Django Admin panel

- The Django admin panel is a built-in, user-friendly interface that allows developers to manage the data of a Django project effortlessly. It provides a convenient way to perform common tasks such as creating, reading, updating, and deleting data in your database without having to write custom views or templates.
- To enter the admin user interface, In the browser window, type 127.0.0.1:8000/admin/ in the address bar.
- The reason why this URL goes to the Django admin login page can be found in the urls.py file of your project.
- To be able to log into the admin application, we need to create a superuser. This is done by typing this command in the terminal.

```
python manage.py createsuperuser
```

- And fill in the login page form with the correct username and password.
- In this admin interface can create, read, update, and delete data.
- The next step is to define models(class). Models define the structure of your database tables.
- Open the "models.py" file within your app and define your models:

```
class student(models.Model):
    name = models.CharField(max_length=50)
    age = models.IntegerFiels()
```

- Now we need to register our models with the admin panel. Open the "admin.py" file within your app and register your models:

```
admin.site.register(student)
```

- In admin user interface when click on a table name data cannot see instead some objects are displayed. Django displays each record as the object. To change this to a more reader-friendly format, in ‘admin.py’

```
class admstd(admin.ModelAdmin):
        list_display = (‘name’,’age’)
admin.site.register(student, admstd)
```