

Report

Name: Week 5

Report date: December 2nd, 2022

Internship Batch: LISUM15

Version: 1.0

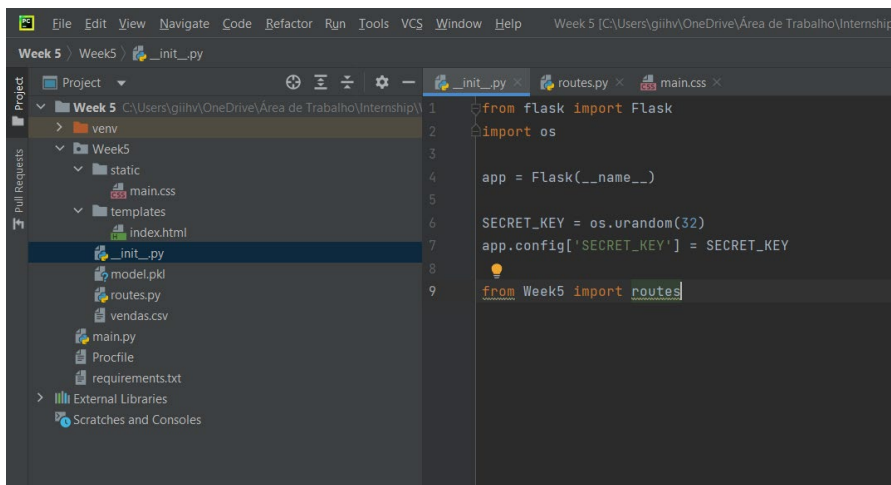
Data intake by: Giovanna Vieira

Data intake reviewer:

Data storage location: GitHub & Heroku

Step by Step – Deployment

1. Creating app



```
from flask import Flask
import os

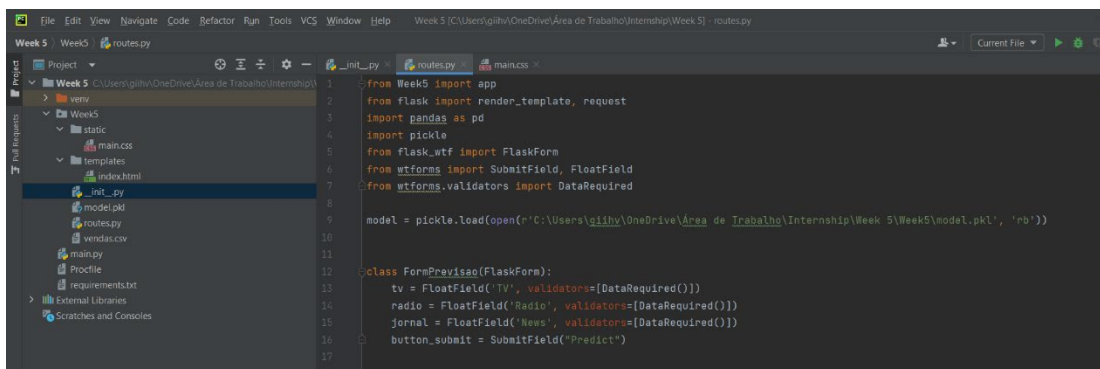
app = Flask(__name__)

SECRET_KEY = os.urandom(32)
app.config['SECRET_KEY'] = SECRET_KEY

from Week5 import routes
```

The screenshot shows the Visual Studio Code editor with a project named 'Week 5'. The file explorer on the left shows the project structure, including a 'venv' directory and files like 'main.py', 'routes.py', and 'vendas.csv'. The main editor window displays the code in 'Week5/_init_.py', which imports Flask, sets a secret key, and imports the 'routes' module from 'Week5'.

