Hardware Requirement to install Android Studio

Windows

1. **Microsoft® Windows®** 7/8/10 (64-bit)
2. 4 GB RAM minimum, 8 GB RAM recommended.
3. 2 GB of available disk space minimum, 4 GB Recommended (500 MB for IDE + 1.5 GB for **Android SDK** and emulator **system** image)
4. 1280 x 800 minimum screen resolution.

What is API Level?

API Level is an integer value that uniquely identifies the framework API revision offered by a version of the Android platform.

The **Android** platform provides a framework **API** that applications can use to interact with the underlying **Android** system. The framework **API** consists of: A core set of packages and classes. .

Android 9 (**API** level **28**) introduces great new features and capabilities for users and developers

The **Android** Hardware Abstraction Layer (**HAL**) is an interface for hardware vendors to implement that allows the **Android** application/framework to communicate with hardware-specific device drivers.

The **Linux® kernel** is the main component of a Linux operating system (OS) and is **the core** interface between a computer's hardware and its processes.

The Android platform provides **Java** framework APIs to expose the functionality of some of these native libraries to apps. For example, you can access **OpenGL** ES through the Android framework's **Java**

OpenGL API to add support for drawing and manipulating 2D and 3D graphics in your app.

Download and Install Android Studio

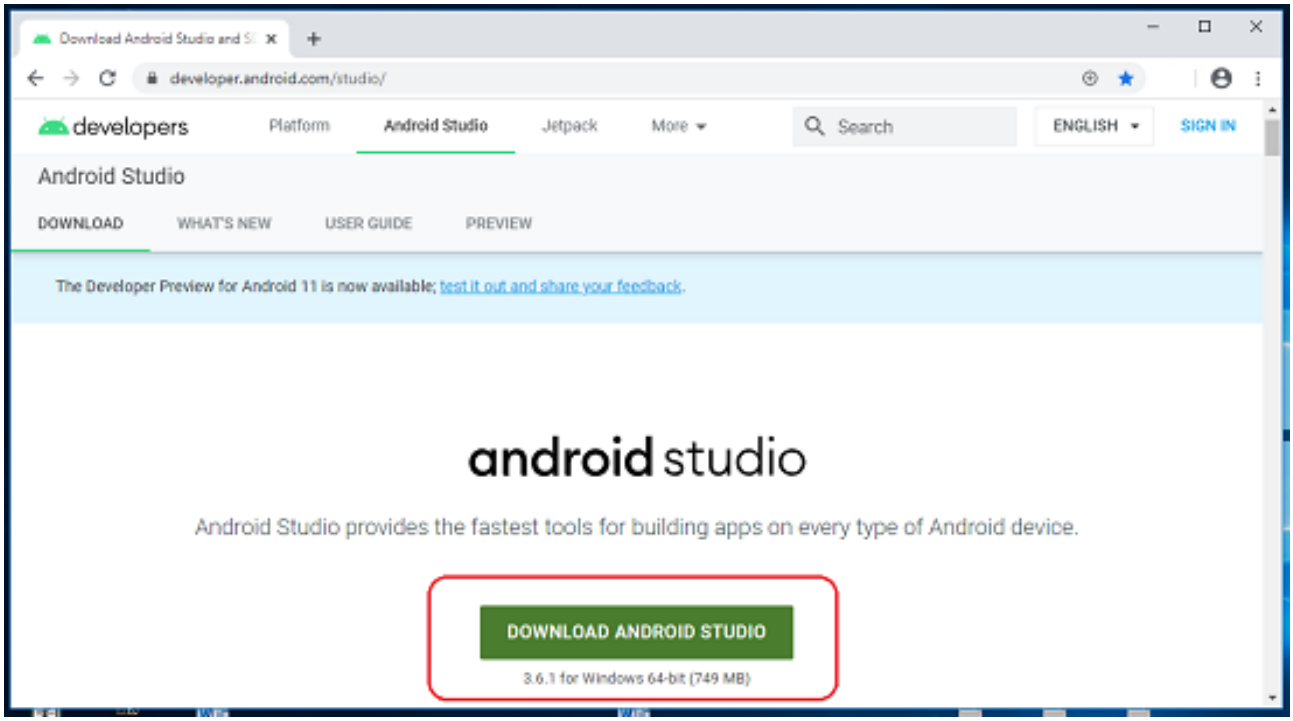
Step 1

To download the Android Studio, visit the official Android Studio website in your web browser.

https://developer.android.com/studio

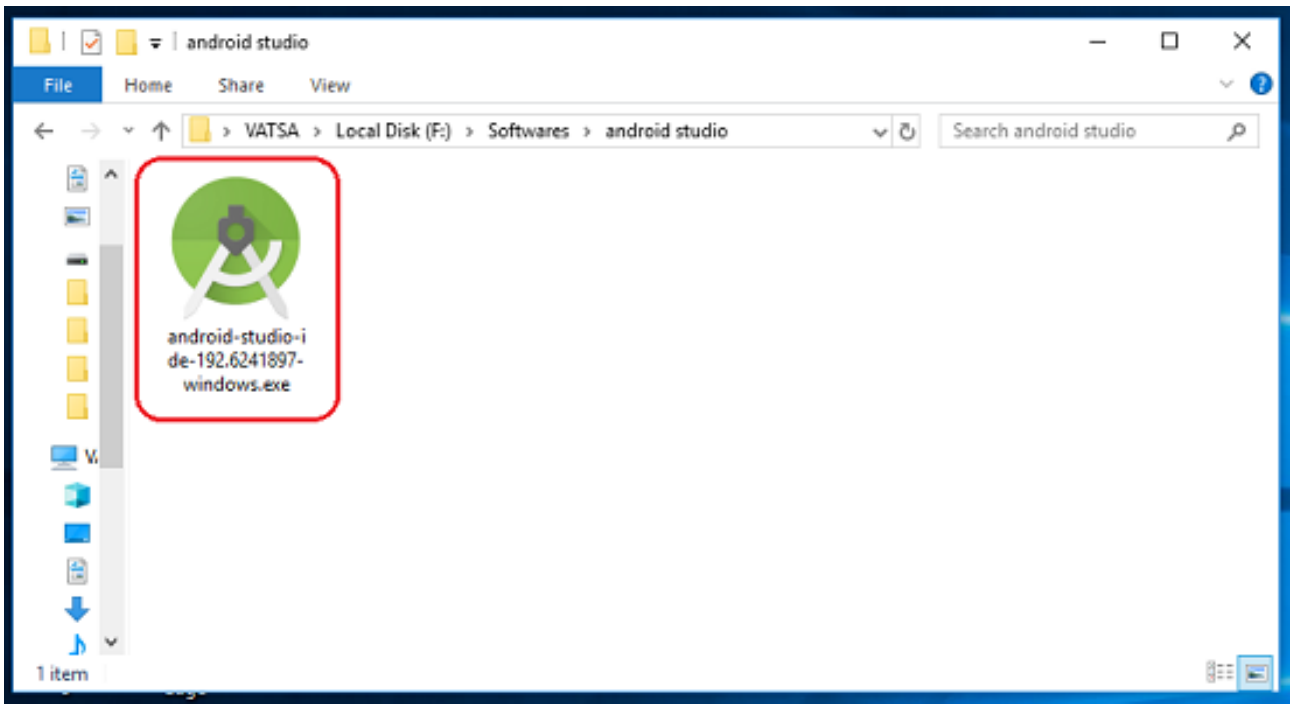
Step 2

Click on the "Download Android Studio" option.



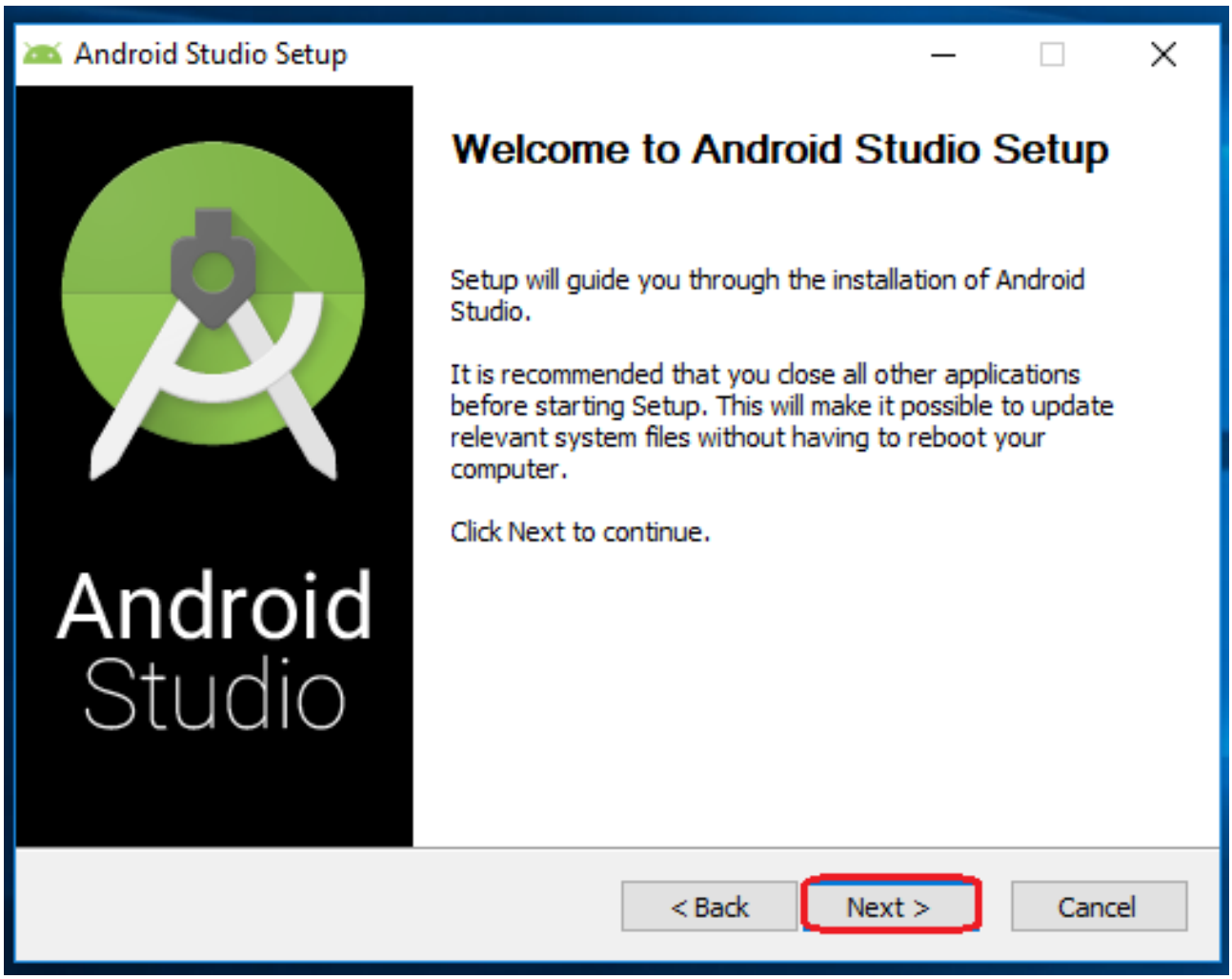
Step 3

Double click on the downloaded "Android Studio-ide.exe" file.



