

Topic: 4. Setting API EndPoints

Speaker: Personal / Notebook: API Development using Django Framework



For other resources on how to create simple API endpoints using Django, use this [reference in Medium](#).

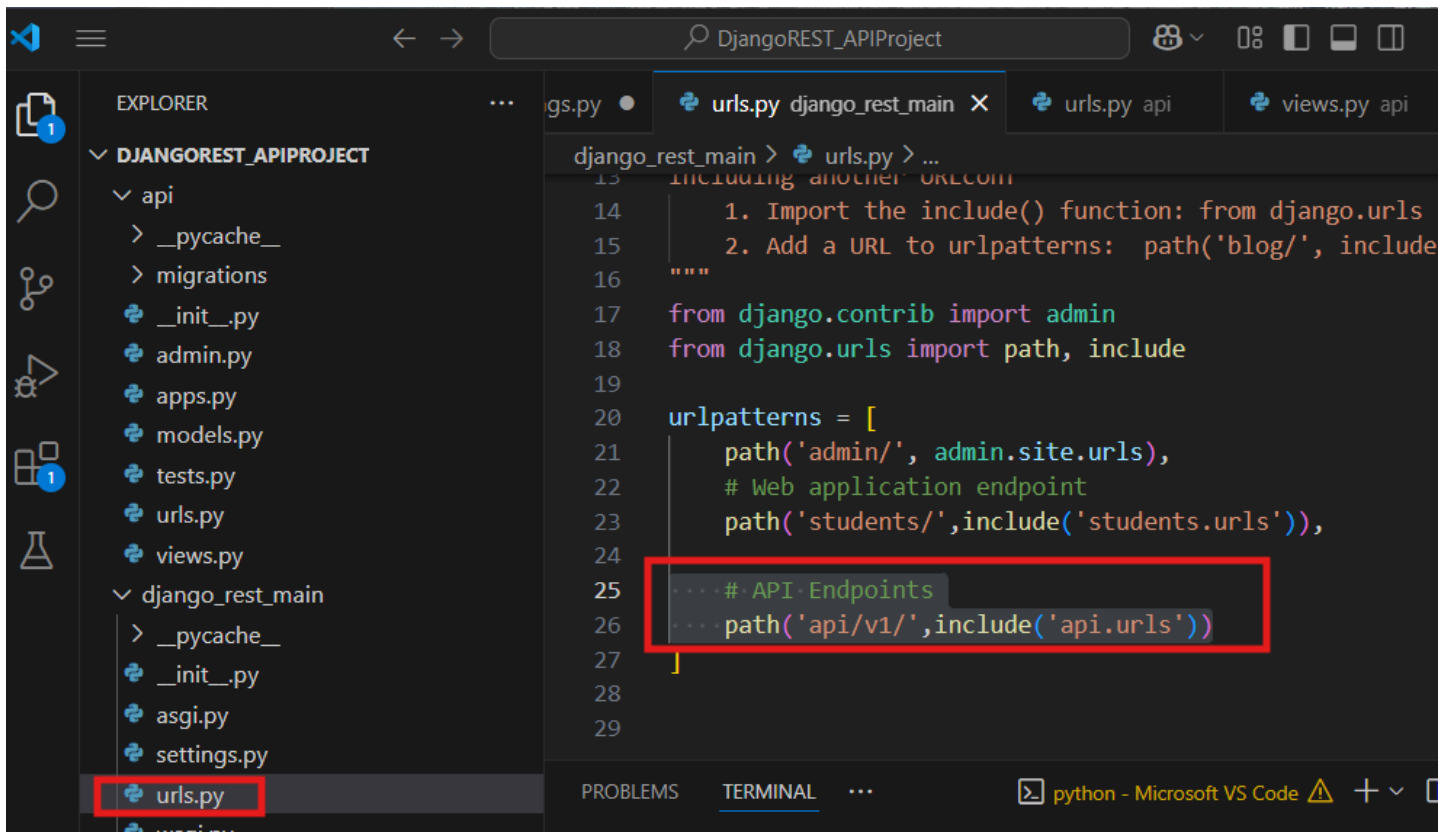
1. Create an API app in your Django project:

```
$ python manage.py startapp api
```