2. Importing libraries and Pickle model; creating the Flask Form.



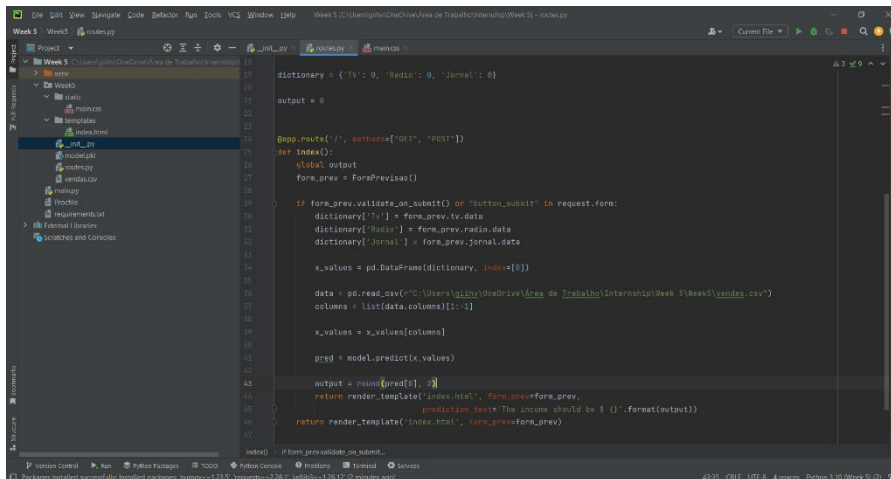
```
from Week5 import app
from flask import render_template, request
import pandas as pd
import pickle
from flask_wtf import FlaskForm
from wtforms import SubmitField, FloatField
from wtforms.validators import DataRequired

model = pickle.load(open(r'C:\Users\giihv\OneDrive\Área de Trabalho\Internship\Week 5\Week5\model.pkl', 'rb'))

class FormPrevisao(FlaskForm):
    tv = FloatField('TV', validators=[DataRequired()])
    radio = FloatField('Radio', validators=[DataRequired()])
    jornal = FloatField('News', validators=[DataRequired()])
    button_submit = SubmitField("Predict")
```

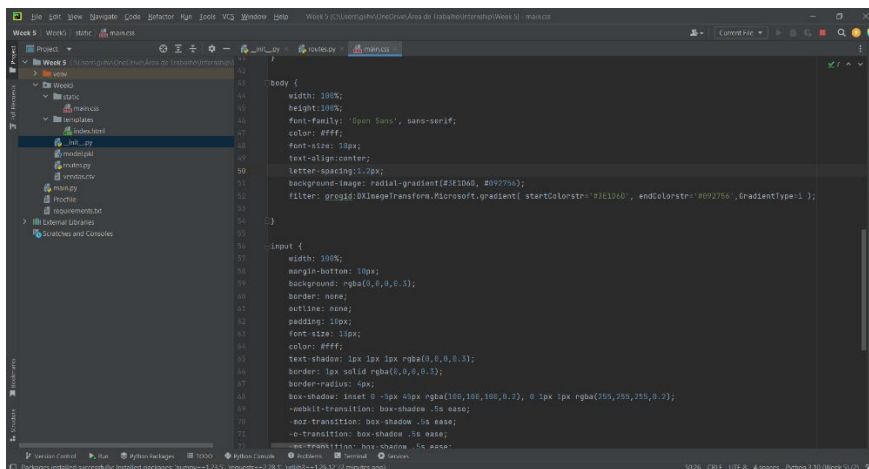
The screenshot shows the 'routes.py' file in the 'Week 5' project. It imports the 'app' from 'Week5' and various libraries for rendering templates, handling requests, and creating forms. It also loads a pickle model. A 'FormPrevisao' class is defined, inheriting from 'FlaskForm', with fields for 'TV', 'Radio', and 'News', each with a 'DataRequired' validator, and a 'button_submit' field of type 'SubmitField' with the label 'Predict'.

