

INNOMATICS[®]
RESEARCH LABS

INNOVATION. AUTOMATION. ANALYTICS

PROJECT ON

CODE REFACTORING AND BUG FIXING

About me

I'm Ananthakrishnan KS, a passionate Data Science Student. With expertise in data analysis, machine learning, 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.

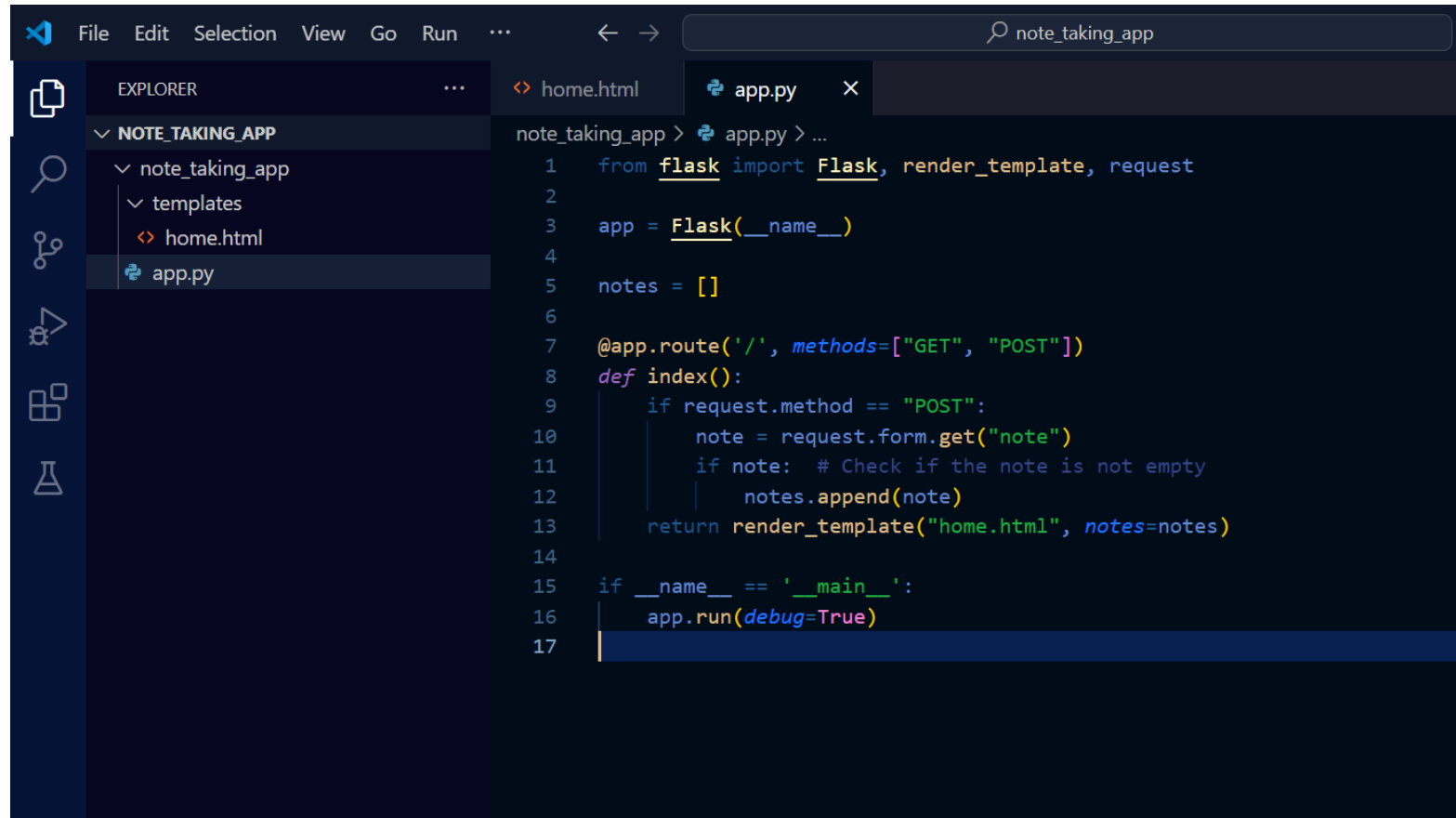
Identifying the Bugs in the Code given:

```
<> home.html • app.py
note_taking_app > templates > <> home.html > ...
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>
```

```
<> home.html • app.py •
note_taking_app > app.py > ...
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)
```

