробити за допомогою коду:

<pre>var path = shade.getDownloadURL({ 'scale': 30, 'crs': 'EPSG:4326', 'region': aoi }); print(path);</pre>	<p>Створення покликання для завантаження getDownloadURL створює лінк з розміром пікселя 30 системою координат EPSG:4326 у регіоні aoi Результат у консоль</p>
--	--

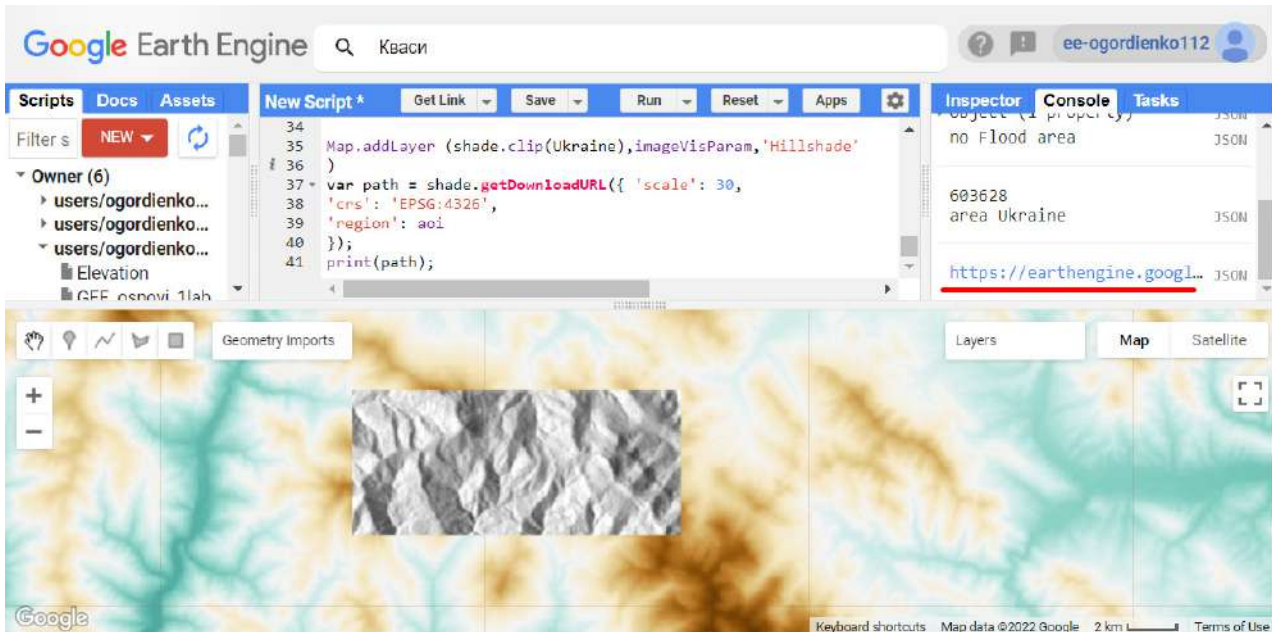


Рис. 111. Завантаження даних

Після того як ви натиснете на лінк у консолі, мапа з відмивкою рельєфу завантажиться на ваш комп'ютер.

? Завдання для перевірки

Завантажте мапу з відмивкою рельєфу на територію гір Говерла та Петрос

9. Створити лінійний об'єкт геометрії для обрахунку профілю рельєфу.

Наступним кроком буде побудова профілю висот. Ця модель також буде працювати на всю Україну. Для подальшої зручності роботи потрібно відключити всі шари в **Layers** (Шари). У коді присутні три команди `Map.addLayer`, до кожної з них додайте змінну **false**.

✔ **Важливо!** Наступний код копіювати не треба. Потрібно поставити параметр **false** у всіх `Map.addLayer`, щоб змінити відображення шарів.

<pre>Map.addLayer(threshold, {palette:['06768F','8F4809']}, 'threshold', false); Map.addLayer(SRTM.clip(Ukraine), visParams, 'Elevation', false); Map.addLayer(shade.clip(Ukraine), imageVisParam, 'Hillshade', false);</pre>	<p>Map.addLayer з параметром false у всіх шарах, доданих на мапу</p>
---	---

Далі потрібно додати нову геометрію в **Geometry imports** (Імпорт геометрії). Маємо створити нову геометрію за допомогою **Geometry imports** (Імпорт геометрії) і назвати її "Line".

Навести курсор миші на **Geometry imports** (Імпорт геометрії), у випадковому меню натиснути **+ new layer** (Створити новий шар).

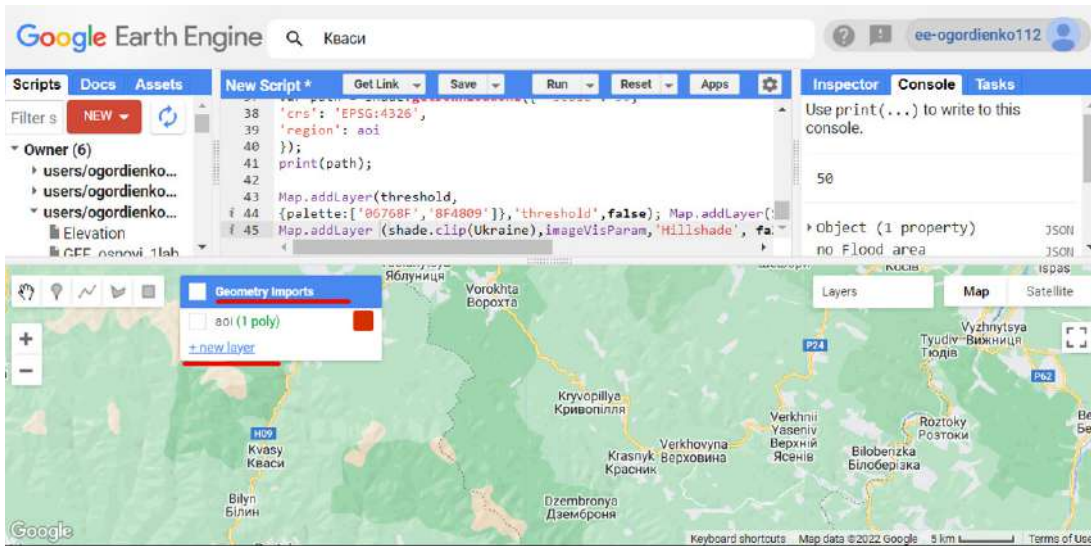




Рис. 112. Відключення шарів

За аналогією з попереднім шаром натиснути на  й у меню, що відкрилося, ввести назву “Line”. Натиснути .

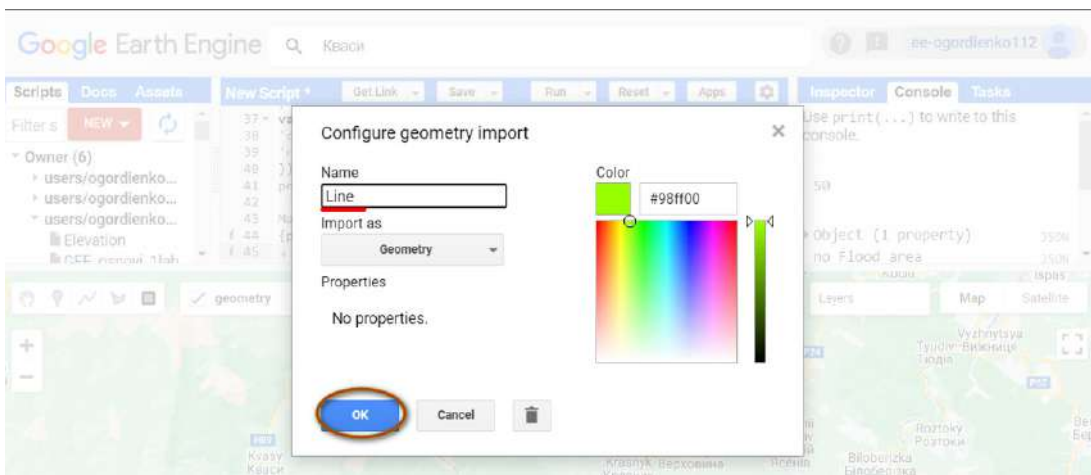


Рис. 113. Зміна назви шару

Наступним кроком буде створення лінії профілю й отримання координат зі списку. Для зручності переключіться спочатку на вид мапи – Супутник.

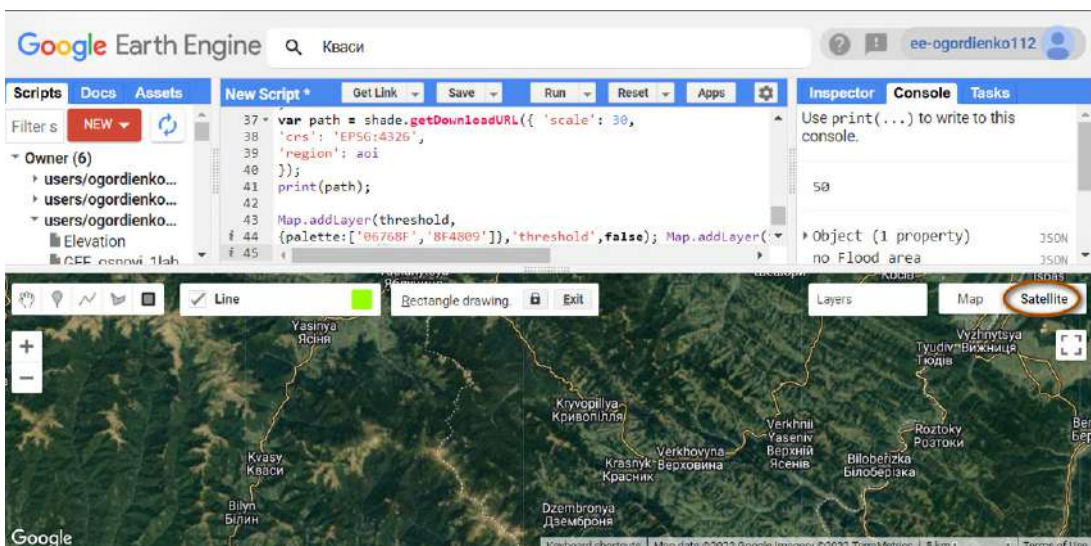



Рис. 114. Підключення даних із супутників високої роздільної здатності

Тепер потрібно створити лінію за допомогою інструменту **Draw line** (Накреслити лінію) в інструментах геометрії . Накресліть лінію, яка проходить через дві вершини – Петрос та Говерлу, приблизно так, як показано на рисунку.

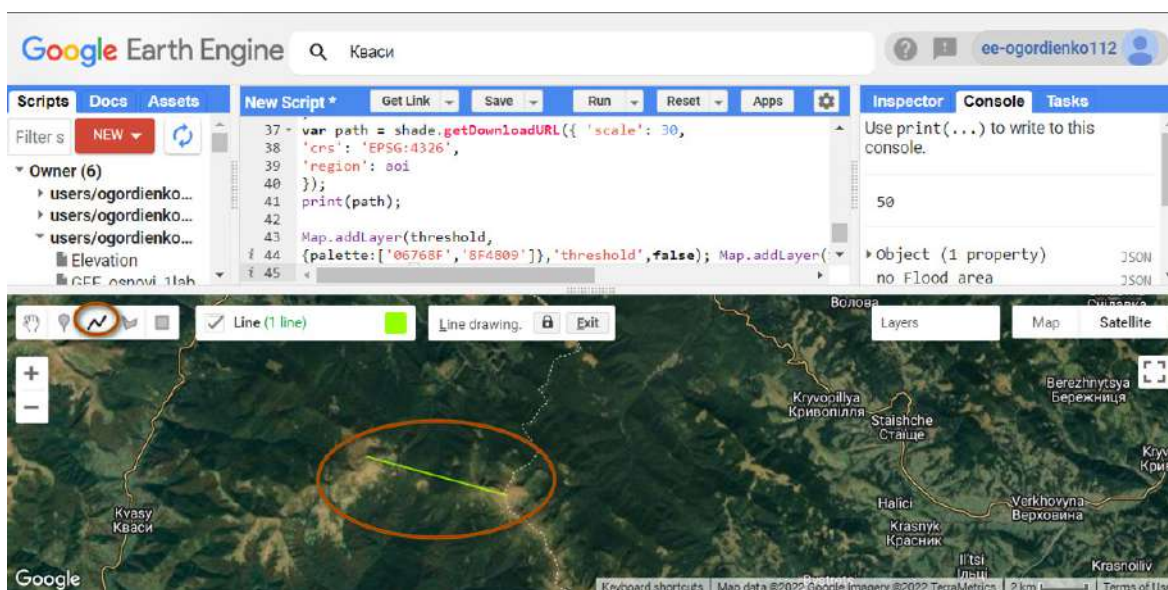


Рис. 115. Створення даних для лінії

У GEE координати й уся атрибутивна інформація зберігається в списках – спеціальному форматі даних. У попередніх роботах вже були використані списки та дані в них. За допомогою ключ-значення можна звернутися до координат лінії. Для кращого розуміння дослідіть наступний код і результат, що з'явився в консолі.

Скопіюйте і натисніть **Run** (Запуск скрипту).

<pre>var line = Line; print(line);</pre>	<p>Змінна line є створеною геометрією з назвою Line; Результат у консоль;</p>
--	---

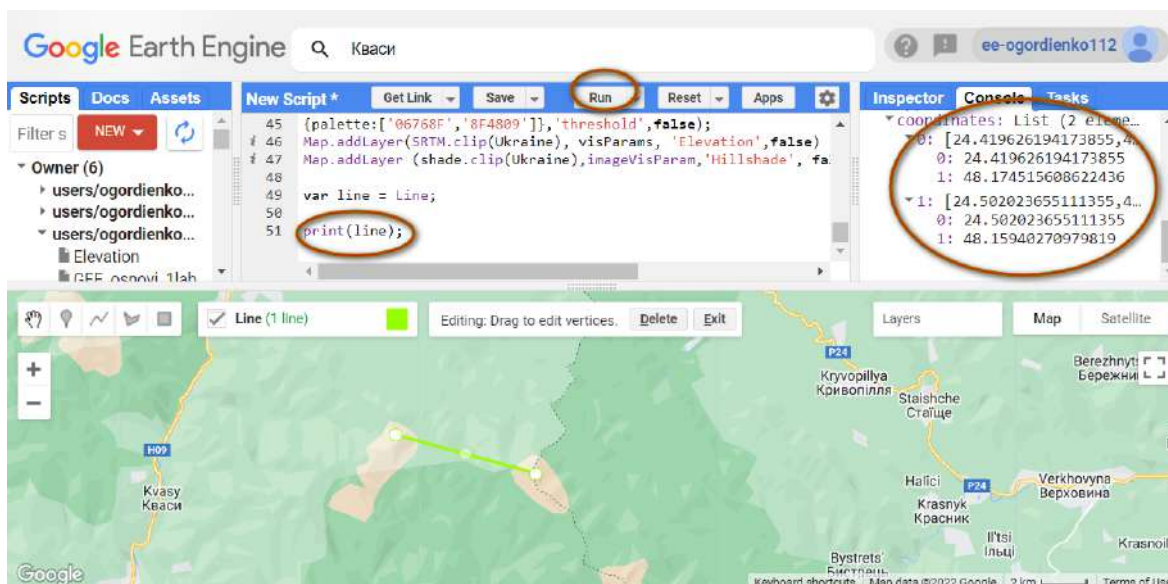


Рис. 116. Координати кінця лінії

Якщо розгорнути в Console (Консоль) результат команди `print (line)`, то можна побачити, як зберігаються координати в GEE.

Після цього потрібно створити змінні, що будуть зберігати координати точок лінії, які потім будуть використовуватися в аналізі.

Для цього скористайтеся кодом:

<pre>var coord = line.coordinates(); var Line_p2 = ee.List(coord.get(0)); var p2_lat = ee.List(Line_p2.get(1)); var Line_p1 = ee.List(coord.get(1)); var p1_lat = ee.List(Line_p1.get(0));</pre>	<p>За допомогою <code>coordinates</code> отримуються координати точок у лінії; У <code>Line_p2</code> поміщаються координати першої точки; <code>p2_lat</code> поміщаються координати широти кінцевої точки; <code>Line_p1</code> поміщаються координати початкової точки; <code>p1_lon</code> поміщаються координати широти початкової точки;</p>
--	--

Цей код створює значення координат для кожного пікселя, а потім застосовує його для ЦМР, яка була додана на початку роботи.

<pre>var latLonImg = ee.Image.pixelLonLat(); var elevImg = SRTM.select('elevation'). addBands(latLonImg);</pre>	<p>За допомогою <code>ee.Image.pixelLonLat</code> до ЦМР додаються параметри широти і довготи положення пікселя; <code>select</code> вибирає <code>'elevation'</code>, в якому зберігається значення пікселя;</p>
--	---

Наступний крок полягає в тому, що потрібно отримати значення висот у список. Зробити це можна за допомогою коду:

<pre>var elevTransect = elevImg.reduceRegion({ reducer: ee.Reducer.toList(), geometry: Line, scale: 30, });</pre>	<p>З попереднього датасету за допомогою <code>ee.Reducer.toList</code> створюється список зі значенням координат та висоти</p>
---	--

Тепер з отриманого списку потрібно створити новий список із значеннями довготи для лінії і значенням висоти для ЦМР. Довгота тут використовується для сортування значень у списку; тепер на графіку значення будуть відображатися, як на мапі – із заходу на схід.

<pre>var lon = ee.List(elevTransect.get('longitude')); var elev = ee.List(elevTransect.get('elevation')); var lonSort = lon.sort(lon); var elevSort = elev.sort(lon);</pre>	<p>Змінна <code>lon</code> обирає значення довготи; Змінна <code>elev</code> обирає значення висот; В обох <code>lonSort</code> та <code>elevSort</code> списках дані сортуються за довготою;</p>
---	---

9. Отримати графік профілю рельєфу г. Говерла та г. Петрос і завантажити його на комп'ютер.

Наступним кроком є створення графіка, на якому будуть відображатися усі значення висот із ЦМР, які перетинаються з лінією.

<pre> var chart = ui.Chart.array.values({array: elevSort, axis: 0, xLabels: lonSort}) .setOptions({ title: 'Elevation Profile Across Longitude', hAxis: { title: 'Longitude', viewWindow: {min: p2_lat, max: p1_lat}, titleTextStyle: {italic: false, bold: true} }, vAxis: { title: 'Elevation (m)', titleTextStyle: {italic: false, bold: true} }, colors: ['654321'], lineSize: 5, pointSize: 0, legend: {position: 'none'} }); print(chart.setChartType('AreaChart')); </pre>	<p><code>ui.Chart.array.values</code> створює графік із заданими параметрами для списків. <code>array: elevSort</code> обирає значення для осі Y, <code>xLabels: lonSort</code> обирає значення для осі X. <code>setOptions</code> дає змогу встановлювати параметри для графіка.</p> <p><code>title</code> – назва графіка <code>hAxis</code> та параметри всередині – це підпис до осі X</p> <p><code>viewWindow</code> дає змогу налаштувати максимальне та мінімальне значення для відображення графіка</p> <p><code>titleTextStyle</code> – параметри відображення тексту</p> <p><code>vAxis</code> параметри всередині – це підпис до осі X <code>titleTextStyle</code> встановлює параметри візуалізації тексту підписів</p> <p><code>colors</code> – колір графіка <code>lineSize</code> – розмір лінії <code>pointSize</code> – розмір точок, з параметром 0 не відображаються</p> <p><code>legend</code> – підпис до графіка, параметр 'none' не відображає його.</p> <p>Результат у консоль з типом графіка 'AreaChart'</p>
---	--

Після цих дій будується графік висот, тобто профіль висот для г. Говерла та г. Петрос. Цей код буде працювати по всій території України.

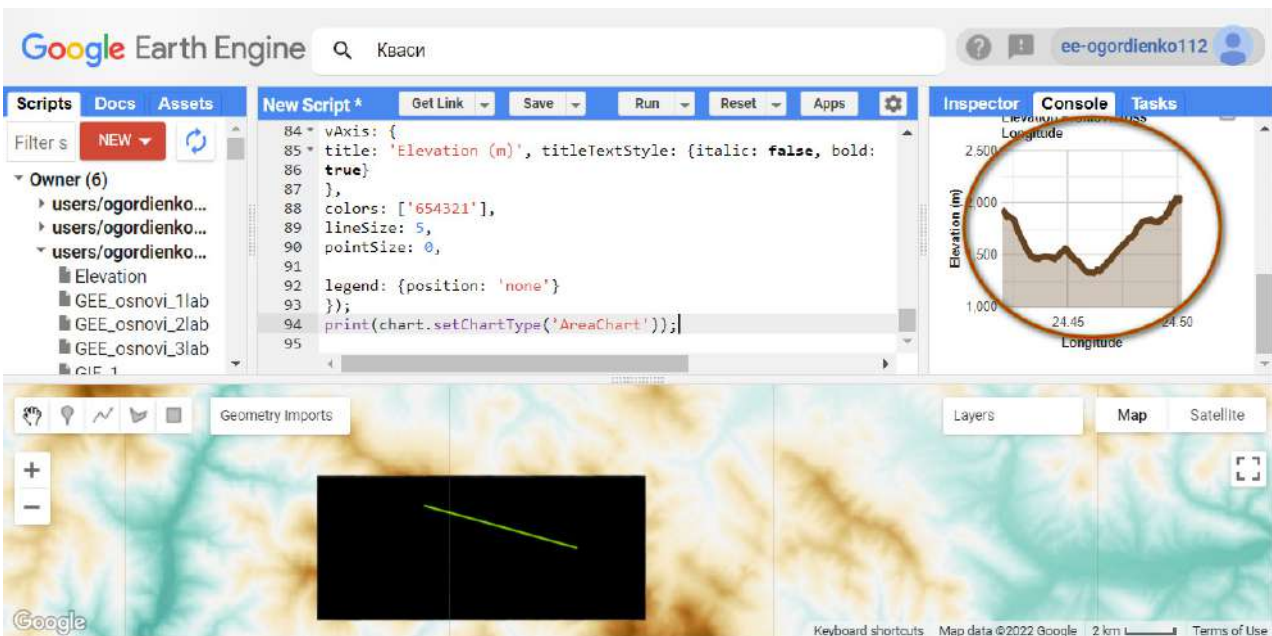



Рис. 117. Створення профілю рельєфу

? Завдання для перевірки

Завантажте профіль висот для г. Говерла та г. Петрос у форматі PNG

Для виконання цього завдання відкрийте графік у новому вікні за допомогою .

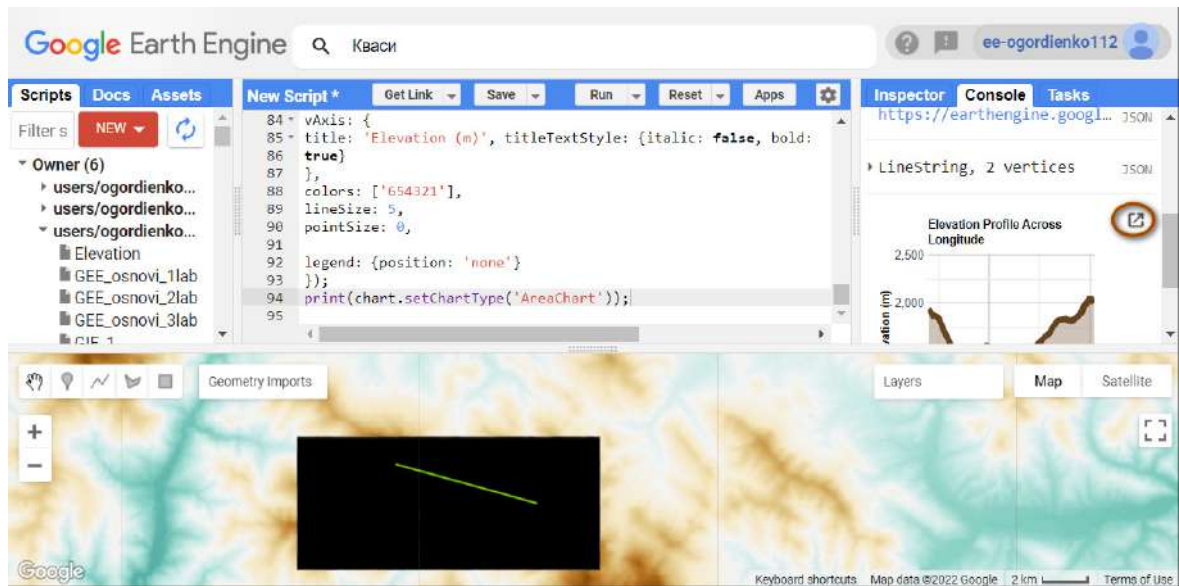


Рис. 118. Виведення в нове вікно профілю рельєфу

Натисніть **Download PNG**, щоб завантажити графік.



Рис. 119. Завантаження рисунка з профілем рельєфу

За наведеним нижче QR-кодом ви можете переглянути відео виконання практичної роботи.



Рис. 120. Відео виконання практичної роботи

Готовий код можна відкрити за лінком:

https://code.earthengine.google.com/?scriptPath=users%2Fogordienko112%2Fcourse_MAN%3ALab_5

Візуальне порівняння радарних і спектральних супутникових зображень, створення водної маски для дослідження паводка на р. Дністер

Ситуація

Оптичні зображення мають широке використання у ДЗЗ. Висока роздільна здатність знімків дає змогу вести моніторинг рослинного покриву, водойм, пожеж чи наслідків антропогенної діяльності. Проте їхнім недоліком є обмеженість використання в нічний час доби й за несприятливих погодних умов. На противагу спектральним зображенням радарні знімки можуть доповнити прогалину в даних, які отримані в нічний час доби або під час суттєвої хмарності. Основною відмінністю між спектральними і радарними знімками є те, що перші отримані з пасивних сенсорів, а другі – з активних, тобто перш ніж отримати дані, сенсор відправляє сигнал для відбиття від певної поверхні.

Паводки і повені, спричинені інтенсивними дощами, є однією з основних кліматичних загроз на території України. Вони завдають збитків на мільйони гривень щороку. Так негода у червні 2020 р. була найбільшою за останні десятки років і спричинила розлив річок на заході країни з подальшим підтопленням цивільної інфраструктури¹. Також дощові дні супроводжувалися значною хмарністю, що унеможливило використання спектральних даних для моніторингу ситуації.

Завдання

Порівняймо радарні дані на період під час паводка і після нього на р. Дністер у районі м. Галич. Проаналізуймо, наскільки радарні знімки можуть бути корисними для моніторингу затоплених територій.

Алгоритм виконання завдання:

1. Відкрити Code Editor і завантажити датасет Sentinel-1 та Sentinel-2.
2. Вказати координати території України, де відбувся паводок, і виділити прямокутником територію інтересу в районі м. Галич на р. Дністер.
3. Вибрати радарні дані Sentinel-1 у період паводка за датою із 24 по 25 червня 2020 р.
4. Вибрати радарні дані Sentinel-1 після паводка за датою із 23 по 24 серпня 2020 р. в період межені з найменшим рівнем води в річці.
5. Вибрати оптичні дані Sentinel-2 в період паводка за датою із 24 по 25 червня 2020 р.
6. Створити маску для водної поверхні в момент паводка.
7. Зберегти на комп'ютер отримане растрове значення маски водної поверхні в червні.
8. Самостійна робота: створити і зберегти маску води після паводка – в серпні.

Покрокова інструкція

1. Відкрити **Code Editor** і завантажити датасет Sentinel-1 та Sentinel-2.

Додамо датасети через **Code Editor** (Редактор коду): <https://code.earthengine.google.com/>. Для початку потрібно завантажити датасет радарних даних **Sentinel-1**. Наберіть у пошуку тег **Sentinel-1** і з випадного списку виберіть **Sentinel-1 SAR GRD: C-band Synthetic Aperture Radar** (Радар із синтезованою апертурою) та натисніть на **import** (імпортувати), щоб датасет з'явився у вікні **Imports** (Імпорт). У вкладці **Imports** (Імпорт) має з'явитися новий датасет з ім'ям **imageCollection**.

¹ URL: https://uk.wikipedia.org/wiki/Паводок_на_заході_України_2020_року

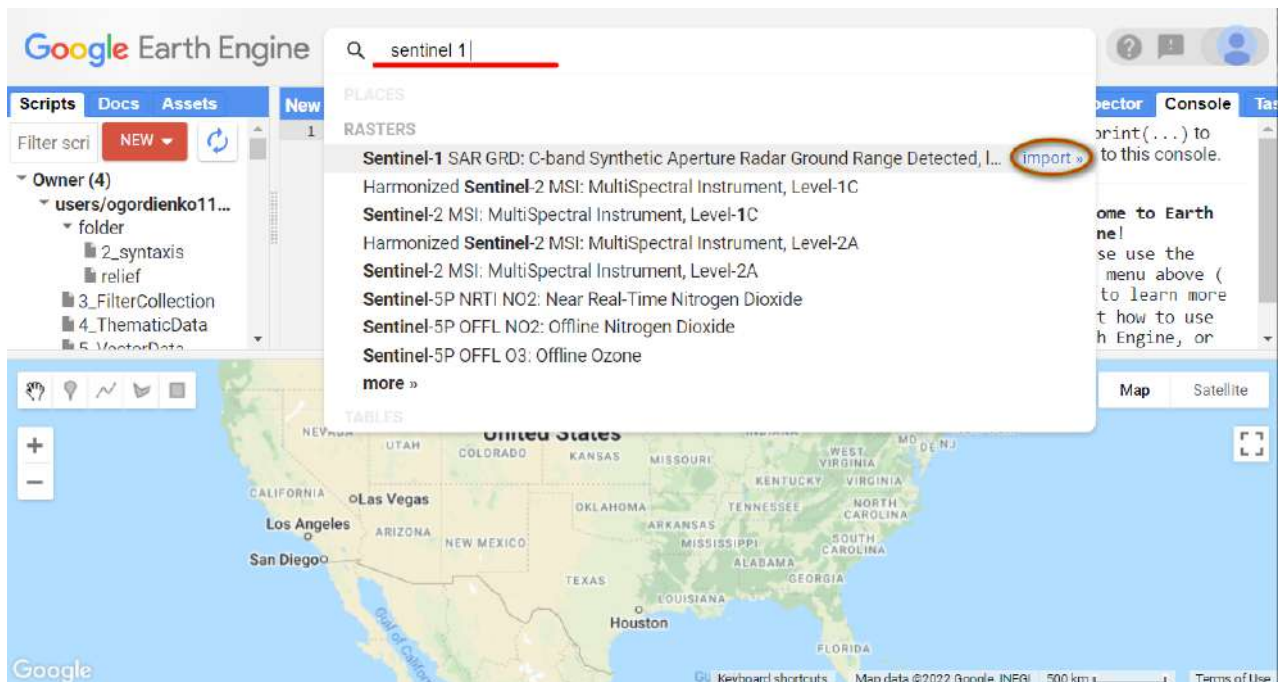


Рис. 121. Пошук даних

Наступним кроком буде підвантаження мультиспектральних супутникових знімків із космічного апарата **Sentinel-2**. Для цього в рядку пошуку наберіть **Sentinel-2** й у випадковому списку виберіть дані з рівнем обробки 2A, а саме **Sentinel-2 MSI: Multispectral Instrument Level-2A**. Натисніть **import** (імпортувати). У вкладці **Imports** (Імпорт) має з'явитися новий датасет з ім'ям **imageCollection2**.

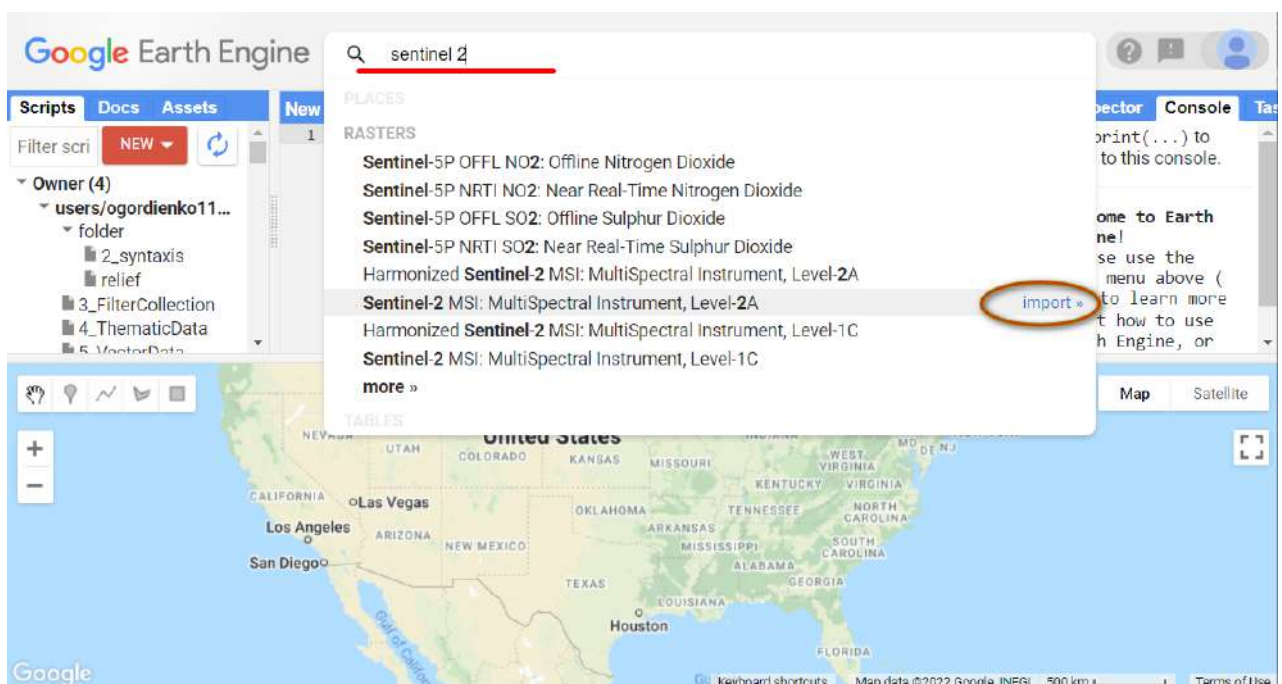


Рис. 122. Імпорт даних

Тепер у вкладці **Imports** (Імпорт) мають з'явитися два датасети з назвами **imageCollection1** й **imageCollection2**, які будуть використані в практичній роботі.

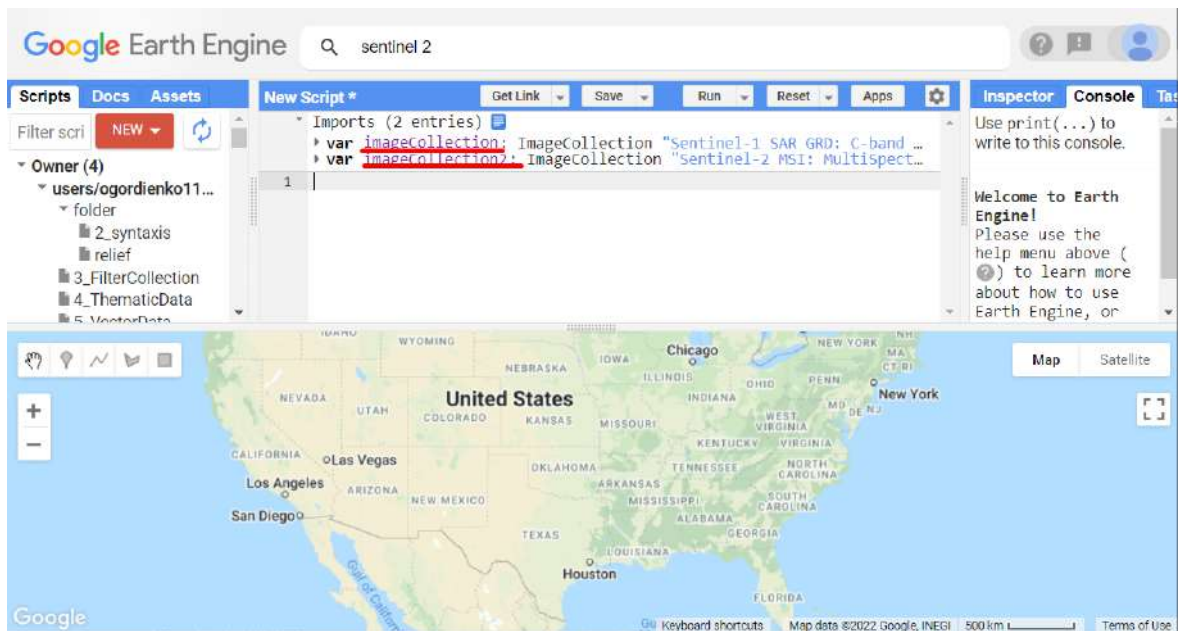


Рис. 123. Імпортовані колекції

2. Вказати координати території України, де відбувся паводок, і виділити прямокутником територію інтересу в районі м. Галич на р. Дністер.

Наступний крок – додавання координат зони інтересу (території нашого дослідження). Для цього потрібно вставити рядок із кодом:

<code>Map.setCenter(24.9099, 49.0101, 9);</code>	<code>Map.setCenter</code> центрує об'єкт на мапі за вказаними координатами
--	---

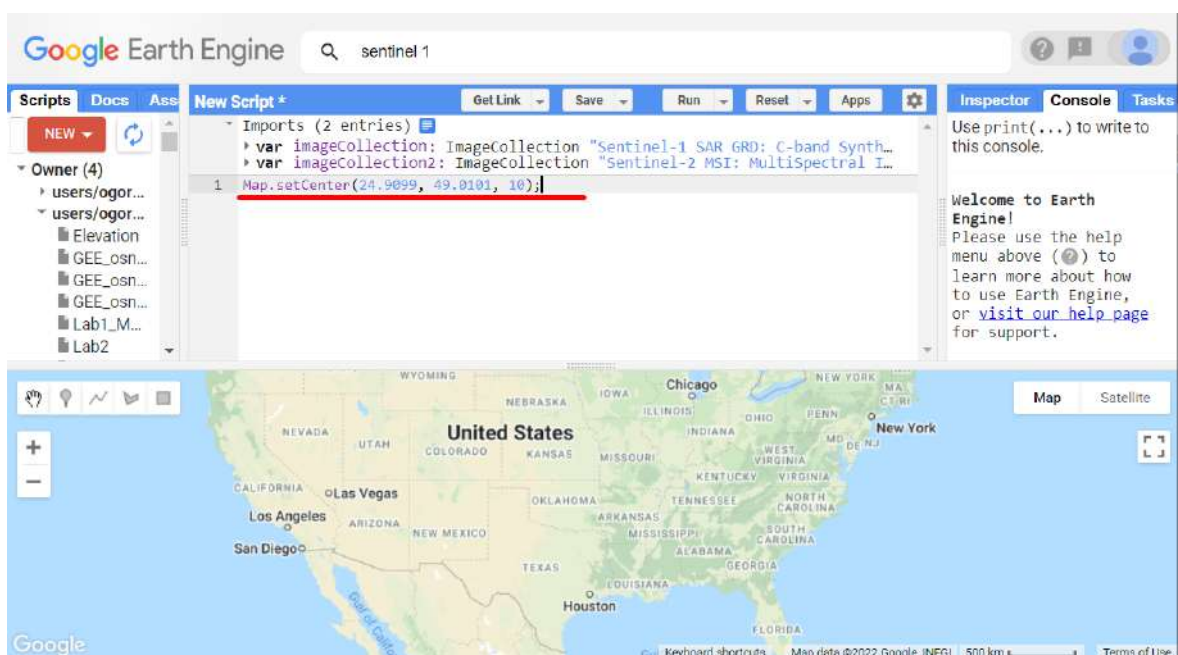


Рис. 124. Центрування мапи

Тепер необхідно змінити вид мапи на гібрид. Зробити це можна за допомогою рядка коду:

<code>Map.setOptions('Hybrid');</code>	<code>Map.setOptions</code> змінює вид мапи
--	---

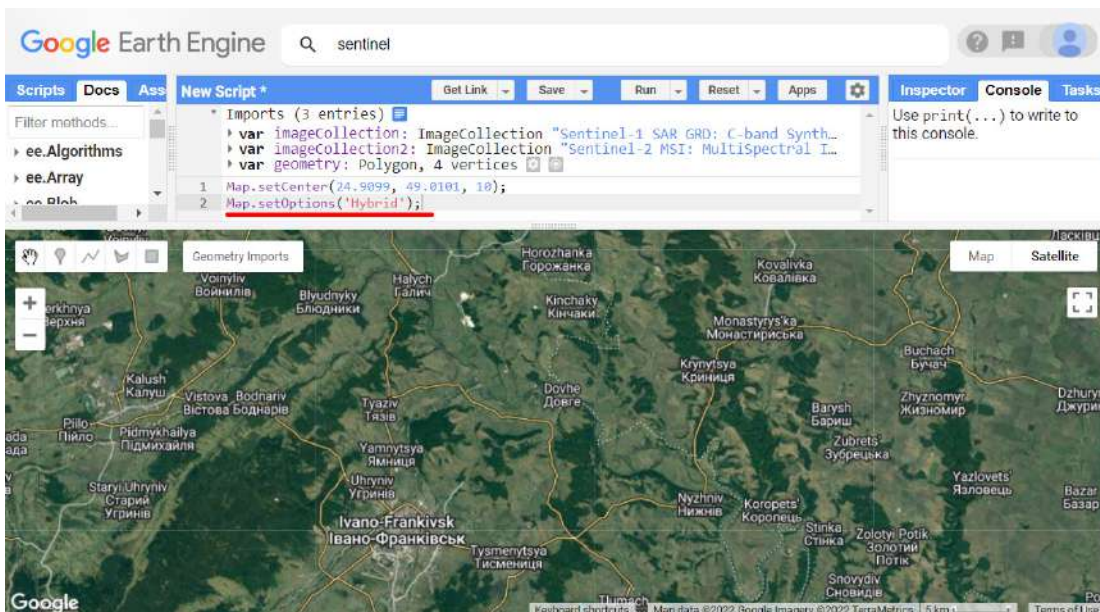



Рис. 125. Зміна вигляду мапи

Далі необхідно вибрати зону інтересу, за якою відбуватимуться пошук та обробка супутникових знімків. Для цього за допомогою інструментів *Geometry imports* (Імпорт геометрії) додамо прямокутну фігуру *Draw a rectangle* (Накреслити прямокутник)  у районі м. Галич, щоб захопити русло р. Дністер. У вкладці *Imports* (Імпорт) з'являться також нові дані з назвою *geometry*.

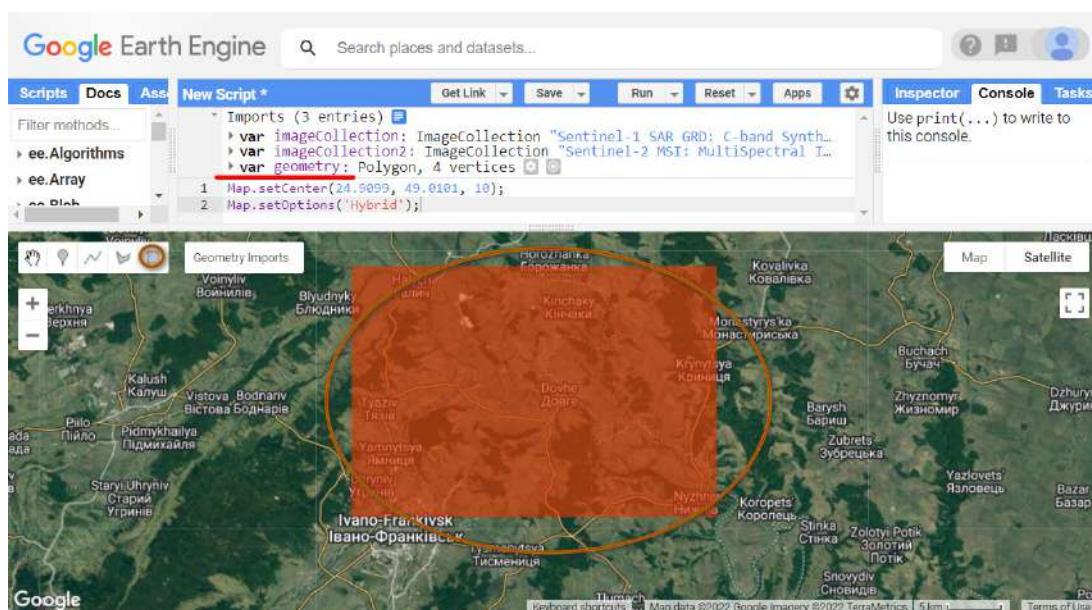


Рис. 126. Створення геометрії

3. Вибрати радарні дані Sentinel-1 у період паводка за датою із 24 по 25 червня 2020 р. На цьому етапі потрібно відфільтрувати колекцію знімків Sentinel-1 за заданими параметрами. Скористайтеся кодом:

<pre>var dataset_1 = imageCollection .filterDate('2020-06-24','2020-06-25') .filterBounds(geometry) .filter(ee.Filter.eq('orbitProperties_pass', 'DESCENDING')) .select('VV','VH') .first();</pre>	<p>змінна <i>dataset_1</i> фільтрує знімки за датою фільтрує дані в межах <i>geometry</i> фільтр за параметрами орбіти</p> <p>обираються два канали з колекції береться перше зображення</p>
--	--

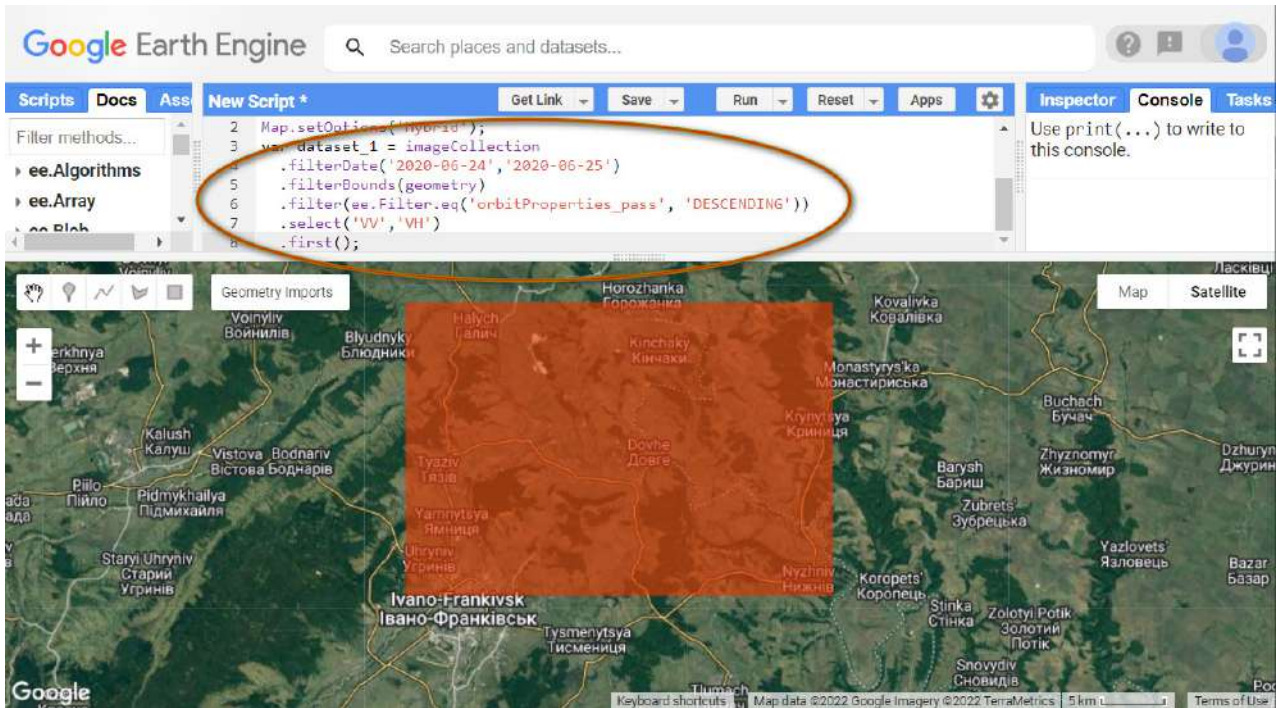


Рис. 127. Фільтрування колекції

Відфільтровані зображення можна додати до мапи. Для цього скористайтеся кодом нижче.

Важливо! Радарні дані виходять за видимий діапазон, який сприймає людське око. Довжина хвилі в Sentinel-1 – 5,5 сантиметрів. Звичайне зображення без псевдокольорів має вигляд, як на рисунку нижче. Можна сказати, що один канал, у цьому випадку VV, відображається в чорно-білому діапазоні.

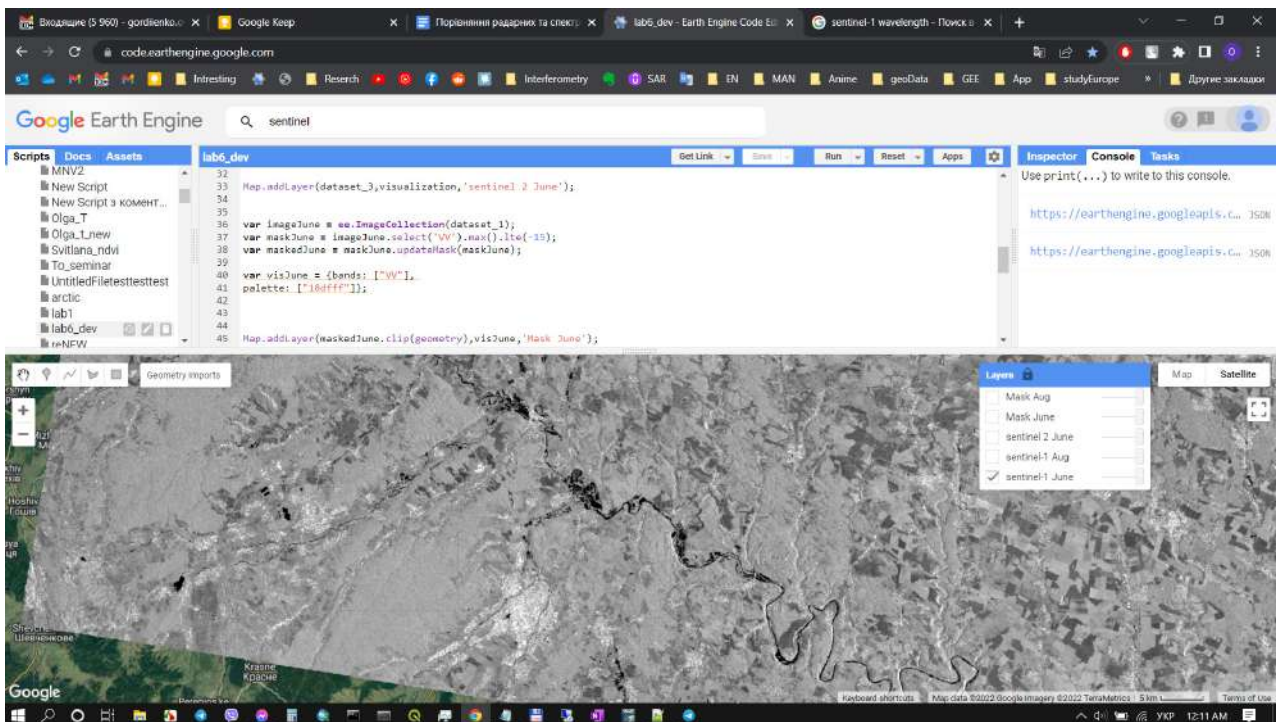


Рис. 128. Додавання шару на мапу

```
Map.addLayer(dataset_1, {'bands':
'VV,VH,VV', min: [-20, -20, -15],max: [0, -15,
0]}, 'sentinel-1 June');
```

Додавання зображення на мапу трьох каналів з мінімальним та максимальним значенням.
Назва **sentinel-1 June**

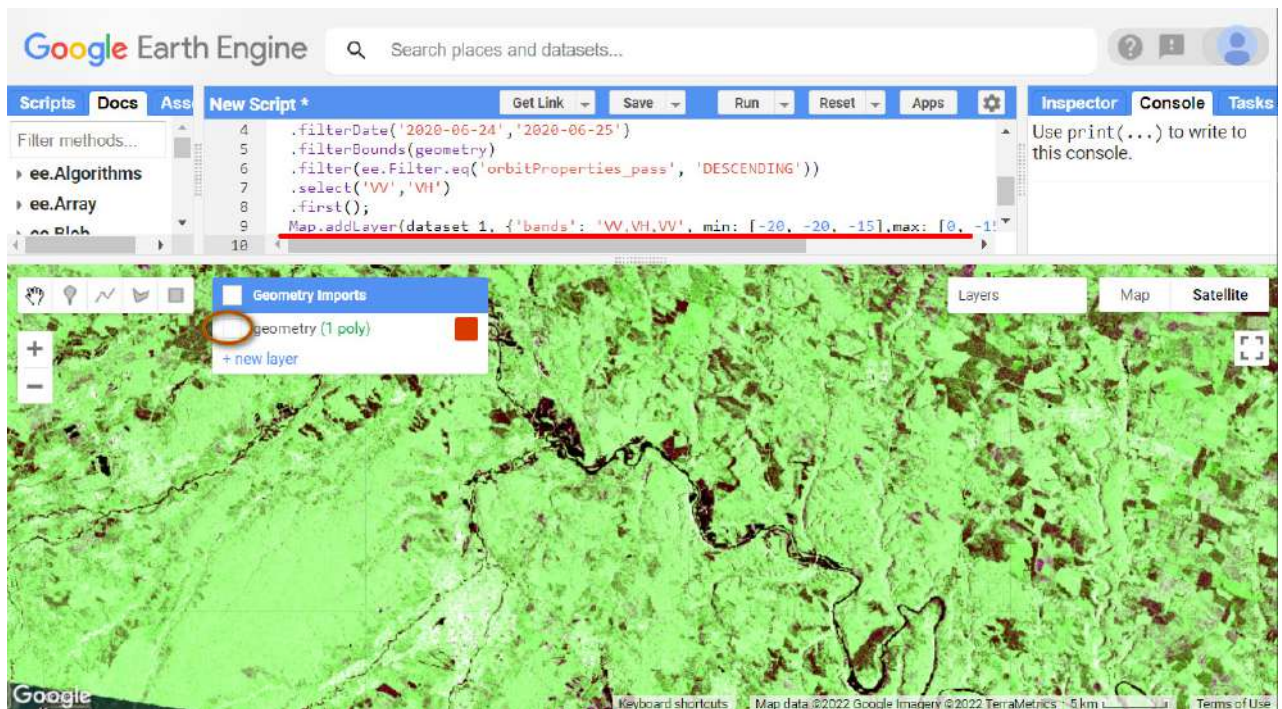


Рис. 129. Зміна відображення геометрії

Зображення з'явилося на мапі. Воно створене з двох каналів, доступних на територію України (**VV,VH**): **VV** (*Vertical receive, Vertical retrieve*) (*Вертикальна віддача, вертикальний прийом*), **VH** (*Vertical receive, Horizontal Retrieve*) (*Вертикальна віддача, горизонтальний прийом*). У такому кольоровому вигляді воду на знімках можна ідентифікувати за темним, ближче до чорного кольором, при цьому с/г поля певного типу рослинності мають коричнево-фіолетовий колір. Усі інші території – зеленого кольору.

Також в інструментах геометрії відключіть додану геометрію.

4. Вибрати радарні дані Sentinel-1 після паводка за датою із 23 по 24 серпня 2020 р. в період межені з найменшим рівнем води в річці.

Тепер необхідно відфільтрувати колекцію Sentinel-1 за новими датами після паводка. Для цього скористайтеся кодом нижче.

<pre>var dataset_2 = imageCollection .filterDate('2020-08-23','2020-08-24') .filterBounds(geometry) .filter(ee.Filter.eq('orbitProperties_pass', 'DESCENDING')) .select('VV','VH') .first();</pre>	<p>Змінна dataset_2 filterDate фільтрує знімки за датою filterBounds фільтрує дані в межах geometry Фільтр за параметрами орбіти select вибирає два канали з колекції береться перше зображення</p>
---	---

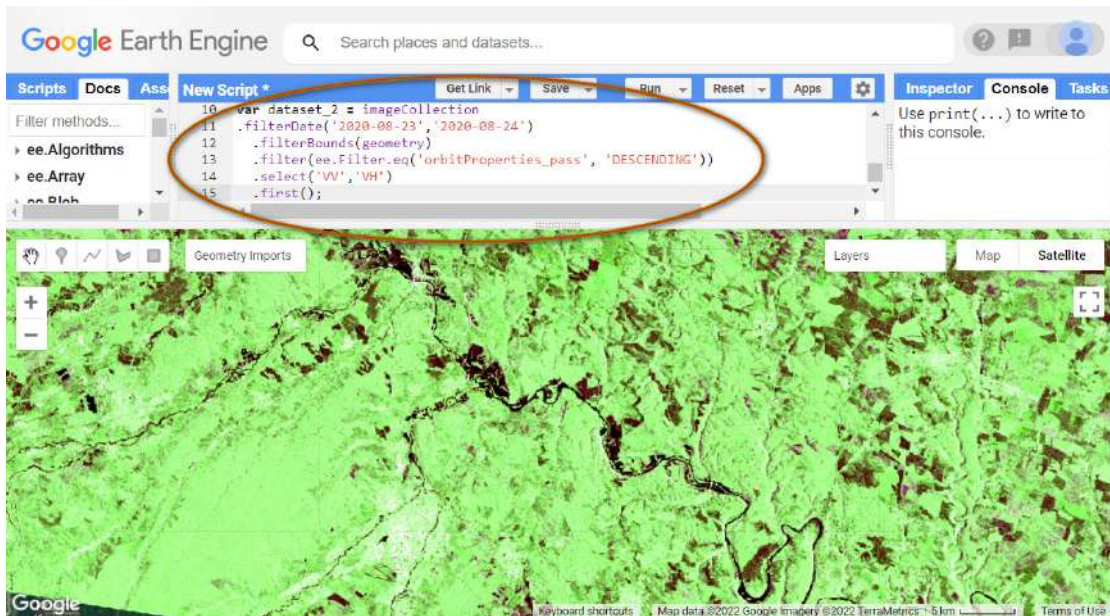


Рис. 130. Фільтрування колекції

Щоби додати шар на мапу, потрібно скористатися рядком:

```
Map.addLayer(dataset_2, {'bands':
'VV,VH,VV', min: [-20, -20, -15],max: [0, -15,
0]}, 'sentinel-1 Aug');
```

Додавання зображення на мапу в трьох каналах із мінімальним та максимальним значенням. Назва **sentinel-1 Aug**

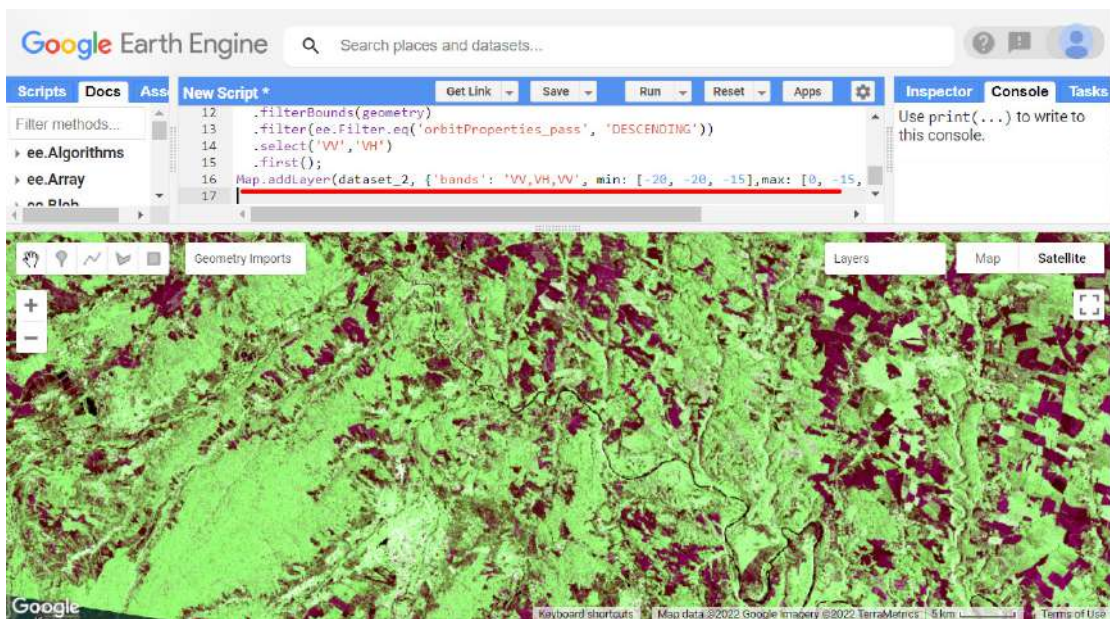


Рис. 131. Додавання шару на мапу

Порівняйте між собою два радарні зображення й ідентифікуйте населені пункти, де спостерігався найбільший розлив р. Дністер на цю дату: 24.06.2020 р.

✓ **Важливо!** Вода має іншу відбивну здатність через свої електромагнітні характеристики. На радарних даних у випадку GEE було вибрано значення -18 , але для кожної зони та похилу кута знімка це значення може трохи змінюватися.

5. Вибрати оптичні дані Sentinel-2 в період паводка за датою із 24 по 25 червня 2020 р.

Також для порівняння вигляду радарних та оптичних даних можна додати мультиспектральний знімок Sentinel-2. Зробити це можна за допомогою коду:

<pre>var dataset_3 = imageCollection2 .filterDate('2020-06-25','2020-06-26') var visualization = { min: 0, max: 6500, bands: ['B12', 'B8', 'B4'], }; Map.addLayer(dataset_3,visualization,'sentinel 2 June');</pre>	<p>dataset_3 звертається до датасету з Sentinel-2 filterDate фільтрує колекцію за датами;</p> <p>visualization візуальні параметри з відображенням у псевдокольорах у комбінації каналів 12,8,4;</p> <p>Map.addLayer додає шар на мапу з параметрами візуалізації visualization та назвою sentinel 2 June</p>
---	---

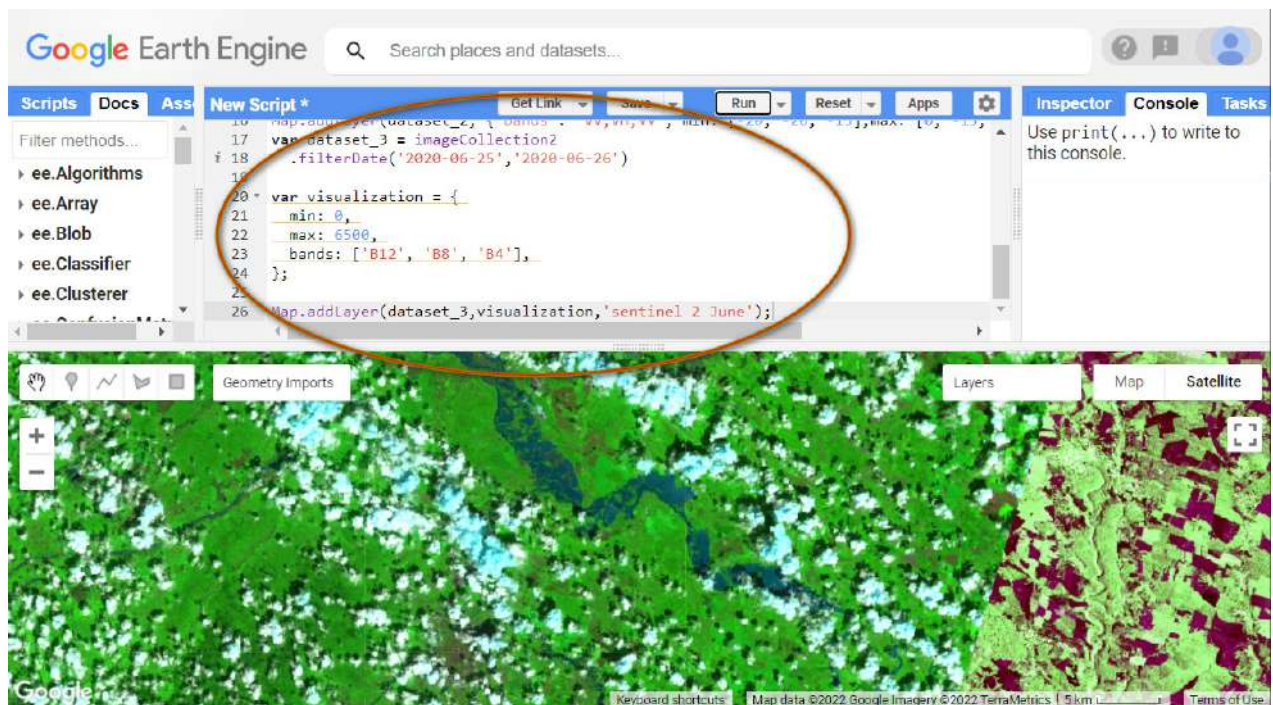


Рис. 132. Зміна візуалізації шару

Порівняйте, який вигляд мав паводок на знімках **Sentinel-1** за 24 червня і **Sentinel-2** за 25 червня. Зверніть увагу на те, як збільшилася площа затоплених територій з різницею в один день. І ще раз подивіться, які переваги є в радарних даних, що не залежать від хмарності і тому є найбільш ефективними для моніторингу паводкової ситуації в період затяжних дощів, коли небо повністю вкрите хмарами.

6. Створити маску для водної поверхні в момент паводка.

Наступним кроком буде накладання маски на перше радарне зображення паводка. Для цього скористайтеся кодом:

<pre>var imageJune = ee.ImageCollection(dataset_1); var maskJune = imageJune.select('VV').max(). lte(-15); var maskedJune = maskJune. updateMask(maskJune);</pre>	<p>imageJune звертається до відфільтрованого датасет dataset_1 maskJune вибирає з imageJune канал 'VV', за допомогою max бере його максимальні значення, за допомогою lte(-15) бере параметри всі, що менші за -15 maskedJune за допомогою updateMask застосовує маску до maskJune</p>
--	--

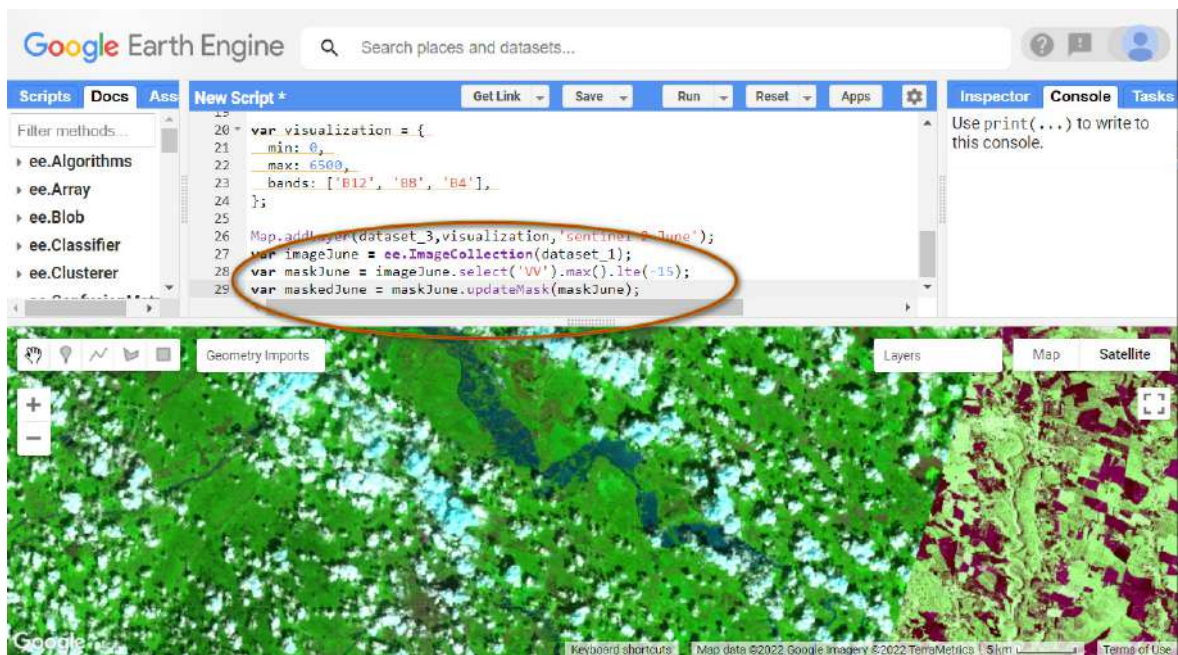


Рис. 133. Маскування даних

Маска отримується за допомогою порогового значення у випадку з територією дослідження, яку ми вивчаємо. Значення водних об'єктів набувають всі об'єкти, яскравість (інтенсивність) пікселя яких менша -15 dB. Також слід зауважити, що для маскування ми використовуємо поляризацію VV, тобто вертикальну передачу й вертикальне прийняття.