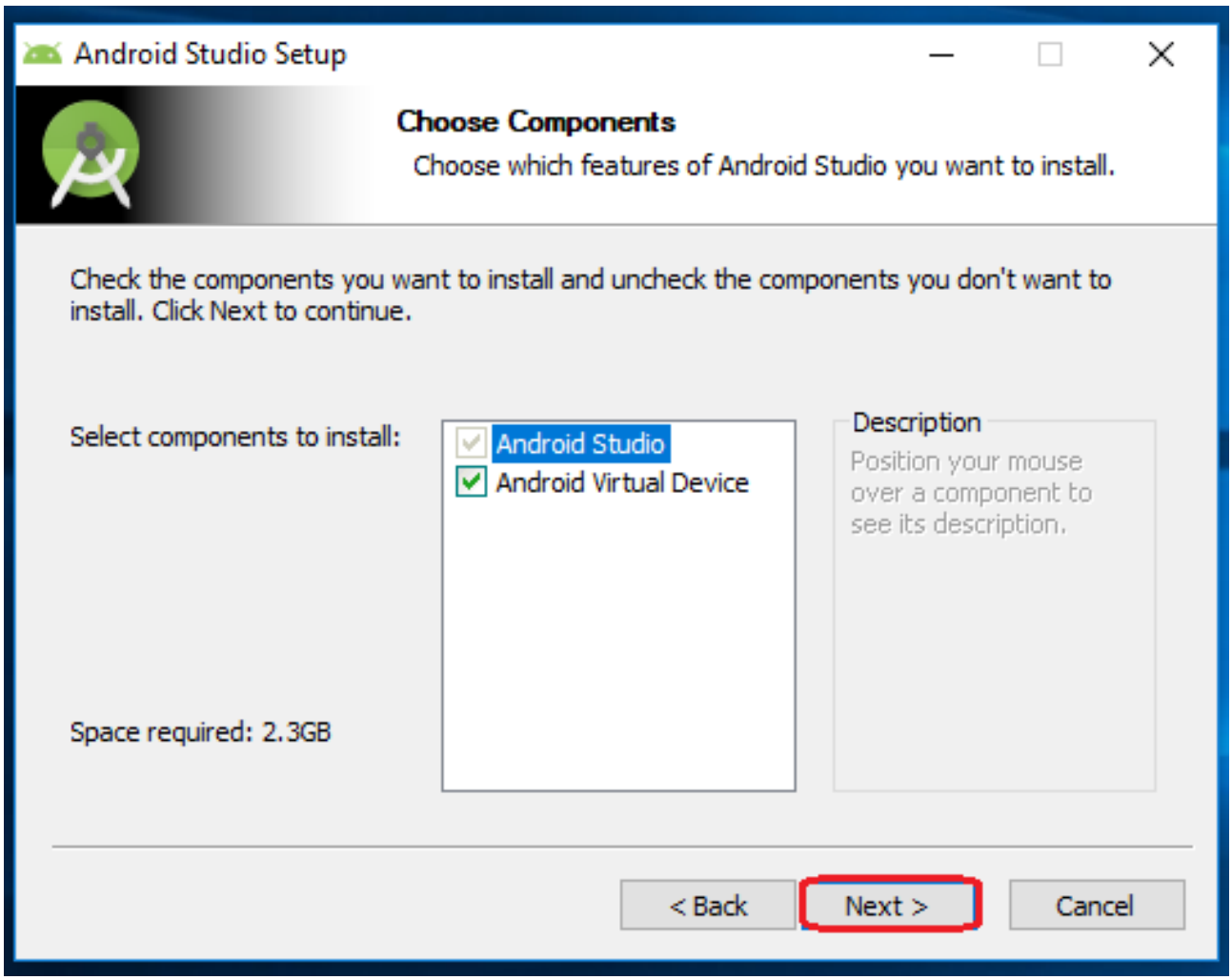
Step 4

"Android Studio Setup" will appear on the screen and click "Next" to proceed.



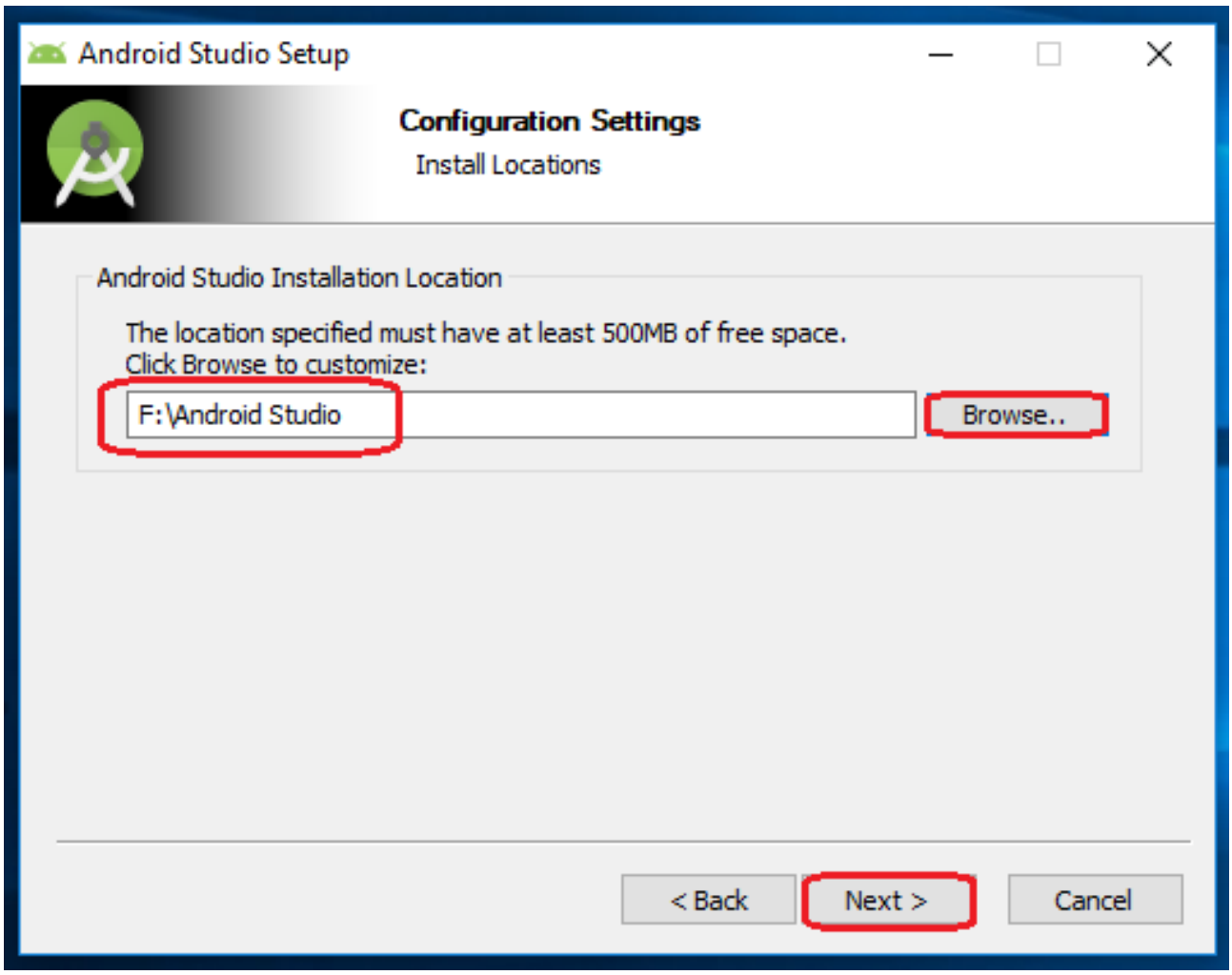
Step 5

Select the components that you want to install and click on the "Next" button.



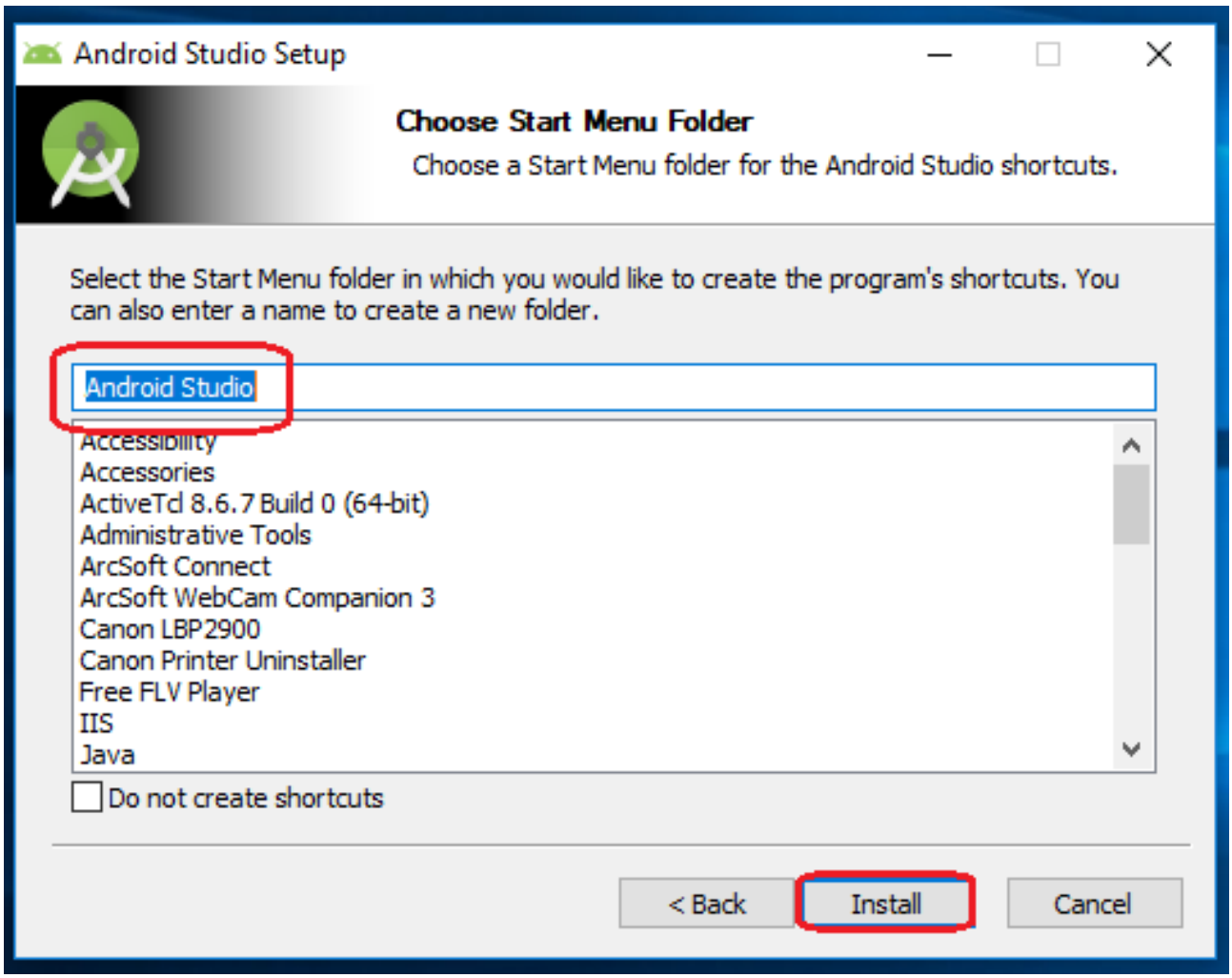
Step 6

Now, browse the location where you want to install the Android Studio and click "Next" to proceed.



