
Flask app deployment Project :

Salary prediction Using Machine Learning

Name :Refka Mejri

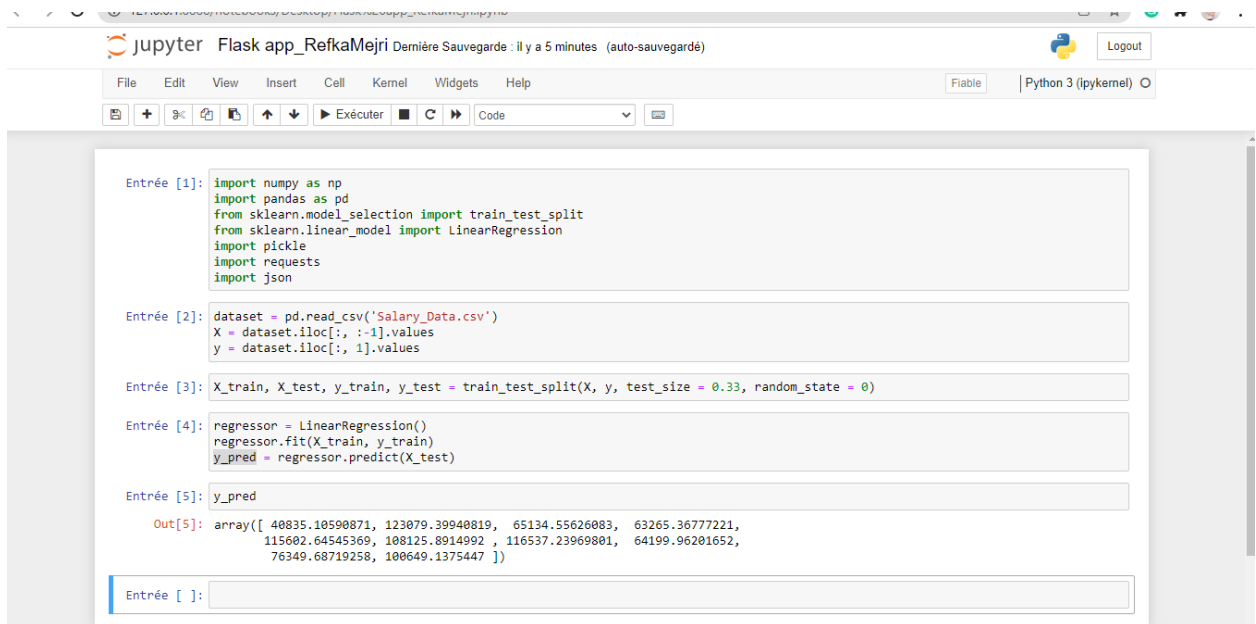
Email : Refka.mejri@enit.utm.tn

Country:Tunisia

College: National Engineering School of Tunis

Specialization :Data Science

First step: *Model prediction :*



The screenshot shows a Jupyter Notebook titled "Flask app_RefkaMejri" with a menu bar (File, Edit, View, Insert, Cell, Kernel, Widgets, Help) and a toolbar. The code is written in Python 3 (ipykernel). The workflow is as follows:

```
Entrée [1]: import numpy as np
import pandas as pd
from sklearn.model_selection import train_test_split
from sklearn.linear_model import LinearRegression
import pickle
import requests
import json

Entrée [2]: dataset = pd.read_csv('Salary_Data.csv')
X = dataset.iloc[:, :-1].values
y = dataset.iloc[:, 1].values

Entrée [3]: X_train, X_test, y_train, y_test = train_test_split(X, y, test_size = 0.33, random_state = 0)

Entrée [4]: regressor = LinearRegression()
regressor.fit(X_train, y_train)
y_pred = regressor.predict(X_test)

Entrée [5]: y_pred

Out[5]: array([ 40835.10590871, 123079.39940819,  65134.55626083,  63265.36777221,
115602.64545369, 108125.8914992 , 116537.23969801,  64199.96201652,
76349.68719258, 100649.1375447 ])
```

Second step :

After prediction, I save my model using pickle library.

Pickle is used for serialization and deserialization of python object structure.

