

Myenv

requirement.txt

src

clarusway

fscohort

views.py

```
from django.http import HttpResponse
```

```
def home_view(request):  
    return HttpResponse("<h1> Hello World </h1>")
```

View oluşturuyoruz
ve
url path tanımlıyoruz

App'in view'ı

Myenv

requirement.txt

src

clarusway



urls.py

fscohort

View oluşturuyoruz
ve
url path tanımlıyoruz

```
from django.urls import path, include (include ekliyoruz)  
path("home/",include("fscohort.urls"))
```

/home yazınca
fscohort url'sine git

Myenv

requirement.txt

src

clarusway

urls.py

fscohort

urls.py

views.py

```
from django.urls import path
from . import views
```

```
urlpatterns = [
    path("", views.home_view, name="home_view")
]
```

Daha önce oluşturduğumuz
«home_view»

View oluşturuyoruz
ve
url path tanımlıyoruz

Myenv

requirement.txt

src

clarusway

settings.py

fscohort

```
INSTALLED_APPS = [  
    ...  
    'fscohort.apps.FscohortConfig'  
    # sadece 'fscohort' da yazabiliriz
```

View oluşturunuz
ve
url path tanımlıyoruz

Yeni bir App'i tanımladık

Myenv

requirement.txt

src

clarusway

fscohort

TERMINAL

py manage.py runserver

View oluşturuyoruz
ve
url path tanımlıyoruz

Myenv

requirement.txt

src

clarusway

fscohort

TERMINAL

py manage.py migrate

database'imizde
-user ve -groups
diye 2 tane table
oluştururuz


Myenv

requirement.txt

src

clarusway

fscohort

 models.py

```
class Student(models.Model):  
    first_name = models.CharField(max_length=50)  
    last_name=models.CharField(max_length=50)  
    number=models.IntegerField()
```

Database için
Model (Tablo)
oluşturalım!!

model field'larını
otomatik oluşturulmuş
django dokümanlarından
bakabilirim.

Myenv

requirement.txt

src

clarusway

fscohort

TERMINAL

py manage.py makemigrations

py manage.py migrate

Database için
Model (Tablo)
oluşturalım!!

django' ya ben bir model
oluşturdum diye deklere
etmem gerekiyor

otomatik
oluşturduğumuz alanları
database'e atmak için

Myenv

requirement.txt

src

clarusway

fscohort

artık ADMIN olarak
işlem yapabiliriz!!

TERMINAL

py manage.py createsuperuser

"username", "email", "password"



127.0.0.1:8000/admin/


Myenv

requirement.txt

src

clarusway

fscohort

 admin.py

```
from .models import Student  
admin.site.register(Students)
```

oluşturduğumuz
model'i
admin panelinde
göstermek istiyoruz


Myenv

requirement.txt

src

clarusway

fscohort

 models.py

```
def __str__(self):  
    return self.first_name
```

??? database'e
kaydettiğimiz her bir
elemanın panelde
hangi field ismi
görüntülenmesini
istersiniz??

Django administration

Site administration

AUTHENTICATION AND AUTHORIZATION

Groups [+ Add](#) [Change](#)

Users [+ Add](#) [Change](#)

FSCOHORT

Students [+ Add](#) [Change](#)

Recent actions

My actions

+ Student object (1)
Student

Artık ADMIN
panelinde
manipölasyon
yapabiliriz!!!!

Django administration

WELCOME, ALIF. [VIEW SITE](#) / [CHANGE PASSWORD](#) / [LOG OUT](#)

Home » Fscohort » Students

AUTHENTICATION AND AUTHORIZATION

Groups [+ Add](#)

Users [+ Add](#)

FSCOHORT

Students [+ Add](#)

Select student to change

Action: 0 of 3 selected

☐ STUDENT

☐ Student object (3)

☐ Student object (2)

☐ Student object (1)

3 students

ADD STUDENT +

Model'e yeni elemanlar
girebiliriz

TERMINAL

```
py manage.py shell
```

```
>>> from fscohort.models import Student
```

```
>>> s1.save()
```

```
>>> s1 = Student(first_name="veli", last_name="asdf", number=122)
```

```
>>> all_s= Student.objects.all()
```

```
>>> s4= Student.objects.get(id=2)
```

```
>>> Student.objects.filter(number=2)
```

```
>>> Student.objects.exclude(first_name="veli")
```

```
>>> s3 = Student.objects.create(first_name="ayşe", last_name="assf", number=112)
```

```
>>> f1 = Student.objects.filter(first_name__startswith="C")
```

```
>>> f1 = Student.objects.filter(first_name__contains="a")
```

**database'de Query yapacağız;
database'den veri alıp
template'e göndereceğiz;**

Shell'den Model'e yönelik çeşitli manipülasyon ve query komutları