Create a DJANGO application to perform CRUD operations in Django admin.

Requirement/Tools:

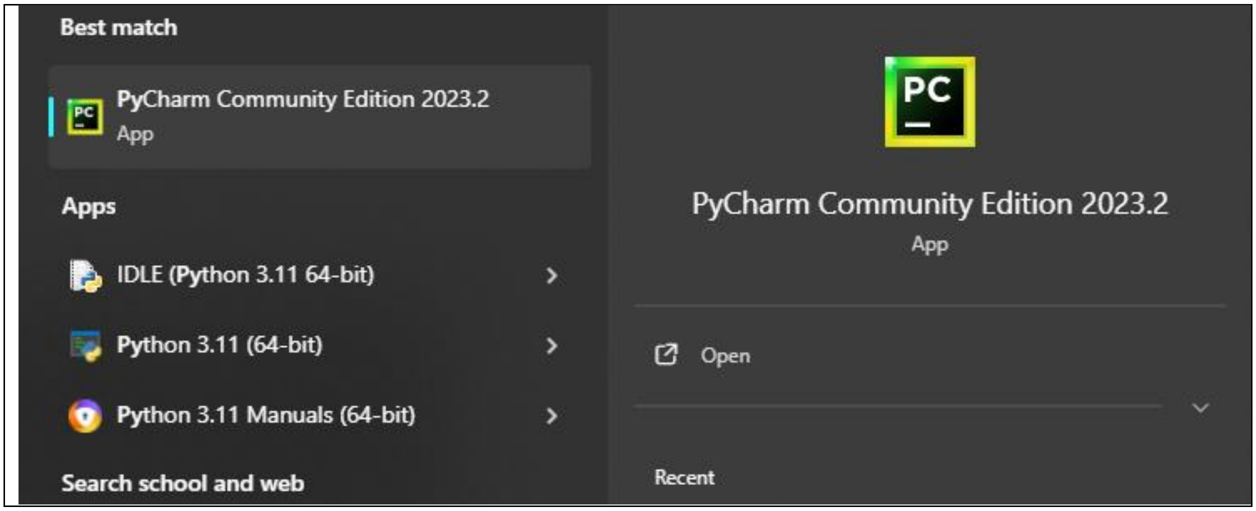
Hardware: Computer/Laptop.

Software: Python, PyCharm Community Edition 2023.2, Django.

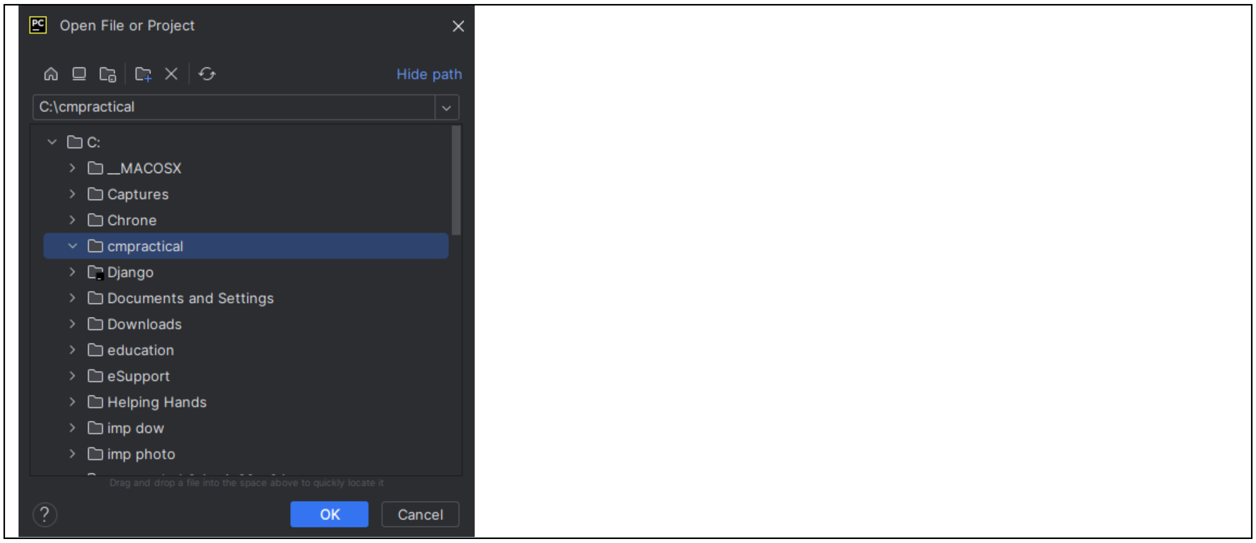
Solution:

Step 1: We need to create a new folder where you want to create the Django project. I am creating a folder inside of C drive that is **cmpractical.**

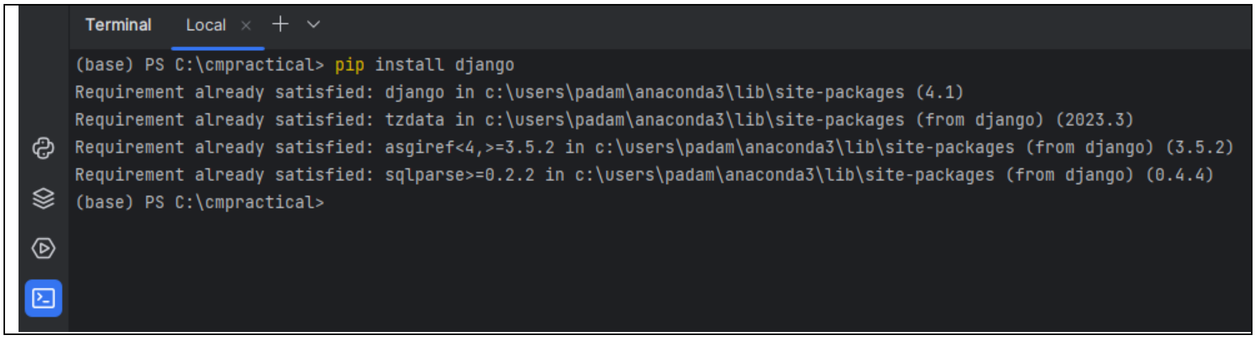
Step 2: Go to the Windows search bar, type PyCharm Community Edition 2023.2, and open it.


Best match
PyCharm Community Edition 2023.2
App
Apps
IDLE (Python 3.11 64-bit)
Python 3.11 (64-bit)
Python 3.11 Manuals (64-bit)
Search school and web
PC
PyCharm Community Edition 2023.2
App
Open
Recent

Step 3: Know we need to open our folder.

