



UANL

UNIVERSIDAD AUTÓNOMA DE NUEVO LEÓN

FCFM

FACULTAD DE CIENCIAS FÍSICO MATEMÁTICAS



Docker - MySQL - Django CRUD

Estas en el menu

Carne Asada
500gr de carne de arrachera acompañada con salchichas y cebolla asada
\$340

Enchiladas Suizas
5 enchiladas banadas en salsa verde con queso, acompañadas de arroz y ensalada.
\$220

Pizza
Pizza de 14", puede elegir de entre los siguientes ingredientes: peperoni, jamon, salchicha, mexicana, hawaiana.
\$119

Mauricio Zuriel Torres Aguilar 2086228

Lenguajes Modernos de Programación 033

(HAY QUE TENER DOCKER INSTALADO)

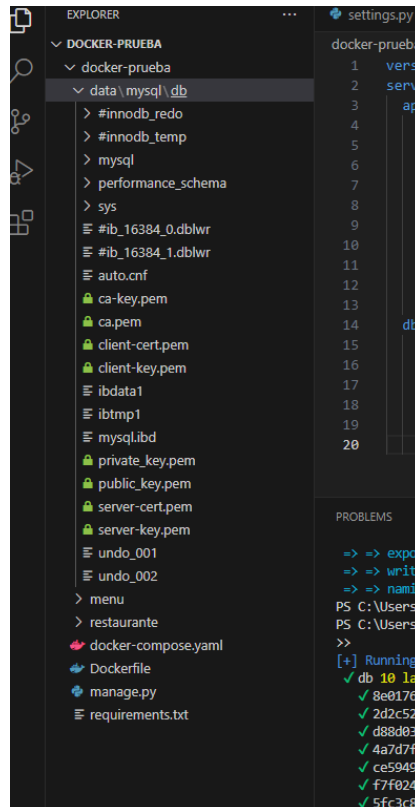
Para desplegar y correr la app primero hay que descargar el archivo zip y descomprimirlo, una vez descomprimido, nos dirigiremos a la ruta de ese archivo mediante la terminal, estando ahí ejecutaremos los siguientes comandos:

- **docker-compose build**

```
PS C:\Users\mauri\OneDrive\Escritorio\docker-prueba\docker-prueba> docker-compose build
>>
[+] Building 5.4s (10/10) FINISHED                                docker:default
=> [app internal] load .dockerignore                             0.8s
=> => transferring context: 2B                                   0.0s
=> [app internal] load build definition from Dockerfile          0.6s
=> => transferring dockerfile: 173B                               0.0s
=> [app internal] load metadata for docker.io/library/python:3.8 2.3s
=> [app auth] library/python:pull token for registry-1.docker.io 0.0s
=> [app 1/4] FROM docker.io/library/python:3.8@sha256:7a82536f5a2895b70416ccaffc49e6469d11ed8d9bf6bfcfc52328faeae7c7710 0.0s
=> [app internal] load build context                             0.2s
=> => transferring context: 37B                                   0.0s
=> CACHED [app 2/4] WORKDIR /django                             0.0s
=> CACHED [app 3/4] COPY requirements.txt requirements.txt       0.0s
=> CACHED [app 4/4] RUN pip3 install -r requirements.txt         0.0s
=> [app] exporting to image                                     0.2s
=> => exporting layers                                           0.0s
=> => writing image sha256:eda773a67f6551bffa21152cb70431d1df69c4bf99499c19dcd105808dfdc4d 0.0s
=> => naming to docker.io/library/app:django                    0.1s
PS C:\Users\mauri\OneDrive\Escritorio\docker-prueba\docker-prueba> █
```

- **docker-compose run --rm app django-admin startproject restaurante .**

```
PS C:\Users\mauri\OneDrive\Escritorio\docker-prueba\docker-prueba> docker-compose run --rm app django-admin startproject restaurante .
>>
[+] Running 11/11
✓ db 10 layers [#####] 0B/0B Pulled
✓ 8e0176adc18c Pull complete
✓ 2d2c52718f65 Pull complete
✓ d88d03ce139b Pull complete
✓ 4a7d7f11aa1e Pull complete
✓ ce5949193e4c Pull complete
✓ f7f024dfb329 Pull complete
✓ 5fc3c840facc Pull complete
✓ 509068e49488 Pull complete
✓ cbc847bab598 Pull complete
✓ 942bef62a146 Pull complete
[+] Building 0.0s (0/0)
[+] Creating 2/2
✓ Network docker-prueba default Created
✓ Container docker-prueba-db-1 Created
[+] Running 1/1
✓ Container docker-prueba-db-1 Started
[+] Building 0.0s (0/0)
CommandError: /django/manage.py already exists. Overlaying a project into an existing directory won't replace conflicting files.
PS C:\Users\mauri\OneDrive\Escritorio\docker-prueba\docker-prueba> █
```



