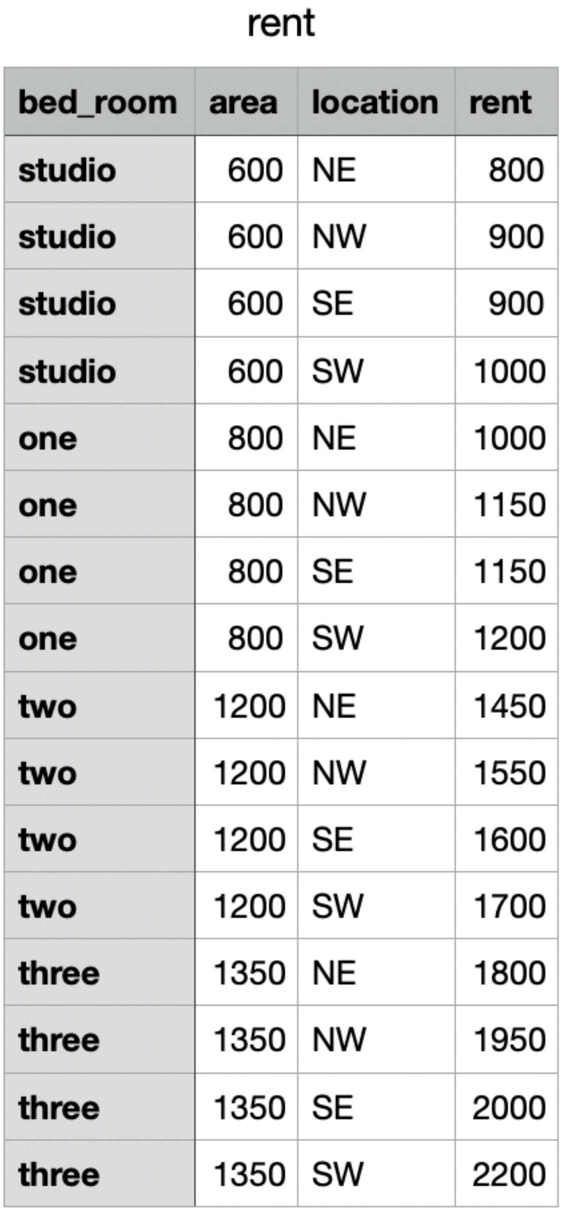
Name: Syed Ahmad Sohail
Batch Code: LISUM18
Submission Date: 28th Feb 2023
Submitted to: Data Glacier

Part 1:

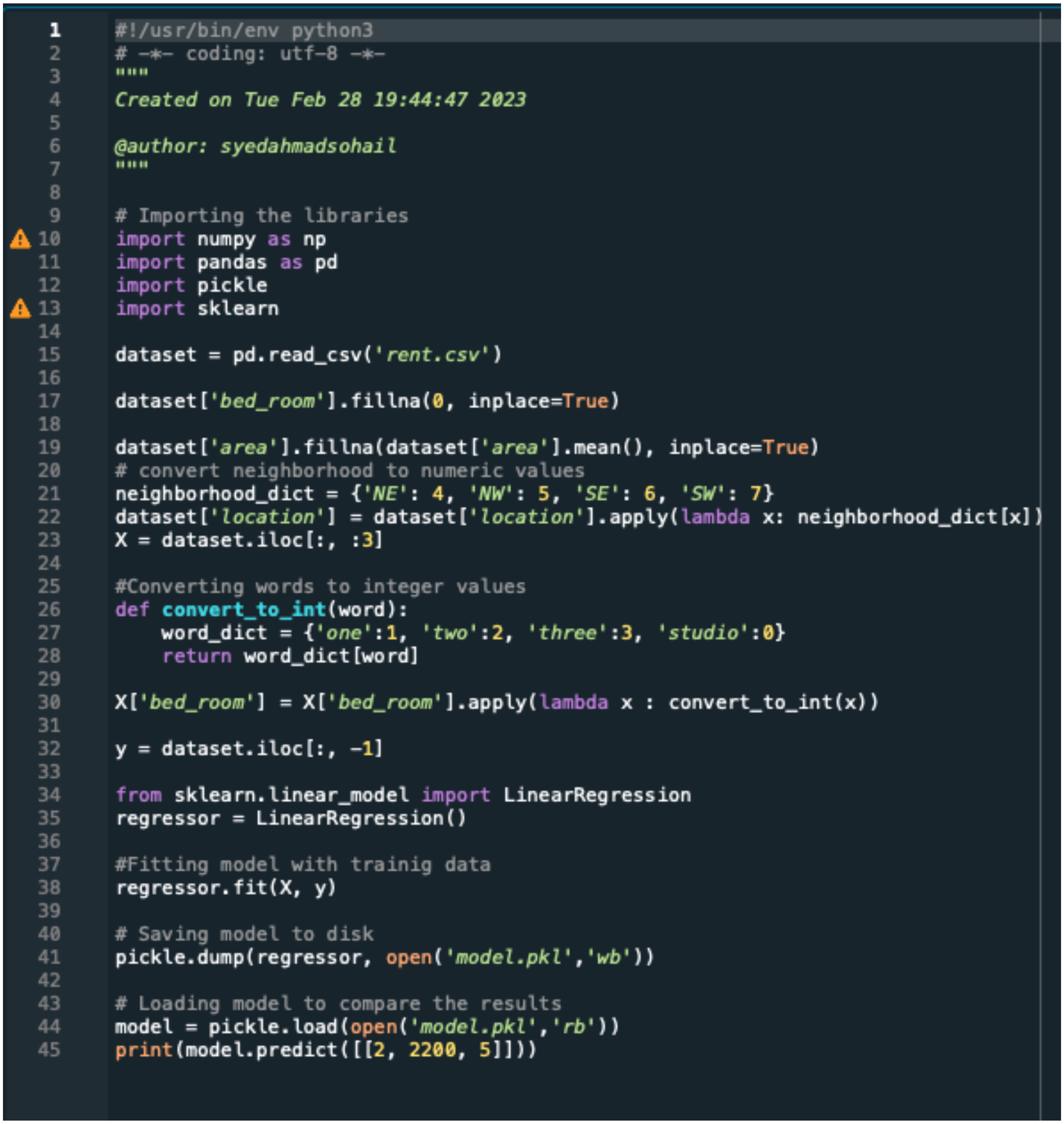


rent

bed_room	area	location	rent
studio	600	NE	800
studio	600	NW	900
studio	600	SE	900
studio	600	SW	1000
one	800	NE	1000
one	800	NW	1150
one	800	SE	1150
one	800	SW	1200
two	1200	NE	1450
two	1200	NW	1550
two	1200	SE	1600
two	1200	SW	1700
three	1350	NE	1800
three	1350	NW	1950
three	1350	SE	2000
three	1350	SW	2200

The data was initially generated by utilizing various websites, such as Rentfaster, to observe rental patterns and subsequently recording them within the file.

