1-Criar uma rede Docker

> docker network create hr-net

2- Baixar uma imagem docker postgres

> docker pull postgres:12-alpine

3- Rodar um container docker

> docker run -p 5432:5432 –name hr-worker-pg12 –network hr-net -e POSGRES\_PASSWORD=1234567 -e POSTGRES\_DB=db\_hr\_worker postgres:12-alpine

> docker run -p 5433:5432 --name hr-user-pg12 --network hr-net -e POSTGRES\_PASSWORD=1234567 -e POSTGRES\_DB=db\_hr\_user postgres:12-alpine

4- hr-config-server

./mvnw clean package

docker build -t hr-config-server:v1 .

docker run -p 8888:8888 --name hr-config-server --network hr-net -e GITHUB\_USER=douglasdjf -e GITHUB\_PASS= hr-config-server:v1

OBS: se o repositório for public não precisa colocar a senha no GITHUB\_PASS

4- hr-eureka-server

mvnw clean package

docker build -t hr-eureka-server:v1 .

docker run -p 8761:8761 --name hr-eureka-server --network hr-net hr-eureka-server:v1

5- hr-worker

mvnw clean package -DskipTests

docker build -t hr-worker:v1 .

docker run -P --network hr-net hr-worker:v1

6- hr-user

mvnw clean package -DskipTests

docker build -t hr-user:v1 .

docker run -P --network hr-net hr-user:v1

7- hr-payrol

mvnw clean package -DskipTests

docker build -t hr-payroll:v1 .

docker run -P --network hr-net hr-payroll:v1

8- hr-oauth

mvnw clean package -DskipTests

docker build -t hr-oauth:v1 .

docker run -P --network hr-net hr-oauth:v1

9- hr-api-gateway-zull

mvnw clean package -DskipTests

docker build -t hr-api-gateway-zuul:v1 .

docker run -p 8765:8765 --name hr-api-gateway-zuul --network hr-net hr-api-gateway-zuul:v1

10- Alguns comandos dokcer

1.**Criar uma rede Docker:**

docker network create <nome-da-rede>

2.**Baixar imagem do Dockerhub**

docker pull <nome-da-imagem:tag>

3.**Ver imagens**

docker images

4.**Criar/rodar um container de uma imagem**

docker run -p <porta-externa>:<porta-interna> --name <nome-do-container> --network <nome-da-rede> <nome-da-imagem:tag>

5. **Listar containers**

docker os

docker ps -a

6. **Acompanhar logs do container em execução**

docker logs -f <container-id>