Для створеної маски води потрібно навести візуальні параметри. Скористайтеся кодом:

<pre>var visJune = {bands: ["VV"], palette: ["18dfff"]};</pre>	<p>visJune візуальні параметри до маски з кольором 18dfff</p>
--	---

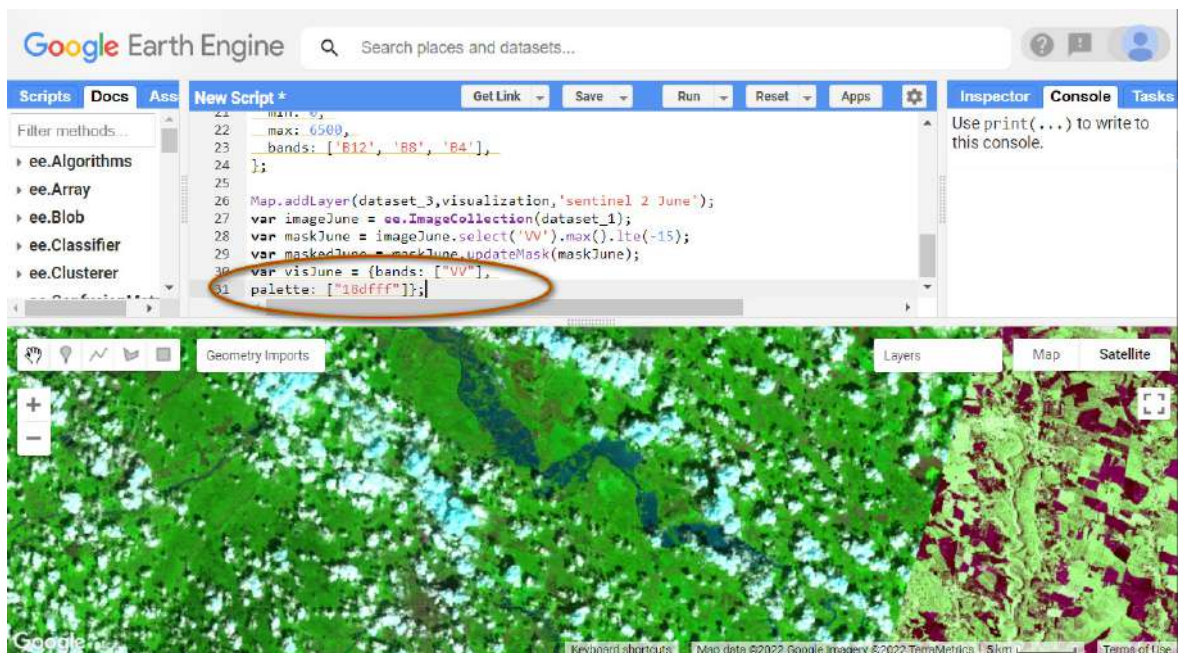


Рис. 134. Налаштування відображення маски

Додамо на екран шар з маскою води станом на червень **Mask June**.

<pre>Map.addLayer(maskedJune, clip(geometry), visJune, 'Mask June');</pre>	<p>Map.addLayer додає шар маски на мапу, clip обрізає зображення в межах geometry з візуальними параметрами visJune та назвою Mask June</p>
--	---

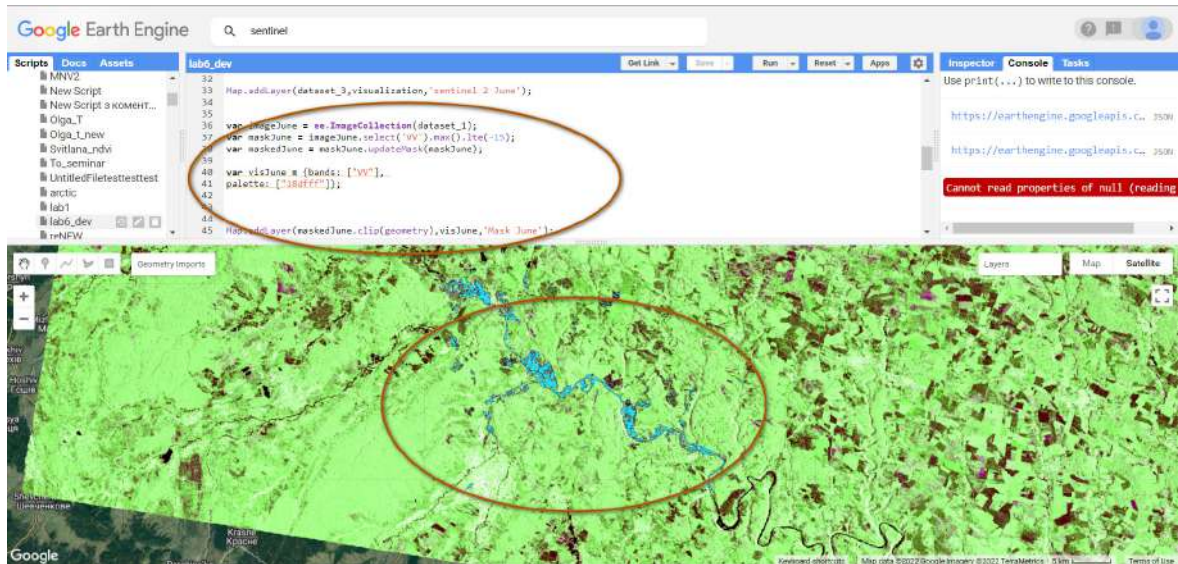


Рис. 135. Маска для води додана на мапу

Маска для води додана на мапу.

Крім води до візуалізації потрапляють поля. Це стається через особливості радарних даних: на деяких полях може бути надмірне зволоження тощо.

7. Зберегти на комп'ютер отримане растрове значення маски водної поверхні в червні.

Тепер потрібно завантажити маску **Mask June** собі на комп'ютер. Зробити це можна за допомогою коду:

<pre>var path = maskedJune.getDownloadURL({ 'scale': 30, 'crs': 'EPSG:4326', 'region': geometry }); print(path);</pre>	<p>path створює лінк для завантаження для датасет maskedJune 'scale' вказує на масштаб пікселів, 'crs' вказує на систему координат, 'region' вказує на зону інтересу print результат у консоль</p>
--	--

У вкладці **Console** (Консоль) з'явиться покликання, на яке потрібно натиснути, щоб завантажити маску собі на комп'ютер.

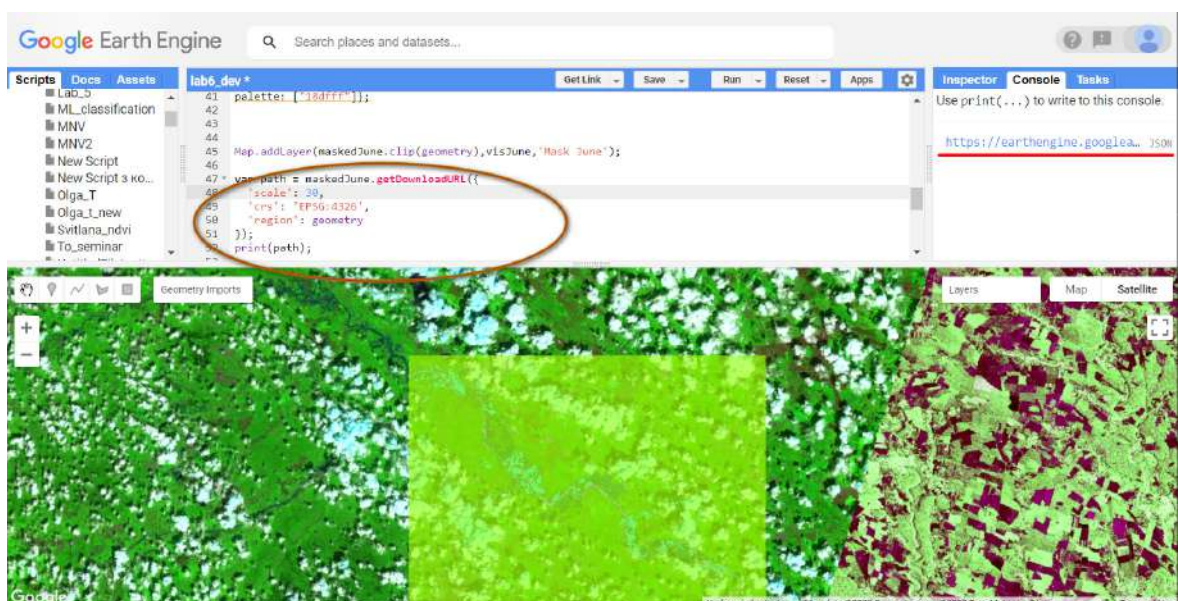


Рис. 136. Інформація в консолі

Завантажений растровий шар маски **Mask June** в QGIS повинен мати такий вигляд:

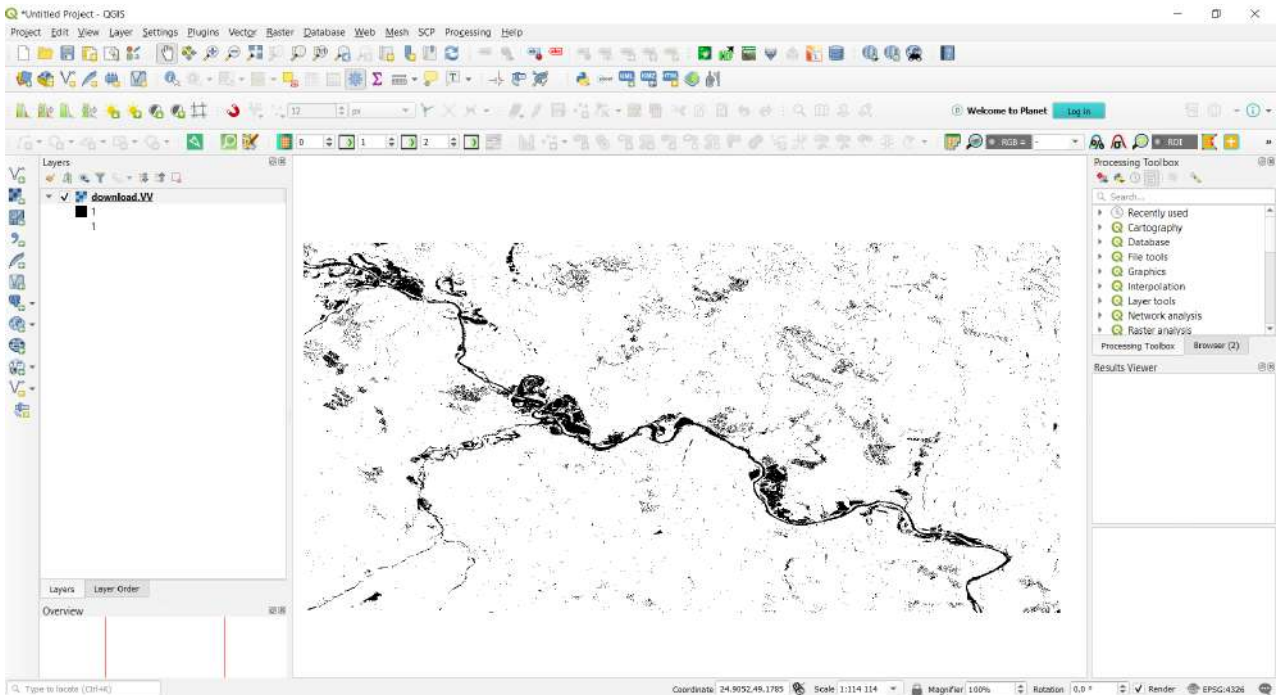


Рис. 137. Вигляд у QGIS

8. Самостійна робота: створити і зберегти маску води після паводка – в серпні.
Для порівняння масштабів розливу Дністра далі потрібно створити маску після паводка.

? Завдання для перевірки

Додати маску після паводка, завантажити її собі на комп'ютер

Для виконання цього завдання вам потрібно самим додати нові змінні за аналогією з накладанням маски. Поверніться до кроку, де додається маска, і створіть змінні з новим ім'ям:

1. Змінна, що буде звертатися до колекції dataset_2.
2. Змінна, що рахує середнє (медіанне) значення й вибирає всі значення менші за -15 до попередньої змінної.
3. За допомогою updateMask використати маску.
4. Створити нові візуальні параметри з новим кольором ("d52bff").
5. Додати на мапу шар з назвою Mask Aug.
6. Додати нову змінну для завантаження нової маски.

Результат повинен мати такий вигляд:

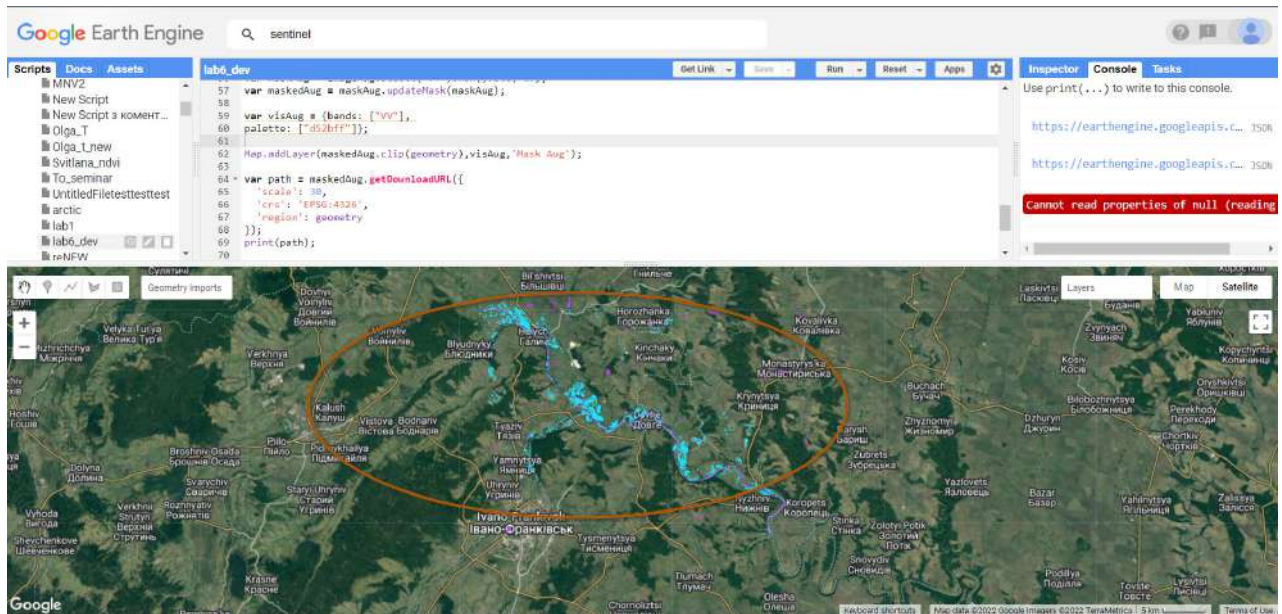


Рис. 138. Маски на мапі

За наведеним нижче QR-кодом ви можете переглянути відео виконання практичної роботи.



Рис. 139. Відео виконання практичної роботи

Готовий код можна відкрити за лінком:

https://code.earthengine.google.com/?scriptPath=users%2Fgis_rs%2Fcourse_MAN%3ALab_6

Вивантаження узагальнених середньостатистичних даних у форматі таблиці в результаті аналізу зміни температурних показників на території України

Ситуація

Ми з вами живемо в епоху глобального потепління і щоденно стикаємося з його наслідками, як-от: частіші й суворіші засухи, підняття рівня океану, танення льодовиків і т. д.

У цій практичній роботі ми проаналізуємо динаміку одного з показників, яким характеризується глобальне потепління, – температури. Зробимо це на прикладі території України та окремих частин м. Києва.

Клімат – багаторічний режим погоди, який ґрунтується на багаторічних метеорологічних спостереженнях.

Зміна клімату – суттєва і тривала зміна погодних умов.

Глобальне потепління – поступове підвищення температури поверхні Землі та океану.

Основною причиною глобального потепління вчені називають парниковий ефект. Гази (переважно вуглекислий газ CO₂), які виділяються під час спалювання викопного палива, затримують сонячне тепло і спричиняють підвищення температури планети.

Швидкі зміни клімату, які ми нині спостерігаємо, викликані постійним ростом кількості нафти, газу, вугілля та інших видів викопного палива, які людство використовує для обігріву своїх будинків, роботи підприємств і транспорту.

На сьогодні у світі приблизно на 1,2 градуса Цельсія тепліше, ніж було у XIX ст., а кількість CO₂ в атмосфері зросла на 50%¹.

Завдання

Візуально порівняти дві мапи температур на територію України. Створити графік температури поверхні для щільно забудованої території і для території з лісовим покривом. Вивантажити значення мінімальних і максимальних температур на ці два типи територій у табличному форматі CSV.

Алгоритм виконання завдання:

1. Підвантажити датасет міжнародних кордонів країн *LSIB 2017* і «взяти» з нього межі України.
2. Відфільтрувати датасет глобальних щомісячних кліматичних показників і водного балансу земної поверхні *TerraClimate* за 1970 р., додати параметри візуалізації і відобразити на мапі.
3. Відфільтрувати датасет *TerraClimate* за 2020 р., додати параметри візуалізації і відобразити на мапі.
4. Підвантажити глобальний датасет температури поверхні Землі *MOD11A1.006 Terra*.
5. Створити дві зони інтересу для щільно забудованої території міста і для лісу.
6. Створити функцію перетворення масштабу та переведення значень температури з Кельвінів у градуси Цельсія.
7. Створити змінні з редукторами мінімального і максимального значення, з редуктором суми та середнього.
8. Створити графік мінімального і максимального значення температури поверхні для забудованої території.
9. Завантажити графік у форматі таблиці CSV на локальний комп'ютер.

¹ URL: https://uk.wikipedia.org/wiki/Вуглекислий_газ_в_атмосфері_Землі

Покрокова інструкція

1. Підвантажити датасет міжнародних кордонів країн **LSIB 2017** і «взяти» з нього межі України.

Для початку потрібно відкрити **Code Editor (Редактор коду)** і підвантажити датасет **LSIB 2017: Large Scale International Boundary Polygons, Simplified (LSIB 2017: Великомасштабні міжнародні кордони країн у полігонах, спрощені)**.

У рядку пошуку наберіть **LSIB** та імпортуйте зазначений вище датасет.

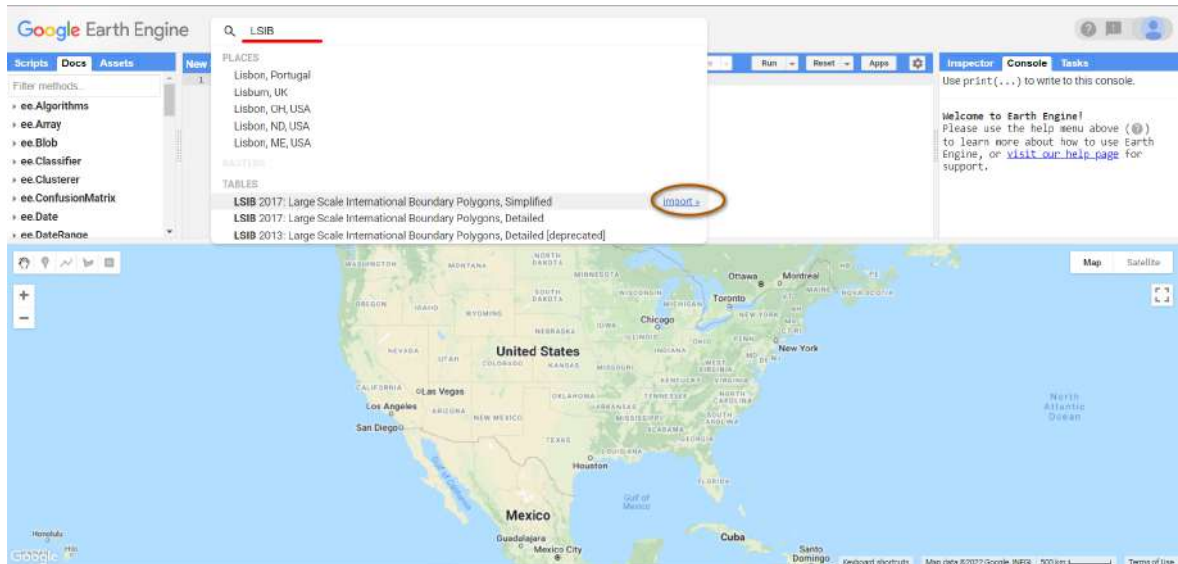


Рис. 140. Імпорт датасету

Далі потрібно перейменувати створений датасет і задати йому назву **LSIB**.

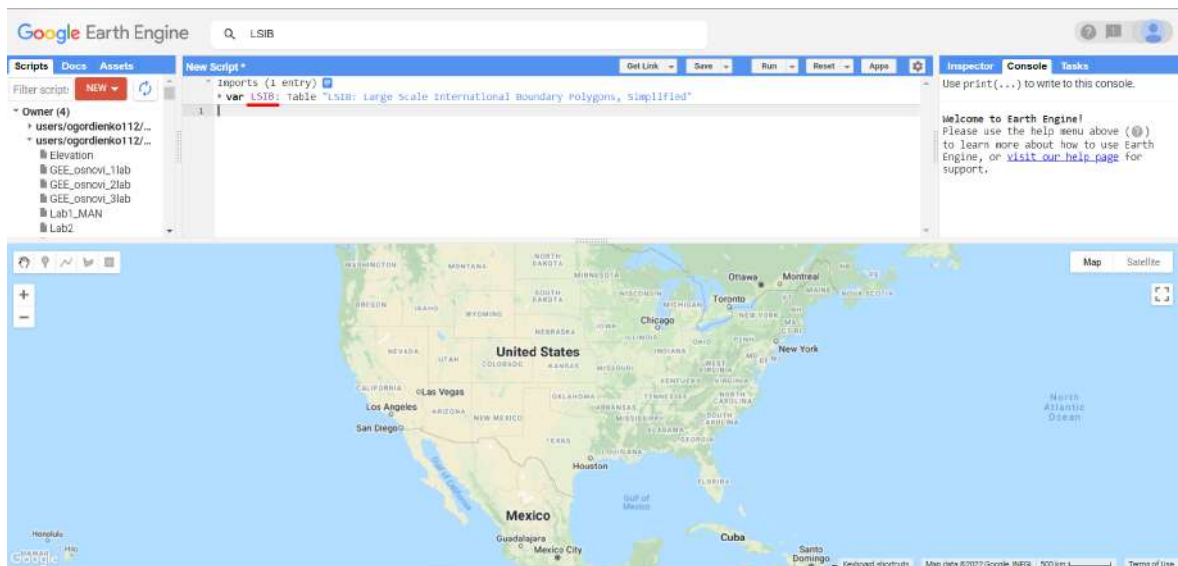


Рис. 141. Імпортований датасет

Наступним кроком буде створити змінну, яка вибере з датасету межі України, і відцентрувати мапу за цими межами.

```
var Ukraine = LSIB.filter(ee.Filter.  
eq('country_na', 'Ukraine'));  
Map.centerObject(Ukraine,5);
```

Змінна **Ukraine** звертається до датасету **LSIB 2017** та фільтрує значення до кордонів України
Центрування мапи по змінній **Ukraine**
з екстендом **5**

Натиснути **Run** (Запуск скрипту).

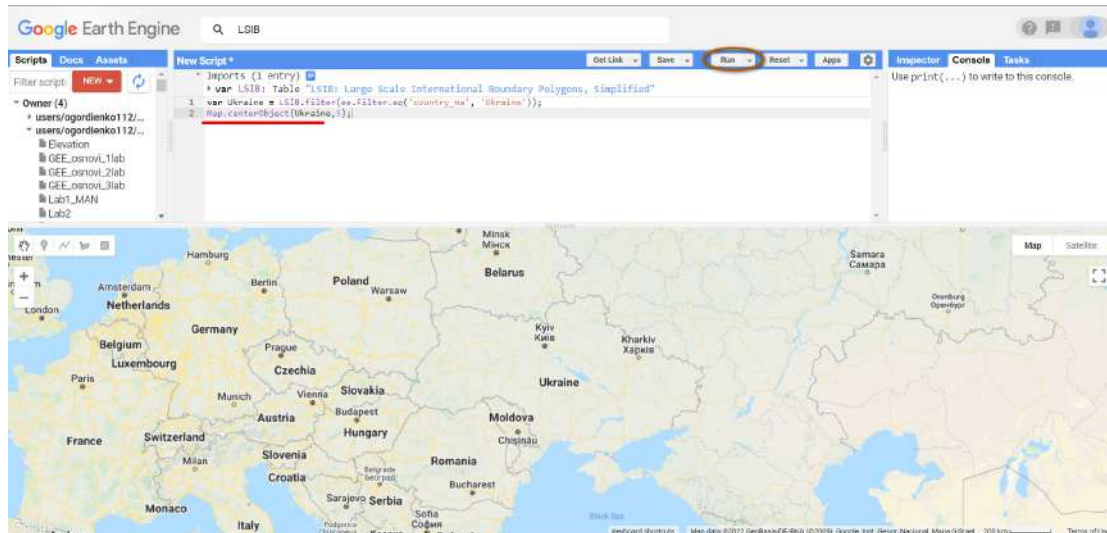


Рис. 142. Додавання на мапу

2. Відфільтрувати датасет глобальних щомісячних кліматичних показників і водного балансу земної поверхні **TerraClimate** за 1970 р., додати параметри візуалізації і відобразити на мапі.

На цьому кроці ми підвантажимо датасет за допомогою коду і не будемо імпортувати його до вкладки **Imports** (*Імпорт*). Необхідно відфільтрувати датасет **TerraClimate: Monthly Climate and Climatic Water Balance for Global Terrestrial Surfaces, University of Idaho** (*TerraClimate: Щомісячний клімат і кліматичний водний баланс для глобальних земних поверхонь, Університет Айдахо*) за датами протягом 1970 р.

Цей датасет має в собі дані про мінімальну і максимальну температуру, евапотранспірацію (сумарне випаровування – кількість вологи, що переходить в атмосферу у вигляді пари), швидкість вітру тощо. Він зберігає в собі дані з 1958 р. по 2021 р. і поки що не оновлюється.

```
var dataset = ee.ImageCollection('IDAHO_EPSCOR/TERRACLIMATE')
  .filter(ee.Filter.date('1970-01-01', '1970-12-31'));
```

Змінна **dataset** фільтрує значення з датасету **TerraClimate** за датами за весь 1970 р.

Тепер потрібно задати канал, який буде вибрано із цього датасету. Це буде канал з максимальними значеннями. Зробити це можна за допомогою коду:

```
var maximumTemperature =
  dataset.select('tmmx');
```

Змінна **maximumTemperature** вибирає канал з темом **'tmmx'**

Наступним кроком є задавання візуальних параметрів.

Візуалізація відбувається за допомогою кодування HEX. Кожен код відповідає кольору. Палітра кольорів, яка була використана для цієї візуалізації:



Задайте візуальні параметри за допомогою коду:

```
var maximumTemperatureVis = {
  min: 8.0,
  max: 16.0,
  palette: [
    '1a3678', '2955bc', '5699ff', '8dbae9', 'acd1ff',
    'caebff', 'e5f9ff',
    'fdffb4', 'ffe6a2', 'ffc969', 'ffa12d', 'ff7c1f',
    'ca531a', 'ff0000',
    'ab0000'
  ],
};
```

Візуальні параметри з мінімальним значенням **8.0** та максимальним значенням **16.0**

Потрібно помножити зображення на значення масштабу, щоб отримати реальні значення температури. Ці значення можна отримати в описі самого датасету під час його пошуку, що представлено на двох наступних рисунках.

У рядку пошуку введіть **IDAHO_EPSCOR/TERRACLIMATE**. Натисніть і перейдіть до вкладки **Bands** (Канали).

TerraClimate: Monthly Climate and Climatic Water Balance for Global Terrestrial Surfaces, University of Idaho

Resolution: 4638.3 meters

Name	Description	Min*	Max*	Units	Scale
aet	Actual evapotranspiration, derived using a one-dimensional soil water balance model	0	3140	mm	0.1
def	Climate water deficit, derived using a one-dimensional soil water balance model	0	4548	mm	0.1

Dataset Availability: 1958-01-01T00:00:00 -

Dataset Provider: University of California Merced

Collection Snippet: ee.ImageCollection("IDAHO_EPSCOR/TERRACLIMATE")

Name	Description	Min*	Max*	Units	Scale
srad	Downward surface shortwave radiation	0	5477	W/m ²	0.1
swe	Snow water equivalent, derived using a one-dimensional soil water balance model	0	32767	mm	0
tmmn	Minimum temperature	-770	387	°C	0.1
tmmx	Maximum temperature	-670	576	°C	0.1
vap	Vapor pressure	0	14749	kPa	0.001
vpd	Vapor pressure deficit	0	1113	kPa	0.01
vs	Wind-speed at 10m	0	2923	m/s	0.01

* = Values are estimated

Рис. 143. Атрибути шару

Це вся інформація про доступні канали, їх короткий опис, мінімальні і максимальні значення для даних із каналів та коефіцієнт масштабу.

Тепер можна відобразити мапу. Зробити це можна за допомогою коду:


```
Map.addLayer(maximumTemperature.  
median().clip(Ukraine).multiply(0.1),  
maximumTemperatureVis, 'Maximum Temperature  
1970',true);
```

Результат на мапу. З відфільтрованої змінної maximumTemperature береться медіанне значення, множиться на 0.1, застосовуються параметри візуалізації maximumTemperatureVis, назва 'Maximum Temperature 1970', відображення true

Натиснути **Run** (Запуск скрипту).

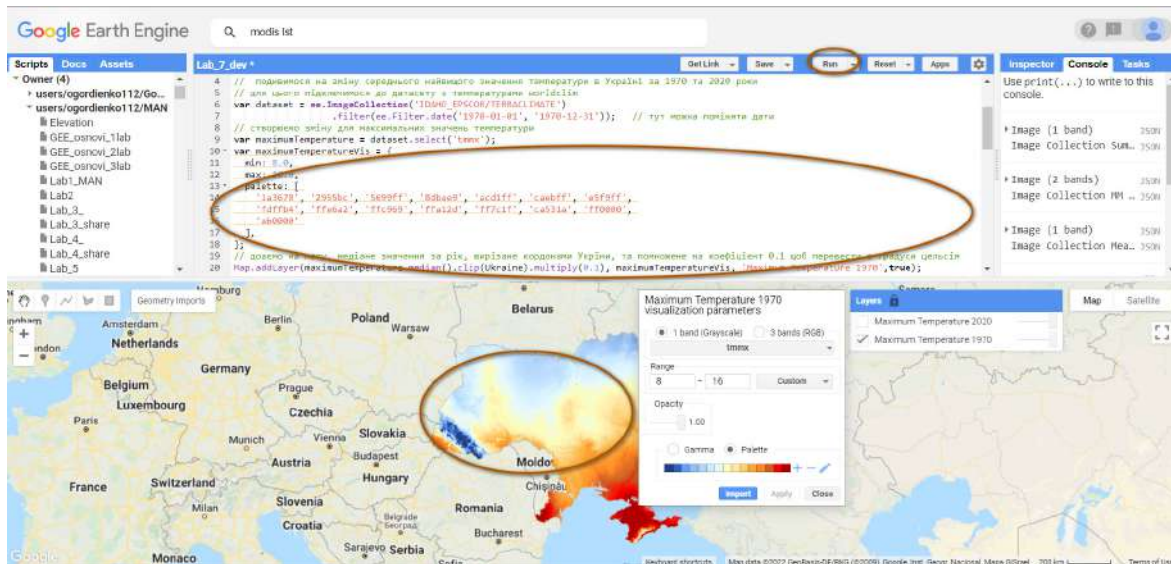


Рис. 144. Налаштування відображення

3. Відфільтрувати датасет **TerraClimate** за 2020 р., додати параметри візуалізації і відобразити на мапі.

Тепер необхідно створити нову змінну з тим самим датасетом (TerraClimate: Monthly Climate and Climatic Water Balance for Global Terrestrial Surfaces, University of Idaho) для 2020 р.

Скопіюйте код:

```
var dataset2 = ee.ImageCollection('IDAHO_  
EPSCOR/TERRA CLIMATE')  
  
.filter(ee.Filter.date('2020-01-01', '2020-12-  
31'));
```

Змінна dataset2 фільтрує значення з датасету **TerraClimate** за датами за весь 2020 р.

І так само, як у попередньому кроці, виберіть канал із середнім максимальним значенням.

```
var maximumTemperature2  
= dataset2.select('tmmx');
```

Змінна maximumTemperature2 вибирає канал з тегом 'tmmx'

Додайте новий шар на мапу з тими самими візуальними параметрами, які були застосовані для попереднього датасету.

```
Map.addLayer(maximumTemperature2.  
median().clip(Ukraine).multiply(0.1),  
maximumTemperatureVis, 'Maximum  
Temperature 2020',true);
```

Результат на мапу. З відфільтрованої змінної maximumTemperature2 береться медіанне значення, множиться на 0.1, застосовуються параметри візуалізації з попередніх кроків maximumTemperatureVis, назва 'Maximum Temperature 2020', відображення true

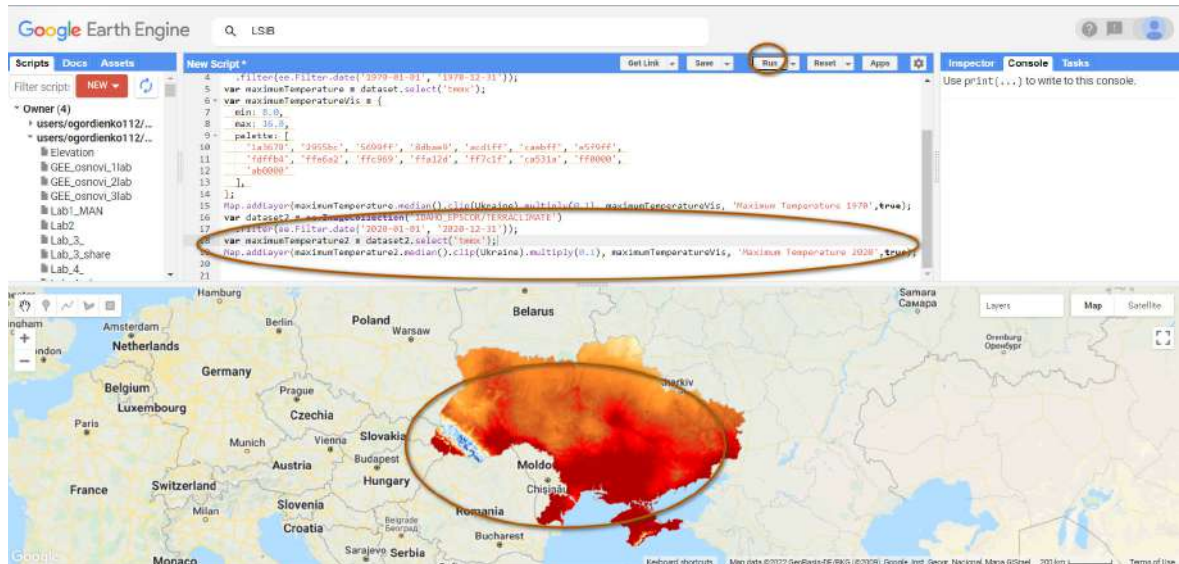



Рис. 145. Додавання на карту

Ці датасети візуалізують різницю середніх річних максимальних температур для України з різницею в 50 років. Палітра  побудована так, що від темно-синього до темно-червоного збільшується показник середньої максимальної температури.

? Завдання для перевірки

Проаналізуйте за допомогою *Inspector* (Інспектор) свій населений пункт. Яка температура поверхні Землі була в 1970 і 2020 рр.?

4. Підвантажити глобальний датасет температури поверхні Землі **MOD11A1.006 Terra**. Наступним кроком буде підвантаження іншого датасету та аналіз його даних. Продукт **MOD11A1 V6 Terra** забезпечує моніторинг добової температури поверхні землі (LST) і значення випромінювання з просторовим розрізненням 1200 × 1200 км. Датасет має дані від початку 2000 р. до сьогоднішнього дня. Наберіть у рядку пошуку MOD11A1.006 і виберіть датасет з назвою **MOD11A1.006 Terra Land Surface Temperature and Emissivity Daily Global 1km** (*MOD11A1.006 Terra Температура Поверхні Землі та Випромінювальна здатність, Глобальний 1 км*). Зверніть увагу, що нам потрібен датасет із тегом Daily. Натисніть **Import** (Імпорт).



Рис. 146. Пошук шару

Тепер потрібно змінити назву імпортованого датасету на *modis_lst*.

✔ **Важливо!** У назвах змінних нема змоги використовувати пробіл, тому замість нього скористайтеся нижнім підкресленням.

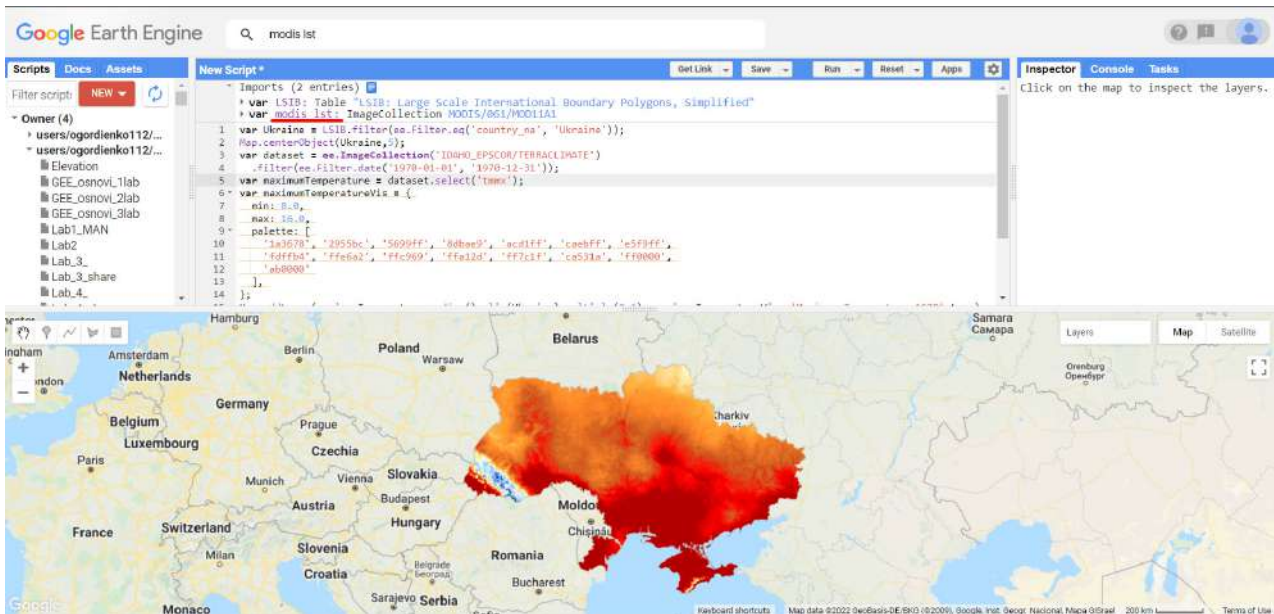


Рис. 147. Зміна назви

5. Створити дві зони інтересу для щільно забудованої території міста і для лісу.

У практичній роботі для наочності зміни температури поверхні на різних типах місцевості ми порівняємо дві ділянки: житловий масив Троєщина в Києві і ліс поряд із цим масивом. Наберіть у рядку пошуку *Kiiv* і виберіть перший результат.

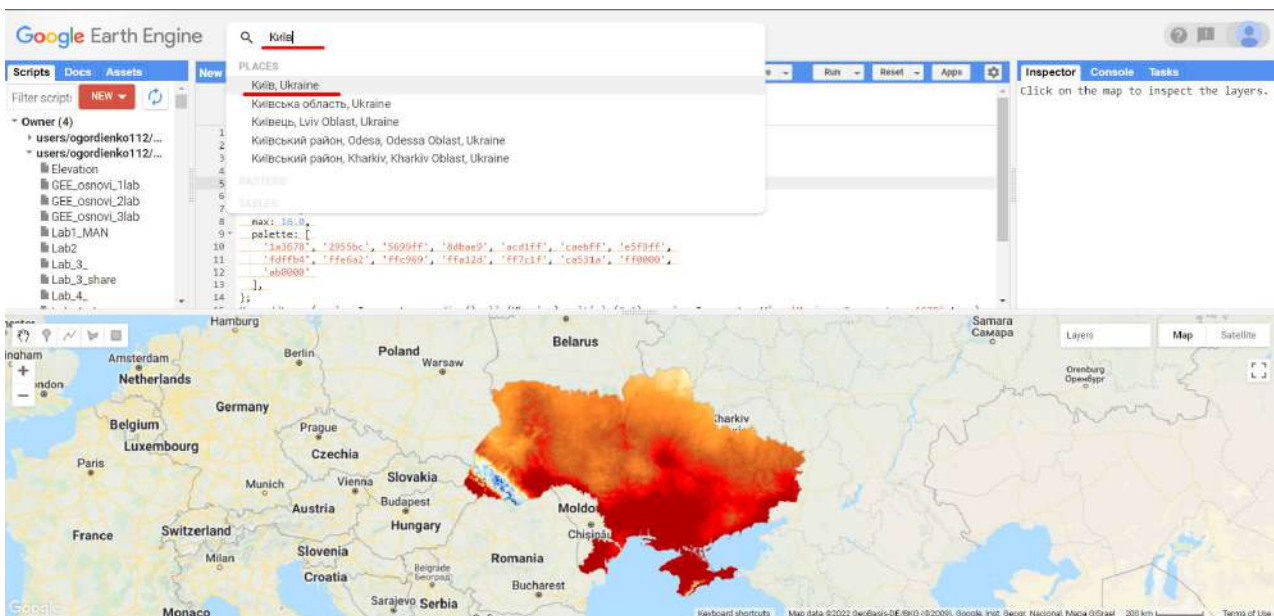


Рис. 148. Пошук місця

Натисніть клавішу *Enter*, і мапа наблизиться до Києва. Для зручності використання вимкніть шари у вкладці *Layers (Шари)*.

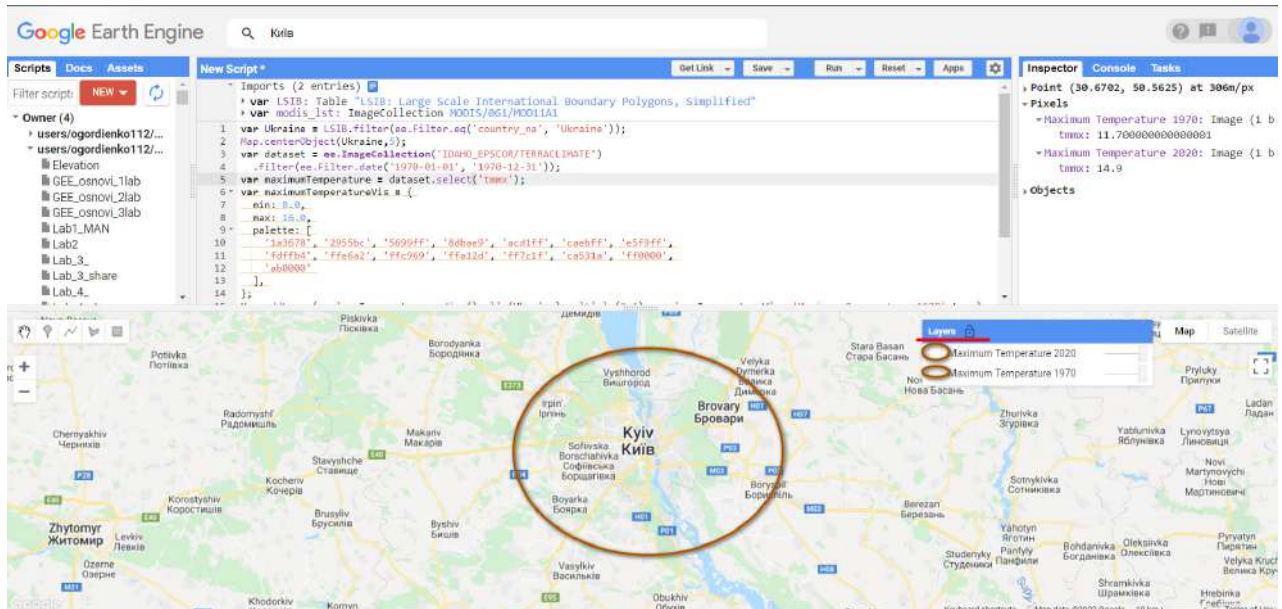


Рис. 149. Наближення до місця

Наблизилися до Троещини. Вона розташована на лівому березі Дніпра, це північний схід Києва. Створіть нову геометрію з назвою `study_area` за допомогою інструменту *Geometry instruments* (Інструменти геометрії).

✔ **Важливо!** Якщо далі будуть проблеми у відображенні мінімальних і максимальних значень, це може статися через розмір та місцезоположення геометрії. Змініть тут розмір та місцезоположення геометрії.

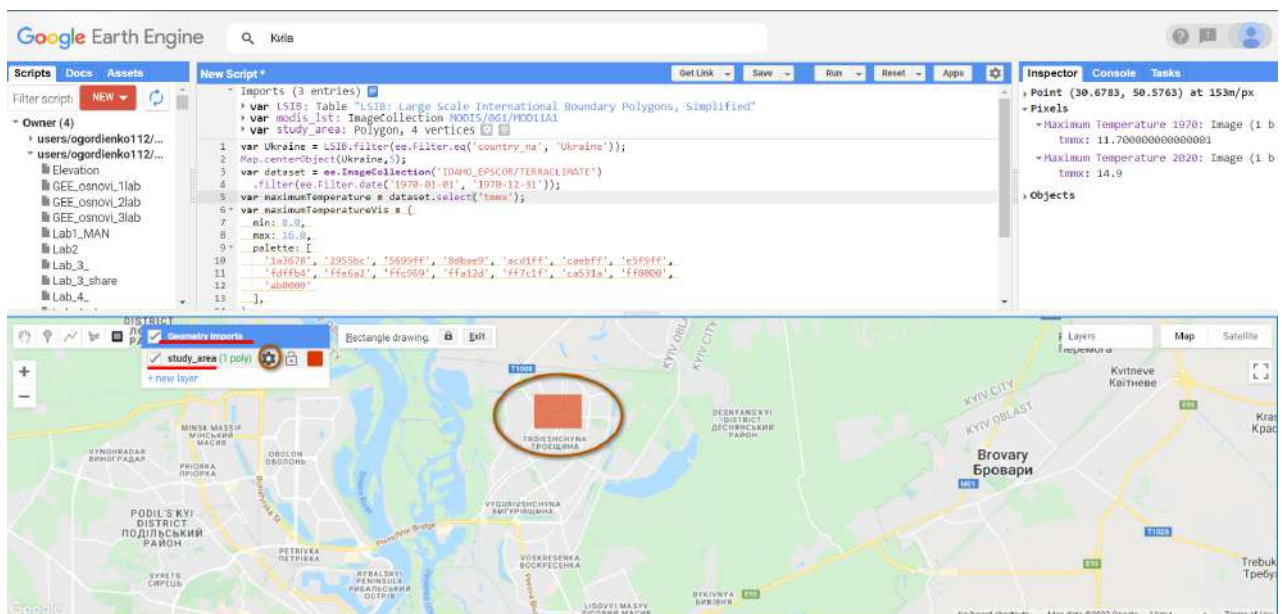


Рис. 150. Створення векторного шару

На мапі видно велику ділянку лісу поряд із Троещиною. Створіть на ній нову геометрію з назвою `study_area_2`.

Створюйте нову геометрію за допомогою [+ new layer](#) приблизно такого самого розміру, як і першу.

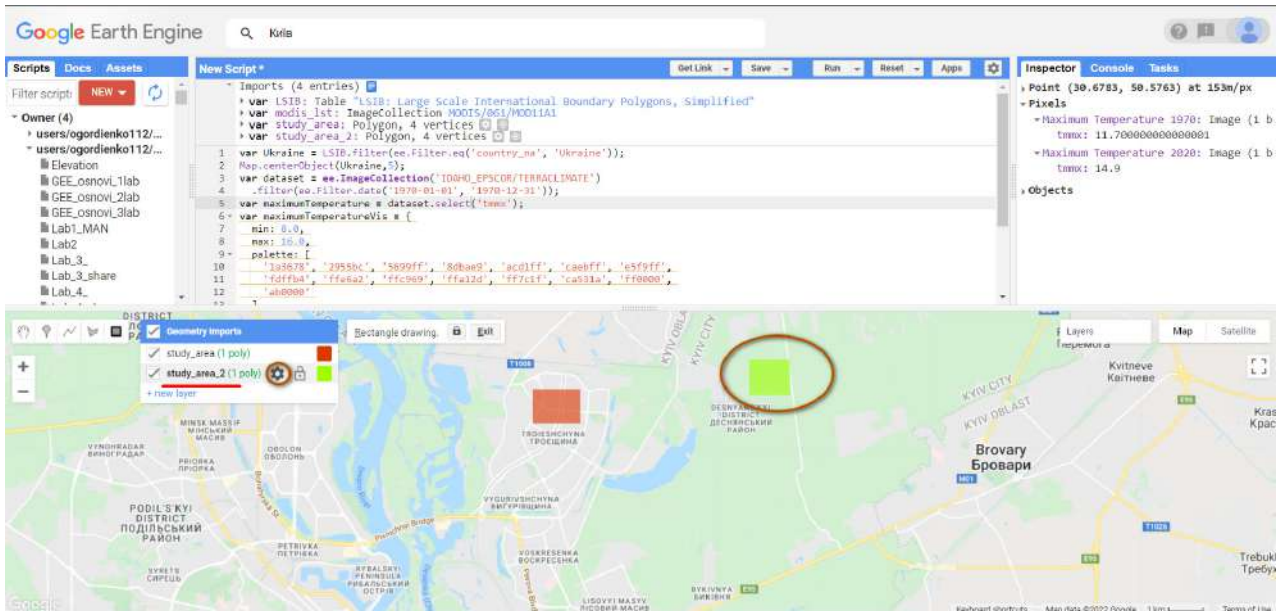


Рис. 151. Створення іншого векторного шару

Після створення нових геометрій можна продовжити роботу зі скриптом.

Наступний код фільтрує колекцію (*MOD11A1.006 Terra Land Surface Temperature and Emissivity Daily Global 1km*) за вказаними датами і вибирає значення температури поверхні.

```
var filtered = modis_lst.  
filterBounds(study_area)  
.filterDate('2010-01-01','2020-12-31')  
.select('LST_Day_1km');
```

Змінна *filtered* фільтрує дані за геометрією *study_area* і за датою за період із 2010 по 2020 рр. вибирає значення *'LST_Day_1km'*

6. Створити функцію перетворення масштабу та переведення значень температури з Кельвінів у градуси Цельсія.

Тепер дані в колекції потрібно перевести до справжніх показників у Цельсіях. Для цього створюється функція, яка множить значення на 0,02 (значення масштабу), віднімає 272, щоби перевести Кельвіни в градуси Цельсія, і бере на вхід значення початку та закінчення дат у системі UNIX, щоби коректно відображалися дати. Мітка часу встановлюється на номінальний час отримання зображення для окремих сцен. Для тимчасових композицій встановлюється номінальний початковий період.

```
var LST = filtered.map(function(img){  
return img.multiply(0.02).subtract(273)  
.copyProperties(img,['system:time_  
start','system:time_end']);  
});
```

Функція *LST* додає за допомогою *map* повернений вираз
Зображення множиться на 0.02 ділиться на 273 та додає значення *'system:time_start'*

7. Створити змінні з редукторами мінімального і максимального значення, з редуктором суми та середнього.

Наступними трьома змінними можна дослідити різні редуктори, а саме: редуктор суми, редуктор мінімального і максимального значення, редуктор середнього значення.

Редуктор об'єднує всі зображення в колекції в одне зображення, яке представляє, наприклад, мінімальне, максимальне, середнє зображення.

Докладніше про редуктори дивіться тут: https://developers.google.com/earth-engine/guides/reducers_image_collection.

<pre> var sum_ic = LST.reduce(ee.Reducer.sum()); print(sum_ic, 'Image Collection Sum Reducer'); var minMax_ic = LST.reduce(ee.Reducer.minMax()); print(minMax_ic, 'Image Collection MM Reducer'); var mean_ic = LST.reduce(ee.Reducer.mean()); print(mean_ic, 'Image Collection Mean Reducer'); </pre>	<p>sum_ic створює нове зображення, за допомогою ee.Reducer.sum. сумує значення</p> <p>print результат на екран, minMax_ic створює нове зображення, за допомогою ee.Reducer.minMax створює мінімальні і максимальні значення</p> <p>print результат на екран, mean_ic створює нове зображення, за допомогою ee.Reducer.mean створює середнє значення</p> <p>print результат на екран</p>
--	---

Натисніть **Run** (Запуск скрипту).

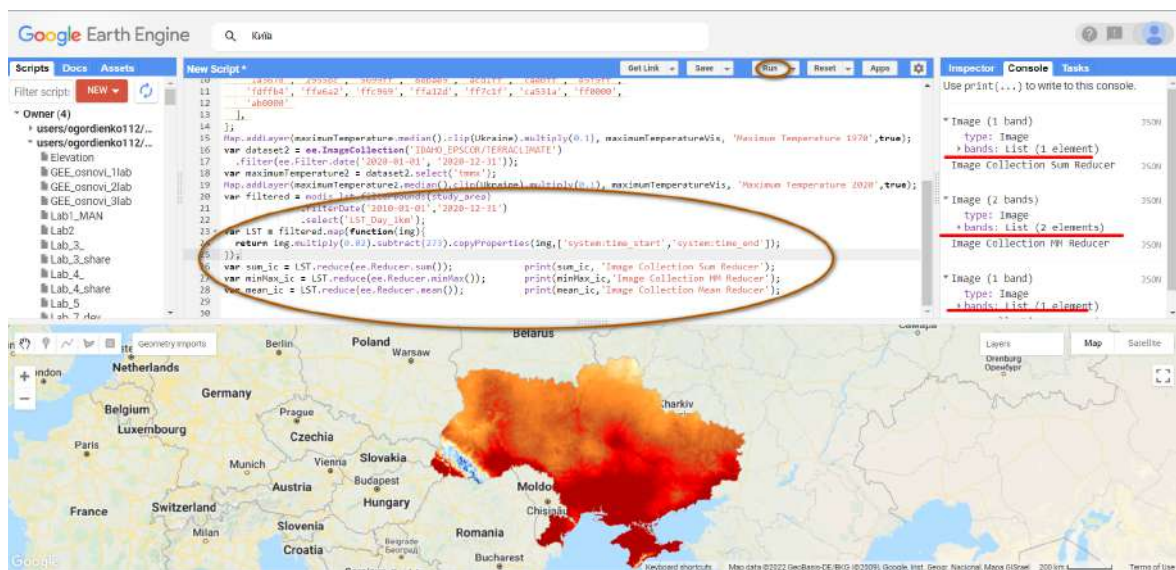


Рис. 152. Виведення даних у консоль

🔍 Завдання для перевірки

Дослідіть виведену інформацію в консолі. Скільки значень має редуктор мінімального і максимального значення?

8. Створити графік мінімального і максимального значення температури поверхні для заведеної території.

Наступним кроком буде створення графіка для мінімальних і максимальних значень для території інтересу study_area (Троещина).

<pre> var chart_lst = ui.Chart.image.series({ imageCollection: LST, region: study_area, reducer: ee.Reducer.minMax() }); .setOptions({ hAxis: {title: 'Month'}, vAxis: {title: 'Temperature (°C)', pointSize: 1, dataOpacity: 1, }); print(chart_lst.setChartType('ScatterChart')); </pre>	<p>chart_lst створює графік LST колекція на вхід region територія для графіка reducer редуктор зменшення</p> <p>setOptions налаштування графіка</p> <p>hAxis підпис ординат</p> <p>vAxis підпис абсцис</p> <p>pointSize розмір точки</p> <p>dataOpacity прозорість даних</p> <p>print результат у консоль, тип графіка 'ScatterChart'</p>
--	---

Натисніть **Run** (Запуск скрипту).

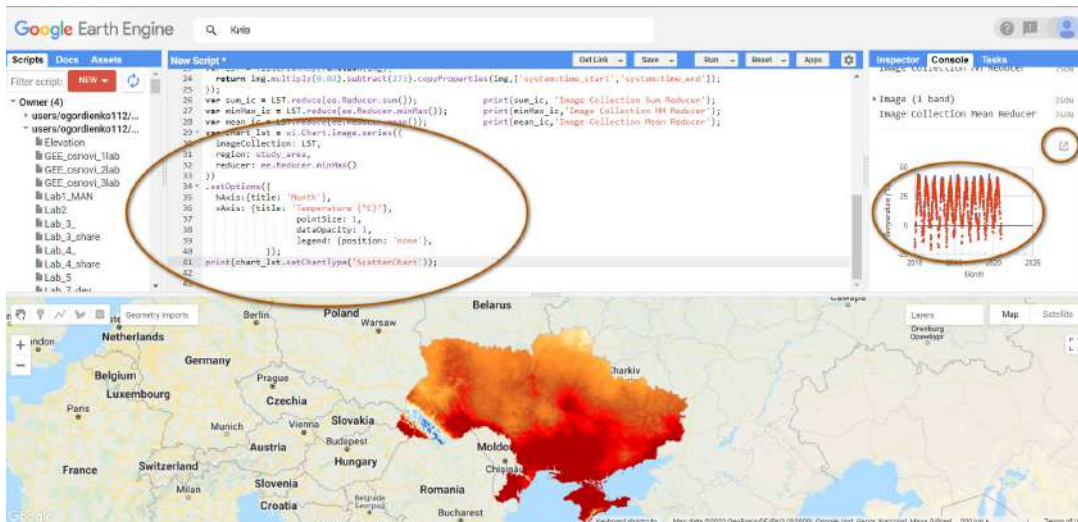


Рис. 153. Виведення таблиці в консоль

Зараз у консолі відображаються мінімальні і максимальні значення для температури поверхні на основі знімків MODIS. Мінімальні значення на графіку показуються червоним кольором, максимальні – синім.

9. Завантажити дані у форматі таблиці CSV на локальний комп'ютер.

Для того щоб зберегти ці дані в табличному вигляді, розгорніть графік за допомогою кнопки . У новому вікні відкриється графік. Для збереження у форматі CSV натисніть на кнопку

Download CSV

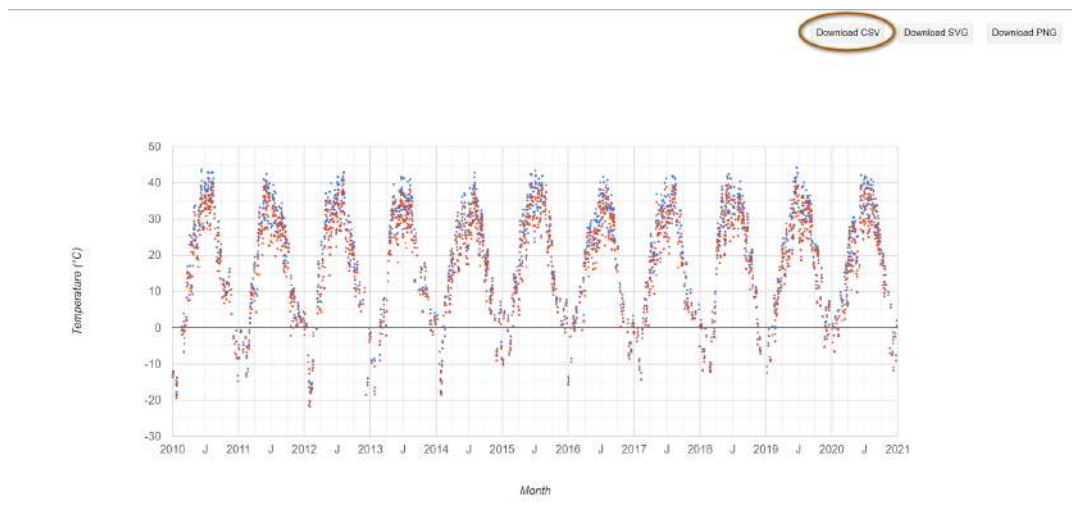


Рис. 154. Виведення таблиці в іншу вкладку

Графік збережеться в табличному вигляді.

Завдання для перевірки

Вставте у створений графік лінію тренду, щоби побачити, як змінюється температура в Україні.

Зробити це можна за допомогою рядка:

```
trendlines: {  
  0: {color: 'CC0000'}  
},
```

Вставити цей рядок потрібно в налаштування графіка

✔ **Важливо!** В редакторі коду зараз обчислюються мінімальні та максимальні значення для території study_area (Троєщина) за період від початку 2010 р. до кінця 2020 р.

? Завдання для перевірки

Для другої території інтересу study_area_2 (лісу) побудуйте за допомогою редуктора мінімального і максимального значення графік та завантажте таблицю зі значеннями у форматі CSV. Дослідіть графіки на предмет того, як змінилася температура в самому місті й у лісі

	A	B	C
1	system:time_start	LST_Day_1km_max	LST_Day_1km_min
2	Jan 1, 2010		
3	Jan 2, 2010		
4	Jan 3, 2010	-12.98	-13.7
5	Jan 4, 2010		
6	Jan 5, 2010	-12	-12.46
7	Jan 6, 2010		
8	Jan 7, 2010		
9	Jan 8, 2010		
10	Jan 9, 2010		
11	Jan 10, 2010		
12	Jan 11, 2010		
13	Jan 12, 2010		
14	Jan 13, 2010		

Рис. 155. Вигляд даних у таблиці

? Додаткове завдання для перевірки

На основі вивантажених даних побудуйте такий самий графік в ексель

За наведеним нижче QR-кодом ви можете переглянути відео виконання практичної роботи.



Рис. 156. Відео виконання практичної роботи

Готовий код можна відкрити за лінком:

https://code.earthengine.google.com/?scriptPath=users%2Fgis_rs%2Fcourse_MAN%3ALab_7

Розрахунки спектральних індексів NDVI та NDWI для моніторингу змін стану рослинного покриву на Кримському півострові

Ситуація

Захоплення частини суверенної території України – півострова Крим російською федерацією у 2014 р. спричинило низку проблем, у тому числі в питанні водозабезпечення анексованої території. Україна забезпечувала до 85% потреб Криму в прісній воді через Північно-Кримський канал, що з'єднує головне русло Дніпра з півостровом. Після анексії Криму росією поставки води на півострів призупинили¹. З перекриттям цього каналу припинилася подача води місцевому населенню; місцеве сільське господарство стикнулося із серйозними проблемами. Так, за даними Андрія Рюмшина, т. зв. міністра сільського господарства (2016–2019 рр.) маріонеткового уряду півострова Крим, площа зрошуваних земель Криму скоротилася із 397 тисяч до 13 тисяч гектарів². За даними видання «Крим. Реалії» перші суттєві проблеми з водою виникли в окупованому Сімферополі і найближчих до нього регіонах влітку 2020 р. На тлі виснаження водоймищ місто, а також Сімферопольський район з вересня отримують воду за графіком: по три години вранці і ввечері¹.

Запаси води в Криму поповнюються із водосховищ природного стоку і підземних джерел. За заявами екологів, регулярне використання води з підземних джерел призвело до засолення ґрунту на півострові.

Загалом відсутність водопостачання до Кримського півострова з материкової частини України мала би призвести до деградації рослинного покриву й осушення резервуарів питної води, але чи так це насправді? Якщо так, то якою мірою? Розібратися в цьому комплексному питанні може допомогти супутниковий моніторинг Землі й індекси NDVI та NDWI зокрема.

NDVI (Normalized Difference Vegetation Index) – це кількісний показник активної (здатної до фотосинтезу) біомаси, який залежить від того, як рослини відбивають і поглинають електромагнітні хвилі різної довжини, а саме червоний та ближній інфрачервоний спектри. Здорова рослина активно поглинає червону частину спектра і відбиває ближню інфрачервону. Отже, формула така: $NDVI = (NIR - RED)/(NIR + RED)$.

NDWI (Normalized Difference Water Index) – це кількісний показник, який використовується для ідентифікації на супутниковому знімку об'єктів відкритих водних дзеркал. $NDWI = (Green - NIR)/(Green + NIR)$.

Завдання

Визначити усереднене значення NDVI та NDWI за п'ять місяців (із травня по вересень) 2013–2021 рр. у межах Кримського півострова, простежити тенденції за лінією тренду.

Алгоритм виконання завдання:

1. Завантажити датасет Landsat 8.
2. Відфільтрувати супутникові знімки Landsat 8 за хмарністю.
3. Створити векторний шар адміністративних меж Кримського півострова.
4. Порахувати й візуалізувати індекс NDVI для Кримського півострова за період із 2013 р. по 2022 р.
5. Створити графік усереднених значень NDVI за п'ять місяців (травень – вересень) із 2013 р. по 2021 р.
6. Створити графік усереднених значень NDWI за п'ять місяців (травень – вересень) із 2013 р. по 2021 р.

¹ URL: <https://www.ukrinform.ua/rubric-crimea/3203911-v-okupovanomu-krimu-miliut-vodoshovisa-voda-znikae.html>

² URL: <https://ru.krymr.com/a/28148230.html>

Покрокова інструкція

1. Завантажити датасет **Landsat 8**.

Відкрийте **Code Editor** (Редактор коду): <https://code.earthengine.google.com/>. Для початку відкрийте Code Editor і наберіть у пошуку тег *Landsat 8*. Коли з'являться назви датасетів, натисніть **more** (більше).

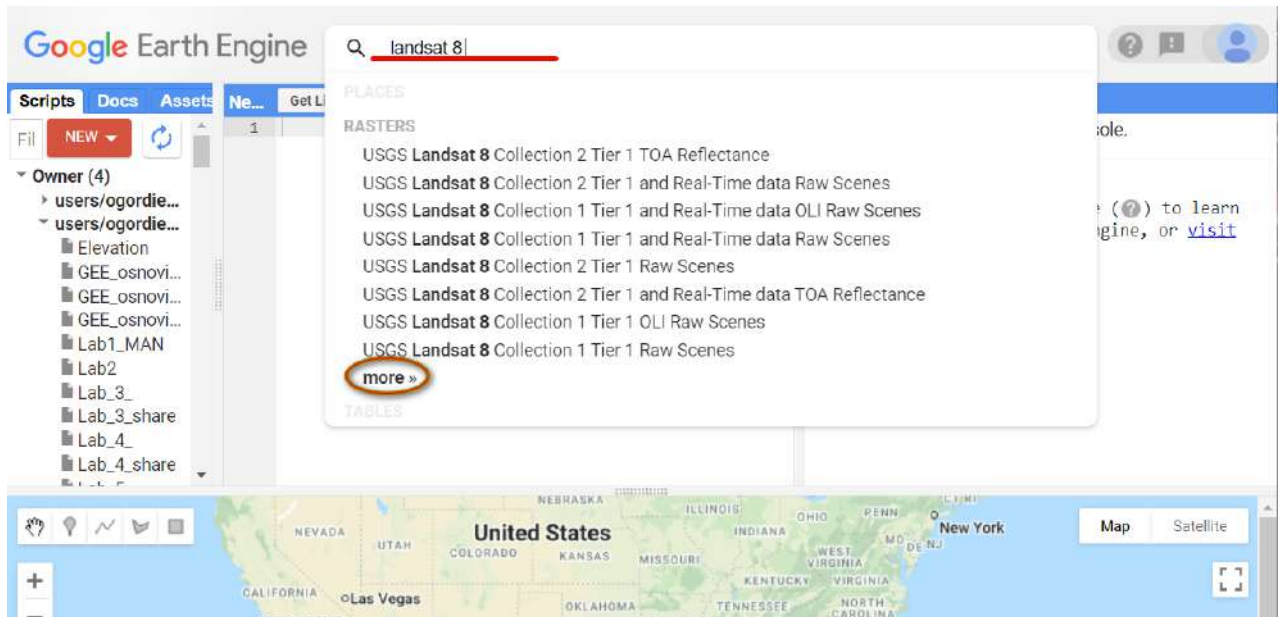


Рис. 157. Пошук датасету

Знайдіть потрібний нам набір даних «*USGS Landsat 8 Collection 1 Tier 1 TOA Reflectance*» і натисніть на нього.

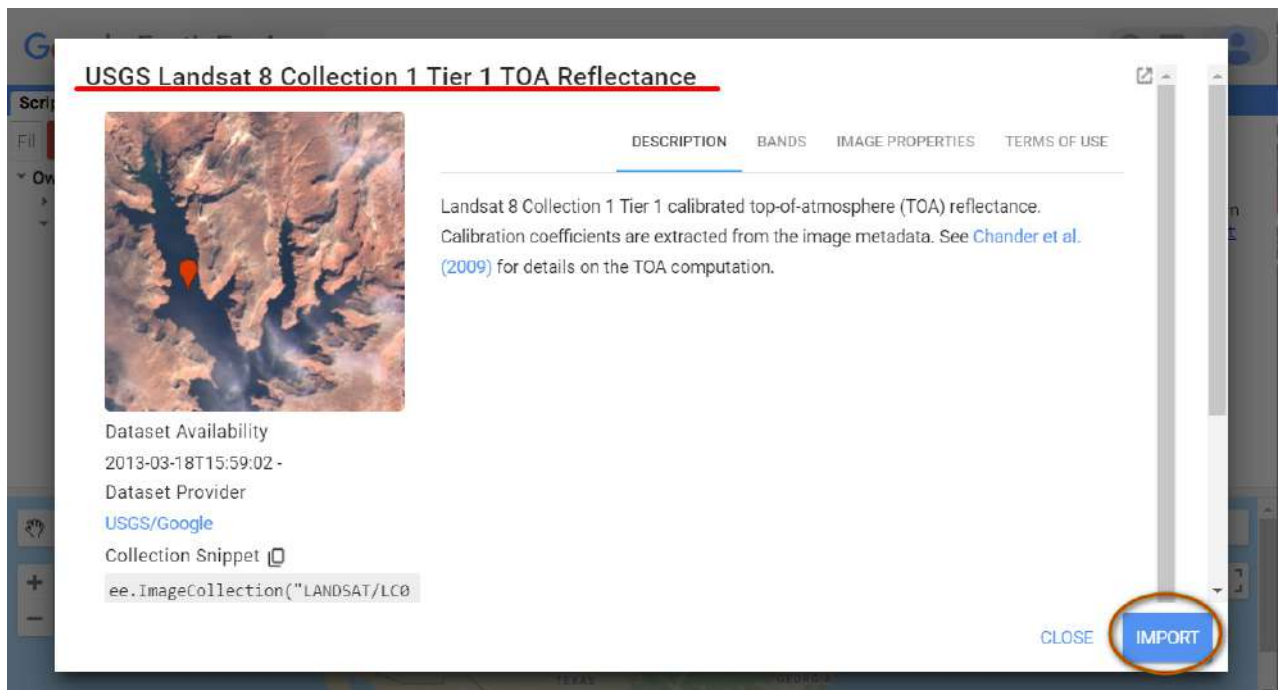


Рис. 158. Імпорт датасету

Після цього закрийте вікно пошуку за допомогою **Close** (Закрити).

