

# SQL\_001

---

## SQL\_001

---

- Criando um banco de dados com docker
- Select + Where + Group by
- CTEs + Window functions
- Deduplicando dados
- Criando, atualizando e deletando tabelas e views
- Acessando um banco de dados com Python
- Conectando a uma ferramenta de BI

### Docker (Criando um banco de dados com docker)

Primeiramente o que é o Docker?

o Docker é uma evolução da VM, acredito que o maior mérito do Docker é poder abolir a frase: "Putz, mas no meu pc roda!"

para mostrar como funciona vamos montar um HTML-SERVER simples com uma página de "Hello world"

1 - Criamos um arquivo *index.html* em uma pasta com o nome de web

2 - na raiz de nossa aula criamos um arquivo chamado *Dockerfile* esse arquivo é a estrutura base de uma imagem

3 - no arquivo colocamos o seguinte código:

```
#FROM: QUAL IMAGEM E QUAL VERSÃO?  
FROM httpd  
#COPIA ARQUIVOS DA MAQUINA PARA O CONTAINER NO BUILD  
COPY ./web/ /usr/local/apache2/htdocs/  
EXPOSE 80
```

pronto! nosso servidor esta preparado para subir ele primeiro precisamos montar a imagem *build* e depois subir o container *run*

```
docker build -t web_apache .
```

Onde:

- *-t* é para nomear nossa imagem, caso contrário o docker dará um nome qualquer
- *web\_apache* é o nome que escolhemos

- `.` é o local onde esta nosso *Dockerfile*

```
docker run -d -p 80:80 web_apache
```

Onde:

- `-d` é o detach ou seja, executa sem travar o terminal
- `-p 80:80` é o mapeamento de portas, aqui ele esta encaminhando a porta 80 do container na porta 80 da nossa maquina, não necessariamente precisa ser o mesmo numero podemos usar um `80:4242`
- `web_apache` é o nome da nossa imagem

Alguns comando importantes:

`docker ps`: mostra os containers que temos ativos

`docker image ls`: lista as imagens que temos na maquina

`docker image rm <<nome>>`: remove a imagem da maquina

`docker stop **id**`: para a imagem com o id selecionado

## Resumo

Aqui pudemos criar uma imagem docker com base em outra imagem o que nos permite automatizar e padronizar soluções para nosso trabalho.

outras imagens pode ser obtidas pelo [Docker Hub](#)