



Create Your First Django App

Estimated time needed: 20 minutes

Learning Objectives

- Create your first Django project and app using command line utils
- Create your first Django view to return a simple HTML page

Create Your First Django Project

Before starting the lab, make sure your current Theia directory is `/home/project`.

First, we need to install Django related packages.

- Go to terminal and run:

```
python3 -m pip install Django
```

- Once the installation is finished, you can create your first Django project by running:

```
django-admin startproject firstproject
```

A folder called `firstproject` will be created which is a container wrapping up settings and configurations for a Django project.

- If your current working directory is not `/home/project/firstproject`, `cd` to the project folder

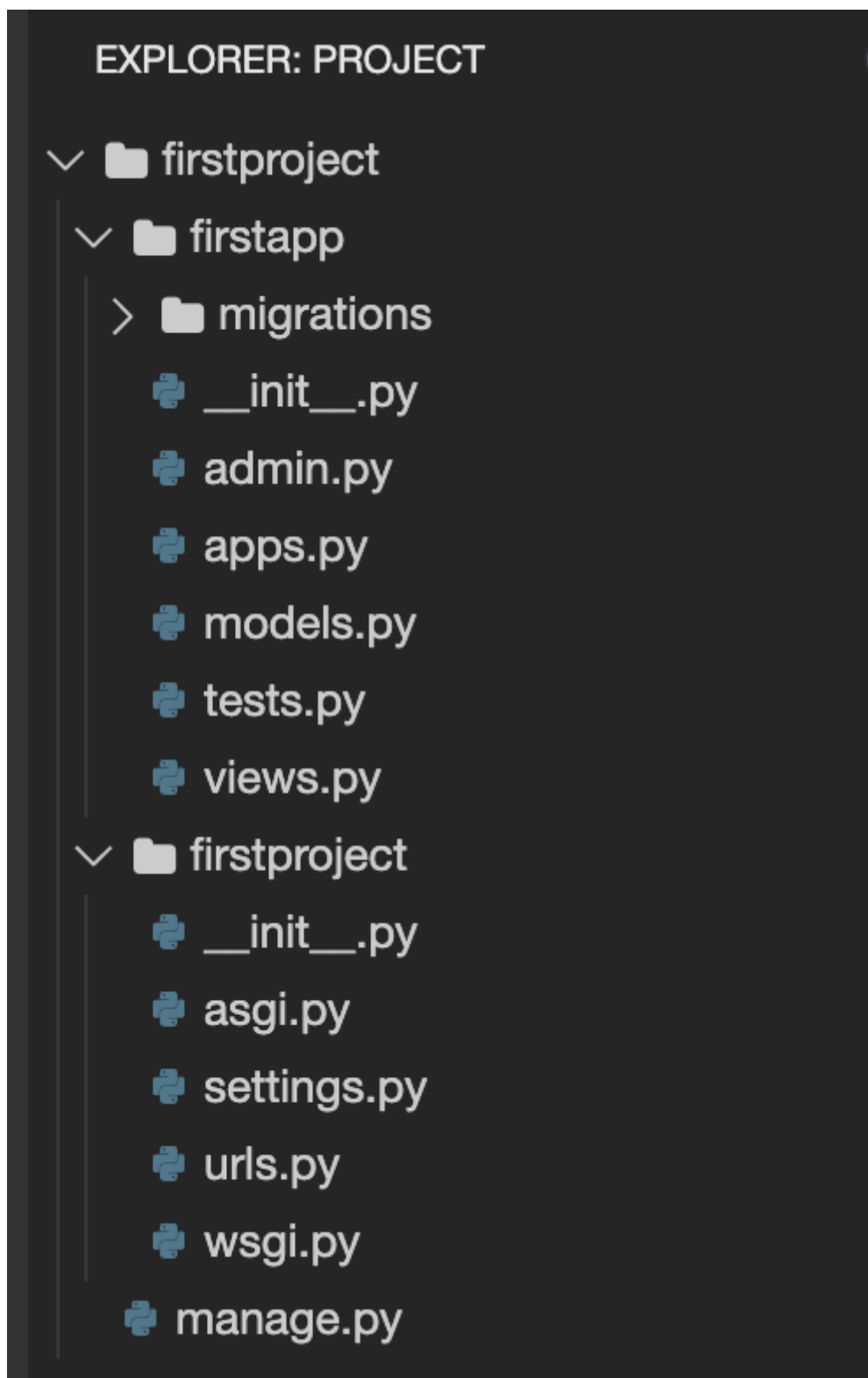
```
cd firstproject
```

- and create a Django app called `firstapp` within the project

```
python3 manage.py startapp firstapp
```

Django created a project scaffold for you containing your first `firstapp` app.

