Docker

* Django it has interfaces for models, views, and templates:
* https://docs.djangoproject.com/en/4.1/intro/tutorial01/

## Initialize

* In the directory run the command:

django-admin startproject mysite

* This creates a set of files such as
  + manage.py: A command-line utility that lets you interact with this Django project in various ways.

## Running the development server

py manage.py runserver 8080

## Writing app

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.

* A project can have multiple apps
* Django apps are “pluggable”
* top-level module => in the same dir as manage.py
* Create app:

py manage.py startapp polls

* What it creates:

polls/

\_\_init\_\_.py

admin.py

apps.py

migrations/

\_\_init\_\_.py

models.py

tests.py

views.py

## Writing views

* In the view file:

from django.http import HttpResponse

def index(request):

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

* Create a file in poll, **urls.py** => we need this to map the view to a url, put the following code inside it:

from django.urls import path

from . import views

urlpatterns = [

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

]

* Mysite.url (the root project) should look like

from django.contrib import admin

from django.urls import include, path

urlpatterns = [

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

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

]

## Database setup

* There are some default apps installed in setting.py ( a module file), to install their databases:

py manage.py migrate

* Models => to update based on changes (migrate). Migrations are how Django stores changes to your models (and thus your database schema)

from django.db import models

class Question(models.Model):

question\_text = models.CharField(max\_length=200)

pub\_date = models.DateTimeField('date published')

class Choice(models.Model):

question = models.ForeignKey(Question, on\_delete=models.CASCADE)

choice\_text = models.CharField(max\_length=200)

votes = models.IntegerField(default=0)

* Add the class 'polls.apps.PollsConfig' of the app to the setting.py, INSTALLED\_APPS
* Run the command

//separated these two commands for version control purposes. To store it in a file

py manage.py makemigrations polls

py manage.py migrate