The object regressor is saved as a file named model.pkl

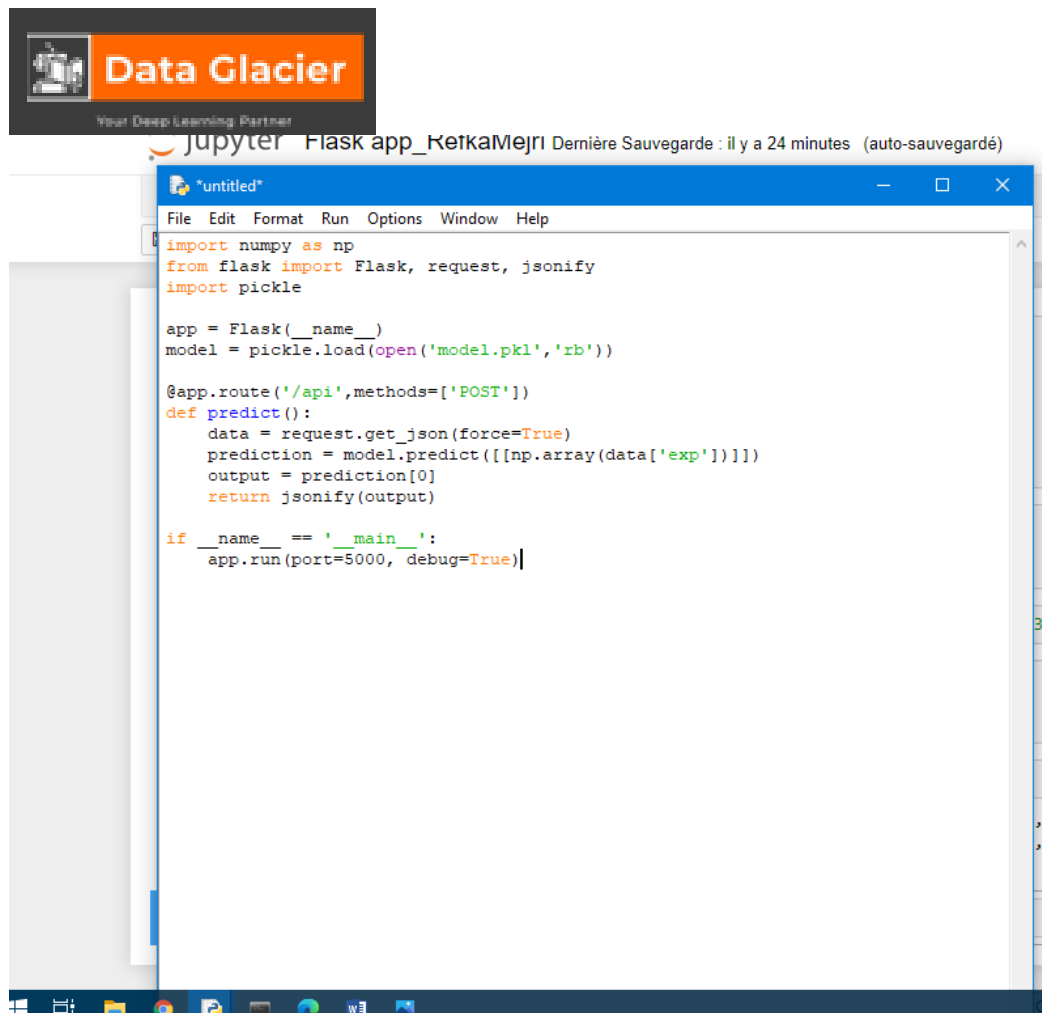


Then I prepare model.py file for training and saving my model:

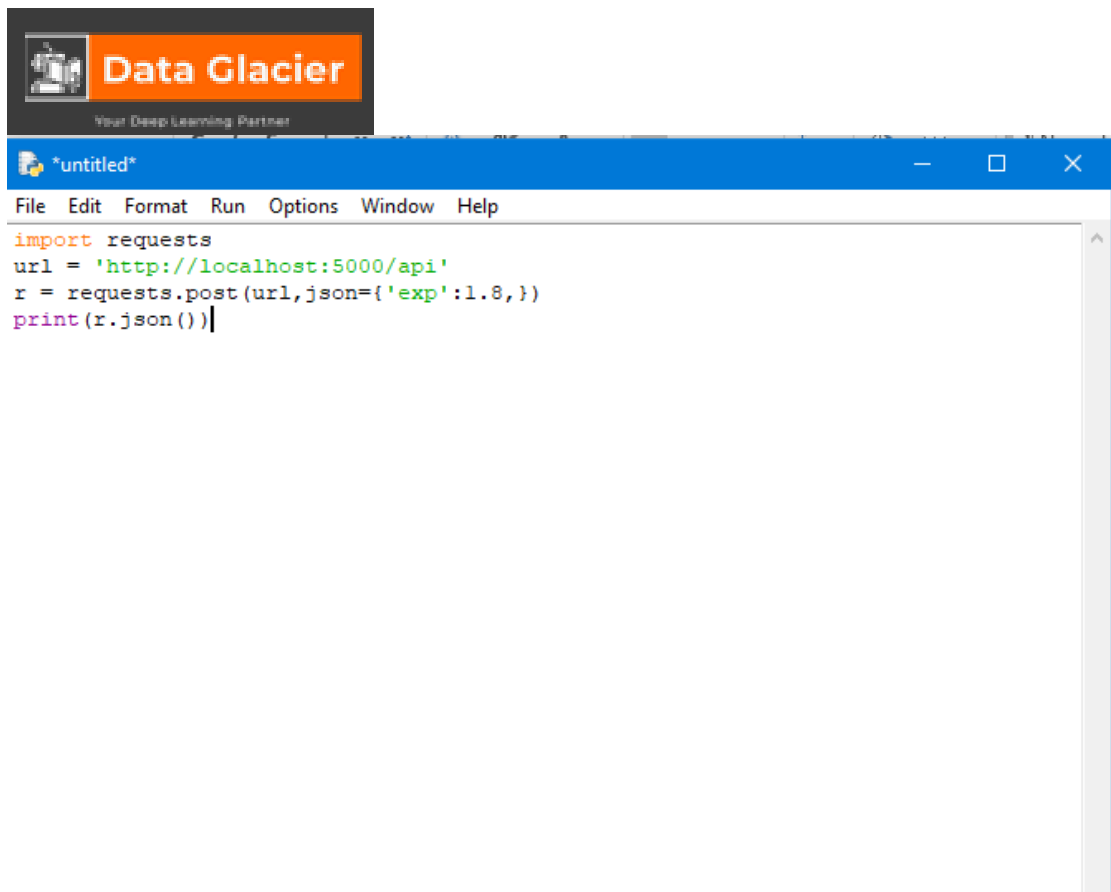
```
File Edit Format Run Options Window Help
# Importing the libraries
import numpy as np
import pandas as pd
from sklearn.model_selection import train_test_split
from sklearn.linear_model import LinearRegression
import pickle
import requests
import json
# Importing the dataset
dataset = pd.read_csv('Salary_Data.csv')
X = dataset.iloc[:, :-1].values
y = dataset.iloc[:, 1].values
# Splitting the dataset into the Training set and Test set
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size = 1/3, random_state = 0)
# Fitting Simple Linear Regression to the Training set
regressor = LinearRegression()
regressor.fit(X_train, y_train)
# Predicting the Test set results
y_pred = regressor.predict(X_test)
# Saving model to disk
pickle.dump(regressor, open('model.pkl', 'wb'))
# Loading model to compare the results
model = pickle.load(open('model.pkl', 'rb'))
print(model.predict([[1.8]]))
```

Step3 :

