



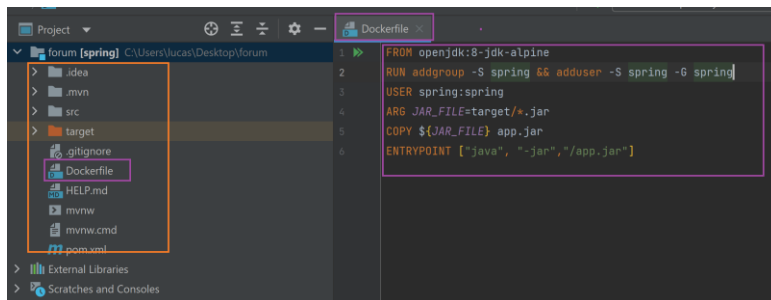
[REST-API]



Deploy com contêiner

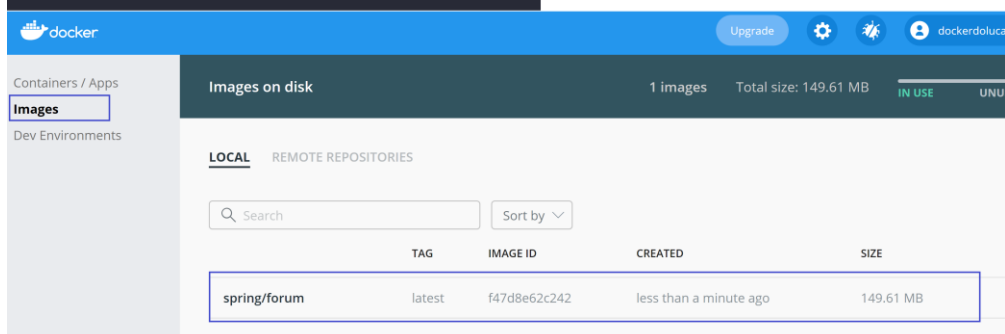
Usando Docker

É comum usar **containers** para isolar o ambiente de uma aplicação. A maneira mais simples de usar Contêiner com o **Docker** é criando um **Dockerfile** (Contém uma série de configurações montam uma imagem) no mesmo nível da **src**:



Depois disso dá pra montar a imagem usando o comando **“docker -t xxxx .”** (-t permite dar um nome pra imagem, e o “.” é pra referenciar o diretório que tem o Dockerfile):

```
lucas@DESKTOP-A5T9N3N MINGW64 ~/Desktop/forum
$ docker build -t spring/forum .
[internal] load build definition from Dockerfile
1: sha256:143808c5918c9b27aed23ceb53a2a1b914522feee27df41aadd91884be7efbe5
1: transferring dockerfile: 232B 0.0s done
1: DONE 0.1s
2: [internal] load .dockerignore
2: sha256:8c3fb3ba6d3e122b6220aff4844a147c67684ef750af25ab0a8513037d62920
2: transferring context: 2B done
2: DONE 0.1s
```



Depois disso **basta rodar a imagem**, dependendo do Perfil que quiser usar é necessário avisar o docker, podemos fazer isso com **“-e xxx”** no comando.

Por exemplo se eu quisesse rodar com o Perfil dev eu simplesmente usaria **“docker run xxx”**, se eu quiser rodar com o Perfil de produção eu preciso passar as variáveis de ambiente que eu configurei no perfil (**“docker run -p 8080:8090 -e perfil -e xxxx nome da img”**):

```
lucas@DESKTOP-A5T9N3N MINGW64 ~/Desktop/forum
$ docker run -e spring.profiles.active='prod' -e FORUM_DATABASE_URL='jdbc:h2:mem:forum' -e FORUM_DATABASE_USERNAME='sa' -e FORUM_DATABASE_PASSWORD='' -e FORUM_JWT_SECRET='123456' spring/forum

Spring
=====
:: Spring Boot ::
(v2.6.3)

2022-02-27 18:20:03.032 INFO 1 --- [          main] br.com.lucastrevizan.forum.Application : Starting Appli
cation v0.0.1-SNAPSHOT using Java 1.8.0_212 on 93b5372cfe2d with PID 1 (/app.jar started by spring in /)
2022-02-27 18:20:03.036 INFO 1 --- [          main] br.com.lucastrevizan.forum.Application : The following
profiles are active: prod
```