

django

Project: Build your application with Django

prepared by
Aya Rabih Mostafa

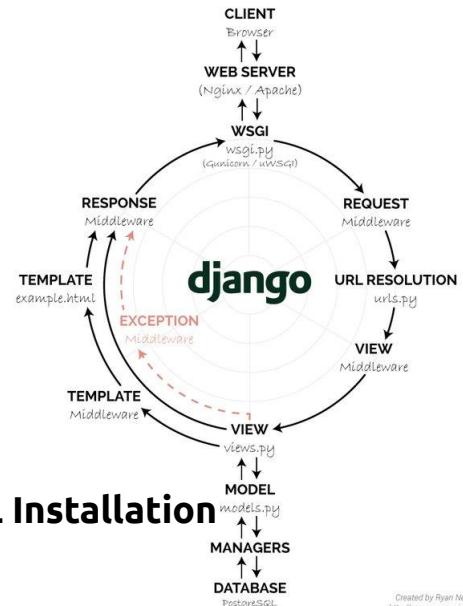
[LinkedIn](#) [GitHub](#)



What is our plan?



before you will run Django
you have to sure that you have python3 and mysql



Install Python 3

1. sudo apt update
2. sudo apt install software-properties-common
3. sudo add-apt-repository ppa:deadsnakes/ppa
4. sudo apt update
5. sudo apt install python3.8
6. python --version

we will Installing MySQL on Ubuntu

1. sudo apt update
2. sudo apt upgrade
3. sudo apt install mysql-server
4. mysql --version

Installing MySQL on Ubuntu: Configure the MySQL Installation

- 1) sudo mysql_secure_installation
- 2) you will chose your level security

There are three levels of password validation policy:

LOW Length >= 8

MEDIUM Length >= 8, numeric, mixed case, and special characters

STRONG Length >= 8, numeric, mixed case, special characters and dictionary file

Please enter 0 = LOW, 1 = MEDIUM and 2 = STRONG:2

Please set the password for root here.

- 3) New password:

Re-enter new password:

- 4) Estimated strength of the password: 100

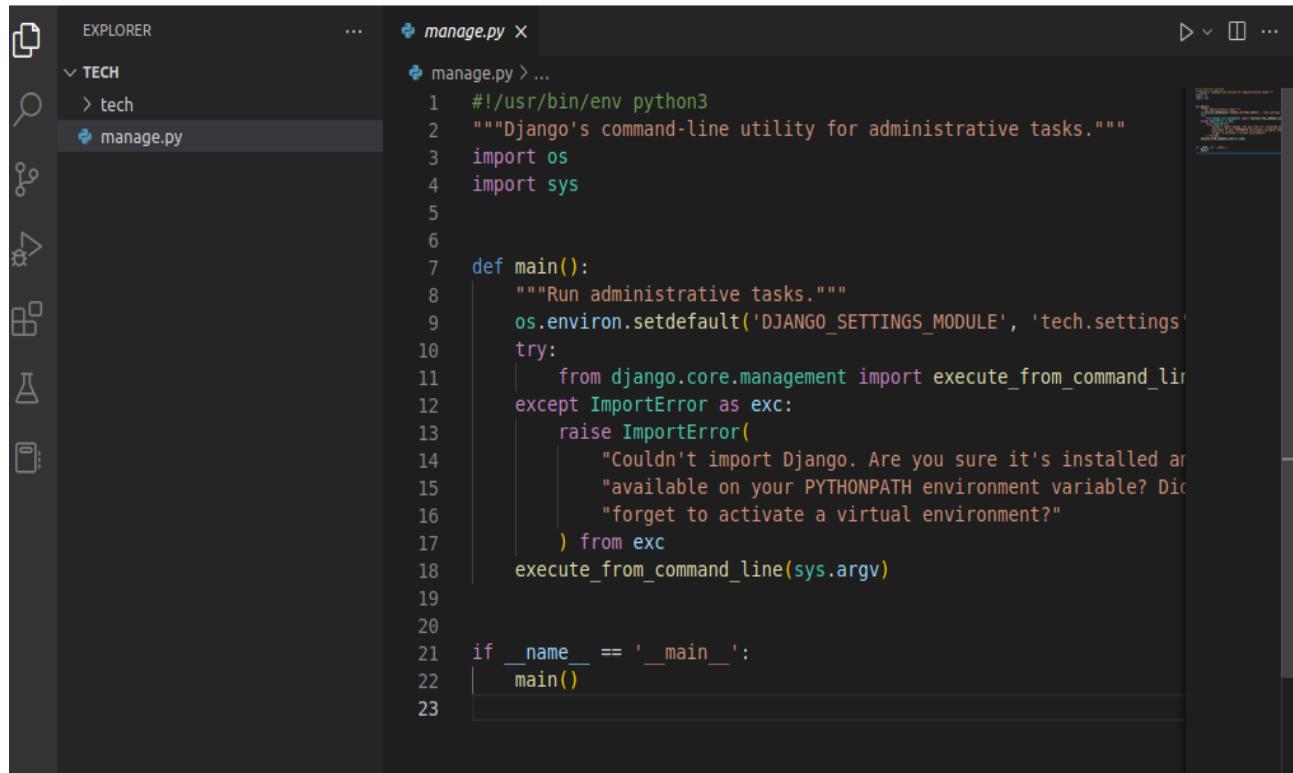
- 5)Do you wish to continue with the password provided? (Press y|Y for Yes, any other key for No) :Y

- 6)sudo mysql

- 7) CREATE USER 'ayarabih'@'localhost' IDENTIFIED by '*****';

Now we will Install the Django Web Framework

- ✓ sudo apt install python3-Django
 - ✓ django-admin –version
 - ✓ We will make project name tech
1. Django-admin startproject tech
 2. it will create file name django and if you open it you will see two file one of them named tech and other it is manage.py



A screenshot of the Visual Studio Code interface. The left sidebar shows a tree view with a folder named 'TECH' expanded, containing a 'tech' folder and a file named 'manage.py'. The main editor area displays the content of the 'manage.py' file. The code is a Python script for a Django command-line utility. It includes imports for os and sys, a main function that runs administrative tasks, and a check for the __name__ variable to determine if it's being run directly or from a command line.

```
EXPLORER    ...  manage.py x
manage.py > ...
1  #!/usr/bin/env python3
2  """Django's command-line utility for administrative tasks."""
3  import os
4  import sys
5
6
7  def main():
8      """Run administrative tasks."""
9      os.environ.setdefault('DJANGO_SETTINGS_MODULE', 'tech.settings')
10     try:
11         from django.core.management import execute_from_command_line
12     except ImportError as exc:
13         raise ImportError(
14             "Couldn't import Django. Are you sure it's installed and "
15             "available on your PYTHONPATH environment variable? Did "
16             "you forget to activate a virtual environment?"
17         ) from exc
18     execute_from_command_line(sys.argv)
19
20
21 if __name__ == '__main__':
22     main()
23
```

✓ sudo subl ./tech/

```
aya@aya-Latitude-E6540:~/tech$ django-admin startproject tech
aya@aya-Latitude-E6540:~/tech$ cd tech
aya@aya-Latitude-E6540:~/tech$ ls
manage.py  tech
aya@aya-Latitude-E6540:~/tech$
```

- ✓ now we will write in our terminal commit to run server
- ✓ python3 manage.py runserver
- ✓ http://127.0.0.1:8000/

```
aya@aya-Latitude-E6540:~/tech$ python3 manage.py runserver
Watching for file changes with StatReloader
Performing system checks...

System check identified no issues (0 silenced).
```

```
You have 18 unapplied migration(s). Your project may not work properly until you apply the migrations for app(s): admin, auth, contenttypes, sessions.
Run 'python manage.py migrate' to apply them.
November 20, 2022 - 12:12:03
Django version 3.2.12, using settings 'tech.settings'
Starting development server at http://127.0.0.1:8000/
Quit the server with CONTROL-C.
```

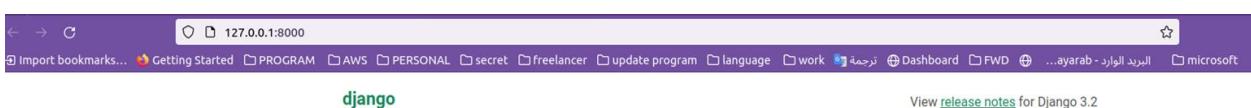
File Edit Selection View Go Run Terminal Help

EXPLORER ... settings.py - tech - Visual Studio Code

TECH > settings.py > ...

```
54     TEMPLATES = [
55         {
56             'BACKEND': 'django.template.backends.django.DjangoTemplates',
57             'DIRS': [],
58             'APP_DIRS': True,
59             'OPTIONS': {
60                 'context_processors': [
61                     'django.template.context_processors.debug',
62                     'django.template.context_processors.request',
63                     'django.contrib.auth.context_processors.auth',
64                     'django.contrib.messages.context_processors.messages',
65                 ],
66             },
67         },
68     ]
69
70 WSGI_APPLICATION = 'tech.wsgi.application'
71
72 # Database
73 # https://docs.djangoproject.com/en/3.2/ref/settings/#databases
74
75 DATABASES = {
76     'default': {
77         'ENGINE': 'django.db.backends.sqlite3',
78         'NAME': BASE_DIR / 'db.sqlite3',
79     }
80 }
81
82
83 # Password validation
84 # https://docs.djangoproject.com/en/3.2/ref/settings/#auth-password-validators
85
86 AUTH_PASSWORD_VALIDATORS = [
87     {
88         'NAME': 'django.contrib.auth.password_validation.UserAttributeSimilarityValidator',
89     },
90     {
91         'NAME': 'django.contrib.auth.password_validation.MinimumLengthValidator',
92     },
93     {
94         'NAME': 'django.contrib.auth.password_validation.CommonPasswordValidator',
95     },
96     {
97         'NAME': 'django.contrib.auth.password_validation.NumericPasswordValidator',
98     },
99 ]
```

Python extension loading... ✓ it is running now



- ✓ now we will open setting.py in part of database to connect our database mysql with django
- ✓ now we will open mysql with our user
- ✓ mysql -u ayarabih -p

```
aya@aya-Latitude-E6540:~/tech$ mysql -u ayarabih -p
Enter password:
Welcome to the MySQL monitor. Commands end with ; or \g.
Your MySQL connection id is 16
Server version: 8.0.31-0ubuntu0.22.04.1 (Ubuntu)

Copyright (c) 2000, 2022, Oracle and/or its affiliates.

Oracle is a registered trademark of Oracle Corporation and/or its
affiliates. Other names may be trademarks of their respective
owners.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.
```

```
mysql> 
✓ create database tech;
```

```
Query OK, 1 row affected (0.00 sec)
```

```
mysql> show databases;
+-----+
| Database      |
+-----+
| information_schema |
| mysql          |
| performance_schema |
| sys            |
| techdb         |
+-----+
5 rows in set (0.00 sec)
```

```
mysql>
```

- ✓ now we will our techdb
- ✓ use techdb;
- ✓ show tables;

```
mysql> show databases;
+-----+
| Database      |
+-----+
| information_schema |
| mysql          |
| performance_schema |
| sys            |
| techdb         |
+-----+
5 rows in set (0.00 sec)
```

```
mysql> use techdb;
Database changed
mysql> show tables;
Empty set (0.00 sec)
```

```
mysql>
```

- ✓ than mean that we don't have any table right now
- ✓ so we will now change database in setting file

```
2
3 # Database
4 # https://docs.djangoproject.com/en/3.2/ref/settings/#databases
5
6 DATABASES = {
7     'default': {
8         'ENGINE': 'django.db.backends.mysql',
9         'NAME': 'techdb',
10        'USER': 'ayarabih',
11        'PASSWORD': 'Ph@rma1911',
12        'HOST': 'localhost',
13        'PORT': '3386'
14    }
15}
16
17
```

```
python3 manage.py migrate
```

```
aya@aya-Latitude-E6540:~/tech$ python3 manage.py migrate
Operations to perform:
  Apply all migrations: admin, auth, contenttypes, sessions
Running migrations:
  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 auth.0012_alter_user_first_name_max_length... OK
  Applying sessions.0001_initial... OK
aya@aya-Latitude-E6540:~/tech$
```

- now we will use our database tech db
- use techdb ;
- show table;

```

mysql> use techdb
Reading table information for completion of table and column names
You can turn off this feature to get a quicker startup with -A

Database changed
mysql> show tables;
+-----+
| Tables_in_techdb |
+-----+
| auth_group
| auth_group_permissions
| auth_permission
| auth_user
| auth_user_groups
| auth_user_user_permissions
| django_admin_log
| django_content_type
| django_migrations
| django_session
+-----+
10 rows in set (0.00 sec)

mysql> █

```

now we will create app name devops

python3 manage.py startapp devops

when you open file you will see that you create app name devops

```

EXPLORER      ...      settings.py ×
TECH > settings.py > ...
tech > settings.py > ...
54     TEMPLATES = [
55         {
56             'BACKEND': 'django.template.backends.django.DjangoTemplates',
57             'DIRS': [],
58             'APP_DIRS': True,
59             'OPTIONS': {
60                 'context_processors': [
61                     'django.template.context_processors.debug',
62                     'django.template.context_processors.request',
63                     'django.contrib.auth.context_processors.auth',
64                     'django.contrib.messages.context_processors.messages',
65                 ],
66             },
67         },
68     ],
69
70     WSGI_APPLICATION = 'tech.wsgi.application'
71
72
73     # Database
74     # https://docs.djangoproject.com/en/3.2/ref/settings/#databases
75
76     DATABASES = {
77         'default': {
78             'ENGINE': 'django.db.backends.mysql',
79             'NAME': 'techdb',
80             'USER': 'ayarabih',
81             'PASSWORD': 'Ph@m1911',
82             'HOST': 'localhost',
83             'PORT': '3386'
84         }
85     }
86
87
88     # Password validation
89     # https://docs.djangoproject.com/en/3.2/ref/settings/#auth-password validators
90
91     AUTH_PASSWORD_VALIDATORS = [
92         {
93             'NAME': 'django.contrib.auth.password_validation.UserAttributeSimilarityValidator'
94         },
95         {
96             'NAME': 'django.contrib.auth.password_validation.MinimumLengthValidator'
97         },
98         {
99             'NAME': 'django.contrib.auth.password_validation.CommonPasswordValidator'

```

Ln 91, Col 29 Spaces: 4 UTF-8 LF ↵ Python 3.10.6 64-bit

The screenshot shows the VS Code interface with the 'settings.py' file open in the editor. The file is located in the 'devops' directory under the 'TECH' folder. The code defines various Django settings, including templates, WSGI application, databases, and password validators.

```
TEMPLATES = [
    {
        'BACKEND': 'django.template.backends.django.DjangoTemplates',
        'DIRS': [],
        'APP_DIRS': True,
        'OPTIONS': {
            'context_processors': [
                'django.template.context_processors.debug',
                'django.template.context_processors.request',
                'django.contrib.auth.context_processors.auth',
                'django.contrib.messages.context_processors.messages',
            ],
        },
    },
]
WSGI_APPLICATION = 'tech.wsgi.application'

# Database
# https://docs.djangoproject.com/en/3.2/ref/settings/#databases

DATABASES = {
    'default': {
        'ENGINE': 'django.db.backends.mysql',
        'NAME': 'techdb',
        'USER': 'ayarabih',
        'PASSWORD': 'Ph@rma1911',
        'HOST': 'localhost',
        'PORT': '3386'
    }
}

# Password validation
# https://docs.djangoproject.com/en/3.2/ref/settings/#auth-password-validators

AUTH_PASSWORD_VALIDATORS = [
    {
        'NAME': 'django.contrib.auth.password_validation.UserAttributeSimilarityValidator',
    },
    {
        'NAME': 'django.contrib.auth.password_validation.MinimumLengthValidator',
    },
    {
        'NAME': 'django.contrib.auth.password_validation.CommonPasswordValidator'
    }
]
```

