**Experiment No. 1**

PART A

(PART A: TO BE REFFERED BY STUDENTS)

A.1 Aim: **Introduction to android by implementing “HelloWorld” application in android and understand building blocks of Android application.**

**A.2 Prerequisite:**

Knowledge of Java programming language

**A.3 Outcome:**

After successful completion of this experiment students will be able to

1. Have knowledge of all the android version and features.
2. Apply Android interface guidelines in development of standardized mobile applications.

**A.4 Theory:**

Android is an open source and Linux-based operating system for mobile devices such as smartphones and tablet computers. Android was developed by the *Open Handset Alliance*, led by Google, and other companies.

Android provides a rich application framework that allows you to build innovative apps and games for mobile devices in a Java language environment.

Android Architecture:

Android operating system is a stack of software components which is roughly divided into five sections and four main layers as shown below in the architecture diagram.



In this lab you will familiarize yourself with Android studio IDE and get your first application running on the emulator.

**Installation:**

Download AndroidStudio.exe file from <https://developer.android.com/studio/index.html> . Then install the same in your system. Make sure Java JDK is install in your system.

After Successfully installation of Android Studio, Launch android studio.

Open SDK Manager  for API setting. Make sure you have at least one API version platform installed on your system. If not installed then select the API which you want to install and accept the license agreement and download the relevant packages.

To test your Android applications, you will need a virtual Android device. To create an Android virtual device. Launch Android AVD Manager  and follow the instructions or install genymotion and test your app.

**Building blocks of Android application:**

1. Activities
2. Services
3. IntentReceiver
4. ContentProviders

Activities:

* Single focus thing that an user can do. Activity is a screen to which user interact to.
* Typically correspond to one UI screen
* Activity performs actions on the screen.
* Example Email application activities will be:
  + Activity that shows list of new emails
  + Activity to compose an email
  + Activity for reading emails
* If an application is having more than one activity, then one of them should be marked as the activity that is presented when the application is launched.
* An activity is implemented as a subclass of Activity class :

**public class MainActivity extends Activity{ }**

**Intent receivers or broadcast receivers:**

* Components that respond to broadcast messages from other applications.
* Way to respond to external notification or alarms
* Apps can also initiate broadcasts to let other applications know that some data has been downloaded to the device and is available for them to use.

**Services:**

* Faceless components that run in the background while user is doing some other activity without interacting with the user.
* E.g. playing music in the background while the user is in a different application.
* Fetch data over the network while user is doing some other activity.
* A service is implemented as a subclass of Service class as follows:

public class MyService extends Service { }

**Content Providers:**

* Enables sharing of data across applications
* E.g. address book, photo gallery
* The data may be stored in the file system, the database or somewhere else entirely.

public class MyContentProvider extends ContentProvide{ }

**A.5 Tasks need to be done:**

**1.** Make your system ready for app development in android.

2. Create a project and display Helloworld on the emulator.

PART B

(PART B: TO BE COMPLETED BY STUDENTS)

**(Students must submit the soft copy as per following segments within two hours of the practical. The soft copy must be uploaded on the Blackboard or emailed to the concerned lab in charge faculties at the end of the practical in case the there is no Black board access available)**

|  |  |
| --- | --- |
| Roll No. E44 | Name: Chinmay Shah |
| Program:BTech CS | Division: E |
| Semester: IV | Batch : E2 |
| Date of Experiment: 17/1/18 | Date of Submission: 24/1/18 |
| Grade : |  |

B.1 Software Code written by student:

***(Paste your xml, java and manifest code of t***

***he project here)***

xml

<?xml version="1.0" encoding="utf-8"?>  
<android.support.constraint.ConstraintLayout 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"  
 tools:context="com.example.mpstmestudent.calme.MainActivity">  
  
 <TextView  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:text="Hello World!"  
 app:layout\_constraintBottom\_toBottomOf="parent"  
 app:layout\_constraintLeft\_toLeftOf="parent"  
 app:layout\_constraintRight\_toRightOf="parent"  
 app:layout\_constraintTop\_toTopOf="parent" />  
  