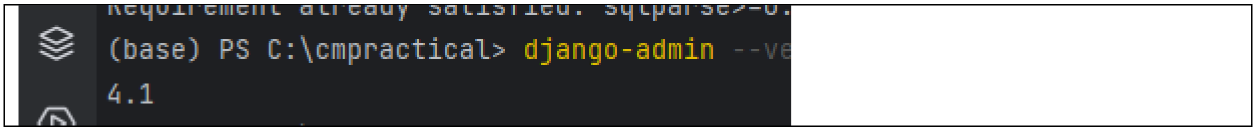


Step 4: After the 3rd step, we need to install Django go to the terminal and type **pip install django** and hit enter.



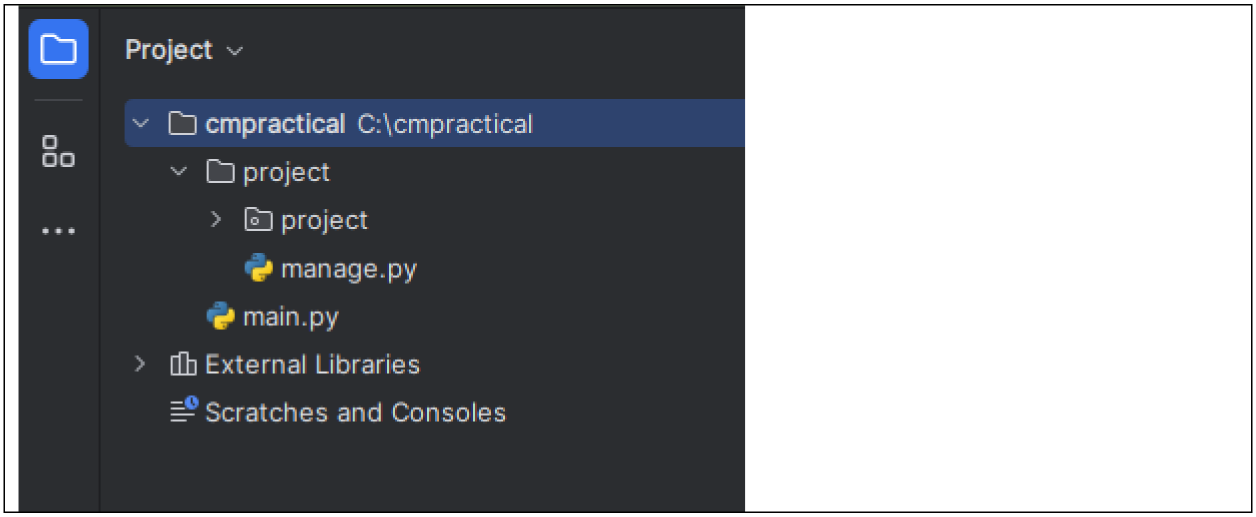
```
(base) PS C:\cmpractical> pip install django
Requirement already satisfied: django in c:\users\padam\anaconda3\lib\site-packages (4.1)
Requirement already satisfied: tzdata in c:\users\padam\anaconda3\lib\site-packages (from django) (2023.3)
Requirement already satisfied: asgiref<4,>=3.5.2 in c:\users\padam\anaconda3\lib\site-packages (from django) (3.5.2)
Requirement already satisfied: sqlparse>=0.2.2 in c:\users\padam\anaconda3\lib\site-packages (from django) (0.4.4)
(base) PS C:\cmpractical>
```

Step 5: Check the Django version type **django-admin –version** and hit enter.

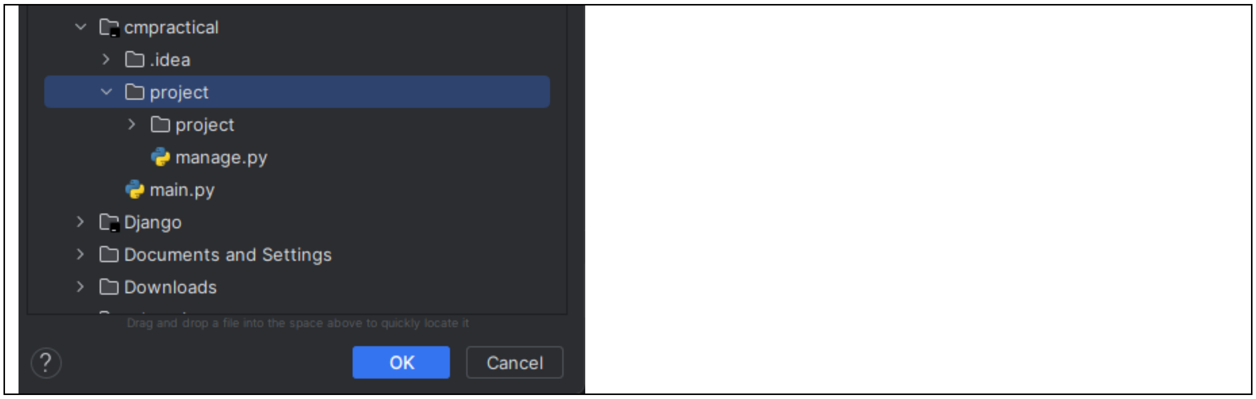


```
Requirement already satisfied: sqlparse>=0.
(base) PS C:\cmpractical> django-admin --ve
4.1
```

Step 6: Create a Django project for type **django-admin startproject project** and hit enter. After some time, we can see a folder created automatically whose name is project.

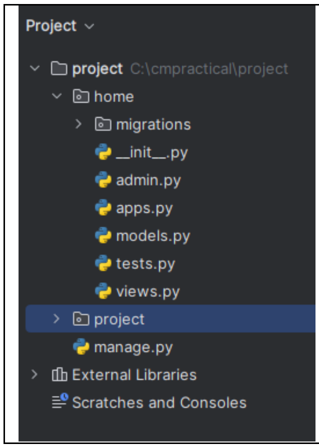


Step 7: Open your Django project folder in the PyCharm.



