

به نام خدا

مستند سازی پروژه درس رایانش ابری

موضوع:

داکرایز پروژه CRUD فیلم با Django

استاد:

دکتر هدی طاهری

دانشجو:

ابراهیم رستمی

4001277272

زمستان 1403

<https://github.com/Ebrahim-Rostami1/cloud-computing-project>

محتوای فایل های docker-compose ، docker file و environment در لینک بالا قرار داده شده است.

در این پروژه دو عدد فایل docker-compose وجود دارد که یکی برای دیتابیس و دیگری برای برنامه اصلی استفاده می شود. در فایل مربوط به دیتابیس شبکه‌ای به نام app-network ساخته می شود و در فایل مربوط به برنامه به این شبکه ملحق می شویم. پس بنابراین ابتدا باید کانتینر مربوط به دیتابیس را بالا بیاوریم و سپس کانتینر مربوط به برنامه.

می دانیم که برای دفعه اول ران کردن کانتینرها، باید از build — استفاده کنیم تا ایمج ها از روی داکر فایل ساخته شوند و در دفعات بعدی فقط دستور up به تنهایی کافیست.

می دانیم که ایمج مربوط به برنامه از روی داکر فایل ساخته می شود اما ایمج مربوط به دیتابیس را از داکرهاب برمی داریم چون قبلا ایمج مناسبی برای دیتابیس Postgres در آن وجود دارد.

بعد از اجرای کانتینرها باید وارد کانتینر برنامه شده و دستورات makemigrations و migrate را وارد کنیم. اولی برای مشخص کردن تیبیل های دیتابیس و دومی برای ساخته شدن آنها در دیتابیس.

بعد از اتمام کار با دستور down به راحتی کانتینرها را از حالت اجرا در می آوریم. با v- می توان والیوم ایمج های ساخته شده را نیز پاک کرد اما برای استفاده مجدد باید هنگام دستور up، build — - را هم بزنیم.

