

**BỘ GIÁO DỤC VÀ ĐÀO TẠO
TRƯỜNG ĐẠI HỌC MỎ - ĐỊA CHẤT**

**BÁO CÁO SINH HOẠT HỌC THUẬT
ỨNG DỤNG CÔNG NGHỆ ĐIỆN TOÁN Đám Mây GOOLE EARTH
ENGINE**

**Cán bộ thực hiện: PGS.TS. Nguyễn Thị Nụ
Đơn vị: Bộ môn Địa chất công trình – Khoa KH& KT Địa chất**

Hà Nội, năm 2024

**BỘ GIÁO DỤC VÀ ĐÀO TẠO
TRƯỜNG ĐẠI HỌC MỎ - ĐỊA CHẤT**

**BÁO CÁO SINH HOẠT HỌC THUẬT
ỨNG DỤNG CÔNG NGHỆ ĐIỆN TOÁN ĐÁM MÂY GOOLE EARTH
ENGINE**

Thực hiện

Bộ môn Địa chất công trình

PGS.TS. Nguyễn Thị Nụ

Hà Nội, năm 2024

MỤC LỤC

1. GIỚI THIỆU GEE.....	4
2. HƯỚNG DẪN SỬ DỤNG GEE.....	4
2.1. Đăng ký	4
2.2. Viết và chạy tập lệnh đầu tiên của bạn	7
2.3. Viết và chạy tập lệnh.....	7
2.4. Bản đồ độ cao.....	8
2.4. Siêu dữ liệu (Metadata).....	10
2.5. Trực quan hóa	11
2.6. Lọc	12
2.8. Độ dốc	15
2.9. Xuất ảnh	16
2.10. Xuất hình ảnh dưới dạng PNG hoặc JPG	17
2.11. Tải lên một shapefile.....	18
2.11. Height Above Nearest Drainage (HAND).....	20
3. ỨNG DỤNG CỦA GEE	21
3.1. Xác định lượng mưa toàn cầu	21
3.2. Xác định phạm vi của một lưu vực	25
3.3. Xác định lượng mưa năm trên lưu vực Sông Hồng.....	26
3.3. Xác định chỉ số thực vật.....	30
3.4. Xác định lượng bốc hơi.....	35
3.5. Phân tích hạn hán với SPI.....	39
3.6. Hình thái sông ngòi	43
3.7. Xác định sự thay đổi hình thái sông ngòi	44
3.9. Bản đồ thảm phủ	2
3.10. Điều kiện loại đất	4
V. KẾT LUẬN	13

1. GIỚI THIỆU GEE

Google Earth Engine (GEE) là một nền tảng điện toán đám mây được phát triển để xử lý ảnh vệ tinh và các dữ liệu địa không gian khác. Nó cung cấp quyền truy cập vào cơ sở dữ liệu khổng lồ ảnh vệ tinh và các thuật toán cần thiết để phân tích ảnh vệ tinh. GEE cho phép giám sát những thay đổi trong các lĩnh vực nông nghiệp, tài nguyên nước và khí hậu... sử dụng dữ liệu địa không gian với các mức độ phân giải theo không gian và thời gian khác nhau. Nó cung cấp một danh mục dữ liệu cùng với thuật toán phân tích, cho phép các nhóm người dùng khác nhau, chẳng hạn như các nhà nghiên cứu và chuyên gia tài nguyên môi trường cộng tác bằng cách sử dụng dữ liệu, thuật toán và minh họa bằng hình ảnh.

Việc sử dụng GEE đã tăng nhanh trong vài năm qua, trong nhiều cơ quan (nghiên cứu, chính phủ, và tư nhân) cũng như trong nhiều lĩnh vực ứng dụng khác nhau (quản lý nước, nông nghiệp, bảo tồn thiên nhiên, v.v.). Sự phổ biến của công cụ có liên quan đến những lợi ích to lớn của nó so với các công nghệ thông thường khác, bao gồm những điều sau:

Xử lý ảnh được thực hiện trên đám mây, có nghĩa là bạn không cần tải hàng gigabyte dữ liệu xuống PC của mình nữa. Điều này giúp tiết kiệm dung lượng ổ cứng, chi phí cơ sở hạ tầng và khắc phục các hạn chế liên quan đến tốc độ internet;

Phần mềm xử lý ảnh viễn thám đặc biệt không còn cần thiết, giúp tiết kiệm chi phí cấp phép phần mềm;

Tốc độ xử lý cao và tiết kiệm thời gian làm việc quý báu;

Cung cấp quyền truy cập vào các bộ dữ liệu khổng lồ không chỉ ở quy mô quốc gia mà còn ở quy mô toàn cầu, cho phép thực hiện các phân tích xuyên biên giới (cần thiết cho các nghiên cứu môi trường!);

Cung cấp quyền truy cập vào hàng thập kỷ dữ liệu lịch sử cho phép phân tích chuỗi thời gian;

Cung cấp quyền truy cập vào các tập dữ liệu vệ tinh đa cảm biến đã được xử lý trước, cho phép dễ dàng tích hợp nhiều nguồn dữ liệu khác nhau mà không tốn thời gian sửa/tiền xử lý;

Cho phép bạn chia sẻ công việc và kiến thức của mình với các chuyên gia khác cũng như với công chúng.

2. HƯỚNG DẪN SỬ DỤNG GEE

2.1. Đăng ký

Việc sử dụng GEE yêu cầu phải có tài khoản Google. Đăng ký GEE có thể được thực hiện tại đây:

<https://earthengine.google.com/signup/>

Nếu bạn đã đăng ký, bạn có thể chuyển sang ("Đăng nhập").

Để đăng ký, trước tiên bạn cần nhập địa chỉ email và mật khẩu Google của mình. Sau đó, bạn sẽ cần điền và gửi biểu mẫu bao gồm tên, đơn vị/tổ chức công tác, quốc gia của bạn và một câu giải thích bạn muốn sử dụng Earth Engine

cho mục đích gì. Bạn cũng sẽ phải chấp nhận các Điều khoản và Điều kiện. Một số Điều khoản dịch vụ quan trọng cần xem xét là:

- GEE được phép sử dụng trong nghiên cứu, phát triển và giáo dục. Nó cũng có thể được sử dụng để đánh giá hoạt động thương mại nhưng không được sử dụng cho mục đích sản xuất bền vững.

- Không đảm bảo rằng GEE sẽ duy trì truy cập miễn phí.
- Công ty Google có thể truy cập các câu lệnh/code của bạn.

Nếu quá trình đăng ký của bạn diễn ra tốt đẹp, bạn sẽ nhận được email xác nhận sau vài phút, hiếm hơn là sau vài ngày.

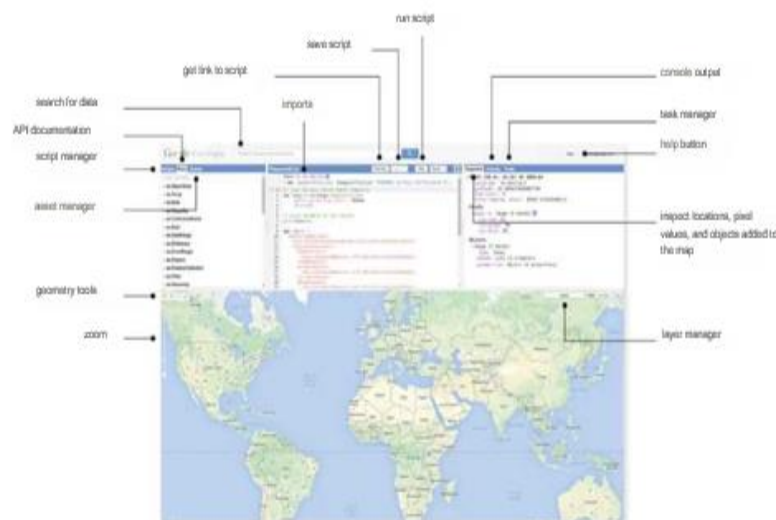
Đăng nhập

Đăng nhập và làm việc thông qua Trình soạn thảo mã của GEE. Đây là nơi bạn có thể phát triển các tập lệnh của riêng mình để phân tích vô số dữ liệu có sẵn. Trình soạn thảo mã có thể được truy cập thông qua liên kết này:

<https://code.earthengine.google.com/>

Trình soạn thảo mã

Bố cục của Trình soạn thảo mã và vị trí của các chức năng cơ bản được hiển thị trong hình bên dưới. Chúng ta sẽ thảo luận về những yếu tố này trong suốt các bài tập tiếp theo.

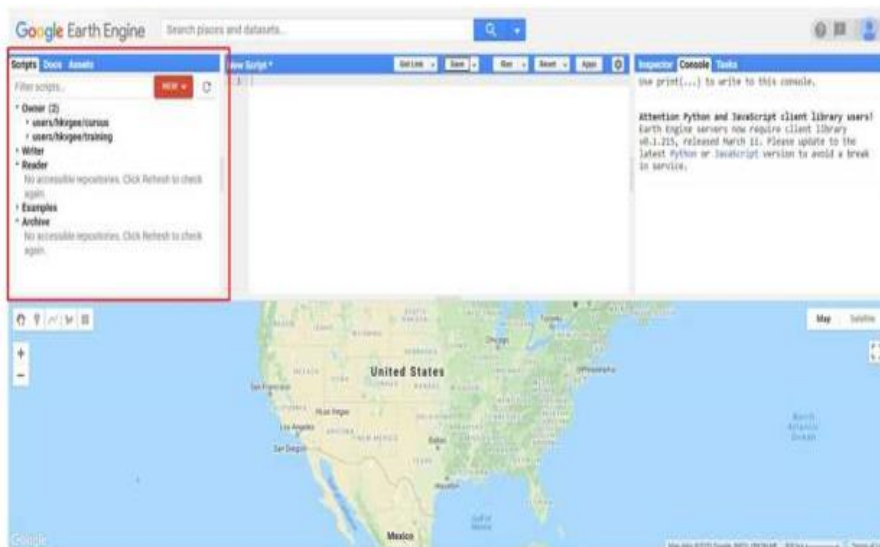


Tạo tập lệnh đầu tiên của bạn

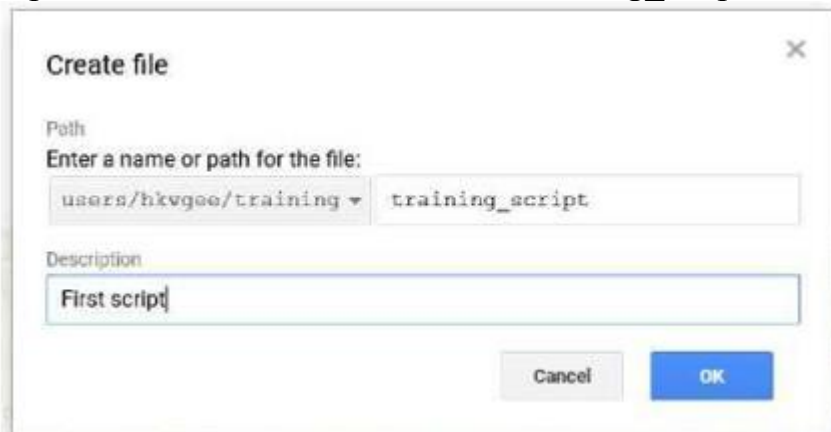
Ở phía trên bên trái màn hình của bạn có một khối với ba tab: Scripts (Tập lệnh), Docs (Tài liệu) và Assets (Dữ liệu) được khoanh trong hình chữ nhật màu đỏ của hình dưới. Đây là trình quản lý tập lệnh, tài liệu API và trình quản lý dữ liệu.

Bấm vào tab Scripts để chọn Trình quản lý tập lệnh. Bạn có thể tạo một tệp mới bằng cách nhấp vào nút màu đỏ

Nếu là lần đầu tiên bạn làm chuyện ấy, bạn sẽ được yêu cầu tạo thư mục chủ của mình. Lưu ý đặt tên của thư mục này một cách cẩn thận, vì nó sẽ hiển thị với những người dùng khác khi bạn chia sẻ công việc của mình và bạn sẽ không thể thay đổi tên sau khi đã tạo.



Tương tự, bạn có thể tạo các thư mục và tệp mới. Tạo một tệp mới bằng cách nhấp vào nút màu đỏ bạn, ví dụ như 'training_script':



Bạn có thể đổi tên các tập lệnh bằng nút **Q** và xóa chúng bằng nút

Theo cách tương tự, bạn có thể tạo các thư mục và kho lưu trữ mới. Bạn có thể di chuyển các tập lệnh và sắp xếp chúng vào các thư mục bằng cách sử dụng chức năng kéo và thả. Nếu bạn kéo tập lệnh sang một kho lưu trữ khác, tập lệnh đó sẽ được sao chép.

Trình quản lý tập lệnh có một số phần phụ, bao gồm cả 'Example'. Tại đây, một danh sách phức tạp gồm các tập lệnh mẫu được cung cấp, tất cả đều đã sẵn sàng để chạy. Bất cứ khi nào bạn muốn phát triển một tập lệnh GEE mới, có thể sẽ hữu ích khi kiểm tra một số ví dụ trước để xem đã có sẵn những gì.

Tài liệu API

Trong tab Docs, bạn có thể tìm thấy tất cả các phương pháp (hàm) có sẵn để xử lý dữ liệu. Bạn có thể sẽ thường sử dụng một số hàm trong ee.Image, chứa tất cả các thao tác có thể được thực hiện trên ảnh. Một ví dụ khác là ee.Reducer, chứa một số chức năng hữu ích để trích xuất số liệu thống kê từ ảnh. Nhấp vào một phương thức bất kỳ sẽ cung cấp cho bạn mô tả về chức năng và cú pháp phù hợp của nó.

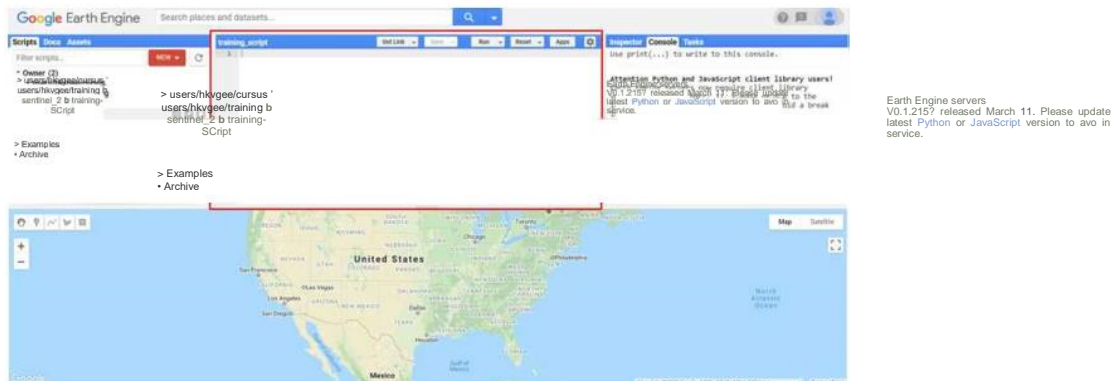
Trình quản lý dữ liệu

Cuối cùng, trong Assets, bạn có thể quản lý và tải lên các bảng và tập dữ liệu địa không gian. Những 'tài sản' này ban đầu là riêng tư nhưng có thể được chia sẻ với những người khác.

2.2. Viết và chạy tập lệnh đầu tiên của bạn

Bây giờ bạn sẽ viết và chạy tập lệnh đầu tiên của mình.

Bạn có thể mở tệp (ví dụ: 'training_script') mà chúng ta đã tạo ở bước trước bằng cách chọn nó trong Trình quản lý tập lệnh ở bên trái. Nó sẽ được mở trong Trình soạn thảo mã Code Editor, là khu vực được khoanh đỏ trong hình bên dưới. Hiện nó vẫn đang trống, nhưng chúng ta sẽ viết nội dung vào nó sau này.



Bạn có thể lưu tập lệnh bằng cách nhấn nút Save trong Trình soạn thảo mã. Điều quan trọng là phải lưu tập lệnh của bạn thường xuyên trong khi bạn đang viết! GEE thêm dấu * vào sau tên tập lệnh nếu có nội dung chưa được lưu trong tập lệnh của bạn. Sau khi bạn đã viết những dòng mã đầu tiên của mình như sẽ được giải thích trong các phần tiếp theo, bạn có thể sử dụng nút 'Run' trong Trình soạn thảo mã để GEE thực thi mã của bạn.

JavaScript

GEE sử dụng ngôn ngữ lập trình JavaScript. Nếu bạn chưa từng lập trình trước đây, đừng lo lắng. Hướng dẫn này sẽ giải thích cho bạn cú pháp của JavaScript và các chức năng của nó.

Nếu bạn đã quen với lập trình, hãy lưu ý rằng những điều sau đây có thể sẽ khác với những điều bạn đã biết:

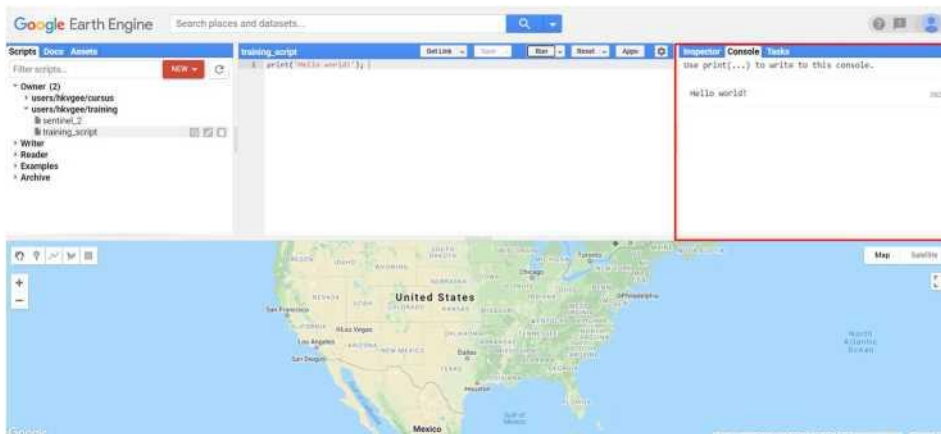
- Để định danh một biến mới, bạn phải sử dụng từ khóa var.
- Mỗi cú pháp trong JavaScript đều kết thúc bằng dấu chấm phẩy (;).
- Bạn phải chạy toàn bộ tập lệnh của mình cùng một lúc bằng JavaScript. Bạn không thể chọn, ví dụ, hai dòng lệnh của mình và chỉ chạy chúng.

- Chỉ mục của lists và arrays bắt đầu từ 0 thay vì 1. Ví dụ, nếu bạn muốn gọi phần tử đầu tiên của một list, bạn sẽ phải dùng lệnh list[0] thay vì list[1].

2.3. Viết và chạy tập lệnh

Nhấp 'Save' và 'Run'.

Hình chữ nhật màu đỏ bên dưới biểu thị Bảng điều khiển (Console). Nó hiển thị tất cả đầu ra từ tập lệnh sau khi tập lệnh được chạy. Trong trường hợp này là: 'Hello World!'.



Lưu và chạy lại tập lệnh. Bây giờ bạn sẽ thấy rằng không có gì được in ra trong Bảng điều khiển. Điều này là do bất kỳ dòng lệnh nào bắt đầu bằng hai dấu gạch chéo (//) đều sẽ bị bỏ qua trong khi chạy mã. Bạn có thể sử dụng chức năng này để viết ghi chú cho chính mình để có thể hiểu mã sau này. Phím tắt cho lệnh này là CTRL + /. Để bỏ ghi chú một dòng, hãy nhấn lại tổ hợp phím tắt đó. Bạn cũng có thể sử dụng tổ hợp phím tắt này để nhanh chóng nhận xét nhiều dòng mã đã chọn. Điều này có thể hữu ích cho việc gỡ lỗi mã của bạn. Ví dụ:

```
training_script * [Get Link] [Save] [Run] [Reset] [Apps] [Settings]
1 // The next line prints 'Hello world!' to the console.
2 print('Hello World !');
```

Lệnh print có thể được sử dụng để in nhiều loại biến khác nhau như văn bản, số, tham số và tập dữ liệu vào bảng điều khiển.

2.4. Bản đồ độ cao

Bây giờ bạn đã sẵn sàng để sử dụng dữ liệu vệ tinh trong GEE. Trong bài tập này, bạn sẽ xem bản đồ độ cao dựa trên dữ liệu vệ tinh SRTM (Shuttle Radar Topography Mission).

Bước 1. Đầu tiên hãy tạo một tập lệnh mới và đặt tên là 'Elevation'.

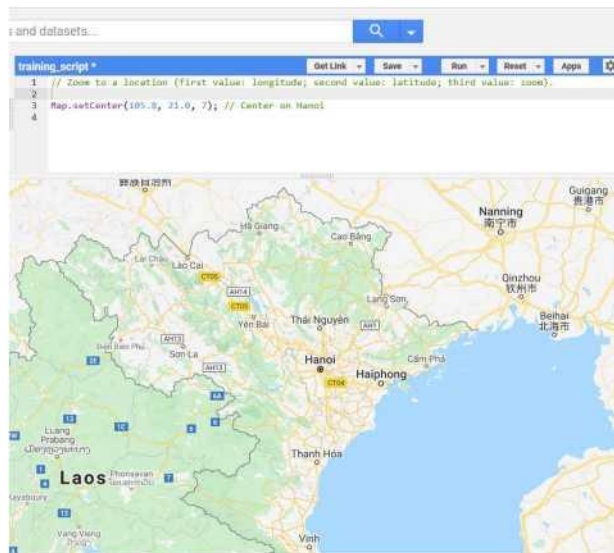
Căn giữa bản đồ

Bạn có thể sẽ nhận thấy chế độ xem bản đồ ở nửa dưới của cửa sổ. Rất có thể, trong màn hình của bạn, nó hiện đang ở Hoa Kỳ. Chúng ta có thể căn nó vào một vị trí khác theo cách sau:

Bước 2. Nhập cú pháp sau vào tập lệnh và nhấp vào Run.

```
// Zoom to a location (first value: longitude; second value: latitude; third value: zoom).
Map.setCenter(105.8, 21.0, 7); // Center on Hanoi
```

Lưu và chạy tập lệnh để xem điều gì sẽ xảy ra. Bản đồ sẽ được căn về Hà Nội:



Thêm dữ liệu vệ tinh SRTM vào bản đồ

Bạn có thể sử dụng dữ liệu vệ tinh SRTM để xem bản đồ độ cao.

Bước 3. Để tải dữ liệu, bạn thêm cú pháp sau vào tập lệnh và nhấp vào

Run:

```
// Define an image using the Image constructor. var image = ee.Image('CGIAR/SRTM90 V4');
```

Để hiểu điều gì xảy ra trong dòng lệnh này, hãy cùng khám phá từng bước, bắt đầu từ phần cuối:

```
var image = ee.Image('CGIAR/SRTM90 V4');
```

Phần mã này tải dữ liệu vào GEE. Bạn có thể thay thế tên giữa dấu ngoặc đơn (‘) bằng tên của một tập dữ liệu khác để chọn các dữ liệu khác nhau. Bạn sẽ làm điều đó trong các bài tập sau.

```
var image = ee.Image('CGIAR/SRTM90 V4');
```

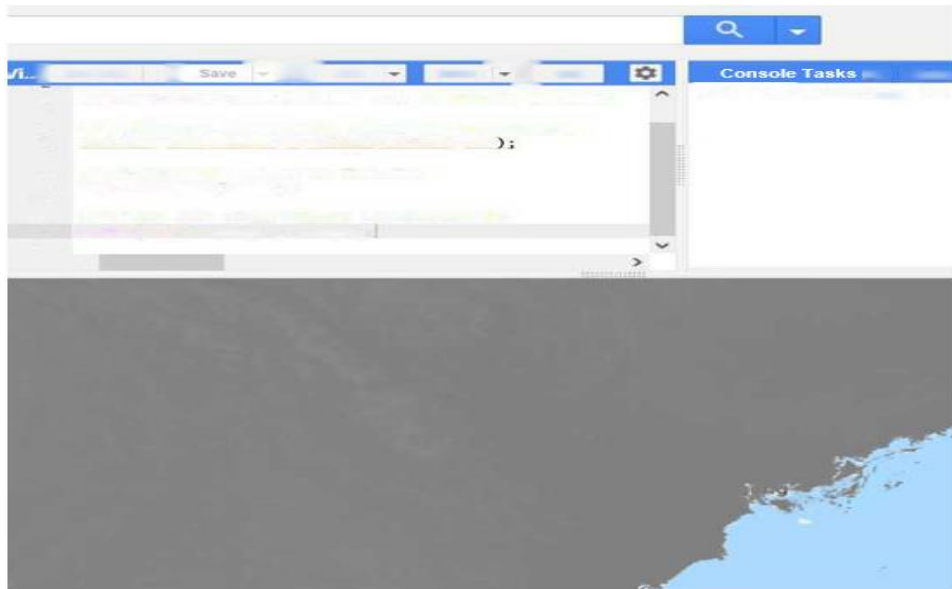
Bạn có thể gán dữ liệu SRTM vào một biến. Để làm điều này, bạn luôn bắt đầu bằng từ var, theo sau là tên của biến, trong trường hợp này là image. Bạn có thể chọn bất kỳ tên nào cho biến của mình.

Bước 4. Bạn có thể hiển thị nó bằng cách thêm nó làm lớp mới trên bản đồ bằng cách thêm cú pháp này vào tập lệnh và nhấp vào ‘Run’:

// Display the image on the map.

```
Map.addLayer(image);
```

Bây giờ hãy lưu và chạy tập lệnh. Những gì bạn thấy, có thể khiến bạn hơi thất vọng: Nó chỉ hoàn toàn là màu xám!



Trong các bài tập tiếp theo, bạn sẽ làm cho bản đồ độ cao sinh động và hữu ích hơn.

Bước 5. Đừng quên lưu tập lệnh của bạn!

Tên giữa dấu ngoặc đơn ở bước 3 ('CGIAR/SRTM90_V4') xác định tập dữ liệu độ cao SRTM. Để biết tên của dữ liệu trong trình soạn thảo mã của GEE. Hãy nhìn vào thanh ở đầu màn hình của bạn:

Go google Earth Engine Search places and datasets.

Nhập SRTM vào thanh này và nhấp vào nút tìm kiếm (hoặc nhấn enter). Bạn sẽ thấy 'SRTM Digital Elevation Data Version 4' trên đầu. Đây là tập dữ liệu mà chúng ta đang tìm kiếm! Nhấp vào tiêu đề và cửa sổ của tập dữ liệu này xuất hiện.

2.4. Siêu dữ liệu (Metadata)

Ảnh vệ tinh thường được tạo thành từ một hoặc nhiều dải quang phổ. Mỗi băng tần (band) quan sát một phần cụ thể của phổ điện từ và có tên riêng, các giá trị pixel và độ phân giải pixel. Như bạn sẽ sớm khám phá ra, hình ảnh SRTM có một band: "độ cao". Nếu một ảnh đơn băng tần được thêm vào bản đồ, theo mặc định, GEE hiển thị band ở thang độ xám, trong đó giá trị nhỏ nhất được gán cho màu đen và giá trị lớn nhất được gán cho màu trắng. Nếu bạn không chỉ định giá trị tối thiểu và tối đa là bao nhiêu, GEE sẽ sử dụng các giá trị mặc định. Ví dụ: hình ảnh bạn đã thêm vào bản đồ trong bài tập trước được hiển thị dưới dạng hình ảnh thang độ xám được kéo dài đến toàn bộ phạm vi dữ liệu.

Kiểm tra một biến trong bảng điều khiển

Bạn có thể khám phá kiểu dữ liệu của ảnh bằng lệnh print và kiểm tra đối tượng ảnh trong tab Console.

Bước 1. Tạo một tập lệnh mới, sao chép và dán cú pháp từ bài tập trước vào đó (hoặc nhấp vào 'Save as' và nhập tên của tập lệnh mới). Sau đó dán cú pháp dưới đây.

```
// Print the image object to inspect it. print('SRTM image', image);
```

Khi bạn chạy tập lệnh, hãy lưu ý rằng một đối tượng xuất hiện trong bảng điều khiển. Để kiểm tra các thuộc tính của đối tượng, hãy mở rộng đối tượng

bằng cách nhấp vào hình tam giác bên trái đối tượng. Mở rộng thuộc tính 'bands', dải độ cao tại chỉ mục '0', và thuộc tính 'data_type' của dải độ cao để nhận thấy rằng đó là kiểu dữ liệu int16 có dấu. Tại đây bạn cũng có thể tìm thấy phạm vi của lớp, trong trường hợp này giá trị pixel tối đa là 32767 và tối thiểu -32768. Một thuộc tính quan trọng khác được chỉ định bằng 'crs' là hệ quy chiếu của lớp. Đây là phương tiện mà hệ tọa độ và dữ liệu được hiển thị trên một bề mặt phẳng, tức là chuyển bề mặt cong của trái đất sang bề mặt phẳng. Trong trường hợp này, hệ quy chiếu là EPSG:4326.

Inspector Tasks

Use print (...) to write to this console.

SRTM image

* Image CGIAR/SRTM90_V4 (1 band)

type: Image

id: CGIBR/SRTM90V4

version: 1624972749315785

"0: I"elevation", signedir.tisjEFSG:4326, 4320Ũ0X14400C i.-l: filiation
|crs: EPSG:4326 I

► crs_transorm: [0.0008333333333333, 0,-180, 0,-0.000833 ' data type: s
igned int16

type: PixelType

max: 32767

min: -32768

precision: int

► dimensions: [432000,144000]

Cú pháp đầy đủ cho bài tập này có thể được tìm thấy ở đây:

<https://code.earthengine.google.com/90ba40a2955d7805b1d0adbe582341>

7c

2.5. Trực quan hóa

Để thay đổi cách kéo dài dữ liệu, bạn có thể cung cấp một tham số bổ sung cho hàm Map.addLayer(). Để khám phá những giá trị nào cần sử dụng, hãy kích hoạt tab Inspector và nhấp vào xung quanh trên bản đồ để có ý tưởng về phạm vi giá trị pixel. Bạn có thể quyết định rằng dữ liệu nên được kéo dài từ 0 đến 1200 m. Để hiển thị hình ảnh trong phạm vi này:

Bước 1. Tạo một tập lệnh mới và sao chép và dán cú pháp từ bài tập trước vào đó (hoặc nhấp vào 'Save as' và nhập tên của tập lệnh mới). Sau đó, thay thế đoạn mã đầu tiên

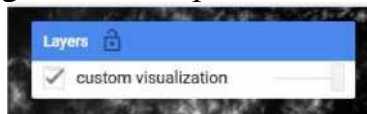
```
// Display the image on the map. Map.addLayer(image);
```

bằng đoạn mã thứ hai bên dưới, sau đó lưu và chạy tập lệnh:

```
// Display the image on the map (incl. range and layer name).  
Map.addLayer(image,  
  {min: 0, max: 1200}, 'custom visualization');
```

Chúng ta đã thêm một đối tượng tham số chứa phạm vi dữ liệu sẽ được hiển thị, trong trường hợp này là 0 đến 1200 m.

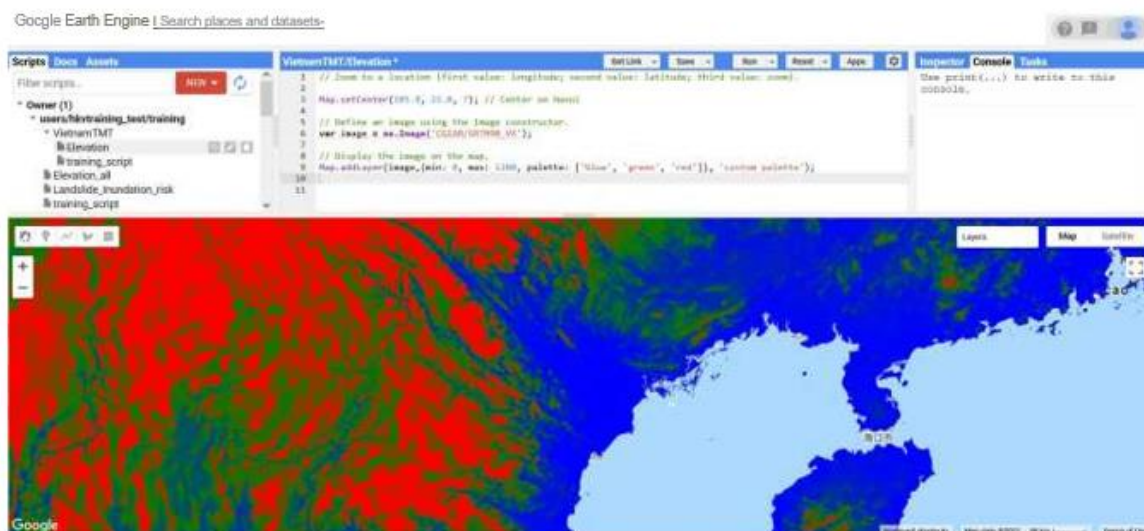
Lưu ý 'custom visualization', là tên của lớp được hiển thị trong trình quản lý Layer. Di chuột qua hộp Layers ở bên phải để xem hiệu quả của việc đổi tên lớp. Bạn có thể bỏ chọn hộp để tắt một lớp. Bạn có thể sử dụng thanh trượt để thay đổi độ trong suốt của lớp.



Bước 2. Chúng ta có thể thêm bảng màu bằng cách thêm thuộc tính bảng màu như sau:

```
// Display the image on the map (incl. range, palette and layer name).
Map.addLayer(image,
  {min: 0, max: 1200, palette: ['blue', 'green', 'red']},
  'custom palette');
```

Bây giờ bạn sẽ thấy bản đồ độ cao giống như sau:



2.6. Lọc

Trong các bài tập trước, mặc dù chúng ta căn bản đồ về Việt Nam, nhưng toàn bộ thế giới đã được tính toán. Để kiểm tra điều này, bạn có thể thu nhỏ bằng cách nhấp vào - (phóng to bằng dấu +) ở bên trái bản đồ (hoặc bằng cách lăn chuột). Việc tính toán cho toàn thế giới chiếm nhiều nhu cầu tính toán hơn chúng ta cần. Vì hiện tại chúng ta chỉ quan tâm đến Việt Nam nên chúng ta sẽ bắt đầu lọc theo quốc gia.

Bước 1. Tạo một tập lệnh mới và sao chép và dán cú pháp từ bài tập trước vào nó (hoặc nhấp vào 'Save as' và nhập tên của tập lệnh mới). Sau đó gõ 'FAO GAUL' vào thanh tìm kiếm dữ liệu ở phía trên màn hình. Nhấp vào lựa chọn thứ hai 'FAO GAUL 500 m: Global Administrative Unit Layers 2015, Second-Level Administrative Units'.

Trong phần mô tả của tập dữ liệu, bạn có thể đọc thông tin dữ liệu này bao gồm: "The *Global Administrative Unit Layers (GAUL)* compiles and disseminates the best available information on *administrative units for all the countries in the world*". Đây là những gì chúng ta đang tìm kiếm!

Bước 2. Nhấp 'Import' trong cửa sổ. Tập dữ liệu sẽ được thêm vào đầu tập lệnh của bạn và bây giờ được gọi là 'table'.

Elevation *

Imports (1 entry)

▶ var table : Table FAO/GAUL_SIMPLIFIED_500m/2ei5/level2

Câu hỏi nhanh 1.7

Bạn có thể sử dụng cách nào khác để nhập tập dữ liệu này?

Bước 3. Đổi tên dữ liệu FAO GAUL đã nhập thành countries bằng cách nhấp đơn vào 'table', nhập countries và nhấn nút Enter.



Bước 4. Bạn có thể chọn quốc gia bằng cách nhập cú pháp sau vào tập lệnh của mình. Vui lòng nhập ở đầu tập lệnh của bạn để đảm bảo rằng các dòng mã tiếp theo có thể sử dụng nó.

```
// Filter by country.
```

```
var Country = countries.filter(ee.Filter.eq('ADM0_NAME', 'Viet Nam'));
```

Bây giờ chúng ta đã chọn Việt Nam nhưng nó vẫn chưa được ghép với bản đồ độ cao mà chúng ta đã thực hiện trước đó. Chúng ta sẽ làm điều này trong bước tiếp theo.

Bước 5. Thêm dòng cú pháp sau bên dưới dòng cú pháp var image:

```
// Clip elevation image by country var srtm Vietnam = image.clip(Country);
```

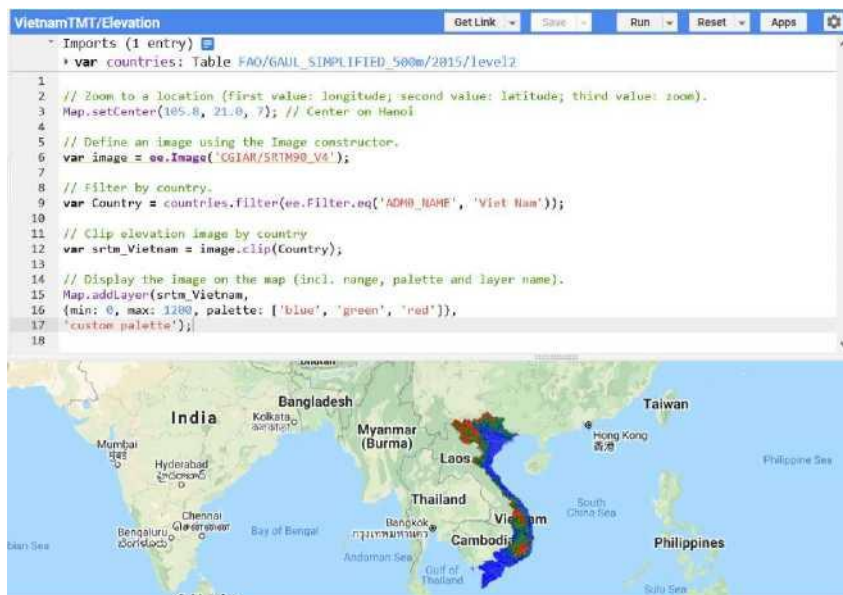
Nếu bạn chạy script, chưa có thay đổi trong bản đồ. Điều này là do chúng ta vẫn chọn biến 'image' trong dòng mã Map.addLayer.

