

PREPARAÇÃO DO AMBIENTE DE DESENVOLVIMENTO HÍBRIDO

1) Instalação do **JDK 1.8**

Para instalar o JDK no Windows, primeiro baixe-o no site da Oracle. É um simples arquivo executável que contém o Wizard de instalação:

1. Acesse o endereço abaixo e clique na imagem a seguir:

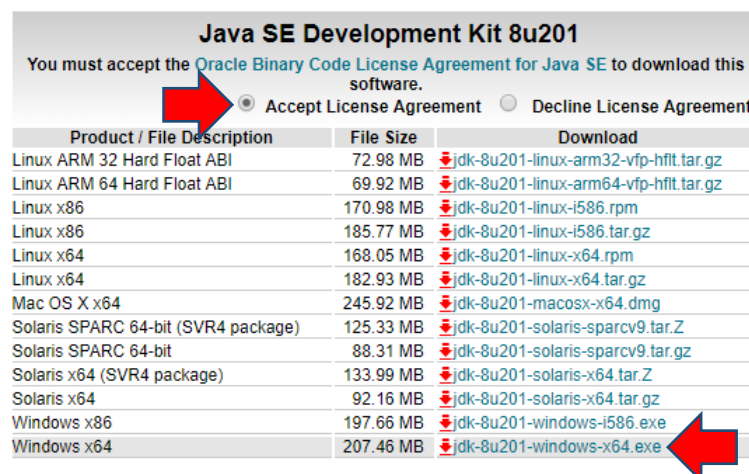
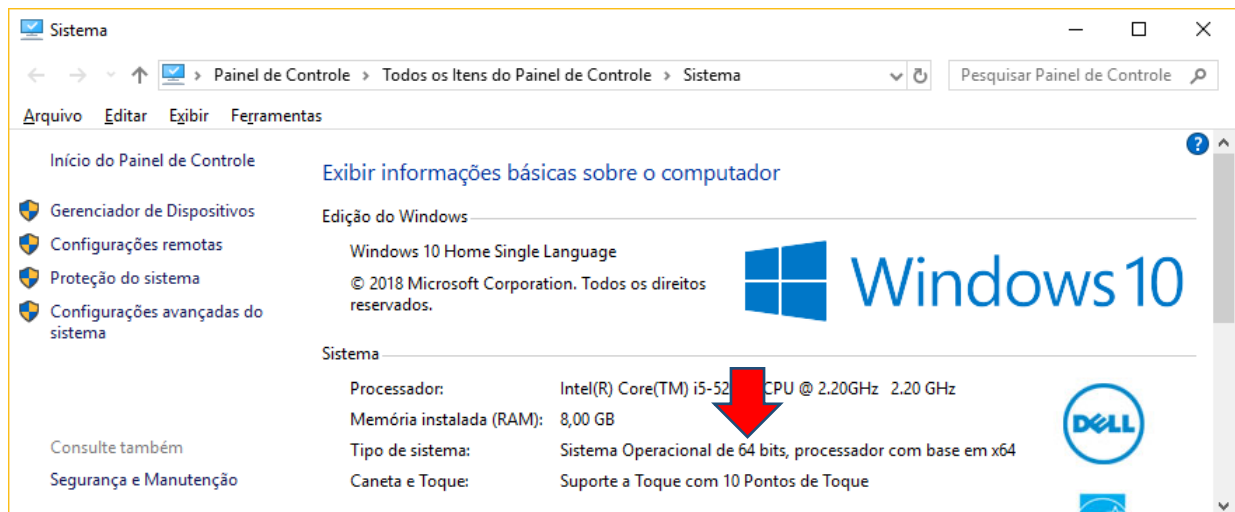
<https://www.oracle.com/technetwork/pt/java/javase/downloads/index.html>

Java SE Downloads

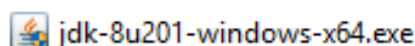


Java Platform (JDK) 8u111 / 8u112

2. Clique em **Accept License Agreement** e selecione o arquivo de instalação de acordo com a versão do seu Sistema Operacional instalado, como por exemplo, pressione as teclas **Windows + Pause/Break** para abrir a janela abaixo. Verifique o tipo do Sistema Operacional.



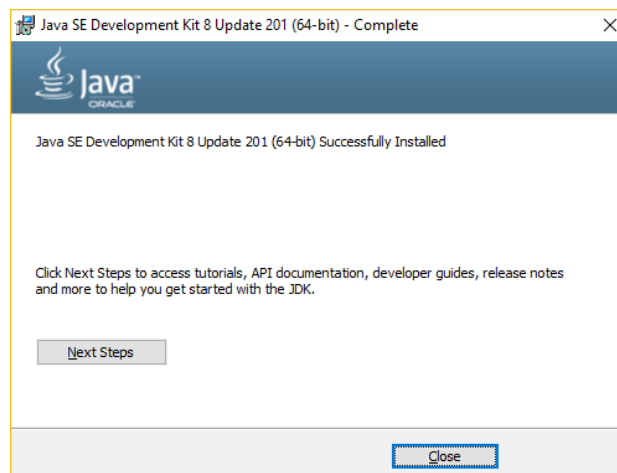
3. Após o download, dê um clique duplo no arquivo **jdk-<versão>-windows-<versão do S.O>.exe** e espere até ele entrar no wizard de instalação, como por exemplo:



4. Aceite os próximos dois passos clicando em Next. Após um tempo, o instalador pedirá para escolher em que diretório instalar o SDK. Pode ser onde ele já oferece como padrão. Anote qual foi o diretório escolhido, vamos utilizar esse caminho mais adiante. A cópia de arquivos iniciará:



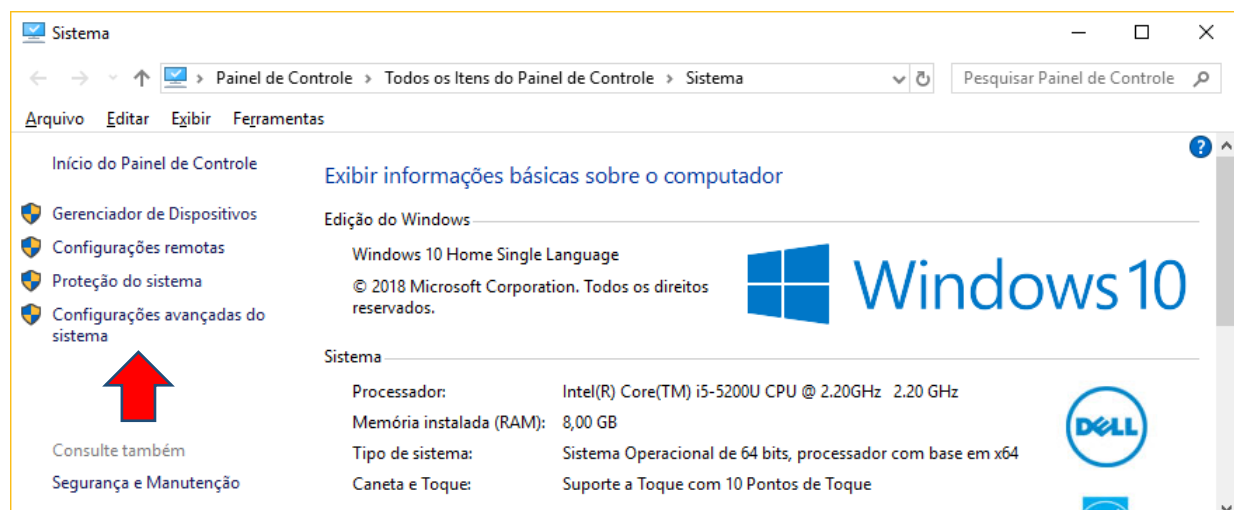
5. Após isso, a instalação estará finalizada e você será direcionado à uma página onde você pode, opcionalmente, criar uma conta na Oracle para registrar sua instalação.



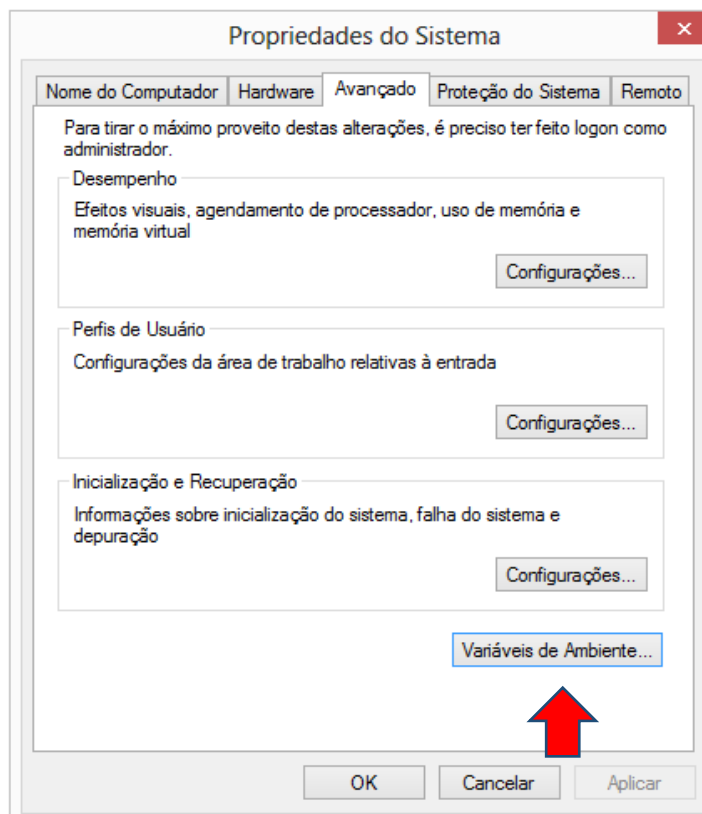
Configurando o ambiente:

Precisamos configurar algumas variáveis de ambiente após a instalação, para que o compilador seja acessível via linha de comando.

