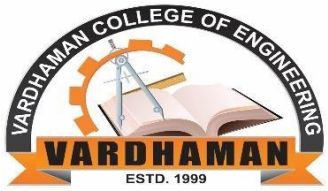
VARDHAMAN COLLEGE OF ENGINEERING, HYDERABAD

(Autonomous)

NAAC-A Grade, NBA Accredited: Affiliated to JNTUH, HyderabadISO 9001:2008 Certified.

DEPARTMENT OF INFORMATION TECHNOLOGY



Laboratory Manual of

MOBILE APPLICATION DEVELOPMENT

(III B.Tech- I SEMESTER)

(VCE-R19)

<table>
<tr><td colspan="2">Course Title</td><td colspan="3">MOBILE APPLICATION DEVELOPMENT</td><td colspan="2">Course Type</td><td colspan="2">Integrated</td></tr>
<tr><td colspan="2">Course Code</td><td>A5606</td><td>Credits</td><td>4</td><td colspan="2">Class</td><td colspan="2">III Year I Semester</td></tr>
<tr><td rowspan="5">Course Structure</td><td>TLP</td><td>Credits</td><td>Contact Hours</td><td>Work load</td><td colspan="2" rowspan="2">Total Number of Classes Per Semester</td><td colspan="2" rowspan="2">Assessment in Weightage</td></tr>
<tr><td>Theory</td><td>3</td><td>3</td><td>3</td></tr>
<tr><td>Practical</td><td>1</td><td>2</td><td>1</td><td rowspan="2">Theory</td><td rowspan="2">Practical</td><td>CIE</td><td>SEE</td></tr>
<tr><td>Tutorial</td><td>-</td><td>-</td><td>-</td><td></td><td></td></tr>
<tr><td>Total</td><td>4</td><td>5</td><td>4</td><td>42</td><td>28</td><td>30</td><td>70</td></tr>
<tr><td colspan="3">Lab Manual Prepared by</td><td colspan="6">Mrs. B Swapna, Assistant Professor</td></tr>
</table>

Component		Duration in Hours	Component Wise Marks	Total Marks	Weightage	Marks
Continuous Internal Evaluation (CIE)	Theory: Test-1	1:30	30	100	0.3	30
	Practical-Test-1	1:00	10			
	Theory: Test-2	1:30	30			
	Practical-Test-2	1:00	10			
	Alternate Assessment*	Theory	15			
		Practical	05			
Semester End Exam (SEE)	Theory	3:00	75	100	0.7	70
	Practical	2:00	25			
Total						100

Course Overview:

The latest mobile devices and applications are changing the way we communicate, do business, and access news and entertainment. Businesses, consumers and programmers have embraced this innovative medium, making mobile application developer one of the most demanded and fastest growing IT career paths. This course teaches students how to build mobile apps for Android mobile operating platform. Students learn to write native apps for Android based devices using Eclipse and the Android SDK. Students are expected to work on a project that produces a professional-quality mobile application. Projects will be deployed in real-world applications.

COURSE OBJECTIVES

With this course, students will be able to:

I. Describe those aspects of mobile programming that make it unique fromprogramming for other platforms
II. Utilize rapid prototyping techniques to design and develop sophisticated mobileinterfaces
III. Program mobile applications for the Android operating system that use basic andadvanced phone features
IV. Deploy applications to the Android marketplace for distribution

COURSE OUTCOMES (COs)

After the completion of the course, the student will be able to:

CO #	Course Outcome	PO	PSO
A5606.1	Understand architecture, the ecosystem, features and tools to design mobile applications		
A5606.2	Create effective user interfaces that leverage evolving mobiledevice capabilities	3,5	1
A5606.3	Design, customize and enhance mobile applications with various widgets	3	1
A5606.4	Develop various user friendly mobile applications with different application components	5	1,2
A5606.5	Build database applications to provide complete mobile development solutions	3,5	1,2

BLOOM'S LEVEL OF THE COURSE OUTCOMES

CO #	Bloom's Level					
	Remember (L1)	Understand (L2)	Apply (L3)	Analyze (L4)	Evaluate (L5)	Create (L6)
A5606.1		✓				
A5606.2			✓			
A5606.3			✓			
A5606.4			✓			
A5606.5			✓			

COURSE ARTICULATION MATRIX

CO # / PO's	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11	PO 12	PSO 1	PSO 2
A5606.1			3		3								3	
A5606.2			3										3	
A5606.3					3								3	3
A5606.4			3		3								3	3
A5606.5			3		3								3	

Note: 1-Low, 2-Medium, 3-High

LIST OF PROGRAMS FOR PRACTICE:

S. No	Experiment	Tools/Techniques Used	Skills/Ability
Week-1	Create an android app to illustrate activity life cycle	Android Studio IDE	Able to Develop Mobile Applicationsusing various widgets
Week-2	a. Create an android app to visit a specified webpage (Use Implicit Intent) b. Create an android app to navigate between activities (Use Explicit Intent)		
Week-3	a. Create an android app to perform mathematical operations (+ , - ,*, / , %).(Use buttons, edittext, toast controls) b. Create an android app to display text in bold, italic, normal style with left, right, center alignments (use RadioButton, CheckBox controls)		
Week-4	a. Create an android app to display name of the country from the list(Use spinnercontrol) b. Create an android app to calculate age of a person (Use DatePicker control) c. Create an android app design login control and validate login details		
Week-5	a. Create an android app to demonstrate AlertDialog Create an android app to demonstrateWebViewcontrol		

Week-6	a. Create an android app to show Analog and Digital clocks b. Create an android app to illustrate a progressbar		
Week-7	a. Create an android app to demonstrate list fragment b. Create an android app to demonstrate dialog fragment	Android Studio IDE	Able to Develop Mobile Applicationsusing various widgets
Week-8	a. Create an android app to demonstrate option menu, handling listeners b. Create an android app to scroll list of images anddisplay details of images (name, size etc) using ImageSwitcher control		
Week-9	a. Create an android app to demonstrate sending e-mail b. Create an android app to demonstrate sending SMS		
Week-10	a. Create an android app to show details phone contacts, implement calling, receiving features		
Week-11	a. Create an android app to demonstrate camera b. Create an android app to demonstrate mediaplyer		
Week-12	a. Create an android app to store details of students in SQLite and display the details b. Create an android app to perform insert, update,delete operations on student database		Able to DevelopMobile Applicationsusing SQLiteDatabase.

LAB MANUAL CONTENTS

This manual includes five Parts.

PART	CONTENT
1	INTRODUCTION TO ANDROID STUDIO & XML
2	WEEKLY LAB EXCERCISES
3	ONLINE RESOURCES
4	POSSIBLE VIVA QUESTIONS
5	KNOWLEDGE BASE

PART-1

INTRODUCTION TO ANDROID STUDIO & XML

Introduction to Android Studio

Android Studio is a popular IDE developed by Google for developing applications that are targeted at the android platform. Android Studio contains tools such as the Android Virtual Device Manager and the Android Device Monitor. It also contains Gradle, which helps you configure your Android application seamlessly. Some of the interesting features of Android Studio include the following:

1) Support for a fast emulator
2) Support for Gradle
3) Support for plenty of code templates and GitHub integration
4) Support for Google Cloud Platform
5) Support for template-based wizards for creating Android designs and components
6) Support for rich layout editor
7) Support for deep code analysis
8) Support for extensive set of tools and frameworks

ANDROID STUDIO INSTALLATION

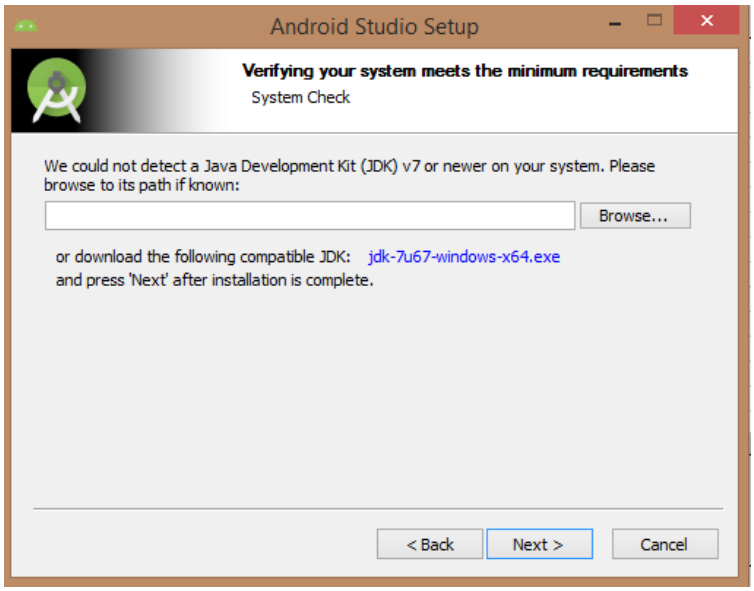
Android Studio is the official IDE for android application development.It works based on IntelliJ IDEA.
So let's launch Android Studio.exe, Make sure before launch Android Studio, Our Machine should required installed Java JDK.

ANDROID STUDIO INSTALLATION

Android Studio is the official IDE for android application development. It works basedon **IntelliJ IDEA**.

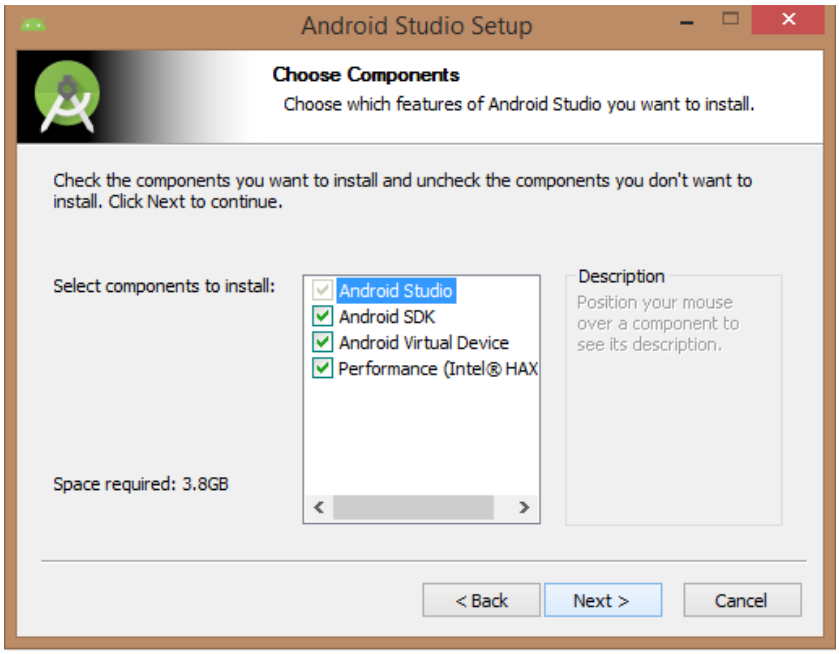
So let's launch *Android Studio.exe*, Make sure before launch Android Studio, OurMachine should require installed Java JDK.

Once you launched Android Studio, its time to mention JDK7 path or later version inandroid studio installer.

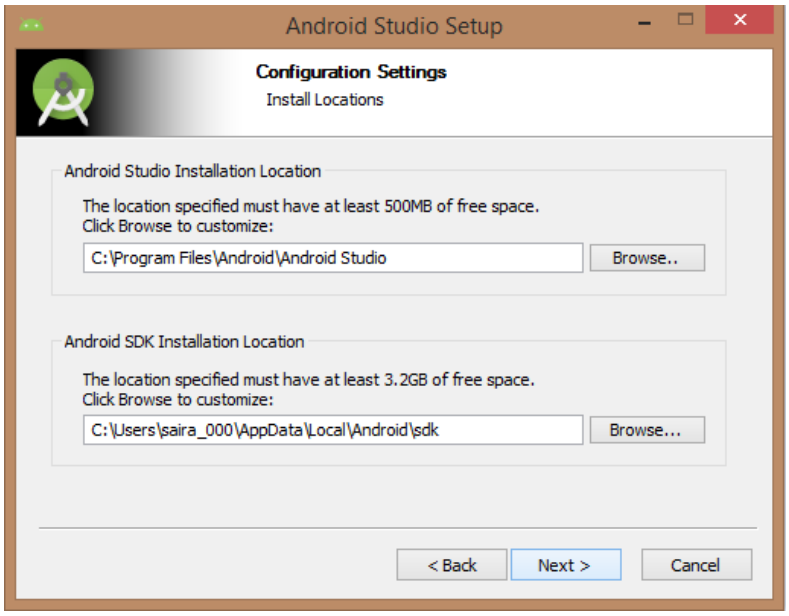


Below the image initiating JDK to android SDK

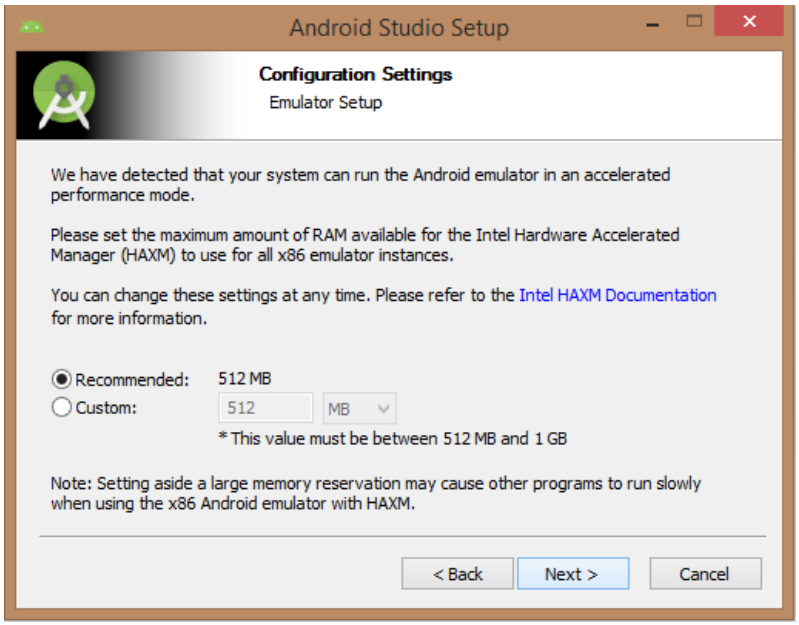
Need to check the components, which are required to create applications, below theimage has selected **Android Studio**, **Android SDK**, **Android Virtual Machine** and **performance(Intel chip).**



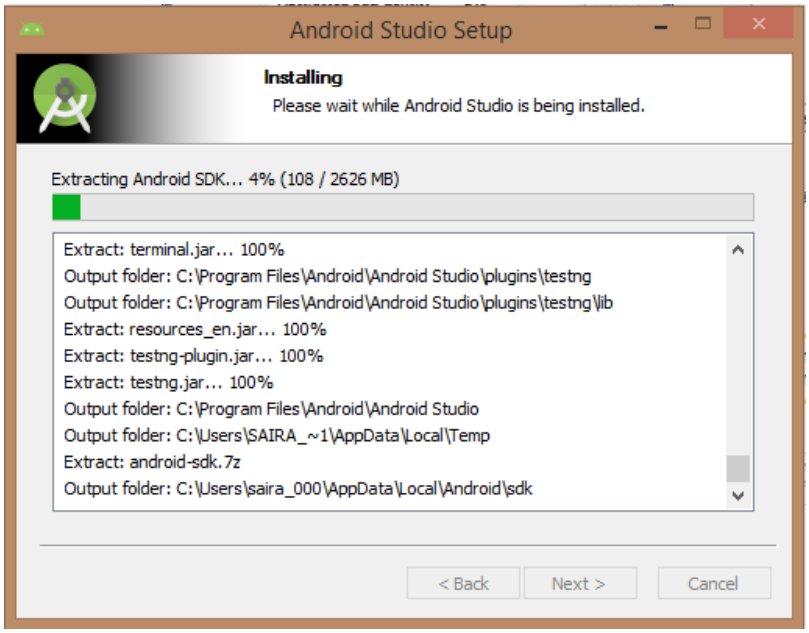
Need to specify the location of local machine path for Android studio and Android SDK, below the image has taken default location of windows 8.1 x64 bit architecture.



Need to specify the ram space for Android emulator by default it would take 512MB of local machine RAM.

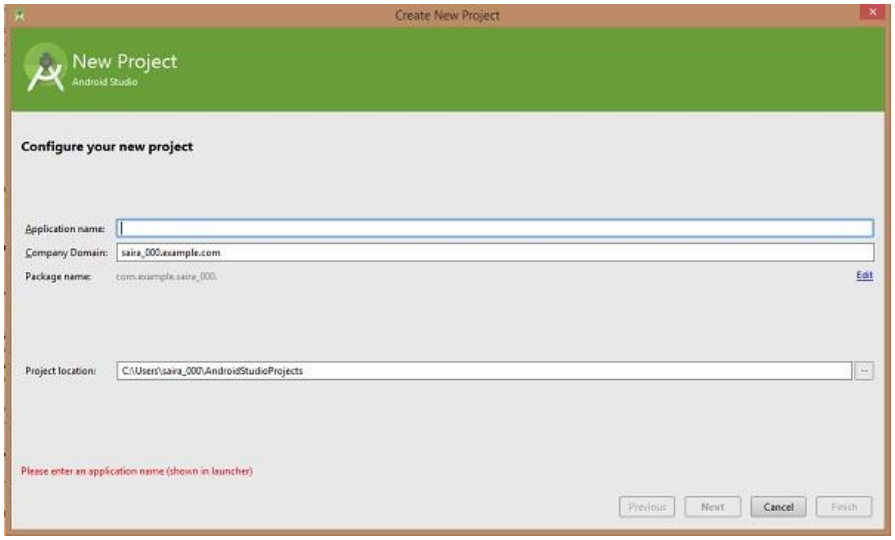


At final stage, it would extract SDK packages into our local machine, it would take a while time to finish the task and would take 2626MB of Hard disk space.

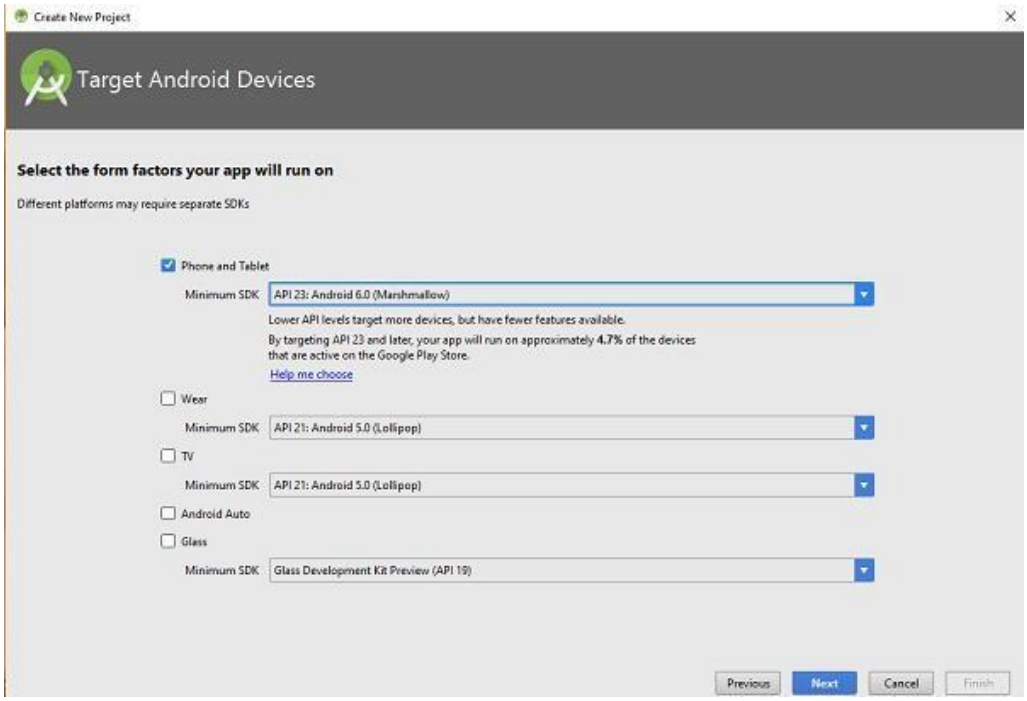


After done all above steps perfectly, you must get finish button and it gonna be open android studio project with Welcome to android studio message as shown below

You can start your application development by calling start a new android studio project. in a new installation frame should ask Application name, package information and location of the project.

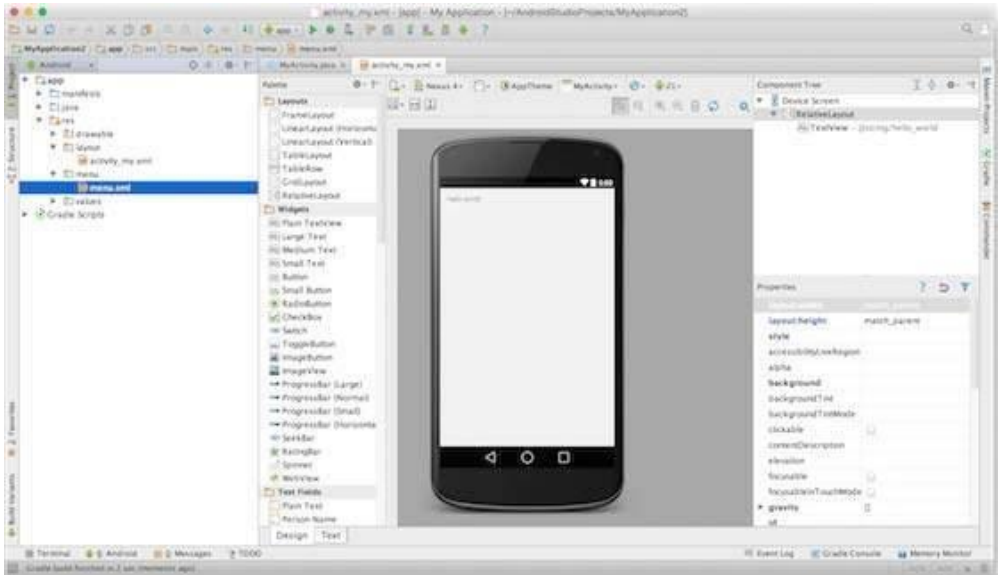


After entered application name, it going to be called select the form factors your application runs on, here need to specify Minimum SDK, in our tutorial, I have declared as API23: Android 6.0(Mashmallow)



The next level of installation should contain selecting the activity to mobile, itspecifies the default layout for Applications

At the final stage it going to be open development tool to write the application code.



Create Android Virtual Device

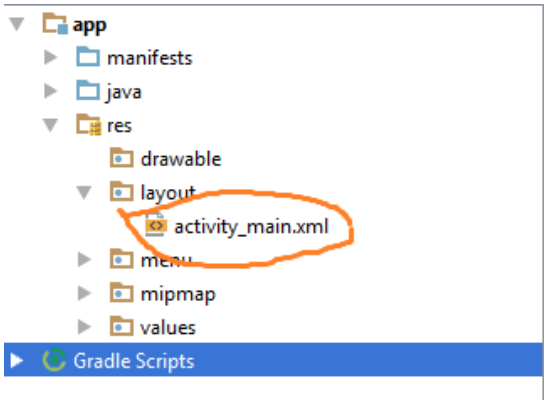
To test your Android applications, you will need a virtual Android device. So before we start writing our code, let us create an Android virtual device. Launch Android AVD Manager Clicking AVD_Manager icon as shown below

After Click on a virtual device icon, it going to be shown by default virtual devices which are present on your SDK, or else need to create a virtual device by clicking **Create new Virtual device** button

If your AVD is created successfully it means your environment is ready for Android application development. If you like, you can close this window using top-right cross button. Better you re-start your machine and once you are done with this last step, you are ready to proceed for your first Android example but before that we will see few more important concepts related to Android Application Development.

