

Project Development phase - Sprint 1

Team ID	PNT2022TMID04422
Project Name	Personal Expense Tracker

Login & Registration Pages

Home



Sign up





Illustration of a person with glasses and an orange shirt sitting at a desk with a computer monitor, keyboard, and a small yellow cat. The monitor displays a green box with a white 'X' and some text.

127.0.0.1:5555/signup

Home



SIGN-UP

Username

E-Mail

Password

CREATE ACCOUNT

ALREADY HAVE AN ACCOUNT

Login





Illustration of a person in a purple suit and blue pants holding a tablet. To the left is a whiteboard on an easel with a diagram of a triangle. The person is also holding a small tablet with a triangle diagram.

127.0.0.1:5555/login

Home



LOGIN

Username

Password

[Forgot Password?](#)

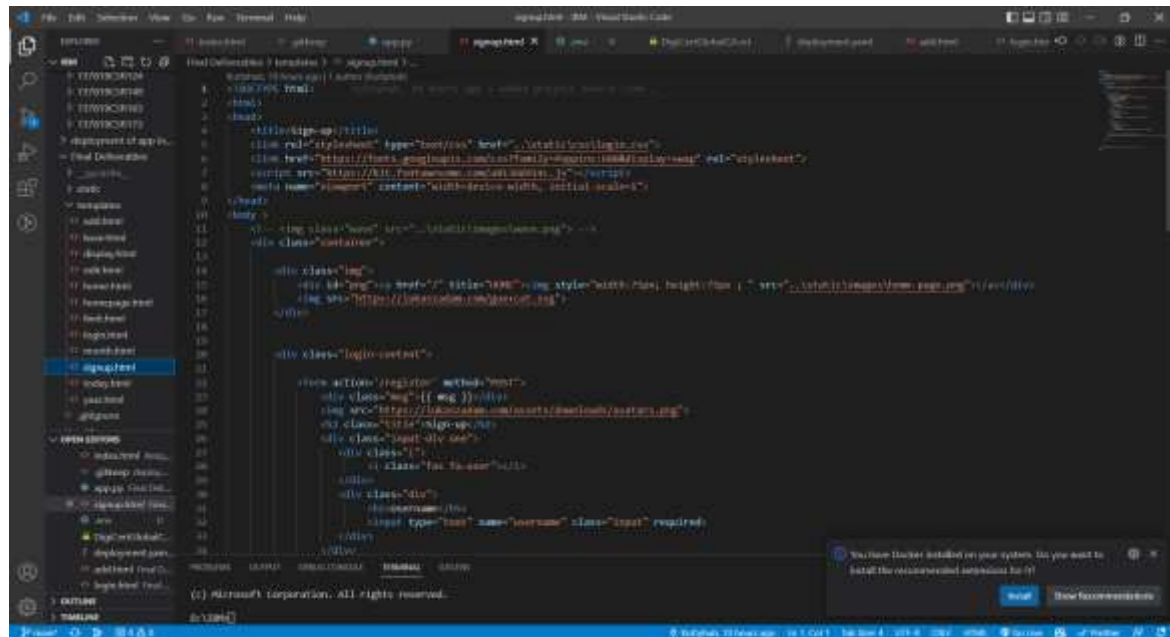
LOGIN

DON'T HAVE AN ACCOUNT

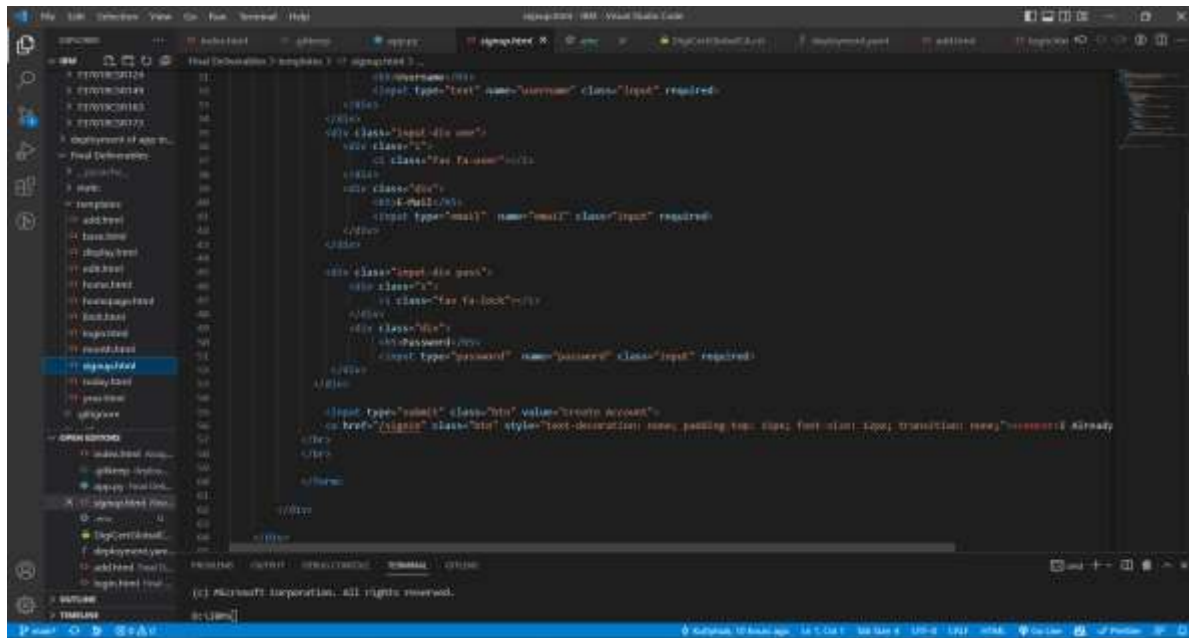
Front End Code

Sign up

Html



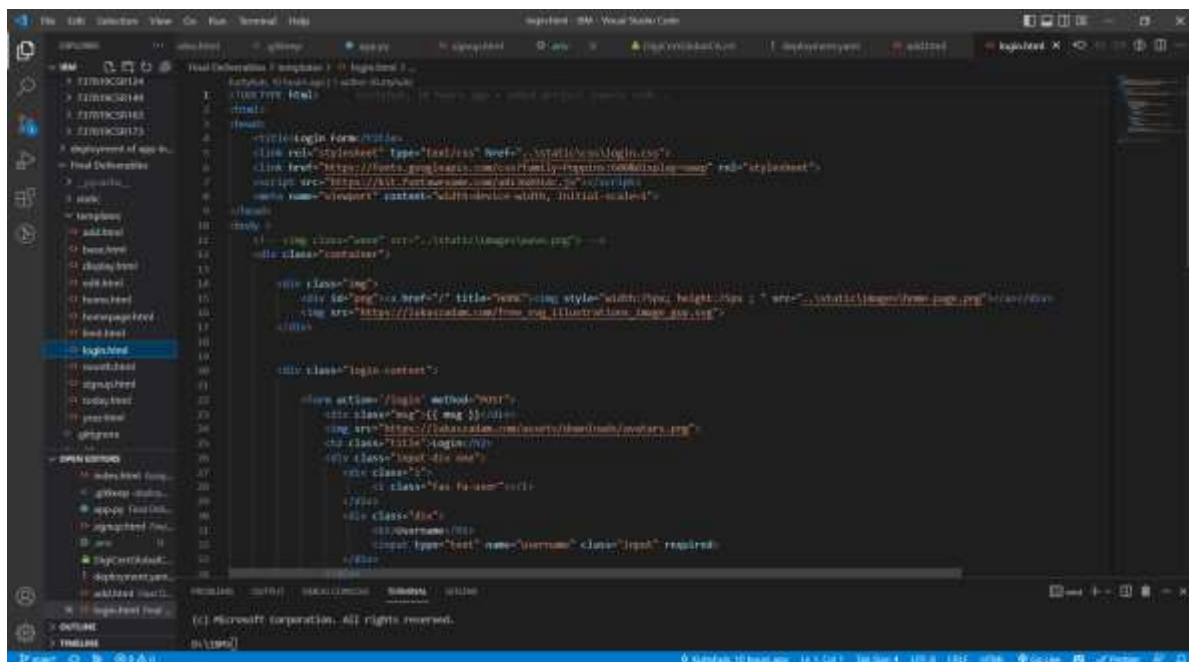
```
1 <!DOCTYPE html>
2 <html>
3 <head>
4   <title>Sign-up</title>
5   <link rel="stylesheet" href="css/sign-up.css" />
6   <link href="https://fonts.googleapis.com/css?family=Roboto:400,400i,700,700i" rel="stylesheet">
7   <script src="https://code.jquery.com/jquery-3.5.1.min.js"></script>
8   <meta name="viewport" content="width=device-width, initial-scale=1">
9 </head>
10 <body>
11   <div class="form" style="background-color: #f9f9f9; padding: 10px; border: 1px solid #ccc;">
12     <div class="img">
13       <img alt="Sign-up icon" title="Sign-up icon" style="width: 100px; height: 100px; border: 1px solid #ccc; margin: 0 auto 10px auto;" />
14     </div>
15     <div class="login-control">
16       <div class="login" style="text-align: center; margin-bottom: 10px;">
17         <a href="#">Sign up</a>
18       </div>
19       <div class="input">
20         <input type="text" value="username" class="input" required />
21       </div>
22     </div>
23   </div>
24 </body>
25 </html>
```