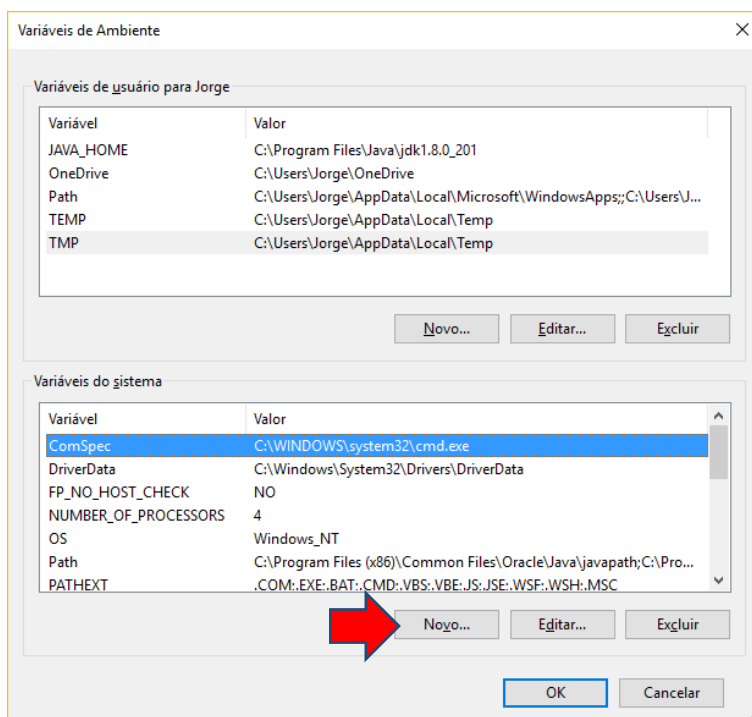
1. Pressione as teclas **Windows + Pause/Break** para abrir a janela abaixo e escolha a aba **“Configurações Avançadas de Sistema”**.



2. Clique no botão “**Variáveis de Ambiente**”.

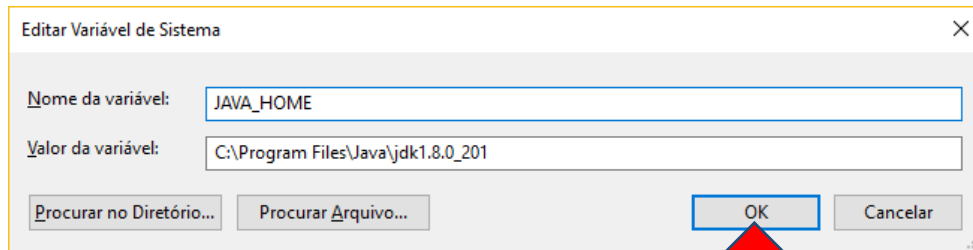


3. Nesta tela, você verá, na parte de cima, as **variáveis de ambiente do usuário** corrente e, embaixo, as **variáveis de ambiente do sistema** (servem para todos os usuários). Clique no botão **Novo...** da parte de baixo.



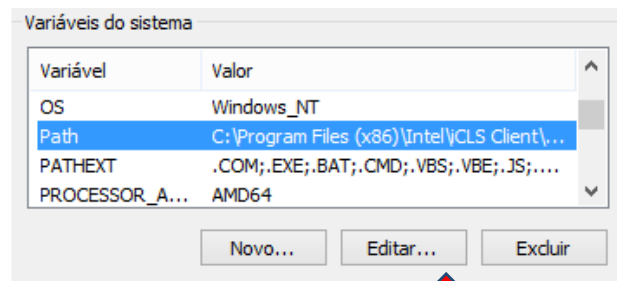
4. Em “**Nome da variável**” digite **JAVA_HOME** e, em “**Valor da variável**”, digite o caminho que você utilizou na instalação do Java.

Provavelmente será algo como: **C:\Program Files\Java\jdk1.8.0_201**

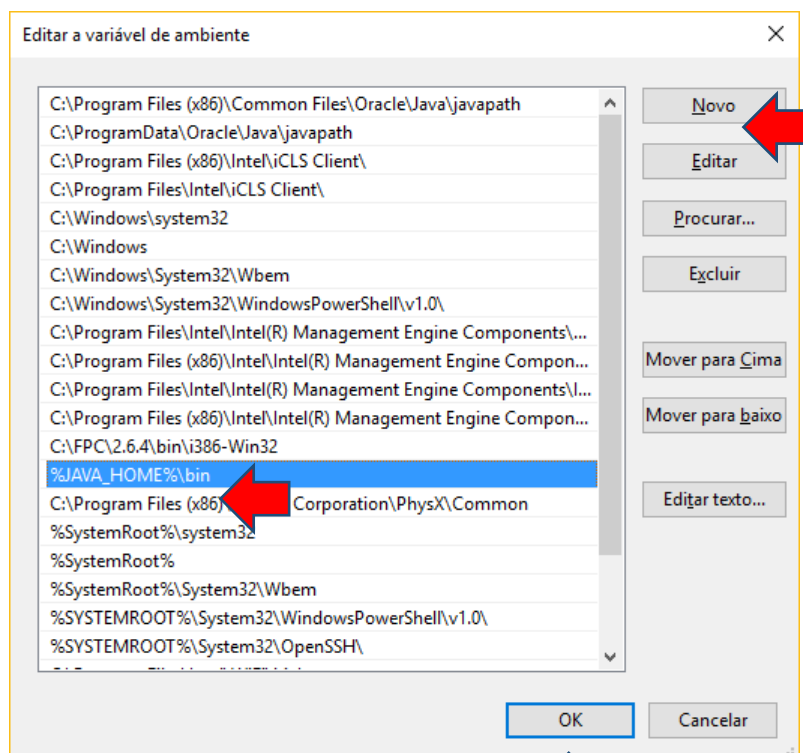


Clique em **Ok**.

5. Não vamos criar outra variável, mas sim alterar. Para isso, procure a variável **PATH**, ou **Path** (dá no mesmo), e clique no botão de baixo “**Editar**”.

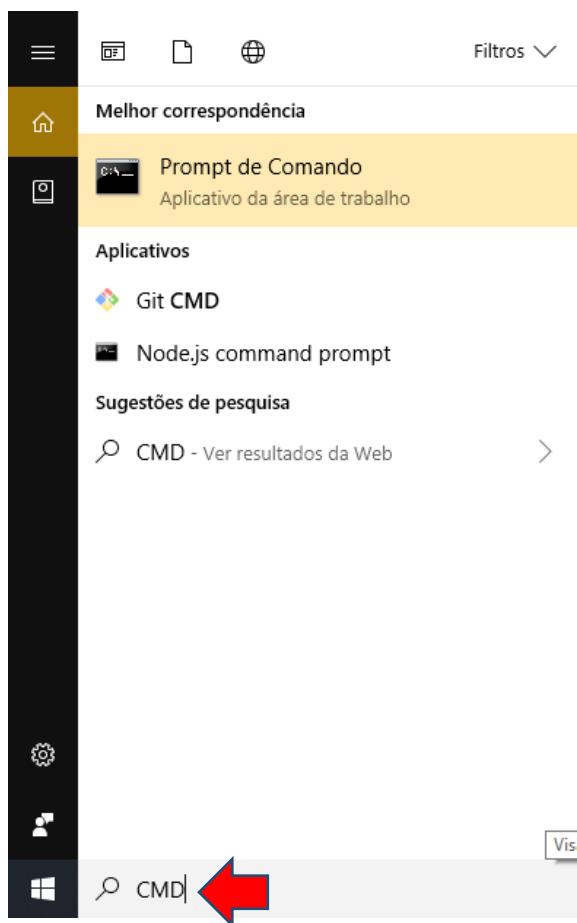


6. Adicione um novo valor **%JAVA_HOME%\bin**. Assim, você está adicionando mais um caminho à sua variável Path. Para isso, clique no botão “**Novo**” e digite o valor citado.



Clique em **Ok**.