Hello Word Example

Before Writing a Hello word code, you must know about XML tags.To write hello word code, you should redirect to **App>res>layout>Activity_main.xml**



To show hello word, we need to call text view with layout (about text view and layout, you must take references at Relative Layout and Text View).

```
<RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"
xmlns:tools="http://schemas.android.com/tools"
android:layout_width="match_parent"
android:layout_height="match_parent"
 android:paddingLeft="@dimen/activity_horizontal_margin"
android:paddingRight="@dimen/activity_horizontal_margin"
android:paddingTop="@dimen/activity_vertical_margin"
android:paddingBottom="@dimen/activity_vertical_margin"
 tools:context=".MainActivity">
<TextView android:text="@string/hello_world"
android:layout_width="550dp"
android:layout_height="wrap_content" />
```

</RelativeLayout>

Need to run the program by clicking **Run>Run App** or else need to call **shift+f10**key. Finally, result should be placed at Virtual devices as shown below

Introduction to XML

- XML stands for eXtensible Markup Language
- XML is a markup language much like HTML
- XML was designed to store and transport data
- XML was designed to be self-descriptive

Maybe it is a little hard to understand, but XML does not DO anything.This note is a

note to Tove from Jani, stored as XML:

```
<note>
  <to>Tove</to>
  <from>Jani</from>
  <heading>Reminder</heading>
  <body>Don't forget me this weekend!</body>
</note>
```

The XML above is quite self-descriptive:

- It has sender information.
- It has receiver information
- It has a heading
- It has a message body.

But still, the XML above does not DO anything. XML is just informationwrapped in tags.

The Difference Between XML and HTML

XML and HTML were designed with different goals:

- XML was designed to carry data - with focus on what data is
- HTML was designed to display data - with focus on how data looks
- XML tags are not predefined like HTML tags are

XML Does Not Use Predefined Tags

The XML language has no predefined tags.

The tags in the example above (like <to> and <from>) are not defined inany XML standard. These tags are "invented" by the author of the XML document.

HTML works with predefined tags like <p>, <h1>, <table>, etc.

With XML, the author must define both the tags and the document structure.

XML is Extensible

Most XML applications will work as expected even if new data is added (orremoved).

Imagine an application designed to display the original version of note.xml(<to> <from> <heading> <body>).

Then imagine a newer version of note.xml with added <date> and <hour>elements, and a removed <heading>.

The way XML is constructed, older version of the application can still work:

```
<note>
  <date>2015-09-01</date>
  <hour>08:30</hour>
  <to>Tove</to>
  <from>Jani</from>
```

```
    <body>Don't forget me this weekend!</body>
</note>
```

XML Simplifies Things

- It simplifies data sharing
- It simplifies data transport
- It simplifies platform changes
- It simplifies data availability

Many computer systems contain data in incompatible formats. Exchangingdata between incompatible systems (or upgraded systems) is a time- consuming task for web developers. Large amounts of data must be converted, and incompatible data is often lost.

XML stores data in plain text format. This provides a software- and hardware-independent way of storing, transporting, and sharing data.

XML also makes it easier to expand or upgrade to new operating systems,new applications, or new browsers, without losing data.

With XML, data can be available to all kinds of "reading machines" likepeople, computers, voice machines, news feeds, etc.

XML Separates Data from Presentation
XML does not carry any information about how to be displayed.
The same XML data can be used in many different presentation scenarios.
Because of this, with XML, there is a full separation between data andpresentation.

XML is Often a Complement to HTML

In many HTML applications, XML is used to store or transport data, whileHTML is used to format and display the same data.

XML Separates Data from HTML

When displaying data in HTML, you should not have to edit the HTML filewhen the data changes.

With XML, the data can be stored in separate XML files.

With a few lines of JavaScript code, you can read an XML file and update thedata content of any HTML page.

PART-2
WEEKLY LAB EXERCISES

Week-1:

Aim: Create an android app to illustrate activity life Cycle
MainActivity.java

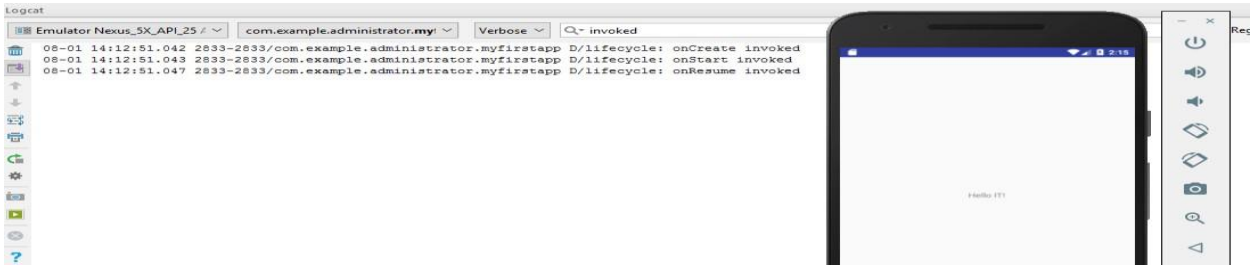
```
import android.os.Bundle;
import android.app.Activity;
import android.util.Log;
import android.view.Menu;
public class MainActivity extends Activity
{@Override
protected void onCreate(Bundle savedInstanceState) {
super.onCreate(savedInstanceState);
setContentView(R.layout.activity_main);
Log.d("lifecycle","onCreate invoked");
}
@Override
protected void onStart(){
super.onStart();
Log.d("lifecycle","onStart invoked");
}
@Override
protected void onResume() {
super.onResume();
Log.d("lifecycle","onResume invoked");
}
@Override
protected void onPause() {
super.onPause();
Log.d("lifecycle","onPause invoked");
}
@Override
protected void onStop() {
super.onStop();
Log.d("lifecycle","onStop invoked");
}
@Override
protected void onRestart() {
super.onRestart();
Log.d("lifecycle","onRestart invoked");
}
@Override
protected void onDestroy() {
super.onDestroy();
Log.d("lifecycle","onDestroy invoked");
}
```

}

Output

Open Log Cat

Now see on the logcat: onCreate, onStart and onResume methods are invoked.



Practice Exercise:

1) Create an android app to display ‘welcome to android’ message when app is loadingand ‘app is closing’ message when closing

2) Create an android app to change the background of an activity

Week-2 (a)

Aim: Create an android app to visit a specified webpage (Use ImplicitIntent)

activity_main.xml

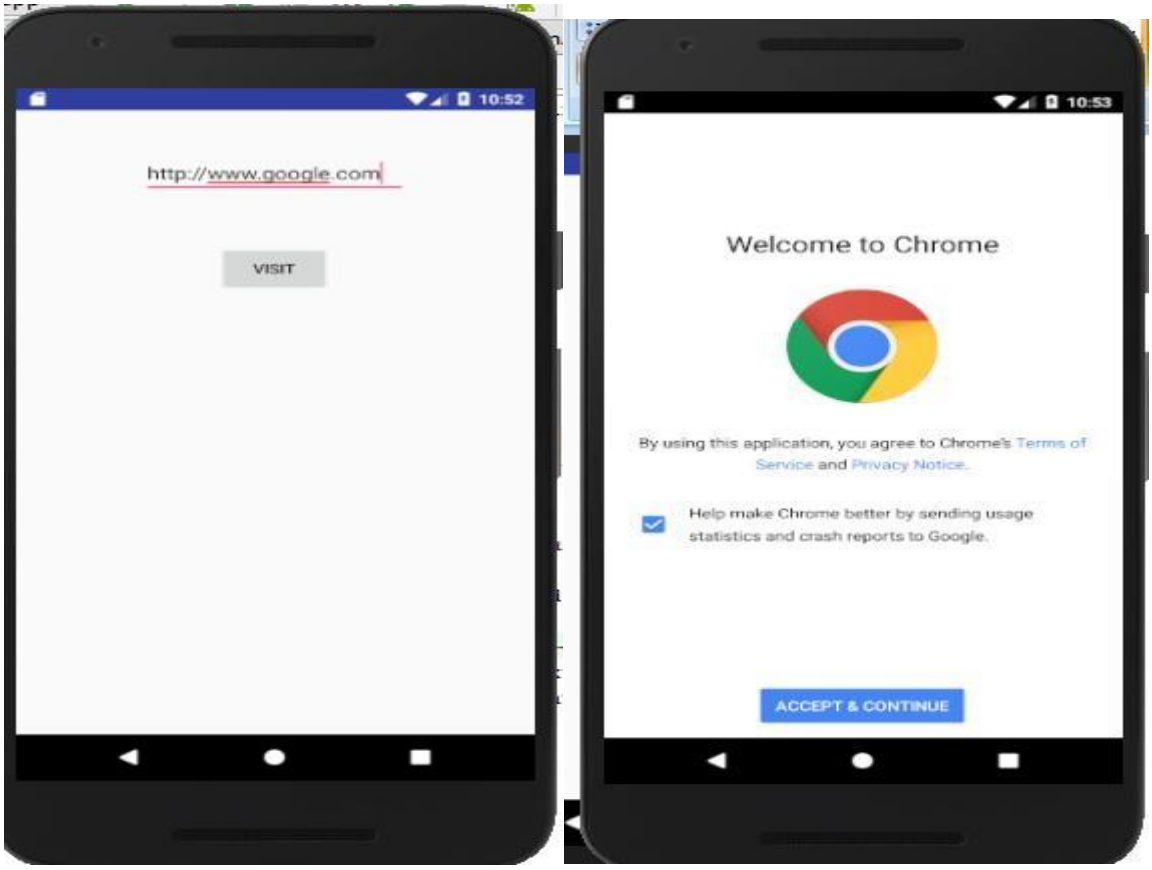
```
<?xml version="1.0" encoding="utf-8"?>
<RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"
 xmlns:app="http://schemas.android.com/apk/res-auto"
 xmlns:tools="http://schemas.android.com/tools"
 android:layout_width="match_parent" android:layout_height="match_parent"
 tools:context="com.example.administrator.implicit.MainActivity">

 <EditText
 android:id="@+id/editText1"
 android:layout_width="wrap_content"
 android:layout_height="wrap_content"
 android:layout_alignParentTop="true"
 android:layout_centerHorizontal="true"
 android:layout_marginTop="44dp" android:ems="10" />

 <Button
 android:id="@+id/button1"
 android:layout_width="wrap_content"
 android:layout_height="wrap_content"
 android:layout_below="@+id/editText1"
 android:layout_centerHorizontal="true"
 android:layout_marginTop="54dp" android:text="Visit" />
</RelativeLayout >
```

MainActivity.java

```
 import android.net.Uri; import
android.os.Bundle; import
android.app.Activity; import
android.content.Intent;import
android.view.View;
```

```
import android.view.View.OnClickListener;
import android.widget.Button;
import android.widget.EditText;

public class MainActivity extends Activity
{
        @Override
        protected void onCreate(Bundle savedInstanceState)
        {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);
       final EditText editText1=(EditText)findViewById(R.id.editText1);
        Button button1=(Button)findViewById(R.id.button1);
       button1.setOnClickListener(new OnClickListener()
       {@Override
       public void onClick(View arg0) {
        String url=editText1.getText().toString();
        Intent intent=new Intent(Intent.ACTION_VIEW,Uri.parse(url));
        startActivity(intent);
        }
        });
       }
}
```

Output


10:52
http://www.google.com
VISIT
10:53
Welcome to Chrome
By using this application, you agree to Chrome's Terms of Service and Privacy Notice.
Help make Chrome better by sending usage statistics and crash reports to Google.
ACCEPT & CONTINUE

Week-2 (b)

Create an android app to navigate between activities (Use Explicit Intent)

Activity_main.xml

```
<?xml version="1.0" encoding="utf-8"?>
<androidx.constraintlayout.widget.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=".MainActivity">

    <Button
        android:id="@+id/b1"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Call Second activity"
        app:layout_constraintBottom_toBottomOf="parent"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintHorizontal_bias="0.529"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toTopOf="parent"
        android:onClick="callsecondactivity"/>

    <EditText
        android:id="@+id/t1"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:ems="10"
        android:inputType="textPersonName"
        android:text="FirstActivity"
        app:layout_constraintBottom_toBottomOf="parent"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintHorizontal_bias="0.597"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toTopOf="parent"
        app:layout_constraintVertical_bias="0.262" />

</androidx.constraintlayout.widget.ConstraintLayout>
```

MainActivity.java

```
package com.vardhaman.myapplication;
import androidx.appcompat.app.AppCompatActivity;
import android.content.Intent;
import android.os.Bundle;
import android.view.View;
public class MainActivity extends AppCompatActivity {
    @Override
```

```
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);
    }
    public void callsecondactivity(View view){
        Intent i = new Intent(getApplicationContext(), secondactivity.class);
        i.putExtra("Value1", "5");
        i.putExtra("Value2", "6");
        startActivity(i);
    }

}
```

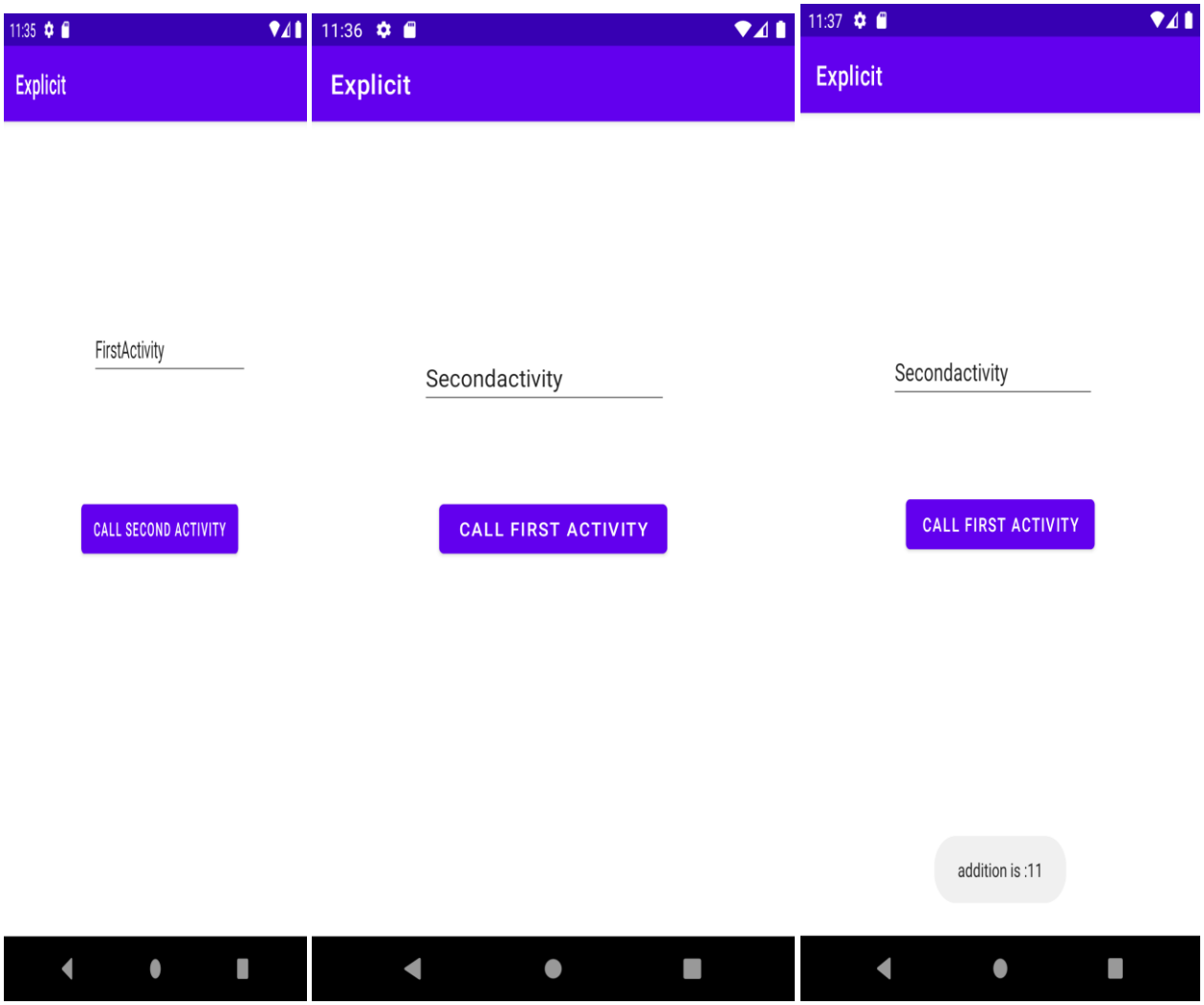
activity_secondactivity.xml

```
<?xml version="1.0" encoding="utf-8"?>
<androidx.constraintlayout.widget.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=".secondactivity">
    <EditText
        android:id="@+id/t2"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:ems="10"
        android:inputType="textPersonName"
        android:text="Secondactivity"
        app:layout_constraintBottom_toBottomOf="parent"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintHorizontal_bias="0.462"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toTopOf="parent"
        app:layout_constraintVertical_bias="0.3" />
    <Button
        android:id="@+id/b2"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Call First Activity"
        android:onClick="callMainActivity"
        app:layout_constraintBottom_toBottomOf="parent"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toTopOf="parent" />
</androidx.constraintlayout.widget.ConstraintLayout>
```

secondactivity.java

```
package com.vardhaman.myapplication;
import androidx.appcompat.app.AppCompatActivity;
import android.content.Intent;
import android.os.Bundle;
import android.view.View;
import android.widget.Toast;
public class secondactivity extends AppCompatActivity {
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_secondactivity);
        Bundle extras = getIntent().getExtras();
        String value1 = extras.getString("Value1");
        String value2 = extras.getString("Value2");
        int a=Integer.parseInt(value1);
        int b=Integer.parseInt(value2);
        int c=a+b;
        Toast t= Toast.makeText(getApplicationContext(),"addition is :" +c, Toast.LENGTH_LONG);
        t.show();
    }
    public void callMainActivity(View view){
        Intent i = new Intent(getApplicationContext(), MainActivity.class);
        startActivity(i);
    }

}
```

Output:



Practice Exercise:

1. Create an android app with multiple activites like login, about, contact and navigate between all

Week-3 (a)

Aim: Create an android app to perform mathematical operations (+,-,*,/,%).(Use buttons, edittext, toast controls)

activity_main.xml

```
<?xml version="1.0" encoding="utf-8"?>
<androidx.constraintlayout.widget.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:id="@+id/b2"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:background="#F4F7F8"
    android:backgroundTint="#009688"
    android:overScrollMode="ifContentScrolls"
    tools:context=".MainActivity">

    <TextView
        android:id="@+id/textView"
        android:layout_width="68dp"
        android:layout_height="39dp"
        android:text="Number1"
        android:textAppearance="@style/TextAppearance.AppCompat.Body2"
        app:layout_constraintBottom_toBottomOf="parent"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintHorizontal_bias="0.144"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toTopOf="parent"
        app:layout_constraintVertical_bias="0.231" />

    <TextView
        android:id="@+id/textView2"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Number2"
        android:textAppearance="@style/TextAppearance.AppCompat.Body2"
        app:layout_constraintBottom_toBottomOf="parent"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintHorizontal_bias="0.144"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toTopOf="parent"
        app:layout_constraintVertical_bias="0.358" />

    <Button
        android:id="@+id/b1"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:onClick="add"
```

```
    android:text="Add"
    app:layout_constraintBottom_toBottomOf="parent"
    app:layout_constraintEnd_toEndOf="parent"
    app:layout_constraintHorizontal_bias="0.12"
    app:layout_constraintStart_toStartOf="parent"
    app:layout_constraintTop_toTopOf="parent"
    app:layout_constraintVertical_bias="0.563" />

<Button
    android:id="@+id/b2"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="Sub"
    android:onClick="sub"
    app:layout_constraintBottom_toBottomOf="parent"
    app:layout_constraintEnd_toEndOf="parent"
    app:layout_constraintHorizontal_bias="0.529"
    app:layout_constraintStart_toStartOf="parent"
    app:layout_constraintTop_toTopOf="parent"
    app:layout_constraintVertical_bias="0.563" />

<Button
    android:id="@+id/b3"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="Mul"
    android:onClick="mul"
    app:layout_constraintBottom_toBottomOf="parent"
    app:layout_constraintEnd_toEndOf="parent"
    app:layout_constraintHorizontal_bias="0.885"
    app:layout_constraintStart_toStartOf="parent"
    app:layout_constraintTop_toTopOf="parent"
    app:layout_constraintVertical_bias="0.563" />

<Button
    android:id="@+id/b4"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:onClick="div"
    android:text="Div"
    app:layout_constraintBottom_toBottomOf="parent"
    app:layout_constraintEnd_toEndOf="parent"
    app:layout_constraintHorizontal_bias="0.12"
    app:layout_constraintStart_toStartOf="parent"
    app:layout_constraintTop_toTopOf="parent"
    app:layout_constraintVertical_bias="0.696" />

<Button
    android:id="@+id/b5"
    android:layout_width="wrap_content"
```

```
    android:layout_height="wrap_content"
    android:onClick="mod"
    android:text="Mod"
    app:layout_constraintBottom_toBottomOf="parent"
    app:layout_constraintEnd_toEndOf="parent"
    app:layout_constraintHorizontal_bias="0.529"
    app:layout_constraintStart_toStartOf="parent"
    app:layout_constraintTop_toTopOf="parent"
    app:layout_constraintVertical_bias="0.696" />

<EditText
    android:id="@+id/e1"
    android:layout_width="211dp"
    android:layout_height="42dp"
    android:background="#EADBDB"
    android:ems="10"
    android:inputType="textPersonName"
    android:shadowColor="#FFFFFF"
    app:layout_constraintBottom_toBottomOf="parent"
    app:layout_constraintEnd_toEndOf="parent"
    app:layout_constraintHorizontal_bias="0.82"
    app:layout_constraintStart_toStartOf="parent"
    app:layout_constraintTop_toTopOf="parent"
    app:layout_constraintVertical_bias="0.232" />

<EditText
    android:id="@+id/e2"
    android:layout_width="208dp"
    android:layout_height="29dp"
    android:background="#F6F0F0"
    android:ems="10"
    android:inputType="textPersonName"
    app:layout_constraintBottom_toBottomOf="parent"
    app:layout_constraintEnd_toEndOf="parent"
    app:layout_constraintHorizontal_bias="0.815"
    app:layout_constraintStart_toStartOf="parent"
    app:layout_constraintTop_toTopOf="parent"
    app:layout_constraintVertical_bias="0.348" />

<TextView
    android:id="@+id/textView3"
    android:layout_width="0dp"
    android:layout_height="wrap_content"
    android:text="CALCULATOR"
    android:textAlignment="center"
    android:textAppearance="@style/TextAppearance.AppCompat.Display2"
    app:layout_constraintBottom_toBottomOf="parent"
    app:layout_constraintEnd_toEndOf="parent"
    app:layout_constraintHorizontal_bias="0.464"
    app:layout_constraintStart_toStartOf="parent"
```

```
        app:layout_constraintTop_toTopOf="parent"
        app:layout_constraintVertical_bias="0.12" />

    <TextView
        android:id="@+id/res"
        android:layout_width="192dp"
        android:layout_height="38dp"
        android:background="#8BC34A"
        android:textColor="#D51F1F"
        android:textSize="20sp"
        android:textStyle="bold"
        app:layout_constraintBottom_toBottomOf="parent"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintHorizontal_bias="0.569"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toTopOf="parent"
        app:layout_constraintVertical_bias="0.858" />

    <TextView
        android:id="@+id/textView4"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Result"
        android:textSize="20sp"
        android:textStyle="bold"
        app:layout_constraintBottom_toBottomOf="parent"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintHorizontal_bias="0.11"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toTopOf="parent"
        app:layout_constraintVertical_bias="0.845" />

</androidx.constraintlayout.widget.ConstraintLayout>
```

