Tutorial: Generate Cloud Free Landsat 8 Composite with GEE

Introduction

Google Earth Engine (GEE) is a cloud based geospatial analysis platform that allows users to analyze and visualize earth surface data at the planetary scale. Since the GEE platform and data are built within Google's infrastructure, the benefit of using this environment is its ability to rapidly process large volumes of data. Although the GEE is powerful to analyze the large data, the deposition of cloud in remotely sensed data is a major problem it prevents it from capturing the land surface data. This can be specially challenging in Hawai'i as it has cloud cover throughout the year especially along the mountain ridge areas where humans cannot easily access due to dynamic topography such as cliffs. Having a cloud-free mosaic imagery enhances the opportunities to effectively understand the land surface .

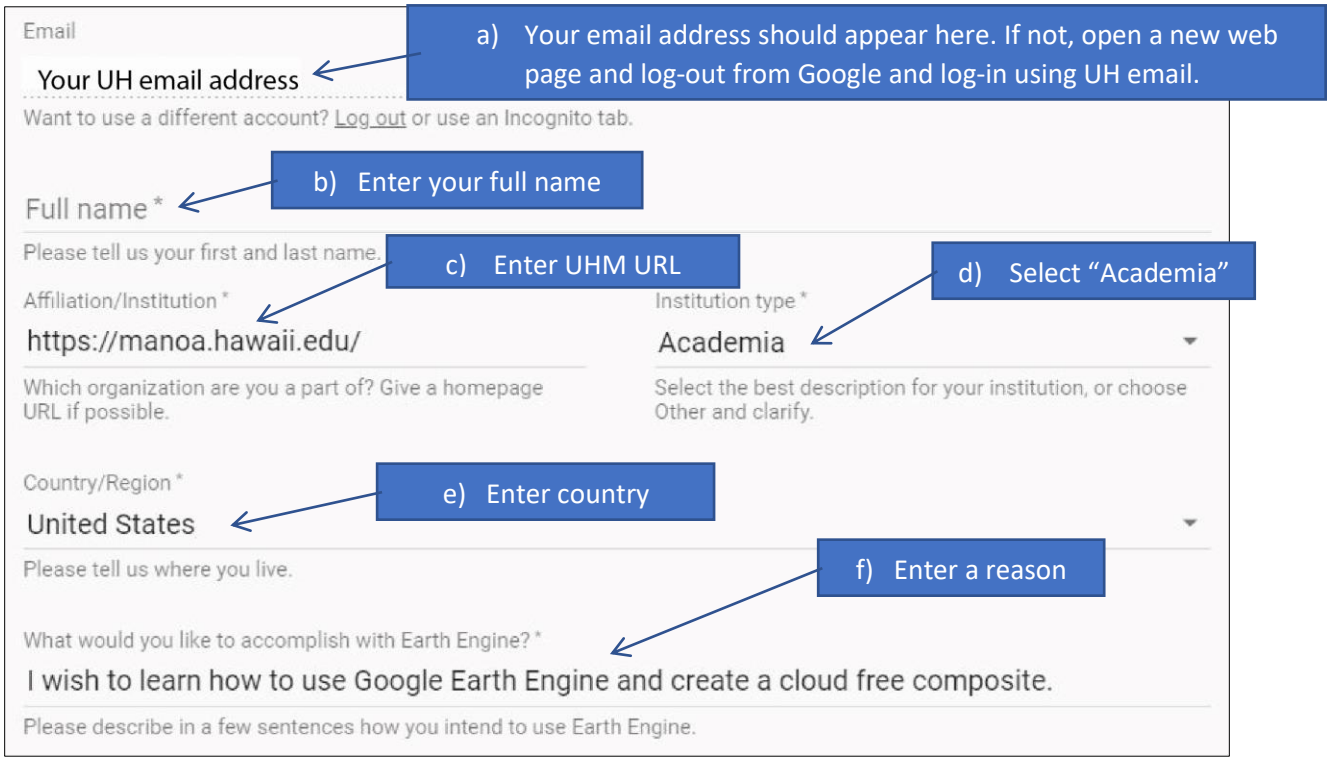
Objective

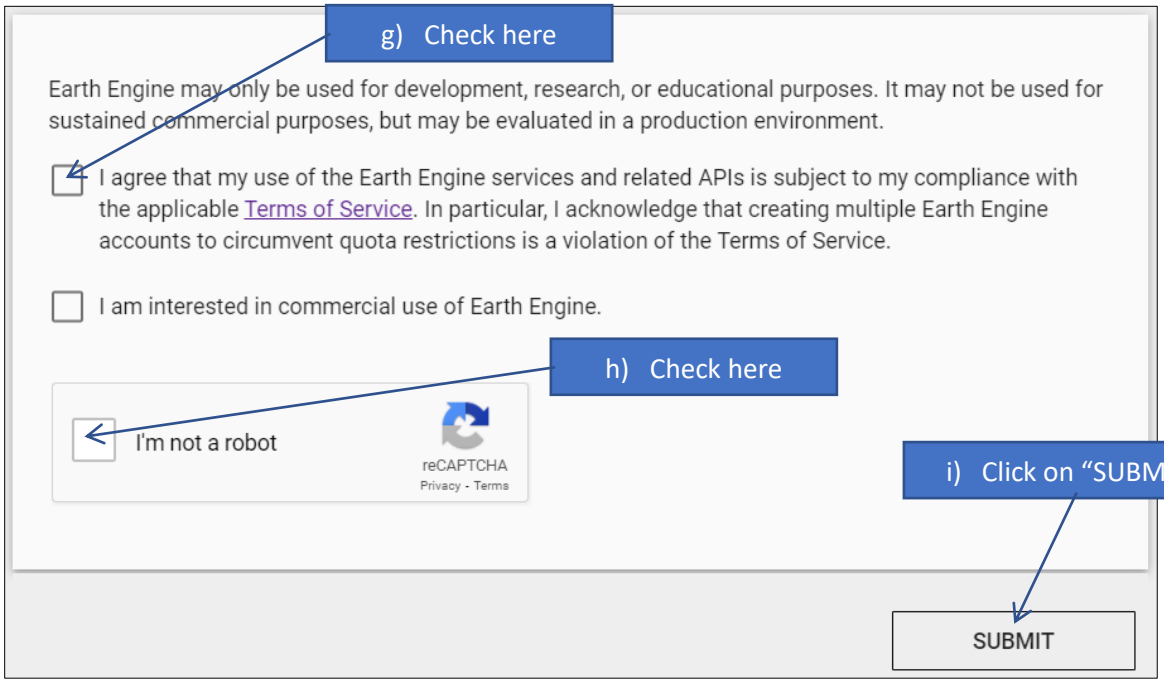
This tutorial will demonstrate how to create a cloud-free Landsat 8 composite for the island of O'ahu in GEE using a GEE code editor and Landsat 8 satellite imagery. Upon completion of the image processing the final cloud-free imagery will be stored in your Google Drive. The entire processes are conducted within GEE's platform using JavaScript and requires a stable Internet connection.

