

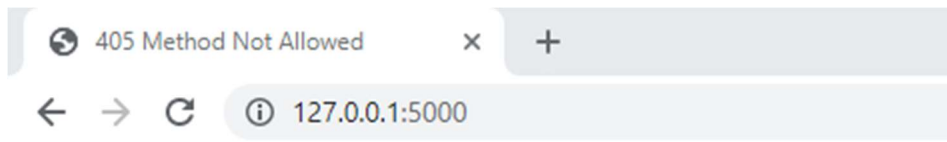
## **DEBUGGING THE NOTE TAKING APPLICATION.**

**NAME : ALMAS BANU**

### **Note Taking App:**

Home route contains a 'text field' and a 'button'. User can add a note and all the notes will be displayed as an unordered list below the text field on the same page.

The given note taking application is giving an error **Method Not Allowed.**



## **Method Not Allowed**

The method is not allowed for the requested URL.

❖ Code given is like this –

```
app.py x
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)
```

PROBLEMS OUTPUT TERMINAL JUPYTER DEBUG CONSOLE Python +

Copyright (C) Microsoft Corporation. All rights reserved.

Try the new cross-platform PowerShell <https://aka.ms/pscore6>

PS C:\Users\Almas Banu\Desktop\internship\_flask\_sessions> & D:/Anaconda/python.exe "c:/Users/Almas Banu/Desktop/internship\_flask\_sessions/note\_taking\_app/app.py"

\* 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  
\* Restarting with windowsapi reloader  
\* Debugger is active!  
\* Debugger PIN: 314-130-766  
\* Running on http://127.0.0.1:5000/ (Press CTRL+C to quit)  
127.0.0.1 - - [13/Aug/2022 20:52:59] "GET / HTTP/1.1" 405 -

- In `app.route` the method specified is POST and while taking input we are using `GET` method since `request.args.get()`.
- Lets divide the work over here, First we will display the html page by default route option GET and while assigning input to notes variable we will use POST method.
- POST method is more secure than GET.
- If we are using POST method then while assigning input we have to use `request.form[]`.
- We have to specify both methods in `@app.route` as `methods=['GET', 'POST']`
- Removing the None method.

❖ After Making the changes

```
app.py x
note_taking_app_new > app.py > ...
1  from flask import Flask, render_template, request, session
2  from flask_session import Session
3
4  app = Flask(__name__)
5  app.config["SESSION_PERMANENT"] = False
6  app.config["SESSION_TYPE"] = "filesystem"
7  Session(app)
8
9  notes = []
10 @app.route('/', methods=["GET", "POST"])
11 def index():
12     if session.get("notes") is None:
13         session["notes"] = []
14     if request.method == "POST":
15         note = request.form["note"]
16         if note!="":
17             session["notes"].append(note)
18     return render_template("home.html", notes=session["notes"])
19
20
21 if __name__ == '__main__':
22     app.run(debug=True)
```

❖ Before :

home.html

```

home.html X
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>
  
```

❖ After:

home.html

```

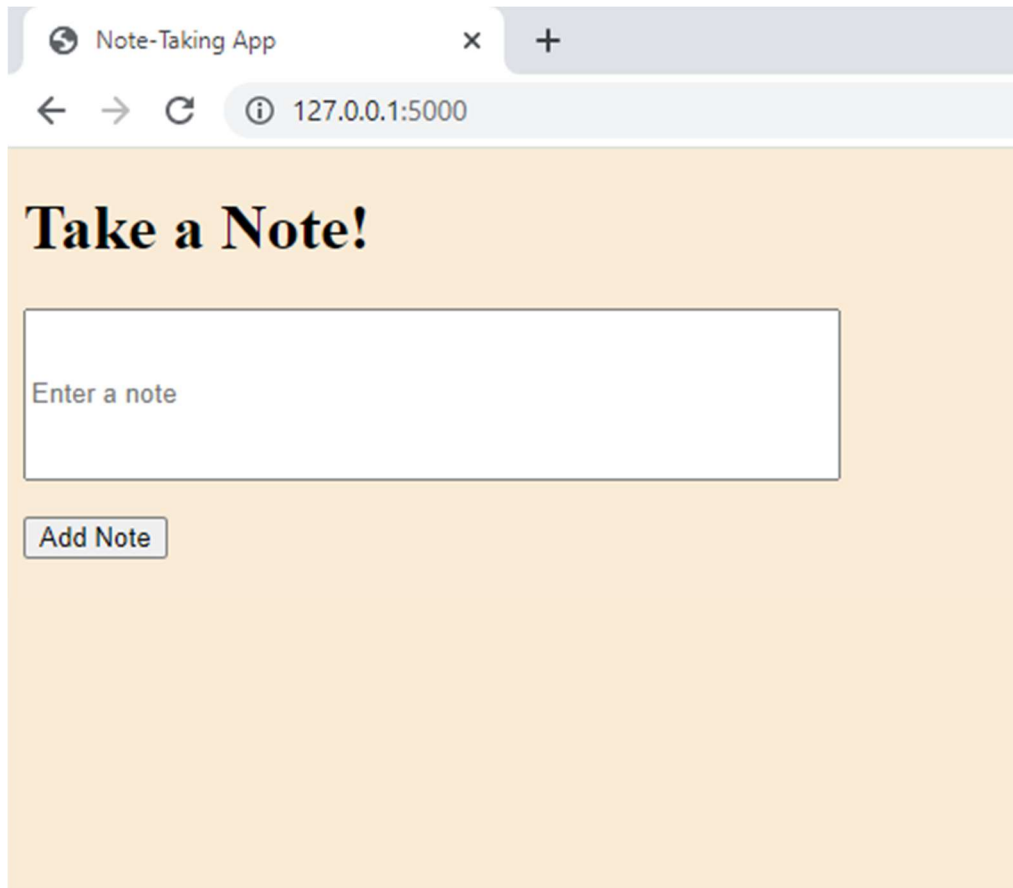
app.py  home.html X
note_taking_app_new > templates > home.html > ...
1  <!DOCTYPE html>
2  <html lang="en">
3  <head>
4      <meta charset="UTF-8">
5      <meta http-equiv="X-UA-Compatible" content="IE=edge">
6      <meta name="viewport" content="width=device-width, initial-scale=1.0">
7      <title>Note-Taking App</title>
8  </head>
9  <body style="background-color: antiquewhite">
10     <h1>Take a Note!</h1>
11     <form action="" method="post">
12         <input type="text" name="note" placeholder="Enter a note" style="height:80px;width:400px">
13         <br/>
14         <br/>
15         <button>Add Note</button>
16     </form>
17     {% if notes %}
18     <ul>
19         {% for note in notes%}
20         <li>{{ note }}</li>
21         {% endfor %}
22     </ul>
23     {% endif %}
24 </body>
25 </html>
  
