Department of Computer Technology and Information Systems

## CTIS 487 - Mobile Application Development

FALL 2019 - 2020

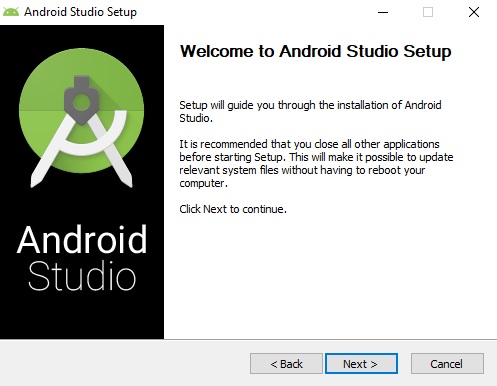
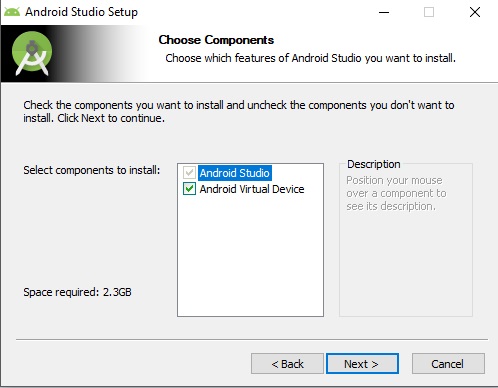
# **Lab Guide #1**

|  |
| --- |
| **OBJECTIVES :** Android Development Environment Installation & Creating First Android Application |
| **Instructor :** Neşe ŞAHİN ÖZÇELİK  **Assistant :** Leyla SEZER |

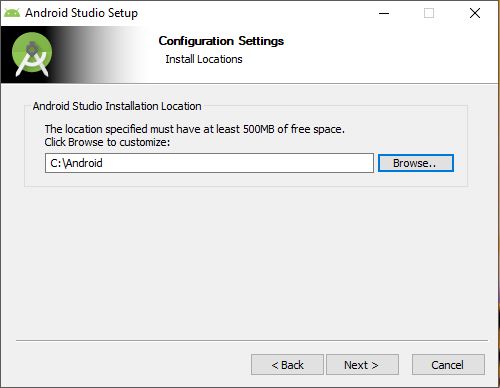
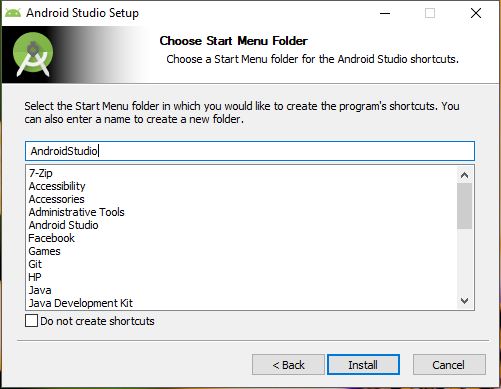
* **Install Java SE Development Kit 10 or upper versions from the below link**

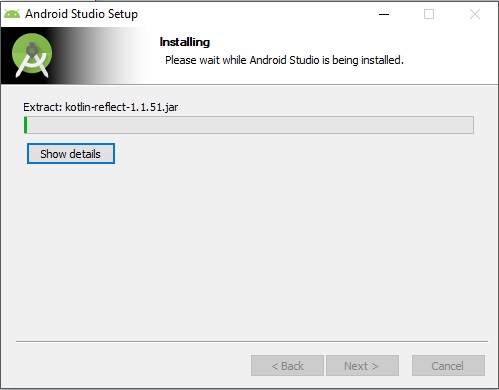
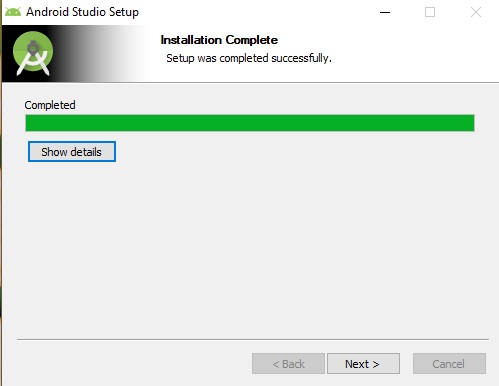
http://www.oracle.com/technetwork/java/javase/downloads/jdk10-downloads-4416644.html

* **Steps to install Android Studio 3.5 and APIs**

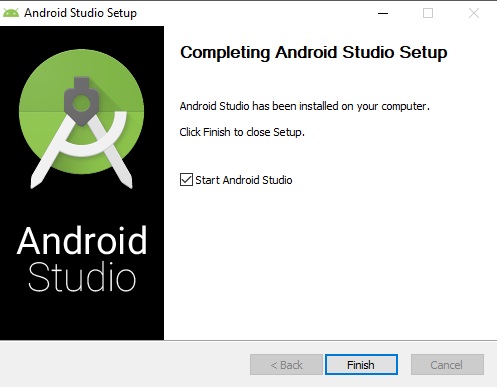


\*\* Specify the location of the installation ( It should NOT include blank character in the path )

****



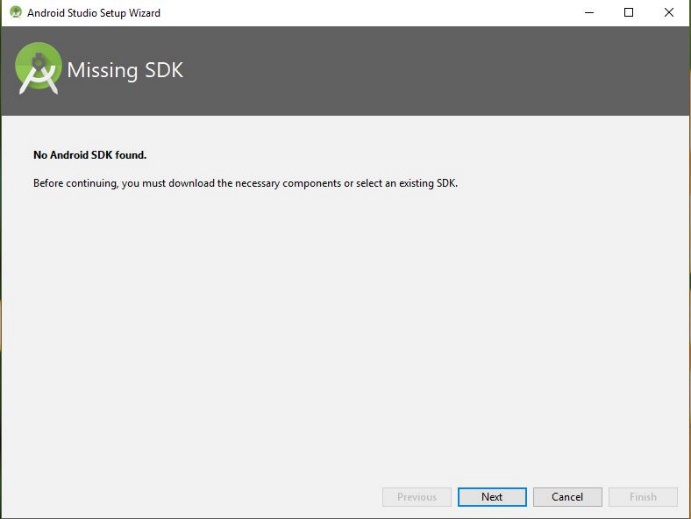
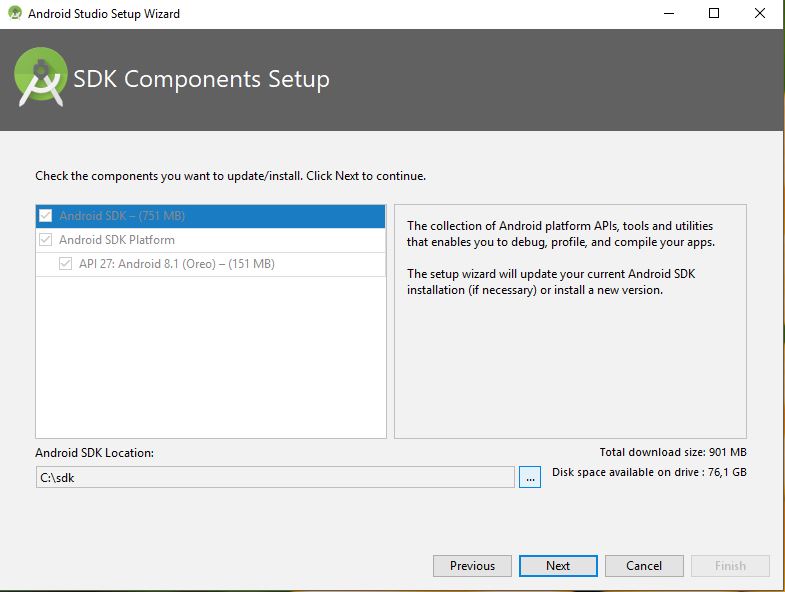


