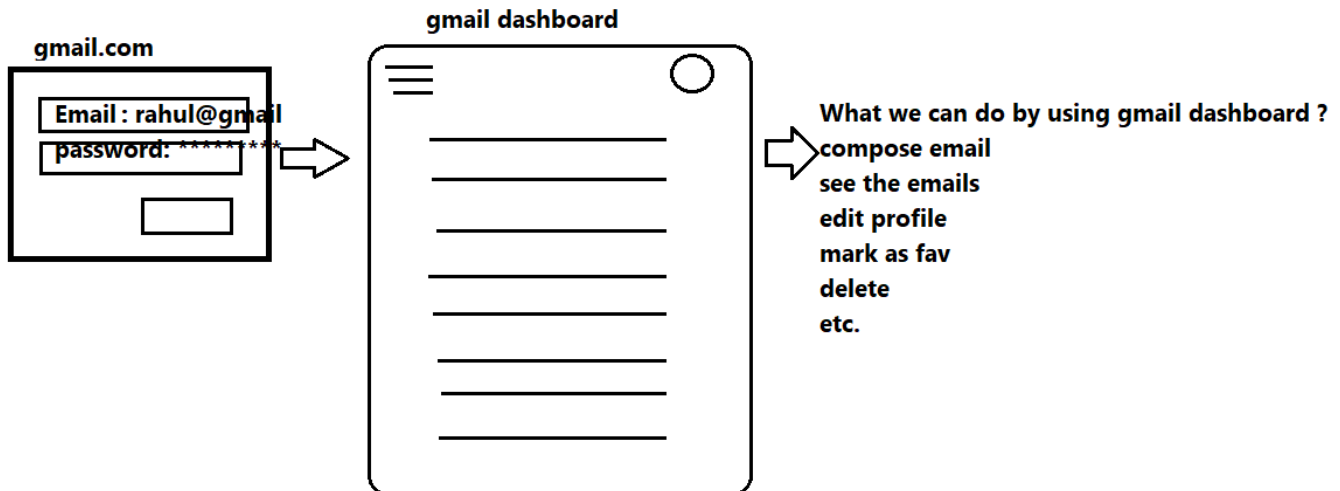


Day: 11

Date: 25/05/2024

Django Admin Interface / Inbuild dashboard in Django:



Dashboard: It is webpage/profile page where a user/admin can perform various operations like add/delete/update/create etc.

Django Admin Interface :

- ❖ It is also called Django admin dashboard.
- ❖ By default, django provide this dashboard for the admin to perform various operations.

What are the different operation we can perform using django admin interface?

- ❖ Insert records.
- ❖ Delete records.
- ❖ Update records.
- ❖ Give privileged.
- ❖ Create users
- ❖ Deletes users etc.

How to create django admin interface in django?

- ❖ Already django created django admin interface.

How to use the django admin interface?

- ❖ Execute makemigrations command. (py mange.py makemigrations)
- ❖ Execute migrate command. (py manage.py migrate)

- ❖ Create superuser(admin) :(py manage.py createsuperuser). Provide userid and password.
- ❖ Open browser and send HTTP request (127.0.0.1:8000/admin), then login page will open the provide superuser user id and password.
- ❖ If the login is success, then it will automatically redirect django admin interface.

P010

Problem Statement: Create a model, create superuser for django admin interface, add records through django admin interface, then display those records in the django admin interface.

All Steps

Step 1: Common steps (Folder, project, app creation then install the application.)

Step 2: Database configuration.

```
DATABASES = {  
    'default': {  
        'ENGINE': 'django.db.backends.mysql',  
        'NAME': 'product',  
        'USER': 'root',  
        'PASSWORD': 'root_db',  
    }  
}
```

Step 3: Define a model inside models.py file.

```
from django.db import models  
  
# Create your models here.  
class Electronics(models.Model):  
    name=models.CharField(max_length=20)  
    brand=models.CharField(max_length=20)  
    price=models.FloatField()
```

Step 4: Convert Model class into SQL specific code.

py manage.py makemigrations

Migrations for 'DAIAPP':

DAIAPP\migrations\0001_initial.py
- Create model Electronics

Step 5: Execute generated SQL code.

```
py manage.py migrate
```

Operations to perform:

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

Running migrations:

Applying DAIAPP.0001_initial... OK

Step 6: Open terminal/cmd then create superuser.

```
py manage.py createsuperuser
```