Your Theia workspace should look like the following:



The scaffold contains the fundamental configuration and setting files for a Django project and app:

- For project-related files:
 - `manage.py` is a command-line interface used to interact with the Django project to perform tasks such as starting the server, migrating models, and so on.
 - `firstproject/settings.py` contains the settings and configurations information.
 - `firstproject/urls.py` contains the URL and route definitions of your Django apps within the project.
- For app-related files:
 - `firstapp/admin.py`: is for building an admin site
 - `firstapp/models.py`: contains model classes
 - `firstapp/views.py`: contains view functions/classes
 - `firstapp/urls.py`: contains URL declarations and routing for the app
 - `firstapp/apps.py`: contains configuration meta data for the app
 - `firstapp/migrations/`: model migration scripts folder
- Before starting the app, you will to perform migrations to create necessary database tables:

```
python3 manage.py makemigrations
```

- and run migration

```
python3 manage.py migrate
```

- Then start a development server hosting apps in the `firstproject`:

```
python3 manage.py runserver
```

To see your first Django app from Theia,

- Click **Launch Application** and enter the port for the development server **8000**

and you should see the following welcome page:

django

View [release notes](#) for Django 3.1



The install worked successfully! Congratulations!

You are seeing this page because **DEBUG=True** is in your settings file and you have not configured any URLs.

When developing Django apps, in most cases Django will automatically load the updated files and restart the development server. However, it might be safer to restart the server manually if you add/delete files in your project.

Let's try to stop the Django server now by pressing:

- **Control + C** or **Ctrl + C** in the terminal

Add Your First View

Next, let's include your **firstapp** into **firstproject**

- Open **firstproject/settings.py** file, find **INSTALLED_APPS** section, and add a new app

entry as

```
'firstapp.apps.FirstappConfig',
```

Your **INSTALLED_APPS** should look like the following

```
# Application definition

INSTALLED_APPS = [
    'firstapp.apps.FirstappConfig',
    'django.contrib.admin',
    'django.contrib.auth',
    'django.contrib.contenttypes',
    'django.contrib.sessions',
    'django.contrib.messages',
    'django.contrib.staticfiles',
]
```

You can also see some pre-installed Django apps such as `admin` for managing the Admin site, `auth` for authentication, etc.

Next, we need to add the `urls.py` of `firstapp` to `firstproject` so that views of `firstapp` can be properly routed.

- Create an empty `urls.py` under `firstapp` folder
- Open `firstproject/urls.py`, you can find a `path` function `from django.urls import path` has been already imported.

Now also import an `include` method from `django.urls` package:

```
from django.urls import include, path
```

- Then add a new `path` entry

```
path('firstapp/', include('firstapp.urls')),
```

Your `firstproject/urls.py` now should look like the following:

```
from django.contrib import admin
from django.urls import include, path

urlpatterns = [
    path('admin/', admin.site.urls),
    path('firstapp/', include('firstapp.urls')),
]
```

Now you can create your first view to receive `HttpRequest` and return a `HttpResponse` wrapping a simple HTML page as its content.

- Open `firstapp/views.py`, write your first view after the comment `# Create your views here.`

```
from django.http import HttpResponse

def index(request):
    # Create a simple html page as a string
    template = "<html>" \
        "This is your first view" \
        "</html>"
    # Return the template as content argument in HTTP response
    return HttpResponse(content=template)
```

Next, configure the URL for the index view.

- Open `firstapp/urls.py`, add the following code:

```
from django.urls import path
from . import views

urlpatterns = [
    # Create a path object defining the URL pattern to the index view
    path(route='', view=views.index, name='index'),
]
```

That's it, now let's test your first view.

- Run Django sever if not started:

```
python3 manage.py runserver
```

- Clicking `Launch Application` and enter `8000`.

After a new browser tab is opened, add a `/firstapp` path to your full URL, note that the `userid` is a place holder and should be replaced by your real id shown after clicking `Launch Application`

`https://userid-8000.theiadocker-1.proxy.cognitiveclass.ai/firstapp/`

Basically, Django will map any HTTP requests starting with `/firstapp` to `firstapp` and search any matches for paths defined in `firstapp/urls.py`.

That's it, you should see your the `HTTPResponse` returned by your first view, which is a simple HTML page with content `This is your first view`.

Coding Practice: Add a View to Return Current Date

Complete and add the following code snippet to create a view to returning today's date.

Add the a `get_date()` view function in `firstapp/views.py` (remember to save the updated file):

```
from datetime import date

def get_date(request):
    today = date.today()
    template = "<html>" \
        "Today's date is {}" \
        "</html>".format(#<HINT> add today here#)
    return HttpResponse(content=#<HINT> use the template object as argument value#)
```

and add a `/date` URL path to `firstapp/urls.py` for `get_date()` view:

```
urlpatterns = [
    # Create a path object defining the URL pattern to the index view
    path(route='', view=views.index, name='index'),
    # Add another path object defining the URL pattern using `/date` prefix
    path(route='#<HINT> add a date URL path here#', view=#add the get_date function name here #, name='date'),
]
```

► [Click here to see solution](#)

Summary

In this lab, you have created your first Django project and your first Django app. You have also created your first Django view to return a simple HTML page.

Author(s)

[Yan Luo]([linkedin.com/in/yan-luo-96288783](https://www.linkedin.com/in/yan-luo-96288783))

Changelog

Date	Version	Changed by	Change Description
30-Nov-2020	1.0	Yan Luo	Initial version created

© IBM Corporation 2020. All rights reserved.