Drag and drop a file into the space above to quickly locate it

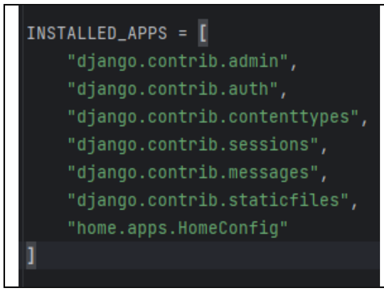
Step 8: Open the terminal and create a Django app type py manage.py startapp home and hit enter. After some time, we can see a folder created automatically whose name is home.



```
Project ⌄
⌄ project C:\cmpractical\project
    ⌄ home
        > migrations
          __init__.py
          admin.py
          apps.py
          models.py
          tests.py
          views.py
    > project
      manage.py
> External Libraries
  Scratches and Consoles
```

Step 9: First we need to configure our app. Go to the setting.py and write this line inside of INSTALLED_APPS.

```
"home.apps.HomeConfig"
```



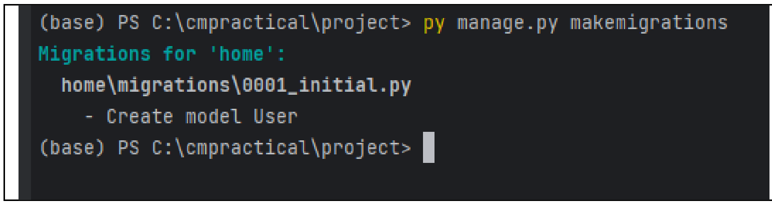
```
INSTALLED_APPS = [
    "django.contrib.admin",
    "django.contrib.auth",
    "django.contrib.contenttypes",
    "django.contrib.sessions",
    "django.contrib.messages",
    "django.contrib.staticfiles",
    "home.apps.HomeConfig"
]
```

Step 10: Go to models.py and type these codes.

Input:

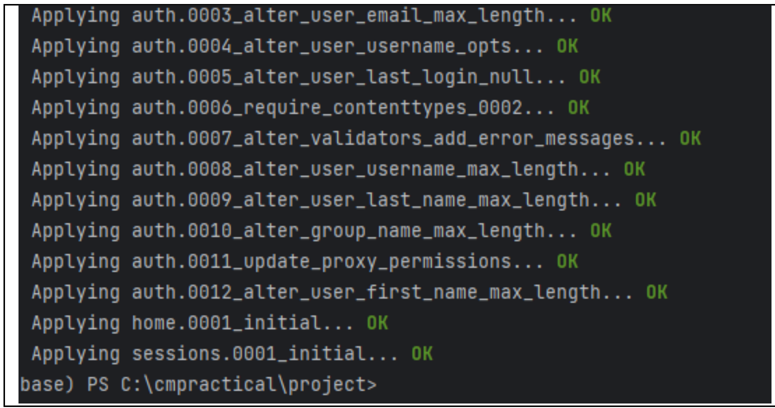
```
from django.db import models

# Create your models here.
class User(models.Model):
    name = models.CharField(max_length=122)
    phone = models.CharField(max_length=122)


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

Step 11: Go to the terminal type **py manage.py makemigrations** and hit enter.



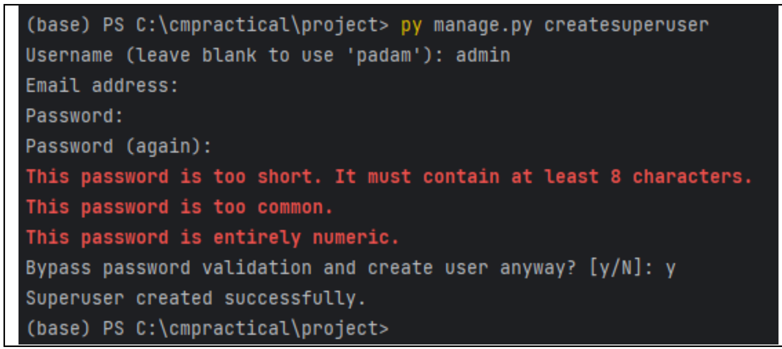
```
(base) PS C:\cmpractical\project> py manage.py makemigrations
Migrations for 'home':
  home\migrations\0001_initial.py
    - Create model User
(base) PS C:\cmpractical\project>
```

Step 12: Know type **py manage.py migrate** and hit enter.



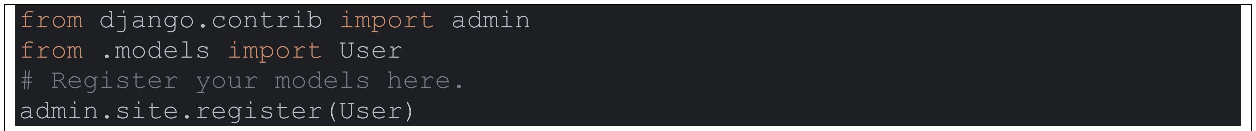
```
 Applying auth.0003_alter_user_email_max_length... OK
 Applying auth.0004_alter_user_username_opts... OK
 Applying auth.0005_alter_user_last_login_null... OK
 Applying auth.0006_require_contenttypes_0002... OK
 Applying auth.0007_alter_validators_add_error_messages... OK
 Applying auth.0008_alter_user_username_max_length... 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 auth.0012_alter_user_first_name_max_length... OK
 Applying home.0001_initial... OK
 Applying sessions.0001_initial... OK
base) PS C:\cmpractical\project>
```

Step 13: Create a superuser type **py manage.py createsuperuser** and hit enter and give the user name and password.



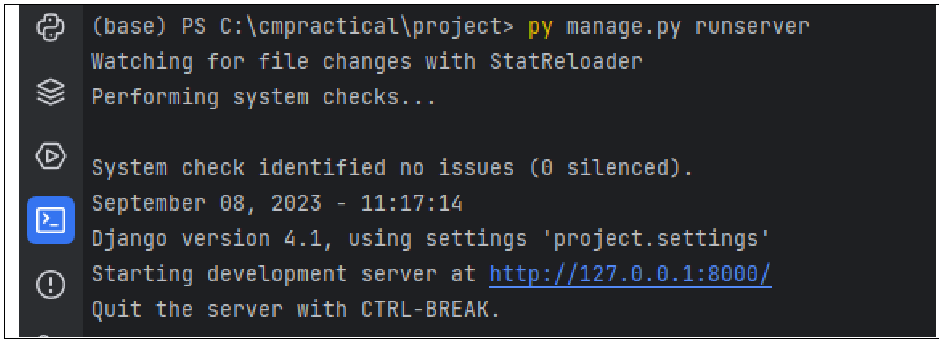
```
(base) PS C:\cmpractical\project> py manage.py createsuperuser
Username (leave blank to use 'padam'): admin
Email address:
Password:
Password (again):
This password is too short. It must contain at least 8 characters.
This password is too common.
This password is entirely numeric.
Bypass password validation and create user anyway? [y/N]: y
Superuser created successfully.
(base) PS C:\cmpractical\project>
```

Step 14: Open admin.py and type these codes.



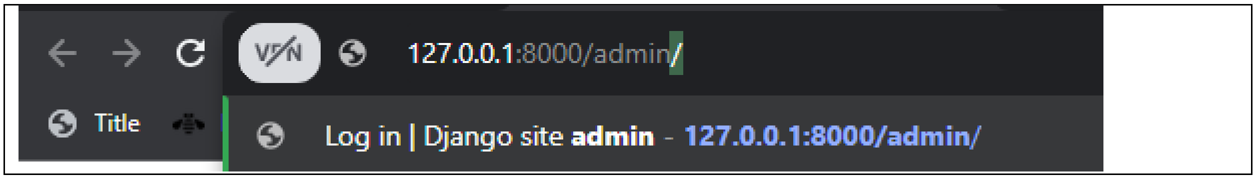
```
from django.contrib import admin
from .models import User
# Register your models here.
admin.site.register(User)
```

Step 15: Open the terminal and type **py manage.py runserver** hit enter and click on the link.