The following web browsers are supported and were confirmed to work, Chrome, Chrome with Incognito mode, Firefox, Microsoft Edge, and Safari browser on both Windows 10 PC, iMac Catalina and iPad Pro iOs 14.7.1. (as of September 1, 2021).

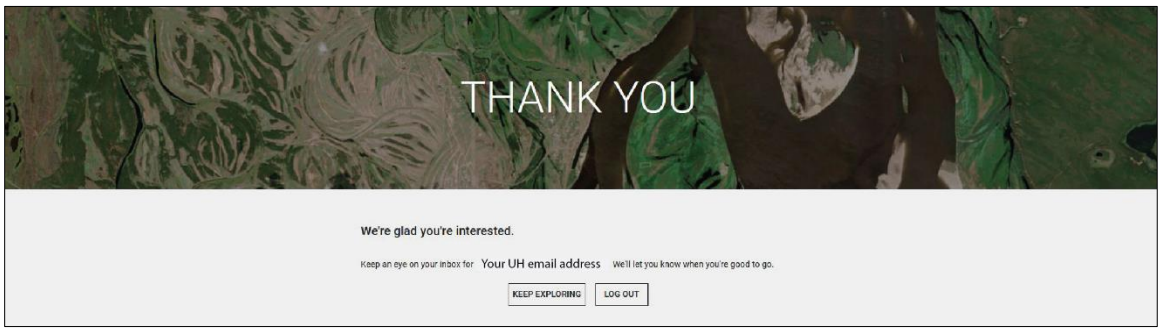
1. Go to https://earthengine.google.com/ or search **Google Earth Engine** in your preferred browser and login using your UH email address or your GEE account if you already have a GEE account.
    a. If you already have a GEE account → skip to the step 3 on page 3.
    b. If you need to create a GEE account → go to the next step

2. To create a GEE account, click "**Sign Up**" and login using your UH email account. Fill out the form and go through GEE's evaluation process.
   See below to complete **steps (a) to (i)**.

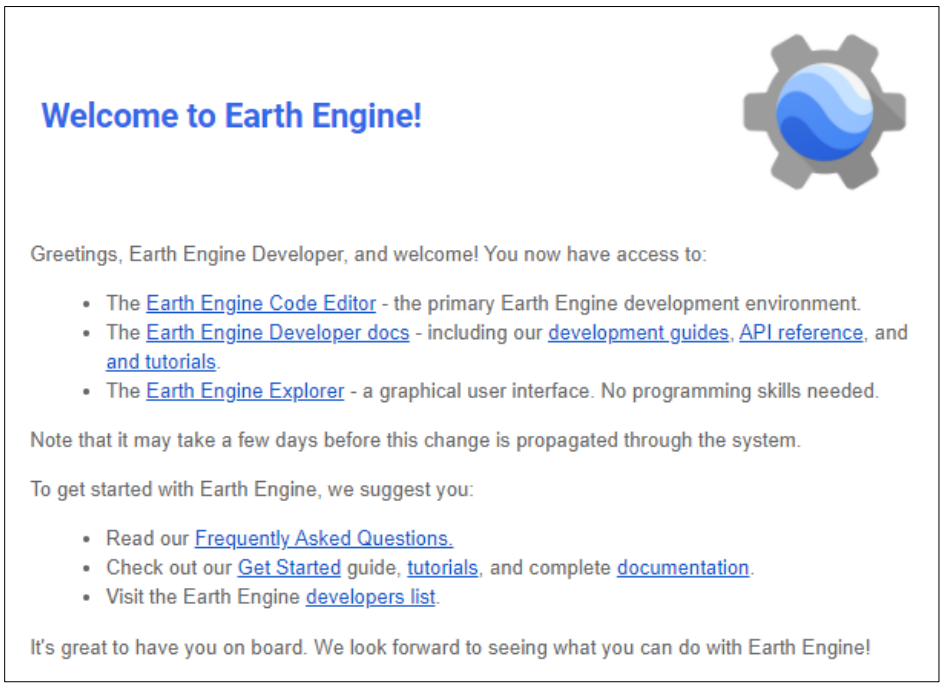




After the submission of the form, you should see this. Then, check you UH email.