7. Abra o Prompt de Comando do Windows conforme abaixo, digitando o comando **CMD**.



8. Por fim, no console, digite **java -version**. O comando deverá mostrar a versão do Java.

```
Microsoft Windows [versão 10.0.17134.523]
(c) 2018 Microsoft Corporation. Todos os direitos reservados.

C:\Users\Jorge>java -version
java version "1.8.0_201"
Java(TM) SE Runtime Environment (build 1.8.0_201-b09)
Java HotSpot(TM) 64-Bit Server VM (build 25.201-b09, mixed mode)

C:\Users\Jorge>
```

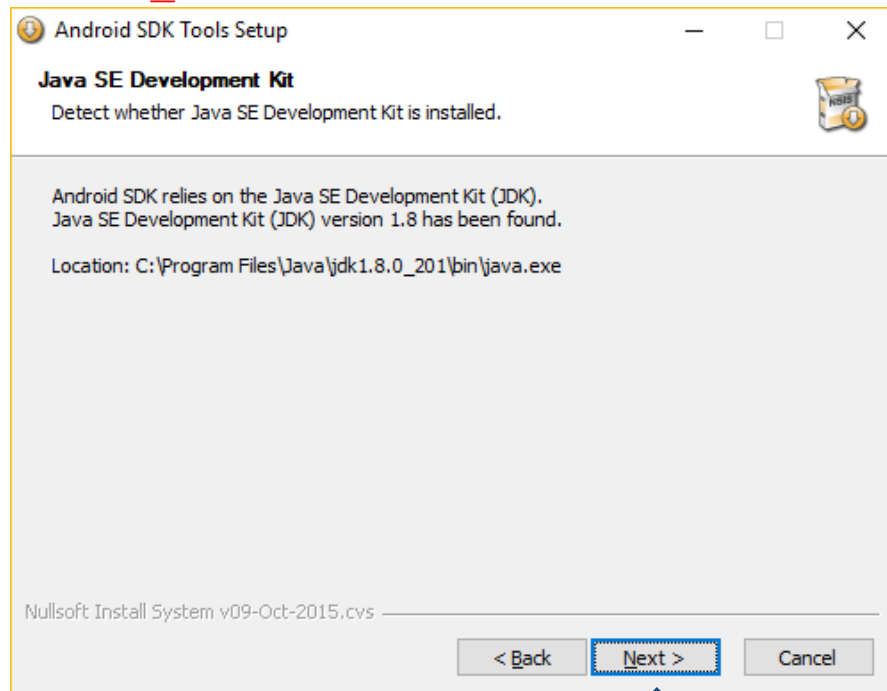
2) Instalação do **Google Android SDK**

Acesse o link abaixo e faça o download instalador do **Android SDK** para a plataforma Windows.

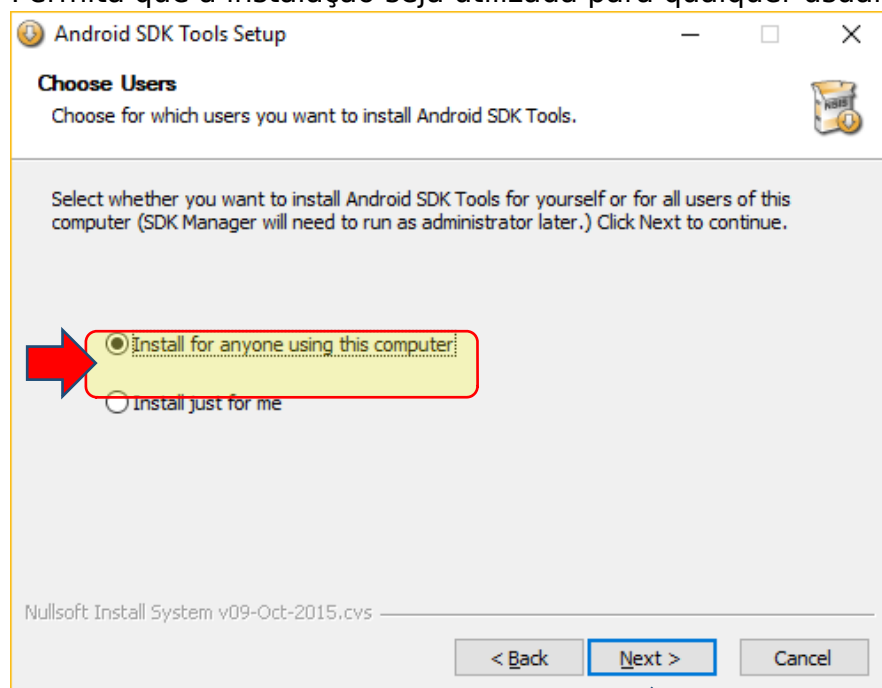
https://dl.google.com/android/installer_r24.4.1-windows.exe

Execute o instalador e siga os seguintes passos:

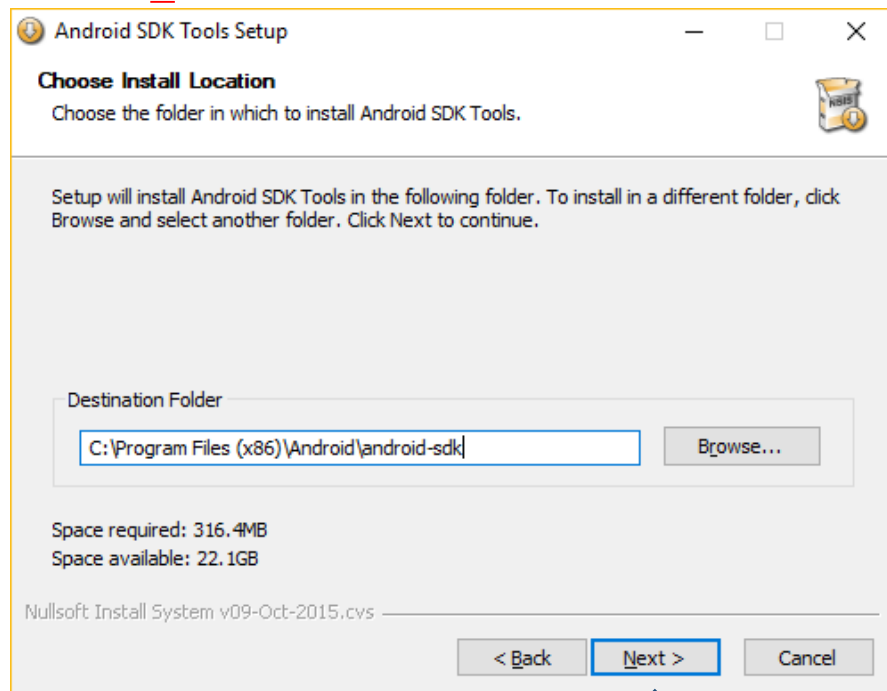
Será identificada a instalação do Java, feita anteriormente em seu computador. Clique no botão **Next**.



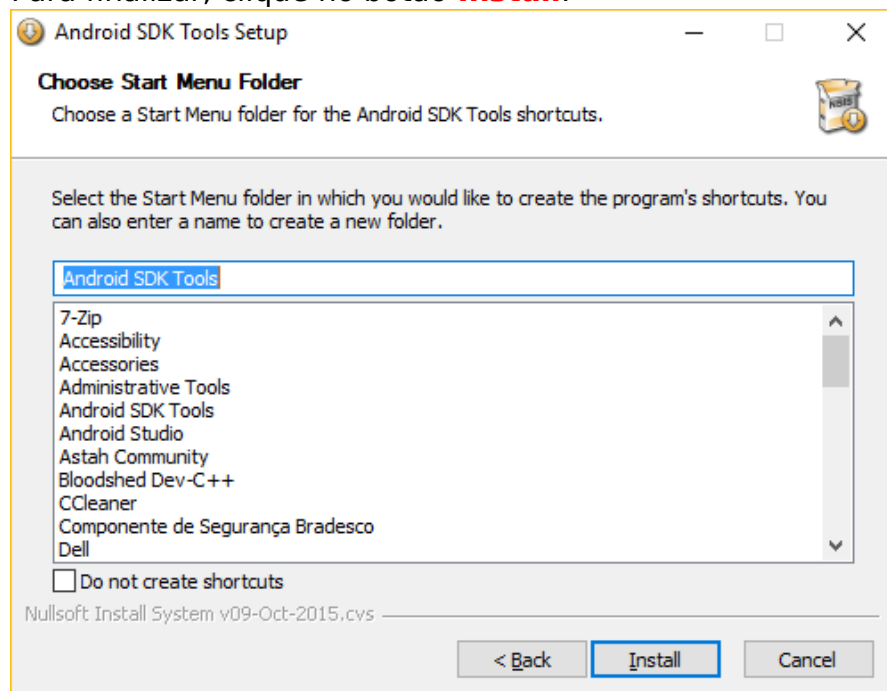
Permita que a instalação seja utilizada para qualquer usuário. Clique no botão **Next**.