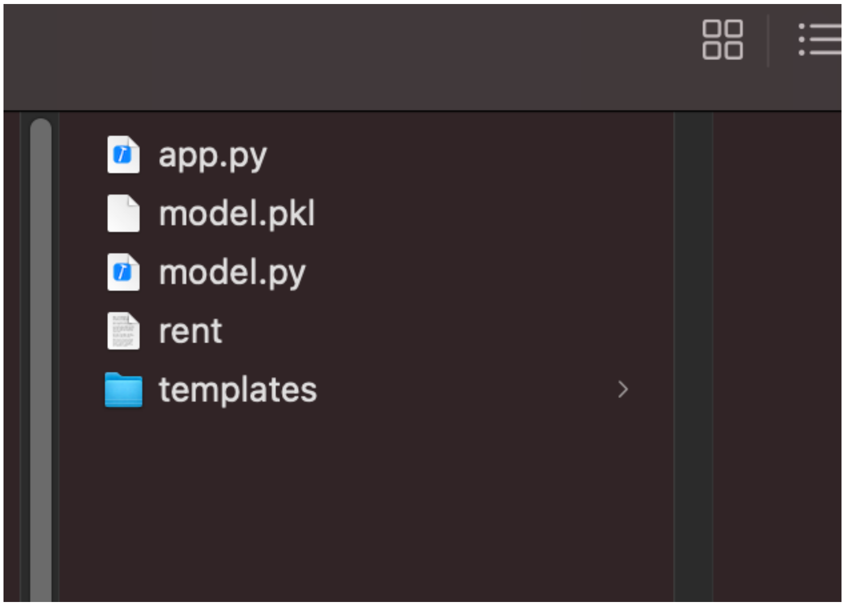
Part 2:



```python
#!/usr/bin/env python3
# -*- coding: utf-8 -*-
"""
Created on Tue Feb 28 19:44:47 2023

@author: syedahmadsohail
"""


# Importing the libraries
import numpy as np
import pandas as pd
import pickle
import sklearn

dataset = pd.read_csv('rent.csv')

dataset['bed_room'].fillna(0, inplace=True)

dataset['area'].fillna(dataset['area'].mean(), inplace=True)
# convert neighborhood to numeric values
neighborhood_dict = {'NE': 4, 'NW': 5, 'SE': 6, 'SW': 7}
dataset['location'] = dataset['location'].apply(lambda x: neighborhood_dict[x])
X = dataset.iloc[:, :3]

#Converting words to integer values
def convert_to_int(word):
    word_dict = {'one':1, 'two':2, 'three':3, 'studio':0}
    return word_dict[word]

X['bed_room'] = X['bed_room'].apply(lambda x : convert_to_int(x))

y = dataset.iloc[:, -1]

from sklearn.linear_model import LinearRegression
regressor = LinearRegression()

#Fitting model with trainig data
regressor.fit(X, y)

# 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([[2, 2200, 5]]))
```

Secondly, a data model was developed to make predictions using the generated data.

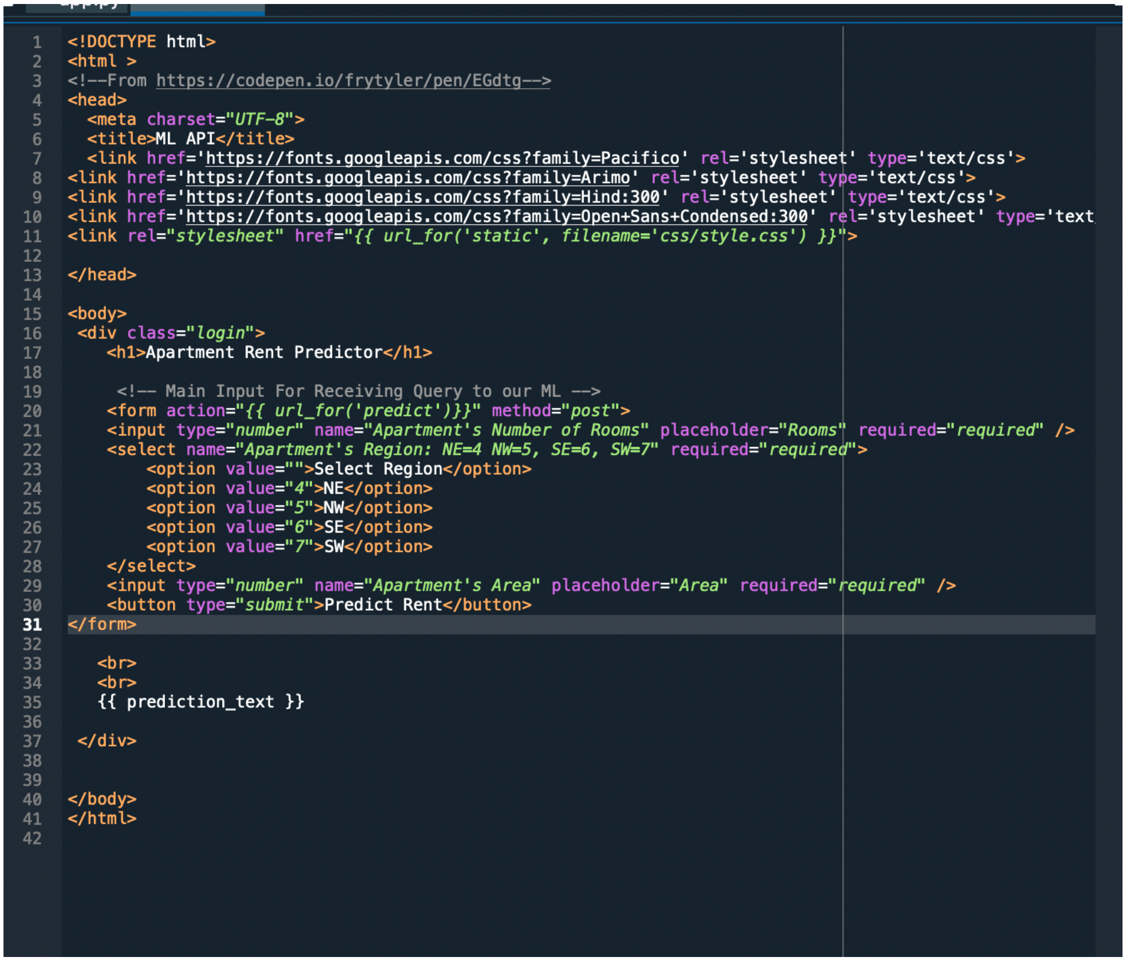
Part 3:



app.py
model.pkl
model.py
rent
templates

Model file was saved to be deployed in web app.

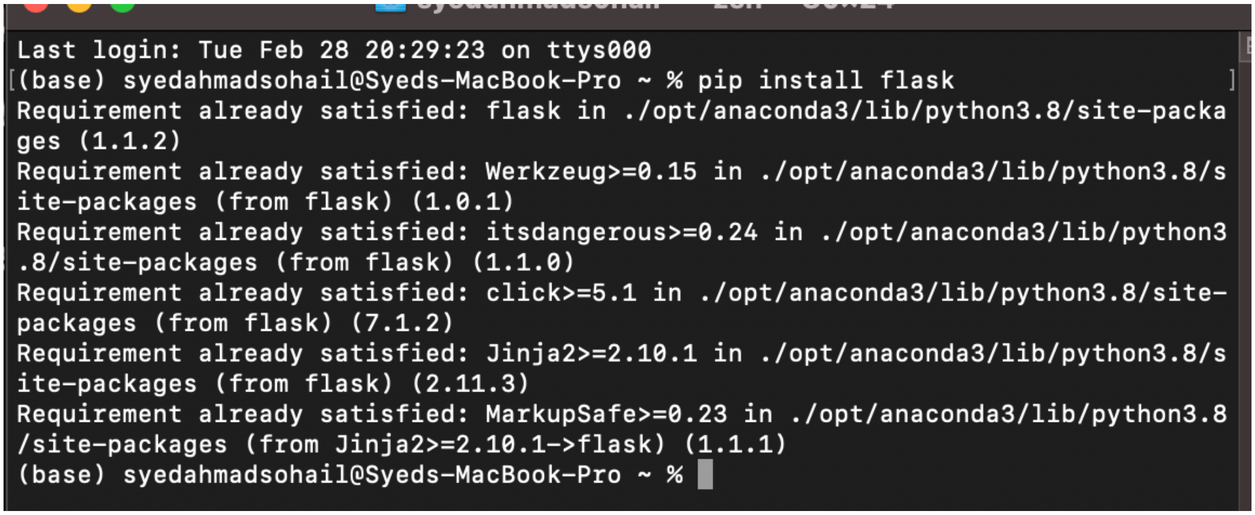
Part 4:



```
<!DOCTYPE html>
<html >
<!--From https://codepen.io/frytyler/pen/EGdtg-->
<head>
  <meta charset="UTF-8">
  <title>ML API</title>
  <link href='https://fonts.googleapis.com/css?family=Pacifico' rel='stylesheet' type='text/css'>
<link href='https://fonts.googleapis.com/css?family=Arimo' rel='stylesheet' type='text/css'>
<link href='https://fonts.googleapis.com/css?family=Hind:300' rel='stylesheet' type='text/css'>
<link href='https://fonts.googleapis.com/css?family=Open+Sans+Condensed:300' rel='stylesheet' type='text
<link rel="stylesheet" href="{{ url_for('static', filename='css/style.css') }}">

</head>

<body>
 <div class="login">
    <h1>Apartment Rent Predictor</h1>

     <!-- Main Input For Receiving Query to our ML -->
    <form action="{{ url_for('predict')}}" method="post">
    <input type="number" name="Apartment's Number of Rooms" placeholder="Rooms" required="required" />
    <select name="Apartment's Region: NE=4 NW=5, SE=6, SW=7" required="required">
        <option value="">Select Region</option>
        <option value="4">NE</option>
        <option value="5">NW</option>
        <option value="6">SE</option>
        <option value="7">SW</option>
    </select>
    <input type="number" name="Apartment's Area" placeholder="Area" required="required" />
    <button type="submit">Predict Rent</button>
</form>

   <br>
   <br>
   {{ prediction_text }}

 </div>


</body>
</html>
```