3. Creating functions and routes



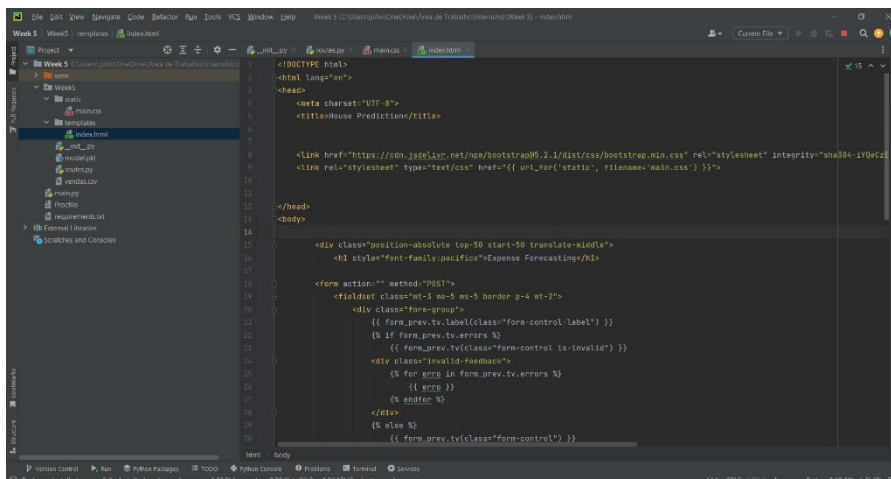
```
18 dictionary = {'TV': 0, 'Radio': 0, 'Jornal': 0}
19
20 output = 0
21
22 @app.route('/', methods=['GET', 'POST'])
23 def index():
24     global output
25     form_prev = FormPrevious()
26
27     if form_prev.validate_on_submit() or "button_submit" in request.form:
28         dictionary['TV'] = form_prev.tv.data
29         dictionary['Radio'] = form_prev.radio.data
30         dictionary['Jornal'] = form_prev.jornal.data
31
32         x_values = pd.DataFrame(dictionary, index=[0])
33
34         data = pd.read_csv(r"C:\Users\gabri\OneDrive\Arquivos de Trabalho\Internship\Week 5\Week5\ vendas.csv")
35         columns = list(data.columns)[1:-1]
36
37         x_values = x_values[columns]
38
39         pred = model.predict(x_values)
40
41         output = round(pred[0], 2)
42
43         return render_template("index.html", form_prev=form_prev,
44                               prediction_text=f"The income should be ${0}.format(output))
45         return render_template("index.html", form_prev=form_prev)
```

4. "main.css" created



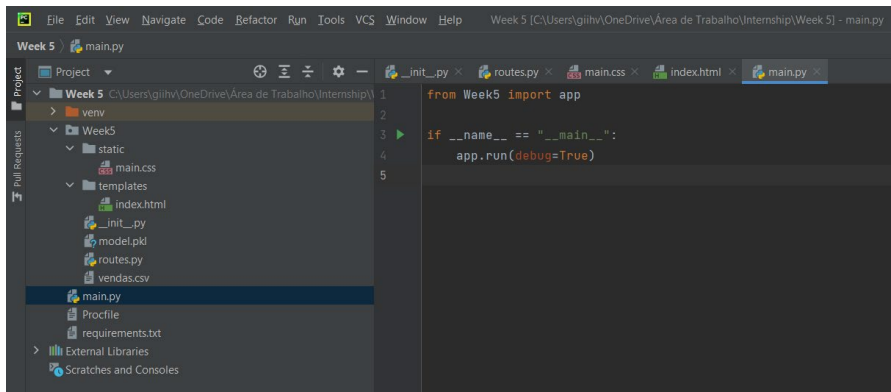
```
1 body {
2     width: 100%;
3     height: 100%;
4     font-family: 'Open Sans', sans-serif;
5     color: #fff;
6     text-align: center;
7     letter-spacing: 1.2px;
8     background-image: radial-gradient(circle at 100% 100%, #000 0%, #000 99%, #000 100%);
9     filter: progid:DXImageTransform.Microsoft.gradient( startColorstr='000000', endColorstr='000000', gradientType=1 );
10 }
11
12 input {
13     width: 100%;
14     margin-bottom: 10px;
15     background: rgba(0,0,0,0.3);
16     border: none;
17     outline: none;
18     padding: 10px;
19     font-size: 16px;
20     color: #fff;
21     text-shadow: 1px 1px 1px rgba(0,0,0,0.3);
22     border: 1px solid rgba(0,0,0,0.3);
23     border-radius: 5px;
24     box-shadow: inset 0 -1px 0px rgba(100,100,100,0.3), 0 1px 1px rgba(255,255,255,0.2);
25     -webkit-transition: box-shadow .5s ease;
26     -moz-transition: box-shadow .5s ease;
27     -o-transition: box-shadow .5s ease;
28     transition: box-shadow .5s ease;
29 }
```