- now we will open our app and chose model.py

The screenshot shows the VS Code interface with the 'models.py' file open in the editor. The file is located in the 'devops' directory under the 'TECH' folder. It contains Python code defining two Django models: 'Track' and 'Student'. The 'Track' model has a single CharField named 'name'. The 'Student' model has three fields: 'fname' (CharField), 'lname' (CharField), and 'age' (IntegerField). It also has a ForeignKey relationship named 'student_track' that points to the 'Track' model with a CASCADE delete rule.

```
from django.db import models

# Create your models here.

class Track(models.Model):
    name = models.CharField(max_length = 50)

class Student(models.Model):
    fname = models.CharField(max_length = 50, null=True, default='someone')
    lname = models.CharField(max_length = 50, null=True, default='someone')
    age = models.IntegerField()
    student_track = models.ForeignKey(Track, on_delete=models.CASCADE)
```

- and write our class we will make 2 class one for track and other for student
- and now we will go to setting.py and write our name in install app(devops)

The screenshot shows the Visual Studio Code interface. The left sidebar displays a project structure under 'EXPLORER' with a tree view of files and folders. The main area shows the 'settings.py' file with the following code:

```

File Edit Selection View Go Run Terminal Help
● settings.py - tech - Visual Studio Code
File Edit Selection View Go Run Terminal Help
EXPLORER ... settings.py ● models.py 3
tech > settings.py > ...
30
31 # Application definition
32
33 INSTALLED_APPS = [
34     'devops',
35     'django.contrib.admin',
36     'django.contrib.auth',
37     'django.contrib.contenttypes',
38     'django.contrib.sessions',
39     'django.contrib.messages',
40     'django.contrib.staticfiles',
41
42 ]

```

- now run `python3 manage.py makemigrations`
- to migrate your app
- `python3 manage.py migrate`
- to migrate data to our database

```

aya@aya-Latitude-E6540:~/tech$ python3 manage.py makemigrations
Migrations for 'devops':
  devops/migrations/0001_initial.py
    - Create model Track
    - Create model Student
aya@aya-Latitude-E6540:~/tech$ python3 manage.py migrate
Operations to perform:
  Apply all migrations: admin, auth, contenttypes, devops, sessions
Running migrations:
  Applying devops.0001_initial... OK
aya@aya-Latitude-E6540:~/tech$ 
```

- now we will open our database to see if my devops app add or no so

```

mysql> use techdb;
Reading table information for completion of table and column names
You can turn off this feature to get a quicker startup with -A

Database changed
mysql> show tables;
+-----+
| Tables_in_techdb |
+-----+
| auth_group
| auth_group_permissions
| auth_permission
| auth_user
| auth_user_groups
| auth_user_user_permissions
| devops_student
| devops_track
| django_admin_log
| django_content_type
| django_migrations
| django_session
+-----+
12 rows in set (0.00 sec)

mysql> 
```

- now we name pf app and name of model

- if you need to show your models
- write describe + name of model

```
mysql> describe devops_student;
+-----+-----+-----+-----+-----+
| Field      | Type       | Null | Key | Default | Extra          |
+-----+-----+-----+-----+-----+
| id         | bigint     | NO   | PRI  | NULL    | auto_increment |
| fname      | varchar(50) | YES  |      | NULL    |                |
| lname      | varchar(50) | YES  |      | NULL    |                |
| age        | int        | NO   |      | NULL    |                |
| student_track_id | bigint     | NO   | MUL  | NULL    |                |
+-----+-----+-----+-----+-----+
5 rows in set (0.01 sec)
```

```
mysql> █
```

- python3 manage.py shell
- from devops.models import Student, Track
- **Track.objects.create(name = 'devops')**

```
aya@aya-Latitude-E6540:~/tech$ python3 manage.py shell
Python 3.10.6 (main, Nov  2 2022, 18:53:38) [GCC 11.3.0] on linux
Type "help", "copyright", "credits" or "license" for more information.
(InteractiveConsole)
>>>
>>>
>>> from devops.models import Student, Track
>>> Track.objects.create(name = 'devops')
<Track: Track object (1)>
>>> Track.objects.create(name = 'devops1')
<Track: Track object (2)>
>>> Track.objects.create(name = 'devops2')
<Track: Track object (3)>
>>> █
```

- that is mean we will create object with our name of app
- if you need to sure that your work cool
- select * from devops_track;

```
mysql> select * from devops_track;
+-----+
| id | name   |
+-----+
|  1 | devops |
|  2 | devops1|
|  3 | devops2|
+-----+
3 rows in set (0.00 sec)
```

```
mysql> █
```

- now open again python3 manage.py shell
- from devops.models import Student, Track
- devops = Track.objects.filter(name = 'devops')
- Student.objects.create(fname = 'mohamed', lname= 'mohamed', age=26, student_track= devops)

```
>>> devops
<Track: Track object (2)>
>>>
>>> Student.objects.create(fname = 'mohamed', lname= 'mohamed', age=26, student_track= devops)
<Student: Student object (1)>
>>>
```

- if you wanna to see your data

- Track.objects.all()
- Student.objects.all()

```
>>> Track.objects.all()
<QuerySet [<Track: devops>, <Track: devops1>]>
>>> Student.objects.all()
<QuerySet [<Student: mohamed mohamed>]>
>>
```

NOW we will use this app

python3 manage.py createsuperuser

Username : admin

Email address: admin@tech.com

```
aya@aya-Latitude-E6540:~/tech$ python3 manage.py createsuperuser
Username (leave blank to use 'aya'): admin
Email address: admin@tech.com
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.
aya@aya-Latitude-E6540:~/tech$
```

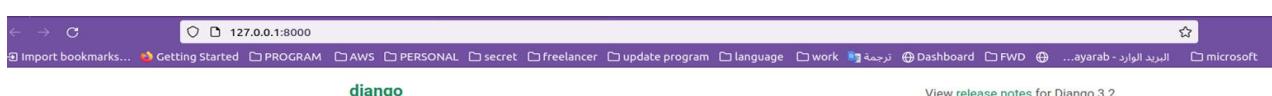
now we will run server

python3 manage.py runserver

```
aya@aya-Latitude-E6540:~/tech$ python3 manage.py runserver
Watching for file changes with StatReloader
Performing system checks...

System check identified no issues (0 silenced).
November 21, 2022 - 12:05:24
Django version 3.2.12, using settings 'tech.settings'
Starting development server at http://127.0.0.1:8000/
Quit the server with CONTROL-C.
```

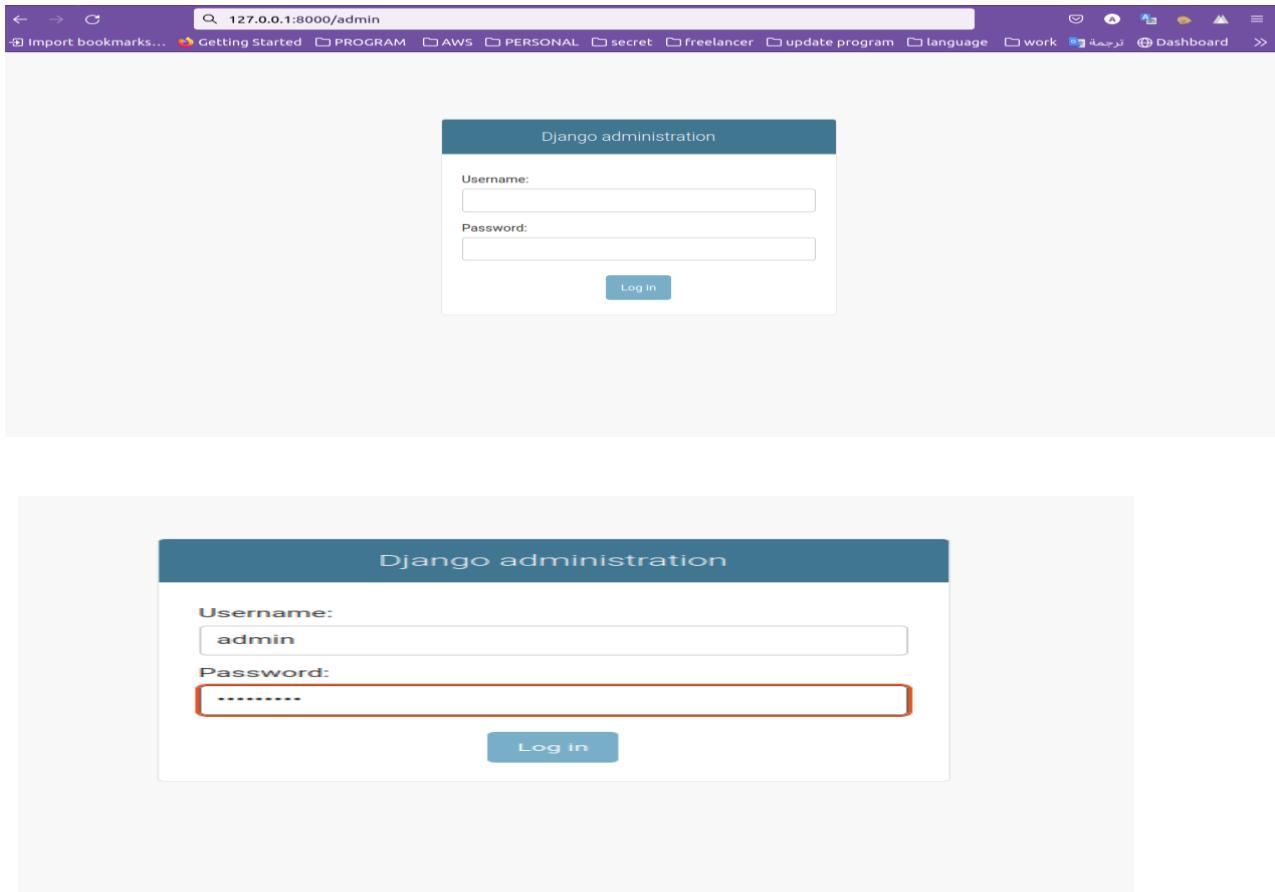
if you remember this one page open to us



The install worked successfully! Congratulations!

You are seeing this page because `DEBUG=True` is in your settings file and you have not configured any URLs.

- now refresh page and write /admin 127.0.0.1:8000/admin
- write your user name (admin) and your password (*****)



- now log in you will see that
- now we need to add our app (devops)
- go to admin.py and add
- now refresh your website to see your app and your data

admin.py - tech - Visual Studio Code

```

admin.py - tech - Visual Studio Code
File Edit Selection View Go Run Terminal Help
EXPLORER ... settings.py models.py admin.py ...
TECH
  devops
    __pycache__
    migrations
    __init__.py
    admin.py
  apps.py
  models.py
  tests.py
  views.py
  tech
  db.sqlite3
  manage.py

devops > admin.py
1   from django.contrib import admin
2   from .models import Student, Track
3   # Register your models here.
4   admin.site.register(Student)
5   admin.site.register[Track]

```

Django administration

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

Site administration

AUTHENTICATION AND AUTHORIZATION

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

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

Recent actions

My actions

None available

- so when we open first name you will see that

Django administration

Home > Devops > Students

AUTHENTICATION AND AUTHORIZATION

Groups [+ Add](#)

Users [+ Add](#)

DEVOPS

Students [+ Add](#)

Tracks [+ Add](#)

Select student to change

Action: ----- [Go](#) 0 of 2 selected

STUDENT

aya rabih

mohamed mohamed

2 students

Change student

aya rabih

Fname: aya

Lname: rabih

Age: 26

Student track: devops1

Delete Save and add another Save and continue editing SAVE

- we need to make two table
- one for (fname . Lname. Age)
- two (student track)
- we will do to admin .py and write that

```

● admin.py - tech - Visual Studio Code
File Edit Selection View Go Run Terminal Help
EXPLORER settings.py models.py admin.py
TECH devops migrations __init__.py admin.py apps.py models.py tests.py views.py db.sqlite3 manage.py

devops > admin.py > ...
1 from django.contrib import admin
2 from .models import Student, Track
3 # Register your models here.
4 class customstudent(admin.ModelAdmin):
5     fieldsets = (
6         ['student information', {'fields': ['fname', 'lname', 'age']}],
7         ['scholarship information', {'fields': ['student_track']}]
8     )
9
10 admin.site.register(Student, customstudent)
11 admin.site.register(Track)

```

- now refresh your website you will see two table

Django administration

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

Home : Devops > Students > aya rabih

AUTHENTICATION AND AUTHORIZATION

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

DEVOPS

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

Change student

aya rabih HISTORY

student information

Fname: aya

Lname: rabih

Age: 26

scholarship information

Student track: devops1 [edit](#) [+ Add](#)

[Delete](#) [Save and add another](#) [Save and continue editing](#) [SAVE](#)

Django administration

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

Home > Devops > Tracks > Add track

AUTHENTICATION AND AUTHORIZATION

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

DEVOPS

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

Add track

Name:

[Save and add another](#) [Save and continue editing](#) [SAVE](#)

- now we need to add new track with new student
- now we add in admin.py

admin.py - tech - Visual Studio Code

File Edit Selection View Go Run Terminal Help

EXPLORER TECH

settings.py models.py apps.py admin.py

devops > admin.py > ...

```

1  from django.contrib import admin
2  from .models import Student, Track
3  # Register your models here.
4
5  class customstudent(admin.ModelAdmin):
6      fieldsets = (
7          ['student information', {'fields' :['fname','lname','age']}],
8          ['scholarship information', {'fields' :[['student_track']]})
9      )
10
11 class InlineStudent(admin.StackedInline):
12     model = Student
13     extra = 2
14
15 class customTrack(admin.ModelAdmin):
16     inlines = [InlineStudent]
17
18 admin.site.register(Student, customstudent)
19 admin.site.register(Track, customTrack)
20

```

Django administration

WELCOME, ADMIN | VIEW SITE / CHANGE PASSWORD / LOG OUT

Home · Devops · Tracks · Add track

AUTHENTICATION AND AUTHORIZATION

- Groups [+ Add](#)
- Users [+ Add](#)

DEVOPS

- Students [+ Add](#)
- Tracks** [+ Add](#)

Add track

Name:

STUDENTS

Student: #1

Fname:	someone
Lname:	someone
Age:	18

Student: #2

Fname:	someone
Lname:	someone
Age:	18

[+ Add another Student](#)

[Save and add another](#) [Save and continue editing](#) **SAVE**

- now refresh your website
- now we need to know all data for student not just for fname and lname

Django administration

Home · Devops · Students

AUTHENTICATION AND AUTHORIZATION

- Groups [+ Add](#)
- Users [+ Add](#)