This screenshot shows the VS Code editor with a file explorer on the left and a code editor in the center. The file explorer shows a project structure with folders like 'index.html', 'login.html', 'register.html', and 'style.css'. The code editor displays the HTML code for 'register.html'. The code includes a form with fields for username, email, password, and confirm password, along with a 'register' button. The code is as follows:

```
<!-- register.html -->
<html>
<head>
<title>Register</title>
<meta charset="utf-8">
</head>
<body>
<div class="container">
<div class="row">
<div class="col-md-6">
<div class="card">
<div class="card-body">
<div class="text-center">
<h3>Register</h3>
</div>
<div class="form">
<div class="form-group">
<input type="text" name="username" class="input" required>
</div>
<div class="form-group">
<input type="email" name="email" class="input" required>
</div>
<div class="form-group">
<input type="password" name="password" class="input" required>
</div>
<div class="form-group">
<input type="password" name="confirm" class="input" required>
</div>
<div class="form-group">
<input type="button" value="Register" class="btn btn-primary" />
</div>
</div>
</div>
</div>
</div>
</div>
</body>
</html>
```

Login



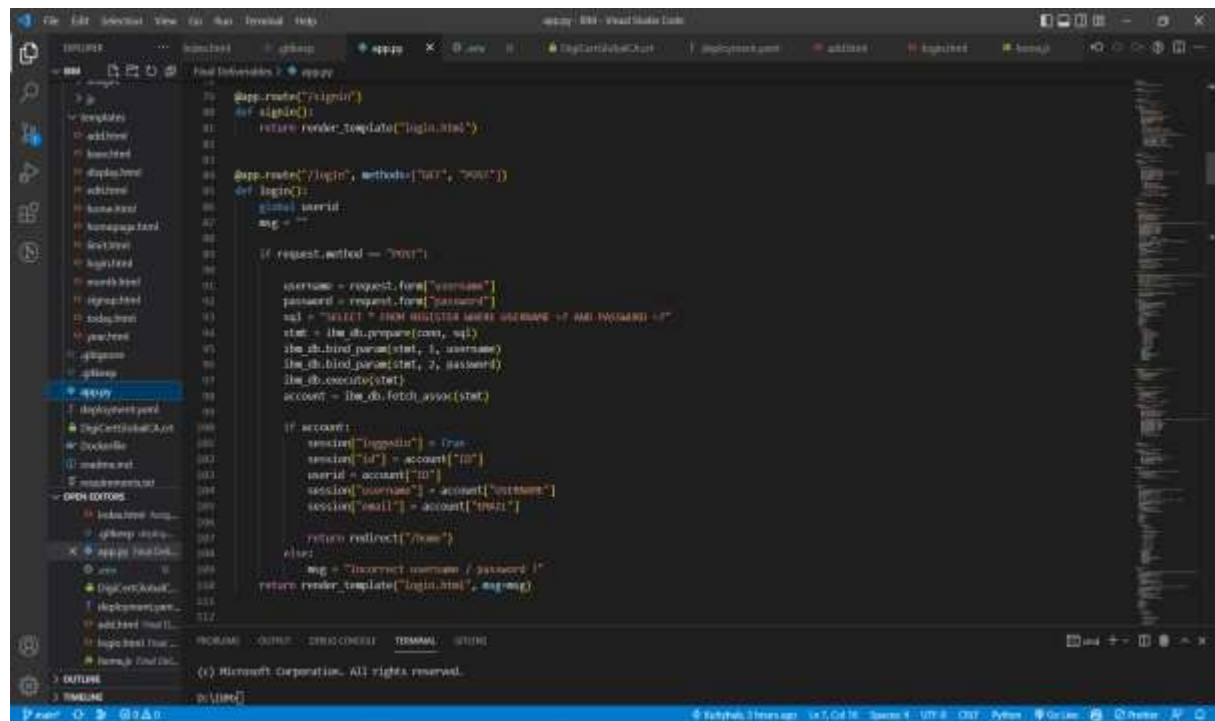
This screenshot shows the VS Code editor with a file explorer on the left and a code editor in the center. The file explorer shows a project structure with folders like 'index.html', 'login.html', 'register.html', and 'style.css'. The code editor displays the HTML code for 'login.html'. The code includes a form with fields for username and password, along with a 'login' button. The code is as follows:

```
<!-- login.html -->
<html>
<head>
<title>Login</title>
<meta charset="utf-8">
</head>
<body>
<div class="container">
<div class="row">
<div class="col-md-6">
<div class="card">
<div class="card-body">
<div class="text-center">
<h3>Login</h3>
</div>
<div class="form">
<div class="form-group">
<input type="text" name="username" class="input" required>
</div>
<div class="form-group">
<input type="password" name="password" class="input" required>
</div>
<div class="form-group">
<input type="button" value="Login" class="btn btn-primary" />
</div>
</div>
</div>
</div>
</div>
</body>
</html>
```