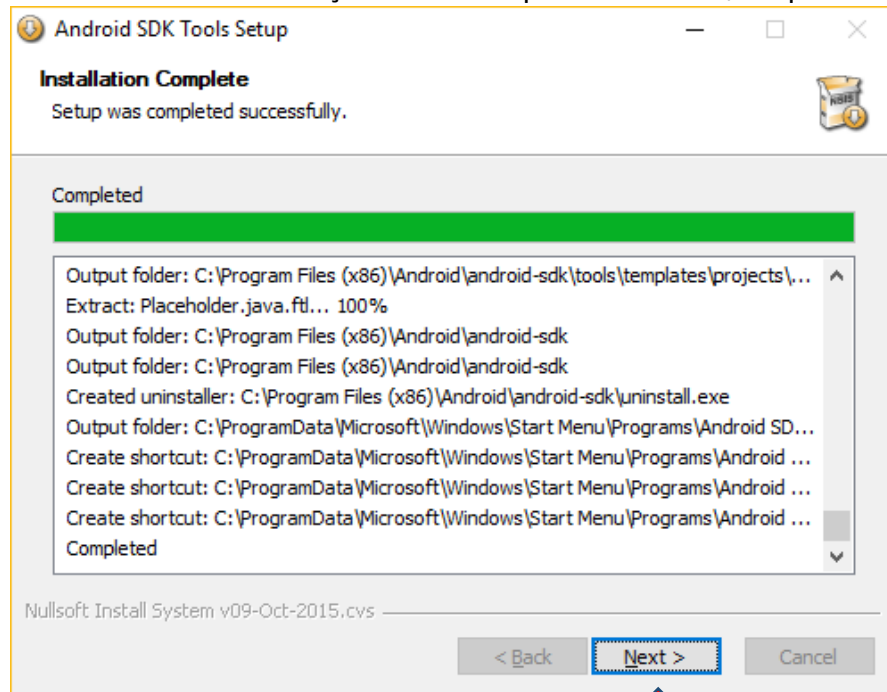
Não altere o caminho de instalação do Android SDK, esse é um caminho padrão. Clique no botão **Next**.



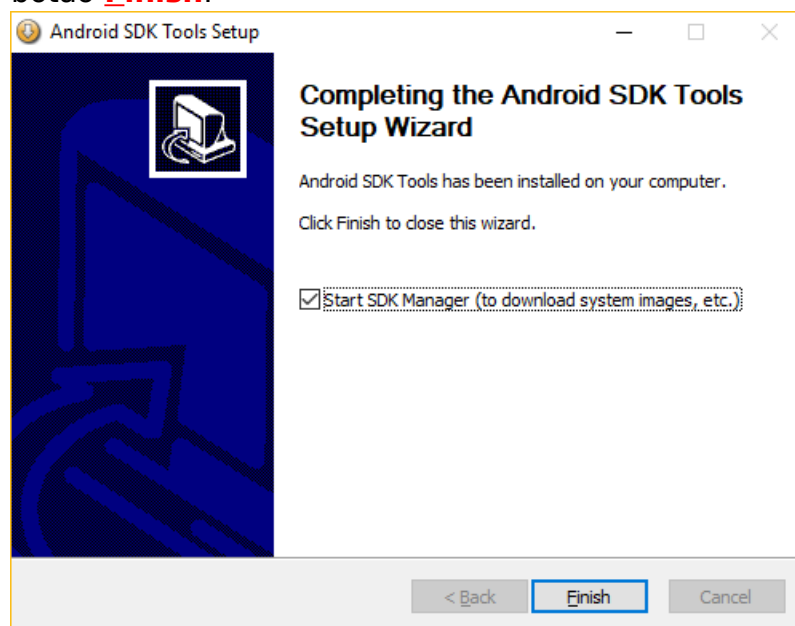
Para finalizar, clique no botão **Install**.



O Processo de instalação está completo. Ao final, clique no botão **Next**.



Deixe a opção **Start SDK Manager** marcada para iniciar o SDK Manager. Clique no botão **Finish**.

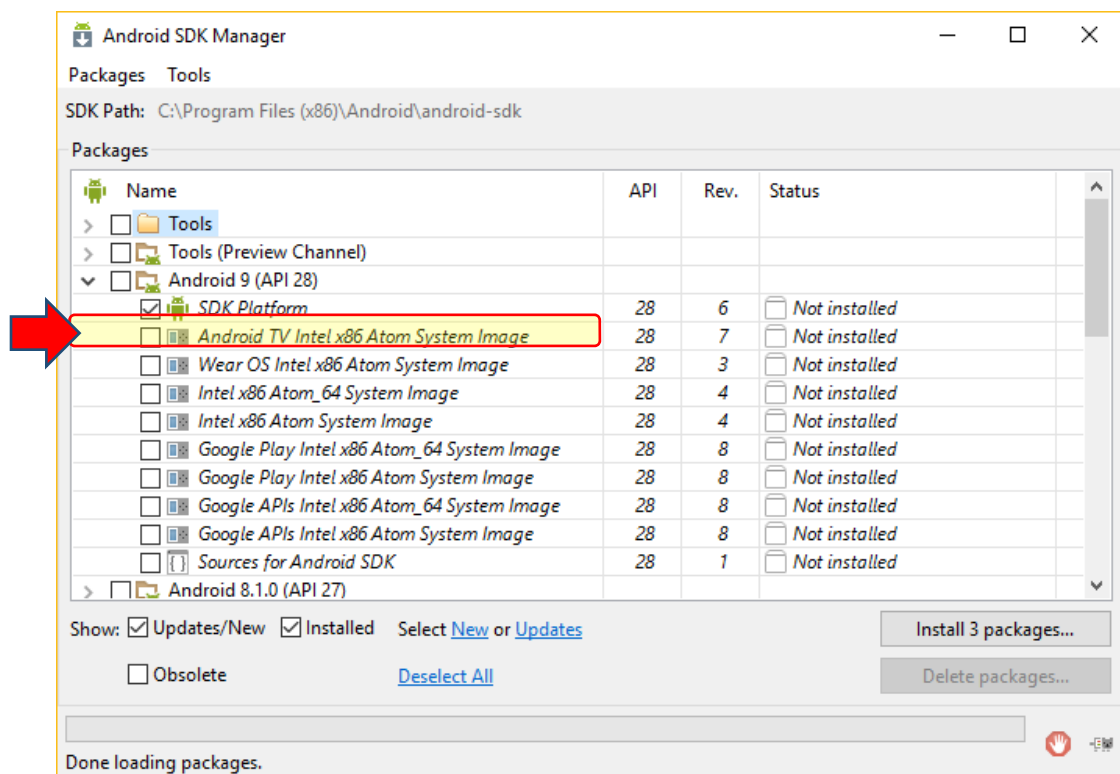
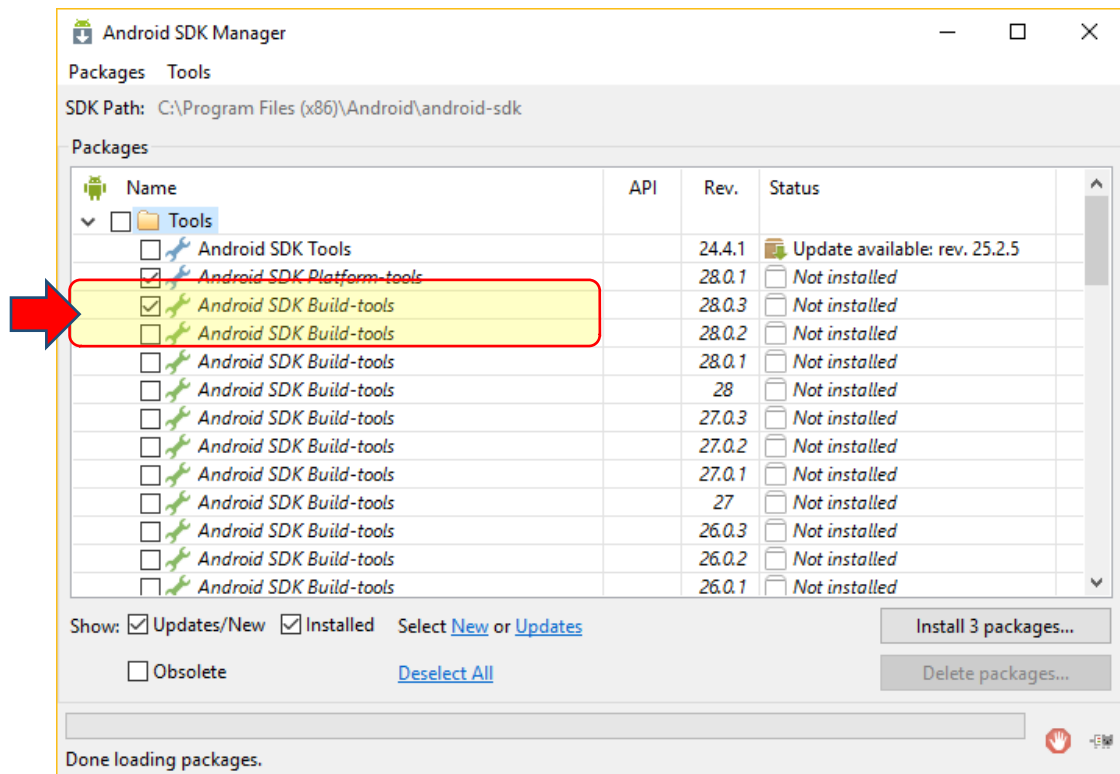


Após abrir o **SDK Manager**, precisamos então adicionar certos componentes a ele.

