**Lab Manual for Application for Mobile Devices**

# 

**Lab No. 1**

# INTRODUCTION TO ANDROID STUDIO

*Objectives*

*The purpose of this lab is to familiarize with the basics of*

**LAB # 01**

**INTRODUCTION TO ANDROID STUDIO**

## **Introduction**

Android Studio is the official integrated development environment for Google's Android operating system, built on JetBrains' IntelliJ IDEA software and designed specifically for Android development. Wikipedia

Operating system: Windows, macOS, Linux

Stable release: 3.0.1 / November 20, 2017; 2 months ago

Size: 683 MB compressed

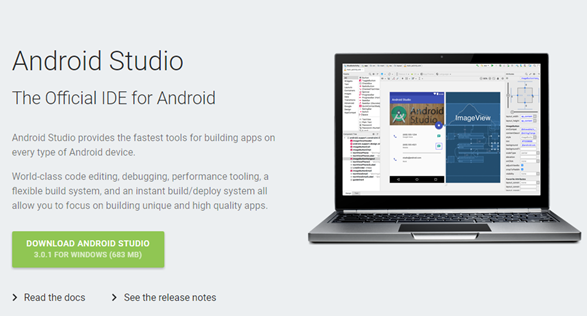
Website <https://developer.android.com/studio/index.html>

Developed by: Google, JetBrains

**Installation Process**

Download Android Studio

Download link <https://developer.android.com/studio/index.html>



Download Java Development Kit JDK SE Standard Edition from Oracle Website

Download Link <http://www.oracle.com/technetwork/java/javase/downloads/index.html>

After installing JDK execute following commands to find an installed version of java and its path.

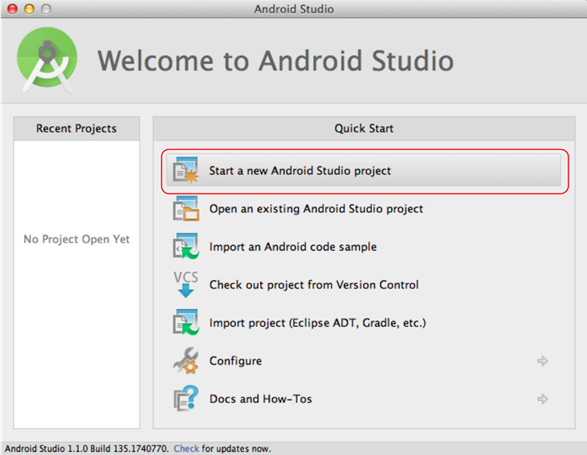
Go to command prompt and execute following commands

* Java -version
* for %i in (java.exe) do @echo. %~$PATH:i

**Making Your First App: Hello World**

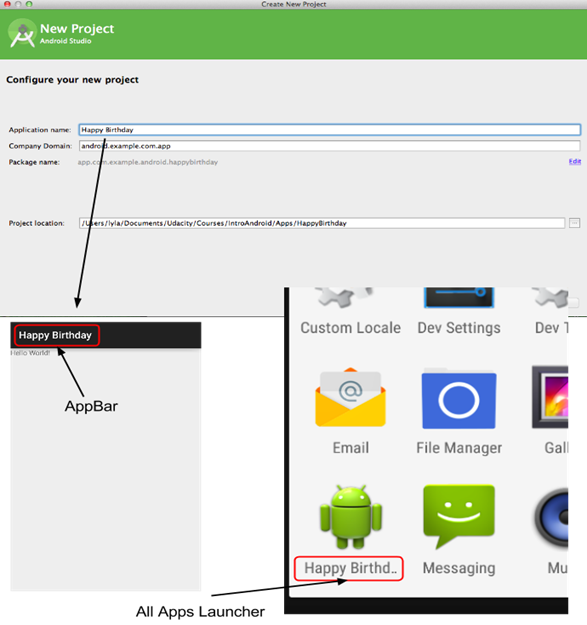
**1. Welcome to Android Studio Screen**

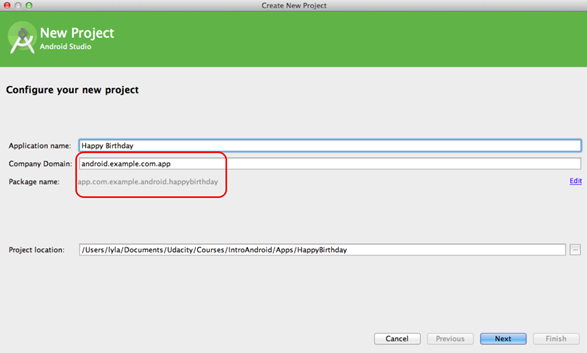
First, open Android Studio. You should see this window. Go ahead and click Start a new Android Studio Project. An Android Studio Project typically means the code and files for one Android Application.



