**Working with Django Framework**

1. **Installation :**
2. Need Python environment to be installed first
3. Using pip:

python –m pip install Django

1. **Creating and Starting Django app**
2. From the command line, cd into a directory where you’d like to store your code, then run the following command:

**$** django-admin startproject mysite

1. A Start project will be created with the following directory structure

mysite/

manage.py

mysite/

\_\_init\_\_.py

settings.py

urls.py

asgi.py

wsgi.py

These files are:

The outer mysite/ root directory is a container for your project. Its name doesn’t matter to Django; you can rename it to anything you like.

manage.py: A command-line utility that lets you interact with this Django project in various ways. You can read all the details about manage.py in django-admin and manage.py.

The inner mysite/ directory is the actual Python package for your project. Its name is the Python package name you’ll need to use to import anything inside it (e.g. mysite.urls).

mysite/\_\_init\_\_.py: An empty file that tells Python that this directory should be considered a Python package. If you’re a Python beginner, read more about packages in the official Python docs.

mysite/settings.py: Settings/configuration for this Django project. Django settings will tell you all about how settings work.

mysite/urls.py: The URL declarations for this Django project; a “table of contents” of your Django-powered site. You can read more about URLs in URL dispatcher.

mysite/asgi.py: An entry-point for ASGI-compatible web servers to serve your project. See How to deploy with ASGI for more details.

mysite/wsgi.py: An entry-point for WSGI-compatible web servers to serve your project. See How to deploy with WSGI for more details.

1. The deployment Server

Let’s verify your Django project works. Change into the outer mysite directory, if you haven’t already, and run the following commands:

$ python manage.py runserver

You’ll see the following output on the command line:

Performing system checks...

System check identified no issues (0 silenced).

You have unapplied migrations; your app may not work properly until they are applied.

Run 'python manage.py migrate' to apply them.

Django version 3.1, using settings 'mysite.settings'

Starting development server at http://127.0.0.1:8000/

Quit the server with CONTROL-C.

1. **Creating a poll app**

* Now that your environment – a “project” – is set up, you’re set to start doing work.
* Each application you write in Django consists of a Python package that follows a certain convention.
* Django comes with a utility that automatically generates the basic directory structure of an app, so you can focus on writing code rather than creating directories.
* To create your app, make sure you’re in the same directory as manage.py and type this command.

**$** python manage.py startapp polls

That’ll create directory polls, which are laid out like this:

polls/

\_\_init\_\_.py

admin.py

apps.py

migrations/

\_\_init\_\_.py

models.py

tests.py

views.py

This directory structure will house the poll application.

**Writing a first view**

Let’s write the first view. Open the file polls/views.py and put the following Python code in it:

**from** **django.http** **import** HttpResponse

**def** index(request):

**return** HttpResponse("Hello, world. You're at the polls index.")

**polls/urls.py**

**from** **django.urls** **import** path

**from** **.** **import** views

urlpatterns = [

path('', views.index, name='index'),

]

**mysite/urls.py**

**from** **django.contrib** **import** admin

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

urlpatterns = [

path('polls/', include('polls.urls')),

path('admin/', admin.site.urls),

]

The idea behind include() is to make it easy to plug-and-play URLs. Since polls are in their own URLconf (polls/urls.py), they can be placed under “/polls/”, or under “/fun\_polls/”, or under “/content/polls/”, or any other path root, and the app will still work.

**$** python manage.py runserver

If you get an error page here, check that you’re going to [http://localhost:8000/polls/](http://localhost:8000/polls/%20) and not <http://localhost:8000/>.