Vamos instalar:

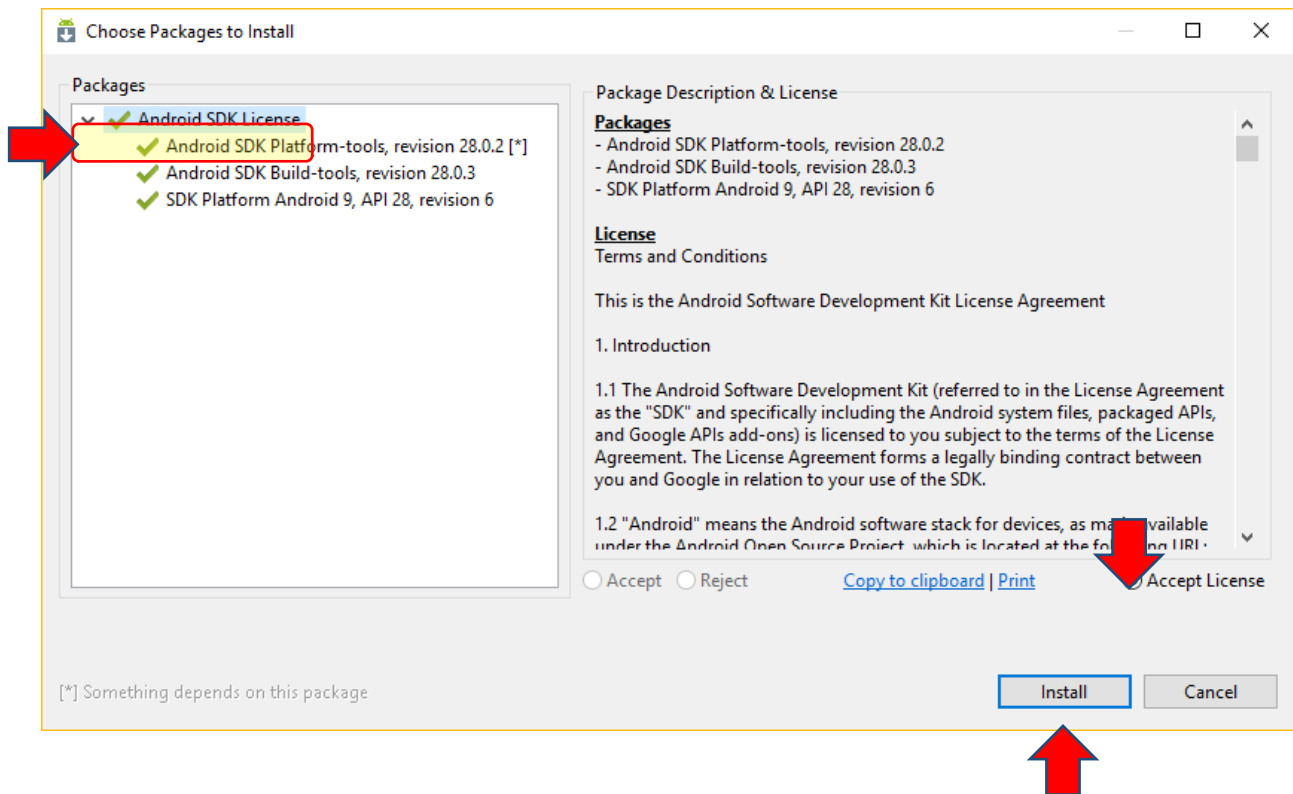
- SDK Platform
- Android SDK Platform-tools
- Android SDK Build-tools

Marque as últimas versões (**Android 9, API 28**) do **Build-tools**, **Platform-tools** e do **SDK Platform**. Clique em **Install 3 Packages**. Ele baixará bastante coisa e é bem demorado para instalar.

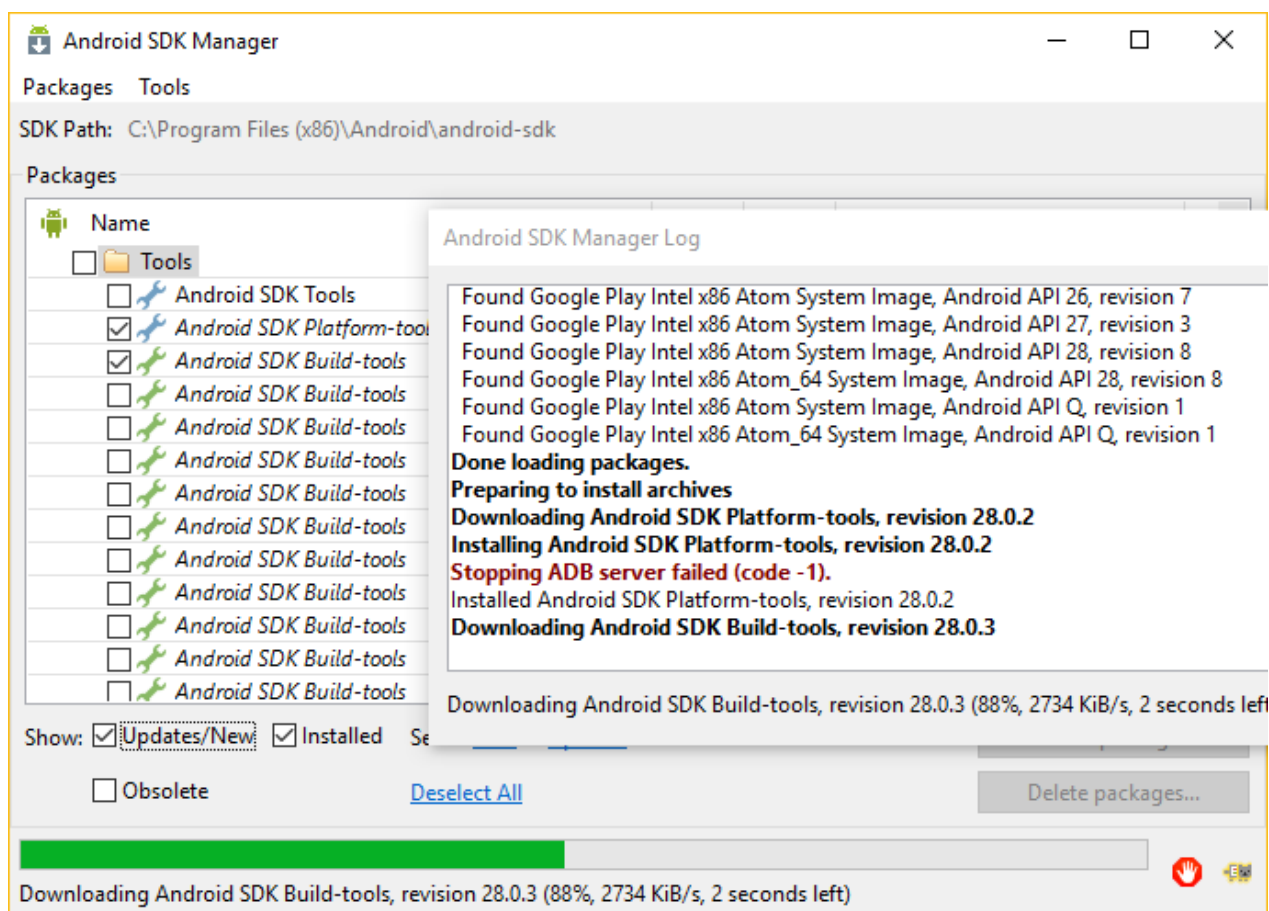


Por fim, você precisa aceitar as licenças para finalizar a instalação dos pacotes do SDK. Faça o seguinte:

- Selecione a opção **Android SDK License**, clique em **Accept License** e clique no botão **Install**.



O Processo de instalação iniciará...



Ao final, clique no botão **Done** e feche o Android SDK Manager.

