



# Curso de Docker

## 5. Docker Compose

Mateus Schwede - UB Social

# Conceito

- Arquivo do tipo *yaml* chamado '*docker-compose.yml*', contendo composto (infraestrutura IAC) de configurações para automatização docker e comunicação entre containers (via nome estabelecido) e volumes, podendo haver versionamento e, com isso, melhor organização.

Arquivo que permite criar toda infraestrutura de Containers, juntamente com suas respectivas configurações

# docker-compose.yml

```
version: "3.7"
services:
  app:
    image: node:12-alpine
    container_name: containerNode
    command: sh -c "yarn install && yarn run dev"
    ports:
      - 3000:3000
    working_dir: /app
    volumes:
      - ./:/app
    environment:
      MYSQL_HOST: mysql
      MYSQL_USER: root
      MYSQL_PASSWORD: senhaMySQL
      MYSQL_DB: nomeBD
    restart: always
  mysql:
    image: mysql:5.7
    container_name: containerBD
    volumes:
      - volumeDockerCompose:/var/lib/mysql
    environment:
      MYSQL_ROOT_PASSWORD: senhaMySQL
      MYSQL_DATABASE: nomeBD
    expose:
      - "5000"
    restart: always
    depends_on:
      - app
volumes:
  volumeDockerCompose: {}
```

# Estrutura

- **version:** Versão do Docker Compose utilizado (Opcional informar no docker-compose.yml);
- **services:** Containers que compõem o docker-compose.yml (Cada service possuirá um nome de preferência do usuário, para referenciá-lo);
- **build:** Alternativo ao *image*, mostra o diretório do Dockerfile para construção da Imagem;
- **image:** Nome da Imagem a ser construída;
- **container\_name:** Nome do Container (De preferência do usuário);
- **ports:** Indica, respectivamente, portaHost:portaContainer que serão liberadas para requisições entre ambos;
- **expose:** Indica porta do Container que será exposta;
- **working\_dir:** Diretório especificado, que direcionará o usuário ao entrar no Container;

# Estrutura

- **deploy:** Determina regras de execução do Container, como nodes referenciados, métodos de execução, limitações de hardware, etc;
- **restart:** Comportamento em caso de queda do Container;
  - *no*: Quando Container cai, o Compose não o excluirá;
  - *always*: Quando Container cai, o Compose sempre o reiniciará;
  - *on-failure*: Em caso de apontamento de falha, executará Container novamente;
  - *unless-stopped*: Sempre tentará garantir o estado do Container em execução, ao menos que o mesmo seja colocado em *stopped* manualmente.
- **restart\_policy:** Usado dentro do *deploy*, para informar regras de reinicialização e prevenção à falhas, como quantidade de tentativas para iniciar o Container, tempo de espera para inicialização, etc;
- **networks:** Definir redes que serão criadas e relacionadas junto ao Container, informando, respectivamente em cada linha dependente: nomeRede, tipoRede, driverRede e subnet;
- **volumes:** Indicar volumes, onde o padrão é: *nomeVolume:diretorioInternoContainer/* ou personalizado: *diretorioInternoHostDocker:diretorioInternoContainer/*

# Estrutura

- **environment:** Variáveis de ambiente do Container, na sintaxe *NOME\_VARIAVEL: valor*;
- **env\_file:** Diretório do arquivo com lista de variáveis de ambiente onde, dentro dele, há uma variável por linha;
- **depends\_on:** Containers que precisam existir para que o em questão funcione (Informar nome do service para referenciá-lo);
- **links:** Mesmo que `depends_on`, mas que indica também link entre outros Containers;
- **command** e **entrypoint:** Comandos shell personalizados que serão executados durante o up da criação do Container;
- **labels:** Para labels personalizadas (tituloLabel: "descricaoLabel");
- **#comentario:** Para comentário no arquivo docker-compose.yml (Não entrará em execução no up);

# Comandos

- **Versão docker-compose:** docker-compose version
- **Executar containers do docker-compose:** docker-compose up (-d executa em background, build ao invés de up somente fará build das Imagens, -f seguido do diretório especificará o local e nome de arquivo do docker-compose.yml)
- **Listar containers via docker-compose:** docker-compose ps
- **Listar logs de container no docker-compose:** docker-compose logs -f nomeContainer
- **Parar docker-compose/containers:** docker-compose stop (Ou down, para parar e remover container e componentes, start inicia e restart reinicia)
- **Entrar em container via docker-compose:** docker-compose exec nomeContainer bash
- **Aumentar réplicas de container:** docker-compose scale

(Pode-se, dentro de Container, citar outros containers pelos nomes personalizados)

# Recapitulando

Container

Docker e Docker Hub

Imagem

Dockerfile

Gestão de Containers - Docker

Volume

Docker Compose





# UB Social

[ubsocial.github.io](https://ubsocial.github.io)