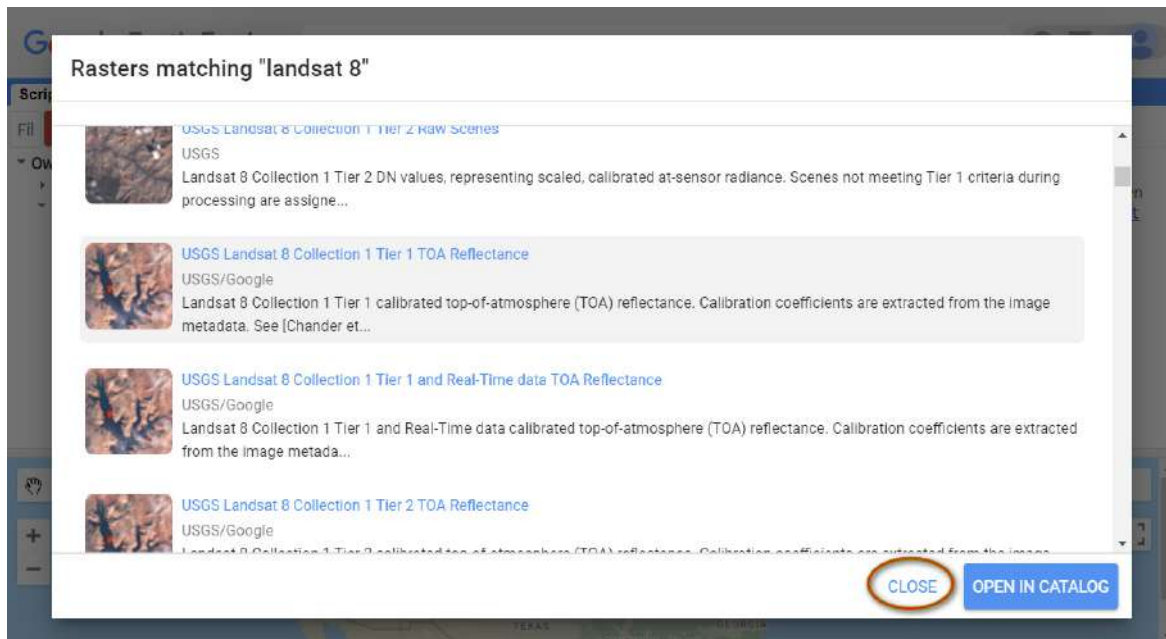


Рис. 159. Закриття вікна пошуку

Тепер потрібно перейменувати датасет. Задайте йому ім'я "L8", натиснувши на назву `imageCollection`. Вона підсвітиться блакитним кольором, а це означає, що ви можете її змінювати.

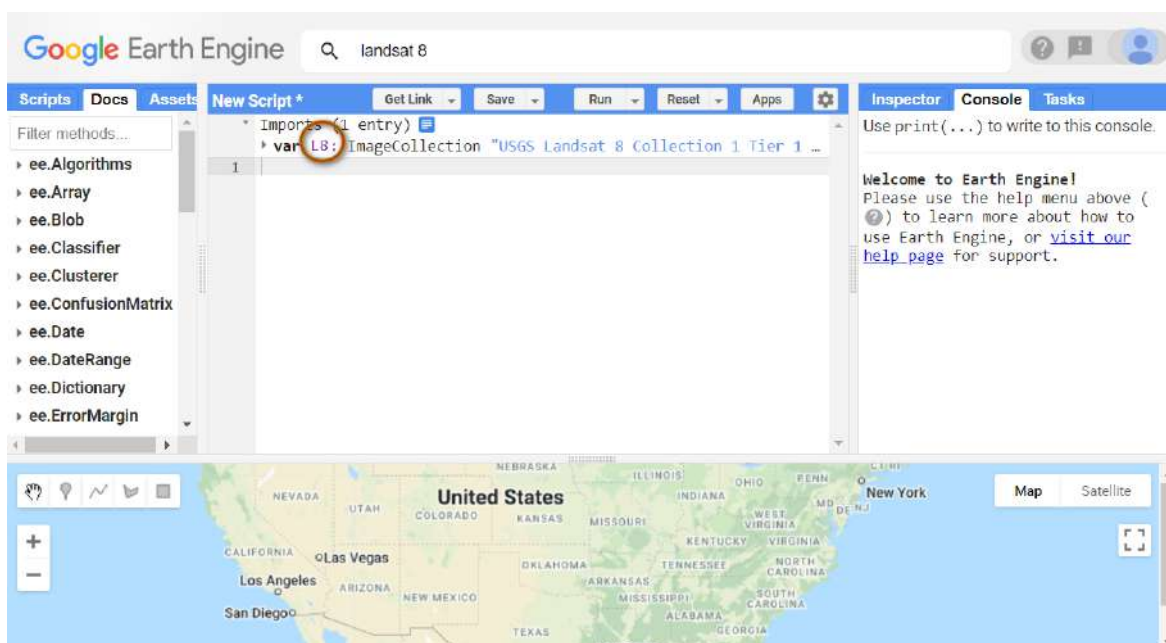


Рис. 160. Зміна назви датасету

2. Відфільтрувати супутникові знімки Landsat 8 за хмарністю.

Доданий нами в першому кроці датасет Landsat 8 надає нам доступ до усіх доступних знімків цього супутника, проте деякі оптичні знімки можуть мати високу хмарність, тому вони малоінформативні в дослідженні стану рослинного покриву. Необхідно їх відфільтрувати і залишити лише ті, де хмарність займає менше 15% площі знімка.

```
var L8 = ee.ImageCollection
('LANDSAT/LC08/C01/T1_TOA')
.filter(ee.Filter.lt('CLOUD_COVER_LAND',15));
```

Ця змінна звертається до датасету Landsat 8 за допомогою `ee.ImageCollection` та відфільтровує за допомогою `ee.Filter.lt` усі знімки, які мають хмарність понад 15%

3. Створити векторний шар адміністративних меж Кримського півострова.

Для цього в рядку пошуку вписуємо ключове слово **FAO**, що означає *Food and Agriculture Organization*.

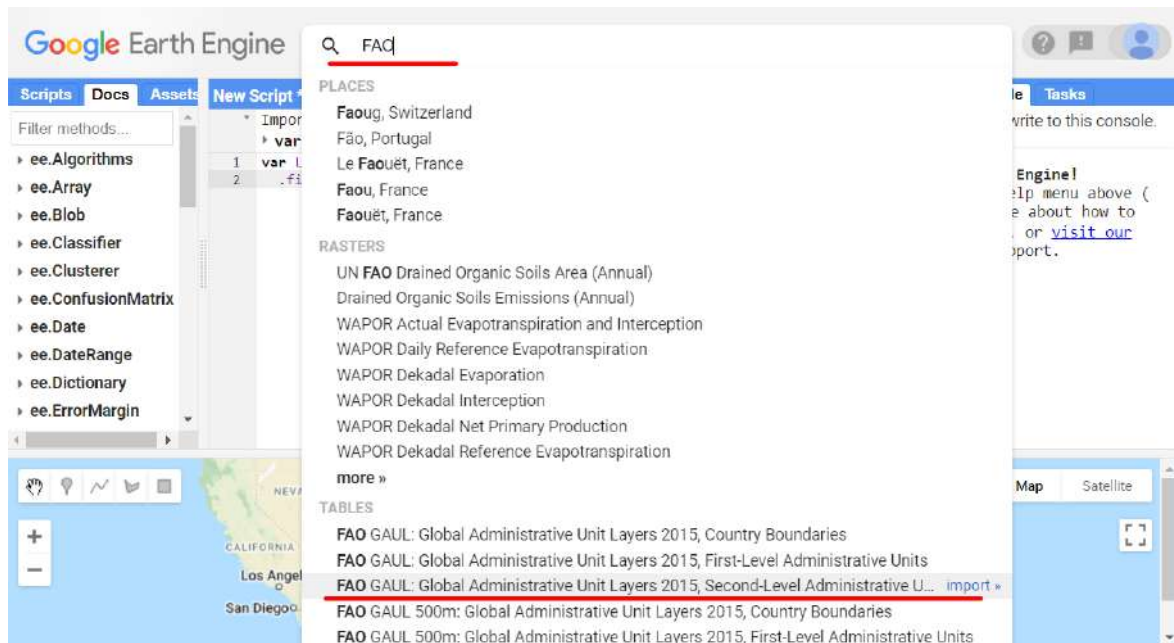


Рис. 161. Пошук датасету

У випадковому списку є «**FAO GAUL: Global Administrative Unit Layers 2015, First-Level Administrative Units**».

✔ **Важливо!** На цьому етапі не потрібно імпортувати датасет до робочого середовища.

Глобальні рівні адміністративних одиниць (GAUL: The Global Administrative Unit Layers) – база даних, ідеєю створення якої є збір, упорядкування й поширення достовірної і загально-доступної інформації про адміністративні одиниці різного рівня для всіх країн світу (здебільшого межі країн та їх найбільших адміністративних одиниць).

Відкриється додаткова інформація про цю базу даних. Копіюємо назву бази даних «`ee.FeatureCollection("FAO/GAUL/2015/level1")`». Вона нам буде потрібна для створення нової змінної, яку ми назвемо «Крым_feature». Натиснути **Run** (Запуск скрипту).

<pre>var Крым_feature = ee.FeatureCollection("FAO/GAUL/2015/level1") Map.addLayer(Крым_feature);</pre>	<p><code>ee.FeatureCollection</code> звертається до датасету з межами адміністративних одиниць першого рівня</p>
--	--

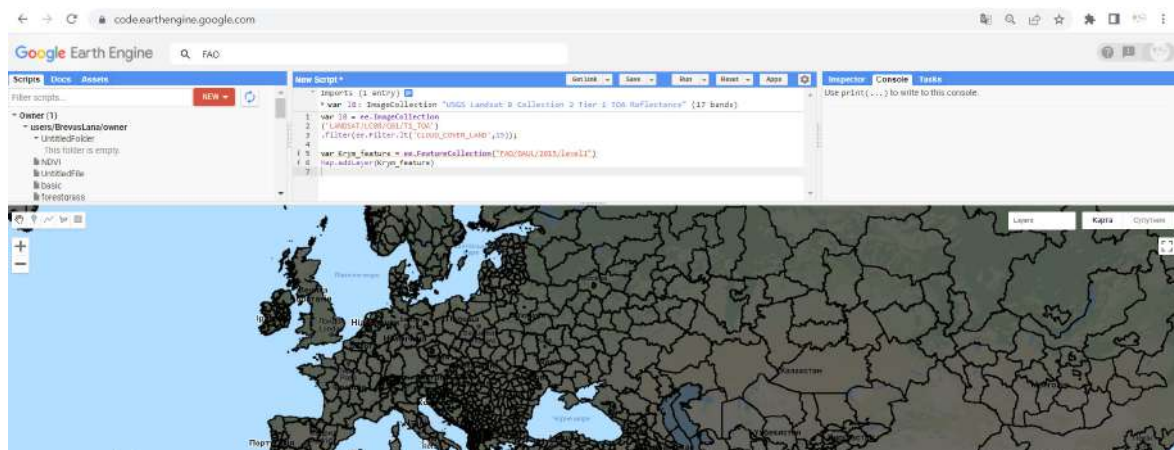


Рис. 161. Пошук датасету

У такий спосіб нам підвантажаться шари адміністративних одиниць першого рівня усіх країн світу. Оскільки ми маємо дослідити лише півострів Крим, то нам потрібно відфільтрувати решту адміністративних одиниць так, щоб на мапі залишилася лише територія інтересу.

Для цього насамперед нам треба подивитися, як звертатися до адміністративної одиниці Кримський півострів у цій базі даних. Для цього скористаємося інструментом *Inspector* (*Інспектор*) і натиснемо лівою клавішею миші на ту територію, назву якої хочемо дізнатися.

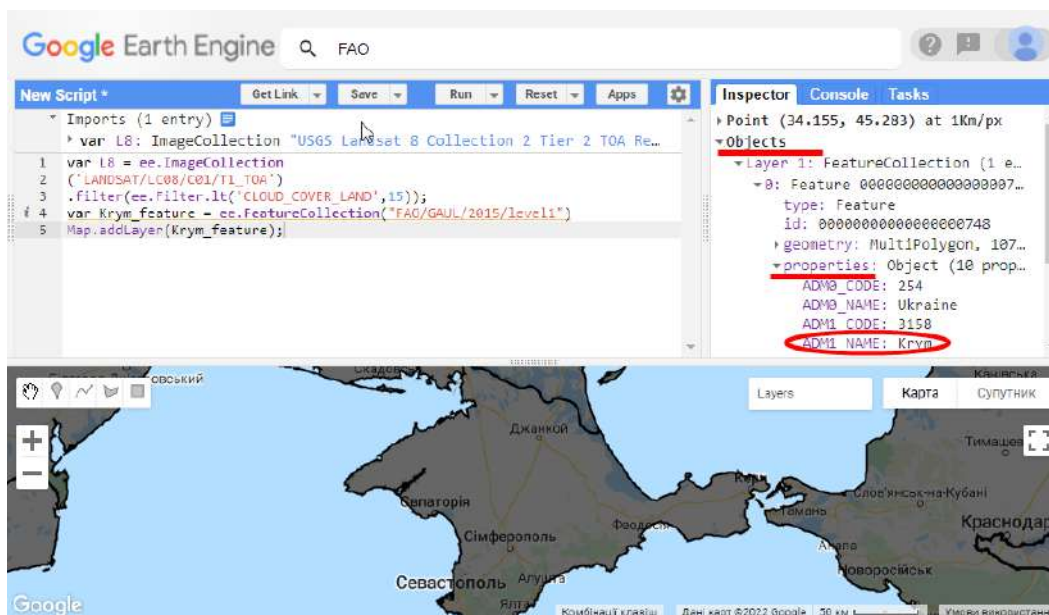


Рис. 163. Наближення до необхідної території

Отже, бачимо назву нашого Кримського півострова в базі даних FAO – «Krym», вона міститься у групі «ADM1_NAME». Тепер необхідно відфільтрувати решту адміністративних одиниць і залишити на мапі лише полігон адміністративних меж півострова. Для цього використовуємо метод фільтра *Filter*.

✔ **Важливо!** Наступний фільтр потрібно додати до змінної `var Krym_feature = ee.FeatureCollection("FAO/GAUL/2015/level1")` щоб вона набула вигляду:
`var Krym_feature = ee.FeatureCollection("FAO/GAUL/2015/level1").filter(ee.Filter.eq('ADM1_NAME','Krym'));`

<code>.filter(ee.Filter.eq('ADM1_NAME', "Krym"));</code>	<code>ee.Filter.eq</code> фільтрує значення, яке в полі 'ADM1_NAME' дорівнює Krym
--	--

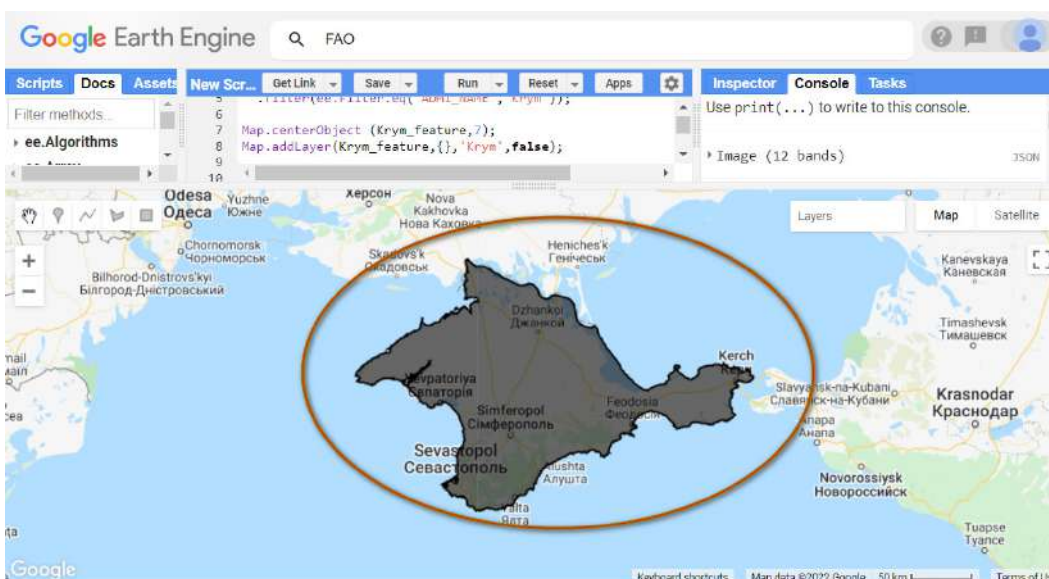


Рис. 164. Фільтрування даних

Натисніть **Run** (Запуск скрипту). Відмасштабуємо мапу так, аби під час запуску скрипту півострів Крим був у центрі картографічного вікна. Для цього впишіть рядок коду:

<code>Map.centerObject (Krym_feature,7);</code>	Центрування до попереднього об'єкта
---	-------------------------------------

✔ **Важливо!** Під час виконання завдання, аби не трапилося помилки, цей рядок краще закоментувати.

Тобто потрібно, щоб він набув вигляду:

`//Map.centerObject (Krym_feature,7);`

Рівень масштабування, який можна вибрати для цієї змінної, перебуває в діапазоні від 0 до 20. Спробуйте замінити цифру 7 на будь-яке натуральне число в цьому проміжку і виберіть, яке вам найбільше до вподоби.

У таблиці шарів створимо і назвемо шар адміністративних меж Кримського півострова – «Крым», для цього скористаємося змінною `Map.addLayer`:

<code>Map.addLayer(Krym_feature,{'Kрым'},false);</code>	Шар на мапу з назвою "Крым"
---	-----------------------------

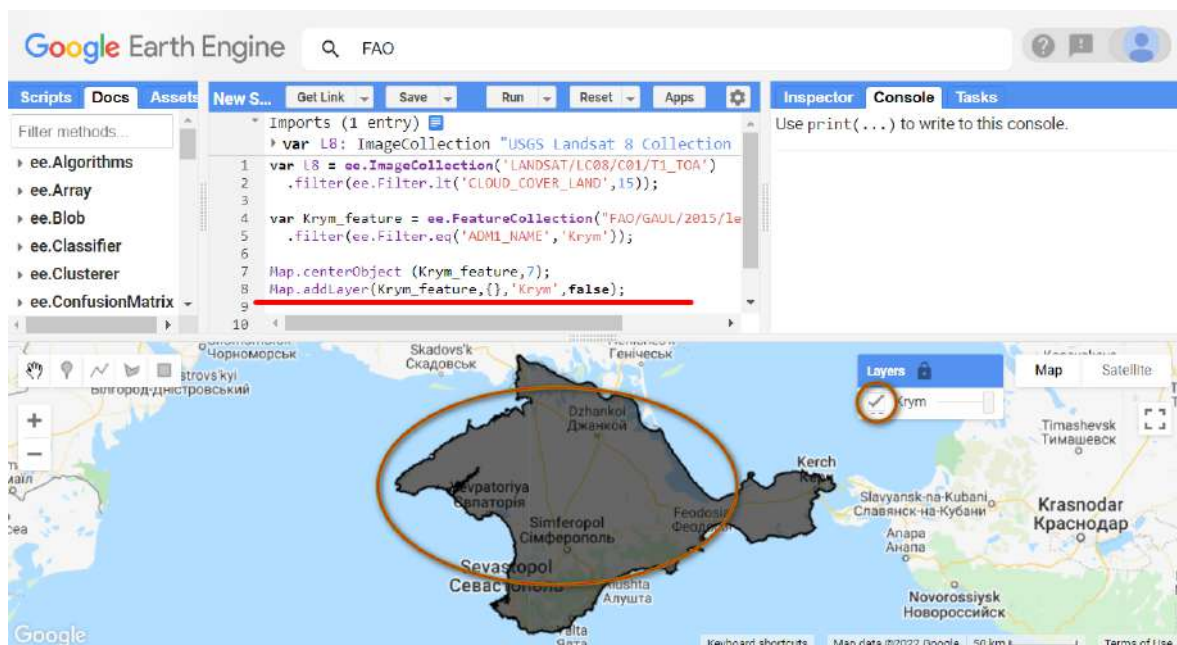


Рис. 165. Зміна відображення даних

4. Порахувати й візуалізувати індекс NDVI для Кримського півострова за період із 2013 р. по 2022 р.

Для цього створимо знімок усереднених значень за період від початку 2013 р. до серпня 2022 р. Ця колекція отримає середнє значення в кожному каналі.

<pre>var image = ee.Image(L8.filterBounds(Krym_feature) .filterDate('2013-01-01', '2022-08-31') .sort('CLOUD_COVER') .median() .clip(Krym_feature)); print(image);</pre>	<p><code>ee.Image</code> робить з колекції одне зображення</p> <p><code>filterDate</code> фільтрує дати за значенням</p> <p><code>sort</code> сортує за хмарністю</p> <p><code>median</code> середнє значення за вказану дату</p> <p><code>clip</code> обрізає за полігоном <code>Krym_feature</code></p> <p>Експортує результат на мапу</p>
--	--

Наступний крок – порахувати показник NDVI на цей знімок усереднених значень, який ми отримали кроком вище. Для цього одним з найбільш простих способів є використання змінної «`image.normalizedDifference(bandName1, bandName2)`», де потрібно вказати назви каналів, які в Landsat 8 відповідають за ближній інфрачервоний та червоний діапазони електромагнітного випромінювання.

На сайті геологічної служби США <https://www.usgs.gov/media/images/landsat-8-band-%20designations> ви можете перевірити, які діапазони електромагнітного спектра фіксує супутник Landsat 8.

Отже, Band 4 фіксує червоний спектр, а Band 5 – ближній інфрачервоний спектр електромагнітного випромінювання. Відповідно вносимо ці значення в рядок коду для визначення NDVI.

<pre>var ndvi = image. normalizedDifference(['B5', 'B4']). rename('NDVI');</pre>	<p><code>normalizedDifference</code> визначає нормалізовану різницю між 'B5' та 'B4' перейменовує на 'NDVI'</p>
--	---

Далі нам потрібно візуалізувати отримані дані на мапі. Для цього ми використовуємо змінну «`coloredVis`»:

<pre>var coloredVis = { min: -1, max: 1, palette: ['blue', 'white', 'green'], }; Map.addLayer(ndvi, coloredVis, 'NDVI image');</pre>	<p>Візуальні параметри з мінімальним та максимальним значенням</p> <p><code>palette</code> вказує на кольори</p> <p>Результат на мапу</p>
--	---

Натиснути **Run** (Запуск скрипту).

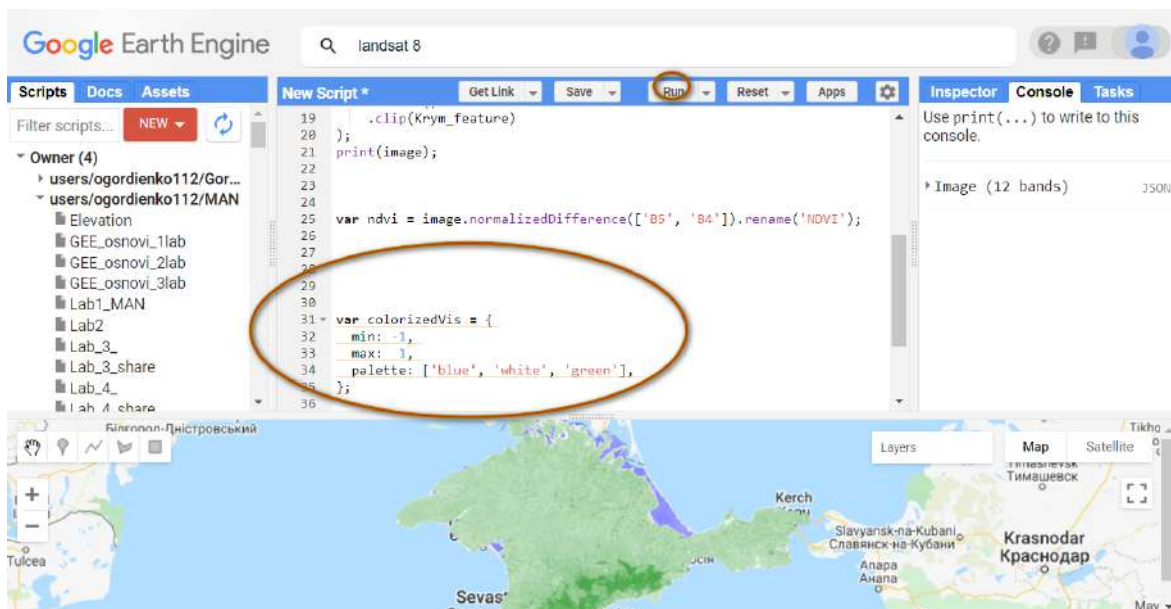


Рис. 166. Налаштування відображення даних

Більш детальну інформацію про коди кольорів ви можете знайти за покликанням: https://en.wikipedia.org/wiki/List_of_colors:_G%E2%80%93M.

У GEE кодування кольорів відбувається за допомогою кодування HEX. Шукайте в таблиці за покликанням саме цю колонку. Для того аби замінити наявний колір, наприклад білий на

червоний, потрібно замінити в коді 'white' на номер червоного – "ff0000". Замініть палетку кольорів на власний розсуд.

5. Створити графік усереднених значень NDVI за п'ять місяців (травень – вересень) із 2013 р. по 2021 р.

Спершу нам потрібно створити функцію, яка додає значення NDVI до знімків Landsat 8:

<pre>function addNDVI(image) { var ndvi = image.normalizedDifference(['B5', 'B4']).rename('ndvi').float(); return image.addBands(ndvi); } var with_ndvi = L8.map(addNDVI);</pre>	<p>Функція з назвою addNDVI на вхід об'єкт для розрахунку нормалізованої різниці. Перейменовуємо за допомогою rename на 'ndvi', значення з плаваючою точкою float Повертає розрахований об'єкт map застосовує функцію addNDVI до колекції L8</p>
--	--

Тепер – створити змінну, яка з попереднього рядка значень «var with_ndvi» відфільтрує NDVI за необхідними нам часовими проміжками:

<pre>var coll_1 = with_ndvi.select('ndvi') .filter(ee.Filter.calendarRange(5, 9, 'month')) .filterDate('2013-06-01', '2022-05-01');</pre>	<p>З колекції with_ndvi вибирається канал з назвою 'ndvi' calendarRange набуває двох значень: 5 початок та 9 кінець Фільтрація за датами</p>
---	--

Щоб створити графік, нам потрібно скористатися методом «ui.Chart.image.series», який дає змогу генерувати графік із певної колекції знімків:

<pre>var chart_1 = ui.Chart.image.series({ imageCollection: coll_1.select('ndvi'), region: Krym_feature, reducer: ee.Reducer.mean(), scale: 100, }).setOptions({ lineWidth: 1, colors: ['green'], title: 'Усереднене NDVI з травня по вересень з 2013 по 2021 роки', vAxis: {title: 'NDVI'}, hAxis: {title: "", format: 'YYYY-MMM'}, trendlines: { 0: {color: 'CC0000'} }, }); print(chart_1);</pre>	<p>Створення графіка за допомогою ui.Chart.image.serie Вибирається coll_1 зі значенням 'ndvi' Krym_feature геометрія розрахунку reducer Приводить колекцію до одного зображення</p> <p>scale роздільна здатність пікселя Налаштування графіка lineWidth вага лінії на графіку colors колір лінії на графіку title Назва графіка</p> <p>vAxis назва по осі абсцис hAxis дати по осі ординат додавання лінії тренду</p> <p>Результат у консоль</p>
--	--

Натисніть **Run** (Запуск скрипту).

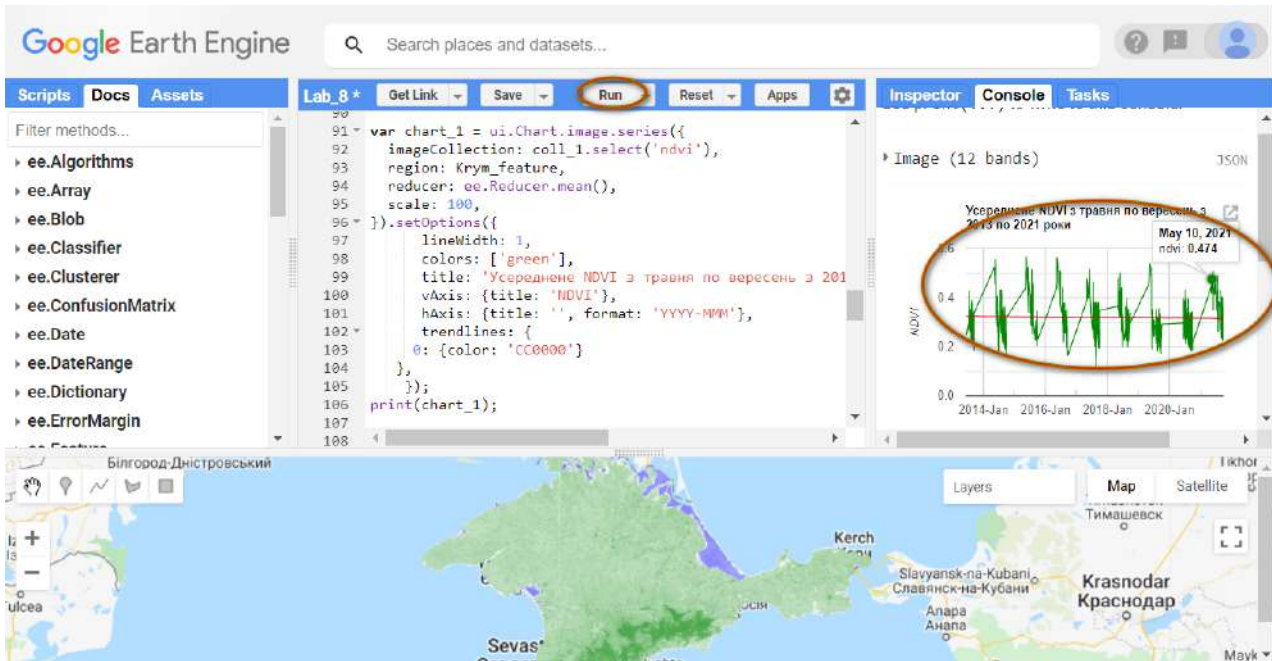


Рис. 167. Виведення графіка в консоль

? Завдання для перевірки

Змініть фільтр хмарності у третьому рядку коду на 2%. Як змінився графік? Як ви гадаєте, чому?

6. Створити графік усереднених значень NDWI за п'ять місяців (травень – вересень) із 2013 р. по 2021 р.

Як і в першому кроці, спершу нам потрібно створити функцію, яка додає значення NDWI до знімків Landsat 8.

? Завдання для перевірки

Формула для розрахунку NDWI = (Green – NIR)/(Green + NIR). Який канал у Landsat 8 фіксує діапазон зеленого кольору електромагнітного спектра?

```
function addNDWI(image) {
  var ndwi = image.
  normalizedDifference(['B5', 'B3']).
  rename('ndwi').float();
  return image.addBands(ndwi);
}
var with_ndwi = L8.map(addNDWI);
```

Функція з назвою addNDWI на вхід об'єкт для розрахунку нормалізованої різниці. Перейменовуємо за допомогою rename на 'ndwi', значення з плаваючою точкою float

Повертає розрахований об'єкт map застосовує функцію addNDWI до колекції L8

Тепер нам потрібно створити змінну, яка з попереднього рядка значень «var with_ndwi» відфільтрує NDWI за необхідними нам часовими проміжками:

```
var coll_2 = with_ndwi.select('ndwi')
  .filter(ee.Filter.calendarRange(5, 9, 'month'))
  .filterDate('2013-06-01', '2022-05-01');
```

З колекції with_ndwi вибираємо канал з назвою 'ndwi' calendarRange набуває двох значень: 5 початок та 9 кінець фільтрація за датами

Щоб створити графік, нам потрібно скористатися методом «`ui.Chart.image.series`», який дає змогу генерувати графік із певної колекції знімків:

<pre>var chart_2 = ui.Chart.image.series({ imageCollection: coll_2.select('ndwi'), region: Krym_feature, reducer: ee.Reducer.mean(), scale: 100, }).setOptions({ lineWidth: 1, title: 'Усереднене NDWI з травня по вересень з 2013 по 2021 роки', vAxis: {title: 'NDWI'}, hAxis: {title: '', format: 'YYYY-MMM'}, trendlines: { 0: {color: 'CC0000'} }, }); print(chart_2);</pre>	<p>Створення графіка за допомогою <code>ui.Chart.image.series</code></p> <p>Вибирається <code>coll_2</code> зі значенням <code>'ndwi'</code> Krym_feature геометрія розрахунку <code>reducer</code> узагальнює колекцію до одного зображення</p> <p><code>scale</code> роздільна здатність пікселя</p> <p>Налаштування графіка <code>lineWidth</code> вага лінії на графіку</p> <p><code>title</code> Назва графіка</p> <p><code>vAxis</code> назва по осі абсцис</p> <p><code>hAxis</code> дати по осі ординат додавання лінії тренду</p> <p>Результат у консоль</p>
---	---

Натисніть **Run** (Запуск скрипту).

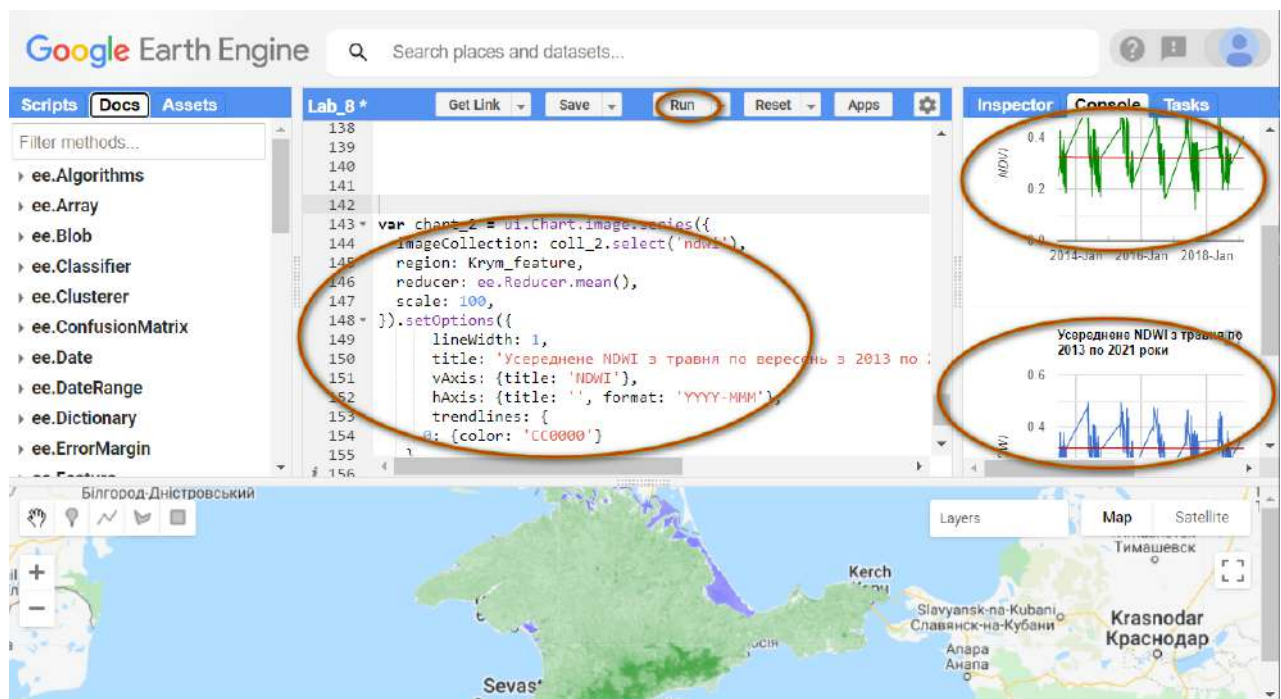


Рис. 168. Виведення інших графіків у консоль

Завантажте графіки.

Для цього скористайтеся в **Console** (Консоль) кнопкою для завантаження.

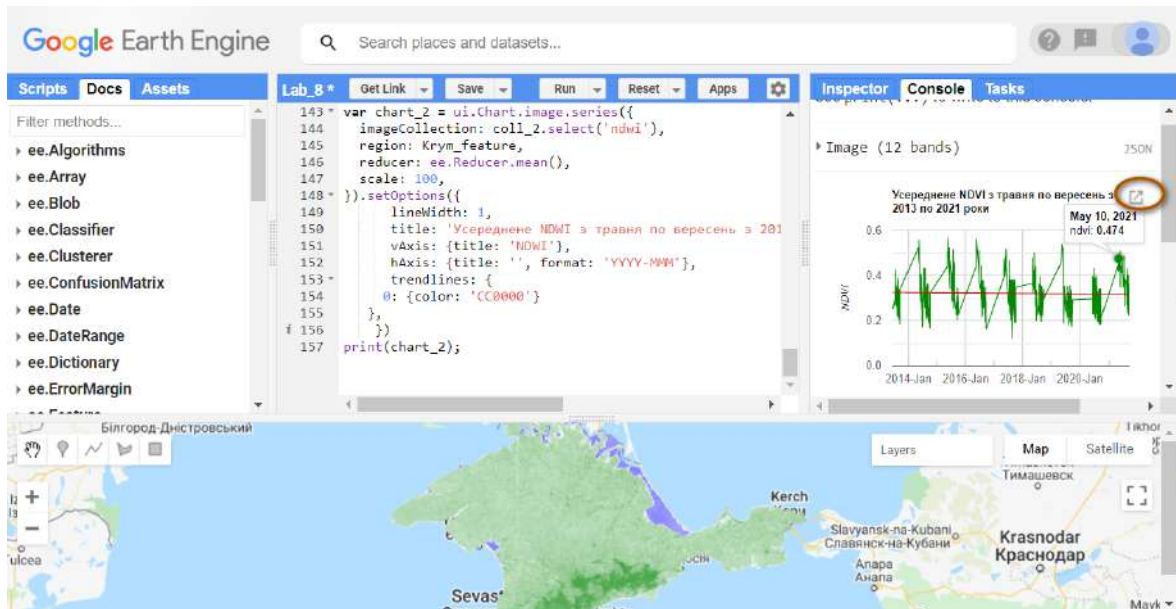


Рис. 169. Відкриття графіка в новій вкладці

Завантажте графіки у форматі PNG.

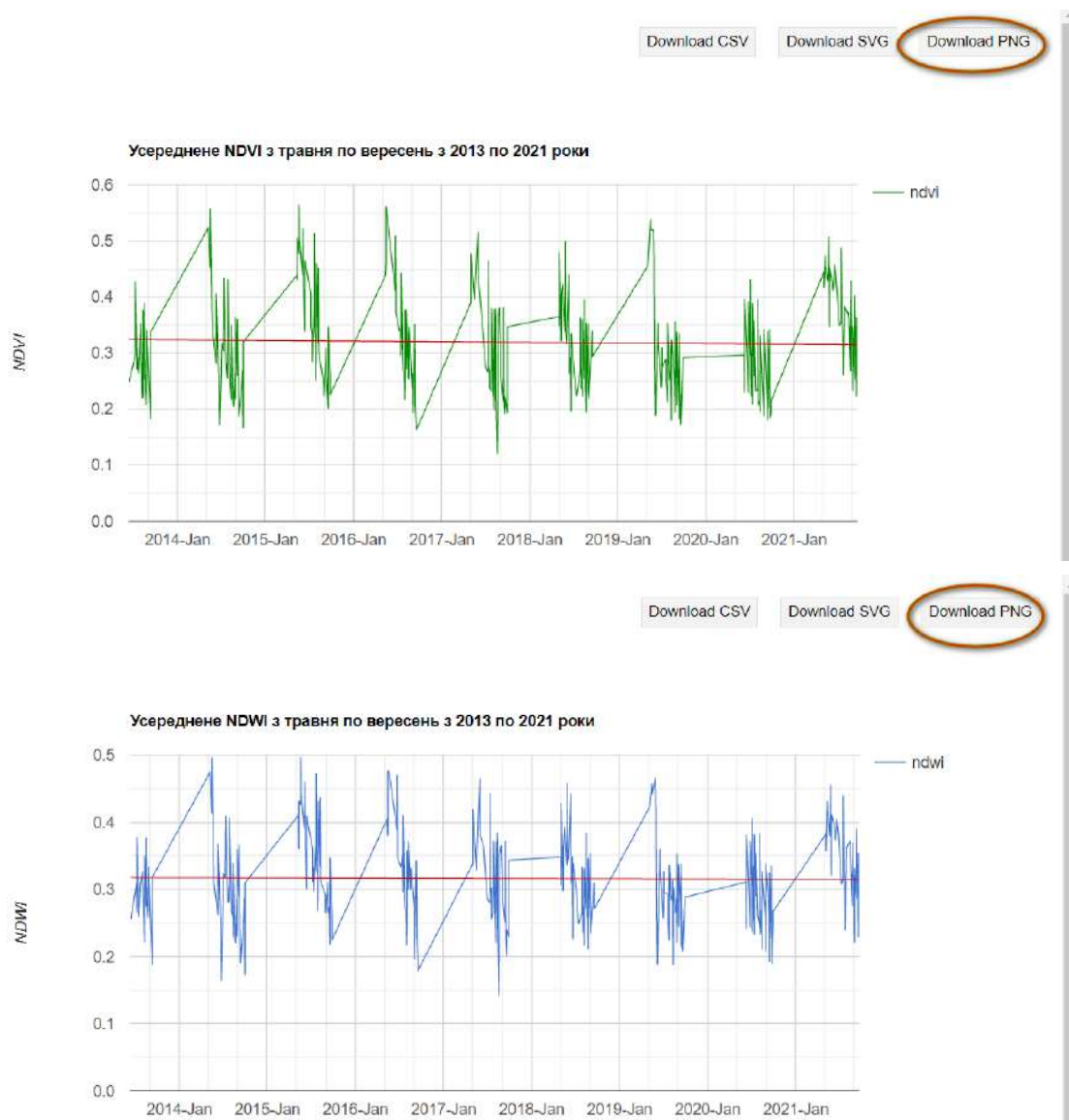


Рис. 170. Нова вкладка з графіками

? Завдання для перевірки

Визначте усереднене значення NDVI та NDWI за п'ять місяців (травень – вересень) з 2013 по 2021 рр. в межах вашої адміністративної області. Простежте тенденцію за лінією тренду (хмарність вибирайте мінімально можливо)

? Додаткове завдання для перевірки

Змініть фільтр хмарності у третьому рядку коду на 20%, на 1%. Як змінився графік? Як ви гадаєте, чому?

За наведеним нижче QR-кодом ви можете переглянути відео виконання практичної роботи.



Рис. 171. Відео виконання практичної роботи

Готовий код можна відкрити за лінком:

https://code.earthengine.google.com/?scriptPath=users%2Fgis_rs%2Fcourse_MAN%3ALab_8

Розрахунок втрати лісу з візуалізацією у користувацькому інтерфейсі на основі даних Global Forest Change для дослідження негативних змін лісового покриву України

Ситуація

Ліси – це екосистеми, які мають одне з вирішальних значень для життя на Землі. Від них залежить стабільність клімату нашої планети й обіг чистої прісної води. Здебільшого в них зосереджене біорізноманіття: 8 з 10 наземних видів рослин і тварин знаходять притулок у лісах. Близько одного мільярда людей напряду залежать від ресурсів лісу: палива, їжі, медикаментів, матеріалів для будівництва.

За останні 30 років площа лісів планети Земля зменшилася на 178 мільйонів гектарів, що майже втричі більше за територію України. Основні загрози для лісів у світі пов'язані з розширенням територій для ведення сільського господарства та заготівлею деревини.

Майже половина території України перебуває у безлісій степовій зоні. Ліси ж займають 16% площі країни, передусім на Поліссі та в Карпатах, і відрізняються різноманіттям типів: хвойні, мішані, листяні і навіть чагарникові ліси середземноморського типу на Південному березі Криму.

Близько половини українських лісів характеризуються штучним походженням, тобто їх посадили люди. Такі здебільшого одновидові лісові насадження є нестійкими до стихійних лих та хвороб, мають низький рівень біорізноманіття порівняно з лісами природного походження. Ми спостерігаємо їхнє масове всихання, в тому числі через зміну клімату і порушення водного балансу. Зокрема це стосується соснових плантацій на Поліссі та смерекових лісів у Карпатах. За оцінками Держлісагентства, на початок 2019 р. загальна площа таких лісів становила 413 тисяч гектарів.

Знеліснення, збезлісіння¹ – процес перетворення порослих лісом земель на угіддя без дерев, як-от: пасовища, пустирі, сільськогосподарські угіддя, міста тощо.

Для моніторингу втрат лісового покриву на території України з 2000 по 2019 рр. може бути використана база даних Global Forest Watch², яка містить у собі шари, що відображають втрату і збільшення лісового покриву на території всієї планети Земля.

Завдання

Створити мобільне вікно для автоматичного розрахунку площі зміни лісового покриву на території України з 2015 по 2018 рр.

Алгоритм виконання завдання:

1. Завантажити датасет *Global Forest Change*.
2. Додати векторний шар адміністративних кордонів України *FAO*.
3. Вибрати шар із бази *Global Forest Change* зі щільністю лісового покриву і вибрати тільки ділянки зі щільністю понад 50%.
4. Вибрати шар із роками знеліснення і діапазон років 2015–2018.
5. Порахувати площу знеліснення за 2015–2018 рр. і вивести на екран.
6. Додати мобільне вікно, з допомогою якого можна рахувати площу знеліснення в конкретній локації, і вивести його на екран.

Покрокова інструкція

1. Завантажити датасет *Global Forest Change*.

¹ URL: <https://uk.wikipedia.org/wiki/Знеліснення>

² URL: <https://www.globalforestwatch.org/map/>

Відкрийте **Code Editor** (Редактор коду): <https://code.earthengine.google.com/> і наберіть у пошуку назву *Global Forest Change*. Коли з'являться назви датасетів, знайдіть **Global Forest Change .1.9 (2000-2021)** і натисніть на нього.

Цей датасет є результатом аналізу часових рядів зображень Landsat для характеристики глобального масштабу лісу та його змін.

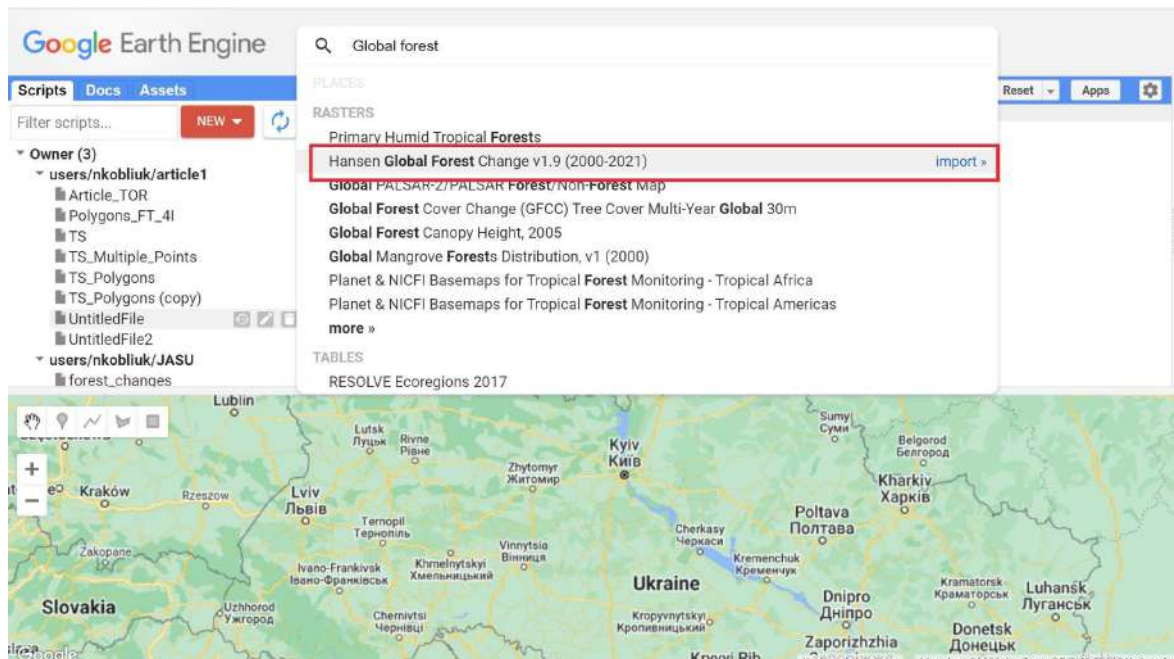


Рис. 172. Пошук датасету

У вікні, що з'явиться на екрані, ви можете побачити інформацію про цей датасет: просторове розрізнення, космічні знімки, з яких він був створений, а також його призначення. Натиснувши на вкладку *Bands* (Шари), ви можете переглянути назви й опис шарів, з яких складається цей датасет. У цій практичній роботі ми будемо використовувати шари "treecover2000" і "forestloss".

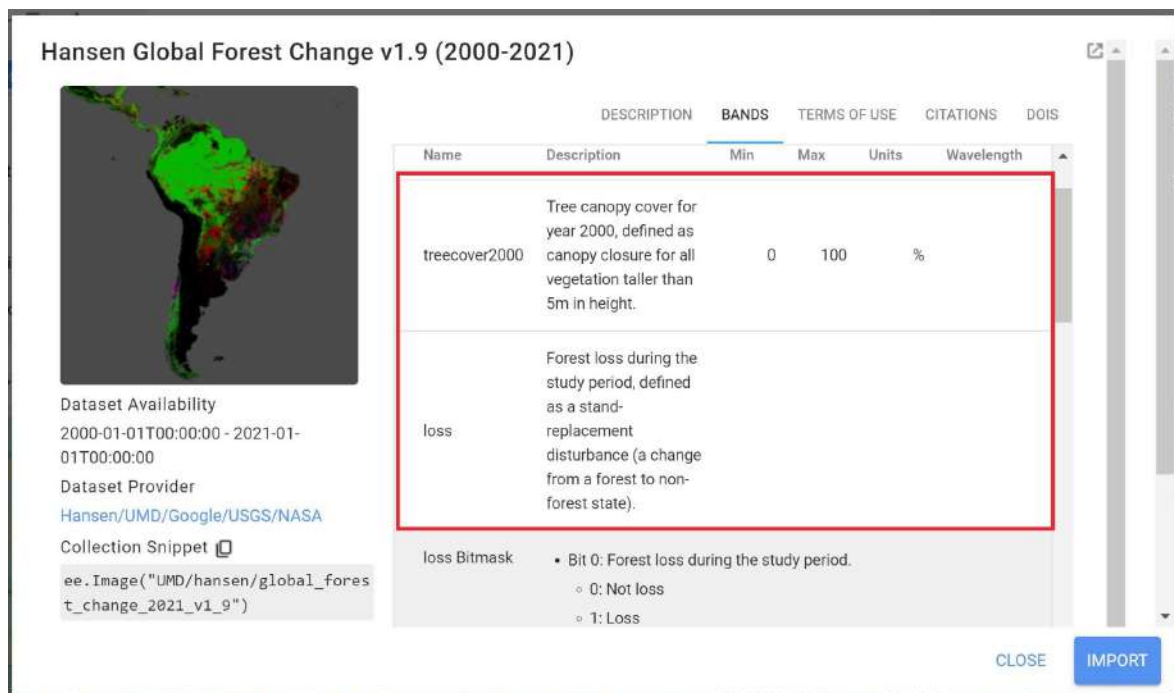


Рис. 173. Атрибути датасету

Наступний крок: натисніть **IMPORT**, щоб додати датасет до робочого середовища. Після цього закрийте вікно пошуку за допомогою **Close** (Закрити).

Тепер потрібно перейменувати датасет. Задайте йому ім'я "GFC", натиснувши на назву **image**. Назва підсвітиться блакитним кольором, а це означає, що ви можете її змінювати.

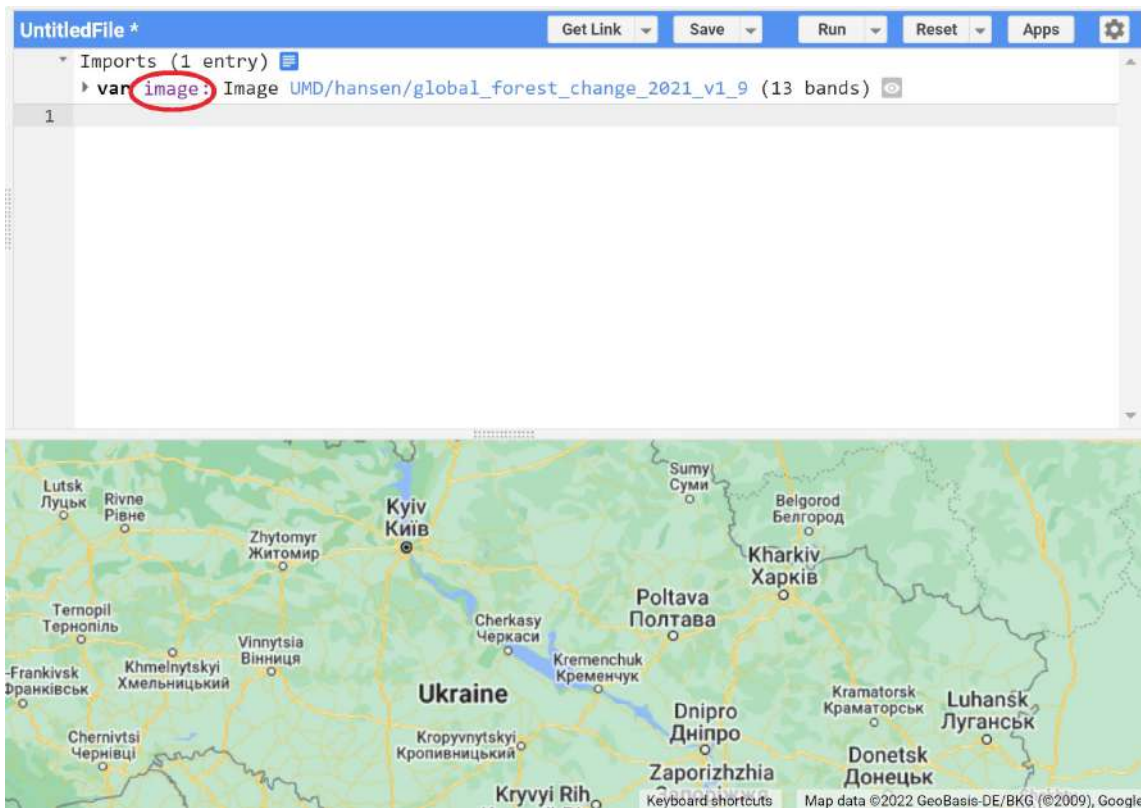


Рис. 174. Зміна назви датасету

2. Створити векторний шар адміністративних меж України.

Для цього в рядку пошуку вписуємо ключове слово **FAO**, що означає *Food and Agriculture Organization*.

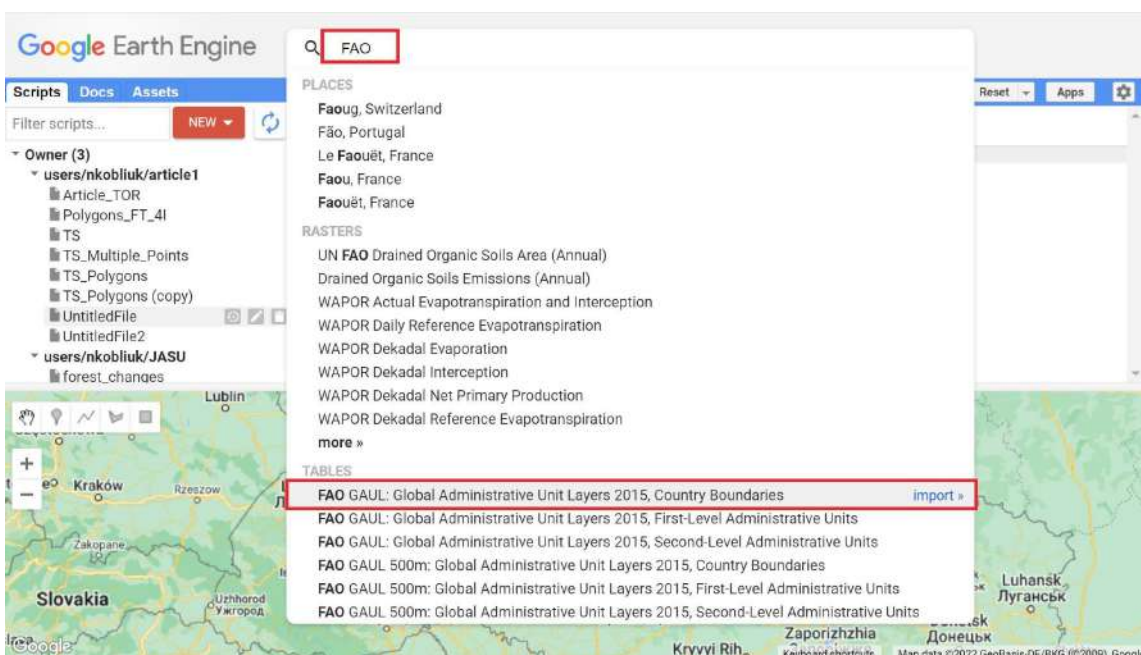


Рис. 175. Пошук необхідних даних

З випадного списку вибираємо «**FAO GAUL: Global Administrative Unit Layers 2015, Country Boundaries**». Далі відкриється інформація про цю базу даних. Необхідно скопіювати її назву `ee.FeatureCollection("FAO/GAUL/2015/level0")`, яка знадобиться нам далі для виділення державного кордону України.

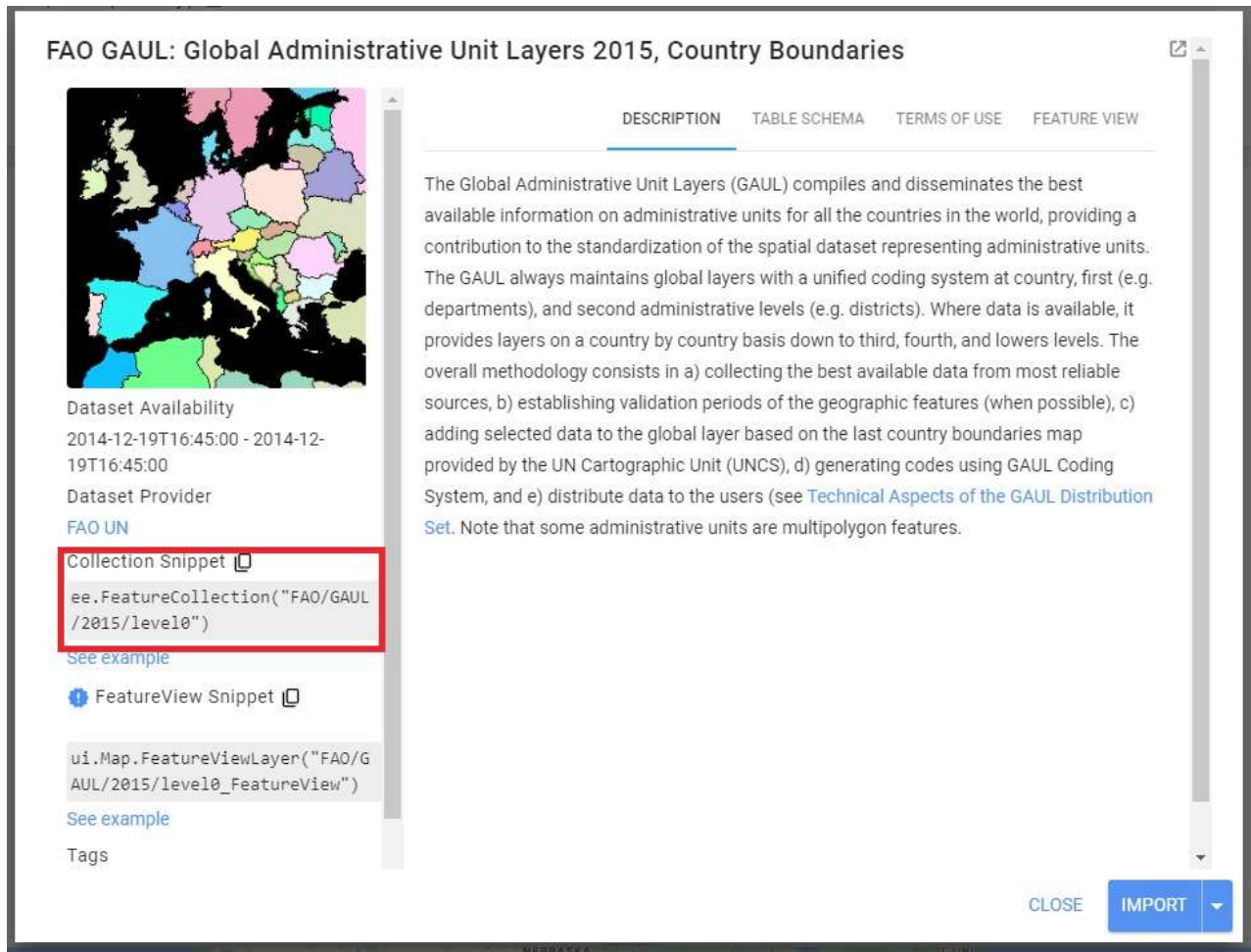


Рис. 176. Копіювання адреси датасету

Для того щоб на мапі відображався кордон лише України, а не всіх країн світу, нам треба відфільтрувати цю базу даних за назвою України. Аби дізнатися, як вона підписана в цій базі даних, ми відобразимо всю інформацію з «**FAO GAUL: Global Administrative Unit Layers 2015, Country Boundaries**» на мапі і за допомогою *Inspector* (Інспектор) дізнаємося назву об'єкта, що нас цікавить. Перед цим ми можемо відцентрувати мапу для полегшення подальшої роботи.

<code>Map.setCenter(31.99, 49.70, 6);</code>	<code>Map.setCenter</code> центрує мапу на визначених координатах і з визначеним рівнем масштабування
--	---

Тепер додамо базу даних «**FAO GAUL: Global Administrative Unit Layers 2015, Country Boundaries**» до проекту і відобразимо її на мапі.

<code>var UA = ee.FeatureCollection("FAO/GAUL/2015/level0"); Map.addLayer(UA);</code>	Створюємо змінну UA, яка набуває значення колекції <code>FAO/GAUL/2015/level0</code> Відображаємо базу даних із кордонами країн
---	--

Натисніть *Run* (Запуск скрипту). Тепер ми можемо переключитися на вкладку *Inspector* (Інспектор) і, клікнувши на полігон України, подивитися, як вона записана в базі даних у розділі *Properties* (Властивості).

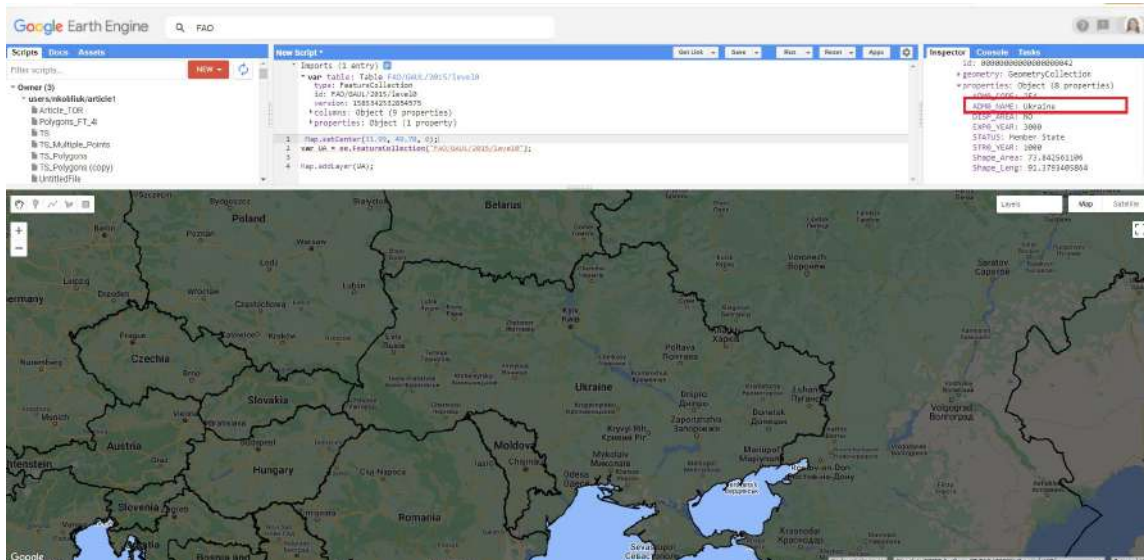


Рис. 177. Виведення даних у консоль

Тепер ми знаємо, як записана назва України в цій базі даних, і можемо відфільтрувати її до кордонів України, додавши до змінної UA функцію фільтрації.

```

var UA = ee.FeatureCollection("FAO/
GAUL/2015/level0") .filter(ee.Filter.eq('ADM0_
NAME', 'Ukraine'));
Map.addLayer(UA, {color: 'green'}, 'Ukraine');

```

Фільтруємо колекцію за назвою стовпчика **ADM0_NAME**, а конкретно – назвою країни **Ukraine** із допомогою функції **.filter**.

Додаємо шар з кордоном України до відображення на мапі, зазначивши колір та назву в легенді мапи. Попередній рядок з функцією **Map.addLayer** можна видалити.

!!!Зверніть увагу, що ми просто додаємо фільтр до вже створеної змінної!

Натисніть **Run** (Запуск скрипту).

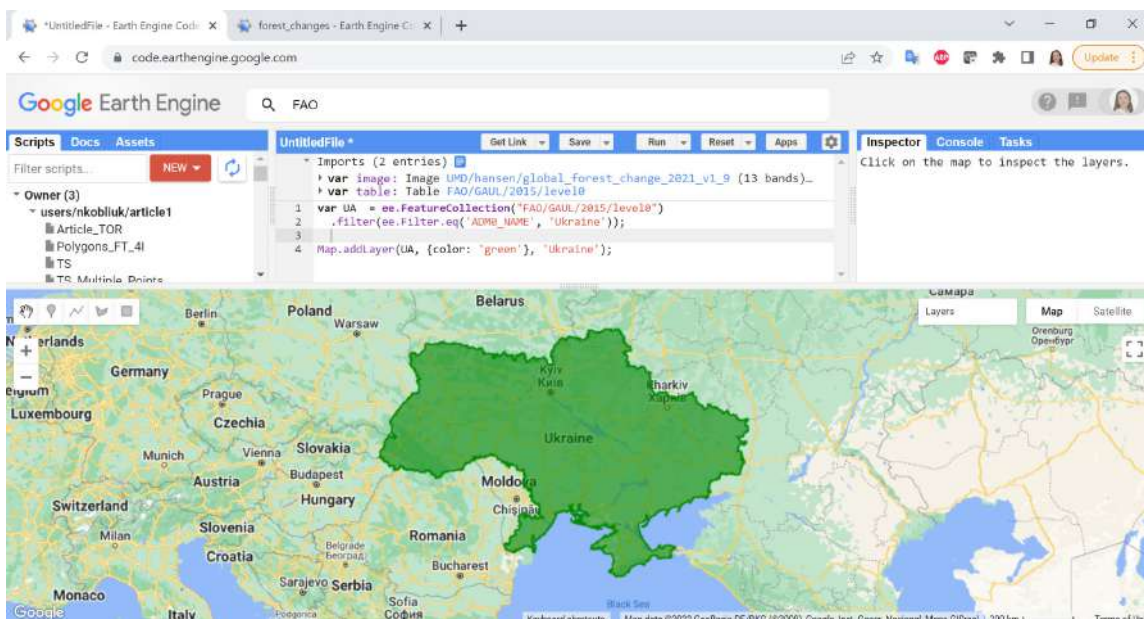


Рис. 178. Зміна відображення даних

3. Вибрати шар із бази Global Forest Change зі щільністю лісового покриву і вибрати тільки ділянки зі щільністю понад 50%:

<pre>var maskedForest = GFC.updateMask(GFC.select('treecover200 0').gte(50));</pre>	<p>Створюємо змінну <code>maskedForest</code>, яка набуває значення бази даних Global Forest Change і фільтруємо її за щільністю лісового покриву >50%</p>
---	---

4. Вибрати шар з роками знеліснення і діапазон років 2015–2018.

