Web Development Lesson Plan

|  |  |  |
| --- | --- | --- |
| Session: 2 | Topic: Creating your first Django App | Day: 2 |

|  |
| --- |
| Lesson Objective:  This exercise will...   * Instruct students how to setup new Django applications.   The purpose of this lesson is also that   * Students can understand basic Django functionality. |
| Materials Needed:   * VS Code * Web browser |
| Agenda:   1. Create Django App (15 minutes) 2. Register App (5 minutes) 3. Hello world (30 minutes) 4. Run app (5 minutes) |
| Procedure:  Student can do this fully independently by reading the instructions, or someone can read the instructions to make sure the task is understood before doing the activity. The reflection part can be done with guided instructions, or in pairs with other students if available to compare notes. |
| Assessment/Check for learning:   * Students can identify concrete things that they missed in their observation (note that this is not a bad thing, the purpose of this lesson is to realize that intentionally looking for information comes with the ‘penalty’ of reduced attention to other things. |

What if:

* Student has already shown proficiency in that skill?
  + Pick a movie or episode in which a lot of things are happening, so that there are more things for the student to keep track of.
* Student is having a hard day and needs special accommodations?
  + Pick a movie or episode in which only a few things are happening, so that there are less things for the student to keep track of.
  + Instead of writing, the student can also take spoken notes with an audio recorder. This will be a bit trickier in the second part but audio in the movie can be supported with subtitles.

NOTES:

* Duration is approximately 50 minutes; can be adjusted by shortening the duration of the film clip. Less than 10 minutes is not recommended unless really necessary.

**Instructions**

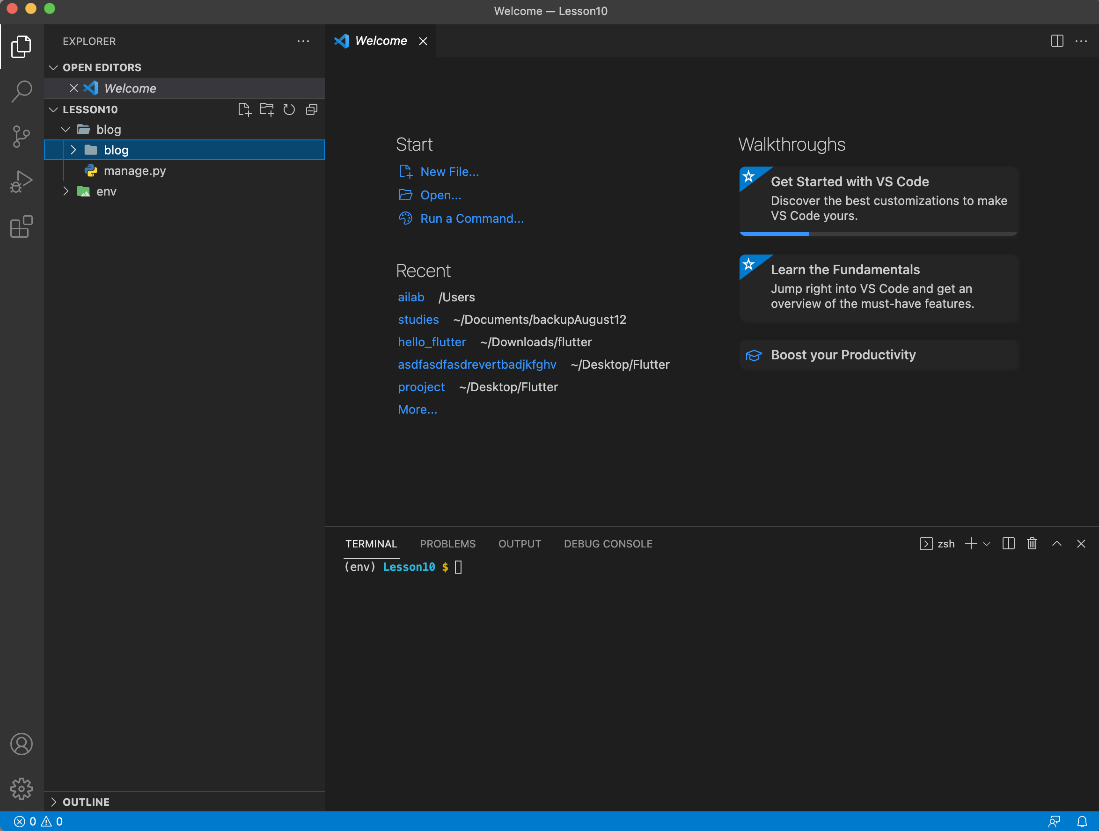
**Create Django App (15 minutes)**

Step 1:

Using the project folder that was created in Lesson 10, we will create our first Django app. A Django project folder contains all our code in a neat structure. Django apps live within Django projects. It is possible to have multiple apps within a Django project folder.

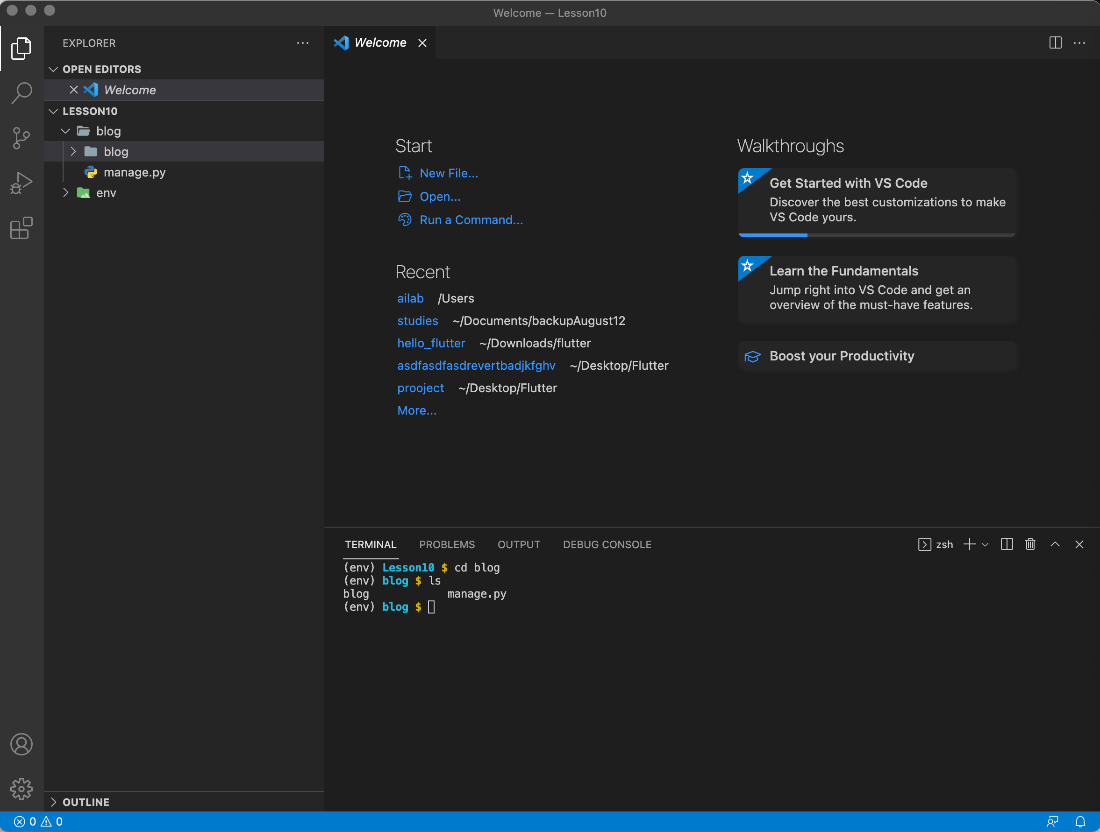
Step2 :

Open the project folder in VS Code and activate the virtual environment. Your window should look like this:



Step 3:

CD into the first “blog” folder. Once you CD into the fist “blog” folder you can type **ls** to confirm that you see “blog” and “manage.py” in the terminal. You should see this:



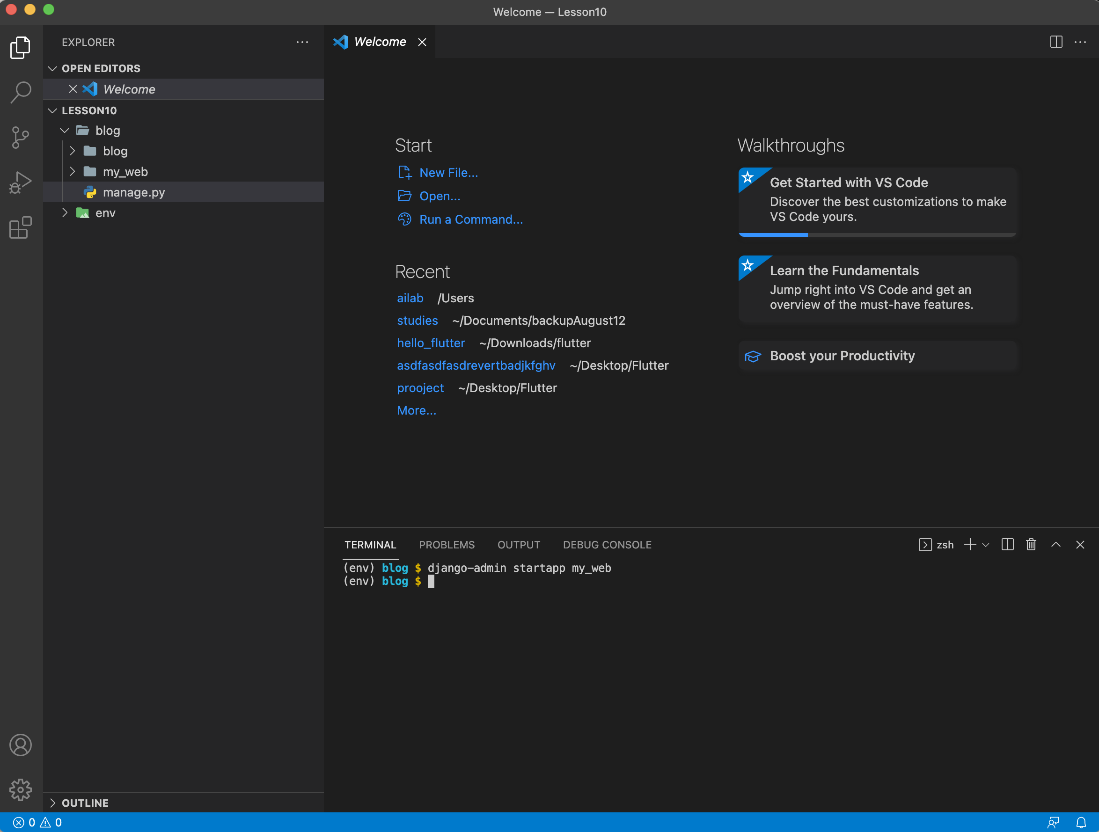
Step 4:

To start a new Django app you will run the command below. We specify the name of the app at the end of the command. In this case the name of our app is called “my\_web” Run the following:

django-admin startapp my\_web

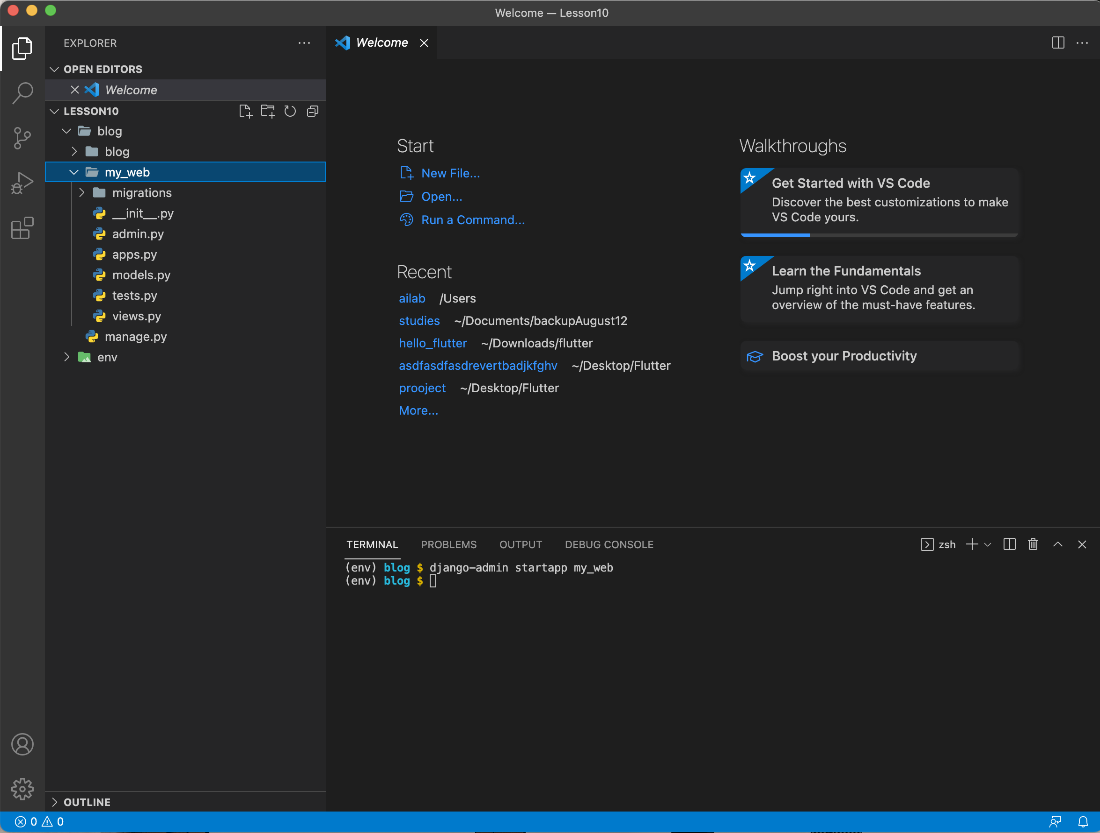
Step 5:

You will now see a folder called “my\_web” in the side bar:



Step 6:

Expand “my\_web” folder. You should see the following:



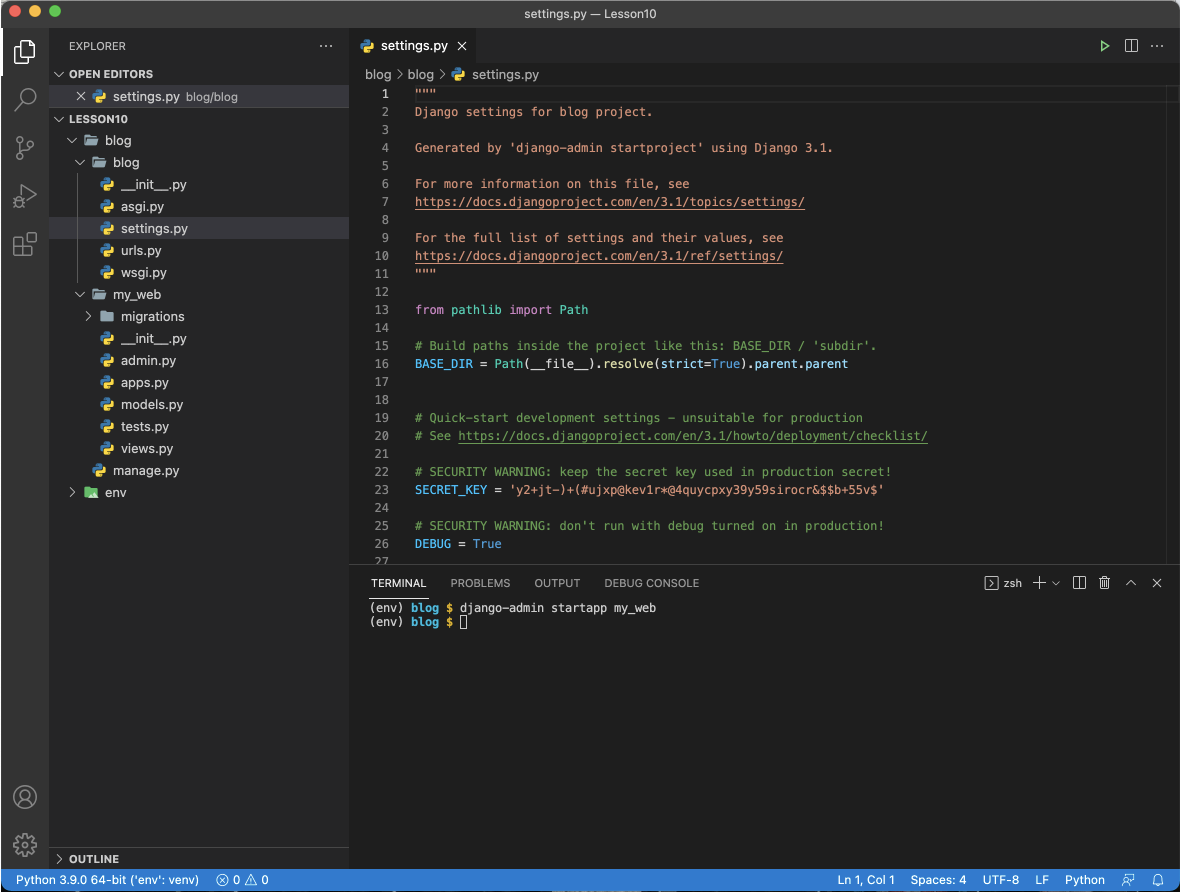
**Register app (30 minutes)**

Step 1:

Now that you have your Django app files setup we will need to register the “my\_web” app to start coding.

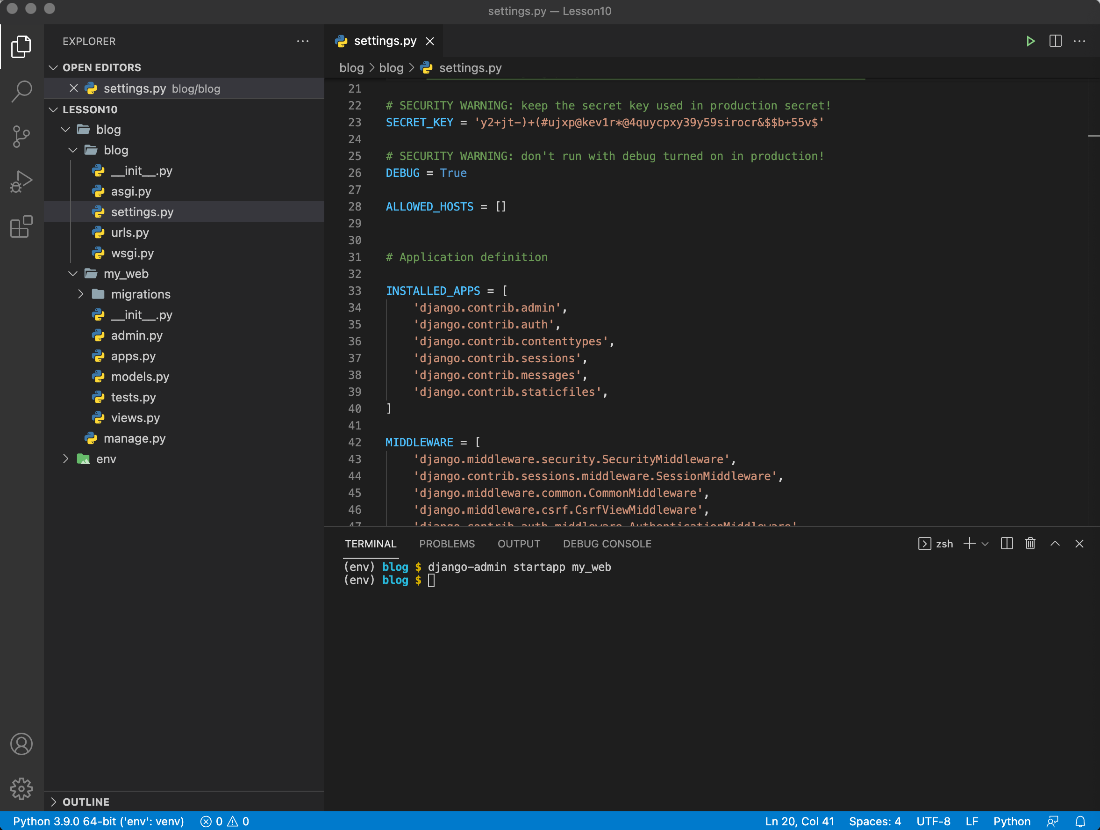
Step 2:

In the side-bar, open the “blog” folder that is directly above “my\_web” folder. Within the “blog” folder there are 5 python files. Double click on the file named “settings.py” the file should open in the editor (middle section) of VS Code. You should see the following:



Step 3:

