

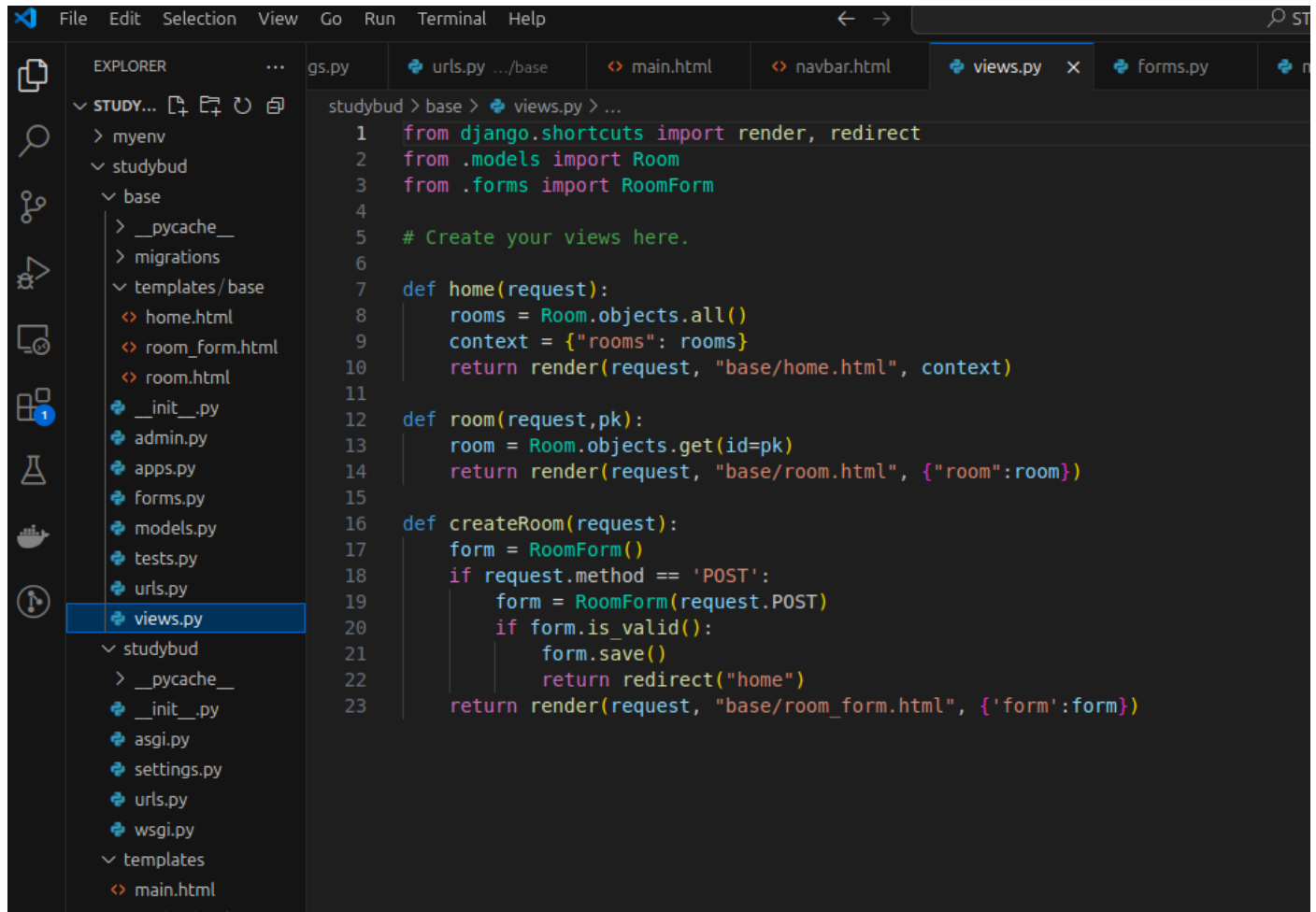
EXPLORER

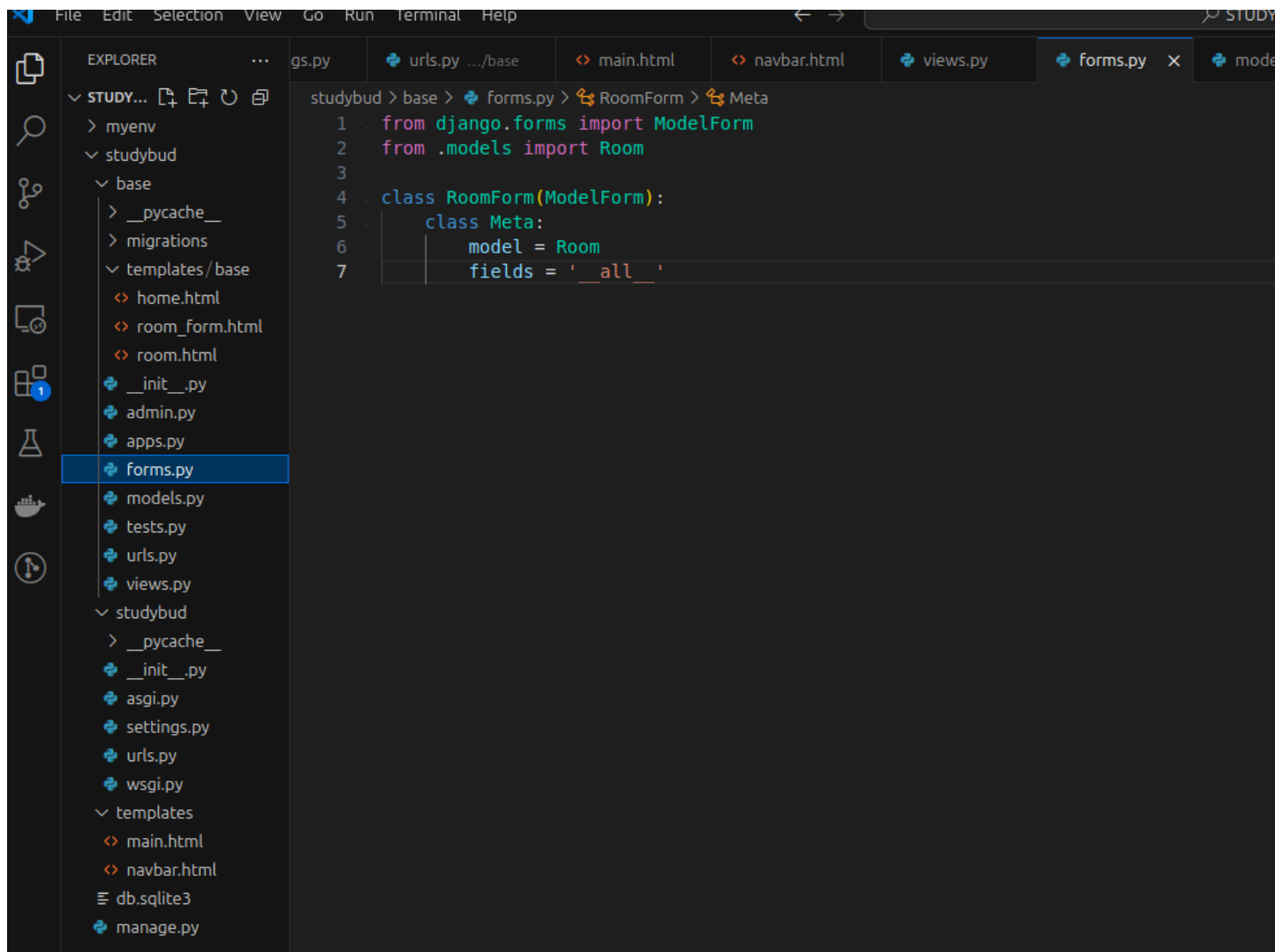
STUDY... myenv studybud base \_\_pycache\_\_ migrations templates/base home.html room\_form.html room.html \_\_init\_\_.py admin.py apps.py forms.py models.py tests.py urls.py views.py studybud \_\_pycache\_\_ \_\_init\_\_.py asgi.py settings.py urls.py wsgi.py templates main.html navbar.html db.sqlite3 manage.py