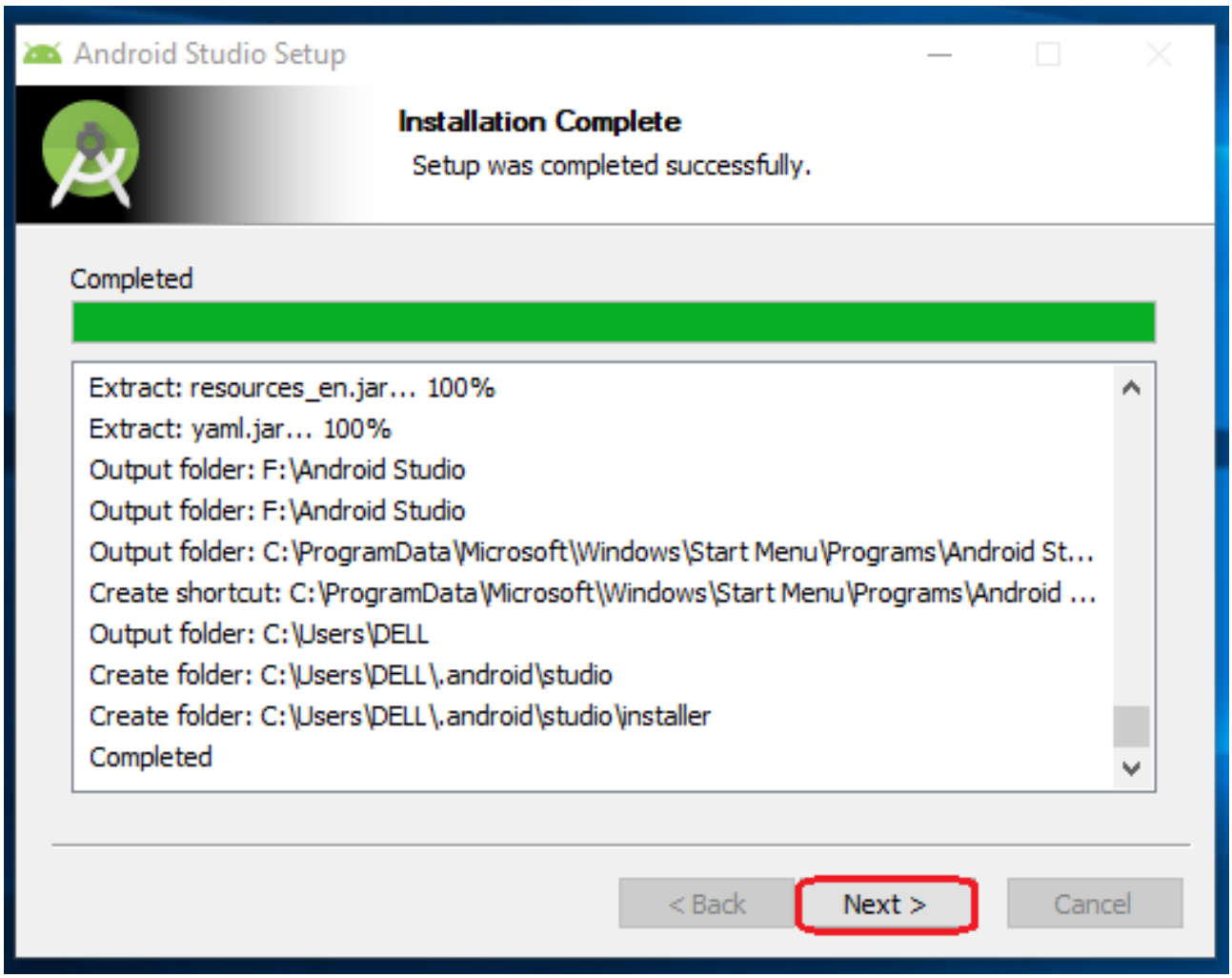
Step 7

Choose a start menu folder for the "Android Studio" shortcut and click the "Install" button to proceed.



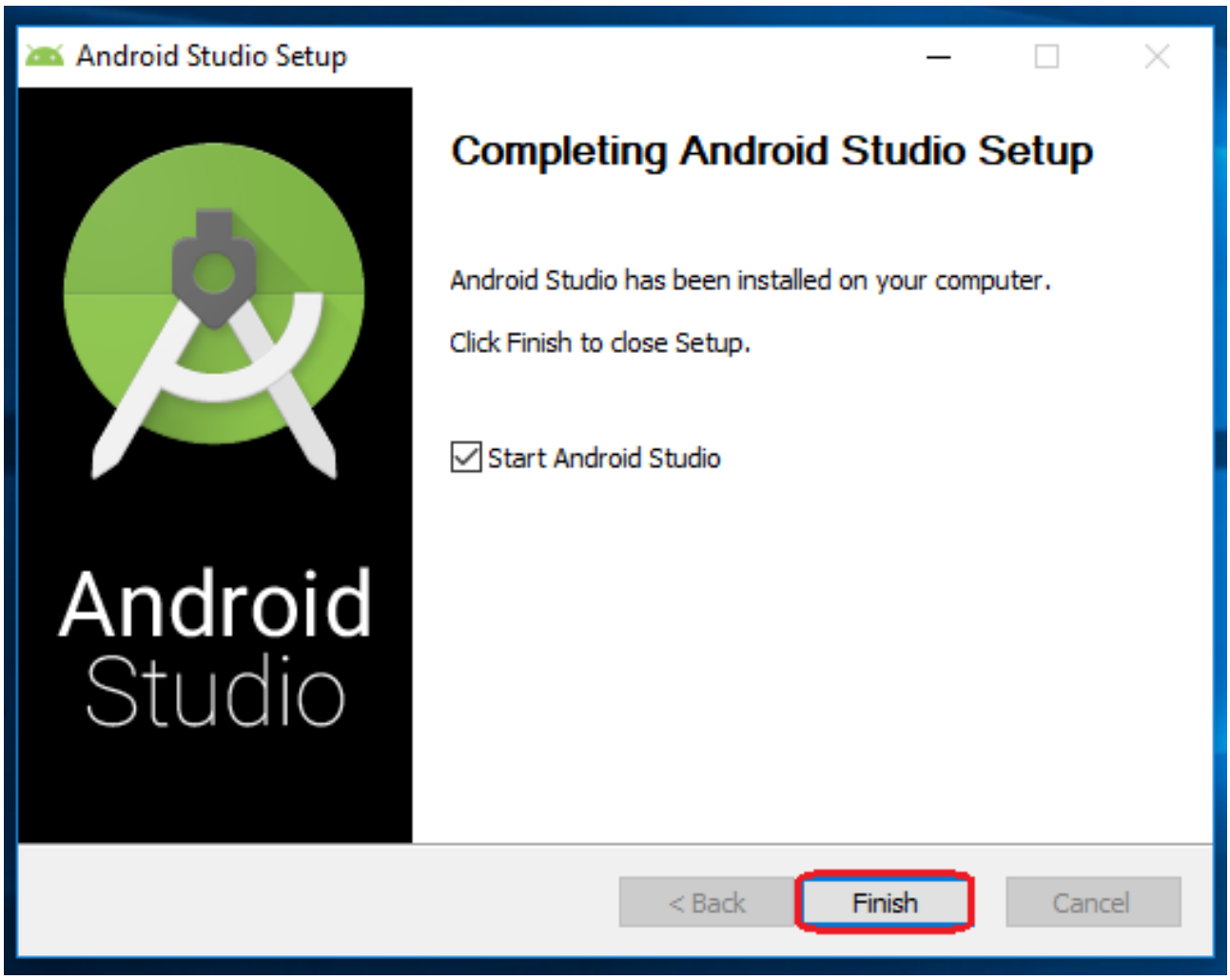
Step 8

After the successful completion of the installation, click on the "Next" button.



Step 9

Click on the "Finish" button to proceed.

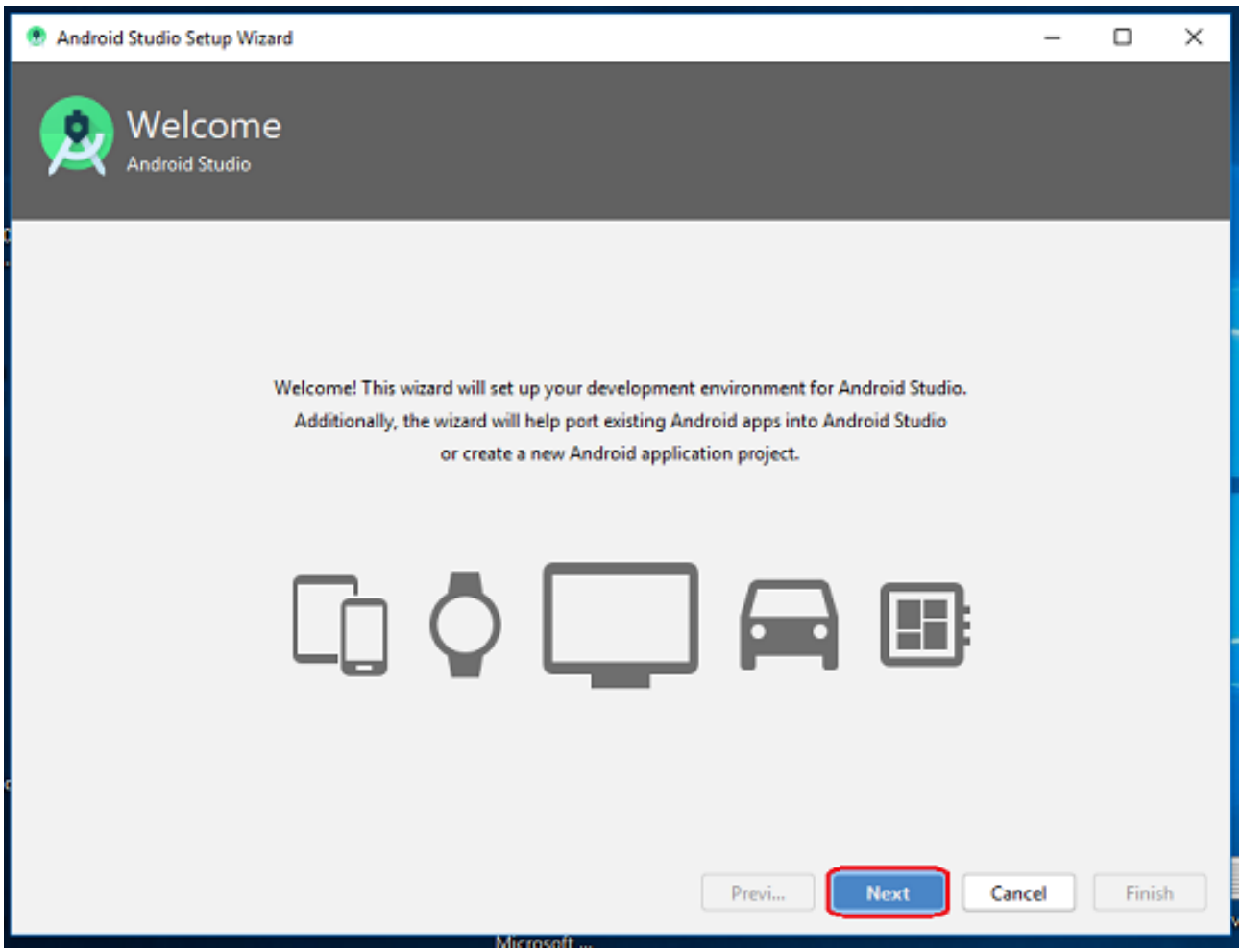


Now, your Android studio welcome screen will appear on the screen.

Android Studio Setup Configuration

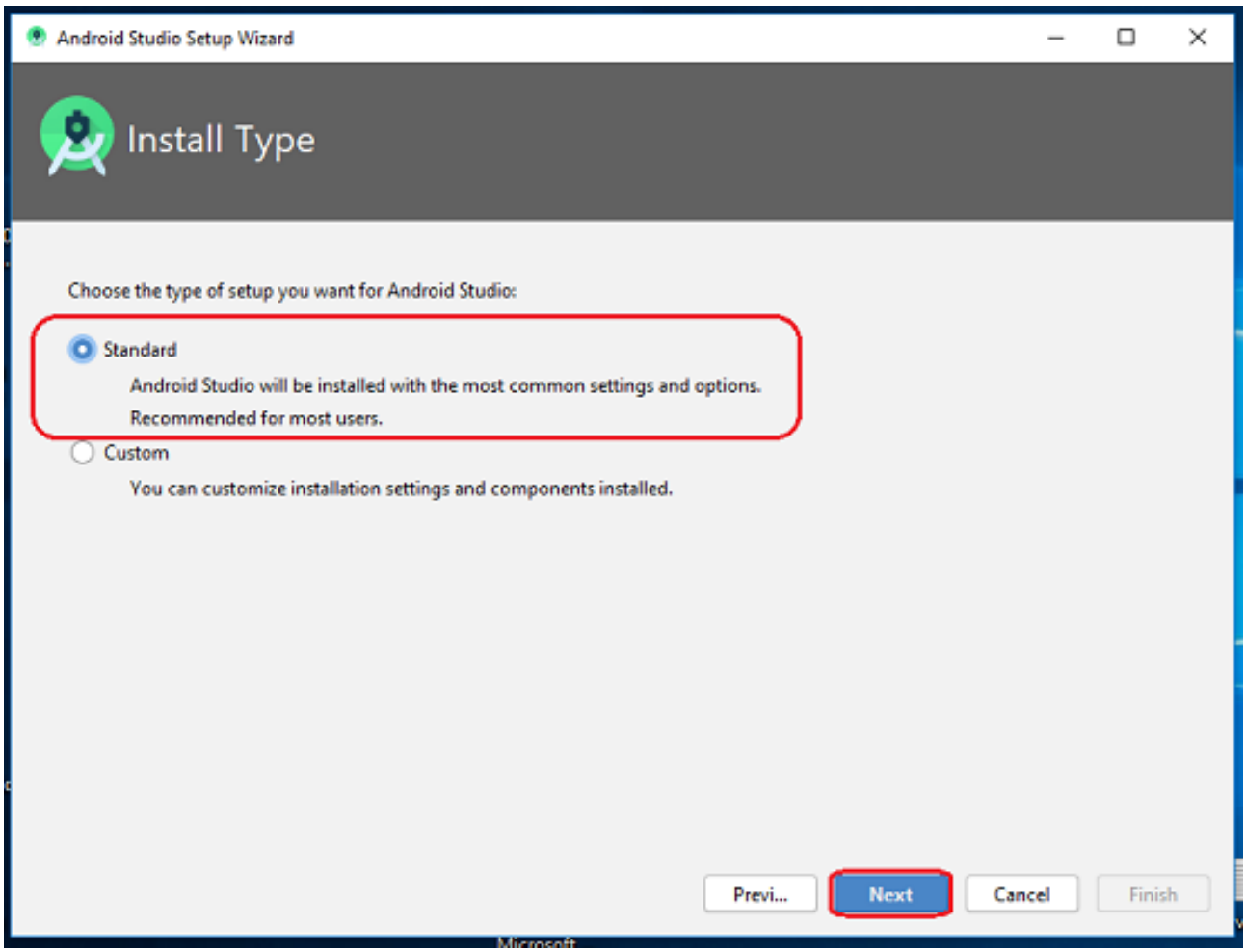
Step 10

"Android Studio Setup Wizard" will appear on the screen with the welcome wizard. Click on the "Next" button.