## نمایش دستورات وارد شده

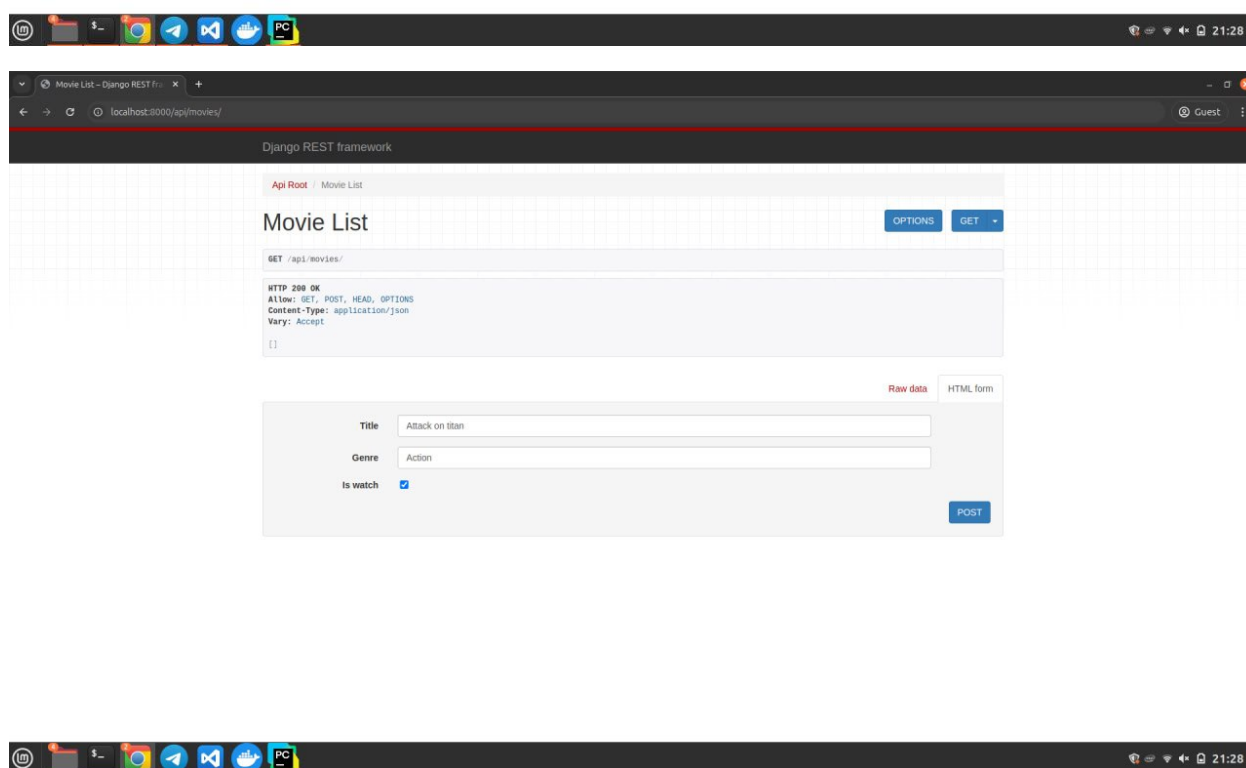
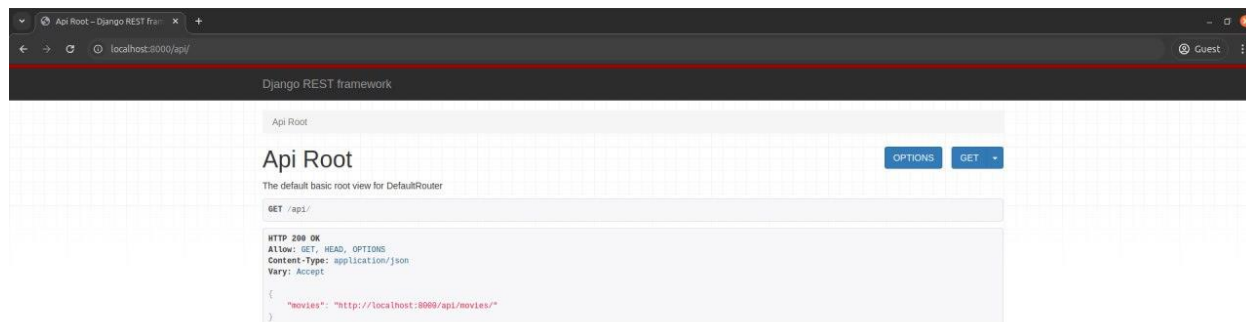
```
Ebrahim@custom host:~/Desktop/Movies$ sudo docker-compose -f docker-compose.db.yml up --build -d
Creating network "app_network" with driver "bridge"
Creating volume "movie_postgres_data" with default driver
Creating postgres db ...
Ebrahim@custom host:~/Desktop/Movies$ sudo docker-compose -f docker-compose.app.yml up --build -d
WARNING: Found orphan containers (postgres_db) for this project. If you removed or renamed this service in your compose file, you can run this command with the --remove-orphans flag to clean it up.
Building web
[*] Building 9.6s (11/11) FINISHED
[*] [internal] Load build definition from Dockerfile
[*] [internal] Load metadata for Dockerfile: 36MB
[*] [internal] Load metadata for Dockerfile/library/python:3.10-slim
[*] [internal] Load Dockerignore
[*] [internal] Transferring context: 2B
[*] [internal] FROM scratch: sha256:3b07729d78040000000000000000000000000000000000000000000000000000
[*] [internal] Load build context
[*] [internal] Transferring context: 880.47MB
[*] CACHED [3/4] RUN apt-get update && apt-get install -y gcc libpq-dev && rm -rf /var/lib/apt/lists/*
[*] CACHED [3/4] RUN pip install --no-cache-dir --upgrade pip && pip install --no-cache-dir -r requirements.txt
[*] CACHED [4/4] RUN pip install --no-cache-dir --upgrade pip && pip install --no-cache-dir -r requirements.txt
[*] [4/4] CMD
[*] exporting to image
[*] exporting layers
[*] writing image image: sha256:1224e25c04940000000000000000000000000000000000000000000000000000
[*] pushing to docker://library/movie-web
Creating django app ...
Ebrahim@custom host:~/Desktop/Movies$ sudo docker ps
CONTAINER ID   IMAGE          COMMAND                  CREATED        STATUS        PORTS                               NAMES
4495ef13f39b   movie-web     "python manage.py ru..." 39 seconds ago Up 39 seconds 0.0.0.0:8000->8000/tcp, :::8000->8000/tcp   django_app
584ef91eff     postgres:15   "docker-entrypoint.s..." About a minute ago Up About a minute 0.0.0.0:5433->5432/tcp, :::5433->5432/tcp   postgres_db
Ebrahim@custom host:~/Desktop/Movies$
```