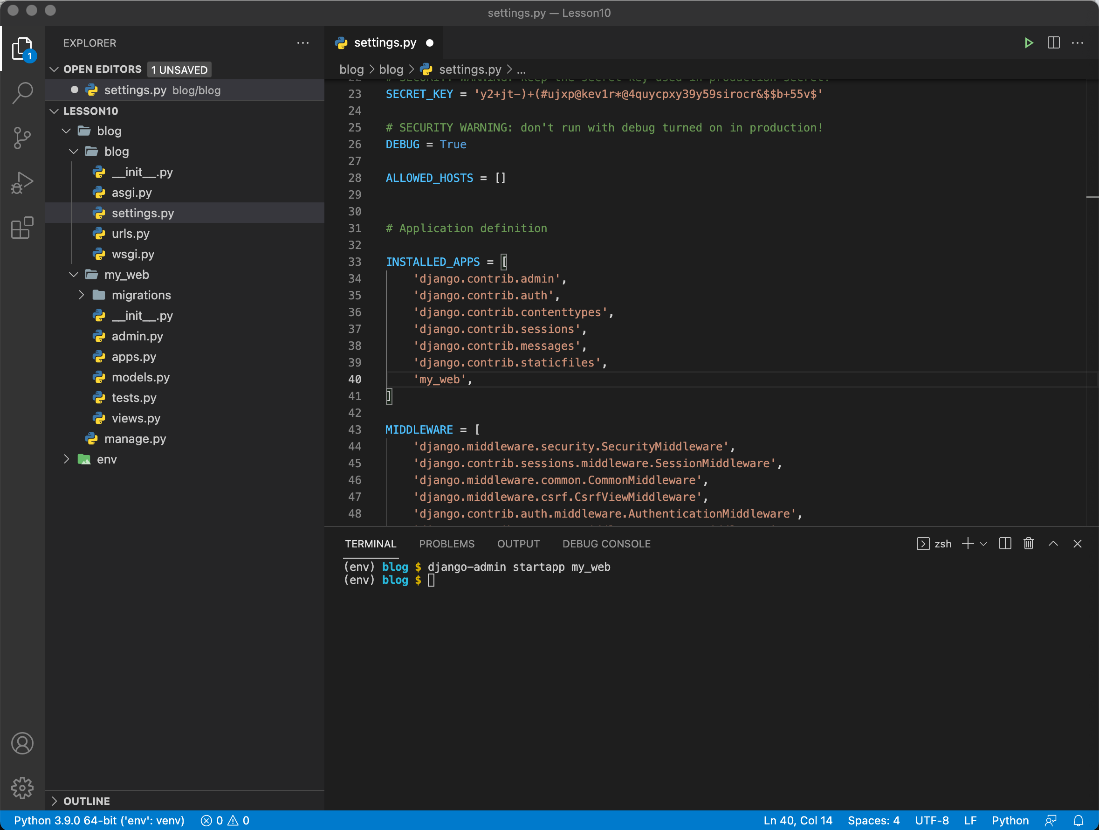
In the code editor, scroll down to line 39. You should see:



Step 4:

After line 39 add the following line:

'my\_web',



Step 5:

Save the file. And now you have successfully registered the app. Each time you create a new app you will need to register the app.

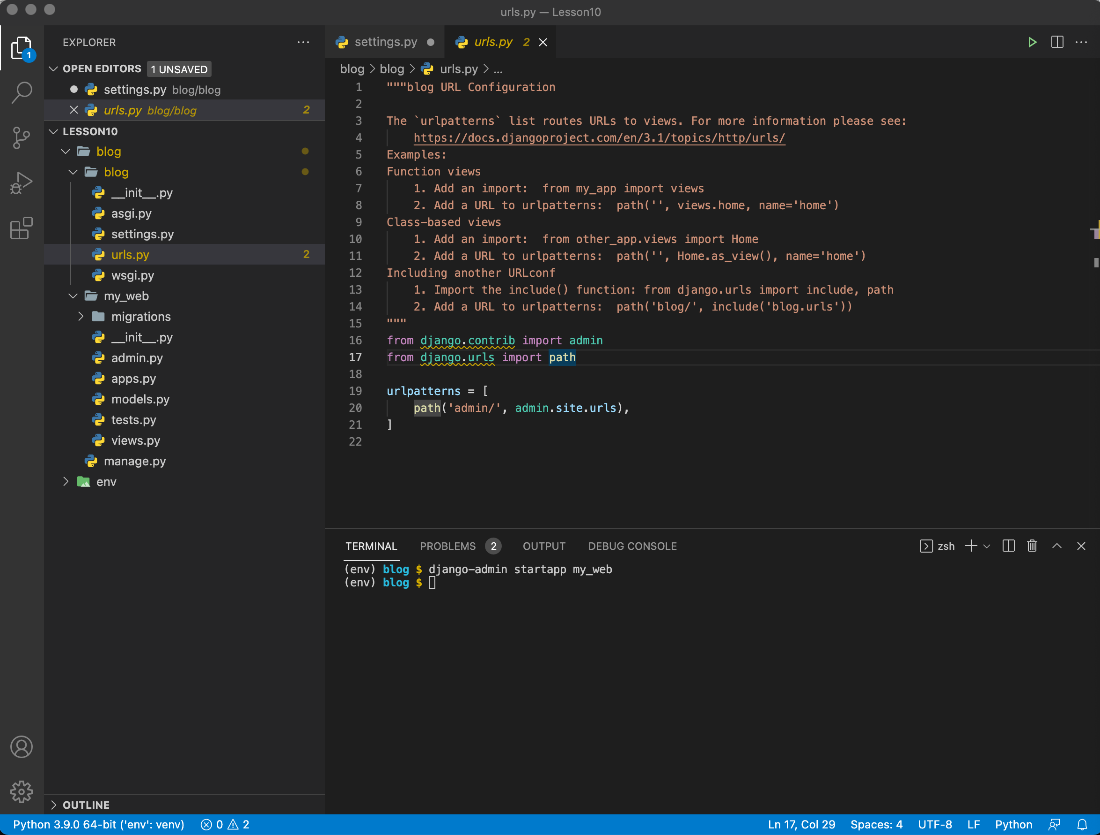
**Hello world (30 minutes)**

Step 1:

We will create a simple web page that displays a “Hello world” text paragraph.

Step 2:

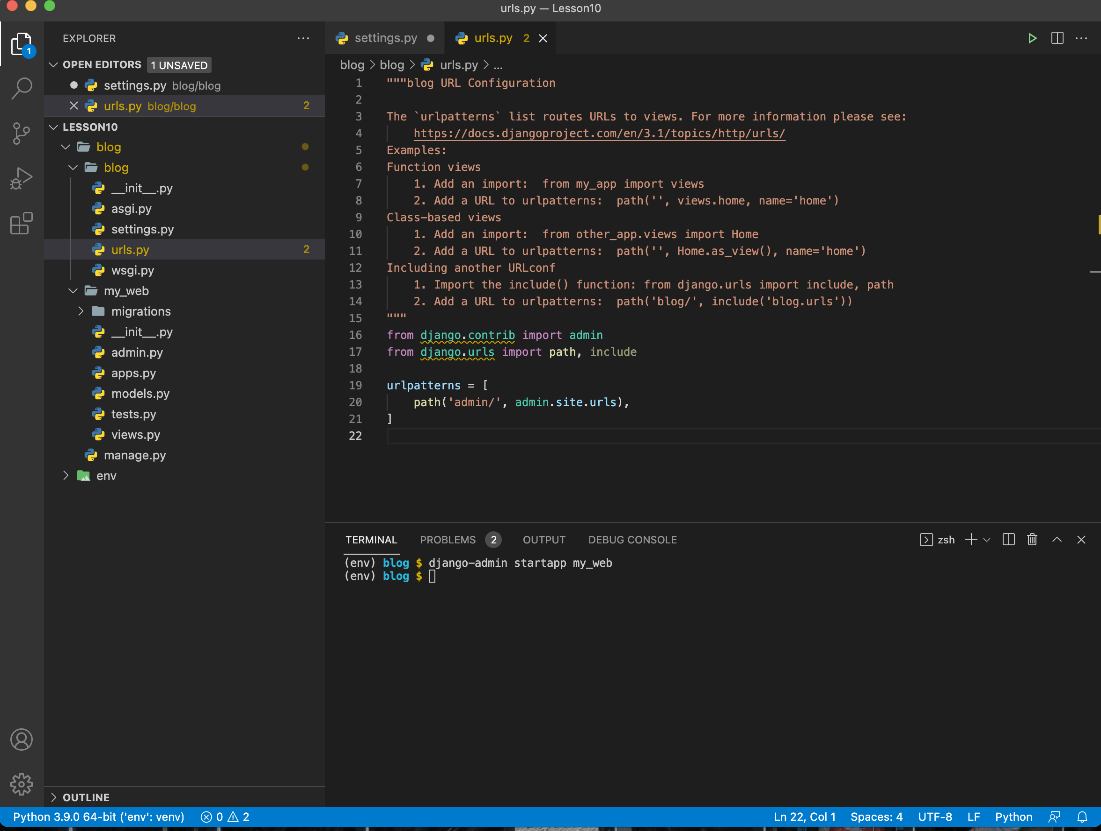
Inside the “blog” folder there is a file called “urls.py”. This file is directly below “settings.py”. Open “urls.py” You should see the following:



This file controls routing for our app. The Django admin function uses this file to display the correct url files for our app.