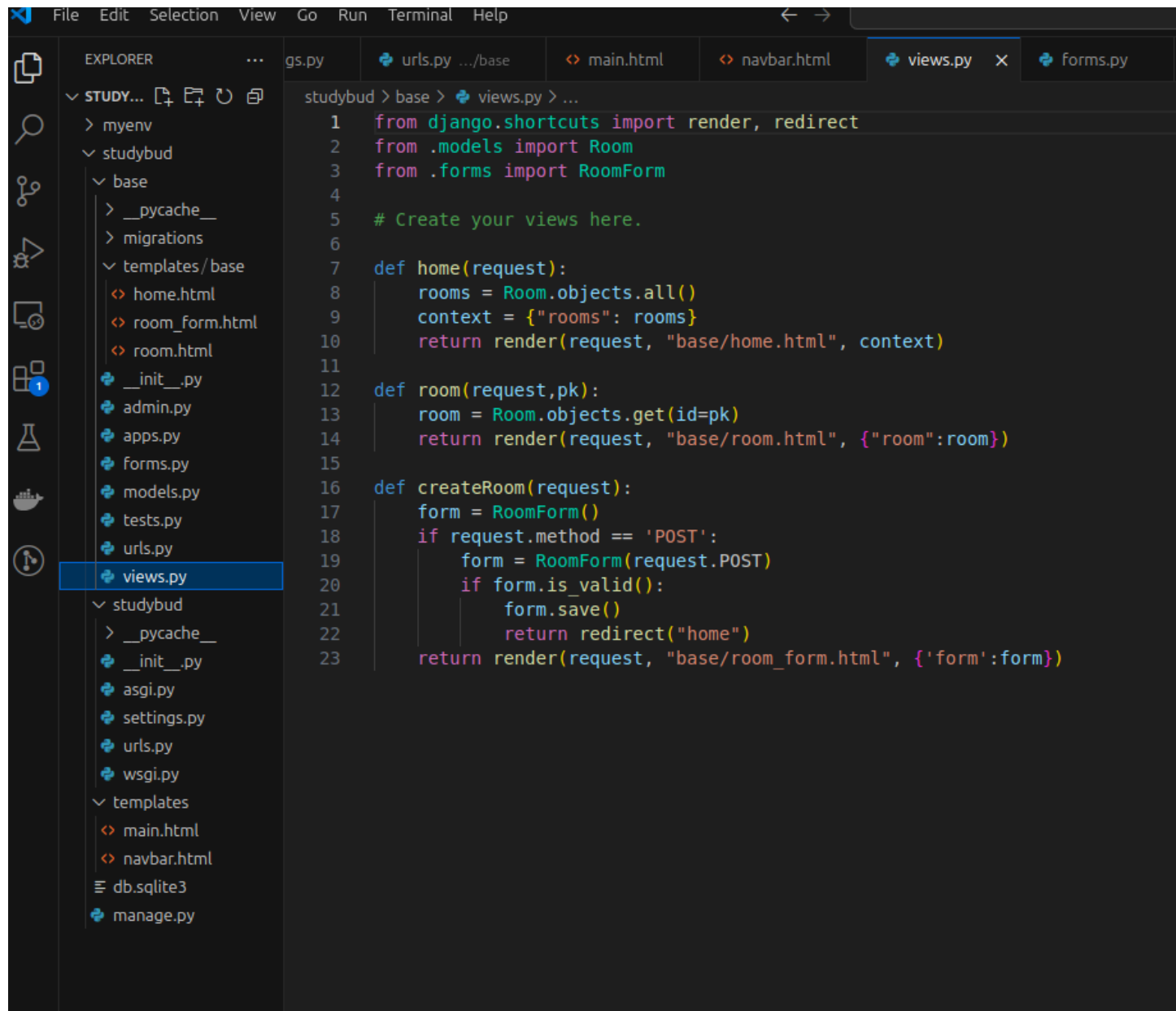
gs.py urls.py .../base main.html navbar.html views.py