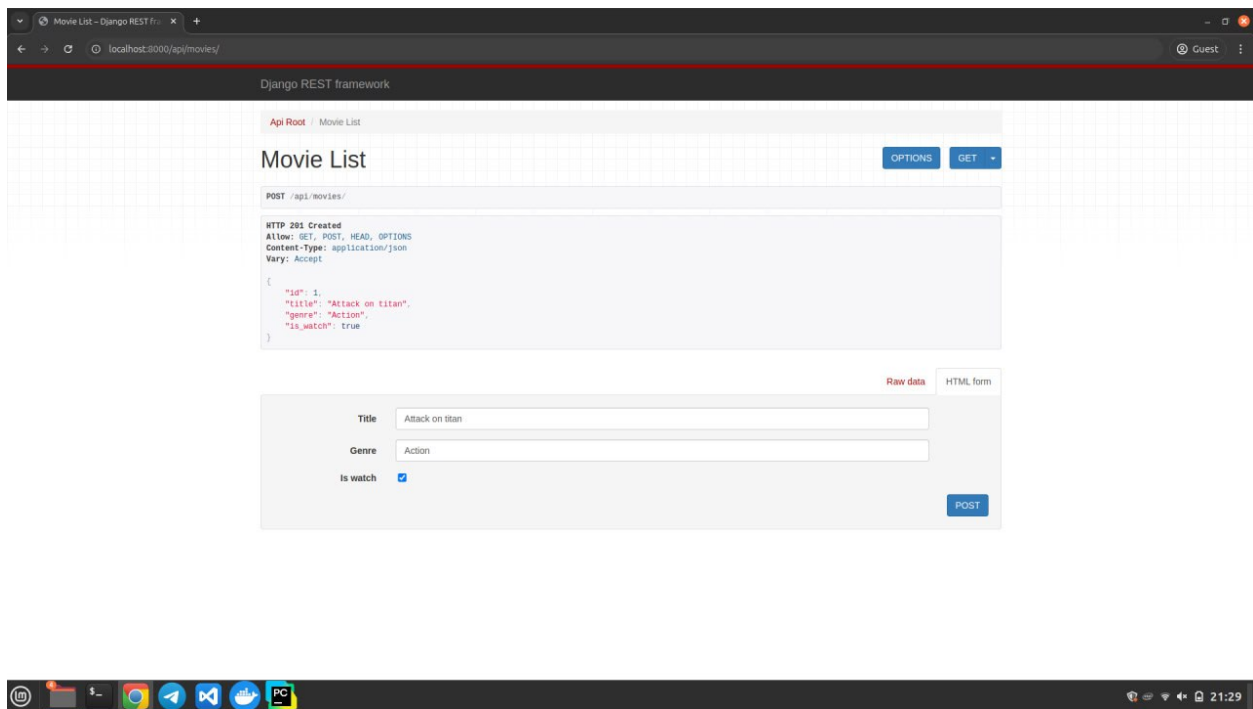
```
Ebrahim@custom host:~/Desktop/Movies$ sudo docker exec -it django_app bash
root@b7310e4285ed:/app# python manage.py makemigrations
No changes detected
root@b7310e4285ed:/app# python manage.py migrate
Operations to perform:
  Apply all migrations: admin, app, auth, contenttypes, sessions
Running migrations:
  Applying app.0001_initial... OK
root@b7310e4285ed:/app# exit
exit
Ebrahim@custom host:~/Desktop/Movies$
```

```
Ebrahim@custom host:~/Desktop/Movies$ sudo docker ps
CONTAINER ID   IMAGE          COMMAND                  CREATED        STATUS        PORTS                               NAMES
b7310e4285ed   movie-web     "python manage.py ru..." 17 minutes ago Up 17 minutes 0.0.0.0:8000->8000/tcp, :::8000->8000/tcp   django_app
a07a860eeb56   postgres:15   "docker-entrypoint.s..." 17 minutes ago Up 17 minutes 0.0.0.0:5433->5432/tcp, :::5433->5432/tcp   postgres_db
Ebrahim@custom host:~/Desktop/Movies$ sudo docker-compose -f docker-compose.app.yml down -v
Stopping django app ... done
WARNING: Found orphan containers (postgres_db) for this project. If you removed or renamed this service in your compose file, you can run this command with the --remove-orphans flag to clean it up.
Removing django app ... done
Network app_network is external, skipping
Ebrahim@custom host:~/Desktop/Movies$ sudo docker-compose -f docker-compose.db.yml down -v
Stopping postgres db ... done
Removing postgres db ... done
Removing network app_network
Removing volume movie_postgres_data
Ebrahim@custom host:~/Desktop/Movies$ sudo docker ps
```