5. 'index.html' template created

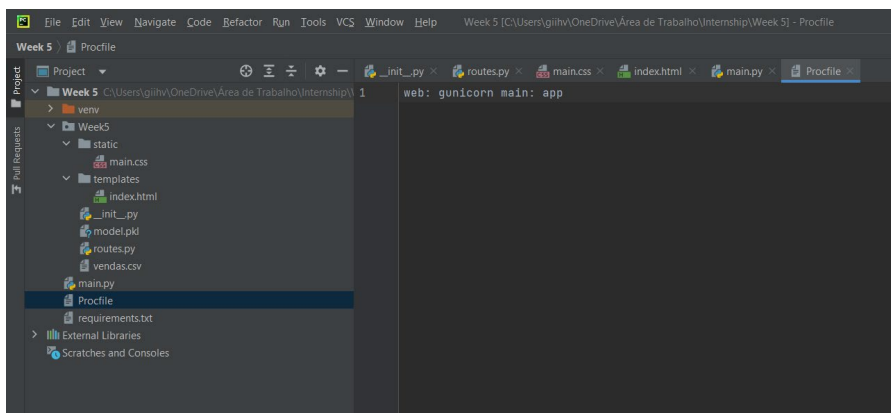


```
1 <!DOCTYPE html>
2 <html lang="en">
3 <head>
4     <meta charset="UTF-8">
5     <title>House Prediction</title>
6
7     <link href="https://cdn.jsdelivr.net/npm/bootstrap@5.2.1/dist/css/bootstrap.min.css" rel="stylesheet" integrity="sha384-170z74fcII5tqb3t67BJzhlgAS-U2fLzqBM506EI8LzqN1wagZ4cX5o3tR3" crossorigin="anonymous">
8     <link rel="stylesheet" type="text/css" href="{{ url_for('static', filename='main.css') }}">
9
10 </head>
11 <body>
12
13     <div class="position-absolute top-50 start-50 translate-middle">
14         <h1 style="font-family:sans-serif">Expense Forecasting</h1>
15
16         <form action="" method="POST">
17             <fieldset class="mb-5 mx-5 mx-sm-5 border p-4 mt-2">
18                 <div class="form-group">
19                     {{ form_prev.tv.label(class="form-control-label") }}
20                     {{ form_prev.tv.errors }}
21                     {{ form_prev.tv(class="form-control is-invalid") }}
22
23                     <div class="invalid-feedback">
24                         {{ form_prev.tv.errors }}
25                     </div>
26                 </div>
27                 {{ form_prev.radio(class="form-control") }}
28             </fieldset>
29         </form>
30     </div>
```