2. Register this new app in your SETTINGS.PY

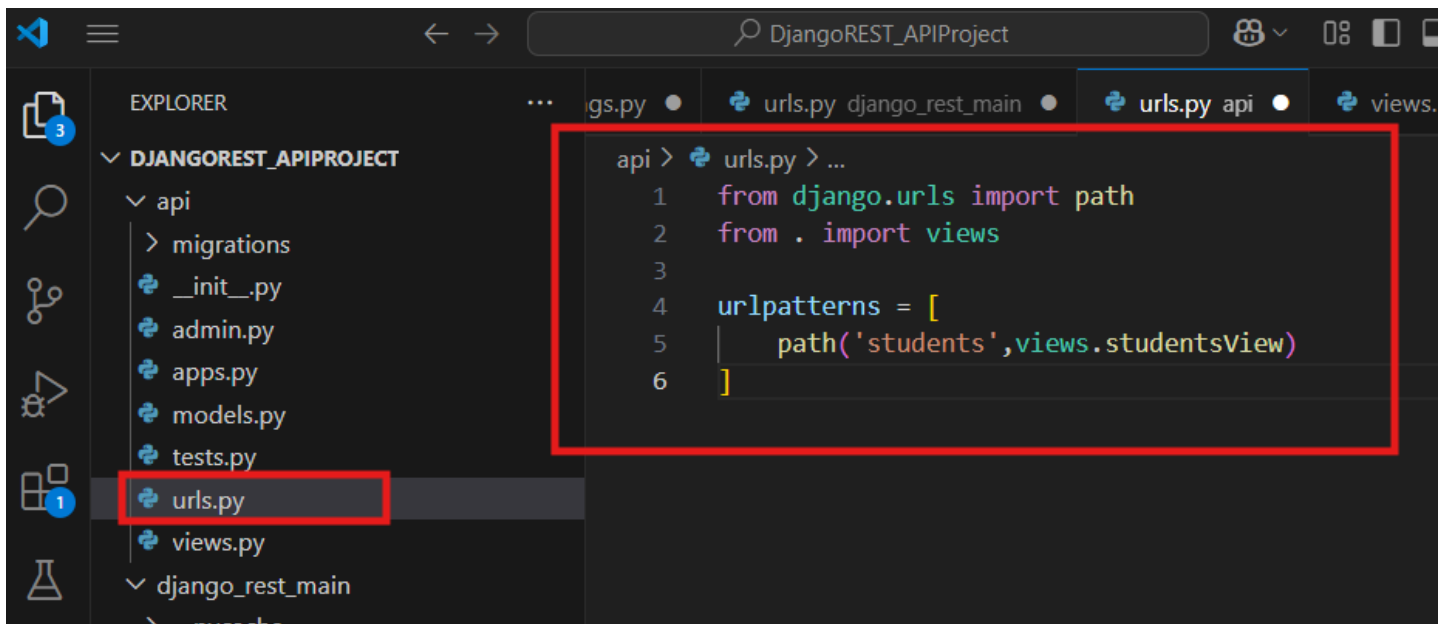
```
31 # Application definition
32
33 INSTALLED_APPS = [
34     'django.contrib.admin',
35     'django.contrib.auth',
36     'django.contrib.contenttypes',
37     'django.contrib.sessions',
38     'django.contrib.messages',
39     'django.contrib.staticfiles',
40     'rest_framework',
41     'students',
42     'api',
43 ]
44
45 MIDDLEWARE = [
46     'django.middleware.security.SecurityMiddleware',
47     'django.contrib.sessions.middleware.SessionMi
```

