# Django level one

## Create Django virtual env by running following commands:

$conda create –name myDjangoEnv django

$conda activate myDjangoEnv (the command does not work in powershell, open cmn prompt in powershell by typing cmd and run the line)

$conda deactivate (after finishing work)

## Install django-admin and start project by running following command:

$conda activate myDjangoEnv

$pip install django-admin

$django-admin startproject first\_project

## Run server:

$cd first\_project

$python manage.py runserver

## Start a Django app:

1. $python manage.py startapp first\_app
2. Add following lines in views.py:

*from django.http import HttpResponse*

*def index(request):*

*return HttpResponse(“Hello World!”)*

1. Link views.py to urls.py by adding following item in urlpatterns list:

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

and add following import in urls.py-

*from first\_app import views*

1. In settings.py add following item in INSTALLED\_APPS list:

“first\_app.apps.FirstAppConfig”

## URL mappings:

1. In url.py file add following item in urlpatters list:

*path('first\_app/', include('first\_app.urls'))*

And add following import in urls.py

*from django.urls import path, include*

1. Create a file named urls.py in first\_app directory and add following lines the file:

*from django.urls import path*

*from . import views*

*urlpatterns = [*

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

*]*

## Templates:

1. Create templates directory in app directory.
2. In templates directory create index.html file and add following lines:

*<!DOCTYPE html>*

*<html lang="en" dir="ltr">*

*<head>*

*<meta charset="utf-8">*

*<title>Templates</title>*

*</head>*

*<body>*

*<h1>Templates Example!</h1>*

*<!-- Template tag -->*

*{{insert\_me}}*

*</body>*

*</html>*

1. Add following lines in views.py:

*def index2(request):*

*my\_dict = {'insert\_me': 'Hello, I am from views.py'}*

*return render(request, 'index.html', context=my\_dict)*

1. Change urls.py accordingly.

## Static Files:

Template tags: simple- {{ }}, for more complex operation- {% %}

1. Create static directory into app directory.
2. Create images directory into static directory and add an image file (sea.jpg)
3. Create template file in templates directory (image.html) and add following lines:

*<!DOCTYPE html>*

*{% load static %}*

*<html lang="en" dir="ltr">*

*<head>*

*<meta charset="utf-8">*

*<title>SEA IMAGE</title>*

*</head>*

*<body>*

*<h1>Sea</h1>*

*<img src="{% static "images/sea.jpg" %}" alt="Uh Oh, didn't show!">*

*</body>*

*</html>*

1. Change views.py and urls.py files accordingly.

# Django level two

## Models:

To incorporate a database into a Django project.

1. Create first\_project and first\_app. Create template and do url mapping.
2. Add following lines in models.py file:

*class Topic(models.Model):*

*self.top\_name = models.CharField(max\_length=264, unique=True)*

*def \_\_str\_\_(self):*

*return self.top\_name*

*class WebPage(models.Model):*

*topic = models.ForeignKey(Topic, on\_delete=models.CASCADE)*

*name = models.CharField(max\_length=264, unique=True)*

*url = models.URLField(unique=True)*

*def \_\_str\_\_(self):*

*return self.name*

*class AccessRecord(models.Model):*

*name = models.ForeignKey(WebPage, on\_delete=models.CASCADE)*

*date = models.DateField()*

*def \_\_str\_\_(self):*

*return str(self.date)*

1. In project directory run following commands:

$python manage.py migrate

$python manage.py makemigrations first\_app

$python manage.py migrate

1. Test whether migration works. Run the command *$python manage.py shell* to open python shell and test with following commands:

*>> from first\_app.models import Topic*

*>> print(Topic.objects.all())*

*>> t = Topic(top\_name=”Social Network”)*

*>> t.save()*

*>> print(Topic.objects.all())*

1. Register models in admin interface. Add following lines in admin.py file:

*from first\_app.models import Topic, WebPage, AccessRecord*

*admin.site.register(Topic)*

*admin.site.register(WebPage)*

*admin.site.register(AccessRecord)*

1. Create superuser by running following command:

$python manage.py createsuperuser

1. Run the server and go to admin interface in <http://127.0.0.1:8000/admin>

## Population scripts:

1. Install library:

$pip install Faker

1. Create populate\_first\_app.py in project directory and add following lines:

*import os*

*# Configure settings for project*

*# Need to run this before calling models from application!*

*os.environ.setdefault('DJANGO\_SETTINGS\_MODULE','first\_project.settings')*

*import django*

*# Import settings*

*django.setup()*

*import random*

*from first\_app.models import Topic,WebPage,AccessRecord*

*from faker import Faker*

*fakegen = Faker()*

*topics = ['Search','Social','Marketplace','News','Games']*

*def add\_topic():*

*t = Topic.objects.get\_or\_create(top\_name=random.choice(topics))[0]*

*t.save()*

*return t*

*def populate(N=5):*

*'''*

*Create N Entries of Dates Accessed*

*'''*

*for entry in range(N):*

*# Get Topic for Entry*

*top = add\_topic()*

*# Create Fake Data for entry*

*fake\_url = fakegen.url()*

*fake\_date = fakegen.date()*

*fake\_name = fakegen.company()*

*# Create new Webpage Entry*

*webpg = WebPage.objects.get\_or\_create(topic=top,url=fake\_url,name=fake\_name)[0]*

*# Create Fake Access Record for that page*

*# Could add more of these if you wanted...*

*accRec = AccessRecord.objects.get\_or\_create(name=webpg,date=fake\_date)[0]*

*if \_\_name\_\_ == '\_\_main\_\_':*

*print("Populating the databases...Please Wait")*

*populate(20)*

*print('Populating Complete')*

1. Populate data:

$python populate\_first\_app.py

## Models-Templates-Views (MTV) paradigm

The idea of how to connect everything - models, templates and views. Steps:

1. In the views.py file we import any models that we will need to use.
2. Use the view to query the model for data that we will need.
3. Pass results from the model to the template.
4. Map a URL to the view.

## Template tags