DEVOPS

- Students [+ Add](#)
- Tracks [+ Add](#)

Select student to change

Action: Go 0 of 2 selected

<input type="checkbox"/> STUDENT
<input type="checkbox"/> aya rabih
<input type="checkbox"/> mohamed mohamed

2 students

- go to admin.py and write

```

EXPLORER ... settings.py models.py admin.py ...
devops > admin.py > ...
Customtrack ...
> __pycache__
> migrations
> __init__.py
> admin.py
> apps.py
> models.py
> tests.py
> views.py
> tech
db.sqlite3
manage.py

1 from django.contrib import admin
2 from .models import Student, Track
3 # Register your models here.
4
5 class customstudent(admin.ModelAdmin):
6     fieldsets = (
7         ['student information', {'fields': ['fname', 'lname', 'age']}],
8         ['scholarship information', {'fields': ['student_track']}]
9     )
10    list_display = ('fname', 'lname', 'age', 'student_track')
11 class InlineStudent(admin.StackedInline):
12     model = Student
13     extra = 2
14
15 class customTrack(admin.ModelAdmin):
16     inlines = [InlineStudent]
17
18 admin.site.register(Student, customstudent)
19 admin.site.register(Track, customTrack)
20

```

- now refresh your website

Django administration

Home · Devops · Students

AUTHENTICATION AND AUTHORIZATION

- Groups [+ Add](#)
- Users [+ Add](#)

DEVOPS

- Students [+ Add](#)
- Tracks [+ Add](#)

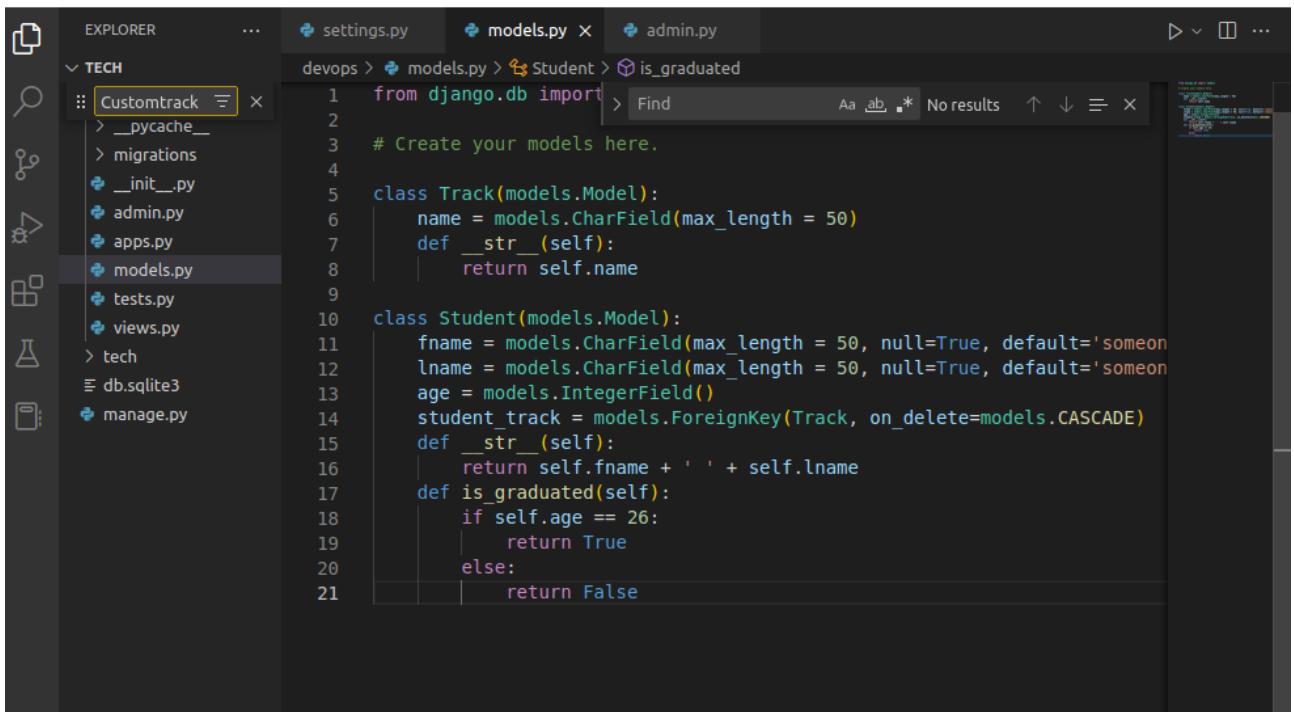
Select student to change

Action: Go 0 of 2 selected

<input type="checkbox"/> FNAME	LNAME	AGE	STUDENT TRACK
<input type="checkbox"/> aya	rabih	26	devops1
<input type="checkbox"/> mohamed	mohamed	26	devops1

2 students

- now we need to add new row to know if this student graduate or no
- go to models.py to add if condition

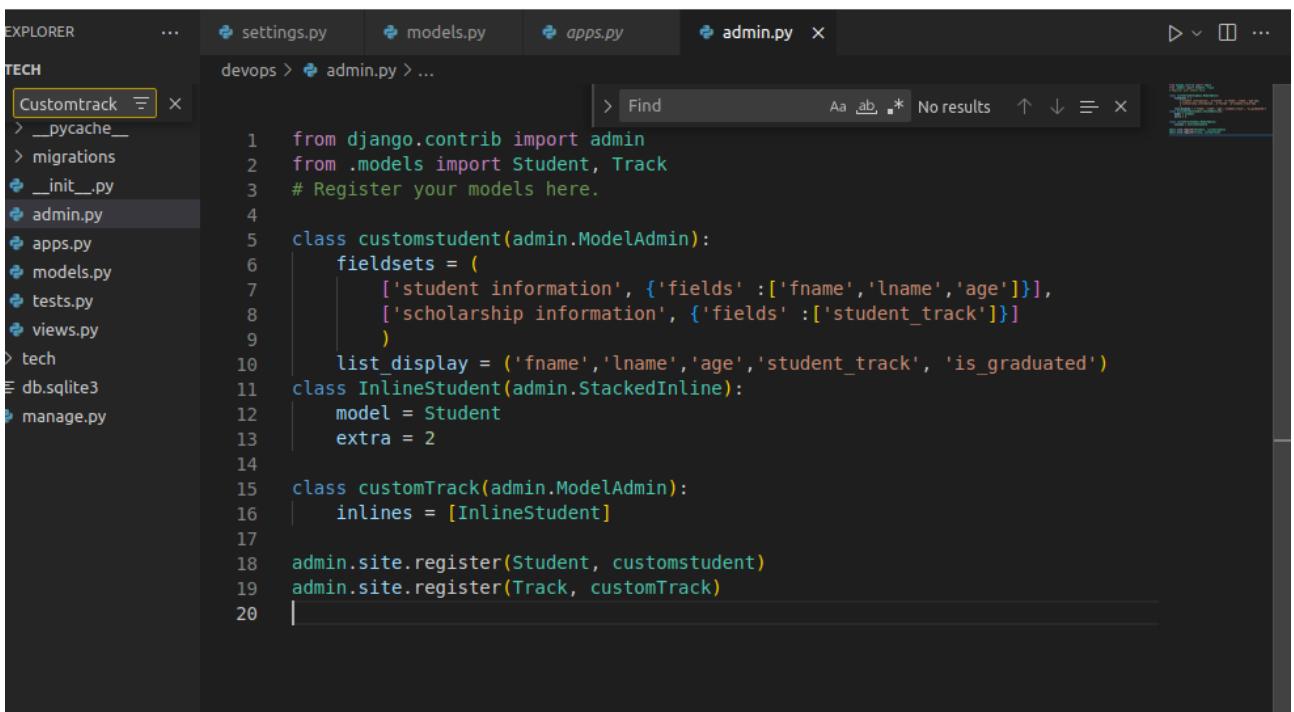


```
from django.db import models
# Create your models here.

class Track(models.Model):
    name = models.CharField(max_length = 50)
    def __str__(self):
        return self.name

class Student(models.Model):
    fname = models.CharField(max_length = 50, null=True, default='someon')
    lname = models.CharField(max_length = 50, null=True, default='someon')
    age = models.IntegerField()
    student_track = models.ForeignKey(Track, on_delete=models.CASCADE)
    def __str__(self):
        return self.fname + ' ' + self.lname
    def is_graduated(self):
        if self.age == 26:
            return True
        else:
            return False
```

- now go to admin.py and add `is_graduated`



```
from django.contrib import admin
from .models import Student, Track
# Register your models here.

class customstudent(admin.ModelAdmin):
    fieldsets = (
        ['student information', {'fields' :['fname','lname','age']}],
        ['scholarship information', {'fields' :['student_track']}]
    )
    list_display = ('fname','lname','age','student_track', 'is_graduated')

class InlineStudent(admin.StackedInline):
    model = Student
    extra = 2

class customTrack(admin.ModelAdmin):
    inlines = [InlineStudent]

admin.site.register(Student, customstudent)
admin.site.register(Track, customTrack)
```

- refresh your website

Django administration

WELCOME, ADMIN | VIEW SITE / CHANGE PASSWORD / LOG OUT

Home > Devops > Students

AUTHENTICATION AND AUTHORIZATION

Groups	+ Add
Users	+ Add
DEVOPS	
Students	+ Add
Tracks	+ Add

Select student to change

[ADD STUDENT +](#)

Action: Go 0 of 2 selected

<input type="checkbox"/>	FNAME	LNAME	AGE	STUDENT TRACK	IS GRADUATED
<input type="checkbox"/>	aya	rabih	26	devops1	True
<input type="checkbox"/>	mohamed	mohamed	25	devops1	False

2 students

- now we need to change true and false to be perfect more
- go to models.py and add
- `is_graduated.short_description = 'Graduated student'`
- `is_graduated.boolean = True`

```

EXPLORER ... settings.py models.py x apps.py views.py admin.py D v I ...
::: Customtrack ...
TECH migrations __init__.py admin.py apps.py models.py tests.py views.py tech db.sqlite3 manage.py
models.py
1 from django.db import model
2
3 # Create your models here.
4
5 class Track(model.Model):
6     name = models.CharField(max_length = 50)
7     def __str__(self):
8         return self.name
9
10 class Student(model.Model):
11     fname = models.CharField(max_length = 50, null=True, default='someone')
12     lname = models.CharField(max_length = 50, null=True, default='someone')
13     age = models.IntegerField()
14     student_track = models.ForeignKey(Track, on_delete=models.CASCADE)
15     def __str__(self):
16         return self.fname + ' ' + self.lname
17     def is_graduated(self):
18         if self.age == 26:
19             return True
20         else:
21             return False
22     is_graduated.short_description = 'Graduated student'
23     is_graduated.boolean = True

```

now we will change from that

Django administration

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

Home > Devops > Students

AUTHENTICATION AND AUTHORIZATION

Groups [+ Add](#)

Users [+ Add](#)

DEVOPS

Students [+ Add](#)

Tracks [+ Add](#)

Select student to change

Action: Go 0 of 2 selected

<input type="checkbox"/>	FNAME	LNAME	AGE	STUDENT TRACK	IS GRADUATED
<input type="checkbox"/>	aya	rabih	26	devops1	True
<input type="checkbox"/>	mohamed	mohamed	25	devops1	False

2 students

[ADD STUDENT +](#)

Django administration

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

Home > Devops > Students

AUTHENTICATION AND AUTHORIZATION

Groups [+ Add](#)

Users [+ Add](#)

DEVOPS

Students [+ Add](#)

Tracks [+ Add](#)

Select student to change

Action: Go 0 of 2 selected

<input type="checkbox"/>	FNAME	LNAME	AGE	STUDENT TRACK	GRADUATED STUDENT
<input type="checkbox"/>	aya	rabih	26	devops1	✓
<input type="checkbox"/>	mohamed	mohamed	25	devops1	✗

2 students

[ADD STUDENT +](#)

- to be that
- now we need to know where is this path for this project
- `python3 manage.py shell`
- `import django`
- `django.__path__`
- `['/usr/lib/python3/dist-packages/django']`

```

aya@aya-Latitude-E6540:~/tech$ python3 manage.py shell
Python 3.10.6 (main, Nov  2 2022, 18:53:38) [GCC 11.3.0] on linux
Type "help", "copyright", "credits" or "license" for more information.
(InteractiveConsole)
>>> import django
>>> django.__path__
Traceback (most recent call last):
  File "/usr/lib/python3.10/code.py", line 90, in runcode
    exec(code, self.locals)
  File "<console>", line 1, in <module>
AttributeError: module 'django' has no attribute 'path'
>>> django.__path__
['/usr/lib/python3/dist-packages/django']
>>> 

```

- now we nwws to change name of our dango administratour
- to our name

Django administration

WELCOME, ADMIN. VIEW SITE / CHANGE PASSWORD / LOG OUT

Home > Devops > Students

AUTHENTICATION AND AUTHORIZATION

Groups	+ Add
Users	+ Add
DEVOPS	
Students	+ Add
Tracks	+ Add

Select student to change

Action: ----- Go 0 of 4 selected

<input type="checkbox"/>	FNAME	LNAME	AGE	STUDENT TRACK	GRADUATED STUDENT
<input type="checkbox"/>	hossam	mostafa	40	devops	✓
<input type="checkbox"/>	sayed	mohamed	20	devops	✗
<input type="checkbox"/>	aya	rabih	26	devops1	✓
<input type="checkbox"/>	mohamed	mohamed	25	devops1	✗

4 students

Inspector

DOM Tree

Elements

Network

Style Editor

Performance

Memory

Storage

Accessibility

Application

0 6

Search HTML

<!- Header-->

<div id="header"> flex

<div id="branding">

<h1 id="site-name">

Django administration

</h1>

</div>

<!-- END Header-->

<div class="breadcrumbs">

body.app-devops.model-student.change-li... > div#container > div#header > div#branding > h1#site-name > a >