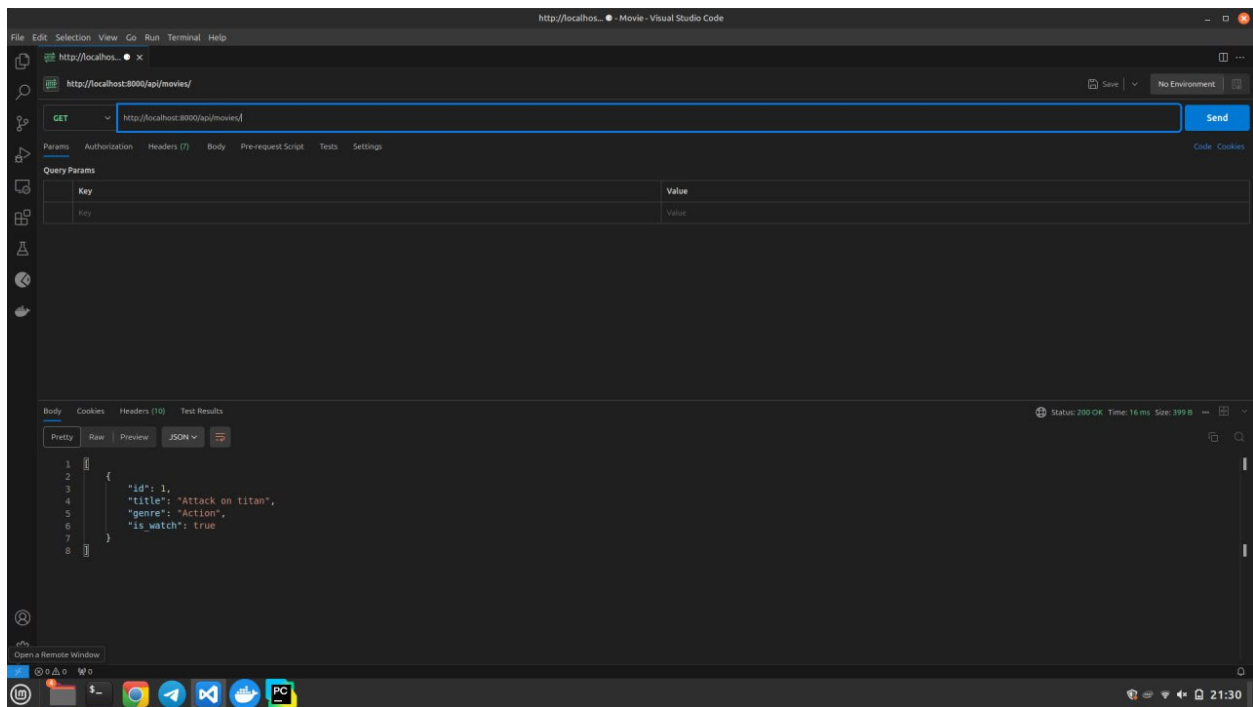
## تست CRUD در Postman

ابتدا وارد برنامه شده و داده ای را در آن وارد می کنیم.