```
(base) PS C:\cmpractical\project> py manage.py runserver
Watching for file changes with StatReloader
Performing system checks...

System check identified no issues (0 silenced).
September 08, 2023 - 11:17:14
Django version 4.1, using settings 'project.settings'
Starting development server at http://127.0.0.1:8000/
Quit the server with CTRL-BREAK.
```

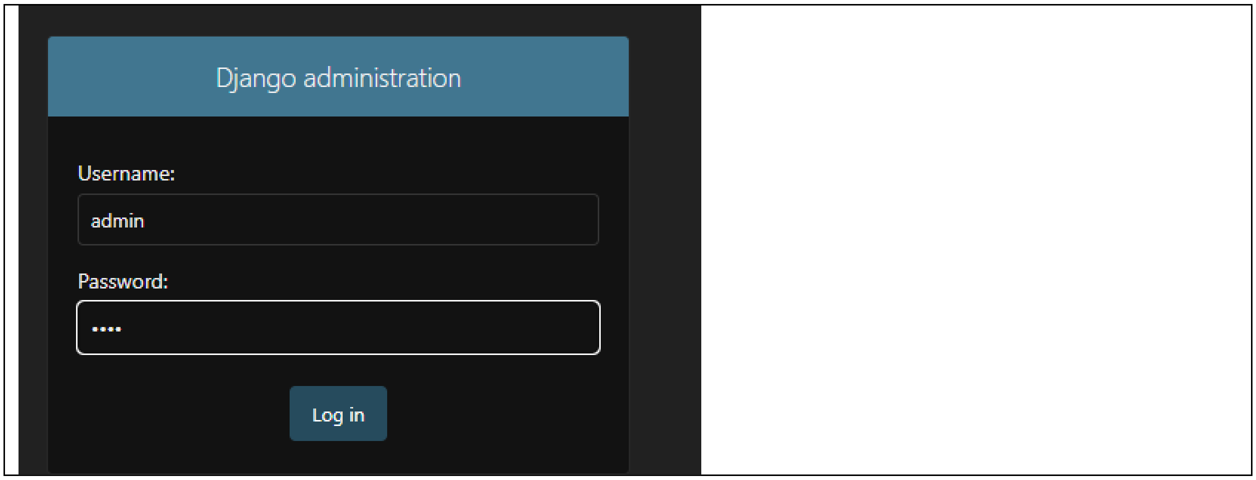
Step 16: Type **/admin** on the browser search bar.



127.0.0.1:8000/admin/

Log in | Django site **admin** - 127.0.0.1:8000/admin/

Step 17: Fill the user **name**, **password** and click on **Log in**.



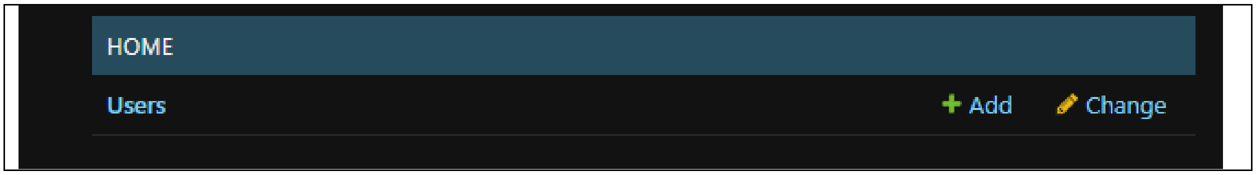
Username: admin

Password: ····

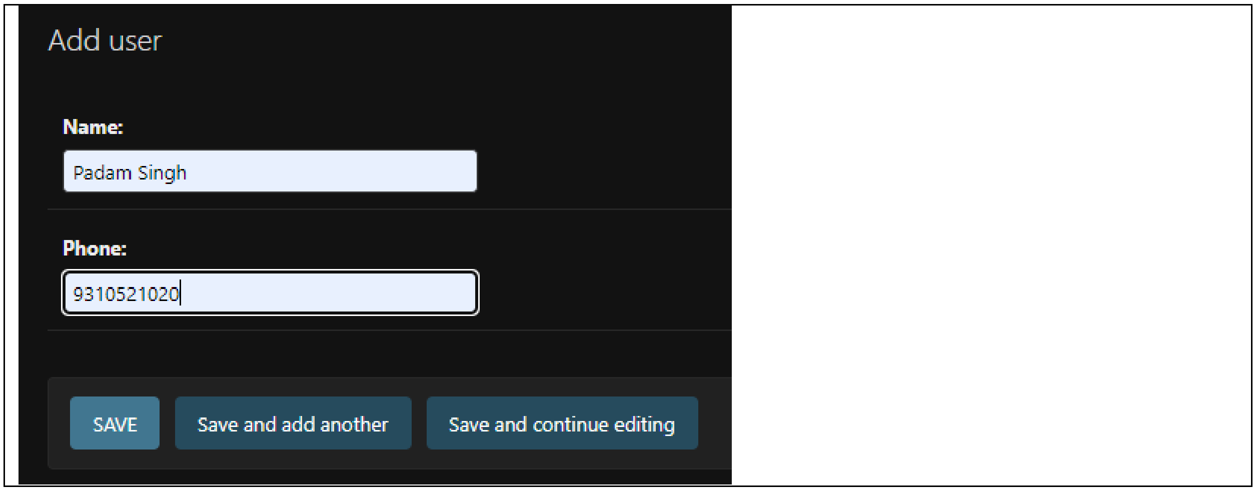
Output:

CRUD - Create

Step 18: You can see here a table whose name is Users.



Step 19: Create a user click on the **add button** fill the **name**, **phone** and click on **save**.

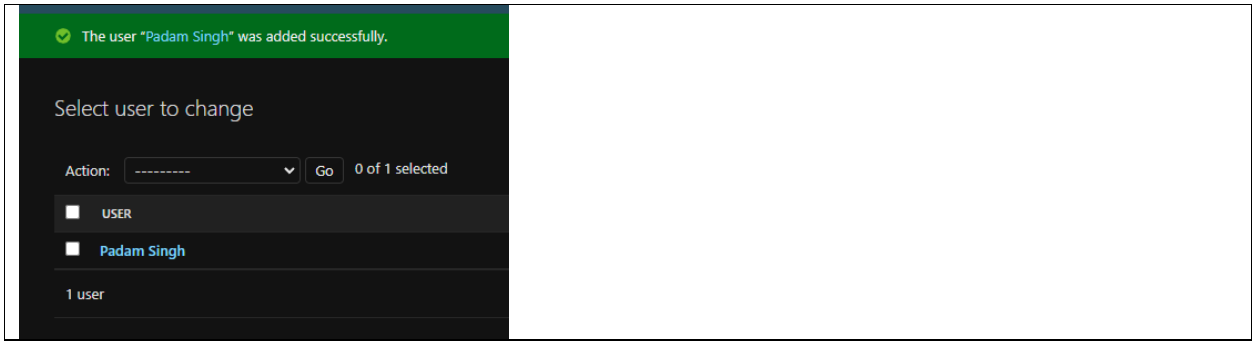


Name: Padam Singh

Phone: 9310521020

CRUD - Read