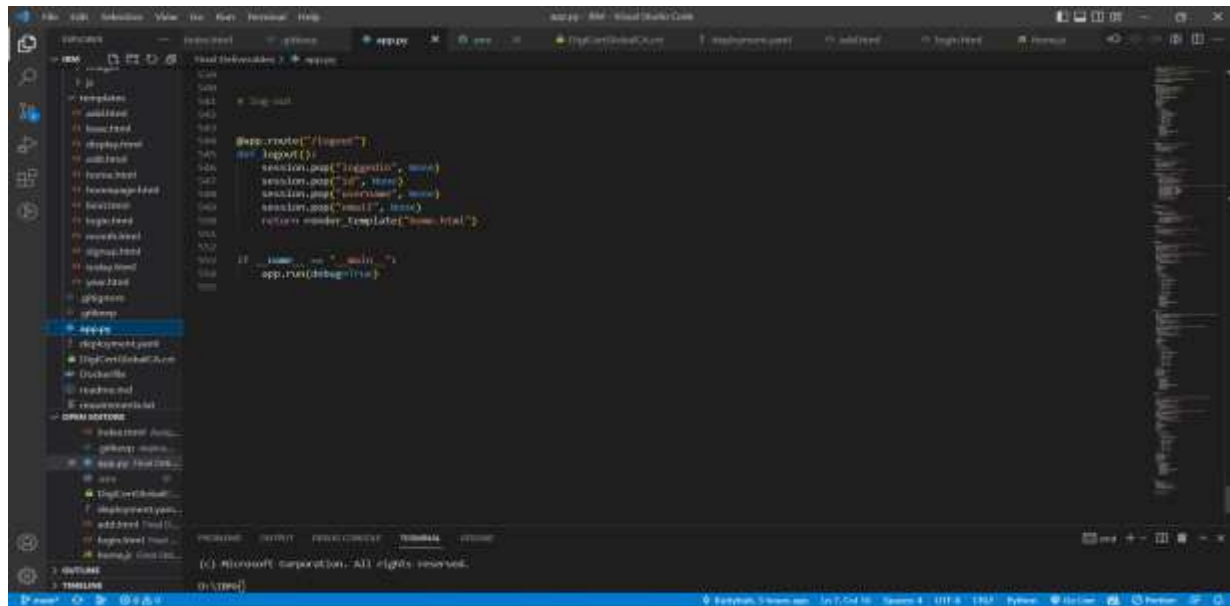
The screenshot shows a Visual Studio Code editor with a Django project. The file explorer on the left shows a project named 'app01' with a 'views.py' file. The main editor displays the code for 'views.py', which includes a 'login' function. The function uses Django's authentication system to handle a login request, including checking the request method, validating the username and password, and creating a session if the login is successful. The status bar at the bottom indicates the file is 'app01/views.py' and the editor is in 'Python' mode.

```

1  # coding: utf-8
2
3  from django.shortcuts import render
4
5  # Create your views here.
6
7  def login(request):
8      if request.method == "POST":
9          username = request.form["username"]
10         password = request.form["password"]
11
12         user = User.objects.filter(username=username).first()
13         if user is None:
14             return render(request, "login.html", {"error": "用户名不存在"})
15         if not user.check_password(password):
16             return render(request, "login.html", {"error": "密码错误"})
17
18         login_success(request, user)
19         return render(request, "index.html")
20
21 def logout(request):
22     if request.method == "POST":
23         auth.logout(request)
24         return render(request, "login.html", {"error": "退出成功"})
25     return render(request, "index.html")
26
27 def register(request):
28     if request.method == "POST":
29         username = request.form["username"]
30         password = request.form["password"]
31         email = request.form["email"]
32
33         if User.objects.filter(username=username).exists():
34             return render(request, "register.html", {"error": "用户名已存在"})
35         if User.objects.filter(email=email).exists():
36             return render(request, "register.html", {"error": "邮箱已存在"})
37
38         user = User.objects.create_user(username=username, password=password, email=email)
39         user.set_password(password)
40         user.save()
41         return render(request, "login.html", {"error": "注册成功"})
42     return render(request, "register.html")
43
44 def index(request):
45     return render(request, "index.html")
46
47 def about(request):
48     return render(request, "about.html")
49
50 def contact(request):
51     return render(request, "contact.html")
52
53 def search(request):
54     return render(request, "search.html")
55
56 def detail(request, pk):
57     return render(request, "detail.html")
58
59 def list(request):
60     return render(request, "list.html")
61
62 def create(request):
63     return render(request, "create.html")
64
65 def update(request, pk):
66     return render(request, "update.html")
67
68 def delete(request, pk):
69     return render(request, "delete.html")
70
71 def error(request):
72     return render(request, "error.html")
73
74 def test(request):
75     return render(request, "test.html")
76
77 def test2(request):
78     return render(request, "test2.html")
79
80 def test3(request):
81     return render(request, "test3.html")
82
83 def test4(request):
84     return render(request, "test4.html")
85
86 def test5(request):
87     return render(request, "test5.html")
88
89 def test6(request):
90     return render(request, "test6.html")
91
92 def test7(request):
93     return render(request, "test7.html")
94
95 def test8(request):
96     return render(request, "test8.html")
97
98 def test9(request):
99     return render(request, "test9.html")
100
101 def test10(request):
102     return render(request, "test10.html")
103
104 def test11(request):
105     return render(request, "test11.html")
106
107 def test12(request):
108     return render(request, "test12.html")
109
110 def test13(request):
111     return render(request, "test13.html")
112
113 def test14(request):
114     return render(request, "test14.html")
115
116 def test15(request):
117     return render(request, "test15.html")
118
119 def test16(request):
120     return render(request, "test16.html")