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

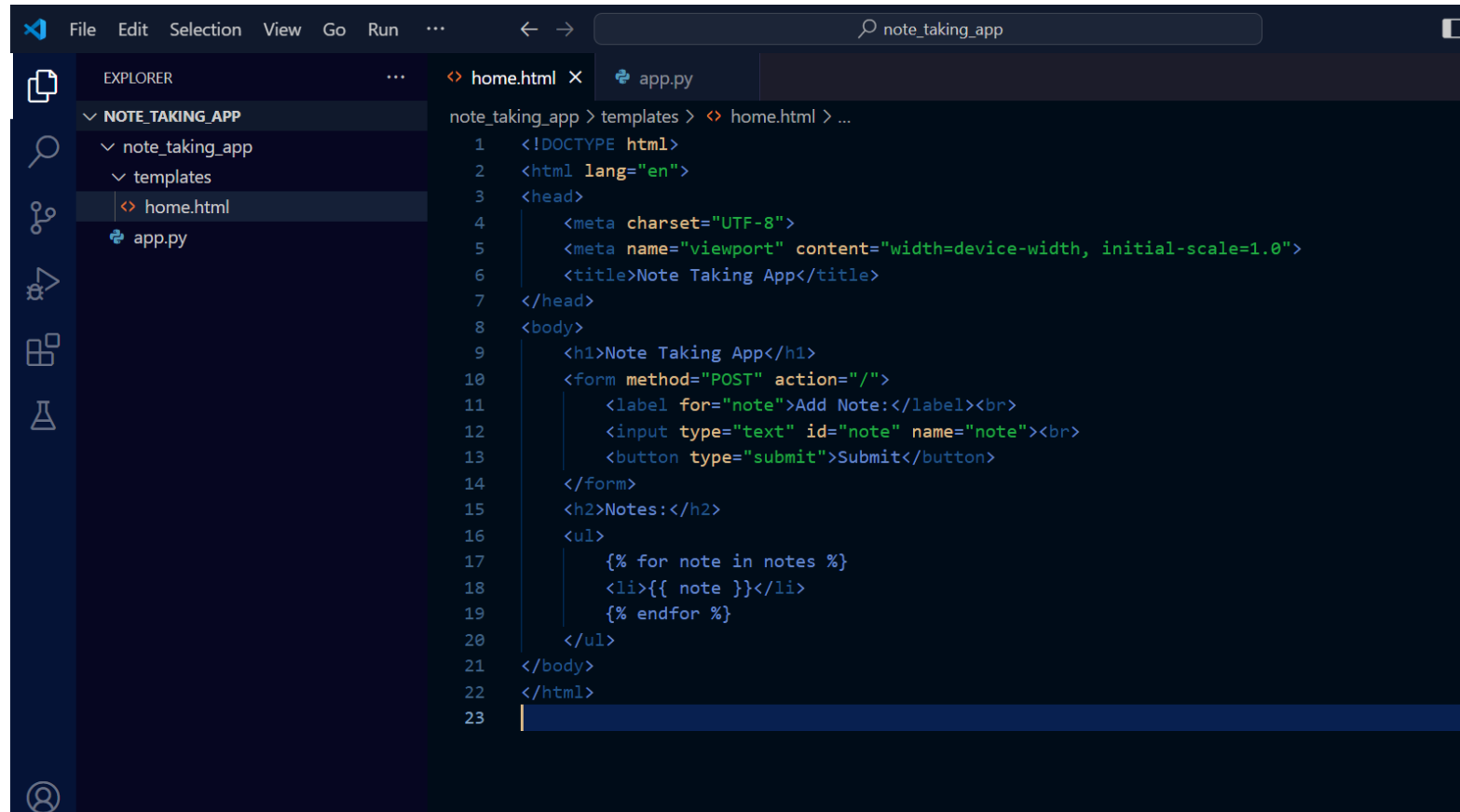


The screenshot shows a code editor with a dark theme. The Explorer panel on the left shows a project named 'NOTE_TAKING_APP' with subfolders 'note_taking_app' and 'templates'. The 'note_taking_app' folder contains 'home.html' and 'app.py'. The 'app.py' file is selected and its code is displayed in the main editor area. The code is a Flask application that handles GET and POST requests to a single route ('/'). It maintains a list of notes and appends new notes to it. The application is run in debug mode.

```
1 from flask import Flask, render_template, request
2
3 app = Flask(__name__)
4
5 notes = []
6
7 @app.route('/', methods=["GET", "POST"])
8 def index():
9     if request.method == "POST":
10         note = request.form.get("note")
11         if note: # Check if the note is not empty
12             notes.append(note)
13     return render_template("home.html", notes=notes)
14
15 if __name__ == '__main__':
16     app.run(debug=True)
17
```

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

Identifying and correcting Bugs in the HTML Code:

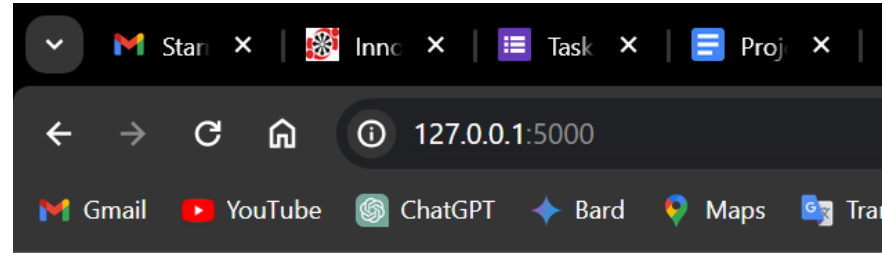


The screenshot shows a code editor with a dark theme. The Explorer panel on the left shows a project named 'NOTE_TAKING_APP' with a subdirectory 'note_taking_app' containing a 'templates' folder. Inside 'templates', the file 'home.html' is selected. The main editor area displays the content of 'home.html'. The code is as follows:

```
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>Note Taking App</title>
7 </head>
8 <body>
9   <h1>Note Taking App</h1>
10  <form method="POST" action="/">
11    <label for="note">Add Note:</label><br>
12    <input type="text" id="note" name="note"><br>
13    <button type="submit">Submit</button>
14  </form>
15  <h2>Notes:</h2>
16  <ul>
17    {% for note in notes %}
18    <li>{{ note }}</li>
19    {% endfor %}
20  </ul>
21 </body>
22 </html>
23
```

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

Developing Output



Note Taking App

Add Note:

Submit

Notes:

- Innomatics
- Intership
- March

THANK
YOU