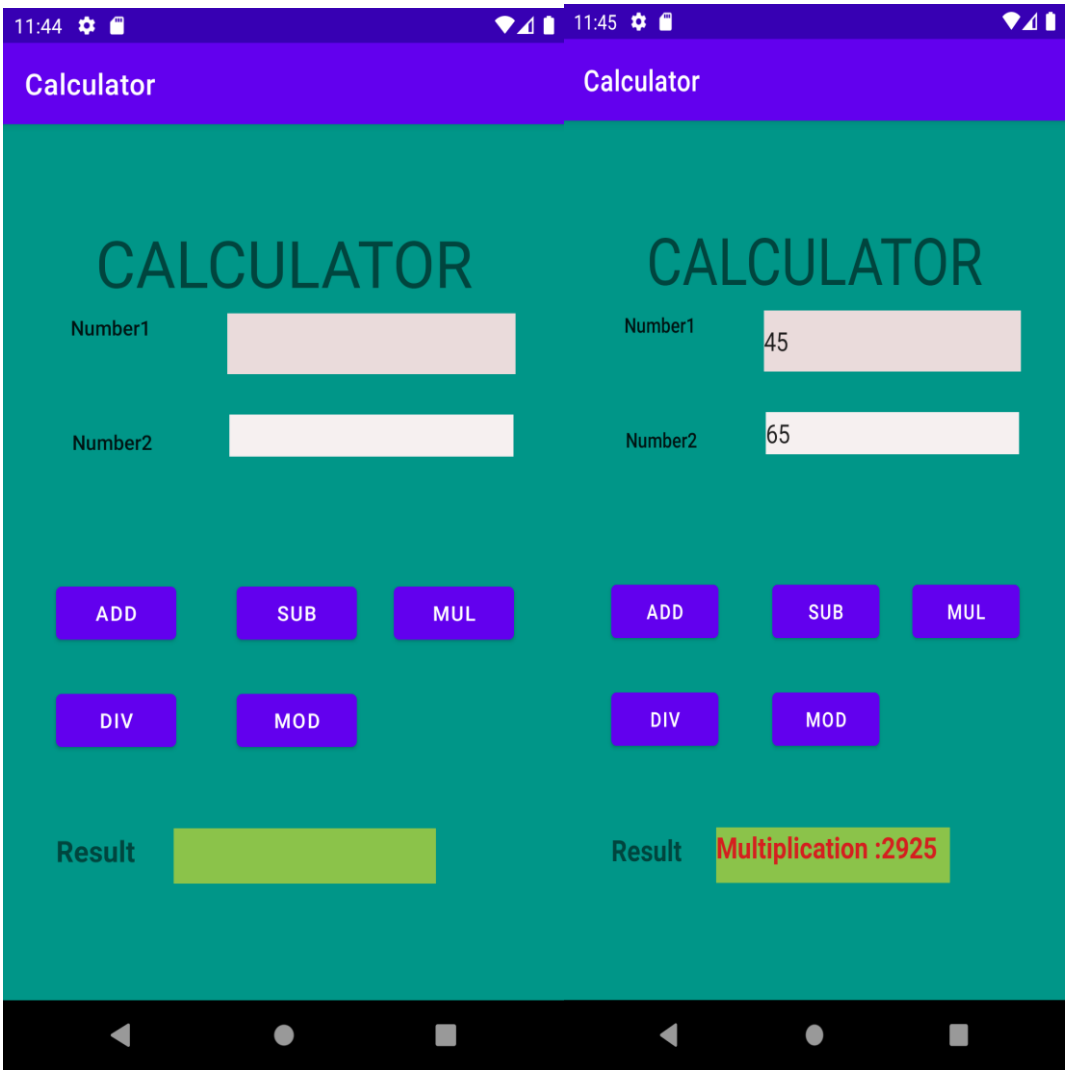
MainActivity.java:

```
package com.vardhaman.calculator;
import androidx.appcompat.app.AppCompatActivity;
import android.os.Bundle;
import android.view.View;
import android.widget.Button;
import android.widget.EditText;
import android.widget.TextView;
public class MainActivity extends AppCompatActivity {
    private EditText et1, et2, res;
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);
    }
    public void add(View view) {
```

```
    et1 = (EditText) findViewById(R.id.e1);
    et2 = (EditText) findViewById(R.id.e2);
    TextView t = (TextView) findViewById(R.id.res);
    int value1 = Integer.parseInt(et1.getText().toString());
    int value2 = Integer.parseInt(et2.getText().toString());
    int c = value1 + value2;
    TextView t2 = (TextView) findViewById(R.id.res);
    t2.setText("Addition :" + c);
}
public void sub(View view) {
    et1 = (EditText) findViewById(R.id.e1);
    et2 = (EditText) findViewById(R.id.e2);
    TextView t = (TextView) findViewById(R.id.res);
    int value1 = Integer.parseInt(et1.getText().toString());
    int value2 = Integer.parseInt(et2.getText().toString());
    int c = value1 - value2;
    TextView t2 = (TextView) findViewById(R.id.res);
    t2.setText("Substraction :" + c);
}
public void mul(View view) {
    et1 = (EditText) findViewById(R.id.e1);
    et2 = (EditText) findViewById(R.id.e2);
    TextView t = (TextView) findViewById(R.id.res);
    int value1 = Integer.parseInt(et1.getText().toString());
    int value2 = Integer.parseInt(et2.getText().toString());
    int c = value1 * value2;
    TextView t2 = (TextView) findViewById(R.id.res);
    t2.setText("Multiplication :" + c);

}
public void div(View view) {
    et1 = (EditText) findViewById(R.id.e1);
    et2 = (EditText) findViewById(R.id.e2);
    TextView t = (TextView) findViewById(R.id.res);
    int value1 = Integer.parseInt(et1.getText().toString());
    int value2 = Integer.parseInt(et2.getText().toString());
    int c = value1 / value2;
    TextView t2 = (TextView) findViewById(R.id.res);
    t2.setText("Divisor is :" + c);
}
public void mod(View view) {
    et1 = (EditText) findViewById(R.id.e1);
    et2 = (EditText) findViewById(R.id.e2);
    TextView t = (TextView) findViewById(R.id.res);
    int value1 = Integer.parseInt(et1.getText().toString());
    int value2 = Integer.parseInt(et2.getText().toString());
    int c = value1 % value2;
    TextView t2 = (TextView) findViewById(R.id.res);
    t2.setText("Remainder is :" + c);
}
```

}

Output


11:44
Calculator
CALCULATOR
Number1
Number2
ADD
SUB
MUL
DIV
MOD
Result
11:45
Calculator
CALCULATOR
Number1
45
Number2
65
ADD
SUB
MUL
DIV
MOD
Result
Multiplication :2925

Week-3 (b):

Aim: Create an android app to display text in bold, italic, normal style with left, right, centeralignments (use RadioButton, CheckBox controls)

activity_ Main . xml

```
<?xml version="1.0" encoding="utf-8"?>
<androidx.constraintlayout.widget.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"
   android:background="#4CAF50"
   tools:context=".MainActivity">
   <TextView
      android:id="@+id/textView"
      android:layout_width="wrap_content"
      android:layout_height="wrap_content"
      android:background="#D82626"
      android:text="Text Styles"
      app:layout_constraintBottom_toBottomOf="parent"
      app:layout_constraintEnd_toEndOf="parent"
      app:layout_constraintHorizontal_bias="0.459"
      app:layout_constraintStart_toStartOf="parent"
      app:layout_constraintTop_toTopOf="parent"
      app:layout_constraintVertical_bias="0.14" />
   <CheckBox
      android:id="@+id/c1"
      android:layout_width="85dp"
      android:layout_height="54dp"
      android:text="Bold"
      app:layout_constraintBottom_toBottomOf="parent"
      app:layout_constraintEnd_toEndOf="parent"
      app:layout_constraintHorizontal_bias="0.165"
      app:layout_constraintStart_toStartOf="parent"
      app:layout_constraintTop_toTopOf="parent"
      app:layout_constraintVertical_bias="0.254" />
   <CheckBox
      android:id="@+id/c2"
      android:layout_width="78dp"
      android:layout_height="43dp"
      android:text="Italic"
      app:layout_constraintBottom_toBottomOf="parent"
      app:layout_constraintEnd_toEndOf="parent"
      app:layout_constraintHorizontal_bias="0.162"
      app:layout_constraintStart_toStartOf="parent"
      app:layout_constraintTop_toTopOf="parent"
      app:layout_constraintVertical_bias="0.38" />
   <CheckBox
```

```
    android:id="@+id/c3"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="Normal"
    app:layout_constraintBottom_toBottomOf="parent"
    app:layout_constraintEnd_toEndOf="parent"
    app:layout_constraintHorizontal_bias="0.168"
    app:layout_constraintStart_toStartOf="parent"
    app:layout_constraintTop_toTopOf="parent"
    app:layout_constraintVertical_bias="0.499" />
<RadioGroup
    android:id="@+id/rg1"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    app:layout_constraintBottom_toBottomOf="parent"
    app:layout_constraintEnd_toEndOf="parent"
    app:layout_constraintHorizontal_bias="0.872"
    app:layout_constraintStart_toStartOf="parent"
    app:layout_constraintTop_toTopOf="parent"
    app:layout_constraintVertical_bias="0.321">
    <RadioButton
        android:id="@+id/r1"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Left"
        app:layout_constraintBottom_toBottomOf="parent"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintHorizontal_bias="0.798"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toTopOf="parent"
        app:layout_constraintVertical_bias="0.3"
        tools:text="Left" />
    <RadioButton
        android:id="@+id/r2"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Right"
        app:layout_constraintBottom_toBottomOf="parent"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintHorizontal_bias="0.798"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toTopOf="parent"
        app:layout_constraintVertical_bias="0.376" />
    <RadioButton
        android:id="@+id/r3"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Center"
        app:layout_constraintBottom_toBottomOf="parent"
        app:layout_constraintEnd_toEndOf="parent"
```

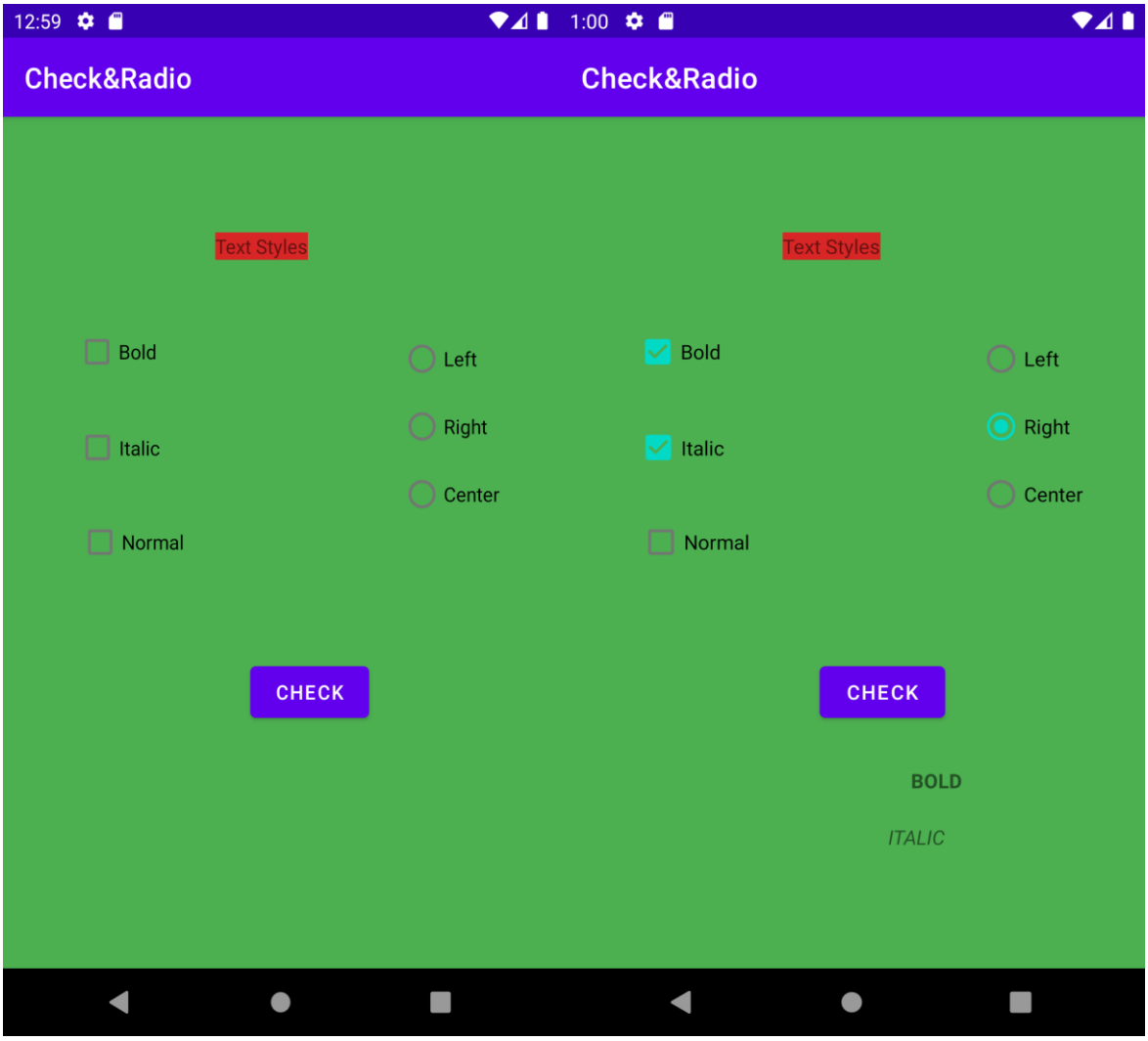
```
        app:layout_constraintHorizontal_bias="0.804"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toTopOf="parent"
        app:layout_constraintVertical_bias="0.455" />
    </RadioGroup>
    <Button
      android:id="@+id/button"
      android:layout_width="wrap_content"
      android:layout_height="wrap_content"
      android:onClick="show"
      android:text="Check"
      app:layout_constraintBottom_toBottomOf="parent"
      app:layout_constraintEnd_toEndOf="parent"
      app:layout_constraintHorizontal_bias="0.567"
      app:layout_constraintStart_toStartOf="parent"
      app:layout_constraintTop_toTopOf="parent"
      app:layout_constraintVertical_bias="0.691" />
    <TextView
      android:id="@+id/t3"
      android:layout_width="209dp"
      android:layout_height="34dp"
      app:layout_constraintBottom_toBottomOf="parent"
      app:layout_constraintEnd_toEndOf="parent"
      app:layout_constraintHorizontal_bias="0.487"
      app:layout_constraintStart_toStartOf="parent"
      app:layout_constraintTop_toTopOf="parent"
      app:layout_constraintVertical_bias="0.943" />
    <TextView
      android:id="@+id/t2"
      android:layout_width="151dp"
      android:layout_height="32dp"
      app:layout_constraintBottom_toBottomOf="parent"
      app:layout_constraintEnd_toEndOf="parent"
      app:layout_constraintHorizontal_bias="0.461"
      app:layout_constraintStart_toStartOf="parent"
      app:layout_constraintTop_toTopOf="parent"
      app:layout_constraintVertical_bias="0.876" />
    <TextView
      android:id="@+id/t1"
      android:layout_width="164dp"
      android:layout_height="23dp"
      app:layout_constraintBottom_toBottomOf="parent"
      app:layout_constraintEnd_toEndOf="parent"
      app:layout_constraintHorizontal_bias="0.481"
      app:layout_constraintStart_toStartOf="parent"
      app:layout_constraintTop_toTopOf="parent"
      app:layout_constraintVertical_bias="0.793" />
</androidx.constraintlayout.widget.ConstraintLayout>
```

Main Activity . java

```
package com.vardhaman.checkradio;
import androidx.appcompat.app.AppCompatActivity;
import android.graphics.Typeface;
import android.os.Bundle;
import android.view.Gravity;
import android.view.View;
import android.widget.CheckBox;
import android.widget.RadioButton;
import android.widget.RadioGroup;
import android.widget.TextView;
public class MainActivity extends AppCompatActivity {
   @Override
   protected void onCreate(Bundle savedInstanceState) {
      super.onCreate(savedInstanceState);
      setContentView(R.layout.activity_main);
   }
   public void show(View view) {
      RadioGroup rg = (RadioGroup) findViewById(R.id.rg1);
      String value
=((RadioButton)findViewById(rg.getCheckedRadioButtonId())).getText().toString();
      CheckBox c1 = (CheckBox)findViewById(R.id.c1);
      CheckBox c2 = (CheckBox)findViewById(R.id.c2);
      CheckBox c3 = (CheckBox)findViewById(R.id.c3);
      TextView tv1=(TextView)findViewById(R.id.t1);
      TextView tv2=(TextView)findViewById(R.id.t2);
      TextView tv3=(TextView)findViewById(R.id.t3);
      if(c1.isChecked()){
         tv1.setText("BOLD");
         tv1.setTypeface(Typeface.defaultFromStyle(Typeface.BOLD));
         if(value.equals("Left"))
            tv1.setGravity(Gravity.LEFT);
         else if(value.equals("Right"))
            tv1.setGravity(Gravity.RIGHT);
         else tv1.setGravity(Gravity.CENTER);
      }
      else tv1.setText("");
      if(c2.isChecked()){ tv2.setText("ITALIC");
         tv2.setTypeface(Typeface.defaultFromStyle(Typeface.ITALIC));
         if(value.equals("Left"))
            tv2.setGravity(Gravity.LEFT);
         else if(value.equals("Right"))
            tv2.setGravity(Gravity.RIGHT);
         else tv2.setGravity(Gravity.CENTER);
      }
      else tv2.setText("");
      if(c3.isChecked()){
         tv3.setText("NORMAL");
      if(value.equals("Left"))
         tv3.setGravity(Gravity.LEFT);
```

```
        else if(value.equals("Right"))
            tv3.setGravity(Gravity.RIGHT);
        else tv3.setGravity(Gravity.CENTER);
        }
        else tv3.setText("");
    }
}
```

Output



Practice Exercise:

1) Create an android app design scientific calculator
2) Create an android app to change font sizes, foreclors and backcolors (use RadioButton, CheckBox controls)

Week-4 (a):

Aim: Create an android app to display name of the country from the list(Usespinner control)

Main.xml

```
<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.administrator.spinner.MainActivity">

   <Spinner android:id="@+id/spinner"
      android:layout_width="149dp"
      android:layout_height="40dp"
      android:layout_marginBottom="8dp"
      android:layout_marginEnd="8dp"
      android:layout_marginStart="8dp"
      android:layout_marginTop="8dp"
      app:layout_constraintBottom_toBottomOf="parent"
      app:layout_constraintEnd_toEndOf="parent"
      app:layout_constraintHorizontal_bias="0.502"
      app:layout_constraintStart_toStartOf="parent"
      app:layout_constraintTop_toTopOf="parent"
      app:layout_constraintVertical_bias="0.498"/>
</android.support.constraint.ConstraintLayout>
```

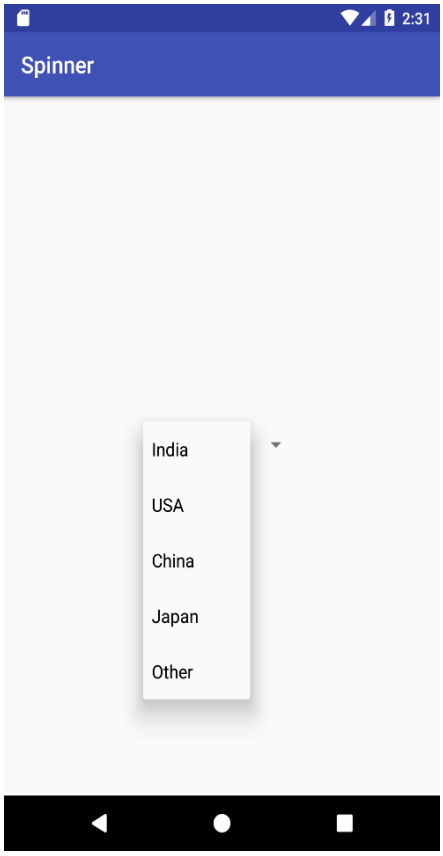
MainActivity.java

```
import android.support.v7.app.AppCompatActivity;import
android.os.Bundle;
import android.view.View;
import android.widget.AdapterView;
import android.widget.ArrayAdapter;
import android.widget.Spinner; import
android.widget.Toast;

public class MainActivity extends AppCompatActivity implements
```

```
AdapterView.OnItemSelectedListener {
   String[] country = { "India", "USA", "China", "Japan", "Other"};
   @Override
   protected void onCreate(Bundle savedInstanceState) {
      super.onCreate(savedInstanceState);
   setContentView(R.layout.activity_main);
      //Getting the instance of Spinner and applying OnItemSelectedListener on it
      Spinner spin = (Spinner) findViewById(R.id.spinner);
      spin.setOnItemSelectedListener(this);

      //Creating the ArrayAdapter instance having the country list
      ArrayAdapter aa = new ArrayAdapter(this,android.R.layout.simple_spinner_item,country);

      aa.setDropDownViewResource(android.R.layout.simple_spinner_dropdown_item);
      //Setting the ArrayAdapter data on the Spinner
      spin.setAdapter(aa);

   }
   //Performing action onItemSelected and onNothing selected
   @Override
   public void onItemSelected(AdapterView<?> arg0, View arg1, int position, longid) {
      Toast.makeText(getApplicationContext(),country[position] ,Toast.LENGTH_LONG).show();
   }
   @Override
   public void onNothingSelected(AdapterView<?> arg0) {
      // TODO Auto-generated method stub
```

```
    }
}
```

Output


2:31
Spinner
India
USA
China
Japan
Other


2:32
Spinner
China
China

Week-4 (b):

Aim: Create an android app to calculate age of a person (Use DatePickercontrol)