<pre>var masked = maskedForest.updateMask(GFC.select('loss year').gt(14)); var maskedLoss = masked.updateMask(GFC.select('lossyear').lte(18)).clip(UA); Map.addLayer(maskedLoss, {}, "Forest loss");</pre>	<p>Створюємо змінну, що з вибраного лісу щільністю >50% фільтрує втрати лісового покриву у часовому проміжку від 2014 р. (<code>gt(14)</code>). Функція <code>updateMask</code> візуалізує тільки ті пікселі растра, що були задані в умові</p> <p>Далі фільтруємо втрати лісового покриву на період до 2018 р. включно <code>lte(18)</code> і обрізаємо по кордону України <code>clip(UA)</code></p> <p>Додаємо результат на карту, вказуючи назву, яка буде відображатися в легенді карти</p>
--	--

Натиснути **Run** (Запуск скрипту).

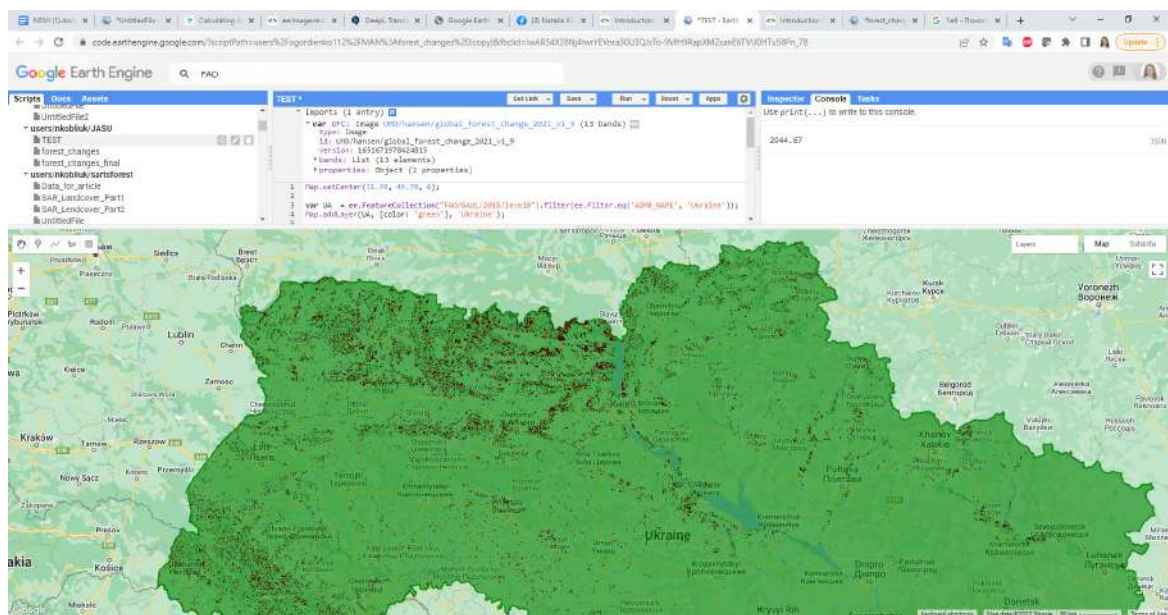


Рис. 179. Додавання нового шару на карту

Найбільш використовувані оператори порівняння:

- gt – більше;
- gte – більше або дорівнює;
- lt – менше;
- lte – менше або дорівнює.

5. Порахувати площу знеліснення за 2015–2018 рр. і вивести на екран.

Розрахунок площі для зображень виконується за допомогою функції `ee.Image.pixelArea()`. Ця функція створює зображення, де значення кожного пікселя – це площа пікселя. Якщо пікселі зображення містять значення 0 або 1, ми можемо помножити це зображення з площею пікселя на наше зображення й обчислити загальну площу за допомогою функції `reduceRegion()`.

```
var areaImage = maskedLoss.multiply(ee.Image.pixelArea());
;
```

Створюємо змінну `areaImage`, у якій множимо значення `maskedLoss` на значення площі, щоб отримати втрати в числовому вигляді

Тепер нам необхідно порахувати сумарну площу всіх пікселів на території України. Для цього будемо використовувати ред'юсер-функцію, яка допомагає зробити агрегацію даних. Наприклад, зібрати багато значень в одне, підсумувавши їх. Reduce з англійської – це зменшення.

```
var areaUA = areaImage.select('loss').
reduceRegion({ reducer: ee.Reducer.sum(),
geometry: UA,
maxPixels: 1e15
});
```

Створюємо нову змінну `areaUA`, якій присвоюємо суму значень всіх пікселів зі змінної `areaImage` з допомогою ред'юсера `reduceRegion` і типу ред'юсера `ee.Reducer.sum()`. Вибираємо геометрію `UA`, всередині якої рахуються пікселі, і вказуємо максимально допустиму їх кількість `1e15`.

```
var LossAreaSqKmUA = (ee.Number(
areaUA.get('loss')).divide(1e6));
```

Для того щоб отримати значення площі в гектарах, а не у квадратних метрах, треба розділити сумарну площу за допомогою `divide(1e6)`

Тепер можемо вивести площу знеліснення на території України в період із 2015 по 2018 рр. на екран у *Console* (Консоль).

```
print(LossAreaSqKmUA.format('%.2f'));
```

Аргумент функції `'%.2f'` дає змогу вивести результат у вигляді числа з плаваючою комою із точністю два знаки після коми

6. Додати мобільне вікно, з допомогою якого можна порахувати площу знеліснення в конкретній локації, і вивести його на екран.

Тепер необхідно додати до мапи полігон, у рамках якого буде розраховуватися площа знеліснення. Для цього треба використати інструменти рисування – нанесення прямокутника чи многокутника. Створений об'єкт відобразиться у списку *Imports* (Імпортовані) в редакторі коду з ім'ям `geometry`.

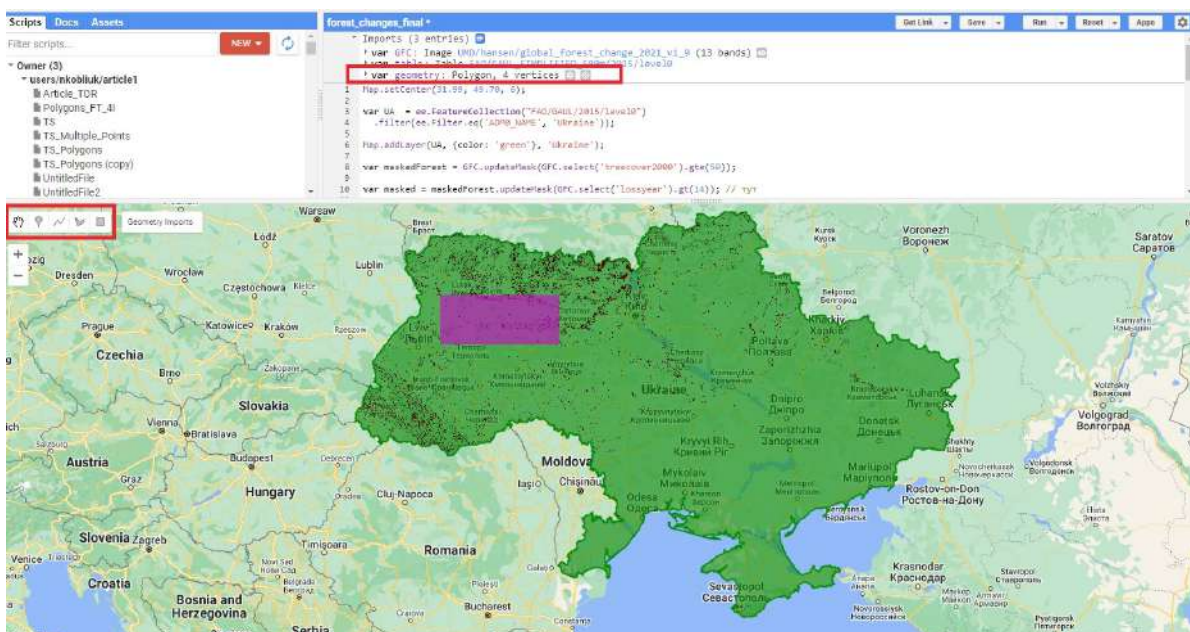


Рис. 180. Налаштування зони цікавості

Наблизи́ть мапу за допомогою коліщатка миші до території, яку щойно накреслили. Скористайтеся кнопкою **Satellite**, щоби перемикаєть вид мапи на **Гібрид (Hybrid)**. Найчастішими причинами знеліснення в Україні є вирубки лісів, пожежі, повені, урагани та ін.

? Завдання для перевірки

Додаткове завдання*
 Подивіться уважно на космічний знімок території.
 Як ви вважаєте, чим було спричинене знеліснення на цій ділянці? Які дешифрувальні ознаки підтверджують ваше припущення?

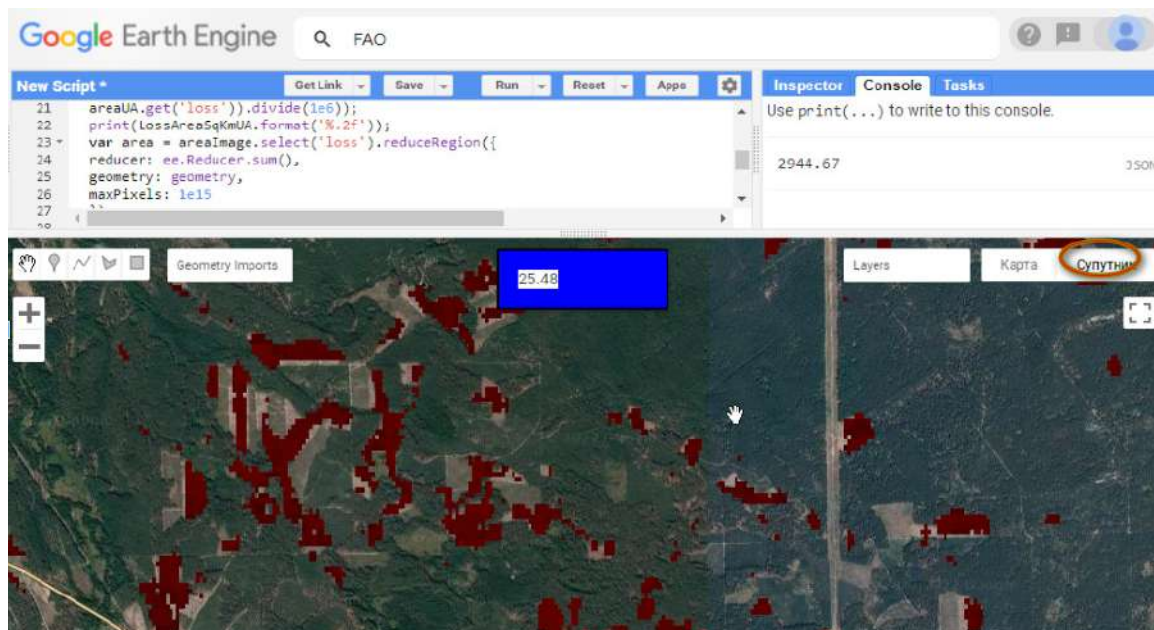


Рис. 181. Зміна відображення мапи

Після цього можемо використати ті самі функції для підрахунку площі, що і для всієї території України.

<pre>var area = areaImage.select('loss'). reduceRegion({ reducer: ee.Reducer.sum(), geometry: geometry, maxPixels: 1e15 }); var LossAreaSqKm = (ee.Number(area.get('loss')).divide(1e6));</pre>	<p>Створюємо змінну <code>area</code> і використовуємо <code>reducer</code> для підрахунку сумарної площі всіх пікселів, вказуючи, що ми хочемо порахувати площу в середині геометрії <code>geometry</code> (полігон, який ми власноруч нанесли на мапу);</p> <p>Конвертуємо отриману площу у квадратних метрах у гектари</p>
---	---

Тепер додамо на нашу мапу панель, на яку буде виводитися значення площі знеліснення на території полігона, що був нарисований.

<pre>var panel = ui.Panel(); panel.style().set({ width: '150px', position: 'top-center', backgroundColor: 'blue', border: '2px solid black', }); Map.add(panel);</pre>	<p>Створюємо змінну <code>panel</code>, якій надаємо значення <code>ui.Panel()</code> – це елемент користувацького графічного інтерфейсу. Далі з допомогою функції <code>style().set</code> задаємо візуальні характеристики панелі (ширину, розташування, колір, товщину границі);</p> <p>Після цього додаємо панель до відображення на мапі з допомогою функції <code>Map.add</code>.</p>
--	---

Для того щоб додати текст на створену панель, треба додати на неї напис і вивести на нього значення площі знеліснення всередині створеного полігона.

```
panel.add(ui.Label({value:LossAreaSqKm.  
format('%.2f').getInfo()}));
```

Додаємо напис із допомогою функції `add(ui.Label)`. Після чого присвоюємо цьому напису значення знеліснення в середині полігона

? Завдання для перевірки

1. Порахувати площу знеліснення на територію вашої області і візуалізувати її на мапі. Зробити це можна, отримавши векторний шар вашої області з бази даних FAO GAUL: Global Administrative Unit Layers 2015, First-Level Administrative Units.

✓ **Важливо!** Цю змінну слід оголошувати перед змінною `panel`. Найкраще це зробити поряд зі змінною, що має назву `UA`, щоби взяти її за взірцем.

Порядок дій такий:

- підвантажити датасет, дати йому ім'я `adm`;
- вивести дані за допомогою `Map.addLayer()`;
- за допомогою `Inspector` (Інспектор) знайти значення назви для своєї адміністративної одиниці;
- відфільтрувати дані за назвою своєї області;
- змінити значення `UA` у коді на значення змінної `adm`;
- нове значення з'явиться в консолі.

✓ **Важливо!** В `JavaScript` не можна змішувати різні лапки, але можна використовувати подвійні й одинарні лапки. Тому якщо в назві області є лапки, візьміть її в подвійні лапки.

2. Змінити місце розташування панелі з площею знеліснення та її колір на ваш вибір. Додаткову інформацію можна знайти тут: <https://developers.google.com/earth-engine/apidocs/ui-panel-style?hl=en>

За наведеним нижче QR-кодом ви можете переглянути відео виконання практичної роботи.



Рис. 182. Відео виконання практичної роботи

Готовий код можна відкрити за лінком:

https://code.earthengine.google.com/?scriptPath=users%2Fgjs_rs%2Fcourse_MAN%3ALab_9

Класифікування водойм за допомогою методу машинного навчання Random Forest на території Шацьких озер

Ситуація

На території України близько 20 тисяч озер. У басейні Західного Бугу та Прип'яті розташовані Шацькі озера (понад 30 прісноводних водойм), вони мають карстове походження. Ці озера разом з навколишнім середовищем утворюють Шацький національний парк. Найбільше і найглибше серед них – озеро Світязь.¹ Воно є цінною туристичною принадою України, але останнім часом помічені зміни щодо обміління цієї пам'ятки природи.² У GEE ми можемо перевірити, чи справді зменшується площа Шацьких озер.

Вода з точки зору дистанційного зондування Землі є відмінною від інших об'єктів за своїми фізичними характеристиками, тому її легше ідентифікувати за допомогою класифікації растра. Щоби познайомитися із класифікуванням за допомогою вбудованої бібліотеки машинного навчання Random Forest, краще «тренуватися» на воді. Іноді на пошук географічних даних для подальшого їх опрацювання витрачається купа часу, а така проста класифікація може допомогти приблизно з'ясувати місце розташування об'єктів. Наприклад, якщо ви хочете опрацювати всі озера України, ви можете звернутися до ресурсу Open Street Map (OSM), але тут може постати проблема актуальності, повноти й точності нанесених об'єктів. Подібний алгоритм, тобто розмежування об'єктів на класи, буде працювати й з іншими типами поверхні Землі. Наприклад, вигорілі території лісу, дороги, нафтові плями та багато іншого можна розділити на класи.

Завдання

Автоматично виділити усі водні об'єкти на території Шацького національного парку, використовуючи алгоритм машинного навчання Random Forest і супутникові знімки Sentinel-2, порівняти площу водної поверхні Шацьких озер за 2021 та 2020 рр.

Алгоритм виконання завдання:

1. Наблизитися до території Шацьких озер і виділити територію інтересу.
2. Створити два нові шари геометрії (суша – *land*, озеро – *lake*) для подальшого вибору еталонів водойм і суші, які будуть використовуватися в класифікації під час розпізнавання території.
3. Нанести на мапу по п'ять еталонів для класу води *lake* і класу суші *land*.
4. Порахувати усереднений знімок із фільтром безхмарності Sentinel-2 і розрахувати вегетаційний індекс, який буде додатковим тематичним шаром для підсилення точності класифікації.
5. Застосувати класифікацію Random Forest.
6. Визначити площі всіх Шацьких озер за літо 2020 та 2021 рр.

Покрокова інструкція

1. Наблизитися до Шацьких озер і виділити територію інтересу.

Світязь – це найглибше і найбільше за площею озеро в Україні. Воно входить до Шацького природного національного парку, разом з іншими озерами утворює групу Шацьких озер. Щоб знайти його на мапі, відкрийте **Code Editor** (Редактор коду): <https://code.earthengine.google.com/> й у рядку пошуку напишіть *Світязь*. Зі списку, що відкрився, виберіть: *Світязь, Волинська область, Україна* і натисніть **Enter**.

¹ Водний фонд України. Штучні водойми: водосховища і ставки : довідник / за ред. В. К. Хільчевського, В. В. Гребеня. Київ : Інтерпрес, 2014. 164 с.

² URL: <https://life.pravda.com.ua/columns/2020/04/24/240724/>

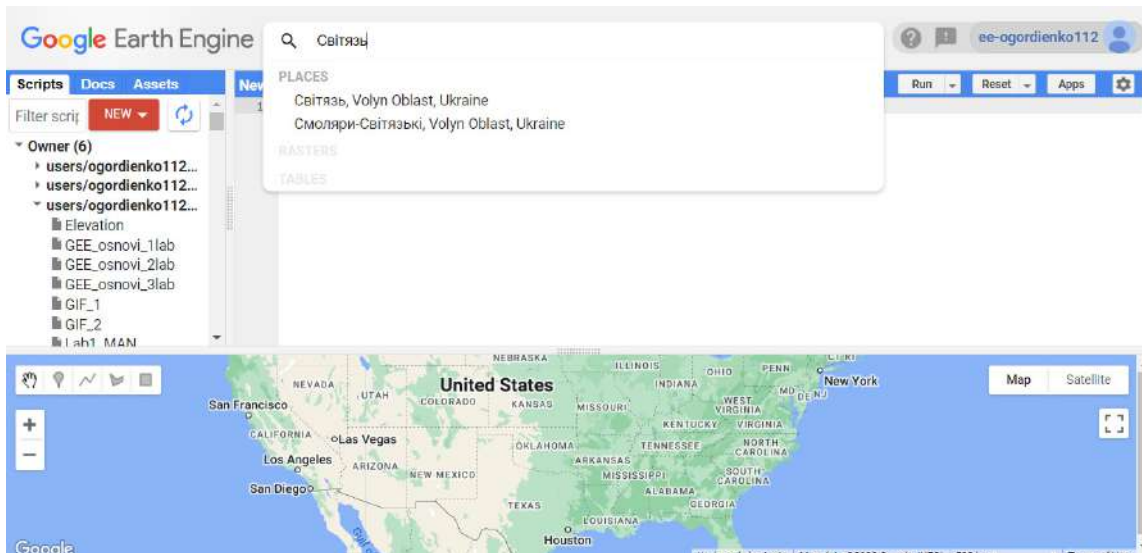


Рис. 183. Пошук зони інтересу

Сервіс покаже територію вашого інтересу згідно із запитом, а саме – Шацькі озера, де і потрібно буде розкласифікувати зображення. Класифікувати ми будемо озеро Світязь і найближчі до нього озера. Для цього нам потрібно створити контур території інтересу, де буде проходити класифікація. Це можна зробити за допомогою інструменту *Geometry imports* (Імпорт геометрії). В інтерфейсі GEE натисніть на іконку із зображенням прямокутника і нарисуйте його так, щоб він покривав водне дзеркало озера Світязь і деяких озер поряд, точність тут не важлива. Після цього у вкладці *Imports* (Імпорт) з'явиться нова змінна, яка називається *geometry*. Вона має тип «Полігон» і чотири вершини.

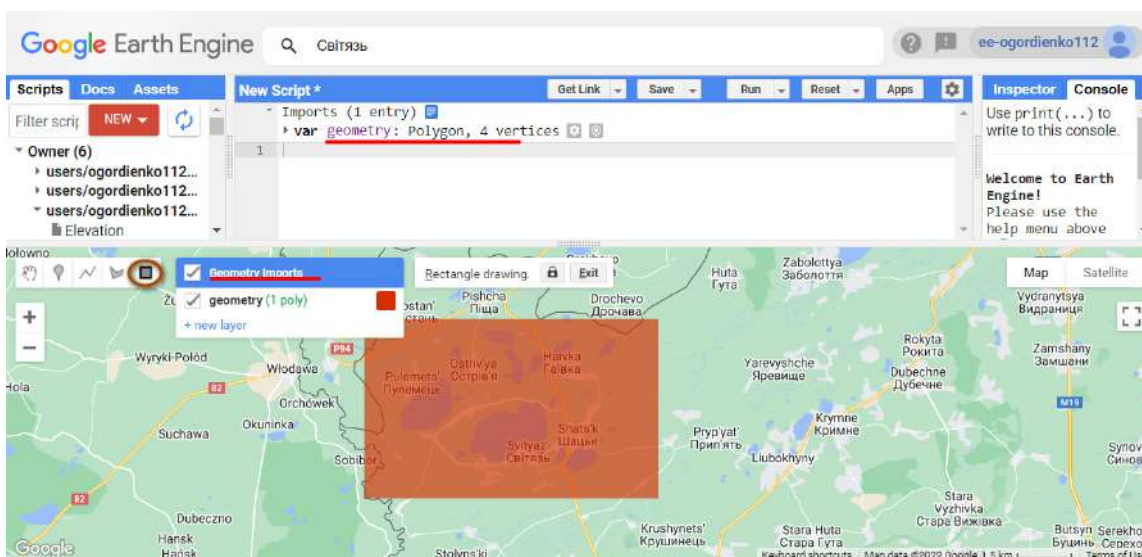


Рис. 184. Створення шару геометрії

2. Створити два нові шари геометрії (суша – *land*, озеро – *lake*) для подальшого вибору еталонів водойм і суші, які будуть використовуватися в класифікації під час розпізнавання території.

Наступним кроком є додавання ще двох класів, які надалі будуть використані для вибору еталонів водойм і суші згідно з нашою моделлю класифікування. Ці класи будуть використовуватися як еталони для машинного навчання. Тобто далі в роботі ми розподілимо нашу модель класифікації на воду та все інше. Для цього необхідно створити два еталонних класи, в які потрапляють об'єкти *land* (всі об'єкти, крім води), *lake* (вода). Таким чином до машинного навчання буде подано два класи: вода та все інше.

Наступним кроком буде зайти до меню **Geometry imports** (Імпорт геометрії) і натиснути там на **+ new layer** (новий шар). GEE створить новий клас з ім'ям **geometry2**. Таке ім'я створюється за замовчуванням, нам потрібно його змінити. Для цього треба натиснути на іконку зі значком шестерні.

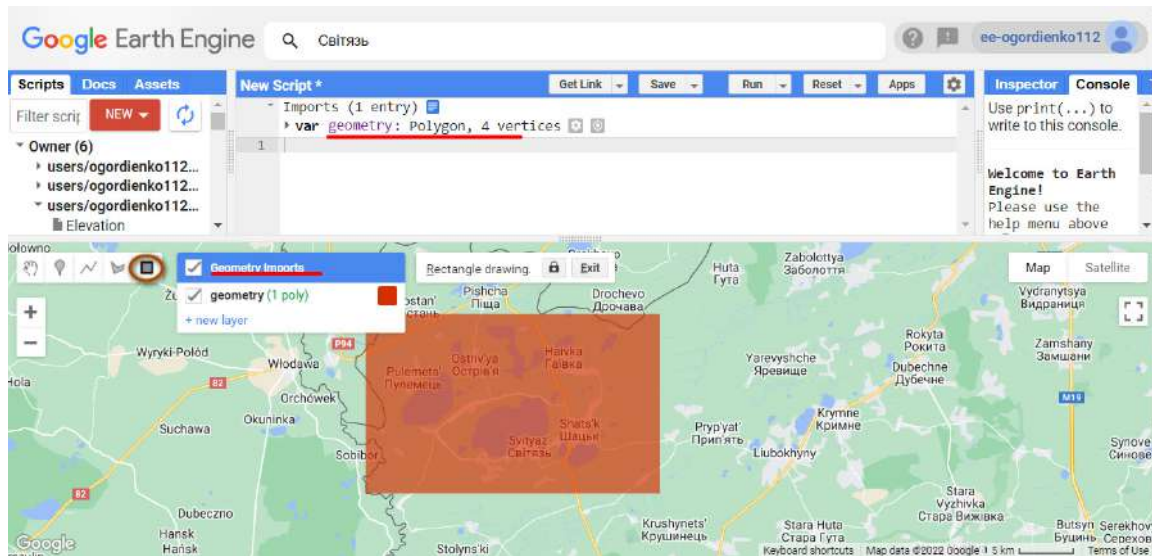


Рис. 185. Налаштування шару геометрії

Після цього відкриється нове меню, в якому можна змінювати назву створених об'єктів, їх тип, а також задавати їм параметри. У новому вікні потрібно змінити назву **Name** (Назва) на назву одного з об'єктів класифікації. Першим створимо об'єкт **land** (у нашому випадку це клас, який об'єднує всі об'єкти суші: ліси, поля, населені пункти тощо), ввівши його назву у відповідне поле. Потім потрібно змінити тип об'єкта, який ми створюємо. Для цього відкрийте випадний список із написом **Geometry** (Геометрія) і змініть його на **Feature Collection** (Колекція об'єктів).

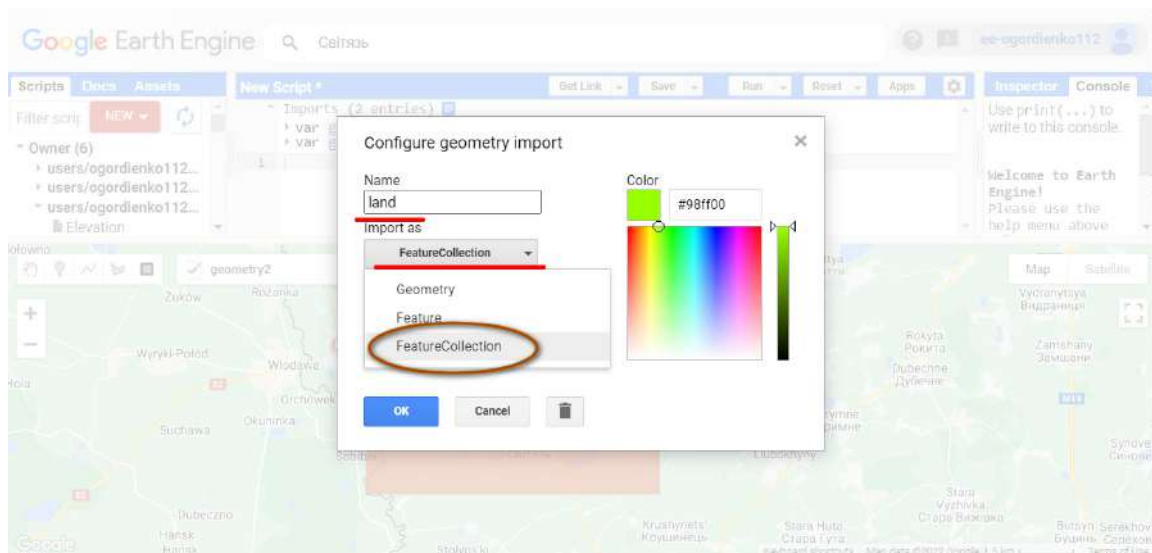


Рис. 186. Надавання налаштувань шару

Після цього з'явиться можливість задати нашому новому класу властивості. Потрібно натиснути на кнопку **Property** (Властивості). Ми будемо задавати властивості, це **class** і **0**. Їх потрібно ввести в колонки **Property** (Властивості) і **Value** (Значення) відповідно. Також на цьому етапі можна змінити колір створеного об'єкта. В GEE є декілька способів змінювати колір векторних об'єктів, один з них – це перетягувати повзунок по палітрі, також можна задавати коди у форматі HEX. Натиснути **OK**.

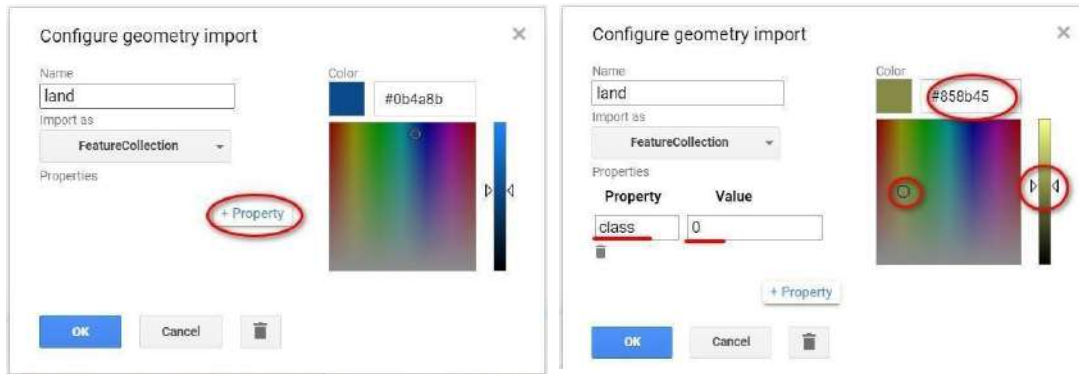


Рис. 187. Зміна вигляду шару

Наступним кроком є створення іншого класу для еталонів водойм з ім'ям *lake* і параметрами *class* та *1*.

Для цього потрібно повторити ті самі дії в *Geometry imports* (Імпорт геометрії), натиснути на *new layer* (новий шар), потім на шестерню і змінити на нові параметри та повторити ті самі кроки, але тепер потрібно створити клас для озер. Тобто в параметрах потрібно задати назву *lake* і створити новий клас зі значенням 1 – все, як у попередньому кроці, але тепер значення дорівнює *1*, а назва – *lake*.

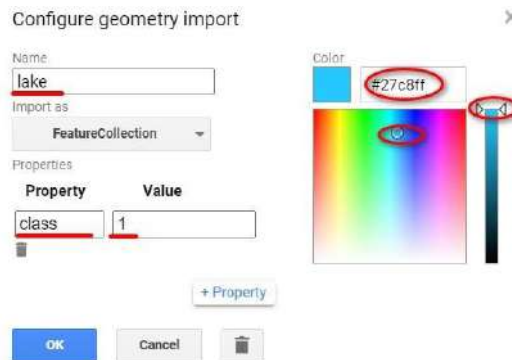


Рис. 188. Надавання налаштувань шару

Після цих дій у вкладці *Imports* (Імпорт) з'являться три шари для об'єктів: один ми нарисували, інші створили, але ще не розмістили їх на мапі. Зараз ми створюватимемо модель для машинного навчання, яка буде класифікувати зображення із супутника Sentinel-2, його ми додамо потім. У цьому випадку класифікацію можна проводити навіть на мапі, яку надає компанія «Google». За допомогою коліщатка миші перемістимося трохи ближче до озер і відключимо шар із назвою *geometry*, щоб він не заважав. Це можна зробити, знявши галочку біля його назви.

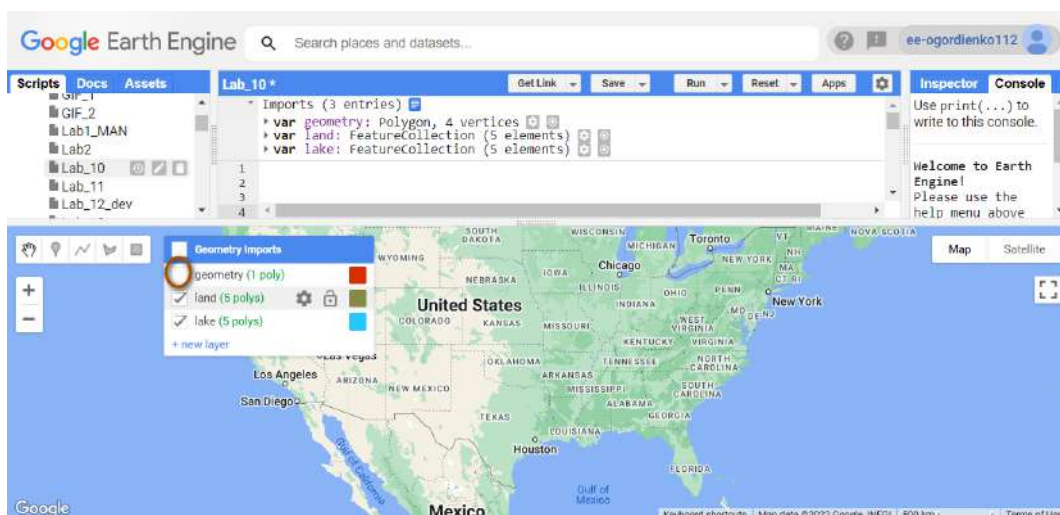



Рис. 189. Зміна відображення шарів

3. Нанести на мапу по п'ять еталонів для класу води *lake* і класу суші *land*.

Тепер можна створювати еталони для класифікації, тобто відділяти озера від всього іншого.

Для цього в тому самому меню, де ми виключили клас *geometry*, потрібно зробити активним клас *lake*, натиснувши на його назву, і скористатися інструментом *прямокутник*. Видаляти об'єкти можна за допомогою інструменту  і кнопки *Delete* (Видалити) на клавіатурі. Створіть п'ять еталонів, що перебуватимуть у межах озер. Тут уже важлива точність: краще не виходити за межі озер.

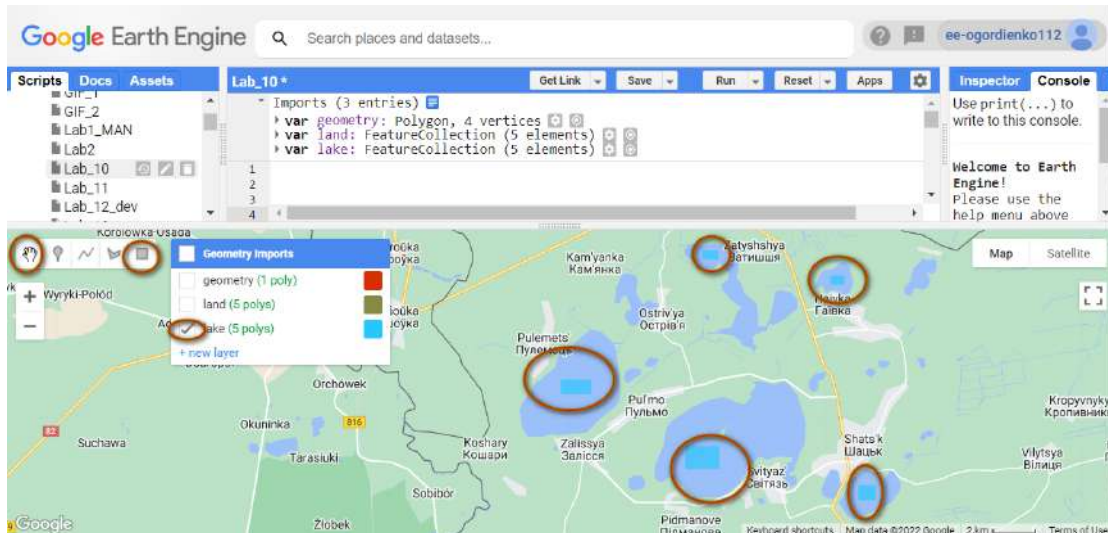


Рис. 190. Ділянки для навчання

Після цього об'єкти, які ви створили, з'являться в *Geometry imports* (Імпорт геометрії), де буде вказано, що створено саме п'ять об'єктів. Таку саму операцію необхідно повторити і для класу *land* – старайтеся захоплювати різні за типом об'єкти: міста, дороги, ліс. Для цього виберіть у меню *Geometry imports* (Імпорт геометрії) клас із назвою *land* і за допомогою інструменту *полігон* створіть п'ять нових полігонів, що покривають інші об'єкти. Різноманітні об'єкти, які ви бачите, це міста, дороги, ліс, ґрунти – все, крім водної поверхні. Для наочності один з полігонів накреслено точно за межами нашої території інтересу – *geometry*, але він все одно буде брати участь у класифікації.

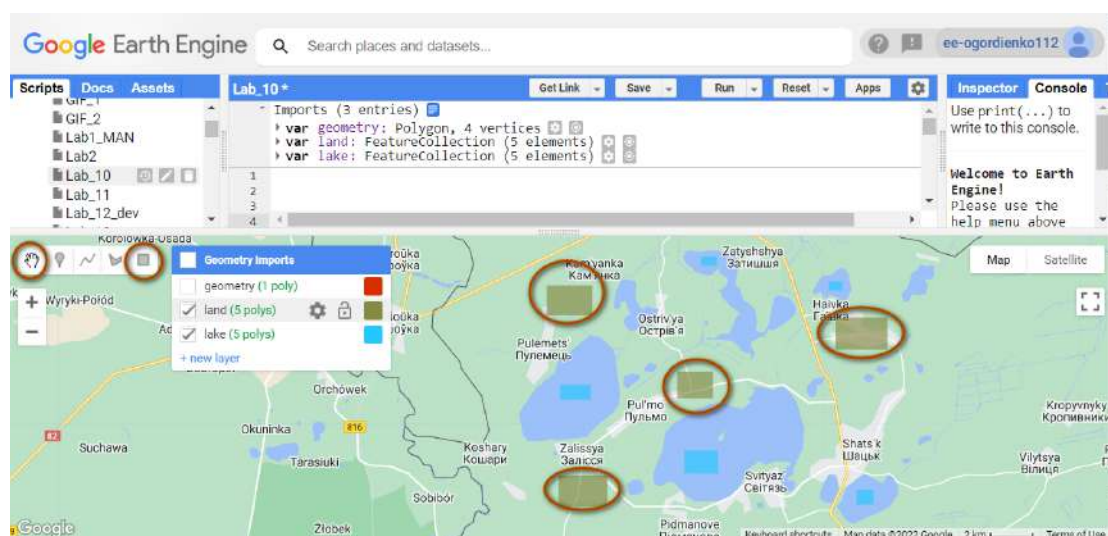


Рис. 191. Вигляд шарів на мапі

Далі нам необхідно об'єднати щойно створені класи. Вони об'єднуються, щоб у подальших кроках їх можна було використовувати для машинного навчання. Це можна зробити, додавши до *Code Editor* (Редактор коду) рядок:

```
var polygons=lake.merge(land);
```

Змінна `polygons` об'єднує два класи `lake` та `land`, які ми створили в попередніх кроках

Цей рядок створить нову змінну з ім'ям `polygons`, в якій будуть зберігатися два об'єкти: `land` та `lake`. Додаймо їх на мапу і в консоль за допомогою рядків:

```
Map.addLayer(polygons,{'Дані для навчання'}, true);  
print(polygons);
```

`Map.addLayer` стандартна конструкція для додавання шарів на мапу (параметри, що будуть відображатися: параметри візуалізації, 'Назва', булевий (ТАК або НІ, True or False) оператор даних для відображення шару)
`print` виводить результат у консоль

Натискаємо **Run** (Запуск скрипту).

Тепер цей шар з'являється у вкладці **Layers** (Шару). І в консолі можна побачити кількість створених об'єктів у шарі `polygons`.

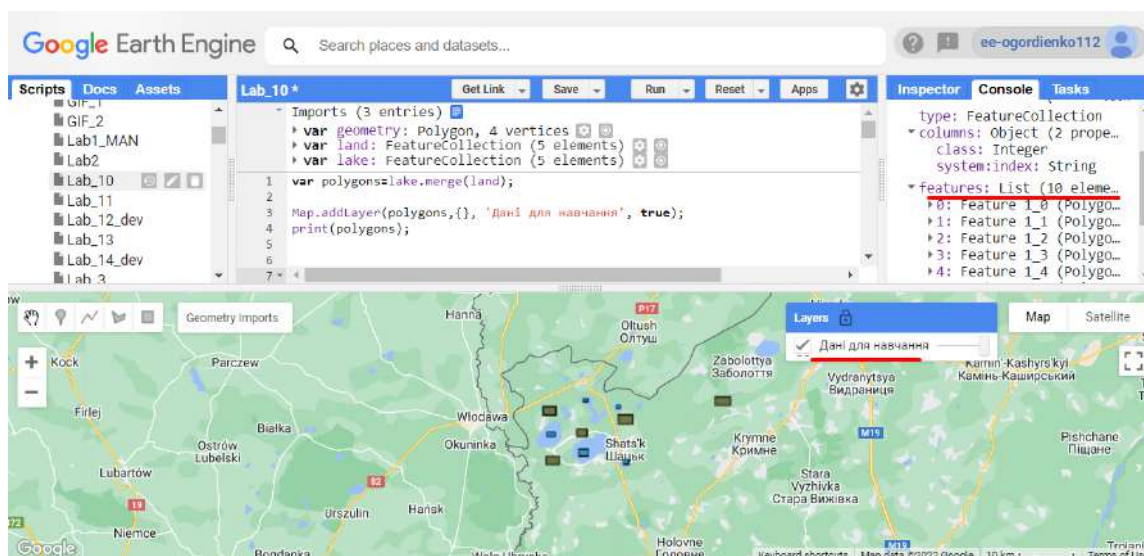


Рис. 192. Виведення списку в консоль

4. Порахувати усереднений знімок із фільтром безхмарності Sentinel-2 і розрахувати вегетаційний індекс, який буде додатковим тематичним шаром для підсилення точності класифікації.

Наступний крок: порахуємо NDVI як додатковий вхідний канал, який дасть змогу зробити точнішу класифікацію. Реалізуємо це за допомогою функції, яку потрібно вставити в редактор коду. Індекс рослинності NDVI гарно відображає воду, усі водойми будуть мати значення менше 0.

```
var addNDVI = function(dataset){  
var ndvi = dataset.normalizedDifference(['B8', 'B4']).rename('NDVI');  
return dataset.addBands(ndvi);  
};
```

Змінна, що створює функцію з назвою `addNDVI` та з аргументами `dataset`, `'B8'`, `'B4'` (які отримує на вході), де за допомогою `normalizedDifference` розраховується індекс `NDVI` та на виході `return`, 'за допомогою `addBands` отримуємо новий канал, що матиме назву `'NDVI'`

Ця функція рахує вегетаційний індекс, але вона поки не задіяна в наступних кроках. Ми її застосуємо згодом. Ми розраховуємо вегетаційний індекс, тому що вода в такому способі відображення буде мати від'ємні значення, і це покращить модель класифікації. Далі необхідно створити змінні, що будуть описувати час, за який ми шукаємо супутникові знімки, а саме: з травня по серпень 2021 р. Для цього створимо дві змінних:

<pre>var start = ee.Date.fromYMD(2021,5,1); var finish = ee.Date.fromYMD(2021,8,1);</pre>	<p>Стандартна конструкція для отримання дати <code>ee.Date.fromYMD(рік,місяць,день)</code></p> <p>Стандартна конструкція для отримання дати <code>ee.Date.fromYMD(рік,місяць,день)</code></p>
---	---

Після цього нам треба задати змінну, де визначено, за якими саме каналами відбуватиметься класифікація. Це можна зробити за допомогою:

<pre>var bands = ['B1', 'B2', 'B3', 'B4', 'B5', 'B6', 'B7', 'B8', 'B8A', 'B9', 'B10', 'B11', 'B12', 'NDVI'];</pre>	<p>Змінна з ім'ям <code>bands</code> зі значеннями каналів <code>'B1'</code> - <code>'B12'</code> та порохом <code>NDVI</code></p>
--	--

Наступним кроком є створення датасету, до якого ми будемо звертатися з усіма функціями, які тільки що створили. Ми звертаємося до датасету, в якому є супутникові зображення Sentinel-2, додаємо до нього створений NDVI, фільтруємо за датою і за покриттям хмар.

<pre>var img = ee.ImageCollection('COPERNICUS/S2') .map(addNDVI) .filterDate(start, finish) .filter(ee.Filter.lt('CLOUD_COVERAGE_ASSESSMENT',10));</pre>	<p>Змінна з ім'ям <code>img</code>, де ми звертаємося до колекції знімків Sentinel-2 <code>'COPERNICUS/S2'</code> за допомогою команди <code>ee.ImageCollection</code>, далі ми застосовуємо декілька фільтрів</p> <p><code>.map</code> застосовує для колекції створену нами функцію <code>NDVI</code>;</p> <p><code>.filterDate</code> вибирає діапазон дат, <code>.filter(ee.Filter.lt</code> фільтрує значення менше ніж заданий параметр, у нашому випадку <code>'CLOUD_COVERAGE_ASSESSMENT'</code></p> <p>кількість хмар, <code>10</code> – відсоток покриття на знімку</p>
---	---

Далі необхідно вибрати найкращий знімок без хмар, який відповідає нашим параметрам. Це можна зробити, додавши змінну:

<pre>var image = ee.Image(img) .filterBounds(geometry).sort('CLOUD_COVER',true).median();</pre>	<p>Змінна <code>image</code> фільтрує територію за допомогою створеного та нарисованого шару <code>geometry</code> за допомогою функції <code>filterBounds</code> та звертається до колекції <code>img</code>, з якої забирає один знімок з медіанним (середнім) значенням <code>.median()</code> хмарності <code>'CLOUD_COVER'</code></p>
---	--

Для того щоб наше дослідження було достовірним, ми створюємо усереднений знімок, а не беремо знімок за конкретну дату, щоб уникнути випадкових помилок.

Змінна візьме з колекції усі зображення без хмар у нашій території інтересу – Шацькому НПП. Тобто алгоритм візьме усі літні зображення, які мають показник хмар менший ніж 10%.

Тепер це зображення можна переглянути, але перед цим потрібно додати до нього параметри візуалізації, які задаються за допомогою такої змінної:

<pre>var visParams = { bands: ['B4','B3','B2'], gamma: 2, min: 300, max: 5000 };</pre>	<p>Змінна <code>visParams</code> має аргументи <code>bands: ['B4','B3','B2']</code>, де перераховані канали, які ми будемо відображати, параметри <code>gamma: 2</code>, <code>min: 300</code>, <code>max: 5000</code> є стандартними для відображення даних Sentinel-2 в природних кольорах;</p>
--	---

Додавання на мапу можна виконати за допомогою команди нижче. Також додаймо інформацію про зображення в консоль.

```
Map.addLayer(image.clip(geometry),visParams,
'Зображення супутникового знімку');
print (image);
```

Map.addLayer стандартна конструкція для додавання шарів на карту (параметр, що буде відображений, параметр візуалізації, 'Назва', булевий (ТАК або НІ, True or False) тип даних для відображення шару)
print виводить результат у консоль

Натискаємо **Run** (Запуск скрипту).

У консолі можна побачити, що до стандартних каналів Sentinel-2 додався канал з NDVI.

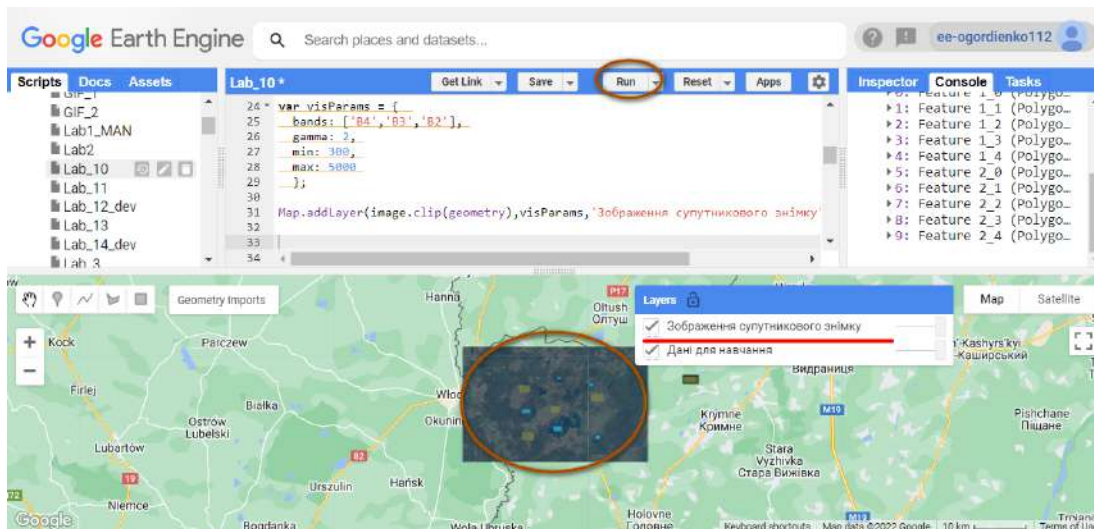


Рис. 193. Запуск скрипту

5. Застосувати класифікацію Random Forest.

Наступним кроком є створення алгоритму для машинного навчання *Random Forest*, де ми додаємо канали, за якими алгоритм буде навчатися.

Random Forest (Випадковий ліс) – це комплексний метод навчання для класифікації, регресії та інших завдань, який діє шляхом побудови безлічі дерев рішень під час навчання. Перший алгоритм для Random Forest був створений у 1995 р. Тін Кам Хо. Нині в Google Earth Engine та в інших сервісах компанії «Google» використовується цей метод.

Для початку створюємо змінну, що «витягує» зразки пікселів, за якими буде навчатися алгоритм.

```
var image = image.select(bands);
var training = image.sampleRegions({
  collection: polygons,
  properties: ['class'],
  scale: 30
});
```

image вибирає, які саме канали необхідні для навчання;
training функція, яка за допомогою sampleRegions перетворює кожен піксель на об'єкт і повертає колекцію об'єктів, за основу бере змінну polygons з об'єднаними lake та land, що ми створили, і до властивостей додає клас 1 або 0, масштаб 30;

Далі створюємо класифікатор *Random Forest* і вказуємо «кількість дерев», за якими алгоритм буде виконувати свої розрахунки. Ці «дерева» є розгалуженнями шляхів, проте алгоритм випадково вибирає шлях, яким піде. Одне дерево не дає високої точності класифікації, але через їх велику кількість результат буде точнішим.

```
var classifier = ee.Classifier.smileRandomForest({
  numberOfTrees: 30,
});
```

classifier, що звертається до класифікатора Random Forest. Тут лише один параметр – кількість дерев, за якими алгоритм буде вирішувати, що саме за об'єкт і якому класу він належить;

Далі застосовуємо команду, яка навчає алгоритм за створеними нами класами і класифікує зображення.

<pre>var trained = classifier.train(training, 'class', bands); var classified = image.classify(trained);</pre>	<p><code>train</code> навчає класифікатор набору об'єктів, використовуючи вказані числові властивості кожної функції як навчальні дані. Геометрія елементів ігнорується. <code>classify</code> класифікує кожну ознаку в колекції;</p>
--	--

Після цього можна додавати візуалізацію до створених алгоритмом класів.

<pre>var palette = ['#858b45', '#27c8ff'];</pre>	<p><code>palette</code> змінна для відображення, параметри передаються для першого та другого об'єкта через кому в кодуванні HEX;</p>
--	---

На початку, коли ми створювали класи *lake* і *land*, можна було змінити колір їх відображення. Щоби створені нами класи мали такі самі кольори, як і відкласифікує алгоритм *Random Forest*, можна зайти в налаштування класів, скопіювати їх з відповідного вікна і вставити до попередньої змінної (спочатку йде *lake*, потім *land*).

? Завдання для перевірки

<p>Задайте колір класу, який буде виводитися на екран, – такий самий, яким ви створили його в другому кроці (Створити два нові шари геометрії (суша – <i>land</i> та озеро – <i>lake</i>) для подальшого набору еталонів водойм і суші, які будуть використовуватися в класифікації під час розпізнавання території).</p> <p>Скористайтеся зображенням нижче, щоб зрозуміти, звідки можна взяти колір. Вставте два кольори в змінну <code>palette</code></p>
--

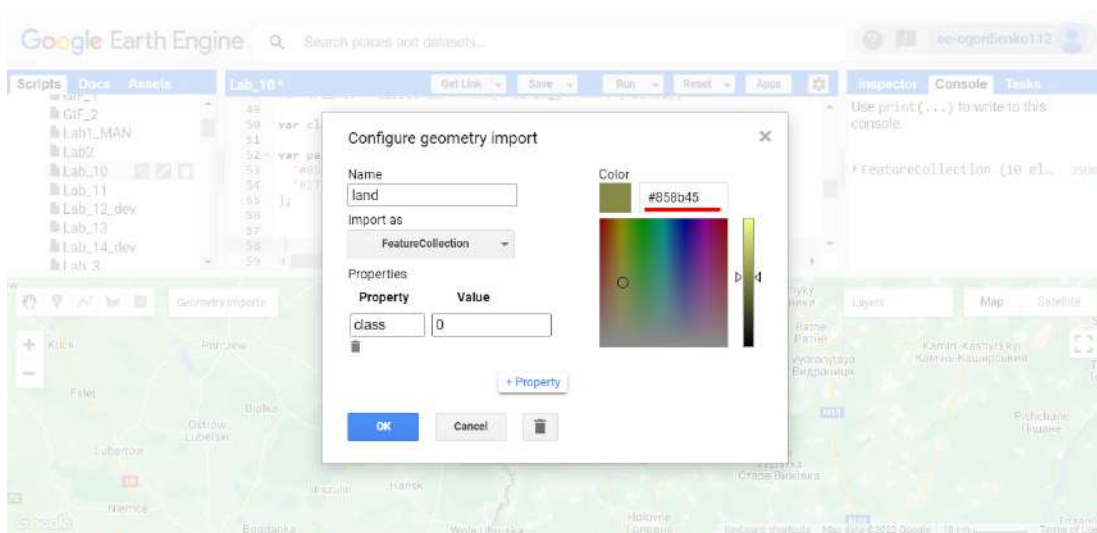


Рис. 194. Зміна відображення шару

Останнім етапом є додавання класифікованого зображення до мапи. Це можна зробити за допомогою команди:

<pre>Map.addLayer(classified.clip(geometry),{min: 0, max: 1, palette: palette},'Класифіковане зображення');</pre>	<p><code>Map.addLayer</code> стандартна конструкція для додавання шарів на мапу (параметр, що буде відображений, параметр візуалізації, 'Назва', булевий тип (ТАК або НІ, True or False) даних для відображення шару)</p>
---	---

Натискаємо **Run** (Запуск скрипту).

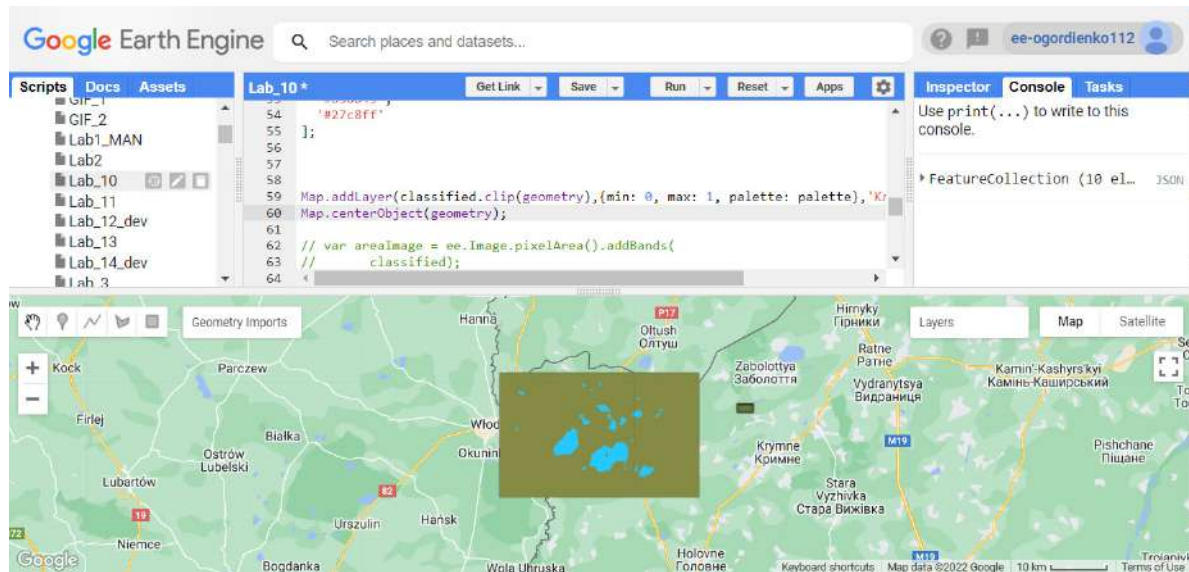


Рис. 195. Навчальні дані

Результатом є класифіковане на два класи зображення території Шацьких озер за літо 2021 р.: озера та інша поверхня.

Щоби повторно запустити код і наблизитися до нашої території інтересу, слід додати такий рядок:

<code>Map.centerObject(geometry);</code>	<code>Map.centerObject</code> стандартна конструкція для того, щоб призумитися до геометричних об'єктів, що створені або додані як шар (назва об'єкта, масштаб (1–20))
--	--

6. Отримати площі всіх Шацьких озер за 2020 і 2021 рр.

Далі нам необхідно порахувати площу озер за різні роки. Для цього скористаємося такими командами:

<code>var areaImage = ee.Image.pixelArea().addBands(classified);</code>	Створює зображення, в якому значення кожного пікселя є площею цього пікселя у квадратних метрах. Зображення має один канал під назвою "area" та один канал з назвою "classification". Це можна дізнатися в метаданих датасету
---	---

Цей рядок коду додасть змінну, в якій будуть зберігатися пікселі для розрахунку площі.

Наступним кроком є створення змінної, якою ми скористаємося, щоб звернутися до нашої території інтересу і задати параметри, за якими буде виконуватися розрахунок площі.

<code>var areas = areaImage.reduceRegion({ reducer: ee.Reducer.sum(), groupField: 1, groupName: 'class', }), geometry: geometry, scale: 500, maxPixels: 1e10 });</code>	Застосовує редуктор до всіх пікселів у певній області. <code>ee.Reducer.sum()</code> рахує суму у всіх заданих об'єктах, у нашому випадку – для класів, що ми створили Повертає словник вихідних даних редуктора
---	---

Після цього нам потрібно створити список зі значень, для того щоб GEE міг до нього звернутися.

<pre>var classAreas = ee.List(areas.get('groups'));</pre>	<p>classAreas створює окремий список із значеннями класу для 0 та 1</p>
---	---

Наступну функцію ми застосовуємо до попередньої змінної. Функцію, яка переведе наші одиниці розрахунку до значень у кілометрах квадратних. Вона звертається до словника, що містить наші значення:

<pre>var classAreaLists = classAreas. map(function(item) { var areaDict = ee.Dictionary(item); var classNumber = ee.Number(areaDict. get('class')).format(); var area = ee.Number(areaDict.get('sum')).divide(1e6).round(); return ee.List([classNumber, area]); });</pre>	<p>За допомогою функції item береться словник, потім беруться всі значення 'class' у цьому словнику; За допомогою 'sum' рахується їх сума і значення конвертуються в метри квадратні за допомогою множення 1e6, ми маємо площу пікселів;</p>
---	--

Останнім кроком у розрахунку площ є згладжування колекції і виведення на екран значень площі:

<pre>var result = ee.Dictionary(classAreaLists. flatten()); print(result);</pre>	<p>flatten() перетворює декілька списків на один список; Побачити це можна за допомогою print</p>
--	---

Натискаємо **Run** (Запуск скрипту).

Після цих дій можна побачити площу в консолі:

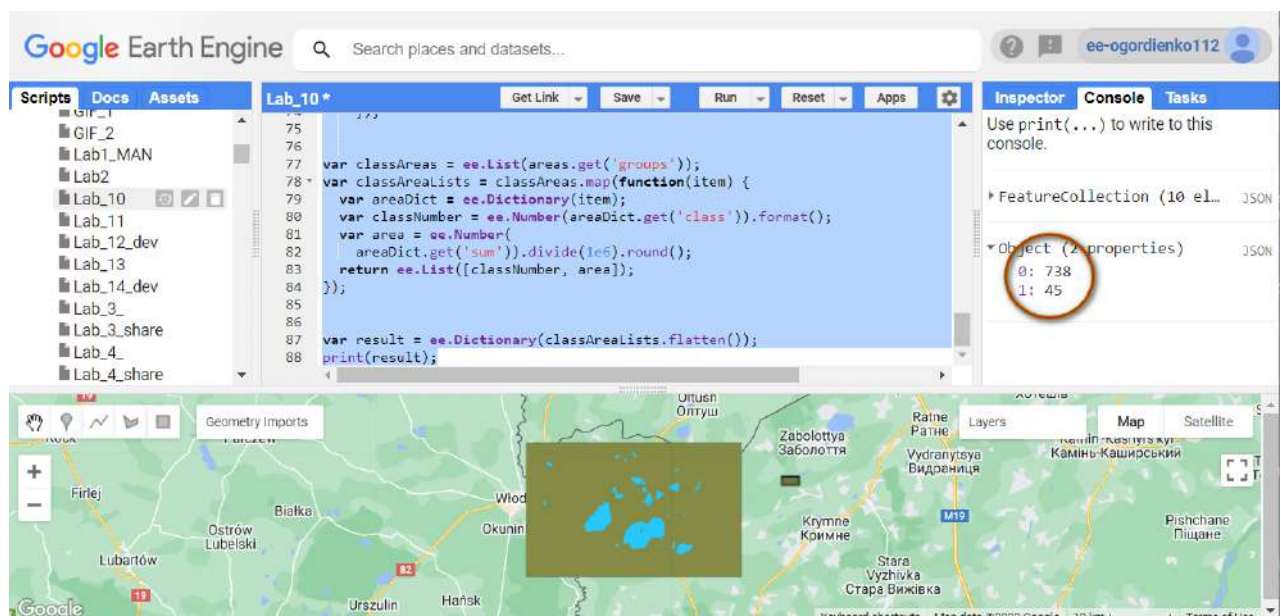


Рис. 196. Площа за навчальними даними

Це площа класу озер і класу всього іншого в нашій зоні інтересу. Дані можуть бути різними, це залежить насамперед від розміру території інтересу, яку ви нарисували. Щоби порівняти, як змінювалася площа водної поверхні, змінімо дати в рядку.

? Завдання для перевірки

Змініть дати в змінній start і finish із 2021 р. на 2020 р.
Порівняйте виведені в консоль значення. Уважно розгляньте два результати класифікації і дайте відповідь, як змінилася територія Шацьких озер. Як ви гадаєте, що вплинуло на зменшення їх площі?

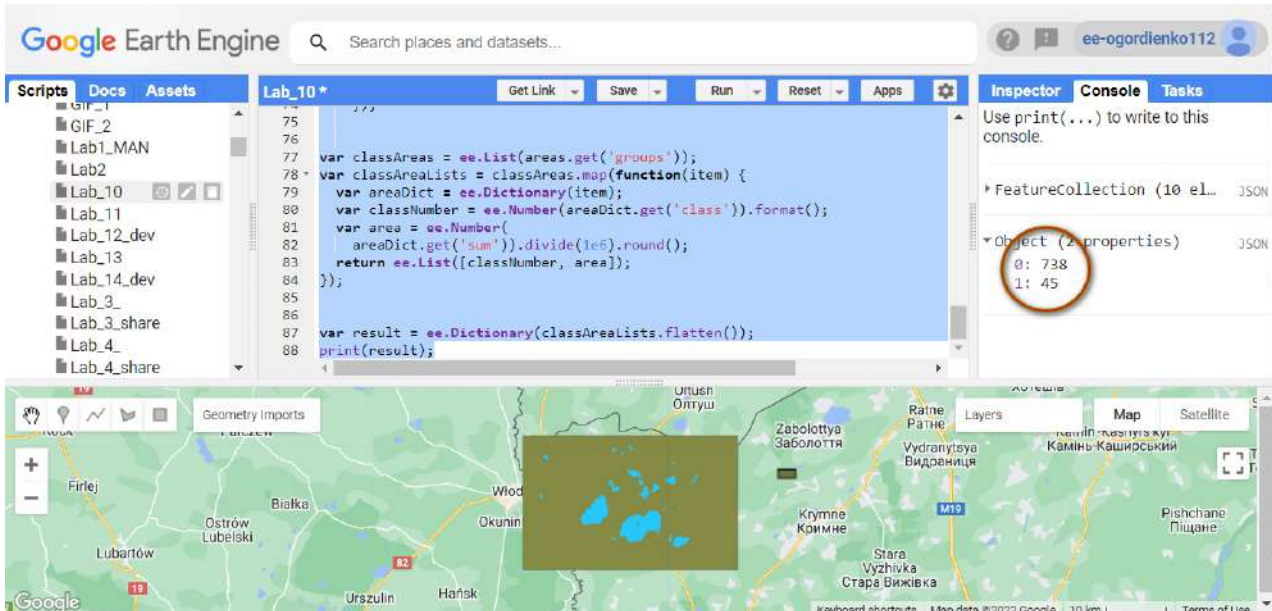


Рис. 197. Зміна площі за роки

За наведеним нижче QR-кодом ви можете переглянути відео виконання практичної роботи.



Рис. 198. Відео виконання практичної роботи

Готовий код можна відкрити за лінком:

https://code.earthengine.google.com/?scriptPath=users%2Fgis_rs%2Fcourse_MAN%3ALab_10

Візуалізація растрових та векторних даних з відображенням легенди на мапі для дослідження впливу промислових об'єктів на стан атмосферного повітря

Ситуація

За даними Всесвітньої організації охорони здоров'я (ВООЗ), забруднення повітря становить серйозну загрозу для здоров'я людей і впливає на кліматичну ситуацію на планеті. Щорічно воно призводить до майже семи мільйонів передчасних смертей, пов'язаних здебільшого з інсультами, серцевими захворюваннями, хронічними захворюваннями легень, раком легень та гострими респіраторними інфекціями. За статистикою 9 з 10 людей дихають повітрям із високим вмістом забруднювальних речовин¹. Основними джерелами забруднення повітря в Україні є металургійні заводи і вугільні електростанції, хімічне виробництво і машинобудування. Найвищі концентрації забруднення спостерігаються поблизу великих міських промислових центрів.

За допомогою супутника «Sentinel 5P» можна відстежити рівні забруднення атмосферного повітря сполуками NO₂, SO₂, CO тощо.

Завдання

Проаналізувати вплив вугільних теплоелектростанцій на стан атмосферного повітря у 2021 р.

Алгоритм виконання завдання:

1. Наблизити карту до території інтересу на Європу.
2. Вибрати часовий період із 1 січня 2021 р. по 1 січня 2022 р.
3. Підвантажити датасет Sentinel-5p для NO₂ і відфільтрувати його за вибраним періодом, порахувати усереднене значення за 2021 р.
4. Додати датасет LSIB (Large Scale International Boundaries) для візуалізації даних лише в межах суходолу.
5. Задати параметри візуалізації шару і застосувати візуалізацію до мапи.
6. Додати легенду до мапи.
7. Додати дані, змінити значення кольорів на коректні, візуалізувати легенду.
8. Підвантажити датасет із тепловими електростанціями (ТЕС), відфільтрувати ті, які працюють на вугіллі.
9. Зберегти усереднений растр для NO₂ за 2021 р. для заданої ділянки.

Покрокова інструкція

1. Наблизити карту до території інтересу на Європу.

Відкрийте **Code Editor** (Редактор коду): <https://code.earthengine.google.com/> і встановіть зум мапи на таке значення: 16, 49, 4. Натисніть **Run** (Запуск скрипту).

<code>Map.setCenter(16, 49, 4);</code>	<code>Map.setCenter</code> задає центр картографічного вікна 16 – перше значення в дужках змінює положення мапи в напрямку схід-захід, 49 – друге значення – північ-південь, 4 – третє значення – зум
--	---

¹ URL: <https://www.who.int/news/item/02-05-2018-9-out-of-10-people-worldwide-breathe-polluted-air-but-more-countries-are-taking-action>

Мапа наблизиться до Європи.

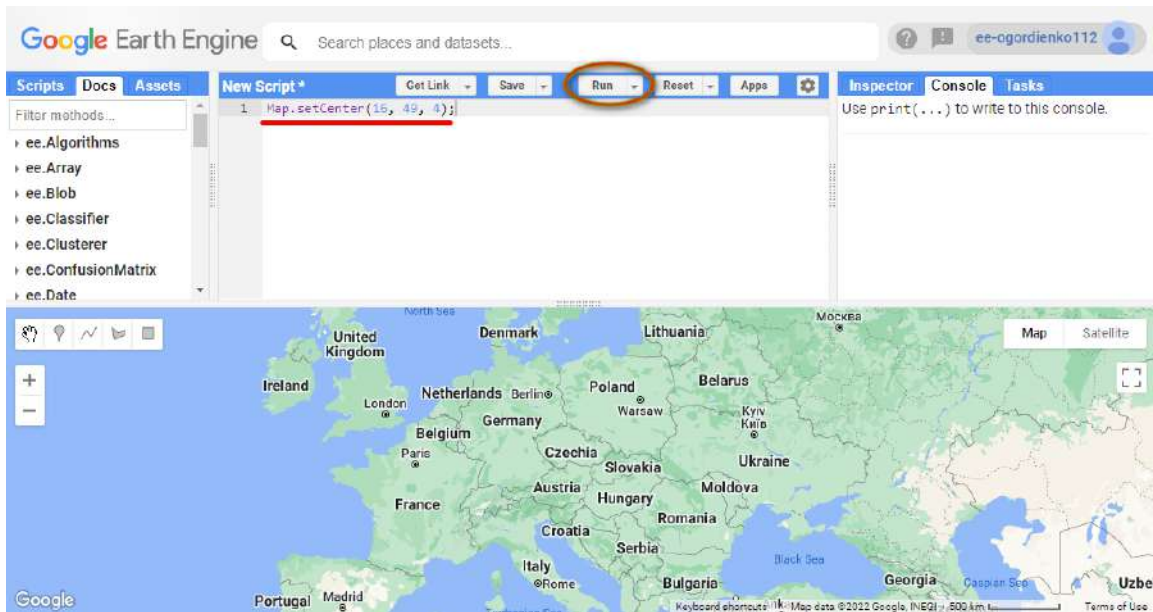


Рис. 199. Центрування мапи

2. Вибрати часовий період з 1 січня 2021 р. по 1 січня 2022 р.