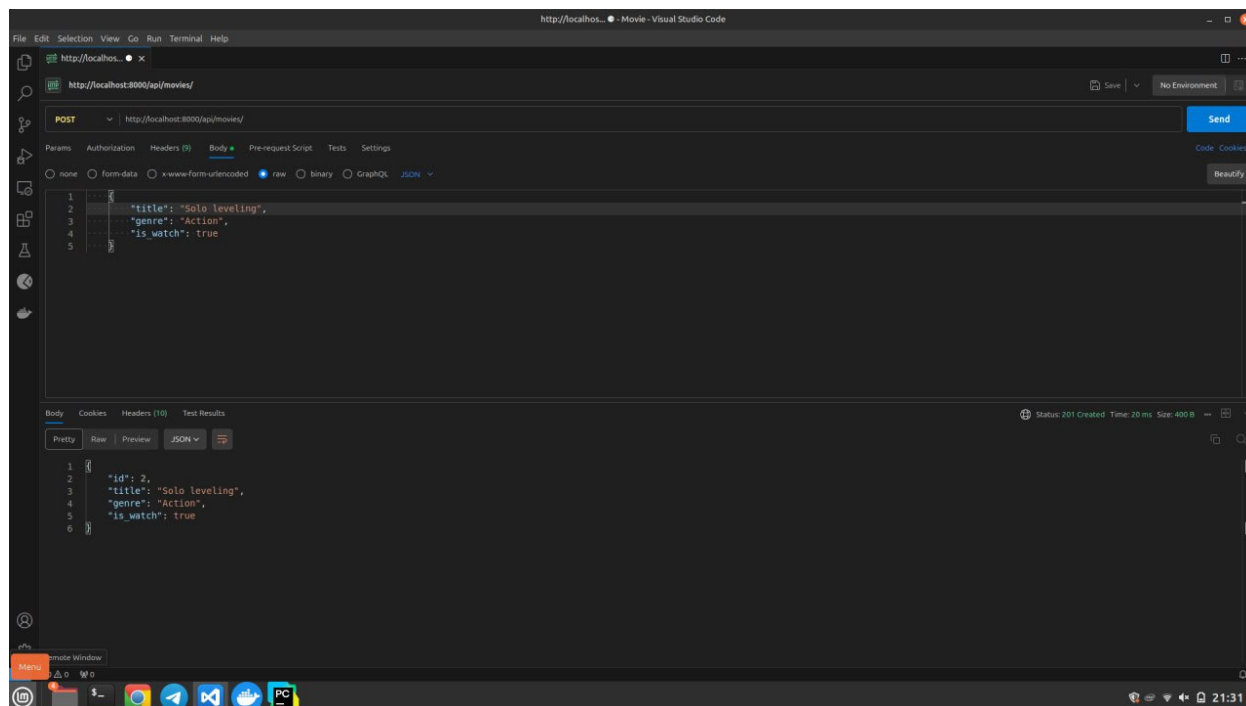




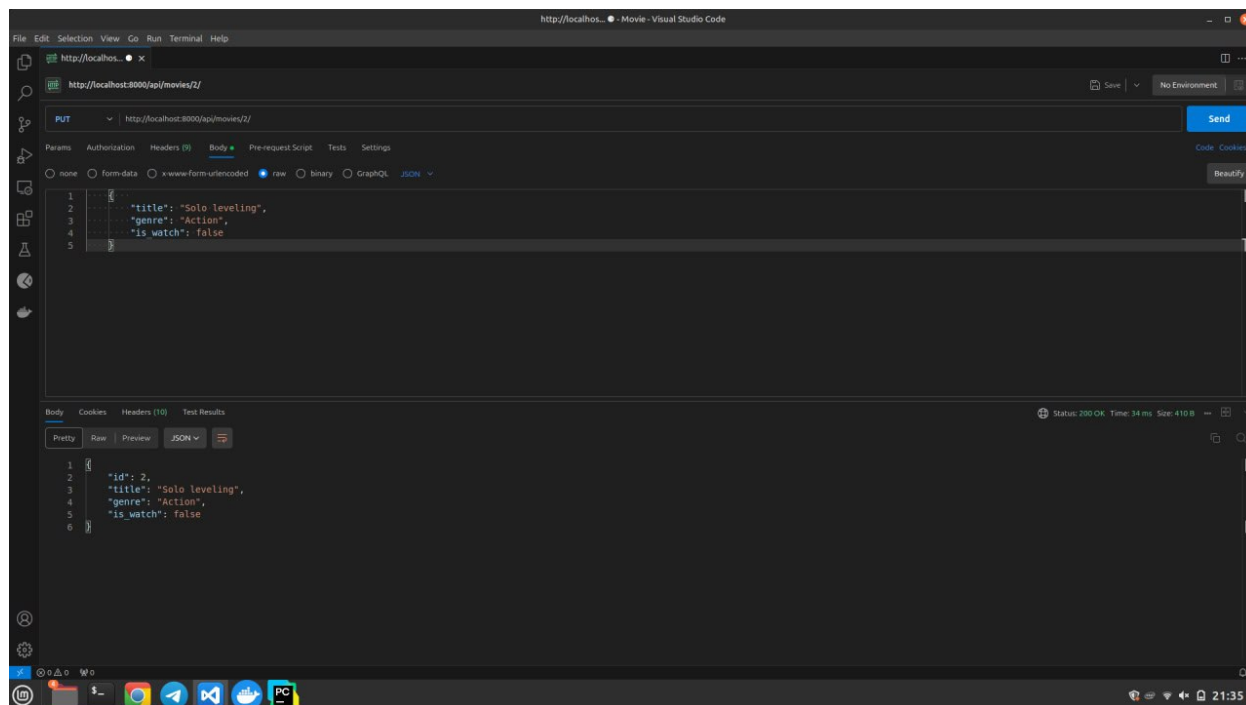
## متد GET و نتیجه آن

