<https://www.statworx.com/en/content-hub/blog/how-to-build-a-machine-learning-api-with-python-and-flask/>

<https://towardsdatascience.com/deploying-a-machine-learning-model-as-a-rest-api-4a03b865c166>

A **Pickle** file is a data file used by a**Python script containing a Python object**. Any object in Python can be pickled, that means the object is "serialised" first before writing it to file. Pickling is a way to convert a python object (list, dict, etc.) into a character stream.

The goal of this **flask** **boilerplate** is to allow developers to quickly write their API with code structured to best practices while giving them flexibility to easily add/change features.

**app = Flask (\_\_name\_\_)** # creates the Flask instance.

\_\_name\_\_ is the name of the current Python module. The app needs to know where it’s located to set up some paths, and \_\_name\_\_ is a convenient way to tell it that.

**Example of the User Requests for URL link:**

Below are some examples of how users can access your API so that they can get predictions.

Using Jupyter Notebook:

url = '<http://127.0.0.1:5000/'>  
params ={'query': 'that movie was boring'}  
response = requests.get(url, params)  
response.json()Output: {'confidence': 0.128, 'prediction': 'Negative'}

Using curl in the terminal:

curl -X GET <http://127.0.0.1:5000/> -d query='that movie was boring'  
{  
 "prediction": "Negative",  
 "confidence": 0.128  
}

Using HTTP in the terminal:

http <http://127.0.0.1:5000/> query=='that movie was boring'

HTTP/1.0 200 OK  
Content-Length: 58  
Content-Type: application/json  
Date: Fri, 31 Aug 2018 18:49:25 GMT  
Server: Werkzeug/0.14.1 Python/3.6.3{  
 "confidence": 0.128,  
 "prediction": "Negative"  
}