Filter Styles

Layout Computed Changes Compatibility

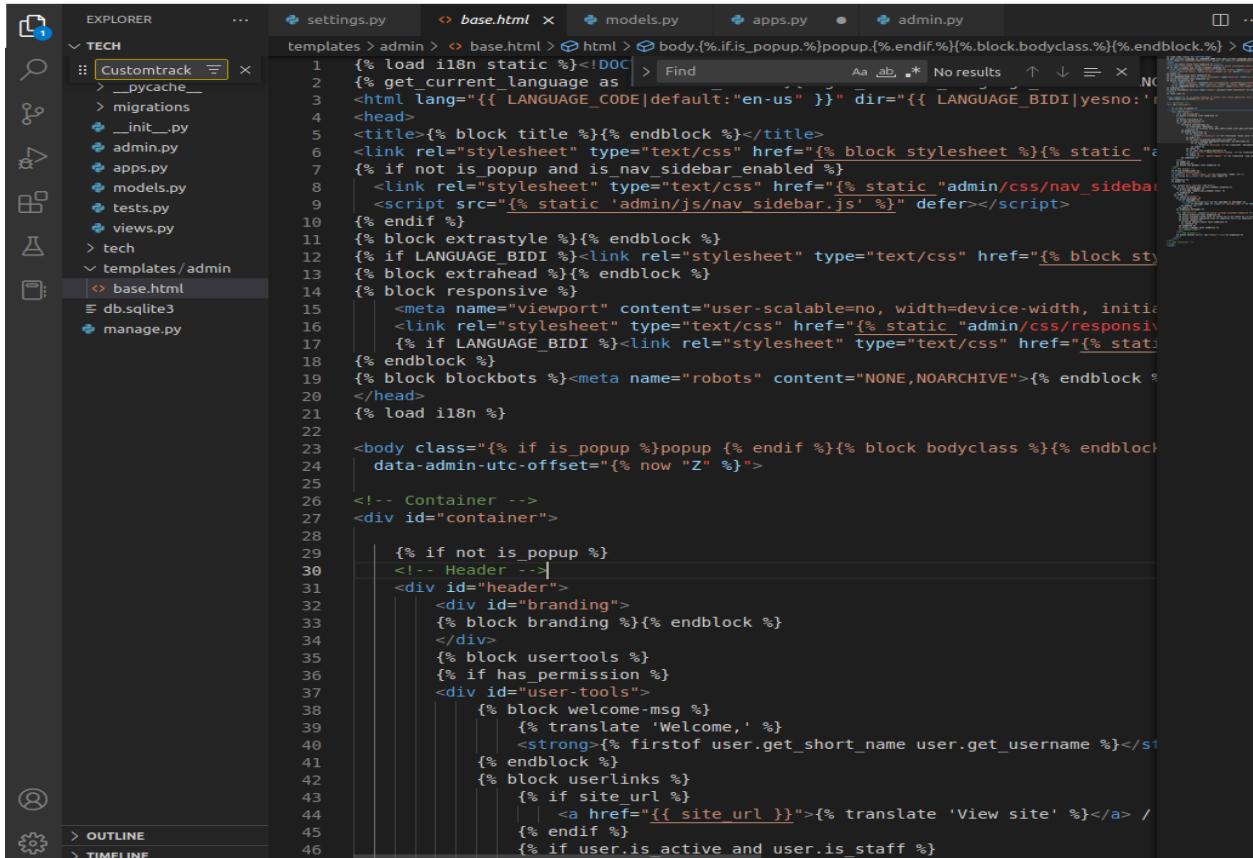
Flex item of div#header

basis/final

Content Size 251

Final Size 251

- now we will change name
- by remove branding and add this line
- <h1 style = 'color:beige'>DEVOPS WITH AYA RABIH </h1>

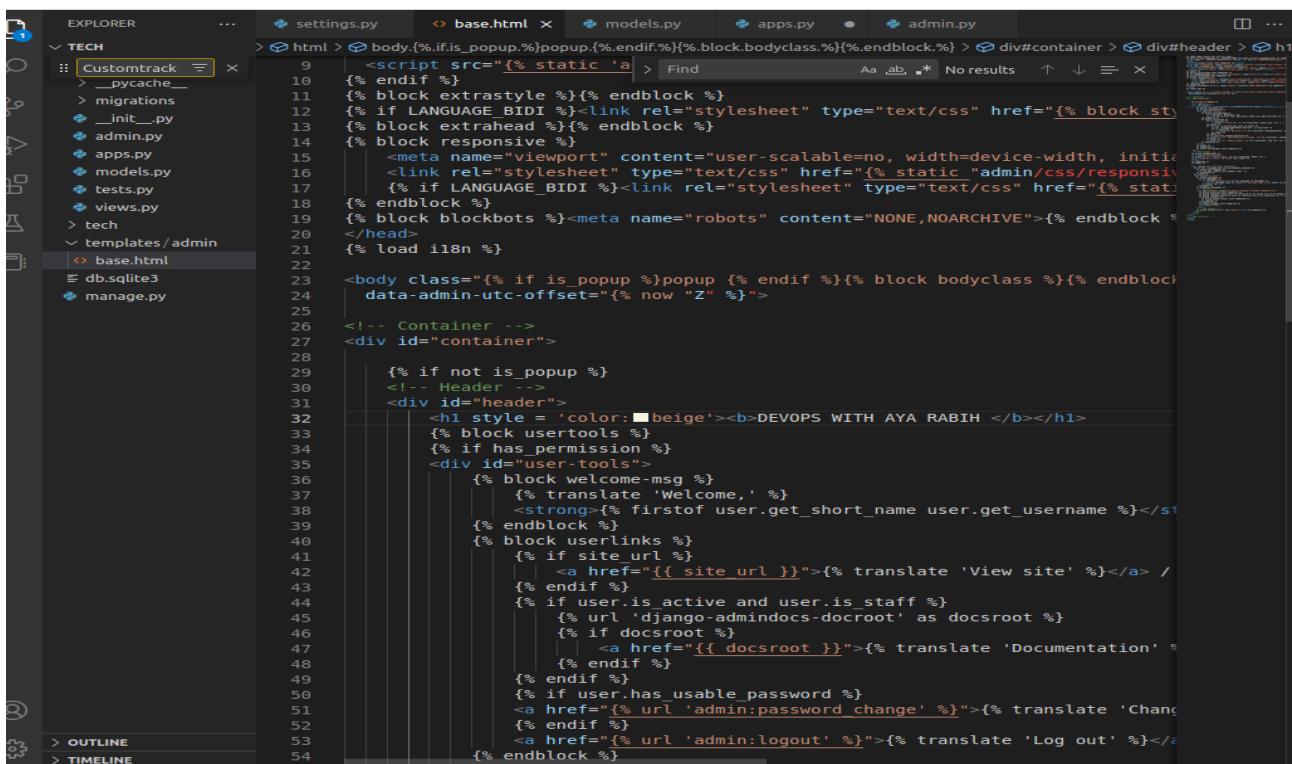


```

EXPLORER ... settings.py base.html models.py apps.py admin.py
templates/admin/base.html
1  {% load i18n static %}<!DOCTYPE html>
2  {% get_current_language as LANGUAGE_CODE %}
3  <html lang="{{ LANGUAGE_CODE|default:"en-us" }}>
4  |<head>
5  |<title>{{ block title }}{{ endblock }}</title>
6  |<link rel="stylesheet" type="text/css" href="{{ block stylesheet }}{{ static "admin/css/nav_sidebar.css" }}>
7  |<% if not is_popup and is_nav_sidebar_enabled %>
8  |<link rel="stylesheet" type="text/css" href="{{ static "admin/css/nav_sidebar.css" }}>
9  |<script src="{{ static 'admin/js/nav_sidebar.js' }}" defer></script>
10 |<% endif %>
11 |<% block extrastyle %>{{ endblock }}<% if LANGUAGE_BIDI %><link rel="stylesheet" type="text/css" href="{{ block extrahead }}>
12 |<% block extrahead %>{{ endblock }}<% block responsive %>
13 |<meta name="viewport" content="user-scalable=no, width=device-width, initial-scale=1, shrink-to-fit=no">
14 |<link rel="stylesheet" type="text/css" href="{{ static "admin/css/responsive.css" }}>
15 |<% endif %>
16 |<% if LANGUAGE_BIDI %><link rel="stylesheet" type="text/css" href="{{ static "admin/css/rtl.css" }}>
17 |<% endif %>
18 |<% block blockbots %><meta name="robots" content="NONE,NOARCHIVE">{{ endblock }}<% if LANGUAGE_BIDI %><meta name="robots" content="NONE,NOARCHIVE">{{ endblock }}<% endif %>
19 |<% block blockbots %><meta name="robots" content="NONE,NOARCHIVE">{{ endblock }}<% if LANGUAGE_BIDI %><meta name="robots" content="NONE,NOARCHIVE">{{ endblock }}<% endif %>
20 |</head>
21 |<% load i18n %>
22 |
23 |<body class="{{ if is_popup }%>popup {{ endif }}{{ block bodyclass }}{{ endblock }}>
24 |<div id="container">
25 |
26 |<!-- Container -->
27 |<div id="container">
28 |
29 |<% if not is_popup %>
30 |<!-- Header -->
31 |<div id="header">
32 |<div id="branding">
33 |<% block branding %>{{ endblock }}</div>
34 |<% block usertools %>
35 |<% if has_permission %>
36 |<div id="user-tools">
37 |<% block welcome-msg %>
38 |<% translate 'Welcome,' %>
39 |<strong>{{ firstof user.get_short_name user.get_username }}</strong>
40 |<% endif %>
41 |<% block userlinks %>
42 |<% if site_url %>
43 |<a href="{{ site_url }}>{{ translate 'View site' }}</a> /<% endif %>
44 |<% if user.is_active and user.is_staff %>
45 |<% url 'django-admindocs-docroot' as docsroot %>
46 |<% if docsroot %>
47 |<a href="{{ docsroot }}>{{ translate 'Documentation' }}</a> /<% endif %>
48 |<% endif %>
49 |<% if user.has_usable_password %>
50 |<a href="{{ url 'admin:password_change' }}>{{ translate 'Change password' }}</a> /<% endif %>
51 |<% if user.is_active and user.is_staff %>
52 |<a href="{{ url 'admin:logout' }}>{{ translate 'Log out' }}</a> /<% endif %>
53 |<% endif %>
54 |<% endblock %>

```

- now save and refresh your web app



```

EXPLORER ... settings.py base.html models.py apps.py admin.py
templates/admin/base.html
1  <script src="{{ static 'a' }}>
2  |<% endif %>
3  |<% block extrastyle %>{{ endblock }}<% if LANGUAGE_BIDI %><link rel="stylesheet" type="text/css" href="{{ block stylesheet }}>
4  |<% block extrahead %>{{ endblock }}<% block responsive %>
5  |<meta name="viewport" content="user-scalable=no, width=device-width, initial-scale=1, shrink-to-fit=no">
6  |<link rel="stylesheet" type="text/css" href="{{ static "admin/css/responsive.css" }}>
7  |<% endif %>
8  |<% if LANGUAGE_BIDI %><link rel="stylesheet" type="text/css" href="{{ static "admin/css/rtl.css" }}>
9  |<% endif %>
10 |<% block blockbots %><meta name="robots" content="NONE,NOARCHIVE">{{ endblock }}<% if LANGUAGE_BIDI %><meta name="robots" content="NONE,NOARCHIVE">{{ endblock }}<% endif %>
11 |<% block blockbots %><meta name="robots" content="NONE,NOARCHIVE">{{ endblock }}<% if LANGUAGE_BIDI %><meta name="robots" content="NONE,NOARCHIVE">{{ endblock }}<% endif %>
12 |</head>
13 |<% load i18n %>
14 |
15 |<body class="{{ if is_popup }%>popup {{ endif }}{{ block bodyclass }}{{ endblock }}>
16 |<div id="container">
17 |
18 |<!-- Container -->
19 |<div id="container">
20 |
21 |<% if not is_popup %>
22 |<!-- Header -->
23 |<div id="header">
24 |<h1 style = 'color:beige'><b>DEVOPS WITH AYA RABIH </b></h1>
25 |<% block usertools %>
26 |<% if has_permission %>
27 |<div id="user-tools">
28 |<% block welcome-msg %>
29 |<% translate 'Welcome,' %>
30 |<strong>{{ firstof user.get_short_name user.get_username }}</strong>
31 |<% endif %>
32 |<% block userlinks %>
33 |<% if site_url %>
34 |<a href="{{ site_url }}>{{ translate 'View site' }}</a> /<% endif %>
35 |<% if user.is_active and user.is_staff %>
36 |<% url 'django-admindocs-docroot' as docsroot %>
37 |<% if docsroot %>
38 |<a href="{{ docsroot }}>{{ translate 'Documentation' }}</a> /<% endif %>
39 |<% endif %>
40 |<% if user.has_usable_password %>
41 |<a href="{{ url 'admin:password_change' }}>{{ translate 'Change password' }}</a> /<% endif %>
42 |<% if user.is_active and user.is_staff %>
43 |<a href="{{ url 'admin:logout' }}>{{ translate 'Log out' }}</a> /<% endif %>
44 |<% endif %>
45 |<% endblock %>

```

DEVOPS WITH AYA RABIH

WELCOME, ADMIN. VIEW SITE / CHANGE PASSWORD / LOG OUT

Home > Devops > Students

AUTHENTICATION AND AUTHORIZATION				
Groups	+ Add			
Users	+ Add			
DEVOPS				
Students	+ Add			
Tracks	+ Add			

Select student to change

Action: 0 of 4 selected

<input type="checkbox"/>	FNAME	LNAME	AGE	STUDENT TRACK	GRADUATED STUDENT
<input type="checkbox"/>	hossam	mostafa	40	devops	✓
<input type="checkbox"/>	sayed	mohamed	20	devops	✗
<input type="checkbox"/>	aya	rabih	26	devops1	✓
<input type="checkbox"/>	mohamed	mohamed	25	devops1	✗

4 students

- now we need to create filter it will help us to search for all track we have
- go to admin.py and add
- `list_filter = ['student_track']`

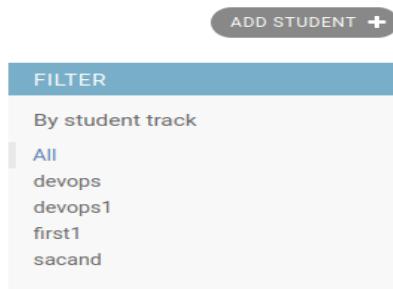
```

EXPLORER ... settings.py base.html models.py apps.py admin.py X
devops > admin.py > ...
Find Aa ab * No results ↑ ↓ ⌂ X
from django.contrib import admin
from .models import Student, Track
# Register your models here.

class customstudent(admin.ModelAdmin):
    fieldsets = (
        ['student information', {'fields': ['fname', 'lname', 'age']}],
        ['scholarship information', {'fields': ['student_track']}]
    )
    list_display = ('fname', 'lname', 'age', 'student_track', 'is_graduated')
    list_filter = ['student_track']

```

- now refresh your website



- now we need to add line to search for people name it will help us to search easy
- go to admin.py and add
- `search_fields = ['fname']`

```

EXPLORER ... settings.py base.html models.py apps.py admin.py
TECH Customtrack ...
    > __pycache__
    > migrations
    __init__.py
    admin.py
    apps.py
    models.py
    tests.py
    views.py
    > tech
    < templates/admin
        > base.html
> devops > admin.py > ...
> Find Aa ab,* No results ↑ ↓ ≡ ×
1 from django.contrib import admin
2 from .models import Student, Track
3 # Register your models here.
4
5 class customstudent(admin.ModelAdmin):
6     fieldsets = (
7         ['student information', {'fields': ['fname', 'lname', 'age']}],
8         ['scholarship information', {'fields': ['student_track']}]
9     )
10    list_display = ('fname', 'lname', 'age', 'student_track', 'is_graduated')
11    list_filter = ['student_track']
12    search_fields = ['fname']