Activity_main.xml

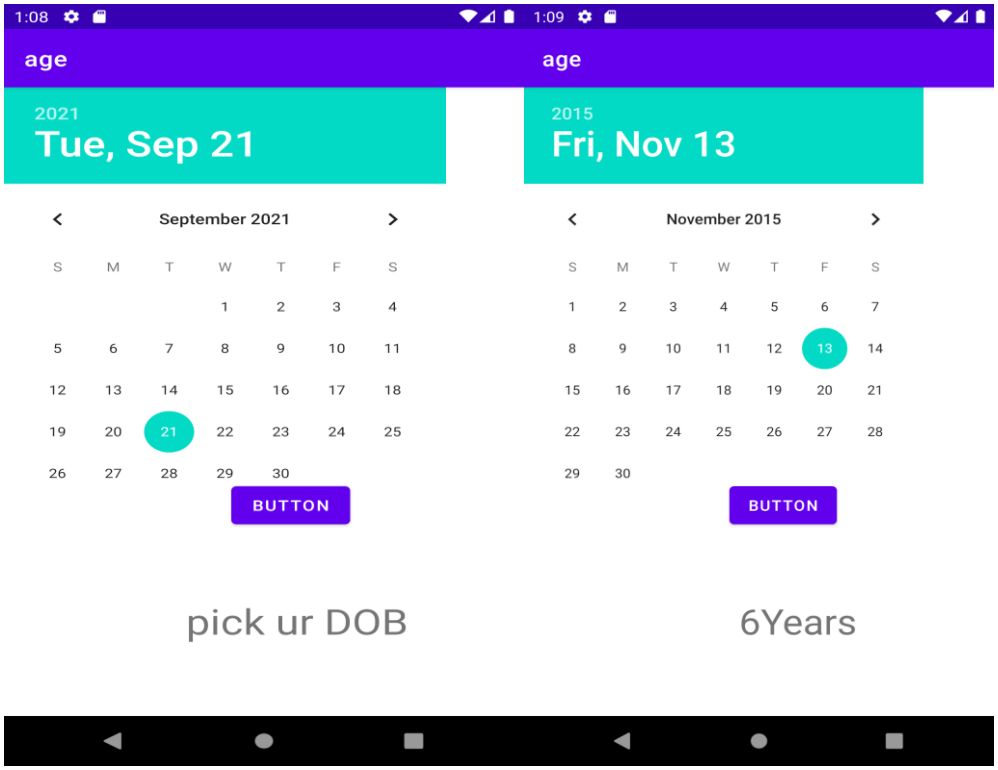
```
<?xml version="1.0" encoding="utf-8"?>
<androidx.constraintlayout.widget.ConstraintLayout
xmlns:android="http://schemas.android.com/apk/res/android"
   xmlns:app="http://schemas.android.com/apk/res-auto"
   xmlns:tools="http://schemas.android.com/tools"
   android:layout_width="match_parent"
   android:layout_height="match_parent"
   tools:context=".MainActivity">
   <DatePicker
      android:id="@+id/dp"
      android:layout_width="wrap_content"
      android:layout_height="wrap_content"
      tools:ignore="MissingConstraints"
      tools:layout_editor_absoluteX="0dp"
      tools:layout_editor_absoluteY="-197dp" />
   <TextView
      android:id="@+id/textView1"
      android:layout_width="wrap_content"
      android:layout_height="wrap_content"
      android:text="TextView"
      android:textSize="34sp"
      app:layout_constraintBottom_toBottomOf="parent"
      app:layout_constraintEnd_toEndOf="parent"
      app:layout_constraintHorizontal_bias="0.611"
      app:layout_constraintStart_toStartOf="parent"
      app:layout_constraintTop_toTopOf="parent"
      app:layout_constraintVertical_bias="0.877" />
   <Button
      android:id="@+id/button"
      android:layout_width="wrap_content"
      android:layout_height="wrap_content"
      android:text="Button"
      app:layout_constraintBottom_toBottomOf="parent"
      app:layout_constraintEnd_toEndOf="parent"
      app:layout_constraintHorizontal_bias="0.567"
      app:layout_constraintStart_toStartOf="parent"
      app:layout_constraintTop_toTopOf="parent"
      app:layout_constraintVertical_bias="0.679" />
</androidx.constraintlayout.widget.ConstraintLayout>
```

MainActivity.java

```
package com.vardhaman.age;
import androidx.appcompat.app.AppCompatActivity;
import android.os.Bundle;
import android.view.View;
```

```
import android.widget.Button;
import android.widget.DatePicker;
import android.widget.TextView;
public class MainActivity extends AppCompatActivity {
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);
        DatePicker picker1;
        Button displayDate;
        TextView textview1;
        textview1=(TextView)findViewById(R.id.textView1);
        picker1=(DatePicker)findViewById(R.id.dp);
        displayDate=(Button)findViewById(R.id.button);
        int i=picker1.getYear();
        textview1.setText("pick ur DOB");
        displayDate.setOnClickListener(new View.OnClickListener()
        {
            public void onClick(View view)
            {
                int j;
                j=picker1.getYear();
                textview1.setText(String.valueOf(i-j)+"Years");
            }
        });
    }
}
```

Output:



Week-4(c):

Aim: Create an android app design login control and validate login details

Activity_main.xml

```
<?xml version="1.0" encoding="utf-8"?>
<androidx.constraintlayout.widget.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=".MainActivity">
    <Button
        android:id="@+id/button"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Validate"
        app:layout_constraintBottom_toBottomOf="parent"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintHorizontal_bias="0.426"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toTopOf="parent"
        app:layout_constraintVertical_bias="0.619" />
    <EditText
        android:id="@+id/un"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:ems="10"
        android:inputType="textPersonName"
        app:layout_constraintBottom_toBottomOf="parent"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintHorizontal_bias="0.353"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toTopOf="parent"
        app:layout_constraintVertical_bias="0.212" />
    <EditText
        android:id="@+id/pass"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:ems="10"
        android:inputType="textPersonName"
        app:layout_constraintBottom_toBottomOf="parent"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintHorizontal_bias="0.353"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toTopOf="parent"
        app:layout_constraintVertical_bias="0.403" />
</androidx.constraintlayout.widget.ConstraintLayout>

 </RelativeLayout>
```

MainActivity.java

```
package com.vardhaman.login;
import androidx.appcompat.app.AppCompatActivity;
import android.content.Intent;
import android.os.Bundle;
import android.view.View;
import android.widget.Button;
import android.widget.EditText;
import android.widget.Toast;
public class MainActivity extends AppCompatActivity {
Button b;
EditText e1,e2;
String w="vardhaman";
String p="it";
   @Override
   protected void onCreate(Bundle savedInstanceState) {
      super.onCreate(savedInstanceState);
      setContentView(R.layout.activity_main);
      b=findViewById(R.id.button);
      e1=findViewById(R.id.un);
      e2=findViewById(R.id.pass);
      b.setOnClickListener(new View.OnClickListener() {
      public void onClick(View V) {
      String un=e1.getText().toString();
      String pass=e2.getText().toString();
      if(w.equals(un)&&(p.equals(pass)))
         {
         Toast.makeText(getApplicationContext(),"success"+un, Toast.LENGTH_LONG).show();
         Intent i=new Intent(getApplicationContext(),secondactivity.class);
         startActivity(i);
          }
          else
         {
            Toast.makeText(getApplicationContext(),"Authentication failed",
Toast.LENGTH_LONG).show();
         }
      }
      });
   }
}
```

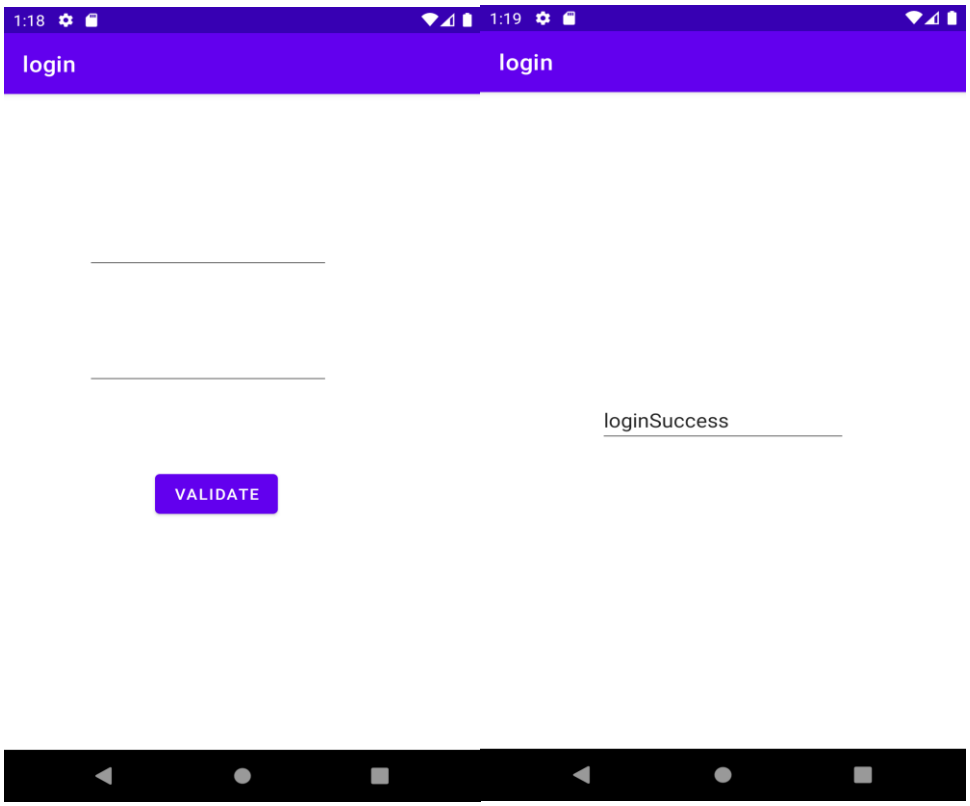
activity_secondactivity.xml

```
<?xml version="1.0" encoding="utf-8"?>
<androidx.constraintlayout.widget.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=".secondactivity">
   <EditText
      android:id="@+id/s"
      android:layout_width="wrap_content"
      android:layout_height="wrap_content"
      android:ems="10"
      android:inputType="textPersonName"
      android:text="loginSuccess"
      app:layout_constraintBottom_toBottomOf="parent"
      app:layout_constraintEnd_toEndOf="parent"
      app:layout_constraintStart_toStartOf="parent"
      app:layout_constraintTop_toTopOf="parent" />
</androidx.constraintlayout.widget.ConstraintLayout>
```

secondactivity.java

```
package com.vardhaman.login;
import androidx.appcompat.app.AppCompatActivity;
import android.os.Bundle;
public class secondactivity extends AppCompatActivity {
   @Override
   protected void onCreate(Bundle savedInstanceState) {
      super.onCreate(savedInstanceState);
      setContentView(R.layout.activity_secondactivity);
   }
}
```

Output:



Practice Exercise:

1) Create an android app to display the names of the India states (Use spinner control)
2) Create an android app to print date, month, year, day of selected date (Use DatePicker control)

Week-5 (a):

Aim: Create an android app to demonstrate AlertDialog

Activity_main.xml

```
<?xml version="1.0" encoding="utf-8"?>
<androidx.constraintlayout.widget.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=".MainActivity">
   <Button android:layout_width="wrap_content"
      android:layout_height="wrap_content"
      android:id="@+id/button"
      android:text="Close app"
      app:layout_constraintBottom_toBottomOf="parent"
      app:layout_constraintLeft_toLeftOf="parent"
      app:layout_constraintRight_toRightOf="parent"
      app:layout_constraintTop_toTopOf="parent" />
</androidx.constraintlayout.widget.ConstraintLayout>
```

strings.xml

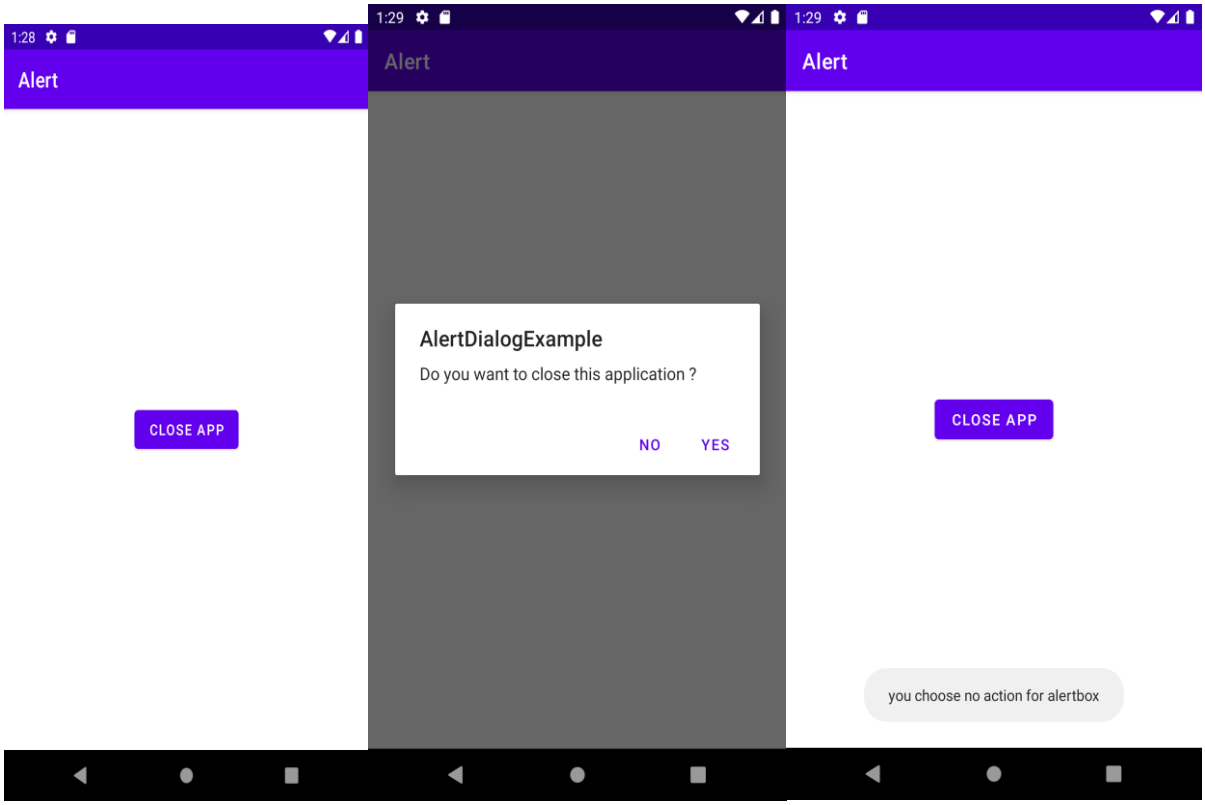
```
<resources>
  <string name="app_name">AlertDialogExample</string>
  <string name="dialog_message">Welcome to Alert Dialog</string>
  <string name="dialog_title">Alert</string>
</resources>
```

MainActivity.java

```
package com.vardhaman.alert;
import androidx.appcompat.app.AppCompatActivity;
import android.app.AlertDialog;
import android.content.DialogInterface;
import android.os.Bundle;
import android.view.View;
import android.widget.Button;
import android.widget.Toast;
public class MainActivity extends AppCompatActivity {
   Button closeButton;
   AlertDialog.Builder builder;
   @Override
   protected void onCreate(Bundle savedInstanceState) {
      super.onCreate(savedInstanceState);
      setContentView(R.layout.activity_main);
      closeButton = (Button) findViewById(R.id.button);
      builder = new AlertDialog.Builder(this);
      closeButton.setOnClickListener(new View.OnClickListener() {
         @Override
         public void onClick(View v) {
```

```
            builder.setMessage("Do you want to close this application ?")
                  .setCancelable(false)
                  .setPositiveButton("Yes", new DialogInterface.OnClickListener() {
                     public void onClick(DialogInterface dialog, int id) {
                        finish();
                        Toast.makeText(getApplicationContext(),"you choose yes action for alertbox",
                              Toast.LENGTH_SHORT).show();
                     }
                  })
                  .setNegativeButton("No", new DialogInterface.OnClickListener() {
                     public void onClick(DialogInterface dialog, int id) {
                       dialog.cancel();
                        Toast.makeText(getApplicationContext(),"you choose no action for alertbox",
                              Toast.LENGTH_SHORT).show();
                     }
                  });
            AlertDialog alert = builder.create();
            alert.setTitle("AlertDialogExample");
            alert.show();
         }
      });
   }
}
```

Output:



Week-5 (b):

Aim: Create an android app to demonstrate WebView control

Activity_main.xml

```
<?xml version="1.0" encoding="utf-8"?>
<androidx.constraintlayout.widget.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=".MainActivity">
    <WebView
        android:id="@+id/webView"
        android:layout_width="match_parent"
        android:layout_height="match_parent"
        app:layout_constraintBottom_toBottomOf="parent"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toTopOf="parent"/>
</androidx.constraintlayout.widget.ConstraintLayout>
```

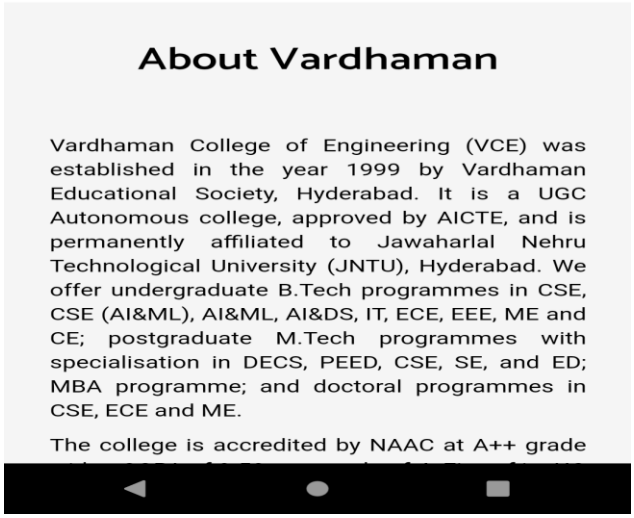
MainActivity.java

```
package com.vardhaman.webview;
import androidx.appcompat.app.AppCompatActivity;
import android.os.Bundle;
import android.webkit.WebView;
public class MainActivity extends AppCompatActivity {
    @Override
 protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);
        WebView mywebview = (WebView) findViewById(R.id.webView);
        //String data = "<html><body><h1>Welcome to Vardhaman</h1><h2>IT
Department</h2></body></html>";
        //mywebview.loadUrl(data);
        //mywebview.loadData(data,null,null);
         String url="https://vardhaman.org/";
       mywebview.loadUrl(url);
    }
}
```

AndroidManifest.xml

```
<?xml version="1.0" encoding="utf-8"?>
<manifest xmlns:android="http://schemas.android.com/apk/res/android"
    package="com.vardhaman.webview">
<uses-permission android:name="android.permission.INTERNET"/>
    <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/Theme.Webview">
        <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>
```

Output:



Practice Exercise:

1) Create an android app to show alertdialog with different buttons and messages
2) Create an android app to display specified webpage using webview

Week-6 (a):

Aim: Create an android app to show Analog and Digital clocks

activity_main.xml

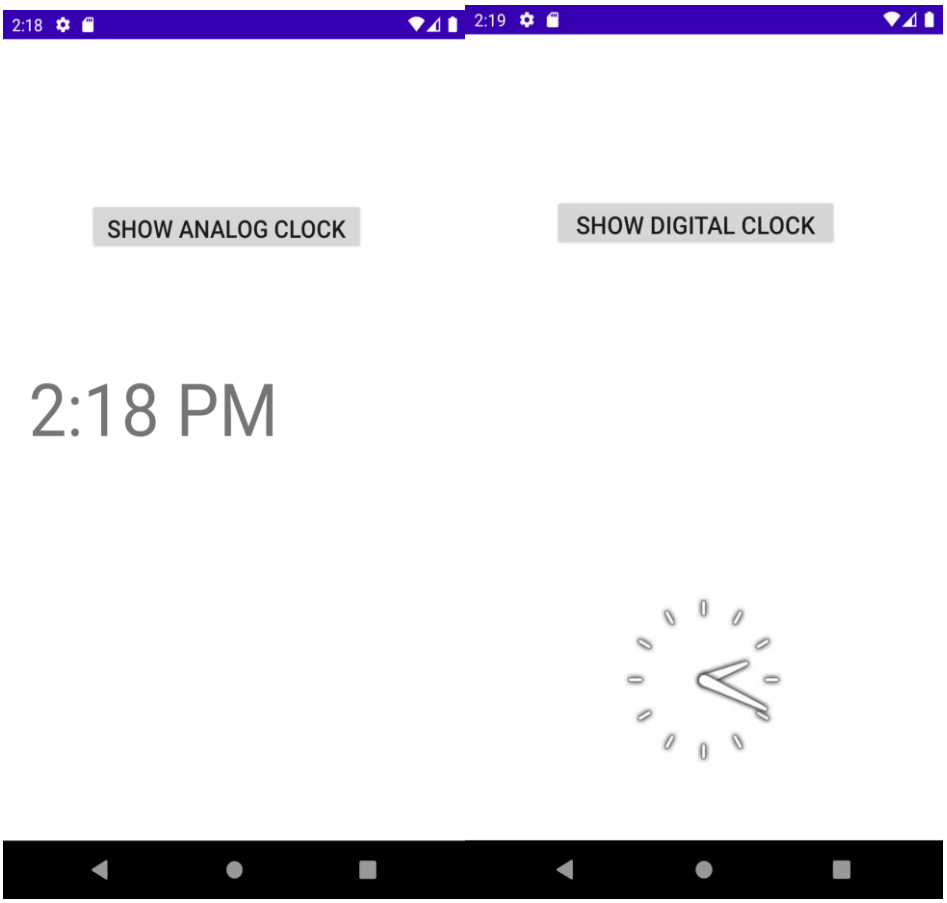
```
<?xml version="1.0" encoding="utf-8"?>
<androidx.constraintlayout.widget.ConstraintLayout
   xmlns:android="http://schemas.android.com/apk/res/android"
   xmlns:app="http://schemas.android.com/apk/res-auto"
   xmlns:tools="http://schemas.android.com/tools"
   android:layout_width="match_parent"
   android:layout_height="match_parent"
   tools:context=".MainActivity">
   <Button
      android:id="@+id/btn"
      android:layout_width="244dp"
      android:layout_height="43dp"
      android:layout_gravity="center"
      android:textSize="20sp"
      app:layout_constraintBottom_toBottomOf="parent"
      app:layout_constraintEnd_toEndOf="parent"
      app:layout_constraintHorizontal_bias="0.458"
      app:layout_constraintStart_toStartOf="parent"
      app:layout_constraintTop_toTopOf="parent"
      app:layout_constraintVertical_bias="0.215" />
   <DigitalClock
      android:id="@+id/dclock"
      android:layout_width="366dp"
      android:layout_height="134dp"
      android:layout_gravity="center"
      android:textSize="60sp"
      app:layout_constraintBottom_toBottomOf="parent"
      app:layout_constraintEnd_toEndOf="parent"
      app:layout_constraintStart_toStartOf="parent"
      app:layout_constraintTop_toTopOf="parent" />
   <AnalogClock
      android:id="@+id/aclock"
      android:layout_width="294dp"
      android:layout_height="156dp"
      android:layout_gravity="center"
      app:layout_constraintBottom_toBottomOf="parent"
      app:layout_constraintEnd_toEndOf="parent"
      app:layout_constraintStart_toStartOf="parent"
      app:layout_constraintTop_toBottomOf="@+id/dclock" />
   </androidx.constraintlayout.widget.ConstraintLayout>
```

MainActivity.java

```
import  android.os.Bundle;
import android.view.View;
import android.widget.Button;
import
```

```
android.widget.TextClock;
import android.widget.AnalogClock;import
android.app.Activity;
public class MainActivity extends
  Activity {TextClock clock;
  AnalogClock
  aclock;Button
  btn; @Override
  protected void onCreate(Bundle
     savedInstanceState) {
     super.onCreate(savedInstanceState);
     setContentView(R.layout.activity_main);
     clock = (TextClock)findViewById(R.id.clock);
     aclock =  (AnalogClock)findViewById(R.id.aclock);
     btn = (Button)findViewById(R.id.btn);
     btn.setText("Show Analog Clock");
     aclock.setVisibility(View.INVISIBLE);
     btn.setOnClickListener(new View.OnClickListener()
     {
        @Override
        public void onClick(View arg0) {
           // TODO Auto-generated method stub
           btn.setText("Show Digital Clock");
           aclock.setVisibility(View.VISIBLE);
           clock.setVisibility(View.INVISIBLE);
           btn.setOnClickListener(new
           View.OnClickListener() {
              @Override
              public void onClick(View arg0) {
                 // TODO Auto-generated method stub
                 onCreate(null);


              }
           });
        }
     });
  }
}
```