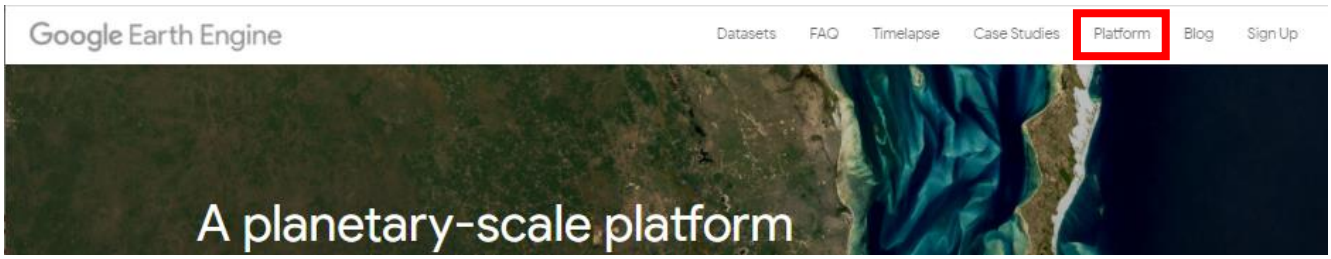


You should receive an email like the image below.

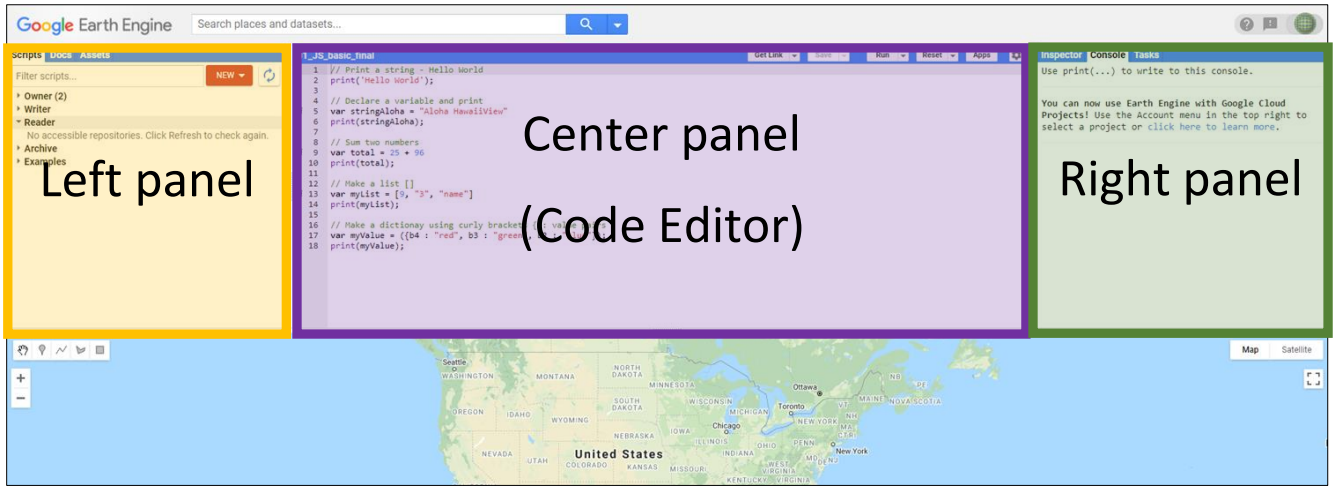


Congratulations! Now, you are ready to use GEE code editor!

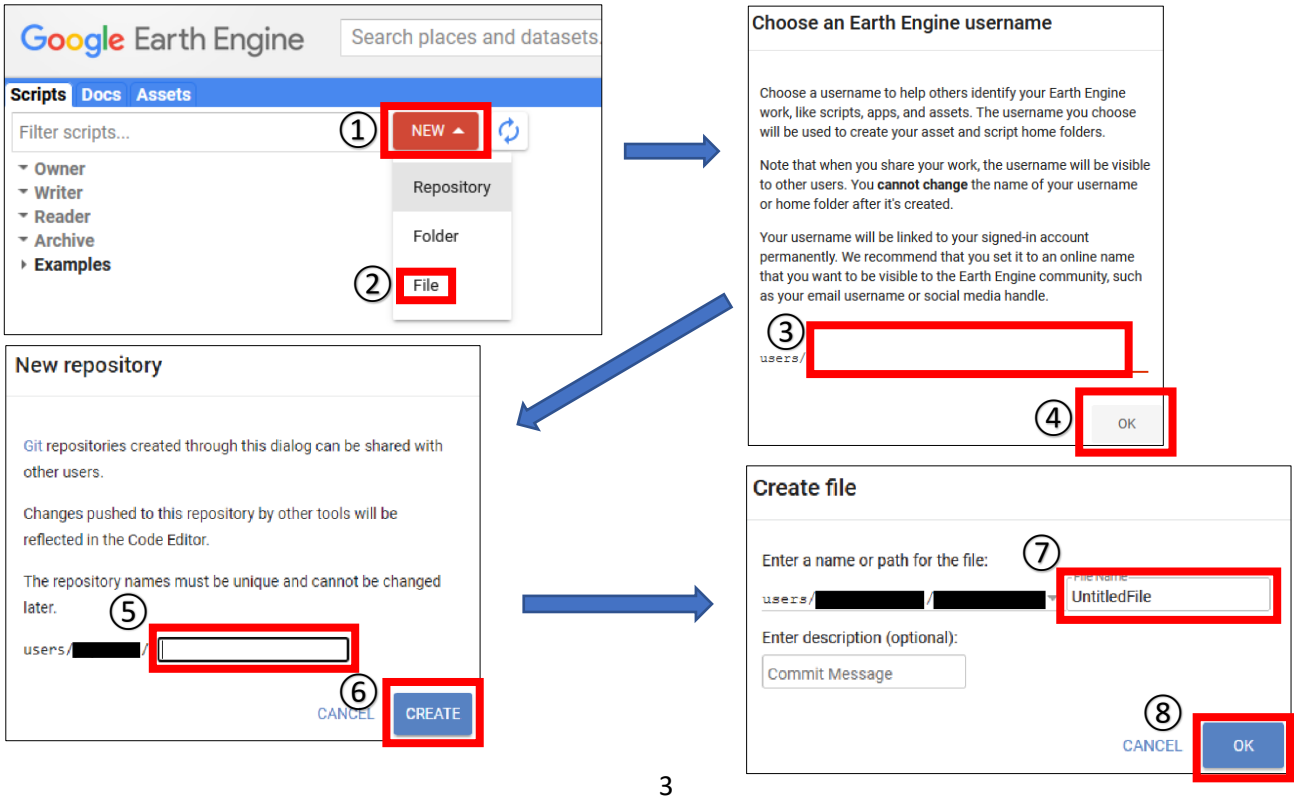
3. Open a GEE code editor and go to https://earthengine.google.com/ or search **Google Earth Engine** in your preferred browser and login using your UH email address or your GEE account if you already have a GEE account.
Move your cursor on "**Platform**" and select "**Code Editor**" from the drop down