121
122 def test17(request):
123     return render(request, "test17.html")
124
125 def test18(request):
126     return render(request, "test18.html")
127
128 def test19(request):
129     return render(request, "test19.html")
130
131 def test20(request):
132     return render(request, "test20.html")
133
134 def test21(request):
135     return render(request, "test21.html")
136
137 def test22(request):
138     return render(request, "test22.html")
139
140 def test23(request):
141     return render(request, "test23.html")
142
143 def test24(request):
144     return render(request, "test24.html")
145
146 def test25(request):
147     return render(request, "test25.html")
148
149 def test26(request):
150     return render(request, "test26.html")
151
152 def test27(request):
153     return render(request, "test27.html")
154
155 def test28(request):
156     return render(request, "test28.html")
157
158 def test29(request):
159     return render(request, "test29.html")
160
161 def test30(request):
162     return render(request, "test30.html")
163
164 def test31(request):
165     return render(request, "test31.html")
166
167 def test32(request):
168     return render(request, "test32.html")
169
170 def test33(request):
171     return render(request, "test33.html")
172
173 def test34(request):
174     return render(request, "test34.html")
175
176 def test35(request):
177     return render(request, "test35.html")
178
179 def test36(request):
180     return render(request, "test36.html")
181
182 def test37(request):
183     return render(request, "test37.html")
184
185 def test38(request):
186     return render(request, "test38.html")
187
188 def test39(request):
189     return render(request, "test39.html")
190
191 def test40(request):
192     return render(request, "test40.html")
193
194 def test41(request):
195     return render(request, "test41.html")
196
197 def test42(request):
198     return render(request, "test42.html")
199
200 def test43(request):
201     return render(request, "test43.html")
202
203 def test44(request):
204     return render(request, "test44.html")
205
206 def test45(request):
207     return render(request, "test45.html")
208
209 def test46(request):
210     return render(request, "test46.html")
211
212 def test47(request):
213     return render(request, "test47.html")
214
215 def test48(request):
216     return render(request, "test48.html")
217
218 def test49(request):
219     return render(request, "test49.html")
220
221 def test50(request):
222     return render(request, "test50.html")
223
224 def test51(request):
225     return render(request, "test51.html")
226
227 def test52(request):
228     return render(request, "test52.html")
229
230 def test53(request):
231     return render(request, "test53.html")
232
233 def test54(request):
234     return render(request, "test54.html")
235
236 def test55(request):
237     return render(request, "test55.html")
238
239 def test56(request):
240     return render(request, "test56.html")
241
242 def test57(request):
243     return render(request, "test57.html")
244
245 def test58(request):
246     return render(request, "test58.html")
247
248 def test59(request):
249     return render(request, "test59.html")
250
251 def test60(request):
252     return render(request, "test60.html")
253
254 def test61(request):
255     return render(request, "test61.html")
256
257 def test62(request):
258     return render(request, "test62.html")
259
260 def test63(request):
261     return render(request, "test63.html")
262
263 def test64(request):
264     return render(request, "test64.html")
265
266 def test65(request):
267     return render(request, "test65.html")
268
269 def test66(request):
270     return render(request, "test66.html")
271
272 def test67(request):
273     return render(request, "test67.html")
274
275 def test68(request):
276     return render(request, "test68.html")
277
278 def test69(request):
279     return render(request, "test69.html")
280
281 def test70(request):
282     return render(request, "test70.html")