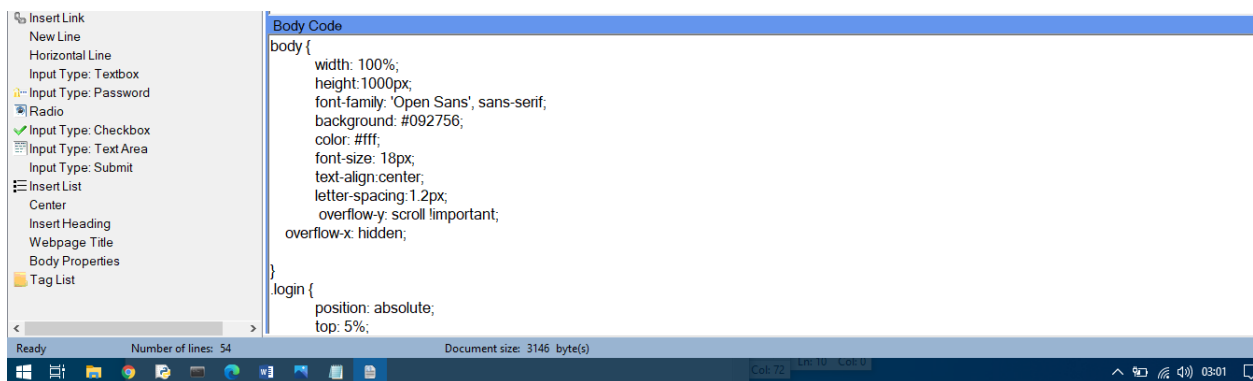
I prepare server.py file where I will use flask web framework to deal with the post request that we will get from request.py



Now our server is ready, let's add request.py file :

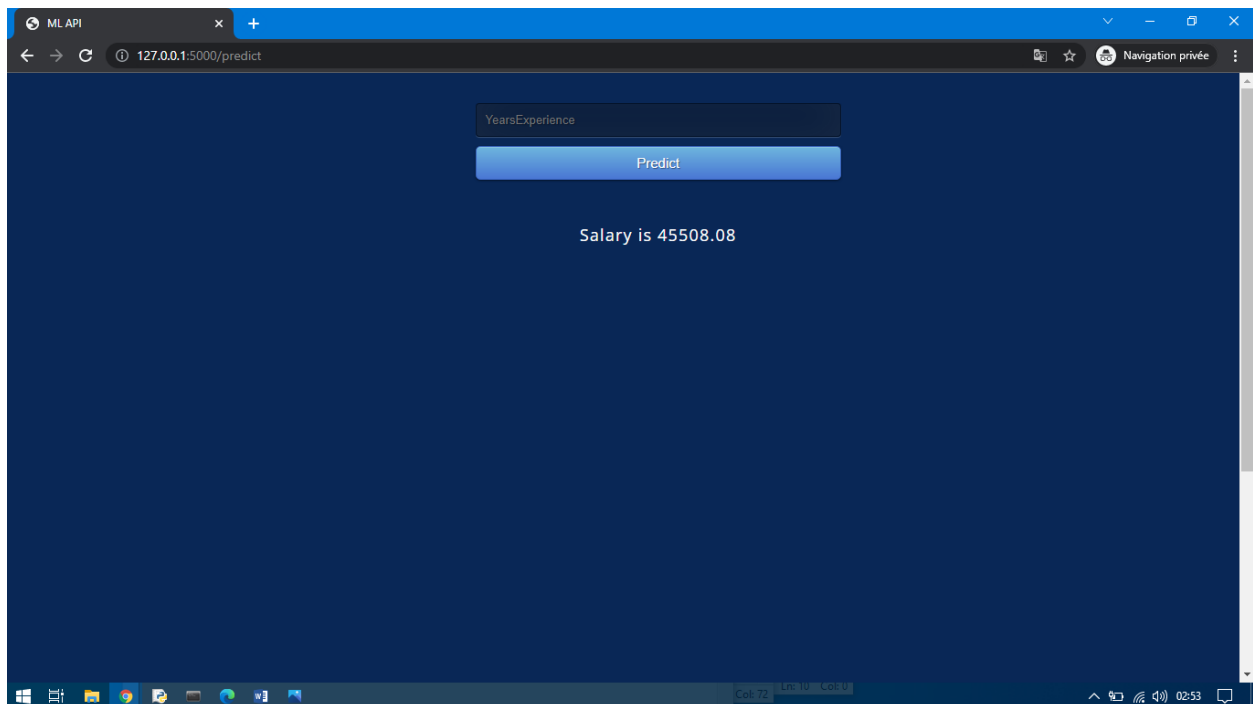


After preparing html and css file for web app.I tested my app using python app.py:





I obtain the url for the app : Running on <http://127.0.0.1:5000/>



All file are saved in the same file named : Flask-app_Salary prediction:

