Name: **Reeka Hazarika**

Batch code: **LISP01**

Submission date: **22-MAR-2021**

Submitted to: **Data Glacier**

Deployment on Flask

Step 1:

Develop ML model:

Predict the salary of an employee based on experience using Linear Regression Model.

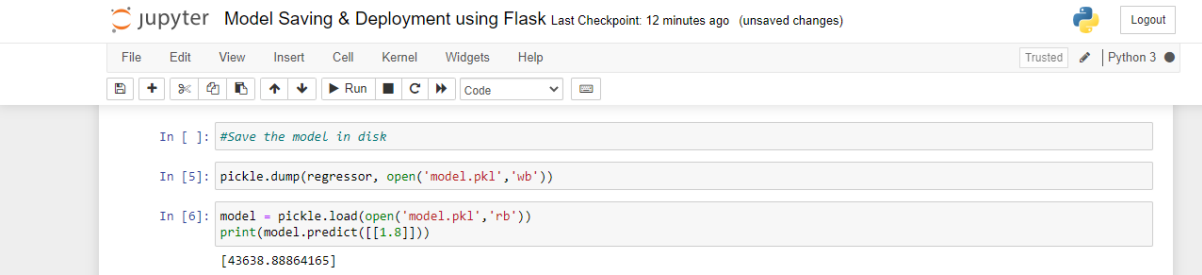
jupyter Model Saving & Deployment using Flask Last Checkpoint: 5 minutes ago (autosaved)

File Edit View Insert Cell Kernel Widgets Help

```
In [1]: import numpy as np
        import pandas as pd
        from sklearn.model_selection import train_test_split
        from sklearn.linear_model import LinearRegression
        from flask import Flask, request, jsonify
        import pickle
        import json

In [2]: salary_data = pd.read_csv('C:/Users/HAZARIKA/Desktop/Salary_Data.csv')
        X = salary_data.iloc[:, :-1].values
        y = salary_data.iloc[:, 1].values

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

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

        [ 40835.10590871 123079.39940819  65134.55626083  63265.36777221
         115602.64545369 108125.8914992  116537.23969801  64199.96201652
          76349.68719258 100649.1375447 ]
```

Step 2:

Saving the trained model to the disk using the *pickle* library.



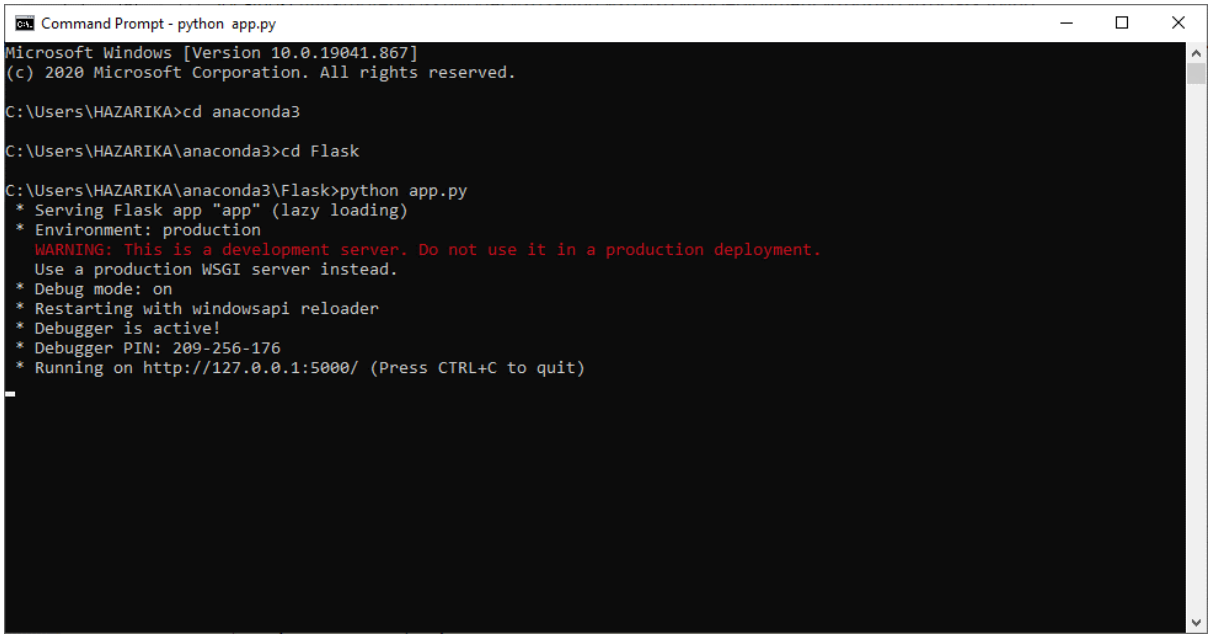
Step 3:

Deployment of Model

- Created the instance of the *Flask()* and loaded the model.
- Bounded " /" with the method *predict()* in which predict method gets the data from the json passed by the requestor.
- *model.predict()* method takes input from the json and converts it into 2D *numpy array* the results are stored into the variable named *output*.
- Return this variable after converting it into the json object using flasks *jsonify()* method.
- Run our server by following above code section and using port 5000.

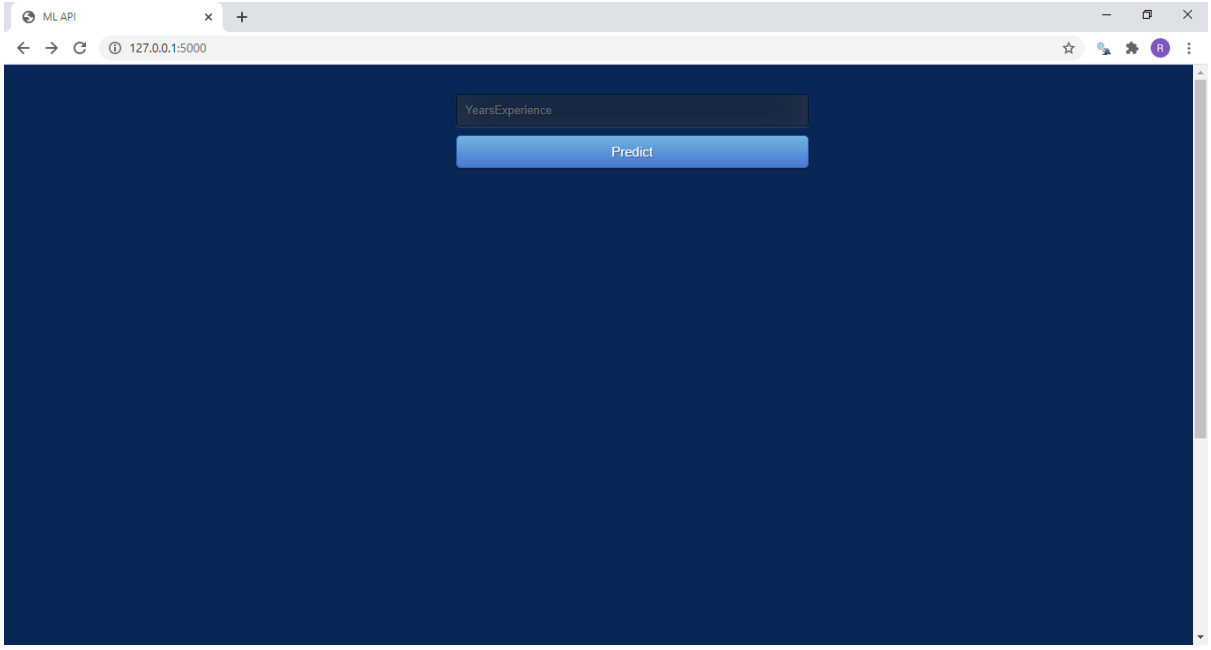
Step 4:

Checking app.py file in CMD



Step 5:

Creating the Web App by typing the URL in the browser



Thank You