Step 20: You can see user created successfully.



The user "Padam Singh" was added successfully.

Select user to change

Action: --------- Go 0 of 1 selected

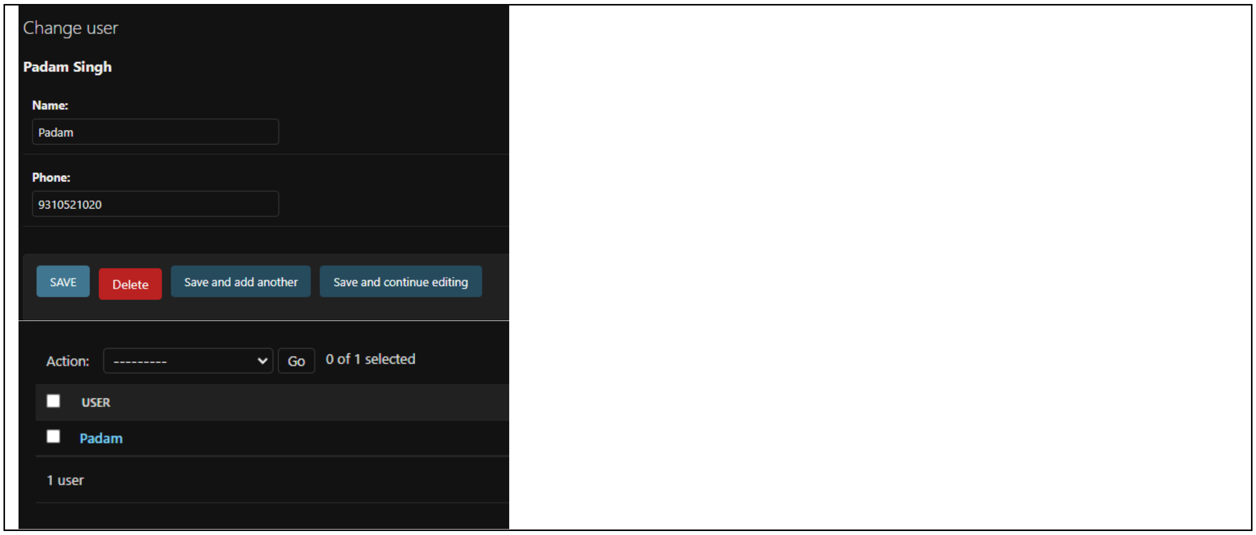
□ USER

□ Padam Singh

1 user

CRUD - Update

Step 21: Click on user and update it and click on save.



Change user

Padam Singh

Name: Padam

Phone: 9310521020

Action: --------- Go 0 of 1 selected

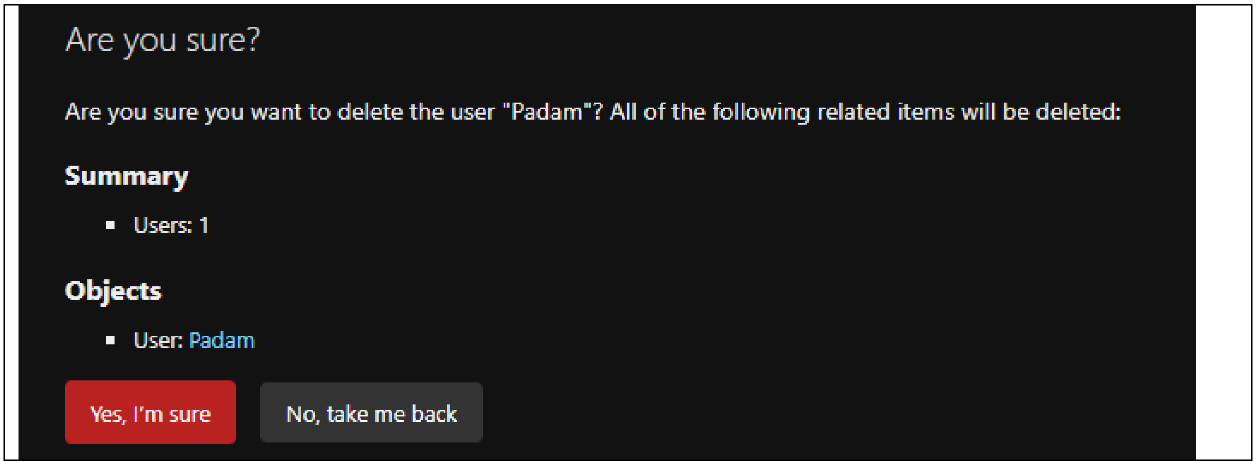
□ USER

□ Padam

1 user

CRUD – Delete

Step 22: Click on user, then click on delete button after that click on Yes, I’m sure.



Are you sure?

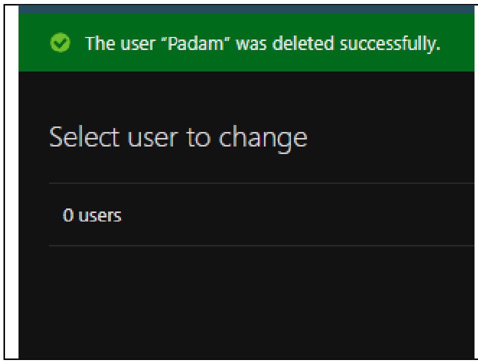
Are you sure you want to delete the user "Padam"? All of the following related items will be deleted:

Summary

- Users: 1

Objects

- User: Padam



The user "Padam" was deleted successfully.

Select user to change

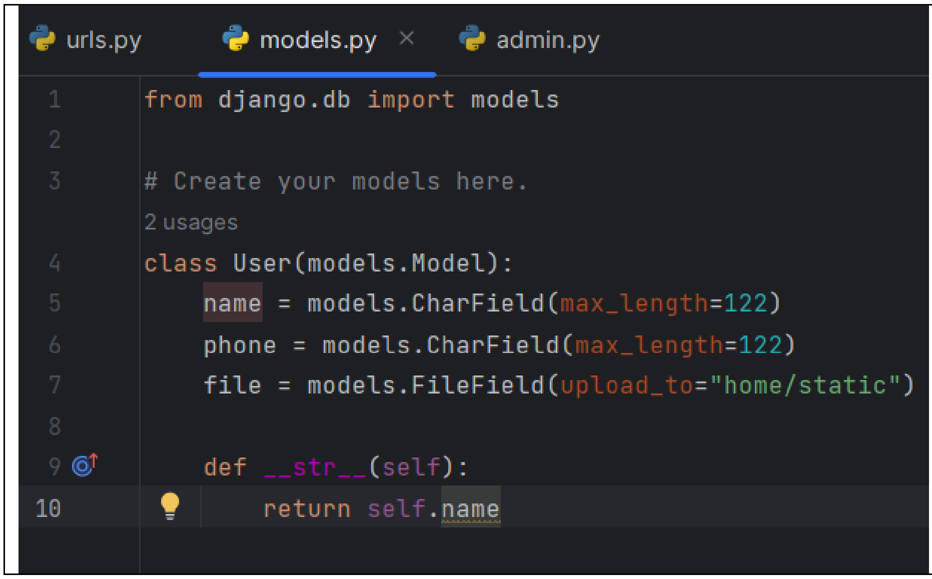
0 users

Our user delete successfully.

File Upload

Step 1. We need to write one line into the models.py

```
file = models.FileField(upload_to="home/static")
```

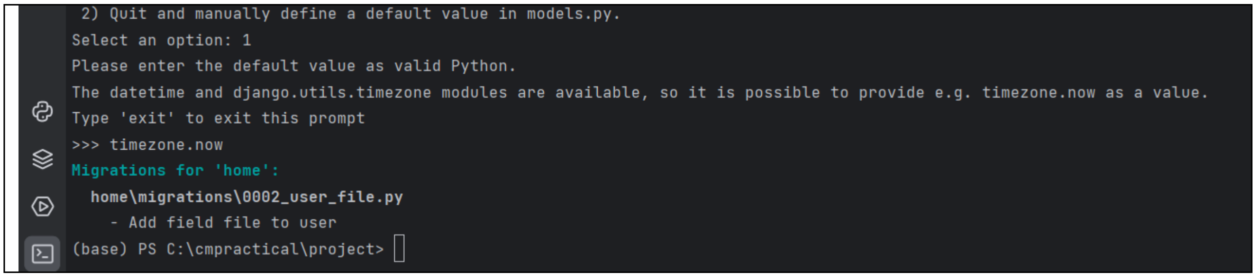


```
from django.db import models

# Create your models here.
class User(models.Model):
    name = models.CharField(max_length=122)
    phone = models.CharField(max_length=122)
    file = models.FileField(upload_to="home/static")

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

Step 2. Go to the setting.py find the Static files and type this code.

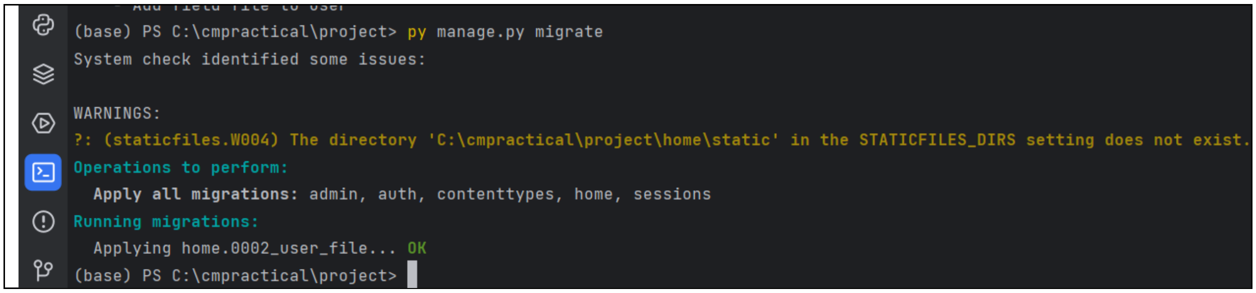
```
STATIC_URL = "home/static/"

STATICFILES_DIRS = [
    BASE_DIR / "home/static/"
]
```

Step 3. Go to the terminal and type **py manage.py makemigrations** and hit enter after that type **timezone.now** and hit enter.



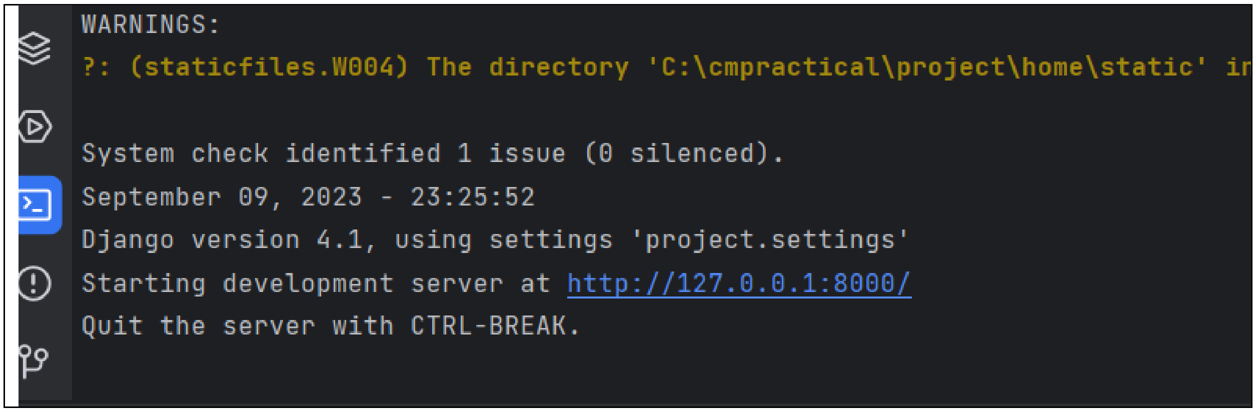
```
 2) Quit and manually define a default value in models.py.
Select an option: 1
Please enter the default value as valid Python.
The datetime and django.utils.timezone modules are available, so it is possible to provide e.g. timezone.now as a value.
Type 'exit' to exit this prompt
>>> timezone.now
Migrations for 'home':
  home\migrations\0002_user_file.py
    - Add field file to user
(base) PS C:\cmpractical\project> 
```

Step 4. Type **py manage.py migrate** and hit enter.