HTML file for deployment was created, I wrote the code using a text editor on the similar file, structured it with HTML tags such as <html>, <head>, and <body>, added necessary CSS and JavaScript code, and saved the file with the .html extension.

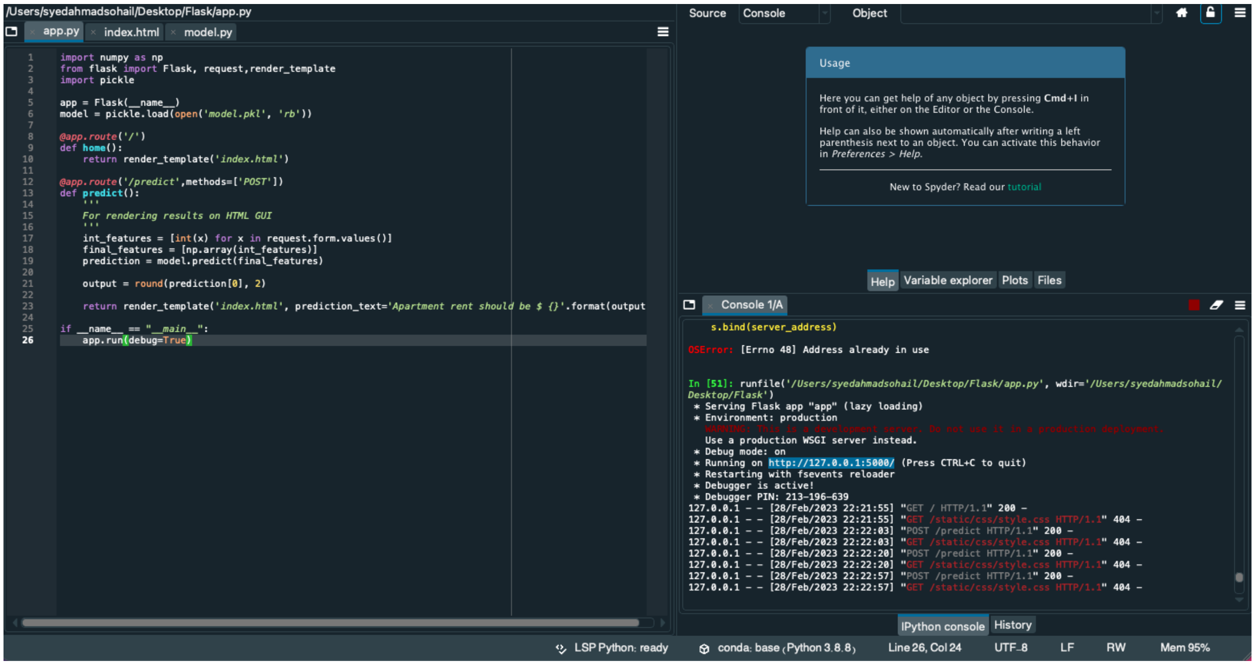
Part 5:



```
Last login: Tue Feb 28 20:29:23 on ttys000
(base) syedahmadsohail@Syeds-MacBook-Pro ~ % pip install flask
Requirement already satisfied: flask in ./opt/anaconda3/lib/python3.8/site-packa
ges (1.1.2)
Requirement already satisfied: Werkzeug>=0.15 in ./opt/anaconda3/lib/python3.8/s
ite-packages (from flask) (1.0.1)
Requirement already satisfied: itsdangerous>=0.24 in ./opt/anaconda3/lib/python3
.8/site-packages (from flask) (1.1.0)
Requirement already satisfied: click>=5.1 in ./opt/anaconda3/lib/python3.8/site-
packages (from flask) (7.1.2)
Requirement already satisfied: Jinja2>=2.10.1 in ./opt/anaconda3/lib/python3.8/s
ite-packages (from flask) (2.11.3)
Requirement already satisfied: MarkupSafe>=0.23 in ./opt/anaconda3/lib/python3.8
/site-packages (from Jinja2>=2.10.1->flask) (1.1.1)
(base) syedahmadsohail@Syeds-MacBook-Pro ~ %
```

Flask was installed using pip install flask, showed already installed as used previously on the system.