Bước 6. Nếu chúng ta thay thành 'srtm_Vietnam' thì chỉ SRTM ở Việt Nam mới được hiển thị.

```
// Display the image on the map (incl. range, palette and layer name).
```

```
Map.addLayer(srtm_Vietnam,  
{min: 0, max: 1200, palette: ['blue', 'green', 'red']}, 'custom palette');
```

Nếu bạn nhấp 'Save' và 'Run' sau đó thu nhỏ một chút, bạn có thể thấy chỉ có Việt Nam được hiển thị.



Bước 7. Đừng quên lưu code của bạn.

Cú pháp đầy đủ cho bài tập này có thể được tìm thấy ở đây:

<https://code.earthengine.google.com/fb38a036aad7f4307888eee28eb06d7>

[a](#)

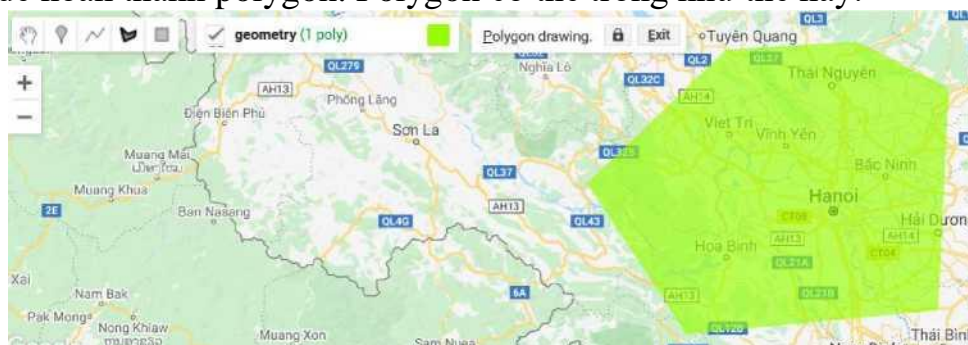
2.7. Polygon

Thay vì chọn một quốc gia như chúng ta đã làm trong bài tập 7, bạn cũng có thể tự mình xác định khu vực quan tâm.

Bước 1. Tạo một script mới, trống. Ở góc trên bên trái của bản đồ, bạn có thể thấy dấu hiệu để 'Vẽ hình': 'Draw a shape':



Nhấn vào nó. Dùng chuột vẽ một hình xung quanh khu vực mà bạn quan tâm. Nhấp đúp để hoàn thành polygon. Polygon có thể trông như thế này:



Lưu ý một đối tượng mới với tên biến 'geometry' đã được tạo tự động trong Trình biên soạn mã Code Editor:

Bước 2. Nhập các cú pháp sau vào Code Editor:

```
// Define an image using the Image constructor. var image = ee.Image('CGIAR/SRTM90_V4');  
// Clip elevation image to polygon var demClip = image.clip(geometry)  
// Display the image on the map (incl. range, palette and layer name).  
Map.addLayer(demClip, {'min':0,'max':400, palette: ['blue', 'green', 'red']}, "DEM Vietnam")
```

Bước 3. Nhấp Run.

Di chuột qua 'Geometry imports' ở góc trên cùng bên trái của bản đồ.

Bước 4. Bỏ chọn lớp 'geometry' và trong nút 'Layers' ở phía bên phải của bản đồ, hãy bỏ chọn lớp 'custom palette'. Bây giờ bạn có thể thấy DEM của mình:



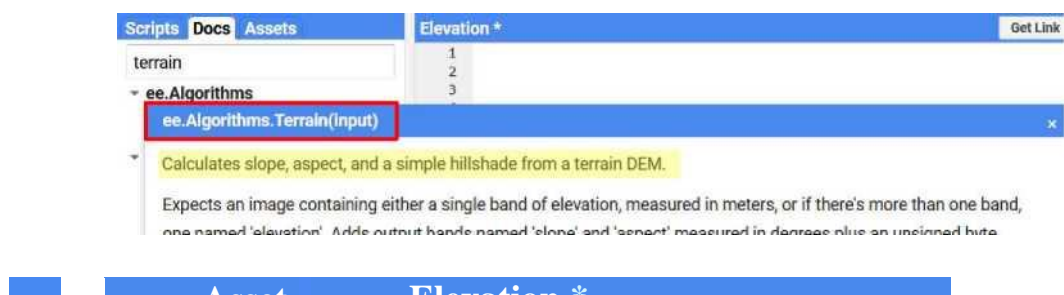
2.8. Độ dốc

Độ dốc của đất đai đóng một vai trò quan trọng trong nhiều quá trình tự nhiên, ví dụ như nước mưa chảy tràn và sạt lở đất (ví dụ, độ dốc từ 25 đến 40 độ có nghĩa là khả năng bị sạt lở cao). Từ bản đồ độ cao, chúng ta có thể xác định độ dốc trong một khu vực theo độ. Hãy làm điều này cho cả Việt Nam.

Bước 1. Tạo một tập lệnh mới và sao chép và dán mã từ Bài tập 7 (Lọc) vào đó (hoặc nhấp vào 'Save as' và nhập tên của tập lệnh mới). Sau đó, thêm các dòng cú pháp sau để xác định độ dốc theo độ.

```
// Slope  
var terrain = ee.Algorithms.Terrain(srtm_Vietnam); var slope = terrain.select('slope');  
Map.addLayer(slope, {min: 0, max: 25}, 'slope')
```

Để hiểu chính xác cách một hàm làm việc, bạn luôn có thể tìm kiếm hàm đó trong Docs và đọc về nó. Xem ở đây hai hàm mới đã được sử dụng trong đoạn mã trước:



```
ee.image.select  
' ee.image  
aside(func, var_args) 5  
hersDescriDtorisetectors. buckets,... 6
```

slect(var_args)

Selects bands from an image.

Returns an image with the selected bands.

Arguments:

- **thì<i>'imani> ilmanel'**

Bước 2. Di chuột qua 'Layers' ở phía bên phải của bản đồ để thấy rằng bây giờ có hai lớp, đó là 'custom palette' (DEM được cắt bớt với bảng màu tùy chỉnh) và 'slope' mà chúng ta vừa tạo. Bạn có thể ẩn các lớp đó bằng cách bỏ chọn các hộp bên cạnh chúng và thay đổi độ mờ của chúng bằng cách di chuyển các thanh trượt.



Bước 3. Đừng quên lưu code của bạn!

2.9. Xuất ảnh

Đôi khi bạn muốn xuất hình ảnh bản đồ của mình để trình bày nó bằng các phương tiện khác hoặc sử dụng nó trong các công cụ GIS khác (ví dụ: ArcGIS hoặc QGIS). Chúng ta sẽ đưa ra một ví dụ về cách xuất trong bài tập này.

Xuất ảnh dưới dạng tiff

Bước 1. Tạo một tập lệnh mới và sao chép và dán mã từ Bài tập 9 (Độ dốc) vào đó (hoặc nhấp vào 'Save as' và nhập tên của tập lệnh mới). Để xuất hình ảnh được tạo trong bài tập trước sang Google Drive của bạn ở định dạng 'GeoTIFF', hãy thêm cú pháp bên dưới:

```
//Export to drive in tiff format  
Export.image.toDrive({  
  image: slope,  
  description: 'imageToGeoTiff',  
  scale: 200,  
  fileFormat: 'GeoTIFF'  
})
```

Bước 2. Kiểm tra Tasks đã chuyển sang màu cam và nhấp vào 'CHẠY'.

Khi

hoàn tất, bạn có thể tìm thấy * .tiff trong Google Drive của mình.



Khi xuất đôi khi xuất hiện 'lỗi' rằng số lượng ô lưới cần xuất quá lớn. Bạn có thể giải quyết vấn đề này bằng cách tăng tỷ lệ hoặc bằng cách chọn một khu vực nhỏ hơn.

2.10. Xuất hình ảnh dưới dạng PNG hoặc JPG

Bước 1. Để xuất hình ảnh dưới dạng PNG hoặc JPG, hãy thêm cú pháp bên dưới vào mã của bạn:

```
//Export as PNG and print download URL (region: array of lon/lat points (E,S,W,N))
var map = slope.visualize({bands:"slope", min:0, max:25, forceRgbOutput:true});
var url = map.getThumbURL({dimensions:'1000',region: [[111,16],[106,8],[101,16],[106,24]], format:'png'}); print(url)
```

Hình ảnh được tạo ra như được chỉ ra trong mô tả "mảng (array) các điểm kinh độ/vĩ độ (E,S,W,N)" (xem hình bên dưới). Điều này có nghĩa là đối với mỗi điểm trung tâm của vùng, một giá trị (lon/lat) cần được đưa ra.

<https://c.tadst.com/gfx/600x337/longitude-and-latitude-simple.png?1>

Bước 2. Kiểm tra Console chuyển sang màu cam và mở liên kết đến hình ảnh. Nhấp vào nút chuột phải khi đang ở trên hình ảnh và sau đó nhấp vào 'Save Image As' để lưu hình ảnh trên máy tính của bạn.

Bước 3. Thêm các cú pháp này vào tập lệnh và chạy lại tập lệnh để xem vị trí đặt các điểm của vùng cho hình ảnh:

```
//Add points to map
var east_point = ee.Geometry.Point([111,16]);
var south_point = ee.Geometry.Point([106,8]); var west_point = ee.Geometry.Point([101,16]);
var north_point = ee.Geometry.Point([106,24]);
Map.addLayer(east_point,{color: 'green'}, 'point east')
Map.addLayer(south_point,{color: 'yellow'}, 'point south')
Map.addLayer(west_point,{color: 'red'}, 'point west')
Map.addLayer(north_point,{color: 'blue'}, 'point north')
```

Bạn có thể xuất bản đồ của mình ở các định dạng khác nhau và đến các vị trí khác nhau. Để biết thêm thông tin, hãy xem ở đây:

<https://developers.google.com/earth-engine/exporting>

Cú pháp đầy đủ cho bài tập này có thể được tìm thấy ở đây:

<https://code.earthengine.google.com/6c09b813a9d7e05c901eb3d5050284>



2.11. Tải lên một shapefile

Trong một số trường hợp, bạn có thể muốn sử dụng dữ liệu mà bạn đã tự tạo hoặc không thể truy cập trực tiếp từ GEE. May mắn thay, GEE cung cấp khả năng tải lên dữ liệu của riêng bạn. Trong bài thực hành này, bạn sẽ học cách thực hiện việc này, sử dụng ví dụ về một shapefile của các cơ sở y tế ở Việt Nam. Mục đích là để hình dung tất cả các cơ sở y tế cũng như làm nổi bật cơ sở nào là nhà thuốc và cơ sở nào là bệnh viện. Hãy bắt đầu!

Bước 1. Tạo một tập lệnh trống mới. Tải tệp zip 'hotsm_vnm_health_facilities_points_shp.zip' chứa shapefile các cơ sở y tế tại Việt Nam từ Moodle hoặc trực tiếp từ trang web sau: https://data.humdata.org/dataset/hotsm_vnm_health_facilities

Bước 2. Trong GEE, shapefiles có thể được tải lên vào Assets ở bảng điều khiển bên trái. Đi đến Assets, nhấp vào 'NEW' và sau đó chọn 'Shape files'.

image collection Folder

Bước 3. Chọn tệp zip có chứa shapefile của các cơ sở y tế trong đó (không cần giải nén tệp zip trước) và sau đó nhấp vào 'UPLOAD':

Bước 4. Trong Tasks trên bảng điều khiển bên phải, bạn sẽ thấy rằng tệp shapefile đang được tải lên. Sau khi hoàn tất, bạn có thể tìm thấy shapefile đã tải lên trong tab Assets ('LEGACYASSETS'). Bạn có thể cần phải nhấp vào để làm mới chế độ xem. Trong các bước tiếp theo, chúng ta sẽ nhập shapefile này vào một tập lệnh mới và trực quan hóa nó trên bản đồ.

Bước 5. Đi đến Scripts tạo một tập lệnh mới cho bài tập này. Mở tập lệnh mới, sau đó nhấp lại vào Assets, nhấp vào 'hotsm_vnm_health_facilities_points_sh' và sau đó nhấp vào 'IMPORT'. Shapefile của bạn hiện đã được nhập vào tập lệnh (xem 'Imports' ở đầu tập lệnh của bạn).

Bước 6. Thay thế tên biến 'table' bằng 'health_facilities', bằng cách nhấp đúp vào nó và nhập tên mới:

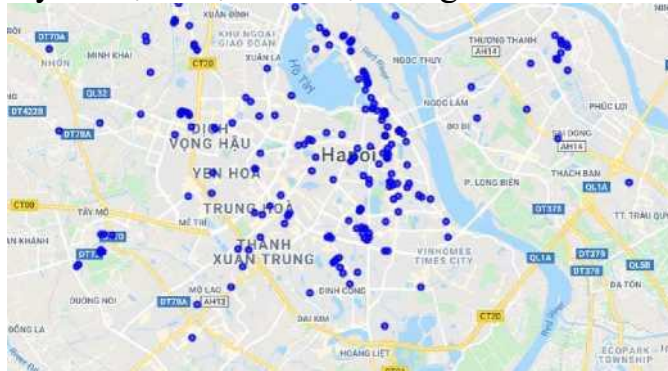
^T Imports (1 entry) [B](#)

► var **I** health facilities:**I** Table
users/hkvtraining_test/hotosrr_vnm_health_facilities_poi.-

Bước 7. Vì chúng ta chưa tự tạo shapefile mà đã tải xuống, trước tiên chúng ta cần hiểu thuộc tính mô tả nó. Thêm các dòng cú pháp sau vào tập lệnh (để căn giữa Hà Nội, in các thuộc tính ra bảng điều khiển và thêm tất cả các cơ sở y tế vào bản đồ), sau đó lưu và chạy tập lệnh:

```
// Zoom to a location (first value: longitude; second value: latitude; third value: zoom).  
Map.setCenter(105.85, 21.01, 12); // Center on Hanoi  
//Print table properties (incl. name) print('Health Facilities',  
health_facilities);  
// Add all health facilities to map (incl. palette and layer name)  
Map.addLayer(health_facilities, {color:'blue'}, 'health facility');
```

Tất cả các cơ sở y tế hiện được thể hiện bằng màu xanh lam trên bản đồ:



Bây giờ hãy kiểm tra bảng điều khiển. Trong 'columns', bạn có thể tìm thấy các thuộc tính được sử dụng. Trường 'amenity' là điều chúng ta quan tâm. Trường này cho chúng ta biết cơ sở y tế nào là bệnh viện, nhà thuốc, phòng khám, nha khoa, v.v.

Tuy nhiên, chúng ta vẫn chưa biết tên chính xác để mô tả 'amenity'.

Bước 8. Để tìm ra tên nào được sử dụng trong trường 'amenity', hãy thêm các cú pháp sau vào tập lệnh, lưu và chạy tập lệnh và kiểm tra bảng điều khiển:

```
// Print property "amenity" (incl. name)  
var amenity_names =  
health_facilities.aggregate_array("amenity").distinct().getInfo ();  
print('Healthcare names', amenity_names);
```

Rõ ràng có một số cơ sở y tế trong shapefile không được mô tả bằng 'amenity'. Những đối tượng khác có tên như 'clinic', 'doctors', 'hospital', v.v.

Hiện tại, chúng ta không tìm hiểu sâu hơn về lý do tại sao lại có những cái tên bị thiếu. Chúng ta chỉ cần chọn những cơ sở được mô tả là 'pharmacy' và những cơ sở được mô tả là 'hospital'.

Bước 9. Thêm các cú pháp sau vào tập lệnh để lọc các cơ sở y tế và thêm các biến mới 'pharmacies' và 'hospitals' vào bản đồ. Lưu và chạy tập lệnh:

```

// Filter pharmacies var pharmacies =
health_facilities.filter(ee.Filter.eq('amenity', 'pharmacy'));
// Filter hospitals var hospitals =
health_facilities.filter(ee.Filter.eq('amenity',
'hospital'));
print('hospitals',hospitals);
// Add pharmacies to map (incl. palette and layer name)
Map.addLayer(pharmacies, {color: 'yellow'},'pharmacy');
// Add hospitals to map (incl. palette and layer name)
Map.addLayer(hospitals, {color: 'red'},'hospital');

```



Bây giờ bạn đã tạo một bản đồ hiển thị tất cả các cơ sở y tế (có trong shapefile) màu xanh lam, các hiệu thuốc màu vàng và bệnh viện màu đỏ. Nếu bạn muốn tìm hiểu thêm về cách tải dữ liệu lên GEE, hãy xem trang này: https://developers.google.com/earth-engine/guides/table_upload

2.11. Height Above Nearest Drainage (HAND)

HAND là viết tắt của Height Above Nearest Drainage. Nó cung cấp thông tin về các địa điểm dễ bị ngập lụt. Các khu vực có giá trị HAND thấp với giá trị cao hơn xung quanh dễ bị ngập lụt hơn.

Đã có ImageCollection được tạo bởi Donchyts mà chúng ta có thể sử dụng: 'Donchyts et al., Global 30m Height Above the Nearest Drainage'.

Bước 1. Tạo một tập lệnh mới và sao chép và dán mã từ Bài tập 7 (Lọc) vào đó (hoặc nhấp vào 'Save as' và nhập tên của tập lệnh mới). Bạn có thể xóa dòng 'Define an image', dòng 'Clip elevation' và dòng 'Display the image on the map'. Bây giờ hãy thêm các cú pháp bên dưới vào tập lệnh của bạn và nhấp vào 'Run'. Bạn có nhận ra những khu vực thường xuyên bị ngập lụt có giá trị HAND thấp ở Việt Nam không?



Cú pháp đầy đủ cho bài tập này có thể được tìm thấy ở đây:

<https://code.earthengine.google.com/cae69bba1effb12a58b454f968d9714>

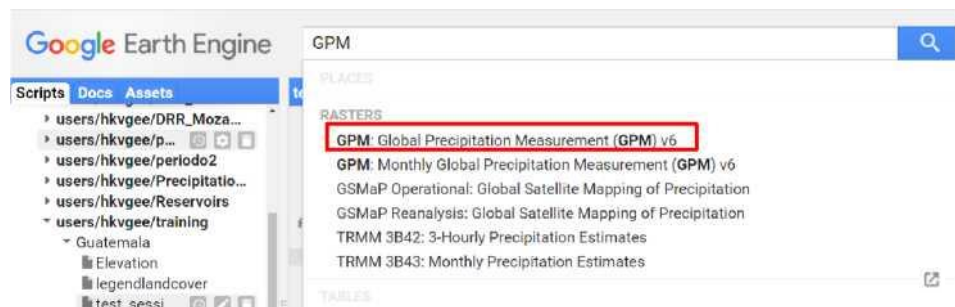
3

3. ỨNG DỤNG CỦA GEE

3.1. Xác định lượng mưa toàn cầu

Google Earth Engine lưu trữ nhiều sản phẩm dữ liệu lượng mưa. Có thể xác định sản phẩm nào hữu ích nhất cho Việt Nam bằng cách sử dụng dữ liệu quan trắc dài hạn. Nếu cơ quan Khí tượng ở Việt Nam có số liệu về lượng mưa phù hợp, bạn có thể nhập số liệu đó vào tập lệnh của mình và so sánh với sản phẩm lượng mưa sẽ sử dụng trong bài tập này. Trong bài tập này, sẽ sử dụng dữ liệu Mưa Toàn cầu - Global Precipitation Measurement (GPM) của NASA vì nó được đánh giá khá đáng tin cậy. GPM cung cấp dữ liệu lượng mưa cứ mỗi 30 phút theo đơn vị mm/giờ.