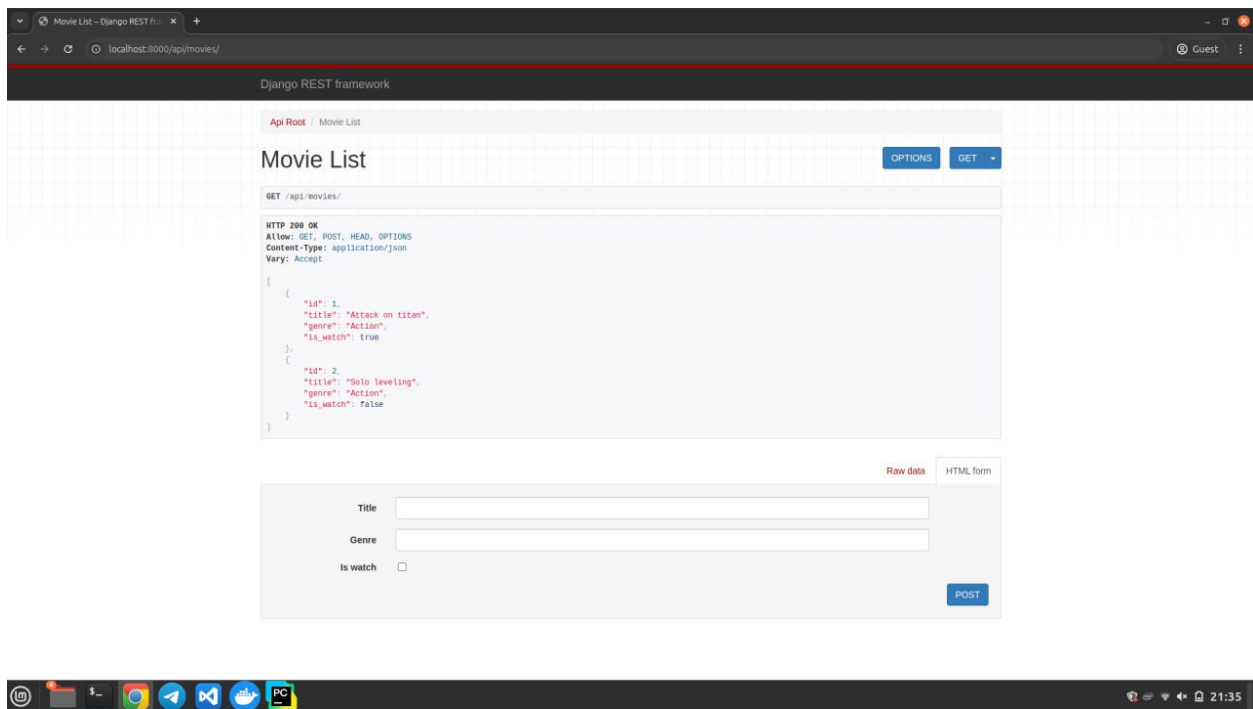


## متد POST و نتیجه آن



## متد PUT (Update) و نتیجه آن





## متد DELETE و نتیجه آن