```

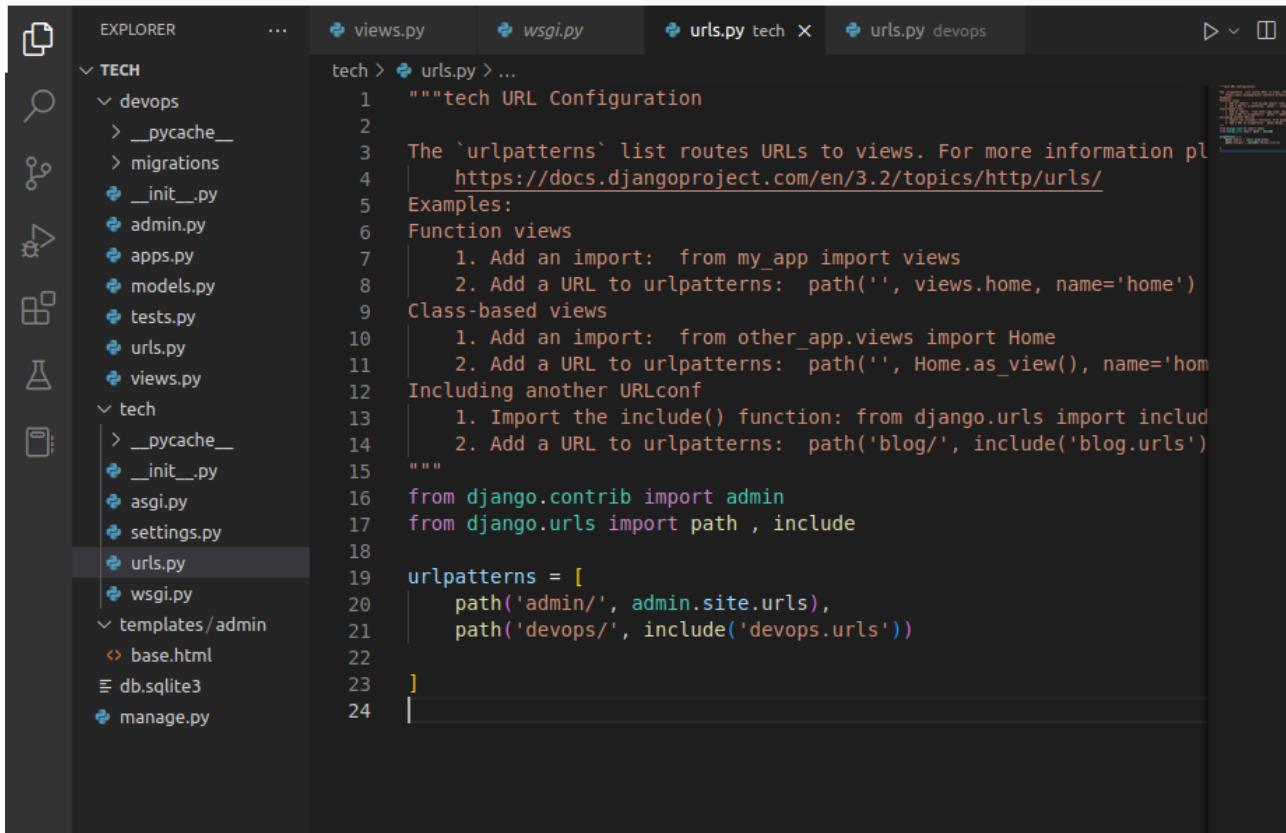
- now refresh your web

FNAME	LNAME	AGE	STUDENT TRACK	GRADUATED STUDENT
amr	mohamed	29	sacand	✓
islam	salah	30	sacand	✓
amena	islam	28	first1	✓
asmaa	amin	27	first1	✓
hossam	mostafa	40	devops	✓
sayed	mohamed	20	devops	✗
aya	rabih	26	devops1	✓
mohamed	mohamed	25	devops1	✗

8 students

- now we want to make page home to our website and want to write like welcome to my website this create by aya rabih

- and also we want to change name of urls website
- go to urls.py and add that



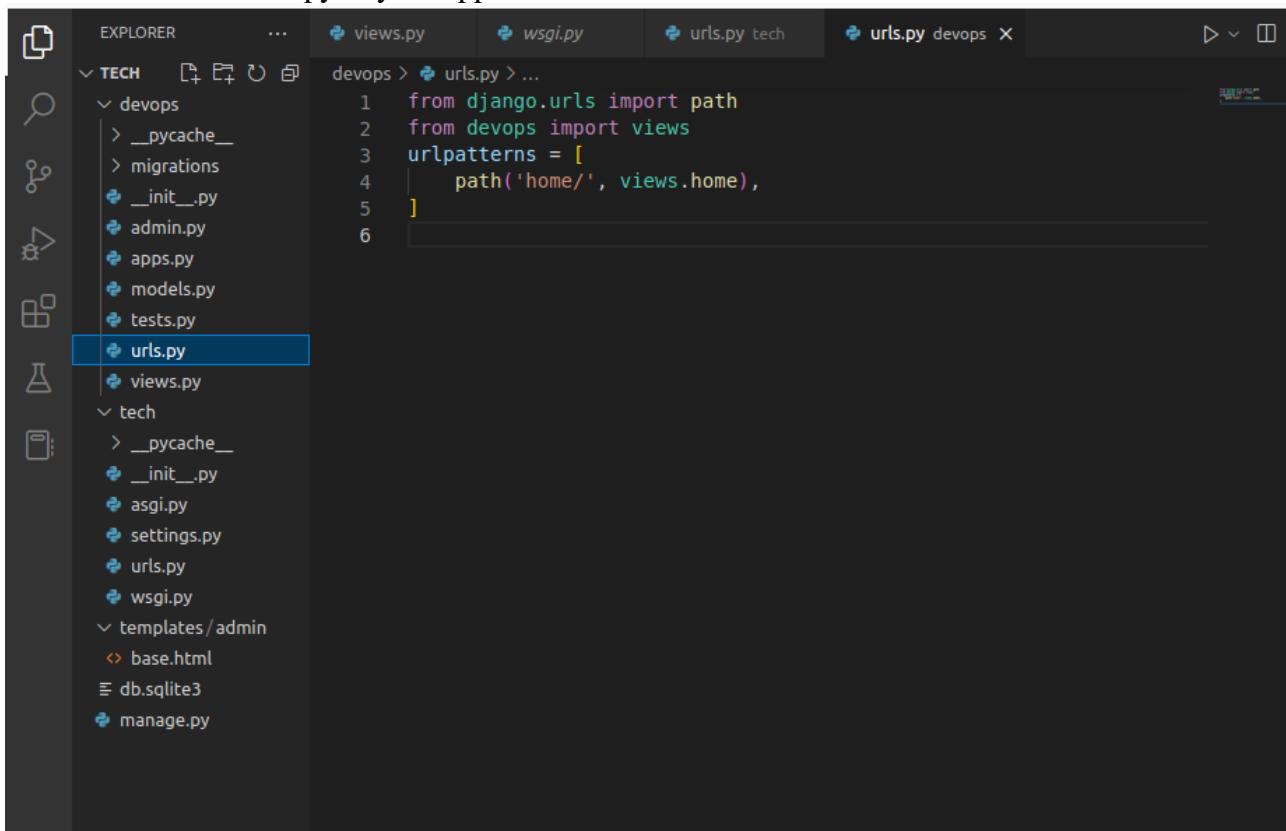
```

EXPLORER ... views.py wsgi.py urls.py tech X urls.py devops D v I

tech > urls.py > ...
1 """tech URL Configuration
2
3 The `urlpatterns` list routes URLs to views. For more information pl
4 | https://docs.djangoproject.com/en/3.2/topics/http/urls/
5 Examples:
6 Function views
7 | 1. Add an import: from my_app import views
8 | 2. Add a URL to urlpatterns: path('', views.home, name='home')
9 Class-based views
10 | 1. Add an import: from other_app.views import Home
11 | 2. Add a URL to urlpatterns: path('', Home.as_view(), name='hom
12 Including another URLconf
13 | 1. Import the include() function: from django.urls import includ
14 | 2. Add a URL to urlpatterns: path('blog/', include('blog.urls'))
15 """
16 from django.contrib import admin
17 from django.urls import path , include
18
19 urlpatterns = [
20 |     path('admin/', admin.site.urls),
21 |     path('devops/', include('devops.urls'))
22 ]
23
24

```

- now create urls.py in your app and add that



```

EXPLORER ... views.py wsgi.py urls.py tech X urls.py devops X D v I

devops > urls.py > ...
1 from django.urls import path
2 from devops import views
3 urlpatterns = [
4 |     path('home/', views.home),
5 ]
6

```

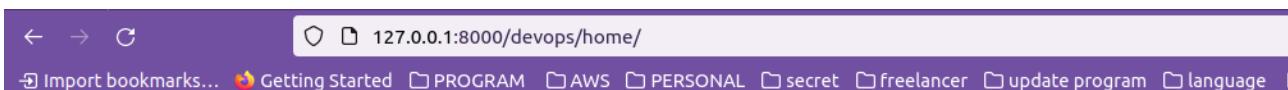
- now go to views.py

The screenshot shows the VS Code interface with the following details:

- Explorer View:** Shows a tree structure of files under a folder named "TECH".
 - Under "devops": __pycache__, migrations, __init__.py, admin.py, apps.py, models.py, tests.py, urls.py, views.py (selected).
 - Under "tech": __pycache__, __init__.py, asgi.py, settings.py, urls.py, wsgi.py.
 - Under "templates/admin": base.html.
- Code Editor:** The "views.py" file is open, showing the following Python code:

```
from django.shortcuts import render
from django.http import HttpResponse
# Create your views here.
def home(request):
    return HttpResponse("<h1> welcome to my website this create by aya rabih </h1>")
```
- Status Bar:** Shows tabs for "views.py", "wsgi.py", "urls.py tech", and "urls.py devops".

- now write your app name
- like that
- `http://127.0.0.1:8000/devops/home/`
- refresh now



welcome to my website this create by aya rabih

- NOW we need to make new path to student and when we write there id give us there name so
- go to urls.py and write
- `path('student/<st_id>', views.getStudent)`

The screenshot shows the VS Code interface with the following details:

- EXPLORER** sidebar: Shows the project structure under the **TECH** folder, including **devops** and **tech** subfolders.
- urls.py** file in the **devops** directory is the active tab, containing the following code:

```
1 from django.urls import path
2 from devops import views
3 urlpatterns = [
4     path('home/', views.home),
5     path('student/<st_id>', views.getStudent)
6 ]
```

- now go to view.py
- and add this
- **def getStudent(request, st_id):**
- **st = Student.objects.get(id = st_id)**
- **return HttpResponse(st.fname)**

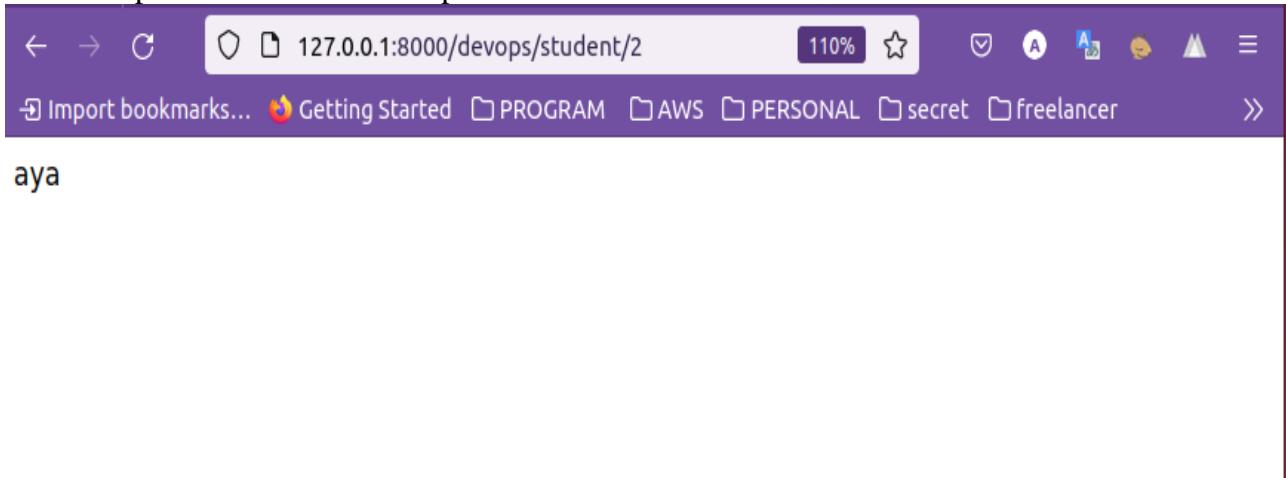
The screenshot shows the VS Code interface with the following details:

- EXPLORER** sidebar: Shows the project structure under the **TECH** folder, including **devops** and **tech** subfolders.
- views.py** file in the **devops** directory is the active tab, containing the following code:

```
1 from django.shortcuts import render
2 from django.http import HttpResponseRedirect
3 from devops.models import Student, Track
4 # Create your views here.
5 def home(request):
6     return HttpResponseRedirect("<h1> welcome to my website this create by")
7
8 def getStudent(request, st_id):
9     st = Student.objects.get(id = st_id)
10    return HttpResponseRedirect(st.fname)
```

now search with id

- http://127.0.0.1:8000/devops/student/2



- now we will make another path
- go to urls.py
- `path('all', views.getAllStudents)`

```

File Edit Selection View Go Run Terminal Help

EXPLORER      ...      views.py      urls.py tech      students.html      urls.py devops X
devops > urls.py > ...
1   from django.urls import path
2   from devops import views
3   urlpatterns = [
4       path('home/', views.home),
5       path('student/<st_id>', views.getStudent),
6       path('all', views.getAllStudents)
7   ]
8 ]
9

```

The screenshot shows a Visual Studio Code interface. The left sidebar has a tree view with a folder named 'TECH' expanded, showing subfolders 'devops' and 'tech', and files like '_init_.py', 'admin.py', 'apps.py', 'models.py', 'tests.py', 'urls.py', 'views.py', and 'settings.py'. The main editor area shows the 'urls.py' file for the 'tech' app. The code defines three URL patterns using the `path` function from `django.urls`. The first two patterns map to the 'home' and 'student/st_id' views respectively. The third pattern, 'all', maps to the `getALLStudents` view.