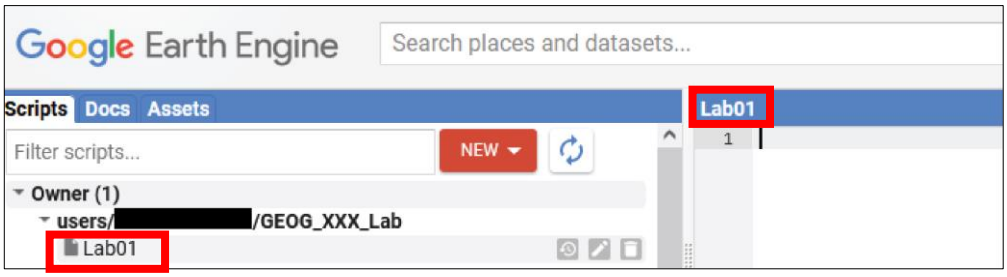
You should see the GEE code editor like the image below. There are three panels beside a map area.



4. In the left panel, there are three tabs. Select **Script** tab and click on **NEW.** In the drop down, click **File**. In the new window, enter the username by your choice and click **OK**. At the pop up window enter "**GEOGXXX_Lab**" then click the **CREATE** button. In the new window, enter the file name as "**LabXX**" and click **OK**.



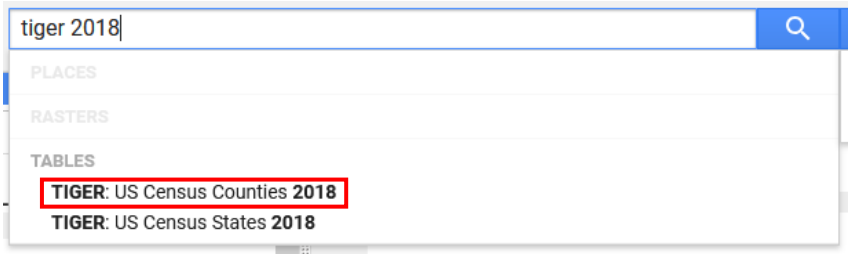
5. In the Script tab, click your file that you created from the previous step and make sure the file name appears in the center panel.



6. Enter code **Map.setCenter(-157.8, 21.48, 10)** to set the map center over O'ahu using its longitude (-157.8), latitude (21.48) and zoom level (10). **Save** and **Run** the code.

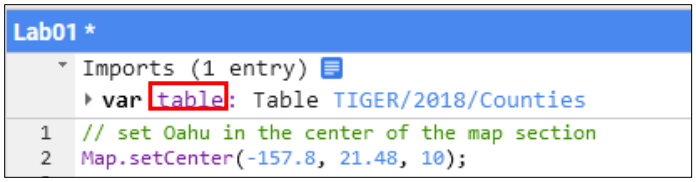
Tip: You can add comments using double slashes **//** at the beginning. Use semicolon at the end of code.

In the search field right above the center panel, enter **tiger 2018** and select **TIGER: US Census Counties 2018**. A new window will pop up then click on the **IMPORT**.

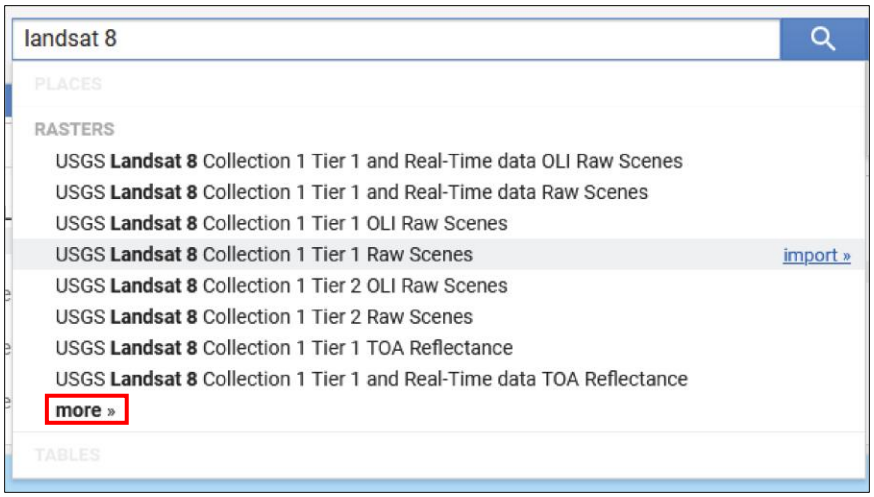


Note: The new window shows the data information including description, table schema, terms of use, and citation. It is a good practice to understand the dataset.