**2. New Project Screen**

Here we see a few options we get to choose.



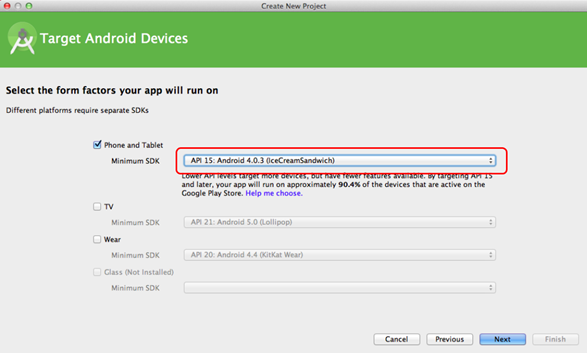


**Application Name** is the name of the application we'll see in the App Bar at the top of our app, as well as shown in the All Apps screen on the device. Use **Happy Birthday**

Next is **company domain**. Since we're practicing together right now, use the domain **android.example.com**, which should be what is automatically shown. If you look below, you'll see that it generates the package name com.example.android.happybirthday, which is combination of the reverse of the company domain and your app name.

**3. Target Android Devices**

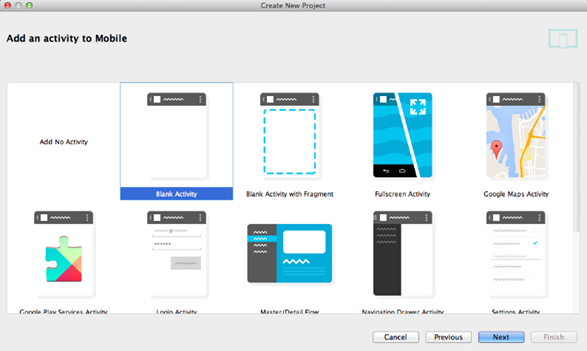
Next you will see this screen:



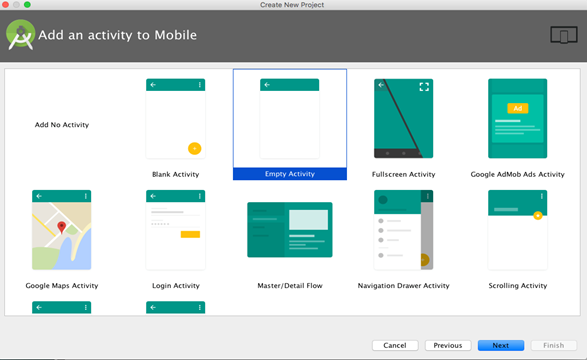
Only select **Phone and Tablet** since we're not making an app for TV or Wear. For **Minimum SDK** choose the deliciously named **API 15: Android 4.0.3 (IceCreamSandwich)**. If you're wondering about SDK naming, check out the [**wikipedia article on Android Version History**](http://en.wikipedia.org/wiki/Android_version_history)

**4. Add an activity to Mobile**

Here we are given a bunch of activity templates that we can choose from, depending on what we're trying to make. Go ahead and choose **Blank Activity** (or **Empty Activity** on Android Studio 1.4 and newer). It's the simplest one that still gives you some default files to start with.

Android Studio 1.3 and Older:  


Android Studio **1.4 and newer**:



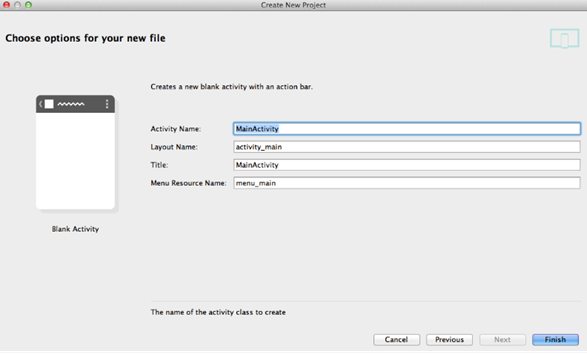
**5. Choose option for your new file**

Finally we have some names for different files that make up our app. As you'll soon see, an application is defined in picture files, sound files, xml files, java files and more. There are a lot of files that work together to make your application.