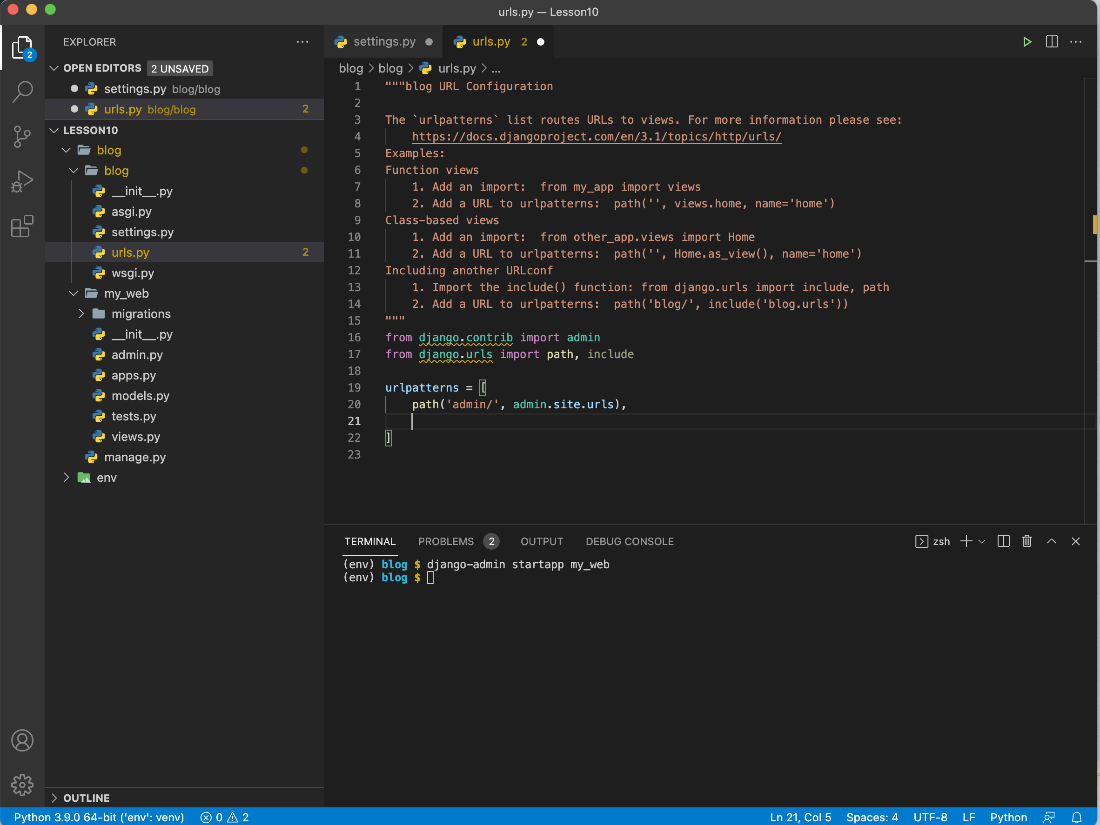
Step 3:

Now add the following text “**, include**” at the end of line 17, directly after the word “path”. **Note: Do not forget the coma.** You should now have the following:



Step 4:

Now after line 20, add a new line like so:



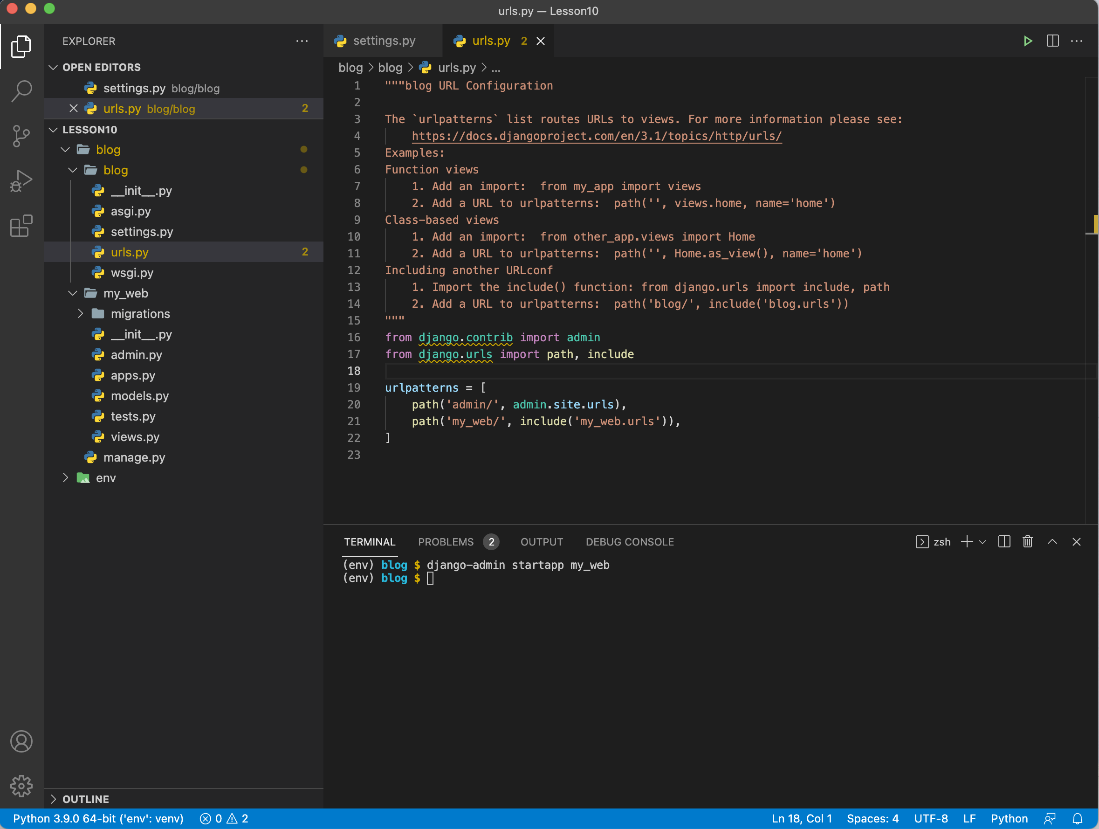
Step 5:

In line 21 add the following code:

path('my\_web/', include('my\_web.urls')),

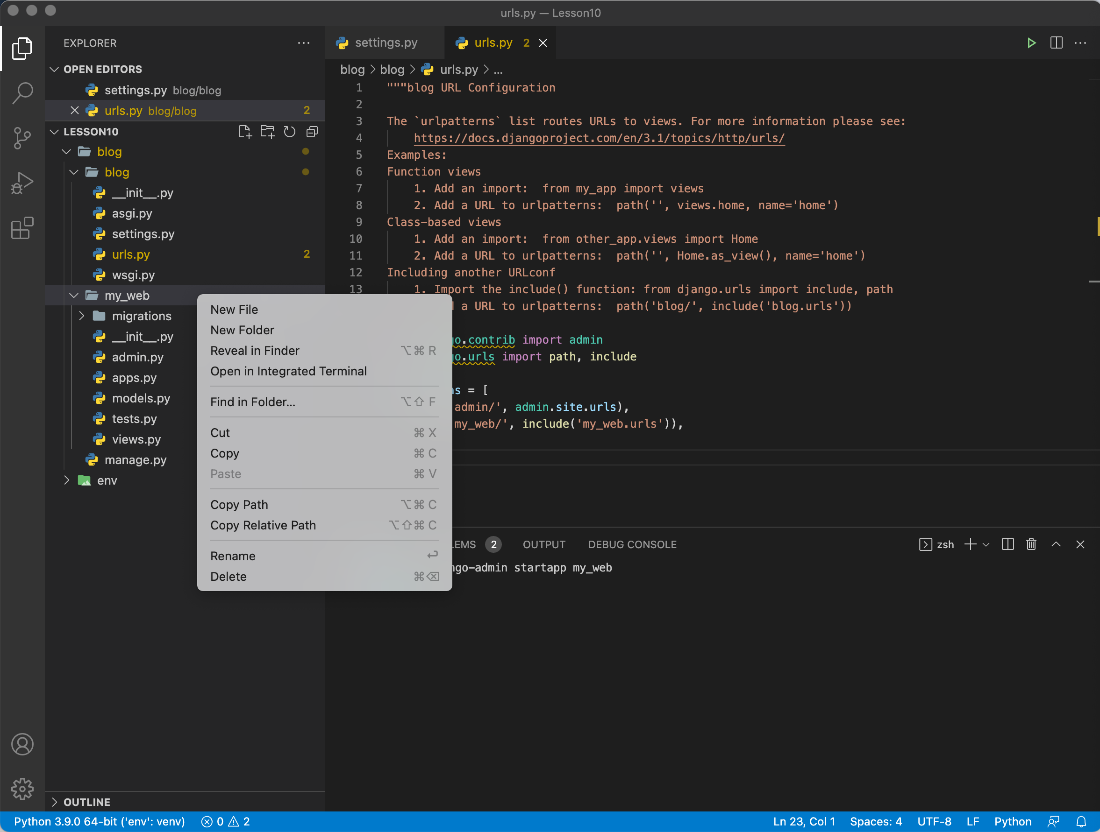
Step 6:

Your file should look like this. Finally save the file.



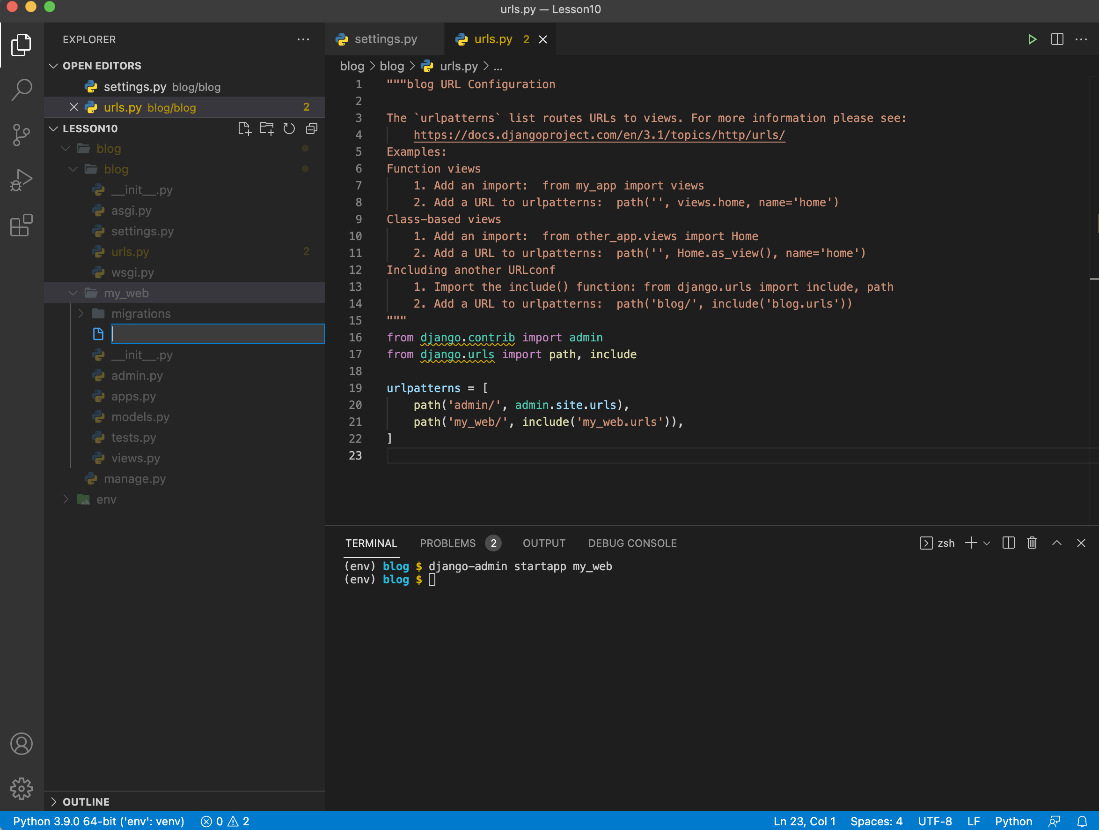
Step 7:

Next, expand the “my\_web” folder. Right click on “my\_web” in the side bar. You should see this:



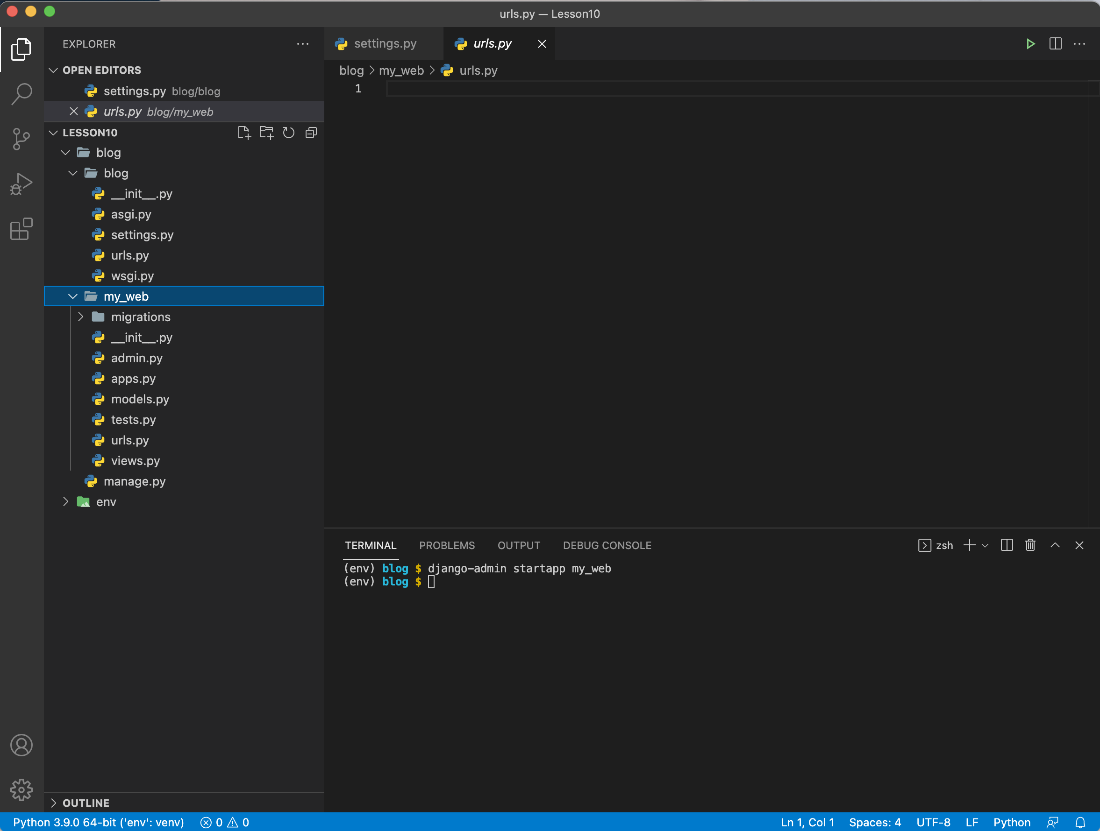
Step 8:

Click on “New File”. After you click “New File” you will see a blank rectangle. In this blank rectangle type “urls.py” and press enter.



Step 9:

Now you should have the following folder structure:

  
Step 12:

The “urls.py” file that you just created should have opened automatically in the text-editor. If not, simply double-click on the file to open it.

Step 13:

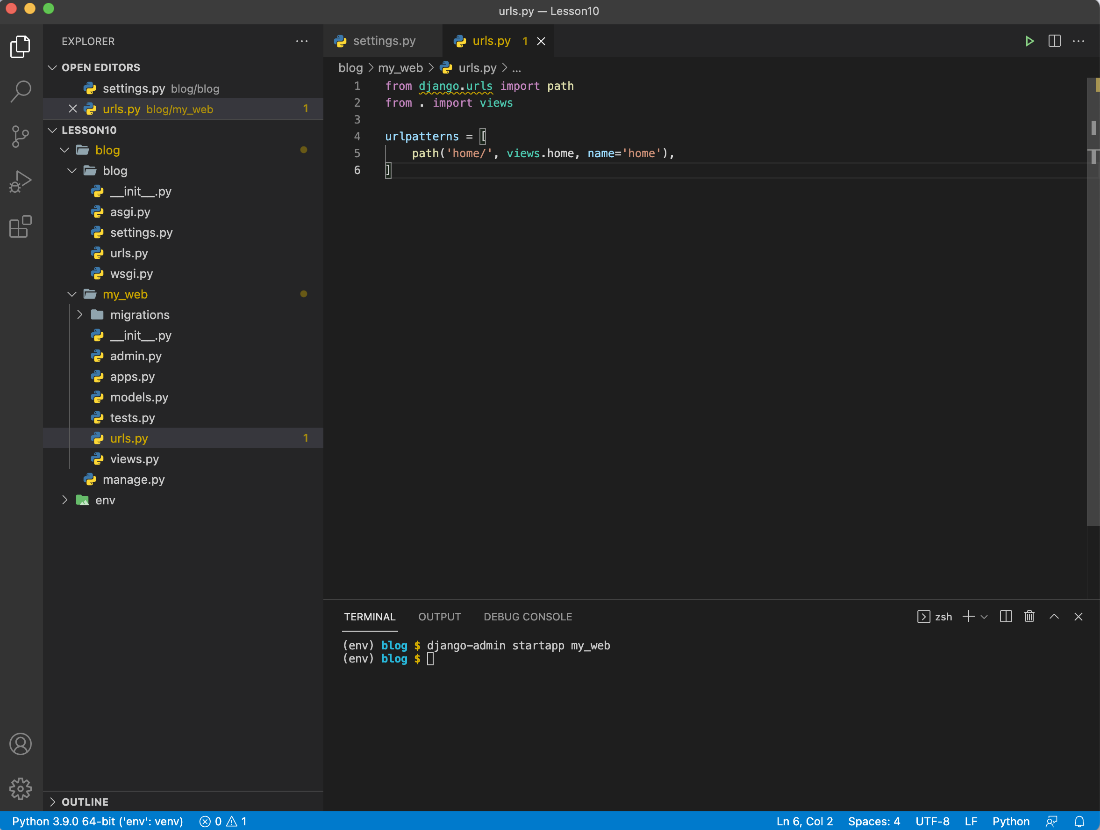
This urls.py will route HTML pages to their correct destination. We will create HTML pages in a few steps but first we need to create a route.

Step 14:

Add the following code to “urls.py” file that you just opened:

from django.urls import path  
from . import views  
  
urlpatterns = [   
 path('home/', views.home, name='home'),  
]

Save your file. Your file should look like this:



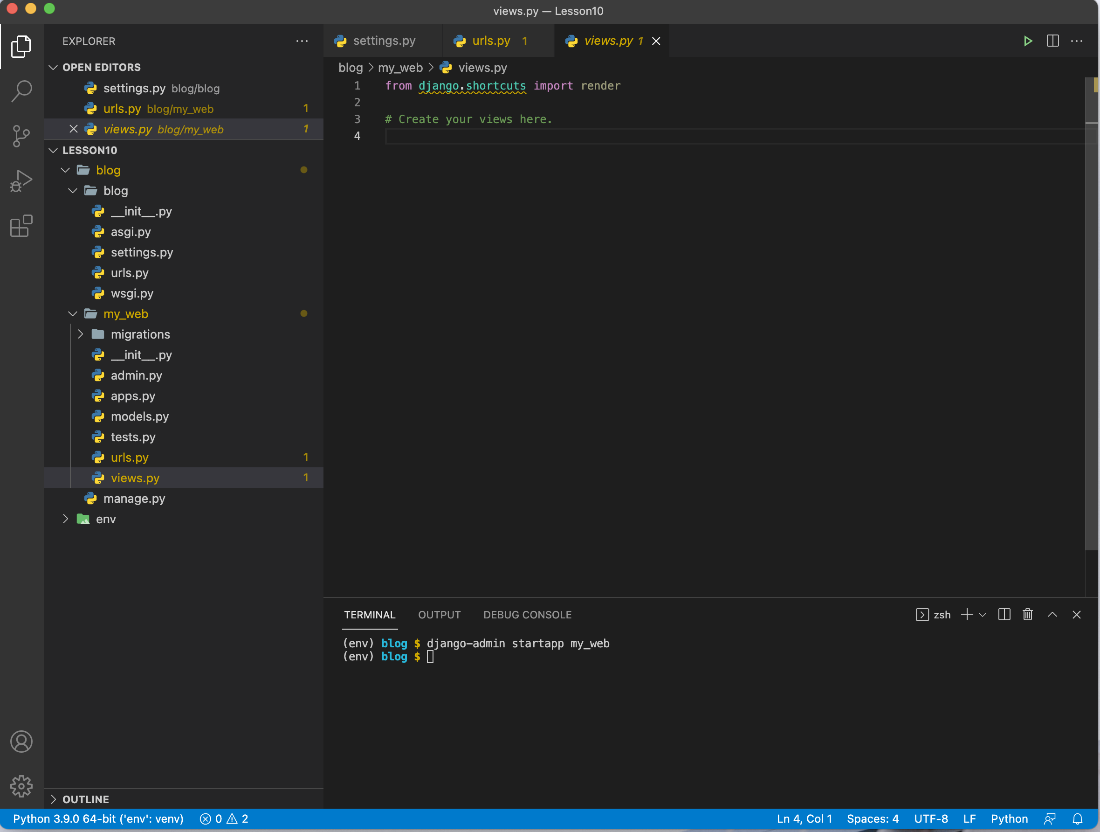
The code above creates a route to an HTML file called “home”. We have not created this file yet, but it is important to first set the route.

Step 15:

Now we will create a View for our home page. A view is a Django function that tells the Django admin what to display on a web browser when a certain directory is visited.