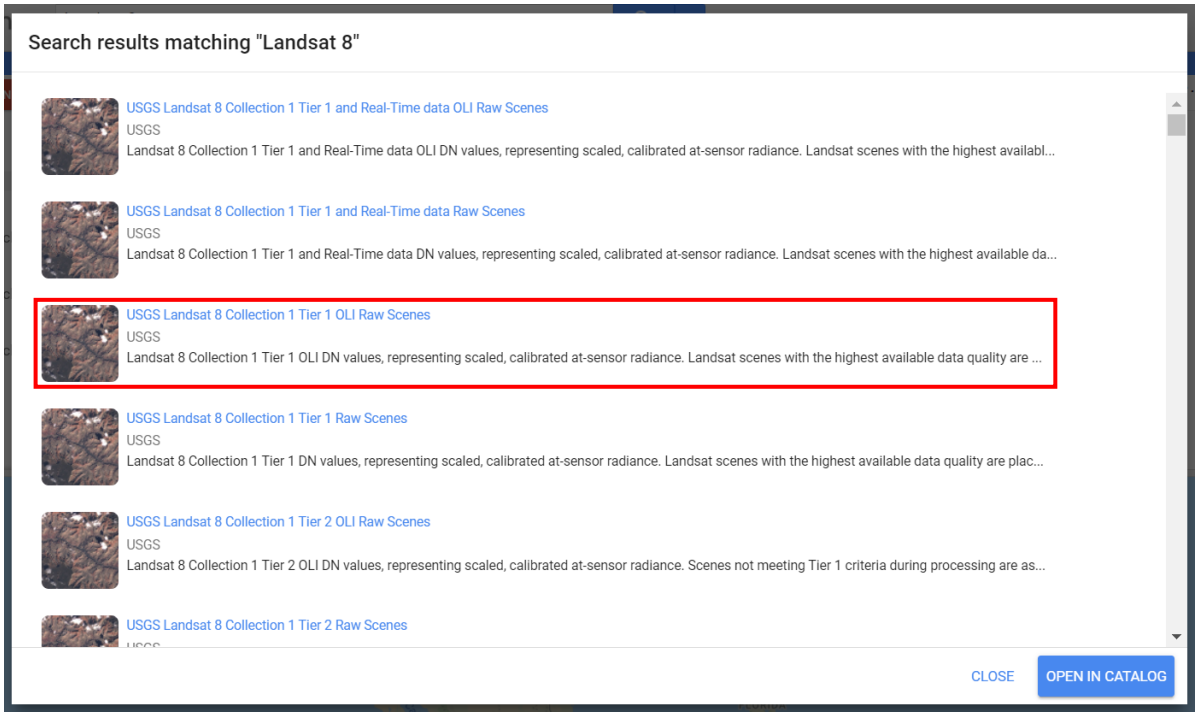
7. Click **table** and rename it as **counties**



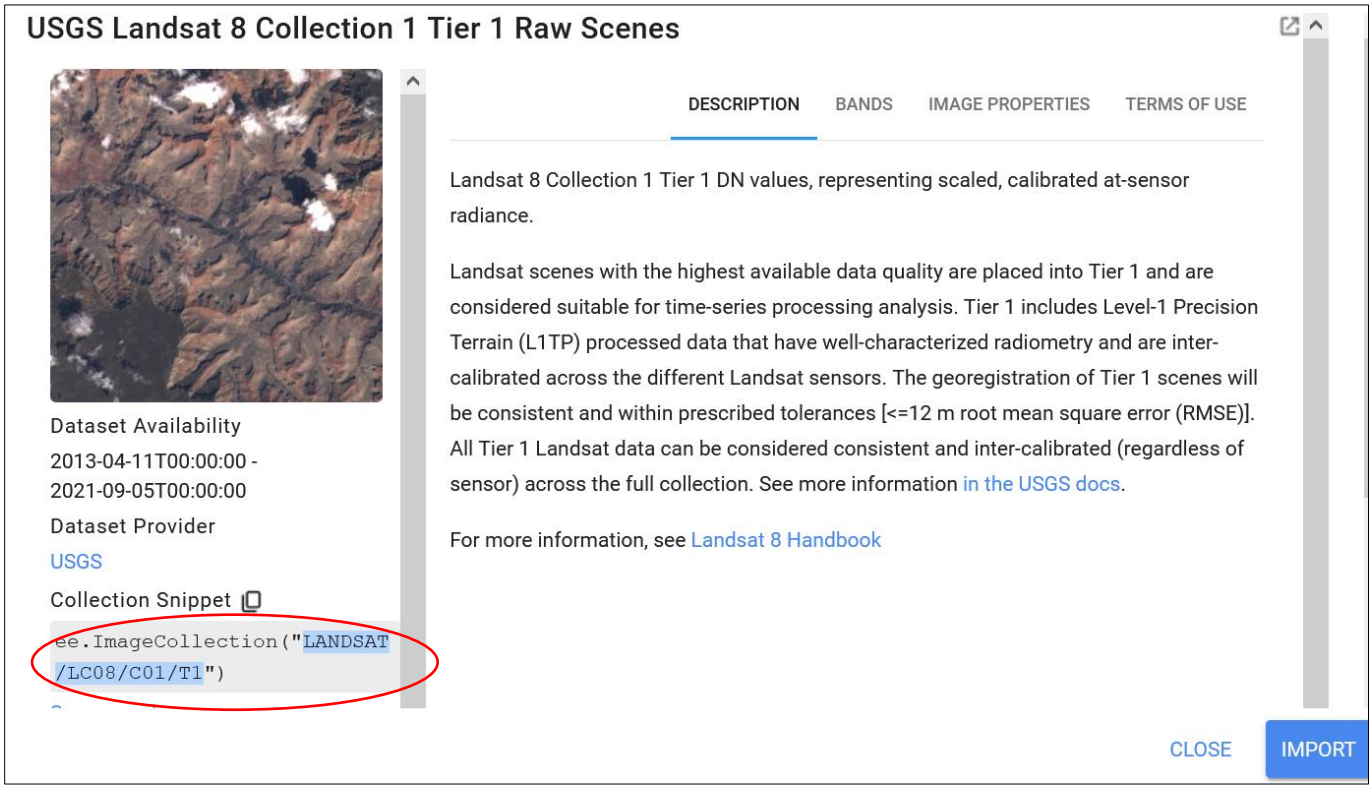
8. In the same search field enter **landsat 8** and click on **more**



A new window pops up showing all available datasets with Landsat 8 tag. Click **USGS Landsat 8 Collection 1 Tier 1 Raw Scenes**.