```

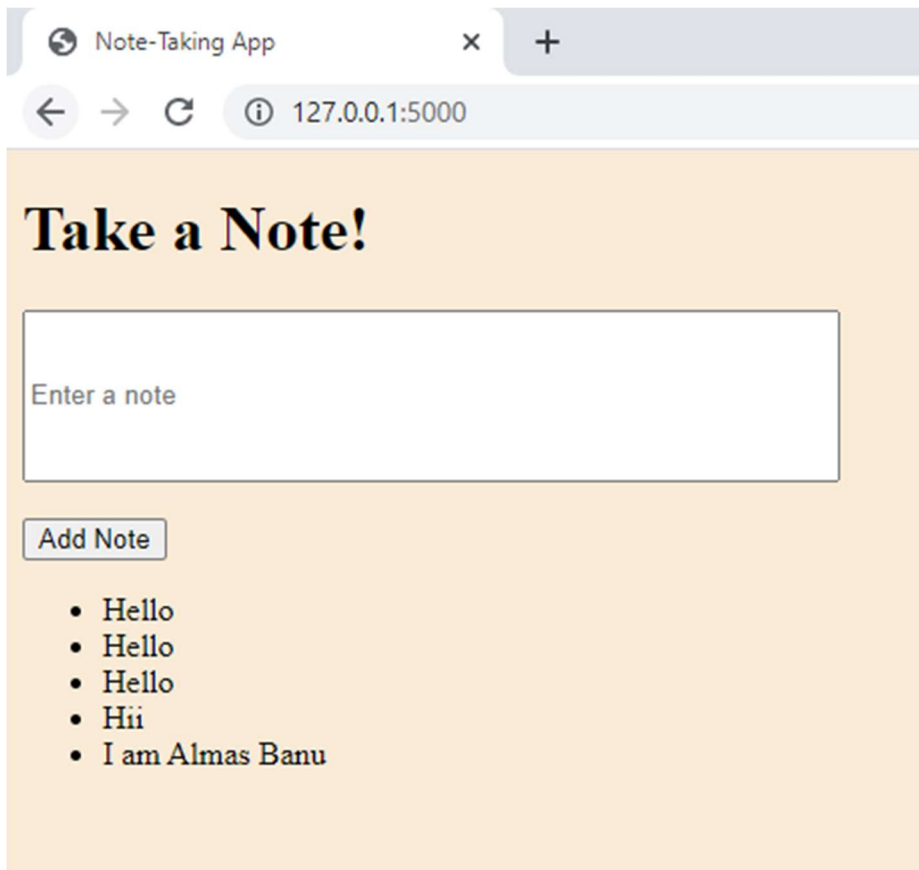
Just increased the textarea and added styles for presentation.

Since we want our inputs to be secured I have specified method="post" in form tag.

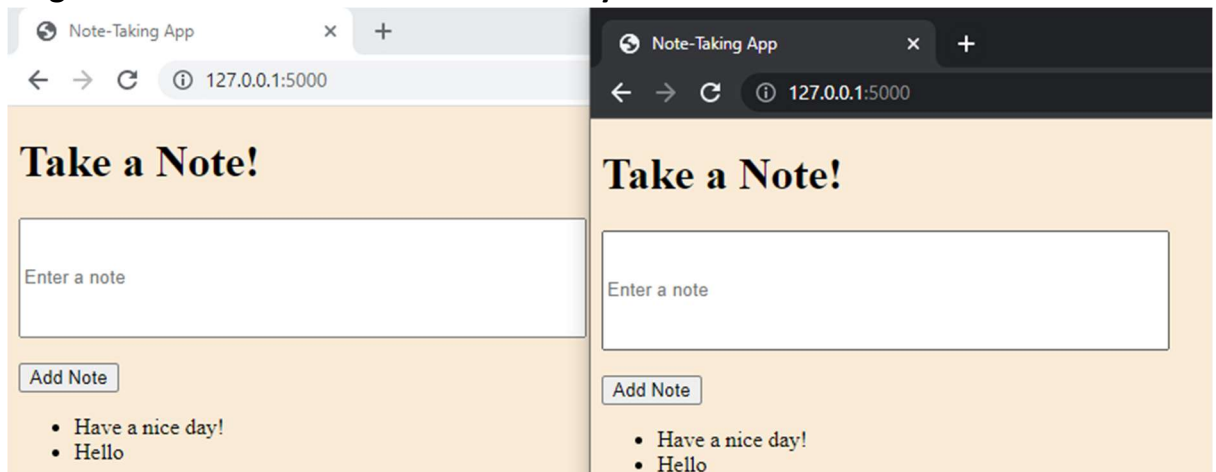
After Solving the bugs,

The Output

A screenshot of a web browser window. The browser has a single tab titled "Note-Taking App". The address bar shows the URL "127.0.0.1:5000". The webpage has a light orange background. At the top, the text "Take a Note!" is displayed in a large, bold, black serif font. Below this is a large white rectangular text input field with a thin black border. Inside the input field, the placeholder text "Enter a note" is visible in a small, gray font. Below the input field is a small, rectangular button with a light gray background and a thin black border, containing the text "Add Note" in a small, black font.



- When we try to access the application in incognito mode parallelly with the normal browser then notes added in chrome browser is also reflected in incognito mode because of same memory .



- In order to solve this we have to use Session module from flask-session to assign particular session to each user. Now, if we run it in 2 user mode then notes added for one user will not be reflected to other user.