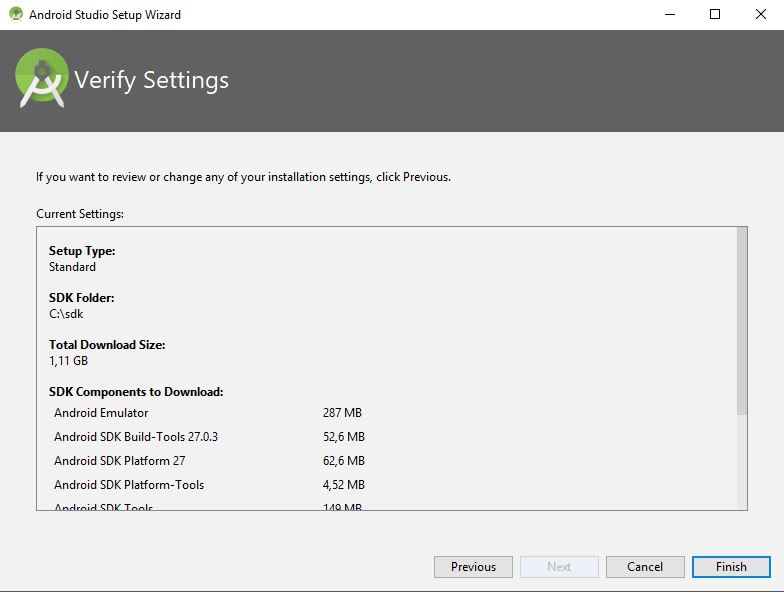
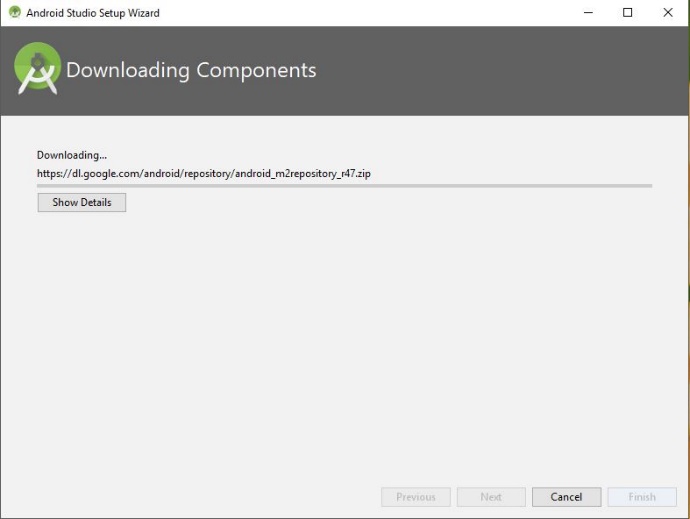


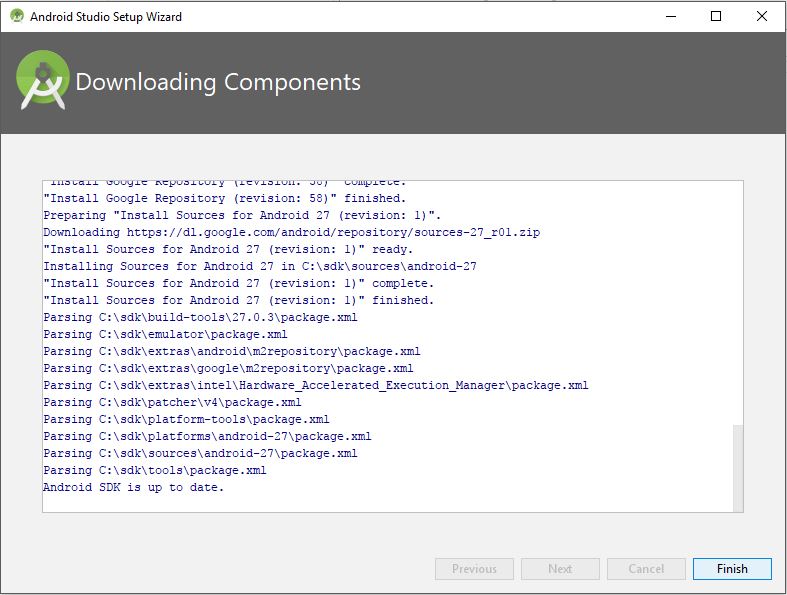
* **To install HAXM installer, be sure Virtualization Technology (VTx) is enabled in the BIOS Menu**