  
</android.support.constraint.ConstraintLayout>

java

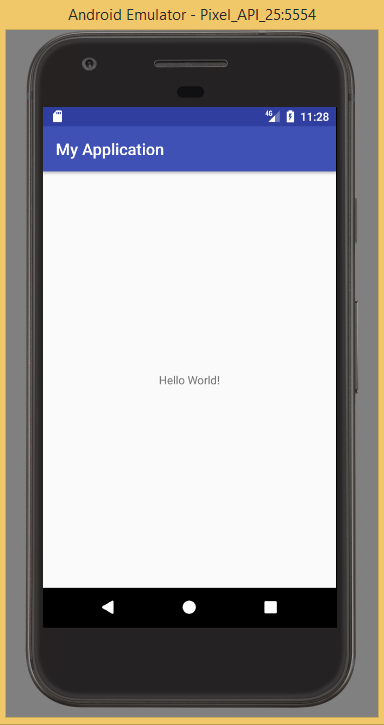
package com.example.mpstmestudent.calme;  
  
import android.support.v7.app.AppCompatActivity;  
import android.os.Bundle;  
  
public class MainActivity extends AppCompatActivity {  
  
 @Override  
 protected void onCreate(Bundle savedInstanceState) {  
 super.onCreate(savedInstanceState);  
 setContentView(R.layout.activity\_main);  
 }  
}

manifest

<?xml version="1.0" encoding="utf-8"?>  
<manifest xmlns:android="http://schemas.android.com/apk/res/android"  
 package="com.example.mpstmestudent.calme">  
  
 <application  
 android:allowBackup="true"  
 android:icon="@mipmap/ic\_launcher"  
 android:label="@string/app\_name"  
 android:roundIcon="@mipmap/ic\_launcher\_round"  
 android:supportsRtl="true"  
 android:theme="@style/AppTheme">  
 <activity android:name=".MainActivity">  
 <intent-filter>  
 <action android:name="android.intent.action.MAIN" />  
  
 <category android:name="android.intent.category.LAUNCHER" />  
 </intent-filter>  
 </activity>  
 </application>  
  
</manifest>

B.2 Input and Output:

***(Paste your screenshot of the project output here with input/output.)***



B.3 Question of Curiosity:

1. List down different activities that can be there is any one real life application.

In an email application, there can be many activities:

LogIn Screen

Email retrieves new screen

Activity that shows list of new emails

Compose an email

Reading emails

Search option for mails

1. What do you mean by launching activity?

Launching an activity refers to the activity being placed on screen.