Another window will pop up and display detailed information of the imagery. The dataset ID is located on the left bottom as **LANDSAT/LC08/C01/T1**. Copy it.



9. Enter the code to specify the dataset ID (See below). Save and Run the code.

```
3
4  // landsat 8 dataset
5  var landsat8 = 'LANDSAT/LC08/C01/T1';
6
```

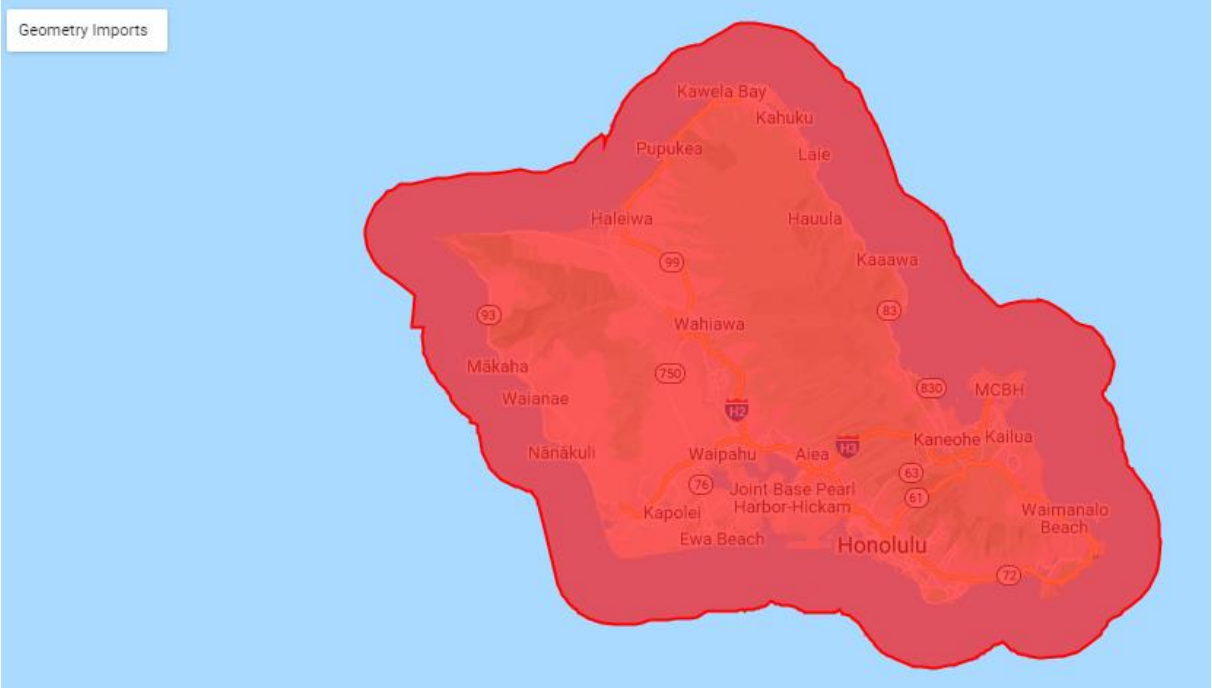
***Note**: A raw Landsat 8 dataset is required to create a cloud free composite.*

10. Specify the timeframe. Follow the date format specified below. **Save** and **Run** the code.

```
6
7  // start and end date for creating cloud-free mosaic
8  var dateStart = '2017-01-01';
9  var dateEnd   = '2020-12-31';
10
```

11. Specify location boundary using the county file that is imported at the step 3. Next, add the location boundary to the map section. **Save** and **Run** the code.

```
10
11 // specify areas
12 var oahu = counties.filter(ee.Filter.eq('NAME','Honolulu'));
13 Map.addLayer(oahu, {color: 'red'});    // display Oahu boundary from counties dataset
14
```