Bước 1. Hãy tạo một tập lệnh mới cho bài tập này. Trong thanh tìm kiếm ở đầu màn hình, nhập 'GPM', kết quả hiển thị ở trên cùng là 'GPM: Global Precipitation Measurement (GPM) v6'. Nhấn vào nó. Ở phía bên trái, bạn có thể thấy 'Cú pháp' để sử dụng tập dữ liệu này. Chúng ta đang tìm kiếm dữ liệu lượng mưa đã được hiệu chỉnh. Kiểm tra phần 'BANDS' để tìm ra dải băng tần nào chứa dữ liệu đó.



GPM: Global Precipitation Measurement (GPM) v6

Bây giờ chúng ta đã biết cách nhập ImageCollection này và băng tần nào

chúng ta cần sử dụng cho mục đích của mình. Trong trường hợp bạn muốn tìm hiểu thêm về tập dữ liệu, hãy nhấp vào 'CITATIONS'. Bạn có thể sao chép tài liệu tham khảo này và dán vào Google để tìm hiểu thêm.

Bước 2. Nhập tập dữ liệu GPM cho giai đoạn từ 2020-10-12 đến 2020-10-19). Trong thời gian này, một cơn bão nhiệt đới đã tràn qua Việt Nam, xem thêm ở đây: [https://vi.wikipedia.org/wiki/B%C3%A3o_Nangka_\(2020\)](https://vi.wikipedia.org/wiki/B%C3%A3o_Nangka_(2020)).

```
// Import dataset (with time filter and band selection) var gpm_dataset = ee.ImageCollection('NASA/GPM_L3/IMERG_V06').filterDate('2020-10-12','2020-10-19').select('precipitationCal');
```

Bước 3. In tập dữ liệu ra bảng điều khiển để kiểm tra những ngày này có trong bộ dữ liệu GPM hay không.

```
// Print dataset print('gpm_dataset', gpm_dataset);
```

Trong **Bảng điều khiển**, một bộ sưu tập hình ảnh ImageCollection với tên 'gpm_dataset' xuất hiện.

Bước 4. Nhấp vào hình tam giác nhỏ bên cạnh 'ImageCollection' và sau đó lại nhấp vào hình tam giác nhỏ bên cạnh 'features' để thấy rằng dữ liệu thực sự có sẵn mỗi nửa giờ trong khoảng thời gian từ ngày 12 tháng 10 năm 2020 đến ngày 19 tháng 10 năm 2020 hay không (Ví dụ, phân đuôi: '20201012233000' có nghĩa là năm 2020, tháng 10, ngày 12, giờ 23, phút 30, giây 0).

Inspector jjwipwa Tasks

Use print(...) to write to this console.

Bây giờ chúng ta muốn tính tổng lượng mưa của những ngày này. Vì dữ liệu được cung cấp cứ nửa giờ một lần theo đơn vị mm/giờ nên chúng ta sẽ cần phải chia dữ liệu cho 2 sau khi tính tổng. Và chúng ta cũng muốn giới hạn tính toán lượng mưa chỉ ở Việt Nam để tiết kiệm thời gian tính toán.

Bước 5. Tính tổng lượng mưa và sau đó chia cho 2 để chuyển đổi sang mm/giờ. Đồng thời giới hạn dữ liệu về Việt Nam (như đã học trong WP1, Bài tập 7).

```
// Calculate the total precipitation between 2020-10-12 and 2020-10-19 var prec_total = gpm_dataset.sum().divide(2); // to sum and calculate per hour, every 30 min a result // Only select Vietnam var countries = ee.FeatureCollection("FAO/GAUL_SIMPLIFIED_500m/2015/level2"); var Country = countries.filter(ee.Filter.eq('ADM0_NAME', 'Viet Nam')); var prec_total_gtm = prec_total.clip(Country);
```

Bước 6. Chúng ta cũng sẽ muốn trực quan hóa dữ liệu lượng mưa để hiển thị các khu vực ẩm ướt nhất.

```
// Define colour palette and visualization parameters var colors_precip = ['white', 'blue', 'yellow', 'red']; Map.addLayer(prec_total_gtm, {min: 0.0, max: 600, palette: colors_precip}, 'GPM total prec. [2020-10-12, 2020-10-19] (mm)'); // Zoom to a location (first value: longitude; second value: latitude; third value: zoom). Map.setCenter(106.9, 17, 5);
```

Bước 7. Chúng ta sẽ thêm một chú giải để giải thích giá trị lượng mưa.

Bạn có thể sao chép các dòng mã sau vào tập lệnh của mình. Trong trường hợp bạn muốn tạo chú thích cho mục đích khác, bạn chỉ cần điều chỉnh các đoạn mã được **in đậm**.

```
//Create legend precipitation
// set position of panel var legend = ui.Panel({ style: {
position: 'bottom-right', padding: '8px 15px'
}
});
// Create legend title
var legendTitle = ui.Label({
value: 'Cumulative precipitation (mm)', style: {
fontWeight: 'bold', fontSize: '18px', margin: '0 0 4px 0', padding: '0'
}
});
// Add the title to the panel legend.add(legendTitle);
// Creates and styles 1 row of the legend. var makeRow = function(color,
name) {
// Create the label that is actually the colored box. var colorBox =
ui.Label({
style: { backgroundColor: color, // Use padding to give the box height and
width. padding: '8px', margin: '0 0 4px 0'
}
});
// Create the label filled with the description text. var description =
ui.Label({
value: name,
style: {margin: '0 0 4px 6px'}
});
// return the panel
return ui.Panel({
widgets: [colorBox, description], layout:
ui.Panel.Layout.Flow('horizontal')
});
};
// Palette with the colors, name of the legend
var palette = colors_precip;
var names = ['0', '200', '400', '600'];
// Add color and names
for (var i = 0; i < 4; i++) { legend.add(makeRow(palette[i], names[i]));
}
// add legend to map
```

```
Map.add(legend);
```

Bổ sung về dữ liệu theo thời gian trong GEE

Trong bài tập trước, chúng ta đã sử dụng dữ liệu theo thời gian. Trong GEE, dữ liệu theo thời gian được tải dưới dạng ImageCollection, một tập hợp các hình ảnh vệ tinh. Nếu bạn muốn tải một tập dữ liệu theo thời gian, bạn phải chỉ định ngày bắt đầu (startdate) và ngày kết thúc (enddate) mà bạn muốn thu thập dữ liệu. Bạn có thể thực hiện việc này bằng lệnh .filterDate(startdate, enddate). Trong bài tập trước, chúng ta đã sử dụng lệnh này ở Bước 2:

```
var gpm_dataset = ee.ImageCollection('NASA/GPM_L3/IMERG_V06')  
.filterDate('2020-10-12','2020-10-19').select('precipitationCal');
```

Không thể vẽ một ImageCollection trực tiếp trên bản đồ. Trước tiên, bạn cần chọn một ảnh từ bộ sưu tập. Ngoài ra, để chỉ định khoảng thời gian, bạn cũng có thể, ví dụ, chỉ chọn hình ảnh đầu tiên trong bộ sưu tập:

```
var gpm_dataset =  
ee.ImageCollection('NASA/GPM_L3/IMERG_V06').first()
```

Ngoài ra, cũng có thể biến một ImageCollection thành một ảnh bằng cách tạo một lớp mới là số liệu thống kê của bộ sưu tập. Một số ví dụ:

Đề tính giá trị	.me
Đề tính giá trị	.su
Đề tính giá trị nhỏ	.mi
Đề tính giá trị lớn	.ma
Đề tính giá trị	.me

Cần lưu ý thời gian tính toán sẽ tăng lên nếu bạn tính toán cho một diện tích lớn hoặc thời gian dài.

Điều tương tự cũng có thể được thực hiện bằng cách sử dụng các hàm tinh giản (Reducer functions) trong GEE. Tinh giản (Reducer) là cách tổng hợp dữ liệu theo thời gian, không gian, dải băng tần, mảng và các cấu trúc dữ liệu khác trong GEE. Lớp hàm ee.Reducer chỉ định các phương thức dữ liệu được tổng hợp. Các cơ chế tinh giản trong lớp có thể chỉ định một phép thống kê đơn giản để tổng hợp (ví dụ: lớn nhất, nhỏ nhất, trung bình, trung vị, độ lệch chuẩn, v.v.) hoặc một tóm tắt phức tạp hơn của dữ liệu đầu vào (ví dụ: biểu đồ histogram, hồi quy tuyến tính, list). Sự tinh giản có thể xảy ra theo

- **thời gian** (imageCollection.reduce()),
- **không gian** (image.reduceRegion(), image.reduceNeighborhood()),
- **băng tần** (image.reduce()),
- hoặc **thuộc tính không gian của một FeatureCollection** (featureCollection.reduceColumns() hoặc phương pháp FeatureCollection bắt đầu với aggregate_).

Trong Bài tập 3 (Lượng mưa năm), chúng ta sẽ sử dụng các hàm này để tính toán tổng lượng mưa năm và sự khác biệt giữa lượng mưa của các năm khác nhau.

3.2. Xác định phạm vi của một lưu vực

Ranh giới lưu vực thường được sử dụng cho các phân tích và mô hình hóa. Lưu vực là phần diện tích mà trong đó dòng chảy sinh ra do mưa được thu gom và đều thoát qua một điểm. Nó thường được bao quanh bởi những ngọn núi. Bạn có thể xác định lưu vực ở các quy mô khác nhau, từ lưu vực quy mô lớn bao gồm lãnh thổ của nhiều quốc gia (ví dụ: Lưu vực sông Hồng) đến quy mô địa phương (ví dụ lưu vực sông Đáy).

Trong bài tập này, chúng ta sẽ sử dụng tập dữ liệu của *WWF HydroSHEDS Basins* để lựa chọn một lưu vực. Trong website của *WWF HydroSHEDS* (<https://www.hydrosheds.org/>) bạn có thể tìm thấy nhiều thông tin hơn về tập dữ liệu này:

"HydroSHEDS là một sản phẩm bản đồ cung cấp các thông tin thủy đặc cho các ứng dụng quy mô toàn cầu và khu vực theo một định dạng nhất quán. Nó cung cấp một bộ dữ liệu tham chiếu địa lý (vector & raster) ở nhiều quy mô khác nhau, bao gồm mạng lưới sông, ranh giới lưu vực, hướng thoát nước, và dòng chảy tích lũy. HydroSHEDS dựa trên dữ liệu độ cao có độ phân giải cao thu được trong chuyến bay của Tàu con thoi cho Nhiệm vụ đo đạc Địa hình - Shuttle Radar Topography Mission (SRTM) của NASA."

Bây giờ chúng ta sẽ làm việc với dữ liệu này:

Bước 1. Đầu tiên, chúng ta sẽ tải dữ liệu từ tập dữ liệu này trong GEE Code Editor và thêm nó vào bản đồ. Để làm được điều đó, hãy tạo một tập lệnh mới và thêm các đoạn mã sau:

```
// import the dataset with watersheds var hydrosheds =  
ee.FeatureCollection("WWF/HydroSHEDS/v1/Basins/hybas_4");  
Map.setCenter(105.8, 21.0, 7); // Center on Hanoi  
// Add to map Map.addLayer(hydrosheds);
```

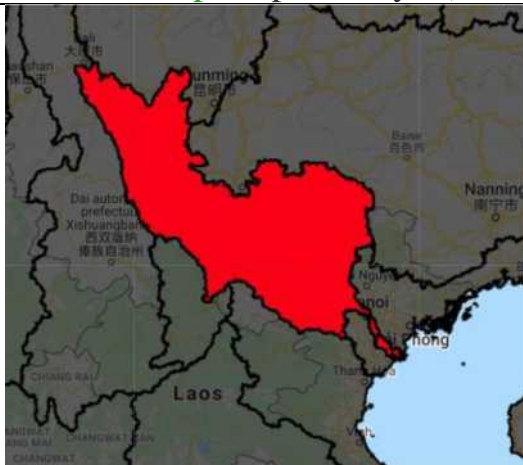
Bản đồ sẽ hiển thị như sau:



Bạn sẽ thấy tất cả các lưu vực được xác định trên thang cấp độ 4. Mức này cho biết quy mô của các lưu vực. Nếu quy mô thấp hơn, nhiều lưu vực sẽ được gộp lại thành các lưu vực lớn hơn.

Bước 3. Bạn có thể sử dụng HYBAS_ID '4040015000' để chọn Lưu vực sông Hồng từ tập dữ liệu này và vẽ nó trong cùng một bản đồ:

```
// Select river basin
var Red_River = hydrosheds.filter(ee.Filter.eq('HYBAS_ID',
4040015000));
// Add to map Map.addLayer(Red_River.style({ fillColor: 'red'}));
```



3.3. Xác định lượng mưa năm trên lưu vực Sông Hồng

Trong bài tập này, chúng ta sẽ sử dụng các hàm thống kê để tính Lượng mưa năm trên lưu vực Sông Hồng (Lượng mưa năm - Annual rainfall là giá trị tổng của các lượng mưa ngày) và các số liệu thống kê. Với những số liệu thống kê này, chúng ta có thể so sánh lượng mưa giữa các năm và phân tích lượng mưa lớn nhất, nhỏ nhất và trung bình trên một khu vực.

Bước 1. Tạo một tập lệnh mới, ví dụ như "Rain_RedRiver". Tìm kiếm tập dữ liệu lượng mưa hàng ngày CHIRPS và nhập nó vào tập lệnh của bạn. Tên mặc định của biến được nhập là "ImageCollection", đổi tên biến này thành "CHIRPS". Đồng thời hãy nhập lưu vực sông Hồng từ bài tập trước một lần nữa:

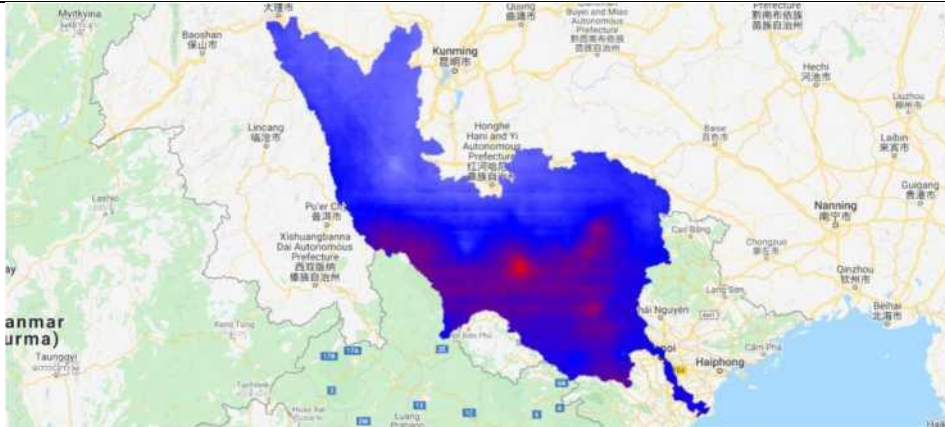
```
// Import the precipitation data
var CHIRPS = ee.ImageCollection("UCSB-CHG/CHIRPS/DAILY");
// Import the boundaries of the Red River Basin var watershed
=ee.FeatureCollection("WWF/HydroSHEDS/v1/Basins/hybas_4"); var
Red_River = watershed.filter(ee.Filter.eq('HYBAS_ID', 4040015000));
```

Bước 2. Chúng ta sẽ tập trung vào lượng mưa trên lưu vực sông Hồng vào năm 2019 và 2020. Đầu tiên, chúng ta tạo một biến với lượng mưa vào năm 2020. Chúng ta sẽ sử dụng bộ lọc từ 1-1-2020 đến 31-12-2020 để chỉ chọn hình ảnh của năm 2020 từ ImageCollection (như được thực hiện trong Bài tập 1 "Giáng thủy" của WP2). Nhập mã sau:

```
// Define period of interest
var Rain2020 = CHIRPS.filterDate('2020-01-01','2020-12-31');
```

Bước 3. Bây giờ chúng ta muốn một cách trực quan kiểm tra dữ liệu CHIRPS cho năm 2020. Trong bước này, chúng ta tạo bản đồ lượng mưa năm 2020 bằng cách chỉ tính toán và hiển thị bộ dữ liệu CHIRPS theo ranh giới của lưu vực Sông Hồng và tính tổng các giá trị mưa ngày. Lớp này sau đó được hiển thị trực quan:

```
// Create annual rainfall map for 2020
var SumYearlyRain2020 = (Rain2020.sum().clip(Red_River));
// Add to map
Map.addLayer(SumYearlyRain2020,{'min': 0,'max':3000, palette: ['white',
'Blue', 'Red']}, "Sum of rainfall in 2020");
```



Bước 4. Bây giờ chúng ta có thể trích xuất các số liệu thống kê từ bản đồ SumYearlyRain. Ví dụ: hãy tính toán lượng mưa trung bình trên lưu vực vào năm 2020 và in nó ra Console:

```
// Calculate geostatistics with reducer functions
var ZonalMean2020 =
SumYearlyRain2020.reduceRegion(ee.Reducer.mean(),Red_River,500
0);
print('Zonal mean 2020',ZonalMean2020);
```

Bước 5. Chúng ta cũng muốn biết các giá trị nhỏ nhất và lớn nhất của

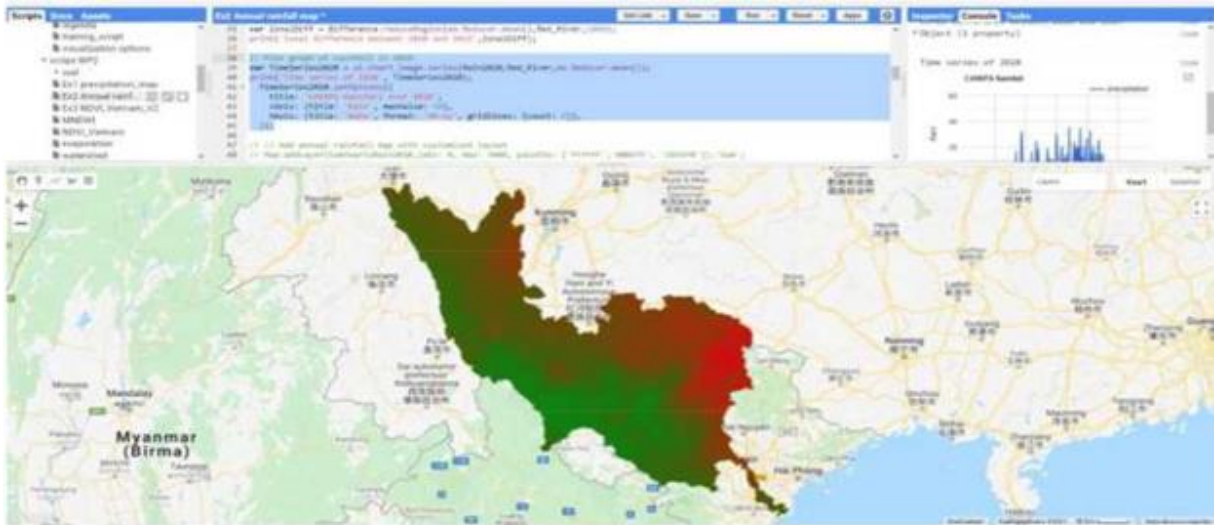
lượng mưa xảy ra trên lưu vực Sông Hồng. Xem trong "Docs" để biết các hàm tính giản thích hợp để tính toán giá trị nhỏ nhất và lớn nhất, sau đó thêm chúng vào mã, tương tự như bước trước.