3. Now, update the main project's URLS.PY to include the URLS.PY of the newly created app.



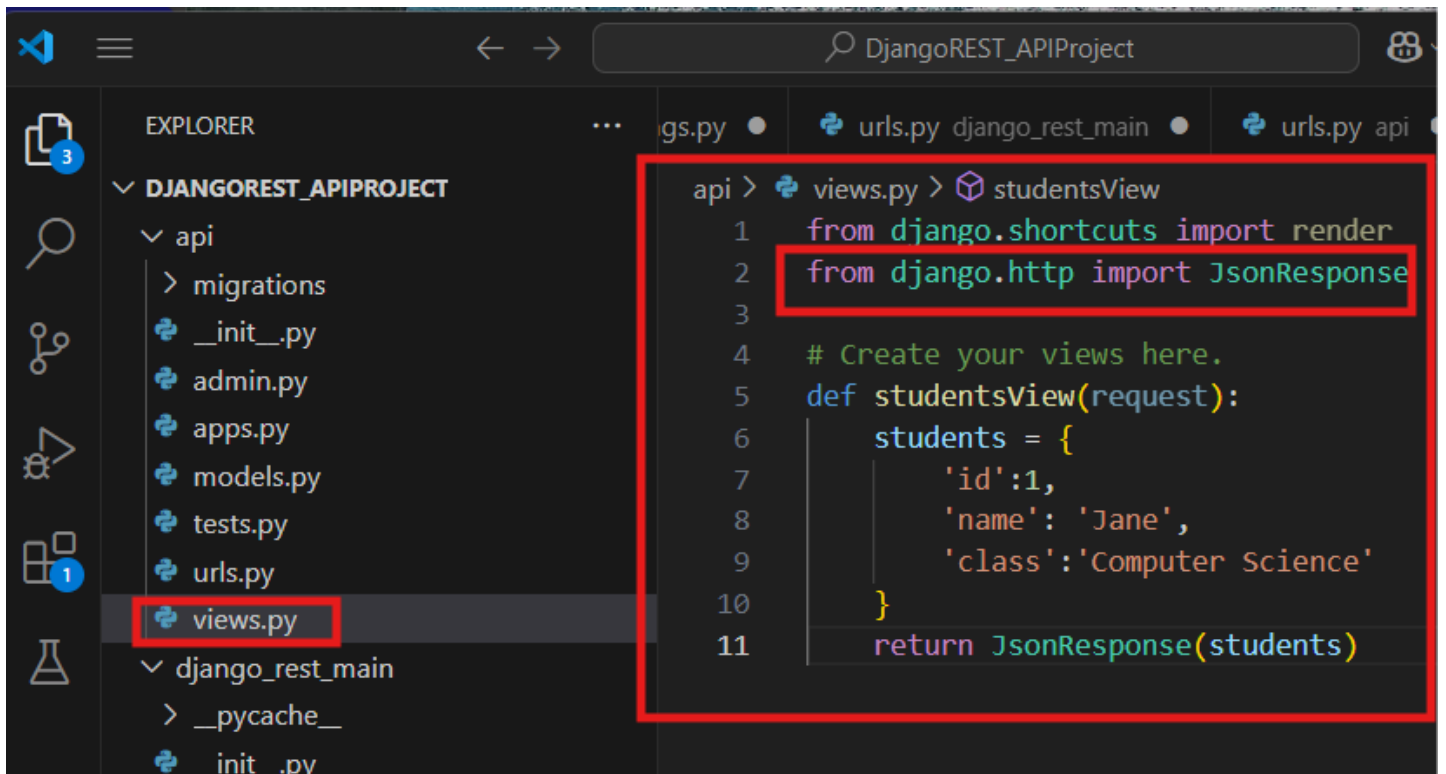
```
13 including another URLCONF
14
15 1. Import the include() function: from django.urls
16 2. Add a URL to urlpatterns: path('blog/', include
17 """
18 from django.contrib import admin
19 from django.urls import path, include
20
21 urlpatterns = [
22     path('admin/', admin.site.urls),
23     # Web application endpoint
24     path('students/',include('students.urls')),
25     ...# API Endpoints
26     ...path('api/v1/',include('api.urls'))
27 ]
28
29
```

4. We then create a URLS.py in our API app and create the path.



```
api > urls.py > ...
1 from django.urls import path
2 from . import views
3
4 urlpatterns = [
5     path('students',views.studentsView)
6 ]
```