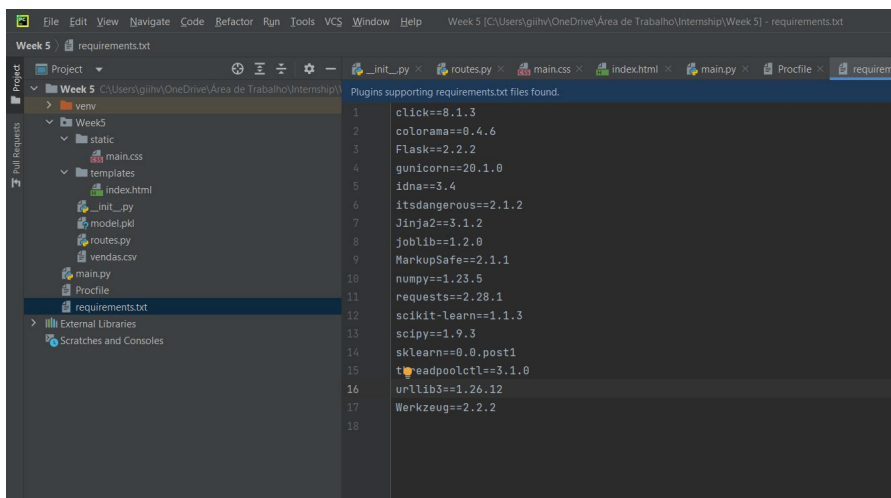
6. Running and Testing the app

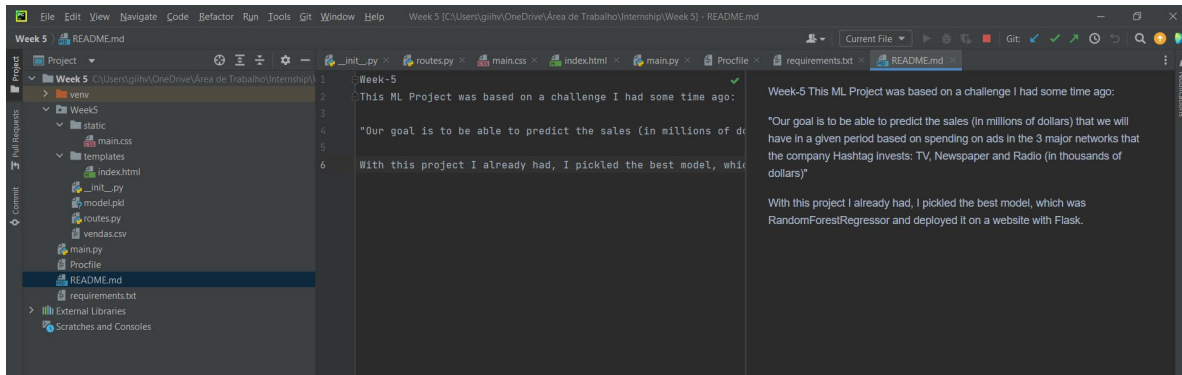


7. 'Procfile' created

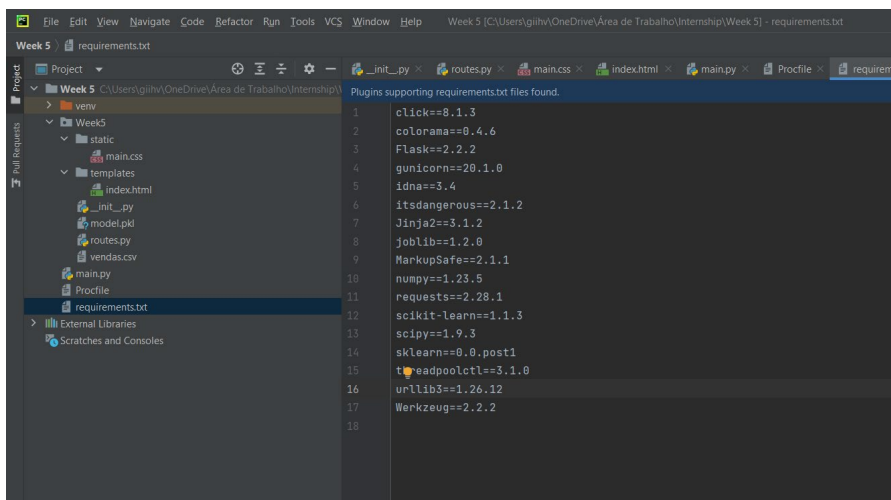


8. Creation of 'requirements.txt' and 'README.md'





9. GitHub Deployment



10. Heroku Deployment