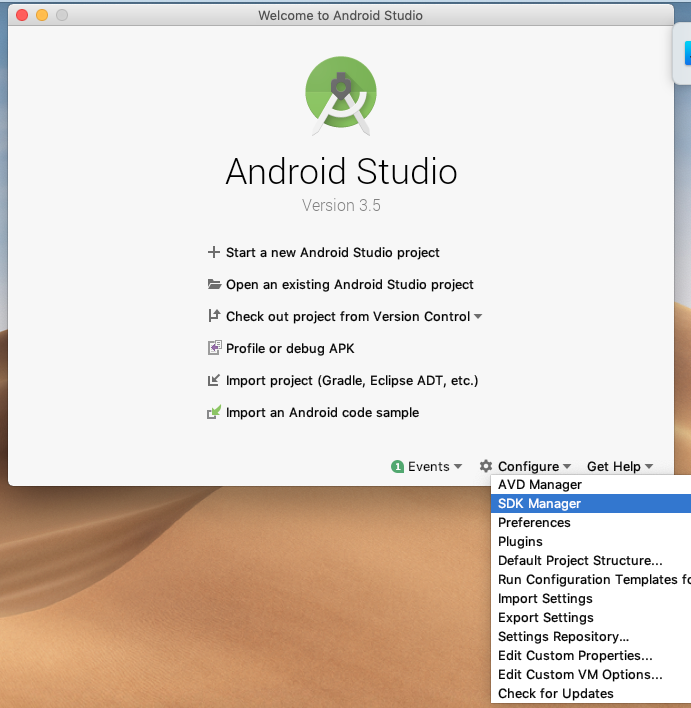
* **Steps to install SDK Manager**

****

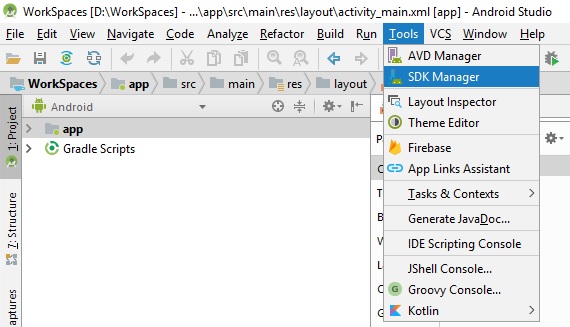
****

****

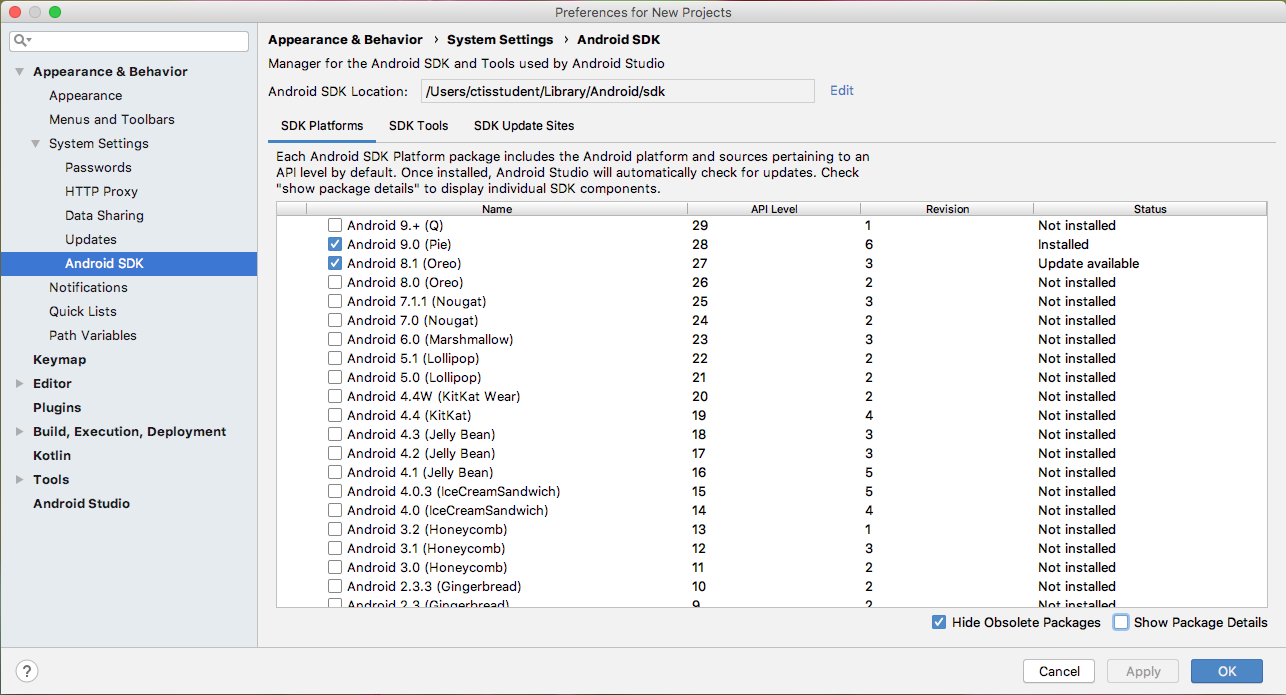
* **Steps to install APIs**
* First Way to open SDK Manager