Bước 6. Cuối cùng, chúng ta muốn so sánh lượng mưa trên lưu vực sông Hồng của năm 2020 với một năm khác. Thực hiện các bước tương tự cho năm 2019 và trả lời các câu hỏi sau:

- Tính toán sự khác biệt giữa giá trị trung bình của lượng mưa năm 2020 và năm 2019. Năm nào ẩm ướt hơn?
- Giá trị lượng mưa trung bình của hai năm chênh lệch bao nhiêu mm?

```
// Define period of interest
var Rain2019 = CHIRPS.filterDate('2019-01-01','2019-12-31');
// Create annual rainfall map for 2019
var SumYearlyRain2019 = (Rain2019.sum().clip(Red_River));
// Calculate geostatistics with reducer functions
var ZonalMean2020 =
SumYearlyRain2020.reduceRegion(ee.Reducer.mean(),Red_River,5000
);
print('Zonal mean 2020',ZonalMean2020);
var ZonalMin2020 =
SumYearlyRain2020.reduceRegion(ee.Reducer.min(),Red_River,5000);
print('Zonal min 2020', ZonalMin2020);
var ZonalMax2020 =
SumYearlyRain2020.reduceRegion(ee.Reducer.max(),Red_River,5000)
; print('Zonal max 2020', ZonalMax2020);
var ZonalMean2019 =
SumYearlyRain2019.reduceRegion(ee.Reducer.mean(),Red_River,5000
);
print('Zonal mean 2019', ZonalMean2019);
```

Bước 7. Bạn có thể sử dụng hàm Subtract trong Code Editor để tính toán sự khác biệt chính xác giữa hai lớp. Trong các dòng mã tiếp theo, chúng ta sẽ tạo một lớp mới chứa giá trị chênh lệch giữa lượng mưa vào năm 2019 và năm 2020. Lớp này có thể được thêm vào bản đồ và chúng ta có thể tính toán tổng chênh lệch trên toàn bộ lưu vực từ lớp này.



```

// Calculate difference between 2019 and 2020 var Difference =
SumYearlyRain2020.subtract(SumYearlyRain2019);
// Add to map
Map.addLayer(Difference,{'min': -500,'max':500, palette: ['red','green']},
"Diff_rainfall");
// Calculate mean value with reducer function var ZonalDiff =
Difference.reduceRegion(ee.Reducer.mean(),Red_River,5000);
print('Zonal Difference between 2020 and 2019',ZonalDiff

```

GEE không chỉ có thể được sử dụng để hiển thị bản đồ không gian mà còn để vẽ biểu đồ để phân tích chuỗi thời gian. Đây là mục đích của bài tập này. Tất cả các chức năng liên quan đến biểu đồ có thể được tìm thấy trong **Docs** với từ khóa 'ui.Chart'.

Bước 1. Tạo một tập lệnh mới, sao chép và dán mã từ bài tập trước vào nó (hoặc nhập vào 'Save as' và nhập tên của tập lệnh mới). Để tạo biểu đồ với giá trị lượng mưa ngày vào năm 2020 cho lưu vực sông Hồng, hãy sử dụng các dòng mã bên dưới. Như bạn có thể thấy trong ví dụ bên dưới, cơ chế tinh giản theo giá trị trung bình có thể là một phần của cùng một dòng mã.

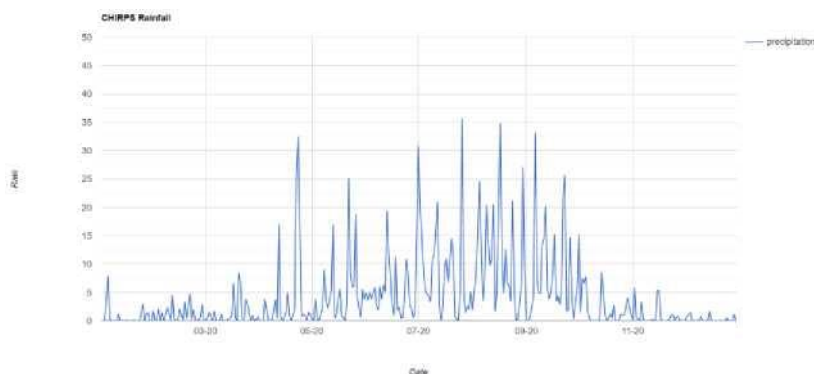
```

// Plot graph of rainfall in 2020
var TimeSeries2020 =
ui.Chart.image.series(Rain2020,Red_River,ee.Reducer.mean());
// Print data to console print('Time series of 2020', TimeSeries2020);
TimeSeries2020.setOptions({
title: 'CHIRPS Rainfall over 2020',
vAxis: {title: 'Rain', maxValue: 50},
hAxis: {title: 'Date', format: 'MM-yy', gridlines: {count: 6}},
});

```

Bạn có thể mở biểu đồ trong một cửa sổ mới bằng cách nhấp vào biểu tượng phía trên biểu đồ trong cửa sổ Bảng điều khiển. Trường hợp của chúng ta

biểu đồ sẽ trông như thế này:



Ở góc trên bên phải, bạn sẽ thấy một số tùy chọn mà bạn có thể sử dụng để xuất biểu đồ sang định dạng CSV, SVG hoặc PNG.

Cú pháp đầy đủ cho bài tập này có thể được tìm thấy ở đây:

<https://code.earthengine.google.com/16e43ddc879b355214d70a3891af2ec>

7

3.3. Xác định chỉ số thực vật

Chỉ số Thực vật (NDVI) là một chỉ số về độ xanh của một khu vực. Các khu vực xanh như khu vực nông nghiệp và rừng có NDVI cao và các khu vực đô thị có NDVI thấp. Công thức cho chỉ số này là:

$$(NIR - Red)$$

NDVI =

$$(NIR + Red)$$

Nói cách khác, NDVI là giá trị mà sự chênh lệch giữa hệ số phản xạ cận hồng ngoại (NIR) và đỏ (RED) của thực vật được chia cho tổng của chúng (dao động trong khoảng từ -1 đến 1). NDVI là một chỉ số rất hữu ích để phát hiện thảm thực vật và cũng đưa ra dấu hiệu về sức khỏe của cây trồng. Một cây khỏe mạnh hấp thụ hầu hết ánh sáng đỏ để quang hợp nhưng phản xạ hầu hết ánh sáng hồng ngoại, cho giá trị NDVI cao. Thảm thực vật không lành mạnh hấp thụ ít ánh sáng đỏ hơn nhưng nhiều ánh sáng hồng ngoại hơn, dẫn đến giá trị NDVI thấp hơn.

Phản xạ quang phổ của cây khỏe mạnh, cây không khỏe mạnh và đất ở bước sóng nhìn thấy và NIR. (Nguồn: <http://extension.usu.edu/nasa/html/on-target/near-infrared-tutorial/>)

Sử dụng NDVI để phân tích các loại lớp phủ xung quanh Hà Nội và quá trình đô thị hóa theo thời gian. Chúng ta sẽ sử dụng vệ tinh Sentinel 2 để xây dựng NDVI.

Bước 1. Tạo một tập lệnh mới. Nhập Sentinel 2 vào thanh tìm kiếm phía trên Trình soạn thảo mã để tìm thêm thông tin về vệ tinh này hoặc truy cập trang web: https://www.esa.int/Applications/Observing_the_Earth/Copernicus/Senti

nel-2

Bước 2. Nhập các dòng mã sau để chỉ định khu vực chúng ta quan tâm và nhập tập dữ liệu Sentinel 2 (biến S2). Chúng ta sẽ giới hạn tập dữ liệu theo phạm vi của khu vực đã chọn. (Xem bài tập 8 của WP1 để biết thêm về cách vẽ polygon).

```
// Create a polygon
var area = /* color: #98ff00 */ ee.Geometry.Polygon(
  [[[105.99806516238482, 21.022269503983605], [105.9699126965645,
21.04502124777042], [105.93970029422076, 21.033485586508295],
[105.9311172253731, 21.052070378680344], // Add polygon to map
Map.addLayer(area);
```

```
// Center on polygon Map.centerObject(area);
// Create image collection of S-2 imagery var S2 =
ee.ImageCollection('COPERNICUS/S2')
// Filter according to drawn polygon .filterBounds(area);
```

Công nghệ viễn thám quang học như sản phẩm Sentinel 2 không có khả năng xuyên qua các đám mây. Tuy nhiên, chúng ta có thể sử dụng dải băng tần Sentinel-2 QA60 (Đánh giá chất lượng) là một dải bit chứa thông tin về mặt nạ đám mây để che các đám mây. Thông tin thêm về quy trình tạo mặt nạ đám mây tự động được cung cấp tại đây: <https://sentinel.esa.int/web/sentinel/technical-guides/sentinel-2-msi/level-1c/cloud-masks>

Bước 3. Với những dòng mã sau, chúng ta có thể che những đám mây:

```
// Function to mask cloud from built-in quality band // information on
cloud
var maskcloud1 = function(image) { var QA60 = image.select(['QA60']);
return image.updateMask(QA60.lt(1));
};
var S2 = S2.map(maskcloud1); //mask clouds
```

Nếu bạn tìm kiếm tập dữ liệu của Sentinel 2, bạn sẽ tìm thấy thêm thông tin về các dải băng tần khác nhau (ví dụ: dải QA60, dải màu đỏ và dải NIR).

Bước 4. Bước tiếp theo là tính toán chỉ số thực vật (NDVI) bằng cách sử dụng band đỏ (Band 4) và hồng ngoại gần (Band 8) của ảnh Sentinel 2. Sau đó, chúng ta thêm NDVI như một band phụ (được gọi là 'nd') vào biến S2.

```
// Function to calculate and add an NDVI band var addNDVI =
function(image) {
return image.addBands(image.normalizedDifference(['B8', 'B4']));
};
```

```
// Add NDVI band to image collection var S2 = S2.map(addNDVI);
// print S2 to check if the NDVI band with the name nd is added
print(S2);
```

Bước 5. Để chuyển đổi ImageCollection của các hình ảnh NDVI thành một ảnh duy nhất, chúng ta tính giá trị trung bình của tất cả các ảnh NDVI vào

năm 2020. Để thực hiện điều này, hãy thêm các dòng mã sau vào tập lệnh:

```
// Extract NDVI band and create NDVI composite image var NDVI =  
S2.select(['nd']);  
var NDVI2020 = NDVI.filterDate('2020-01-01', '2020-12- 31').mean();
```

Bước 6. Bây giờ chúng ta sẽ thêm vào các thông số trực quan hóa và thêm lớp vào bản đồ:

```
// Create palettes for display of NDVI  
var visParams_ndvi = {min: -0.2, max: 0.8, palette: 'FFFFFF, CE7E45,  
DF923D, F1B555, FCD163, 99B718, 74A901, 66A000, 529400,' +  
'3E8601, 207401, 056201, 004C00, 023B01, 012E01, 011D01, 011301'};  
// Display NDVI results of 2020 on map  
Map.addLayer(NDVI2020.clip(area), visParams_ndvi, 'NDVI_mean2020');
```



Bước 7. Bây giờ chúng ta sẽ sử dụng chỉ số này để so sánh sự thay đổi trong việc sử dụng đất từ năm 2016 đến năm 2020. Chúng ta sẽ thực hiện điều này bằng cách thêm một lớp bổ sung với NDVI trung bình của năm 2016:

```
// Calculate the mean NDVI of 2016  
var NDVI2016 = NDVI.filterDate('2016-01-01', '2016-12- 31').mean();  
// Display NDVI results of 2016 on map  
Map.addLayer(NDVI2016.clip(area), visParams_ndvi, 'NDVI_mean2016');
```

Bước 8. Chọn khu vực có mức độ thay đổi lớn về NDVI (thay đổi mục đích sử dụng đất). Vẽ một polygon ở đó và thêm nó vào bản đồ (hoặc sao chép các dòng mã sau vào tập lệnh):

```
// Area with big change in NDVI  
var geometry =
```



```

/* color: #0000ff */
/* shown: false */
ee.Geometry.Polygon(
  [[[105.94358937473933, 20.98499792654286],
    [105.95972554417293, 20.982673909312954],
    [105.96264378758113, 20.993171768856804],
    [105.94753758640925, 21.004790676492835],
    [105.93277470799129, 20.997098259504117]]]);

```

```

// Add geometry to map Map.addLayer(geometry);
// Center on geometry Map.centerObject(geometry);

```



Bây giờ chúng ta sẽ tạo một biểu đồ histogram các giá trị NDVI trung bình của năm 2016 và 2020. Để làm được điều này, trước tiên chúng ta sẽ tạo một biến NDVI2, trong đó chúng ta lưu trữ hai ảnh: NDVI trung bình của năm 2016 và NDVI trung bình của năm 2020. Tiếp theo, chúng ta sử dụng hàm *ui.Chart.image.histogram* để vẽ biểu đồ histogram của hai ảnh đó. Biểu đồ histogram cho biết số lần một giá trị cụ thể xuất hiện trong mỗi ảnh. Nếu bạn thấy rằng các giá trị NDVI cao hơn (gần 1) xảy ra thường xuyên hơn, thì đây có thể là dấu hiệu của một vùng màu xanh lá cây.

Bước 9. Thêm mã sau vào tập lệnh của bạn:

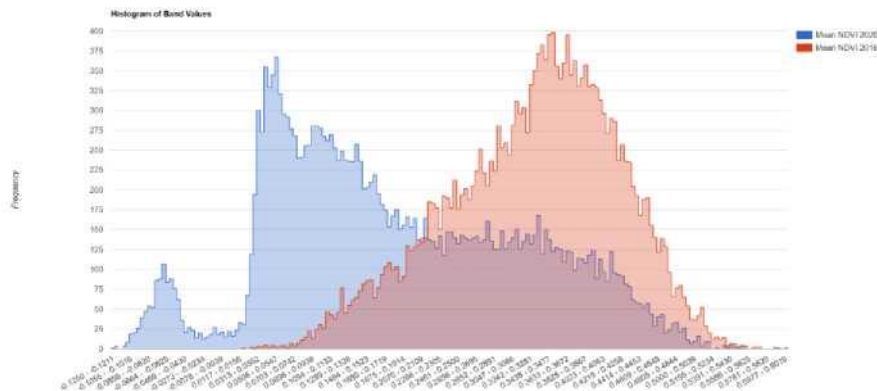
```

//Create histogram
// Add image of mean NDVI of 2020
var NDVI2 = S2.select(['nd']).filterDate('2020-01-01', '2020- 12-
31').mean();

// Add image of mean NDVI of 2016
var NDVI2 = NDVI2.addBands(S2.select(['nd']).filterDate('2016- 01-
01', '2016-12-31').mean());
// Chart variable
var hist_NDVI = ui.Chart.image.histogram(NDVI2, geometry.bounds(),
20)

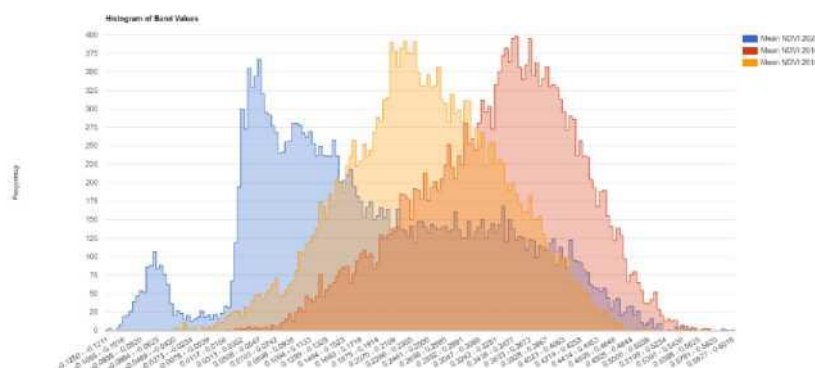
.setSeriesNames(['Mean NDVI 2020', 'Mean NDVI 2016']);
//Print chart to console print("NDVI", hist_NDVI);
//Print chart to console print("NDVI", hist_NDVI);

```



Bước 10. Bây giờ chúng ta hãy thử thêm một năm nữa vào biểu đồ. Thêm vào dòng mã sau ngay bên dưới dòng mã chúng ta đã thêm ảnh NDVI trung bình năm 2016. Đừng quên đặt tên cho nó trong phần tạo biểu đồ. Điều gì đã thay đổi?

```
// Add image of mean NDVI of 2018
var NDVI2 = NDVI2.addBands(S2.select(['nd']).filterDate('2018- 01-01',
'2018-12-31').mean());
```



Bạn có thể sử dụng nút phía trên biểu đồ để mở biểu đồ trong một cửa sổ mới. Tại đây, bạn cũng có thể chọn tải xuống biểu đồ dưới dạng PNG hoặc dữ liệu đằng sau biểu đồ ở định dạng CSV.

Google Earth Engine chứa bộ dữ liệu độ ẩm đất từ sứ mệnh Soil Moisture Active Passive (SMAP) của NASA. Vệ tinh SMAP đã đi vào quỹ đạo từ năm 2015, có độ phân giải không gian 25x25 km và cung cấp dữ liệu 2-3 ngày một lần. Vệ tinh có thể đo lượng nước trong 5 cm lớp đất bề mặt ở tất cả các vị trí trên bề mặt Trái đất không bị bao phủ bởi nước hoặc đóng băng.

3.4. Xác định lượng bốc hơi

Lượng nước bốc hơi từ bề mặt đất vào khí quyển phụ thuộc vào độ ẩm của đất. Thông tin về độ ẩm đất là chìa khóa để hiểu sự vận động của nước và thông lượng nhiệt giữa bề mặt và khí quyển tác động đến thời tiết và khí hậu. Chúng ta vẫn chỉ biết rất ít về sự thay đổi độ ẩm của đất ở cả quy mô khu vực hoặc toàn cầu, nhưng với sứ mệnh SMAP cùng với các chương trình hiện có (như AMSR-E và AMSR-2), điều này sắp thay đổi. Các phép đo độ ẩm đất thường xuyên và đáng tin cậy từ SMAP sẽ giúp cải thiện năng lực dự báo của các mô hình thời tiết và khí hậu.

Dữ liệu độ ẩm đất cho Việt Nam và so sánh nó với các thành phần khác của chu trình thủy văn : lượng mưa và lượng bốc thoát hơi nước.