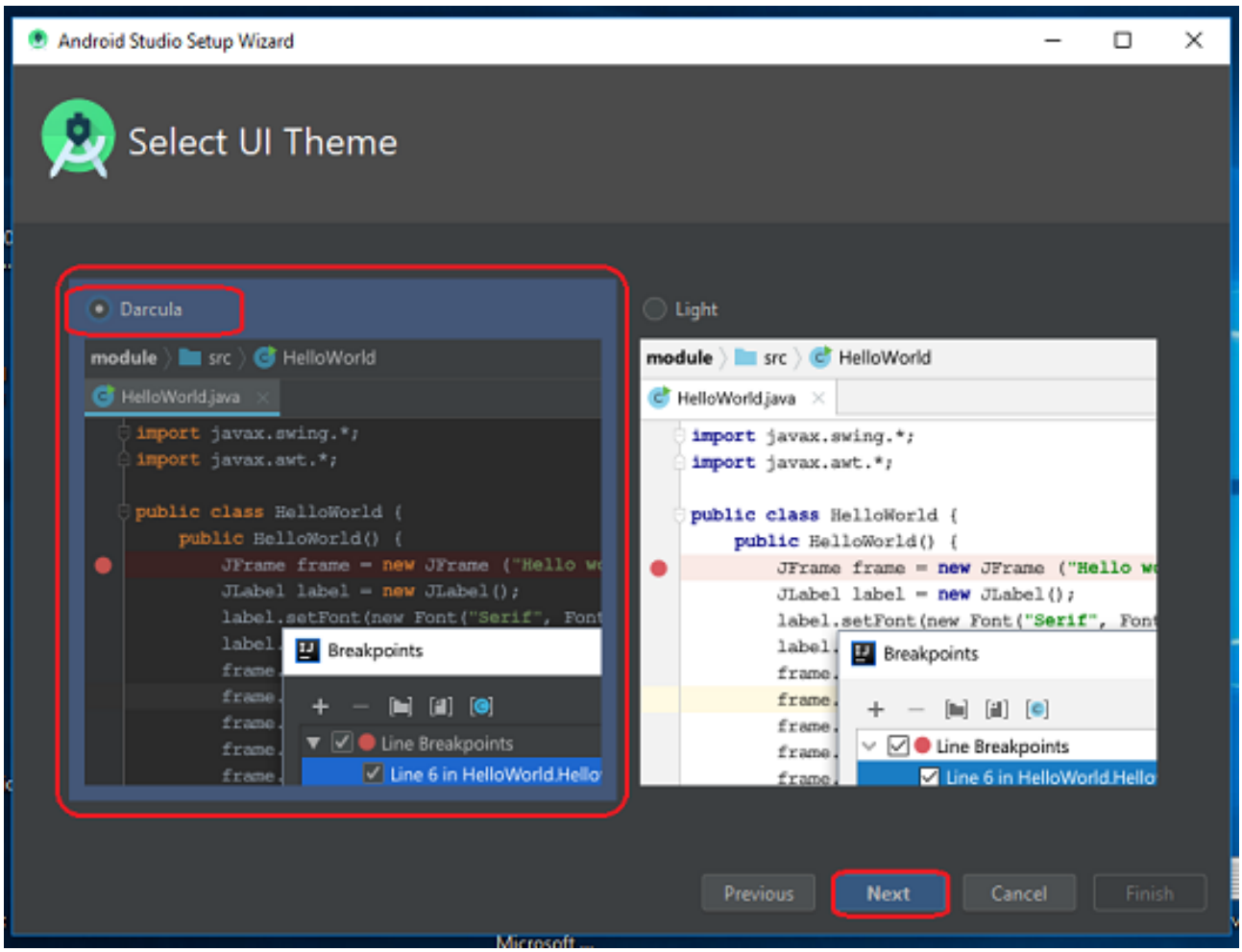
Step 11

Select (check) the "Standard" option if you are a beginner and do not have any idea about Android Studio. It will install the most common settings and options for you. Click "Next" to proceed.



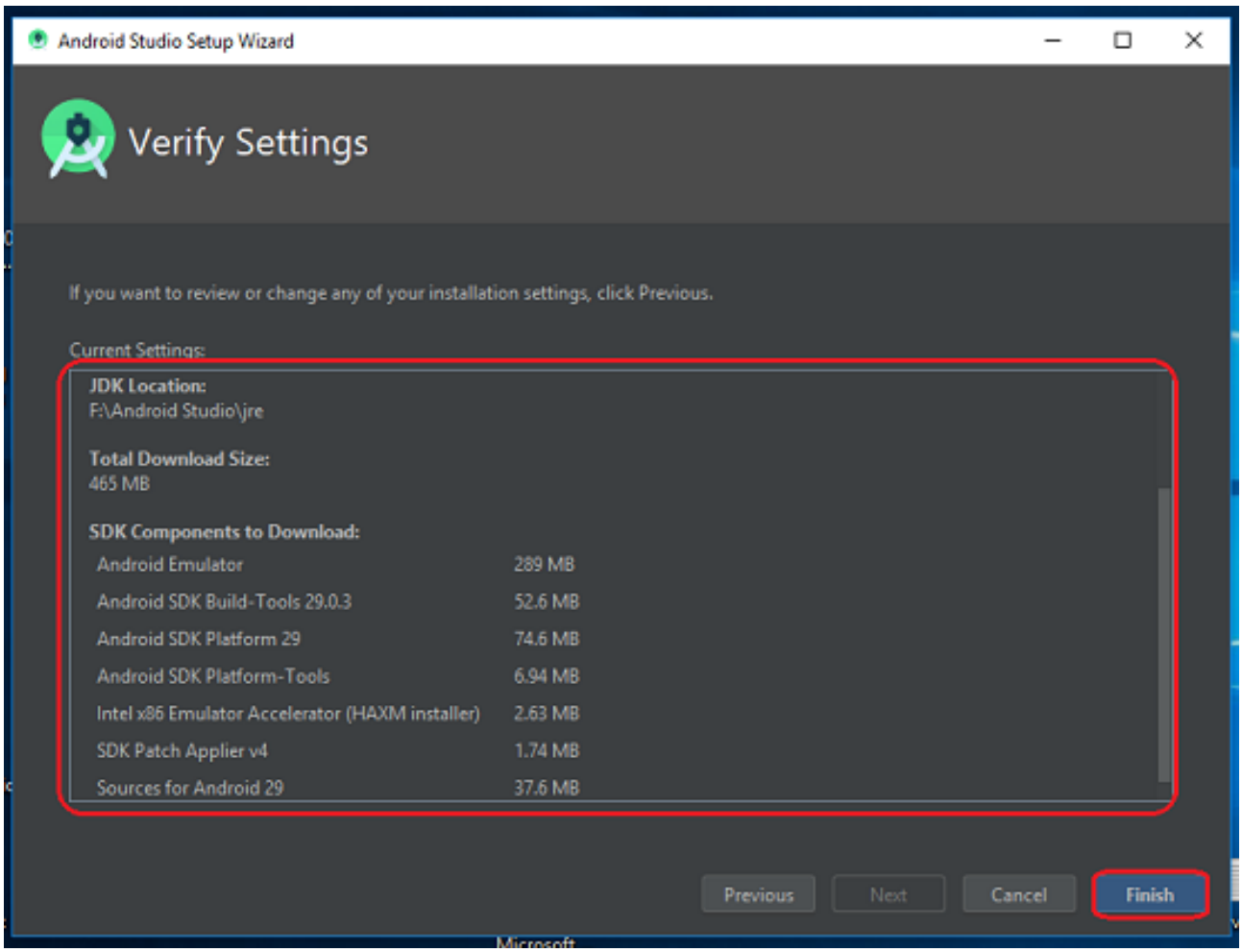
Step 12

Now, select the user interface theme as you want. (I prefer Dark theme (Dracula) that is most liked by the coders). Then, click on the "Next" button.



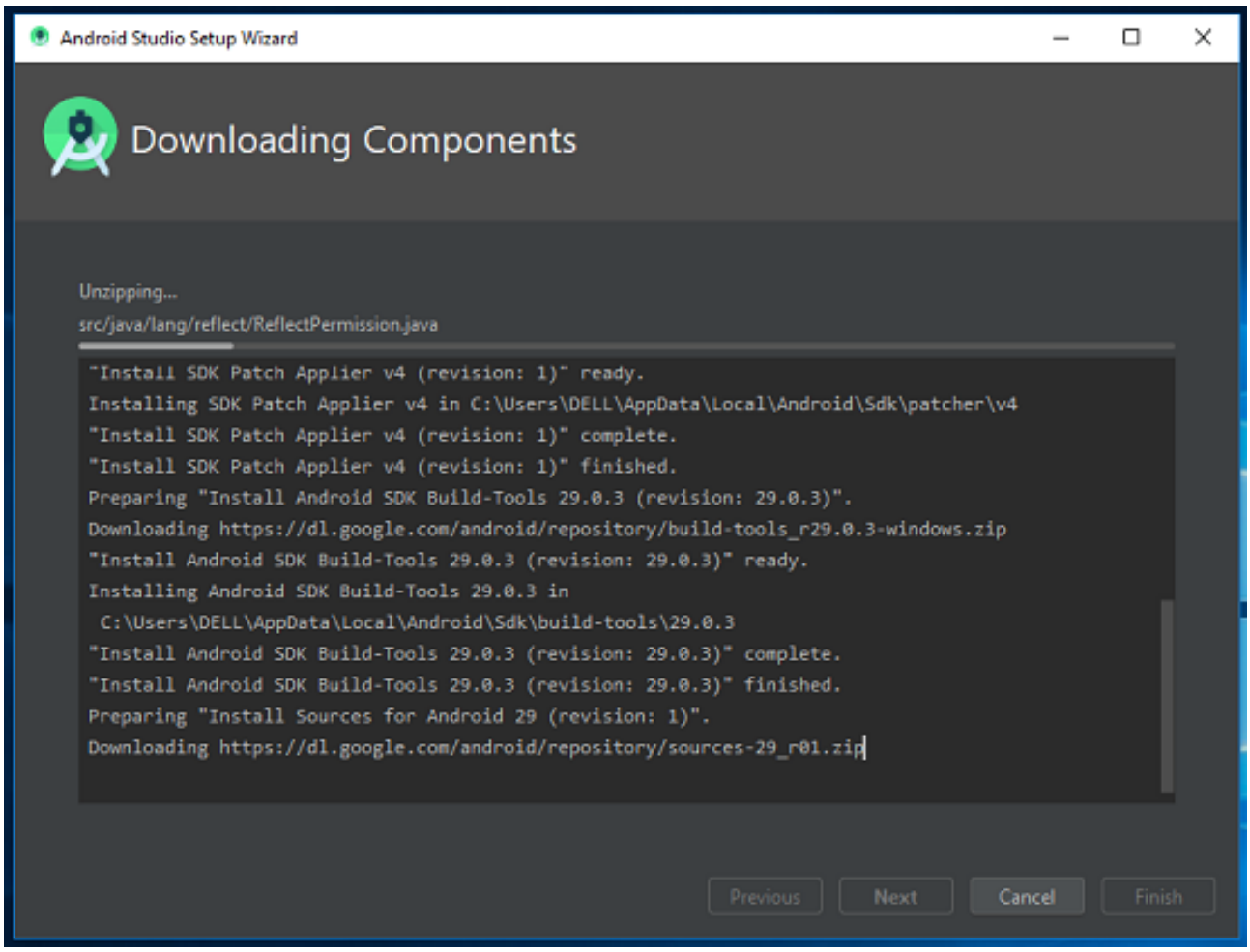
Step 13

Now, click on the "Finish" button to download all the SDK components.



