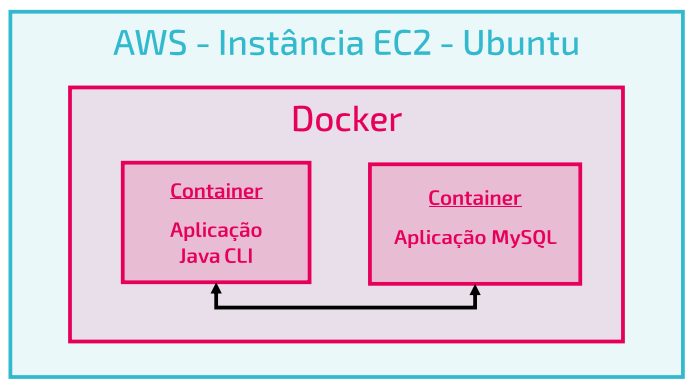
Linkando containers docker

O Docker tem como objetivo criar, testar e implementar aplicações em um ambiente separado da máquina original, chamado de container. Dessa forma, o desenvolvedor consegue empacotar o software de maneira padronizada, pois o Docker é uma plataforma de containers, ou seja, dentro de um Docker pode se haver um ou mais containers com configurações variadas e independentes um do outro.

[Docker | O que é Docker, o que são Containers e como isolar ambientes? (opservices.com.br)](https://www.opservices.com.br/o-que-e-docker/)

Demonstração da estrutura que iremos realizar:

Dentro de uma instância EC2 teremos o Docker instalado e criaremos dois containers independentes um do outro (um com a aplicação java e outro com o banco MySQL) e faremos ambos se comunicarem para trocar informações.

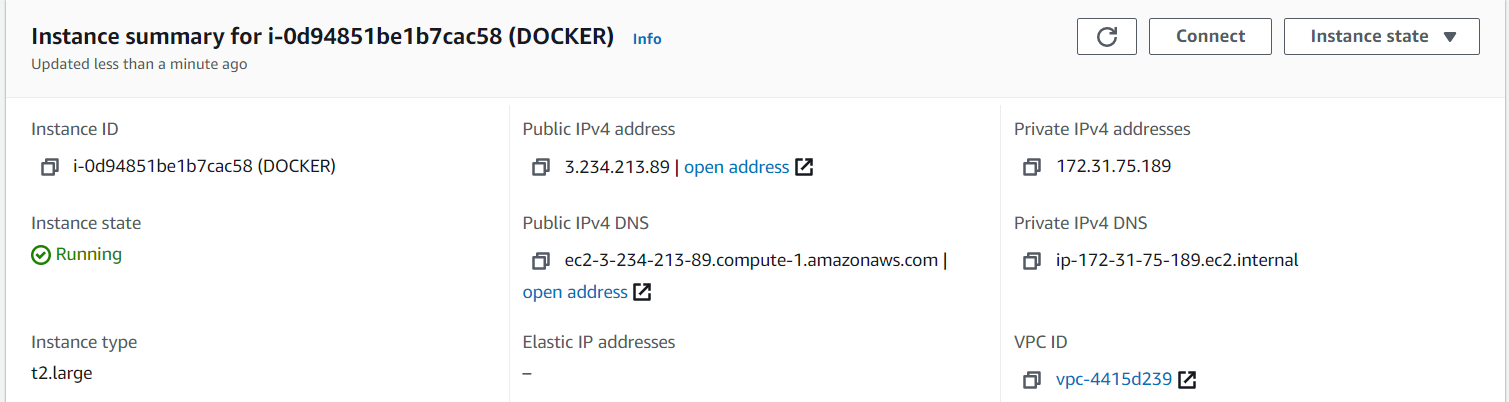


Passo 1:

Criar uma instância EC2.

Material de apoio:

Já temos a instância EC2 criada:



Utilizamos uma instância do tipo t2.large, que mesmo sendo um pouco mais cara, traz uma eficiência maior na utilização do docker com dois ou mais containers.

Dica: quando forem mudar os tipos de instancias sempre consultem os valores e fiquem ligados nisto para não gastarem todos seus créditos de forma indevida

Link para consultar valores: [Amazon EC2 tipos de instâncias - AWS](https://aws.amazon.com/pt/ec2/instance-types/)

Passo 2:

Instalar dentro da instância EC2 o Docker e criar um container com MySQL

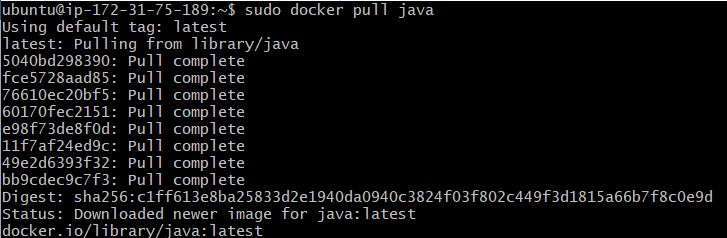
Material de apoio:

Passo 3:

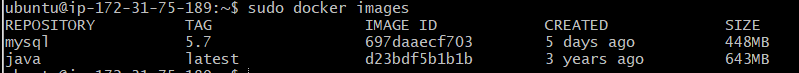
Criar um segundo docker para hospedar a nossa aplicação em java

Primeiramente iremos instalar a imagem java para utilizar no container

sudo docker pull java



sudo docker images



Vamos criar o container que conterá o java já fazendo um link com o container que há o MySQL, para isso precisamos pegar o name ou id do container MySQL

sudo docker ps



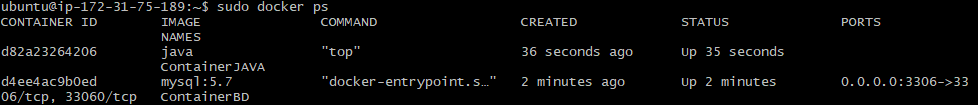
sudo docker run –name “NomeContainer” –link “NomeContainerLink”:”NomeImagemLink” –t –d “imagemContainer” top

sudo docker run --name ContainerJAVA --link ContainerBD:mysql -t -d java top



Verificar se foi criado o container

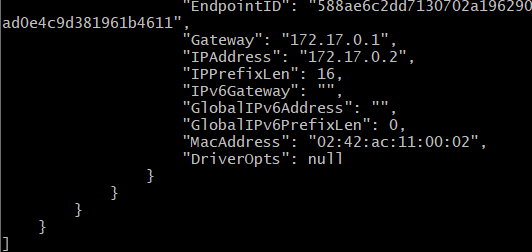
sudo docker ps



Passo 4:

Agora para verificar se a conexão foi realizada precisamos pegar o ip do container que foi linkado (neste caso o ContainerBD)

sudo docker inspect ContainerBD



Procurar pelo IPAddress. OBS: GUARDAR ESTE IP, POIS UTILIZAREMOS POSTERIORMENTE.

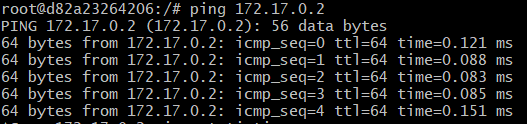
Entrar no Container que contem o JAVA com o comando:

sudo docker exec -it ContainerJAVA bash



Após entrar no container JAVA verificaremos se o link entre eles esta funcionando, utilizando o comando ping + IPAddress:

ping 172.17.0.2



Dar ctrl + c para parar o comando

Com isto nossos container estão linkados um ao outro, prontos para compartilhar informações entre si.

* Criar tabela MYSQL
* Criar classe de conexão java
* Criar classe MAIN para inserir dados
* Baixar o projeto no container JAVA
* Executar no container
* Verificar se os dados estão sendo enviados