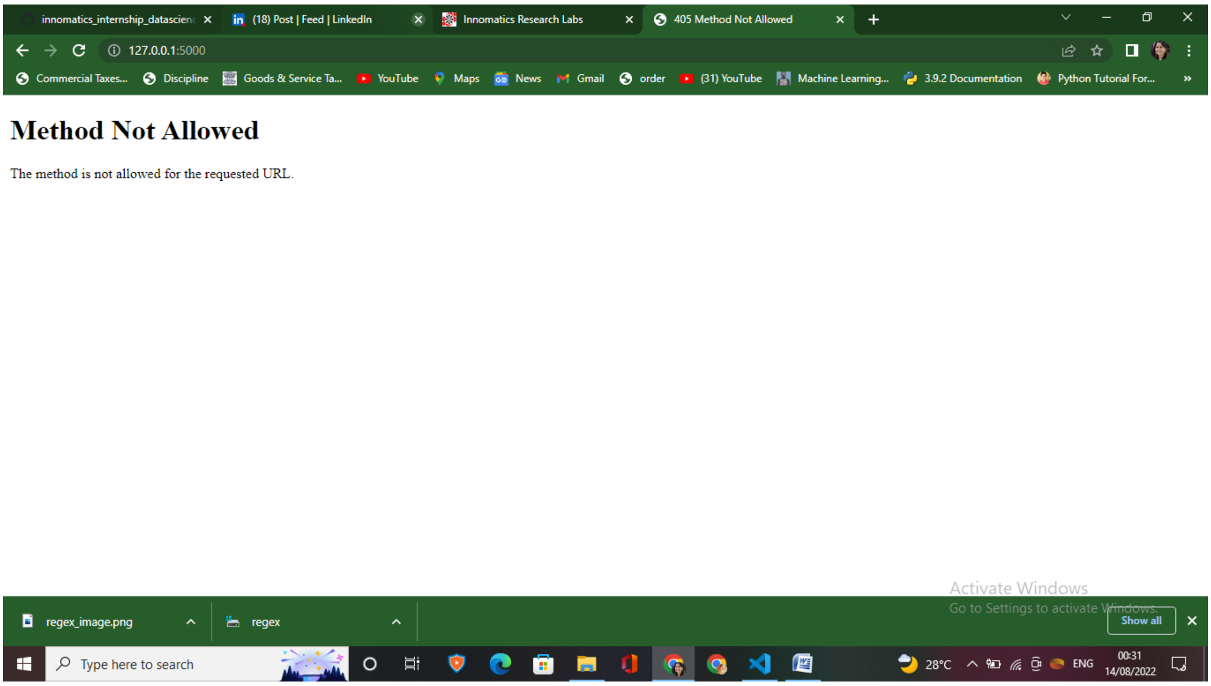
NOTE TAKING APPLICATION

We have a note taking application with only one route (i.e. Home Route). This application is broken.

We need to refactor the code and try to make sure the application works properly.