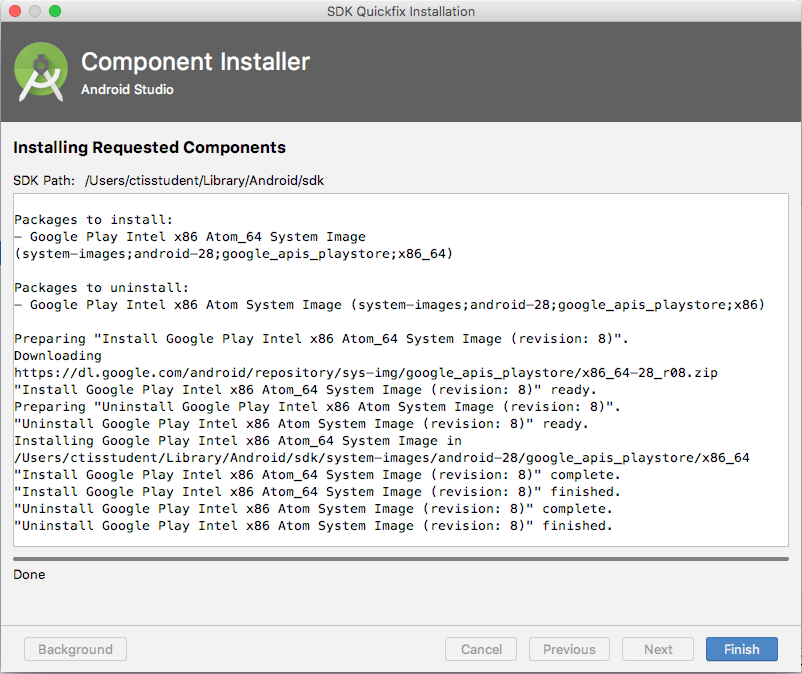
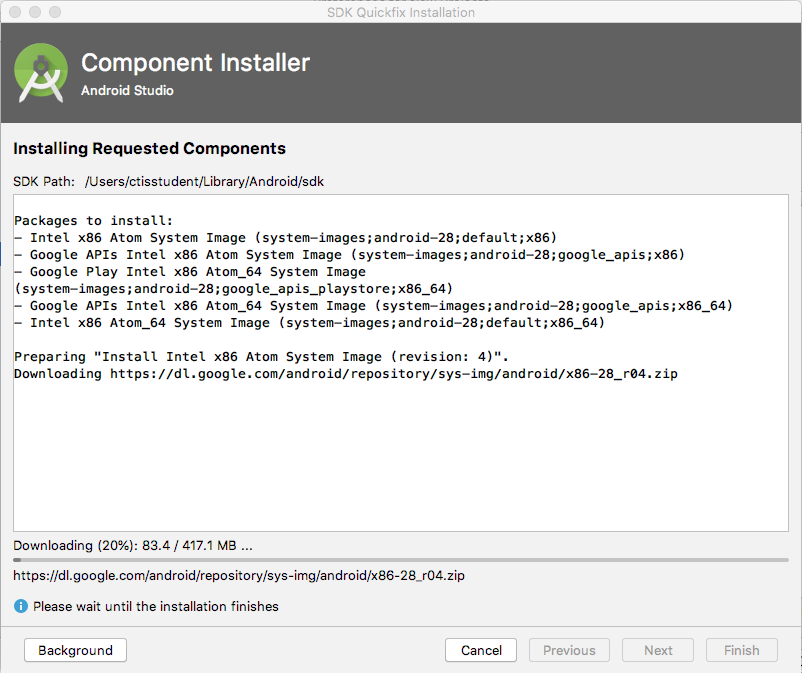
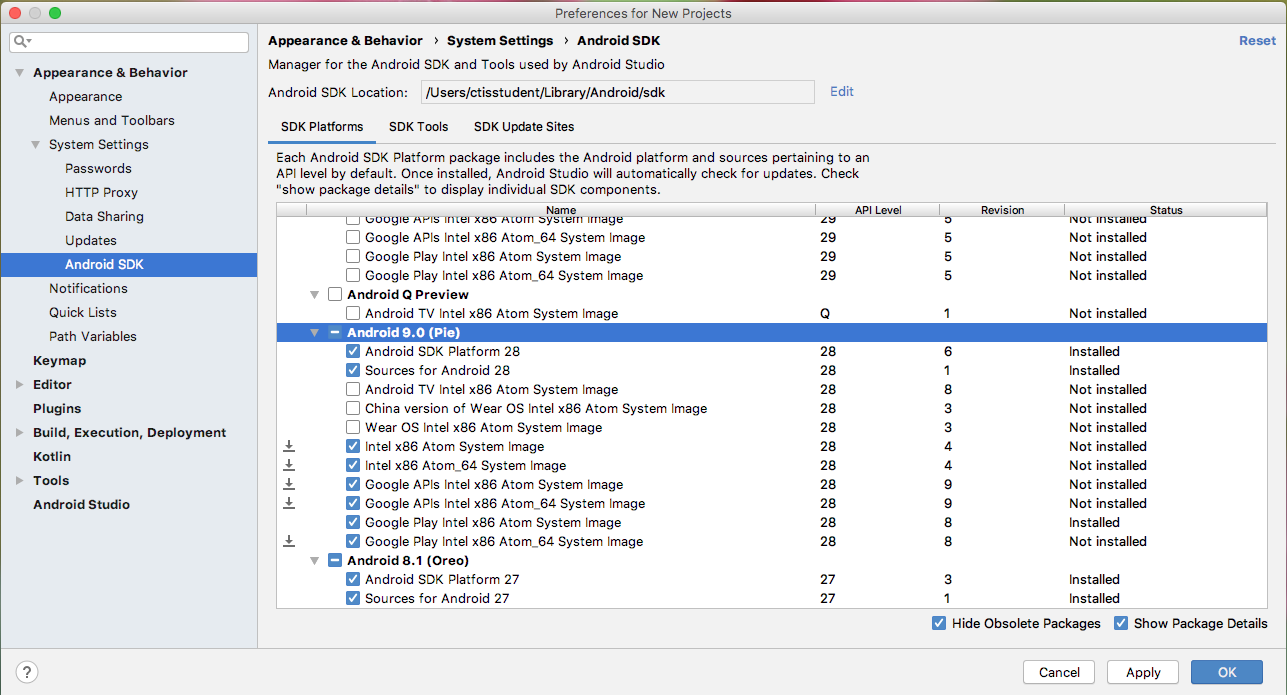
* Second Way to open SDK Manager, select SDK Manager from the Tools Menu like in the below



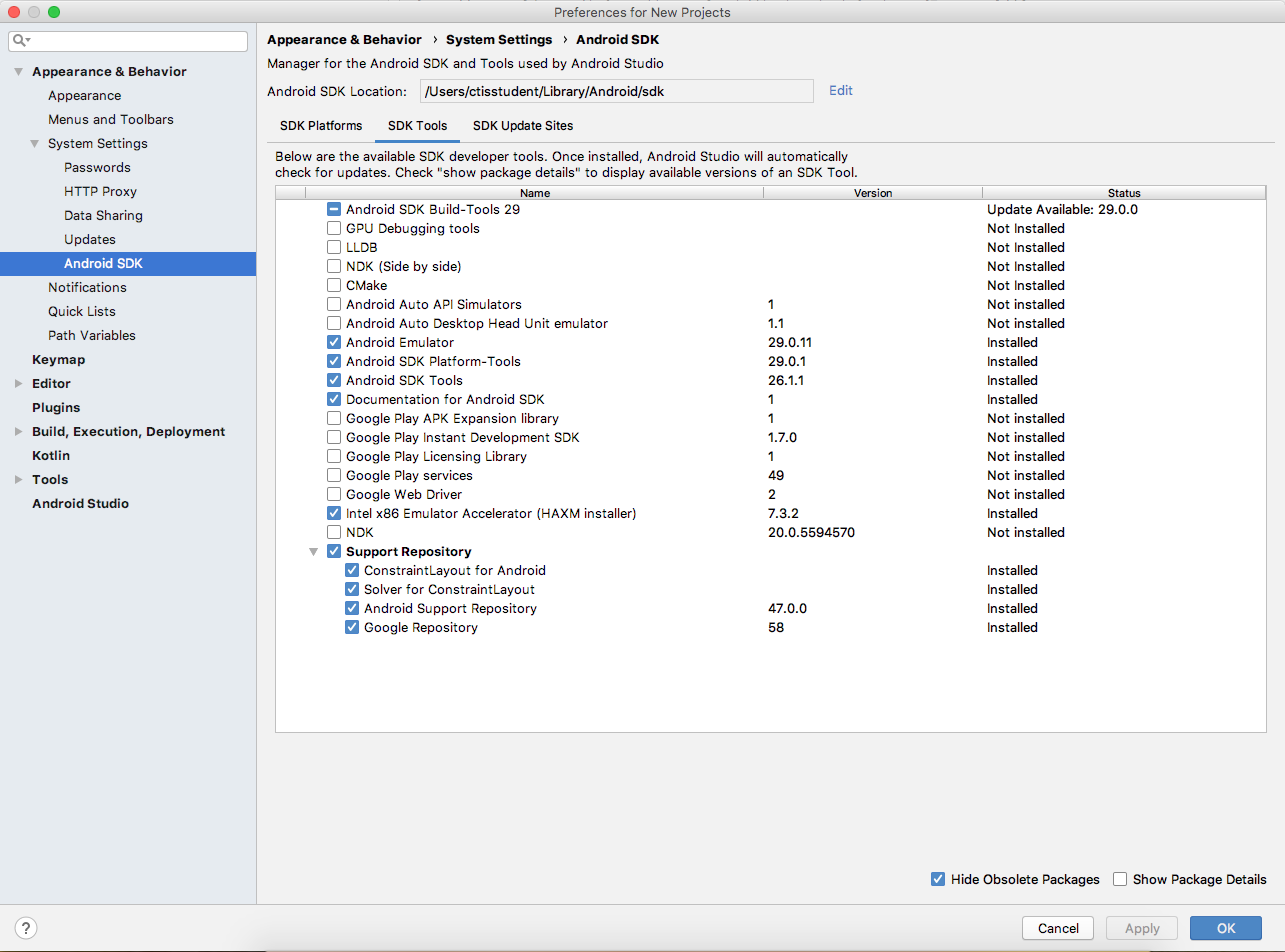
* Select Android 9.0 ( API 28 ) to install