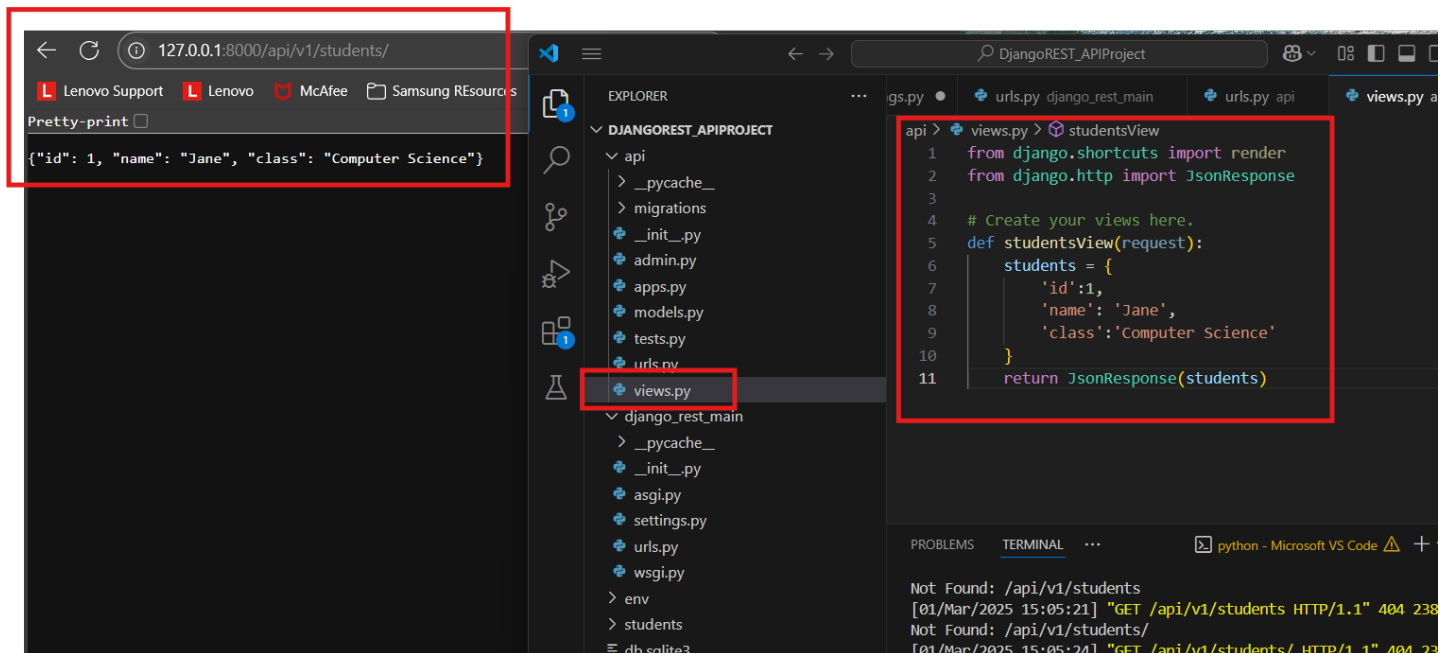
5. Update the API APP'S VIEWS.PY . Unlike regular views of web-based endpoints, APIs return JSON data.



6. Run the server and add the API endpoint:

```
$ python manage.py runserver
```

In your browser: 127.0.0.1:8000/api/v1/students/



7. Run the migrations command and create superuser.

```
$ python manage.py migrate
```

This will create `user_auths` default table.

The screenshot shows a VS Code editor window for a project named 'DjangoREST_APIProject'. The Explorer panel on the left shows the project structure, with 'views.py' selected under the 'api' directory. The main editor shows the 'studentsView' function in 'views.py', which returns a JSON response for a student named Jane. The Terminal panel at the bottom shows the command 'python manage.py migrate' being executed, with a list of migrations being applied successfully.