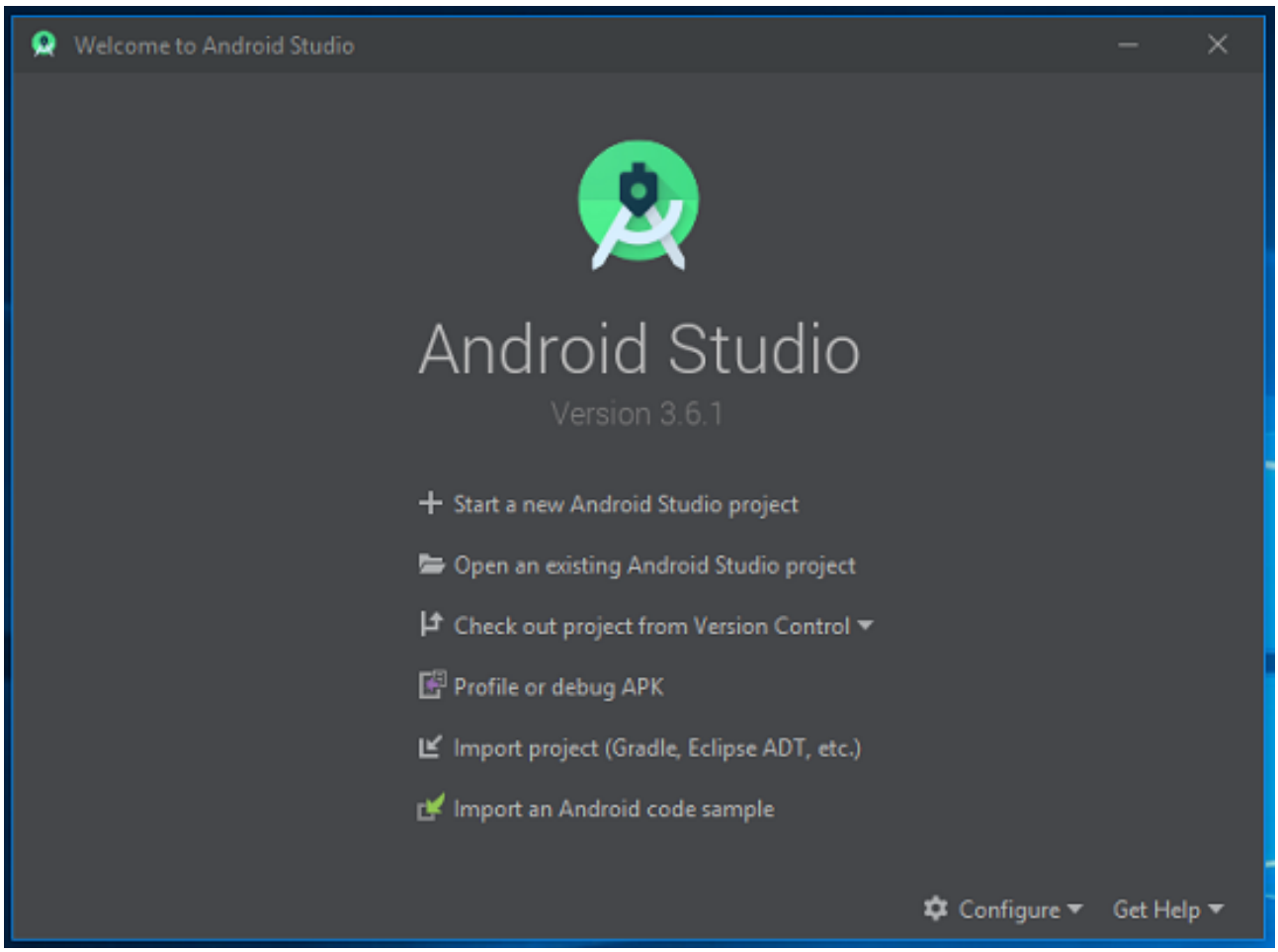
And, the downloading and installation process of components gets started.

C:\Program Files\Java\jdk1.8 and jre1.8



Step 14

After downloading all the necessary components, click on the "Finish" button.



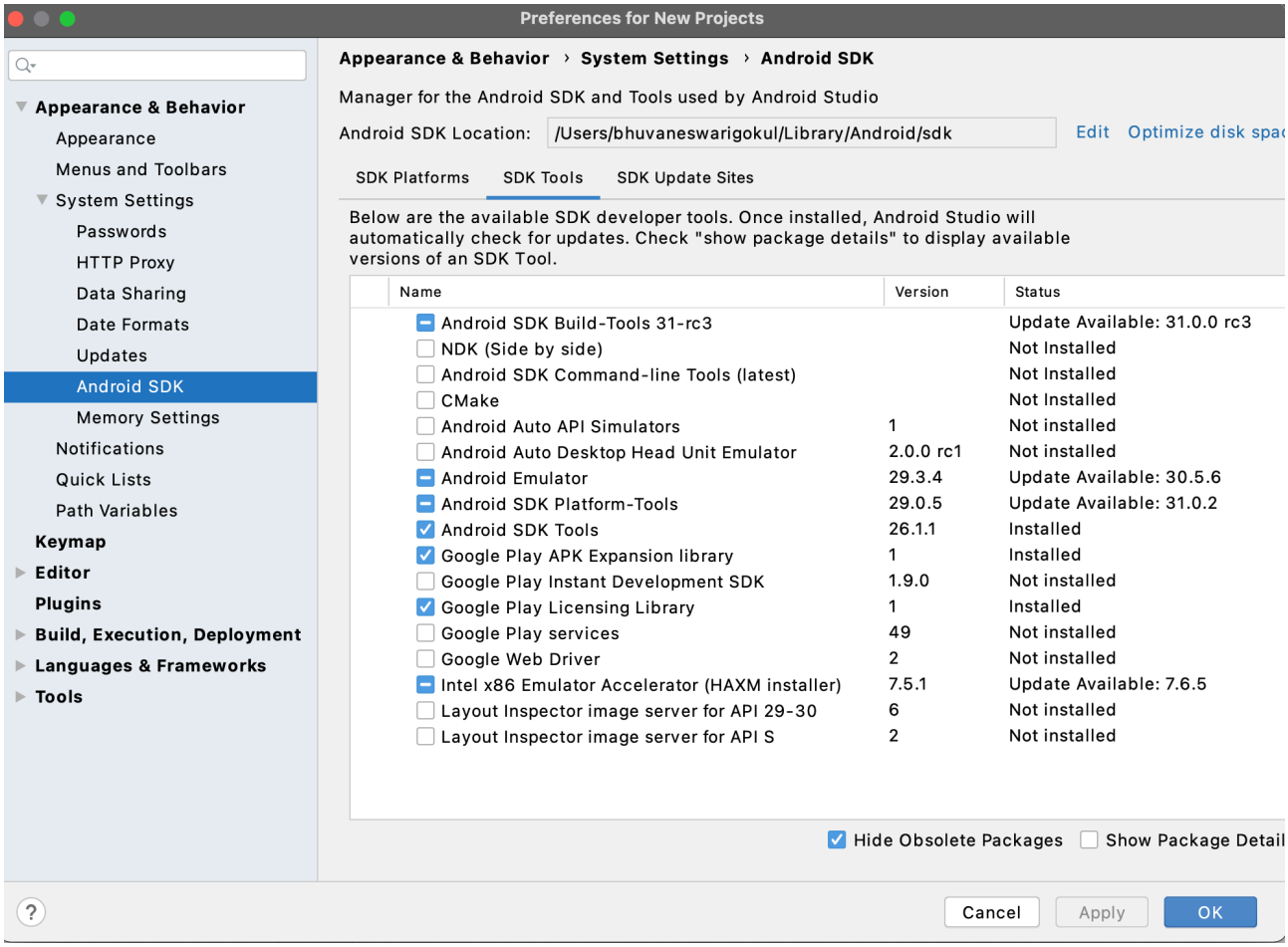
Congrats, your Android Studio has been successfully installed in your system and you can start a new Android studio project.

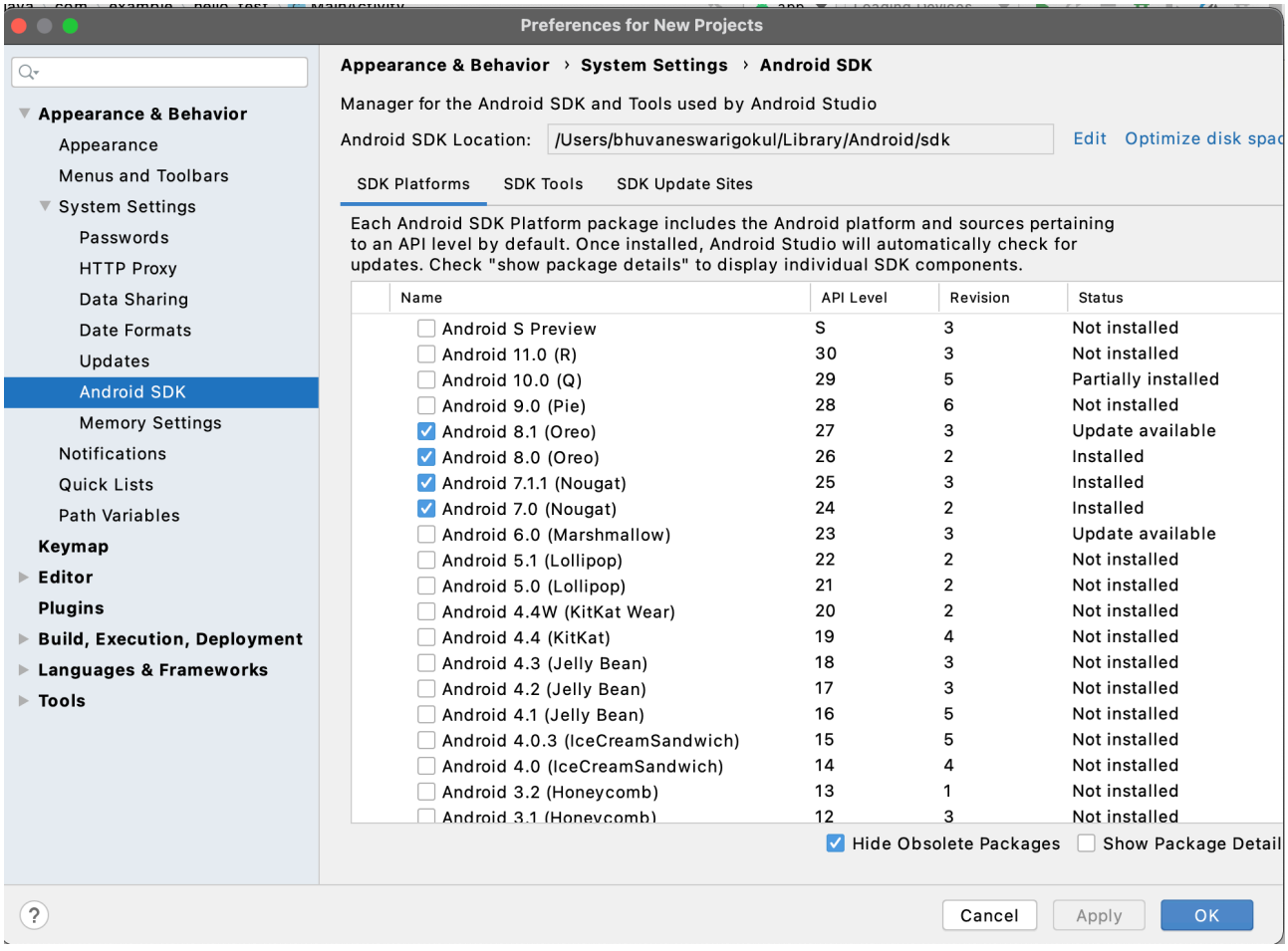
Android - SDK Manager

Running Android SDK Manager

Once downloaded, you can launch Android SDK Manager in one of the following ways −

- Click **tools->Android-> SDK Manager** option in Eclipse.
- Double Click on the **SDK Manager.exe** file in the Android SDK folder.