****** ATENÇÃO: SOMENTE EXECUTE ESSE PASSO SE HOUVER FALHA NO BUILD DO APLICATIVO. ******

Pode ser que, dependendo da versão do seu dispositivo móvel, ao executar o comando **CORDOVA RUN ANDROID**, você não possua a versão do **Android SDK Platform** compatível, veja o seguinte erro abaixo:

```

Prompt de Comando
Checking the license for package Android SDK Platform 27 in C:\Program Files (x86)\Android\android-sdk\l
licenses
Warning: License for package Android SDK Platform 27 not accepted.

FAILURE: Build failed with an exception.

* What went wrong:
A problem occurred configuring project ':CordovaLib'.
> You have not accepted the license agreements of the following SDK components:
  [Android SDK Platform 27].
  Before building your project, you need to accept the license agreements and complete the installation
of the missing components using the Android Studio SDK Manager.
  Alternatively, to learn how to transfer the license agreements from one workstation to another, go to
http://d.android.com/r/studio-ui/export-licenses.html

* Try:
Run with --stacktrace option to get the stack trace. Run with --info or --debug option to get more log o
utput.

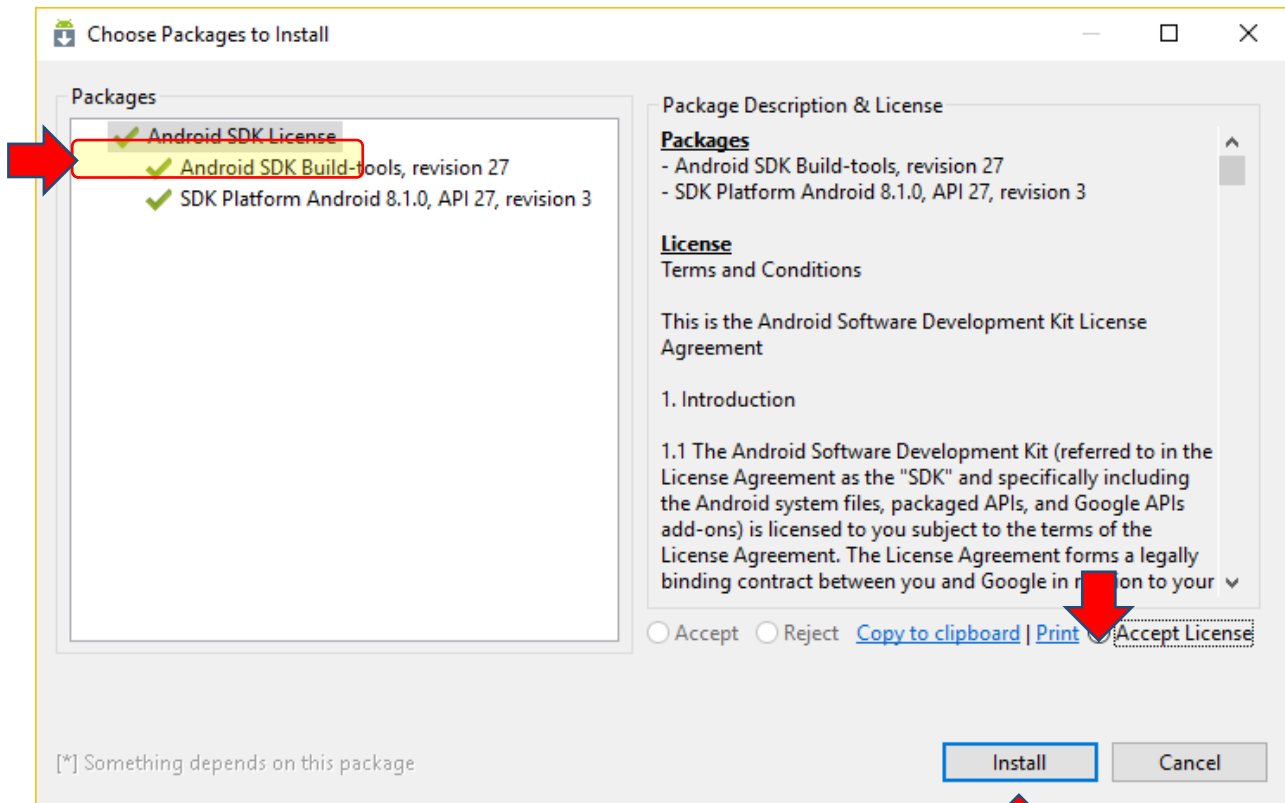
* Get more help at https://help.gradle.org

BUILD FAILED in 3m 59s
cmd: Command failed with exit code 1 Error output:
FAILURE: Build failed with an exception.
  
```

Para resolver esse problema, execute o **SDK Manager** como **Administrador**.

Clique com o botão direito do mouse sobre o arquivo **C:\Program Files (x86)\Android\android-sdk\SDK Manager.exe** e selecione a opção **Executar como Administrador**.

No SDK Manager, selecionei as opções **Android SDK Build-tools Rev. 27** e o **SDK Platform Android 8.1.0 API 27**, como indicado na falha do build acima. Ao final, clique em **Install 2 Packages**.



****** ATENÇÃO: SOMENTE EXECUTE ESSE PASSO SE HOUVER FALHA NO LAUNCH DO APLICATIVO. ******

Ao executar o comando **CORDOVA RUN ANDROID**, normalmente temos um problema de timeout para o deploy no device. Portanto, você deverá aumentar o arquivo **q.js** do seu projeto cordova.

Solução para não dar mais o erro de timed out:

- No seu projeto, abra o arquivo **platforms\android\cordova\node_modules\q\q.js**
- Altere a linha **1825** para a linha **1826** e salve o arquivo (Ctrl+S).

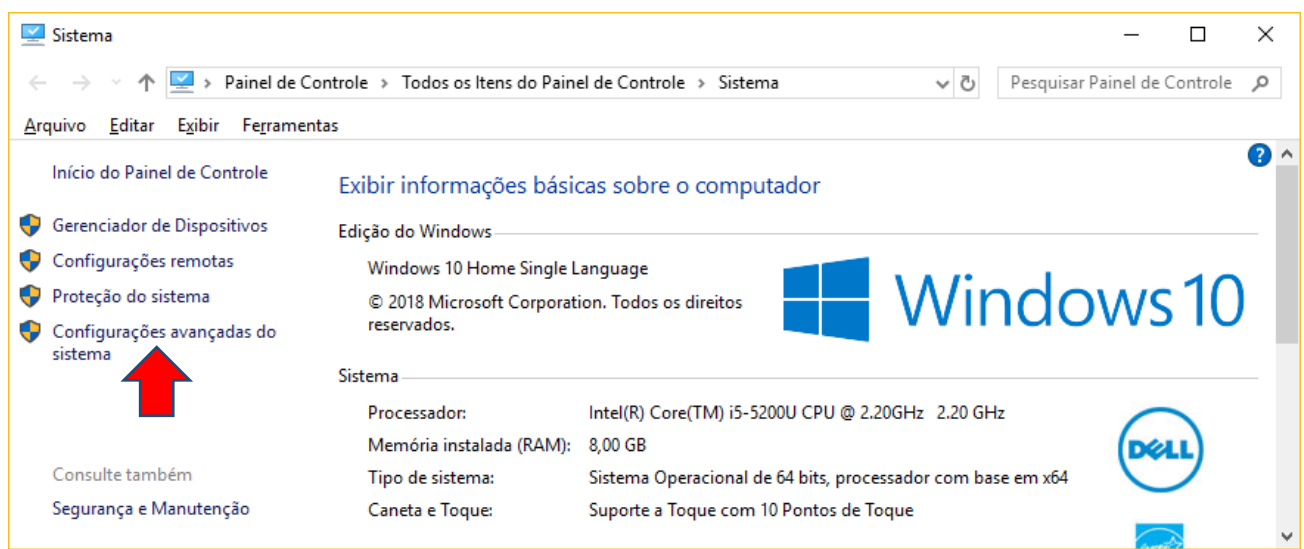
```
1810 * @returns a promise
1811 * fulfilled before
1812 */
1813 Q.timeout = function (object, ms, error) {
1814     return Q(object).timeout(ms, error);
1815 };
1816
1817 Promise.prototype.timeout = function (ms, error) {
1818     var deferred = defer();
1819     var timeoutId = setTimeout(function () {
1820         if (!error || "string" === typeof error) {
1821             error = new Error(error || "Timed out after " + ms + " ms");
1822             error.code = "ETIMEDOUT";
1823         }
1824         deferred.reject(error);
1825     }, ms); => Esta linha era assim, troque pelo conteúdo abaixo */
1826 }, 40000);
1827
1828 this.then(function (value) {
1829     clearTimeout(timeoutId);
1830     deferred.resolve(value);
1831 }, function (exception) {
```

Após isso, execute novamente o projeto com o comando:

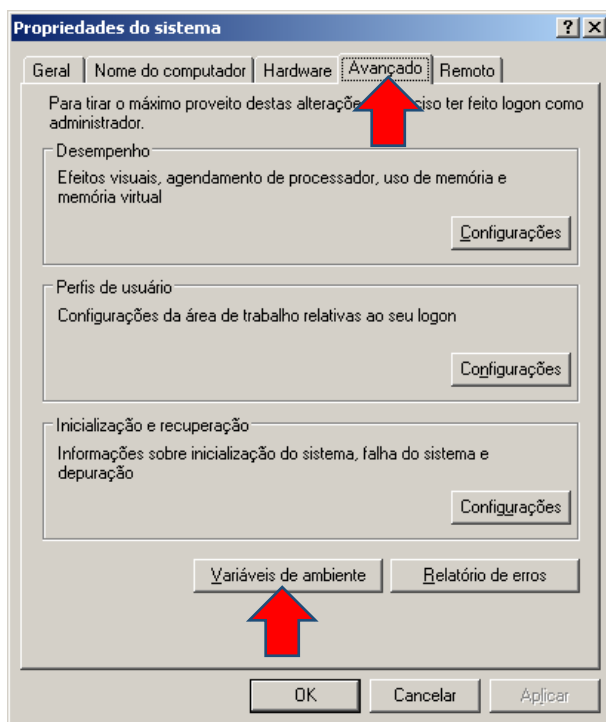
- **CORDOVA RUN ANDROID**

Agora, você vai precisar adicionar o caminho no **PATH**. Para isso, faça o seguinte:

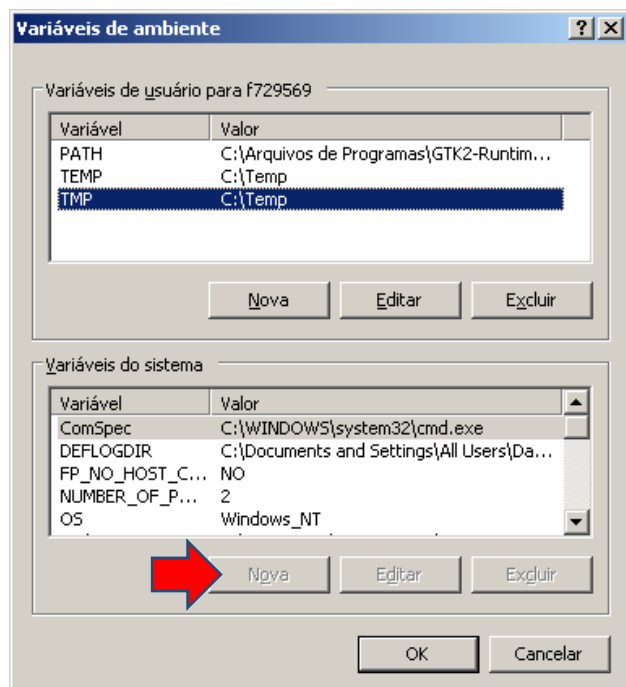
Pressione as teclas **Windows + Pause/Break** para abrir a janela abaixo e escolha a aba **“Configurações Avançadas de Sistema”**.



