



INNOVATION. AUTOMATION. ANALYTICS

PROJECT ON

Code Refactoring and Bug Fixing

About me

I'm Gowrabathina Rakesh, a passionate Data Analyst. With expertise in EDA, SQL, PowerBI, I specialize in deriving actionable insights from complex datasets. I thrive on solving challenging problems and stay updated with the latest advancements in data science. Collaborative by nature, I enjoy working in dynamic teams to drive innovation.

Scenario

A team of enthusiastic data scientists embarked on a mission to develop a Note Taking application using Python, Flask, and HTML. However, their lack of experience in backend development has led to challenges in making the application fully functional. Recognizing your proficiency in backend development, you have been tasked with fixing the broken code and ensuring the application works seamlessly.

Task

Refactor the existing codebase and ensure the proper functioning of the Note Taking Application. Document all identified bugs during the debugging process. Remember, the task is not about recreating the app from scratch. Your goal is to fix the already existing codebase and make the application work as intended.

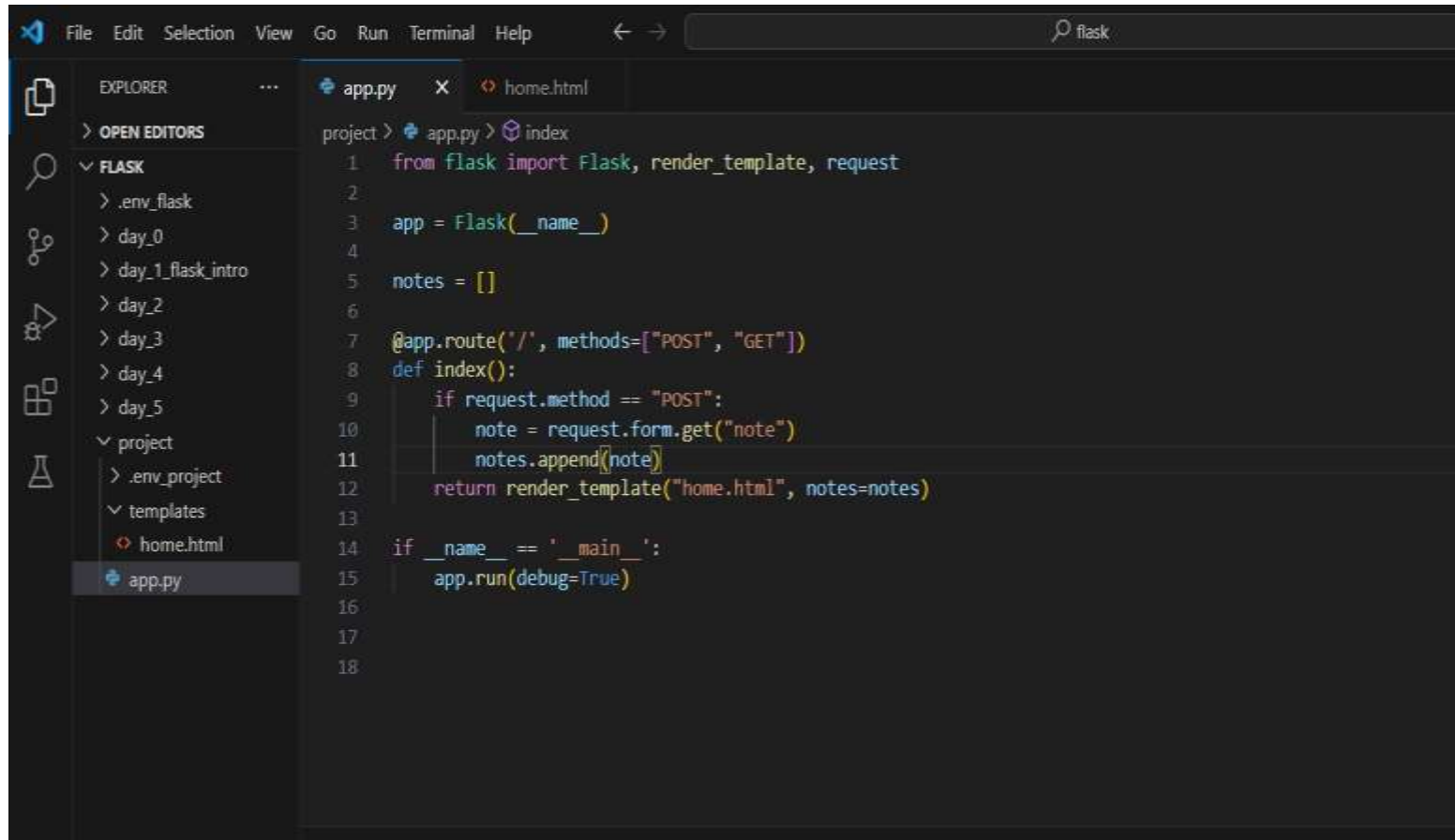
Identifying the Bugs in the Code given:

```
1  from flask import Flask, render_template, request
2
3  app = Flask(__name__)
4
5  notes = []
6  @app.route('/', methods=["POST"])
7  def index():
8      note = request.args.get("note")
9      notes.append(note)
10     return render_template("home.html", notes=notes)
11
12
13 if __name__ == '__main__':
14     app.run(debug=True)
```

```
1  <!DOCTYPE html>
2  <html lang="en">
3  <head>
4      <meta charset="UTF-8">
5      <meta http-equiv="X-UA-Compatible" content="IE=edge">
6      <meta name="viewport" content="width=device-width, initial-scale=1.0">
7      <title>Document</title>
8  </head>
9  <body>
10     <form action="">
11         <input type="text" name="note" placeholder="Enter a note">
12         <button>Add Note</button>
13     </form>
14
15     <ul>
16         {% for note in notes%}
17         <li>{{ note }}</li>
18         {% endfor %}
19     </ul>
20 </body>
21 </html>
```

The application codes containing bugs, therefore to make the application run we need to make changes by identifying.

Identifying and correcting Bugs in python application program

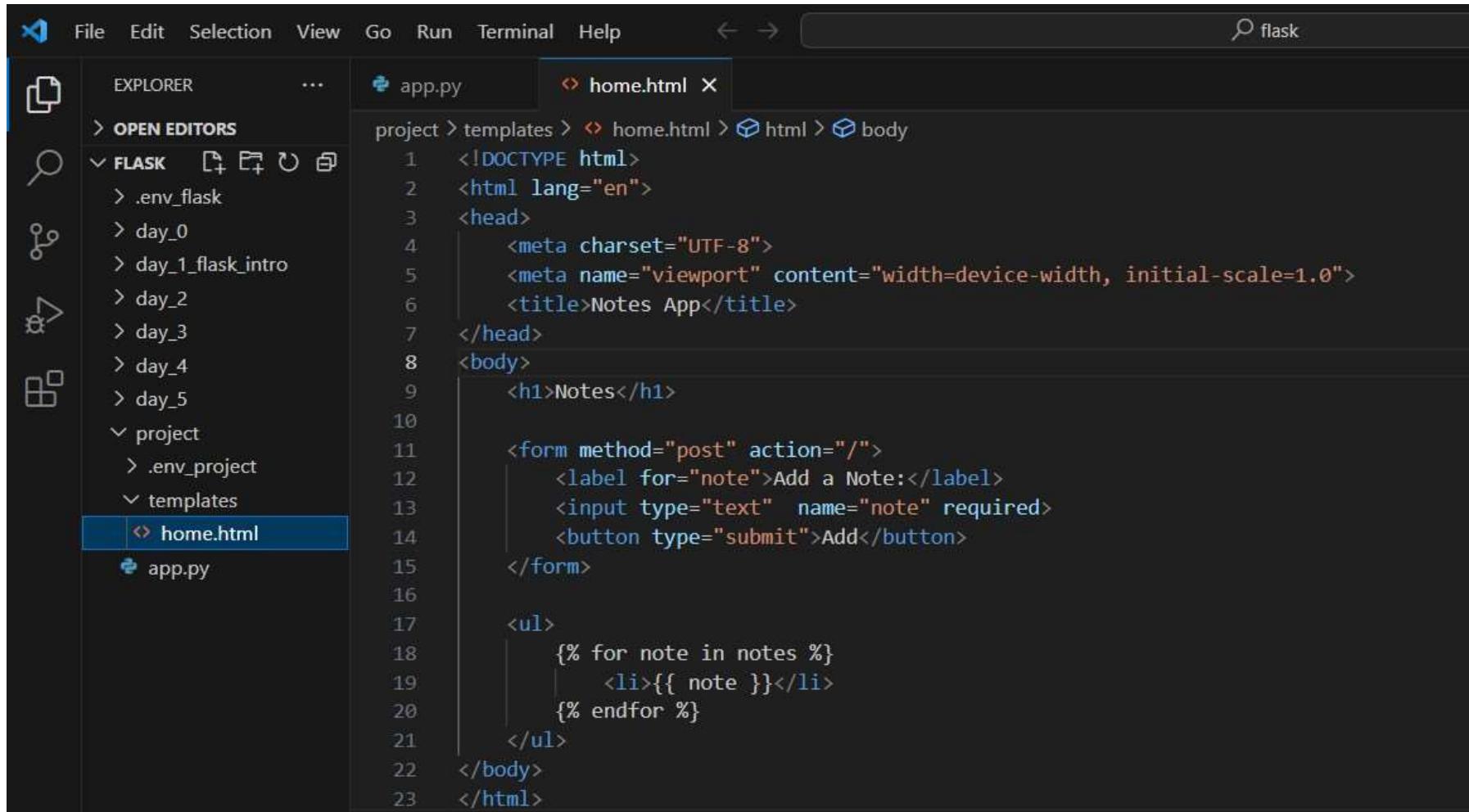


The image shows a screenshot of the Visual Studio Code editor interface. The Explorer panel on the left shows a project structure with folders like 'FLASK', 'project', and 'templates', and files like 'app.py' and 'home.html'. The main editor window displays the code for 'app.py'. The code is a Flask application that imports Flask, render_template, and request. It creates an app instance, sets a route for '/', and defines an index function that handles POST and GET requests. The index function appends the note from the request form to a list of notes and renders the 'home.html' template. The app is run in debug mode.

```
project > app.py > index
1  from flask import Flask, render_template, request
2
3  app = Flask(__name__)
4
5  notes = []
6
7  @app.route('/', methods=["POST", "GET"])
8  def index():
9      if request.method == "POST":
10         note = request.form.get("note")
11         notes.append(note)
12         return render_template("home.html", notes=notes)
13
14  if __name__ == '__main__':
15     app.run(debug=True)
16
17
18
```

This is the code snippet after identifying bugs in python application program.

Identifying and correcting the Bugs in the HTML Code :



```
1 <!DOCTYPE html>
2 <html lang="en">
3 <head>
4   <meta charset="UTF-8">
5   <meta name="viewport" content="width=device-width, initial-scale=1.0">
6   <title>Notes App</title>
7 </head>
8 <body>
9   <h1>Notes</h1>
10
11   <form method="post" action="/">
12     <label for="note">Add a Note:</label>
13     <input type="text" name="note" required>
14     <button type="submit">Add</button>
15   </form>
16
17   <ul>
18     {% for note in notes %}
19     <li>{{ note }}</li>
20     {% endfor %}
21   </ul>
22 </body>
23 </html>
```

This is the HTML code after identifying & correcting the bugs.

The output of the code:

← ↻ ⓘ 127.0.0.1:5000

Notes Application

Add a Note:

- rakesh
- hai
- hello
- how r u

THANK
YOU

