

# Criando serviços em Java com AWS ECS e Fargate

---

## Instruções de instalação das ferramentas

---

Para que o aluno possa acompanhar as sessões desse curso, em sua totalidade, é necessário preparar seu ambiente de desenvolvimento, instalando algumas ferramentas e principalmente a IDE a ser utilizada para criação das aplicações de exemplo.

É importante observar que os passos descritos a seguir estão em linhas gerais do que deve ser feito, pois cada sistema operacional possui suas peculiaridades. Por isso, é necessário seguir os passos definidos nos links que serão fornecidos, de acordo com o sistema operacional que será utilizado.

### a) Java Development Kit

Antes de mais nada é necessário instalar o **Java Development Kit** (JDK), um conjunto de aplicações e bibliotecas necessário para se desenvolver aplicações em Java.

Esse curso está baseado no JDK 15. Ele pode ser encontrado através do seguinte link:

<https://www.oracle.com/java/technologies/javase-jdk15-downloads.html>

Para baixá-lo, é necessário aceitar os termos da licença de uso, escolher o sistema operacional e sua versão. Note que alguns sistemas operacionais podem ter passos adicionais para a configuração do JDK 15.

### b) IntelliJ IDEA Community Edition

Para desenvolver as aplicações de exemplo que serão criadas nesse curso, é necessário instalar um ambiente integrado de desenvolvimento, ou como é conhecido pela sua sigla em Inglês - IDE.

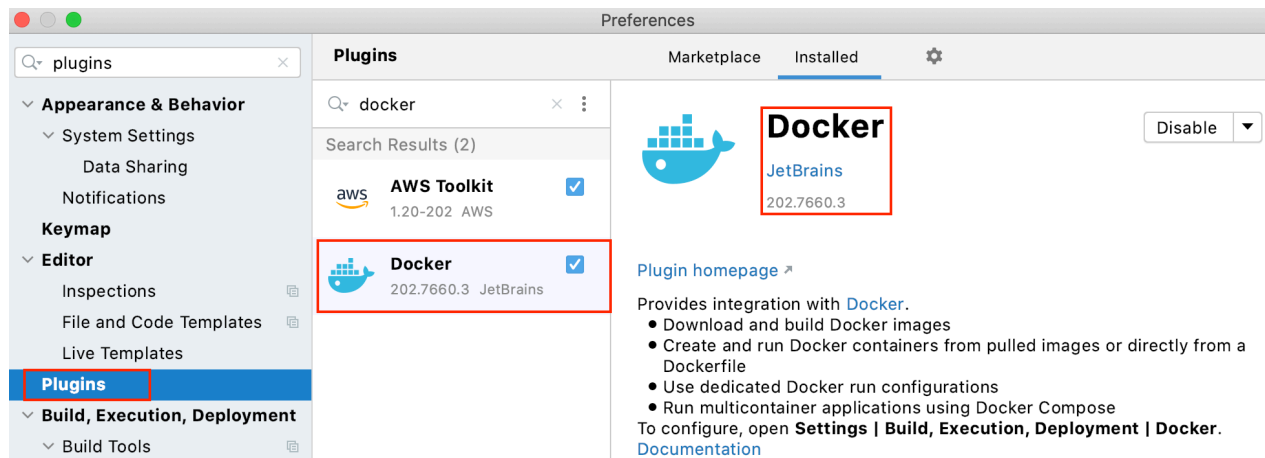
A IDE que será utilizada é o **IntelliJ IDEA Community Edition**, do fabricante JetBrains. Essa é uma das mais modernas IDE para desenvolvimento em Java e outras linguagens de programação.

Ele pode ser encontrado no seguinte endereço: <https://www.jetbrains.com/idea/download>

Baixe e instale a versão compatível com seu sistema operacional, seguindo as instruções definidas no site da JetBrains.