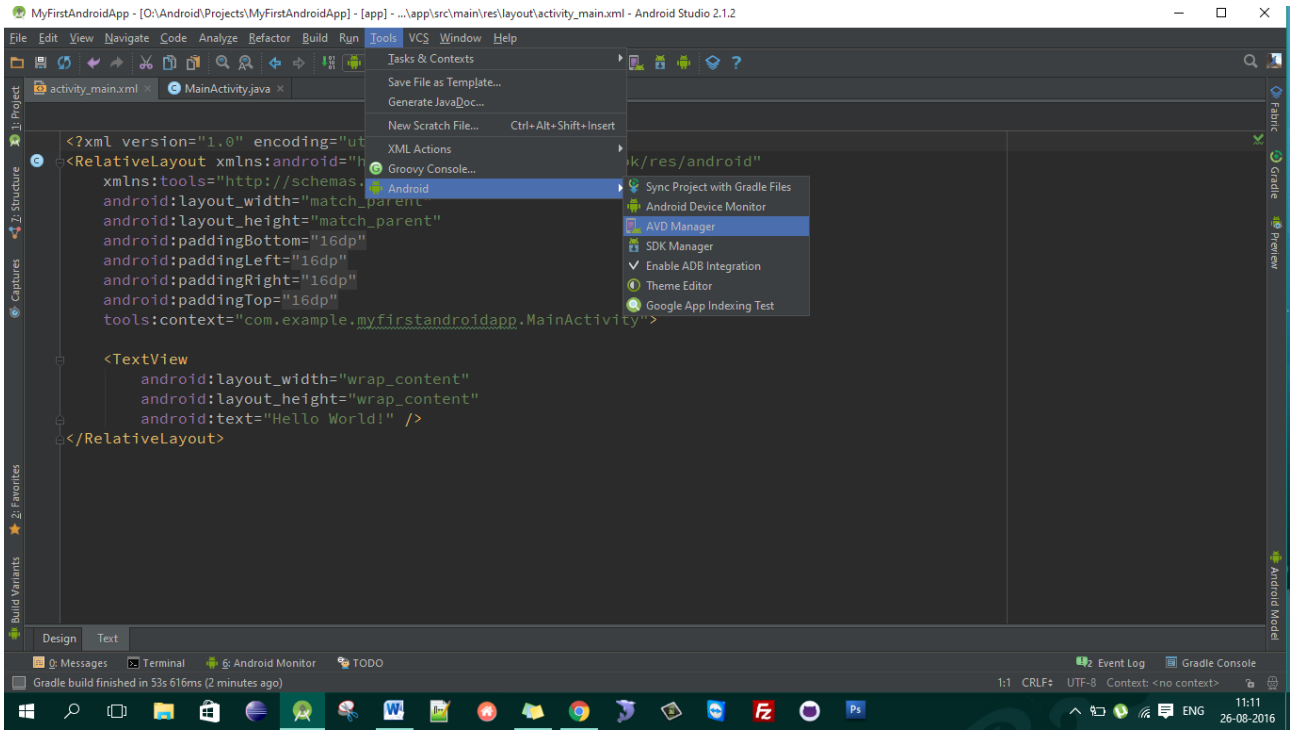

Preferences for New Projects
Appearance & Behavior
Appearance
Menus and Toolbars
System Settings
Passwords
HTTP Proxy
Data Sharing
Date Formats
Updates
Android SDK
Memory Settings
Notifications
Quick Lists
Path Variables
Keymap
Editor
Plugins
Build, Execution, Deployment
Languages & Frameworks
Tools
Appearance & Behavior › System Settings › Android SDK
Manager for the Android SDK and Tools used by Android Studio
Android SDK Location: /Users/bhuvaneswarigokul/Library/Android/sdk
Edit
Optimize disk spac
SDK Platforms
SDK Tools
SDK Update Sites
Below are the available SDK developer tools. Once installed, Android Studio will automatically check for updates. Check "show package details" to display available versions of an SDK Tool.
Name
Version
Status
Android SDK Build-Tools 31-rc3 — Update Available: 31.0.0 rc3
NDK (Side by side) — Not Installed
Android SDK Command-line Tools (latest) — Not Installed
CMake — Not Installed
Android Auto API Simulators — 1 — Not installed
Android Auto Desktop Head Unit Emulator — 2.0.0 rc1 — Not installed
Android Emulator — 29.3.4 — Update Available: 30.5.6
Android SDK Platform-Tools — 29.0.5 — Update Available: 31.0.2
Android SDK Tools — 26.1.1 — Installed
Google Play APK Expansion library — 1 — Installed
Google Play Instant Development SDK — 1.9.0 — Not installed
Google Play Licensing Library — 1 — Installed
Google Play services — 49 — Not installed
Google Web Driver — 2 — Not installed
Intel x86 Emulator Accelerator (HAXM installer) — 7.5.1 — Update Available: 7.6.5
Layout Inspector image server for API 29-30 — 6 — Not installed
Layout Inspector image server for API S — 2 — Not installed
Hide Obsolete Packages
Show Package Detail
Cancel
Apply
OK

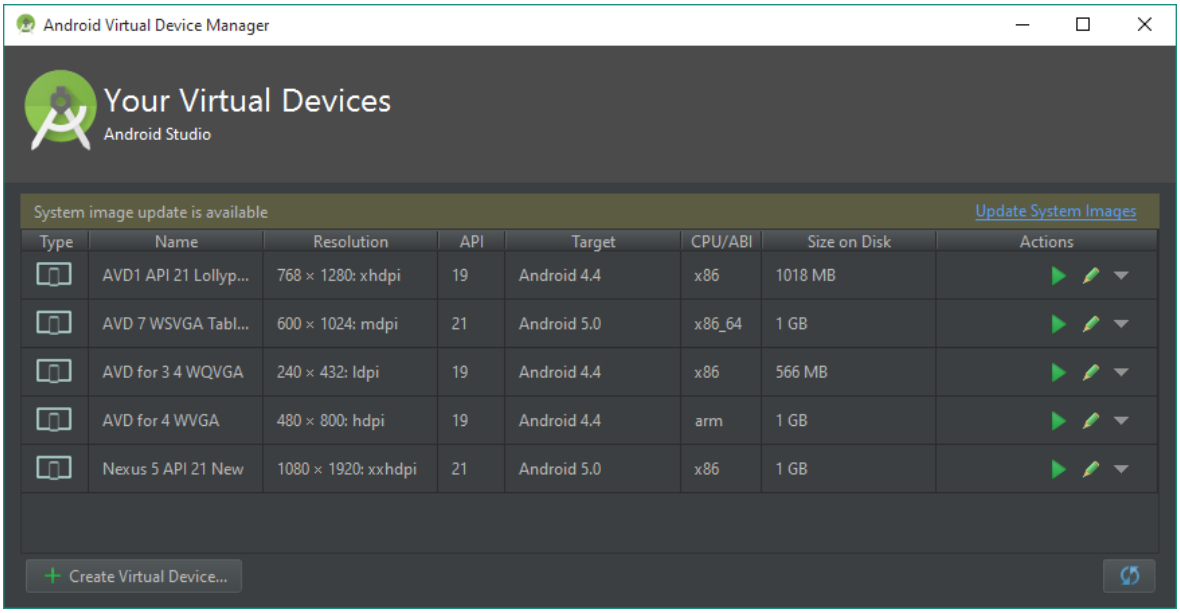

Preferences for New Projects
Appearance & Behavior
Appearance
Menus and Toolbars
System Settings
Passwords
HTTP Proxy
Data Sharing
Date Formats
Updates
Android SDK
Memory Settings
Notifications
Quick Lists
Path Variables
Keymap
Editor
Plugins
Build, Execution, Deployment
Languages & Frameworks
Tools
Appearance & Behavior › System Settings › Android SDK
Manager for the Android SDK and Tools used by Android Studio
Android SDK Location: /Users/bhuvaneswarigokul/Library/Android/sdk
Edit
Optimize disk spac
SDK Platforms
SDK Tools
SDK Update Sites
Each Android SDK Platform package includes the Android platform and sources pertaining to an API level by default. Once installed, Android Studio will automatically check for updates. Check "show package details" to display individual SDK components.
Name
API Level
Revision
Status
Android S Preview — S — 3 — Not installed
Android 11.0 (R) — 30 — 3 — Not installed
Android 10.0 (Q) — 29 — 5 — Partially installed
Android 9.0 (Pie) — 28 — 6 — Not installed
Android 8.1 (Oreo) — 27 — 3 — Update available
Android 8.0 (Oreo) — 26 — 2 — Installed
Android 7.1.1 (Nougat) — 25 — 3 — Installed
Android 7.0 (Nougat) — 24 — 2 — Installed
Android 6.0 (Marshmallow) — 23 — 3 — Update available
Android 5.1 (Lollipop) — 22 — 2 — Not installed
Android 5.0 (Lollipop) — 21 — 2 — Not installed
Android 4.4W (KitKat Wear) — 20 — 2 — Not installed
Android 4.4 (KitKat) — 19 — 4 — Not installed
Android 4.3 (Jelly Bean) — 18 — 3 — Not installed
Android 4.2 (Jelly Bean) — 17 — 3 — Not installed
Android 4.1 (Jelly Bean) — 16 — 5 — Not installed
Android 4.0.3 (IceCreamSandwich) — 15 — 5 — Not installed
Android 4.0 (IceCreamSandwich) — 14 — 4 — Not installed
Android 3.2 (Honeycomb) — 13 — 1 — Not installed
Android 3.1 (Honeycomb) — 12 — 3 — Not installed
Hide Obsolete Packages
Show Package Detail
Cancel
Apply
OK

Creating Android Virtual Device with AVD Manager

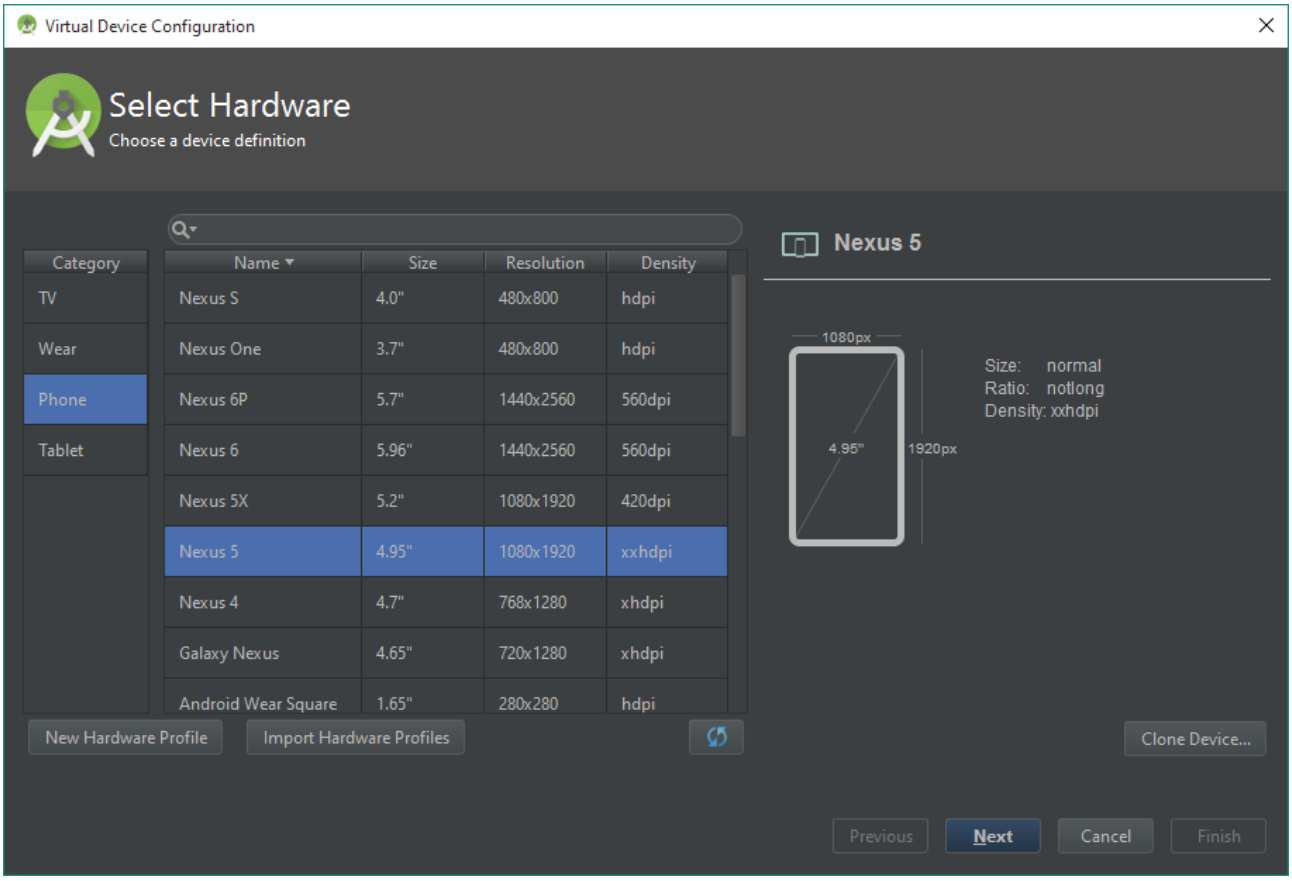
To open AVD manager, go to **Tools → Android → AVD Manager**

