



Running Spring Boot with Docker Compose

Ömer Naci Soydemir

Senior Software Engineer





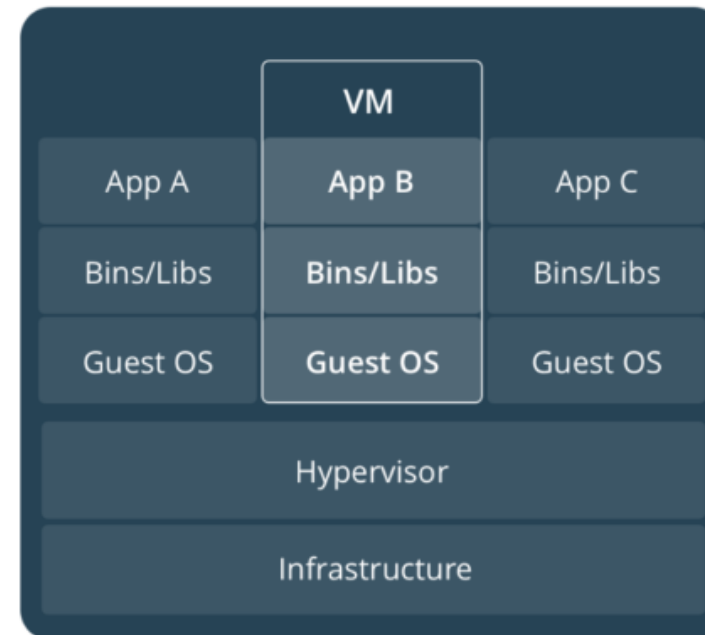
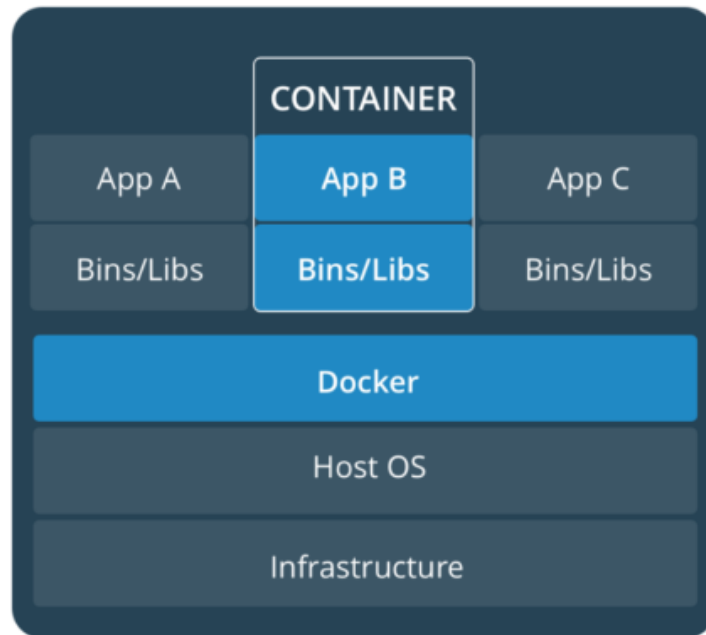
Roadmap

- Introduction to Docker
- Basic Docker Commands
- Docker Compose
- Running Spring Boot with PostgreSQL in Docker Compose
- Conclusion