Caso o aluno queira aprender mais como trabalhar com o IntelliJ IDEA, há um excelente conjunto de tutoriais em vídeo na seguinte página de documentação do site da JetBrains: <https://www.jetbrains.com/idea/documentation/>

Para a geração da imagem do Docker das aplicações, será necessário instalar um plugin dentro do IntelliJ IDEA. Para isso, abra o IntelliJ IDEA e vá até o menu **Preferences** ou **Settings**, dependendo do seu sistema operacional. Na tela que aparecer, localize a seção **Plugins** e em seguida o plugin de nome **Docker**, criado pela JetBrains, como pode ser visto na figura a seguir:



## c) Cliente MySQL

Como algumas aplicações de exemplo irão utilizar uma base de dados relacional MySQL, será interessante ter um cliente para acessá-lo, ver suas tabelas e seus dados.

Existem várias aplicações clientes para esse tipo de trabalho. Uma excelente opção é o MySQL Workbench em sua versão gratuita. Ele pode ser baixado nesse endereço:

<https://dev.mysql.com/downloads/workbench/>

Selecione a opção adequada ao seu sistema operacional e siga as instruções de instalação oferecidas na página do fabricante.

Na sessão em que uma base de dados for utilizada por uma aplicação, haverá instruções específicas de como configurar o **MySQL Workbench** para se conectar a tal base. Por isso, não se preocupe em fazer passos adicionais. No momento, só é necessário instalar o cliente do MySQL Workbench.

## d) Conta no Docker

Um dos processos que será realizado ao longo desse curso é a criação de uma imagem de um *container* Docker para a execução da aplicação em um AWS ECS. A princípio parece algo complexo, mas na verdade é bem simples se alguns passos forem seguidos corretamente.

Para a utilização dessa imagem, é interessante colocá-la em um repositório para que a AWS possa baixá-la e utilizá-la. No momento, apenas crie uma conta no **Docker Hub**, no link a seguir:

<https://hub.docker.com>

Essa conta é gratuita e servirá totalmente para os objetivos desse curso.

## e) Docker Desktop

Da mesma forma como as demais aplicações apresentadas nesse capítulo, você deve instalar o Docker de acordo com a versão do seu sistema operacional. Para baixá-lo, veja as instruções contidas no link à seguir:

<https://docs.docker.com/install/>

Se quiser ir diretamente aos links para Mac e Windows, aqui estão:

- Mac: <https://docs.docker.com/docker-for-mac/install/>
- Windows: <https://docs.docker.com/docker-for-windows/install/>
- Windows Home: <https://docs.docker.com/docker-for-windows/install-windows-home/>

É importante ler as instruções de preparação e instalação. Normalmente não há problemas em computadores mais novos, mas como o Docker trabalha com virtualização de hardware, assim como o VMWare e o VirtualBox, podem existir certas incompatibilidades ou configurações adicionais que deverão ser feitas.

## f) Repositório no Docker Hub

Depois que a conta foi criada no Docker Hub, será necessário criar um repositório para cada imagem que for sendo criada. Esse passo será detalhado mais adiante, quando a primeira imagem for criada.

Nesse momento, basta que o aluno tenha apenas a conta no Docker Hub.

## g) Postman

**Postman** é um aplicativo gratuito muito útil e versátil. Com ele será possível fazer requisições às aplicações desenvolvidas nesse curso, tanto quando estiverem em execução na própria máquina de desenvolvimento, bem como quando estiverem hospedadas na AWS.

Para baixá-lo, siga as instruções contidas nesse link: <https://www.getpostman.com>

Com o Postman é possível organizar e salvar requisições montadas nele em forma de coleções, organizadas por projeto ou por contexto.

Também é possível criar uma conta no Postman, para armazenar e compartilhar as coleções de requisições que forem sendo criadas nele. Isso facilita bastante o trabalho em uma equipe de desenvolvedores e testadores.

### Instruções adicionais

Ainda serão detalhados passos adicionais em sessões mais adiante, principalmente no que se refere ao AWS CDK.

siecola.com.br

