



# **Curso completo de Python!**





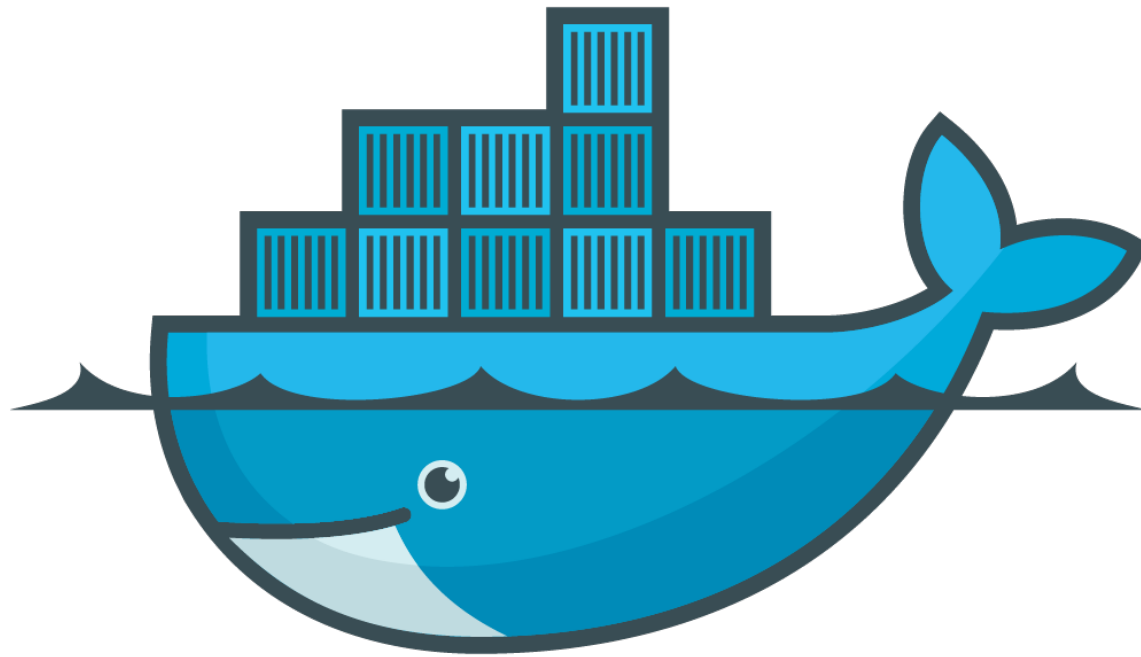
**Instrutor: Vitor Mazuco**

**<http://facebook.com/vitormazuco>**

**Email: [vitor.mazuco@gmail.com](mailto:vitor.mazuco@gmail.com)**

**WebSite:**

**<http://vmzsolutions.com.br>**



docker



## Gerenciando o nosso Container

Para remover o container criado antes, execute:

```
# docker rm hello
```



## Gerenciando o nosso Container

Para re-criar execute:

```
# docker run -dit --name hello debian /bin/bash
```



## Gerenciando o nosso Container

Para inspecionar execute:

```
# docker inspect hello
```



## Gerenciando o nosso Container

O comando `inspect` é usado para sabermos sobre as informações de nosso container. O formato retornado é do tipo JSON, e podemos ver o seu ID, data de criação, comando executado, status, etc.



## Gerenciando o nosso Container

Para parar execute:

```
# docker stop hello
```





## Gerenciando o nosso Container

Podemos remover também vários containers ao mesmo tempo:

```
# docker rm container1 container2 container3 ...
```



## Gerenciando o nosso Container

O comando para interagir com o intrepertador é o:

```
# docker attach hello
```

E para desconectar do container digite:

```
# CTRL^P e CTRL^Q
```



## Gerenciando o nosso Container

Caso você digite o comando **exit**, o seu docker irá parar!

Quando usamos o exit em nosso docker, estamos parando o interpretador de nosso container e por consequência o nosso sistema por completo. Caso isso ocorra, podemos voltar o Docker com o comando start!

<http://mcseolution.com.br/blog/2016/07/27/15-comandos-para-trabalhar-com-docker/>