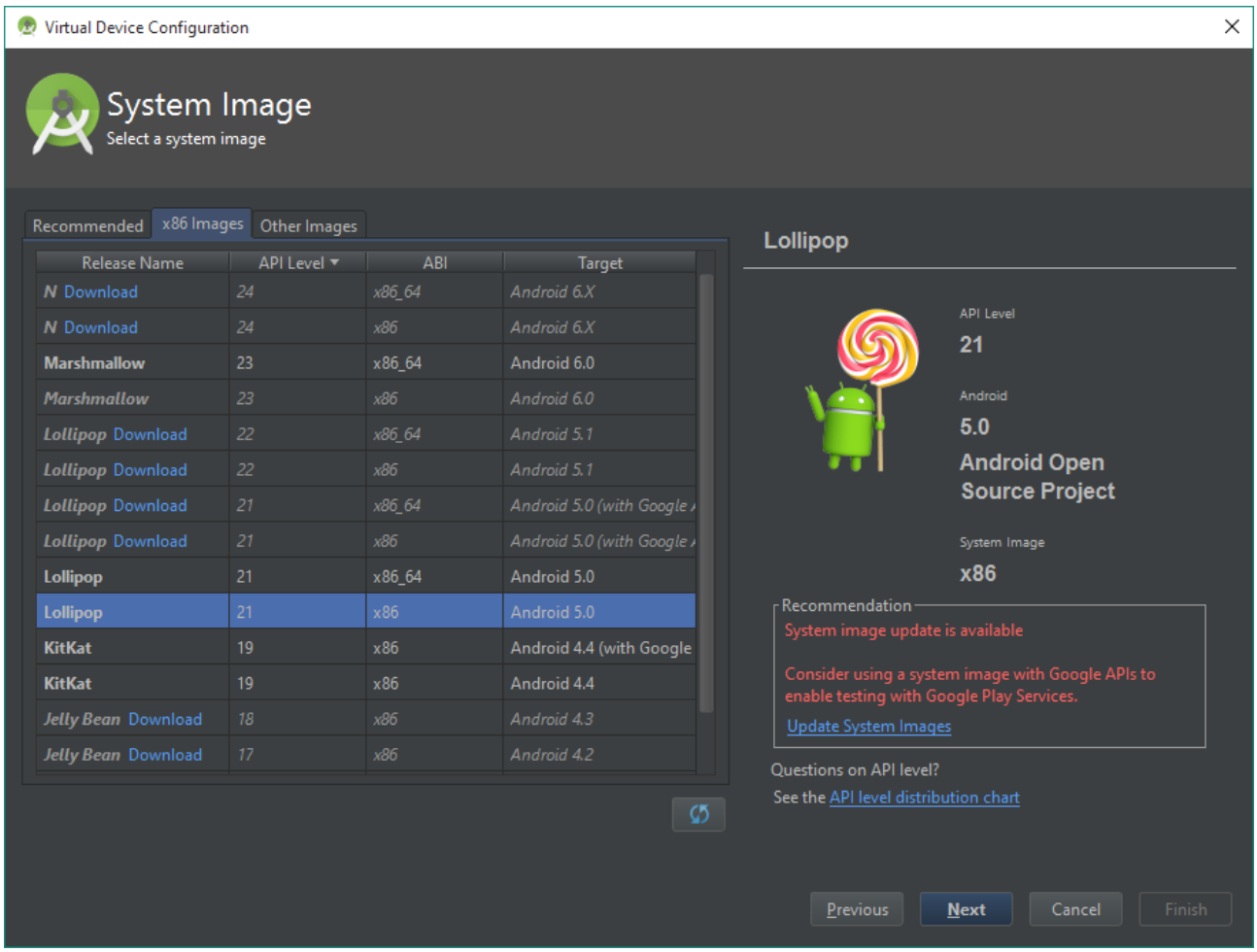
To create a new device, click on **Create Virtual Device** button at the bottom-left corner.

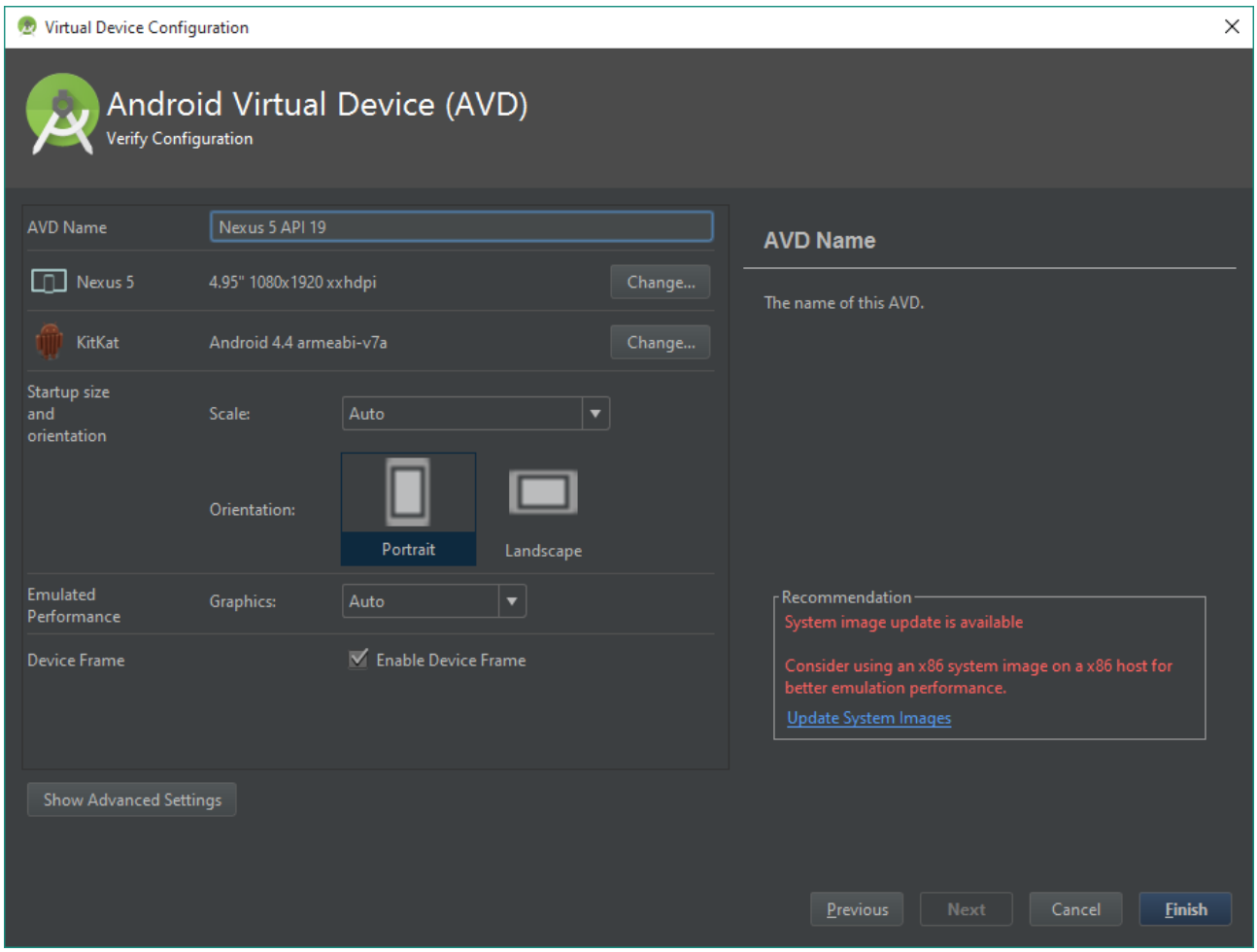


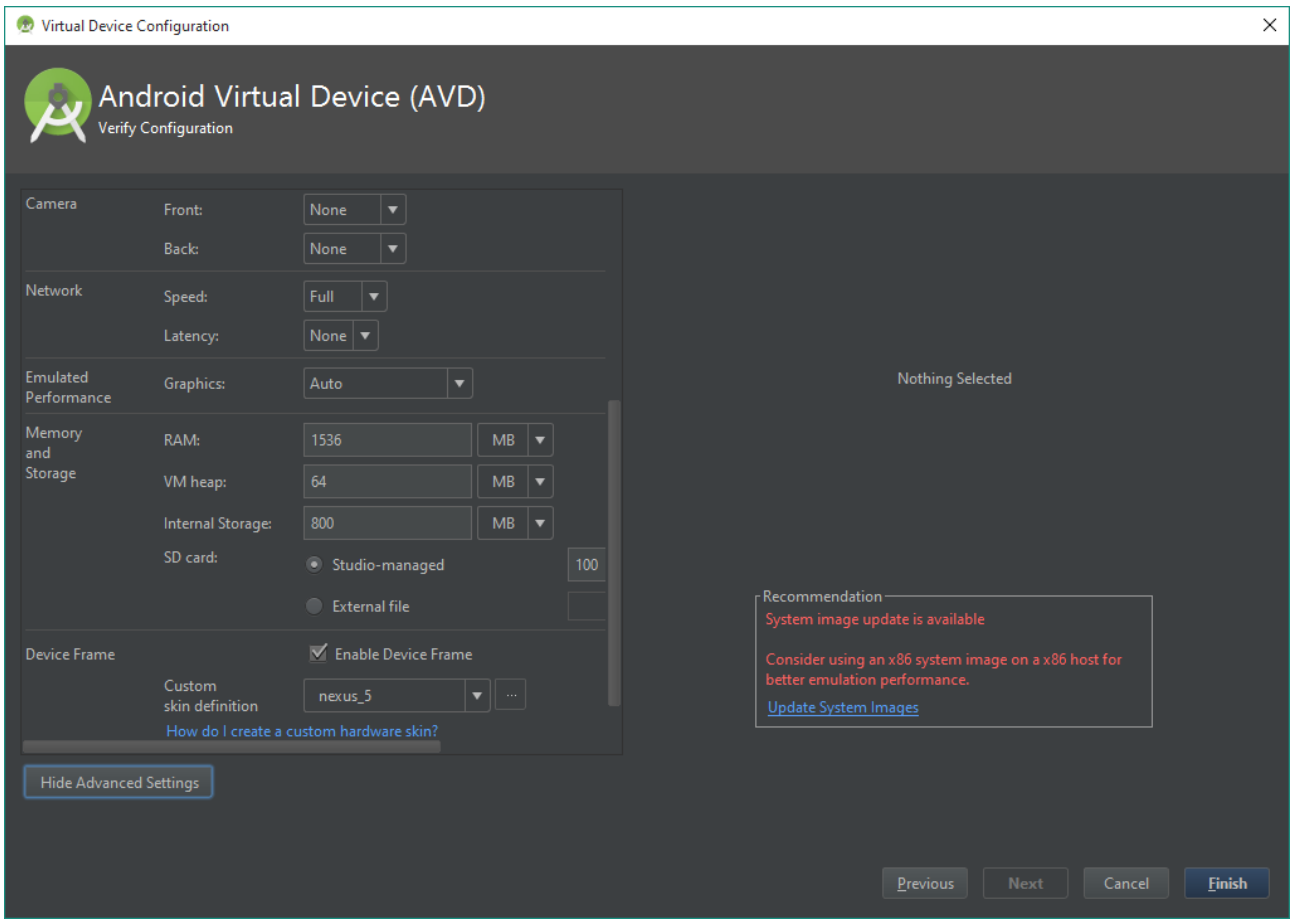


It will open a window to **Select Hardware** type for your virtual device. This list contains almost all the Android devices with their respective settings. Select any one out of all the devices listed, with your required configuration (like Size of the screen, Resolution and Density) and click on **Next**.