Bước 1: Đầu tiên, hãy tạo một tập lệnh mới và đặt tên là 'WP3_soilmoisture'.

Bước 2: Chúng ta quan tâm đến độ ẩm đất theo không gian và thời gian ở lưu vực sông Hồng. Chúng ta sẽ sử dụng các ranh giới của bộ dữ liệu WWF HydroSHEDS (xem thêm Bài tập 2 của Gói đào tạo số 2). Bạn cũng có thể tải lên ranh giới của riêng mình thông qua thẻ Assets ở góc trên bên trái (xem thêm Bài tập 11 của Gói đào tạo số 1).

```
//Import the boundaries of the Red River basin
var watershed
=ee.FeatureCollection("WWF/HydroSHEDS/v1/Basins/hybas_4");
var Red_River = watershed.filter(ee.Filter.eq('HYBAS_ID',
4040015000));
```

Bước 3: Căn giữa bản đồ vào lưu vực sông Hồng. Bạn có thể sử dụng cú pháp *Map.centerObject* hoặc *Map.setCenter*(105.8, 21.0, 7);. Sau đó, hãy thêm ranh giới lưu vực vào bản đồ bằng cú pháp *Map.addLayer*.

```
// Add Red River basin and center the map
Map.addLayer(Red_River);
Map.centerObject(Red_River);
```

Bước 4: Chúng ta quan tâm đến của dữ liệu độ ẩm của đất theo mùa. Để thực hiện, chúng ta sẽ thu thập dữ liệu từ năm 2018 đến năm 2020. Chúng ta cũng tạo hai biến startdate và endddate để lọc tập dữ liệu.

// Set period

```
var startdate = '2018-01-01'; var endddate = '2020-12-31';
```

Bước 5: Tìm ba tập dữ liệu sau trong danh mục dữ liệu GEE và nhập chúng theo các biến (sao chép đoạn mã bộ sưu tập):

- NASA-USDA Enhanced SMAP Global Soil Moisture Data,
- CHIRPS Daily: Climate Hazards Group InfraRed Precipitation With Station Data (Version 2.0 Final)
- MOD16A2.006: Terra Net Evapotranspiration 8-Day Global 500m

Nếu bạn thực hiện tìm kiếm trong Google Earth Engine với tên tập dữ liệu của mình, một cửa sổ như dưới đây sẽ xuất hiện:

NASA-USDA Enhanced SMAP Global Soil Moisture Data

Resolution: 10000 meters

Dataset Availability: 2015-04-01T00:00:00 - 2021-11-02T00:00:00

Dataset Provider: NASA GSFC

Name	Description	Min*	Max*	Units
ssm	Surface soil moisture	0	25.39	mm
susm	Subsurface soil moisture	0	274.6	mm
smp	Soil moisture profile	0	1	fraction
ssma	Surface soil moisture anomaly	-4	4	
susma	Subsurface soil moisture anomaly	-4	4	

* = Values are estimated

Collection Snippet

```
ee.ImageCollection("NASA_USDA/HSL/SMAP10KM_soil_moisture")
```

Tags: geophysical, soil, moisture, nasa, usda, hsl, smap

CLOSE IMPORT

Tại đây bạn có thể tìm thấy rất nhiều thông tin về tập dữ liệu. Mỗi tập dữ liệu chứa các dải băng tần khác nhau được liệt kê trong cửa sổ này, cũng như các đơn vị, tỷ lệ và các thuộc tính liên quan khác. Chọn từ mỗi tập dữ liệu các dải băng tần có liên quan và lọc ImageCollection với startdate và enddate đã xác định. Bạn có thể sao chép đoạn mã bộ sưu tập từ mỗi tập dữ liệu để nhập tập dữ liệu vào tập lệnh của mình.

```
// Filter datasets for bands and period var SSM = ee.ImageCollection('NASA_USDA/HSL/SMAP10KM_soil_moisture')
.select('ssm') .filterDate(startdate, enddate);
var ET = ee.ImageCollection('MODIS/006/MOD16A2') .select('ET')
.filterDate(startdate, enddate);
var PRCP = ee.ImageCollection('UCSB-CHG/CHIRPS/DAILY')
.select('precipitation')
.filterDate(startdate, enddate);
```

Bước 6: In tập dữ liệu và kiểm tra thuộc tính của chúng.

```
// Filter datasets for bands and period print('Soil moisture dataset',SSM);
print('Evapotranspiration dataset',ET); print('Precipitation dataset',PRCP);
```

Bước 7: Trước khi có thể hiển thị dữ liệu, chúng ta cần gộp các ảnh trong ImageCollections thành một ảnh duy nhất. Chúng ta sẽ làm điều này với hàm reducer. Với hàm này, chúng ta sẽ tính giá trị trung bình của mỗi ImageCollection. Chúng ta sẽ lấy giá trị trung bình của dữ liệu độ ẩm đất và lượng bốc hơi. Hệ số chuyển đổi của tập dữ liệu bốc thoát hơi nước là 0,1 vì vậy chúng ta sẽ nhân tập dữ liệu này với 0,1. Đơn vị của lượng bốc thoát hơi là

mm/8 ngày, để đổi sang mm/ngày, chúng ta sẽ chia dữ liệu cho 8. Chúng ta sẽ vẽ biểu đồ lượng mưa trung bình hàng năm trong lưu vực. Để tạo tham số này, chúng ta sẽ cần tính tổng tất cả các ảnh trong ImageCollection và chia nó cho 3. Bởi vì, ImageCollection chứa 3 năm dữ liệu. Đừng quên chạy và lưu tập lệnh của bạn sau mỗi bước.

```
// Calculate means
var SSM_mean = SSM.reduce(ee.Reducer.mean());
var ET_mean = ET.reduce(ee.Reducer.mean()).divide(8).multiply(0.1);
var PRCP_sum = PRCP.reduce(ee.Reducer.sum()).divide(3);
```

Chúng ta muốn hiển thị các bộ dữ liệu này trong bản đồ. Để thực hiện điều này, chúng ta cần khai một bảng màu. Màu sắc được xác định bằng cách sử dụng bảng giá trị màu CSS chuẩn web (https://en.wikipedia.org/wiki/Web_colors).

Màu sắc có thể được chỉ định bằng tên (như được hiển thị trong Bước 12) hoặc các chuỗi chỉ ra sự kết hợp của màu đỏ, xanh lá cây và xanh lam. Giá trị thấp nhất ở bất kỳ vị trí nào trong ba vị trí là 00 (đại diện cho số thập phân 0), trong khi giá trị cao nhất là FF (đại diện cho số thập phân 255). Chuỗi '000000' đại diện cho màu đen, 'FFFFFF' là màu trắng, 'FF0000' là màu đỏ, '00FF00' là màu xanh lục và '0000FF' là màu xanh lam.

Nhưng chúng ta nên chọn những màu nào? Google Earth Engine đưa ra một gợi ý. Bạn có thể duyệt qua tất cả các tập dữ liệu có sẵn trong Danh mục dữ liệu: <https://developers.google.com/earth-engine/datasets/catalog/>. Ở phía dưới cùng, bạn luôn có thể tìm thấy một vài dòng mã để hiển thị trực quan dữ liệu. Trong trường hợp dưới đây, tìm kiếm 'MOD11A1.006 Terra Land Surface Temperature and Emissivity Daily Global 1km' cho kết quả sau:

Explore in Earth Engine

```
var dataset = ee.ImageCollection('MODIS/666/NOD11A1')
  .filter(ee.Filter.date('2018-01-01', '2018-05-01'));
var landSurfaceTemperature = dataset.select('LST_Day_1km');
var landSurfaceTemperatureVis = {
  min: 13000.0,
  max: 16568.0,
};
Map.setCenter(6.746, 46.529, 2);
Map.addLayer(
  landSurfaceTemperature, landSurfaceTemperatureVis,
  'land Surface Temperature');
```

Điều này rất tiện lợi, vì giờ đây bạn có thể xem cả giá trị nhỏ nhất, giá trị lớn nhất và bảng màu do Google Earth Engine đề xuất.

Bước 8: Tìm kiếm Danh mục dữ liệu GEE cho từng tập dữ liệu trong số ba tập dữ liệu mà chúng ta quan tâm. Thêm bảng màu được đề xuất vào mã của bạn.

```

// Set visualization parameters
var ETVis = {
  min: 0.0,
  max: 5.0,
  palette: [
    'ffffff', 'fcd163', '99b718', '66a000', '3e8601', '207401', '056201',
    '004c00', '011301'
  ],
};
var PRCPVis = {
  min: 1000.0,
  max: 2500.0,
  palette: ['001137', '0aab1e', 'e7eb05', 'ff4a2d', 'e90000'], };
var SSMVis = {
  min: 0.0,
  max: 25.0,
  palette: ['0300ff', '418504', 'efff07', 'efff07', 'ff0303'], };

```

Bước 9: Bây giờ chúng ta có một ảnh cho mỗi biến. Nhưng những ảnh này vẫn chứa dữ liệu trên toàn thế giới. Chúng ta chỉ quan tâm đến lưu vực sông Hồng vì vậy hãy sử dụng `.clip(Red_River)` khi thêm các biến vào bản đồ.

```

// Add variables to the map and filter on Red River
Map.addLayer(SSM_mean.clip(Red_River), SSMVis, 'SSM');
Map.addLayer(ET_mean.clip(Red_River), ETVis, 'ET'); Map.addLayer(PRCP
sum.clip(Red_River), PRCPVis, 'PRCP');

```

Bước 11: Sử dụng cú pháp `ui.Chart.image.seriesByRegion` để tính giá trị trung bình (trên lưu vực sông Hồng) và lập biểu đồ cho từng bộ dữ liệu. Hãy nhớ đặt tiêu đề thích hợp cho biểu đồ cho cả trục hoành (hAxis) và trục tung (vAxis). Đây là một ví dụ cho tập dữ liệu LST:

```

// Define graphs
var SSMGraph = ui.Chart.image.seriesByRegion(SSM, Red_River,
ee.Reducer.mean()).setOptions({
  title: 'Average soil moisture in the Red River basin', hAxis: {title: 'Date'},
  vAxis: {title: 'Soil moisture'}});
var ETGraph = ui.Chart.image.seriesByRegion(ET, Red_River,
ee.Reducer.mean()).setOptions({
  title: 'Average evapotranspiration in the Red River basin',
  hAxis: {title: 'Date'}, vAxis: {title: 'Evapotranspiration'}});
var PRCPGraph = ui.Chart.image.seriesByRegion(PRCP, Red_River,
ee.Reducer.mean()).setOptions({
  title: 'Average precipitation in the Red River basin', hAxis: {title: 'Date'},
  vAxis: {title: 'Precipitation'}});

```

```
// Display graphs print(SSMGraph); print(ETGraph); print(PRCPCGraph);
```

3.5. Phân tích hạn hán với SPI

Chỉ số Lượng mưa Chuẩn hóa (SPI) là một chỉ số để đặc trưng hóa hạn hán khí tượng. Hạn hán khí tượng được xác định dựa trên mức độ khô hạn (lượng mưa thực tế so với lượng mưa chuẩn) và độ dài của thời kỳ khô hạn. Hạn hán nông nghiệp mô tả tác động đối với nông nghiệp dựa trên các khía cạnh khí tượng và thủy văn như thiếu hụt lượng mưa, độ ẩm đất, sự khác biệt giữa lượng bốc hơi thực tế và tiềm năng và sự sụt giảm mực nước ngầm. Ngoài ra còn có các định nghĩa về hạn hán thủy văn, hạn hán kinh tế xã hội và hạn hán sinh thái. Bạn có thể tìm thêm thông tin và ứng dụng về tất cả các loại hạn hán trên internet.

SPI so sánh lượng mưa thực tế với lượng mưa chuẩn. Lượng mưa chuẩn được xác định bằng cách tính toán phân bố điển hình của lượng mưa trong năm từ một chuỗi dữ liệu dài (hơn 30 năm). Lượng mưa thực tế sẽ được so sánh với lượng mưa phân bố chuẩn. Giá trị SPI lớn hơn 1,0 cho biết một thời kỳ ẩm ướt hơn bình thường và giá trị nhỏ hơn -1,0 cho biết thời kỳ khô hơn nhiều so với bình thường. Tất cả các giá trị từ -0,99 đến 0,99 cho thấy lượng mưa trong khoảng bình thường. Bạn có thể tìm thêm thông tin về chỉ số này qua trang web này: <https://climatedataguide.ucar.edu/climate-data/standardized-precipitation-index-spi>

Nếu bạn tìm kiếm SPI trong Google, bạn có thể tìm thấy các tập lệnh mã nguồn mở để tính toán chỉ số này. Chúng ta sẽ sử dụng tập lệnh này: <https://code.earthengine.google.com/3f41f3d15e8cd7a16d7d923c3bc8831a?noload=true>

Hãy cố gắng thay đổi khu vực nghiên cứu trong tập lệnh này thành lưu vực sông Hồng hoặc khu vực bạn quan tâm. Bạn thấy sao? Nó có phù hợp với kỳ vọng của bạn không?

Bạn có thể tìm thêm thông tin về tập lệnh và cách sử dụng tại đây: <https://www.un-spider.org/advisory-support/recommended-practices/recommended-practice-drought-monitoring-spi/step-by-step#:~:text=The%20SPI%20is%20used%20as,on%20two%20independent%20SPI%20calculations.>

Bạn có thể tìm thấy cú pháp đầy đủ của bài tập này tại đây: <https://code.earthengine.google.com/c3f98793857459072961aea813fcc485?noload=true>

3.6. Lượng mưa trên lưu vực sông Hồng

Trong bài tập này, chúng ta sẽ đánh giá lượng mưa trên lưu vực sông Hồng và so sánh nó với sự bốc thoát hơi nước để phân tích tổng lượng nước hiệu quả của lưu vực.

Bước 1: Tạo một tập lệnh mới, đặt tên là “WP3_Wateryield” và nhập các ranh giới sông Hồng.

```
//Import the boundaries of the Red River basin
var watershed
=ee.FeatureCollection("WWF/HydroSHEDS/v1/Basins/hybas_4");
var Red_River = watershed.filter(ee.Filter.eq('HYBAS_ID',
4040015000));
// Add Red River basin and center map Map.addLayer(Red_River);
Map.centerObject(Red_River);
```

Bước 2: Chúng ta quan tâm đến dữ liệu từ năm 2018 đến năm 2020, nhưng hãy viết mã để có thể dễ dàng thay đổi khoảng thời gian quan tâm sau này. Chúng ta có thể làm điều này bằng cách đặt năm và tháng bắt đầu và kết thúc thành các biến riêng biệt:

```
// Set period
var startyear = 2018;
var endyear = 2020;
var startmonth = 1;
var endmonth = 12;
var startdate = ee.Date.fromYMD(startyear,startmonth,1);
var enddate = ee.Date.fromYMD(endyear,endmonth,31);
```

Bước 3: Nhập bộ dữ liệu lượng mưa ngày CHIRPS và bộ dữ liệu bốc thoát hơi nước MODIS Terra (giống như trong Bài tập 1 của Gói đào tạo này).

```
// Filter datasets for bands and period
var ET = ee.ImageCollection('MODIS/006/MOD16A2') .select('ET')
.filterDate(startdate, enddate);
var PRCP = ee.ImageCollection('UCSB-CHG/CHIRPS/DAILY')
.select('precipitation') .filterDate(startdate, enddate);
```

Bước 4: Chúng ta sẽ xác định lượng bốc thoát hơi và lượng mưa năm trong lưu vực. Chúng ta sẽ sử dụng hàm .sum và sẽ nhân tập dữ liệu bốc thoát hơi với 0,1 vì đây là hệ số chuyển đổi của tập dữ liệu này.

```
// Calculate sum and multiply ET with 0.1 var ET_sum =
ET.reduce(ee.Reducer.sum()).multiply(0.1).divide(3);
var PRCP sum = PRCP.reduce(ee.Reducer.sum()).divide(3);
```

Bước 5: Bây giờ chúng ta sẽ hiển thị bản đồ lượng mưa. Sao chép các thông số hiển thị mà bạn đã sử dụng để hiển thị trực quan lượng mưa CHIRPS trong các bài tập trước đó. Chạy tập lệnh và kiểm tra kết quả. Bây giờ thay đổi cài đặt lượng mưa nhỏ nhất và lớn nhất liên quan đến màu đỏ và xanh lam cho đến khi bạn hài lòng với bản đồ kết quả. Đừng quên chạy và lưu tập lệnh của bạn sau mỗi bước.

```
// Set visualization parameters
var ETVis = {
min: 0.0,
max: 1000.0,
palette: [
```



```

'ffffff', 'fcd163', '99b718', '66a000', '3e8601', '207401', '056201',
'004c00', '011301'
],
};
var PRCPVis = {
min: 1000,
max: 2500.0,
palette: ['001137', '0aab1e', 'e7eb05', 'ff4a2d', 'e90000'], };
// Add variables to the map and filter on Red River
Map.addLayer(ET_sum.clip(Red_River), ETVis, 'ET'); Map.addLayer(PRCP
sum.clip(Red River), PRCPVis, 'PRCP');

```

Bằng cách trừ lượng mưa bằng lượng nước bốc thoát hơi, chúng ta có thể tính được “lượng nước hiệu quả”: lượng nước thấm vào đất hoặc trở thành dòng chảy bề mặt.