Username (leave blank to use 'user'): dj27

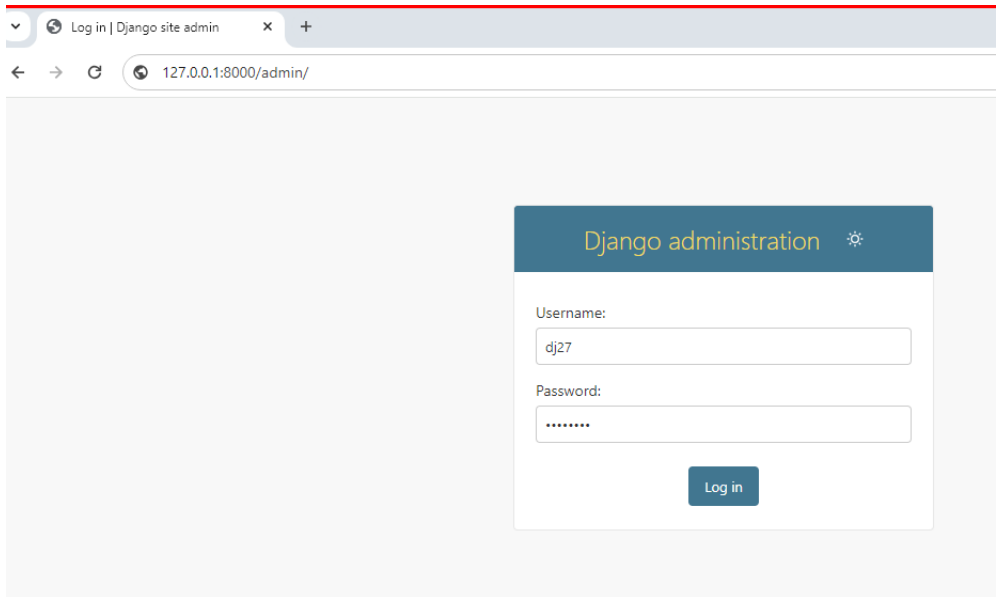
Email address: dj27@gmail.com

Password: dj102423

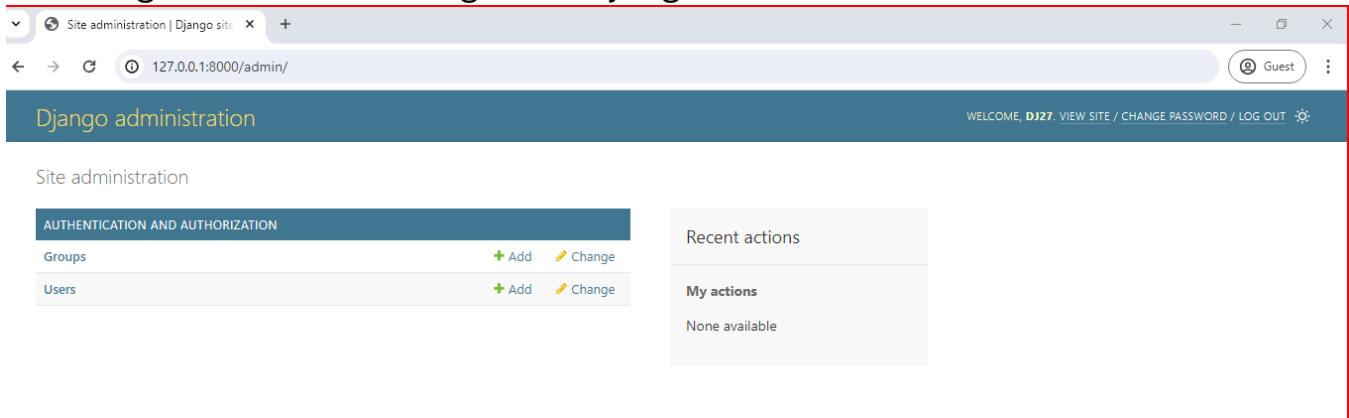
Password (again): dj102423

Superuser created successfully.

Step 7: Runserver, Open browser then send http request (127.0.0.1:8000/admin), provide user id and password.



Once login success we will get this django dashboard



Step 8: Now admin can perform various operation.

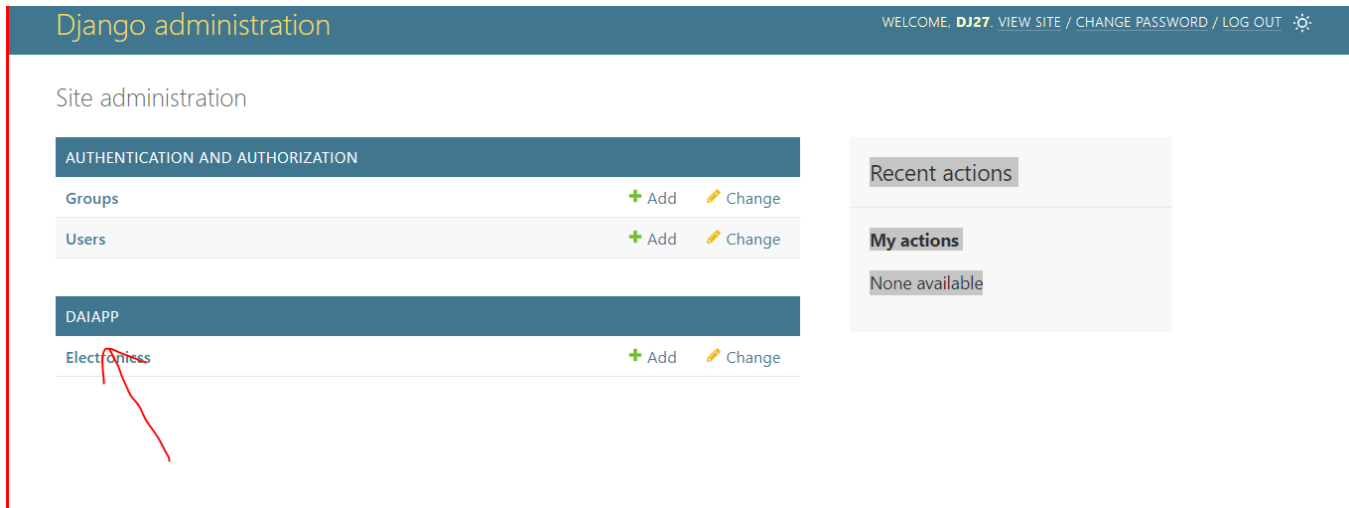
Target : add records into the table through django admin interface.

Register our model class inside admin.py file.

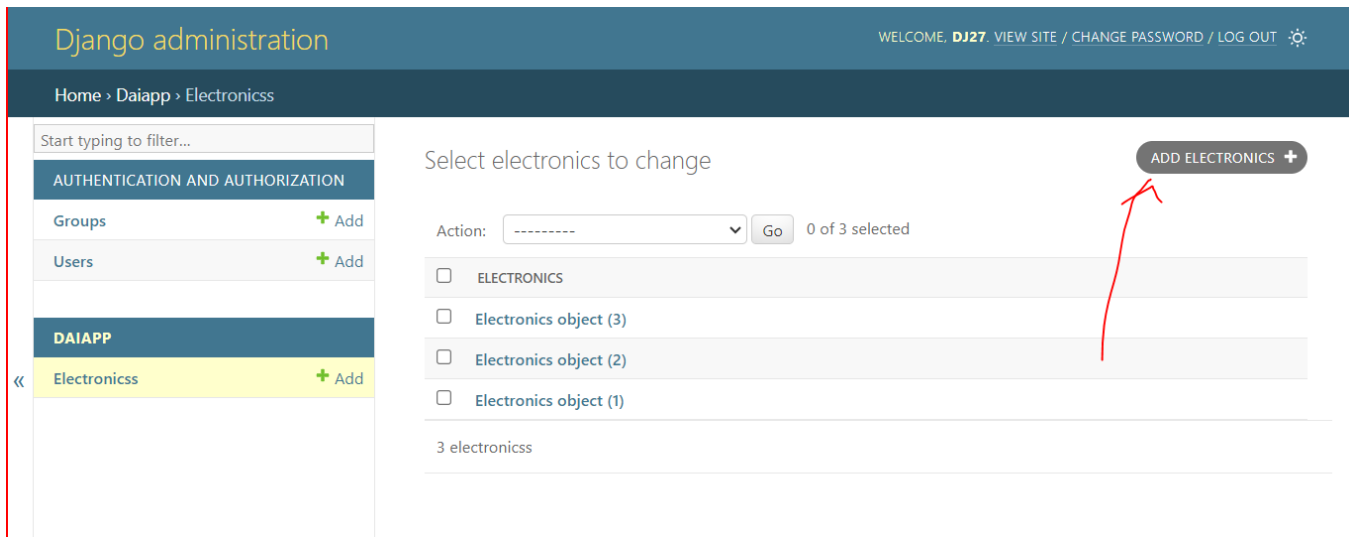
admin.py

```
from django.contrib import admin
from .models import Electronics
# Register your models here.

admin.site.register(Electronics)
```



Add electronics product through django admin interface



```
mysql> select * from daiapp_electronics;
+----+-----+-----+-----+
| id | name      | brand | price |
+----+-----+-----+-----+
| 1  | Fan       | USHA  | 2399  |
| 2  | Watch     | Titan | 6900  |
| 3  | Water Heater | Kent  | 2100  |
+----+-----+-----+-----+
3 rows in set (0.00 sec)
```

Default product details for each record

Select electronics to change

Action: 0 of 3 selected

<input type="checkbox"/>	ELECTRONICS
<input type="checkbox"/>	Electronics object (4)
<input type="checkbox"/>	Electronics object (3)
<input type="checkbox"/>	Electronics object (1)

3 electronicss

We can provide good name to all the product records

```
from django.db import models

# Create your models here.
class Electronics(models.Model):
    name=models.CharField(max_length=20)
    brand=models.CharField(max_length=20)
    price=models.FloatField()

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

`__str__()` is a magic method used to return the string representation of an object.

Select electronics to change

Action: 0 of 3 selected

<input type="checkbox"/>	ELECTRONICS
<input type="checkbox"/>	Mobile
<input type="checkbox"/>	Water Heater
<input type="checkbox"/>	Fan

3 electronicss