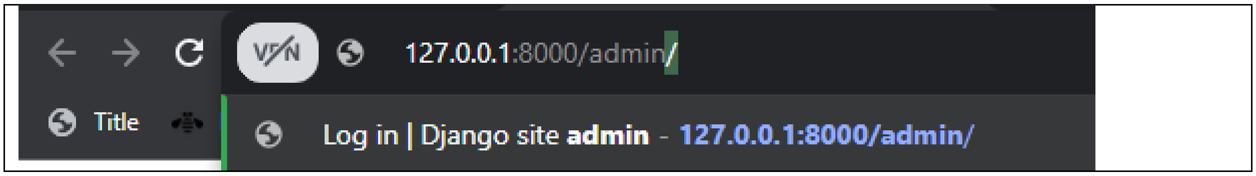
```
    - Add field file to user
(base) PS C:\cmpractical\project> py manage.py migrate
System check identified some issues:

WARNINGS:
?: (staticfiles.W004) The directory 'C:\cmpractical\project\home\static' in the STATICFILES_DIRS setting does not exist.
Operations to perform:
  Apply all migrations: admin, auth, contenttypes, home, sessions
Running migrations:
  Applying home.0002_user_file... OK
(base) PS C:\cmpractical\project> 
```

Step 5. Type **py manage.py runserver** and hit enter. You get a link simple click on it.



```
WARNINGS:
?: (staticfiles.W004) The directory 'C:\cmpractical\project\home\static' in

System check identified 1 issue (0 silenced).
September 09, 2023 - 23:25:52
Django version 4.1, using settings 'project.settings'
Starting development server at http://127.0.0.1:8000/
Quit the server with CTRL-BREAK.
```

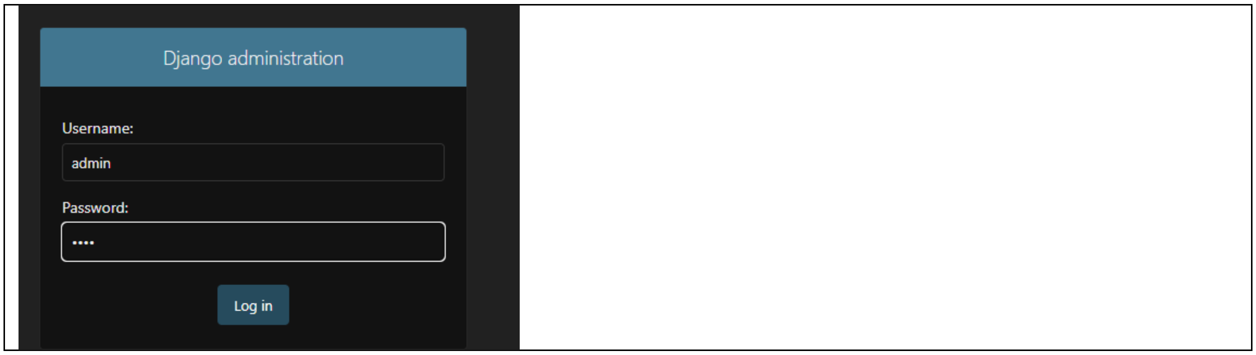
Step 6. Type **/admin** on browser search bar.



127.0.0.1:8000/admin/

Log in | Django site admin - 127.0.0.1:8000/admin/

Step 7. Fill the user **name**, **password** and click on **Log in**.

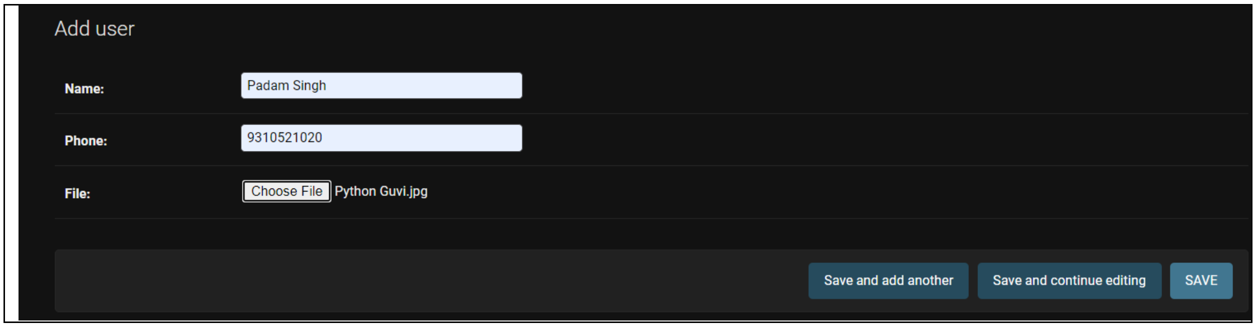


Django administration

Username: admin

Password: ····

Step 8. Create a user click on the **add button** fill the **name**, **phone** and click on **save**.