- now go to views py

- add `def getALLStudents(request):`
- `all_students = Student.objects.all()`
- `context = {'students': all_students}`
- `return render(request,"devops/students.html", context)`

The screenshot shows a code editor interface with two tabs open: `views.py` and `urls.py`. The `views.py` tab contains Python code for a Django application named 'tech'. It includes imports for `render` and `HttpResponse` from Django, and for `Student` and `Track` models from the `devops.models` module. It defines three functions: `home`, `getStudent`, and `getALLStudents`. The `getALLStudents` function retrieves all students and passes them to a template. The `urls.py` tab shows URL patterns for the 'tech' application, including a path to the 'students' view.

```

File Edit Selection View Go Run Terminal Help
EXPLORER ... views.py X urls.py tech students.html ● urls.py devops
devops > views.py > ...
1   from django.shortcuts import render
2   from django.http import HttpResponseRedirect
3   from devops.models import Student, Track
4   # Create your views here.
5   def home(request):
6       return HttpResponseRedirect("<h1> welcome to my website this create by aya rabih </h1>")
7
8   def getStudent(request, st_id):
9       st = Student.objects.get(id = st_id)
10      return HttpResponseRedirect(st.fname)
11
12  def getALLStudents(request):
13      all_students = Student.objects.all()
14      context = {'students': all_students}
15      return render(request,"devops/students.html", context)
16

```

- now create students . Html

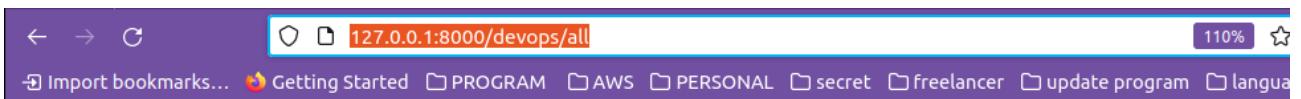
The screenshot shows a code editor interface with several tabs open. The `students.html` tab is active and displays an HTML table template. The template starts with an `<h1>ALL Students</h1>` heading and a `<table border="1">` tag. Inside the table, there is a header row with columns for First Name, Last Name, Age, Track, and Graduated. Below the header, there is a loop that iterates over a list of students, creating a row for each with cells containing the student's first name, last name, age, track, and graduation status. The `manage.py` tab is also visible in the background.

```

File Edit Selection View Go Run Terminal Help
EXPLORER ... views.py db.sqlite3 students.html ● urls.py
templates > devops > students.html > table > table
1   <h1>ALL Students</h1>
2   <table border="1">
3       <th> First Name</th> <th>Last Name </th> <th>Age</th> <th>Track</th> <th>Graduated</th>
4       {% for st in students %}
5           <tr>
6               <td>{{st.fname}}</td>
7               <td>{{st.lname}}</td>
8               <td>{{st.age}}</td>
9               <td>{{st.student_track}}</td>
10              <td>{{st.is_graduated}}</td>
11          </tr>
12      {% endfor %}
13  </table>
14

```

- refresh your web site now
- `http://127.0.0.1:8000/devops/all`



- now need to make hyper link when we click to any name give us all details
- we will create student_detials.html and write that

The screenshot shows a code editor interface with the following structure:

- EXPLORER**: Shows the project structure:
 - TECH
 - devops
 - __pycache__
 - migrations
 - __init__.py
 - admin.py
 - apps.py
 - models.py
 - tests.py
 - urls.py
 - views.py
 - tech
 - __pycache__
 - __init__.py
 - asgi.py
 - settings.py
 - urls.py
 - wsgi.py
 - templates
 - admin
 - base.html
 - devops
 - base.html
 - student_details.h...
 - students.html
 - views.py**, **urls.py**, **students.html**, **student_details.html**, **base.html**

- now we will create base.html and write this block

The screenshot shows the VS Code interface with the following details:

- EXPLORER:** Shows a project structure under 'TECH' named 'devops'. It includes subfolders like 'migrations', 'models.py', 'tests.py', 'urls.py', 'views.py', and 'tech' which contains 'admin', 'base.html', 'devops', 'students.html', 'db.sqlite3', and 'manage.py'.
- RIGHT PANEL:** Displays the content of 'base.html' located at 'templates > devops > base.html'. The code is as follows:

```

1  {% block title_block %}
2  {% endblock %}
3
4
5
6
7  {% block main_block %}
8
9
10 {% endblock %}

```

- now we will add this block to student.html

The screenshot shows the VS Code interface with the following details:

- EXPLORER:** Shows a project structure under 'TECH' named 'devops'. It includes subfolders like 'migrations', 'models.py', 'tests.py', 'urls.py', and 'views.py' (which is currently selected).
- RIGHT PANEL:** Displays the content of 'views.py' located at 'devops > views.py > ...'. The code is as follows:

```

1  from django.shortcuts import render
2  from django.http import HttpResponseRedirect
3  from devops.models import Student, Track
4  # Create your views here.
5  def home(request):
6      return HttpResponseRedirect("<h1> welcome to my website this create by aya rabih")
7
8  def getStudent(request, st_id):
9      st = Student.objects.get(id = st_id)
10     context = {'student': st}
11     return render(request, 'devops/student_details.html', context)
12
13 def getAllStudents(request):
14     all_students = Student.objects.all()
15     context = {'students': all_students}
16     return render(request, "devops/students.html", context)
17
18

```

and now add details in views.py

- now refresh your website



ALL Students

First Name	Last Name	Age	Track	Graduated student
mohamed	mohamed	25	devops1	False
aya	rabih	26	devops1	True
sayed	mohamed	20	devops	False
hosam	mostafa	40	devops	True
asmaa	amin	27	first1	True
amena	islam	28	first1	True
islam	salah	30	sacand	True
amr	mohamed	29	sacand	True

welcome: sayed mohamed

First Name: sayed

Last Name: mohamed

Age: 20

Track: devops

- now we need to make a button when we click to here back us to start page
- ` click Back `

The screenshot shows the VS Code interface with the following details:

- File Explorer:** Shows the project structure under the 'TECH' folder:
 - devops
 - migrations
 - __init__.py
 - admin.py
 - apps.py
 - models.py
 - tests.py
 - urls.py
 - views.py
 - tech
 - __init__.py
 - asgi.py
 - settings.py
 - urls.py
 - wsgi.py
 - templates
 - admin
 - base.html
 - devops
 - base.html
 - student_details.h...
 - students.html
- Terminal:** Shows the command `python manage.py runserver` being run.
- Code Editor:** Displays the contents of the `student_details.html` template file:

```
1  {% extends 'devops/base.html' %} 2 3 4 5  <h1>welcome: {{student.fname}} {{student.lname}}</h1> 6  <br /> 7  <a href="/devops/all"> click Back </a> 8  <br /> 9  {% endblock %} 10 11 12 13 14 15 16 17 18 19 20 21 22
```

- now refresh your website



welcome: aya rabih

[click Back](#)

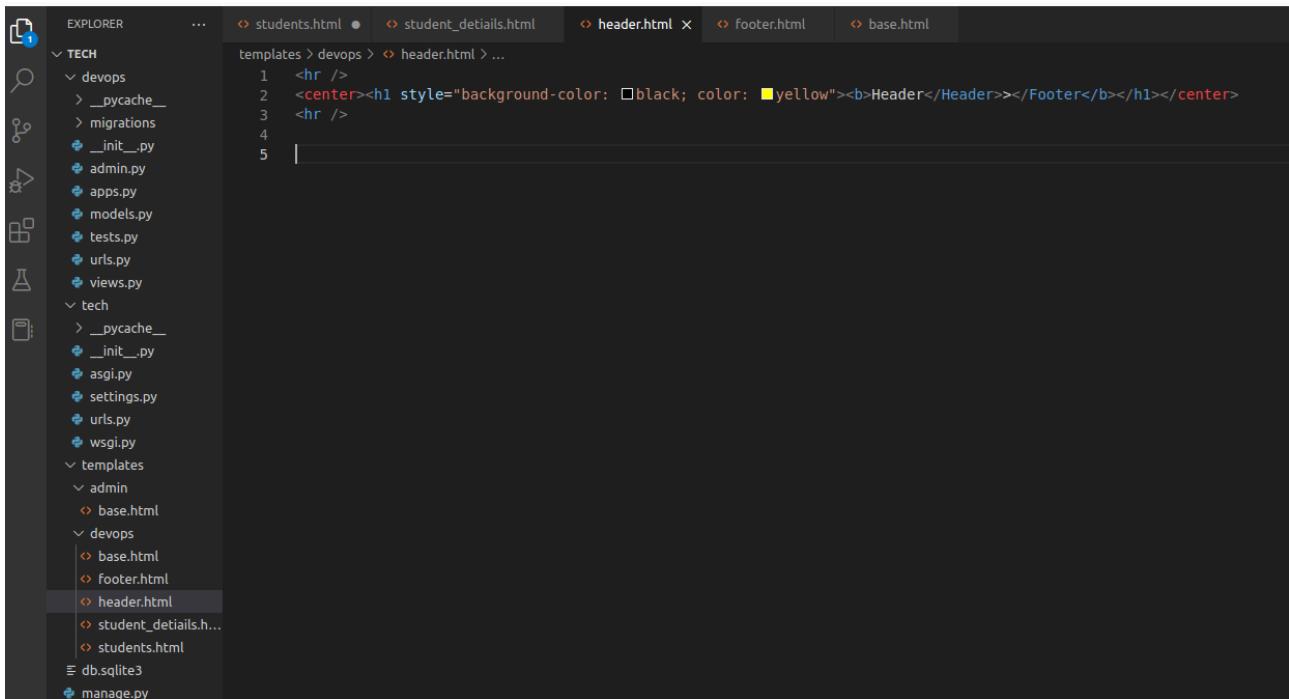
First Name: aya

Last Name: rabih

Age: 26

Track: devops1

- now we need to make header and footer



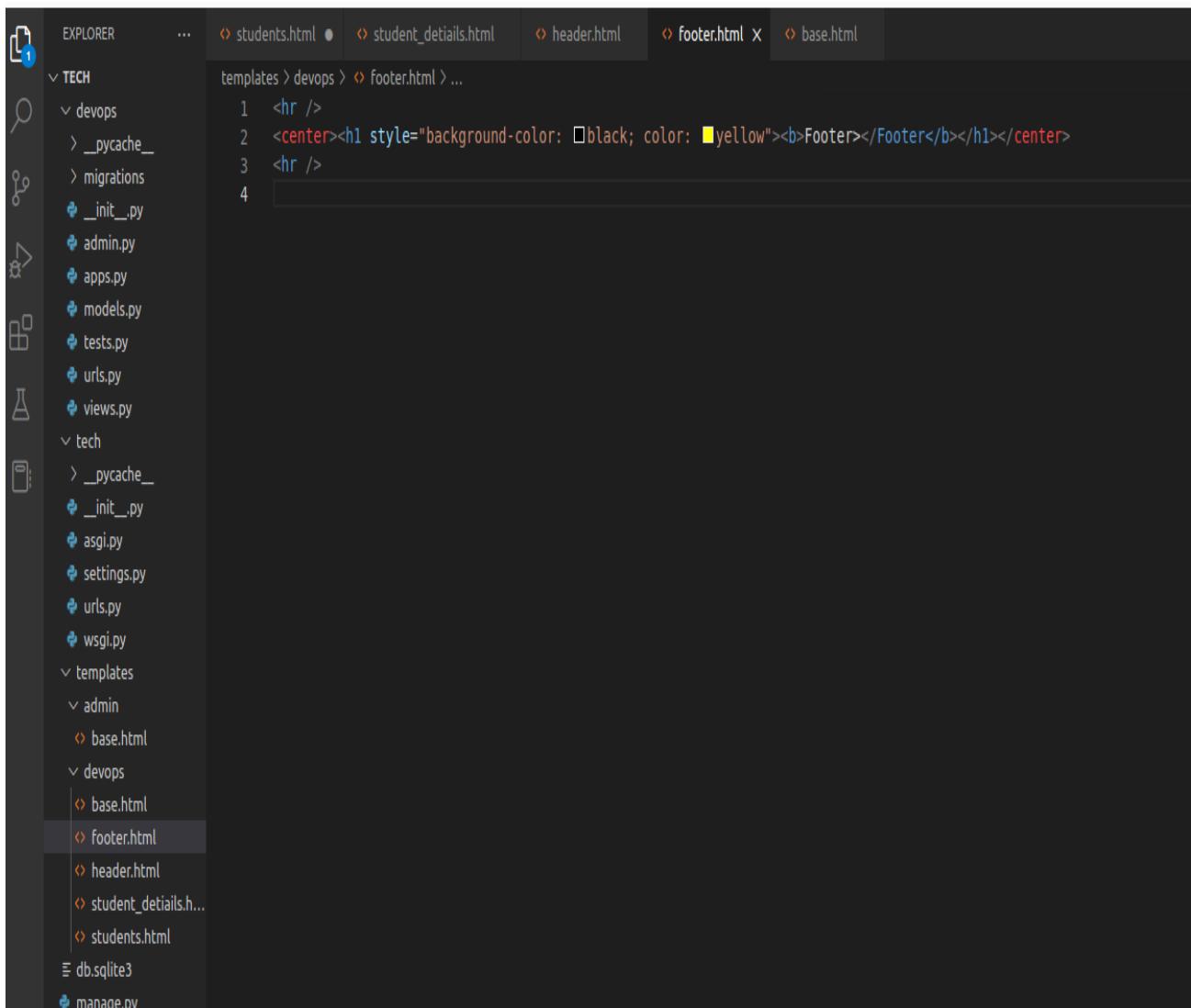
```

EXPLORER      ...
              ...
              > students.html  ...
              > student_details.html
              > header.html X
              > footer.html
              > base.html

templates > devops > > header.html > ...
1   <hr />
2   <center><h1 style="background-color: black; color: yellow"><b>Header</b></h1></center>
3   <hr />
4
5

```

- and make footer also



```

EXPLORER      ...
              ...
              > students.html  ...
              > student_details.html
              > header.html
              > footer.html X
              > base.html

templates > devops > > footer.html > ...
1   <hr />
2   <center><h1 style="background-color: black; color: yellow"><b>Footer</b></h1></center>
3   <hr />
4

```

- and include them in base.html

The screenshot shows a code editor interface with the following details:

- EXPLORER** sidebar: Shows the project structure with folders like TECH, devops, migrations, models.py, tests.py, urls.py, views.py, and tech. Inside tech, there are __pycache__, __init__.py, asgi.py, settings.py, urls.py, and wsgi.py.
- base.html** file content (shown in the main editor area):

```

1  {% include 'devops/header.html' %}
2  {% block title_block %}
3
4  {% endblock %}
5
6
7
8
9  {% block main_block %}
10
11 {% endblock %}
12 {% include 'devops/footer.html' %}
13

```

- now refresh your website

The screenshot shows a web browser displaying the following content:

Header>

ALL Students

First Name	Last Name	Age	Track	Graduated
mohamed	mohamed	25	devops1	False
aya	rabih	26	devops1	True
sayed	mohamed	20	devops	False
hossam	mostafa	40	devops	True
asmaa	amin	27	first1	True
amena	islam	28	first1	True
islam	salah	30	sacand	True
amr	mohamed	29	sacand	True