- **docker-compose up**

```
PS C:\Users\mauri\OneDrive\Escritorio\docker-prueba\docker-prueba> docker-compose up
>>
[+] Building 0.0s (0/0)
[+] Running 2/2
  ✓ Container docker-prueba-db-1 Running
  ✓ Container django_container Created
Attaching to django_container, docker-prueba-db-1
docker-prueba-db-1 | 2023-11-20T06:55:37.324907Z 0 [Warning] [MY-010068] [Server] CA certificate ca.pem is self signed.
docker-prueba-db-1 | 2023-11-20T06:55:37.325382Z 0 [System] [MY-013602] [Server] Channel mysql_main configured to support TLS. Encrypted connections are now supported for
docker-prueba-db-1 | 2023-11-20T06:55:37.475397Z 0 [Warning] [MY-011810] [Server] Insecure configuration for --pid-file: Location '/var/run/mysqld' in the path is accessible
docker-prueba-db-1 | 2023-11-20T06:55:37.510269Z 0 [System] [MY-011323] [Server] X Plugin ready for connections. Socket: /var/run/mysqld/mysqlx.sock
docker-prueba-db-1 | 2023-11-20T06:55:37.510504Z 0 [System] [MY-010931] [Server] /usr/sbin/mysqld: ready for connections. Version: '8.2.0' socket: '/var/run/mysqld/mysqld
docker-prueba-db-1 | 2023-11-20T06:55:37.512444Z 0 [System] [MY-015016] [Server] MySQL Server - end.
docker-prueba-db-1 | 2023-11-20 06:55:37+00:00 [Note] [Entrypoint]: Temporary server started.
docker-prueba-db-1 | '/var/lib/mysql/mysql.sock' -> '/var/run/mysqld/mysqld.sock'
django_container | Watching for file changes with StatReloader
django_container | Performing system checks...
django_container |
django_container | System check identified no issues (0 silenced).
django_container | Exception in thread django-main-thread:
django_container | Traceback (most recent call last):
django_container |   File "/usr/local/lib/python3.8/site-packages/pymysql/connections.py", line 644, in connect
```

Iniciaremos el contenedor, pero no funcionará correctamente, para eso hay que hacer las migraciones, abrimos otra terminal e insertamos los siguientes comandos:


- `docker exec -it django_container /bin/bash`
- `python manage.py migrate`

```

PS C:\Users\mauri\OneDrive\Escritorio\docker-prueba\docker-prueba> docker exec -it django_container /bin/bash
>>
root@9bde29242c88:/django# python manage.py migrate
Operations to perform:
  Apply all migrations: admin, auth, contenttypes, menu, 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 menu.0001_initial... OK
  Applying sessions.0001_initial... OK
root@9bde29242c88:/django#

```

Después de las migraciones reiniciamos el contenedor, esto lo podemos hacer desde docker desktop solamente dándole click en la sección Actions, detenemos y volvemos a iniciar.