AFTER THE INITIAL RUN



Method Not Allowed

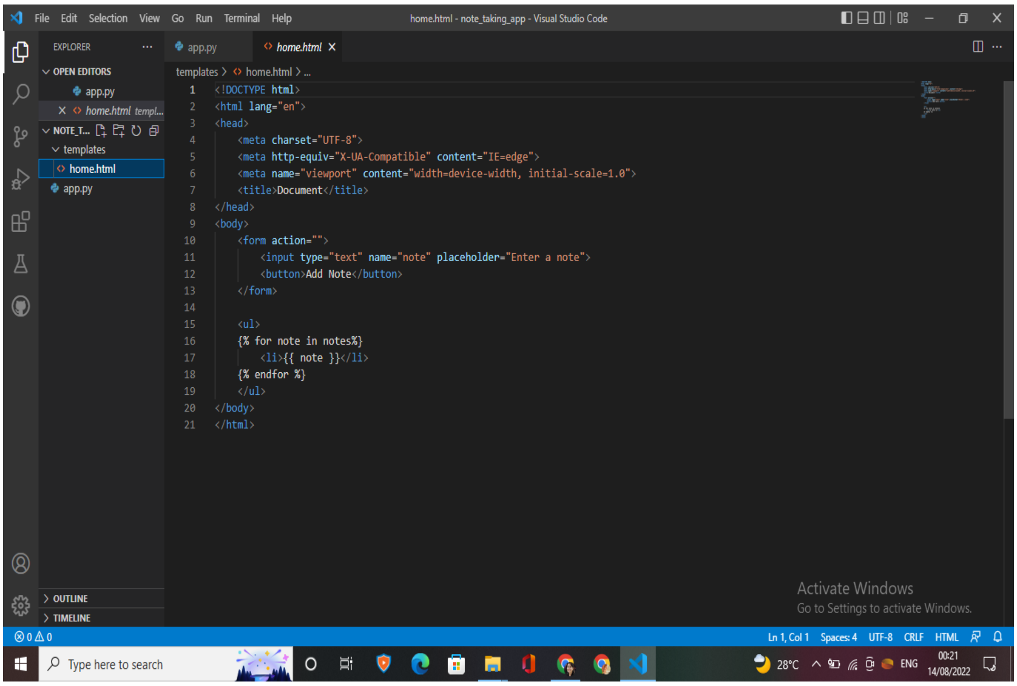
The method is not allowed for the requested URL.

Output displayed before debugging the code

CHANGES DONE IN BACKEND i.e app.py

- As we get the error ,method not allowed in the above image
- We get to know that the method passed in the route is “POST” and the request to fetch the data from frontend is called as ‘args’ which is used for “GET” , So I changed it to ‘.form’ which is used for “POST” method.
- I added both “GET” and “POST” request to my route as i wanted that the home page to be of GET request and my entered text should be secured as POST request.

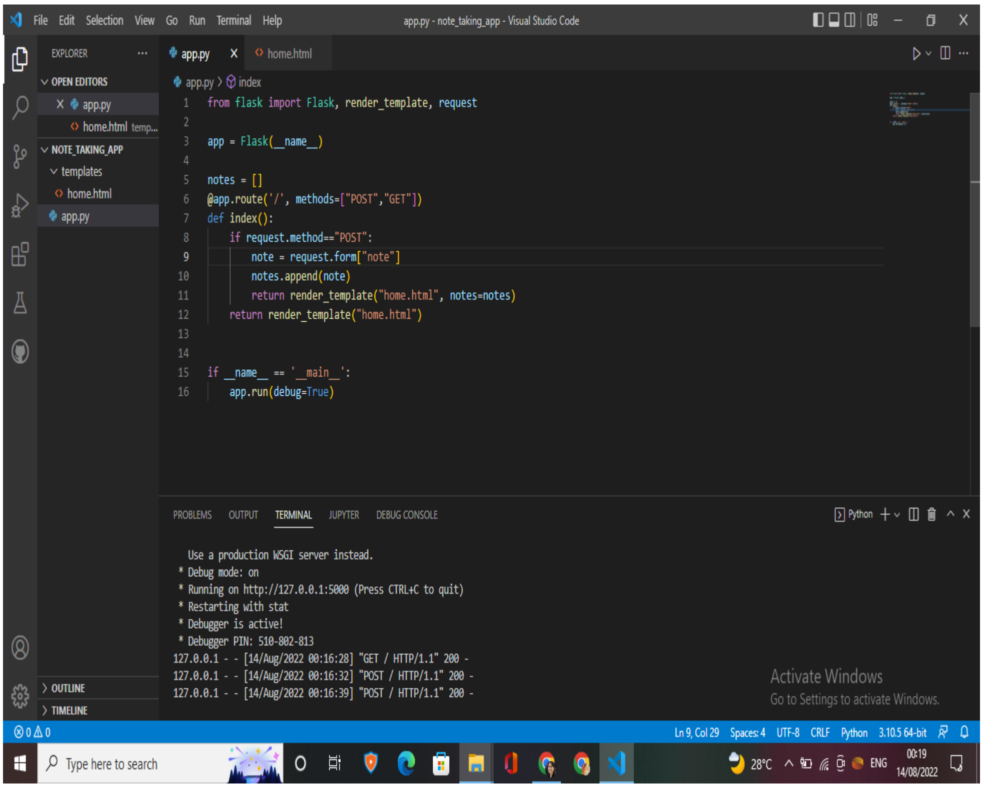
- Now there should be two return statements i.e one for GET method and another one for POST method. GET method return .html page and POST method return entered text as output in a variable to the frontend along with .html page.