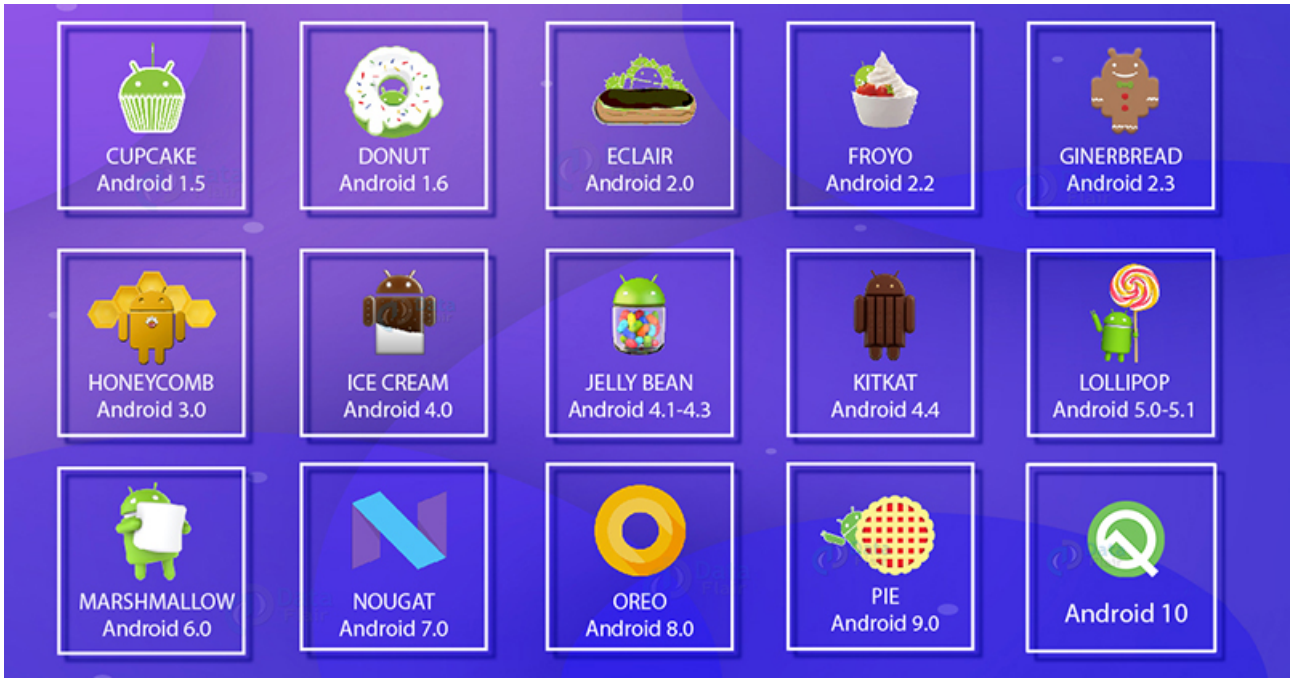

Virtual Device Configuration
System Image
Select a system image
Recommended
x86 Images
Other Images
Release Name
API Level
ABI
Target
N Download 24 x86_64 Android 6.X
N Download 24 x86 Android 6.X
Marshmallow 23 x86_64 Android 6.0
Marshmallow 23 x86 Android 6.0
Lollipop Download 22 x86_64 Android 5.1
Lollipop Download 22 x86 Android 5.1
Lollipop Download 21 x86_64 Android 5.0 (with Google
Lollipop Download 21 x86 Android 5.0 (with Google
Lollipop 21 x86_64 Android 5.0
Lollipop 21 x86 Android 5.0
KitKat 19 x86 Android 4.4 (with Google
KitKat 19 x86 Android 4.4
Jelly Bean Download 18 x86 Android 4.3
Jelly Bean Download 17 x86 Android 4.2
Lollipop
API Level
21
Android
5.0
Android Open Source Project
System Image
x86
Recommendation
System image update is available
Consider using a system image with Google APIs to enable testing with Google Play Services.
Update System Images
Questions on API level?
See the API level distribution chart
Previous
Next
Cancel
Finish


Virtual Device Configuration
Android Virtual Device (AVD)
Verify Configuration
AVD Name
Nexus 5 API 19
Nexus 5
4.95" 1080x1920 xxhdpi
Change...
KitKat
Android 4.4 armeabi-v7a
Change...
Startup size and orientation
Scale:
Auto
Orientation:
Portrait
Landscape
Emulated Performance
Graphics:
Auto
Device Frame
Enable Device Frame
Show Advanced Settings
AVD Name
The name of this AVD.
Recommendation
System image update is available
Consider using an x86 system image on a x86 host for better emulation performance.
Update System Images
Previous
Next
Cancel
Finish



History of Android Technology

After the introduction to Android, we will discuss it's history part. It is an operating system developed by Android Inc.

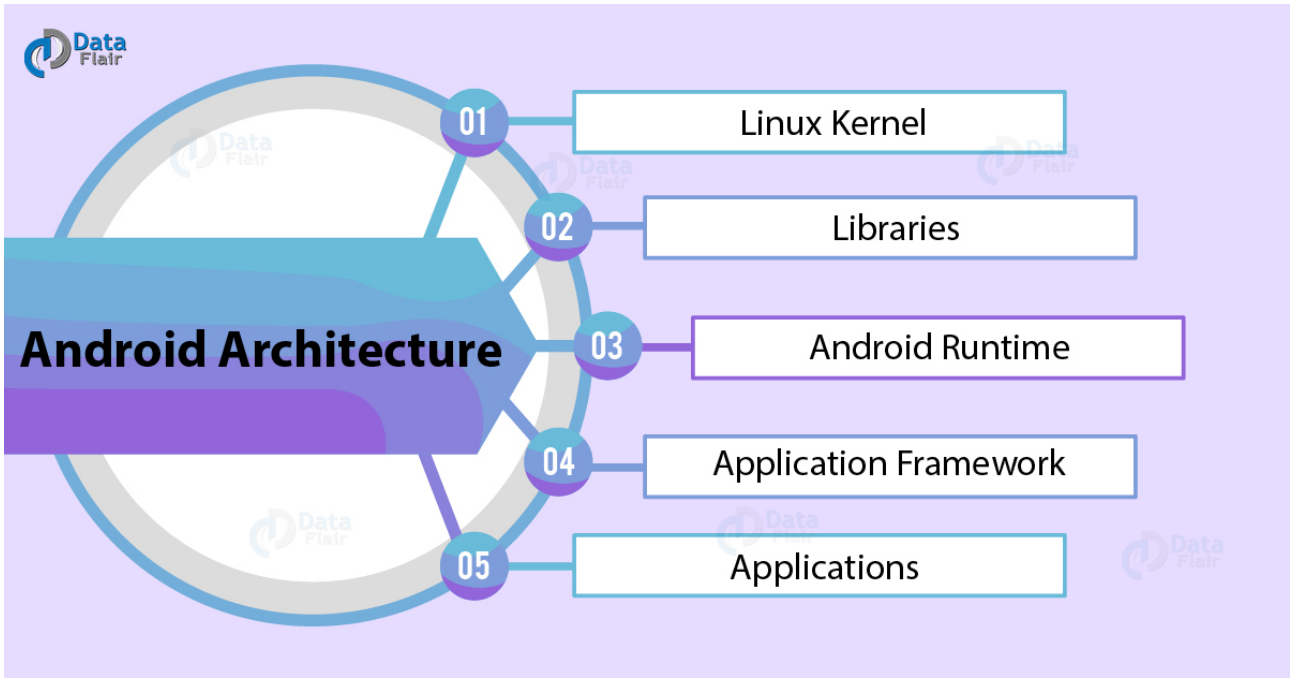


and then overtaken by Google. Android Inc. was developed in Palo Alto California, in October 2003 by Andy Rubin, Rich Miner, Nick Sears, and Chris White.
In 2005, Google acquired Android Inc, in 2007. The first version was released by Google and the commercial version was released in 2008 known as Android 1.0.

- The first version of Android 1.0, released on 23 September 2003.
- The second version, released on 9 Feb 2009.
- Then version 1.5 known as Cupcake, released on 27 April 2009.
- Version 1.6 known as Donut, released on 15 Sept 2009.
- Then on 26 Oct 2009 version 2.0-2.1 was released.
- Froyo version 2.2-2.2.3, released on 20 May 2010.
- On 6 Dec 2010 gingerbread version 2.3-2.3.7 was released.
- Honeycomb version 3.0-3.2.6, released on 22 Feb 2011.
- Ice cream version 4.0-4.0.4, released on 18 Oct 2011.
- Jelly bean version 4.1-4.3.1, released on 9 July 2012.
- Kit-kat version 4.4-4.4.4, released on 31 Oct 2013.
- Lollipop version 5.0-5.1..1, released on 12 Nov 2014.
- Marshmallow 5 version 6.0-6.0.1, released on Oct 2015.
- Nougat version 7.0-7.1.2, released on 22 Aug 2016.
- Oreo version 8.0-8.1, released on 21 Aug 2017.
- Version 9 also known as Pie, released on 6 Aug 2018.
- Version 10 also known as Android 10, released on 3 September 2019.

i. Linux Kernel

Linux kernel is the bottom-most and important layer of the Android architecture and it is the core part of Android architecture.



It provides features such as:

- **Security**
- **Process management**
- **Memory management**
- **Device management**
- **Multitasking**

Android Runtime

It comprises of DVM (Dalvik Virtual Machine). Just like **JAVA uses JVM**, Android uses DVM to optimize battery life, memory and performance. The byte code generated by the **Java compiler** has to be converted to .dex file by DVM, as it has its own byte code

Application Framework

Android operating system are available to us through API's written in form of **JAVA classes**.
It also consists of an Android **Hardware Abstraction Layer (HAL)** that allows the Android Application

framework to communicate with hardware-specific device drivers.

The application framework consists of following key services:

- **Activity Manager:** The method in this class uses testing and debugging methods.
- **Content provider:** It provides data from application to other layers.
- **Resource Manager:** It provides access to non-code resources.
- **Notification Manager:** The users get notification about all the actions happening in the background.
- **View System:** It acts as a base class for widgets and is responsible for event handling.

Applications

It is the top-most layer of Android architecture. This layer consists of native Android applications and third-party installed apps. They are bundled in an Android package and all the applications that are to be installed are written in this layer only such as contacts, games, settings, and messages.

Keyboard shortcuts

Here are 10 shortcuts to help you through your Android Application Development:

- Ctrl + F – It will help you find words in your file.
- Ctrl + R – It will help you replace a word/ name.
- F5 – It will help you to copy.
- Ctrl+ (+/-) – It will let you zoom in or out.
- F11 – It will get the Android Studio in full-screen mode.
- Ctrl + B – It will open the XML file.
- Ctrl + N – It will help you find a class.

- Shift + F6 – It will help you to rename a class name or variable name.
- Ctrl + Shift+ N – It will help you find a file.
- Ctrl + I – It will help to implement methods.

Some other Tips

- You can change the theme as per your convenience as there is a light and a dark theme to use Android Studio.
- You can make a very fast search if you need to find the files or classes or symbols. It is very simple. You can simply press the Shift key two times, and there you go; you can search it.
- While writing the code, you would be given suggestions, pressing the tab key will type that automatically.
- You can also check the CPU, memory, and network usage of the applications in Android Studio. This can be done through the profile check feature in it.
- Also, if there is some issue with the name of the class or variable or method, you can easily change it too. You just need to right click and choose the refactor and then rename. And you are done.

Android Activity Lifecycle