Output:



Week-6 (b):

Aim: Create an android app to illustrate a progressbar

Activity_main.xml

```
<?xml version="1.0" encoding="utf-8"?>
<RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"
  xmlns:tools="http://schemas.android.com/tools"
  android:layout_width="match_parent"
  android:layout_height="match_parent"
  tools:context=".MainActivity" >

  <ProgressBar
    android:id="@+id/progressBar"
    style="?android:attr/progressBarStyleHorizontal"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_alignParentLeft="true"
    android:layout_alignParentTop="true"
    android:layout_marginLeft="23dp"
    android:layout_marginTop="20dp"
    android:indeterminate="false"
     android:max="100"
    android:minHeight="50dp"
    android:minWidth="200dp"
    android:progress="1" />

  <ProgressBar android:id="@+id/progressBar_cyclic"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:minHeight="50dp"
    android:minWidth="50dp"
    android:layout_centerVertical="true"
    android:layout_centerHorizontal="true" />
  <TextView android:id="@+id/textView"
```

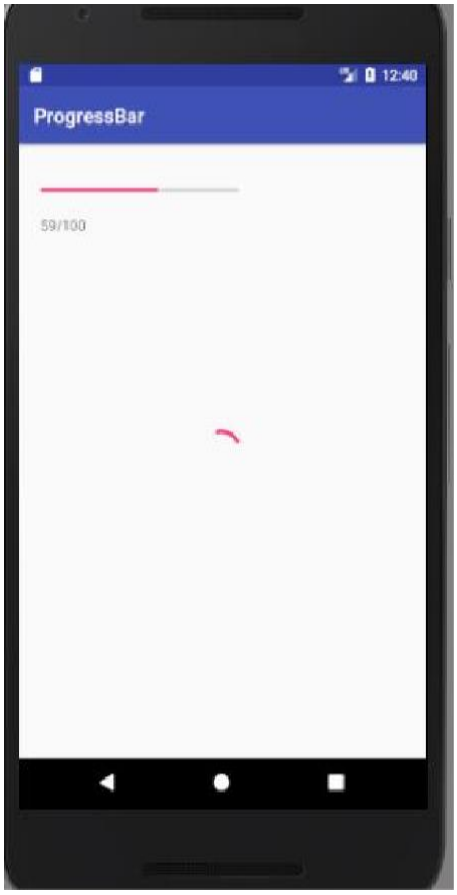
```
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
     android:layout_alignLeft="@+id/progressBar"
    android:layout_below="@+id/progressBar"/>
</RelativeLayout>
```

MainActivity.java

```
import android.os.Handler;
import android.support.v7.app.AppCompatActivity;import
android.os.Bundle;
import android.widget.ProgressBar;
import android.widget.TextView;

public class MainActivity extends AppCompatActivity {private
  ProgressBar progressBar;
  private int progressStatus = 0;
  private TextView textView;
  private Handler handler = new Handler();

  @Override
  protected void onCreate(Bundle savedInstanceState) {super.onCreate(savedInstanceState);
    setContentView(R.layout.activity_main);
    progressBar = (ProgressBar) findViewById(R.id.progressBar);
    textView = (TextView) findViewById(R.id.textView);
    // Start long running operation in a background thread
    new Thread(new Runnable() {
      public void run() {
        while (progressStatus < 100) {
          progressStatus += 1;
```

```
                    // Update the progress bar and display the
                    //current value in the text view
                    handler.post(new
                       Runnable() {
                       public void run()
                       {
                          progressBar.setProgress(progressStatus);
                          textView.setText(progressStatus+"/"+progressBar.getMax());
   }
});
                    try {
                       // Sleep for 200 milliseconds.
                       Thread.sleep(200);
                    } catch
                       (InterruptedExcepti
                       on e) {
                       e.printStackTrace();
                    }
                 }
              }
           }).start();
        }
     }
```

Output:



Practice Exercise:

1) Create an android app to display the progressbar status on Textview widget
2) Create an android app to display hour, minutes, seconds using both analog anddigital widgets

Week-7

a. Create an android app to demonstrate list fragment

Activitymain.xml

```
<?xml version="1.0" encoding="utf-8"?>
    <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"
      tools:context=".MainActivity">
      <fragment
        android:layout_height="match_parent"
        android:layout_width="350px"
        class="com.vardhaman.fragments.ListMenuFragment"
        android:id="@+id/fragment"/>
      <fragment
        android:layout_width="match_parent"
        android:layout_height="match_parent"
        class="com.vardhaman.fragments.DetailsFragment"
        android:id="@+id/fragment2"/>
    </LinearLayout>
```

Detailinfo.xml

```
<?xml version="1.0" encoding="utf-8"?>
    <LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
      android:orientation="vertical"
      android:layout_width="match_parent"
      android:layout_height="match_parent">
      <TextView
        android:id="@+id/item"
        android:layout_width="297dp"
        android:layout_height="34dp"
        android:layout_marginLeft="200px"
        android:layout_marginTop="200px"
        android:textColor="#CD5C5C" />

      <TextView
        android:id="@+id/price"
        android:layout_width="282dp"
        android:layout_height="64dp"
        android:layout_marginLeft="200px"
        android:textColor="#6495ED" />

    </LinearLayout>
```

Listitemsinfo.xml

```
<?xml version="1.0" encoding="utf-8"?>
    <LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
      android:orientation="vertical"
      android:layout_width="match_parent"
      android:layout_height="match_parent">
      <ListView
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:id="@android:id/list" />
    </LinearLayout>
```

Mainactivity.java

```
package com.vardhaman.fragments;
    import androidx.appcompat.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);
      }
    }
```

listmenufragment.java

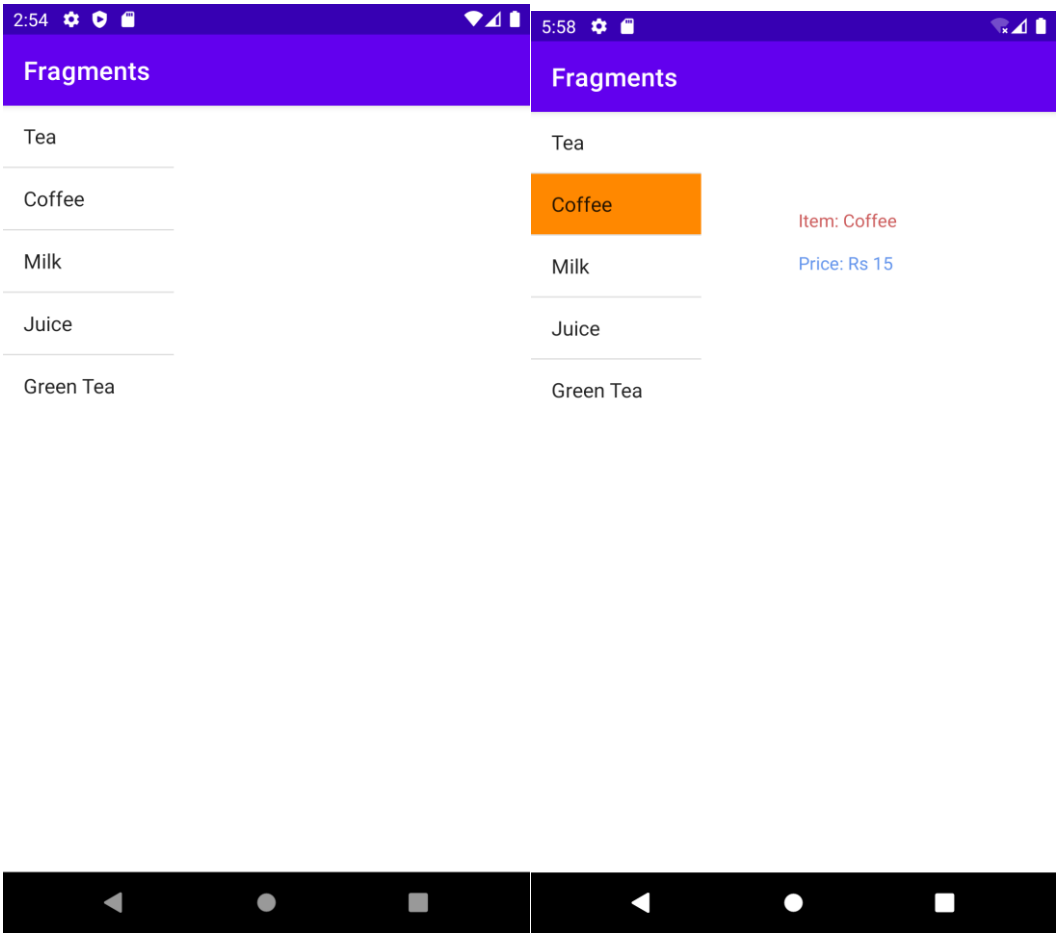
```
    package com.vardhaman.fragments;
    import android.app.ListFragment;
    import android.os.Bundle;
    import android.view.LayoutInflater;
    import android.view.View;
    import android.view.ViewGroup;
    import android.widget.ArrayAdapter;
    import android.widget.ListView;
    public class ListMenuFragment extends ListFragment {
      String[] items = new String[] { "Tea","Coffee","Milk","Juice","Green Tea" };
      String[] prices = new String[]{"Rs 10","Rs 15","Rs 20","Rs 30","Rs 10"};
      @Override
      public View onCreateView(LayoutInflater inflater, ViewGroup container,
                  Bundle savedInstanceState) {
        View view =inflater.inflate(R.layout.listitemsinfo, container,
            false);
        ArrayAdapter<String> adapter = new ArrayAdapter<String>(getActivity(),
            android.R.layout.simple_list_item_1, items);
        setListAdapter(adapter);
        return view;
      }
      @Override
      public void onListItemClick(ListView l, View v, int position, long id) {
```

```
            DetailsFragment txt = (DetailsFragment)getFragmentManager().
                findFragmentById(R.id.fragment2);
            txt.change("Item: "+ items[position],"Price: "+
                prices[position]);
            getListView().setSelector(android.R.color.holo_orange_dark);
        }
    }
```

detailsfragment.java

```
package com.vardhaman.fragments;
    import android.app.Fragment;
    import android.os.Bundle;
    import android.view.LayoutInflater;
    import android.view.View;
    import android.view.ViewGroup;
    import android.widget.TextView;
    public class DetailsFragment extends Fragment {
        TextView item, price;
        @Override
        public View onCreateView(LayoutInflater inflater, ViewGroup container,
                        Bundle savedInstanceState) {
            View view = inflater.inflate(R.layout.detailsinfo, container,
                false);
            item = (TextView) view.findViewById(R.id.item);
            price = (TextView) view.findViewById(R.id.price);
            return view;
        }
        public void change(String items, String prices) {
            item.setText(items);
            price.setText(prices);
        }
    }
```

OUTPUT


2:54
Fragments
Tea
Coffee
Milk
Juice
Green Tea
5:58
Fragments
Tea
Coffee
Milk
Juice
Green Tea
Item: Coffee
Price: Rs 15

b. Create an android app to demonstrate dialog fragment

activitymain.xml

```
<?xml version="1.0" encoding="utf-8"?>
    <androidx.constraintlayout.widget.ConstraintLayout
      xmlns:android="http://schemas.android.com/apk/res/android"
      xmlns:tools="http://schemas.android.com/tools"
      android:layout_width="match_parent"
      android:layout_height="match_parent"
      tools:context=".MainActivity">
      <FrameLayout
        android:layout_width="match_parent"
        android:layout_height="match_parent"
        android:orientation="horizontal">

        <Button
          android:id="@+id/b1"
          android:layout_width="match_parent"
          android:layout_height="70dp"
          android:layout_margin="100dp"
          android:text="open fragment"></Button>
      </FrameLayout>
    </androidx.constraintlayout.widget.ConstraintLayout>
```

dialogfrag.xml

```
<?xml version="1.0" encoding="utf-8"?>
    <androidx.constraintlayout.widget.ConstraintLayout
    xmlns:android="http://schemas.android.com/apk/res/android"
      android:layout_width="match_parent"
      android:layout_height="match_parent">
      <LinearLayout
        android:layout_width="300dp"
        android:layout_height="400dp"
        android:orientation="horizontal">
        <TextView
          android:layout_width="317dp"
          android:layout_height="412dp"
          android:background="#ccff99"
          android:text="You are in Dialog Fragment"
          android:textSize="50dp" />
      </LinearLayout>
    </androidx.constraintlayout.widget.ConstraintLayout>
```

mainactivity.java

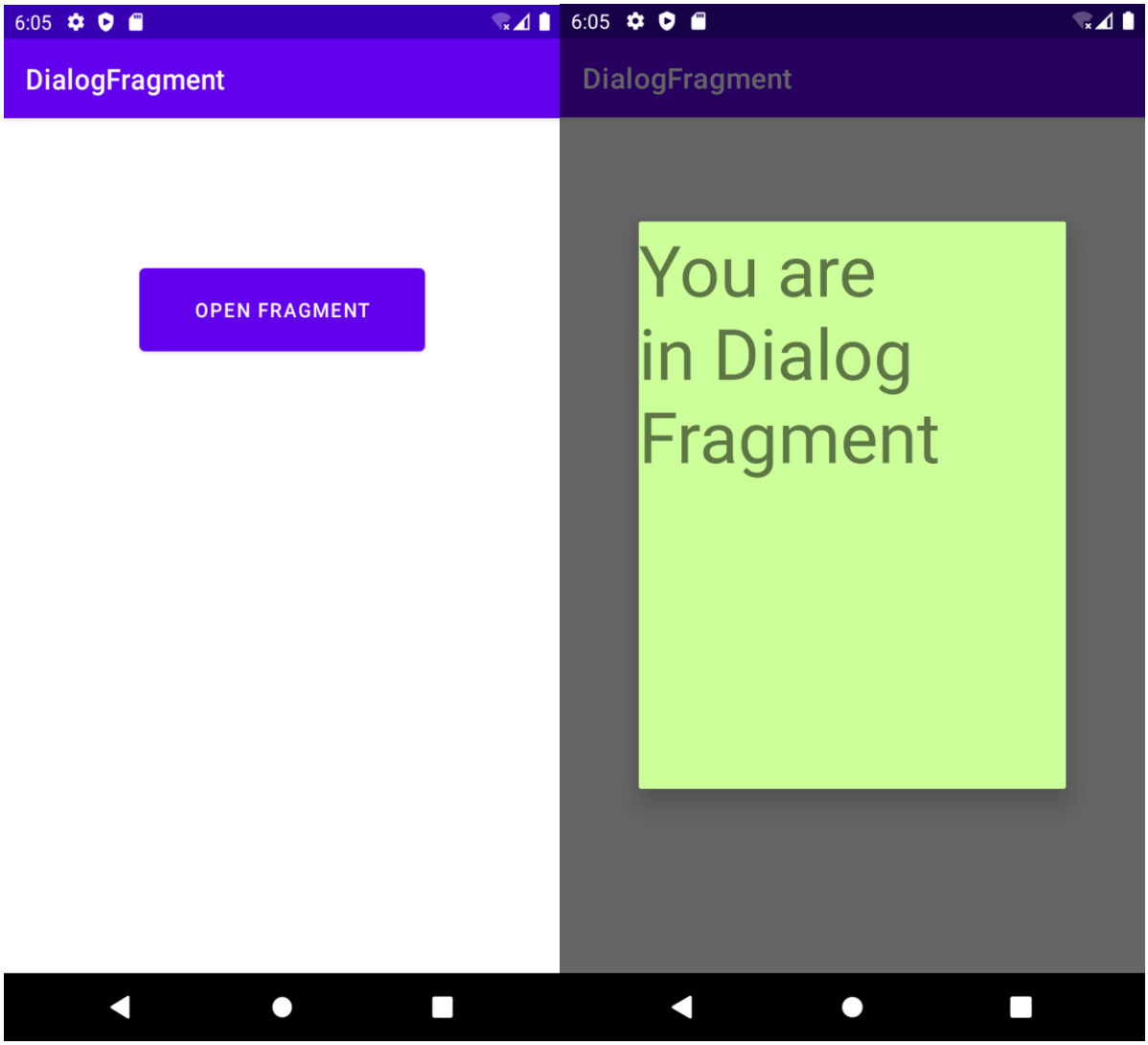
```
    package com.vardhaman.dialogfragment;
    import androidx.appcompat.app.AppCompatActivity;
    import android.os.Bundle;
    import android.view.View;
    import android.widget.Button;
```

```
public class MainActivity extends AppCompatActivity {
Button button;
  @Override
  protected void onCreate(Bundle savedInstanceState) {
    super.onCreate(savedInstanceState);
    setContentView(R.layout.activity_main);
    button=findViewById(R.id.b1);
    button.setOnClickListener(new View.OnClickListener() {
      @Override
      public void onClick(View v) {
        MyDialogFragment one=new MyDialogFragment();
        one.show(getSupportFragmentManager(),"Myfragment");
      }
    });
  }
}
```

mydialogfragment.java

```
package com.vardhaman.dialogfragment;
  import android.os.Bundle;
  import android.view.LayoutInflater;
  import android.view.View;
  import android.view.ViewGroup;
  import androidx.fragment.app.DialogFragment;
  public class MyDialogFragment extends DialogFragment {
      public View onCreateView(LayoutInflater inflater,
                  ViewGroup container,
                  Bundle SavedInstanceState) {
       super.onCreateView(inflater, container, SavedInstanceState);
    return inflater.inflate(R.layout.dialog_frag,container,  false);
    }
  }
```

OUTPUT:


6:05
DialogFragment
OPEN FRAGMENT
6:05
DialogFragment
You are
in Dialog
Fragment

Week-8

a. Create an android app to demonstrate option menu, handling listeners

Activitymain.xml

```
<?xml version="1.0" encoding="utf-8"?>
    <androidx.constraintlayout.widget.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=".MainActivity">
</androidx.constraintlayout.widget.ConstraintLayout>
```

Listmenu.xml

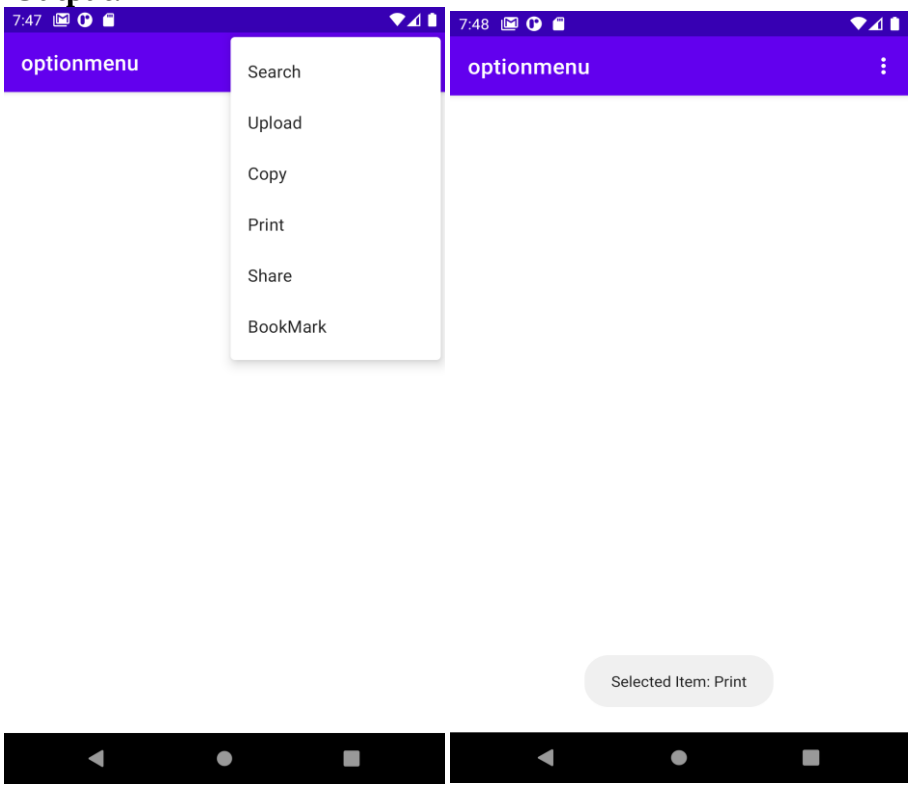
```
<?xml version="1.0" encoding="utf-8"?>
    <menu xmlns:android="http://schemas.android.com/apk/res/android">
      <item android:id="@+id/search_item"
        android:title="Search" />
      <item android:id="@+id/upload_item"
        android:title="Upload" />
      <item android:id="@+id/copy_item"
        android:title="Copy" />
      <item android:id="@+id/print_item"
        android:title="Print" />
      <item android:id="@+id/share_item"
        android:title="Share" />
      <item android:id="@+id/bookmark_item"
        android:title="BookMark" />
    </menu>
```

Mainactivity.java

```
package com.vardhaman.optionmenu;
import androidx.appcompat.app.AppCompatActivity;
import android.os.Bundle;
import android.view.Menu;
import android.view.MenuItem;
import android.widget.Toast;
public class MainActivity extends AppCompatActivity {
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);
    }
    public boolean onCreateOptionsMenu(Menu menu) {
        getMenuInflater().inflate(R.menu.list_menu, menu);
        return true;
    }
    @Override
```

```
    public boolean onOptionsItemSelected(MenuItem item) {
        Toast.makeText(this, "Selected Item: " +item.getTitle(),
                Toast.LENGTH_SHORT).show();
        switch (item.getItemId()) {
            case R.id.search_item:
                // do your code
                return true;
            case R.id.upload_item:
                // do your code
                return true;
            case R.id.copy_item:
                // do your code
                return true;
            case R.id.print_item:
                // do your code
                return true;
            case R.id.share_item:
                // do your code
                return true;
            case R.id.bookmark_item:
                // do your code
                return true;
            default:
                return super.onOptionsItemSelected(item);
        }
    }
}
```

Output:



b. Create an android app to scroll list of images and display details of images (name, size etc) usingImage Switcher control

