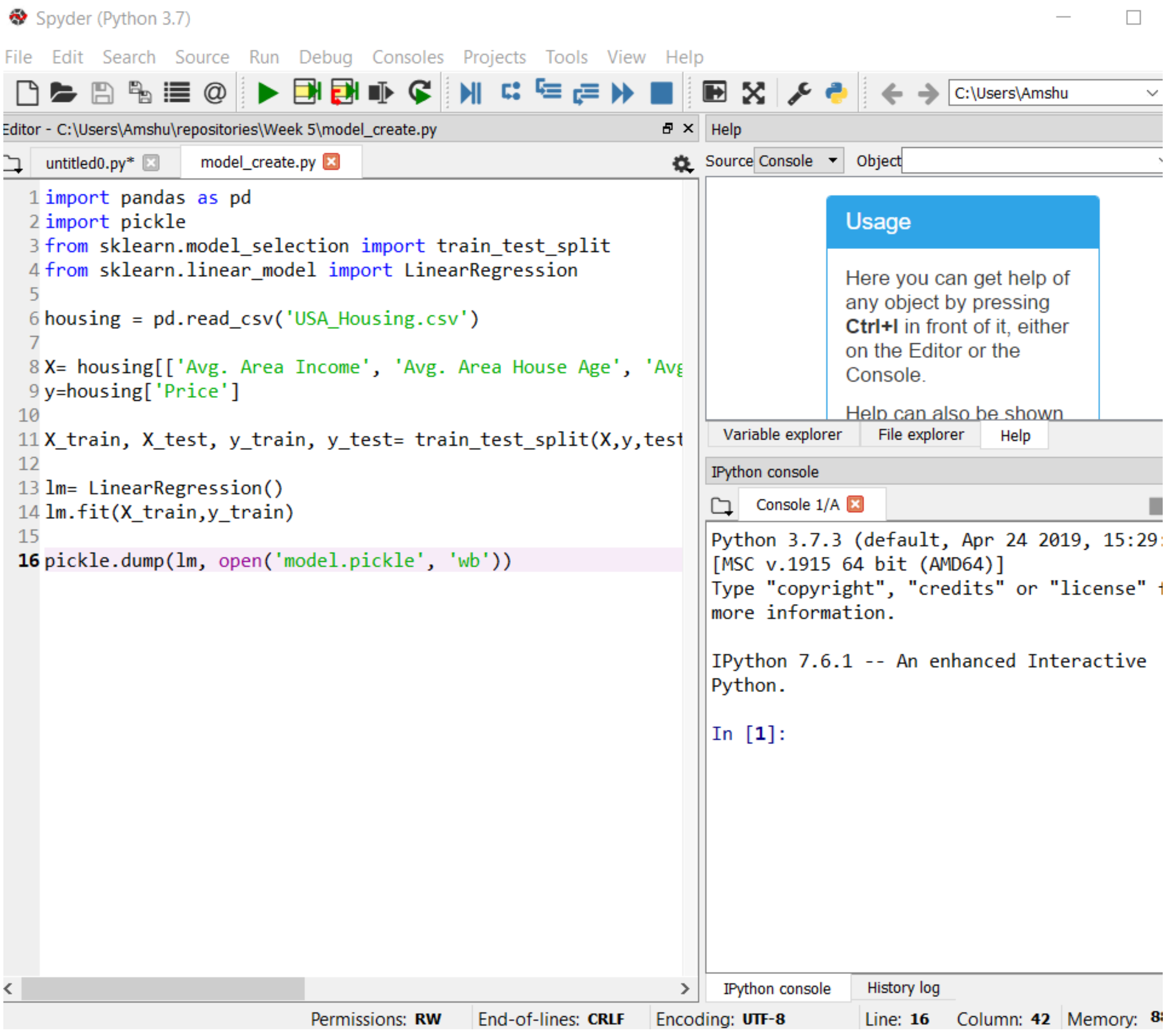
Cloud and API Deployment:

Name: Amshumann Singh

Batch: LISUM02

Submission Date: 23th Aug 2021

Step 1: Creating the dummy model which will be deployed



```
import pandas as pd
import pickle
from sklearn.model_selection import train_test_split
from sklearn.linear_model import LinearRegression

housing = pd.read_csv('USA_Housing.csv')

X= housing[['Avg. Area Income', 'Avg. Area House Age', 'Avg
y=housing['Price']

X_train, X_test, y_train, y_test= train_test_split(X,y,test

lm= LinearRegression()
lm.fit(X_train,y_train)

pickle.dump(lm, open('model.pickle', 'wb'))
```

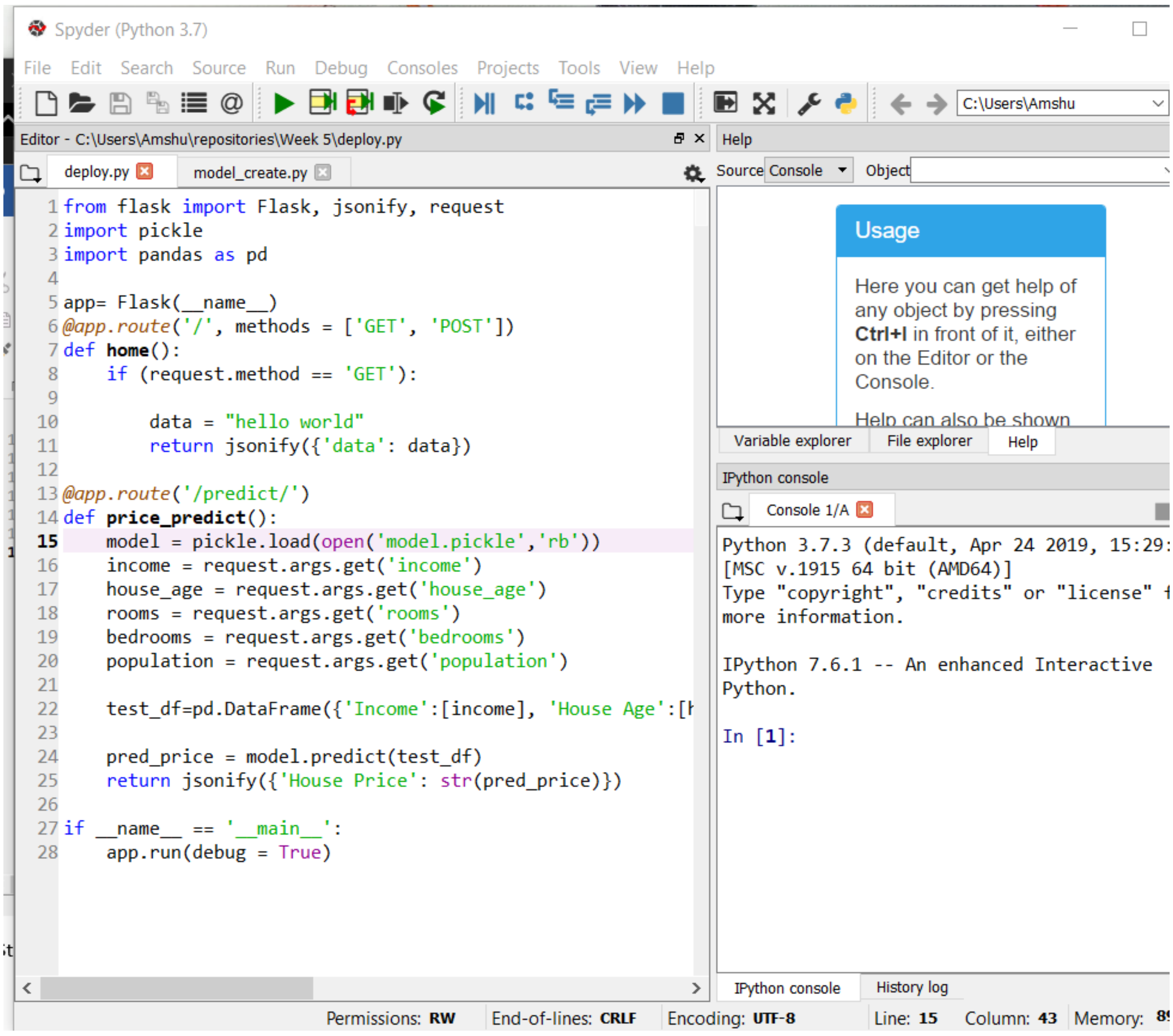
```
Python 3.7.3 (default, Apr 24 2019, 15:29
[MSC v.1915 64 bit (AMD64)]
Type "copyright", "credits" or "license"
more information.

IPython 7.6.1 -- An enhanced Interactive
Python.

In [1]:
```