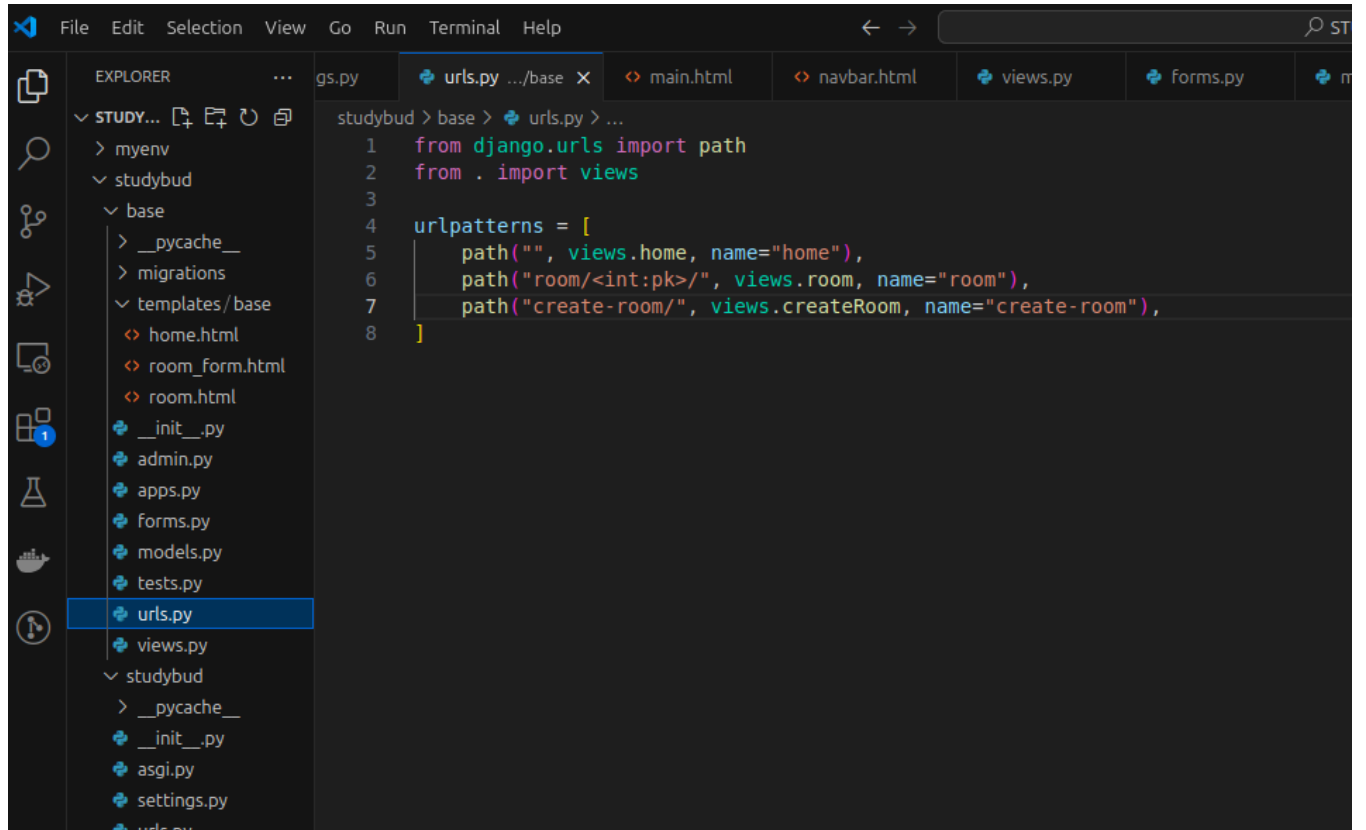
studybud > base > templates > base > room\_form.html > div > form

1 {% extends 'main.html' %}  
2  
3 {% block content %}  
4  
5 <div>  
6 <form method="POST" action="">  
7 {% csrf\_token %}  
8  
9  
10 {{form}}  
11 <input type="submit" value="Submit">  
12 </form>  
13 </div>  
14  
15 {% endblock content %}









```
21         form.save()
22         return redirect("home")
23     return render(request, "base/room_form.html", {'form':form})
24
25 def updateRoom(request, pk):
26     room = Room.objects.get(id=pk)
27     form = RoomForm(instance=room)
28
29     if request.method == "POST":
30         form = RoomForm(request.POST, instance=room)
31         if form.is_valid():
32             form.save()
33             return redirect("home")
34
35
36     return render(request, "base/room_form.html", {"form":form})
```

```
2 from . import views
3
4 urlpatterns = [
5     path("", views.home, name="home"),
6     path("room/<int:pk>/", views.room, name="room"),
7     path("create-room/", views.createRoom, name="create-room"),
8     path("update-room/<int:pk>/", views.updateRoom, name="update-room"),
9 ]
```