Introduction to Docker

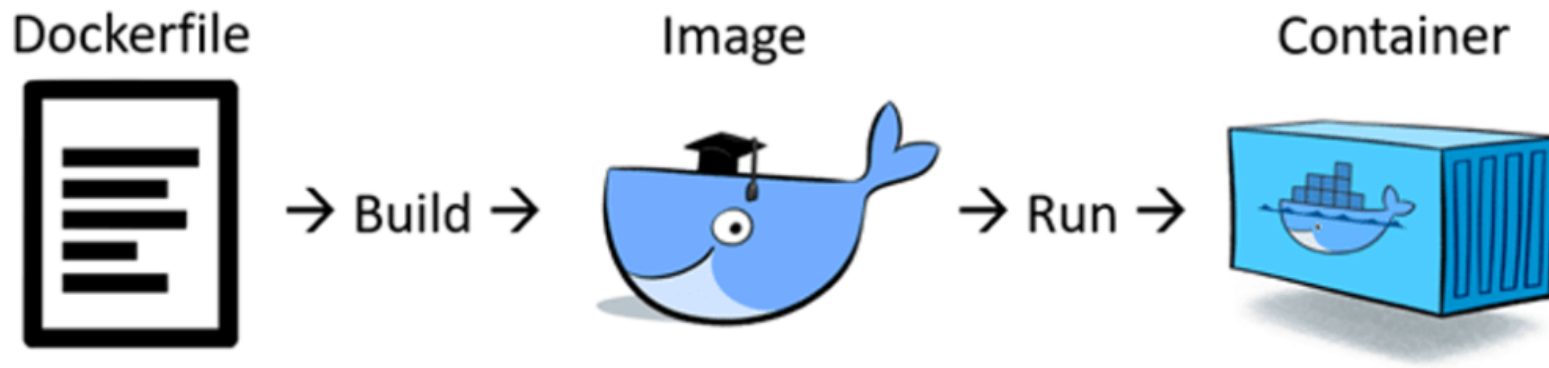
- Docker is basically a container engine that uses the Linux kernel features.
- Docker's virtualization structure does not have a Hypervisor layer.
- It accesses the host operating system through the Docker Engine and uses shared system tools.
- It consumes less system resources than conventional VMs.

Introduction to Docker



Introduction to Docker

- *Dockerfile* : A script that defines how an application packages to be run as a container.



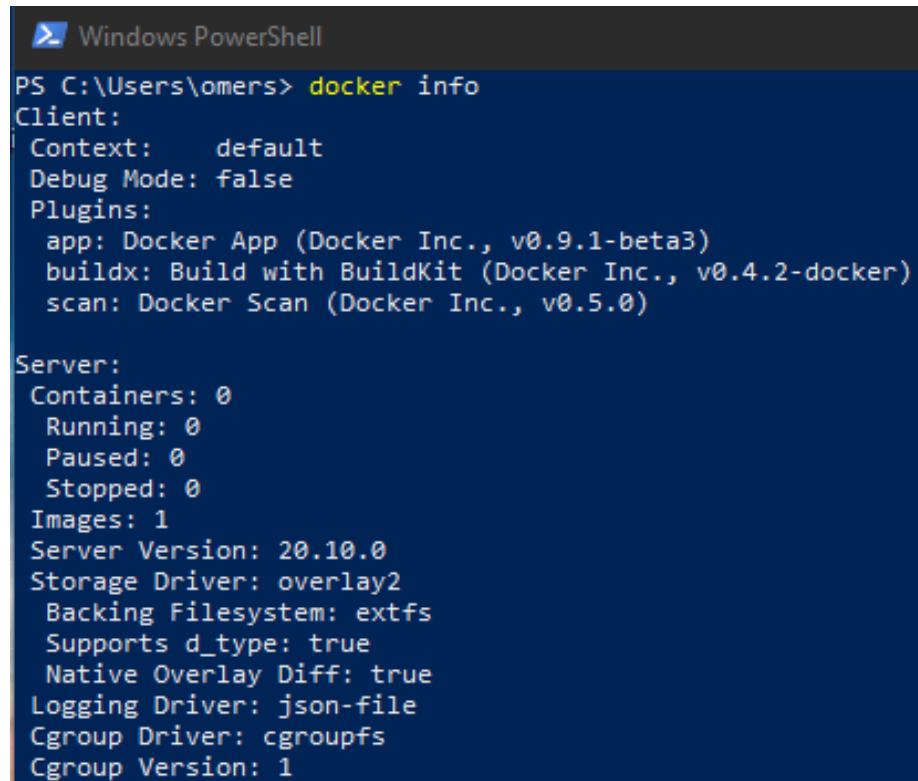
```
FROM adoptopenjdk/openjdk11:jre-11.0.6_10-alpine
COPY build/libs/ /opt/apps/
EXPOSE 8080
ENTRYPOINT ["java", "-jar", "/opt/apps/docker-demo-0.0.1-SNAPSHOT.jar"]
```

Introduction to Docker

- ***Docker registry*** : It is an environment in which images are kept and distributed.
- ***Docker daemon*** : Allows containers to work in isolation from each other, makes the use and distribution of resources.
- ***Docker image*** : An executable application package that the docker service creates that will use dockerfile and base image.
- ***Docker container*** : Containers are working examples of docker images. Running an image creates a docker container.

