Google Earth Engine API

September 2, 2026

maXbox Starter Tutorial 194

Google Earth Engine combines a multi-petabyte catalog of satellite imagery and geospatial datasets with planetary-scale analysis capabilities. Scientists, researchers, and developers use Earth Engine to detect changes, map trends, and quantify differences on the Earth's surface. Earth Engine is now available for commercial use, and remains free for **academic and research** use.

We take a short look at the following API:

GitHub – google/earthengine-api: Python and JavaScript bindings for calling the Earth Engine API. · GitHub

Python and JavaScript client libraries for calling the Google Earth Engine API.

- Earth Engine Homepage
- Web Code Editor
- Python Installation

Here's an example screenshot and the corresponding Code Editor JavaScript code:

22:34

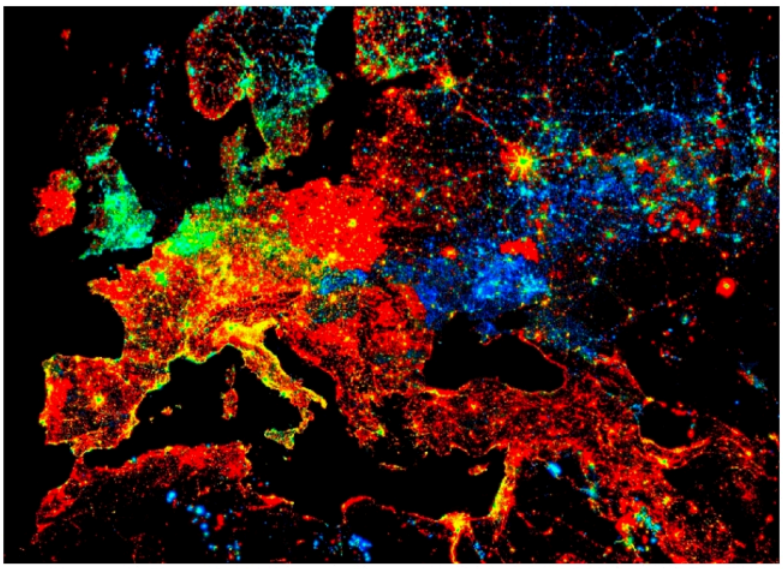
README Code of conduct More

Google Earth Engine API

Python and JavaScript client libraries for calling the Google Earth Engine API.

- Earth Engine Homepage
- Web Code Editor
- Python Installation

Here's an example screenshot and the corresponding Code Editor JavaScript code:



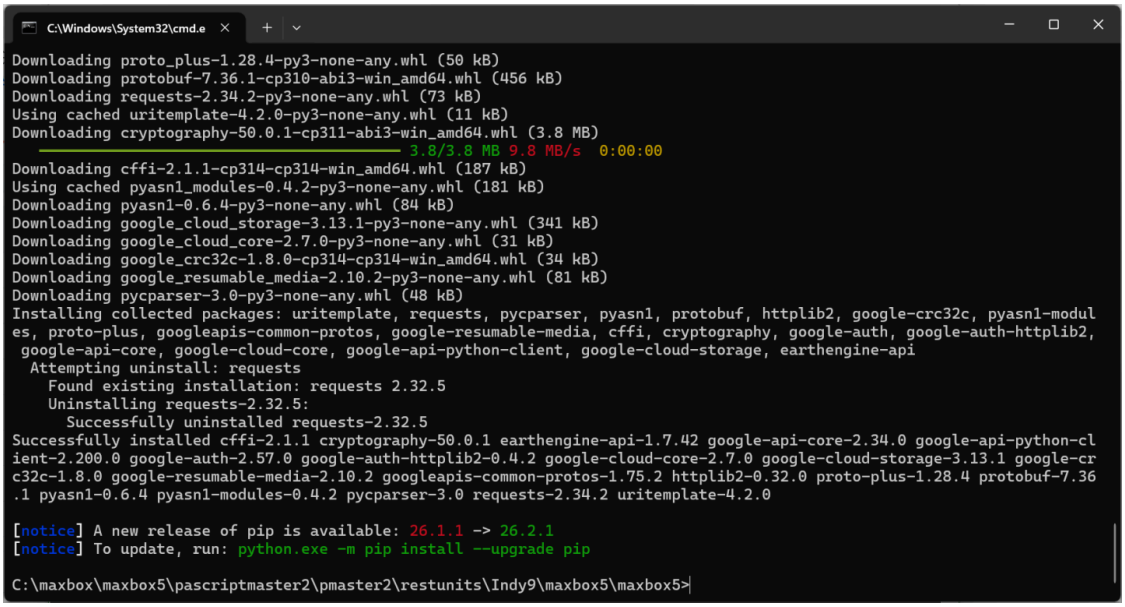
```
// Compute the trend of night-time

// Adds a band containing image dat
function createTimeBand(img) {
  var year = ee.Date(img.get('syste
  return ee.Image(year).byte().addB
}
```