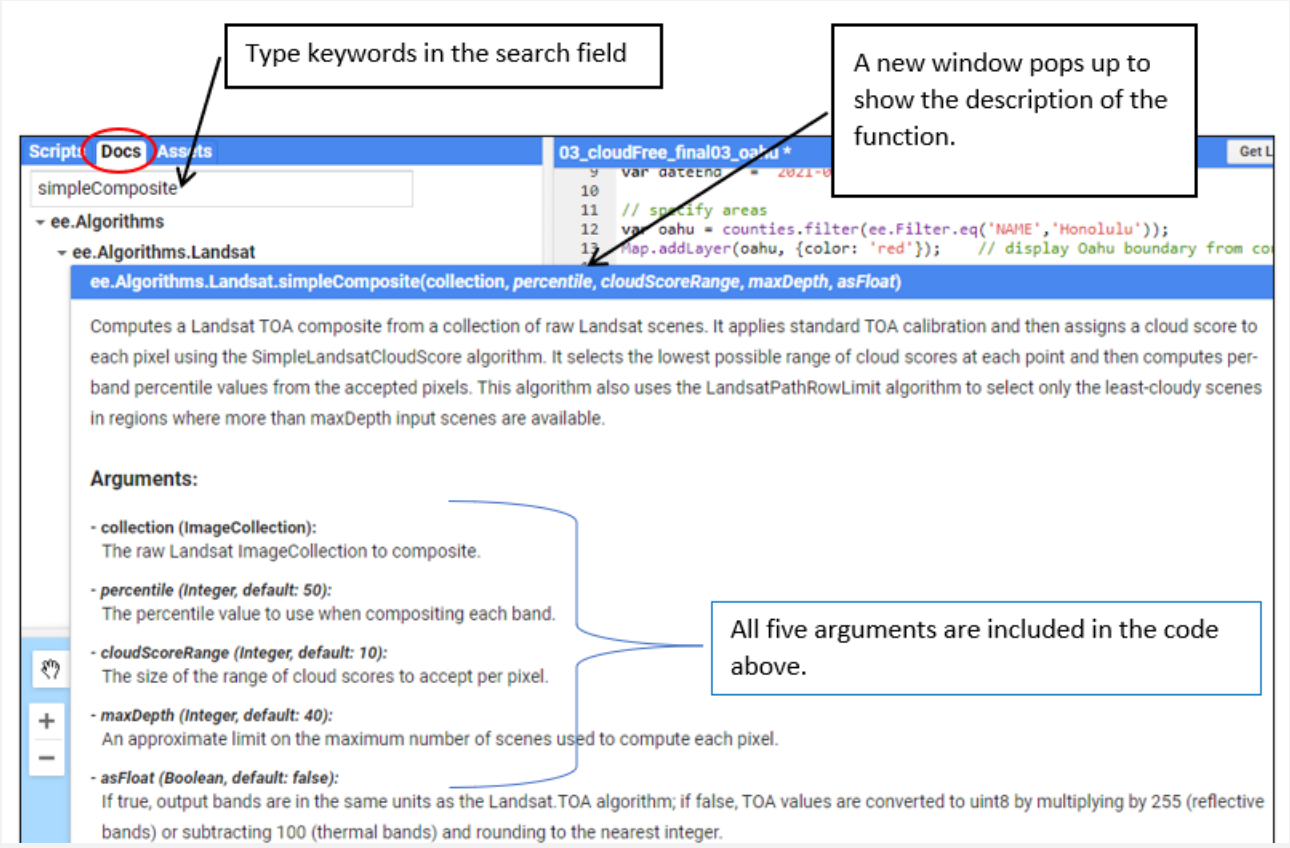
12. Narrow down the size of Landsat 8 Raw dataset using the timeframe and the location boundary data. **Save** and **Run** the code.

```
14
15  // Create a Landsat 8 collection for oahu
16  var collectionOahu = ee.ImageCollection(landsat8)
17      .filterDate(dateStart, dateEnd)
18      .filterBounds(oahu);
19
```

13. At this point all preparation is done and now it is ready to use the cloud-free composite function to process the Landsat 8 raw imagery. Add the code shown in the image below. **Save** and **Run** the code.

```
19
20  // Create a cloud-free composite with custom parameters
21  var oahuComposite = ee.Algorithms.Landsat.simpleComposite({
22      'collection': collectionOahu,
23      'percentile': 50,
24      'cloudScoreRange': 10,
25      'maxDepth':1000,
26      'asFloat': true
27  });
28
```

***Note**: To understand **ee.Algorithms.Landsat.simpleComposite()**, click the Docs tab in the left panel. Enter **simplecomposite** in the search field.*

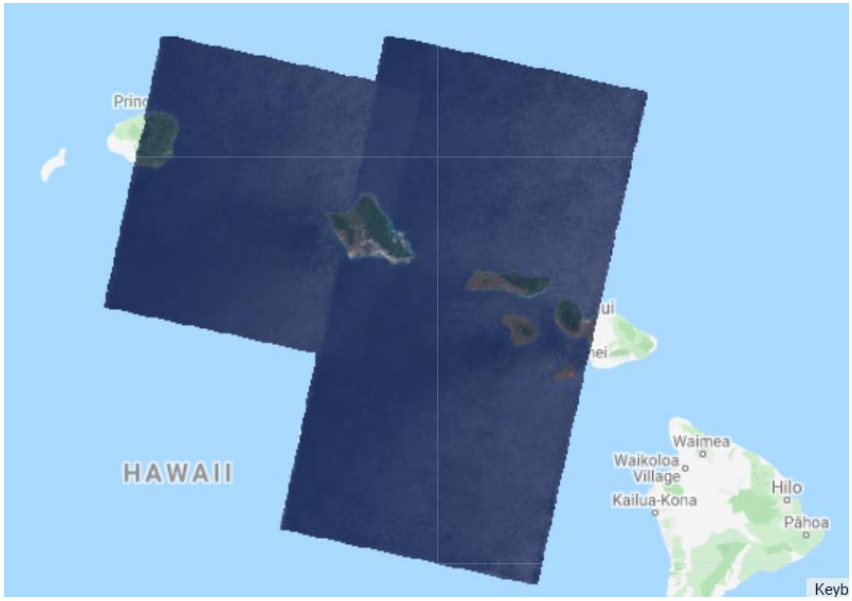


Let's take a look at the cloud-free image.

14. Add the code below to set the band combination assignment for RGB (Red, Green, and Blue) and add the image layer to the map section. **Save** and **Run** the code.

```
28
29  // create a band combination assignment for RGB, then add the cloud free image to the map section
30  var bandComb = ['B4', 'B3', 'B2']
31  Map.addLayer(oahuComposite, {bands: bandComb, max: 0.3},'Oahu TOA composite');
32
```

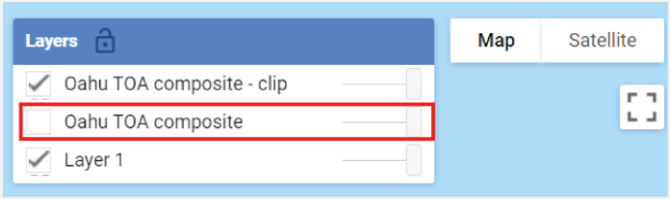
Notice, the area of the cloud-free composite mosaic is much greater than O'ahu.