Тепер необхідно вибрати часовий проміжок для відображення даних. Для цього скористайтеся такими змінними:

<pre>var start = '2021-01-01'; var end = '2022-01-01';</pre>	start початкова дата end кінцева дата
--	--

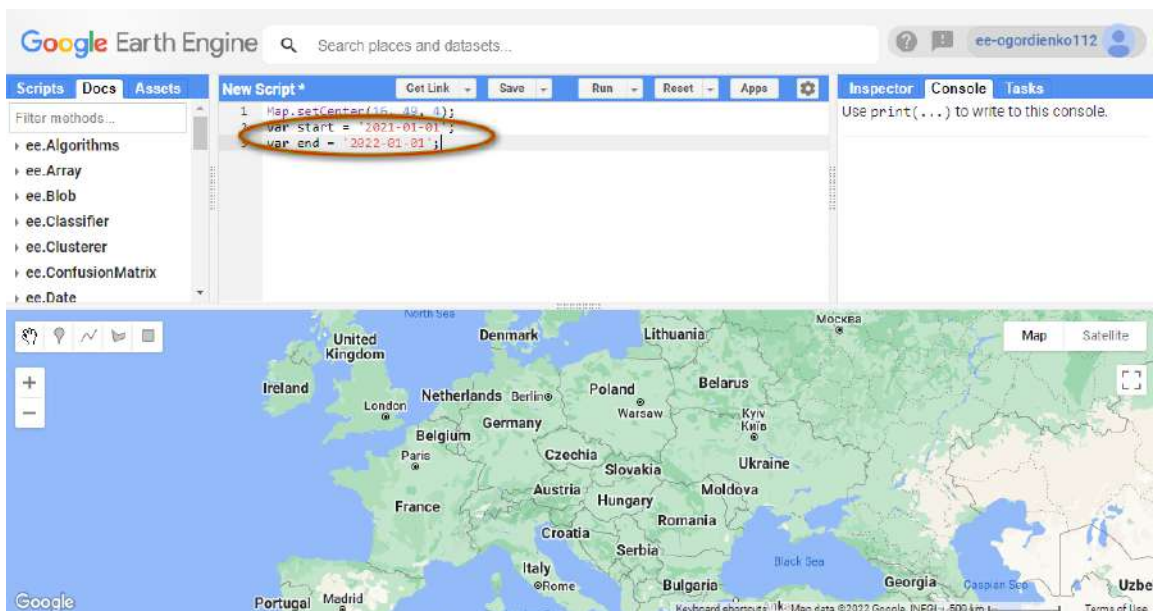


Рис. 200. Задавання часового проміжку

Часовий проміжок за весь 2021 р. буде відображатися на екрані.

3. Підвантажити датасет Sentinel-5р для NO₂ і відфільтрувати його за вибраний період, порахувати усереднене значення за 2021 р.

За допомогою цієї змінної можна відобразити концентрацію NO₂ і відсортувати дані за створеними вище датами:

<pre>var collection = ee.ImageCollection('COPERNICUS/S5P/OFFL/L3_NO2') .select('tropospheric_NO2_column_number_density') .filterDate(start, end);</pre>	<p>collection за допомогою ee.ImageCollection звертається до бази даних Sentinel-5P OFFL NO2: Offline Nitrogen Dioxide select обирає значення зі щільністю NO₂ filterDate фільтрує дані за заданими датами</p>
---	--

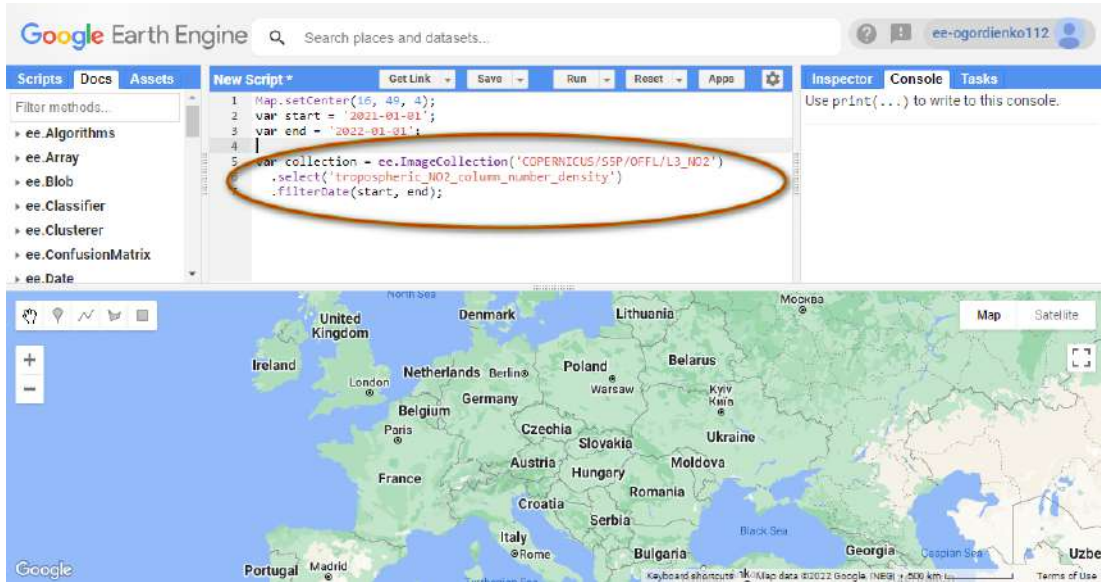


Рис. 201. Додавання колекції

Наступна змінна візьме на вхід усі супутникові зображення, відсортовані за типом забруднювача, датами, і порахує для них одне усереднене значення.

<pre>var mean = collection.mean();</pre>	<p>рахує середнє значення mean() у collection знімків</p>
--	--

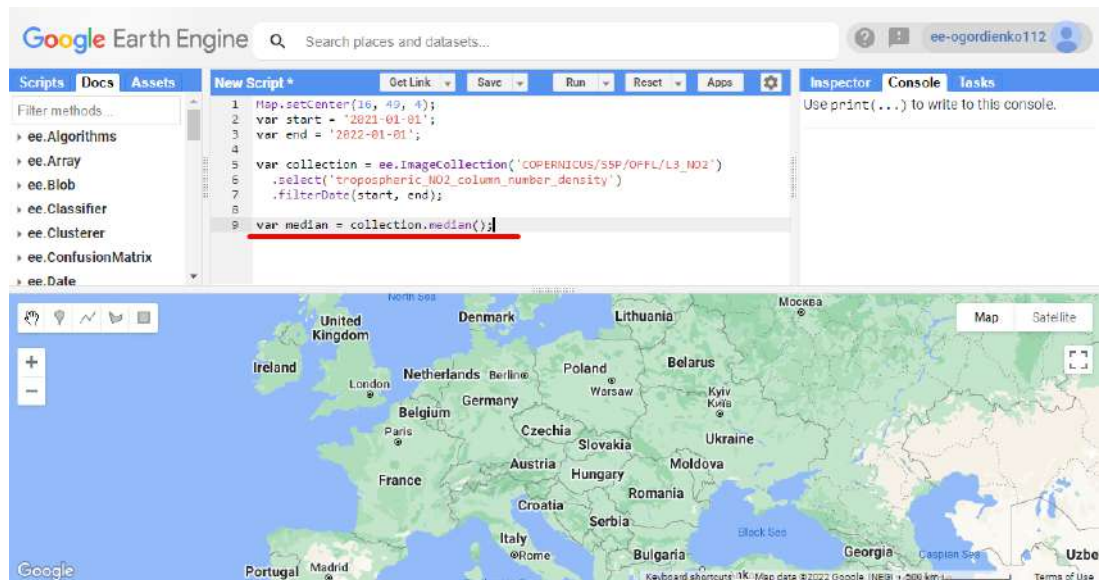


Рис. 202. Зведення колекції до медіани

4. Додати датасет LSIB (Large Scale International Boundaries) для візуалізації даних лише в межах суходолу.

Для того щоб шар візуалізувався лише в межах суходолу, можна скористатися датасетом *LSIB (Large Scale International Boundaries)*. Це набір даних від *The United States Office of the*

Geographer (Управління географів Сполучених Штатів Америки), версія отримана з двох інших наборів даних: векторного файлу кордонів країн і берегових ліній. У наборі даних спрощені лінії кордонів (рідко зміщуються більш ніж на 100 метрів) і берегова лінія зазвичай зміщена від 100 метрів до 1 кілометра. У цьому спрощеному наборі даних відсутні середні та малі острови.

У рядку пошуку набираємо LSIB й імпортуємо датасет з ім'ям LSIB 2017: Large Scale International Boundary Polygons, Simplified.

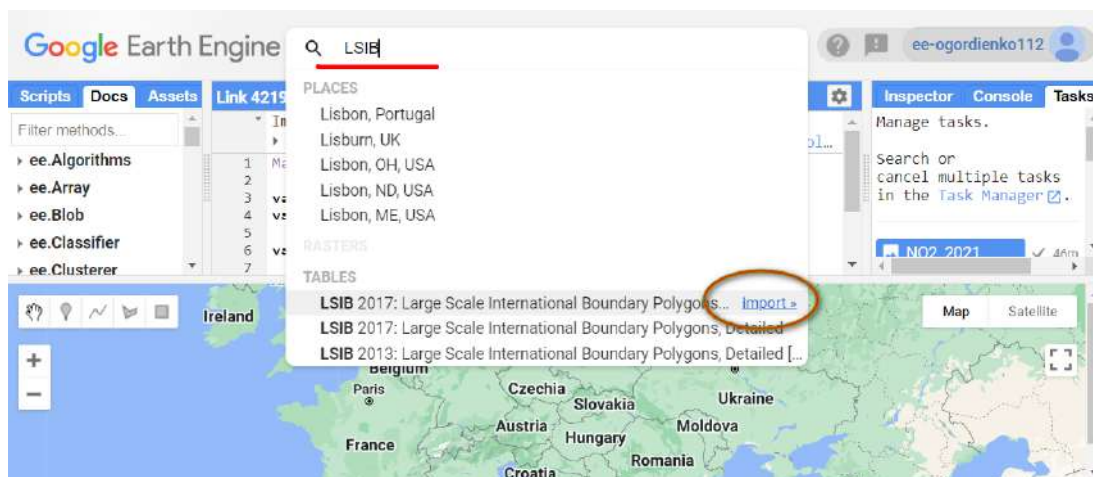


Рис. 203. Імпорт даних

Задайте датасету ім'я LSIB.

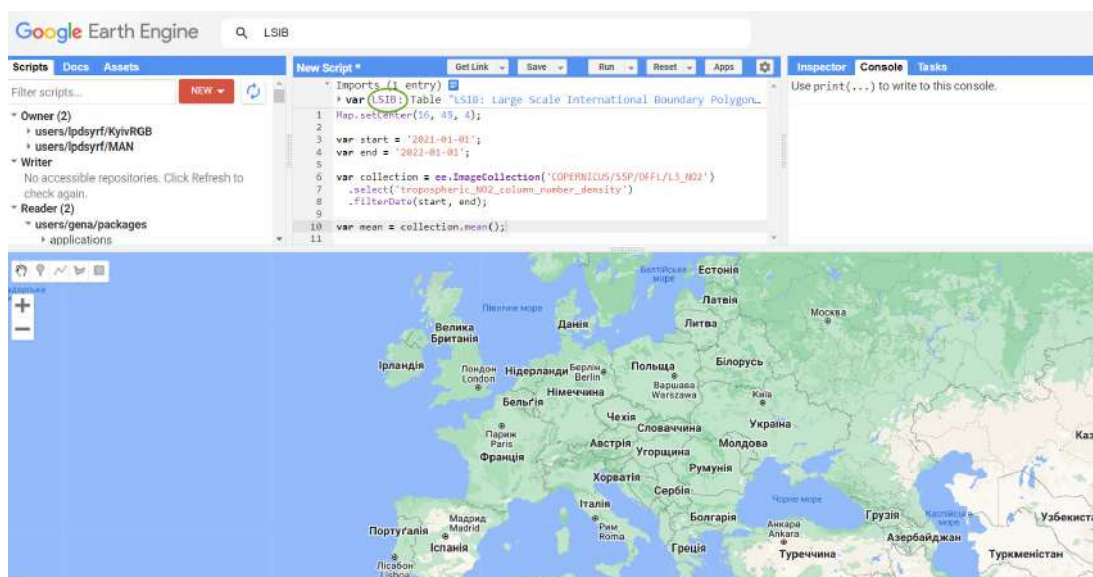


Рис. 204. Зміна назви шару

Цей шар буде використано як шар для обрізки наступних даних.

Наступний крок – додати шар на мапу. Зробити це можна за допомогою команди “Map.addLayer”. Натиснути **Run** (Запуск скрипту).

<pre>Map.addLayer(mean.clip(LSIB), {}, 'S5P N02')</pre>	<p>Map.addLayer додає шар до мапи, в дужках зазначаються умови для цього шару: mean – усереднене значення clip(LSIB) – обрізане по шару LSIB {} – тут пізніше вкажемо параметри візуалізації даних 'S5P N02' – назва колекції знімків</p>
---	---

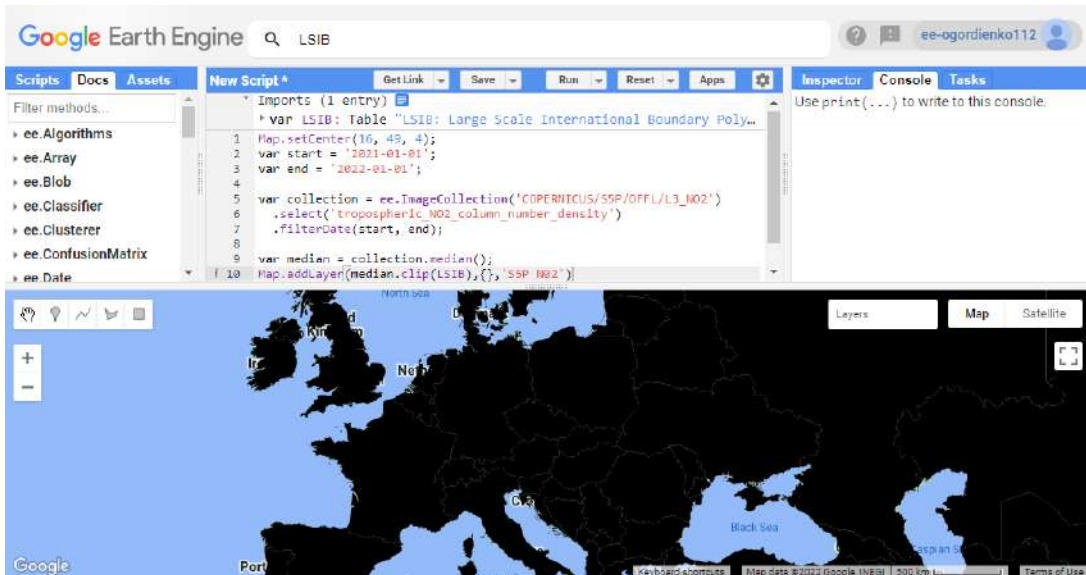



Рис. 205. Відображення всіх даних

5. Задати параметри візуалізації шару і застосувати візуалізацію до карти.

Шар візуалізується неправильно: він має чорну суцільну заливку і не відображає зміни значень у кольорі. Потрібно задати йому візуальні параметри. Для цього скористайтесь інструментом налаштування , що міститься в **Layers** (Шари).

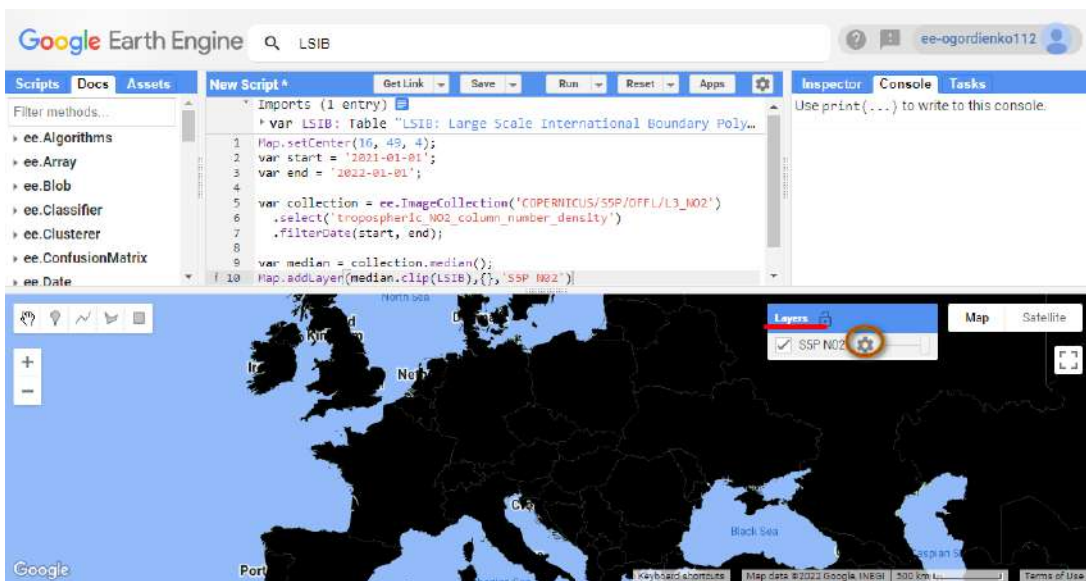


Рис. 206. Налаштування відображення

У новому вікні можна візуалізувати дані, які додаються на карту. Виберіть розтягування гістограми **Stretch:1 σ** (Розтягування: 1 σ). І для того, щоб задавати кольори, натисніть на вкладку **Palette** (Палітра). Скористайтесь інструментом із зображенням олівця і плюсиком для додавання нових кольорів. Радимо використовувати світліші відтінки для позначення найнижчих значень, а темніші – для найбільших.



Рис. 207. Налаштування відображення

Поступово додайте шість кольорів до палітри, перетягуючи значок кружечка  на кольоровій палітрі.

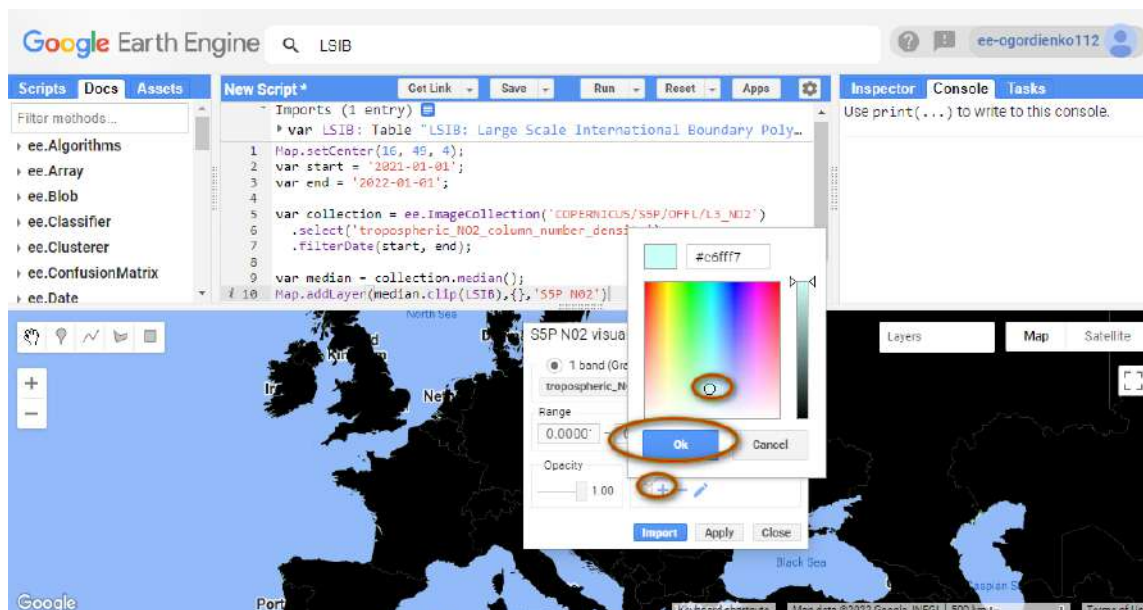


Рис. 208. Зміна відображення даних

? Завдання для перевірки

Додайте шість кольорів до палітри

Після того як ви додасте шість кольорів до палітри, натисніть спочатку **Apply** (Виконати), а потім **Import** (Імпортувати). Після цього ваші налаштування з'являться у вкладці **Imports** (Імпорт). Ця операція може зайняти трохи часу.

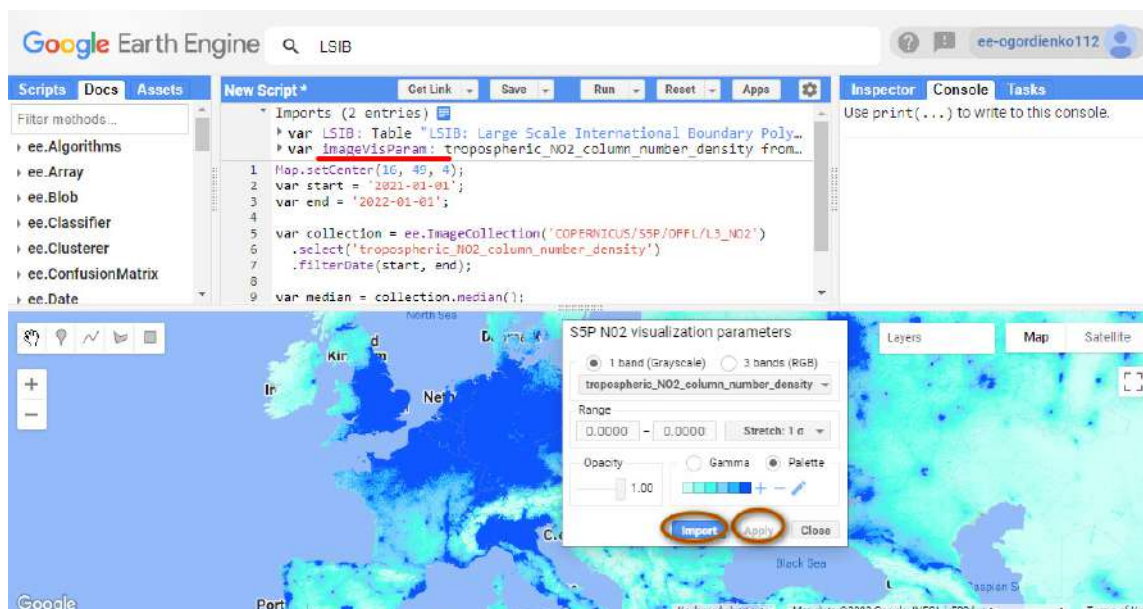


Рис. 209. Зміна відображення на мапі

Тепер візуалізацію, яку ви створили, можна додати до мапи. У попередній змінній замініть фігурні дужки `{}` на значення `imageVisParam`.

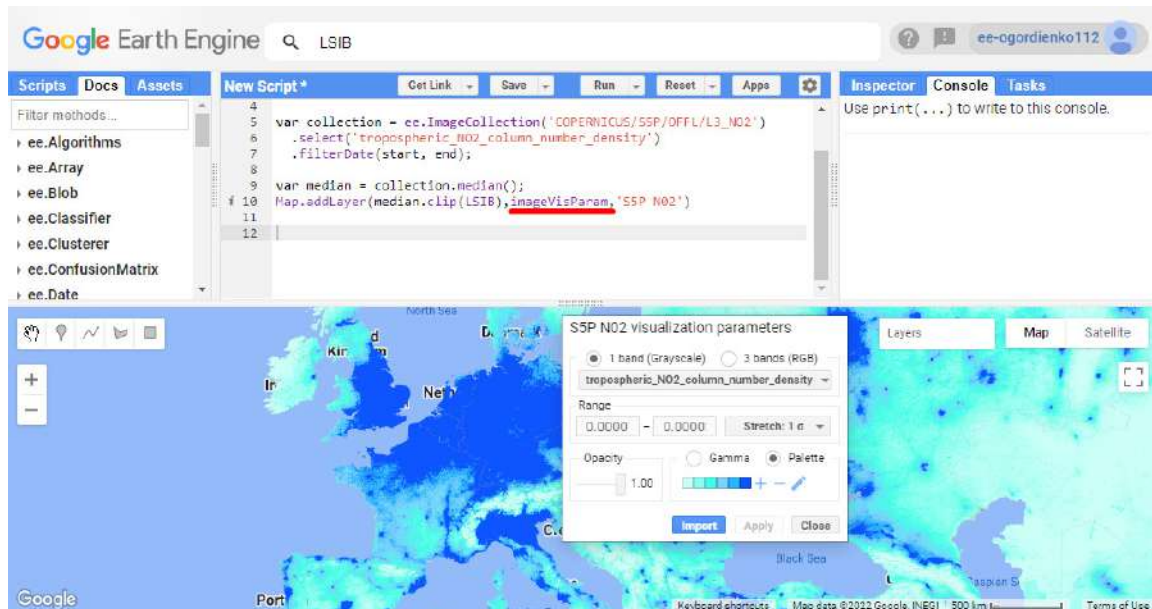


Рис. 210. Імпортовані параметри відображення

Можна закрити вікно налаштування палітри за допомогою кнопки **Close** (Закрити).

6. Додати легенду до мапи.

Для того щоб створити блок легенди і розмістити її в лівому нижньому куті мапи, скористайтеся кодом:

<pre>var legend = ui.Panel({ style: { position: 'bottom-left', padding: '8px 15px' } }); Map.add(legend);</pre>	<p>Додає змінну legend style – вибір її розміщення і розмірів Map.add(legend) додає легенду до картографічного вікна</p>
---	--

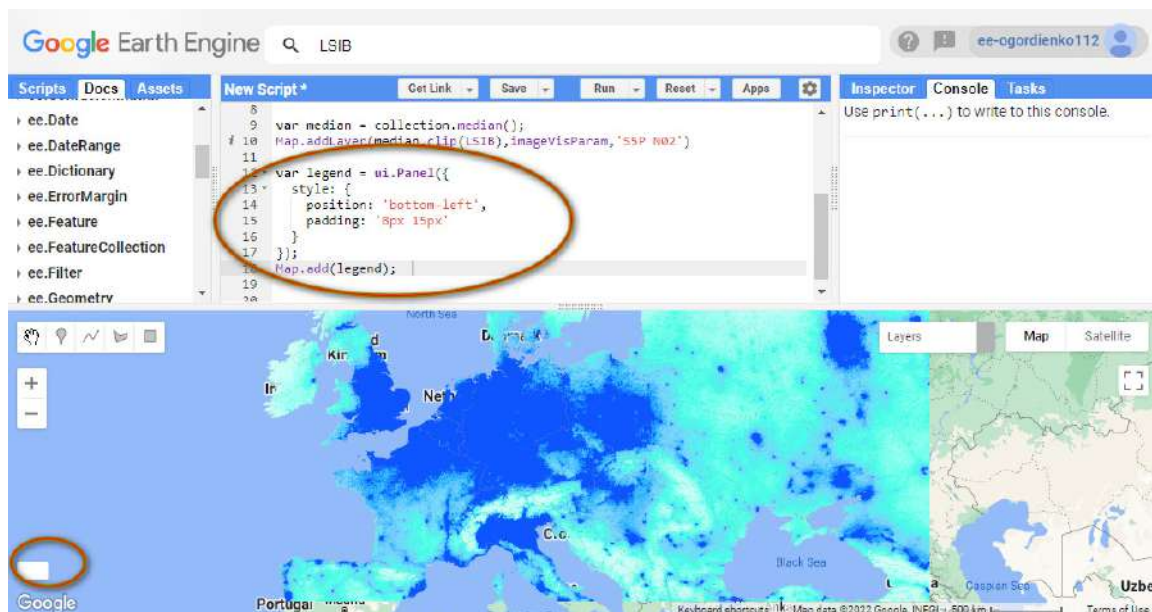


Рис. 211. Створення легенди

Нове вікно з'являється в лівому нижньому кутку, але поки в ньому немає жодних даних. Тому наступні кроки будуть описувати, як додати дані в легенду.

7. Додати дані, змінити значення кольорів на коректні, візуалізувати легенду.

Щоб задати назву в легенді Sentinel-5 NO₂ з налаштуваннями шрифту та відносною позицією, скористайтеся кодом:

<pre>var legendTitle = ui.Label({ value: 'Sentinel-5 NO2', style: { fontWeight: 'bold', fontSize: '18px', margin: '0 0 4px 0', padding: '0' } }); legend.add(legendTitle);</pre>	<p>legendTitle додає заголовок до легенди 'Sentinel-5 NO2' – текст у заголовку параметри відображення: жирний шрифт розмір 18</p> <p>додає заголовок легенди у вікно мапи</p>
--	---

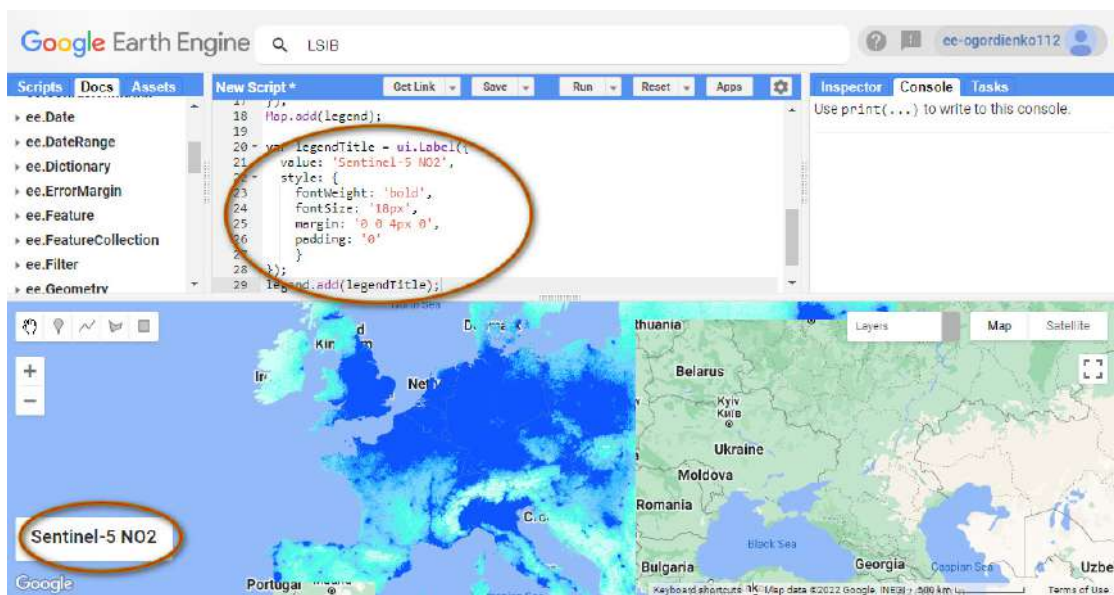


Рис. 212. Створення назви легенди

Назва з'являється в новому створеному блоці.

За допомогою наступної функції коду додаються дві змінні до легенди, які мають на вхід значення кольору й опису.

<pre>var makeRow = function(color, name) { var colorBox = ui.Label({ style: { backgroundColor: '#' + color, padding: '8px', margin: '0 0 4px 0' } }); var description = ui.Label({ value: name, style: {margin: '0 0 4px 6px'} }); return ui.Panel({ widgets: [colorBox, description], layout: ui.Panel.Layout.Flow('horizontal') }); };</pre>	<p>функція makeRow бере на вхід два значення color та name за допомогою ui.Label повертає стиль віджета, який можна змінити, щоб оновити стилі віджета. backgroundColor '#' вказує на початок за id елемента для кольорів у легенді. Знак # означає, що береться початок рядка padding відступ у пікселях margin відступ у пікселях description за допомогою ui.Label задає відступи та створює місце, куди буде розміщений текст</p> <p>return повертає за допомогою ui.Panel віджети для colorBox – кольорових елементів та description – текстової інформації. ui.Panel.Layout.Flow повертає макет, який розміщує віджети горизонтально, в робочому середовищі.</p>
--	--

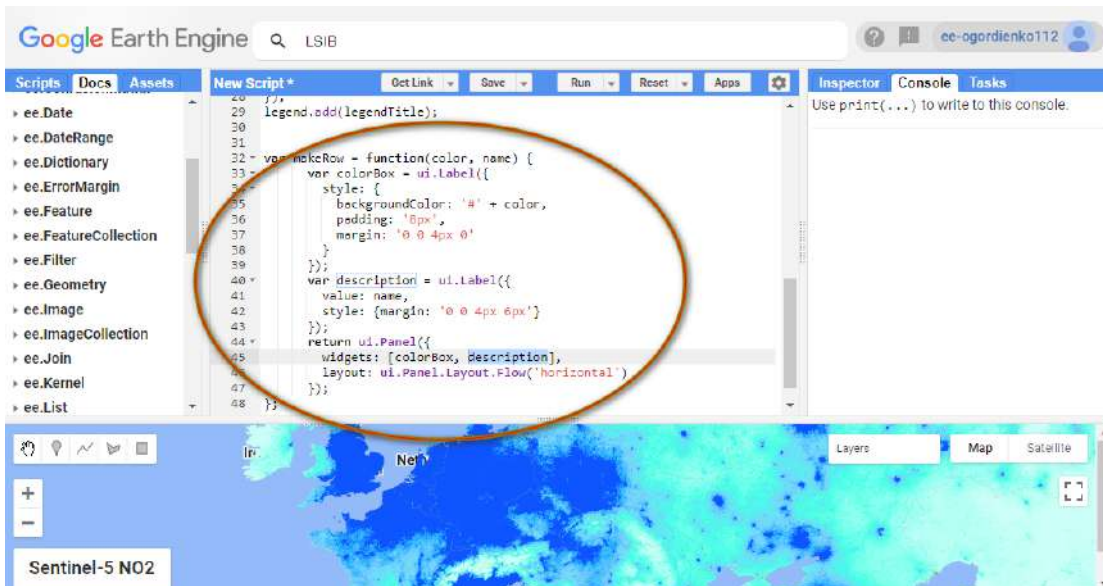


Рис. 213. Додавання даних до легенди

Ця змінна додає значення до наведених вище змінних і задає їм коректні значення.

✓ Важливо! У коді нижче всі значення *palette* дорівнюють *ff0000*, тобто червоному кольору. У пункті 7 були створені кольори для візуалізації, ними потрібно замінити всі значення *ff0000*.

<pre> var palette=["ff0000", "ff0000", "ff0000", "ff0000", "ff0000", "ff0000"]; var names = ['низький показник NO2', 'нижче середнього показник NO2', 'середній показник NO2 ', 'вище середнього показник NO2', 'високий показник NO2', 'дуже високий показник NO2', 'ТЕС']; </pre>	<p><i>palette</i> списком у [] квадратних дужках задаються кольори.</p> <p><i>names</i> списком у [] квадратних дужках задаються назви підписів</p>
---	---

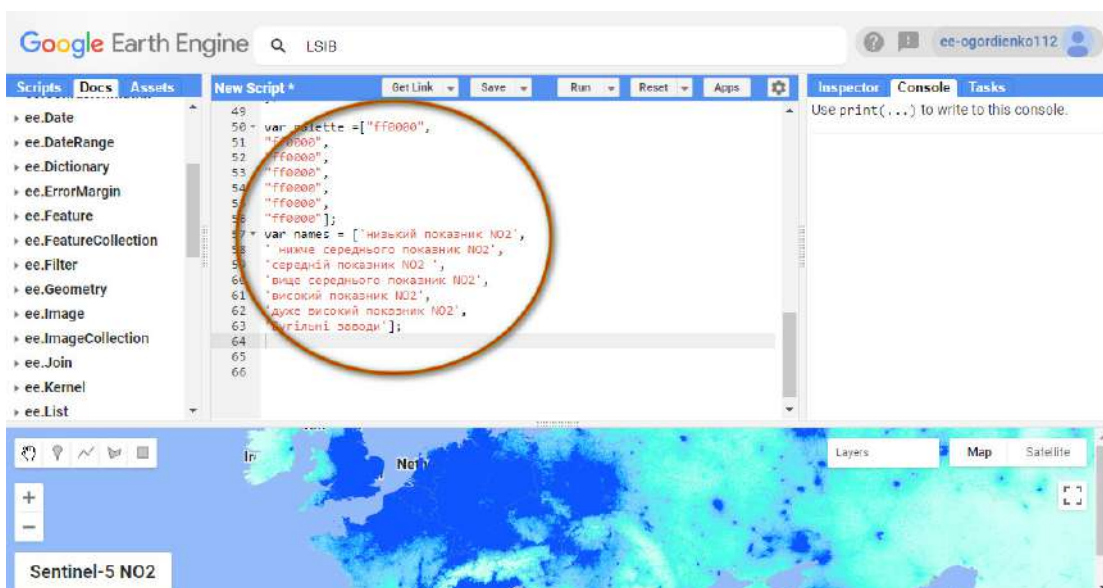


Рис. 214. Додавання візуалізації до легенди

Для того щоб вставити правильні кольори, потрібно звернутися до вкладки **Imports** (Імпорт). Натисніть на кнопку із зображенням листка 📄.

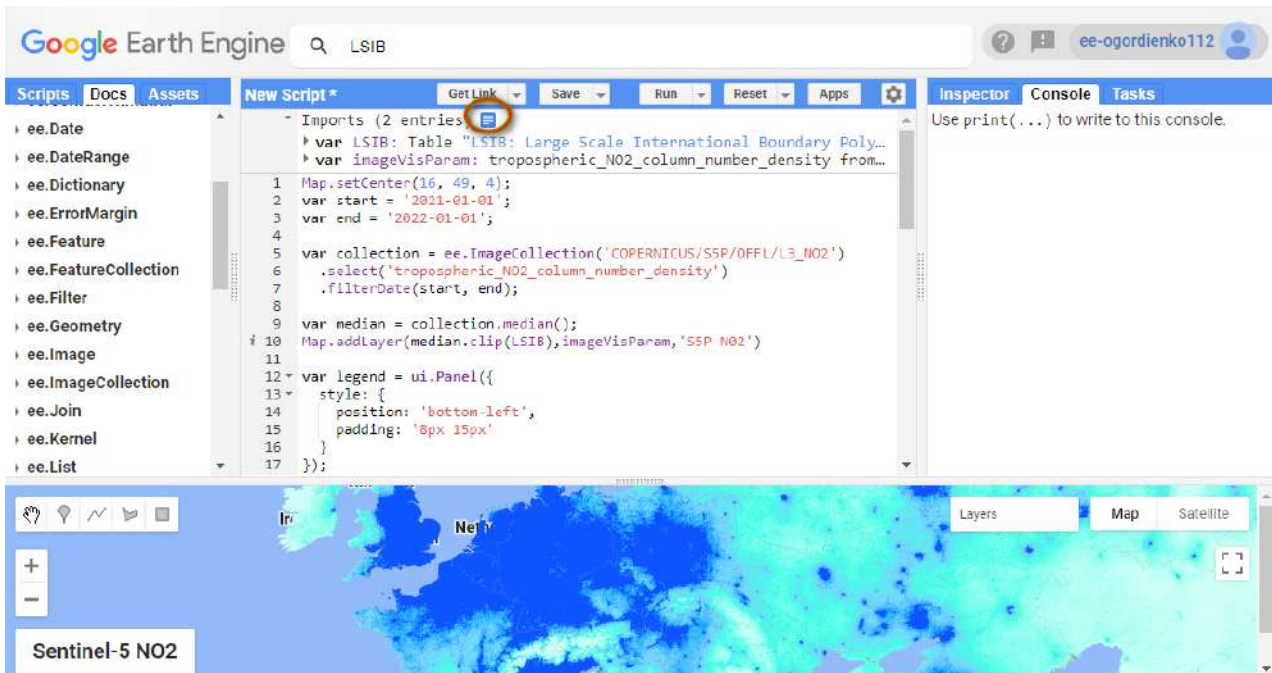


Рис. 215. Імпортовані дані в скрипті

Відкривається нове вікно, в якому зберігаються значення, які ви імпортували до датасету. Тут зберігається колекція, за допомогою якої вирізаний суходіл, а також потрібні параметри візуалізації.

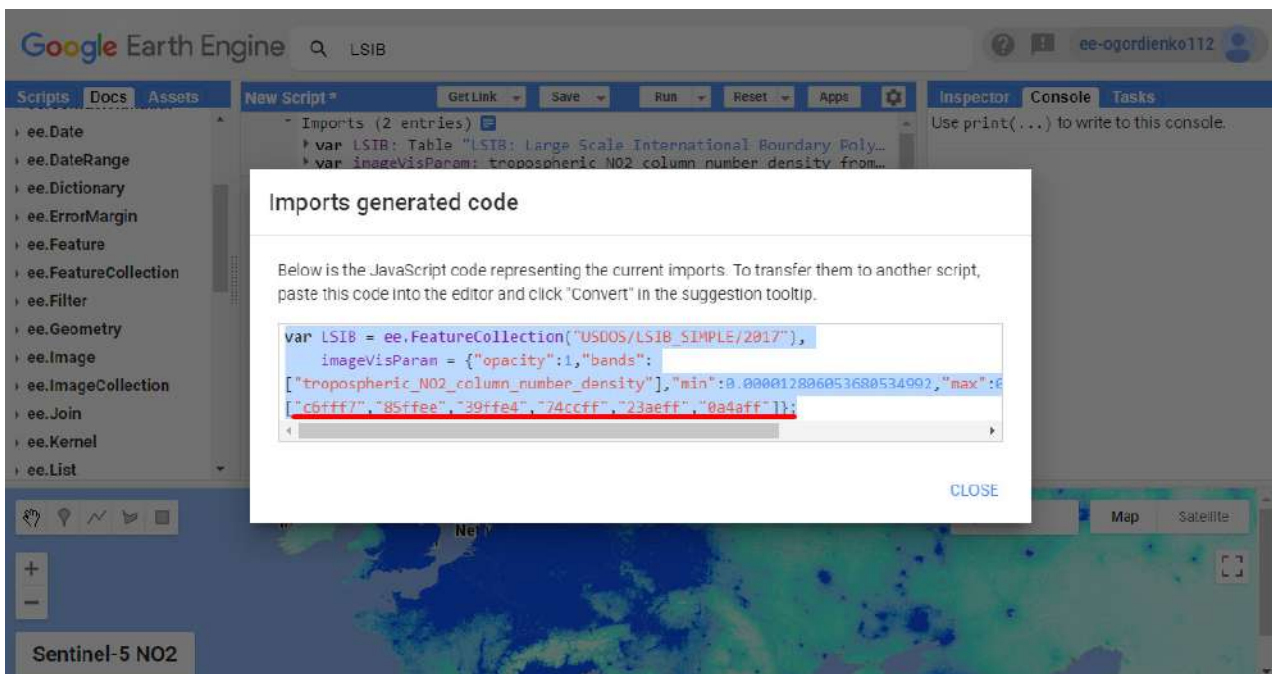


Рис. 216. Копіювання даних

Скопіюйте тільки параметри візуалізації.

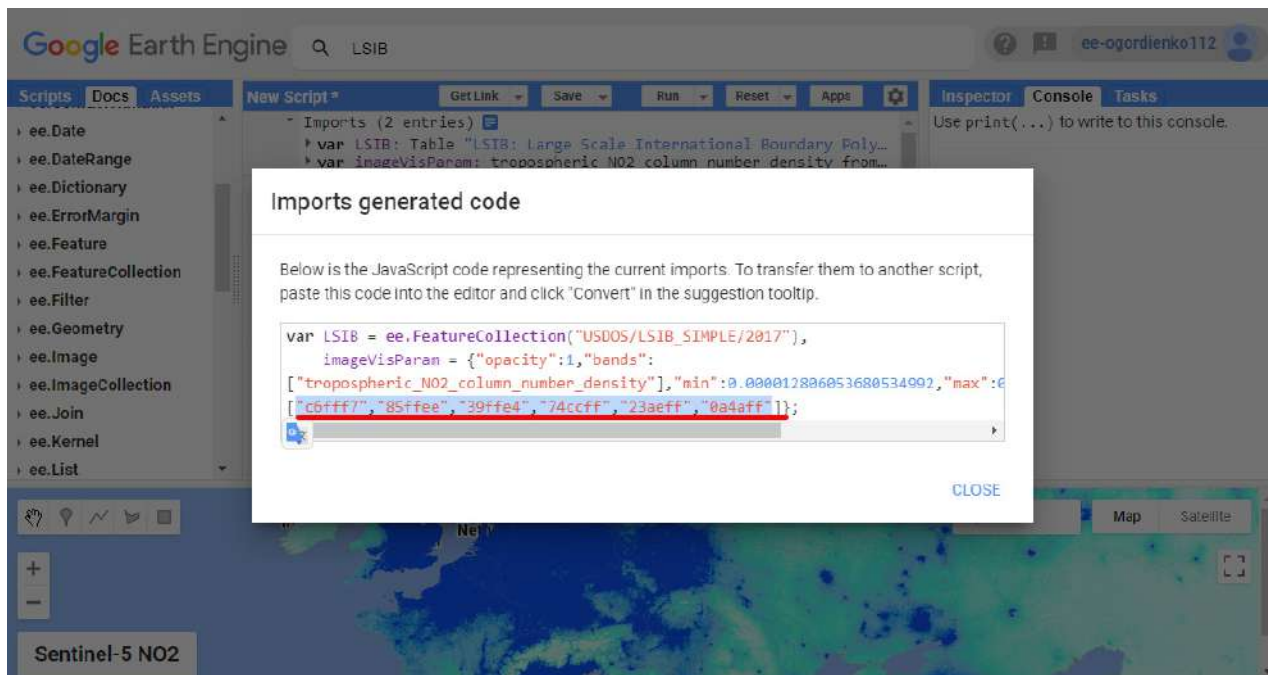


Рис. 217. Відображення даних, як на мапі

Тепер ці значення можна вставити в легенду. Видалить усі значення ff0000 і вставте нові, як на наступному зображенні.

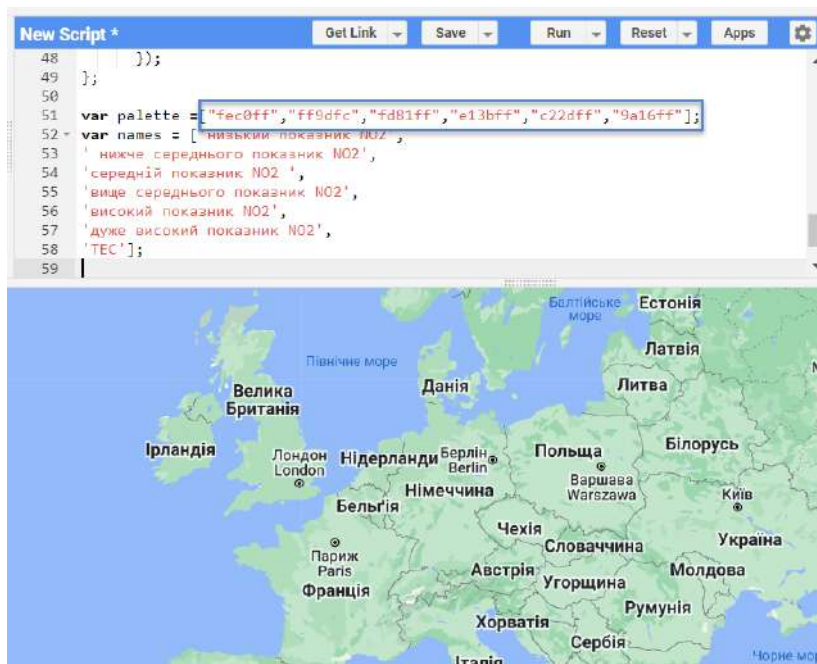


Рис. 218. Зміна відображення легенди

Також у легенді мають бути ТЕС, які зараз не підвантажені, тому одразу вставте ще одне значення із чорним кольором. Для цього через кому, в лапках вставте значення "000000", що відповідає чорному кольору.

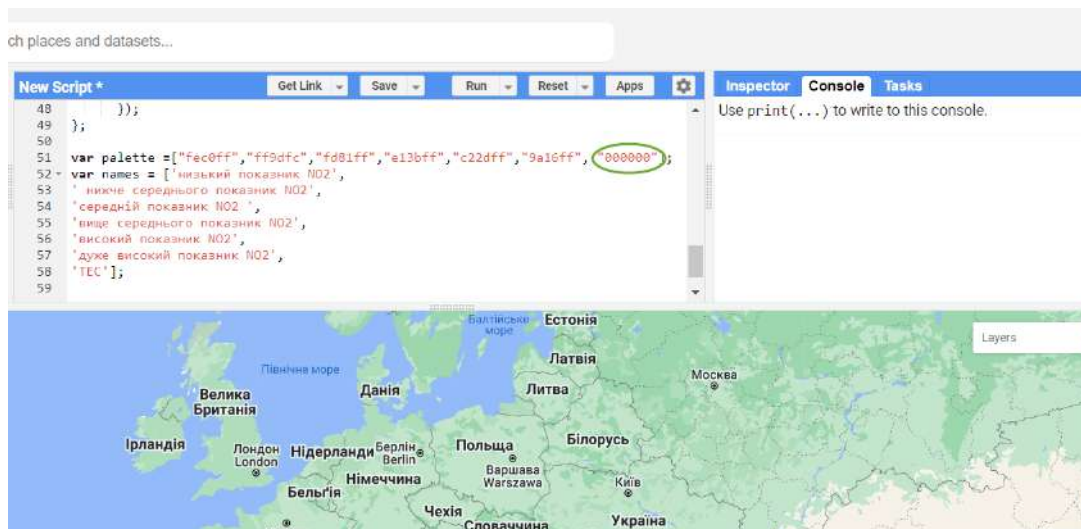


Рис. 219. Зміна кольору

Цикл “for” допомагає візуалізувати легенду в середовищі мапи.

Щоб не додавати кожен елемент окремо, можна скористатися циклом for, який послідовно додасть кожен елемент до легенди.

Опис наступних рядків такий: коли **for** число $i = 0$ та менше 7 (сім не включаючи), виконувати цикл. Тобто **legend.add** (додати) до **makeRow** palette зі значеннями [0,1,2,3,4,5,6] та **names** зі значеннями [0,1,2,3,4,5,6].

Докладніше про цей цикл: [https://uk.wikipedia.org/wiki/For_\(цикл\)](https://uk.wikipedia.org/wiki/For_(цикл)).

<pre>for (var i = 0; i < 7; i++) { legend.add(makeRow(palette[i], names[i])); }</pre>	<p>for задається трьома значеннями (var i = 0 перший елемент початок циклу; i < 7 другий елемент кінець циклу; i++ третій елемент додавання +1 до попереднього елементу)</p>
--	---

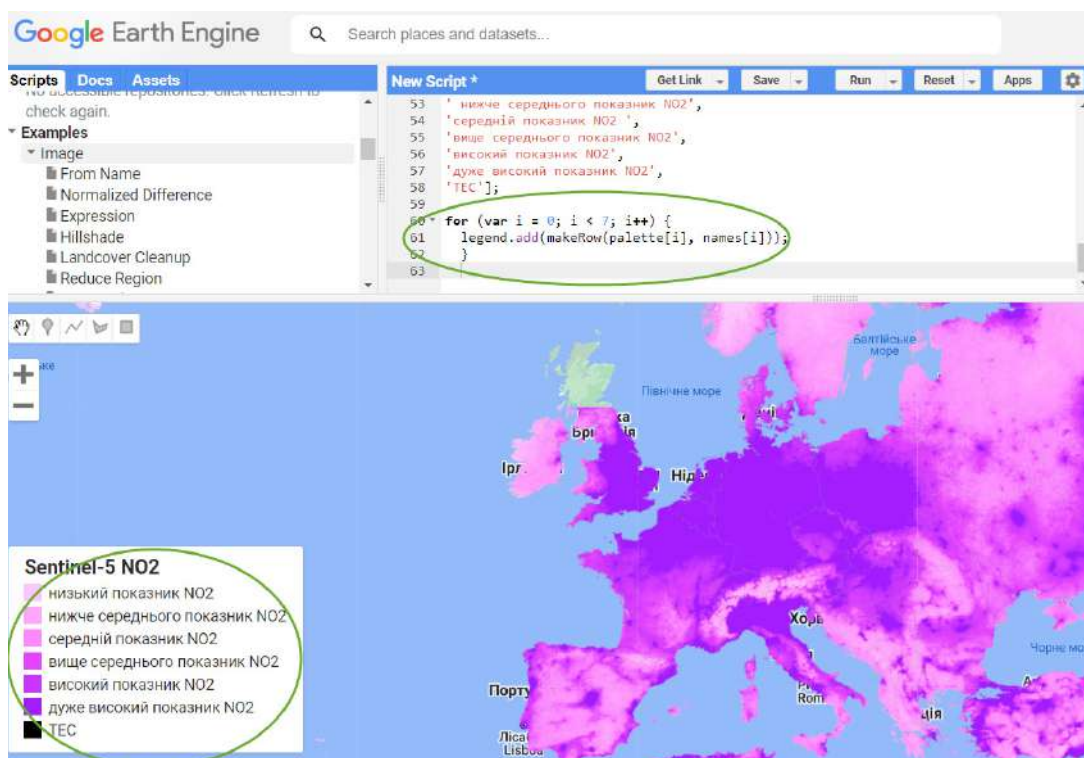


Рис. 220. Відображення легенди

✔ **Важливо!** Зображення на наведених скриншотах і у вас на екрані буде різнитися, тому що кольори для візуалізації були набрані вручну.

8. Підвантажити датасет із тепловими електростанціями (ТЕС), відфільтрувати ті, які працюють на вугіллі.

Потрібно додати один з основних чинників забруднення NO_2 – теплові вугільні електростанції. У Google Earth Engine існує датасет **Global Power Plant Database** – це повна база даних електростанцій з відкритим кодом у всьому світі. Кожна електростанція геолокована, а записи містять інформацію про потужність, виробництво, власність і тип палива.

Для візуалізації вугільних електростанцій скористайтеся таким кодом:

```
var poi = ee.FeatureCollection("WRI/GPPD/  
power_plants")  
.filter(ee.Filter.eq('fuel1','Coal'));  
Map.addLayer(poi,{},'Coal');
```

collection за допомогою **ee.FeatureCollection** звертається до бази даних Global Power Plant Database **filter** вибирає теплоелектростанції за типом палива 'fuel1','Coal' (переклад “паливо”, “вугілля”) **Map.addLayer** додає шар до мапи, в цьому випадку – точковий шар

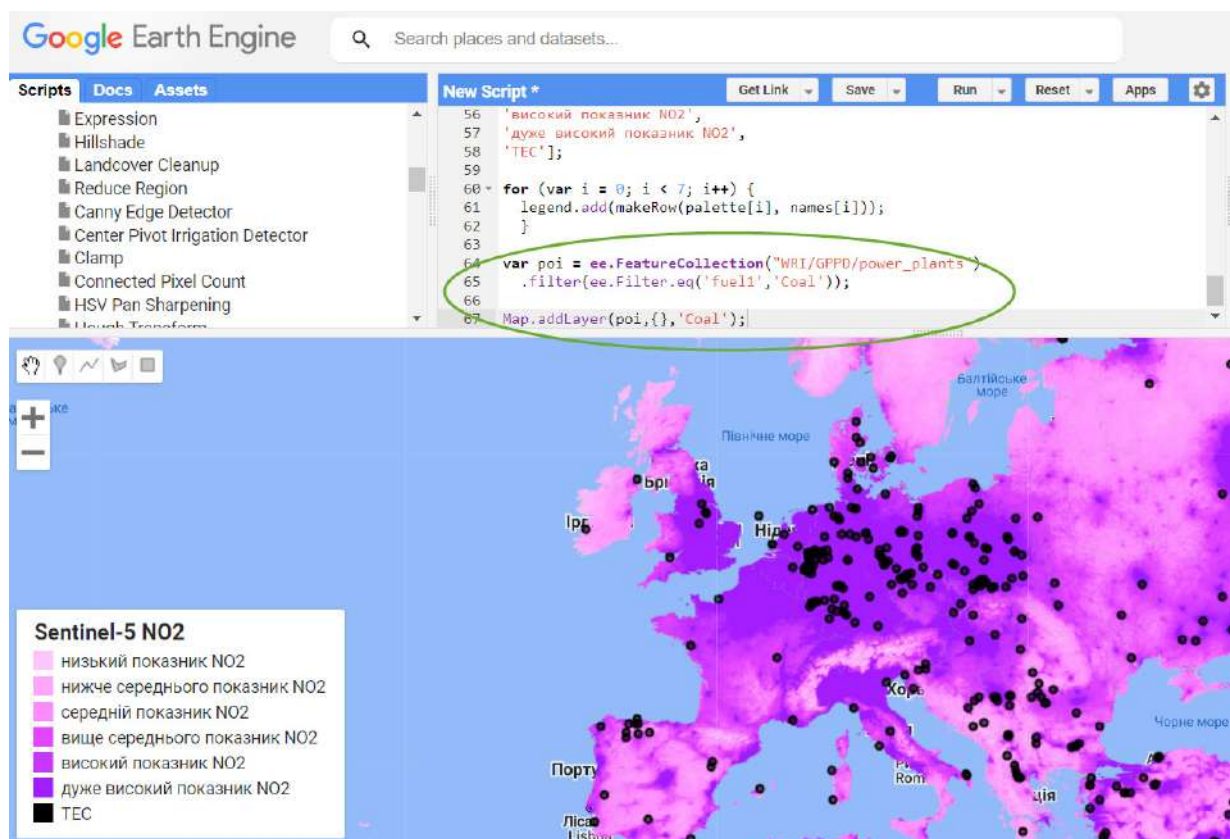


Рис. 221. Додавання шару на мапу

? Завдання для перевірки

Збережіть скриншот екрана з вашою візуалізацією і легендою для NO_2 та ТЕС

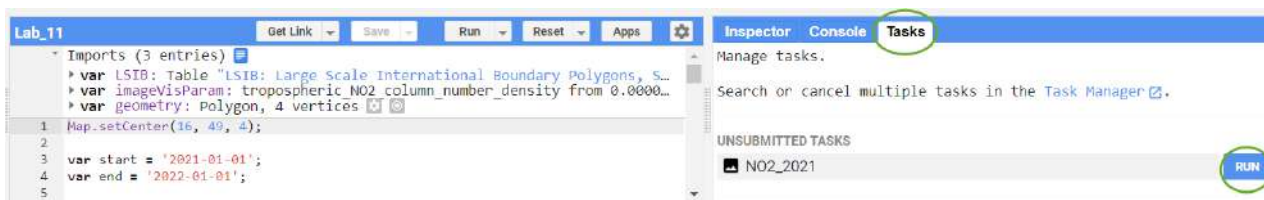
9. Зберегти усереднений растр для NO_2 за 2021 р. для заданої ділянки.

Для подальшого аналізу, особливо якщо ви працюєте в ГІС-програмах, може бути корисним збереження усередненого растра за 2021 р.

Рекомендуємо вибрати територію, виділивши її полігоном, тоді ви уникнете помилкової обрізки растра в разі строкатої адміністративної межі. Отже, окреслюємо необхідну нам територію й обрізаємо по ній наш усереднений растр. І за допомогою [Export.image.toDrive](#) зберігаємо растр на своєму Google-диску.

<pre>var clip = mean.clip(geometry) Export.image.toDrive({ image: clip, description: 'NO2_2021', scale: 300, region: geometry, });</pre>	<p><code>clip(geometry)</code> обрізає по геометрії розраховані середні значення <code>mean</code> <code>Export.image.toDrive</code> зберігає растр із заданими параметрами на Google-диску: <code>clip</code> – обрізаний знімок по геометрії <code>'NO2_2021'</code> – назва файлу <code>scale</code> – масштаб <code>geometry</code> – заданий полігон</p>
--	---

Після запуску коду підсвітиться панель **Tasks** (Менеджер завдань) і тут у рядку файлу, який ми створюємо, тиснемо на **Run** (Запуск скрипту). У вікні, яке відкриється, можемо змінити назву, масштаб, формат і теж тиснемо **Run** (Запуск скрипту). Процес збереження растра буде тривати декілька хвилин (тривалість збереження великою мірою залежить від розміру території, яку ви вибрали).



Task: Initiate image export

Task name (no spaces) *

Coordinate Reference System (CRS)

Scale (m/px)

DRIVE
 CLOUD STORAGE
 EE ASSET

Drive folder

Filename *

File format *

Рис. 222. Експорт на диск

? Додаткове завдання для перевірки

Для іншого порівняння можна додати забруднювач SO₂.

Порядок дій:

1. Скористайтесь пошуком за датасетами <https://developers.google.com/earth-engine/datasets/catalog>.
2. Введіть тер SO₂.
3. Виберіть Sentinel-5P NRTI SO₂: Near Real-Time Sulphur Dioxide.
4. Прогорніть до кінця сторінки, де ви побачите код, який потрібно скопіювати і вставити до свого редактора коду.
5. Задайте такий самий діапазон дат, як і для датасету з NO₂. Натисніть **Run** (Запуск скрипту).

Результат повинен мати приблизно такий вигляд:

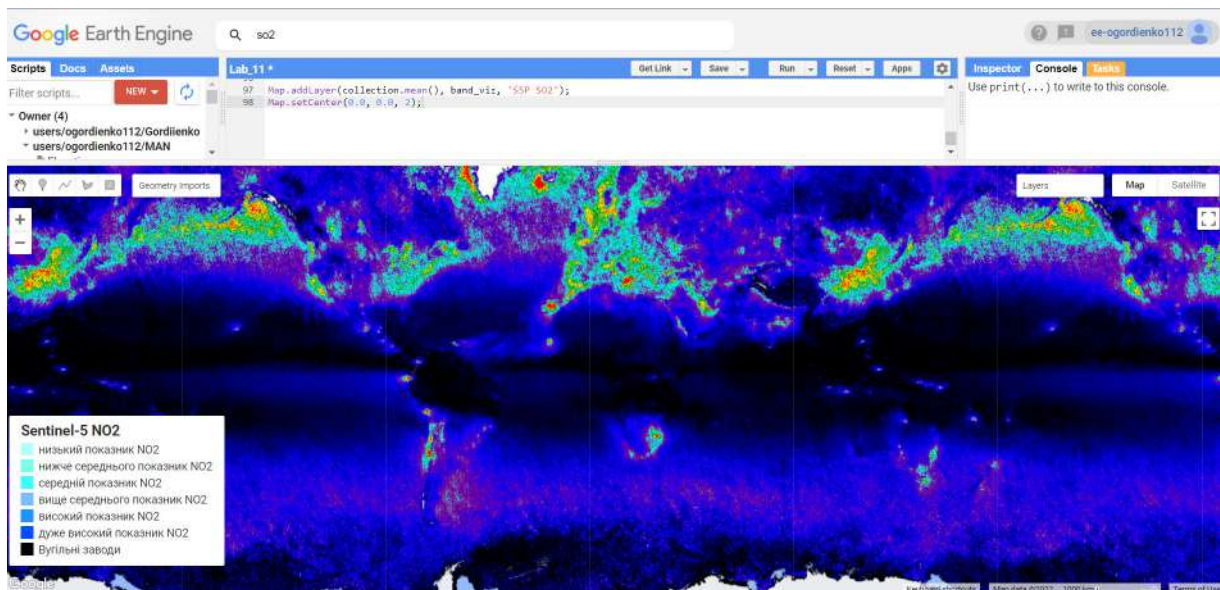


Рис. 223. Відображення даних на весь світ

За наведеним нижче QR-кодом ви можете переглянути відео виконання практичної роботи.



Рис. 224. Відео виконання практичної роботи

Готовий код можна відкрити за лінком:

https://code.earthengine.google.com/?scriptPath=users%2Fgfs_rs%2Fcourse_MAN%3ALab_11

Створення композитних анімаційних зображень із різних супутникових даних для відображення карстових процесів у районі Солотвино

Ситуація



Рис. 225. Провали в районі Солотвино

На території населеного пункту Солотвино відбувався видобуток кам'яної солі. Через низку чинників у деяких Солотвинських шахтах він призупинений. На території шахтних виробок ДП «Солотвинський солерудник» внаслідок активізації карстових явищ стався провал 3000 метрів кубічних ґрунту. Наступним став провал на території шахти № 8, який зруйнував електропідстанцію і компресорну станцію ДП «Солотвинський солерудник». У зону небезпеки потрапили житлові будинки, системи життєзабезпечення населення та об'єкти інфраструктури.¹

Новоутворене карстове провалля з гострими краями свідчить про те, що ґрунт і далі просідає.

У цій практичній роботі буде досліджено, як саме збільшувалися провали в Солотвино.

Завдання

Відфільтрувати колекції USGS Landsat 8 Collection 1 Tier 1 TOA Reflectance та Sentinel-2 MSI: MultiSpectral Instrument, Level-2A, помістити їх в одну колекцію. Додати до кожного зображення дату. Зробити з отриманих зображень анімацію розвитку техногенного карсту за період із 2013 по 2021 рр., завантажити анімацію на Google-диск.

Алгоритм виконання завдання:

1. Визначити територію інтересу, відцентрувати мапу, створити змінні для дат.
2. Створити змінні для візуалізації.
3. Відфільтрувати датасет USGS Landsat 8 Collection 1 Tier 1 TOA Reflectance.
4. Відфільтрувати датасет Sentinel-2 MSI: MultiSpectral Instrument, Level-2A.
5. Об'єднати дві колекції і відсортувати за датою.
6. Створити анімаційні зображення з підписами дат для комбінації каналів swir nir green.
7. Завантажити анімаційні зображення з підписами дат для комбінації каналів swir nir green.
8. Створити анімаційні зображення з підписами дат для комбінації каналів red green blue.

Покрокова інструкція

1. Визначити територію інтересу, відцентрувати мапу, створити змінні для дат.

Відкрити **Code Editor** (Редактор коду) і скористатися цими рядками для визначення області інтересу:

¹ URL: <https://www.ukrinform.ua/rubric-regions/2893760-poratunok-solotvina-plan-dij-vid-ukrainskih-naukovciv.html>

<pre> var geometry = /* color: #98ff00 */ /* displayProperties: [{ "type": "rectangle" }] */ ee.Geometry.Polygon([[[23.847871981697597, 47.9654058710124], [23.847871981697597, 47.94620717657372], [23.883405886726894, 47.94620717657372], [23.883405886726894, 47.9654058710124]]], null, false); </pre>	<p>Змінна <code>aoi</code> із закоментованою частиною, що описує візуалізацію.</p> <p>В <code>ee.Geometry.Polygon</code> зберігаються координати</p>
--	--

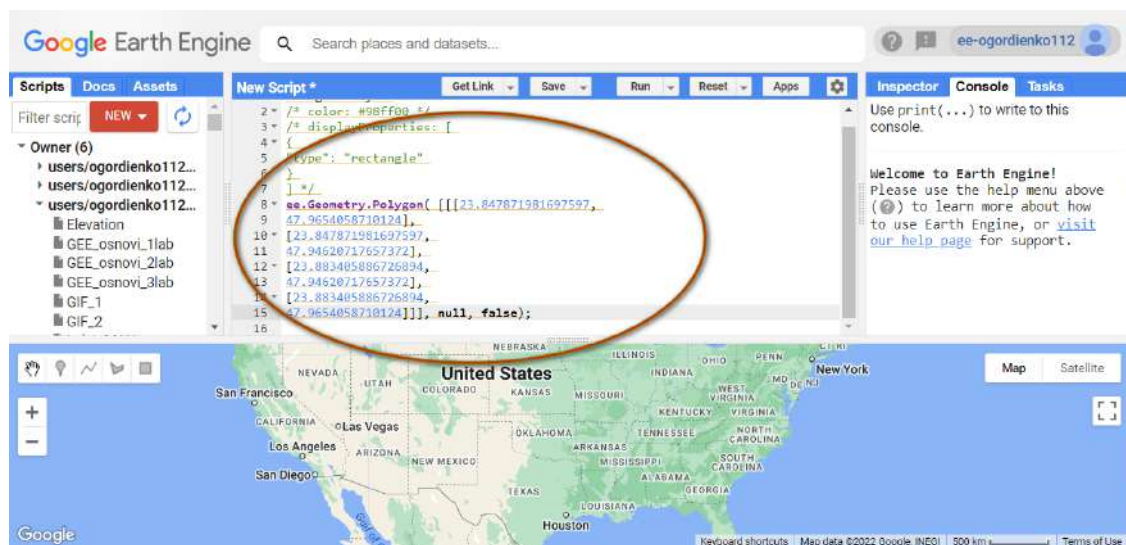


Рис. 226. Задання зони цікавості

Ці координати потрібно імпортувати в *Imports* (Імпорт). Зробити це можна, якщо навести мишку на координати і натиснути *Convert* (Конвертувати).

