**Setup React Native Development Environment**

**Introduction:**

You will need Node, the React Native command line interface, a JDK, and Android Studio.

**Step 1:**

Installing Nodejs:

* Visit https://nodejs.org/en and download the latest LTS version of Node.js.
* Install Node.js on your Windows system.

**Step 2:**

Installing Java JDK:

* Go to https://www.openlogic.com/openjdk-downloads and download the Java 17 JDK installer.
* Install the Java 17 JDK on your Windows system.

Now, follow carefully these steps are important:

* Open the Advanced System Properties from the Windows Control Panel.
* Click on "Environment Variables".
* Under "System variables", click "New".
* Set the variable name to JAVA\_HOME.
* Set the variable value to your Java installation directory, for example, "c:\programfiles\openjdk".
* Edit the "Path" variable and add "%JAVA\_HOME%\bin" to the variable value.

**Step 3:**

Installing Android Studio:

* Go to https://developer.android.com/studio and download the latest version of Android Studio.
* During the installation wizard:
  + Ensure that the boxes next to the following items are checked:
    - Android SDK
    - Android SDK Platform
    - Android Virtual Device

Then, click "Next" to install all of these components.

**Step 4:**

After Installing Android Studio:

* Open Android Studio and click on the "More Actions" button.
* Select "SDK Manager".
* In the SDK Manager:
* Go to the "SDK Platforms" tab.
* Check the box next to "Show Package Details" in the bottom right corner.
* Find and expand the "Android 14 (UpsideDownCake)" entry.
* Ensure the following items are checked:
  + Android SDK Platform 34
  + Intel x86 Atom\_64 System Image or Google APIs Intel x86 Atom System Image
* Switch to the "SDK Tools" tab.
* Check the box next to "Show Package Details".
* Look for and expand the "Android SDK Build-Tools" entry.
* Make sure version 34.0.0 is selected.

Finally, click "Apply" to download and install the Android SDK and related build tools.

**Step 5:**

Configure the ANDROID\_HOME environment variable:

* Open the Windows Control Panel.
* Click on User Accounts, then click User Accounts again
* Click on Change my environment variables
* Click on New... to create a new ANDROID\_HOME user variable that points to the path to your Android SDK (C:\users\user\AppData\Local\Android\Sdk)
* The SDK is installed, by default, at the following location: (%LOCALAPPDATA%\Android\Sdk)

Open a new Command Prompt window to ensure the new environment variable is loaded before proceeding to the next step.

* Open powershell
* Copy and paste **Get-ChildItem -Path Env:\** into powershell
* Verify ANDROID\_HOME has been added.

**Step 6:**

Adding platform-tools to Path:

* Open the Windows Control Panel.
* Click on "User Accounts", then click "User Accounts" again.
* Click on "Change my environment variables".
* Select the "Path" variable.
* Click "Edit".
* Click "New" and add the path to platform-tools to the list.
  + The default location for this folder is: %LOCALAPPDATA%\Android\Sdk\platform-tools

**Step 7:**

Let's create a First React Native project called "TodoList":

* Open Command Prompt in the directory where you want to create the project.
* Run the following command:

npx react-native@latest init TodoList

* Wait for the initialization process to complete.
* Once completed, open the project in Android Studio.

Step 8:

* Open Android Studio and go to "Android Virtual Devices (AVDs)" by opening the "AVD Manager".
* If necessary, create a new AVD by selecting "Create Virtual Device...", then choose any Phone from the list and click "Next", followed by selecting the UpsideDownCake API Level 34 image.
* Click on "Run" to start the emulator.
* In your project folder's terminal, run the following command:
  + npm start

Once the Metro Bundler is running, run the following command to launch the app in the emulator:

* npm run andriod

If everything is set up correctly, you should see your new app running in your Android emulator shortly.