283
284 def test71(request):
285     return render(request, "test71.html")
286
287 def test72(request):
288     return render(request, "test72.html")
289
290 def test73(request):
291     return render(request, "test73.html")
292
293 def test74(request):
294     return render(request, "test74.html")
295
296 def test75(request):
297     return render(request, "test75.html")
298
299 def test76(request):
300     return render(request, "test76.html")
301
302 def test77(request):
303     return render(request, "test77.html")
304
305 def test78(request):
306     return render(request, "test78.html")
307
308 def test79(request):
309     return render(request, "test79.html")
310
311 def test80(request):
312     return render(request, "test80.html")
313
314 def test81(request):
315     return render(request, "test81.html")
316
317 def test82(request):
318     return render(request, "test82.html")
319
320 def test83(request):
321     return render(request, "test83.html")
322
323 def test84(request):
324     return render(request, "test84.html")
325
326 def test85(request):
327     return render(request, "test85.html")
328
329 def test86(request):
330     return render(request, "test86.html")
331
332 def test87(request):
333     return render(request, "test87.html")
334
335 def test88(request):
336     return render(request, "test88.html")
337
338 def test89(request):
339     return render(request, "test89.html")
340
341 def test90(request):
342     return render(request, "test90.html")
343
344 def test91(request):
345     return render(request, "test91.html")
346
347 def test92(request):
348     return render(request, "test92.html")
349
350 def test93(request):
351     return render(request, "test93.html")
352
353 def test94(request):
354     return render(request, "test94.html")
355
356 def test95(request):
357     return render(request, "test95.html")
358
359 def test96(request):
360     return render(request, "test96.html")
361
362 def test97(request):
363     return render(request, "test97.html")
364
365 def test98(request):
366     return render(request, "test98.html")
367
368 def test99(request):
369     return render(request, "test99.html")
370
371 def test100(request):
372     return render(request, "test100.html")
373
374 def test101(request):
375     return render(request, "test101.html")
376
377 def test102(request):
378     return render(request, "test102.html")
379
380 def test103(request):
381     return render(request, "test103.html")
382
383 def test104(request):
384     return render(request, "test104.html")
385
386 def test105(request):
387     return render(request, "test105.html")
388
389 def test106(request):
390     return render(request, "test106.html")
391
392 def test107(request):
393     return render(request, "test107.html")
394
395 def test108(request):
396     return render(request, "test108.html")
397
398 def test109(request):
399     return render(request, "test109.html")
400
401 def test110(request):
402     return render(request, "test110.html")
403
404 def test111(request):
405     return render(request, "test111.html")
4
```



Log out



The screenshot shows the Visual Studio Code editor with a Flask application. The file explorer on the left shows a project structure with a `templates` folder and a `static` folder. The main editor displays the `app.py` file, which contains the following code:

```
from flask import Flask, session, redirect, url_for, flash, request
from flask_login import LoginManager, login_required, login_user, logout_user

app = Flask(__name__)
login_manager = LoginManager()
login_manager.init_app(app)

@app.route("/")
def index():
    return render_template("index.html")

@app.route("/login")
def login():
    session.pop("logged_in", None)
    session.pop("id", None)
    session.pop("username", None)
    session.pop("email", None)
    return render_template("login.html")

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

The bottom status bar indicates the file is `app.py` and the current line is `11`.