File Edit Selection View Go Run Terminal Help
home.html - note_taking_app - Visual Studio Code
EXPLORER
OPEN EDITORS
app.py
home.html templ...
NOTE_T...
templates
home.html
app.py
OUTLINE
TIMELINE
app.py
home.html
templates > home.html > ...
Activate Windows
Go to Settings to activate Windows.
Ln 1, Col 1 Spaces: 4 UTF-8 CRLF HTML
Type here to search
28°C ENG 00:21 14/08/2022

```html
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta http-equiv="X-UA-Compatible" content="IE=edge">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Document</title>
</head>
<body>
    <form action="">
        <input type="text" name="note" placeholder="Enter a note">
        <button>Add Note</button>
    </form>

    <ul>
    {% for note in notes%}
        <li>{{ note }}</li>
    {% endfor %}
    </ul>
</body>
</html>
```

BEFORE DEBUGGING, CODE LOOKS LIKE THIS



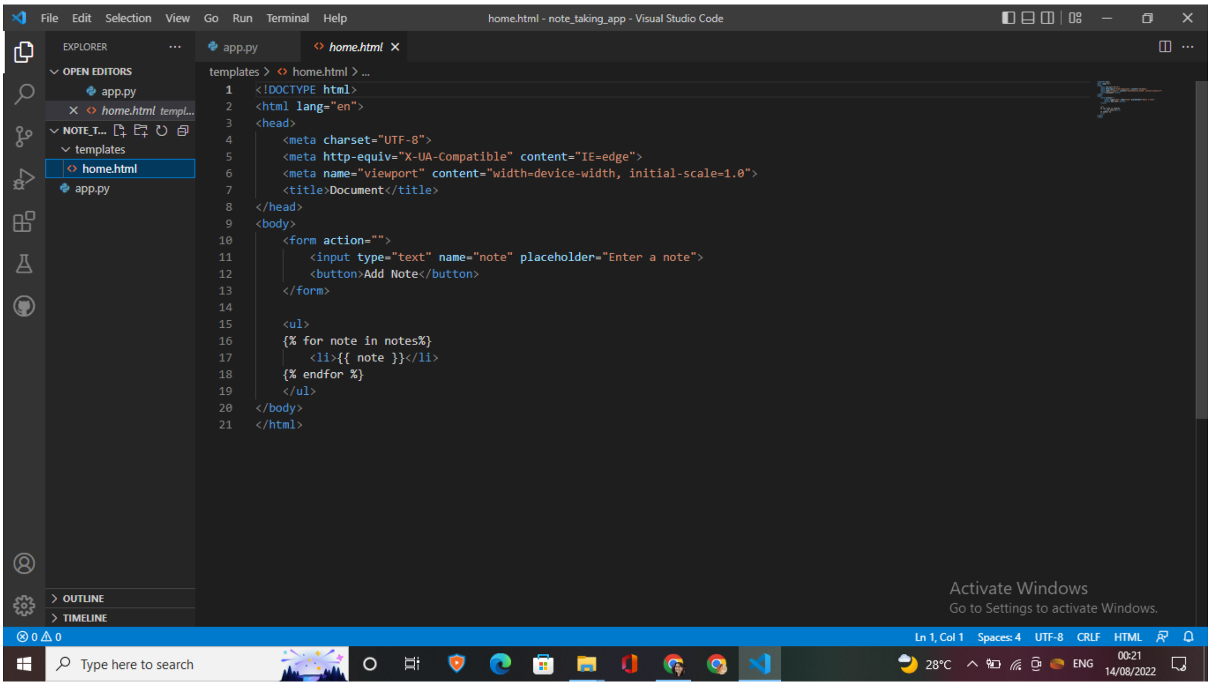
```python
from flask import Flask, render_template, request

app = Flask(__name__)

notes = []
@app.route('/', methods=["POST","GET"])
def index():
    if request.method=="POST":
        note = request.form["note"]
        notes.append(note)
        return render_template("home.html", notes=notes)
    return render_template("home.html")


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