Basic Docker Commands

- *docker info* : This command displays system wide information regarding the Docker installation.

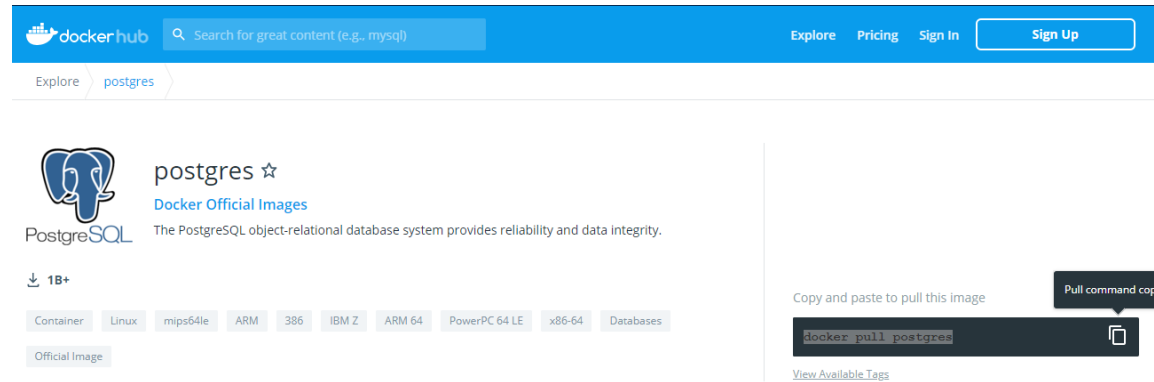


```
Windows PowerShell
PS C:\Users\omers> docker info
Client:
 Context:    default
 Debug Mode: false
 Plugins:
  app: Docker App (Docker Inc., v0.9.1-beta3)
  buildx: Build with BuildKit (Docker Inc., v0.4.2-docker)
  scan: Docker Scan (Docker Inc., v0.5.0)

Server:
 Containers: 0
  Running: 0
  Paused: 0
  Stopped: 0
 Images: 1
 Server Version: 20.10.0
 Storage Driver: overlay2
  Backing Filesystem: extfs
  Supports d_type: true
  Native Overlay Diff: true
 Logging Driver: json-file
 Cgroup Driver: cgroupfs
 Cgroup Version: 1
```

Basic Docker Commands

- *`docker pull image_name`* : Download an image, and all its parents, from the registry.



```
Windows PowerShell
PS C:\Users\omers> docker pull postgres
Using default tag: latest
latest: Pulling from library/postgres
6ec7b7d162b2: Already exists
a7d6065bfd90: Pull complete
1b99004acb42: Pull complete
e11ab41f0489: Pull complete
eaa59f6a7cf0: Pull complete
e27743da9368: Pull complete
10ffdd8ade29: Pull complete
bdaa066489bc: Pull complete
18fcf6a7b457: Downloading [=====>] 14.51MB/73.68MB
7e48453e674b: Download complete
d6a3ea95116d: Download complete
0cd33e8b20bb: Download complete
85eddbec7cca: Download complete
a4a32e0796c6: Download complete
```


Basic Docker Commands

- *docker images* : Lists all local docker images.
- *docker image rm image_name (docker rmi image_name)* : Delete an image from the local image store.

```
Windows PowerShell
PS C:\Users\omers> docker images
REPOSITORY          TAG             IMAGE ID        CREATED         SIZE
omernaci/demo        latest          97bed29f14a1    7 hours ago    194MB
postgres             latest          1f1bd4302537    3 days ago     314MB
PS C:\Users\omers> docker image rm postgres
Untagged: postgres:latest
Deleted: sha256:a160a47b02773829c9927b1a85561e138f71f52ca7a517c309773c1c44137c1e
Deleted: sha256:1f1bd43025375353fab1d82299c57b7c0b1c136b79279f3595f6ec3078775d4d
Deleted: sha256:82a93dc4c7e272756d024484afb12bde8be48a1e5ab3859a176b2e0ca6814b7
Deleted: sha256:570b6cf875c2391e9f75d77389c791752c95c76bc53e905ed5fe221c8a0bafdb
Deleted: sha256:e0554ab481fcc3a8b5bc99ade5b3ddcfff975bf8e9af21397af315117653b794
Deleted: sha256:73faa64d73ecc9cb5aed151eb8153f06b450b1bc5bdd8c8dba4fe942e2195791
Deleted: sha256:1398d248f494ed6b8f151a78731fbed8ffd8466a7dac3231d3c2ef507499ec57
Deleted: sha256:da0ccb282303628c993237898904ee145e5aa408c3a2de4dab99709d520c5771
Deleted: sha256:502536ce3a8f6549572c469ec8ba9388a3a4a317325ffa410e789319b5ab64f9
Deleted: sha256:c83e929f0b2bc4a1b058835bf052bad7ca07727fc243842759c6fc57976776d5
Deleted: sha256:66b90dd42432662a1749c2aaec3b8b7e1b3133f77df3fb1252c60de824115f5c
Deleted: sha256:9bec221aabbf0f3ce8a8ebba1ffb5efb0dbc8fb01d261a12a9544bd8effba76a
Deleted: sha256:620d34dc219010dd0ac62c566ba0266345fe30d43cfd38fc39bc020e37c451a1
Deleted: sha256:808b9a2b3f94c125e3ecfb95946fb606ecbf8cac2fb1135fbf299f664d529d9c
Deleted: sha256:701b4a21a8a2cf7f30d665dd7d81868eeb62c943d8edcbcd698d9ea21b07d48c
Deleted: sha256:87c8a1d8f54f3aa4e05569e8919397b65056aa71cdf48b7f061432c98475eee9
PS C:\Users\omers>
```

Basic Docker Commands

- *`docker build -t image_tag`* : Performs container build according to the definitions in dockerfile.

```
C:\Users\omers\Documents\workspace\workshops\docker\docker-demo>docker build -t omernaci/demo:latest .
[+] Building 1.4s (7/7) FINISHED
=> [internal] load build definition from Dockerfile
=> => transferring dockerfile: 32B
=> [internal] load .dockerignore
=> => transferring context: 2B
=> [internal] load metadata for docker.io/adoptopenjdk/openjdk11:jre-11.0.6_10-alpine
=> [internal] load build context
=> => transferring context: 120B
=> [1/2] FROM docker.io/adoptopenjdk/openjdk11:jre-11.0.6_10-alpine@sha256:a9295aa69d9dbbeb123c7d8b72a8bef5f3523f89523b77802a4d7f74f33df6a9
=> CACHED [2/2] COPY build/libs/ /opt/apps/
=> exporting to image
=> exporting layers
=> => writing image sha256:97bed29f14a112eec162ce1b63b181e1f2d48c5fcc1b8680cee5f247a4dc0e6d
=> => naming to docker.io/omernaci/demo:latest
```

Basic Docker Commands

- ***docker run*** : Creates a container from a given image and starts the container using a given command.

```
Windows PowerShell
PS C:\Users\omers> docker images
REPOSITORY    TAG       IMAGE ID       CREATED        SIZE
postgres      latest    1f1bd4302537   3 days ago    314MB
PS C:\Users\omers> docker run --name postgresql-container -p 5432:5432 -e POSTGRES_PASSWORD=password -d postgres
7734eb481505cfe55e3dc87e19177d5752bd197d2303c2be91af87026c1d8469
PS C:\Users\omers> docker ps
CONTAINER ID   IMAGE     COMMAND                  CREATED        STATUS        PORTS                    NAMES
7734eb481505   postgres  "docker-entrypoint.s..." 41 minutes ago Up 41 minutes  0.0.0.0:5432->5432/tcp    postgresql-container
PS C:\Users\omers>
```

Basic Docker Commands

- *docker ps (docker container ls)* : Creates a container from a given image and starts the container using a given command.

```
Windows PowerShell
PS C:\Users\omers> docker images
REPOSITORY    TAG       IMAGE ID       CREATED        SIZE
postgres      latest    1f1bd4302537   3 days ago    314MB
PS C:\Users\omers> docker run --name postgresql-container -p 5432:5432 -e POSTGRES_PASSWORD=password -d postgres
7734eb481505cfe55e3dc87e19177d5752bd197d2303c2be91af87026c1d8469
PS C:\Users\omers> docker ps
CONTAINER ID   IMAGE     COMMAND                  CREATED        STATUS        PORTS                    NAMES
7734eb481505   postgres  "docker-entrypoint.s..." 41 minutes ago Up 41 minutes  0.0.0.0:5432->5432/tcp    postgresql-container
PS C:\Users\omers>
```