Spyder (Python 3.7)
File Edit Search Source Run Debug Consoles Projects Tools View Help
C:\Users\Amshu
Editor - C:\Users\Amshu\repositories\Week 5\model_create.py
untitled0.py*
model_create.py
Help
Source Console Object
Usage
Here you can get help of any object by pressing Ctrl+I in front of it, either on the Editor or the Console.
Help can also be shown
Variable explorer File explorer Help
IPython console
Console 1/A
IPython console History log
Permissions: RW End-of-lines: CRLF Encoding: UTF-8 Line: 16 Column: 42 Memory:

Step 2: Create the flask app to deploy on Postman



```python
from flask import Flask, jsonify, request
import pickle
import pandas as pd

app= Flask(__name__)
@app.route('/', methods = ['GET', 'POST'])
def home():
    if (request.method == 'GET'):

        data = "hello world"
        return jsonify({'data': data})

@app.route('/predict/')
def price_predict():
    model = pickle.load(open('model.pickle','rb'))
    income = request.args.get('income')
    house_age = request.args.get('house_age')
    rooms = request.args.get('rooms')
    bedrooms = request.args.get('bedrooms')
    population = request.args.get('population')

    test_df=pd.DataFrame({'Income':[income], 'House Age':[h

    pred_price = model.predict(test_df)
    return jsonify({'House Price': str(pred_price)})

if __name__ == '__main__':
    app.run(debug = True)
```

Usage

Here you can get help of any object by pressing **Ctrl+I** in front of it, either on the Editor or the Console.

Help can also be shown

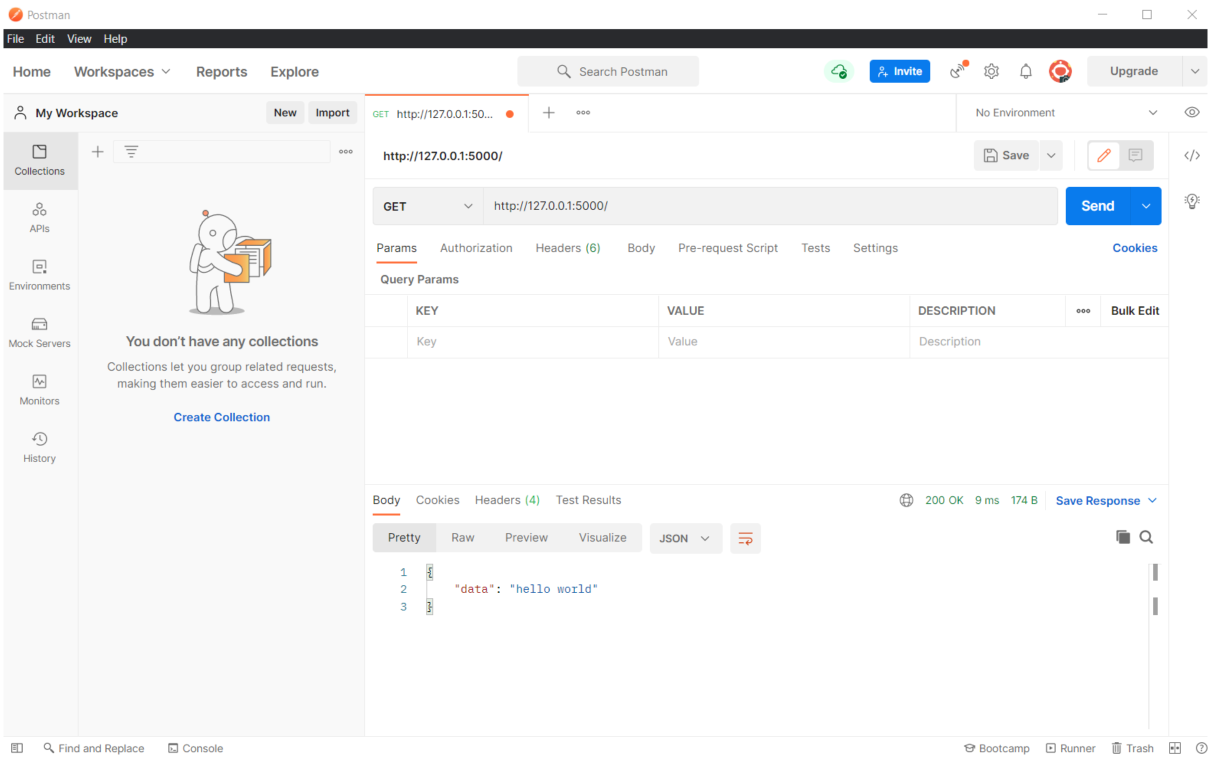
```
Python 3.7.3 (default, Apr 24 2019, 15:29:
[MSC v.1915 64 bit (AMD64)]
Type "copyright", "credits" or "license" f
more information.

IPython 7.6.1 -- An enhanced Interactive
Python.

In [1]:
```