Bước 6: Thêm các dòng mã sau để trừ lượng thoát hơi nước hàng năm khỏi lượng mưa và lập bản đồ lượng nước hiệu quả. Lưu ý rằng lượng nước hiệu quả trung bình của lưu vực được in trong Bảng điều khiển là “Zonal mean P-ET”.

```

// Calculate P - ET
var water_yield = PRCP_sum.subtract(ET_sum);
var wateryieldVis = {
min: 0.0,
max: 3000.0,
palette: ['001137', '0aab1e', 'e7eb05', 'ff4a2d', 'e90000'],
};
Map.addLayer(water_yield.clip(Red_River), wateryieldVis, 'Water
yield');
// Get P-ET statistics
var ZonalMeanWY =
water_yield.reduceRegion(ee.Reducer.mean(),Red_River,5000); print("Zonal
mean P-ET", ZonalMeanWY);

```

Bước 7: Bây giờ chúng ta sẽ tạo một biểu đồ lượng nước hiệu quả theo thời gian. Chúng ta sẽ sử dụng một hàm tổng hợp dữ liệu lượng mưa và lượng bốc thoát hơi thành tổng hàng tháng. Chúng ta sẽ lấy tổng lượng mưa hàng tháng trừ đi tổng lượng nước bốc thoát hơi hàng tháng để có được lượng nước hiệu quả hàng tháng. Cuối cùng, chúng ta vẽ các giá trị đó trong biểu đồ.

```

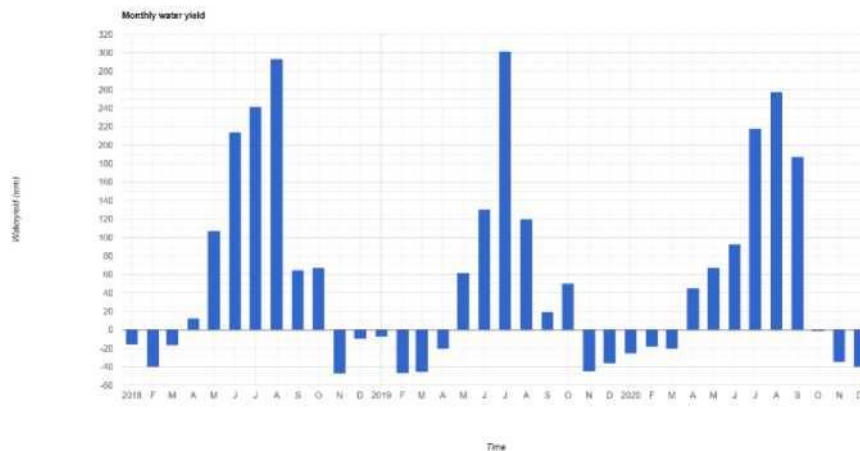
// Water yield graph
// calculate the et for each month
var months = ee.List.sequence(startmonth,endmonth);
var years = ee.List.sequence(startyear,endyear);
var allET = ee.ImageCollection.fromImages( years.map(function (y) {
return months.map(function(m){

```

```

    var w = ET.filter(ee.Filter.calendarRange(y, y, 'year'))
    .filter(ee.Filter.calendarRange(m, m, 'month')) .sum();
    return w.set('year', y) .set('month', m) .set('date',
ee.Date.fromYMD(y,m,1)) .set('system:time_start',ee.Date.fromYMD(y,m,1));
  });
  }).flatten());
  // calculate the P for each month
  var allP = ee.ImageCollection.fromImages( years.map(function (y) {
return months.map(function(m){
  var w = PRCP.filter(ee.Filter.calendarRange(y, y, 'year'))
.filter(ee.Filter.calendarRange(m, m, 'month')) .sum();
  return w.set('year', y) .set('month', m) .set('date',
ee.Date.fromYMD(y,m,1)) .set('system:time_start',ee.Date.fromYMD(y,m,1));
  });
  }).flatten());
  // Combine the two datasets and calculate monthly water yield var
P_ET_combi = allP.combine(allET);
  P_ET_combi = P_ET_combi.map(function(img){
  var myimg = img.expression("(p - et) ",
  {
  p: img.select("precipitation"),
  et: img.select("ET").multiply(0.1),
  });
  return img.addBands(myimg.rename('P-ET'));
  });
  // setup the chart var title = {
  title: 'Monthly water yield',
  hAxis: {title: 'Time'},
  vAxis: {title: 'Wateryield (mm)'},
  };
  var chart = ui.Chart.image.seriesByRegion(P_ET_combi, Red_River,
ee.Reducer.mean(), 'P-ET', 2500, 'system:time_start',
'SITE')
.setOptions(title) .setChartType('ColumnChart');
  // plot the chart print(chart);
  Biểu đồ này sẽ giống như thế này:

```



Câu hỏi nhanh 3.5

Cú pháp đầy đủ của bài tập này có thể được tìm thấy tại đây:

<https://code.earthengine.google.com/d8045830b61df1d6cabd3515d0bf6a4>

c? noload=true

3.6. Hình thái sông ngòi

Việt Nam có một số sông và hệ thống đồng bằng lớn. Trong phần đầu tiên của gói huấn luyện này, chúng ta sẽ tập trung vào hình thái sông ngòi và nguyên nhân của các biến đổi hình thái.

Aqua Monitor

Nền tảng Aqua Monitor cung cấp cái nhìn tổng quan về sự thay đổi của các thủy vực bề mặt Trái đất trong 30 năm qua.

Bước 1. Truy cập http://aqua-monitor.appspot.com/?from=2000&to=2017&view=17.21179758232761,95.72558832305796,11z&max_doy=365

Màu xanh lục thể hiện nơi nước mặt đã trở thành đất (bồi lắng). Màu xanh lam ngược lại, cho biết nơi đất đã trở thành nước mặt (xói mòn).

xuất hiện trên màn hình của bạn.

Với thanh thời gian ở phía dưới cùng, bạn có thể chọn khoảng thời gian mà bạn muốn xem sự thay đổi của nước bề mặt. Bằng cách di chuyển ba thanh trượt ở góc dưới cùng bên phải, bạn có thể thấy các thay đổi hoặc ảnh vệ tinh của hai năm (năm bắt đầu và năm gần nhất) bạn đã chọn trên thanh thời gian.

Khi thanh trượt hết cỡ sang bên phải có nghĩa là đang "bật". Nếu bạn muốn xem các thay đổi trong khoảng thời gian đã chọn, bạn phải di chuyển thanh trượt 'Changes' sang bên phải. Khi đó bạn có thể thấy các màu xanh lục và xanh lam biểu thị những thay đổi của nước và bề mặt đất. Lưu ý rằng thanh trượt cao nhất trong ba thanh trượt "bật" sẽ là thanh hiển thị trên màn hình. Vì vậy, nếu thanh trượt 'Changes' được bật, nó sẽ là thanh được hiển thị. Nếu nó "tắt", ảnh vệ tinh của năm gần nhất được chọn sẽ được hiển thị. Và nếu nó cũng bị "tắt" thì ảnh vệ tinh của năm bắt đầu sẽ được hiển thị.

Bước 2. Phóng to Đồng bằng sông Hồng (Việt Nam) (bạn có thể tìm

Đồng bằng sông Hồng trên thanh tìm kiếm).

- Đặt thanh thời gian ở phía dưới cùng từ năm 1987 đến năm 2017 (30 năm)

- Xem ảnh vệ tinh năm 1987

- Xem ảnh vệ tinh năm 2017

- Kiểm tra sự thay đổi của hình thái sông ở khu vực này.

Bước 3. Bây giờ hãy di chuyển bản đồ về phía Tây đến Bangladesh (hoặc sử dụng thanh tìm kiếm). Ở đây bạn có thể thấy một mô hình sông phân dòng (braided river) rất đẹp, đã thật sự thay đổi rất nhiều theo thời gian.

3.7. Xác định sự thay đổi hình thái sông ngòi

Để hiểu tại sao sông Hồng không thay đổi hình dạng đáng kể trong 30 năm qua, chúng ta sẽ phải xem xét các yếu tố gây ra điều này. Trong bài tập này, chúng ta sẽ xét đến độ cao, độ dốc và lớp phủ bề mặt của khu vực.

Để bắt đầu, chúng ta sẽ căn giữa bản đồ đến Đồng bằng sông Hồng (khu vực quan tâm).

Bước 1. Đầu tiên hãy tạo một tập lệnh mới gọi là: 'WP4_Understanding_River_behaviour'.

Căn giữa bản đồ

Bạn có thể sẽ nhận thấy chế độ xem bản đồ ở nửa dưới của cửa sổ. Rất có thể, trong màn hình của bạn, nó hiện đang được căn giữa ở Hoa Kỳ. Chúng ta có thể căn giữa nó vào một vị trí khác.

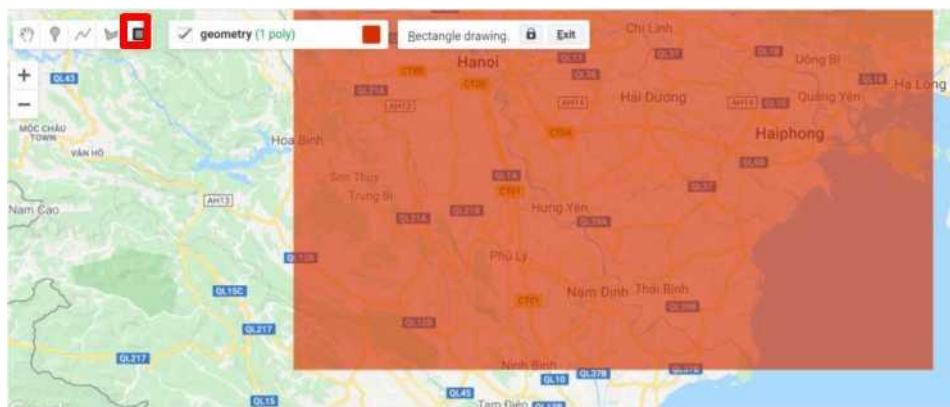
Bước 2. Nhập cú pháp sau vào tập lệnh và nhấp vào 'Run'

```
// Zoom to a location.
```

```
Map.setCenter(106.29, 20.89, 9); // Center on Red River Delta
```



Bước 3. Bây giờ chúng ta sẽ xác định khu vực chúng ta quan tâm. Thực hiện bằng cách sử dụng công cụ 'Draw a Rectangle' ở góc trên cùng bên trái khu vực bản đồ của bạn.



Bước 4. Vẽ hình chữ nhật xung quanh Đồng bằng sông Hồng. Khi thả chuột, bạn sẽ thấy rằng một biến geometry đã xuất hiện ở đầu màn hình trong mục 'Imports'.

```
t Imports (1 entry) s
```

```
▶ var geometry: Polygon, 4 vertices 0 ©
```

Đổi tên biến này thành AOI bằng cách nhấp vào từ geometry.

3.8. Bản đồ độ cao + độ dốc

Trong bài tập này chúng ta sẽ phân tích số liệu độ cao của khu vực đồng bằng sông Hồng. Dữ liệu độ cao mà chúng ta sử dụng thực tế cũng chính là tập dữ liệu mà bạn đã sử dụng trong Gói đào tạo số 1. Vì vậy, nếu bạn đã biết cách chèn tập dữ liệu này vào tập lệnh của mình, vui lòng thực hiện việc này theo cách của bạn.

Bây giờ chúng ta sẽ thêm dữ liệu Độ cao vào bản đồ. Đối với mục tiêu này, chúng ta sẽ sử dụng dữ liệu SRTM. SRTM có nghĩa là Sứ mệnh Địa hình Radar của tàu Con thoi (Shuttle Radar Topography Mission). Đây là một dự án của NASA, ở đó họ tạo ra dữ liệu độ cao từ các vệ tinh radar. Bạn có thể sử dụng dữ liệu vệ tinh SRTM để tạo và xem bản đồ độ cao.

Bước 1. Để tải dữ liệu, bạn thêm cú pháp sau vào tập lệnh và nhấp vào 'Run':

```
// Define an image using the Image constructor.
```

```
var SRTM = ee.Image('CGIAR/SRTM90 V4');
```

Bước 2. Chúng ta muốn xem thông tin bên trong tập dữ liệu SRTM. Để làm như vậy, chúng ta sẽ in tập dữ liệu ra Bảng điều khiển

print('SRTM image', SRTM) Khi bạn chạy tập lệnh, hãy lưu ý một đối tượng xuất hiện trong Bảng điều khiển. Để điều tra các thuộc tính của đối tượng, hãy mở rộng đối tượng bằng cách nhấp vào hình tam giác bên trái đối tượng. Mở rộng đối tượng hình ảnh, thuộc tính 'bands', băng tần 'elevation' tại chỉ mục '0' và thuộc tính 'data_type' của băng tần 'elevation' để thấy rằng đó là kiểu dữ liệu int16 có dấu.

Inspector uglWjXJ Tasks

Use print(...) to write to this console.

SRTM image

JSON

'Image

CGIAR/SRTM90_V4

(1

band)

DSO

N

```
type: Image
```

```
id: CGIAR/SRTM90_V4
```

```
version: 1590671702209836
```

```
'bands: List (1 element)
```

```
▶ 0: "elevation", signed int16, EPSG:4326, 432000...
```

```
▶ properties: object (24 properties)
```

Bước 3. Trong toàn bộ tập dữ liệu SRTM này, chúng ta chỉ cần quan tâm

một phần nhỏ, đó là khu vực xung quanh đồng bằng sông Hồng. Để làm như vậy, chúng ta giới hạn hình ảnh tại khu vực chúng ta quan tâm:

```
// Clip the digital elevation model to our area of interest
var SRTM_AOI = SRTM.clip(AOI);
```

Bước 4. Bạn có thể hiển thị dữ liệu này bằng cách thêm nó thành một lớp mới trên bản đồ. Bằng cách thêm giá trị tối thiểu và tối đa cho vùng trưng này của Việt Nam, bản đồ sẽ hiển thị nhiều thông tin hơn. Hãy thử nghiệm với các giá trị tối thiểu và tối đa khác nhau để hiển thị nhiều chi tiết nhất có thể. Thêm cú pháp bên dưới vào tập lệnh và nhấp vào 'Run':

```
// Display the image on the map.
Map.addLayer(SRTM_AOI, {min: 0, max: 200, palette: ['blue','cyan',
'white']}, 'SRTM_AOI');
```

Lưu và chạy tập lệnh.

Bài thuyết trình giải thích rằng độ dốc cũng có thể có ảnh hưởng lớn đến động lực của sông. Do đó, chúng ta cũng sẽ xác định độ dốc của khu vực này.

Bước 5. Bằng cách sử dụng thuật toán Địa hình, chúng ta có thể xác định độ dốc (tính bằng độ) của khu vực quan tâm.

```
// Slope
var terrain = ee.Algorithms.Terrain(SRTM_AOI); var slope =
terrain.select('slope');
Map.addLayer(slope,{min: 0, max: 15},'slope');
```

Lưu và chạy tập lệnh.

Trong bài thực hành này, chúng ta đã xem xét các số liệu về độ cao ở Đồng bằng sông Hồng.

Đừng quên lưu tập lệnh của bạn! Cú pháp đầy đủ có thể được tìm thấy ở đây: <https://code.earthengine.google.com/aa179bb3a18408c69b9cf9513ae2dc69>

3.9. Bản đồ thảm phủ

Nhiều nguồn dữ liệu khác nhau có sẵn trong Google Earth Engine cung cấp thông tin chi tiết về sử dụng đất của một khu vực bất kỳ. Trong phần này, chúng ta sẽ bắt đầu với việc khai thác và hiển thị dữ liệu này trên bản đồ.

Nhìn chung, Google Earth Engine có dữ liệu cơ bản trên toàn thế giới có thể được sử dụng kết hợp với dữ liệu vệ tinh. Nếu bạn muốn sử dụng một nguồn dữ liệu khác với nguồn mà chúng ta sử dụng ở đây, hãy thử tra cứu nó trong công cụ tìm kiếm Google Earth Engine và xem nó có sẵn hay không.

Bây giờ chúng ta sẽ bắt đầu với việc tạo bản đồ với thông tin về lớp phủ và điều kiện loại đất. Trong Google Earth Engine, bản đồ thảm phủ của MODIS từ NASA có sẵn từ năm 2001 đến năm 2019. Nếu bạn muốn biết thêm về nguồn dữ liệu này. Hãy xem tại: <https://lpdaac.usgs.gov/products/mcd12q1v006/>.

Mọi nguồn dữ liệu trong Google Earth Engine đều chứa tệp siêu dữ liệu miêu tả thông tin dữ liệu như độ phân giải, cách sử dụng, loại thông tin dữ liệu chứa đựng và ví dụ như bảng màu được chọn trước: <https://developers.google.com/earth->

engine/datasets/catalog/ESA_GLOBCOVER_L4_200901_200912_V2_3

Bước 6. Tiếp tục tập lệnh của bạn trong bài tập trước và thêm thông tin về thảm phủ của năm 2019 bằng cách sử dụng tập cú pháp bên dưới

```
//Add land cover map //2019
var dataset = ee.Image('MODIS/006/MCD12Q1/2019_01_01'); var
igbpLandCover_2019 = dataset.select('LC_Type1');
// clip the Land cover map
var RedRiver_lc_2019 = igbpLandCover_2019.clip(AOI);
```

Bây giờ bạn đã thêm bản đồ thảm phủ vào tập lệnh của mình nhưng vẫn chưa thêm hình ảnh vào bản đồ. Nếu bạn muốn biết màu sắc cần sử dụng, danh mục dữ liệu có thể rất thuận tiện. Nó có thể được tìm thấy ở đây: <https://developers.google.com/earth-engine/datasets/catalog/>

Bước 7. Nếu bạn nhập 'MODIS land cover' vào thanh tìm kiếm trên trang web Danh mục dữ liệu của Google Earth Engine (liên kết ở trên), bạn có thể tìm thấy tập dữ liệu mà chúng ta đang sử dụng. Khi bạn nhấp vào nó, bạn có thể tìm thấy một đoạn mã ở dưới cùng có thể được sử dụng để hiển thị dữ liệu thảm phủ này lên bản đồ. Bạn có thể nhận ra một vài dòng đầu tiên chúng ta đã sử dụng ở bước trước. Vì vậy, chúng ta chỉ cần sao chép cú pháp 'var igbpLandCoverVis' và 'Map.addLayer' vì chúng ta đã căn bản đồ về Việt Nam trước đó. Hãy thêm cú pháp này vào tập lệnh của bạn.



```
// visualization land cover var igbpLandCoverVis = { min: 1.0, max: 17.0,
palette: [
],
};
// show layer on the map
Map.addLayer(RedRiver_lc_2019, igbpLandCoverVis, 'IGBP Land
Cover');
```

Danh mục dữ liệu tỏ ra rất hữu ích khi bạn muốn trực quan hóa tập dữ liệu một cách nhanh chóng.

Bước 8. Bây giờ chúng ta sẽ thêm chú giải vào bản đồ để có thể giải thích

bản đồ thảm phủ. Bạn có thể sử dụng cú pháp trong liên kết này cho việc đó: <https://code.earthengine.google.com/c32097aaa64a20669898d3f1c219f981>

3.10. Điều kiện loại đất

Vì chúng ta vừa tải dữ liệu lớp phủ, nên cũng có thể xem dữ liệu về hàm lượng các loại đất của lớp đất dưới bề mặt. Điều này cung cấp thông tin tổng quan về loại đất dưới bề mặt của khu vực. Bạn có thể sử dụng thông tin này cho các phân tích về hạn hán hoặc nông nghiệp. Xin lưu ý, dữ liệu có thể không chính xác nhưng nó cung cấp các dấu hiệu ban đầu về hàm lượng đất sét của lớp đất dưới bề mặt.

Bước 9. Với cú pháp bên dưới, bạn có thể tải thông tin về hàm lượng loại đất. Trong trường hợp này, chúng ta đã đưa ra một ví dụ với hàm lượng đất sét.

Câu hỏi 7: Hàm lượng đất sét được hiển thị với cú pháp bên dưới ở độ sâu nào?

```
////soil clay////
//import, filter and display
var dataset = ee.Image("OpenLandMap/SOL/SOL_CLAY-
WFRATION_USDA- 3A1A1A_M/v02");
var RedRiver_clay = dataset.clip(AOI); //Band of interest
var clay_band = 'b100';
var visualization = { bands: [clay_band], min: 0.0, max: 50.0, palette: [
"FFFF00", "FFA500", "FF0000" ]
};
Map.addLayer(RedRiver_clay, visualization, 'Clay content');
```

Hình ảnh cho ra màu cam. Nếu bạn google các màu trong bảng màu, bạn có thể thấy điều này cho thấy hàm lượng đất sét cao (% kg/kg).

Đừng quên lưu tập lệnh của bạn! Cú pháp đầy đủ có thể được tìm thấy ở đây: <https://code.earthengine.google.com/f35be741a6bd7601d840282f9847be3a>

3.11.