Clique na aba “**Avançado**” e clique em “**variáveis de ambiente**”.

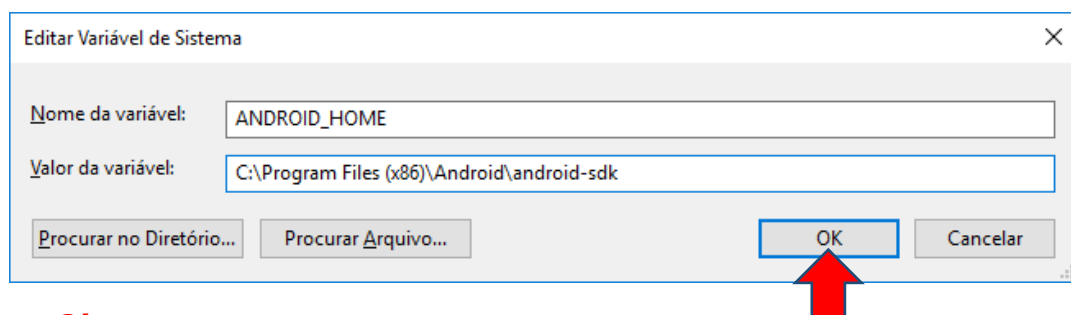


Em variáveis do sistema, clique no botão “**Nova**”.



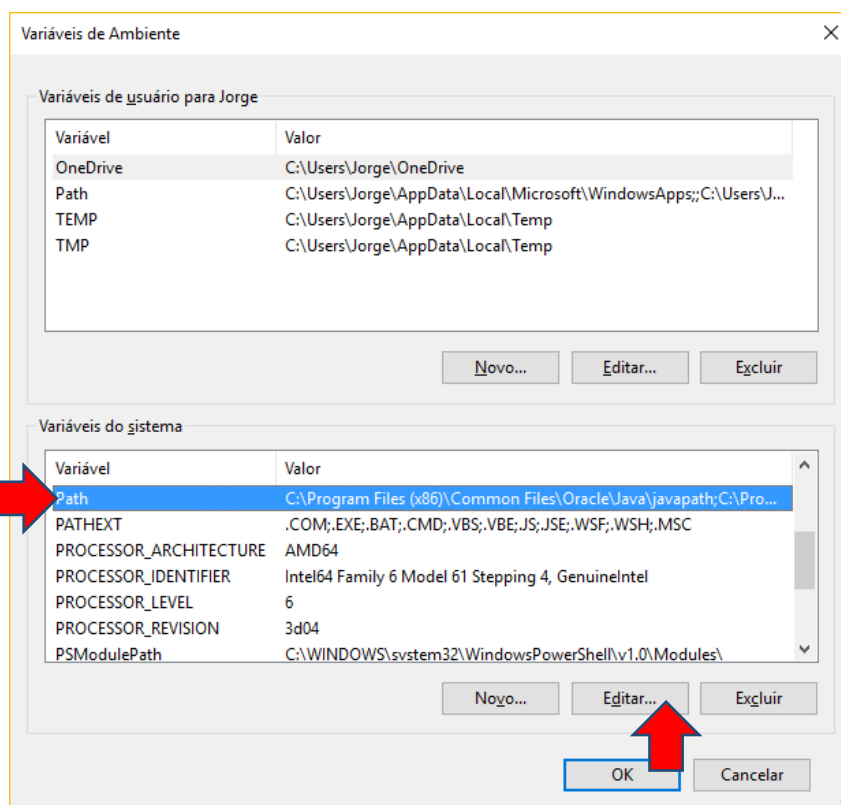
O nome da variável deverá ser **"ANDROID_HOME"** e seu conteúdo será:

"C:\Program Files (x86)\Android\android-sdk".



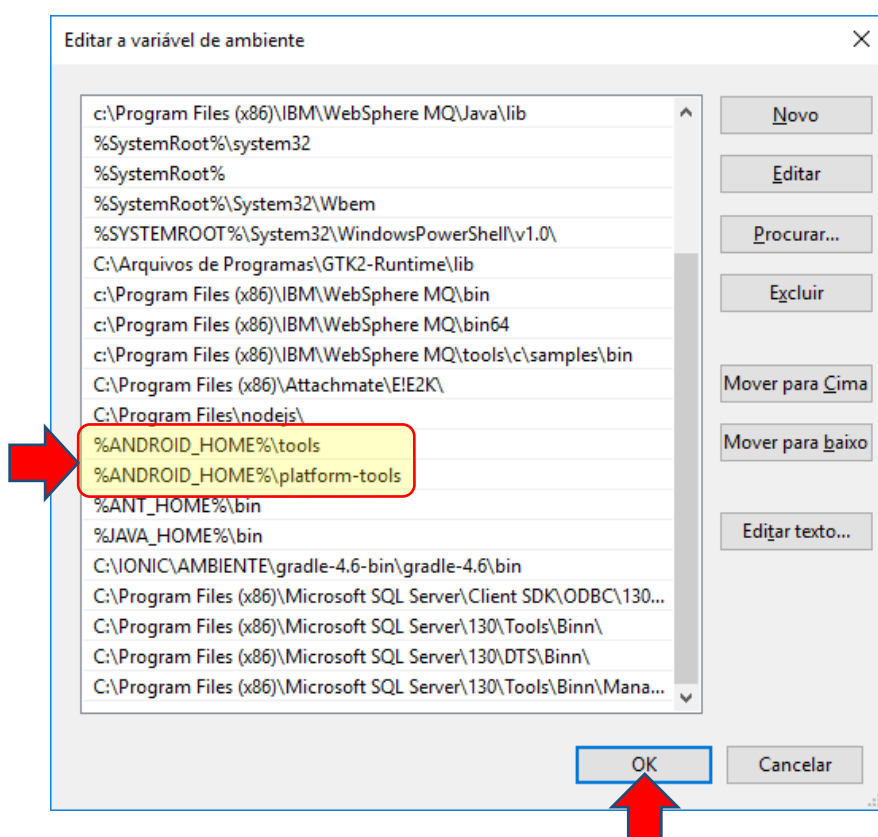
Clique em **Ok**.

Selecione a variável de sistema **"PATH"** e clique em **"Editar"**.



Adicione as duas variáveis abaixo. Para isso, clique no botão **"Novo"**.

%ANDROID_HOME%\tools
%ANDROID_HOME%\platform-tools



Clique em **Ok**.

3) Instalação do **Gradle**

Imagine que você está desenvolvendo um aplicativo para **Android** e precisa empacotar tudo e distribuir para seus usuários, nesse momento é que entram em ação os sistemas de **build**, eles são responsáveis por "pegar" automaticamente todos os recursos (sejam eles arquivos Java ou XML, por exemplo) e utilizar ferramentas específicas (exemplo: compilador de classes Java) em um grupo de arquivos empacotando em um único arquivo, no caso do **Android** um arquivo **.apk**.

Mas então, o que é o Gradle ?

O Gradle é um sistema avançado de automatização de **builds** que une o melhor da flexibilidade do **Ant** com o gerenciamento de dependências e as convenções do **Maven**. Os arquivos de **build** do **Gradle** são *scripts* escritos na linguagem **Groovy**, ao contrário dos formatos de construção do **Ant** e **Maven** que usam arquivos XML para a configuração. Por serem baseados em *scripts*, os arquivos do **Gradle** permitem que você realize tarefas de programação em seu arquivo de configuração. O **Gradle** ainda conta com um sistema de plug-ins que adicionam funcionalidades extras ao seu core.

Vamos iniciar a instalação ?

Passo 1: Download do Gradle

A atual versão do Gradle é a 5.2.1, distribuída em **08/02/2019**, através de um arquivo *.ZIP. Acesse o link abaixo para download automático do arquivo **gradle-5.2.1-bin.zip**:

<https://gradle.org/next-steps/?version=5.2.1&format=bin>

Passo 2: Descompactando o Gradle

Crie um novo diretório **C:\Gradle** pelo Windows Explorer e descompacte o arquivo **gradle-5.2.1-bin.zip** para o novo diretório **C:\Gradle**. Ou então, abra o arquivo **gradle-5.2.1-bin.zip** pelo windows explorer através de um duplo clique e arraste o conteúdo da pasta **gradle-5.2.1** para o novo diretório **C:\Gradle**.