Step 16:

Within the folder “my\_web”, open the “views.py” file. You should see the following:

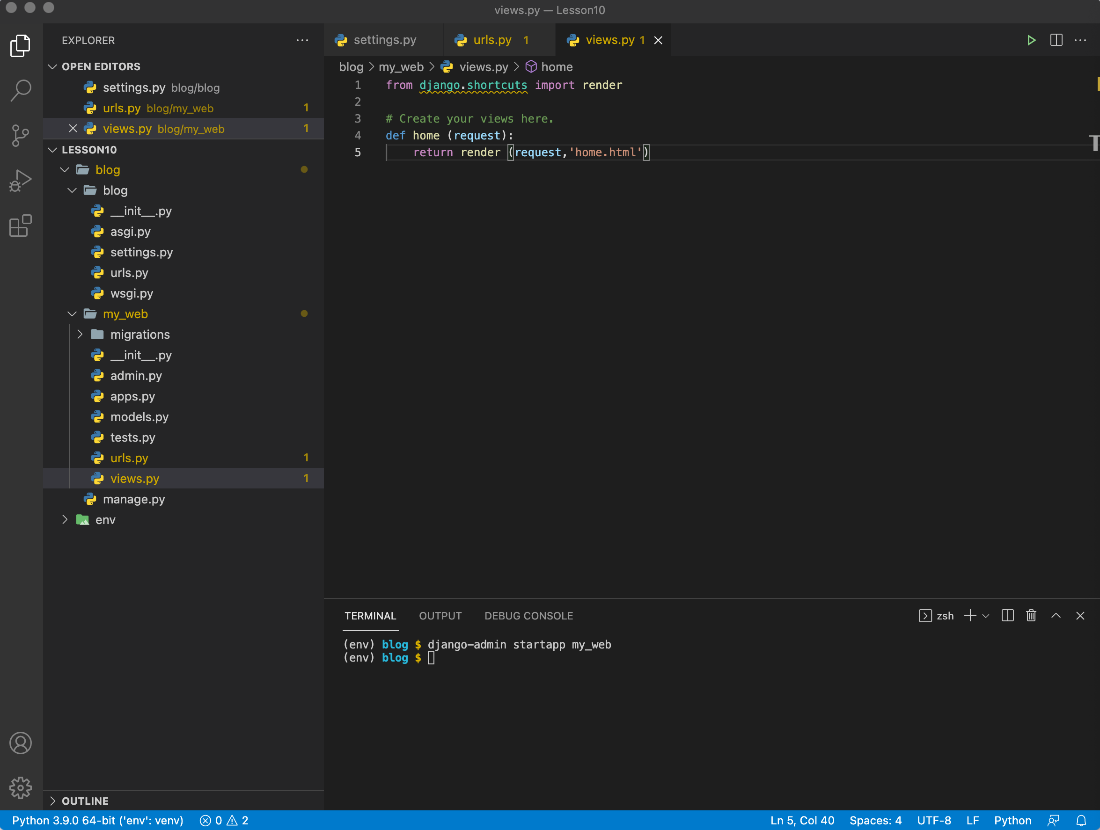


Step 17:

At line 4 we will create our first view. Type the following code:

def home (request):  
 return render (request, ‘home.html’)

Your file should now look like this:

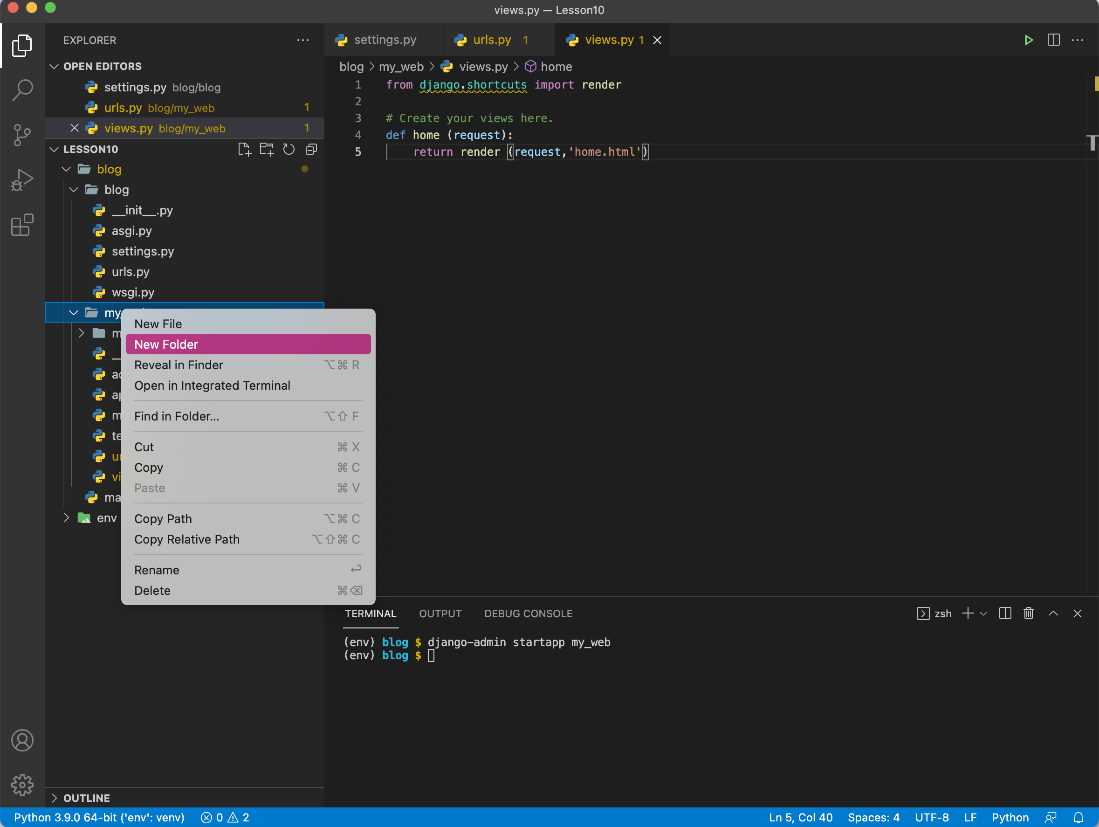


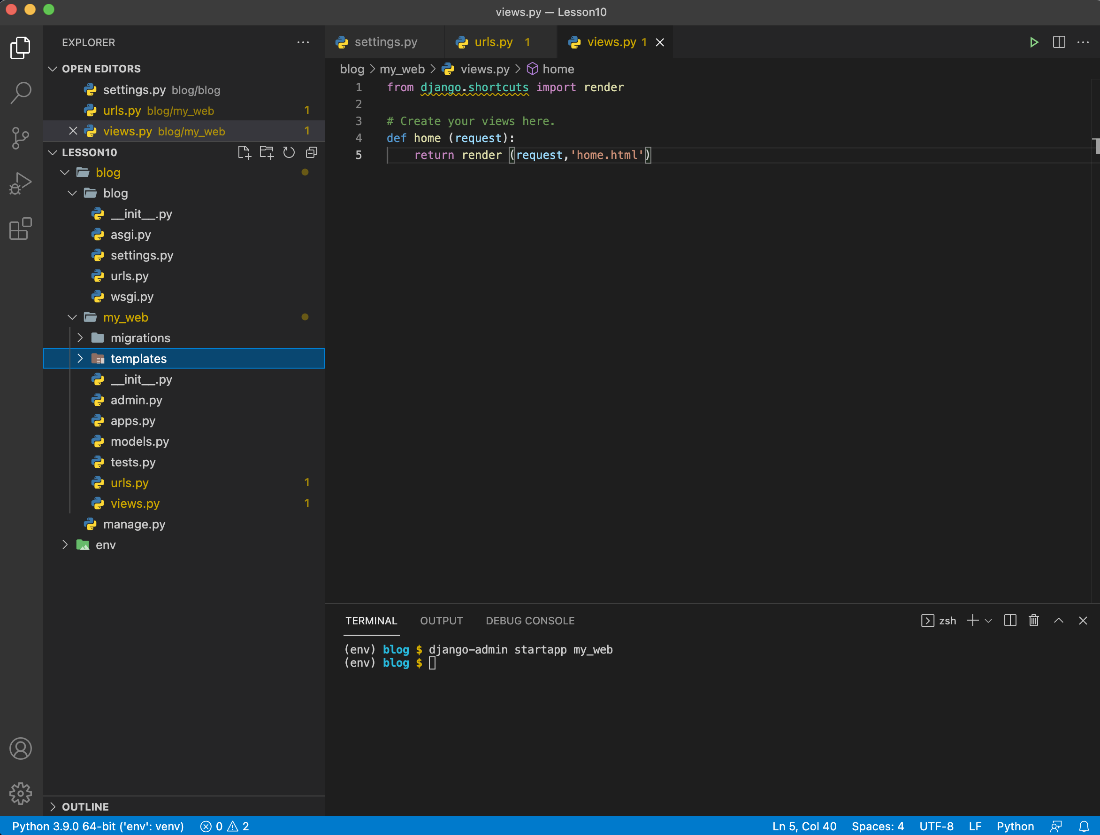
Step 18:

Save the file.