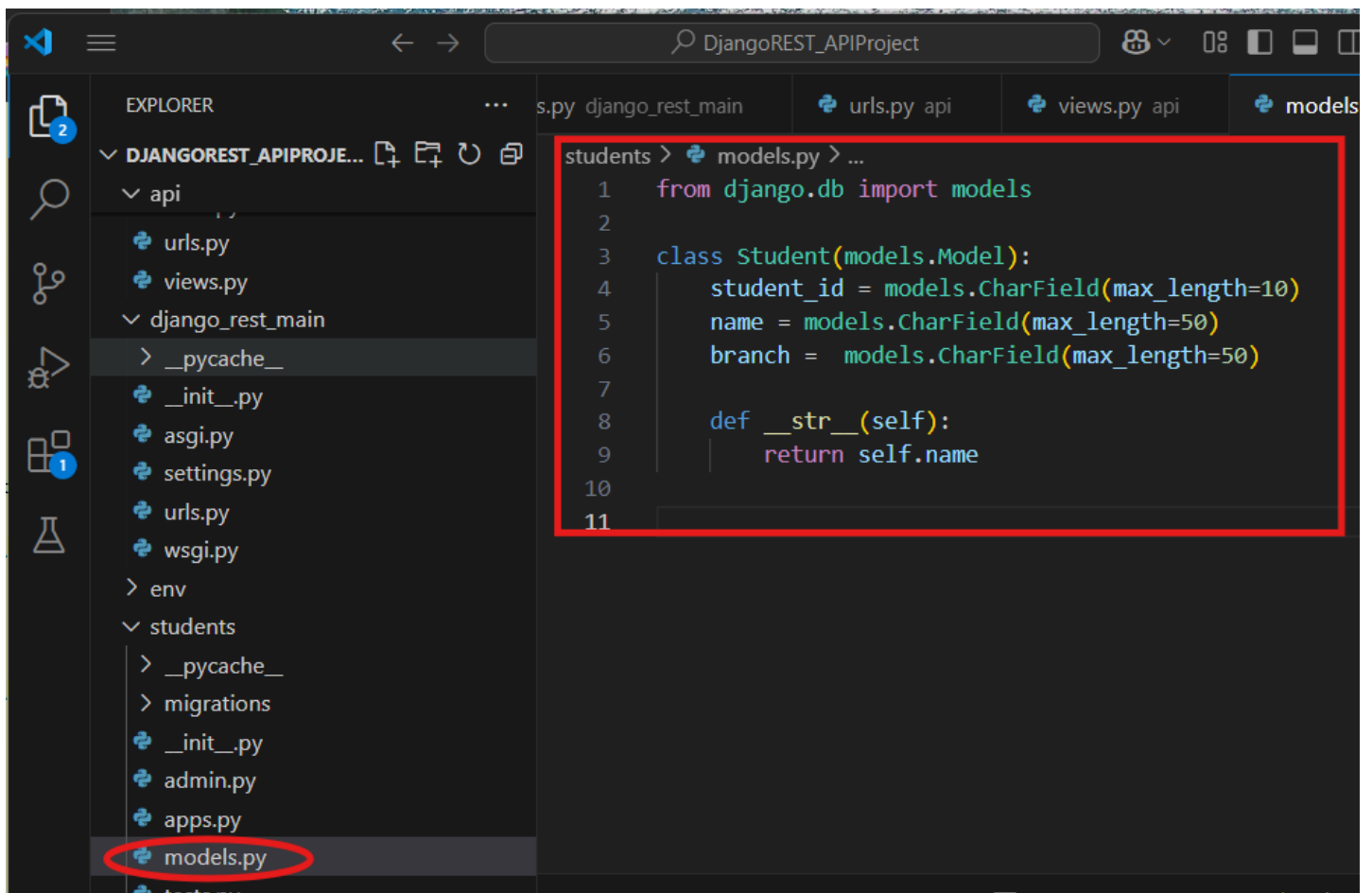
```
api > views.py > studentsView
5 def studentsView(request):
6     students = {
7         'id': 1,
8         'name': 'Jane',
9         'class': 'Computer Science'
10    }
11    return JsonResponse(students)
```

rosil@LearnCodeRepeat MINGW64 C:/Users/rosil/AppData/Local/Programs/Microsoft VS Code
\$ python 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.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
(env)
rosil@LearnCodeRepeat MINGW64 C:/Users/rosil/AppData/Local/Programs/Microsoft VS Code
\$