Permissions: **RW** End-of-lines: **CRLF** Encoding: **UTF-8** Line: **15** Column: **43** Memory: **8**

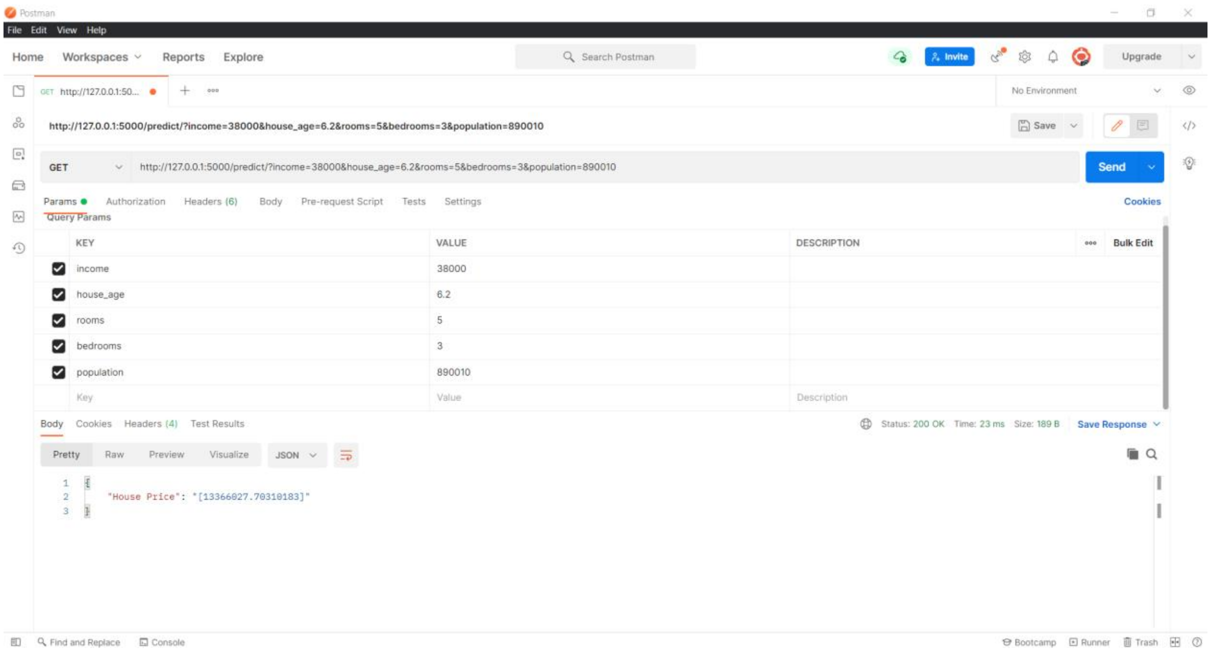
Step 3: Run Postman app and create HTTP request for the Flask app deployed on local server. First we run the root end point.


GET http://127.0.0.1:5000/

Status: 200 OK 9 ms 174 B

```
{
    "data": "hello world"
}
```

Step 4: Now we run the predict end point, fill in the key values which will be the features for the model we have deployed.


GET http://127.0.0.1:5000/predict/?income=38000&house_age=6.2&rooms=5&bedrooms=3&population=890010

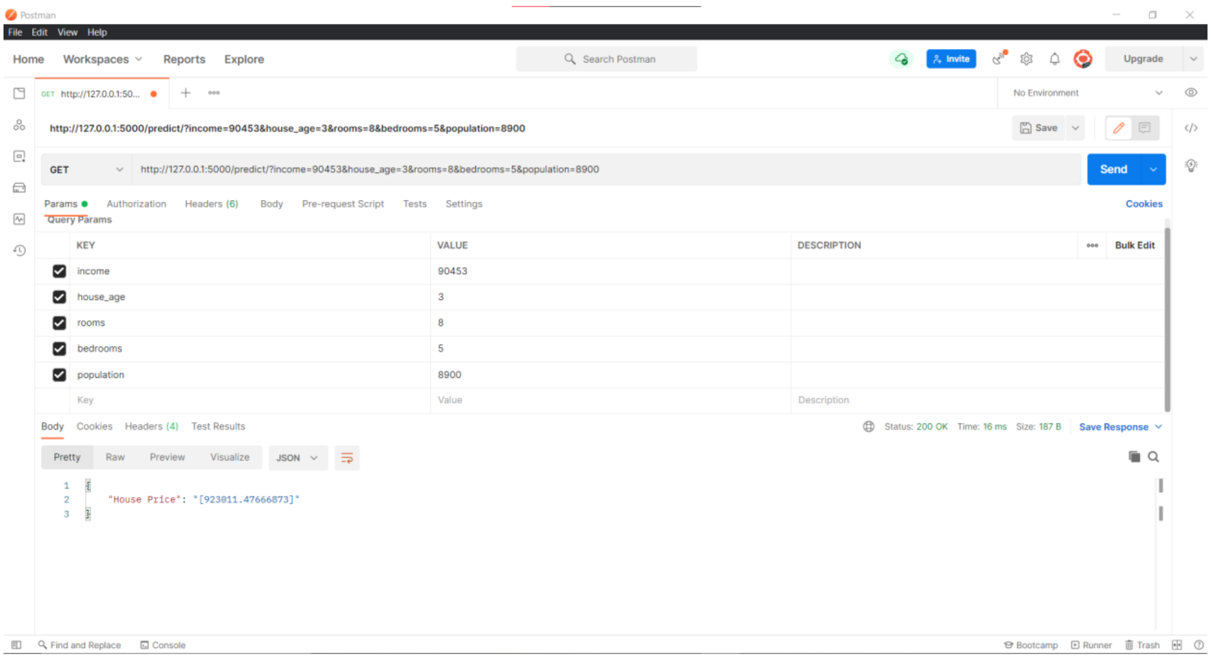
Query Params

	KEY	VALUE	DESCRIPTION
✓	income	38000	
✓	house_age	6.2	
✓	rooms	5	
✓	bedrooms	3	
✓	population	890010	

Status: 200 OK Time: 23 ms Size: 189 B

```
{
    "House Price": "[13366027.70310183]"
}
```

Step 5: Another example with different values.



```
GET http://127.0.0.1:5000/predict/?income=90453&house_age=3&rooms=8&bedrooms=5&population=8900
```

Query Params

KEY	VALUE	DESCRIPTION
income	90453	
house_age	3	
rooms	8	
bedrooms	5	
population	8900	

Status: 200 OK Time: 16 ms Size: 187 B

```
{
    "House Price": "[923011.47666873]"
}
```