Activity_main.xml

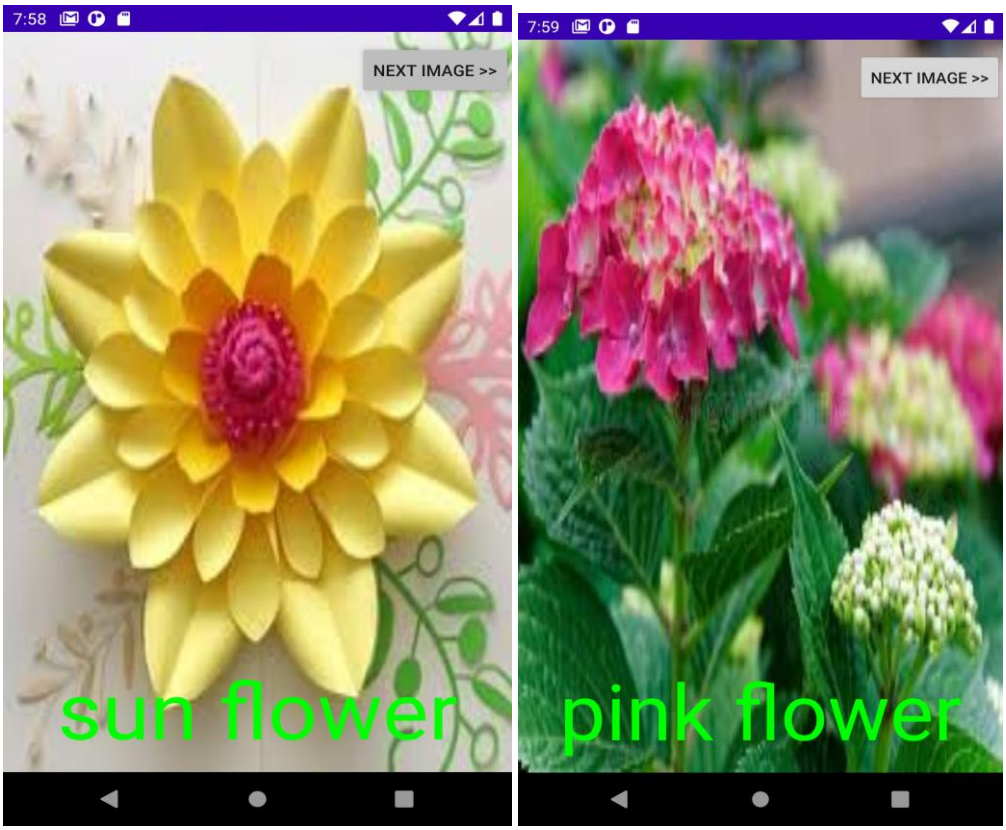
```
<?xml version="1.0" encoding="utf-8"?>
<RelativeLayout
  xmlns:android="http://schemas.android.com/apk/res/android"
  xmlns:app="http://schemas.android.com/apk/res-auto"
  xmlns:tools="http://schemas.android.com/tools"
  android:layout_width="match_parent"
  android:layout_height="match_parent"
  tools:context=".MainActivity">
  <ImageSwitcher
      android:id="@+id/imageSwitcher"
      android:layout_width="fill_parent"
      android:layout_height="fill_parent"
      android:layout_alignParentLeft="true" >
  </ImageSwitcher>
  <TextSwitcher
      android:id="@+id/textSwitcher"
      android:layout_width="fill_parent"
      android:layout_height="wrap_content"
      android:layout_alignParentBottom="true"
      android:layout_centerHorizontal="true"
      android:padding="10dp" />
  <Button
      android:layout_width="wrap_content"
      android:layout_height="wrap_content"
      android:layout_alignParentRight="true"
      android:layout_alignParentTop="true"
      android:layout_marginTop="10dp"
      android:onClick="onSwitch"
      android:text="Next Image >>" />
  </RelativeLayout>
```

Mainactivity.java

```
package com.vardhaman.image1;
import android.app.Activity;
import android.os.Bundle;
import android.view.Gravity;
import android.view.View;
import android.widget.ImageSwitcher;
import android.widget.ImageView;
import android.widget.TextSwitcher;
import android.widget.TextView;
import android.widget.ViewSwitcher;
public class MainActivity extends Activity {
  private static final String[] TEXTS = { "pink flower", "sun flower" };
  private static final int[] IMAGES = { R.drawable.red, R.drawable.yellow};
  private int mPosition = 0;
  private TextSwitcher mTextSwitcher;
  private ImageSwitcher mImageSwitcher;
  @Override
  public void onCreate(Bundle savedInstanceState) {
    super.onCreate(savedInstanceState);
    setContentView(R.layout.activity_main);
    mTextSwitcher = (TextSwitcher) findViewById(R.id.textSwitcher);
    mTextSwitcher.setFactory(new ViewSwitcher.ViewFactory() {
      @Override
      public View makeView() {
        TextView textView = new TextView(MainActivity.this);
        textView.setTextSize(70);
        textView.setGravity(Gravity.CENTER);
        textView.setTextColor(0xFF00FF00);
        return textView;
      }
    });
    mTextSwitcher.setInAnimation(this, android.R.anim.fade_in);
    mTextSwitcher.setOutAnimation(this, android.R.anim.fade_out);
    mImageSwitcher = (ImageSwitcher) findViewById(R.id.imageSwitcher);
    mImageSwitcher.setFactory(new ViewSwitcher.ViewFactory() {
      @Override
      public View makeView() {
```

```
            ImageView imageView = new ImageView(MainActivity.this);
            return imageView;
        }
    });
    mImageSwitcher.setInAnimation(this, android.R.anim.slide_in_left);
    mImageSwitcher.setOutAnimation(this, android.R.anim.slide_out_right);
    onSwitch(null);
  }
  public void onSwitch(View view) {
    mTextSwitcher.setText(TEXTS[mPosition]);
    mImageSwitcher.setBackgroundResource(IMAGES[mPosition]);
    mPosition = (mPosition + 1) % TEXTS.length;
  }
}
```

Output:



Practice Exercise:

1) Create an android app to create a simple notepad
2) Create an android app to create a image slide with animations

Week-9

a. **Create an android app to demonstrate sending e-mail**
   Activitymain.xml

```
<?xml version="1.0" encoding="utf-8"?>
<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">
   <EditText
      android:id="@+id/recipient"
      android:layout_width="wrap_content"
      android:layout_height="wrap_content"
      android:inputType="textEmailAddress"
      android:hint="Recipient email" />
   <EditText
      android:id="@+id/subject"
      android:layout_width="wrap_content"
      android:hint="Subject"
      android:layout_height="wrap_content" />
   <EditText
      android:id="@+id/Message"
      android:layout_width="match_parent"
      android:hint="Enter Message here"
      android:minHeight="150dp"
      android:gravity="start"
      android:layout_height="wrap_content" />
   <Button
      android:id="@+id/button"
      android:layout_width="match_parent"
      android:layout_height="wrap_content"
      android:gravity="end"
      android:text="Send Email" />
</LinearLayout>
```

Mainactivity.java

```
package com.vardhaman.email;
import androidx.appcompat.app.AppCompatActivity;
import android.content.Intent;
import android.net.Uri;
import android.os.Bundle;
import android.view.Menu;
import android.view.View;
import android.widget.Button;
import android.widget.EditText;
import android.widget.Toast;
```

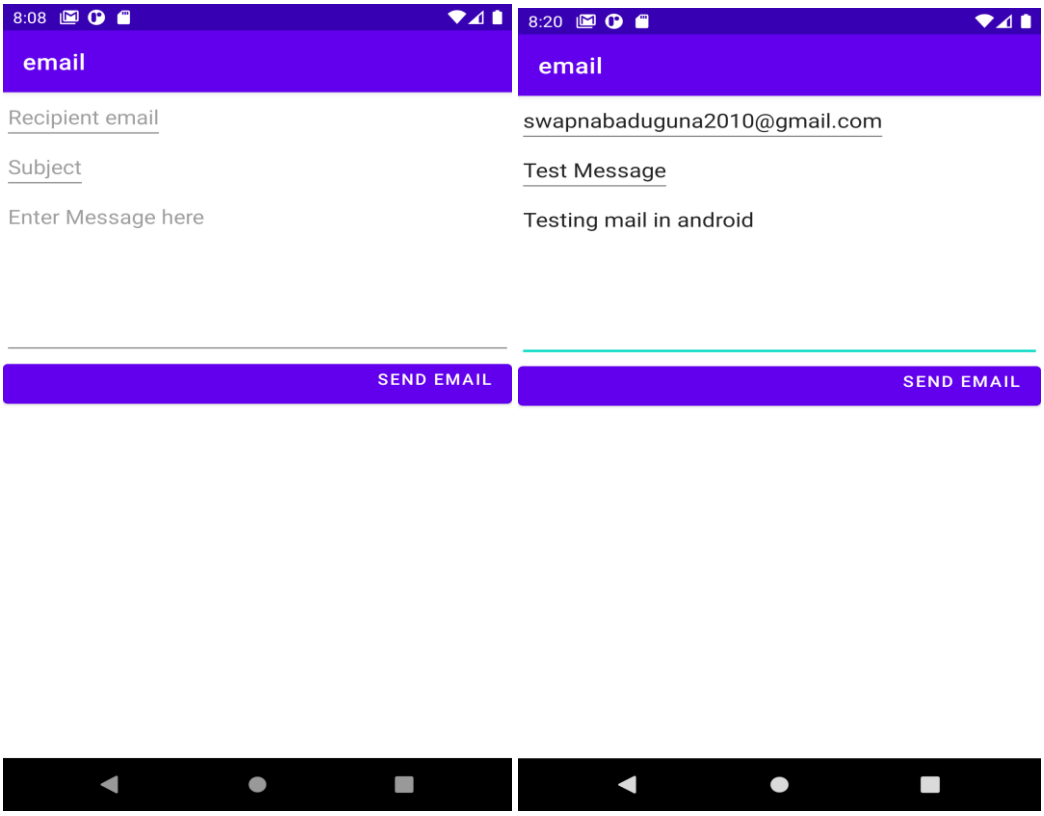
```
public class MainActivity extends AppCompatActivity {
   EditText editTextTo,editTextSubject,editTextMessage;
   Button send;
   @Override
   protected void onCreate(Bundle savedInstanceState) {
      super.onCreate(savedInstanceState);
      setContentView(R.layout.activity_main);
      editTextTo=(EditText)findViewById(R.id.recipient);
      editTextSubject=(EditText)findViewById(R.id.subject);
      editTextMessage=(EditText)findViewById(R.id.Message);
      send=(Button)findViewById(R.id.button);
      send.setOnClickListener(new View.OnClickListener() {
         @Override
         public void onClick(View v) {
            String re=editTextTo.getText().toString().trim();
            String su=editTextSubject.getText().toString().trim();
            String me=editTextMessage.getText().toString().trim();
            sendEmail(re,su,me);
         }
      });
   }
   private void sendEmail(String re, String su, String me) {
      Intent i=new Intent(Intent.ACTION_SEND);
      i.setData(Uri.parse("Mail to"));
      i.setType("text/plain");
      i.putExtra(Intent.EXTRA_EMAIL,new String[]{re});
      i.putExtra(Intent.EXTRA_SUBJECT,su);
      i.putExtra(Intent.EXTRA_TEXT,me);
      try{
         startActivity(Intent.createChooser(i,"choocse an email client"));
      }
      catch(Exception e){
         Toast.makeText(this,e.getMessage(),Toast.LENGTH_SHORT).show();
      }

      }
   }
```

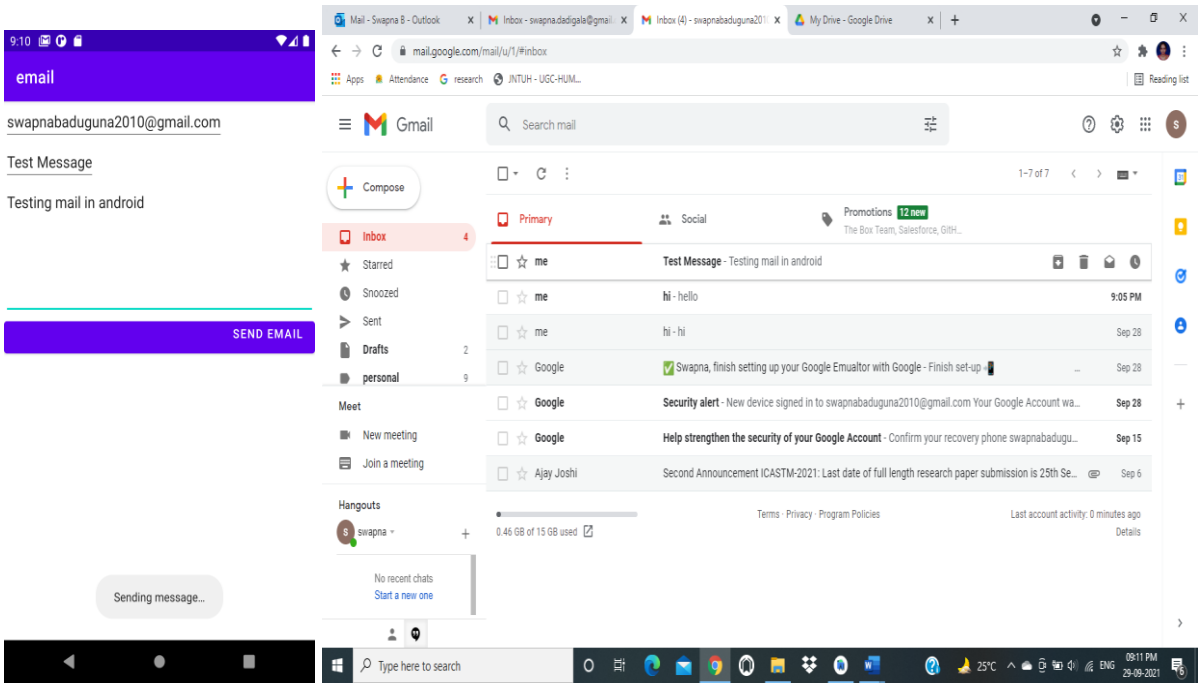
AndroidManifest.xml

```
<?xml version="1.0" encoding="utf-8"?>
<manifest xmlns:android="http://schemas.android.com/apk/res/android"
   package="com.vardhaman.email">
   <uses-permission android:name="android.permission.INTERNET"></uses-permission>
   <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/Theme.Email">
      <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>
```

Output:





b. Create an android app to demonstrate sending SMS

Activity_main.xml

```
<?xml version="1.0" encoding="utf-8"?>
<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:gravity="center_horizontal"
   android:orientation="vertical"
   android:padding="16dp"
   tools:context=".MainActivity">
   <EditText
      android:layout_width="match_parent"
      android:layout_height="wrap_content"
      android:id="@+id/phone"
      android:hint="enter phone number"
      android:inputType="phone"
      android:padding="12dp"
      android:maxLength="10"/>
   <EditText
      android:id="@+id/message"
      android:layout_width="wrap_content"
      android:layout_height="wrap_content"
      android:padding="16dp"
      android:ems="10"
      android:minLines="6"
      android:gravity="top"
      android:inputType="textMultiLine"
      android:hint="enter message"
      android:layout_margin="8dp"/>
   <Button
      android:layout_width="wrap_content"
      android:layout_height="wrap_content"
      android:id="@+id/send"
      android:text="send sms"
      android:layout_marginTop="30dp"/>
</LinearLayout>
```

MainActivity,java

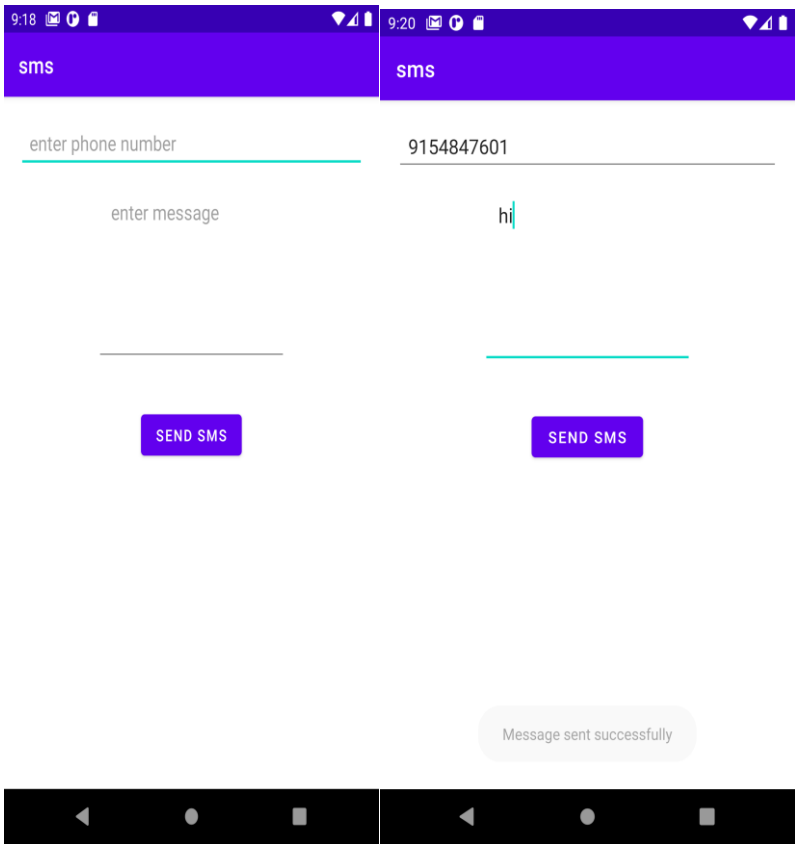
```
package com.vardhaman.sms;
import androidx.appcompat.app.AppCompatActivity;
import androidx.core.app.ActivityCompat;
import androidx.core.content.ContextCompat;
import android.Manifest;
import android.content.pm.PackageManager;
import android.os.Bundle;
import android.telephony.SmsManager;
import android.view.View;
import android.widget.Button;
import android.widget.EditText;
```

```
import android.widget.Toast;
public class MainActivity extends AppCompatActivity {
   EditText e1,e2;
   Button b;
   @Override
   protected void onCreate(Bundle savedInstanceState) {
      super.onCreate(savedInstanceState);
      setContentView(R.layout.activity_main);
      e1=findViewById(R.id.phone);
      e2=findViewById(R.id.message);
      b=findViewById(R.id.send);
      b.setOnClickListener(new View.OnClickListener() {
         @Override
         public void onClick(View v) {
            if (ContextCompat.checkSelfPermission(MainActivity.this,
Manifest.permission.SEND_SMS)
                  == PackageManager.PERMISSION_GRANTED) {
               sendMessage();
            } else {
               ActivityCompat.requestPermissions(MainActivity.this,
                     new String[]{Manifest.permission.SEND_SMS},
                     100);
            }
         }
      });
      }
      public void sendMessage() {
      String number=e1.getText().toString().trim();
      String sms=e2.getText().toString().trim();
      if((!number.equals(""))&&(!sms.equals("")))
      {
         SmsManager sm=SmsManager.getDefault();
         sm.sendTextMessage(number,null,sms,null,null);
         Toast.makeText(this, "Message sent successfully", Toast.LENGTH_SHORT).show();
      }else
      {
         Toast.makeText(getApplicationContext(),
               "enter number",Toast.LENGTH_LONG).show();      }
   }
   @Override
   public void onRequestPermissionsResult(int requestCode, String[] permissions,int[]
grantResults) {
      super.onRequestPermissionsResult(requestCode, permissions, grantResults);
   if(requestCode==100&&grantResults.length>0&&grantResults[0]==
   PackageManager.PERMISSION_GRANTED)
   {sendMessage();
   }
   else{
      Toast.makeText(getApplicationContext(),"permission
denied",Toast.LENGTH_SHORT).show();

   }
   }
}
```

AndroidManifest.xml

```
<?xml version="1.0" encoding="utf-8"?>
<manifest xmlns:android="http://schemas.android.com/apk/res/android"
   package="com.vardhaman.sms">
   <uses-permission android:name="android.permission.SEND_SMS"/>
   <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/Theme.Sms">
      <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>
```

Output:



Week-10

Create an android app to show details phone contacts, implement calling, receiving features

activitymain.xml

```
<?xml version="1.0" encoding="utf-8"?>
    <androidx.constraintlayout.widget.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=".MainActivity">
      <EditText
        android:id="@+id/et1"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:hint="Enter Phone Number"
        app:layout_constraintBottom_toBottomOf="parent"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toTopOf="parent"
        app:layout_constraintVertical_bias="0.161" />
      <Button
        android:id="@+id/bt"
        android:layout_width="153dp"
        android:layout_height="46dp"
        android:layout_weight="1"
        android:text="Call"
        app:layout_constraintBottom_toBottomOf="parent"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintHorizontal_bias="0.461"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toTopOf="parent"
        app:layout_constraintVertical_bias="0.302" />
    </androidx.constraintlayout.widget.ConstraintLayout>
```

Androidmanifest.xml

```
<?xml version="1.0" encoding="utf-8"?>
    <manifest xmlns:android="http://schemas.android.com/apk/res/android"
      package="com.vardhaman.telephonemanager"
 <uses-permission android:name="android.permission.CALL_PHONE">
</uses-permission>
      <application
        android:allowBackup="true"
        android:icon="@mipmap/ic_launcher"
        android:label="@string/app_name"
        android:roundIcon="@mipmap/ic_launcher_round"
        android:supportsRtl="true"
```

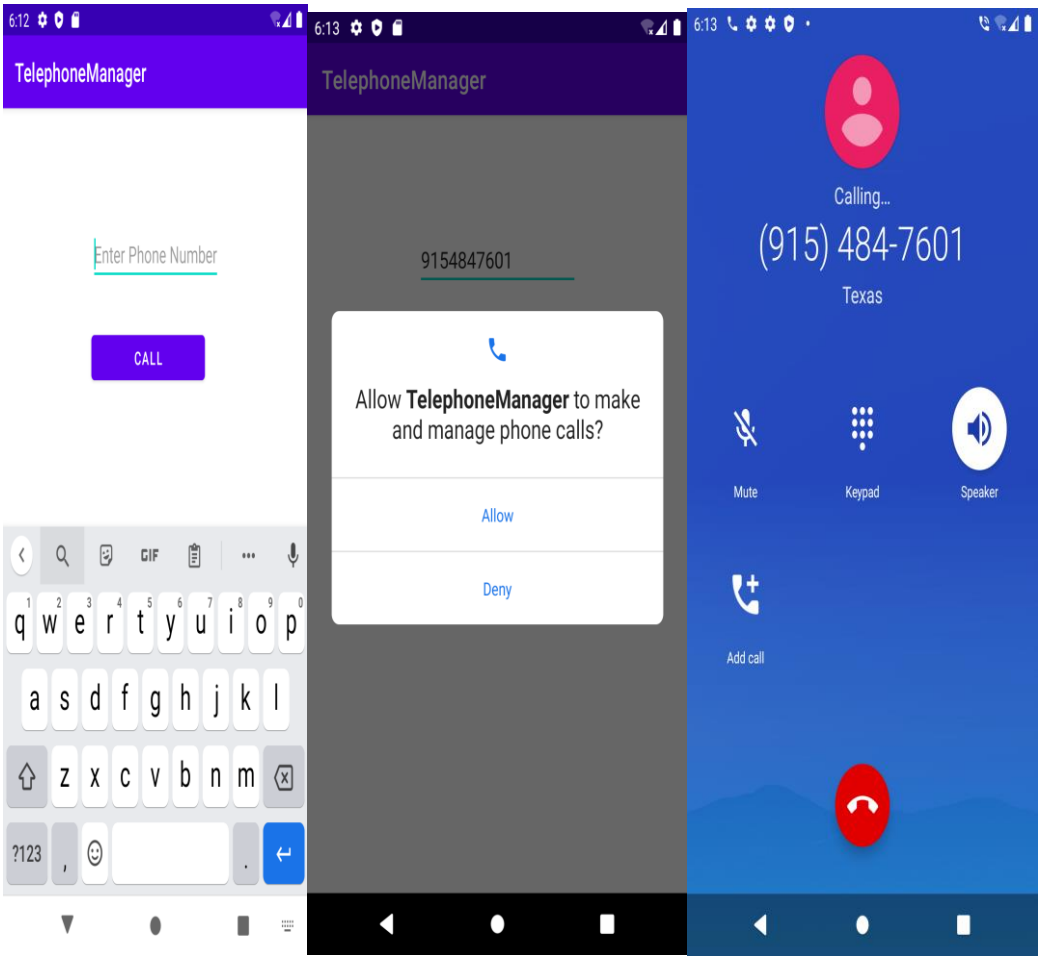
```
        android:theme="@style/Theme.TelephoneManager">
        <activity android:name=".MainActivity">
          <intent-filter>
            <action android:name="android.intent.action.MAIN" />
            <category android:name="android.intent.category.LAUNCHER" />
          </intent-filter>
        </activity>
      </application>
    </manifest>
```

