4th April lecture wasn’t recorded sir

5th git lecture by Emmanuel.

**6th April lecture by Jayson. Notes below**

Virtual environment: A virtual environment is a Python environment such that the Python interpreter, libraries and scripts installed into it are isolated from those installed in other virtual environments, and (by default) any libraries installed in a “system” Python, i.e., one which is installed as part of your operating system.

URL: Uniform Resource Locator

HTTP or HTTPS: same mother but diff fathers. It is the way web pages move around over the internet. The http or https makes it possible to navigate through web pages. The s stands for a secure site

<https://docs.djangoproject.com/en/1.8/intro/tutorial01/> -- registering your Django app

include ‘apps’, after the list of INSTALLED\_APPS = in your settings.py from your created app in VS code. Don’t forget to put that comma.

Template in Django: Templates contain html files only. The other folder is called static, where we hace css and javascript files.

Both your static and template should be in same directory as your manage.py

Create a folder and call it Templates and then go to settings.py and add “Templates” in front of DIRS:

OUR CREATED demo folder is the base dir

Hypertext transfer protocol

Hypertext markup language: it is the skeleton of every webpage using tags. It is not a programming language

Create a model that will ask for my name:

Django works on the basis of what we call MVT – Model view template

M-

V: Decide what you see and order

T: Gives you what you see – frontend

#ORM: Object Relational mapper

CBV: Class Based View: under this, we have different kind of views. The view itself, APIs, generic views, template views.

Function comes under/after class. When a function is inside a class, it is no longer a function, it is a method. If it is standing alone, it is a called a function in Python. It is called a function-based view in Django. They give us the permission to manipulate data on our own

Templates work together with url

Authentication: the ability to log in log out, create a new user, reset your password

Authorisation: Ability to limit your user to some certain things e.g.. if you are not a super user, you can’t have access to the Django dashboard

Models where you write a python object that will turn into table in python

**USE CLS to clear your command line in windows**

Create superuser from your command line to have access to your dashboard using python manage.py createsuperuser

A screenshot of a computer

Description automatically generated with medium confidence

**To get our browser running, type**

python manage.py runserver in your cmd or vscode terminal

**This will be the result**

A screenshot of a computer

Description automatically generated with medium confidence

**Then go ahead and type the url for our server as shown on the image** *127.0.0.1:8000*

Typing that on our browser will give this

A screenshot of a computer

Description automatically generated with medium confidence

Add admin to the url i.e. ***127.0.0.1:8000/admin*** to open the administration log in page

Graphical user interface, application

Description automatically generated

Use the superuser credential created in cmd to log in. After logging in our interface will look like:

Graphical user interface, application

Description automatically generated

This was our model’s name. It has a space because of how it was written under verbose\_name\_plural



Everything on that admin page is from our code. Clicking on Todo Data 🡪 Add todo data, it will return some fields like this

A screenshot of a computer

Description automatically generated with medium confidence

Every attribute on the right hand side (e.g. Name of task, Task, etc) are from the modelwe created in models.py. The attributes are appearing same way we specify

Creating a model is what we turned into a database logic

A dashboard comes with Django by default

Register your model in admin.py. By registering it, it will appear in your dashboard

Every webpage can either be a function or a class or we can decide our webpage under a particular function