8. Access the admin panel by running the server. Access it using:

<http://127.0.0.1:8000/admin/>

9. Create a student model in the STUDENTS app to create a new table.



10. Run the migrations.

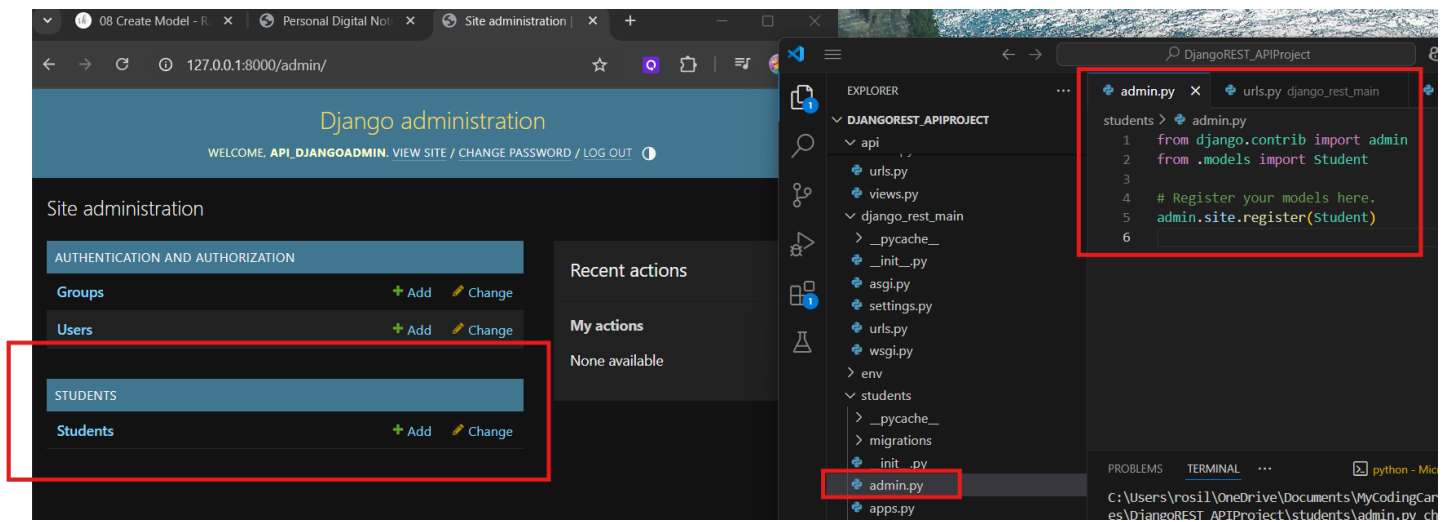
```
$ python manage.py makemigrations
```

```
$ python manage.py migrate
```