Mainactivity.java

```
package com.vardhaman.telephonemanager;
    import androidx.appcompat.app.AppCompatActivity;
    import androidx.core.app.ActivityCompat;
    import android.Manifest;
    import android.content.Intent;
    import android.content.pm.PackageManager;
    import android.net.Uri;
    import android.os.Build;
    import android.os.Bundle;
    import android.view.View;
    import android.widget.Button;
    import android.widget.EditText;
    public class MainActivity extends AppCompatActivity {
      Button b;
      EditText phno;
      @Override
      protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);
        phno = (EditText) findViewById(R.id.et1);
        b = (Button) findViewById(R.id.bt);
        b.setOnClickListener(new View.OnClickListener() {
          @Override
          public void onClick(View v) {
            callPhoneNumber();
          }
        });
      }
      public void requestPermissionsResult(int requestCode, String[]
    permissions, int[] grantResults) {
        if (requestCode == 101) {
          if (grantResults[0] == PackageManager.PERMISSION_GRANTED) {
            callPhoneNumber();
          }
        }
      }
      public void callPhoneNumber() {
        try {
```

```
        if (Build.VERSION.SDK_INT > 22) {
            if (ActivityCompat.checkSelfPermission(this,
Manifest.permission.CALL_PHONE) !=
PackageManager.PERMISSION_GRANTED) {
                ActivityCompat.requestPermissions(MainActivity.this, new
String[]{Manifest.permission.CALL_PHONE}, 101);
                return;
            }
            Intent callIntent = new Intent(Intent.ACTION_CALL);
            callIntent.setData(Uri.parse("tel:" + phno.getText().toString()));
            startActivity(callIntent);
        } else {
            Intent callIntent = new Intent(Intent.ACTION_CALL);
            callIntent.setData(Uri.parse("tel:" + phno.getText().toString()));
            startActivity(callIntent);
        }
    } catch (Exception ex) {
        ex.printStackTrace();
    }
  }
}
```

OUTPUT



Week-11

a. Create an android app to demonstrate camera

activitymain.xml

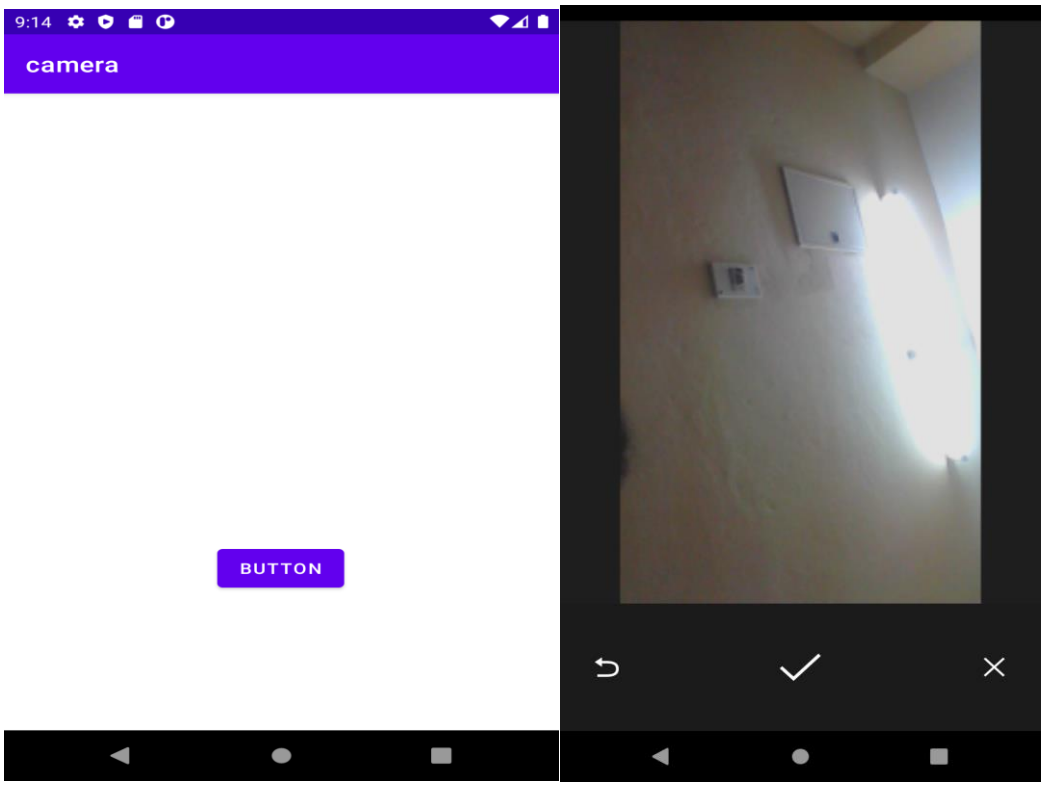
```
<?xml version="1.0" encoding="utf-8"?>
<androidx.constraintlayout.widget.ConstraintLayout
xmlns:android="http://schemas.android.com/apk/res/android"
   xmlns:app="http://schemas.android.com/apk/res-auto"
   xmlns:tools="http://schemas.android.com/tools"
   android:layout_width="match_parent"
   android:layout_height="match_parent"
   tools:context=".MainActivity">

   <ImageView
      android:id="@+id/imageView1"
      android:layout_width="wrap_content"
      android:layout_height="wrap_content"
      app:layout_constraintBottom_toBottomOf="parent"
      app:layout_constraintEnd_toEndOf="parent"
      app:layout_constraintStart_toStartOf="parent"
      app:layout_constraintTop_toTopOf="parent"
      tools:srcCompat="@tools:sample/avatars" />

   <Button
      android:id="@+id/button1"
      android:layout_width="wrap_content"
      android:layout_height="wrap_content"
      android:text="Button"
      app:layout_constraintBottom_toBottomOf="parent"
      app:layout_constraintEnd_toEndOf="parent"
      app:layout_constraintHorizontal_bias="0.498"
      app:layout_constraintStart_toStartOf="parent"
      app:layout_constraintTop_toTopOf="parent"
      app:layout_constraintVertical_bias="0.767" />

</androidx.constraintlayout.widget.ConstraintLayout>
```

mainactivity.java

```
package com.vardhaman.camera;
import androidx.appcompat.app.AppCompatActivity;
import android.content.Intent;
import android.graphics.Bitmap;
import android.os.Bundle;
import android.view.View;
import android.widget.Button;
import android.widget.ImageView;

public class MainActivity extends AppCompatActivity {
   private static final int CAMERA_REQUEST = 1888;
   ImageView imageView;
   @Override
   protected void onCreate(Bundle savedInstanceState) {
      super.onCreate(savedInstanceState);
```

```
        setContentView(R.layout.activity_main);
        imageView = (ImageView) this.findViewById(R.id.imageView1);
        Button photoButton = (Button) this.findViewById(R.id.button1);
        photoButton.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                Intent cameraIntent = new
Intent(android.provider.MediaStore.ACTION_IMAGE_CAPTURE);
                startActivityForResult(cameraIntent, CAMERA_REQUEST);
               }
        });
    }

    protected void OnActivityResult(int requestCode, int resultCode, Intent data) {
        if (requestCode == CAMERA_REQUEST) {
            Bitmap photo = (Bitmap) data.getExtras().get("data");
            imageView.setImageBitmap(photo);

        }
    }
}
```

OUTPUT



b. Create an android app to demonstrate media player

Mainactivity.xml

```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
   xmlns:tools="http://schemas.android.com/tools"
   android:layout_width="match_parent"
   android:layout_height="match_parent"
   tools:context=".MainActivity"
   android:padding="20dp">
      <Button
         android:id="@+id/button"
         android:layout_width="wrap_content"
         android:layout_height="wrap_content"
         android:layout_weight="1"
         android:onClick="music"
         android:text="PLAY"
         android:layout_marginLeft="20dp"/>
      <Button
         android:id="@+id/button2"
         android:layout_width="wrap_content"
         android:layout_height="wrap_content"
         android:layout_weight="1"
         android:text="PAUSE"
         android:onClick="music"
         android:layout_marginLeft="30dp"/>
      <Button
         android:id="@+id/button3"
         android:layout_width="wrap_content"
         android:layout_height="wrap_content"
         android:layout_weight="1"
         android:text="STOP"
         android:onClick="music"
         android:layout_marginLeft="60dp"/>
   </LinearLayout>
```

mainactivity.java

```
package com.vardhaman.mediaplayer;
import androidx.appcompat.app.AppCompatActivity;
import android.media.MediaPlayer;
import android.os.Bundle;
import android.view.View;
public class MainActivity extends AppCompatActivity {
   MediaPlayer mp;
   @Override
   protected void onCreate(Bundle savedInstanceState) {
      super.onCreate(savedInstanceState);
      setContentView(R.layout.activity_main);
```

```
        mp = null;
    }
    public void music(View v) {
        switch (v.getId()) {
            case R.id.button:
                if (mp == null) {
                    mp = MediaPlayer.create(this, R.raw.roja);
                }
                mp.setOnCompletionListener(new
MediaPlayer.OnCompletionListener() {
                    @Override
                    public void onCompletion(MediaPlayer mp) {
                        stopMusic();
                    }
                });
                mp.start();
                break;
            case R.id.button2:
                if (mp != null) {
                    mp.pause();
                }
                break;
            case R.id.button3:
                if (mp != null) {
                    mp.stop();
                    stopMusic();
                }
                break;
        }
    }
    public void stopMusic() {
        mp.release();
        mp = null;
    }
    public void onStop() {
        super.onStop();
        stopMusic();
    }
}
```

OUTPUT:


6:21
mediaplayer
PLAY
PAUSE
STOP

Week-12

a. Create an android app to store details of students in SQ Lite and display the details

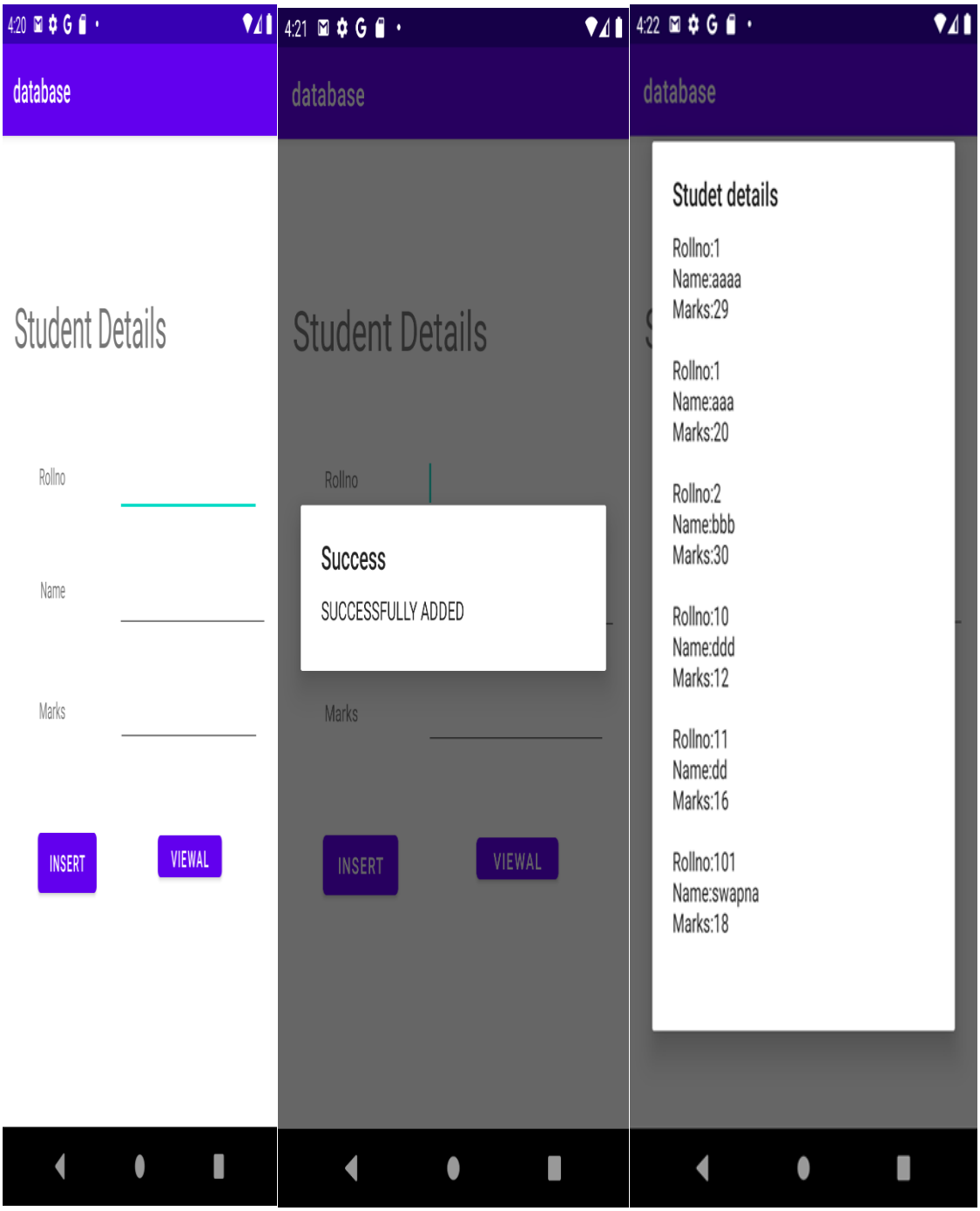
```
<?xml version="1.0" encoding="utf-8"?>
<androidx.constraintlayout.widget.ConstraintLayout
xmlns:android="http://schemas.android.com/apk/res/android"
  xmlns:app="http://schemas.android.com/apk/res-auto"
  xmlns:tools="http://schemas.android.com/tools"
  android:layout_width="match_parent"
  android:layout_height="match_parent"
  tools:context=".MainActivity">

  <TextView
    android:id="@+id/textView"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="Student Details"
    android:textSize="34sp"
    app:layout_constraintBottom_toBottomOf="parent"
    app:layout_constraintEnd_toEndOf="parent"
    app:layout_constraintHorizontal_bias="0.096"
    app:layout_constraintStart_toStartOf="parent"
    app:layout_constraintTop_toTopOf="parent"
    app:layout_constraintVertical_bias="0.167" />

  <TextView
    android:id="@+id/textView2"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="Rollno"
    app:layout_constraintBottom_toBottomOf="parent"
    app:layout_constraintEnd_toEndOf="parent"
    app:layout_constraintHorizontal_bias="0.147"
    app:layout_constraintStart_toStartOf="parent"
    app:layout_constraintTop_toTopOf="parent"
    app:layout_constraintVertical_bias="0.338" />

  <TextView
    android:id="@+id/textView3"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="Name"
    app:layout_constraintBottom_toBottomOf="parent"
    app:layout_constraintEnd_toEndOf="parent"
    app:layout_constraintHorizontal_bias="0.152"
    app:layout_constraintStart_toStartOf="parent"
    app:layout_constraintTop_toTopOf="parent"
    app:layout_constraintVertical_bias="0.456" />

  <TextView
    android:id="@+id/textView4"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
```

```
    android:text="Marks"
    app:layout_constraintBottom_toBottomOf="parent"
    app:layout_constraintEnd_toEndOf="parent"
    app:layout_constraintHorizontal_bias="0.147"
    app:layout_constraintStart_toStartOf="parent"
    app:layout_constraintTop_toTopOf="parent"
    app:layout_constraintVertical_bias="0.582" />

  <EditText
    android:id="@+id/Rollno"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:ems="10"
    android:inputType="textPersonName"
    app:layout_constraintBottom_toBottomOf="parent"
    app:layout_constraintEnd_toEndOf="parent"
    app:layout_constraintHorizontal_bias="0.86"
    app:layout_constraintStart_toStartOf="parent"
    app:layout_constraintTop_toTopOf="parent"
    app:layout_constraintVertical_bias="0.336" />

  <EditText
    android:id="@+id/Name"
    android:layout_width="223dp"
    android:layout_height="50dp"
    android:ems="10"
    android:inputType="textPersonName"
    app:layout_constraintBottom_toBottomOf="parent"
    app:layout_constraintEnd_toEndOf="parent"
    app:layout_constraintHorizontal_bias="0.92"
    app:layout_constraintStart_toStartOf="parent"
    app:layout_constraintTop_toTopOf="parent"
    app:layout_constraintVertical_bias="0.459" />

  <EditText
    android:id="@+id/Marks"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:ems="10"
    android:inputType="textPersonName"
    app:layout_constraintBottom_toBottomOf="parent"
    app:layout_constraintEnd_toEndOf="parent"
    app:layout_constraintHorizontal_bias="0.865"
    app:layout_constraintStart_toStartOf="parent"
    app:layout_constraintTop_toTopOf="parent"
    app:layout_constraintVertical_bias="0.588" />

  <Button
    android:id="@+id/INSERT"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="INSERT"
    app:layout_constraintBottom_toBottomOf="parent"
    app:layout_constraintEnd_toEndOf="parent"
```

```
    app:layout_constraintHorizontal_bias="0.164"
    app:layout_constraintStart_toStartOf="parent"
    app:layout_constraintTop_toTopOf="parent"
    app:layout_constraintVertical_bias="0.754" />

  <Button
    android:id="@+id/VIEWALL"
    android:layout_width="96dp"
    android:layout_height="37dp"
    android:text="VIEWALL"
    app:layout_constraintBottom_toBottomOf="parent"
    app:layout_constraintEnd_toEndOf="parent"
    app:layout_constraintHorizontal_bias="0.738"
    app:layout_constraintStart_toStartOf="parent"
    app:layout_constraintTop_toTopOf="parent"
    app:layout_constraintVertical_bias="0.742" />
</androidx.constraintlayout.widget.ConstraintLayout>

Mainactivity.java
package com.vardhaman.database;
import androidx.appcompat.app.AppCompatActivity;
import android.app.AlertDialog;
import android.content.Context;
import android.database.Cursor;
import android.database.sqlite.SQLiteDatabase;
import android.os.Bundle;
import android.view.View;
import android.widget.Button;
import android.widget.EditText;
import android.widget.Toast;
public class MainActivity extends AppCompatActivity  {
  Button Insert,viewAll;
  EditText Rollno,Name,Marks;
  SQLiteDatabase db;
  @Override
  protected void onCreate(Bundle savedInstanceState) {
    super.onCreate(savedInstanceState);
    setContentView(R.layout.activity_main);
    Insert = findViewById(R.id.INSERT);
    viewAll = findViewById(R.id.VIEWALL);
    Rollno = findViewById(R.id.Rollno);
    Name = findViewById(R.id.Name);
    Marks = findViewById(R.id.Marks);
    db = openOrCreateDatabase("StudentDB", Context.MODE_PRIVATE, null);
    db.execSQL("CREATE TABLE IF NOT EXISTS student(rollno varchar,name
varchar,marks number)");
    Insert.setOnClickListener(new View.OnClickListener() {
      @Override
      public void onClick(View v) {
        if (Rollno.getText().toString().length() == 0 || Name.getText().toString().length()
== 0 || Marks.getText().toString().length() == 0) {
          Toast.makeText(getApplicationContext(), "Please enter details",
Toast.LENGTH_LONG).show();
          return;
```

```
            }
            db.execSQL("INSERT INTO STUDENT
VALUES('"+Rollno.getText()+"','"+Name.getText()+"','"+Marks.getText()+"');");
            showMessage("Success","SUCCESSFULLY ADDED");
            clearText();
          }
        });
        viewAll.setOnClickListener(new View.OnClickListener() {
          public void onClick(View v) {
            Cursor c = db.rawQuery("select * from student", null);
            if (c.getCount() == 0) {
              showMessage("error", "No records inserted");
              return;
            }
            StringBuffer bf = new StringBuffer();
            while (c.moveToNext()) {
              bf.append("Rollno:" + c.getString(0) + "\n");
              bf.append("Name:" + c.getString(1) + "\n");
              bf.append("Marks:" + c.getString(2) + "\n\n");
            }
            showMessage("Studet details", bf.toString());
          }
        });
    }
    public void showMessage(String title,String message)
    {
      AlertDialog.Builder b=new AlertDialog.Builder(this);
      b.setCancelable(true);
      b.setTitle(title);
      b.setMessage(message);
      b.show();
    }
    private void clearText()
    {
      Rollno.setText("");
      Name.setText("");
      Marks.setText("");
      Rollno.requestFocus();
    }
}
```

Output:


4:20
database
Student Details
Rollno
Name
Marks
INSERT
VIEWAL
4:21
database
Student Details
Rollno
Success
SUCCESSFULLY ADDED
Marks
INSERT
VIEWAL
4:22
database
Studet details
Rollno:1
Name:aaaa
Marks:29
Rollno:1
Name:aaa
Marks:20
Rollno:2
Name:bbb
Marks:30
Rollno:10
Name:ddd
Marks:12
Rollno:11
Name:dd
Marks:16
Rollno:101
Name:swapna
Marks:18

b. Create an android app to perform insert, update, delete operations on student database