```
   Use a production WSGI server instead.
 * Debug mode: on
 * Running on http://127.0.0.1:5000 (Press CTRL+C to quit)
 * Restarting with stat
 * Debugger is active!
 * Debugger PIN: 510-802-813
127.0.0.1 - - [14/Aug/2022 00:16:28] "GET / HTTP/1.1" 200 -
127.0.0.1 - - [14/Aug/2022 00:16:32] "POST / HTTP/1.1" 200 -
127.0.0.1 - - [14/Aug/2022 00:16:39] "POST / HTTP/1.1" 200 -
```

AFTER DEBUGGING, CODE LOOKS LIKE THIS

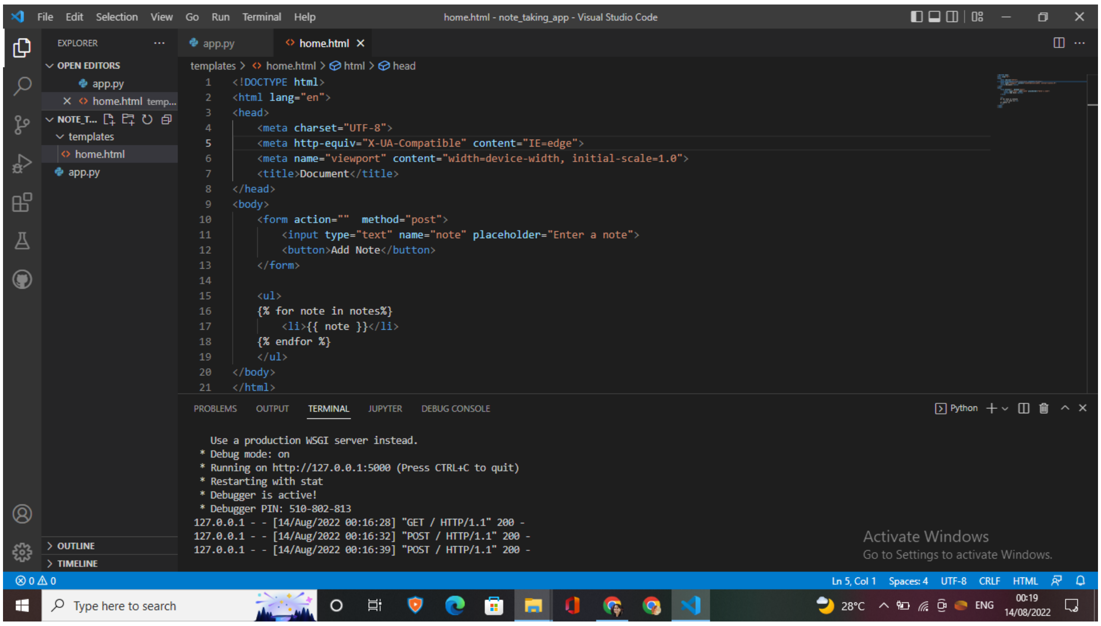
<u>CHANGES DONE IN FRONTEND i.e home.html</u>

- Only change done in the frontend was that we added the method type as “POST” next to action as we wanted the output to get displayed in post request after clicking the button.



