$ python -m venv djangoenv – to create djangoenv if this does not exist.

Or conda create --name <myenv>

$ conda env list

$ conda activate <env name>

To deactivate an environment, type:

$ conda deactivate

$ **python -m django --version**

# Django-admin – to see if Django is installed correctly and show all its commands.

$django-admin startproject <name of project> 🡺 To Start a Django project. Project has Setting.py.

I created Django project as FirstDJangoProject and it automatically creates following in project directory - **FirstDjangoProject**

Text

Description automatically generated

Now go to IDE – pycharm and open up this project and you will see following folders and files in that:

Text

Description automatically generated

You should not edit manage.py, init.py and wsgi.py file right now.

**Now Run the local server by following command:**

**Go to pycharm – Terminal which is your project inside terminal and then run following -**

$ python manage.py runserver 🡺 To run the Django server

Text

Description automatically generated

**First page of your running server you get here:**

Graphical user interface, text

Description automatically generated

**You can terminate server using Ctrl+Break**

Please note that running the server first time will create a new file and visible on pycharm:

Graphical user interface, text, website

Description automatically generated

You can ignore this right now as this nothing but used for sqlite db.

$ python manage.py 🡺 List all the functions which can be performed on Django Available subcommands.

**All the project work in Django project mainly happen in different apps. In order to create Django applications. First you need to quit \ exit from running server and then run the following command.**

$ python manage.py startapp <name of app> 🡺 to create an app of a project. But app do not have urls.py and setting.py but it has admin.py which is not in project.

$ python manage.py startapp challenges



The above folder will get created. This folder will contain 6 files and a migrations sub folder–

Text

Description automatically generated

Apps are building block for the overall project. App = Module.

Graphical user interface

Description automatically generated

$ python manage.py makemigration

**URLS.py – are action mappings with HTML which ensure that certain results are “achieved” when certain URLs are entered by the user.**

**Views.py are – actual logic which gives actual actions for those URLs.**

My-page.com/ 🡺 Show starting page if this is mentioned in URLS and Actual actions are mentioned in Views.py

My-page.com/posts 🡺 Show list of all post.

A picture containing graphical user interface

Description automatically generated

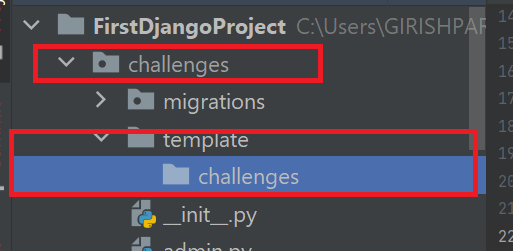
<https://consideratecode.com/2018/05/11/the-hidden-powers-of-custom-django-2-0-path-converters/>

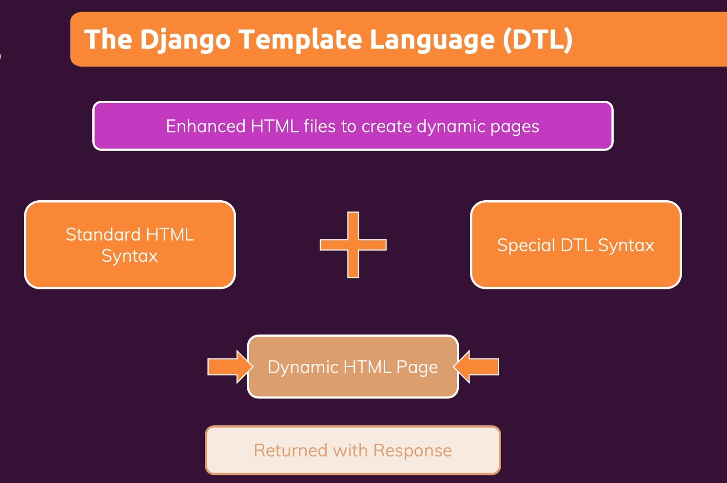
<https://consideratecode.com/2018/05/02/django-2-0-url-to-path-cheatsheet/>

Diagram

Description automatically generated

**Templates:** In order to keep Static files which then contains Html\ CSS\ Java script etc. you should create template withing the Django App you just created – challenges. The convention is to create a template folder and then <app name> folder within the template folder. Look at screen shot:

****

****

Built-in template tags and filters

**https://docs.djangoproject.com/en/3.2/ref/templates/builtins/**