Antes de iniciar o desenvolvimento precisamos configurar o nosso ambiente de desenvolvimento. O ambiente devidamente configurado é premissa para o desenvolvedor.

O nosso ambiente será configurado em uma máquina com o sistema operacional U ubntu 18.0.4, porém você pode montar o seu ambiente no sistema operacional da sua preferência. Seja qual for o sistema operacional que você for utilizar, você precisará instalar as seguintes ferramentas:

* OpenJDK 10
* MySQL
* IDE STS
* Tomcat 9

Configuração

Utilizaremos o gerenciador de pacotes apt-get, sendo assim, antes de iniciar a instalação das nossas ferramentas atualizaremos a lista de repositório do nosso gerenciador de pacotes. Caso deseja conhecer melhor sobre esse gerenciar de pacotes, esse link <https://www.infowester.com/aptget1.php> possui um tutorial bem bacana sobre o apt-get.

Todo o processo de instalação pode demorar um pouco, dependendo da sua conexão no momento da instalação.

Acesse o seu terminal e digite o comando abaixo e aguarde a conclusão da atualização

$ sudo apt-get update

Quando você estiver instalando qualquer pacote com o apt-get você será informado do total de espaço que aquele pacote utilizará do seu disco, para continuar com o processo de instalação você deve pressionar “Y”. Estou explicando isso aqui para não ficar redundante em todas as etapas que utilizaremos o apt-get install.

**Java**

Agora com o nosso APT atualizado iniciaremos a instalação das nossas ferramentas. Primeiro instalaremos o kit de desenvolvimento Java, do projeto OpenJDK. Um JDK (Java Development Kit) nos fornecerá toda a tecnologia Java para iniciar os nossos projetos. A Oracle é a detentora da tecnologia Java e possui sua própria JDK, porém o nosso ambiente trabalhará com o kit da OpenJDK, para mais informações consulte o site do projeto <https://openjdk.java.net>.

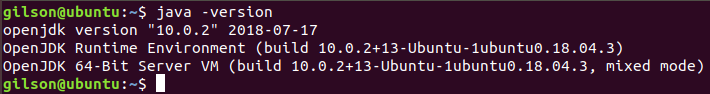
Novamente no terminal, vamos instalar a ultima versão do OpenJDk disponível no nosso gerenciador de pacotes, digite e aguarde o final da instalação

$ sudo apt-get install default-jdk

Para verificar a versão do OpenJDK instalado, no terminal digite:

$ java -version

Você obterá o seguinte resultado (no momento que configurei o meu ambiente, a última versão do OpenJDK é a 10.0.2):



O kit de desenvolvimento Java já está instalado.

**MySQL**

Praticamente toda aplicação necessita de um repositório para armazenar, ou persistir, os dados. Utilizaremos um dos mais famosos sistemas de gerenciamento de banco de dados banco de dados (SGDB), o MySQL, que pertence a Oracle.

Primeiro instalaremos o MySQL, novamente no seu terminal digite:

$ sudo apt-get install mysql-server

Depois de concluída a instalação será necessário configurar o MySQL. As configurações a seguir são bem explicadas em cada processo, então leia com atenção e tome a decisão que melhor se encaixar para as suas necessidades. Uma observação, na primeira etapa aonde você será questionado se aceita que o MySQL aceite apenas senhas fortes, se você manter essa configuração, você precisará atender a política de senhas do MySQL quando for criar um usuário no MySQL.

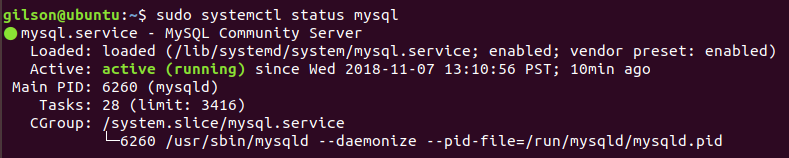
Vamos iniciar a configuração, novamente no terminal digite:

$ sudo mysql\_secure\_installation

Se ocorreu tudo certinho, o MySQL server já está instalado e executando. Para verificar se já está executando, digite no terminal:

$ sudo systemctl status mysql

Se você receber o seguinte retorno, o serviço já está ativo:



Você pode gerenciar o serviço, parando ou subindo-o através dos comandos:

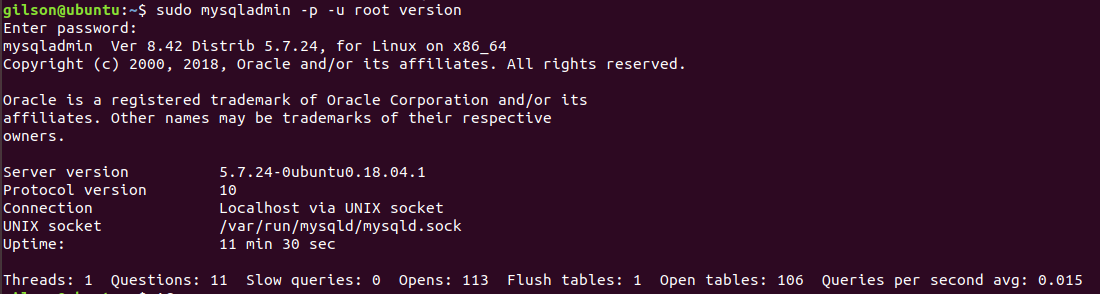
$ sudo systemctl start mysql

$ sudo systemctl stop mysql

Agora verificaremos a versão do nosso MySQL Server, digite o comando, logo na sequência será solicitada a senha que você configurou:

$ sudo mysqladmin -p -u root version

Nesse comando, estamos utilizando o utilitário mysqladmin, o “-p” indica que forneceremos a senha e o “-u” indica que forneceremos o usuário, no nosso caso já esta digitado logo a frente, o usuário é “root”. Você deve receber o seguinte retorno (no meu caso o gerenciador de pacotes instalou a versão 5.7 do MysQL Server e a 8.42 do utilitário de administração do nosso gerenciador de banco de dados):



IDE – STS

Você pode trabalhar com qualquer editor de texto para desenvolver suas soluções em Java, mas uma IDE (Integrated Development Envirement ou Ambiente de Desenvolvimento Integrado) facilita muito a vida de um desenvolvedor. O IDE disponibiliza enorme gama de ferramentas que auxiliarão durante o ciclo de vida de desenvolvimento e até mesmo execução do seu software. Para conhecer um pouco mais acesse esse site(<https://pt.wikipedia.org/wiki/Ambiente_de_desenvolvimento_integrado>).

O nosso ambiente utilizará o IDE STS, do projeto Spring. Esse IDE é baseada no IDE Eclipse, então o STS e o Eclipse são IDEs muito parecidos, claro que cada um tem algumas particularidades.

Para instalar o STS não utilizaremos o APT-GET, porém continuaremos utilizando o terminal. Acesse o diretório /opt, caso a sua máquina não possua esse diretório cria-o. Para isso digite o comando para criar e acessar o diretório /opt:

$ sudo mkdir /opt && cd /opt

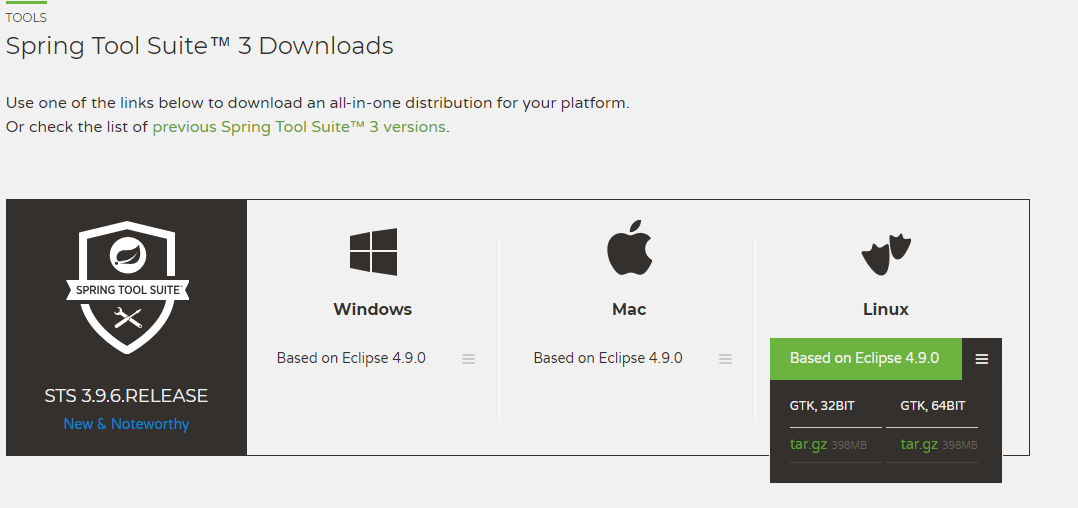


Caso você possua esse diretório, apenas acesse-o com o seguinte comando:

$ sudo cd /opt



Para obter a última versão do STS diretamente do fabricante, acesse o site <https://spring.io/tools3/sts/all/> e clique em Linux, abrirá um menu, clique com o botão diretório em cima da versão que atenderá a estrutura do seu computador e clique em copiar link

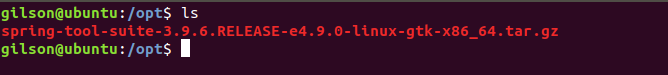


Volte para o terminal para iniciar o download do pacote do STS, utilizaremos o comando wget:

$ sudo wget <https://download.springsource.com/release/STS/3.9.6.RELEASE/dist/e4.9/spring-tool-suite-3.9.6.RELEASE-e4.9.0-linux-gtk-x86_64.tar.gz>

Após o download, verifique se o pacote compactado esta no seu diretório, liste com o comando ls:

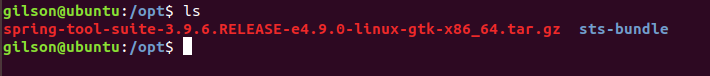
$ ls



Agora iremos extrair o arquivo tar.gz utilizando o comando tar

$ sudo tar -xvf spring-tool-suite-3.9.6.RELEASE-e4.9.0-linux-gtk-x86\_64.tar.gz sts-bundle

Aguarda até que pacote seja descompactado. Se você utilizar o comando ls novamente, verá que agora existe um diretório:



O STS, assim como o Eclipse, não possui um instalador, você já pode executar o arquivo /opt/sts-bundle/sts-3.9.6.RELEASE/STS para acessar o IDE, porém será muito trabalhoso acessar esse arquivo toda vez que você for abrir o IDE. Para facilitar a nossa vida deixaremos o seu acesso a partir do nosso Desktop. Para isso, primeiro, devemos alterar a permissão no nosso diretório de instalação, execute o comando:

$ sudo chmod 77 -Rf /opt/sts-bundle/

Agora criaremos o nosso ícone no Desktop. Para isso anote o endereço absoluto do executável do STS e do ícone, ambos dentro do diretório /opt/sts-bundle/sts-3.9.6.RELEASE/. Agora criaremos o nosso ícone, para isso utilizaremos o editor de texto vi (pode utilizar qualquer editor de texto da sua preferência) para criar o arquivo STS.desktop utilizaremos o editor de texto nano, mas você pode utilizar qualquer um da sua preferência:

sudo nano /usr/share/applications/STS.desktop

Dentro do editor de texto, insira o conteúdo abaixo, preste atenção no caminho do seu ícone e do executável:

[Desktop Entry]

Name=SpringSource Tool Suite

Comment=SpringSource Tool Suite

Exec=/opt/sts-bundle/sts-3.9.6.RELEASE/STS

Icon=/opt/sts-bundle/sts-3.9.6.RELEASE/icon.xpm

StartupNotify=true

Terminal=false

Type=Application

Categories=Development;IDE;Java;

Salve o arquivo. Agora você pode acessar o STS a partir do seu desktop, digitando apenas STS

**Tomcat**

Toda aplicação web necessita de uma plataforma que execute a nossa aplicação deixando-a disponível na rede. Nesse caso não utilizaremos um servidor web, pois um servidor web é robusto e possui características que não necessitaremos em um primeiro momento, por isso utilizaremos um container, uma plataforma que possui o básico para disponibilizar um conteúdo web, nesse caso utilizaremos o Tomcat, ele é um container muito utilizado pela comunidade. Você também pode utilizar o Tomcat em ambiente de produção, funcionará normalmente, uma vez que o Spring já abstrai muitas tecnologias que um servidor web fornece.

Voltemos ao terminal. Por questão de segurança, criaremos um usuário apenas para o tomcat. Primeiro criaremos um grupo

$ sudo groupadd tomcat9

Agora criaremos o usuário tomcat9. No comando abaixo estamos configurando o shell desse usuário para false, assim ninguém conseguirá efetuar login com essa conta; depois adicionamos no grupo criado acima sendo o diretório home o diretório /opt/tomcat (esse diretório será o diretório de instalação do Tomcat, criaremos mais a frente); por fim informado o usuário, tomcat9:

$ sudo useradd -s /bin/false -g tomcat9 -d /opt/tomcat tomcat9

Acesse o diretório /opt/. Primeiro faremos o download do pacote de instalação Tomcat, assim como a instalação do STS também não utilizaremos o APT-GET. Para fazer o download acesse o site do fabricante <https://tomcat.apache.org>, na área de Download clique em Tomcat 9, na próxima página, localize a seção 9.0.13, no tópico Core localize o link “tar.gz (pgp, sha512)”, clique com o botão direito em cima deste link e selecione a opção Copiar Link. Agora no terminal digite:

$ sudo wget https://www.apache.org/dist/tomcat/tomcat-9/v9.0.16/bin/apache-tomcat-9.0.16.tar.gz

Aguarde o termino do download. Se você digitar o comando ls verificará que possui um arquivo “apache-tomcat-9.0.13.tar.gz.”. Agora criaremos o nosso diretório de instalação:

$ sudo mkdir tomcat

Agora descompactaremos o conteúdo do arquivo para dentro do nosso diretório de instalação:

sudo tar xzvf apache-tomcat-9.0.13.tar.gz -C /opt/tomcat --strip-components=1

Vamos acessar o diretório de instalação:

$ cd tomcat

Dê ao grupo de usuário que criamos, permissão de dono para o diretório /opt/tomcat:

$ sudo chgrp -R tomcat9 /opt/tomcat

Agora, conceda permissão de leitura para o diretório conf e todo seu conteúdo e permissão de execução.

$ sudo chmod -R g+r conf

$ sudo chmod -R g+x conf

Faça com que o usuário que criamos seja dono dos diretórios webapps, work, temp e logs

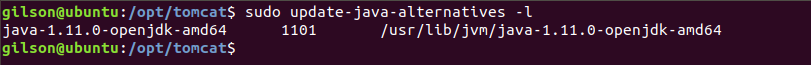
$ sudo chown -R tomcat9 webapps/ work/ temp/ logs/

Agora vamos criar um arquivo para você controlar o Tomcat como serviço, facilitando quando for necessário reiniciar o serviço do Tomcat.

Primeiro você precisa localizar o caminho de instalação do nosso JDK, o Kit de desenvolvimento do Java. Execute o comando para listar o diretório do JDK:

$ sudo update-java-alternatives -l

Você terá o seguinte resultado, porém o seu caminho pode estar diferente ou até mesmo o nome do pacote JDK.



Caso localize o pacote, vamos criar o arquivo. Para criar esse arquivo você pode utilizar o seu editor de texto favorito, aqui utilizo o nano:

sudo nano /etc/systemd/system/tomcat.service

Copie e cole o conteúdo abaixo, porém verifique se os caminhos e o usuário estão corretos:

[Unit]

Description=Apache Tomcat Web Application Container

After=network.target

[Service]

Type=forking

Environment=JAVA\_HOME=/usr/lib/jvm/java-1.11.0-openjdk-amd64

Environment=CATALINA\_PID=/opt/tomcat/temp/tomcat9.pid

Environment=CATALINA\_HOME=/opt/tomcat

Environment=CATALINA\_BASE=/opt/tomcat

Environment='CATALINA\_OPTS=-Xms512M -Xmx1024M -server -XX:+UseParallelGC'

Environment='JAVA\_OPTS=-Djava.awt.headless=true -Djava.security.egd=file:/dev/./urandom'

ExecStart=/opt/tomcat/bin/startup.sh

ExecStop=/opt/tomcat/bin/shutdown.sh

User=tomcat9

Group=tomcat9

UMask=0007

RestartSec=10

Restart=always

[Install]

WantedBy=multi-user.target

Agora iremos recarregar o módulo de serviços do Linux:

$ sudo systemctl daemon-reload

Agora iremos iniciar o serviço do tomcat e habilitar o start deste serviço no boot do Linux