Step 19:

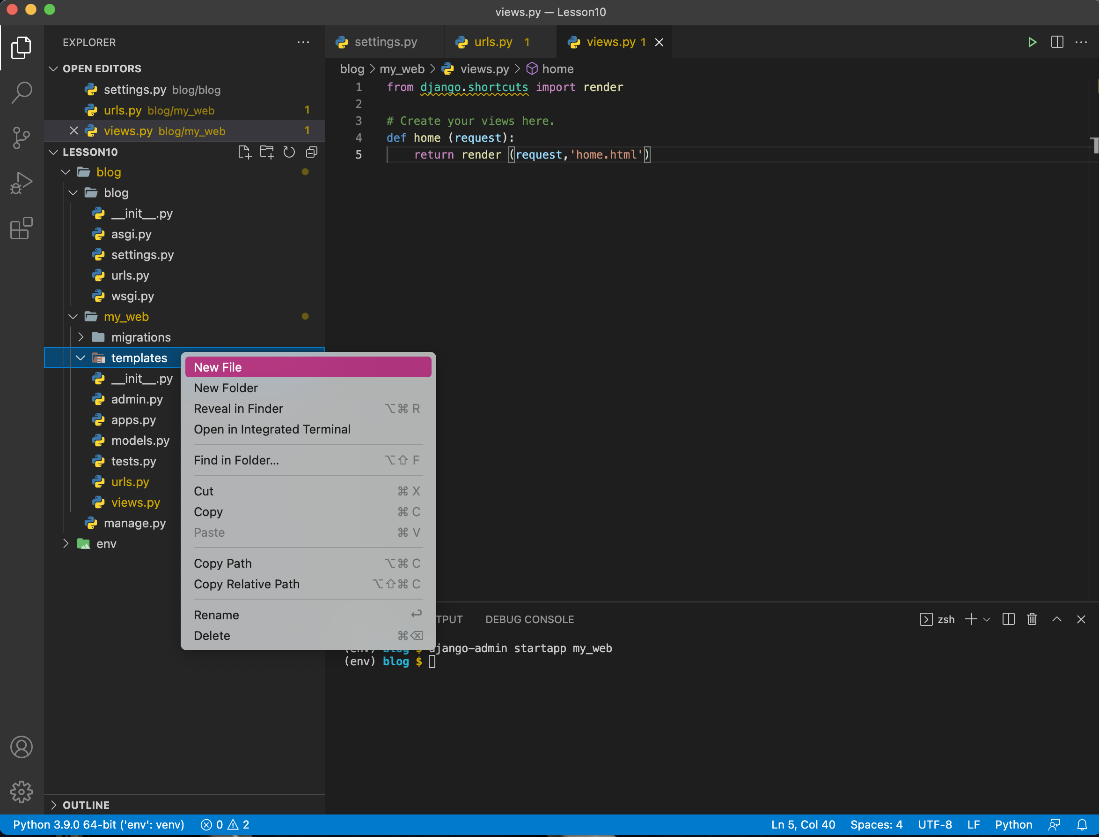
Next, we will create the actual HTML page for home. Within the “my\_web” folder, right click on “my\_web” and create a new folder called “templates”

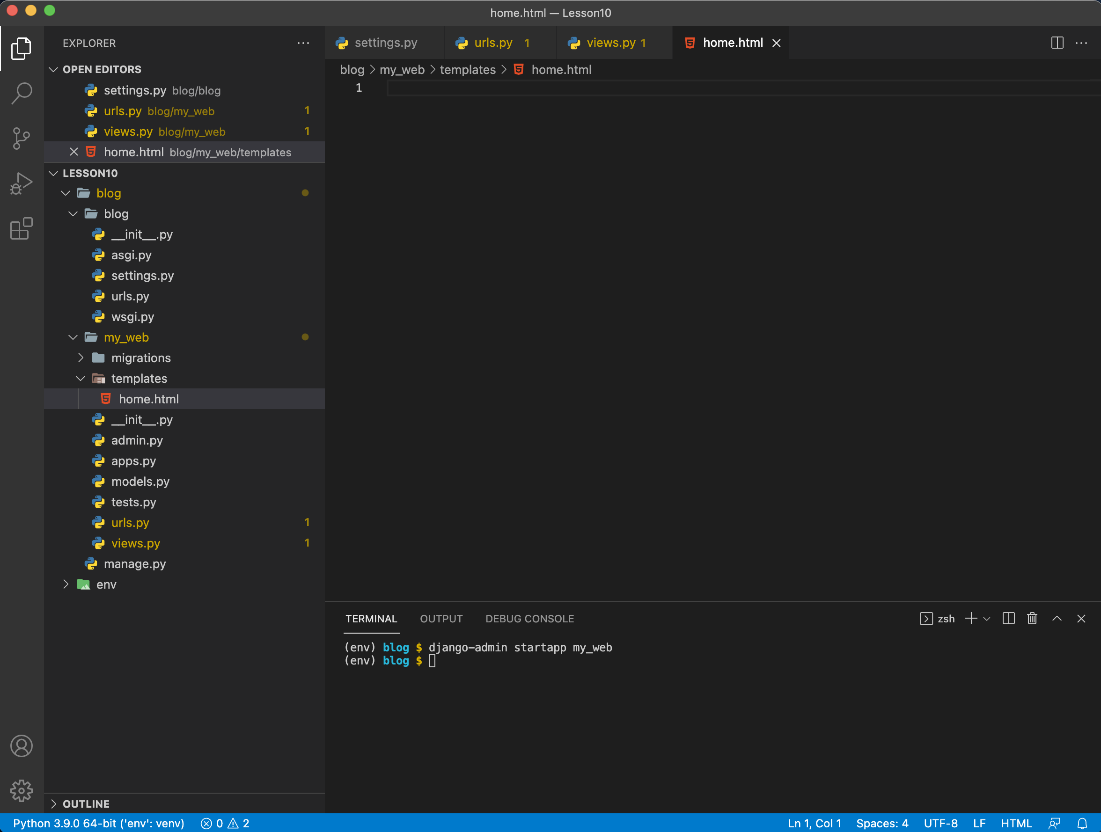




Step 20:

Right click on the “templates” folder and add a new file called “home.html”



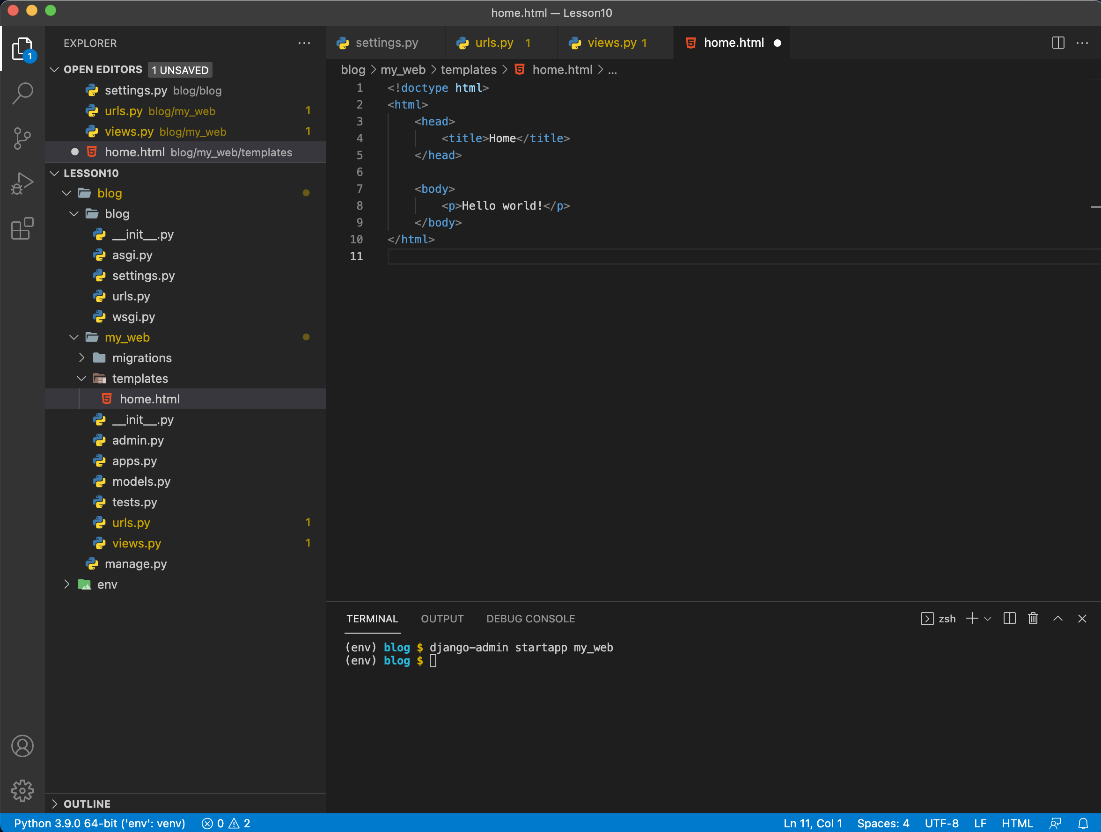


Step 21:

Once “home.html” is open, type the following code in the file:

<!doctype html>  
<html>  
 <head>  
 <title>Home</title>  
 </head>  
  
 <body>  
 <p>Hello world!</p>  
 </body>  
</html>

Your file should look like this:



Step 22:

Save the file. We have now created your first app. Next we will run the app.

**Run app(5 minutes)**

Step 1:

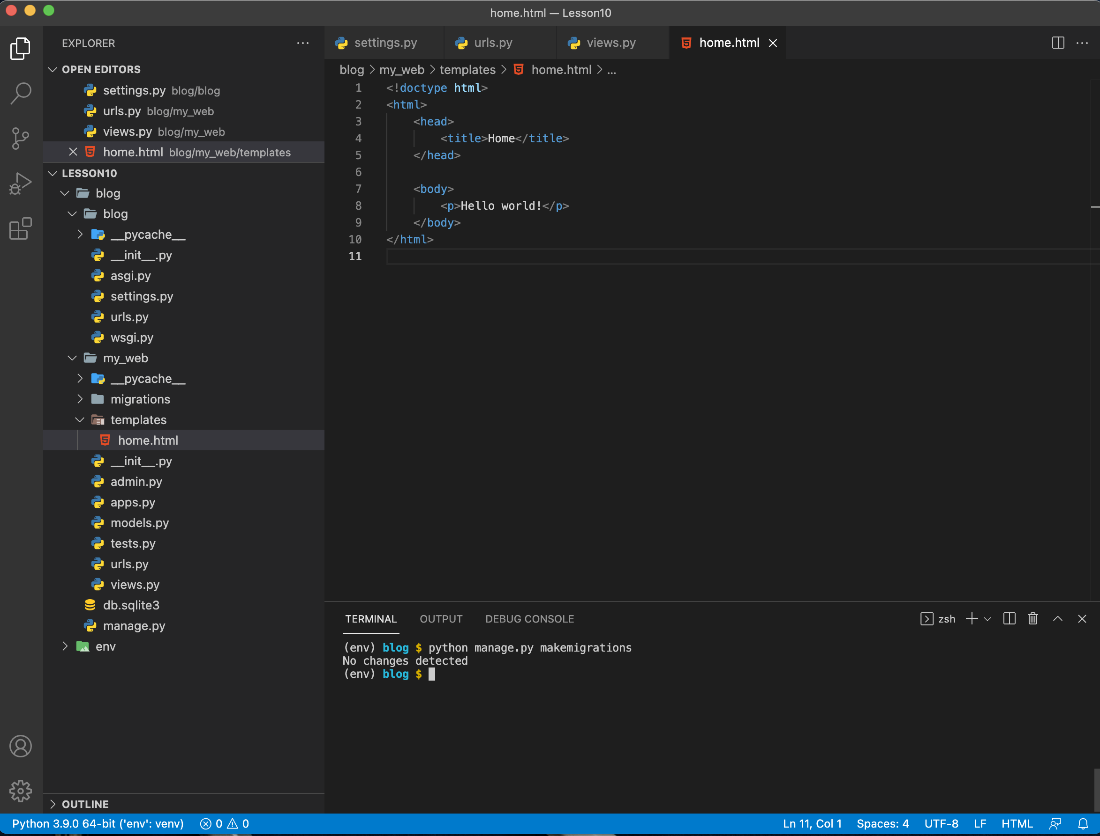
Make sure you have saved all your files.