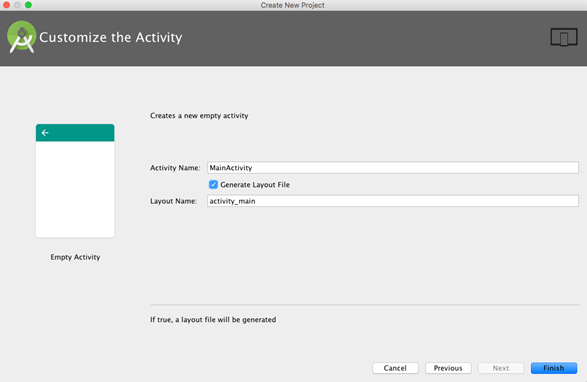
We'll talk about this all shortly. For now, take a look at the layout name. You've been working on XML layout files and this is the name of the XML file you'll be editing in the coming exercises.

For Android Studio 1.3 and older, **Use the defaults** which should be:

* Activity Name: **MainActivity**
* Layout Name: **activity\_main**
* Title: **MainActivity**
* Menu Resource Name: **menu\_main**



For Android Studio **1.4 and newer**, use the defaults shown here:



Depending on your computer speed it might take a minute to set up your project. Go ahead and get some tea or do a stretch.

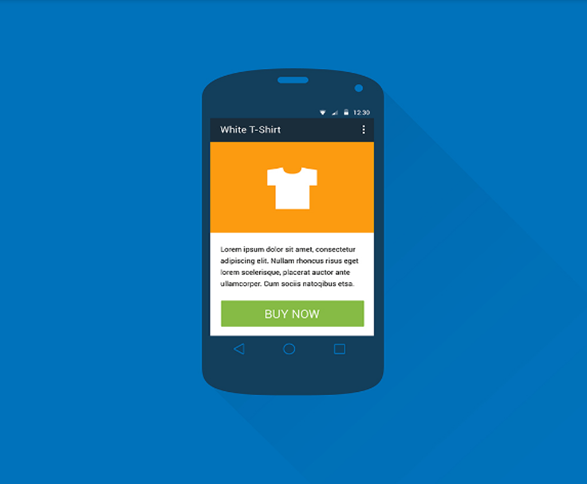
## **User Interface**

Your app's user interface is everything that the user can see and interact with. Android provides a variety of pre-built UI components such as structured layout objects and UI controls that allow you to build the graphical user interface for your app. Android also provides other UI modules for special interfaces such as dialogs, notifications, and menus.

### SELECTING, STYLING AND POSITIONING XML VIEWS

A View is a rectangular area visible on the screen. It has a width and height, and sometimes a background color.

The illustration shows Views of three different types. An **ImageView** displays an image such as an icon or photo. A **TextView** displays text. A Button is a TextView that is sensitive to touch: tap it with your finger and it will respond. And a **ViewGroup** is a big View—often invisible—that contains and positions the smaller Views inside of it.



A View on the screen is drawn by a Java object inside the Android device. In fact, the Java object is the real View. But when talking about what the user sees, it’s convenient to refer to the rectangular area on the screen as the “View”.

**XML CODE**

**TextView**

<TextView

android:id="@+id/text\_view\_id"

android:layout\_height="wrap\_content"

android:layout\_width="wrap\_content"

android:text="@string/hello" />

**ImageView**

<ImageView

android:layout\_height="wrap\_content"

android:layout\_width="wrap\_content"

android:src="@drawable/myimage" />

**ButtonView**

<Button

android:id="@+id/button\_id"

android:layout\_height="wrap\_content"

android:layout\_width="wrap\_content"

android:text="self\_destruct" />

## **Time Boxing**

|  |  |  |
| --- | --- | --- |
| Activity Name | Activity Time | Total Time |
| Login Systems + Setting up android studio Environment | 3 mints + 5 mints | 8 mints |
| Walk through Theory & Tasks | 60 mints | 60 mints |
| Implement Tasks | 80 mints | 80 mints |
| Evaluation Time | 30 mints | 30 mints |
|  | Total Duration | 178 mints |

## **Objectives/Outcomes**

The purpose of this lab is to familiarize with basic of Android studio IDE

• Understanding and Installing Android Studio

• Creating First Android App

## 

## **Lab Tasks/Practical Work**

1. Download, install Android Studio on the system.
2. Verify SDK settings within the Android Studio
3. Creating shortcut of Studio.exe or Studio64.exe
4. Updating PATH environment variable for SDK\platform-tools
5. Stopping WSearch service
6. Excluding folders in Windows Defender antivirus.