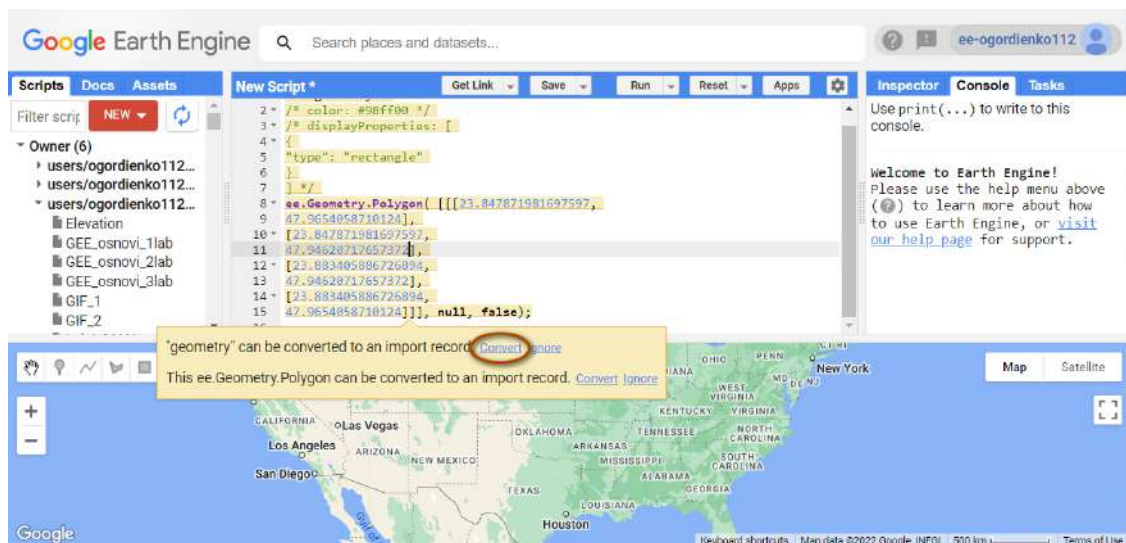


Рис. 227. Конвертація даних в імпорт

Координати і створена геометрія з'являться в *Imports* (Імпорт) і в *Geometry Imports* (Імпорт геометрії).

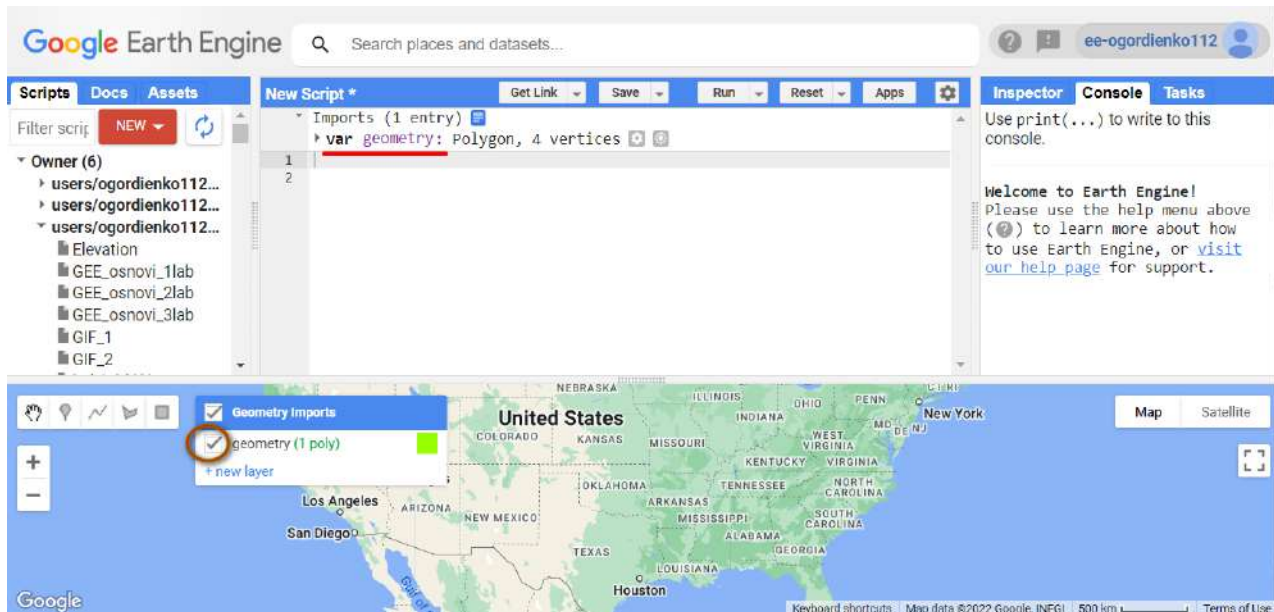


Рис. 228. Зміна відображення імпортованих даних

Для того щоб наблизитися до геометрії і задати їй іншу назву, скористайтеся рядками коду нижче й натисніть **Run** (Запуск скрипту).

<pre>var studyArea = geometry; Map. centerObject(studyArea, 12);</pre>	<p>Змінна <i>geometry</i> тепер має назву <i>studyArea</i> центрування мапи на <i>studyArea</i></p>
--	---

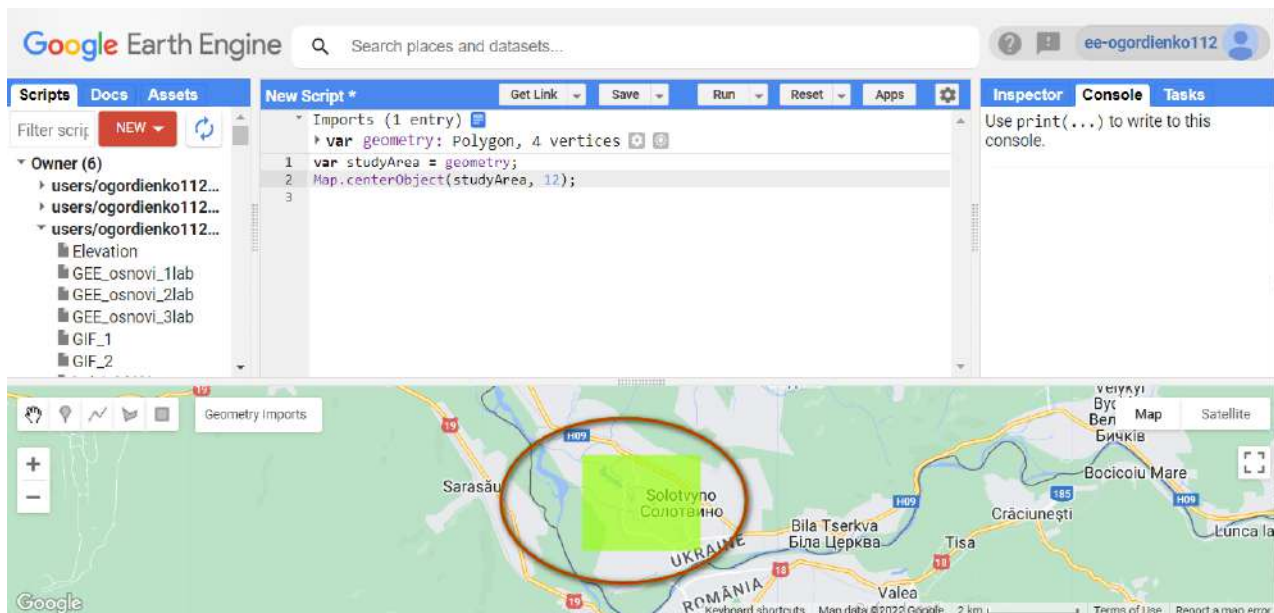


Рис. 229. Створення геометрії

Наступним кроком буде створення змінних для визначення років. У цій практичній роботі ми візьмемо дві дати: 2013 р. – початок роботи Landsat-8 на орбіті і 2022 р.

Для задання змінних скористайтеся кодом:

<pre>var startyear = 2013; var endyear = 2022;</pre>	<p>Змінна <i>startyear</i> початок дослідження Змінна <i>endyear</i> кінець дослідження</p>
--	---

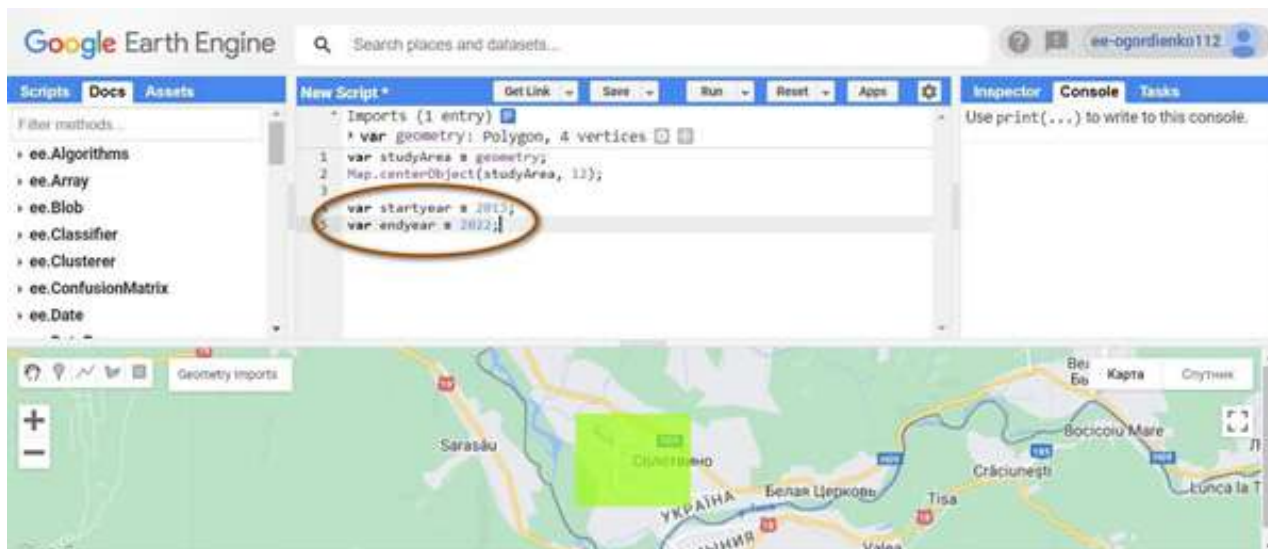


Рис. 230. Створення змінних для часового діапазону

2. Створити змінні для візуалізації.

У цій практичній роботі будуть візуалізовані дві анімації: одна в природних кольорах, інша – в штучних.

Для створення змінної, яка буде зберігати інформацію про візуалізацію в природних кольорах, скористайтеся кодом:

<pre> var visParamsRGB = { bands: ['red', 'green', 'blue'], min: 0.0, max: 0.3, }; </pre>	<p>Візуальні параметри для 'red', 'green', 'blue' bands, що будуть використані min мінімальне значення max максимальне значення</p>
---	---

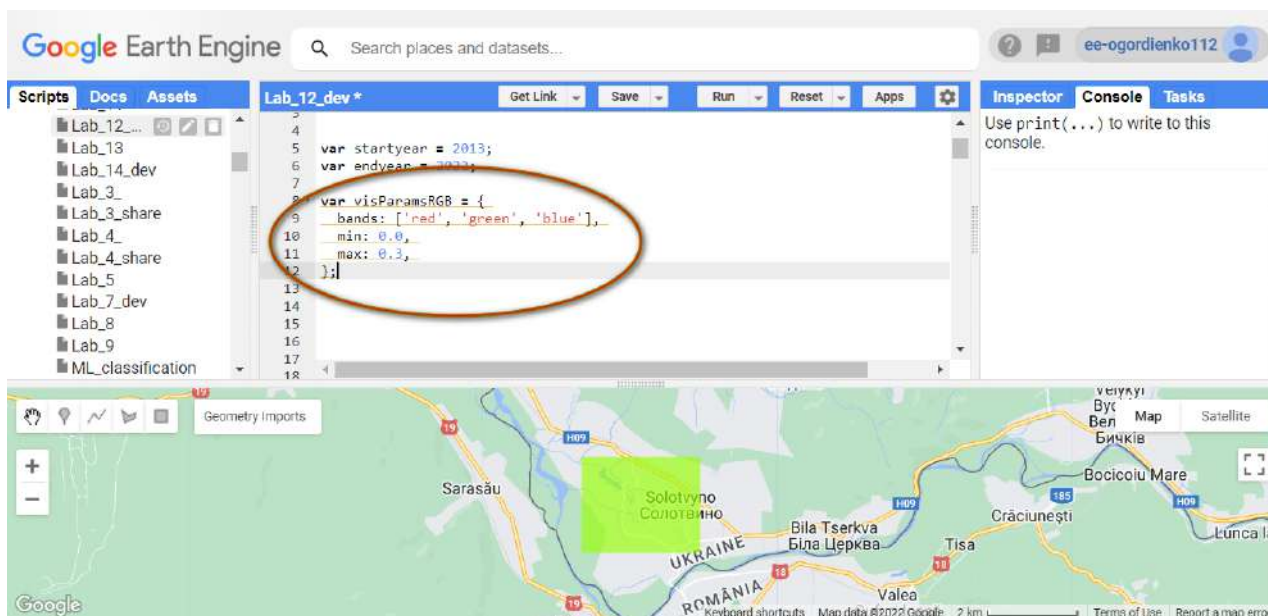


Рис. 231. Додавання візуальних параметрів RGB

Для створення змінної, яка буде зберігати інформацію про візуалізацію в штучних кольорах, скористайтеся кодом:

```
var visParamsSNG = { bands:
  ['swir', 'nir', 'green'], min: 0.0,
  max: 0.3,
};
```

Візуальні параметри для 'swir', 'nir', 'green' bands, що будуть використані
 min мінімальне значення
 max максимальне значення

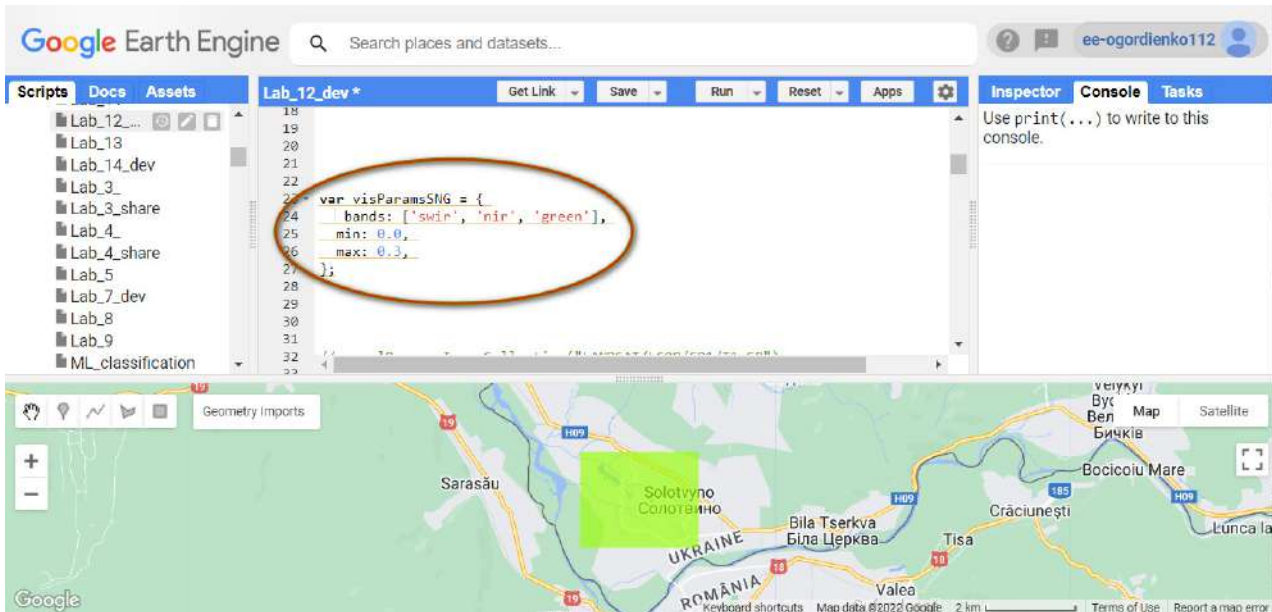


Рис. 232. Додавання візуальних параметрів RGB

3. Відфільтрувати датасет USGS Landsat 8 Collection 1 Tier 1 TOA Reflectance.

Наступним кроком буде додавання в змінну колекції знімків Landsat-8 Collection 1 Tier 1 з каліброваним коефіцієнтом відбиття верхньої частини атмосфери (TOA). Коефіцієнти калібрування витягуються з метаданих зображення. Цей датасет доступний із 2013 р. по цей час.

Для додавання скористайтеся кодом:

```
var l8 = ee.ImageCollection("LANDSAT/LC08/
C01/T1_SR")
  .filterBounds(studyArea)
  .filter(ee.Filter.calendarRange(startyear,
endyear, 'year'))
  .filter(ee.Filter.dayOfYear(182,212))
  .filter('CLOUD_COVER < 10')

  .map(function (image) {
var opticalBands = image.select('B.').
multiply(0.0001);
return image.addBands(opticalBands, null,
true)
;
})
  .map(function(image){
return image.select('B2', 'B3',
'B4', 'B5', 'B10').rename(['blue', 'green',
'red', 'nir', 'swir']);
});
```

Змінна l8
 ee.ImageCollection звертається до колекції landsat-8
 filterBounds фільтр за зоною інтересу
 calendarRange фільтр за роками
 dayOfYear фільтр за днями в році, липень
 'CLOUD_COVER показник хмарності менше 10'

Функція, що множить всі значення на коефіцієнт масштабу 0.0001 та повертає помножені значення

Функція, що перейменовує 'B2', 'B3', 'B4', 'B5', 'B10' на 'blue', 'green', 'red', 'nir', 'swir'

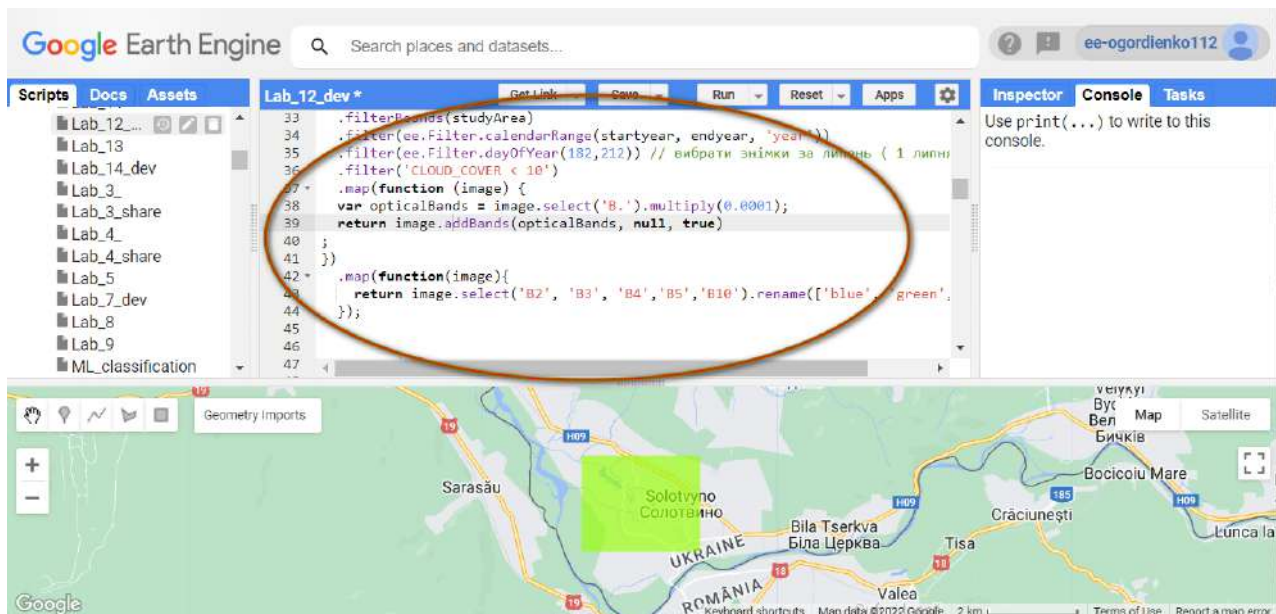


Рис. 233. Зведення зображень

Для того щоби побачити інформацію про відфільтровану колекцію і вивести перше зображення з колекції на екран, скористайтеся кодом. Натисніть **Run** (Запуск скрипту).

<pre>print (l8); Map.addLayer(l8.first(),visParamsRGB,'l8');</pre>	<p>Результат у консоль; Додавання першого зображення з колекції з параметрами візуалізації visParamsRGB та назвою 'l8';</p>
--	---

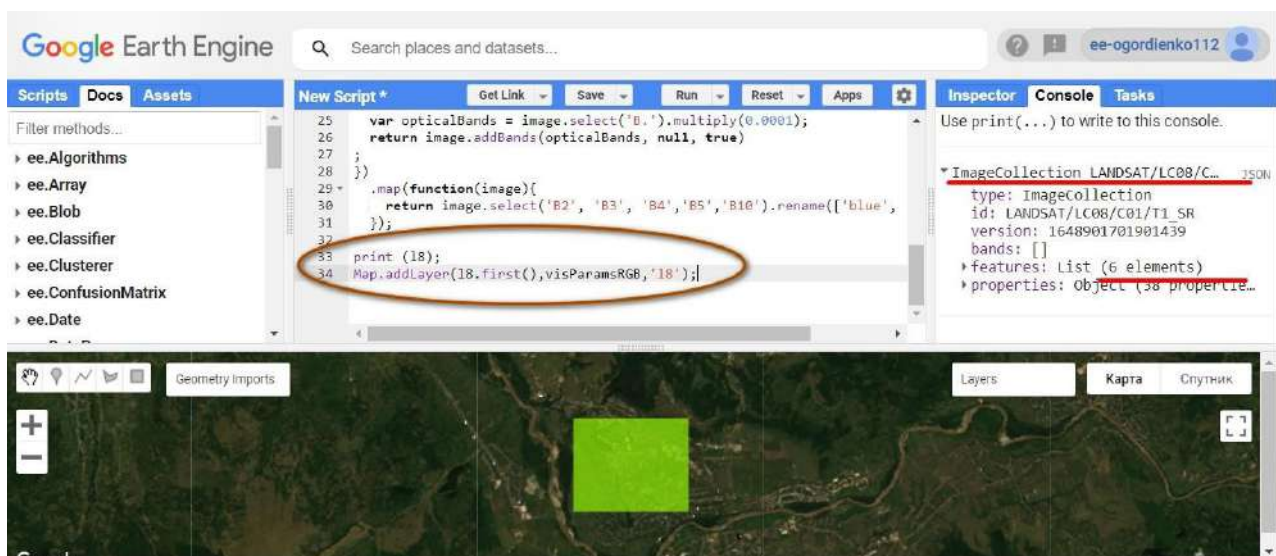


Рис. 234. Додавання шару на мапу

Якщо розгорнути в консолі ImageCollection LANDSAT/LC08/C01/T1_SR, то можна побачити дату першого зображення в колекції зображень, саме воно виводиться на мапу.

Цікаво! У програмуванні відлік починається з 0, тому перше зображення має індекс 0, друге – 1, третє – 2 і т. д.

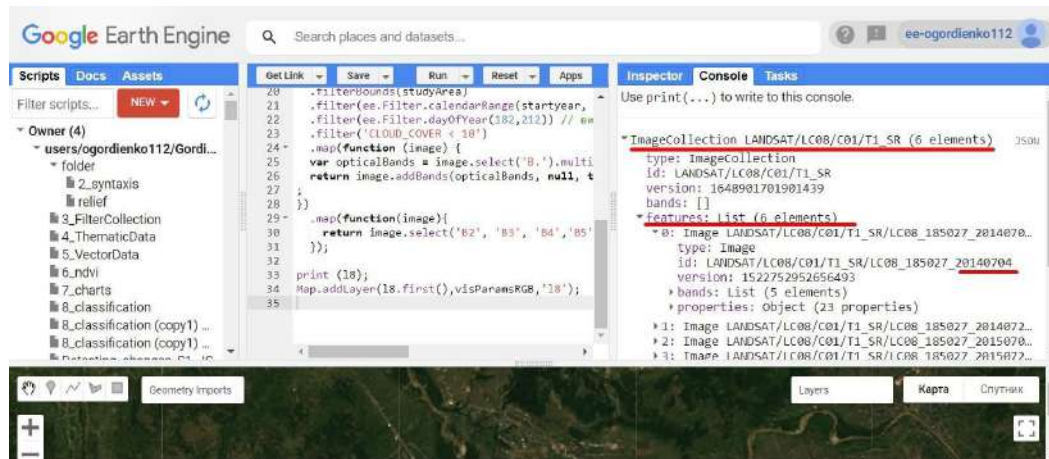


Рис. 235. Дані в консолі

Також на цьому етапі можна відключити шар геометрії, щоб він не перекривав область інтересу. Це можна зробити за допомогою *Geometry Imports* (Імпорт геометрії) і прибрати галочку навпроти назви шару.

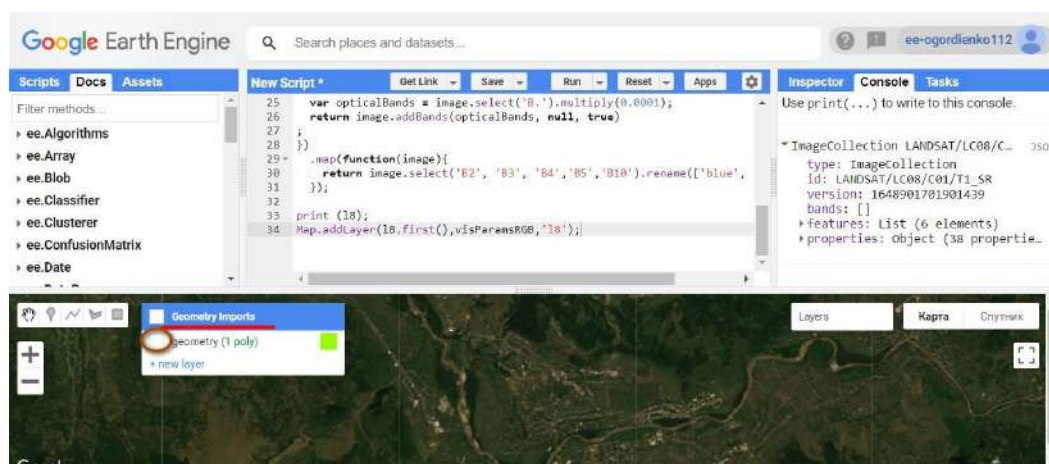


Рис. 236. Зміна відображення мапи

4. Відфільтрувати датасет Sentinel-2 MSI: MultiSpectral Instrument, Level-2A.

Наступним кроком буде створення змінної для даних із супутника Sentinel-2 – це широко-зонна місія з високим просторовим розрізненням і багатоспектральною зйомкою, що використовується для моніторингу рослинності, ґрунту та водного покриву, а також спостереження за внутрішніми водними шляхами та прибережними районами.

```

var s2 = ee.ImageCollection("COPERNICUS/S2_SR")
  .filterBounds(studyArea)
  .filter(ee.Filter.calendarRange(startyear, endyear, 'year'))
  .filter(ee.Filter.dayOfYear(182, 212))
  .filter('CLOUD_COVERAGE_ASSESSMENT < 15')
  .map(function(image) {
    var opticalBands = image.select('B').multiply(0.0001);
    return image.addBands(opticalBands, null, true)
  });
  .map(function(image){
    return image.select('B2', 'B3', 'B4','B8','B12').
    rename(['blue', 'green', 'red','nir','swir']);
  });

```

Змінна **s2**
ee.ImageCollection звертається до колекції sentinel-2
filterBounds фільтр за зоною інтересу
calendarRange фільтр за роками
dayOfYear фільтр за днями в році, липень
'CLOUD_COVERAGE' показник хмарності менше 15
 Функція, що множить всі значення на коефіцієнт масштабу 0.0001 та повертає помножені значення
 Функція, що перейменовує 'B2', 'B3', 'B4','B5','B10' на 'blue', 'green', 'red','nir','swir'

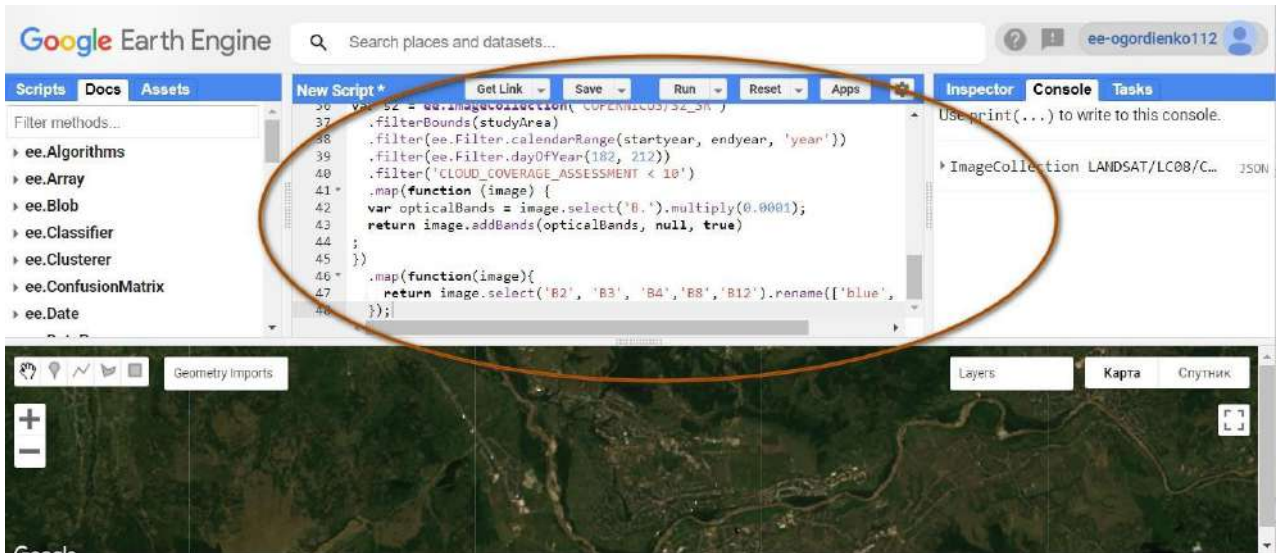


Рис. 237. Налаштування зображення

Для того щоби побачити інформацію про відфільтровану колекцію і вивести перше зображення з колекції на екран, скористайтеся кодом. Натисніть **Run** (Запуск скрипту).

<pre>print(s2); Map.addLayer(s2.first(),visParamsRGB,'s2');</pre>	<p>Результат у консоль; Додавання першого зображення з колекції з параметрами візуалізації visParamsRGB та назвою 's2';</p>
---	---

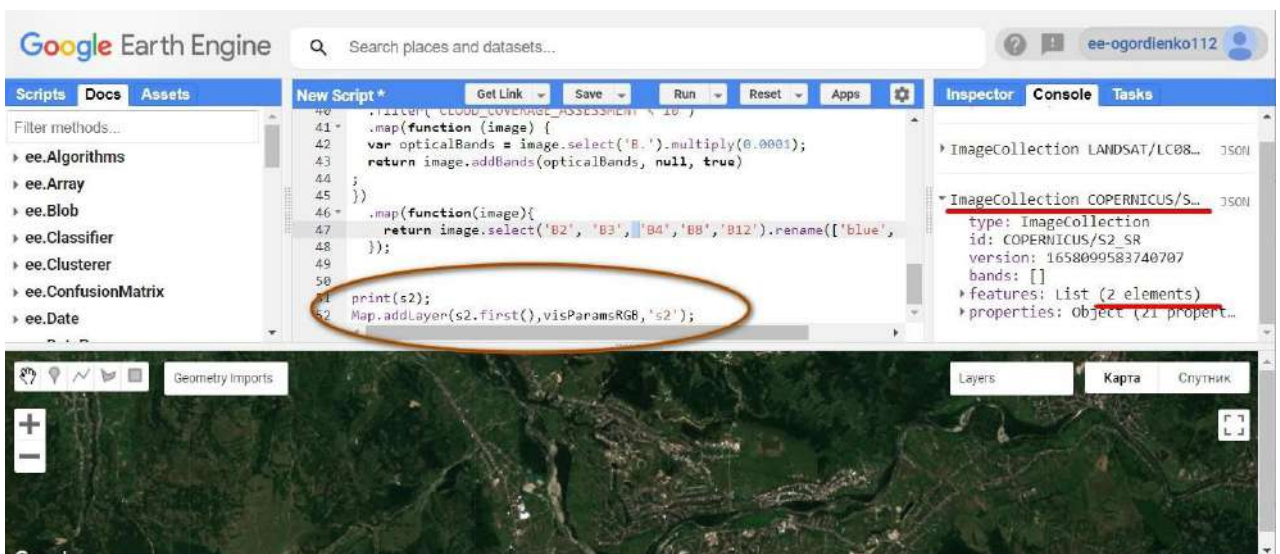


Рис. 238. Додавання шару на мапу

5. Об'єднати дві колекції і відсортувати за датою.

Наступним кроком буде створення нової змінної, в якій зберігатимуться відфільтровані дані з обох колекцій – Landsat та Sentinel.

<pre>var merged_collection = ee.ImageCollection(l8.merge(s2));</pre>	<p>Змінна, що об'єднує відфільтровані дані в одну колекцію, яка має назву merged_collection</p>
--	---

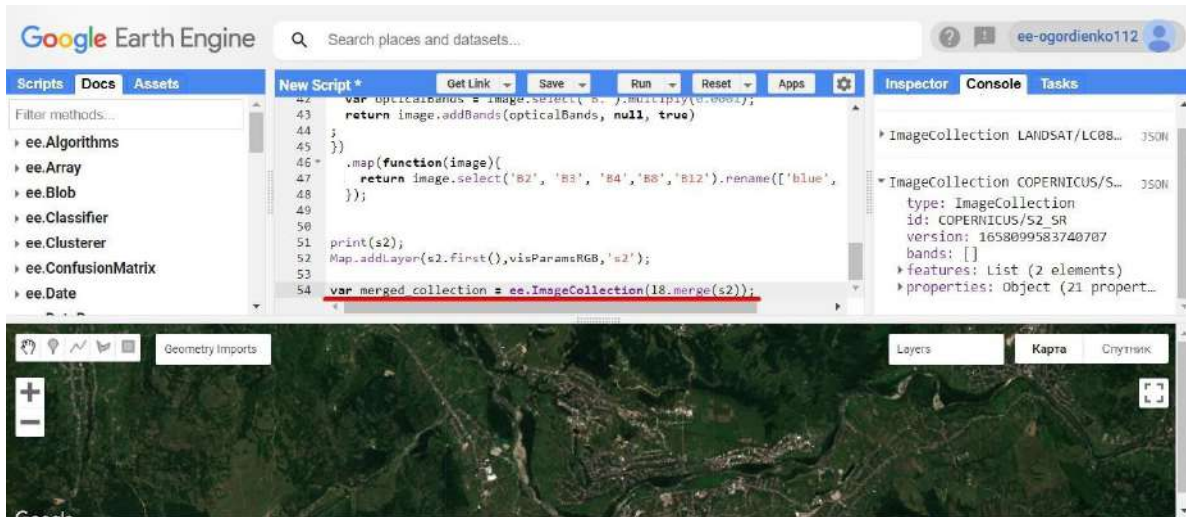


Рис. 239. Об'єднання двох колекцій

6. Створити анімаційні зображення з підписами дат для комбінації каналів swir nir green.

Наступний крок: потрібно відфільтрувати отримані дані і розмістити їх за датою, використовуючи `system:time_start`.

Позначка часу *Earth Engine* у мілісекундах з епохи UNIX. Перегляньте це покликання для отримання додаткової інформації: https://en.wikipedia.org/wiki/Unix_time.

Мітка часу встановлюється на номінальний час отримання зображення для окремих сцен. Для тимчасових композитів встановлено номінальний період початку композиту.

Результат відфільтрованої колекції вивести в консоль. Зробити це можна за допомогою коду:

<pre>var merged_collection_sorted = merged_collection.sort("system:time_start"); print(merged_collection_sorted);</pre>	<p>Сортування даних за датою, першою буде початкова дата;</p> <p>Результат у консоль;</p>
---	---

Натисніть **Run** (Запуск скрипту).

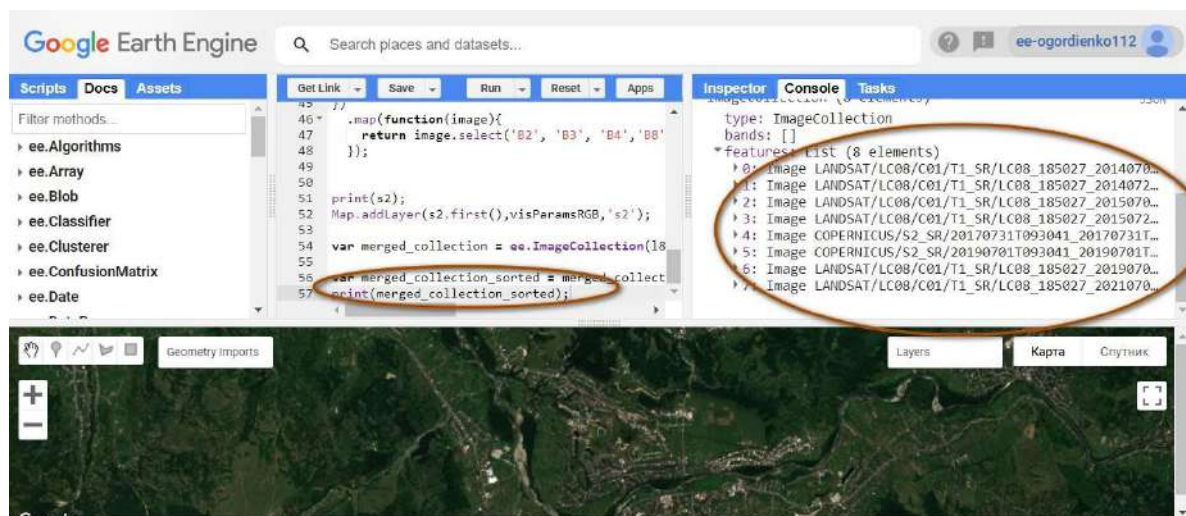


Рис. 240. Виведення у консоль

У консолі можна побачити, що дані тепер розташовані за датою.

Наступним кроком буде створення змінної, яка звертатиметься до сторонньої користувачької бази даних із текстом і за допомогою `system:time_start` додаватиме до анімації дату, задаватиме її місцезположення, розмір.

Щоб створити анімацію з візуалізацією 'swir', 'nir', 'green', в якій вода й урболандшафти будуть відображені червоним кольором, а все інше жовтим, скористайтеся кодом:

<pre> var text = require('users/gena/packages:text') ; var annotations = [{position: 'left', offset: '1%', margin: '1%', property: 'label', scale: 30}]; var animation_label = merged_collection_ sorted.select('swir', 'nir', 'green'). map(function(img) { var timeStamp = ee.Date(img.get('system:time_ start')).format().slice(0,10); timeStamp = ee.String('Date: ').cat(ee.String(timeStamp)); var image = img.visualize(visParamsSNG).set({'label': timeStamp}); var annotated = text.annotateImage(image, {}, studyArea, annotations); return annotated; }); </pre>	<p>Змінна <code>text</code> звертається до користувачької бібліотеки <code>users/gena/packages:text</code> <code>annotations</code> задає параметри розміщення та розмір підписів для дат</p> <p><code>animation_label</code> з відсортованої колекції знімків бере канали 'swir', 'nir', 'green'</p> <p><code>timeStamp</code> бере з 'system:time_start' 10 символів</p> <p><code>timeStamp</code> перетворює формат дати на список</p> <p><code>image</code> за допомогою <code>visualize</code> задає параметри візуалізації і бере підпис <code>timeStamp</code> <code>annotated</code> створює колекцію з текстом за геометрією <code>studyArea</code> та повертає колекцію з підписами</p>
---	---

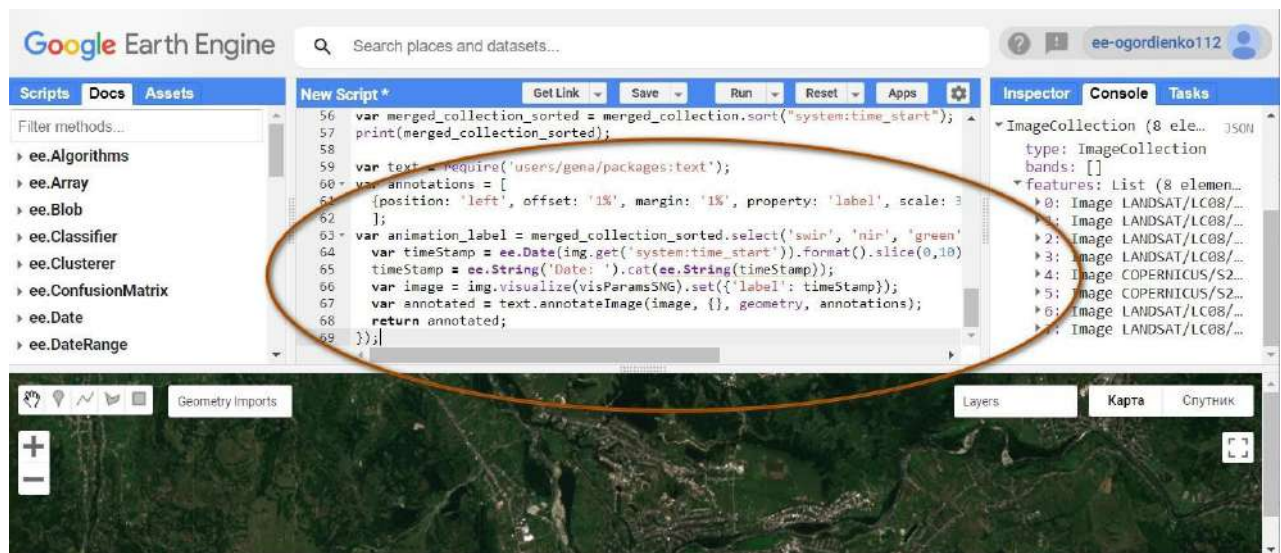


Рис. 241. Створення анімації

Для параметрів потрібно задати аргументи, які будуть використовуватися в композитній анімації. Зробити це можна за допомогою коду:

<pre> var videoArgs = { dimensions: 1000, region: studyArea, framesPerSecond: 1 }; </pre>	<p><code>videoArgs</code> параметри відео <code>dimensions</code> розширення відео <code>region</code> зона інтересу <code>framesPerSecond</code> кадрів на секунду</p>
---	--

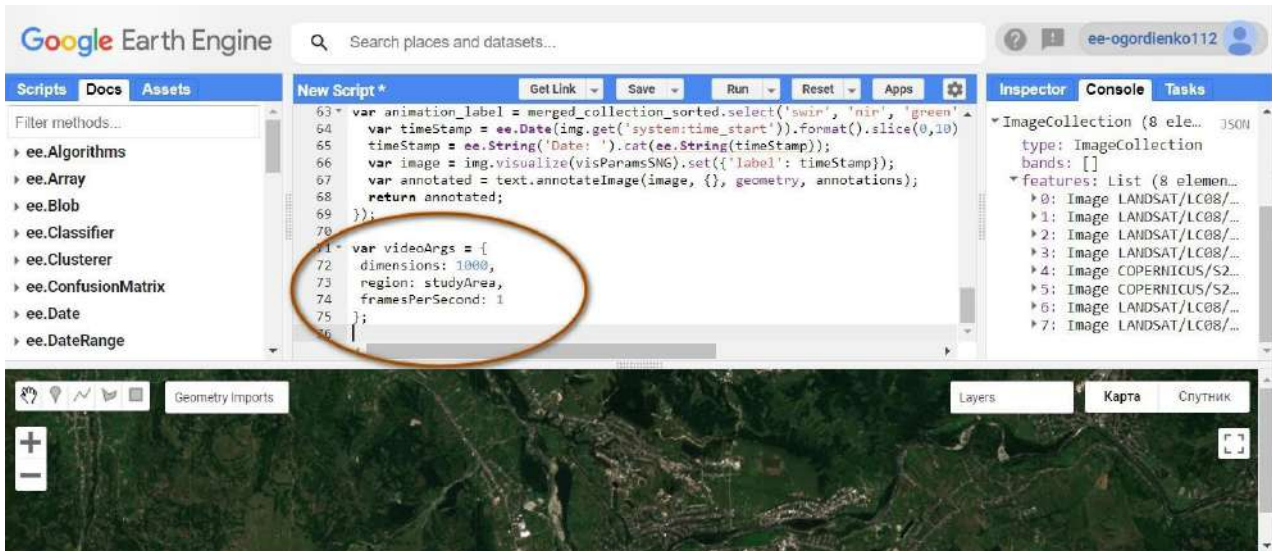


Рис. 242. Надання налаштувань анімації

Для того щоб вивести анімацію в консоль, скористайтеся кодом нижче. Натисніть **Run** (Запуск скрипту).

<pre>print(ui.Thumbnail(animation_label, videoArgs));</pre>	<p>Результат у консоль;</p>
---	-----------------------------

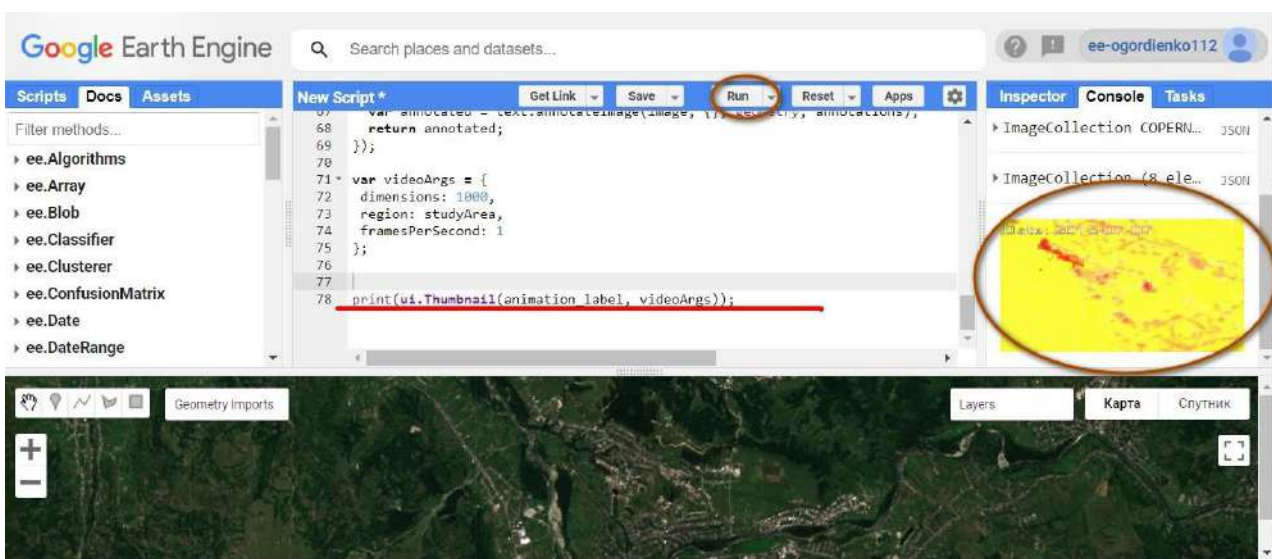


Рис. 243. Виведення анімації в консоль

7. Завантажити анімаційні зображення з підписами дат для комбінації каналів swir nir green. Для завантаження анімації на Google-диск скористайтеся кодом:

<pre>Export.video.toDrive({ collection: animation_label, region: studyArea, description: 'Solotvino_2013-2022', dimensions: 1000, framesPerSecond: 1, });</pre>	<p>Export.video.toDrive експортує відео на Google-диск collection колекція з підписами region зона інтересу description назва експортованого файлу dimensions розширення зображень framesPerSecond кадрів на секунду</p>
---	---

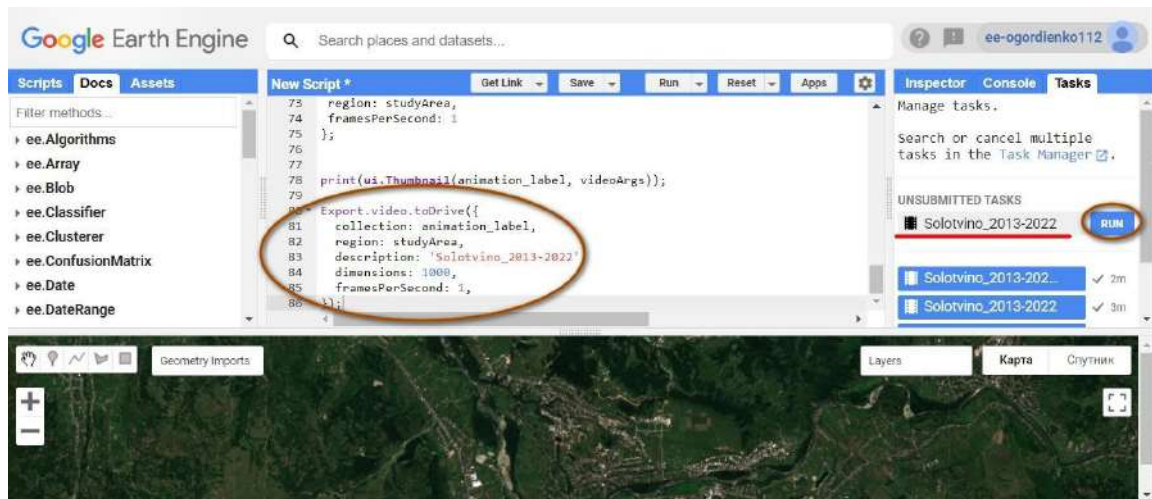


Рис. 244. Завантаження анімації

8. Створити анімаційні зображення з підписами дат для комбінації каналів red green blue.

Для створення анімації за тими самими даними, але з візуалізацією в природних кольорах скористайтеся кодом:

<pre> var animation_label_ui = merged_ collection_sorted.select('red', 'green', 'blue').map(function(img) { var timeStamp = ee.Date(img. get('system:time_start')).format ().slice(0,10); timeStamp = ee.String('Date: ').cat(ee.String(timeStamp)); var image = img.visualize(visParamsRGB).set({'label': timeStamp}); var annotated = text.annotateImage(image, {}, studyArea, annotations); return annotated; }); </pre>	<p>Змінна <code>animation_label_ui</code> бере колекцію <code>merged_collection_sorted</code>, обирає <code>red</code>, <code>green</code>, <code>blue</code> та застосовує <code>timeStamp</code> для додавання інформації про дату, яку бере з <code>'system:time_start'</code> з довжиною символів 10</p> <p><code>timeStamp</code> перетворює формат дати на список</p> <p><code>image</code> за допомогою <code>visualize</code> задає параметри візуалізації та бере підпис <code>timeStamp</code></p> <p><code>annotated</code> створює колекцію з текстом за геометрією <code>studyArea</code> та повертає колекцію з підписами</p>
---	---

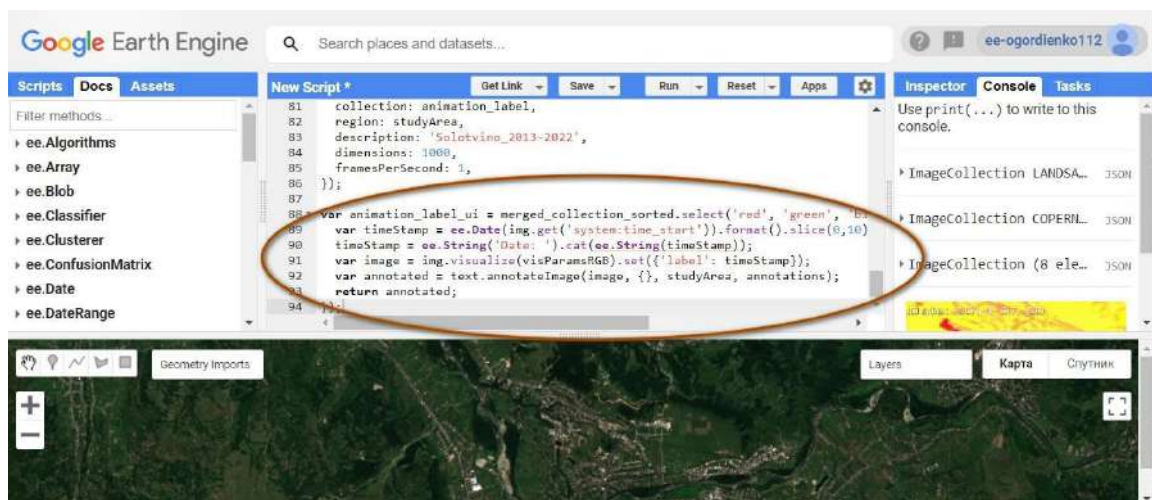


Рис. 245. Надавання налаштувань анімації

Ця анімація буде виводитися в користувацький інтерфейс для зручності порівняння двох візуалізацій.

Користувачський інтерфейс – це елементи мапи, що містяться всередині робочого простору програми. До користувачького інтерфейсу можна виводити зображення, графіки, анімації тощо.

Для виведення анімації в природних кольорах до користувачького інтерфейсу скористайтесь кодом нижче. Натисніть **Run** (Запуск скрипту).

<pre>var Thumbnail = ui.Thumbnail({ image: animation_label_ui, params: videoArgs, style: { position: 'bottom-right', width: '320px' } }); Map.add(Thumbnail);</pre>	<p>Змінна <code>Thumbnail</code> за допомогою <code>ui.Thumbnail</code> створює елемент користувачького інтерфейсу <code>image</code> колекція зображень <code>params</code> параметри відео <code>style</code> параметри розміщення вікна користувачького інтерфейсу</p> <p><code>Map.add</code> додає вікно користувачького інтерфейсу на мапу</p>
---	--

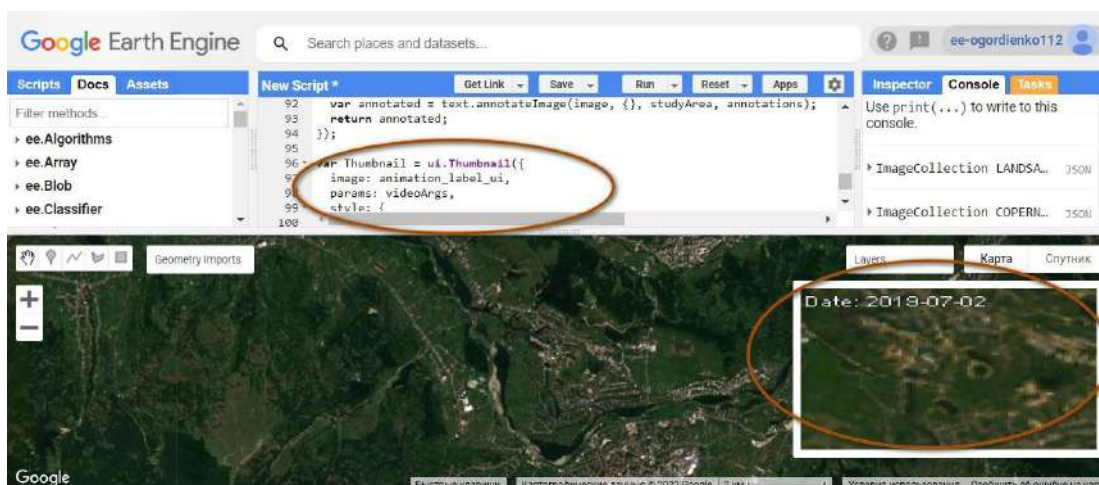


Рис. 246. Виведення анімації в інтерфейс користувача

? Завдання для перевірки

Скористайтесь кодом і заповніть пробіли, щоб завантажити анімацію в природних кольорах на Google-диск:

```
Export.video.toDrive({
  collection:_, region:_____,
  description: 'Solotvino_2013-2022_rgb',
  dimensions:_, framesPerSecond:____,
});
```

За наведеним нижче QR-кодом ви можете переглянути відео виконання практичної роботи.



Рис. 247. Відео виконання практичної роботи

Готовий код можна відкрити за лінком:

https://code.earthengine.google.com/?scriptPath=users%2Fgis_rs%2Fcourse_MAN%3ALab_12

Створення анімаційних зображень часових рядів вегетаційного індексу на прикладі аналізу динаміки лісового покриву

Ситуація

Масштабна й хаотична експлуатація природних ресурсів призводить до загострення екологічної ситуації і не приносить економічного зростання, до того ж насамперед місцевим громадам. А за регіоном Українських Карпат поступово закріплюється слава лісосировинного придатка. Сотні тисяч кубічних метрів деревини, які заготовляють тут щорічно із застосуванням екологічно небезпечних технологій, вивозять за кордон кругляком або у вигляді пиломатеріалів. Такі методи лісокористування не приносять соціально-економічних вигод, а навпаки, загострюють паводкову ситуацію¹, активізують ерозійні процеси, руйнують ландшафти та дорожню інфраструктуру, створюють негативний імідж для розвитку туризму тощо.

Дуже показовим у цьому контексті є стан лісокористування в надзвичайно паводконебезпечному верхів'ї басейну Тиси, на території Рахівського району Закарпатської області. Саме тут, попри прийняття закону України про мораторій на проведення суцільних рубок у лісах на гірських схилах, спостерігається чітка тенденція до збільшення обсягів лісозаготівель². До того ж основна маса деревини заготовлюється шляхом суцільних рубок, різко знижуються показники посадок лісу. Супутникові дані космічного моніторингу Землі можуть бути використані не тільки для дослідження динаміки змін на великих територіях, а й для відстежування динаміки змін на конкретних ділянках із допомогою часових рядів спектральних індексів.

Завдання

Для дослідження динаміки лісового покриву на території Рахівського району Закарпатської області потрібно розрахувати мапи індексу NDVI і побудувати графік змін цього індексу на території вибраної ділянки станом із 2014 по 2020 рр. Для того щоб уникнути впливу хмарного покриву на значення вегетаційного індексу, необхідно провести маскування хмар із космічних зображень. Для більш наочного представлення результатів треба створити анімаційне зображення з розрахованих мап індексу NDVI на вибрану ділянку.

Алгоритм виконання завдання:

1. Підвантажити архів даних Landsat і зробити фільтрацію за датами.
2. Зробити маскування хмар на вибраних знімках архіву Landsat.
3. Розрахувати і візуалізувати індекс NDVI на вибрану ділянку.
4. Побудувати графік змін індексу NDVI на конкретну точку.
5. Створити анімацію змін індексу NDVI на вибрану територію.

Покрокова інструкція

1. Підвантажити архів даних Landsat і зробити фільтрацію за датами.

Відкрийте **Code Editor** (Редактор коду) і скористайтеся цим рядком, щоб наблизитися до зони інтересу:

```
Map.setCenter(24.42,48.44, 12);
```

Відцентрувати мапу за координатами

Мапа наблизиться до зони Карпат.

¹ URL: <https://www.slovoidilo.ua/2020/06/24/novyna/suspilstvo/derzhlisahentstvi-nazvaly-prychyny-pavodkiv-karpatax>

² URL: <https://suspilne.media/65518-u-peredgiri-karpat-zrubano-50-lisu/>

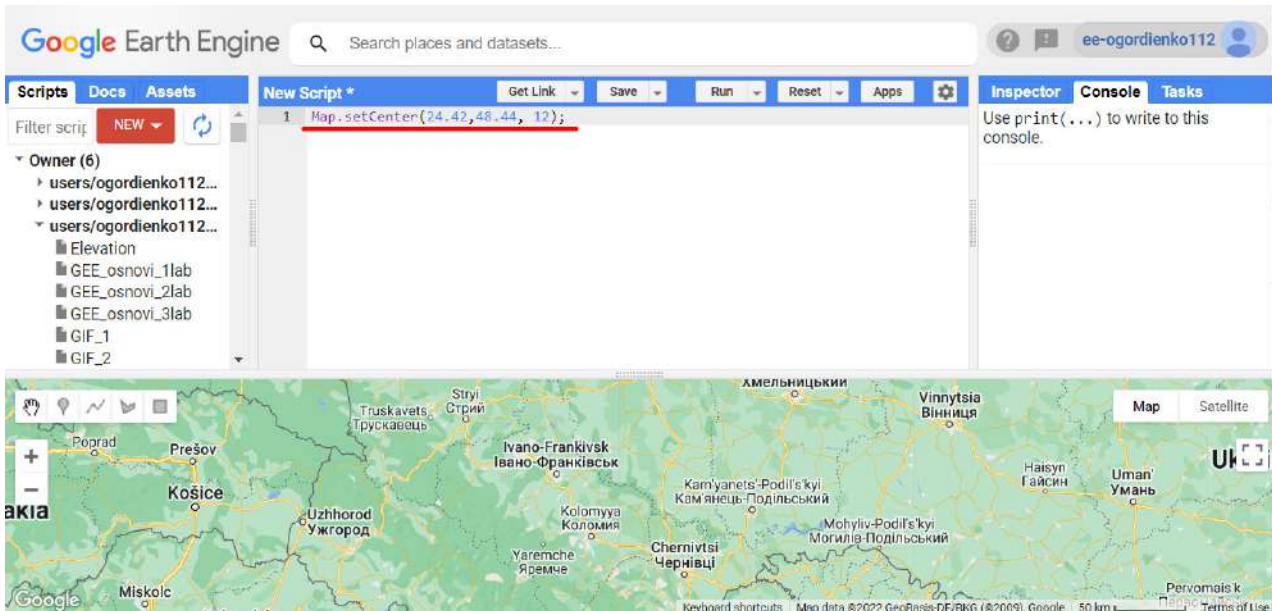


Рис. 248. Наближення до зони цікавості

AOI – зона інтересу, з англійської – *Area Of Interest*.

POI – точка інтересу, з англійської – *Point Of Interest*.

У Google Earth Engine є можливість ділитися геометрією користувача. Скопіюйте область інтересу і точку інтересу, яку ми створили для вас.

<pre> var aoi = /* color: #d63000 */ /* shown: false */ /* displayProperties: [{ "type": "rectangle" }] */ ee.Geometry.Polygon([[[24.384706096165687, 48.46485976000598], [24.384706096165687, 48.42625838203151], [24.45628889035514, 48.42625838203151], [24.45628889035514, 48.46485976000598]]], null, false), poi = /* color: #d63000 */ee.Geometry.Point([24.392841370399633, 48.439243238881105]); </pre>	<p>Змінна aoi із закоментованою частиною, що описує візуалізацію.</p> <p>В ee.Geometry.Polygon зберігаються координати</p> <p>poi змінна, що описує точку ee.Geometry.Point дві координати точки</p>
---	--

Якщо після скопійованих вище параметрів натиснути на будь-яке місце, то з'явиться нове меню, яке пропонує скопіювати ці параметри в **Imports** (Імпорт геометрії).

Натисніть **Convert** (Конвертувати).

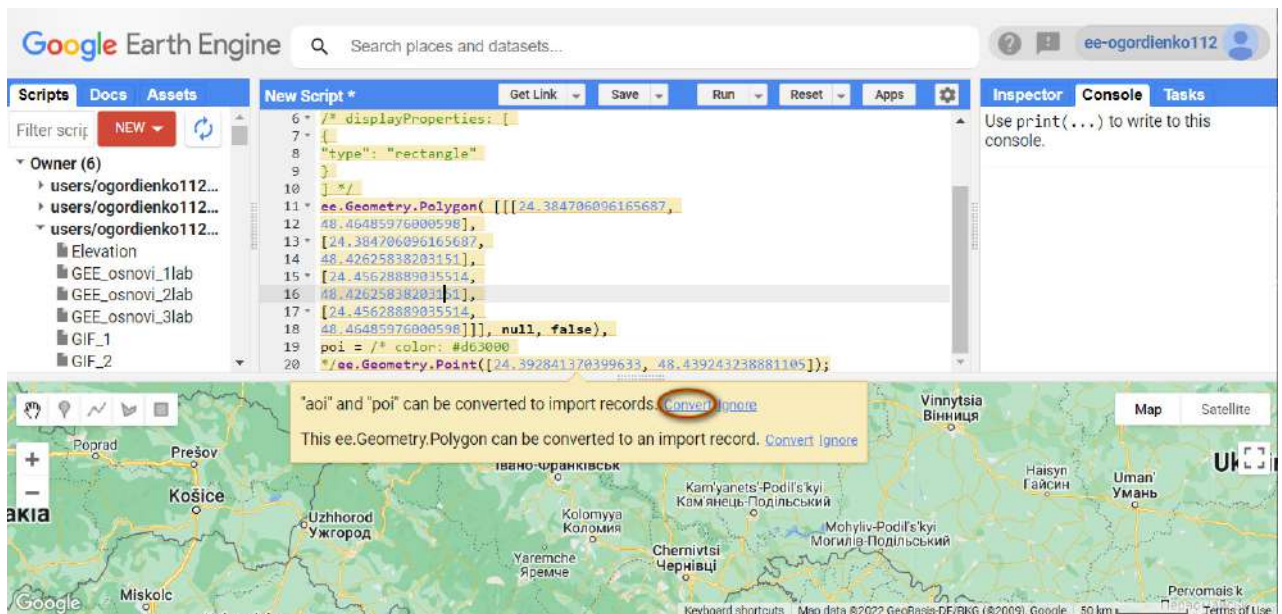


Рис. 249. Імпорт даних

Шар з'явиться в **Imports** (Імпорт геометрії).

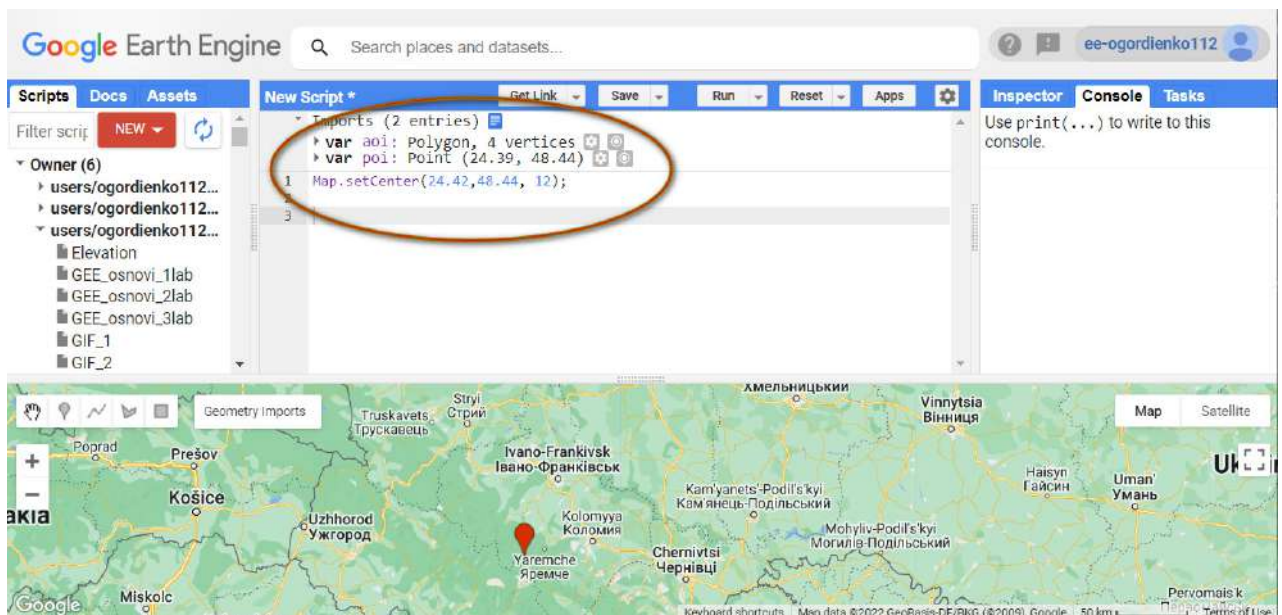


Рис. 250. Центрування мапи

Натисніть **Run** (Запуск скрипту).

Наступним кроком буде підвантаження датасету LANDSAT і фільтрація його за областю інтересу, яка була щойно створена. Для цього скористайтеся рядком коду:

```

var imageCollection =
ee.ImageCollection('LANDSAT/LC08/C01/T1
_SR')
.filterBounds(aoi);

```

imageCollection звертається до 'LANDSAT/LC08/C01/T1_SR' і бере дані, що перетинаються з **aoi**

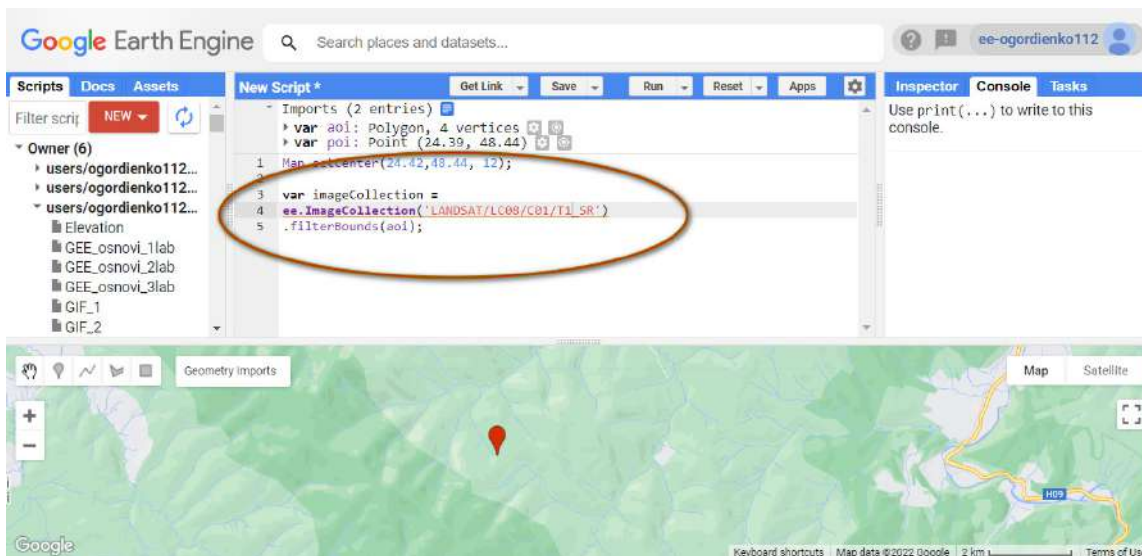


Рис. 251. Додавання колекції

2. Зробити маскування хмар на вибраних знімках архіву Landsat.