Step 2:

In the terminal type the following command:

python manage.py makemigrations

You should see this:

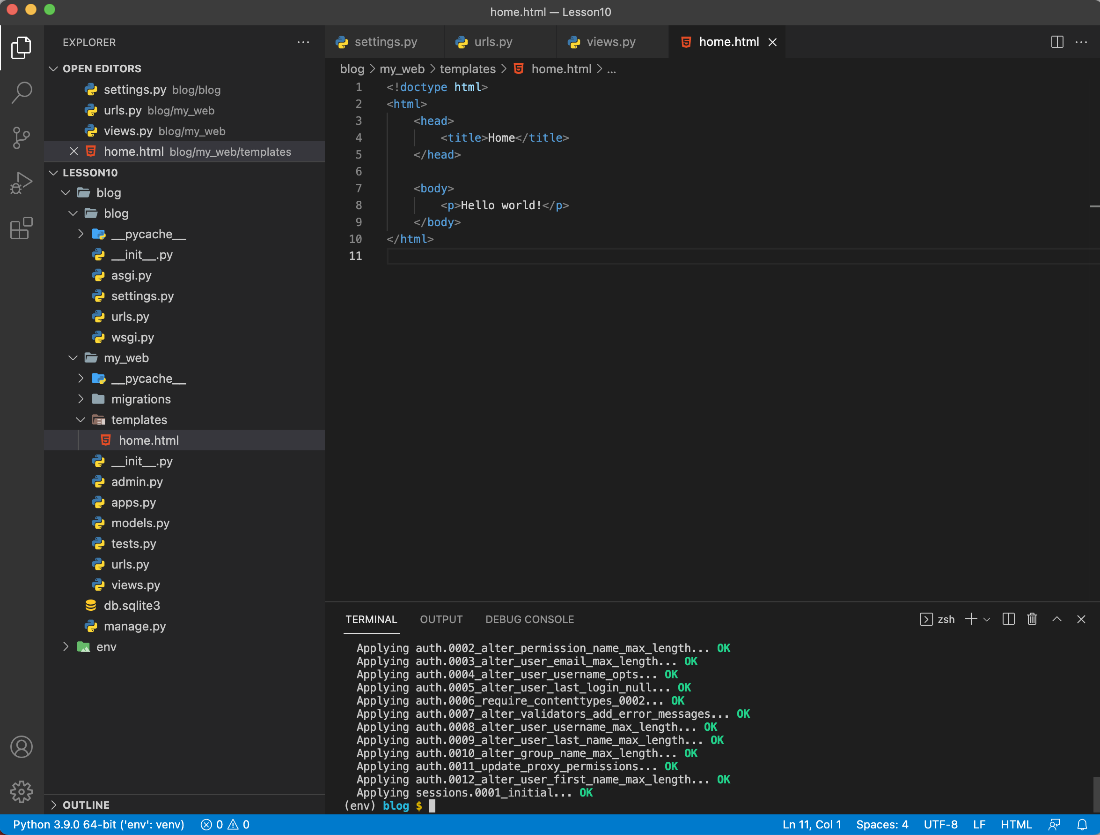


Step 3:

Now enter the following command:

python manage.py migrate

You should see the following:

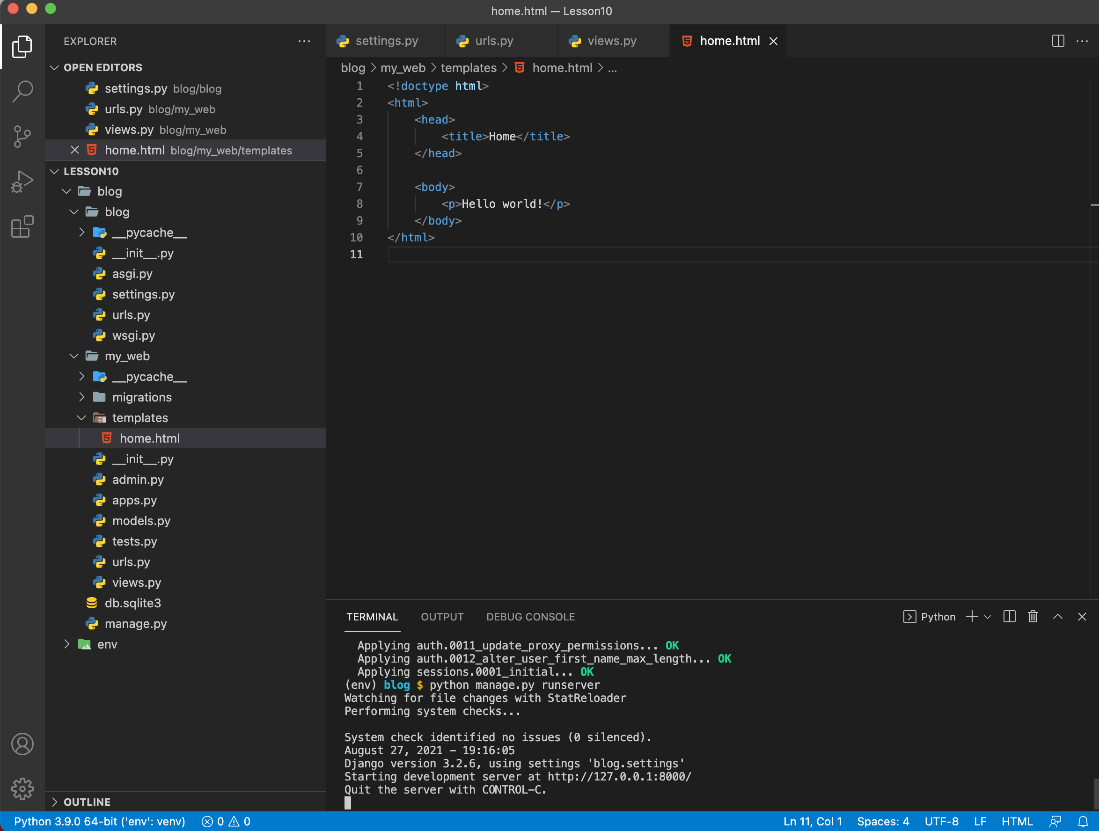


Step 4:

Finally, enter the following command to run the app:

python manage.py runserver

This will run the server. You will see this:

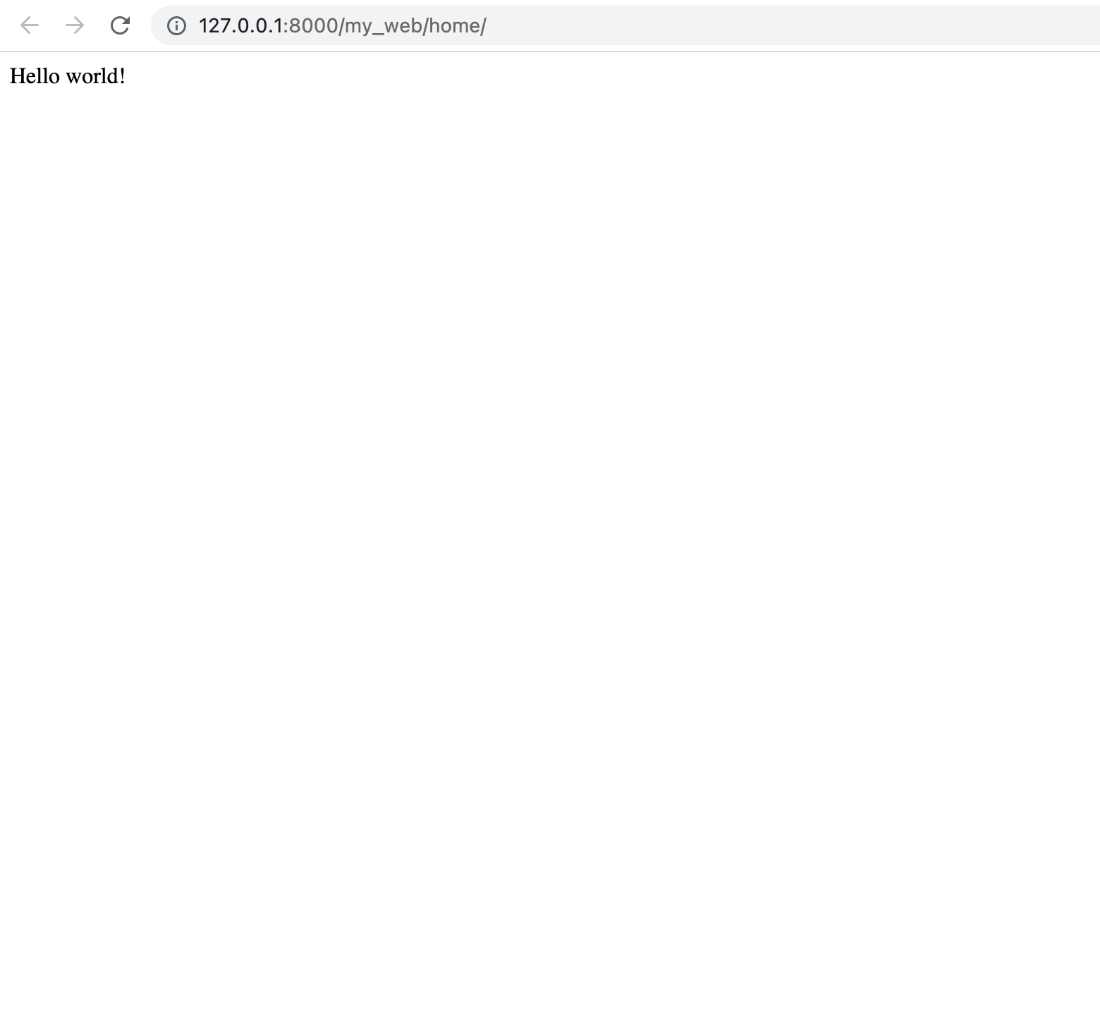


Step 5:

Open a web browser and navigate to the following url:

<http://127.0.0.1:8000/my_web/home>

You should see your web page:



In the following lesson you will be introduced to the Django web framework.