Footer>

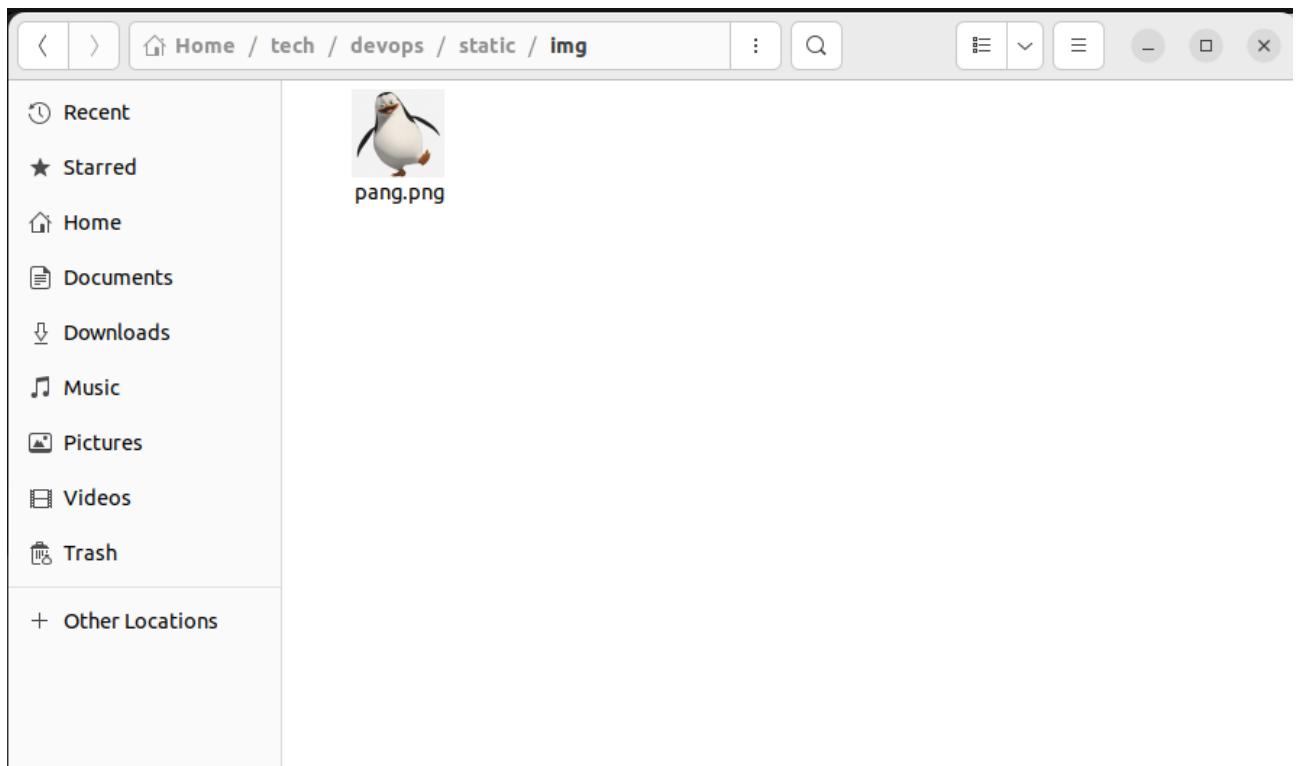
- we need to add static file so we will create folder under our app and we will name it static and inside it create folder named img

```
108
109
110 # Internationalization
111 # https://docs.djangoproject.com/en/3.2/topics/i18n/
112
113 LANGUAGE_CODE = 'en-us'
114
115 TIME_ZONE = 'UTC'
116
117 USE_I18N = True
118
119 USE_L10N = True
120
121 USE_TZ = True
122
123
124 # Static files (CSS, JavaScript, Images)
125 # https://docs.djangoproject.com/en/3.2/howto/static-files/
126
127 STATIC_URL = '/static/'
128
129 # Default primary key field type
130 # https://docs.djangoproject.com/en/3.2/ref/settings/#default-auto-field
131
132 DEFAULT_AUTO_FIELD = 'django.db.models.BigAutoField'
133
```

- now change static to

```
108
109
110 # Internationalization
111 # https://docs.djangoproject.com/en/3.2/topics/i18n/
112
113 LANGUAGE_CODE = 'en-us'
114
115 TIME_ZONE = 'UTC'
116
117 USE_I18N = True
118
119 USE_L10N = True
120
121 USE_TZ = True
122
123
124 # Static files (CSS, JavaScript, Images)
125 # https://docs.djangoproject.com/en/3.2/howto/static-files/
126
127 STATIC_URL = os.path.join(BASE_DIR, 'devops/static/')
128
129 # Default primary key field type
130 # https://docs.djangoproject.com/en/3.2/ref/settings/#default-auto-field
131
132 DEFAULT_AUTO_FIELD = 'django.db.models.BigAutoField'
133
```

- now we download any photo you wanna and add it in img



- so now in student_detials add

The screenshot shows the Visual Studio Code interface with the title bar "student_detials.html - tech - Visual Studio Code". The menu bar includes File, Edit, Selection, View, Go, Run, Terminal, and Help. The Explorer sidebar on the left shows a project structure under "TECH": devops (with __pycache__ and migrations), static/img (with pang.png), and tech (with __pycache__, __init__.py, admin.py, apps.py, models.py, tests.py, urls.py, views.py). The main editor pane displays the "student_detials.html" template file:

```
templates > devops > student_detials.html > ...
1  {% extends 'devops/base.html' %}
2  {% load static %}
3
4  {% block title_block %}
5
6  <h1>welcome: {{student.fname}} {{student.lname}}</h1>
7  <br />
8  <img src ="{{ static 'img/pang.png' }}" width="60" height="60"/>
9  <a href="/devops/all"> click Back </a>
10 <br />
11 {% endblock %}
12 {% block main_block %}
13
14 <h3><b>First Name: </b>{{student.fname}}</h3>
15 <h3><b>Last Name: </b>{{student.lname}}</h3>
16 <h3><b>Age: </b>{{student.age}}</h3>
17 <h3><b>Track: </b>{{student.student_track}}</h3>
18 {% endblock %}
19
20
21
22
23
24
```

welcome: aya rabih



[click Back](#)

First Name: aya

Last Name: rabih

Age: 26

Track: devops1

- now refresh your website
- now we need to make a page we can add any new student in it and we can save also and return all data to our database

```

EXPLORER      ...      forms.py      urls.py      views.py      ▷ □
                ↴ TECH
                ↳ devops
                  > __pycache__
                  > migrations
                  ↳ static/img
                  < pang.png
                  < __init__.py
                  < admin.py
                  < apps.py
                  < forms.py
                  < models.py
                  < tests.py
                  < urls.py
                  < views.py
                > tech
                ↳ templates
                  > admin
                  > devops
                    < base.html
                    < footer.html
                    < header.html
                    < new_student.html
                    < student_details.h...
                    < students.html
                  < db.sqlite3
                  < manage.py

devops > < views.py > ...
1   from django.shortcuts import render
2   from django.http import HttpResponseRedirect
3   from devops.models import Student, Track
4   from devops.forms import studentform, trackform
5
6   # Create your views here.
7   def home(request):
8       return HttpResponseRedirect("h1> welcome to my website this c")
9
10  def getStudent(request, st_id):
11      st = Student.objects.get(id = st_id)
12      context = {'student': st}
13      return render(request, 'devops/student_details.html', context)
14
15  def getAllStudents(request):
16      all_students = Student.objects.all()
17      context = {'students': all_students}
18      return render(request, "devops/students.html", context)
19
20  def newStudent(request):
21      st_form = studentform()
22      if request.method == 'POST':
23          st_form = studentform(request.POST)
24          if st_form.is_valid():
25              st_form.save()
26              return HttpResponseRedirect('/devops/all')
27
28  context = {'student_form': st_form}
29  return render(request, "devops/new_student.html", context)
30

```

- and create new_student.html and add that

The screenshot shows the VS Code interface with the following details:

- EXPLORER** sidebar: Shows a project structure under "TECH". Key items include "devops" (containing "__pycache__", "migrations", "static/img" with "pang.png", and "views.py"), "admin.py", "apps.py", "forms.py", "models.py", "tests.py", "urls.py", and "views.py". Other folders like "tech" and "templates" are also listed.
- CODE EDITOR**: The "new_student.html" file is open. The code is as follows:

```
<h1> Add New Student </h1>
<form method="POST"> {% csrf_token %}
    {{ student_form.as_p}}
    <button type="submit">save</button>
    <button type="reset">clear</button>
</form>
```

- now add our path

The screenshot shows the VS Code interface with the following details:

- EXPLORER** sidebar: Shows a project structure under "TECH". Key items include "devops" (containing "__pycache__", "migrations", "static/img" with "pang.png", and "views.py"), "admin.py", "apps.py", "forms.py", "models.py", "tests.py", and "urls.py".
- CODE EDITOR**: The "urls.py" file is open. The code is as follows:

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

urlpatterns = [
    path('home/', views.home),
    path('student/<st_id>', views.getStudent),
    path('all', views.getAllStudents),
    path('new', views.newStudent)
]
```

- and now save and refresh your website
- <http://127.0.0.1:8000/devops/new>

← → ⌂ 127.0.0.1:8000/devops/new
Import bookmarks... Getting Started PROGRAM AWS PERSONA

Add New Student

Fname:

Lname:

Age:

Student track:

- and when we click save it is return to first page

← → ⌂ 127.0.0.1:8000/devops/all
Import bookmarks... Getting Started PROGRAM AWS PERSONAL

He

ALL Students

First Name	Last Name	Age	Track	Graduted
mohamed	mohamed	25	devops1	False
aya	rabih	26	devops1	True
sayed	mohamed	20	devops	False
hossam	mostafa	40	devops	True
asmaa	amin	27	first1	True
amenah	islam	28	first1	True
islam	salah	30	sacand	True
amr	mohamed	29	sacand	True
mazen	kamal	20	first1	False
morad	salim	24	sacand	False

- now we need to add in this page a title named add new student when we click to it return us to add student

The screenshot shows a code editor with the following file structure:

```

EXPLORER      ...
  TECH
    __pycache__
    migrations
    static/img
      pang.png
    __init__.py
    admin.py
    apps.py
    forms.py
    models.py
    tests.py
    urls.py
    views.py
  > tech
  templates
    > admin
    <> devops
      <> base.html
      <> footer.html
      <> header.html
      <> new_student.html
      <> student_detials.h...
      <> students.html
    db.sqlite3
    manage.py

```

The file `new_student.html` is open in the editor, containing the following Python code:

```

1  {% extends 'devops/base.html' %} 
2  {% block title_block %} 
3  | 
4  <h1> Add New Student </h1>
5  {% endblock %} 
6  | 
7  {% block main_block %} 
8  <form method="POST"> {{ csrf_token }} 
9  {{ student_form.as_p}} 
10 <button type="submit">save</button>
11 <button type="reset">clear</button>
12 </form>
13 {% endblock %} 
14

```

- and now refresh your website
- now click to add new student

Header

ALL Students

[Add New Student](#)

First Name	Last Name	Age	Track	Graduated
mohamed	mohamed	25	devops1	False
aya	rabih	26	devops1	True
sayed	mohamed	20	devops	False
hossam	mostafa	40	devops	True
asmaa	amin	27	first1	True
amenah	islam	28	first1	True
islam	salah	30	sacand	True
amr	mohamed	29	sacand	True
mazen	kamal	20	first1	False
morad	salim	24	sacand	False

Footer>

Header

Add New Student

Fname:

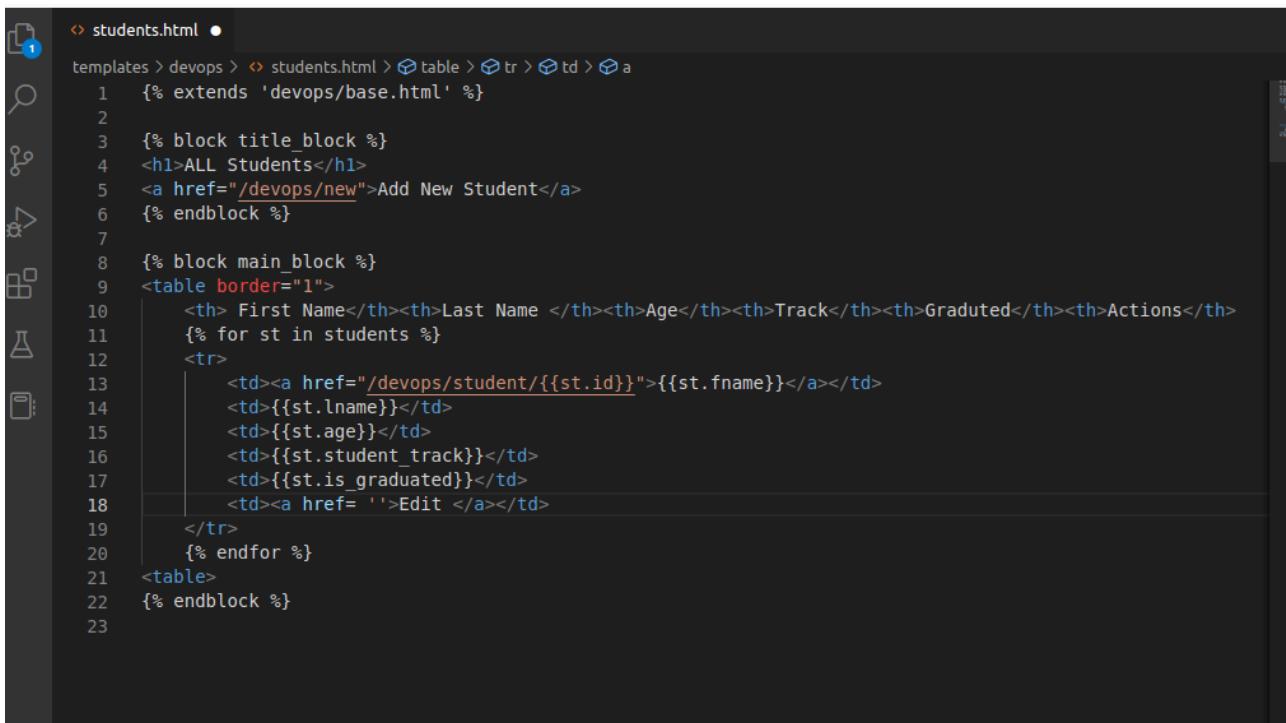
Lname:

Age:

Student track:

Footer>

- now we need to edit our data for any student
- go to student.html and add action and also `<td>Edit </td>`



The screenshot shows a code editor with the file "students.html" open. The code is a Django template for displaying a list of students and allowing editing of individual records. It includes blocks for title and main content, a table with columns for First Name, Last Name, Age, Track, Graduated status, and Actions (with an edit link), and logic to iterate over a list of students.