Python Code Example

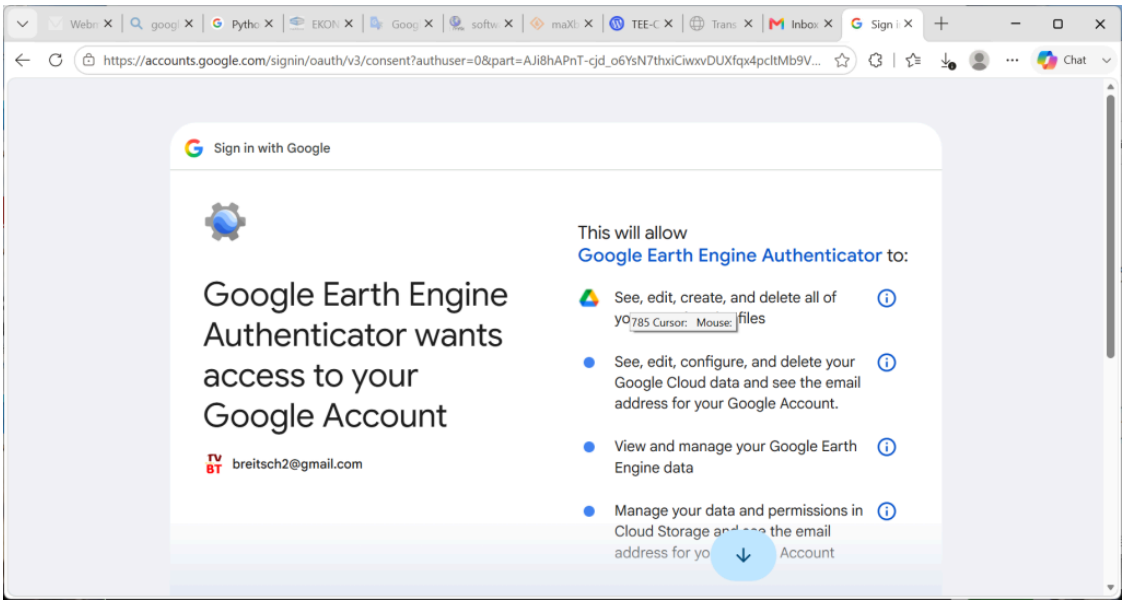
First we have to install the Python Lib:

```
py -3.14 -m pip install earthengine-api
```



PIP Protocol

Then we open a project (used or new) with an existing google account:



Authenticator to use the API

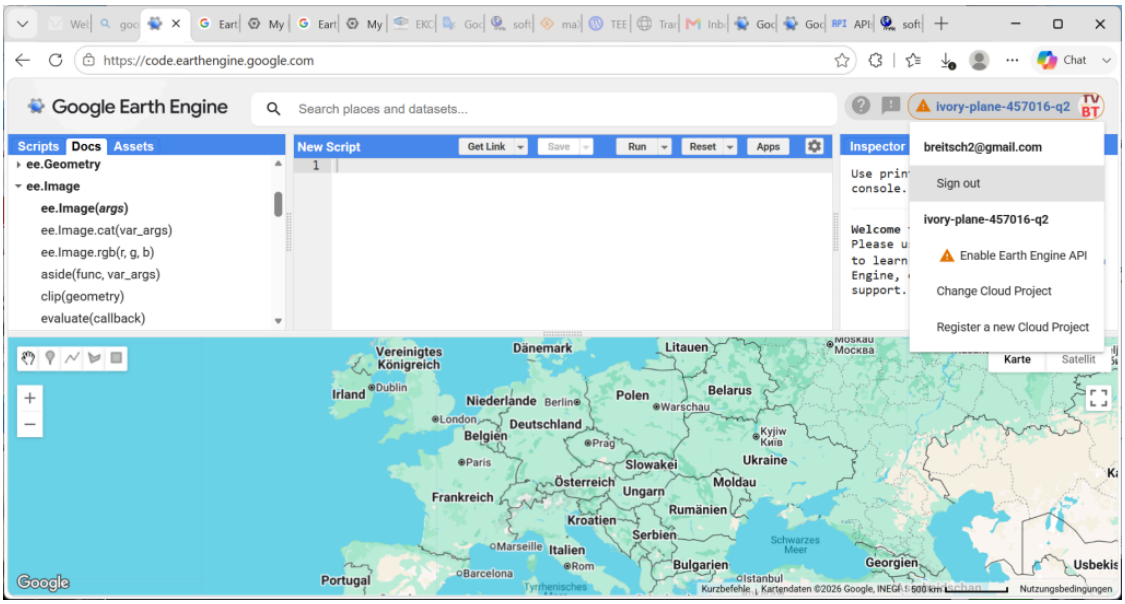
Earth Engine runs in Google Cloud and requires Cloud Projects for access and API management. All Earth Engine calls (whether from the code editor, client

libraries, apps, or the REST API) are routed through a Cloud Project. This allows you to use the Cloud Console for access control, resource management, and usage monitoring. After that we got an access:

```
Google Earth Engine authorization successful!

Credentials have been retrieved. Please close this window.
```

Projects created during registration can be managed in the Google Cloud Console. You can enable or disable the Earth Engine API on the “Earth Engine API” page in the Cloud Console.



Each Project has Enabler / Disabler

Clientlibraries (Python, JavaScript)

The `ee.Initialize()` function specifies a project for Earth Engine requests originating from the Python and JavaScript client libraries. There are several ways to configure the project specification. See the Authentication and Initialization page for more information.

This code can be automatised with Python4Delphi in maXbox:

```
1 function PyCode2GoogleEarthEngineAPI(my_project, aAPIKey: str
```

```
begin
  with TPythonEngine.Create(Nil) do begin
  try
    loadDLL;
    autofinalize:= false;
    ExecString('import io,sys');
    execString('import ee');
    execStr('ee.Authenticate()');
    execstr('output = io.StringIO()')     //for print out
    ExecStr('ee.Initialize(project="'+my_project+'")'+lf+
            //'files = {"image": image_file_descriptor}'+lf+
            //'headers= {"X-Api-Key":"'+aAPIKey+'"}'+lf+
            'print(ee.String("Hello from Earth Engine servers
    println(stringreplace(evalstr('output.getvalue().strip()
                                #10,#13#10,[rfReplaceAll, rfIgnor
    //result:= ((EvalStr('r.json()')));
  except
    raiseError;
  finally
    free;
  end;
 end;
end;
```

ngine_API1_7.txt/download

If you want to associate Earth Engine calls from the command-line tool with a cloud project, use the –project parameter in your call to earthengine or set a default project with set_project.

The path to the credentials is stored in the `GOOGLE_APPLICATION_CREDENTIALS` environment variable. To authenticate and initialize Earth Engine using the credentials, see the snippet below.

```
import json
import os

from google.auth import identity_pool

scopes = [
    "https://www.googleapis.com/auth/cloud-platform",
    "https://www.googleapis.com/auth/earthengine",
]
path = os.environ["GOOGLE_APPLICATION_CREDENTIALS"]
info = json.load(open(path))
credentials = identity_pool.Credentials.from_info(info).with_
ee.Initialize(credentials, project=credentials.project_numbe
```

When no used project can be found or the enabler is blocked you got an exception:

```
Exception: EEException: Google Earth Engine API has not been us
ed in project ivory-plane-457016-q2 before or it is disabled.
Enable it by visiting https://console.developers.google.com/api
s/api/earthengine.googleapis.com/overview?project=ivory-plane-
457016-q2 then retry.
```

If you enabled this API recently, wait a few minutes for the action to propagate to our systems and retry.

Posted in Code, Geo Intelligence, Geomap

Leave a comment