У цій практичній роботі буде розрахований індекс NDVI. Для його коректного відображення з колекції потрібно скористатися функцією, що маскує хмари.

Функція для маскування хмар на Landsat 8 має таку логіку: в кожному зображенні міститься інформація про наявність у пікселі хмар, тіні від хмар.

Бітність числа в обчислювальній техніці – це нумерація бітів. Це правило, яке використовується для визначення позицій бітів у двійковому числі. Це властивість самого числового значення у двійковому форматі. Часто використовується в програмуванні через бітовий зсув: значення $1 \ll n$ відповідає n -му біту двійкового цілого числа.

Приклад на наступному зображенні.

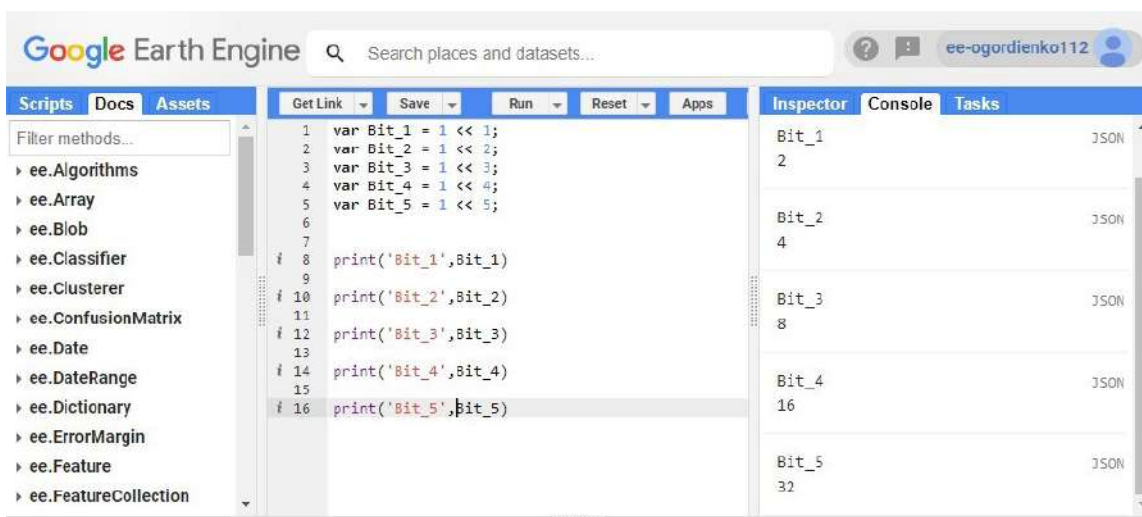


Рис. 252. Побітовий зсув

Більше інформації про біти: https://en.wikipedia.org/wiki/Bit_numbering.

Нижче наведено таблицю, що показує, які значення зберігаються в цьому пікселі.

Біт	Присвоєнні значення
Bit 0	<i>Fill (Заповнений)</i>
Bit 1	<i>Clear (Чистий)</i>

Bit 2	<i>Water (Вода)</i>
Bit 3	<i>Cloud Shadow (Тінь хмари)</i>
Bit 4	<i>Snow (Сніг)</i>
Bit 5	<i>Cloud (Хмара)</i>
Bits 6–7	<i>Cloud Confidence (Впевненість у хмарі)</i>
Bits 8–9	<i>Cirrus Confidence (Впевненість у перистій хмарі)</i>
Bit 10	<i>Terrain Occlusion (Тінь від місцевості)</i>

Маскування хмар проводилося за рекомендацією розробника. Більше інформації за покликанням: https://developers.google.com/earth-engine/datasets/catalog/LANDSAT_LC08_C01_T2_SR

Щоби створити функцію маскування хмар, скористайтеся рядками коду:

<pre>function maskL8sr(imageCollection) { var cloudShadowBitMask = 1 << 3; var cloudsBitMask = 1 << 5; var qa = imageCollection.select('pixel_qa'); var mask = qa.bitwiseAnd(cloudShadowBitMask).eq(0) .and(qa.bitwiseAnd(cloudsBitMask).eq(0)); return imageCollection.updateMask(mask).divide(10000) .select("B[0-9]*") .copyProperties(imageCollection, ["system:time_start"]); }</pre>	<p>Функція для маскування хмар на Landsat 8 << означає зсув у 16-бітному кодуванні чисел, які відсилають до попередньої таблиці (тут замасковані Bit 5 та Bit 3) з imageCollection береться канал 'pixel_qa'</p> <p>mask бере значення, що дорівнюють 0, з каналу 'pixel_qa'</p> <p>Повертає маску, за допомогою updateMask застосовує маску і множить на коефіцієнт масштабу 10000 вибирає всі значення каналів від 0 до 9 Вставляє значення time_start для коректного відображення</p>
---	--

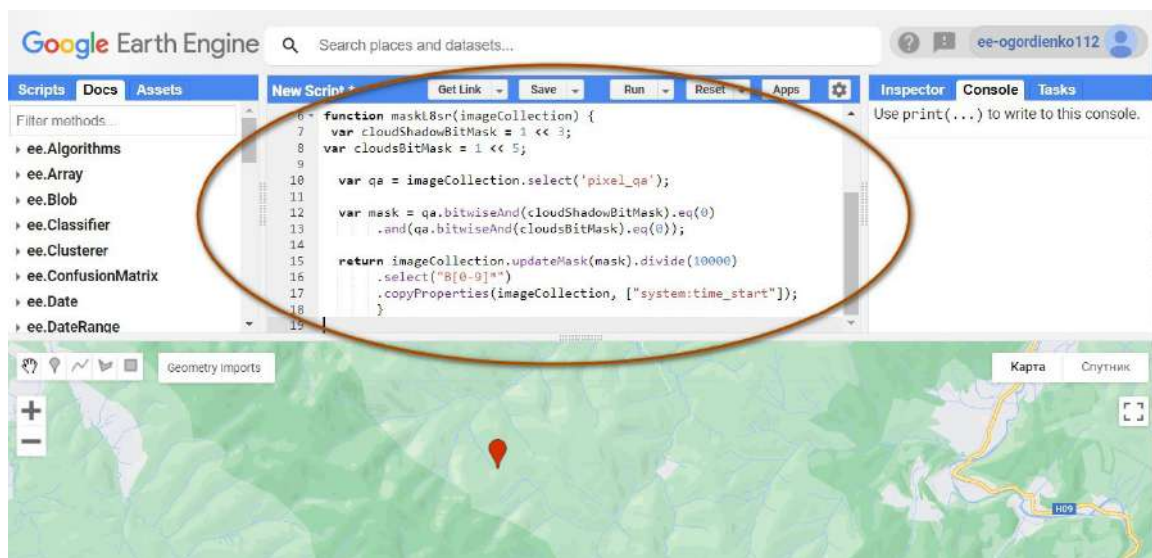


Рис. 253. Маскування хмар

Для того щоб надалі відображати середні значення за роками (в нашому випадку – з 2014 р. по 2020 р.), є можливість створити список дат за роками. Для цього потрібно створити команду:

```
var stepList = ee.List.sequence(2014,2020);
```

Створення списку зі значеннями 2014, 2015, 2016, 2017, 2018, 2019, 2020

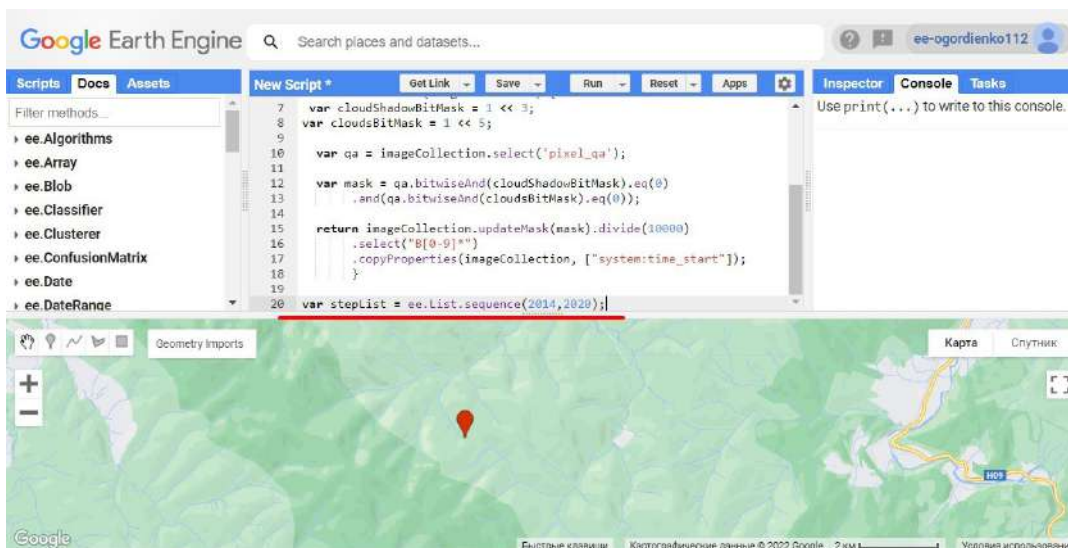


Рис. 254. Створення списку

Наступна функція створює медіанні значення за датами і застосовує маску хмар. За допомогою `ee.Date.fromYMD` створюються дані за датами. Функція створює композитне зображення за кожен рік і на вихід подає сім медіанних замаскованих зображень.

```
var filterCollection = stepList.map(function(year){
  var startDate = ee.Date.fromYMD(year,5,1);
  var endDate = ee.Date.fromYMD(year,9,15);
  var composite_i = imageCollection.
    filterDate(startDate, endDate)
      .map(maskL8sr)

      .median()
    .set('system:time_start',startDate);

  return composite_i;
});
```

Фільтрація колекції за `stepList` де зберігаються дати років.
`startDate` бере дані за 2014.05.01 `endDate` бере дані за 2020.09.15
фільтрація даних за початковою і кінцевою датою.
`map` застосовує функцію `maskL8sr`, що маскує хмари
`median` рахує медіанне значення
`.set('system:time_start',startDate);`
використовується для коректного відображення дат на графіках, зображеннях тощо.
`composite_i` повертає розраховані значення

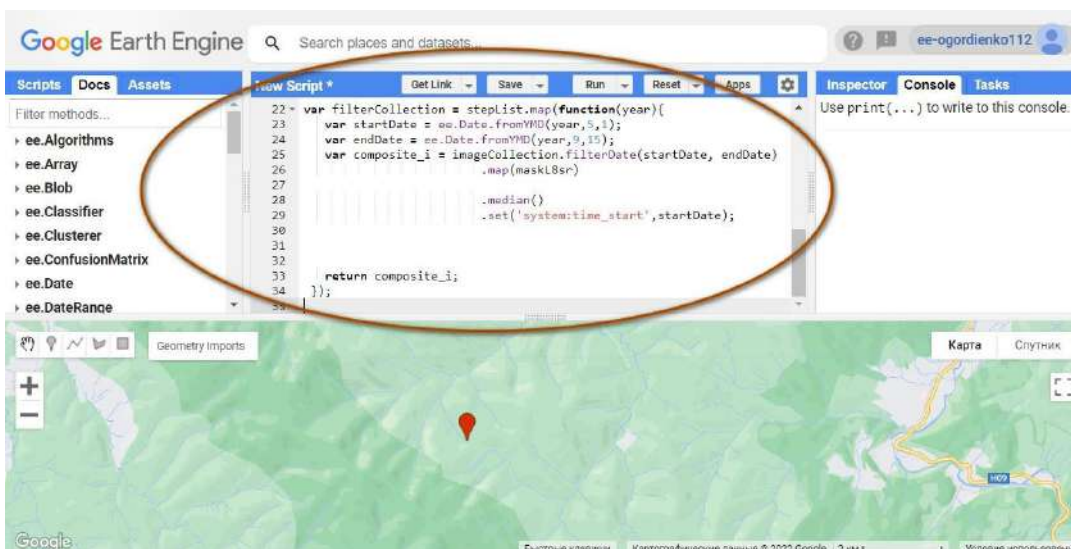


Рис. 255. Сортування даних

Наступна команда створить змінну, в якій будуть поміщені медіанні значення індексу NDVI за кожен рік в одній колекції зображень.

Команда **print** (друк) виводить результат у консоль.

<pre>var yearlyComposites = ee.ImageCollection(filterCollection); print(yearlyComposites, 'Masked and Filtered Composites');</pre>	Вся інформація з filterCollection поміщається до yearlyComposites print дані в консоль;
--	---

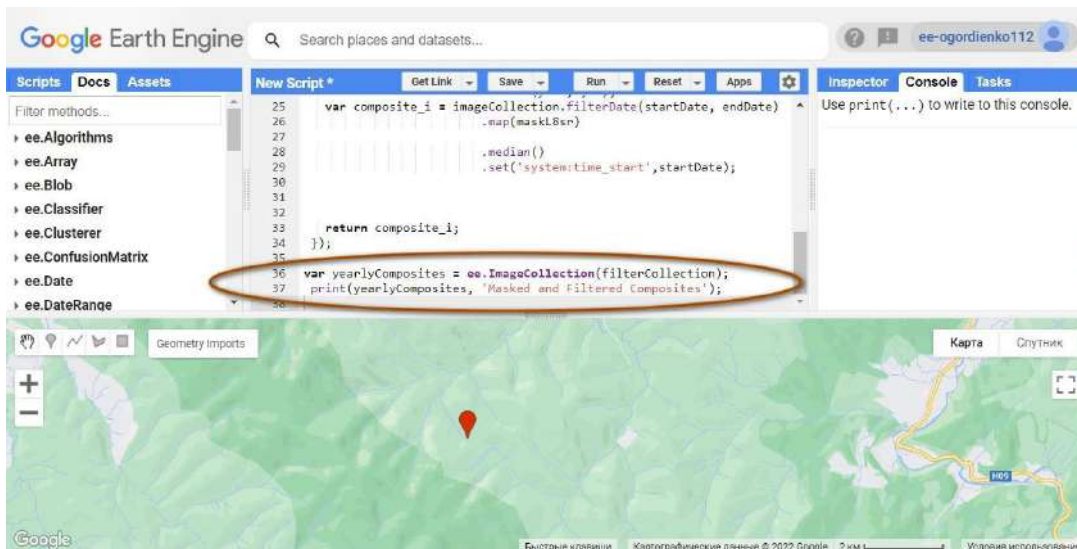


Рис. 256. Створення медіанних зображень

У консолі можна побачити, що в колекції зображень міститься сім зображень: це медіанні значення за період із 2014 р. по 2020 р. включно.

3. Розрахувати і візуалізувати індекс NDVI на вибрану ділянку.

Для розрахунку NDVI можна скористатися функцією. За допомогою **normalizedDifference** береться значення з каналів 'B5' та 'B4' і перейменовується розраховане значення на 'NDVI'.

За допомогою **addBands** до колекції додається новий канал, який має назву 'NDVI'.

<pre>var addNDVI = function(image) { var ndvi = image.normalizedDifference(['B5', 'B4']).rename('NDVI'); return image.addBands(ndvi); };</pre>	Функція з назвою addNDVI бере на вхід 'B5' та 'B4'. Розраховує нормалізовану різницю за допомогою normalizedDifference. Повертає новий канал зі значенням 'NDVI'
--	--

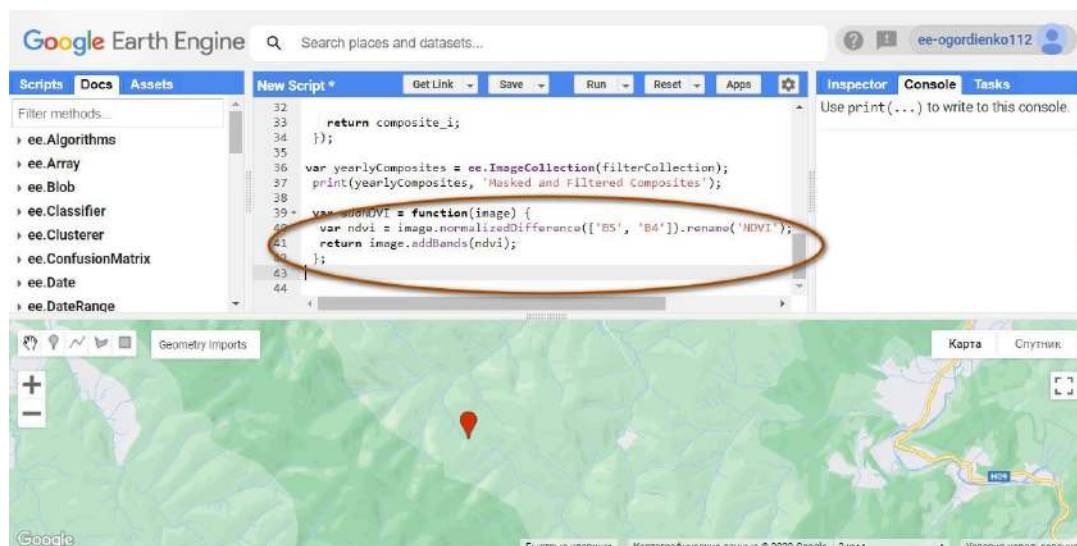


Рис. 257. Налаштування каналів

Для візуалізації індексу NDVI скористайтеся рядком коду:

```
var ndvi_palette = {"bands":["NDVI"],"min":-1,"max":1,"palette":["d73027','f46d43','fdae61','fee08b','ffffbf','d9ef8b','a6d96a','66bd63','1a9850','006837']};
```

bands параметр, що візуалізується **min max** значення для індексу NDVI **palette** всі значення кольорів у HEX-кодунанні

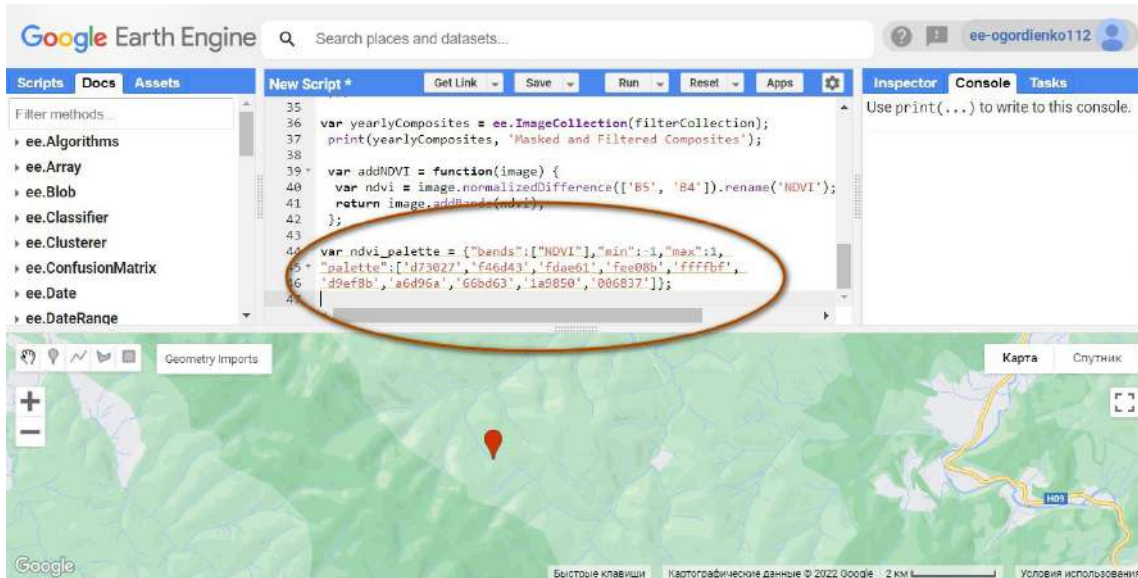


Рис. 258. Налаштування відображення

Для створення окремого каналу, який буде містити тільки значення індексу NDVI, скористайтеся кодом. Змінна *yearly* застосовує функцію *yearlyComposites*, в якій містяться замасковані значення, й *addNDVI*, де зберігається значення індексу NDVI.

У змінній *ndviCollection* будуть міститися лише значення індексу NDVI, перевірити це можна за допомогою команди **print** (друк).

```
var yearly= yearlyComposites.  
map(function(image){  
  return addNDVI(image);  
});  
var ndviCollection = yearly.select('NDVI');  
print(ndviCollection, 'With NDVI as Band');
```

До *yearlyComposites* застосовується функція *addNDVI*; з *yearly* обираються всі значення 'NDVI'; Результат у консоль;

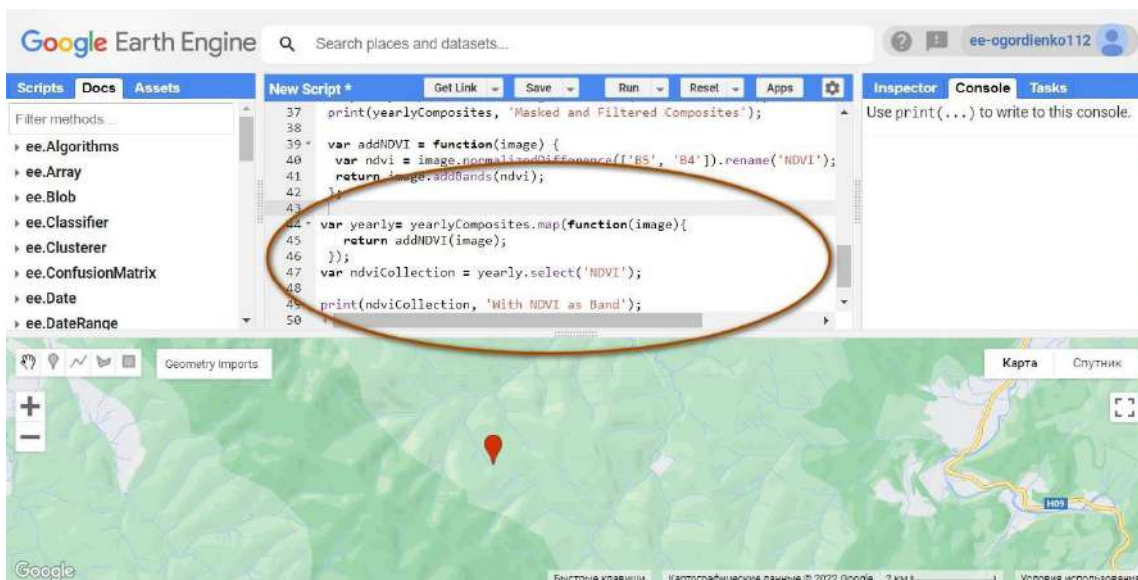


Рис. 259. Створення функції

У консолі з'явиться новий шар, який буде містити лише канал з NDVI.
Для створення зображення медіанного значення за один рік скористайтеся рядком:

<pre>var y2020 = ndviCollection.filterDate('2020-01-01','2020-12-31') .first() .clip(aoi); var y2014 = ndviCollection.filterDate('2014-01-01','2014-12-31') .first() .clip(aoi);</pre>	<p>Змінна y2020 звертається до колекції відфільтрованих, замаскованих зображень. Фільтрує їх за датою, бере перше зображення, фільтрує за aoi</p>
--	---

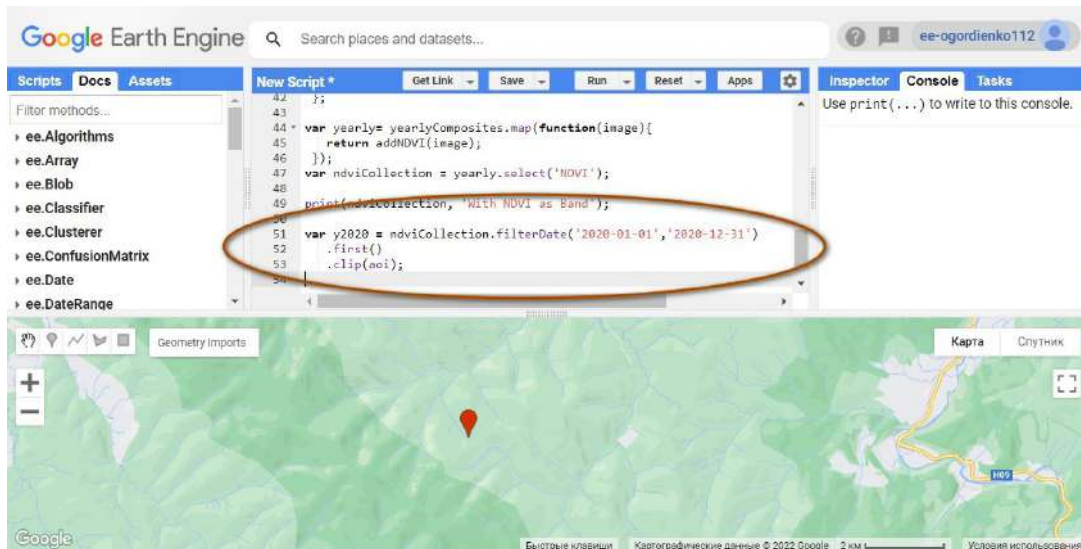


Рис. 260. Додавання змінних до шару

Для додавання його на мапу скористайтеся командою:

<pre>Map.addLayer(y2020, ndvi_palette, '2020 NDVI'); Map.addLayer(y2014, ndvi_palette, '2014 NDVI');</pre>	<p>Додавання шару y2020 на мапу з візуалізацією ndvi_palette та назвою '2020 NDVI'</p>
--	--

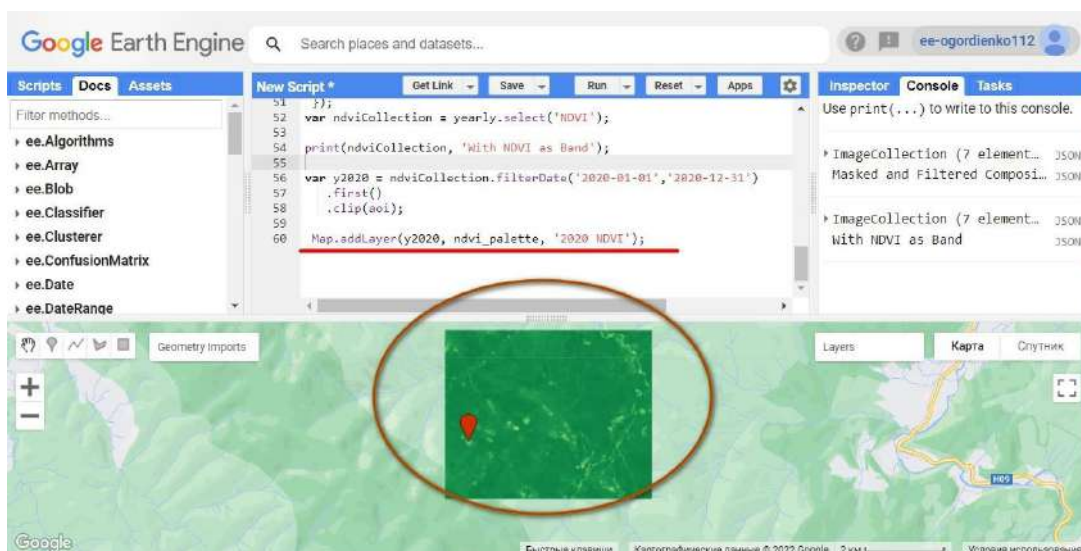


Рис. 261. Додавання шару на мапу

4. Побудувати графік змін індексу NDVI на конкретну точку.

Тепер в аналізі буде ще використано графік змін для імпортованого шару aoi. Для цього скористайтеся кодом нижче. Натисніть **Run** (Запуск скрипту).

<pre>var chart = ui.Chart.image. series({ imageCollection: ndviCollection.select('NDVI'), region: poi, scale: 30 }).setOptions({title: 'Point : NDVI Over Time'}); print(chart);</pre>	<p>До chart додається колекція зображень ndviCollection обирається значення 'NDVI'</p> <p>зона інтересу poi масштаб 30 title назва графіка</p> <p>print результат у консоль</p>
--	---

У результаті з'явиться графік.

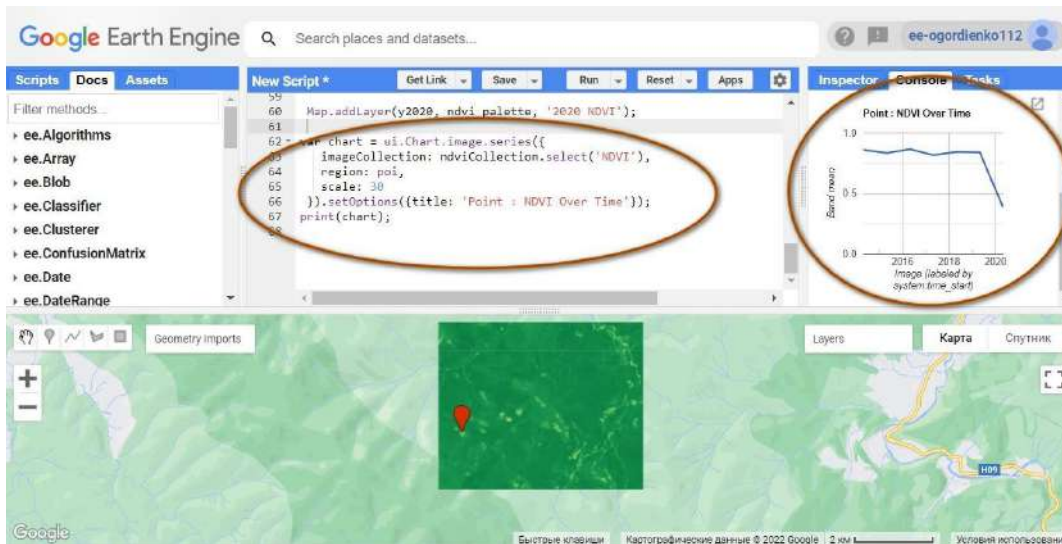


Рис. 262. Створення графіка

5. Створити анімацію змін індексу NDVI на вибрану територію.

Наступним кроком буде створення анімації для колекції. Ця анімація створена за допомогою медіанних значень за кожен рік. Завдяки такому способу можна побачити, як щороку змінювався ліс і з'явилися вирубки.

Для визначення зони для анімації та візуалізації скористайтеся кодом:

<pre>var Vis = ndviCollection.map(function(img) { return img.visualize(ndvi_palette).clip(aoi); });</pre>	<p>Змінна Vis застосовує функцію img до ndviCollection, за допомогою visualize відображає із заданими параметрами анімацію</p>
---	--

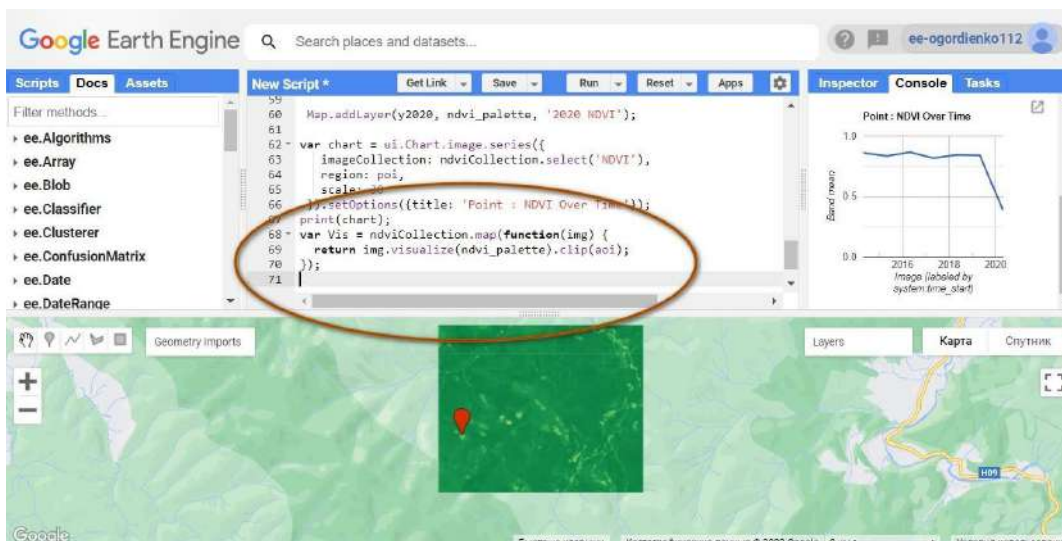


Рис. 263. Створення графіка за параметрами

Для задавання параметрів для анімації скористайтеся рядком коду:

<pre>var gifParams = { 'region': aoi, 'framesPerSecond': 1 };</pre>	gifParams на вхід має зону цікавості aoi та кількість кадрів на секунду 1
---	---

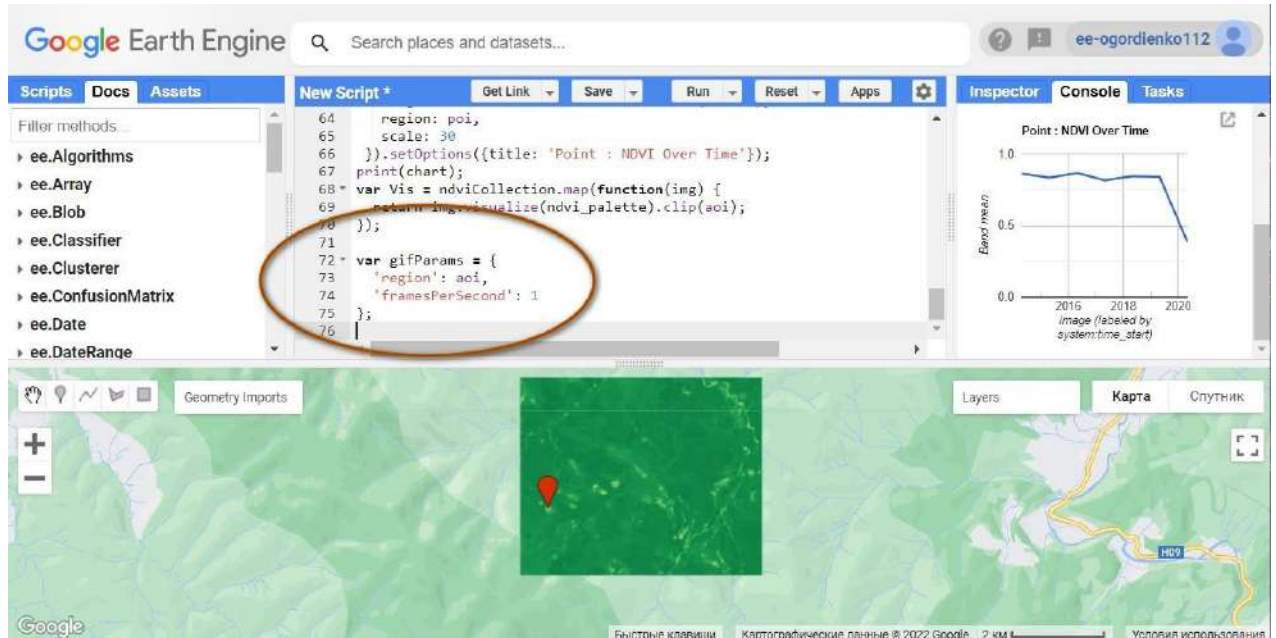


Рис. 264. Створення анімації

Для виведення в консоль скористайтеся кодом:

<pre>print(ui.Thumbnail(Vis, gifParams));</pre>	Результат <code>ui.Thumbnail</code> гіфки в консоль, з параметрами <code>Vis</code> (складена гіфка) та <code>gifParams</code> (параметри анімації)
---	---

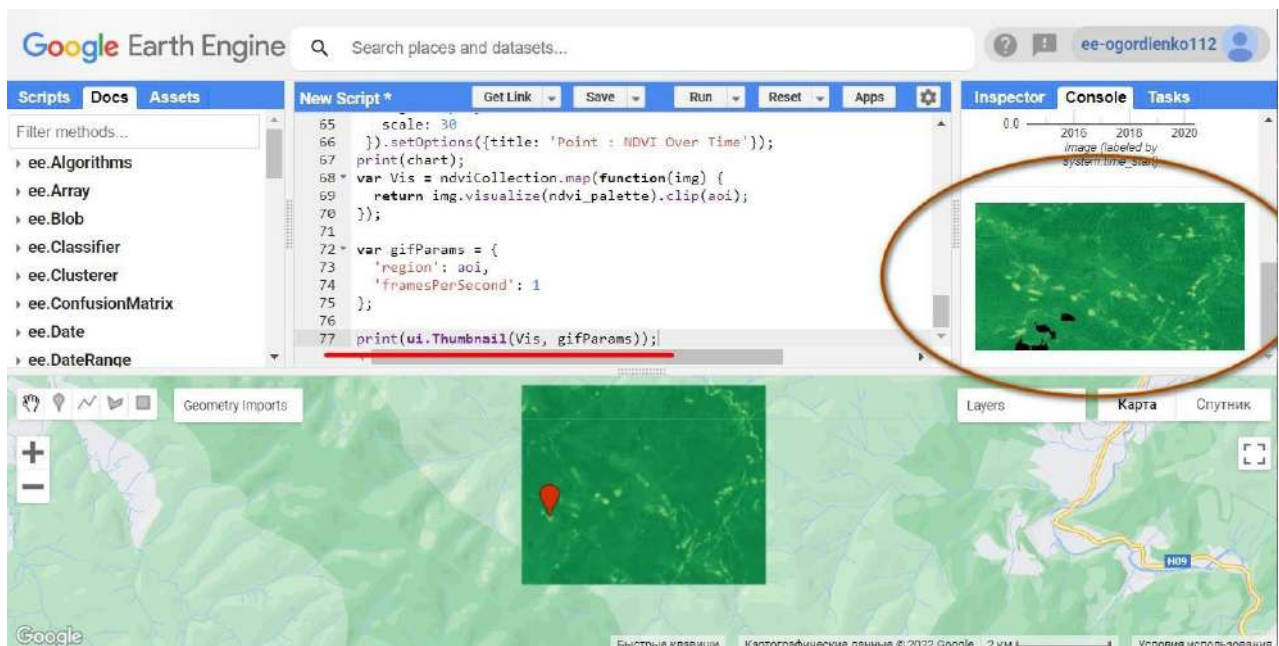


Рис. 265. Виведення анімації в консоль

? Завдання для перевірки

Додайте шари з 2014 р. по 2020 р.

Поки що у вкладці *Layers (Шари)* відображається один шар з назвою '2020 NDVI'. Він бере інформацію із шару у 2020 р.

Створіть ще шість нових шарів для 2014, 2015, 2016, 2017, 2018, 2019 рр.

Для цього за аналогією для змінної

```
var y2020 = ndviCollection.filterDate('2020-01-01','2020-12-31')  
  .first()  
  .clip(aoi);
```

створіть шар для зазначених вище років.

Потрібно, щоб змінні мали різні назви і в кожній змінній були відповідні дати для кожного року.

За аналогією

```
Map.addLayer(y2020, ndvi_palette, '2020 NDVI');
```

додайте шість шарів на мапу

Результат повинен мати такий вигляд:

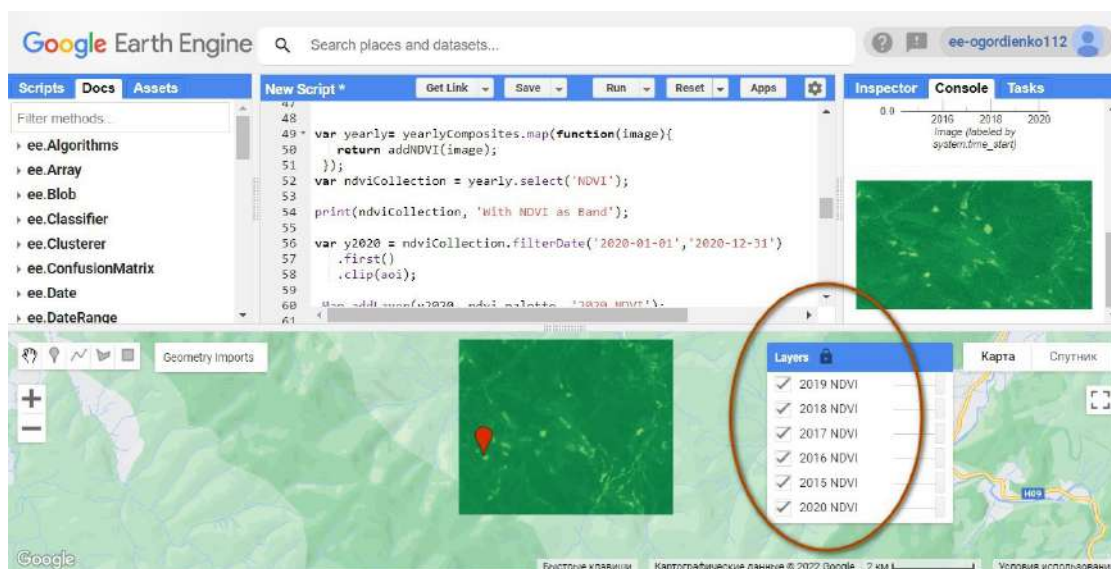


Рис. 266. Додавання шарів на мапу

? Завдання для перевірки

Додати шари NDVI ще за п'ять років (2015–2019).

Відобразити створені шари на мапі

За наведеним нижче QR-кодом ви можете переглянути відео виконання практичної роботи.



Рис. 267. Відео виконання практичної роботи

Готовий код можна відкрити за лінком:

https://code.earthengine.google.com/?scriptPath=users%2Fgis_rs%2Fcourse_MAN%3ALab_13

Створення інтерактивного застосунку з метою дослідження верхньої межі лісу в Карпатах

Ситуація

Верхня межа лісу (ВМЛ) – це умовна межа лісового поясу та високогір'я, де дерева мають мінімальну висоту 5 метрів, мінімальну зімкнутість крон 0,3¹. Чому взято такі показники? П'ятиметрові дерева здебільшого піднімаються над середньою або максимальною (4 метри) потужністю снігового покриву. Зімкнутість крон деревного ярусу 0,3 (проективне покриття) є показником ступеня ценотичного впливу дерев одного на одне¹.

У більшості ландшафтів Українських Карпат ВМЛ перебуває на висоті 1200–1300 метрів над рівнем моря, а подекуди й 800 метрів. В Українських Карпатах трапляється ландшафтна (природна) й антропогенна (господарська) ВМЛ. Ландшафтна (природна) ВМЛ знижена під впливом людської діяльності (наприклад, надмірного випасання тварин тощо) на 300–400 метрів і збереглася фрагментарно, зокрема в Чорногорі. Наслідком антропогенного чинника стало знищення цілого поясу смерекових лісів у ландшафтах Бескидів, на південних схилах Свидовця і Горганів та частково в південно-західній частині Чорногори, Чивчинських і Мармароських горах¹.

Висота місцевості над рівнем моря і її рельєф безпосередньо впливають на лісистість території. Рельєф є механічною перешкодою для перенесення повітряних мас. Теплі повітряні маси проникають через хребет вільно. Холодне повітря, з огляду на свою природу, не може піднятися вгору по хребту, а змушене обтікати його. Якщо висота гірських масивів перевищує потужність холодної повітряної маси, то гори стають для холодного повітря орографічним бар'єром. Тому і простежується різниця середньорічних температур по обидва боки хребта¹.

Завдання

У цьому завданні користувач буде взаємодіяти із застосунком, який виводить значення висоти рельєфу та вегетаційного індексу на екран. **Інтерактивна інформація на сайті – це інформація про взаємодію інших користувачів із сайтом, яка відображається в реальному часі, наприклад: кількість відвідувачів на сайті, останні коментарі, останні перегляди відео, фото, дописи.**

Також сюди можна включити розрахунки, які відбуваються на екрані, якщо вони відображаються за кліком чи взаємодією із сайтом.

Тож основним завданням цієї практичної роботи буде створити елементи користувацького інтерфейсу й окремий інтерактивний застосунок.

Алгоритм виконання завдання:

1. Створити два вікна для відображення інформації на мапі.
2. Підвантажити датасет адміністративних кордонів *FAO GAUL* для обрізання даних.
3. Підвантажити датасет цифрової моделі рельєфу *NASA SRTM* з позначками висоти.
4. Змінити стиль курсора.
5. Створити панель користувацького інтерфейсу і додати до неї функцію підрахунку висот.
6. Підвантажити датасет супутникових даних *Landsat 7 Collection 1 Tier 1 8-Day NDVI Composite* з розрахунком вегетаційним індексом.
7. Створити панель користувацького інтерфейсу і додати до неї функцію відображення *NDVI*.
8. Відфільтрувати колекцію *Sentinel-2 MSI: MultiSpectral Instrument, Level-2A* і підвантажити її на мапу.

¹ Байцар А. Верхня межа лісу в ландшафтах Українських Карпат, її охорона та оптимізація. *Вісник Львівського університету. Серія географічна*. 2014. Вип. 45. С. 166–177. URL: http://nbuv.gov.ua/UJRN/VLNU_Geograf_2014_45_21

9. Створити зони інтересу для гір Бребенескул, Говерла та Хом'як.

10. Створити інтерактивний застосунок і дослідити його.

Покрокова інструкція

Відкрийте *Code Editor* (Редактор коду): <https://code.earthengine.google.com/> і подивіться уважно, який вигляд має картографічне вікно (1).

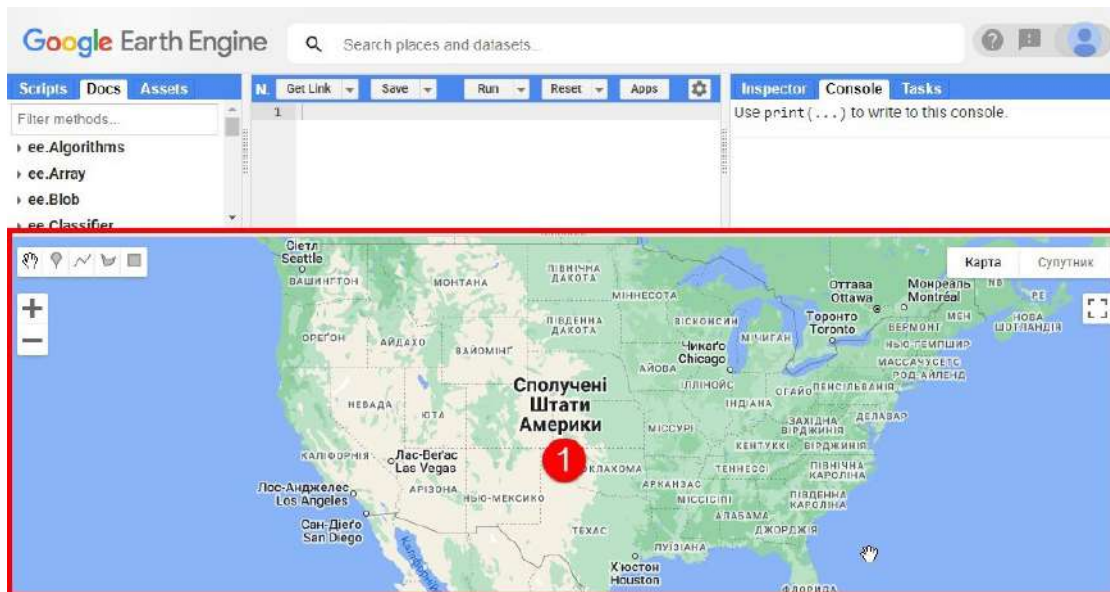


Рис. 268. Загальний вигляд програми

1. Створити два вікна відображення інформації на мапі.

Зараз ви маєте одну мапу (1) для відображення даних, і в цій практичній роботі ми розглянемо, як створити два паралельних вікна, з якими можна буде одночасно працювати.

Для того щоб створити два вікна мапи (див. рис. 1, 2), скористайтеся кодом нижче і натисніть *Run* (Запуск скрипту).

<pre>var left = ui.Map(); var right = ui.Map(); ui.root.clear(); ui.root.add(left); ui.root.add(right);</pre>	<p>Змінні <code>left</code> та <code>right</code> створюють нові мапи за допомогою <code>ui.Map</code></p> <p>Очищає кореневу панель. <code>add</code> додає <code>left</code> та <code>right</code></p>
---	--

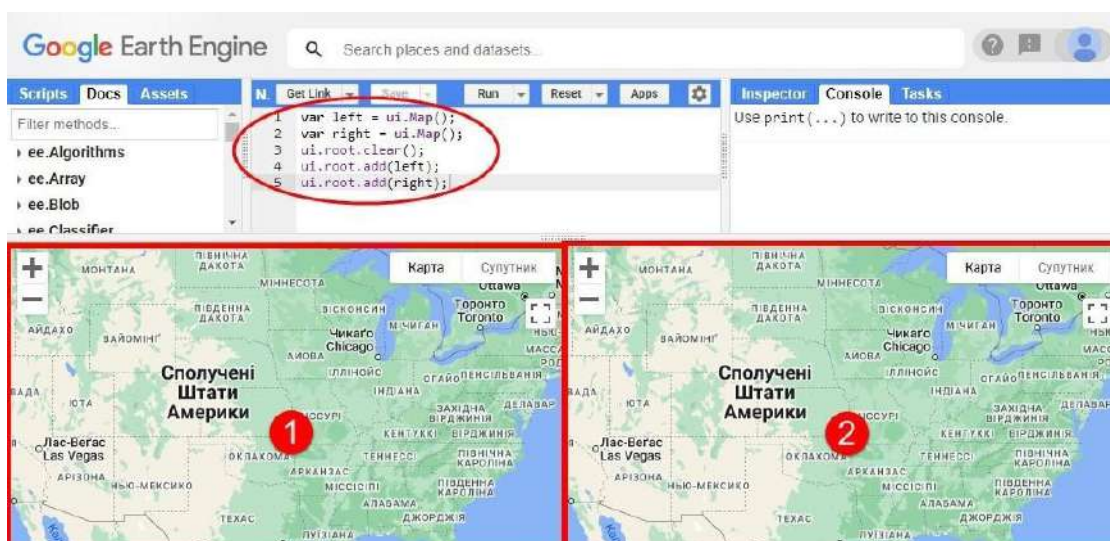


Рис. 269. Код для створення розподіленого екрана

Спробуйте змінити місцезположення мапи за допомогою миші в правому вікні (2).

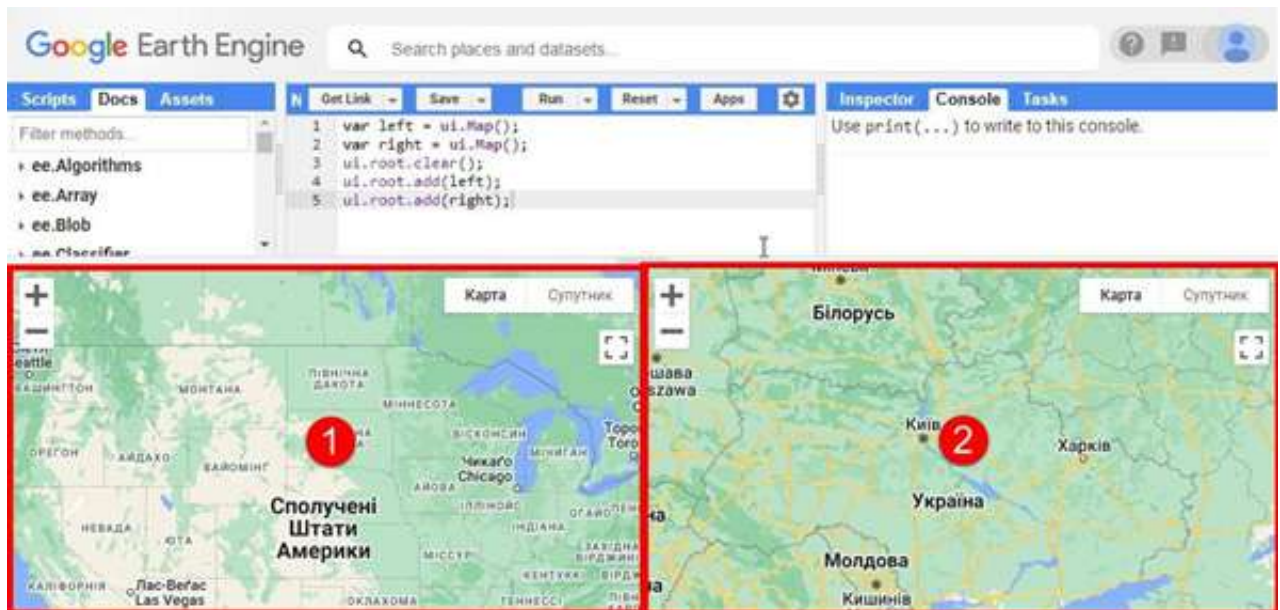


Рис. 270. Код для розподілення екрана

Для того щоб ці два вікна зумилися (працювали), паралельно потрібно скористатися рядком коду нижче. Натисніть **Run** (Запуск скрипту).

<code>ui.Map.Linker([left, right], 'change-bounds');</code>	Утиліта для створення зв'язаних мап;
---	--------------------------------------

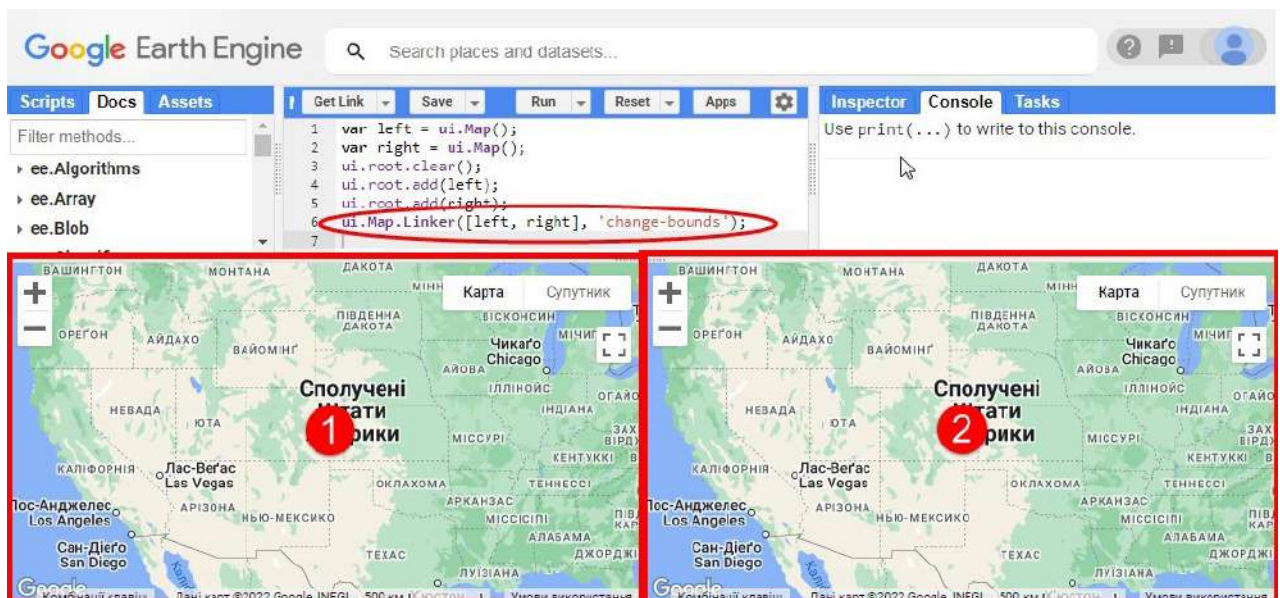


Рис. 271. Код для лінування мап

Тепер масштабування та зміна місцезположення, подані для одного з вікон, будуть паралельно виконуватися в обох вікнах.

Тепер потрібно призумитися до України. Для цього скористайтеся кодом нижче. Натисніть **Run** (Запуск скрипту).

<code>left.setCenter(31.672, 49.277,5);</code>	Центрування мапи за координатами
--	----------------------------------

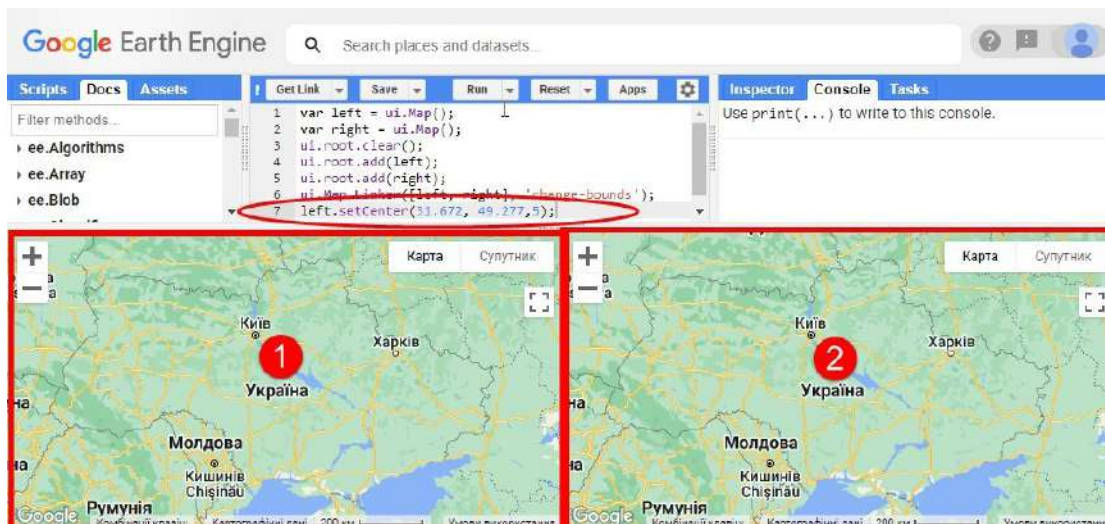


Рис. 272. Код для центрування мапи

2. Підвантажити датасет адміністративних кордонів *FAO* для обрізання даних.

Дані, використані в цій практичній роботі, будуть обрізатися в межах кордонів України за допомогою датасету *FAO GAUL: Global Administrative Unit Layers 2015, Country Boundaries* (Глобальні дані адміністративних одиниць 2015, Межі країн). Для завантаження даних скористайтеся кодом:

<pre>var UA = ee.FeatureCollection("FAO/GAUL/2015/level0").filter(ee.Filter.eq('ADM0_NAME', 'Ukraine'));</pre>	<p>Змінна UA бере інформацію зі списку 'ADM0_NAME' з тегом 'Ukraine'</p>
--	--

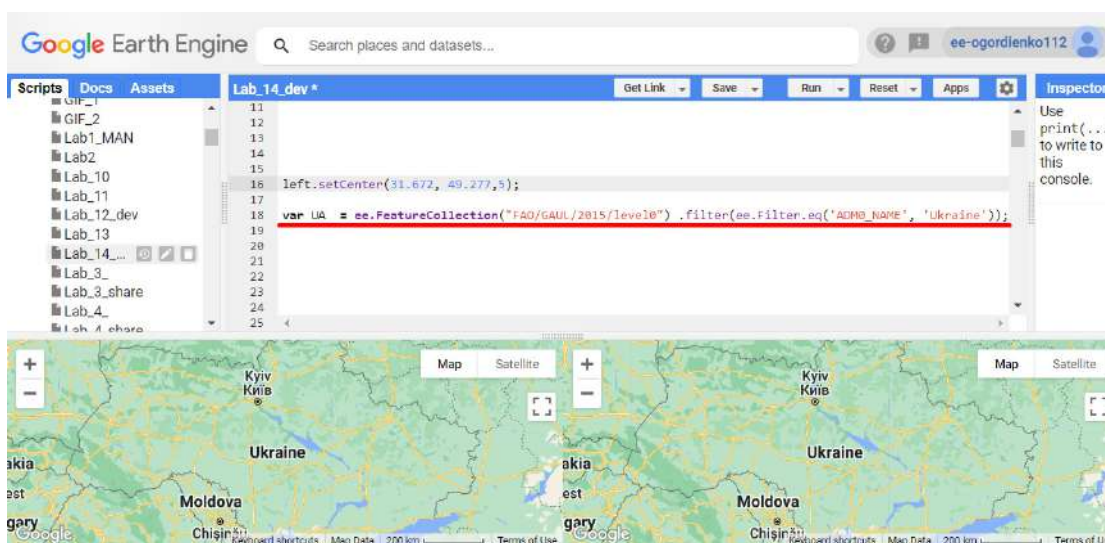


Рис. 273. Код для визначення зони цікавості

3. Підвантажити датасет цифрової моделі рельєфу *NASA SRTM* з позначками висоти.

На цьому кроці до лівого вікна буде завантажена й відображена *NASA SRTM Digital Elevation 30m* (*NASA SRTM Цифрова модель рельєфу 30 м*).

Щоби підвантажити дані, скористайтеся рядком:

<pre>var srtm = ee.Image('USGS/SRTMGL1_003'); left.addLayer(srtm.clip(UA),{min: 500, max: 2200, palette: ['ffffd4', 'fee391', 'fec44f', 'fe9929', 'd95f0e', '993404']}, 'SRTM');</pre>	<p>Змінна srtm бере зображення SRTM і за допомогою left.addLayer додає його на ліву мапу, обрізає за геометрією UA, встановлює відображення від 500 до 2200 з кольоровою гамою palette, задає назву 'SRTM'</p>
--	--

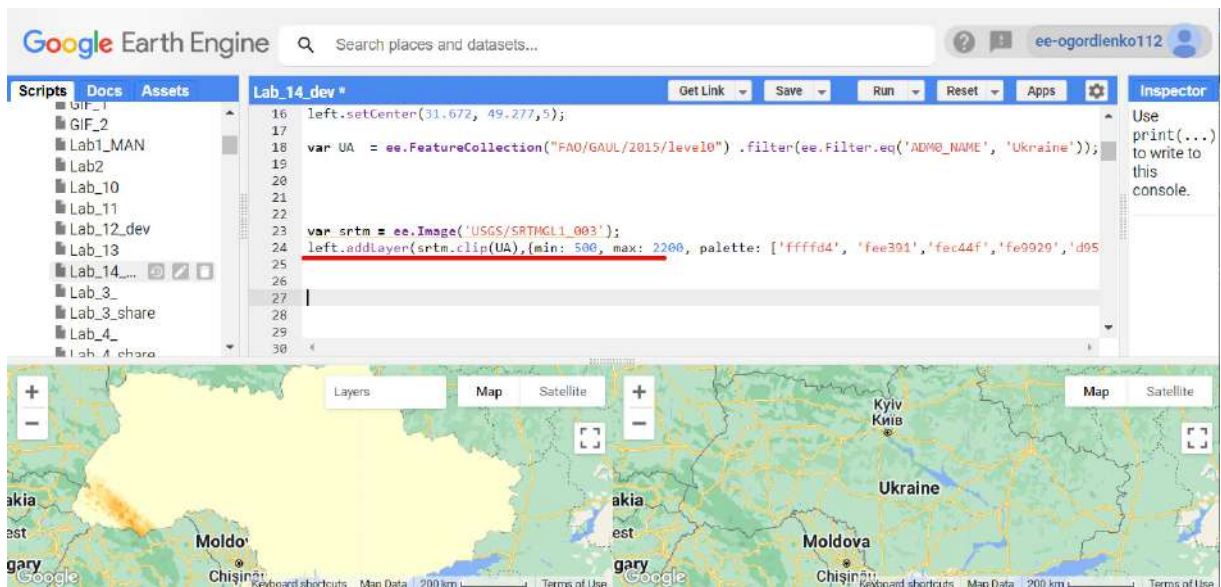


Рис. 274. Код для додавання даних

4. Змінити стиль курсора.

Наступний крок змінить вигляд курсора на мапі. За замовчуванням він подібний до «руки». Ці рядки змінять його вигляд на «хрест».

<pre>left.style().set('cursor', 'crosshair'); right.style().set('cursor', 'crosshair');</pre>	<p>Зміна вигляду курсора за допомогою <code>set</code> та <code>'crosshair'</code></p>
---	--

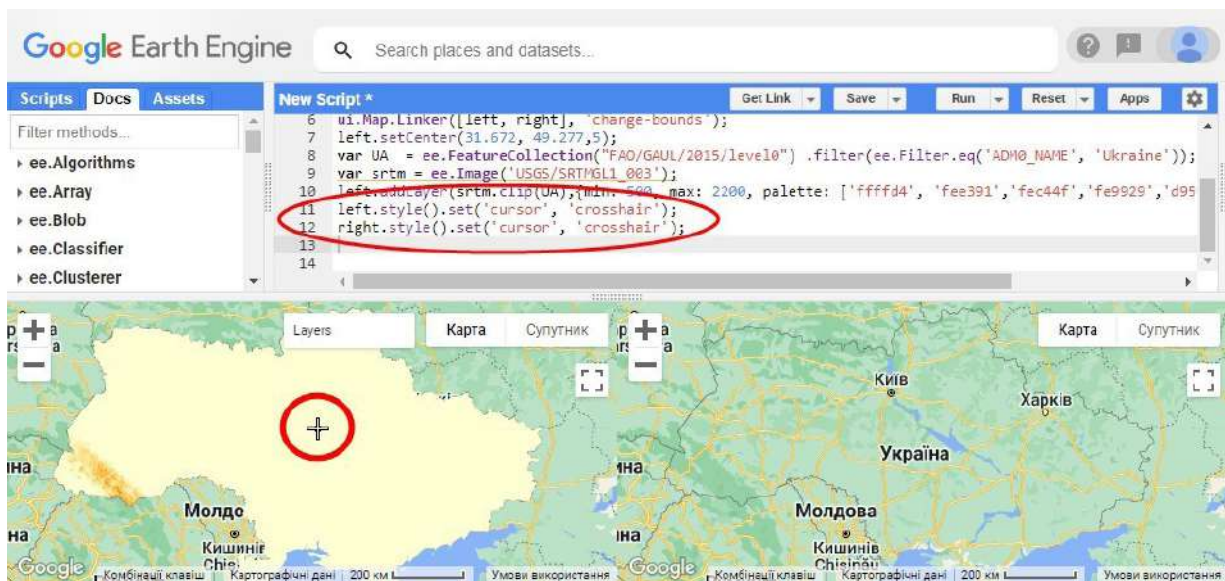


Рис. 275. Код для зміни курсора

5. Створити панель користувацького інтерфейсу і додати до неї функцію підрахунку висот.

На цьому етапі будуть створені елементи користувацького інтерфейсу, як-от панель з даними висот. Для створення панелі та підпису до неї (для прикладу задамо підпис «Дізнатися висоту») скористайтеся кодом нижче. Натисніть **Run** (Запуск скрипту).

<pre>var inspector_srtm = ui.Panel([ui.Label('Дізнатися висоту')]); left.add(inspector_srtm);</pre>	<p>Змінна <code>inspector_srtm</code> створює підпис до панелі користувацького інтерфейсу <code>left.add</code> додає панель до мапи</p>
---	--

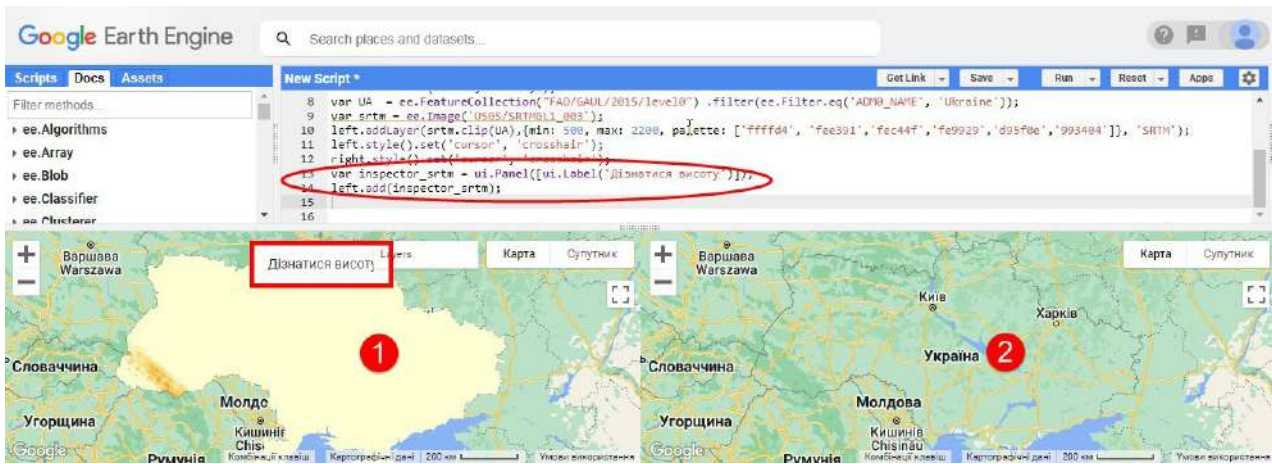


Рис. 276. Виведення панелі на екран

Наступним кроком буде встановлена команда, яка буде робити напис «Завантаження під час підвантаження даних» і встановлюватиме колір цього тексту. Визначатиме координати і за цими координатами братиме з **elevation** значення, що зберігається в srtm висоти. Виводитиме на екран результат зі значенням два знаки після коми. Натисніть **Run** (Запуск скрипту).