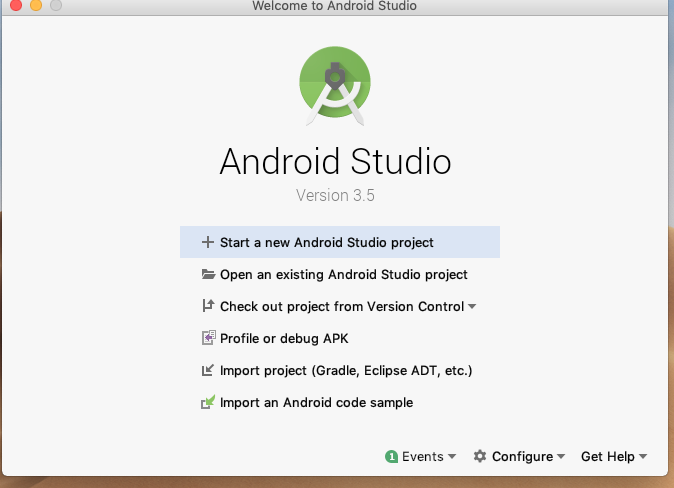
* Select Show Package Details check box
* Select the options without TV and Wear one like in the below



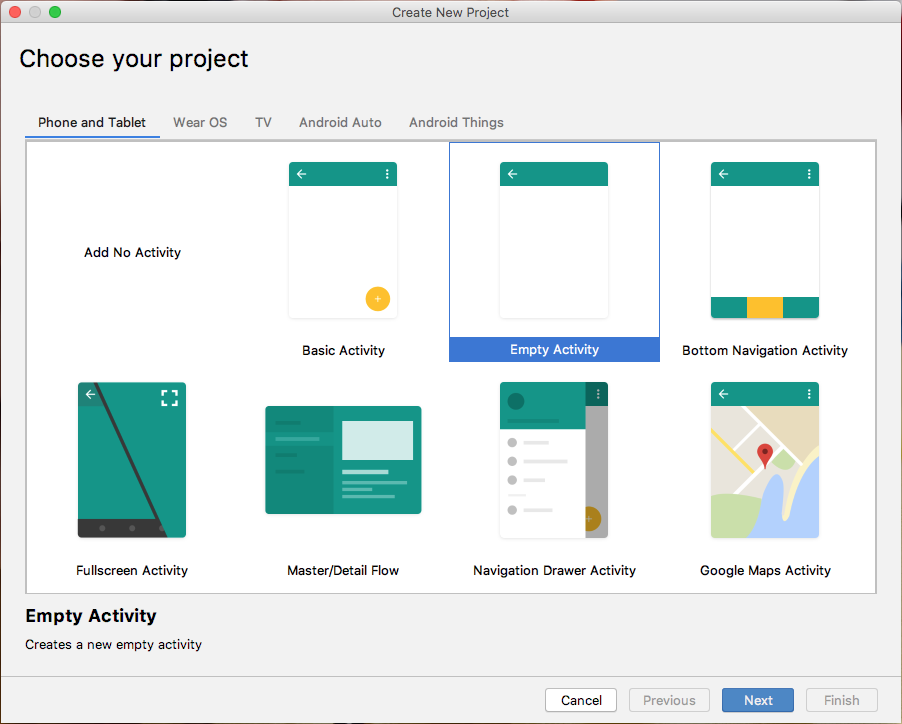
* It starts to install APIs
* Install SDK Build Tools belongings to API 28 in the SDK Tools like in the below
* If HAXM Installer is not installed, install it



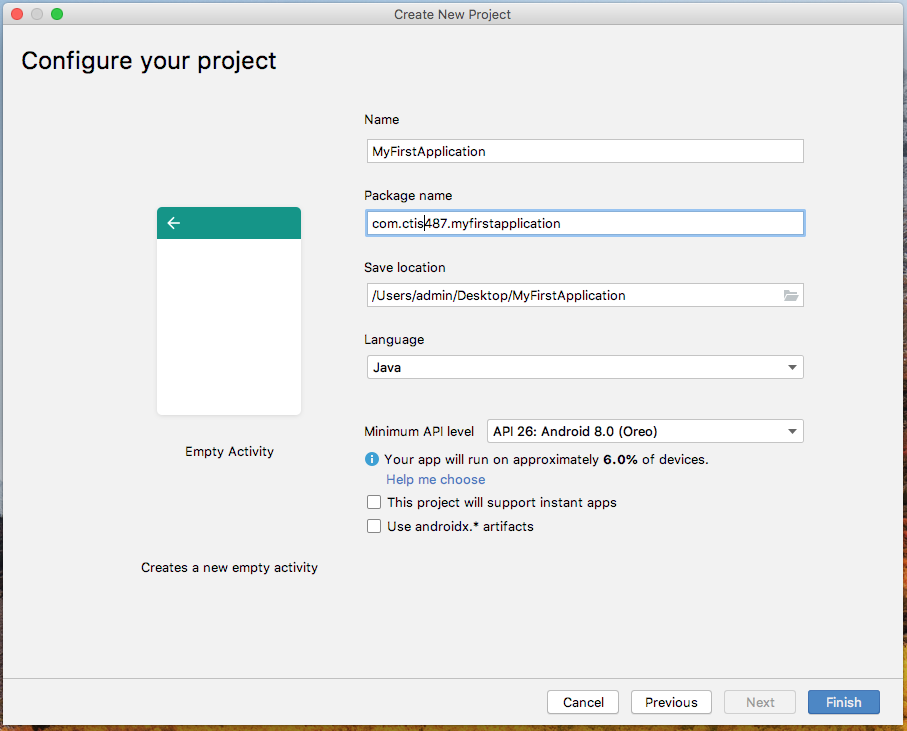
* **Creating a new project**
* Select “Start a new Android Studio project” option