```html
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta http-equiv="X-UA-Compatible" content="IE=edge">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Document</title>
</head>
<body>
    <form action="">
        <input type="text" name="note" placeholder="Enter a note">
        <button>Add Note</button>
    </form>

    <ul>
    {% for note in notes%}
        <li>{{ note }}</li>
    {% endfor %}
    </ul>
</body>
</html>
```

BEFORE DEBUGGING

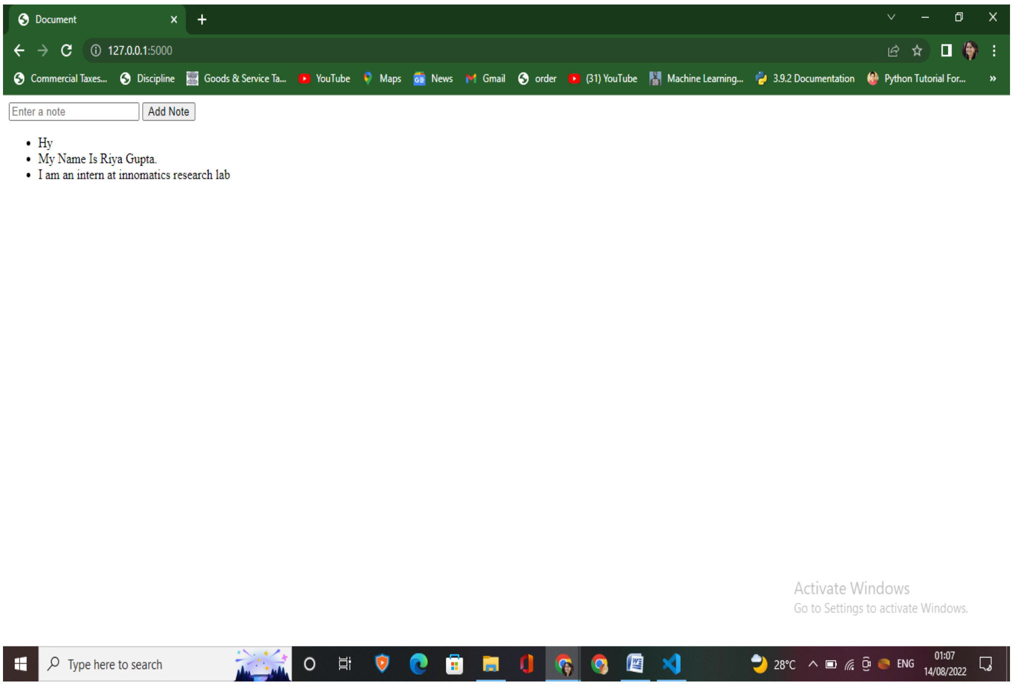


```html
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta http-equiv="X-UA-Compatible" content="IE=edge">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Document</title>
</head>
<body>
    <form action=""  method="post">
        <input type="text" name="note" placeholder="Enter a note">
        <button>Add Note</button>
    </form>

    <ul>
    {% for note in notes%}
        <li>{{ note }}</li>
    {% endfor %}
    </ul>
</body>
</html>
```

```
   Use a production WSGI server instead.
 * Debug mode: on
 * Running on http://127.0.0.1:5000 (Press CTRL+C to quit)
 * Restarting with stat
 * Debugger is active!
 * Debugger PIN: 510-802-813
127.0.0.1 - - [14/Aug/2022 00:16:28] "GET / HTTP/1.1" 200 -
127.0.0.1 - - [14/Aug/2022 00:16:32] "POST / HTTP/1.1" 200 -
127.0.0.1 - - [14/Aug/2022 00:16:39] "POST / HTTP/1.1" 200 -
```

AFTER DEBUGGING

IN THE END APPLICATION RUNS SUCCESSFULLY.HENCE THE OUTPUT DISPLAYED AFTER DEBUGGING THE CODE:



- Hy
- My Name Is Riya Gupta.
- I am an intern at innomatics research lab