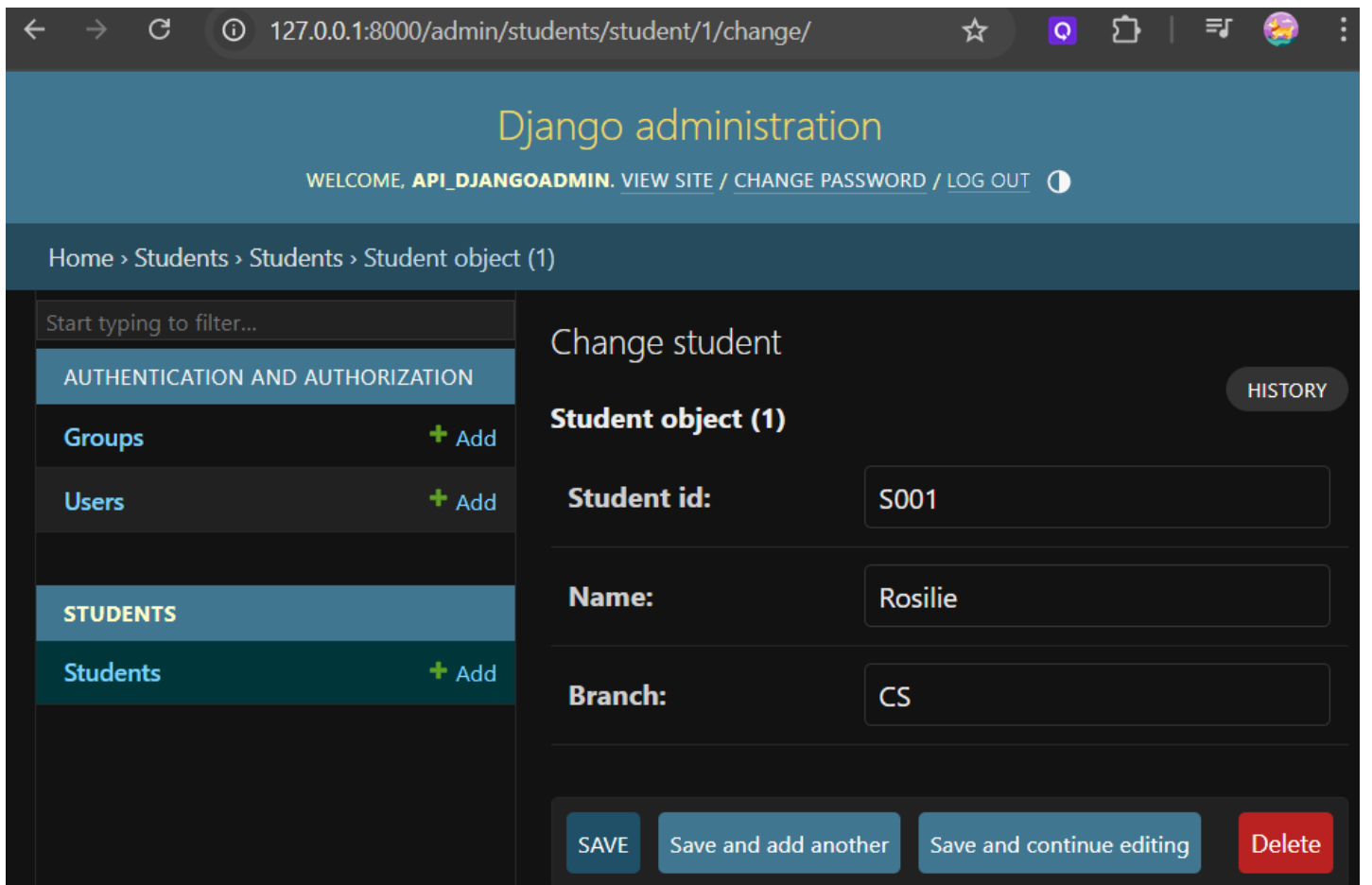
```
• $ python manage.py makemigrations
Migrations for 'students':
  students\migrations\0001_initial.py
    + Create model Student
(env)
```

```
(env)
rosil@LearnCodeRepeat MINGW64 C:/Users/rosil/AppData/Local/Programs/Mi
Code
$ python manage.py migrate
Operations to perform:
  Apply all migrations: admin, auth, contenttypes, sessions, students
Running migrations:
  Applying students.0001_initial... OK
(env)
```

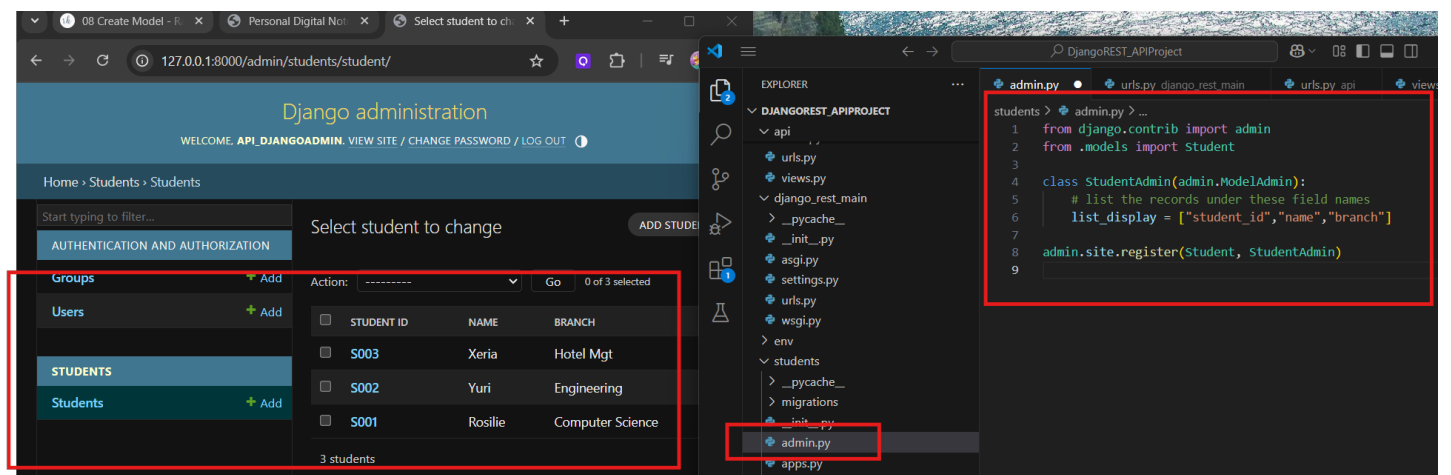
11. To make this table appear on your admin dashboard, register this in the ADMIN.PY of STUDENTS app.