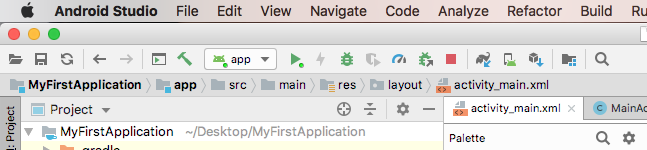
* Select “Empty Activity”

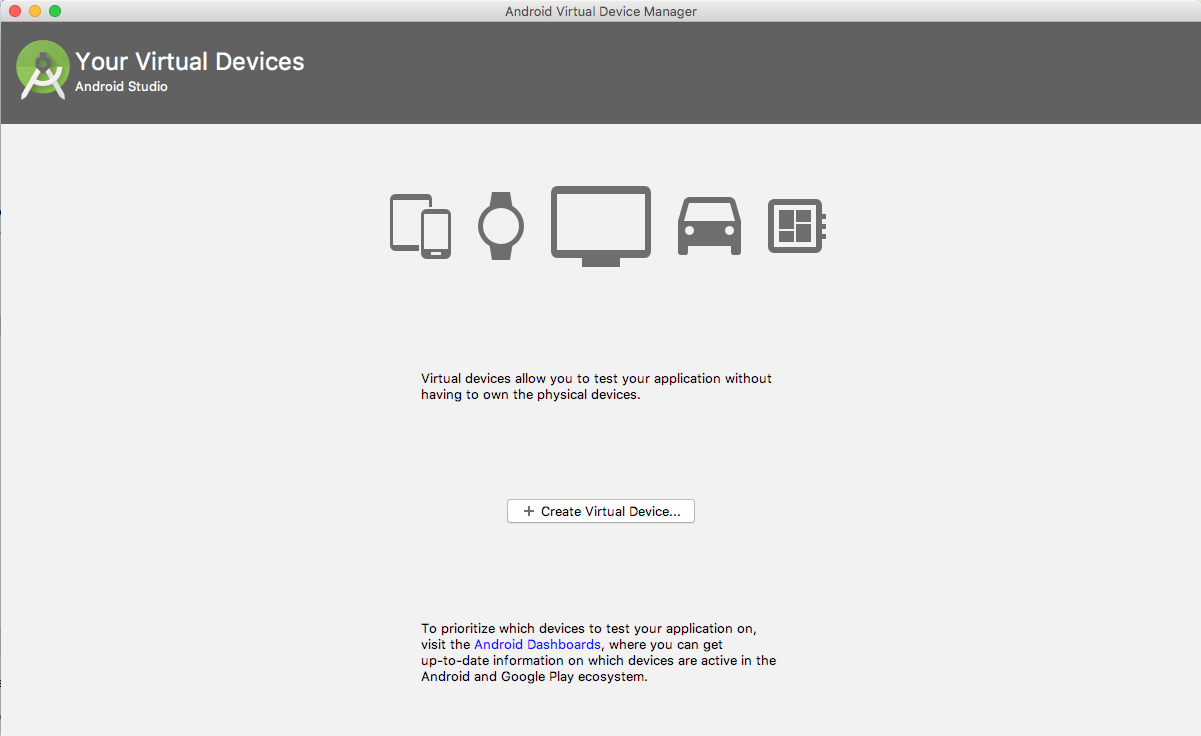


* Write a Project Name, Select API 26 and select language to JAVA and click Finish button.