<pre> left.onClick(function(coords) { inspector_srtm.widgets().set(0, ui.Label({ value: 'Завантаження...', style: {color: 'gray'} })); var point = ee.Geometry.Point(coords.lon, coords.lat); var select = srtm.select('elevation'); var sample = select.sample(point, 30); var computedValue = sample.first(). get('elevation'); computedValue.evaluate(function(result) { inspector_srtm.widgets().set(0, ui.Label({ value: 'Висота: ' + result.toFixed(2), })); }); }); </pre>	<p>onClick реєструє зворотний виклик, який запускається під час натискання на мапу зі значеннями 'Завантаження' кольором color: 'gray'</p> <p>Змінна point бере координати точки, обирає значення в растрі 'elevation'</p> <p>sample бере вибірку пікселів зображення, повертаючи їх у вигляді значень.</p> <p>computedValue бере розраховане значення</p> <p>Додає нову мітку з написом Висота та розрахованим значенням 'elevation'</p>
---	---

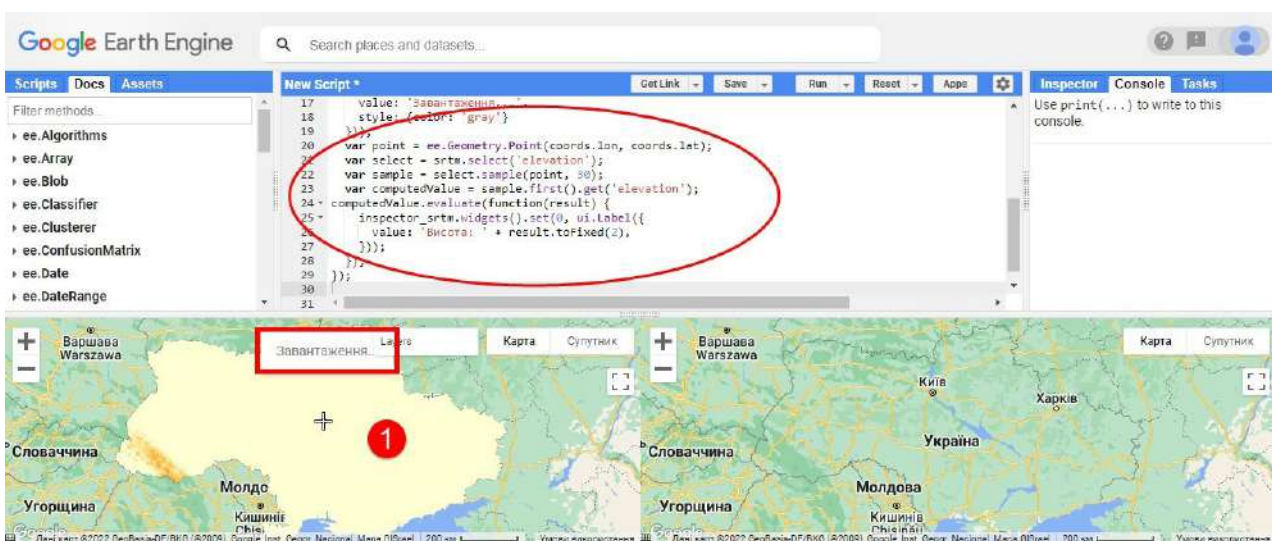


Рис. 277. Код для створення візуальних ефектів кнопок

Дослідіть мапу: спробуйте натиснути на будь-яке місце в Україні на лівій мапі (1).

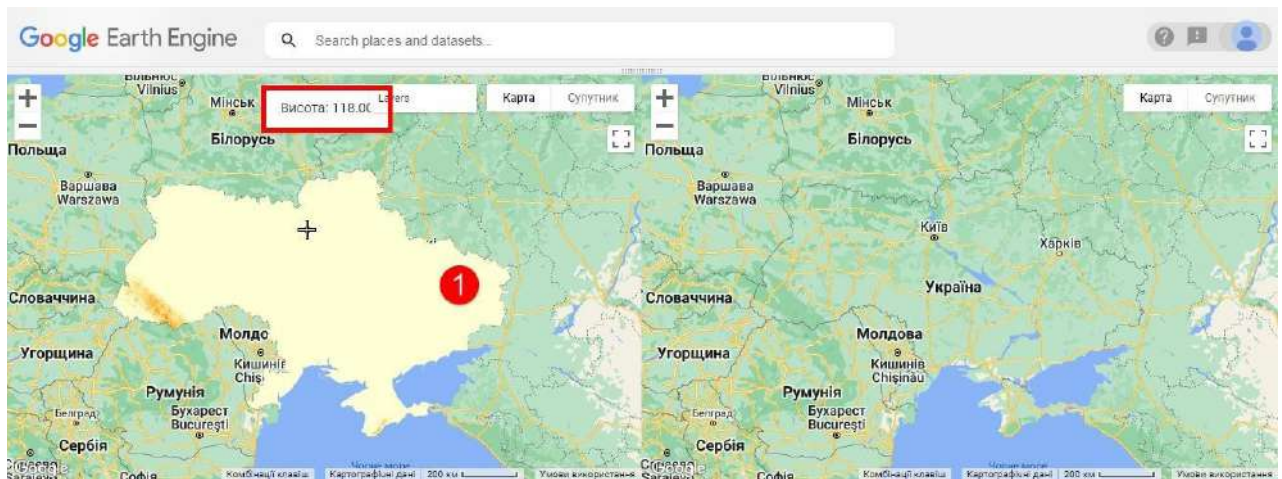


Рис. 278. Код для створення інспектора висот

З кожним натисканням на різні ділянки на мапі буде змінюватися значення висоти.

6. Підвантажити датасет супутникових даних Landsat 7 Collection 1 Tier 1 8-Day NDVI Composite з розрахованим вегетаційним індексом.

Наступним кроком буде підвантаження даних до правої мапи (2), а саме даних супутника **Landsat 7** з індексом NDVI.

Датасет **Landsat 7 Collection 1 Tier 1 8-Day NDVI Composite** (Landsat 7 Колекція 1 Рівень 1 8-денні комбіновані NDVI).

Ці комбіновані зображення Landsat 7 Collection 1 Tier 1 виготовлені з орторектифікованих сцен Tier 1 із використанням обчисленого коефіцієнта TOA (відбиття у верхній частині атмосфери).

Нормалізований індекс різниці рослинності NDVI генерується на основі каналів ближнього ІЧ-діапазону та червоного діапазону кожного знімка і має значення від $-1,0$ до $1,0$.

Для підвантаження індексу NDVI скористайтеся кодом нижче. Натисніть **Run** (Запуск скрипту).

<pre>var ls = ee.ImageCollection('LANDSAT/LC8_ L1T_8DAY_NDVI') .filterDate('2015-04-01', '2021-09-01') .filter(ee.Filter.dayOfYear(121,273)); var vis = {min: 0, max: 1, palette: ['8c510a', 'd8b 365', 'f6e8c3', 'c7eae5', '5ab4ac', '01665e']} ; right.addLayer(ls.median().clip(UA), vis, 'NDVI');</pre>	<p>Змінна ls</p> <p>Фільтрує за датами Фільтрує за днями в році (теплі дні) vis задає параметри візуалізації</p> <p>right.addLayer додає шар на мапу з медіанним значенням median та вирізає за геометрією UA</p>
---	---

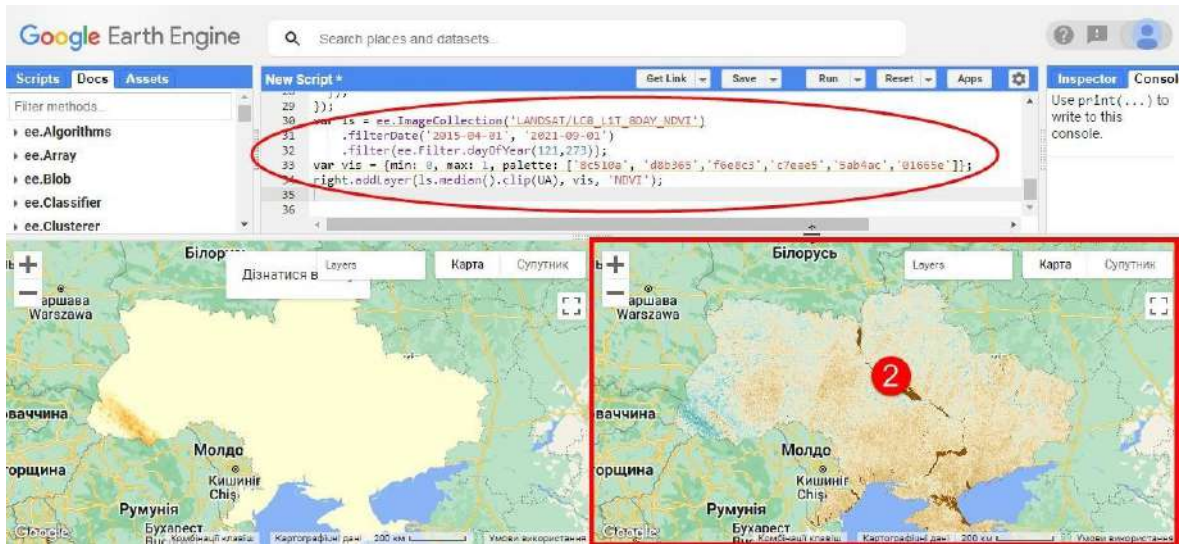


Рис. 279. Код для створення датасету

На мапі з'явиться розрахований індекс NDVI за теплий весняно-літній період.

7. Створити панель користувацького інтерфейсу і додати до неї функцію відображення NDVI.

На цьому етапі будуть створені елементи користувацького інтерфейсу. На панель будуть виводитися дані з індексом NDVI. Для створення панелі та підпису до неї (для прикладу задамо підпис "Дізнатися NDVI") скористайтесь кодом. Натисніть **Run** (Запуск скрипту).

<pre>var inspector_ndvi = ui.Panel({ui. Label('Дізнатися NDVI')}); right. add(inspector_ndvi);</pre>	<p>inspector_ndvi створює нову панель з підписом Дізнатися NDVI right.add додає панель на праву мапу</p>
--	--

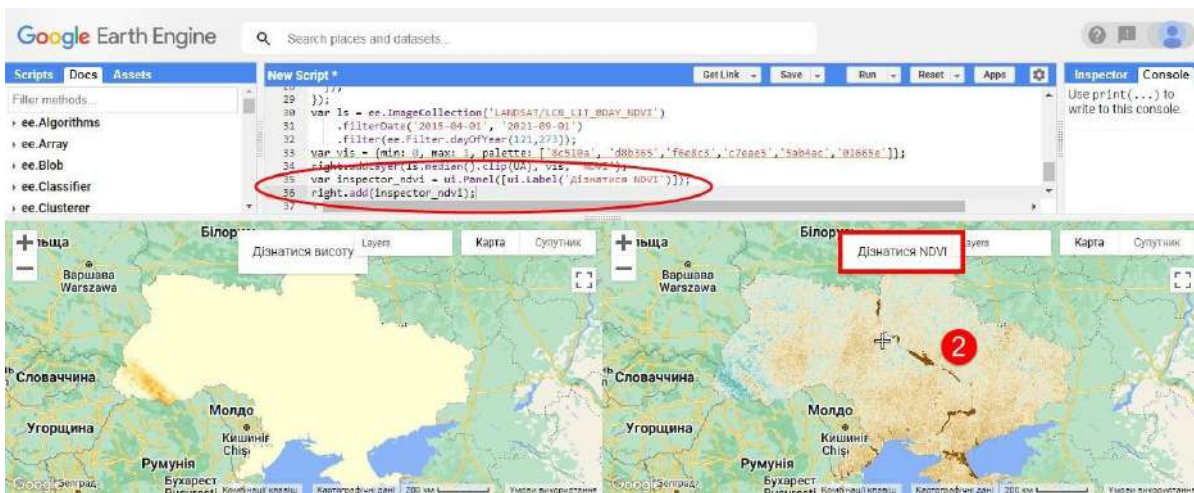


Рис. 280. Код для створення інспектора

Наступним кроком буде встановлена команда, яка буде робити напис «Завантаження під час підвантаження даних», визначатиме координати і за цими координатами братиме з NDVI значення, що зберігаються в ls. Виводитиме на екран результат зі значенням два знаки після коми. Натисніть **Run** (Запуск скрипту).

```

right.onClick(function(coords) {
  inspector_ndvi.widgets().set(0, ui.Label({
    value: 'Завантаження...',
    style: {color: 'gray'}
  }));var point = ee.Geometry.Point(coords.lon,
  coords.lat);
  var meanNdvi = ls.reduce('mean');
  var sample = meanNdvi.sample(point, 30);
  var computedValue = sample.first().get('NDVI_mean');
  computedValue.evaluate(function(result) {
    inspector_ndvi.widgets().set(0, ui.Label({
      value: 'Середній NDVI: ' + result.toFixed(2),
    }));
  });
});
});

```

onClick реєструє зворотний виклик, який запускається під час натискання на мапу зі значеннями 'Завантаження' кольором color: 'gray' Змінна point бере координати точки, обирає значення в растрі 'mean'

sample бере вибірку пікселів зображення, повертаючи їх у вигляді значень. computedValue бере розраховане значення

Додає нову мітку з написом Середній NDVI та розрахованим значенням 'NDVI_mean'

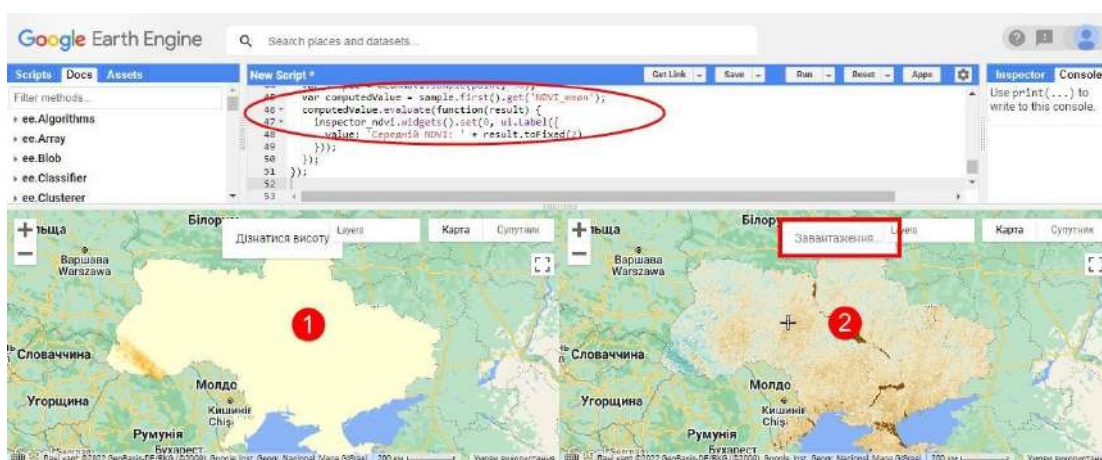


Рис. 281. Код для розрахунку вегетаційного індексу

Дослідіть праву мапу, натисніть у будь-якому місці на території України, щоб дізнатися індекс NDVI.

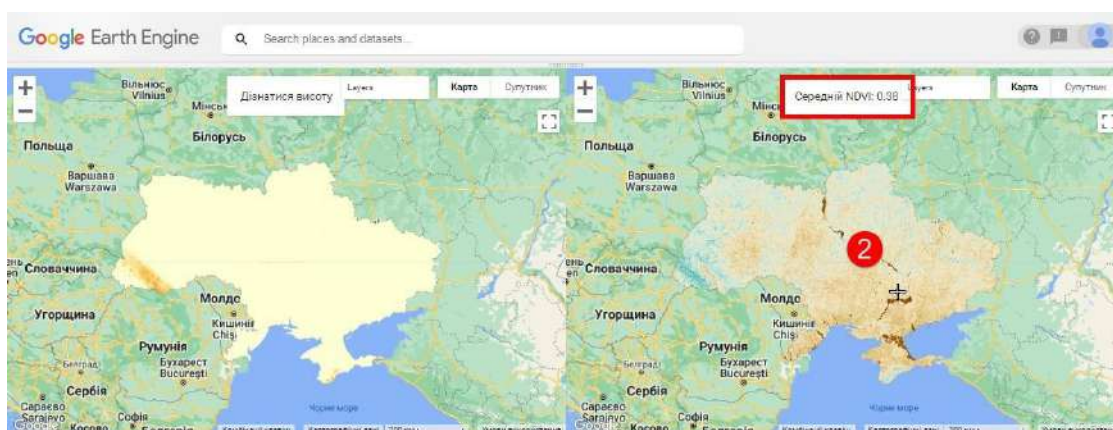


Рис. 282. Видягд розподіленого екрана

8. Відфільтрувати колекцію Sentinel-2 MSI: MultiSpectral Instrument, Level-2A і підвантажити її на мапу.

Наступним кроком буде підвантаження знімків у природних кольорах для України із супутника Sentinel-2. Для маскування хмар скористайтеся кодом:

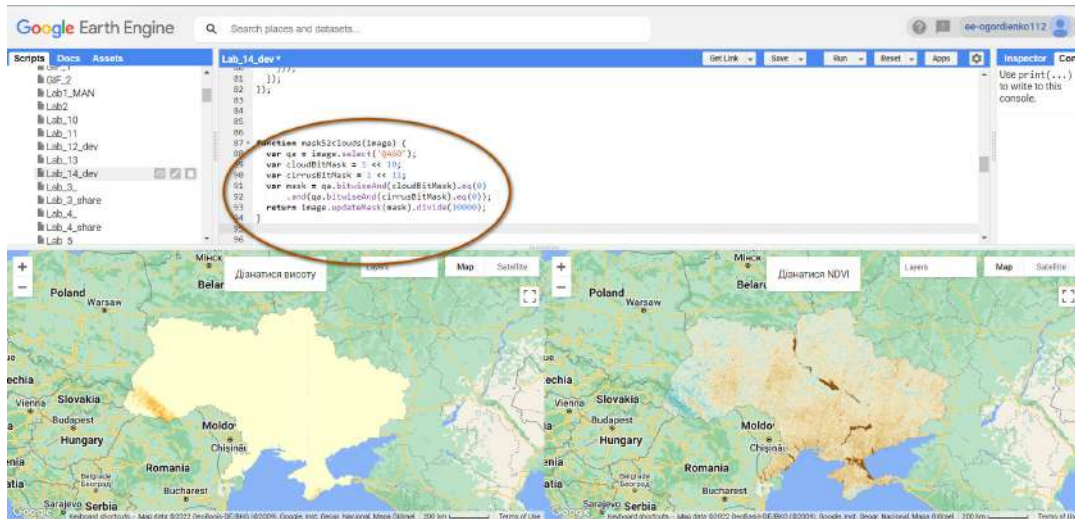


Рис. 283. Код для маскування хмар

```
function maskS2clouds(image) {
  var qa = image.select('QA60');
  var cloudBitMask = 1 << 10;

  var cirrusBitMask = 1 << 11;

  var mask = qa.bitwiseAnd(cloudBitMask).eq(0)
    .and(qa.bitwiseAnd(cirrusBitMask).eq(0));
  return image.updateMask(mask).divide(10000);
}
```

Функція `maskS2clouds` задає змінну `qa`, де вибирає канал `'QA60'`, який може набувати значень, представлених вище. `cloudBitMask` зсуває значення побітово на 10 знаків
`cirrusBitMask` зсуває значення побітово на 11 знаків

Змінна `mask` на поелементній основі обчислює бітове значення та вхідне значення, які дорівнюють 0 для `cloudBitMask` та `cirrusBitMask`

Вся функція встановлює маску для зображення і ділить її на 10 тисяч

Для підвантаження датасету Sentinel-2 скористайтеся кодом:

```
var dataset =
ee.ImageCollection('COPERNICUS/S2_SR')
  .filterDate('2016-05-01', '2021-01-30')
  .filter(ee.Filter.dayOfYear(121,273))
  .filter(ee.Filter.lt('CLOUDY_PIXEL_PERCENTAGE', 1))
  .map(maskS2clouds)
  .select('B4', 'B3', 'B2');
```

Змінна `dataset` звертається до колекції зображень `COPERNICUS/S2_SR`

Фільтрує за датами

Фільтрує за теплими днями в році

Обирає знімки з хмарністю менше 1 відсотка

Застосовує маску хмар `maskS2clouds`, обирає канали `'B4', 'B3', 'B2'`

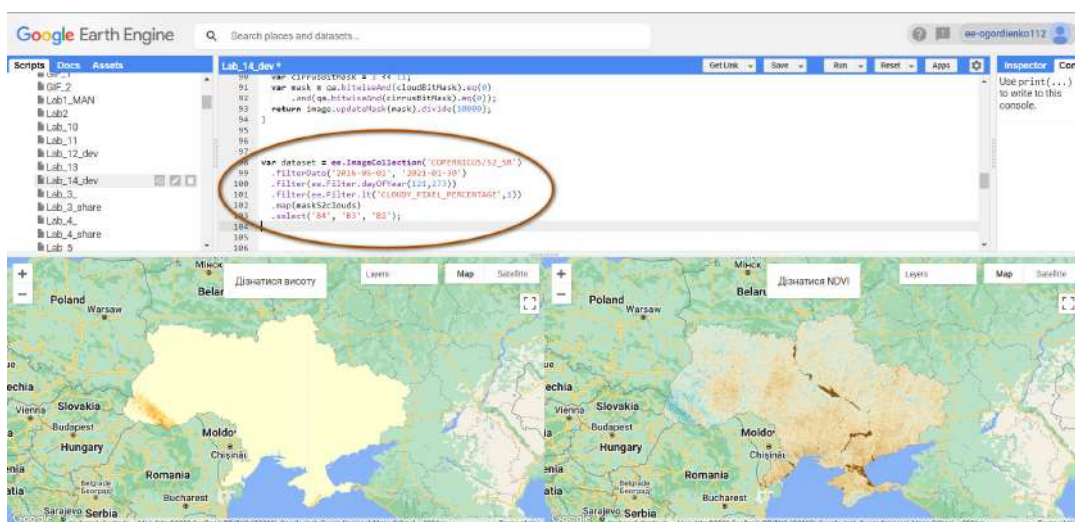


Рис. 284. Створення датасету

Для візуалізації супутникових знімків у природних кольорах скористайтеся кодом:

<pre>var visualization = { min: 0.0, max: 0.3, bands: ['B4', 'B3', 'B2'], };</pre>	<p>Змінна <code>visualization</code> встановлює значення відображення мінімальне <code>min</code> та максимальне <code>max</code> обирає канали <code>'B4', 'B3', 'B2'</code></p>
--	---

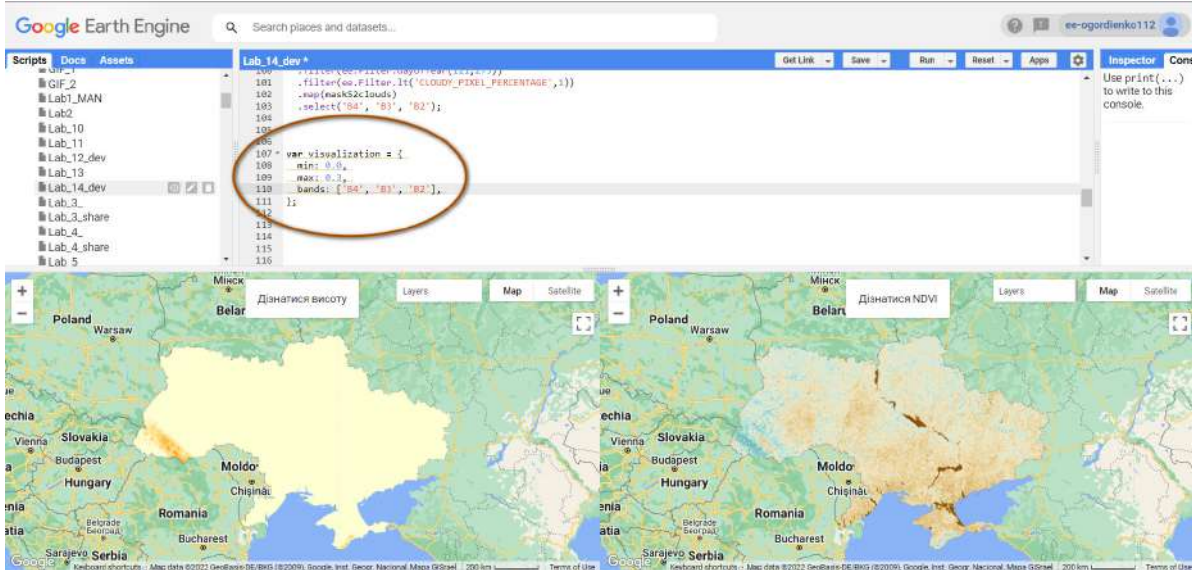


Рис. 285. Код для візуалізації

Наступним кроком буде завантаження відфільтрованих візуалізованих даних до обох мап – лівої і правої.

Скористайтеся кодом. Натисніть **Run** (Запуск скрипту).

<pre>right.addLayer(dataset.median().clip(UA), visualization, 'True Color (432)'); left.addLayer(dataset.median().clip(UA), visualization, 'True Color (432)');</pre>	<p><code>right.addLayer</code> та <code>left.addLayer</code> додають до обох мап дані з <code>dataset</code> з медіанним знімком <code>median</code> та обрізаними за геометрією <code>UA</code> з назвою <code>'True Color (432)'</code></p>
---	---

Нові шари з'являться у вкладці **Layers** (Шари). Їх відображення можна змінювати.

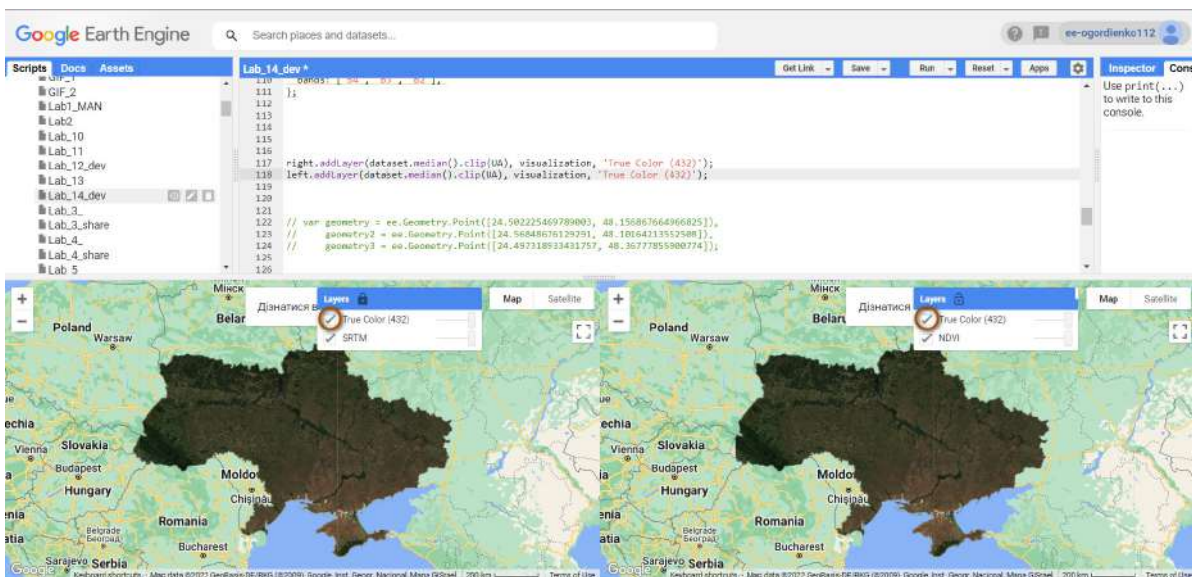


Рис. 286. Відображення даних

9. Створити територію інтересу для гір Бребенескул, Говерла, Хом'як.

Наступними кроками будуть створені зони інтересу гір Бребенескул, Говерла, Хом'як. Скопіюйте координати цих гір.

<pre>var geometry = ee.Geometry.Point([24.502225469789003, 48.156867664966825]), geometry2 = ee.Geometry.Point([24.56848676129291, 48.10164213552508]), geometry3 = ee.Geometry.Point([24.497318933431757, 48.36777855900774]);</pre>	<p>Змінна, що задає три точки інтересу з назвами geometry, geometry2, geometry3</p>
---	---

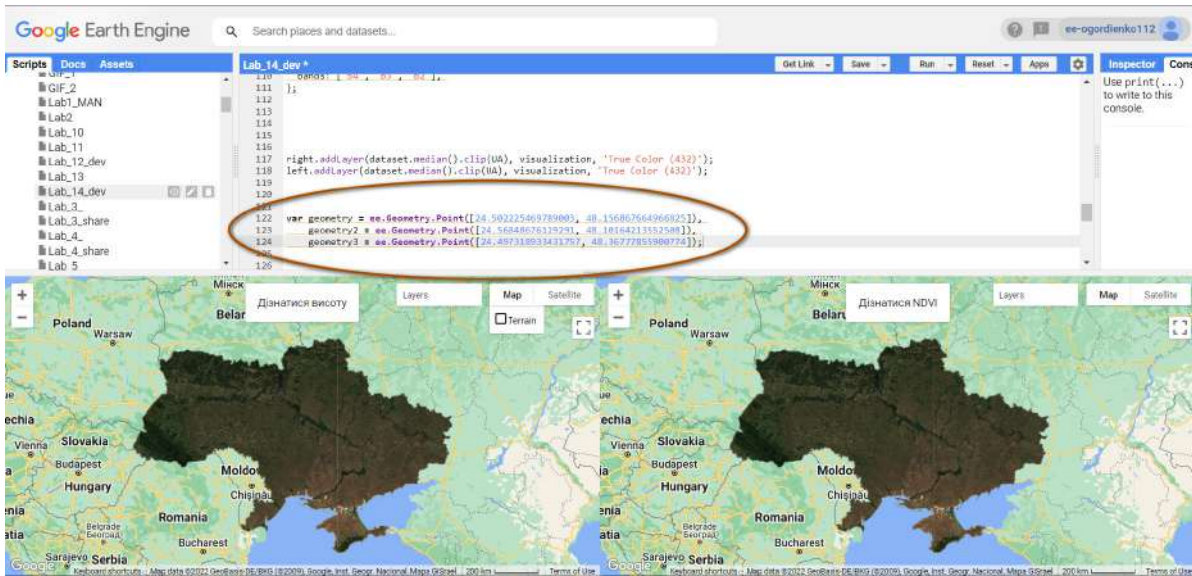


Рис. 287. Код для створення кнопок

За допомогою наступного коду буде створена панель, в якій розміщуватимуться кнопки з назвами гір. Натисніть **Run** (Запуск скрипту).

<pre>var panel = ui.Panel({style: {width:'5%'}}); ui.root.insert(0,panel);</pre>	<p>Змінна panel за допомогою ui.Panel створює блок з розміром width:'5%' від вашого розміру екрана ui.root.insert вставляє віджет у кореневу панель за вказаним індексом</p>
--	--

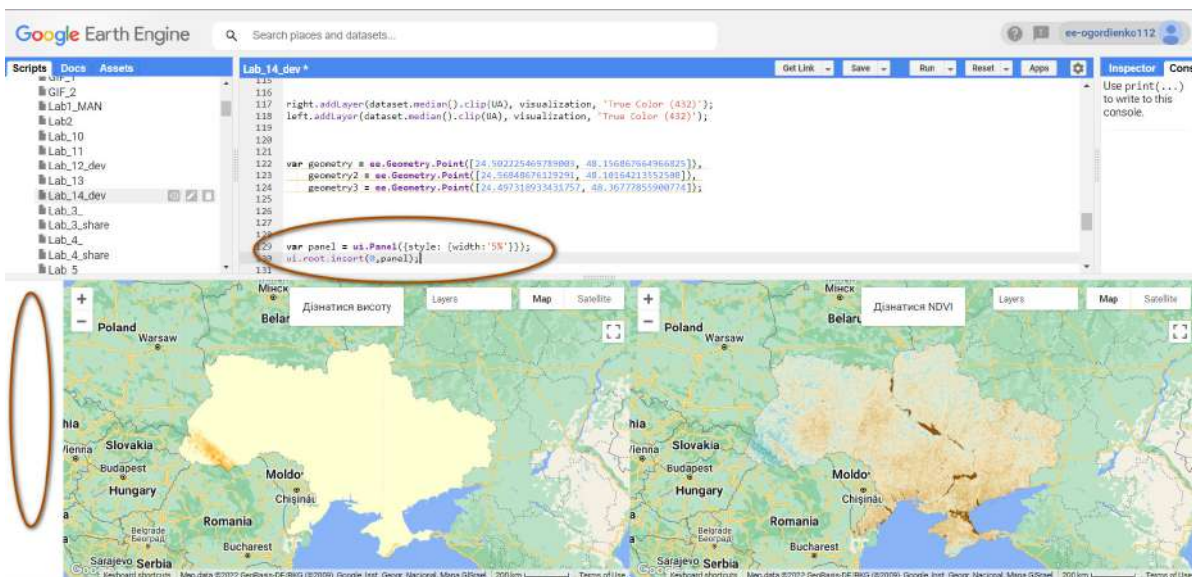


Рис. 288. Код для налаштування розміру панелі

Наступний код створить кнопки для кожної гори і під час натискання на кнопку буде призумлюватися до потрібної гори.

<pre> var button_Brebeneskul = ui.Button('Бребе- нескул', Brebeneskul_co); panel.add(button_Brebeneskul); function Brebeneskul_co(){ left.centerObject(geometry2,12); } var button_Hoverla = ui.Button('Товерла', Hoverla_co); panel.add(button_Hoverla); function Hoverla_co(){ left.centerObject(geometry,12); } var button_Khomiak = ui.Button('Хом'як", Khomiak_co); panel.add(button_Khomiak); function Khomiak_co(){ left.centerObject(geometry3,13); } </pre>	<p>Змінні button_Brebeneskul, button_Hoverla, button_Khomiak за допомогою ui.Button створюють кнопки з відповідними назвами та додають їх до створеного блоку. Функції Brebeneskul_co, Hoverla_co, Khomiak_co центрують мапу за відповідними координатами</p>
--	--

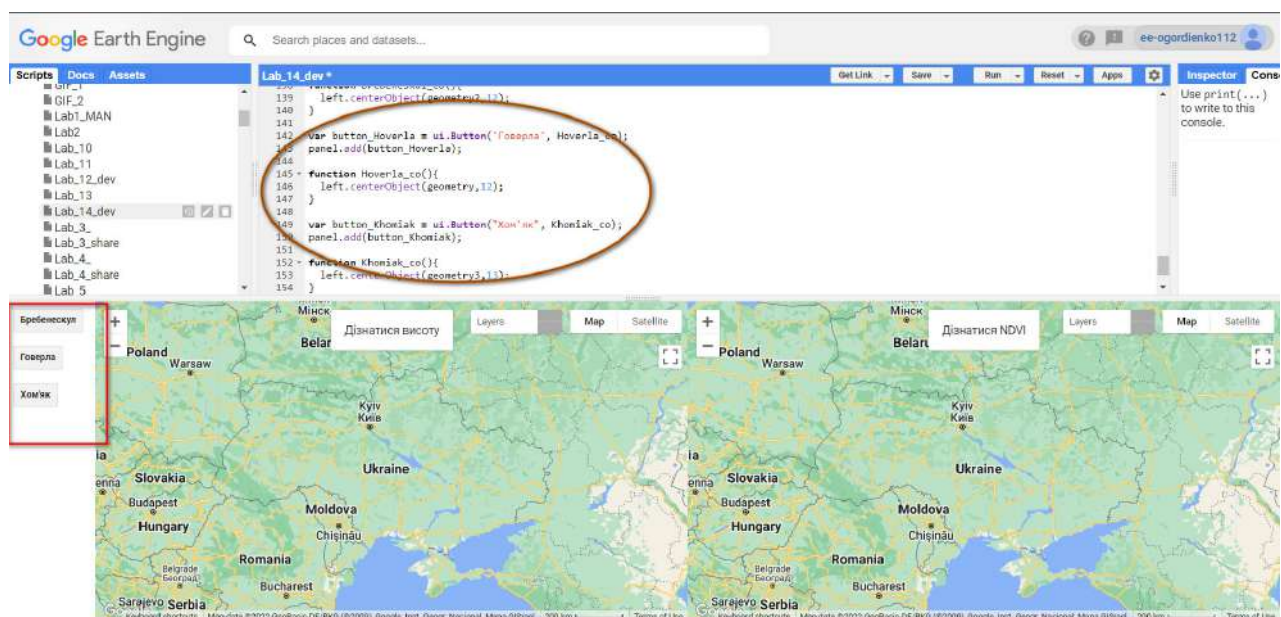


Рис. 289. Код для відображення кнопок

Спробуйте натиснути на Барбенескул – обидві мапи призумляться до потрібної гори.

Натисніть на верхівку гори, користуючись зображенням у натуральних кольорах спочатку в лівому вікні, а потім у правому. Так ви дізнаєтеся значення середнього індексу рослинності та значення висоти.

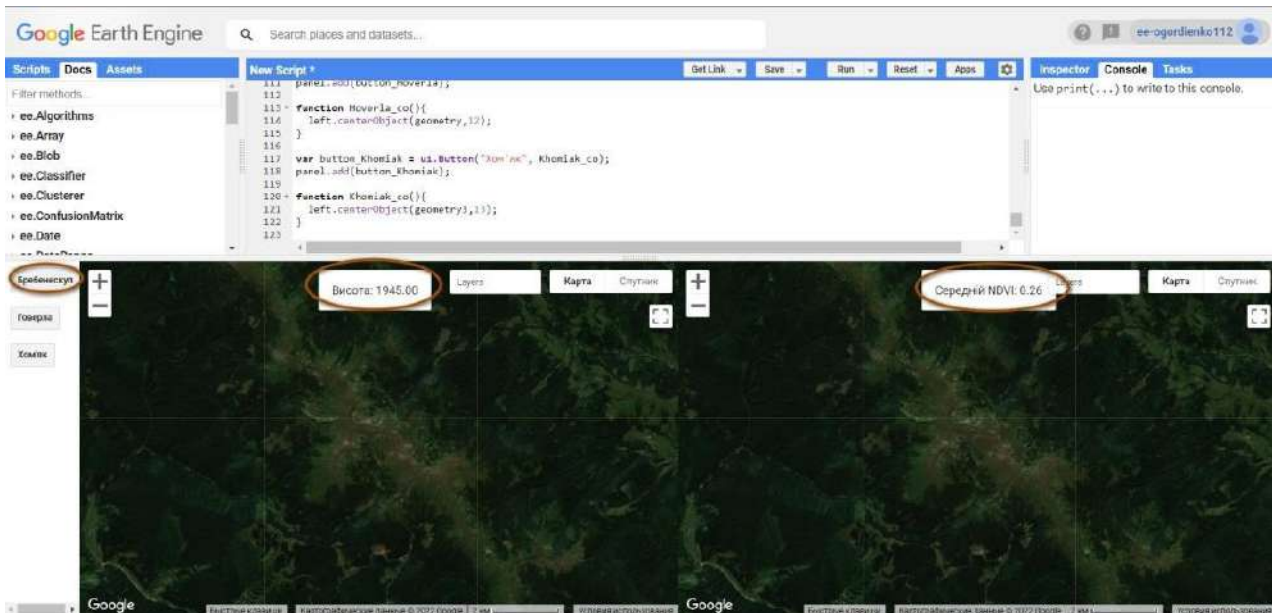


Рис. 290. Кнопка з висотою

Збережіть створений код за допомогою кнопки *Save (Зберегти)* – *Save as (Зберегти як)*. Дайте йому назву *Lab_14_JASU* або якусь іншу довільну назву.

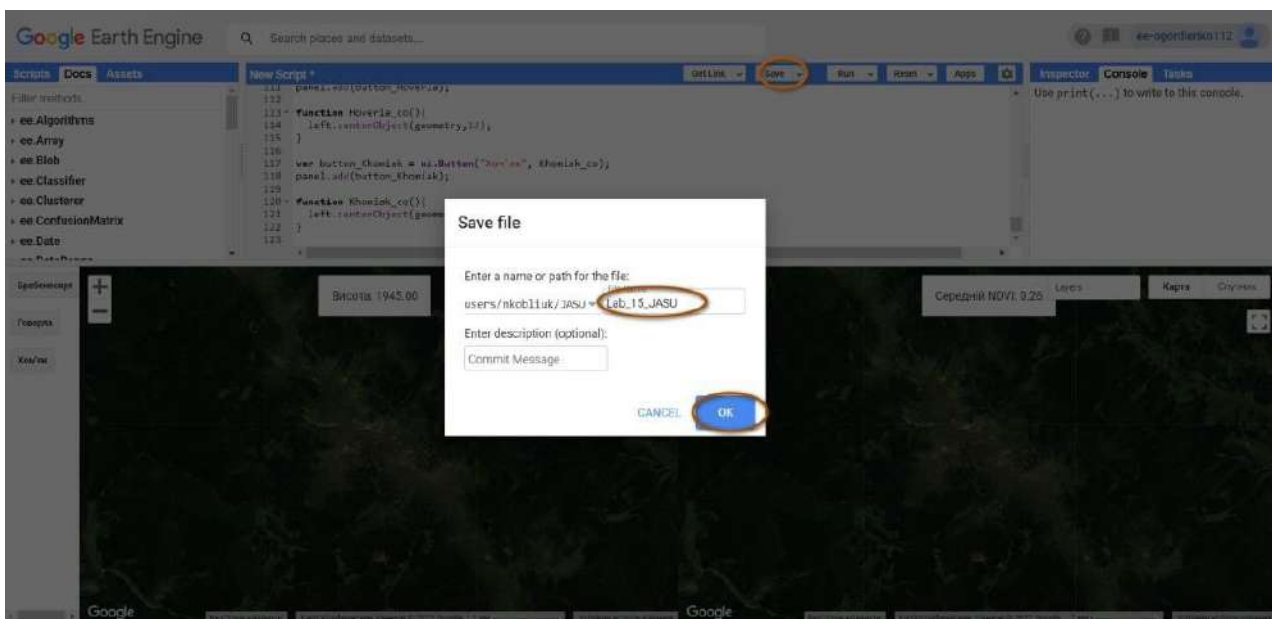


Рис. 291. Назва застосунку

10. Створити інтерактивний застосунок і дослідити його.

Наступним кроком буде вивантаження інтерактивного застосунку. Для цього існує кнопка *Apps (Застосунки)*. Скористайтеся нею.

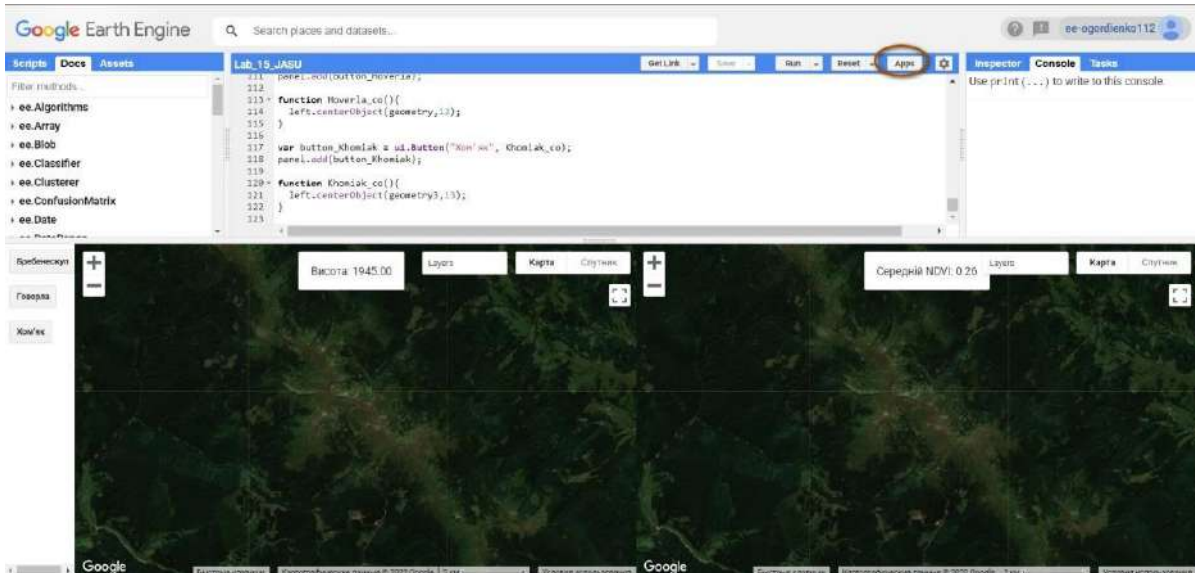


Рис. 292. Знаходження кнопки *Застосунок*

У новому вікні виберіть *New app* (Новий застосунок).

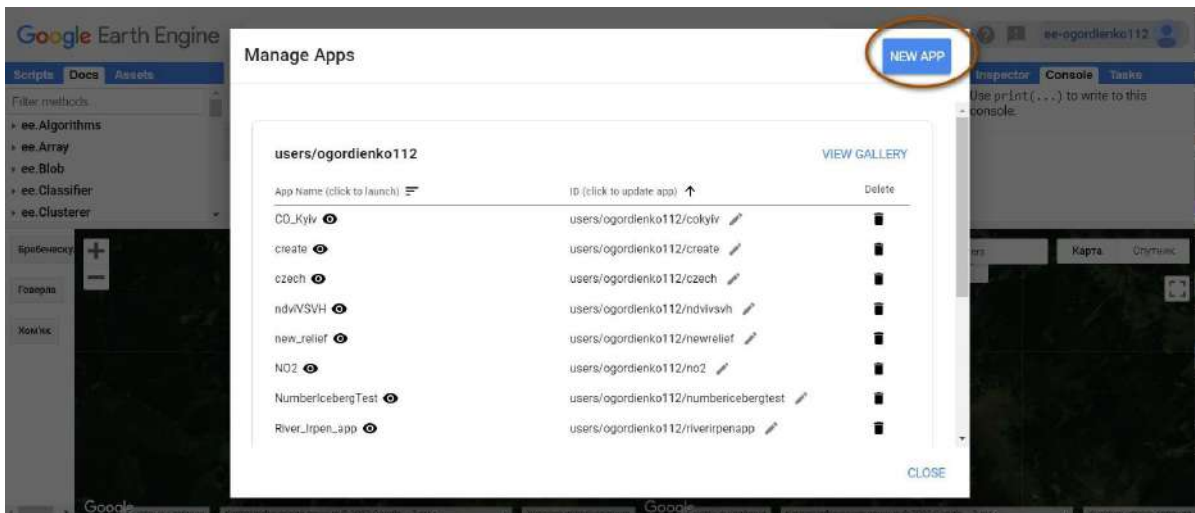


Рис. 293. Створення застосунку

У рядку *App Name* (Назва застосунку) задайте назву застосунку Lab_14_JASU, де замість JASU буде ваше прізвище. Натисніть *Next* (Далі).

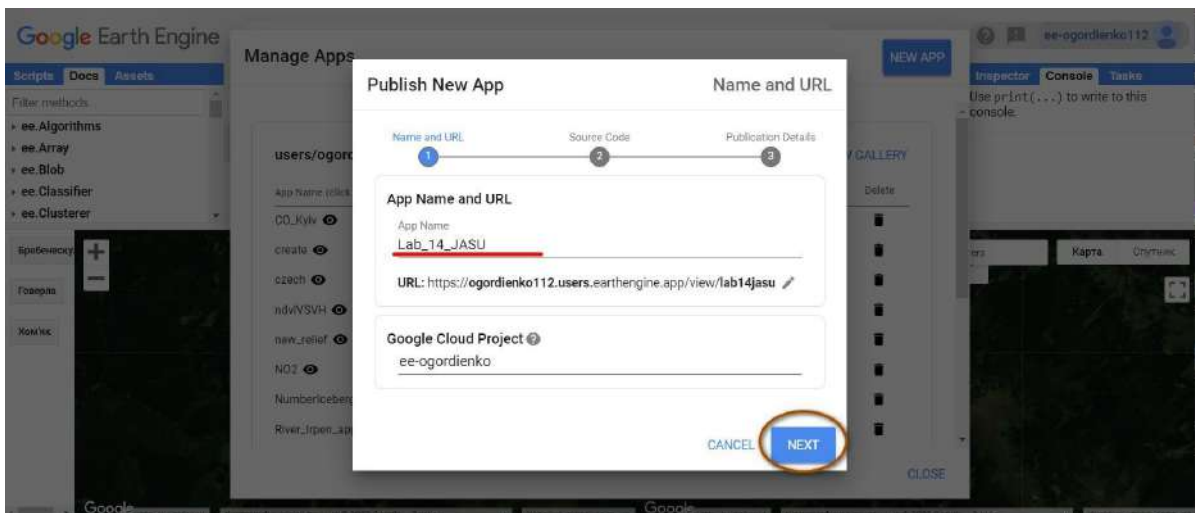


Рис. 294. Створення застосунку

✔ **Важливо!** Якщо ви уперше створюєте застосунок, вам потрібно створити новий *Google Cloud Project* (Проект *Google Cloud*).

Якщо у вас вже є проекти *Google Cloud*, пропустіть цей крок.

Натисніть на *Google Cloud Project* і в новому вікні задайте унікальне ім'я вашого проекту.

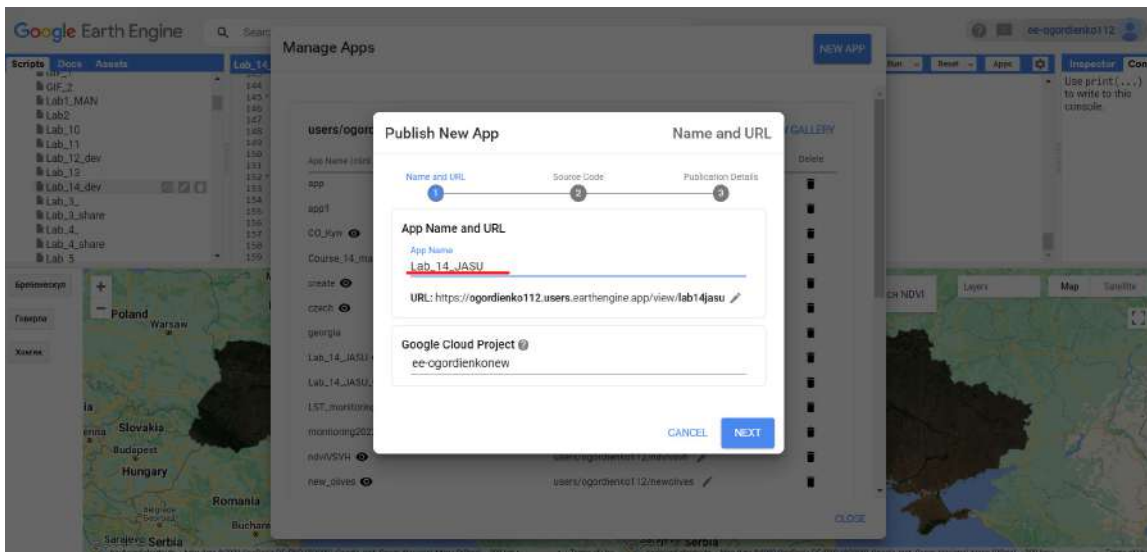


Рис. 295. Назва застосунку

Воно має містити латиницю, цифри і тільки тире.

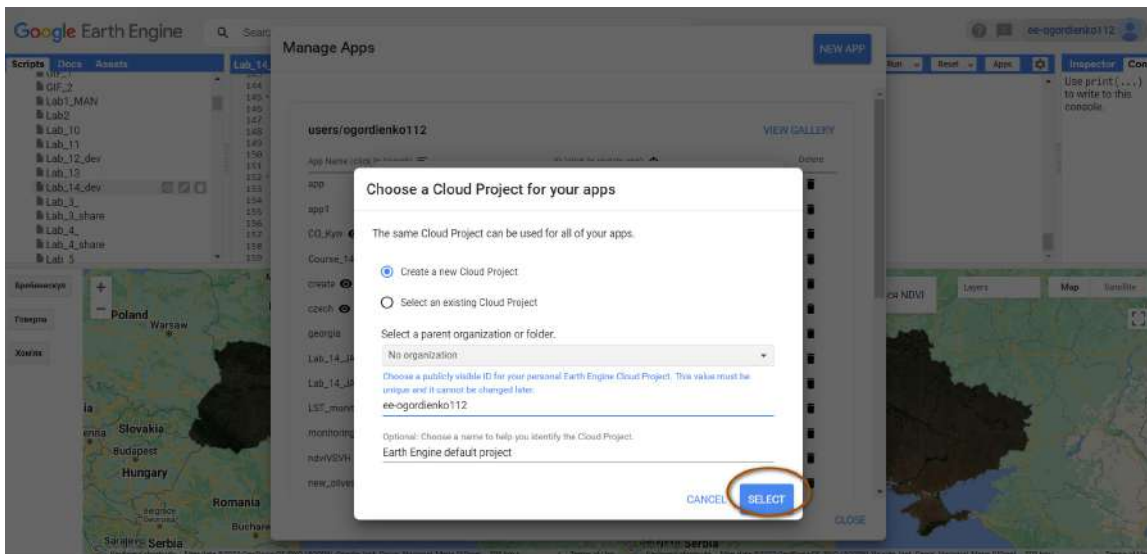


Рис. 296. Вибір репозитарію для застосунку

Натисніть *Select* (Вибрати).

Поверніться до попереднього кроку, де потрібно задавати назву

У наступній вкладці виберіть *Current contents of editor* (Поточний вміст редактора). Натисніть *Next* (Далі).

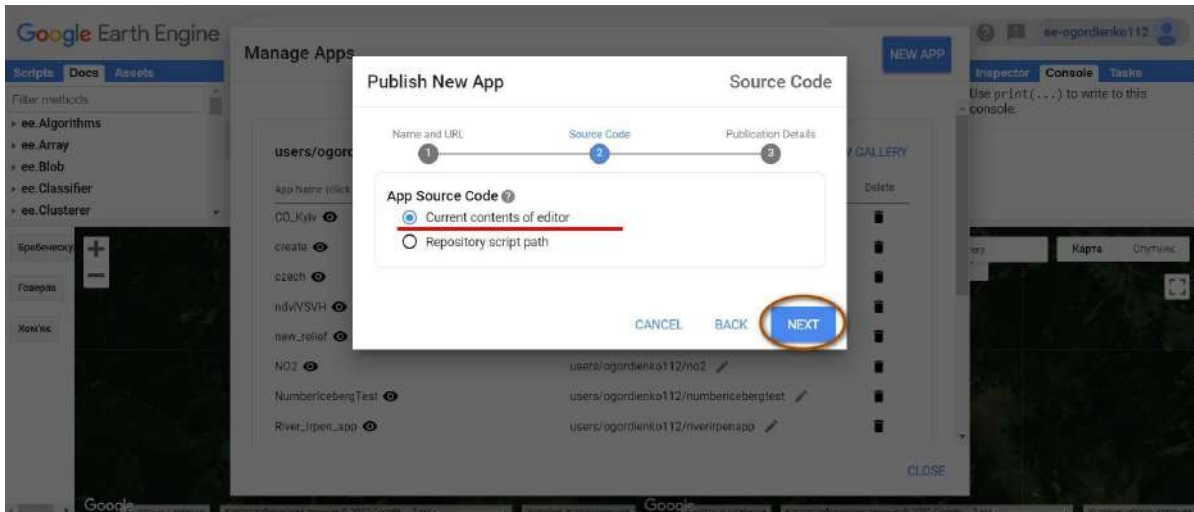


Рис. 297. Місцезнаходження інтерактивного застосунку

У наступному діалоговому вікні відмітьте **Feature this app in Public Apps Gallery** (Розмістити цю програму в галереї публічних програм). Натисніть **Publish** (Опублікувати).

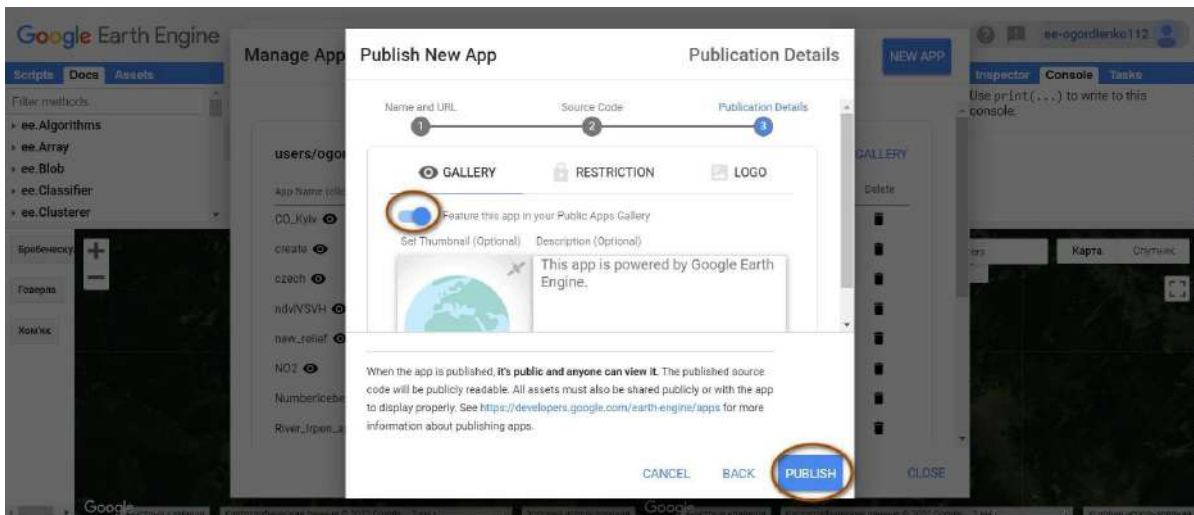


Рис. 298. Налаштування доступу застосунку

✔ **Важливо!** Проект створюється деякий час, тому потрібно трохи почекати.

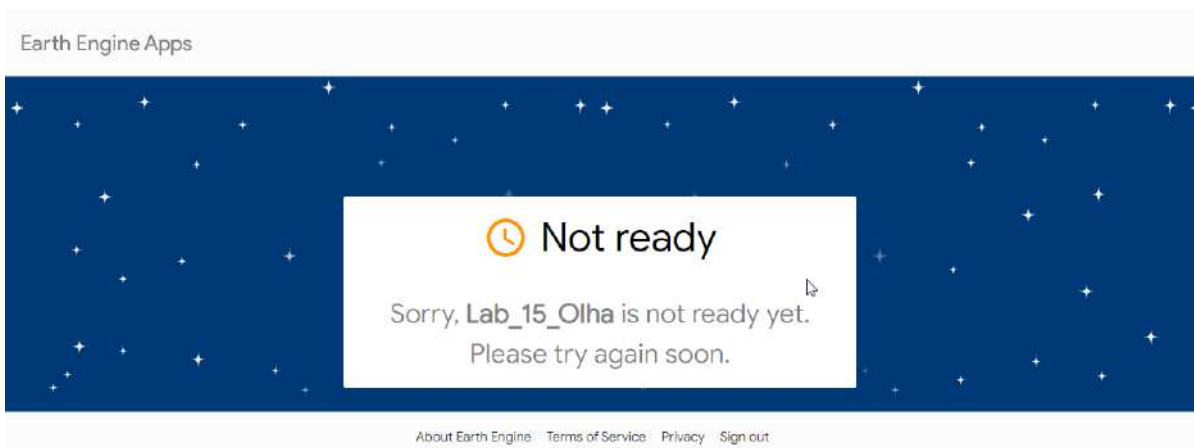


Рис. 299. Можлива помилка відображення

У **Manage Apps** (Менеджер застосунків) з'явиться новий застосунок із назвою, яку ви задали. Натисніть на нього.

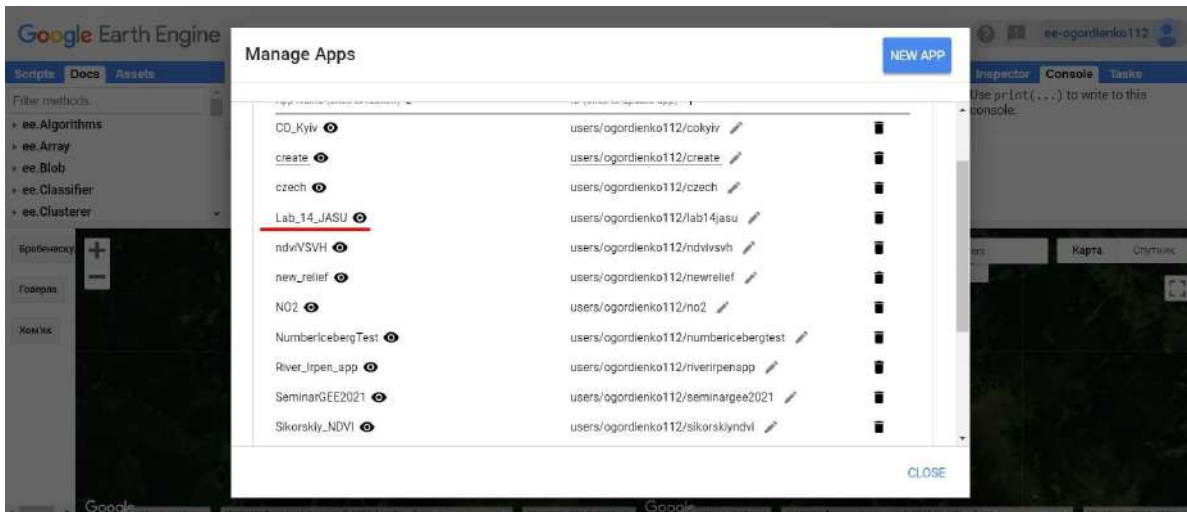


Рис. 300. Місцезнаходження інтерактивного застосунку

У новій вкладці відкривається готовий інтерактивний застосунок без редактора коду, яким так само можна користуватися. Дослідіть його.

Також у цьому застосунку можна увімкнути шар з рельєфом. Потрібно вимкнути всі шари в одному з вікон і під назвою **Map (Mapa)** знайти кнопку, яка вмикає рельєф.

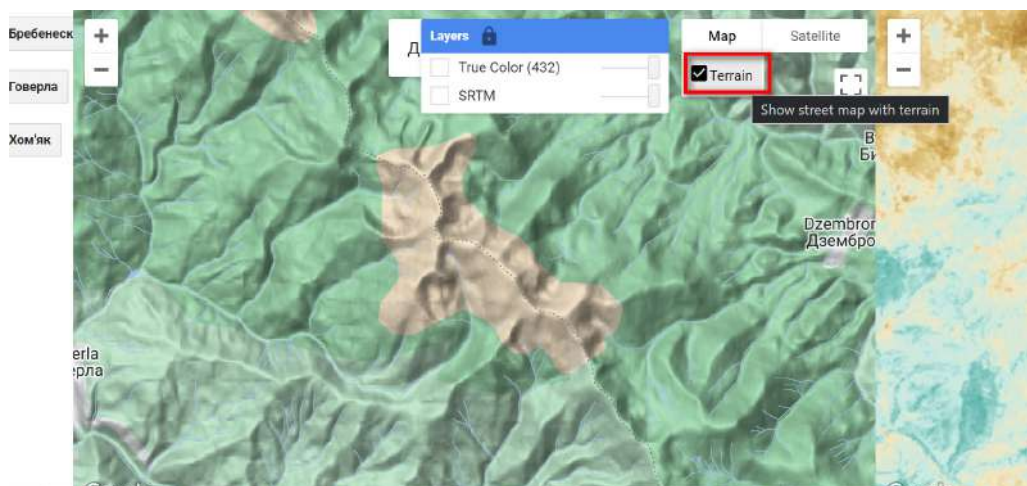


Рис. 301. Налаштування відображення ландшафту

? Завдання для перевірки

Користуючись створеним застосунком, дослідіть гори Бребенескул, Говерла, Хом'як у Карпатах. Яку вони мають висоту та середній індекс NDVI?

За наведеним нижче QR-кодом ви можете переглянути відео виконання практичної роботи.



Рис. 302. Відео виконання практичної роботи

Готовий код можна відкрити за лінком:

https://code.earthengine.google.com/?scriptPath=users%2Fgis_rs%2Fcourse_MAN%3ALab_14

Орієнтовна тематика науково-дослідницьких робіт учнів – членів МАНУ

Гідрологія

Евтрофування водойм / «цвітіння» води (Дніпровських водосховищ).

Моніторинг повеней на річках внаслідок інтенсивних опадів (на заході України, в Румунії).

Моніторинг затоплення території під час паводка або весняного водопілля (річки Дніпро, Десна).

Динаміка дельтоутворення чи зростання дельт річок (Дунаю, Гангу, Нілу).

Заболочування / заростання водосховищ, озер (Дніпровських водосховищ).

Моніторинг льодової обстановки (скресання льоду на великих суднохідних річках, водосховищах).

Танення льодовиків (Арктики, Антарктики, Гренландії).

Моніторинг айсбергів.

Ландшафтознавство / урбаністика

Зростання міст або розбудова / розширення (міста Париж, Київ, Варшава).

Зміна берегової лінії (Дубай, Одеса).

Висихання озер, морів / обміління річок (Аральське море).

Зростання площ земельних угідь, зайнятих під будівництво сонячних електростанцій (найбільша в Україні – в Дніпропетровській обл., площею 400 га).

Лісознавство

Негативні процеси, що впливають на лісові масиви: вплив шкідників і хвороб, осушення або перезволоження лісів, які призводять до їх деградації і загибелі (ураження сосновим ко-роїдом хвойних лісів півночі України).

Динаміка лісових пожеж (Чорнобильська зона в Україні, Австралія, Каліфорнія).

Виявлення і моніторинг вирубок лісу (будь-яка територія лісового фонду України).

Насадження лісів / лісовідновлення (Олешківські піски – українська пустеля).

Агрономія

Порівняння типів сільськогосподарського господарювання в різних країнах (Ірак, Німеччина).

Меліоративні заходи (зрошування), побудова греблі (Іран).

Моніторинг процесів деградації земельних ресурсів (ерозія, засолення, заболочування, заростання сільськогосподарських угідь чагарниками).

Картографування сівозмін, визначення реальної структури посівних площ.

Охорона довкілля

Виявлення стихійних сміттєзвалищ (у межах одного адміністративного району будь-якої області).

Збереження природно-заповідного фонду.

Спалювання сухої трави, плавнів на території об'єктів природно-заповідного фонду (нинішні пожежі навесні в Дунайських плавнях).

Виявлення незаконних вирубок лісу

Виявлення незаконного добування бурштину.

Наслідки надзвичайних ситуацій

Цунамі (26 грудня 2004 р., спричинене землетрусом в Індійському океані, постраждали жителі Індонезії, Шрі-Ланки, півдня Індії, Таїланду).

Виверження вулканів (2010 р., вулкан Ейяф'ятлайокутль).

Зсуви, обвали, осипи, карстові процеси, абразія.

Сходження лавин (23.02.1999 р. – Galtur, Австрія).

Паводок (06.05.2018 р. – Анкара, Туреччина).

Торнадо (20.05.2013 р. – штат Оклахома, США).

Землетруси (27.02.2010 р. – Чилі, Пд. Америка).

Великі терористичні акти (11.09.2001 р. – Нью-Йорк).

Катастрофи й аварії вантажних потягів, суден, авіа- і космічні катастрофи, аварії на нафтопроводах, бурових платформах, викиди на нафтових і газових родовищах (вибухи і пожежі на військових складах в Україні 26.09.2017 р. – Калинівка, Калинівський район, Вінницька область).

Розлив річки Ірпінь – березень 2022 р.

Кліматологія, метеорологія

Теплові аномалії земної поверхні (виверження вулканів).

Формування циклонів, тайфунів, буревіїв.

Пилові бурі, дефляція (в пустелі Сахара, в м. Сідней на сході Австралії).

Забруднення атмосферного повітря (знаходження найбільших забруднювачів повітря і порівняння динаміки за місяцями).

Список рекомендованої літератури для педагогів

1. Community Tutorials. URL: <https://developers.google.com/earth-engine/tutorials/community/explore> (дата звернення: 22.10.2022).
2. Kumar L., Mutanga O. Applications. Remote Sensing. 2019. URL: <https://www.mdpi.com/books/book/1262-google-earth-engine-applications> (дата звернення: 22.10.2022).
3. Cloud-Based Remote Sensing with Google Earth Engine. URL: <https://www.eefabook.org/go-to-the-book.html> (дата звернення: 22.10.2022).

Навчальне видання

**Робочий зошит
з основ дистанційного
зондування Землі**

Частина 3

**Обробка та аналіз супутникових знімків
на платформі Google Earth Engine**

Коректура *І. В. Браташук*
Верстання *О. А. Жупанська*
Дизайн обкладинки *Б. Л. Лісовський*

Формат 60×84 1/8. Папір офс. 80 г/м².
Друк цифровий. Ум. друк. арк. 23,25.
Наклад 300 прим.

Видавництво: Національний центр «Мала академія наук України»,
Кловський узвіз, буд. 8, м. Київ, 01021

Свідоцтво суб'єкта видавничої справи:
ДК № 6999 від 04.12.2019