```
1  {% extends 'devops/base.html' %} 
2 
3  {% block title_block %} 
4  <h1>ALL Students</h1> 
5  <a href="/devops/new">Add New Student</a> 
6  {% endblock %} 
7 
8  {% block main_block %} 
9  <table border="1"> 
10 <thead> 
11   <th> First Name</th><th>Last Name </th><th>Age</th><th>Track</th><th>Graduated</th><th>Actions</th> 
12 </thead> 
13 <tbody> 
14   {% for st in students %} 
15     <tr> 
16       <td><a href="/devops/student/{{st.id}}"/>{{st.fname}}</a></td> 
17       <td>{{st.lname}}</td> 
18       <td>{{st.age}}</td> 
19       <td>{{st.student_track}}</td> 
20       <td>{{st.is_graduated}}</td> 
21       <td><a href= ''>Edit </a></td> 
22     </tr> 
23   {% endfor %} 
24 </tbody> 
25 {% endblock %}
```

- now in urls.py add new path `path('edit/<st_id>', views.editstudent)`

The screenshot shows the VS Code interface with the following details:

- EXPLORER** sidebar: Shows the project structure under the **TECH** folder, including **devops**, **migrations**, **static/img**, and several Python files (**__init__.py**, **admin.py**, **apps.py**, **forms.py**, **models.py**, **tests.py**, **urls.py**).
- CODE** tab: The **urls.py** file is open, showing URL patterns for a Django application. The code includes imports from `django.urls` and `views`, and defines `urlpatterns` with paths for home, student detail, all students, new student, and edit student.

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

urlpatterns = [
    path('home/', views.home),
    path('student/<st_id>', views.getStudent),
    path('all', views.getAllStudents),
    path('new', views.newStudent),
    path('edit/<st_id>', views.editstudent)
]
```

now in views.py add

The screenshot shows the VS Code interface with the following details:

- EXPLORER** sidebar: Shows the project structure under the **TECH** folder, including **devops**, **migrations**, **static/img**, and several Python files (**__init__.py**, **admin.py**, **apps.py**, **forms.py**, **models.py**, **tests.py**, **urls.py**, **views.py**).
- CODE** tab: The **views.py** file is open, showing three functions: `getALLStudents`, `newStudent`, and `editstudent`.
 - `getALLStudents` retrieves all students from the database and returns them with a context dictionary.
 - `newStudent` handles POST requests for new students, creating a form, validating it if POSTed, saving it, and then redirecting to the all students page.
 - `editstudent` retrieves a student by ID, creates a form instance with it, and returns it with a context dictionary.

```
def getALLStudents(request):
    all_students = Student.objects.all()
    context = {'students': all_students}
    return render(request, "devops/students.html", context)

def newStudent(request):
    st_form = studentform()
    if request.method == 'POST':
        st_form = studentform(request.POST)
        if st_form.is_valid():
            st_form.save()
            return HttpResponseRedirect('/devops/all')

    context = {'student_form': st_form}
    return render(request, "devops/new_student.html", context)

def editstudent(request, st_id):
    st = Student.objects.get(id= st_id)
    st_form = studentform(instance= st)
    context = {'student_form': st_form}
    return render(request, "devops/new_student.html", context)
```

- now refresh your website

ALL Students

[Add New Student](#)

First Name	Last Name	Age	Track	Graduated	Actions
mohamed	mohamed	25	devops1	False	Edit
aya	rabih	26	devops1	True	Edit
sayed	mohamed	20	devops	False	Edit
hossam	mostafa	40	devops	True	Edit
asmaa	amin	27	first1	True	Edit
amenah	islam	28	first1	True	Edit
islam	salah	30	sacand	True	Edit
amr	mohamed	29	sacand	True	Edit
mazen	kamal	20	first1	False	Edit
morad	salim	24	sacand	False	Edit

- if you go to
- <http://127.0.0.1:8000/devops/edit/1>
- you can add any
- now edit

[REDACTED]

Add New Student

Fname:

Lname:

Age:

Student track:

[REDACTED]

- so now click save
- now we need to active button edit

Add New Student

Fname:

Lname:

Age:

Student track:

ALL Students

[Add New Student](#)

First Name	Last Name	Age	Track	Graduted	Actions
saly	mohamed	25	devops1	False	Edit
aya	rabih	26	devops1	True	Edit
sayed	mohamed	20	devops	False	Edit
hossam	mostafa	40	devops	True	Edit
asmaa	amin	27	first1	True	Edit
amena	islam	28	first1	True	Edit
islam	salah	30	sacand	True	Edit
amr	mohamed	29	sacand	True	Edit
mazen	kamal	20	first1	False	Edit
morad	salim	24	sacand	False	Edit

```
views.py          students.html
templates > devops > students.html > table > table
1   {% extends 'devops/base.html' %} 
2
3   {% block title_block %} 
4   <h1>ALL Students</h1>
5   <a href="/devops/new">Add New Student</a>
6   {% endblock %} 
7
8   {% block main_block %} 
9   <table border="1">
10  <th> First Name</th><th>Last Name </th><th>Age</th><th>Track</th><th>
11  {% for st in students %} 
12  <tr>
13      <td><a href="/devops/student/{{st.id}}">{{st.fname}}</a></td>
14      <td>{{st.lname}}</td>
15      <td>{{st.age}}</td>
16      <td>{{st.student_track}}</td>
17      <td>{{st.is_graduated}}</td>
18      <td><a href= '/devops/edit/{{st.id}}'>Edit </a></td>
19  </tr>
20  {% endfor %} 
21 </table>
22 {% endblock %} 
```

refresh now your website

ALL Students

Add New Student

First Name	Last Name	Age	Track	Graduated	Actions
saly	mohamed	25	devops1	False	Edit
aya	rabih	26	devops1	True	Edit
sayed	mohamed	20	devops	False	Edit
hosam	mostafa	40	devops	True	Edit
asmaa	amin	27	first1	True	Edit
amenia	islam	28	first1	True	Edit
islam	salah	30	sacand	True	Edit
amr	mohamed	29	sacand	True	Edit
mazen	kamal	20	first1	False	Edit
morad	salim	24	sacand	False	Edit

Add New Student

Fname:

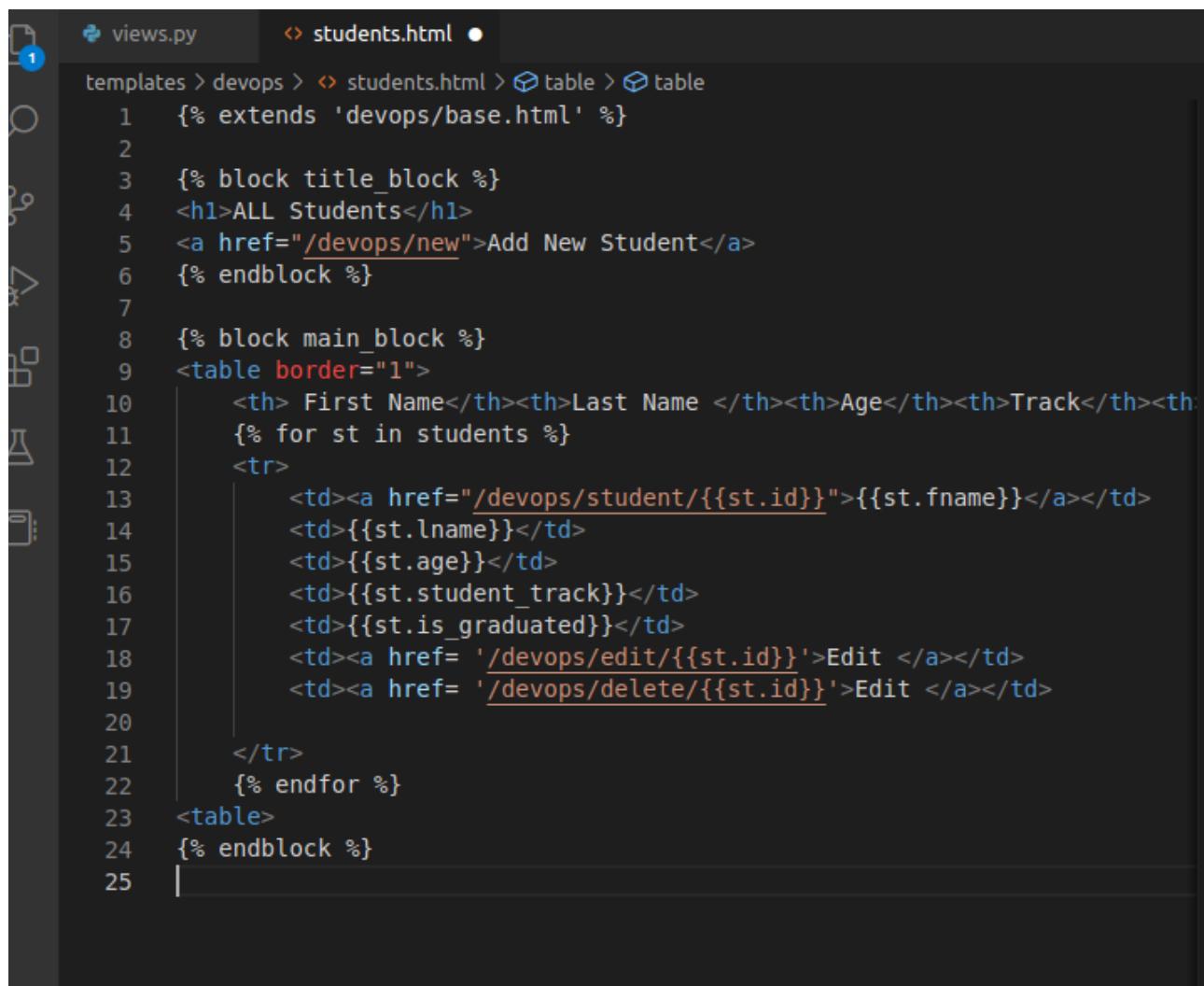
Lname:

Age:

Student track:

For

- click to edit
- now we need to add new roll for delete



The screenshot shows a code editor with a dark theme. On the left, there's a sidebar with icons for file operations like open, save, and search. The main area displays a Python template file named 'students.html'. The code uses Jinja2 syntax with blocks and loops to render a table of student data. It includes links for editing and deleting each student record.

```
views.py          students.html ●
templates > devops > students.html > table > table
1   {% extends 'devops/base.html' %} 
2
3   {% block title_block %}
4     <h1>ALL Students</h1>
5     <a href="/devops/new">Add New Student</a>
6   {% endblock %}
7
8   {% block main_block %}
9     <table border="1">
10       <th> First Name</th><th>Last Name </th><th>Age</th><th>Track</th><th>
11         {% for st in students %}
12           <tr>
13             <td><a href="/devops/student/{{st.id}}">{{st.fname}}</a></td>
14             <td>{{st.lname}}</td>
15             <td>{{st.age}}</td>
16             <td>{{st.student_track}}</td>
17             <td>{{st.is_graduated}}</td>
18             <td><a href= '/devops/edit/{{st.id}}'>Edit </a></td>
19             <td><a href= '/devops/delete/{{st.id}}'>Edit </a></td>
20           </tr>
21         {% endfor %}
22     </table>
23   {% endblock %}
```

and go urls.py

The screenshot shows the VS Code interface with the Explorer sidebar on the left and the code editor on the right. The Explorer sidebar shows a project structure under 'TECH' with several subfolders like 'devops', 'migrations', 'static/img', etc., and files like 'urls.py' and 'views.py'. The code editor displays the 'urls.py' file which contains the following code:

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

urlpatterns = [
    path('home/', views.home),
    path('student/<st_id>', views.getStudent),
    path('all', views.getAllStudents),
    path('new', views.newStudent),
    path('edit/<st_id>', views.editstudent),
    path('delete/<st_id>', views.deletestudent)
]
```

go now to view.py add delete .student

The screenshot shows the VS Code interface with the Explorer sidebar on the left and the code editor on the right. The Explorer sidebar shows the same project structure as before. The code editor displays the 'views.py' file which contains the following code:

```
if request.method == 'POST':
    st_form = studentform(request.POST)
    if st_form.is_valid():
        st_form.save()
        return HttpResponseRedirect('/devops/all')

context = {'student_form': st_form}
return render(request,"devops/new_student.html", context)

def editstudent(request, st_id):
    st = Student.objects.get(id= st_id)
    st_form = studentform(instance= st)
    if request.method == 'POST':
        st_form = studentform(request.POST, instance=st)
        if st_form.is_valid():
            st_form.save()
            return HttpResponseRedirect('/devops/all')
    context = {'student_form': st_form}
    return render(request,"devops/new_student.html", context)

def deletestudent(request, st_id):
    student = Student.objects.get(id= st_id)
    student.delete
    return HttpResponseRedirect('/devops/all')
```

- go to student.html

```

1  views.py   ●  ◊ students.html ✘
2  templates > devops > ◊ students.html > table > table
3
4      <h1>ALL Students</h1>
5      <a href="/devops/new">Add New Student</a>
6      {% endblock %}
7
8      {% block main_block %}
9      <table border="1">
10         <th> First Name</th><th>Last Name </th><th>Age</th><th>Track</th><th>Graduated</th><th>Actions</th>
11         {% for st in students %}
12             <tr>
13                 <td><a href="/devops/student/{{st.id}}">{{st.fname}}</a></td>
14                 <td>{{st.lname}}</td>
15                 <td>{{st.age}}</td>
16                 <td>{{st.student_track}}</td>
17                 <td>{{st.is_graduated}}</td>
18                 <td><a href= '/devops/edit/{{st.id}}'>Edit </a></td>
19                 <td><a href= '/devops/delete/{{st.id}}'>Delete </a></td>
20             </tr>
21         {% endfor %}
22     </table>
23     {% endblock %}
24
25

```

- save and refresh

ALL Students

[Add New Student](#)

First Name	Last Name	Age	Track	Graduted	Actions	Delete
saly	mohamed	25	devops1	False	Edit	Delete
aya	rabih	26	devops1	True	Edit	Delete
sayed	mohamed	20	devops	False	Edit	Delete
hossam	mostafa	40	devops	True	Edit	Delete
asmaa	amin	27	first1	True	Edit	Delete
amenia	islam	28	first1	True	Edit	Delete
islam	salah	30	sacand	True	Edit	Delete
amr	mohamed	29	sacand	True	Edit	Delete
mazen	kamal	20	first1	False	Edit	Delete
morad	salim	24	sacand	False	Edit	Delete

The screenshot shows the VS Code interface with the following details:

- EXPLORER** sidebar: Shows a project structure under 'TECH'. The 'devops' folder contains __pycache__, migrations, static/img (with a pang.png file), __init__.py, admin.py, apps.py, forms.py, models.py, tests.py, urls.py, and views.py.
- CODE** tab: The 'views.py' file is open. The code defines three functions: newStudent, editstudent, and deletestudent. The newStudent function handles POST requests to add a new student. The editstudent function handles POST requests to update an existing student. The deletestudent function deletes a student by their ID.

```
def newStudent(request):
    st_form = studentform()
    if request.method == 'POST':
        st_form = studentform(request.POST)
        if st_form.is_valid():
            st_form.save()
            return HttpResponseRedirect('/devops/all')

    context = {'student_form': st_form}
    return render(request,"devops/new_student.html", context)

def editstudent(request, st_id):
    st = Student.objects.get(id= st_id)
    st_form = studentform(instance= st)
    if request.method == 'POST':
        st_form = studentform(request.POST, instance=st)
        if st_form.is_valid():
            st_form.save()
            return HttpResponseRedirect('/devops/all')
    context = {'student_form': st_form}
    return render(request,"devops/new_student.html", context)

def deletestudent(request, st_id):
    student = Student.objects.get(id= st_id)
    student.delete()
    return HttpResponseRedirect('/devops/all')
```

- now we need to make this button of delete work
now refresh

ALL Students

[Add New Student](#)

First Name	Last Name	Age	Track	Graduated	Actions	Delete
aya	rabih	26	devops1	True	Edit	Delete
sayed	mohamed	20	devops	False	Edit	Delete
hossam	mostafa	40	devops	True	Edit	Delete
asmaa	amin	27	first1	True	Edit	Delete
amena	islam	28	first1	True	Edit	Delete
islam	salah	30	sacand	True	Edit	Delete
amr	mohamed	29	sacand	True	Edit	Delete
mazen	kamal	20	first1	False	Edit	Delete
morad	salim	24	sacand	False	Edit	Delete

Footer

- now we just change cloure of our page

welcome

ALL Students

Add New Student

First Name	Last Name	Age	Track	Graduted	Actions
aya	rabih	26	devops1	True	Edit Delete
sayed	mohamed	20	devops	False	Edit Delete
hossam	mostafa	40	devops	True	Edit Delete
asmaa	amin	27	first1	True	Edit Delete
amenia	islam	28	first1	True	Edit Delete
islam	salah	30	sacand	True	Edit Delete
amr	mohamed	29	sacand	True	Edit Delete
mazen	kamal	20	first1	False	Edit Delete
morad	salim	24	sacand	False	Edit Delete
hossam	amin	28	first1	True	Edit Delete
ahmed	mosamed	55	sacand	True	Edit Delete

connect with us

- done