Activitymain.xml

```
<?xml version="1.0" encoding="utf-8"?>
    <AbsoluteLayout
      xmlns:android="http://schemas.android.com/apk/res/android"
      xmlns:tools="http://schemas.android.com/tools"
      android:layout_width="match_parent"
      android:layout_height="match_parent"
      tools:context=".MainActivity">
        <TextView
          android:layout_width="wrap_content"
          android:layout_height="wrap_content"
          android:layout_x="50dp"
          android:layout_y="20dp"
          android:text="Student Details"
          android:textSize="30sp" />
        <TextView
          android:layout_width="wrap_content"
          android:layout_height="wrap_content"
          android:layout_x="20dp"
          android:layout_y="110dp"
          android:text="Enter Rollno:"
          android:textSize="20sp" />
        <EditText
        android:id="@+id/Rollno"
        android:layout_width="150dp"
        android:layout_height="wrap_content"
        android:layout_x="175dp"
        android:layout_y="100dp"
        android:textSize="20sp" />
        <TextView
          android:layout_width="wrap_content"
          android:layout_height="wrap_content"
          android:layout_x="20dp"
          android:layout_y="160dp"
          android:text="Enter Name:"
          android:textSize="20sp" />
        <EditText
          android:id="@+id/Name"
          android:layout_width="150dp"
          android:layout_height="wrap_content"
          android:layout_x="175dp"
          android:layout_y="150dp"
          android:inputType="text"
          android:textSize="20sp" />
        <TextView
          android:layout_width="wrap_content"
```

```
    android:layout_height="wrap_content"
    android:layout_x="20dp"
    android:layout_y="210dp"
    android:text="Enter Marks:"
    android:textSize="20sp" />
<EditText
    android:id="@+id/Marks"
    android:layout_width="150dp"
    android:layout_height="wrap_content"
    android:layout_x="175dp"
    android:layout_y="200dp"
    android:inputType="number"
    android:textSize="20sp" />
<Button
    android:id="@+id/Insert"
    android:layout_width="150dp"
    android:layout_height="wrap_content"
    android:layout_x="25dp"
    android:layout_y="300dp"
    android:text="Insert"
    android:textSize="20dp" />
<Button
    android:id="@+id/Delete"
    android:layout_width="150dp"
    android:layout_height="wrap_content"
    android:layout_x="200dp"
    android:layout_y="300dp"
    android:text="Delete"
    android:textSize="20dp" />
<Button
    android:id="@+id/Update"
    android:layout_width="150dp"
    android:layout_height="wrap_content"
    android:layout_x="25dp"
    android:layout_y="400dp"
    android:text="Update"
    android:textSize="20dp" />
<Button
    android:id="@+id/View"
    android:layout_width="150dp"
    android:layout_height="wrap_content"
    android:layout_x="200dp"
    android:layout_y="400dp"
    android:text="View"
    android:textSize="20dp" />
<Button
    android:id="@+id/ViewAll"
    android:layout_width="200dp"
    android:layout_height="wrap_content"
```

```
            android:layout_x="100dp"
            android:layout_y="500dp"
            android:text="View All"
            android:textSize="20dp" />
        </AbsoluteLayout>
```

mainactivity.java

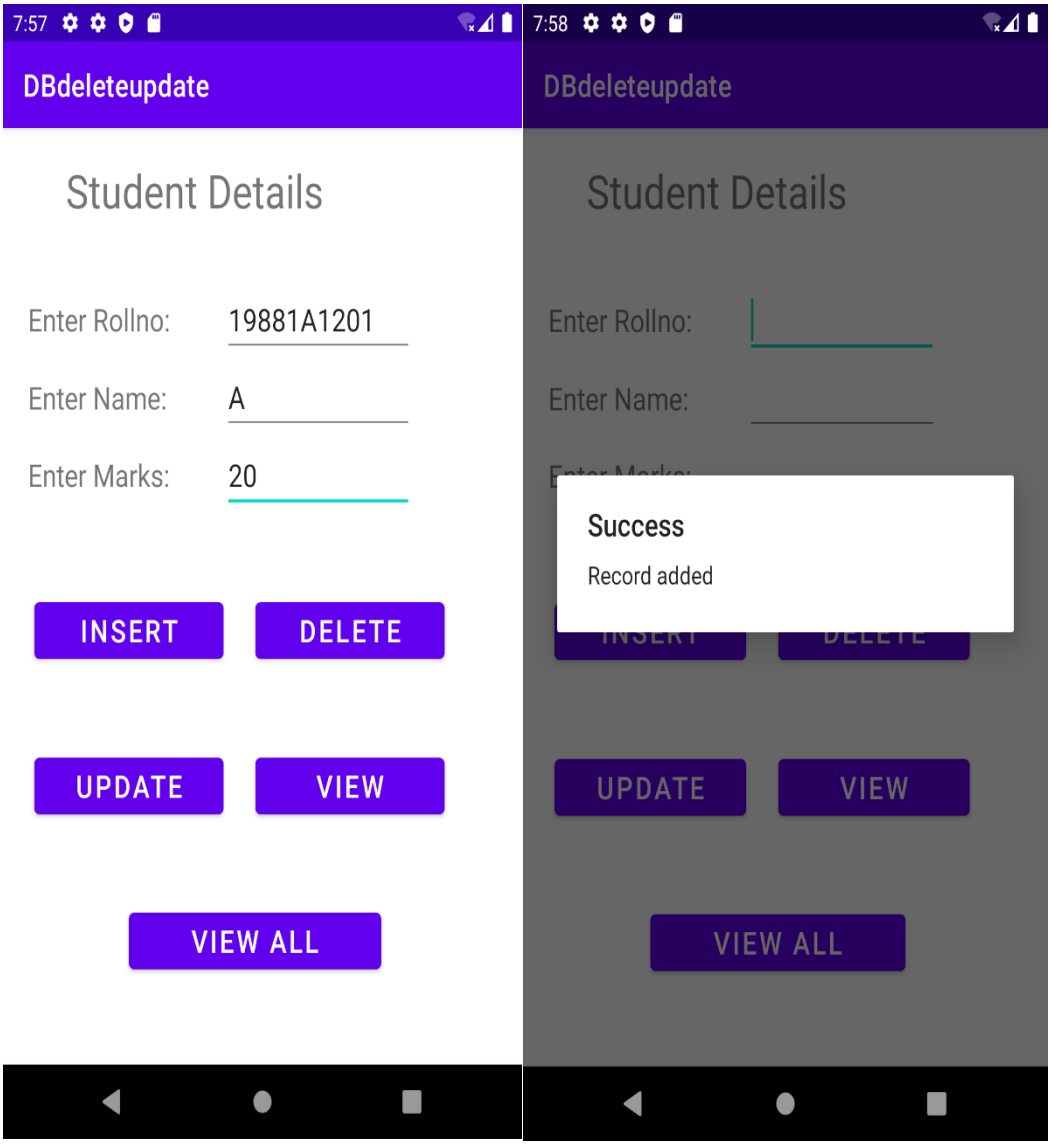
```
package com.vardhaman.dbdeleteupdate;
import android.app.AlertDialog;
import android.os.Bundle;
import androidx.appcompat.app.AppCompatActivity;
import android.content.Context;
import android.database.Cursor;
import android.database.sqlite.SQLiteDatabase;
import android.view.View;
import android.widget.Button;
import android.widget.EditText;

public class MainActivity extends AppCompatActivity implements
      android.view.View.OnClickListener {
   EditText Rollno,Name,Marks;
   Button Insert,Delete,Update,View,ViewAll;
   SQLiteDatabase db;
   /** Called when the activity is first created. */
   @Override
   public void onCreate(Bundle savedInstanceState)
   {
      super.onCreate(savedInstanceState);
      setContentView(R.layout.activity_main);
      Rollno=(EditText)findViewById(R.id.Rollno);
      Name=(EditText)findViewById(R.id.Name);
      Marks=(EditText)findViewById(R.id.Marks);
      Insert=(Button)findViewById(R.id.Insert);
      Delete=(Button)findViewById(R.id.Delete);
      Update=(Button)findViewById(R.id.Update);
      View=(Button)findViewById(R.id.View);
      ViewAll=(Button)findViewById(R.id.ViewAll);
      Insert.setOnClickListener(this);
      Delete.setOnClickListener(this);
      Update.setOnClickListener(this);
      View.setOnClickListener(this);
      ViewAll.setOnClickListener(this);
      // Creating database and table
      db=openOrCreateDatabase("StudentDB", Context.MODE_PRIVATE, null);
      db.execSQL("CREATE TABLE IF NOT EXISTS student(rollno VARCHAR,name
VARCHAR,marks VARCHAR);");
   }
   public void onClick(View view)
   {
      // Inserting a record to the Student table
      if(view==Insert)
      {
```

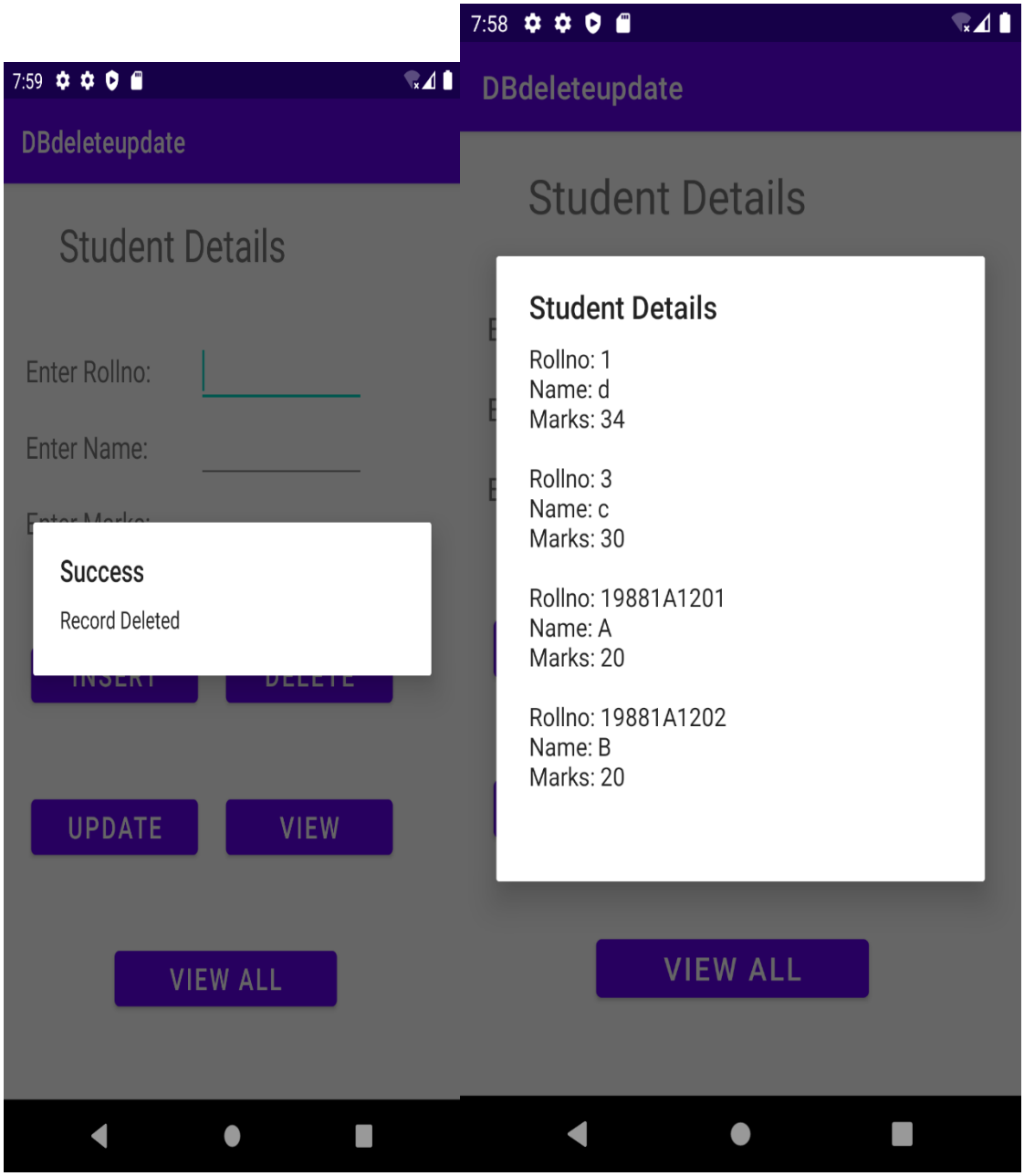
```
        // Checking for empty fields
        if(Rollno.getText().toString().trim().length()==0||
             Name.getText().toString().trim().length()==0||
             Marks.getText().toString().trim().length()==0)
        {
          showMessage("Error", "Please enter all values");
          return;
        }
        db.execSQL("INSERT INTO student
VALUES('"+Rollno.getText()+"','"+Name.getText()+
                  "','"+Marks.getText()+"');");
        showMessage("Success", "Record added");
        clearText();
      }
      // Deleting a record from the Student table
      if(view==Delete)
         {
        // Checking for empty roll number
        if(Rollno.getText().toString().trim().length()==0)
        {
          showMessage("Error", "Please enter Rollno");
          return;
        }
        Cursor c=db.rawQuery("SELECT * FROM student WHERE
rollno='"+Rollno.getText()+"'", null);
        if(c.moveToFirst())
        {
          db.execSQL("DELETE FROM student WHERE rollno='"+Rollno.getText()+"'");
               showMessage("Success", "Record Deleted");
        }
        else
        {
          showMessage("Error", "Invalid Rollno");
        }
        clearText();
      }
      // Updating a record in the Student table
      if(view==Update)
      {
        // Checking for empty roll number
        if(Rollno.getText().toString().trim().length()==0)
        {
          showMessage("Error", "Please enter Rollno");
          return;
        }
        Cursor c=db.rawQuery("SELECT * FROM student WHERE
rollno='"+Rollno.getText()+"'", null);
        if(c.moveToFirst()) {
          db.execSQL("UPDATE student SET name='" + Name.getText() +
               "',marks='" + Marks.getText() +
               "' WHERE rollno='"+Rollno.getText()+"'");
          showMessage("Success", "Record Modified");
        }
        else {
```

```
            showMessage("Error", "Invalid Rollno");
          }
          clearText();
        }
        // Display a record from the Student table
        if(view==View)
        {
          // Checking for empty roll number
          if(Rollno.getText().toString().trim().length()==0)
          {
            showMessage("Error", "Please enter Rollno");
            return;
          }
          Cursor c=db.rawQuery("SELECT * FROM student WHERE
rollno='"+Rollno.getText()+"'", null);
          if(c.moveToFirst())
          {
            Name.setText(c.getString(1));
            Marks.setText(c.getString(2));
          }
          else
            {
            showMessage("Error", "Invalid Rollno");
            clearText();
          }
        }
        // Displaying all the records
        if(view==ViewAll)
        {
          Cursor c=db.rawQuery("SELECT * FROM student", null);
          if(c.getCount()==0)
          {
            showMessage("Error", "No records found");
            return;
          }
          StringBuffer buffer=new StringBuffer();
          while(c.moveToNext())
          {
            buffer.append("Rollno: "+c.getString(0)+"\n");
            buffer.append("Name: "+c.getString(1)+"\n");
            buffer.append("Marks: "+c.getString(2)+"\n\n");
          }
          showMessage("Student Details", buffer.toString());
        }
      }
      public void showMessage(String title,String message)
      {
        AlertDialog.Builder builder=new AlertDialog.Builder(this);
        builder.setCancelable(true);
        builder.setTitle(title);
        builder.setMessage(message);
        builder.show();
      }
      public void clearText()
```

```
    {
        Rollno.setText("");
        Name.setText("");
        Marks.setText("");
        Rollno.requestFocus();
    }
}
```

OUTPUT:




7:59
DBdeleteupdate
Student Details
Enter Rollno:
Enter Name:
Success
Record Deleted
UPDATE
VIEW
VIEW ALL
7:58
DBdeleteupdate
Student Details
Student Details
Rollno: 1
Name: d
Marks: 34
Rollno: 3
Name: c
Marks: 30
Rollno: 19881A1201
Name: A
Marks: 20
Rollno: 19881A1202
Name: B
Marks: 20
VIEW ALL

PART-3
ONLINE RESOURCES

ONLINE RESOURCES
OBJECTIVE
To help students on acquiring more practice on the course using various online resources

DESCRIPTION
These open tools form an excellent practice platform for the students, which he can explore anytime from anywhere. The links of the websites providing SQL tutorials are given. Students are needed to explore the websites.

LINKS

1. https://www.javatpoint.com/android-tutorial
2. https://www.tutorialspoint.com/mobile_development_tutorials.htm
3. https://developer.android.com/training/basics/firstapp
4. https://www.udemy.com/course/learn-android-application-development-y/
5. https://data-flair.training/blogs/android-tutorial/
6. https://www.tutlane.com/tutorial/android

PART-4
POSSIBLE VIVA QUESTIONS

DESCRIPTION

Possible viva questions include, questions whose answers are provided. The student can have a practice of them.

1) What is Android?

Android is an open-source, Linux-based operating system used in mobiles, tablets, televisions, etc.

2) Who is the founder of Android?

Andy Rubin.

3) Explain the Android application Architecture.

Following is a list of components of Android application architecture:

o Services: Used to perform background functionalities.

o Intent: Used to perform the interconnection between activities and the data passing mechanism.

o Resource Externalization: strings and graphics.

o Notification: light, sound, icon, notification, dialog box and toast.

o Content Providers: It will share the data between applications.

4) What are the code names of android?

1. Aestro
2. Blender
3. Cupcake
4. Donut
5. Eclair
6. Froyo
7. Gingerbread
8. Honeycomb
9. Ice Cream Sandwich
10. Jelly Bean
11. KitKat
12. Lollipop
13. Marshmallow

5) What are the advantages of Android?

Open-source: It means no license, distribution and development fee.

Platform-independent: It supports Windows, Mac, and Linux platforms.

Supports various technologies: It supports camera, Bluetooth, wifi, speech, EDGE etc. technologies.

Highly optimized Virtual Machine: Android uses a highly optimized virtual machine for mobile devices, called DVM (Dalvik Virtual Machine).

6) Does android support other languages than java?

Yes, an android app can be developed in C/C++ also using android NDK (Native Development Kit). It makes the performance faster. It should be used with Android SDK.

7) What are the core building blocks of android?

The core building blocks of Android are:

o Activity

o View

o Intent

o Service

o Content Provider

o Fragment etc.

8) What is activity in Android?

Activity is like a frame or window in java that represents GUI. It represents one screen of android.

9) What are the life cycle methods of android activity?

There are 7 life-cycle methods of activity. They are as follows:

1. onCreate()
2. onStart()
3. onResume()
4. onPause()
5. onStop()
6. onRestart()
7. onDestroy()

10) What is intent?

It is a kind of message or information that is passed to the components. It is used to launch an activity, display a web page, send SMS, send email, etc. There are two types of intents in android:

1. Implicit Intent
2. Explicit Intent

11) How are view elements identified in the android program?

View elements can be identified using the keyword findViewById.

12) Define Android toast.

An android toast provides feedback to the users about the operation being performed by them. It displays the message regarding the status of operation initiated by the user.

13) Give a list of impotent folders in android

The following folders are declared as impotent in android:

- o AndroidManifest.xml
- o build.xml
- o bin/
- o src/
- o res/
- o assets/

14) Explain the use of 'bundle' in android?

We use bundles to pass the required data to various subfolders.

15) What is an application resource file?

The files which can be injected for the building up of a process are called as application resource file.

16) What is the use of LINUX ID in android?

A unique Linux ID is assigned to each application in android. It is used for the tracking of a process.

17) Can the bytecode be written in java be run on android?

No

18) List the various storages that are provided by Android.

The various storage provided by android are:

- o Shared Preferences
- o Internal Storage
- o External Storage
- o SQLite Databases
- o Network Connection

19) How are layouts placed in Android?

Layouts in Android are placed as XML files.

20) Where are layouts placed in Android?

Layouts in Android are placed in the layout folder.

21) What is the implicit intent in android?

The Implicit intent is used to invoke the system components.

22) What is explicit intent in android?

An explicit intent is used to invoke the activity class.

23) How to call another activity in android?

1. Intent i = new Intent(getApplicationContext(), ActivityTwo.class);
2. startActivity(i);

24) What is service in android?

A service is a component that runs in the background. It is used to play music, handle network transaction, etc.

25) What is the name of the database used in android?

SQLite: An opensource and lightweight relational database for mobile devices.

26) What is AAPT?

AAPT is an acronym for android asset packaging tool. It handles the packaging process.

27) What is a content provider?

A content provider is used to share information between Android applications.

28) What is fragment?

The fragment is a part of Activity by which we can display multiple screens on one activity.

29) What is ADB?

ADB stands for Android Debug Bridge. It is a command line tool that is used to communicate with the emulator instance.

30) What is NDK?

NDK stands for Native Development Kit. By using NDK, you can develop a part of an app using native language such as C/C++ to boost the performance.

31) What is ANR?

ANR stands for Application Not Responding. It is a dialog box that appears if the application is no longer responding.

32) What is the Google Android SDK?

The Google Android SDK is a toolset which is used by developers to write apps on Android-enabled devices. It contains a graphical interface that emulates an Android-driven handheld environment and allows them to test and debug their codes.

33) What is an APK format?

APK is a short form stands for Android Packaging Key. It is a compressed key with classes, UI's, supportive assets and manifest. All files are compressed to a single file is called APK.

34) Which language does Android support to develop an application?

Android applications are written by using the java (Android SDK) and C/C++ (Android NDK).

35) What is ADT in Android?

ADT stands for Android Development Tool. It is used to develop the applications And test the applications.

36) What is View Group in Android?

View Group is a collection of views and other child views. It is an invisible part and the base class for layouts.

37) What is the Adapter in Android?

An adapter is used to create a child view to present the parent view items.

38) What is nine-patch images tool in Android?

We can change bitmap images into nine sections with four corners, four edges, and an axis.

39) Which kernel is used in Android?

Android is a customized Linux 3.6 kernel.

40) What is application Widgets in Android?

Application widgets are miniature application views that can be embedded in

other applications and receive periodic updates.

41) Which types of flags are used to run an application on Android?

Following are two types of flags to run an application in Android:

- o FLAG_ACTIVITY_NEW_TASK
- o FLAG_ACTIVITY_CLEAR_TOP

42) What is a singleton class in Android?

A singleton class is a class which can create only an object that can be shared by all other classes.

43) What is sleep mode in Android?

In sleep mode, CPU is slept and doesn't accept any commands from android device except Radio interface layer and alarm.

44) What do you mean by a drawable folder in Android?

In Android, a drawable folder is compiled a visual resource that can use as a background, banners, icons, splash screen, etc.

45) What is DDMS?

DDMS stands for Dalvik Debug Monitor Server. It gives the wide array of debugging features:

1. Port forwarding services
2. Screen capture
3. Thread and heap information
4. Network traffic tracking
5. Location data spoofing

46) Define Android Architecture?

The Android architecture consists of 4 components:

1. Linux Kernal
2. Libraries
3. Android Framework
4. Android Applications

47) What is a portable wi-fi hotspot?

The portable wi-fi hotspot is used to share internet connection to other wireless devices.

48) Name the dialog box which is supported by Android?

- o Alert Dialog
- o Progress Dialog
- o Date Picker Dialog
- o Time picker Dialog

49) Name some exceptions in Android?

- o Inflate Exception
- o Surface.OutOfResourceException
- o SurfaceHolder.BadSurfaceTypeException
- o WindowManager.BadTokenException

50) What are the basic tools used to develop an Android app?

- o JDK
- o Eclipse+ADT plugin
- o SDK Tools

51)What is an iOS ?

iPhone Operating system for apple phones

PART-5
KNOWLEDGE BASE

DESCRIPTION

Knowledge base includes possible viva questions whose answers are not provided. The student has to explore for the answers.

POSSIBLE VIVA QUESTIONS (WITHOUT ANSWERS)

1) Who developed android?
2) How many versions are there in android stdio?
3) List few other android app development IDEs
4) What is activity?
5) How many methods are there in activity life cycle?
6) What is an intent?
7) What is an iOS?
8) What are the layers in iOS architecture ?
9) What are the benefits of iOS ?
10) How many types of intents are there?
11) What is an implicit intent?
12) What is explicit intent?
13) Which method is used to invoke an intent?
14) List few implicit intent
15) What is widget?
16) Which is the package for all widgets?
17) What is an attribute?
18) How a widget is accessed in java code?
19) How to set attribute value in xml file?
20) Which method is used for to display the activity?
21) How items are added to spinner control?
22) List important attributes of datepicker control
23) How to create a radiogroup?
24) What is the differerence between radiobutton and checkbox?
25) How to get year from the datepicker widget?
26) What is the use of webview?
27) How many butons are displayed by default in alertdialog?
28) Which method is used to display title message in alertdialog?
29) How to load a page using webview?
30) Does webview support all browsers?
31) Why we need thread with progressbar?
32) How to display progress status?
33) How to display hour using analog clock?
34) How to display hour using digital clock?
35) Which tag is sed for analog and digital clocks?
36) What is a fragment?
37) How many types of fragments are there?
38) What is list fragment?
39) What is dialogue fragment?
40) What is the differenece between activity and fragment?
41) What is menu?
42) How many types of menus support android?
43) What is popup menu?
44) What is context menu?
45) What is option menu?
46) Which tag is used to create a menu in xml?
47) How to add permission for SMS in manifest.xml file?

48) How to add permission for email in manifest.xml file?
49) Which class is used for email?
50) Which class is used for SMS?
51) List the package for email and sms
52) Which intent is used for SMS
53) Which is the package for phone calls?
54) Which is the class for phone calls?
55) How to find SIM number?
56) How to get phone contacts?
57) Which permission should we add in manifest.xml file for phone calls?
58) Which package is used for camera?
59) Which class is used for camera?
60) Which package is used for mediplayer?
61) Does media player support all formats?
62) How to find current time when audio or video file is playing?
63) How to find total time when audio or video file is playing?
64) Where to store audio or video files?
65) What is SQLite?
66) Which package is sed for databases?
67) How much data is stored in SQLite?