Phần thứ hai của Gói đào tạo này sẽ giới thiệu các khái niệm về rủi ro lũ lụt và cách phân tích bằng cách sử dụng Google Earth Engine, dựa trên khái niệm rủi ro = tần suất xuất hiện x tác động. Do đó, rủi ro lũ lụt không chỉ phụ thuộc vào khả năng lũ lụt xảy ra tại một địa điểm nhất định mà còn phụ thuộc vào tác động của lũ lên một khu vực cụ thể dưới hình thức hoạt động kinh tế (lớp phủ đất) và khả năng thiệt hại về nhân mạng (dân số). Bên cạnh đó, gói đào tạo này cũng sẽ tập trung vào các nhóm người dễ bị tổn thương trong một trận lũ lụt như người già, trẻ nhỏ và bệnh nhân trong bệnh viện. Họ có thể phải chịu nhiều thiệt hại trong trường hợp có lũ lụt hơn những đối tượng khác.

Nguồn: <https://lapaas.com/how-to-learn-risk-impact-and-probability-chart/>

Hướng dẫn này sẽ mô tả một số thao tác cơ bản theo các bước ngắn gọn.

Trong trường hợp bạn muốn biết thêm các thông tin khác, ví dụ: về các chức năng không được đề cập trong hướng dẫn, vui lòng tham khảo:

<https://developers.google.com/earth-engine/playground>
hoặc đặt câu hỏi cho người hướng dẫn.

Tập dữ liệu JRC sẽ xuất hiện ở đầu tập lệnh của bạn.

Bước 2. Tập dữ liệu có tên là 'ImageCollection', hãy thay đổi tập dữ liệu này thành 'jrc'.

► var jrc: Image "JRC Global Surface Water Mapping Layers, v1.3" (7 bands) c

Chúng ta không cần phải phân tích tần suất xuất hiện của lũ lụt trên toàn thế giới. Chúng ta chỉ quan tâm đến Việt Nam. Do đó, chúng ta sẽ lọc bộ dữ liệu JRC ở Việt Nam. Điều này cũng sẽ tiết kiệm thời gian tính toán.

Bước 4. Đổi tên bảng đã nhập từ "table" thành "country" bằng cách nhấp vào từ "table" ở đầu màn hình của bạn.

danhsach '500m: Global Administrative Unit Layers'.



► var [Table] Table [FAO/GAUL SIHPLIFIED 500I1/2015/level2](#)

Bước 5. Chúng ta sẽ chọn quốc gia Việt Nam từ bộ dữ liệu FAO GAUL. Bạn có thể xem phần 'Bands' của tập dữ liệu FAO GAUL khi nhập nó vào thanh tìm kiếm để thấy rằng băng tần "ADMO_NAME" đại diện cho Tên quốc gia theo Liên hợp quốc mà tập dữ liệu đó chứa. Chúng ta cần sử dụng "Viet Nam".

```
// Filter by country
```

```
var Vietnam = countries.filter(ee.Filter.eq('ADMO_NAME', 'Viet Nam'));
```

Bước 6. Bởi vì chúng ta không muốn kéo bản đồ của mình đến Việt Nam mỗi khi làm việc với tập lệnh này, chúng ta sẽ căn giữa đối tượng vào Việt Nam (mà chúng ta vừa xác định ở bước 5).

```
// Centre on Vietnam
```

```
Map.centerObject(Vietnam,6);
```

Bước 7. Chúng ta cũng sẽ xác định ngày bắt đầu và ngày kết thúc bằng

cách tạo biến startDate và endDate mà sau này sẽ sử dụng cho tập dữ liệu JRC.

```
// Define study period
var startDate = ee.Date.fromYMD(2000, 1, 1); var endDate =
ee.Date.fromYMD(2021, 1, 1);
```

Bước 8. Lọc dữ liệu JRC theo ngày bắt đầu và ngày kết thúc và quốc gia Việt Nam mà bạn đã xác định ở Bước 5.

```
// Filter JRC on Vietnam and period
var myjrc = jrc.filterBounds(Vietnam).filterDate(startDate, endDate);
```

Kiểm tra dữ liệu của bạn là rất quan trọng trong mọi nghiên cứu và cả khi làm việc với Google Earth Engine.

Bước 9. Kiểm tra tính đầy đủ của dữ liệu JRC mà bạn đang sử dụng bằng cách sử dụng chức năng *print*.

```
//Print myjrc dataset
```

```
print(myjrc);
```

Trong bảng điều khiển bên phải, bạn sẽ thấy trong 'features' tất cả các tháng của khu vực giữa ngày bắt đầu và ngày kết thúc mà bạn đã chọn.

Chúng ta muốn xây dựng một bản đồ tần suất lũ. Để làm như vậy, chúng ta cần biết tần suất có nước của mỗi ô lưới trong giai đoạn nghiên cứu. Chúng ta có thể làm điều này bằng cách xác định tần suất nước được phát hiện trong một ô lưới nhất định chia cho số lần quan sát ở đó. Nếu bạn nhân con số này với 100, bạn sẽ nhận được phần trăm thời gian mà nước có trong ô lưới.

Bước 10. Để tìm ra cách thực hiện điều này có thể là một thách thức lúc đầu. Do đó, chúng tôi cung cấp cho bạn thông tin đầu tiên, nó được mô tả bên dưới.

```
// Detect observations
var myjrc = myjrc.map(function(img){
// observation is img > 0
var obs = img.gt(0);
return img.addBands(obs.rename('obs').set('system:time_start',
img.get('system:time_start')));
});
```

```
// Calculate the total amount of observations
```

```
var totalObs =
```

```
ee.Image(ee.ImageCollection(myjrc.select("obs")).sum().toFloat());
```

Bây giờ bạn đã xác định tổng số quan sát.

Bước 11. Bạn có thể tự mình nghĩ ra cách xác định 'only water' trong ô lưới không? Sau đó, bạn sẽ phải chia giá trị này cho totalObs và nhân nó với 100 để có phần trăm.

```
// Detect all observations with water
```

```
var myjrc = myjrc.map(function(img){
```

```
// if water
```

```
var water = img.select('water').eq(2);
```

```

return img.addBands(water.rename('onlywater').set('system:time_start',
img.get('system:time_start')));
});
// Calculate all observations with water
var totalWater =
ee.Image(ee.ImageCollection(myjrc.select("onlywater")).sum().toFloat
());
// Calculate the percentage of observations with water
var floodfreq = totalWater.divide(totalObs).multiply(100);

```

Bây giờ chúng ta muốn che những khu vực chưa bao giờ có nước, chúng ta không muốn nhìn thấy những khu vực đó trên bản đồ, do nguy cơ lũ lụt rất thấp.

Bước 12. Che những khu vực chưa bao giờ có nước trong thời gian nghiên cứu.

```

// Mask areas that are not water
var myMask = floodfreq.eq(0).not();
var floodfreq_select = floodfreq.updateMask(myMask);

```

Bước 13. Thêm bảng màu và cắt hình ảnh theo khu vực quan tâm (Việt Nam) để thêm lớp vào bản đồ

```

// Add a palette var viz = { min:0, max:90,
palette:[ 'FFFFFF','008000','001556','b7d2f7']};
Map.addLayer(floodfreq_select.clip(Vietnam),viz,"Flood frequency");

```

Bước 14. Thêm một chú giải đi kèm với bản đồ tần suất lũ. Các cú pháp này

hơi dài nên có thể được sao chép từ ô bên dưới.

```

// set position of panel var legend = ui.Panel({ style: {
position: 'bottom-left', padding: '8px 15px'
}
});

```

// Create legend title

```

var legendTitle = ui.Label({ value: 'Floodfrequency 2000 - 2020', style: {
fontWeight: 'bold', fontSize: '18px', margin: '0 0 4px 0', padding: '0' }
});

```

// Add the title to the panel legend.add(legendTitle);

// Creates and styles 1 row of the legend.

```

var makeRow = function(color, name) {

```

```

// Create the label that is actually the colored box. var colorBox =
ui.Label({
style: { backgroundColor: '#' + color, // Use padding to give the box
height and width. padding: '8px', margin: '0 0 4px 0'
}
});

```

```
// Create the label filled with the description text. var description =
ui.Label({
  value: name,
  style: {margin: '0 0 4px 6px'}
});
// return the panel return ui.Panel({ widgets: [colorBox, description],
layout: ui.Panel.Layout.Flow('horizontal')
});
};
```

```
// Palette with the colors
var palette = ['FFFFFF','008000','001556','b7d2f7'];
// name of the legend
var names = ['0%', '30%', '60%', '90%'];
// Add color and names for (var i = 0; i < 4; i++) {
legend.add(makeRow(palette[i], names[i]));
}
// add legend to map (alternatively you can also print the legend to the
console)
Map.add(legend);
```



Đừng quên lưu tệp lệnh của bạn. Cú pháp đầy đủ có thể được tìm thấy ở đây: <https://code.earthengine.google.com/da4a9370d7b41e7869e071b4a8d4af37>

3.11. Đánh giá tác động đến các cơ sở y tế

Rủi ro không chỉ liên quan đến khả năng xảy ra một điều gì đó mà còn là tác động của sự kiện đó. Để ước tính các vị trí có rủi ro cao nhất, tốt nhất là bạn phải biết nơi có nhiều người nhất sẽ phải gánh chịu hậu quả của lũ lụt. Đối với điều này, chúng ta sẽ sử dụng dữ liệu dân số có sẵn trong Google Earth Engine.

4.2.1 Dữ liệu dân số

Bắt đầu với tập lệnh mà bạn đã thực hiện trong bài tập 3 để có thể kết hợp dữ liệu dân số và tần suất lũ.

Bước 1. Trong thanh tìm kiếm ở trên cùng của màn hình, hãy nhập 'GHSL

population'. Chọn kết quả xuất hiện trên cùng với 'population grid'.

GHSL population

I GHSL: Global Human Settlement Layers, Population Grid 1975-1990-2000-2014 (P2016) I¹

GHSL: Global Human Settlement Layers, Settlement Grid 1975-1990-2000-2014 (P2016)

Trong cửa sổ hiện lên, hãy xem qua phần 'Bands'. Bảng tần duy nhất là

'population_count' mà chúng ta sẽ sử dụng.

Bước 2. Sao chép đoạn mã bộ sưu tập ở dưới cùng bên trái của cửa sổ hiện lên. Tạo một biến gọi là 'population_2015' và thêm đoạn mã của bộ sưu tập vào đó. Lọc biến theo dữ liệu có sẵn gần đây nhất (2015) bằng cách đặt ngày bắt đầu và ngày kết thúc.

```
// Add population to make a flood risk map
//      select      population      var      population_2015      =
ee.ImageCollection("JRC/GHSL/P2016/POP_GPW_GLOBE_V1")
.filter(ee.Filter.date('2015-01-01', '2015-12-31'));
```

Bước 3. Nếu bạn in biến 'population_2015' này, bạn sẽ thấy rằng nó thực sự chỉ chứa một ảnh. Tuy nhiên, vì GEE xem nó như một bộ sưu tập hình ảnh (nhiều hình ảnh) nên chúng ta sẽ phải chọn ảnh đầu tiên này. Bên cạnh đó, hãy chọn băng tần cần thiết ('population_count') và giới hạn dữ liệu về Việt Nam.

```
var population_Count_2015 =
population_2015.select('population_count').first().clip(Vietnam);
```

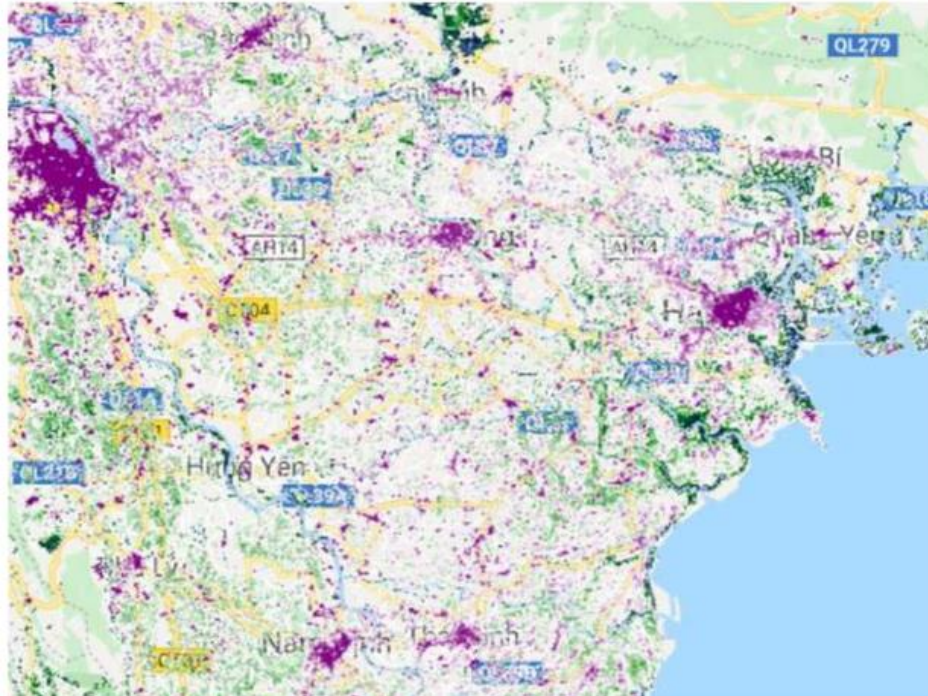
Bước 4. Bây giờ chúng ta sẽ tạo một biến để tô màu dân số trên bản đồ. Chúng ta cũng nên cung cấp các giá trị tối thiểu và tối đa.

```
var vispar_population = { min: 10, max: 1000, palette: [
"white", "purple" ] };
```

Câu hỏi 10: Các giá trị tối thiểu và tối đa này có nghĩa là gì?

Bước 5. Cuối cùng nhưng không kém phần quan trọng, chúng ta sẽ phải thêm lớp vào bản đồ. Bởi vì chúng ta không quan tâm đến bất kỳ khu vực nào có dưới 10 người trên mỗi ô lưới 250x250 m, chúng ta sẽ chỉ hiển thị các giá trị cao hơn 10.

```
// add the population layer to the map without showing cells with
//      lower      than      10      people      per      cell
Map.addLayer(population_Count_2015.updateMask(population_Count_2015.
gt(10)),vispar_population, 'Population per 250m cell 2015');
```



Đừng quên lưu tệp lệnh của bạn. Cú pháp đầy đủ có thể được tìm thấy ở đây: <https://code.earthengine.google.com/10f75fc9ffc24e3ffd18cc8f2a664373>

Không ai giống ai. Một số người ít có khả năng di chuyển hoặc độc lập hơn và sẽ gặp khó khăn hơn trong việc sơ tán trong trường hợp lũ lụt. Mặc dù điều này là cá nhân, nhưng một số nhóm có thể dễ bị tổn thương hơn những nhóm khác. Ví dụ như trẻ em, bệnh nhân trong bệnh viện và người cao niên. Khi biết vị trí của họ có thể giúp xác định nhu cầu hỗ trợ thêm trong trường hợp lũ lụt. Trong bài tập này, chúng ta sẽ thêm các cơ sở y tế vào bản đồ. Chúng ta sẽ sử dụng cùng một tập dữ liệu từ bài tập 11 của gói đào tạo 1.

4.3.1 Cơ sở y tế

Người dân trong các cơ sở y tế (bệnh viện, v.v.) sẽ ít di chuyển hơn trong trường hợp lũ lụt. Do đó, chúng ta muốn biết nơi đặt các cơ sở y tế để xác định nơi có nhu cầu cụ thể để được hỗ trợ sơ tán hoặc cần bảo vệ thêm.

Để kết hợp tất cả thông tin, hãy bắt đầu từ kịch bản của bài tập 4.

Bước 1. Tải *Health_facilities_Vietnam.zip* xuống máy tính của bạn từ trang Moodle trong Gói đào tạo 4.

Chúng ta tải shapefile này lên Google Earth Engine.

Bước 2. Ở trên cùng bên trái của màn hình, hãy tìm 'Assets' và nhấp vào nó. Tiếp theo nhấp vào 'NEW' (nút màu đỏ) và sau đó nhấp vào 'Shape files'.



Image Upload

GeoTIFF (.tif, .tiff) or TFRecord (tfrecord)

Table Upload

Shapefiles (.shp, .shx, .dbf, .prj, or .zip)

CSV file (.csv)

Image collection

Folder

Bước 3. Chọn tất cả các tệp trong zip (sau khi giải nén zip trên máy tính của bạn). Đổi tên assetname thành 'Vietnam_healthfacilities' và nhấp vào 'Upload'.

Upload a new shapefile asset

Source files

Please drag and drop or select files for this asset.

Allowed extensions: shp, zip, dbf, prj, shx, cpg, fix, qix, sbn or shp.xml.

hotosm_vnm_health_facilities_points.cpg

hotosm_vnm_health_facilities_points.dbf

hotosm_vnm_health_facilities_points.prj

hotosm_vnm_health_facilities_points.shp

hotosm_vnm_health_facilities_points.shx

Bây giờ, bạn có thể thấy nút 'Tasks' chuyển sang màu cam. Tệp shapefile đang được tải lên. Sau một lúc, hãy làm mới dữ liệu của bạn bằng cách nhấp vào nút này ở trên cùng bên trái của màn hình.

Bước 4. Bạn sẽ thấy 'Vietnam_health_facilities' bây giờ xuất hiện trong Assets. Nhập dữ liệu này vào tập lệnh của bạn bằng cách nhấp vào nó và sau đó nhấp vào 'Import'.

Bây giờ bạn có thể thấy rằng tập dữ liệu này là một phần của Imports.

Bước 5. Đổi tên dữ liệu thành 'healthfac'.

► var **healthfac** : Table [users/carolien/vietnam_health_facilities](#)

Luôn luôn tốt để kiểm tra những gì bạn đã nhập bằng cách in tập dữ liệu vào Bảng điều khiển của bạn.

Bước 6. In 'healthfac' vào Bảng điều khiển.

```
// health facilities
```

```
// take a look at the shapefile that you have imported print(healthfac);
```

Cuối cùng, chúng ta sẽ thêm các cơ sở y tế vào bản đồ.

```
// add the health facilities to the map Map.addLayer(healthfac, {color: 'black'}, 'health facilities');
```



Đừng quên lưu tệp lệnh của bạn. Cú pháp đầy đủ có thể được tìm thấy ở đây: <https://code.earthengine.google.com/cfdc4c830cd0a88f41fad620c293fc55>

V. KẾT LUẬN

Từ kết quả xác định các ứng dụng của GEE rút ra một số kết luận sau:

- GEE có thể ứng dụng trong rất nhiều lĩnh vực khác nhau;
- Khai thác dữ liệu khổng lồ các loại dữ liệu về độ ẩm, dữ liệu ảnh sentinel, các dữ liệu về lượng mưa toàn cầu, dữ liệu về việc sử dụng đất,...
- So sánh, đánh giá các giai đoạn và thời kỳ khác nhau,
- Xây dựng các dạng bản đồ theo định hướng của người sử dụng như bản đồ nhạy cảm lũ lụt, bản đồ biến đổi đất,....

TÀI LIỆU THAM KHẢO

1. Báo cáo học thuật được lược dịch từ khóa đào tạo về GEE của trường Đại học Tài Nguyên và Môi trường.