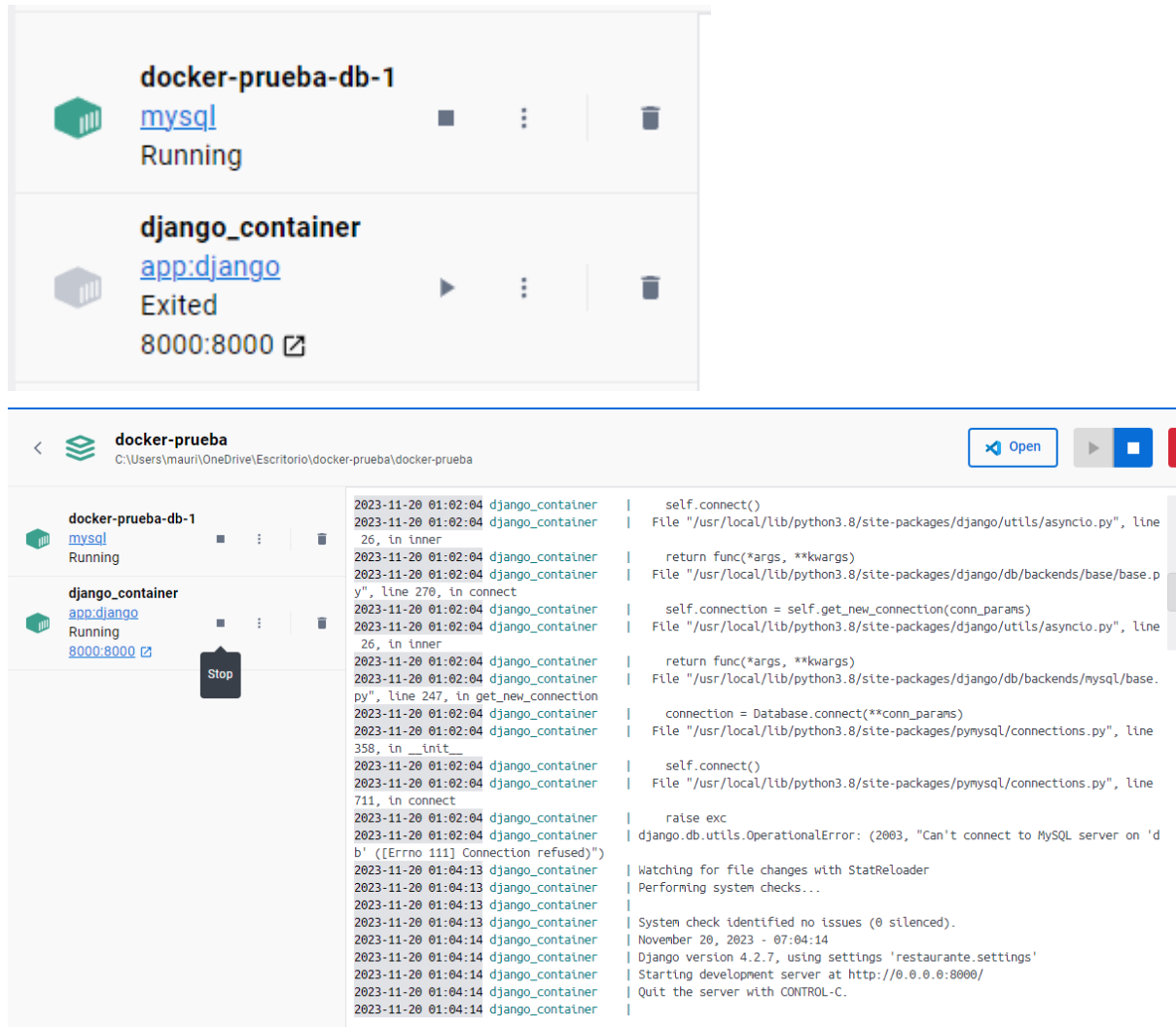
<input type="checkbox"/>	Name	Image	Status	CPU (%)	Port(s)	Last started	Actions
<input type="checkbox"/>	>  docker-prueba		Exited	N/A		3 minutes ago	▶ ⋮ 🗑

```

2023-11-20 01:02:04 django_container | File "/usr/local/lib/python3.8/site-packages/django/db/backends/base/base.py", line 289, in ensure_connection
2023-11-20 01:02:04 django_container | self.connect()
2023-11-20 01:02:04 django_container | File "/usr/local/lib/python3.8/site-packages/django/db/utils.py", line 91, in __exit__
2023-11-20 01:02:04 django_container | raise dj_exc_value.with_traceback(traceback) from exc_value
2023-11-20 01:02:04 django_container | File "/usr/local/lib/python3.8/site-packages/django/db/backends/base/base.py", line 289, in ensure_connection
2023-11-20 01:02:04 django_container | self.connect()
2023-11-20 01:02:04 django_container | File "/usr/local/lib/python3.8/site-packages/django/utils/asyncio.py", line 26, in inner
2023-11-20 01:02:04 django_container | return func(*args, **kwargs)
2023-11-20 01:02:04 django_container | File "/usr/local/lib/python3.8/site-packages/django/db/backends/base/base.py", line 270, in connect
2023-11-20 01:02:04 django_container | self.connection = self.get_new_connection(conn_params)
2023-11-20 01:02:04 django_container | File "/usr/local/lib/python3.8/site-packages/django/utils/asyncio.py", line 26, in inner
2023-11-20 01:02:04 django_container | return func(*args, **kwargs)
2023-11-20 01:02:04 django_container | File "/usr/local/lib/python3.8/site-packages/django/db/backends/mysql/base.py", line 247, in get_new_connection
2023-11-20 01:02:04 django_container | connection = Database.connect(**conn_params)
2023-11-20 01:02:04 django_container | File "/usr/local/lib/python3.8/site-packages/pymysql/connections.py", line 358, in __init__
2023-11-20 01:02:04 django_container | self.connect()
2023-11-20 01:02:04 django_container | File "/usr/local/lib/python3.8/site-packages/pymysql/connections.py", line 711, in connect
2023-11-20 01:02:04 django_container | raise exc
2023-11-20 01:02:04 django_container | django.db.utils.OperationalError: (2003, "Can't connect to MySQL server on 'c
b' ([Errno 111] Connection refused)")