Basic Docker Commands

- *docker exec* : When dealing with the interactive processes like bash, use the -i and -t options to start the container.

```
Windows PowerShell
PS C:\Users\omers> docker exec -it postgresql-container bash
root@7734eb481505:/# psql -h localhost -p 5432 -U postgres -W
Password:
psql (13.1 (Debian 13.1-1.pgdg100+1))
Type "help" for help.

postgres=# \l

               List of databases
  Name      | Owner   | Encoding | Collate |  Ctype  | Access privileges
-----+-----+-----+-----+-----+-----
 articledb  | postgres | UTF8     | en_US.utf8 | en_US.utf8 | 
 postgres  | postgres | UTF8     | en_US.utf8 | en_US.utf8 | 
 template0  | postgres | UTF8     | en_US.utf8 | en_US.utf8 | =c/postgres +
            |          |          |          |          | postgres=CTc/postgres
 template1  | postgres | UTF8     | en_US.utf8 | en_US.utf8 | =c/postgres +
            |          |          |          |          | postgres=CTc/postgres
(4 rows)

postgres=#
```

Basic Docker Commands

- *docker logs* : Displays the logs that have occurred up to that moment in the corresponding container's terminal.
- *-f* continues to show logs formed after that moment with the follow parameter.

```
Windows PowerShell
PS C:\Users\omers> docker ps
CONTAINER ID   IMAGE          COMMAND                  CREATED        STATUS        PORTS                    NAMES
c18665f6748f   omernaci/demo:latest   "java -jar /opt/apps..."  14 seconds ago  Up 11 seconds        0.0.0.0:8080->8080/tcp    article-app
24166f18684f   postgres:latest       "docker-entrypoint.s..."  14 seconds ago  Up 13 seconds (health: starting)  0.0.0.0:5432->5432/tcp    postgresql-container

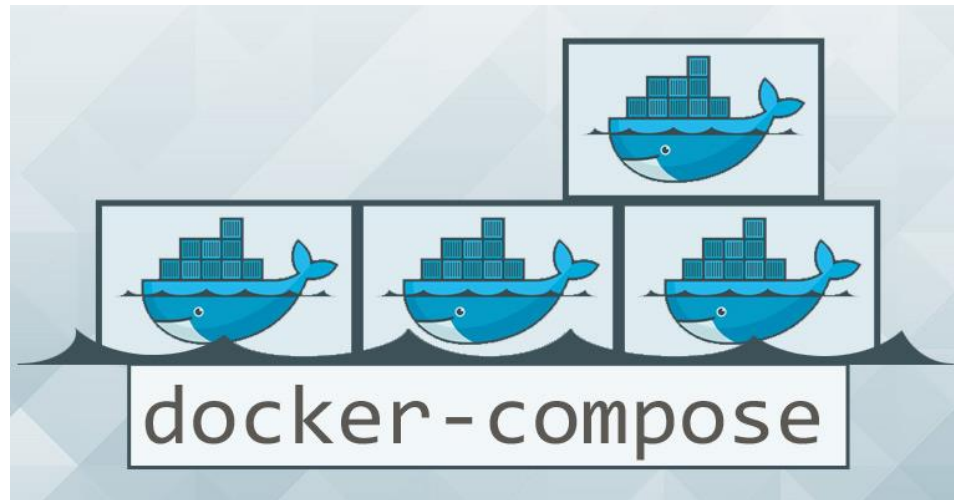
PS C:\Users\omers> docker logs -f c18665f6748f

:: Spring Boot ::
(v2.4.1)

2021-01-03 12:55:43.521 INFO 1 --- [main] c.o.dockerdemo.DockerDemoApplication : Starting DockerDemoApplication using Java 11.0.6 on c18665f6748f with P
ID 1 (/opt/apps/docker-demo-0.0.1-SNAPSHOT.jar started by root in /)
2021-01-03 12:55:43.534 INFO 1 --- [main] c.o.dockerdemo.DockerDemoApplication : No active profile set, falling back to default profiles: default