Part 6:



```python
import numpy as np
from flask import Flask, request,render_template
import pickle

app = Flask(__name__)
model = pickle.load(open('model.pkl', 'rb'))

@app.route('/')
def home():
    return render_template('index.html')

@app.route('/predict',methods=['POST'])
def predict():
    '''
    For rendering results on HTML GUI
    '''
    int_features = [int(x) for x in request.form.values()]
    final_features = [np.array(int_features)]
    prediction = model.predict(final_features)

    output = round(prediction[0], 2)

    return render_template('index.html', prediction_text='Apartment rent should be $ {}'.format(output

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

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

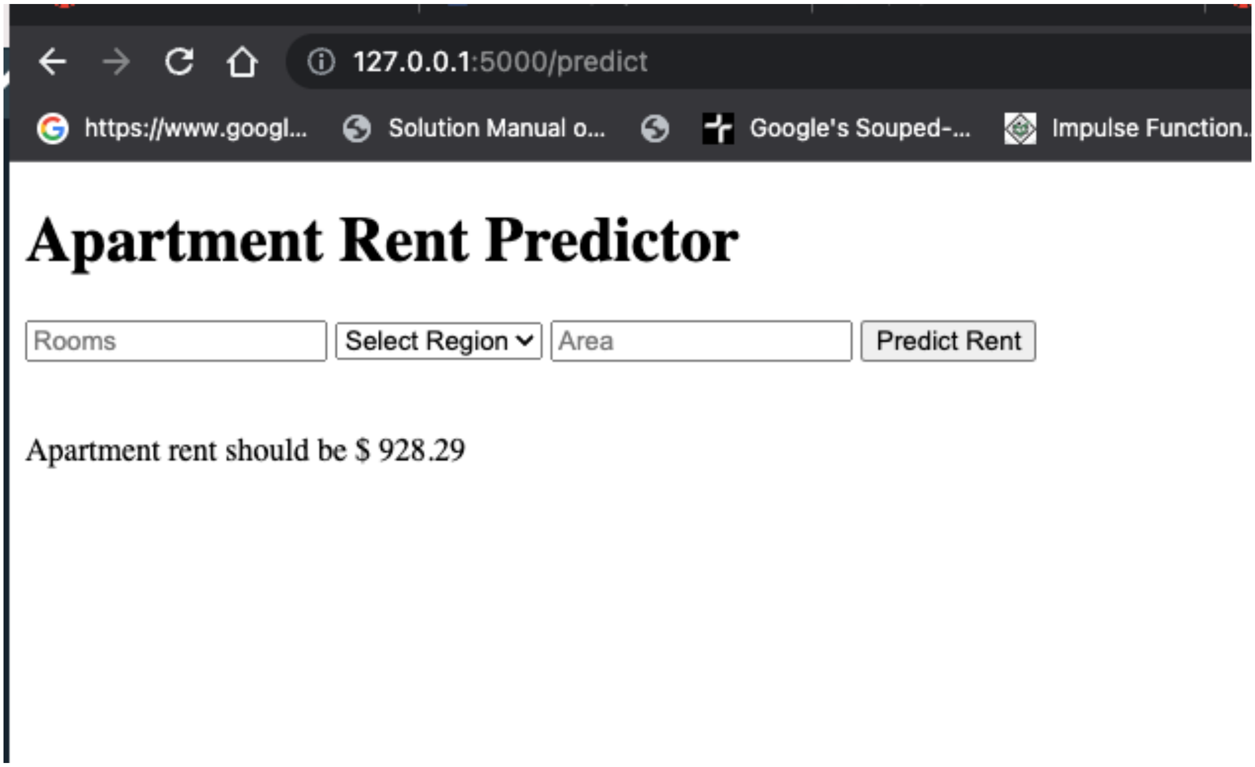
Help can also be shown automatically after writing a left parenthesis next to an object. You can activate this behavior in *Preferences > Help*.

New to Spyder? Read our tutorial