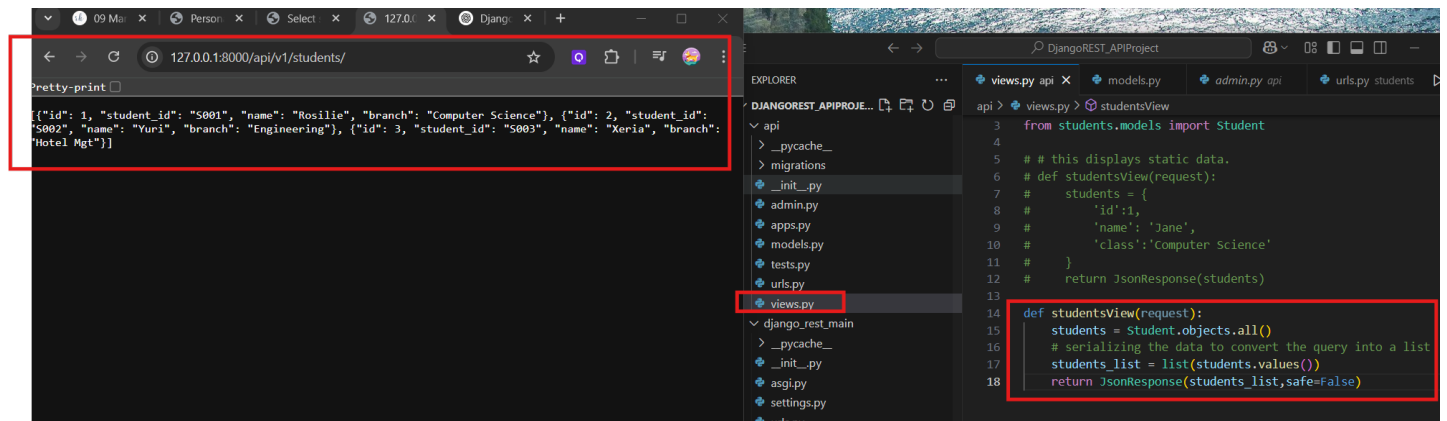
12. Create a sample record in the table.



13. To display the model student records using the admin dashboard:



14. To display the dynamic data from the database. We need to **SERIALIZE** to convert the returned query set into a list.



```
127.0.0.1:8000/api/v1/students/
[{"id": 1, "student_id": "S001", "name": "Rosilie", "branch": "Computer Science"}, {"id": 2, "student_id": "S002", "name": "Yuri", "branch": "Engineering"}, {"id": 3, "student_id": "S003", "name": "Xeria", "branch": "Hotel Mgt"}]
```

```
views.py
from students.models import Student

def studentsView(request):
    # this displays static data.
    # def studentsView(request):
    #     students = {
    #         'id': 1,
    #         'name': 'Jane',
    #         'class': 'Computer Science'
    #     }
    #     return JsonResponse(students)

    def studentsView(request):
        students = Student.objects.all()
        # serializing the data to convert the query into a list
        students_list = list(students.values())
        return JsonResponse(students_list, safe=False)
```

15. **SERIALIZERS** are like **TRANSLATORS** that convert certain data i.e **QUERYSET** from your database into other types of data like **JSON** data that can be used on **HTML**. While **DESERIALIZERS** will reverse the translation i.e from **JSON** file into **Query set** (database records).