```

En algunas ocasiones aparecerá este error, lo que pasa es que como los dos contenedores se inician con un solo click, el de la app al buscar la base de datos puede que aún no haya iniciado totalmente, esto se resuelve reiniciando el contenedor con la app.



The screenshot shows the Docker Desktop interface. At the top, two containers are listed: 'docker-prueba-db-1' (mysql) which is 'Running', and 'django_container' (app:django) which is 'Exited' with port '8000:8000'. Below this, a terminal window displays the logs for the 'django_container'. The logs show a series of timestamps and messages from the Django application, including a connection error to the MySQL server on 'localhost:8000'.

```
2023-11-20 01:02:04 django_container | self.connect()
2023-11-20 01:02:04 django_container | File "/usr/local/lib/python3.8/site-packages/django/utils/asyncto.py", line
26, in inner
2023-11-20 01:02:04 django_container | return func(*args, **kwargs)
2023-11-20 01:02:04 django_container | File "/usr/local/lib/python3.8/site-packages/django/db/backends/base/base.p
y", line 270, in connect
2023-11-20 01:02:04 django_container | self.connection = self.get_new_connection(conn_params)
2023-11-20 01:02:04 django_container | File "/usr/local/lib/python3.8/site-packages/django/utils/asyncto.py", line
26, in inner
2023-11-20 01:02:04 django_container | return func(*args, **kwargs)
2023-11-20 01:02:04 django_container | File "/usr/local/lib/python3.8/site-packages/django/db/backends/mysql/base.
py", line 247, in get_new_connection
2023-11-20 01:02:04 django_container | connection = Database.connect(**conn_params)
2023-11-20 01:02:04 django_container | File "/usr/local/lib/python3.8/site-packages/pymysql/connections.py", line
358, in __init__
2023-11-20 01:02:04 django_container | self.connect()
2023-11-20 01:02:04 django_container | File "/usr/local/lib/python3.8/site-packages/pymysql/connections.py", line
711, in connect
2023-11-20 01:02:04 django_container | raise exc
2023-11-20 01:02:04 django_container | django.db.utils.OperationalError: (2003, "Can't connect to MySQL server on 'd
b' ([Errno 111] Connection refused)")
2023-11-20 01:04:13 django_container | Watching for file changes with StatReloader
2023-11-20 01:04:13 django_container | Performing system checks...
2023-11-20 01:04:13 django_container |
2023-11-20 01:04:13 django_container | System check identified no issues (0 silenced).
2023-11-20 01:04:14 django_container | November 20, 2023 - 07:04:14
2023-11-20 01:04:14 django_container | Django version 4.2.7, using settings 'restaurant.settings'
2023-11-20 01:04:14 django_container | Starting development server at http://0.0.0.0:8000/
2023-11-20 01:04:14 django_container | Quit the server with CONTROL-C.
2023-11-20 01:04:14 django_container |
```

Y como vemos ya funciona, para usar la aplicación vamos hacia

<http://localhost:8000/menu/>

(si no funciona intentar con <http://0.0.0.0:8000/menu/>)

Imagen del funcionamiento:

