**[Requests Module in Python: Find Weather Using API](https://copyassignment.com/requests-module-in-python-find-weather/)**

## **Requests Module in Python**

Request Module in Python works for websites. It provides a facility to send an HTTP request with a set of functionalities like POST, GET, PUT, DELETE, etc. It deals with cookies, timeouts, multi-part file uploads, streaming downloads.

Let’s get started:

First you need to install ‘requests’ in your pip. Here’s the command:

pip install requests

This module works for websites or HTTP. So, for any operation to be done, you need to pass ‘URL’ first. Let’s pass ‘URL’ for the official documentation of ‘requests’: <https://pypi.org/project/requests/>

import requests

r = requests.get("https://pypi.org/project/requests/")

print(r)

Now, if you print r here, you’ll get a response code which is basically a response from the website whether it is ok or not. 200’s are successful responses and 400’s represent errors while establishing connections.  
Output:

200 #shows successful response

As your know dir() function is used to check available functions in the object. If you check dir(r) you will notice all the available functions like ‘header’: prints header details of URL, ‘content’: this works for bytes, ‘JSON’: convert JSON format to the dictionary, etc.

You may wonder but you can download an image from the website using this code! Just copy the image address and save write the image file with the content method as the image is in bytes.

r = requests.get("https://encrypted-tbn0.gstatic.com/images?q=tbn:ANd9GcR-LDYLhu96jjkP8dAervpsP8dRjFqk1\_MmFg&usqp=CAU")

with open("image.png","wb") as f:

f.write(r.content)

We have open the file in binary mode as the image is in bytes and hence, using the ‘content’ functionality provided by requests, this byte image will be encoded and will save the image in a new file ‘image.png’ as specified.

As we discussed earlier basic authentication, cookies, delay, dynamic data methods available inside this module, you can find and test these methods in this website created by ‘requests’ module creators themselves: <https://httpbin.org/>

Now, as you’ve got a little bit of idea about ‘requests’, let’s get on to our main project i.e. to find weather using API.

## **What is API?**

APPLICATION PROGRAMMING INTERFACE, API, is a server used to communicate with each other by means of code. These API’s are of various types by which we can really build cool and amazing projects.

### **How to get these API’s?**

Thankfully, there are thousands of websites that provide us required API. Some websites cost for it while some provide it free for a limited period of time. All have their different prices.

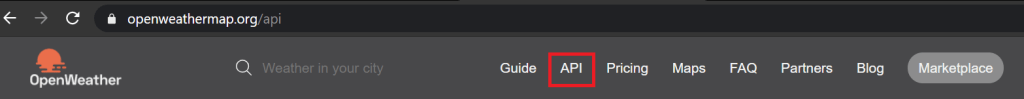
### **How to access the data from this API?**

We have to send API requests by the ‘requests’ module. The data thus we get is of course in ‘JSON’ format that can be converted into a dictionary by the ‘JSON’ method in requests.

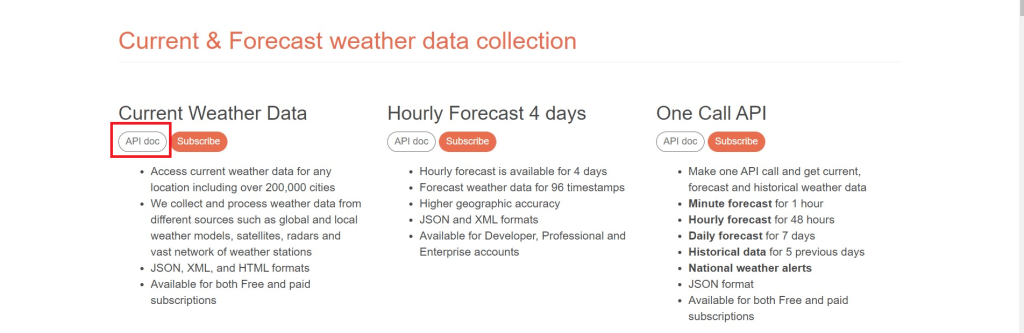
Seems pretty cool, let’s find weather:

Now here, I m using weather key from the website <https://openweathermap.org/>

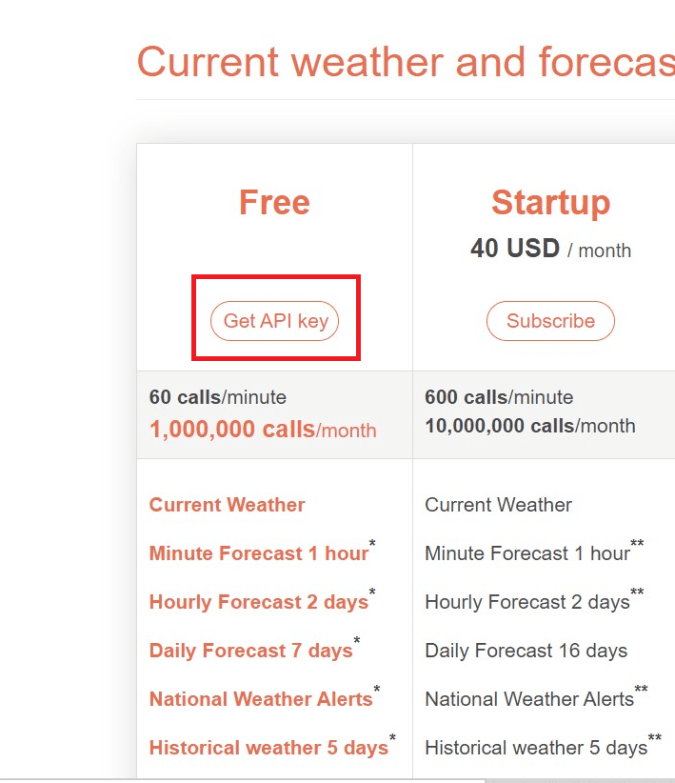
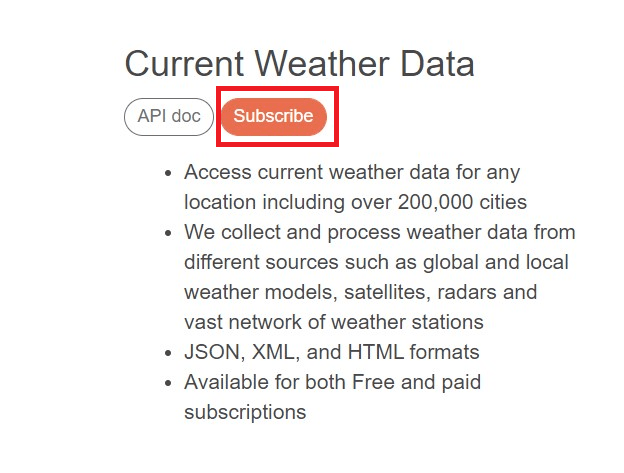
You need to first register if you are not registered and then sign in. Now in the API section, select desired API and you can also read its documentation like here:



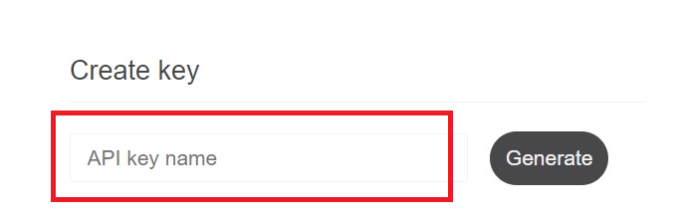
Now, here, I am using the first API for our code. So, click on the read documentation to know how it works:



Now, to use this API, we need to subscribe to it first so we can get the API key. After clicking on subscribe, it will provide pricing details. Now, here I am using a free one, so let’s get the API key:



Now, in API Keys, generate an API key with the desired name:

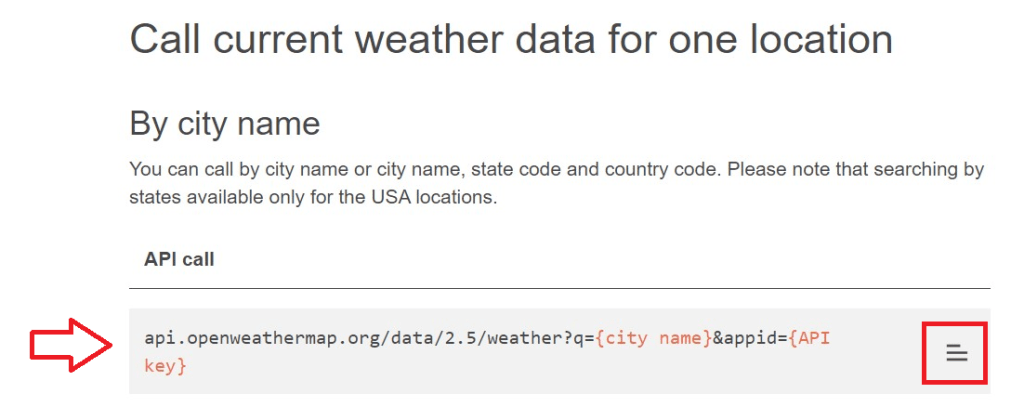
https://i2.wp.com/copyassignment.com/wp-content/uploads/2021/01/5-2.png?resize=675%2C61&ssl=1

Now you can see the API key on the right side after generating it.



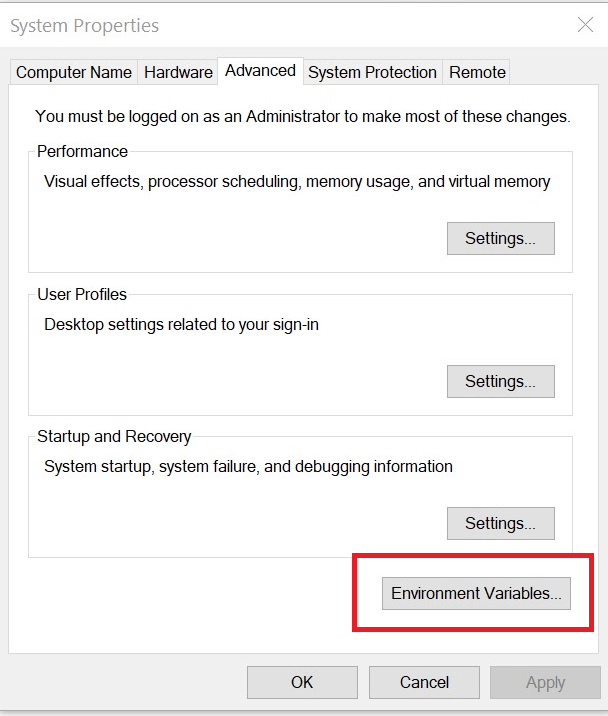
Make sure that this key will be accessed by only you, as if anyone accesses it, there will be security issues with your account. Security is Important!

Now, let’s head on to documentation again, to select how we will call the data. Here I am calling by city name only. You can select anything once you’ll understand how this will work.  
So, copy the corresponding call and paste it into your code editor for use, and yes also copy the key id.

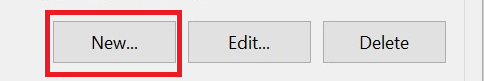


Now, for security measures, you can save this key to your system environment variable, to remain confidential.

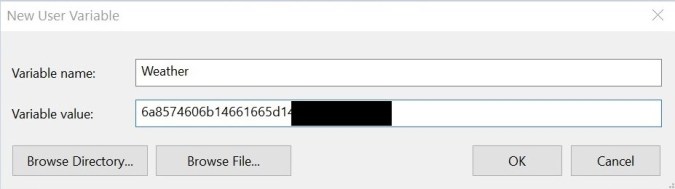
To do this, search for Environment variables in your system’s search bar. I am using windows here, so you’ll get a window of ‘edit the environment variables’ and select ‘Environment Variables’ to store the key like here:



Now select ‘New’ right after the first block. Please make sure that you won’t disturb other variables. So be careful as it can affect your system.



Now, Enter the desired variable name and set its value to the API key, and press OK.



Now, this was all about the security of our API key. Now let’s get started to code:

Do you remember the code you pasted from the documentation? Yes, it was the API link and now we will use it. From it, we can get that we require 2 parameters: city id, which will be of course dynamic here, and API key which we’ve recently stored in the environment variable.

Now, to access environment variables we need to import the os module and the variable name in which you’ve saved the key. This is how we accessed it:

import os

weather\_key = os.environ["Weather"]

We are taking input i.e. city name from the user, so, we’ll take input from the user and then modify our API link. Note, that this will be the URL we are actually passing for HTTP requests.

location = input("Enter city name: ")

api\_link = "https://api.openweathermap.org/data/2.5/weather?q="+location+"&appid="+ weather\_key

So, now we’ve to send an HTTP request, we need to import the module first and pass the request. As we discussed earlier, the data will be retrieved in JSON format for which, ‘requests’ module provides the facility to convert it into a dictionary by the ‘JSON’ method.

And as from documentation, data generated, we will print temp, pressure, humidity, speed, and description for our project. Also to make it look well we will also print the date and time. And if the user inputs an unavailable city, the dictionary has an attribute of ‘cod’, which will generate the response code as an error.

So here goes the complete code:

#importing modules

import os

import requests

from datetime import datetime

import time

#description of project

print("Hello")

time.sleep(0.5)

print("Here, you can find weather in your city!")

time.sleep(1)

#access API key

weather\_key = os.environ["Weather"]

#location from user

location = input("Enter location: ")

#URL

api\_link = "https://api.openweathermap.org/data/2.5/weather?q="+location+"&appid="+ weather\_key

#HTTP request

r=requests.get(api\_link)

#convert the data in 'r' into dictionary

data=r.json()

if(data["cod"]=="404"):

print(f"Invalid city: {location}\nEnter valid city")

exit()

else:

temp= data["main"]["temp"]

pressure = data["main"]["pressure"]

humidity = data["main"]["humidity"]

speed = data["wind"]["speed"]

descr = data["weather"][0]["description"]

date\_time = datetime.now().strftime("%D %M %Y | %H:%M:%S %p")

print("||----------------------------------------------------------------||")

print(f" Location: {location} |||| Date&Time : {date\_time}")

time.sleep(1)

print(f" Scenerio : {descr}")

time.sleep(1)

print(f" Temperature : {temp}")

time.sleep(1)

print(f" Pressure : {pressure}")

time.sleep(1)

print(f" Speed : {speed}")

time.sleep(1)

print("||----------------------------------------------------------------||")

time.sleep(1)

print("Thank You!!")