15. Enter the code below to clip the cloud-free composite mosaic using the **O'ahu** variable that is created in the **step 11.** **Save** and **Run** the code.

```
32
33  // Clip the cloud-free mosaic by a polygon and add it to the map section
34  var cloudFreeMosaicOahu = oahuComposite.clip(oahu);
35  Map.addLayer(cloudFreeMosaicOahu, {bands: bandComb, max: 0.3},'Oahu TOA composite - clip');
36
```

Note: Now, there are three layers on the map. Mover the cursor over the **Layer** box in the map section and uncheck **Oahu TOA composite**. Then, you can see the clipped cloud-free composite mosaic just for the island of O'ahu without any clouds!



16. Add the code below to export the cloud free composite to the Google Drive. **Save** and **Run** the code.

```
36
37  // export the cloud-free mosaic to google drive (folder name with a date range)
38  var folder = '_GEE_cloudFree_Lab01'
39  Export.image.toDrive({
40    image: cloudFreeMosaicOahu,
41    description: 'oahu_mosaic_' +  dateStart + '_' + dateEnd + '_resolution30m',
42    folder: folder,
43    scale: 30, // spatial resolution in meter
44    maxPixels: 1e13,
45  });
46
```

***Note**: The folder name in the Google Drive is assigned as **_GEE_cloudFree_Lab01**. The cloud-free composite file is named as **oahu_mosaic2017-01-01_ 2020-12-31_resolution30m**. Notice, the variables are replaced with the values and "+" signs disappear. "+" sign is to connect each element. So, more elements can be added to the file name. It is important to enclose the string within a pair of apostrophes. Refer **Export.image.toDrive()** in the **Docs**.*

Export.image.toDrive(image, description, folder, fileNamePrefix, dimensions, region, scale, crs, crsTransform, maxPixels, shardSize, fileDimensions, skipEmptyTiles, fileFormat, formatOptions)

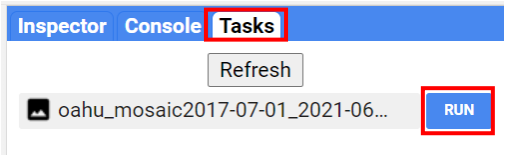
Creates a batch task to export an Image as a raster to Drive. Tasks can be started from the Tasks tab. "crsTransform", "scale", and "dimensions" are mutually exclusive.

Arguments:

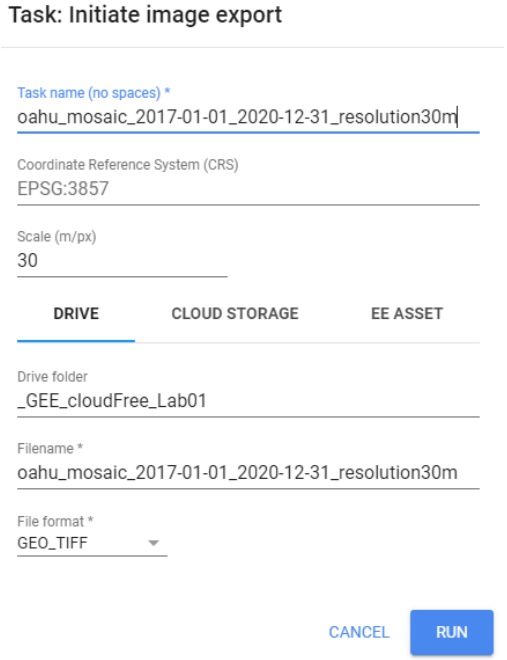
- **image (Image):**
  The image to export.
- ***description (String, optional):***
  A human-readable name of the task. May contain letters, numbers, -, _ (no spaces). Defaults to "myExportImageTask".
- ***folder (String, optional):***
  The Google Drive Folder that the export will reside in. Note: (a) if the folder name exists at any level, the output is written to it, (b) if duplicate folder names exist, output is written to the most recently modified folder, (c) if the folder name does not exist, a new folder will be created at the root, and (d) folder names with separators (e.g. 'path/to/file') are interpreted as literal strings, not system paths. Defaults to Drive root.
- ***fileNamePrefix (String, optional):***
  The filename prefix. May contain letters, numbers, -, _ (no spaces). Defaults to the description.
- ***dimensions (Number|String, optional):***
  The dimensions to use for the exported image. Takes either a single positive integer as the maximum dimension or "WIDTHxHEIGHT" where WIDTH and HEIGHT are each positive integers.
- ***region (Geometry.LinearRing|Geometry.Polygon|String, optional):***
  A LinearRing, Polygon, or coordinates representing region to export. These may be specified as the Geometry objects or coordinates serialized as a string. If not specified, the region defaults to the viewport at the time of invocation.
- ***scale (Number, optional):***
  Resolution in meters per pixel. Defaults to 1000.
- ***crs (String, optional):***
  CRS to use for the exported image.

Important: After running the code, the entire island of O'ahu should be in the map view as GEE will export the image as shown in the map view.

17. In the right panel, select **Tasks** tab and click **RUN**.



A new window pops up. You can change the folder or file name in this section before exporting. It may take more than 5 minutes. If it continues running for more than 10 minutes, click **Refresh**.



18. The exported cloud-free composite mosaic looks like below if you open it in GIS or remote sensing software. This mosaic is TOA (Top of Atmosphere) reflectance.