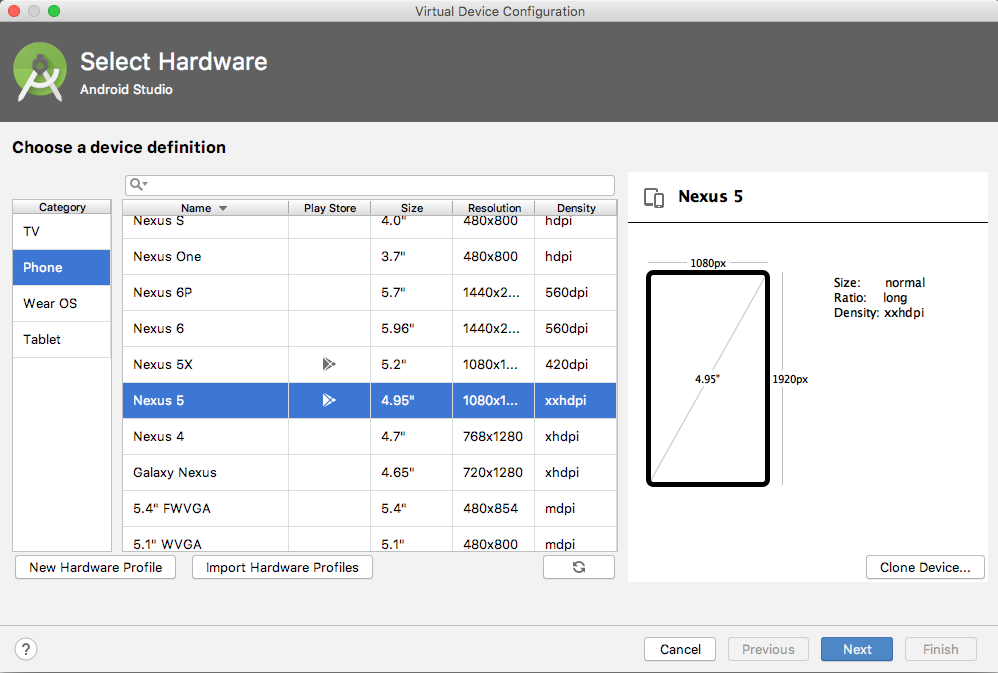


* **Creating a new virtual device**

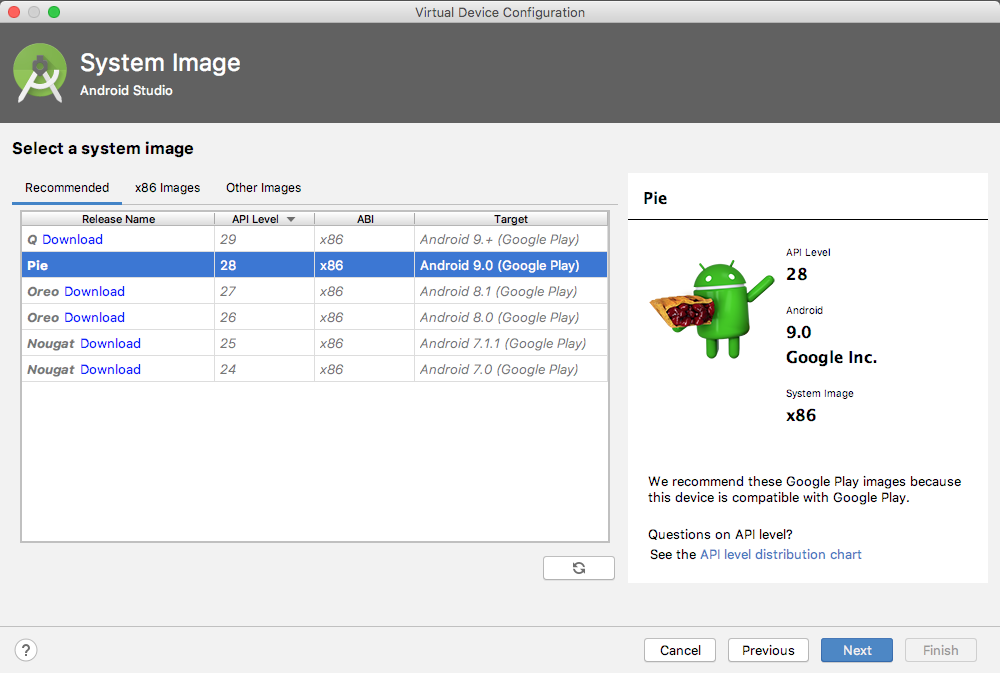




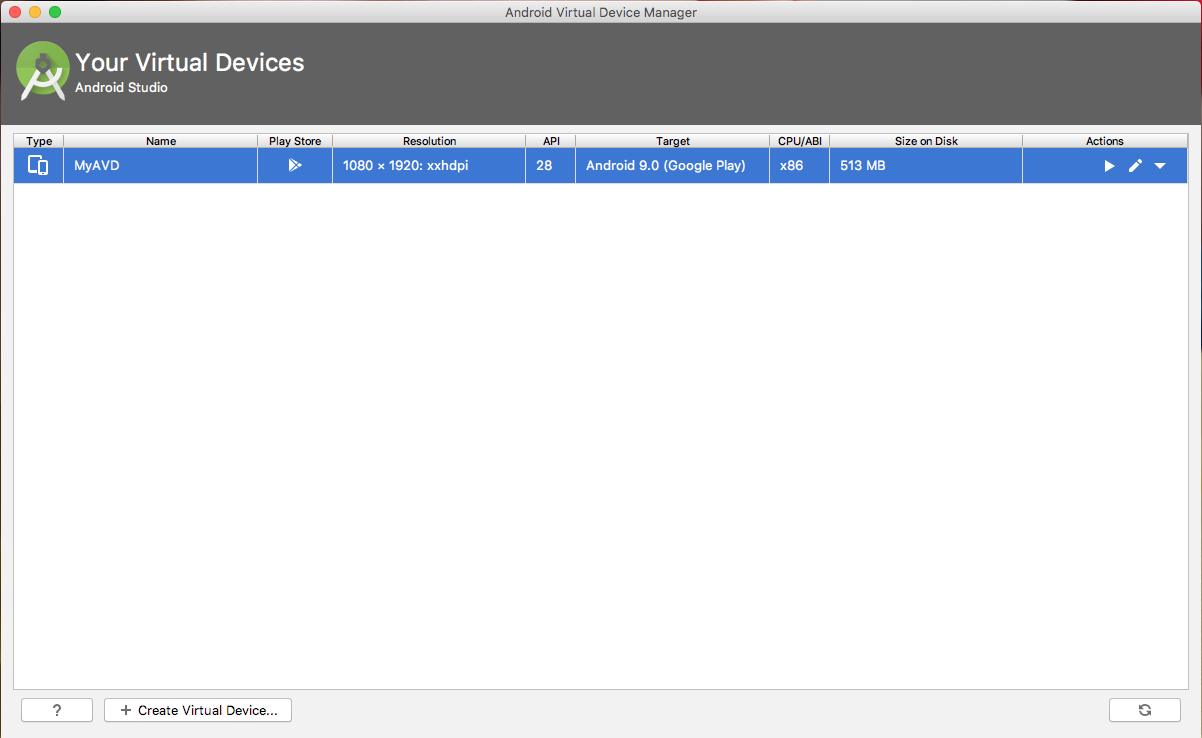
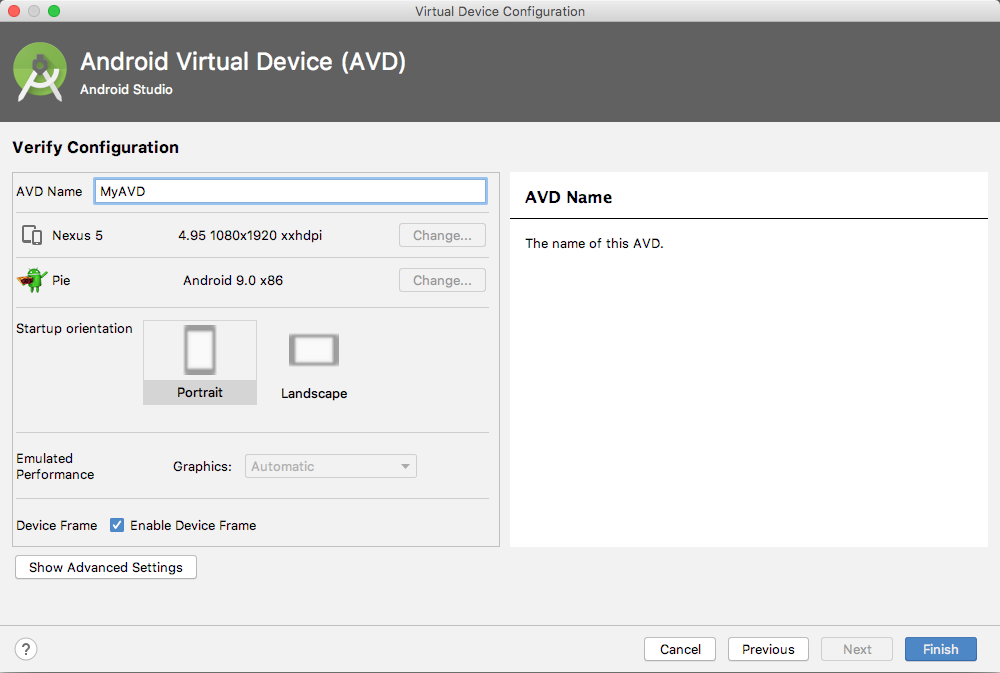
* Select Nexus 5 from the list and click next button