2021-01-03 12:55:45.257 INFO 1 --- [main] o.apache.catalina.core.StandardService : Starting service [Tomcat]
2021-01-03 12:55:45.257 INFO 1 --- [main] org.apache.catalina.core.StandardEngine : Starting Servlet engine: [Apache Tomcat/9.0.41]
2021-01-03 12:55:45.326 INFO 1 --- [main] o.a.c.c.C.[Tomcat].[localhost].[/] : Initializing Spring embedded WebApplicationContext
2021-01-03 12:55:45.573 INFO 1 --- [main] o.hibernate.jpa.internal.util.LogHelper : HHH000204: Processing PersistenceUnitInfo [name: default]
2021-01-03 12:55:45.665 INFO 1 --- [main] org.hibernate.Version : HHH000412: Hibernate ORM core version 5.4.25.Final
2021-01-03 12:55:45.882 INFO 1 --- [main] o.hibernate.annotations.common.Version : HCANN000001: Hibernate Commons Annotations {5.1.2.Final}
2021-01-03 12:55:45.987 INFO 1 --- [main] com.zaxxer.hikari.HikariDataSource : HikariPool-1 - Starting...
2021-01-03 12:55:46.073 INFO 1 --- [main] com.zaxxer.hikari.HikariDataSource : HikariPool-1 - Start completed.
2021-01-03 12:55:46.107 INFO 1 --- [main] org.hibernate.dialect.Dialect : HHH000400: Using dialect: org.hibernate.dialect.PostgreSQL10Dialect
2021-01-03 12:55:46.655 INFO 1 --- [main] o.h.e.t.j.p.i.JtaPlatformInitiator : HHH000490: Using JtaPlatform implementation: [org.hibernate.engine.tran
saction.jta.platform.internal.NoJtaPlatform]
2021-01-03 12:55:47.267 INFO 1 --- [main] pertySourcedRequestMappingHandlerMapping : Mapped URL path [/v2/api-docs] onto method [springfox.documentation.swa
gger2.web.Swagger2Controller#documentation(String, HttpServletRequest)]
2021-01-03 12:55:47.710 INFO 1 --- [main] d.s.w.p.DocumentationPluginsBootstrapper : Context refreshed
2021-01-03 12:55:47.754 INFO 1 --- [main] d.s.w.p.DocumentationPluginsBootstrapper : Found 1 custom documentation plugin(s)
2021-01-03 12:55:47.787 INFO 1 --- [main] s.d.s.w.s.ApiListingReferenceScanner : Scanning for api listing references
2021-01-03 12:55:47.924 INFO 1 --- [main] c.o.dockerdemo.DockerDemoApplication : Started DockerDemoApplication in 4.994 seconds (JVM running for 5.692)
```