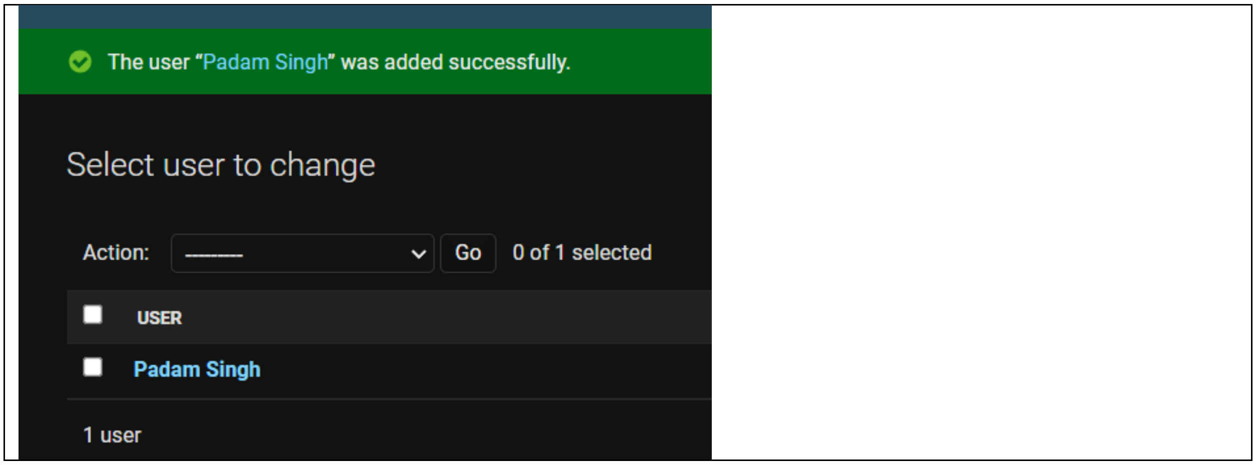
Add user

Name: Padam Singh

Phone: 9310521020

File: Choose File Python Guvi.jpg

Step 9. The user added successfully and the file was also uploaded.



The user “Padam Singh” was added successfully.

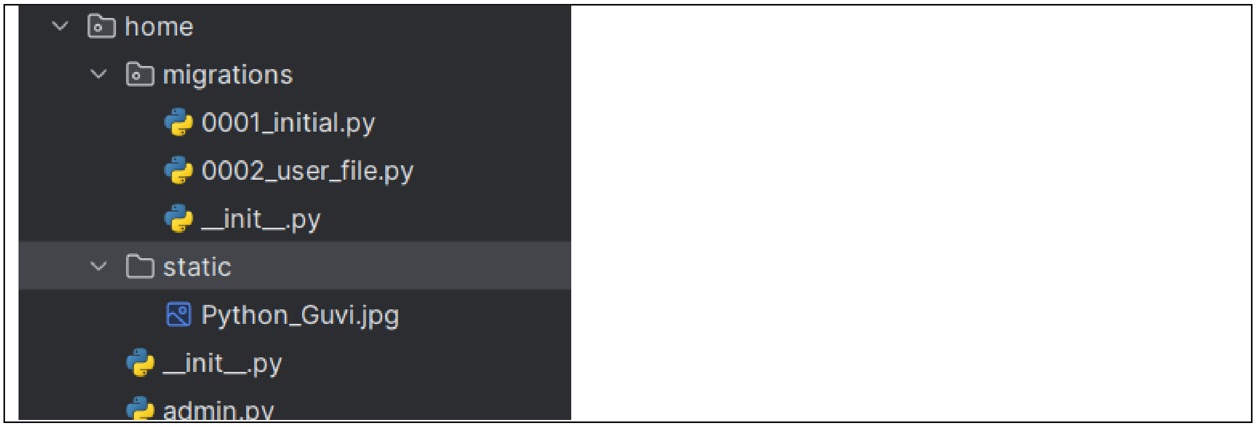
Select user to change

Action: --------- Go 0 of 1 selected

□	USER
□	Padam Singh

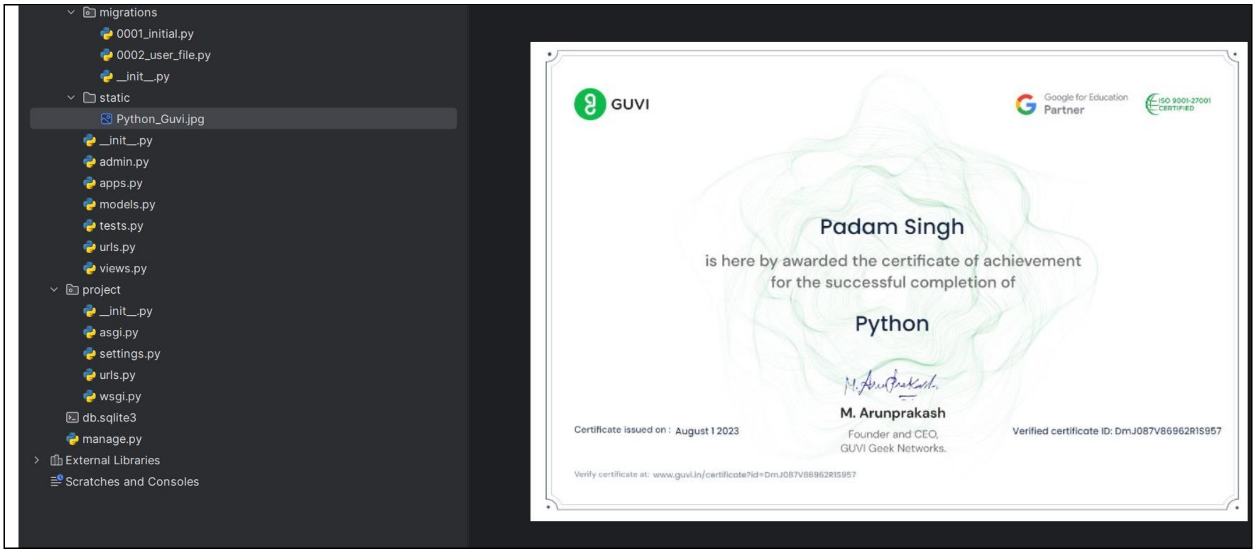
1 user

Step 10. To check file upload or not go to the static file.



```
home
  migrations
    0001_initial.py
    0002_user_file.py
    __init__.py
  static
    Python_Guvi.jpg
  __init__.py
  admin.py
```

I can read my file.



```
    migrations
      0001_initial.py
      0002_user_file.py
      __init__.py
    static
      Python_Guvi.jpg
    __init__.py
    admin.py
    apps.py
    models.py
    tests.py
    urls.py
    views.py
  project
    __init__.py
    asgi.py
    settings.py
    urls.py
    wsgi.py
  db.sqlite3
  manage.py
External Libraries
Scratches and Consoles
```

GUVI

Google for Education Partner

ISO 9001-27001 CERTIFIED

Padam Singh

is here by awarded the certificate of achievement
for the successful completion of

Python

M. Arunprakash

M. Arunprakash
Founder and CEO,
GUVI Geek Networks.

Certificate issued on : August 1 2023

Verified certificate ID: DmJ087V86962R1S957

Verify certificate at: www.guvi.in/certificate?id=DmJ087V86962R1S957

We have done with CRUD and File uploading.