FileEditSelectionViewGoRunTerminalHelp

STUDYBU

gs.pyurls.py .../basemain.htmlnavbar.htmlviews.pyforms.pyapps.py

EXPLORER

STUDY...  
myenv  
studybud  
base  
\_\_pycache\_\_  
migrations  
templates/base  
home.html  
room\_form.html  
room.html  
\_\_init\_\_.py  
admin.py  
apps.py  
forms.py  
models.py  
tests.py  
urls.py  
views.py  
studybud  
templates  
main.html  
navbar.html  
db.sqlite3  
manage.py

studybud > base > templates > base > home.html > ...

1 {% extends "main.html" %}  
2  
3  
4  
5 {% block content %}  
6 <h1>HomePage</h1>  
7  
8 <a href="{% url 'create-room' %}">Create Room</a>  
9  
10 {% for room in rooms%}  
11 <a href="{% url 'update-room' room.id %}">Update</a>  
12 <p>@{{room.host.username}}</p>  
13 <p>{{room.id}}--<a href="/room/{{room.id}}">{{room.name}}</a></p>  
14 <small>{{room.topic.name}}</small>  
15 <hr>  
16 {% endfor %}  
17 {% endblock %}

PROBLEMSOUTPUTDEBUG CONSOLETERMINALPORTS

[30/Apr/2025 10:07:43] "GET /update-room/2/ HTTP/1.1" 200 1710  
[30/Apr/2025 10:07:49] "POST /update-room/2/ HTTP/1.1" 302 0  
[30/Apr/2025 10:07:49] "GET / HTTP/1.1" 200 1135  
[30/Apr/2025 10:07:54] "GET /room/1/ HTTP/1.1" 200 743  
[30/Apr/2025 10:07:58] "GET /room/2/ HTTP/1.1" 200 426

OUTLINE  
TIMELINE

Launchpad 0 0

# 🔧 Step-by-Step Instructions for Building the Django Blog Project

## 1. Setup Your Django Project

Create a Django project:

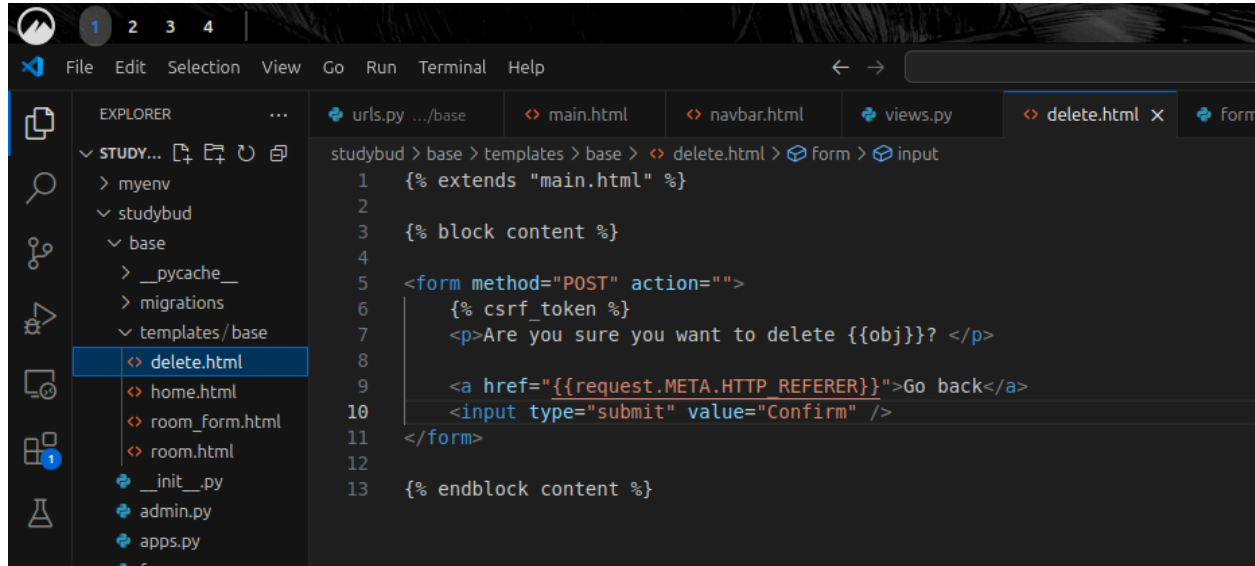
```
bash

django-admin startproject blog_project
cd blog_project
```