Passo 3: Configurando a variável de ambiente

No Windows Explorer, clique com o botão direito do mouse em **Este Computador**, selecione a opção **Propriedades**, selecione a opção **Configurações Avançadas do Sistema** e a opção **Variáveis de Ambiente**. Em variáveis do sistema, selecione a variável **PATH** e adicione o endereço **C:\Gradle\gradle-5.2.1\bin**. Ao final, clique nos botões **OK**, **OK** e **OK**.

Passo 4: Verificando a instalação

Abra o Prompt do Windows com o comando **CMD** e digite o comando **gradle -v** para exibir a versão instalada:

```
Prompt de Comando
Microsoft Windows [versão 10.0.17134.590]
(c) 2018 Microsoft Corporation. Todos os direitos reservados.

C:\Users\Jorge>gradle -v

-----
Gradle 5.2.1
-----

Build time:   2019-02-08 19:00:10 UTC
Revision:     f02764e074c32ee8851a4e1877dd1fea8ffb7183

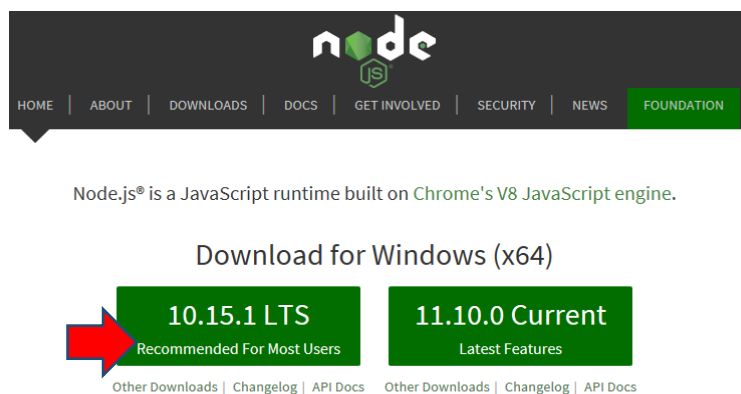
Kotlin DSL:   1.1.3
Kotlin:       1.3.20
Groovy:       2.5.4
Ant:          Apache Ant(TM) version 1.9.13 compiled on July 10 2018
JVM:          1.8.0_201 (Oracle Corporation 25.201-b09)
OS:           Windows 10 10.0 amd64

C:\Users\Jorge>
```

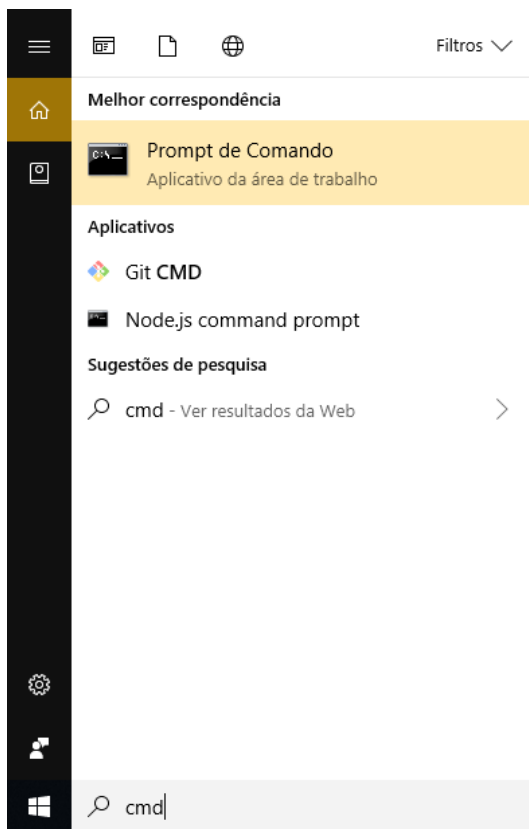
4) Instalação do **Node.js**

Para isso, precisamos do **Node.js** para usar o **Cordova**. Entre no site do Node em <http://nodejs.org>

Clique na versão recomendada.



Ele vai baixar o executável da instalação e você pode simplesmente executá-lo. Faça a instalação padrão com Next, Next, Finish e deixando as opções padrões marcadas. Ao fim, para saber se a instalação foi bem-sucedida, abra o terminal (comando CMD), conforme abaixo:



Agora, digite o comando: **node -v**

Ele deve confirmar a versão instalada do node.

```
Microsoft Windows [versão 10.0.17134.590]
(c) 2018 Microsoft Corporation. Todos os direitos reservados.

C:\Users\Jorge>node -v
v10.15.1

C:\Users\Jorge>
```

5) Instalação do **Cordova**

O Cordova é um pacote do **Node.js** que vamos instalar com sua ferramenta de pacotes, chamada **npm**.

No Windows, feche e abra novamente o terminal, e digite:

npm -g install cordova

No Mac e no Linux, precisamos do **sudo** (usuário administrador) para instalar globalmente, então faça:

sudo npm -g install cordova

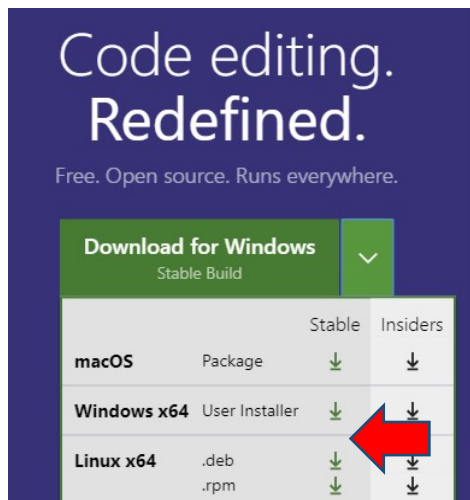
Deve demorar um pouco para fazer a instalação completa.

```
Microsoft Windows [versão 10.0.17134.590]
(c) 2018 Microsoft Corporation. Todos os direitos reservados.

C:\Users\Jorge>npm -g install cordova
[.....] \ extract:glob: sill extract glob@7.1.3 extracted to C:\Users\Jorge\Ap
```

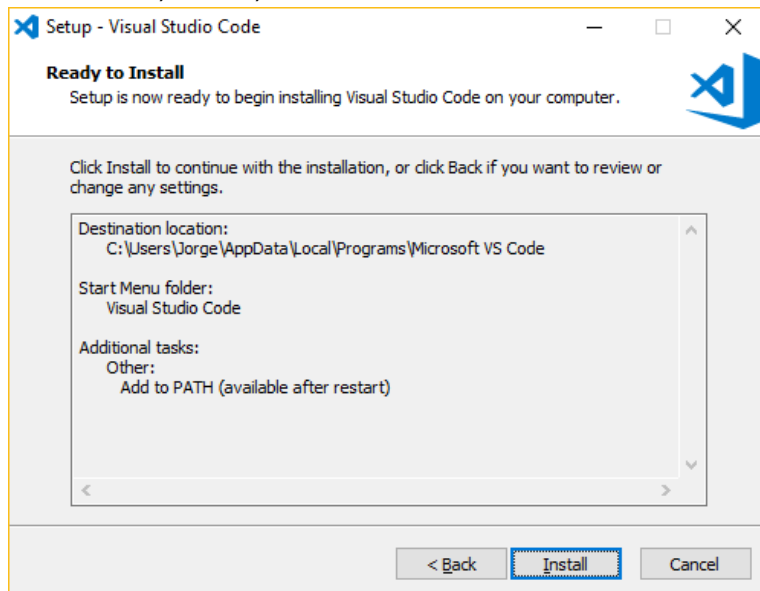
6) Instalação do **Visual Studio Code**

Para isso, acesse o endereço: <https://code.visualstudio.com/> e selecione a opção abaixo:



Selecione a versão **Windows x64 User Installer (Stable)**

Execute o instalador **VSCodeUserSetup-x64-1.31.1.exe** e faça a instalação padrão com **Next, Next, Install**.

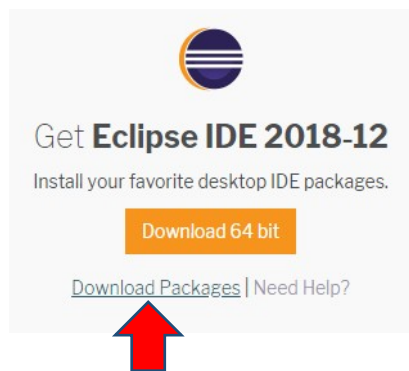


7) Instalação do **Eclipse**, conforme visto no laboratório.

1. Para isso, acesse o endereço abaixo:

<https://www.eclipse.org/downloads/>

2. Clique em:



3. Selecione a opção abaixo:



Ao final, se existir, apague a pasta **eclipse** que está no diretório **C:** e descompacte o arquivo ZIP para o diretório **C:**. A pasta **Eclipse** será criada e a instalação efetuada com sucesso. Após isso, basta dar um duplo clique no arquivo **Eclipse.exe** para execução do ambiente de desenvolvimento. Durante a execução, defina o diretório **Workspace**.