$ sudo systemctl start tomcat

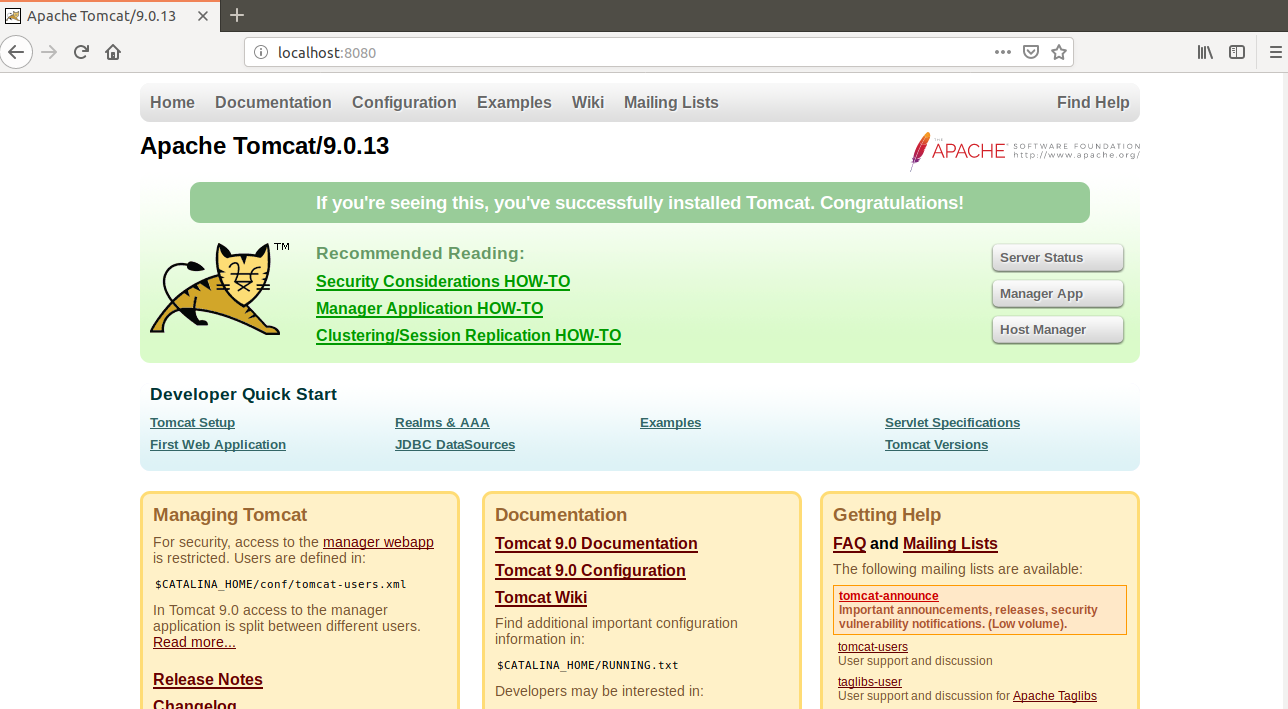
$ sudo systemctl enable tomcat

Você pode verificar o status do serviço com o comando abaixo:

$ systemctl status tomcat

A partir desse momento você já pode acessar o portal do seu container Tomcat, para isso, no browser digite

<http://localhost:8080>



Caso você não conseguiu acessar o container, talvez o seu firewall esteja bloqueando a porta 8080, para liberar a porta padrão do tomcat, 8080, digite o comando abaixo:

$ sudo ufw allow 8080

Mesmo acessando a home do seu Tomcat, você não conseguirá acessar o gerenciador, pois não possui um usuário configurado.

Para configurar o acesso vamos configurar uma conta no Tomcat. Acesse o arquivo de configuração de usuários do Tomcat, é um arquivo xml:

$ sudo nano /opt/tomcat/conf/tomcat-users.xml

Agora insira a linha abaixo dentro do nó <tomcat-users>, por motivos de segurança recomendo que altere o login e senha. Com relação a roles, não altere nesse momento, mas vale a pena consultar a documentação do Tomcat para configurar as permissões que realmente lhe atenderão:

<user username="admin" password="password" roles="manager-gui,admin-gui"/>

Caso você tente acessar o gerenciador do Tomcat em outra estação da sua rede, isso não será permitido, pois por default esse acesso é apenas para a estação que o Tomcat está instalado.   
Para alterar essa configuração, você deve editar os arquivos contexto.xml do manager e do host-manager. Dentro de cada arquivo você deve procurar o nó <Valve ....> e comentá-lo, adicionando o “<!--" no ínicio e “-->” no final da tag

<!-- <Valve className="org.apache.catalina.valves.RemoteAddrValve"

allow="127\.\d+\.\d+\.\d+|::1|0:0:0:0:0:0:0:1" />

-->

Caminho dos arquivos contexto.xml que deverão sofrer a alteração:

$ sudo nano /opt/tomcat/webapps/manager/META-INF/context.xml

$ sudo nano /opt/tomcat/webapps/host-manager/META-INF/context.xml

Agora você tem um ambiente para desenvolver e testar suas aplicações Java.

Fontes:

<https://www.digitalocean.com/community/tutorials/install-tomcat-9-ubuntu-1804>

<https://joseandresmacias.wordpress.com/2015/06/08/how-do-i-modify-my-hosts-file/>