```
    s.bind(server_address)

OSError: [Errno 48] Address already in use


In [51]: runfile('/Users/syedahmadsohail/Desktop/Flask/app.py', wdir='/Users/syedahmadsohail/
Desktop/Flask')
 * Serving Flask app "app" (lazy loading)
 * Environment: production
   WARNING: This is a development server. Do not use it in a production deployment.
   Use a production WSGI server instead.
 * Debug mode: on
 * Running on http://127.0.0.1:5000/ (Press CTRL+C to quit)
 * Restarting with fsevents reloader
 * Debugger is active!
 * Debugger PIN: 213-196-639
127.0.0.1 - - [28/Feb/2023 22:21:55] "GET / HTTP/1.1" 200 -
127.0.0.1 - - [28/Feb/2023 22:21:55] "GET /static/css/style.css HTTP/1.1" 404 -
127.0.0.1 - - [28/Feb/2023 22:22:03] "POST /predict HTTP/1.1" 200 -
127.0.0.1 - - [28/Feb/2023 22:22:03] "GET /static/css/style.css HTTP/1.1" 404 -
127.0.0.1 - - [28/Feb/2023 22:22:20] "POST /predict HTTP/1.1" 200 -
127.0.0.1 - - [28/Feb/2023 22:22:20] "GET /static/css/style.css HTTP/1.1" 404 -
127.0.0.1 - - [28/Feb/2023 22:22:57] "POST /predict HTTP/1.1" 200 -
127.0.0.1 - - [28/Feb/2023 22:22:57] "GET /static/css/style.css HTTP/1.1" 404 -
```

This code was run which uses a trained machine learning model from a saved file, creates a Flask web application, defines routes for the home page and a prediction page, and renders the prediction results on an HTML template.

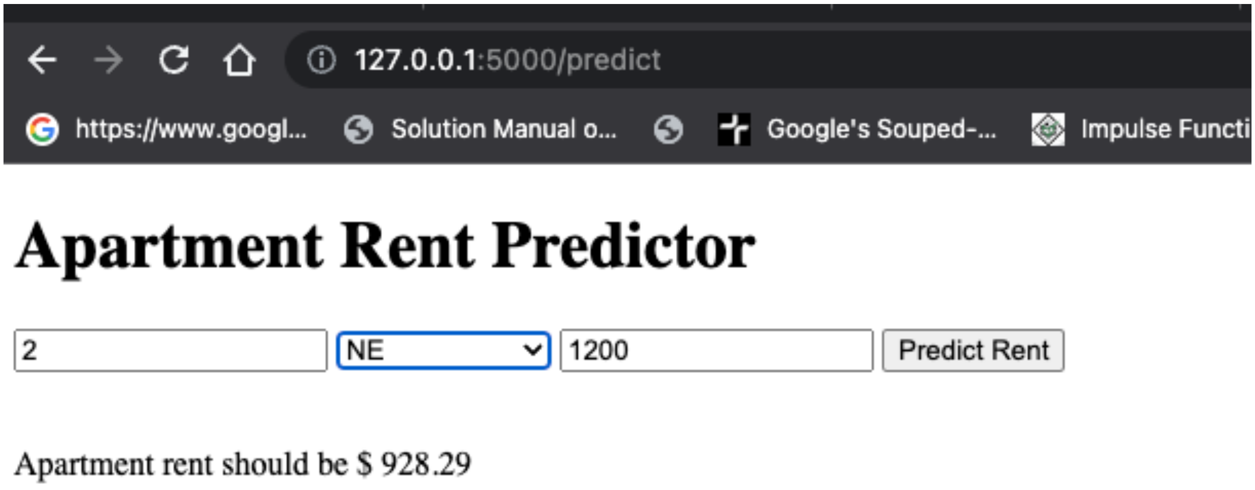
Part 7:



Apartment Rent Predictor

Rooms | Select Region | Area

Apartment rent should be $ 928.29



Apartment Rent Predictor

2 | NE | 1200

Apartment rent should be $ 928.29

A web app was created which was able to take region, rooms and area value to calculate rent in Calgary Area.