* Select a system image with the API Level 28



* Write a name for Android Virtual Device ( AVD ) and click finish button

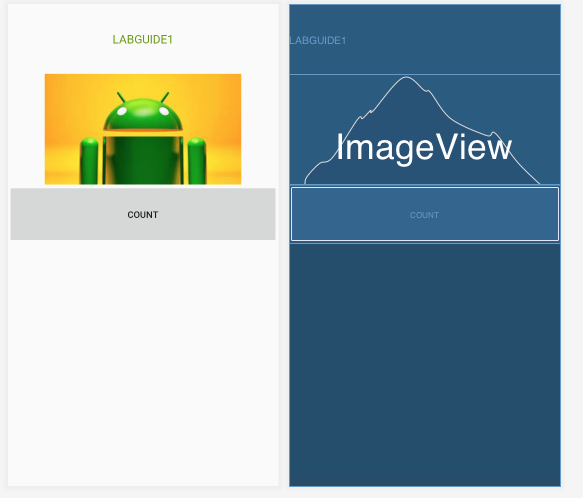


* You can use Genymotion for creating Virtual Device easily by the following url; <https://www.genymotion.com/fun-zone/>

**Q1.** Create the following layout, bu using the Android Components;

* Use Vertical LinearLayout
* Textview; text is “**LABGUIDE1**” give any color which you want,
* ImageView; put the **android.jpg** to the drawable folder
* Button; text is “**COUNT**”

Check the activity\_main.xml file content.



**activity\_main.xml**

<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"

xmlns:app="http://schemas.android.com/apk/res-auto"

xmlns:tools="http://schemas.android.com/tools"

android:layout\_width="match\_parent"

android:layout\_height="match\_parent"

android:orientation="vertical"

tools:context=".MainActivity">

<TextView

android:id="@+id/txtLab1"

android:layout\_width="match\_parent"

android:layout\_height="106dp"

android:gravity="center"

android:text="LABGUIDE1"

android:textColor="@android:color/holo\_green\_dark"

android:textSize="18sp" />

<ImageView

android:id="@+id/imgAndroid"

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

app:srcCompat="@drawable/android" />

<Button

android:id="@+id/btnCOUNT"

android:layout\_width="match\_parent"

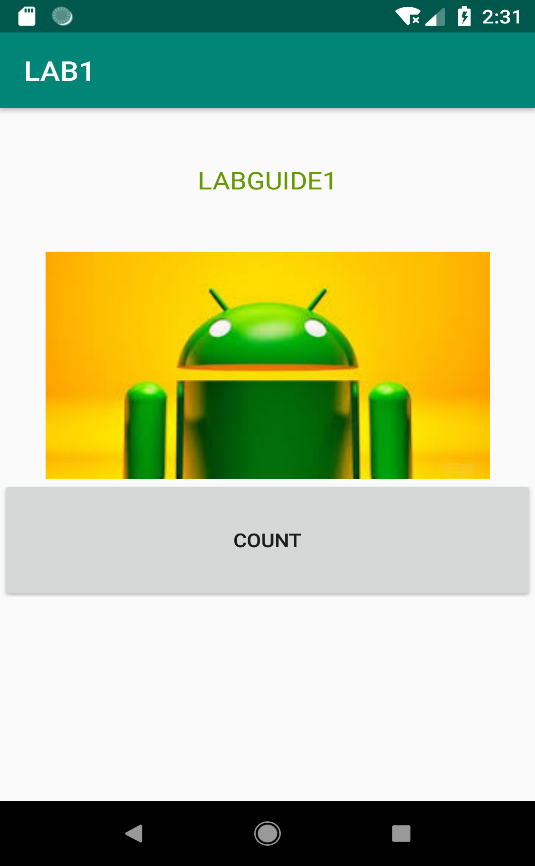
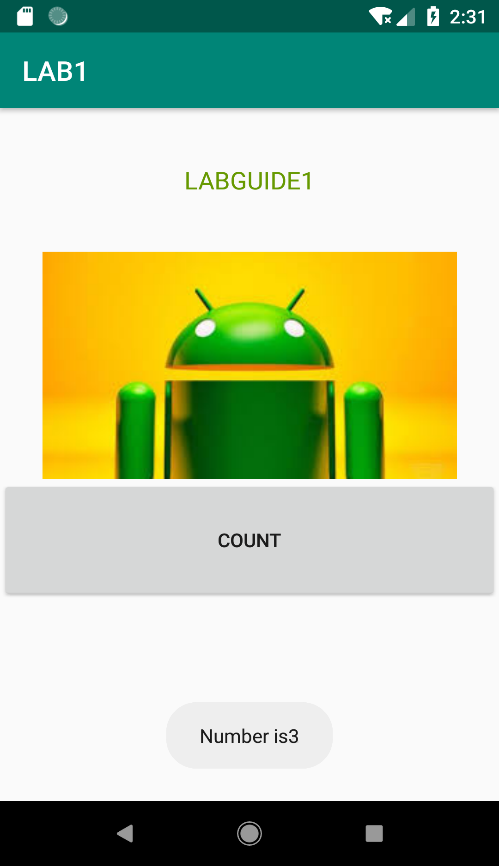
android:layout\_height="90dp"

**android:onClick="onClick"**

android:text="COUNT" />

</LinearLayout>

In the main part, when the user clicks on the **COUNT** button Toast Messeage will be shown and count operation will be held on.

**MainActivity.java**

public class MainActivity extends AppCompatActivity {

int number=0;

@Override

protected void onCreate(Bundle savedInstanceState) {

super.onCreate(savedInstanceState);

setContentView(R.layout.activity\_main);

}

**public void onClick(View view) {**

**number++;**

**Toast.makeText(this,"Number is"+number,Toast.LENGTH\_LONG).show();**

**}**

}