**Django Start Project -> Database**

1. Install Django

>>> pip install django==2.2

1. Start project

>>> django-admin startproject DBProject

1. Location change

>>> cd DBProject

1. Run Server   
   >>> python manage.py runserver
2. Create APP

>>> python manage.py startapp A2

1. Create template dir
2. Create a template director inside the main project (DBProject)
3. Main app -> Settings.py -> TEMPLATES -> '**’DIRS': [‘templates’]**

7. Create an HTML page inside the templates dir

8. Create a views.py file in main app (DBProject)

9. Inside DBProject->views.py write the following codes (exactly)

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

**from** django.shortcuts **import** render

**def** home(request):

**return** render(request, **'home.html'**)

10. Inside DBProject->urls.py import views.py (add the following code)

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

12. Inside DBProject->urls.py connect home function with a path (add the following code)

path(**''**, views.home),

13. Run server to open the home page that you have created

>>> python manage.py runserver

14. Create a student model inside the A2->models.py (use the following code)

**class** Student(models.Model):

name = models.CharField(max\_length=100, default=**""**)

student\_ID = models.IntegerField(default=0)

email\_address = models.CharField(max\_length=500, default=**""**)

15. To Install APP: Go to DBProject->settings.py->INSTALLED\_APPS = [

**'A2.apps.A2Config'**,

16. Create table in Database (in terminal). To turn of the server use Ctrl+C. Then write the following command.

>>> python manage.py makemigrations A2

Output:

Migrations for 'A2':

A2\migrations\0001\_initial.py

- Create model Student

17. Migrate the project to reflect the change

>>> python manage.py migrate

Output:

Operations to perform:

Apply all migrations: A2, admin, auth, contenttypes, sessions

Running migrations:

Applying A2.0001\_initial... OK

Applying contenttypes.0001\_initial... OK

Applying auth.0001\_initial... OK

Applying admin.0001\_initial... OK

Applying admin.0002\_logentry\_remove\_auto\_add... OK

Applying admin.0003\_logentry\_add\_action\_flag\_choices... OK

Applying contenttypes.0002\_remove\_content\_type\_name... OK

Applying auth.0002\_alter\_permission\_name\_max\_length... OK

Applying auth.0003\_alter\_user\_email\_max\_length... OK

Applying auth.0004\_alter\_user\_username\_opts... OK

Applying auth.0005\_alter\_user\_last\_login\_null... OK

Applying auth.0006\_require\_contenttypes\_0002... OK

Applying auth.0007\_alter\_validators\_add\_error\_messages... OK

Applying auth.0008\_alter\_user\_username\_max\_length... OK

Applying auth.0009\_alter\_user\_last\_name\_max\_length... OK

Applying auth.0010\_alter\_group\_name\_max\_length... OK

Applying auth.0011\_update\_proxy\_permissions... OK

Applying sessions.0001\_initial... OK

18. Create a super user

>>> python manage.py createsuperuser

**Username (leave blank to use 'lab5projector'): tsr**

**Email address: tsr@gmail.com**

**Password:**

**Password (again):**

**The password is too similar to the username.**

**This password is too short. It must contain at least 8 characters.**

**Bypass password validation and create user anyway? [y/N]: y**

**Superuser created successfully.**

19. After running the server go to Browser: /admin

20. Use your username and password to go inside the admin panel

21. Register your model in Admin. Go to **A2->admin.py** and add the following codes

**from** . models **import** Student

admin.site.register(Student)

22. Now runserver and go to the admin panel again. You will see the Student table there.

23. Add some students

24. Add a method in the models.py->Student

**class** Student(models.Model):

name = models.CharField(max\_length=100, default=**""**)

student\_ID = models.IntegerField(default=0)

email\_address = models.CharField(max\_length=500, default=**""**)

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

**return** self.name

25. Create a students.html in templates dir

26. DBProject -> views.py add the following code

**def** showStudent(request):

**return** render(request, **'students.html'**)

27. DBProject -> urls.py add a new path like the following

path(**'students/'**, views.showStudent),

28. Check the students page from browser after running the server

29. Import Student model into the main views.py

**from** A2.models **import** Student

30. Search the table and print all the students in terminal to check

**def** showStudent(request):

allStudents = Student.objects.all()

print(allStudents)

**return** render(request, **'students.html'**)

31. Run server -> go to students page -> check terminal. You should see a list of all students in the terminal like the following

<QuerySet [<Student: Tanmoy Sarkar Pias>, <Student: Afia>, <Student: Mim>, <Student: shakil>]>

32. Create a context dictionary

context = { **"allStudents"** : allStudents}

33. Pass the context with the html like the following

**return** render(request, **'students.html'**, context)

34. Inside the **students.html** page add the following code to print the students table

{% for s in allStudents %}

{{s.name}} <**br**>

{{s.student\_ID}} <**br**>

{{s.email\_address}} <**br**>

<**br**>

{% endfor %}

35. Run server -> go to students page



The end !!!!