1. Which function will be used as a starting point for any activity?

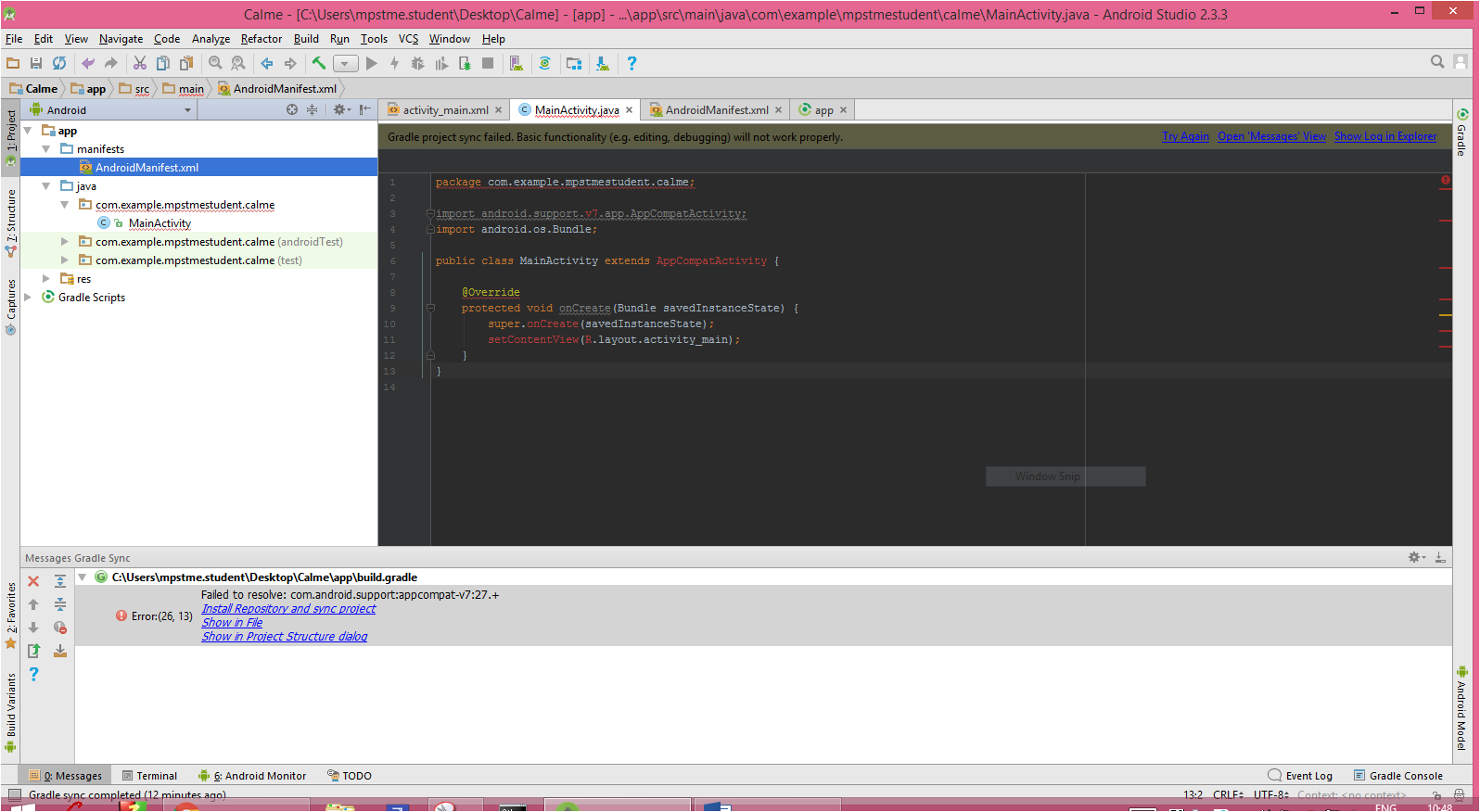
OnCreate() function will be present at starting point of each activity. This function initializes the activity.

1. Explain use of manifest file? Where it is located in android studio? Locate it in android studio and put a screen shot over here.

Every application must have an AndroidManifest.xml file (with precisely that name) in its root directory. The manifest file provides essential information about your app to the Android system, which the system must have before it can run any of the app's code.

Among other things, the manifest file does the following:

* It names the Java package for the application. The package name serves as a unique identifier for the application.
* It describes the components of the application, which include the activities, services, broadcast receivers, and content providers that compose the application. It also names the classes that implement each of the components and publishes their capabilities, such as the Intent messages that they can handle. These declarations inform the Android system of the components and the conditions in which they can be launched.
* It determines the processes that host the application components.
* It declares the permissions that the application must have in order to access protected parts of the API and interact with other applications. It also declares the permissions that others are required to have in order to interact with the application's components.
* It lists the Instrumentation classes that provide profiling and other information as the application runs. These declarations are present in the manifest only while the application is being developed and are removed before the application is published.
* It declares the minimum level of the Android API that the application requires.
* It lists the libraries that the application must be linked against.



B.4 Conclusion:

*(****Students must write the conclusion as per the attainment of individual outcome listed above and learning/observation noted in section B.1)***

Understood the basic of android layout and launched a basic android app.