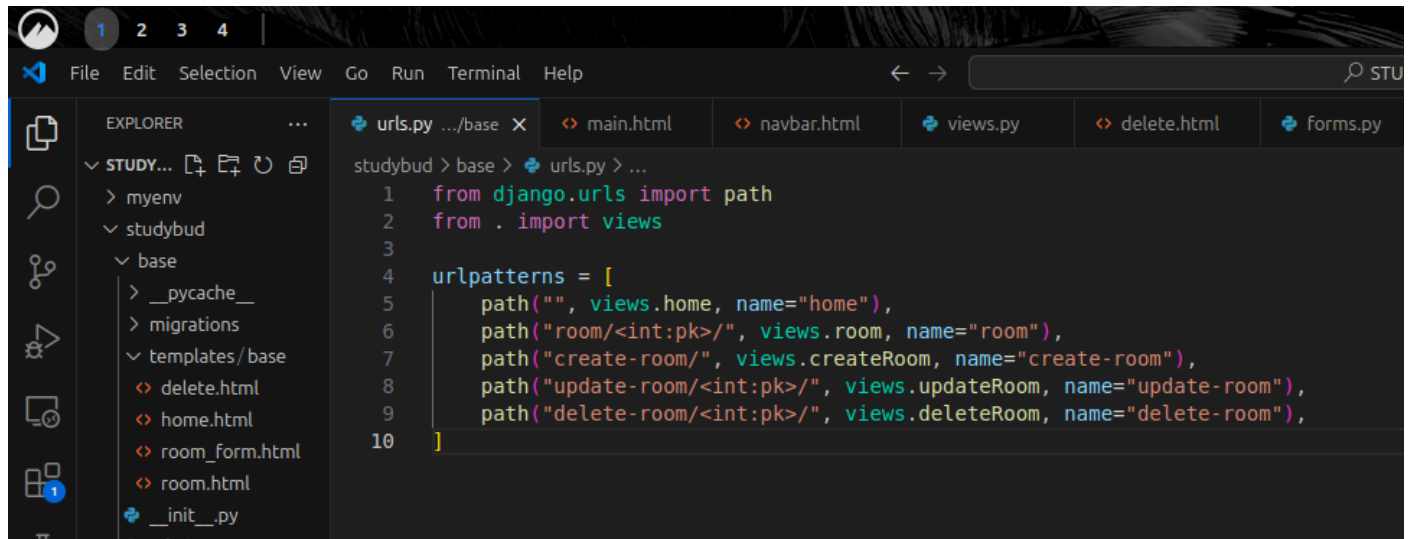
Create an app inside the project:

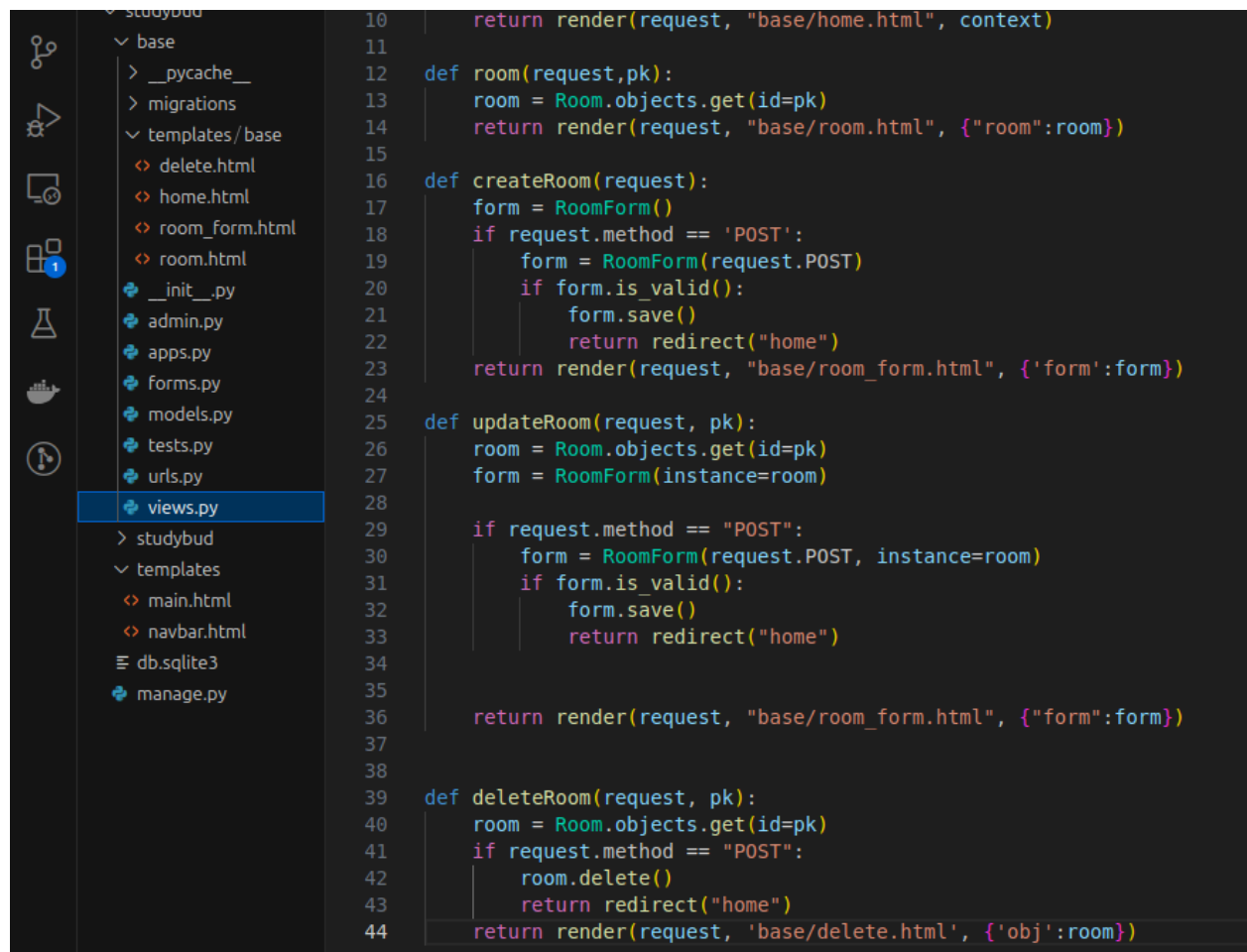
```
bash

python manage.py startapp post
```









The image shows a code editor with a dark theme. On the left is a file explorer sidebar showing the project structure. The main area displays the code for `views.py`.

**File Explorer (Left Sidebar):**

- studybud
  - base
    - \_\_pycache\_\_
    - migrations
    - templates/base
      - delete.html
      - home.html
      - room\_form.html
      - room.html
    - \_\_init\_\_.py
    - admin.py
    - apps.py
    - forms.py
    - models.py
    - tests.py
    - urls.py
    - views.py**
  - studybud
  - templates
    - main.html
    - navbar.html
  - db.sqlite3
  - manage.py

**Code Editor (Main Area):**

```
10     return render(request, "base/home.html", context)
11
12 def room(request, pk):
13     room = Room.objects.get(id=pk)
14     return render(request, "base/room.html", {"room": room})
15
16 def createRoom(request):
17     form = RoomForm()
18     if request.method == 'POST':
19         form = RoomForm(request.POST)
20         if form.is_valid():
21             form.save()
22             return redirect("home")
23     return render(request, "base/room_form.html", {'form': form})
24
25 def updateRoom(request, pk):
26     room = Room.objects.get(id=pk)
27     form = RoomForm(instance=room)
28
29     if request.method == "POST":
30         form = RoomForm(request.POST, instance=room)
31         if form.is_valid():
32             form.save()
33             return redirect("home")
34
35     return render(request, "base/room_form.html", {"form": form})
36
37
38
39 def deleteRoom(request, pk):
40     room = Room.objects.get(id=pk)
41     if request.method == "POST":
42         room.delete()
43         return redirect("home")
44     return render(request, 'base/delete.html', {'obj': room})
```