Docker Compose

- Docker compose is a “yaml” - based file that allows applications that can run in container to be run and stopped with a single command.
- The *docker-compose* tool is currently offered by Docker Inc, which also provides the Docker CLI and Docker daemon




```
version: "3"

networks:
  demo-net:

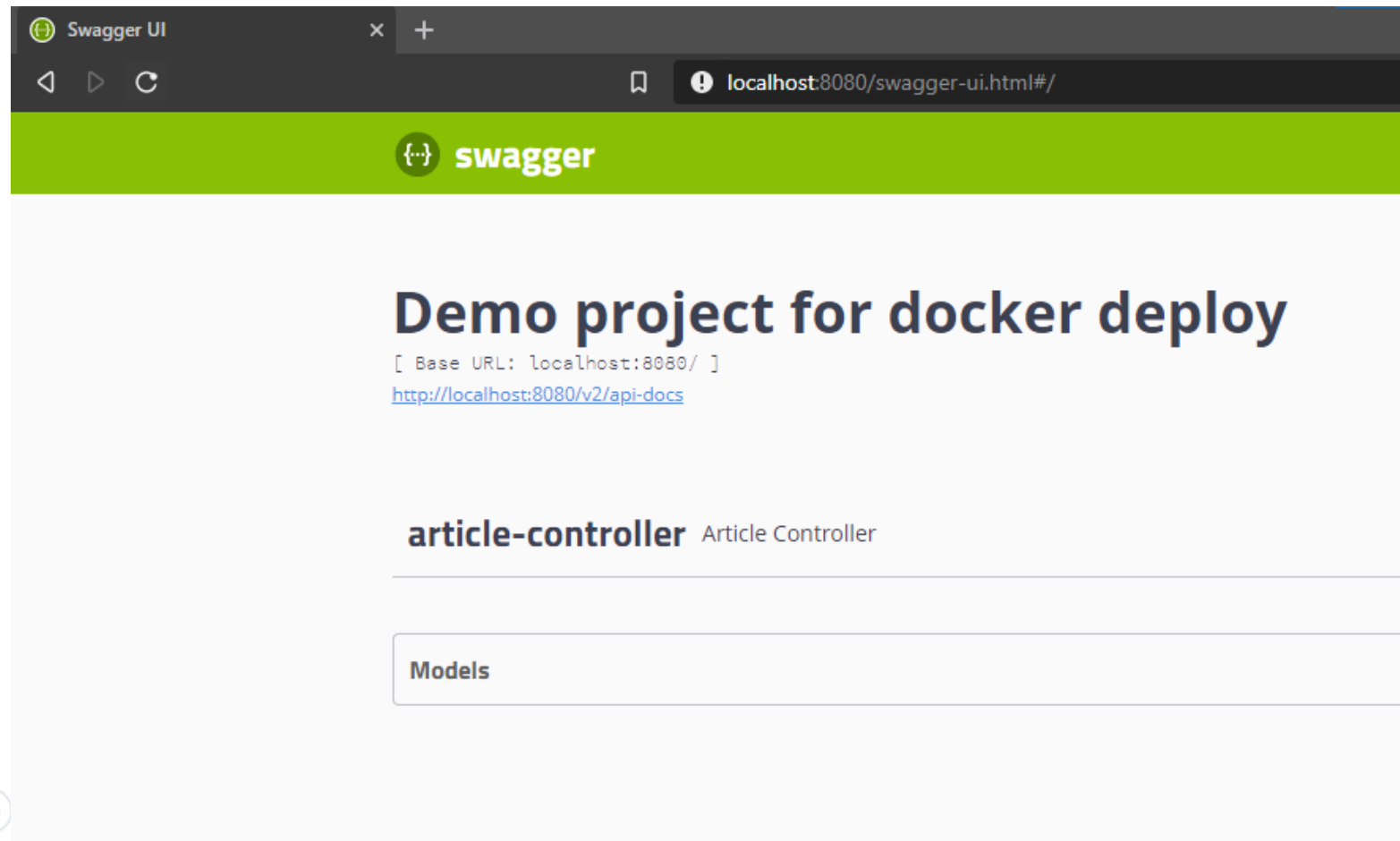
services:
  # Database Service (Postgres)
  db:
    image: postgres:latest
    container_name: postgresql-container
    restart: always
    #volumes:
    # - postgres-data:/var/lib/postgresql/data
    networks:
      - demo-net
    ports:
      - "5432:5432"
    environment:
      POSTGRES_USER: postgres
      POSTGRES_PASSWORD: password
      POSTGRES_DB: articledb
      PGDATA: /var/lib/postgresql/data/pgdata
    healthcheck:
      test: pg_isready -U postgres
      interval: 1m
      timeout: 10s
      retries: 2

  # Article Service
  app:
    image: omernaci/demo:latest
    container_name: article-app
    ports:
      - "8080:8080"
    networks:
      - demo-net
    environment:
      - SPRING_DATASOURCE_URL=jdbc:postgresql://db/articledb
      - SPRING_DATASOURCE_USERNAME=postgres
      - SPRING_DATASOURCE_PASSWORD=password
    depends_on:
      - db
```


Running Spring Boot with PostgreSQL in Docker Compose

- ***docker-compose up*** : It sets up the network for the broadcast of services and stands up all the services contained in the yml file.
- **-d** parameter, docker compose runs in detached mode.
- ***docker-compose down*** : It completely deletes services.
- ***docker-compose stop***: Docker environment does not work, but the network is available.

Running Spring Boot with PostgreSQL in Docker Compose



Conclusion

- It starts in seconds.
- The containers it contains are just a process its lightweight structure makes Docker fast.
- Enables faster deployment process.
- Programmers who want to work on different servers just need to run Docker images.



Time for Questions!

