

**NAGARJUNA COLLEGE OF ENGINEERING AND TECHNOLOGY**

[Madugurki Venkatagiri Kote Devanahalli, Bengaluru, Karnataka 562110](https://www.bing.com/local?lid=YN4070x518690048&id=YN4070x518690048&q=Nagarjuna+College+Of+Engineering+&+Technology&name=Nagarjuna+College+Of+Engineering+&+Technology&cp=13.35106086730957~77.72819519042969&ppois=13.35106086730957_77.72819519042969_Nagarjuna+College+Of+Engineering+&+Technology))

**DEPARTMENT OF INFORMATION SCIENCE AND ENGINEERING**

**VI SEMESTER**

**ANDROID PROGRAMMING**

**LABORATORY MANUAL**

**21ISI62**

**Prepared By:**

**Mohan D.N**

**Assistant Professor**

**Dept of ISE**

**2023-2024**

## ANDROID PROGRAMMING(IC)

| **Course Code** | **L:T:P:S** | **Credits** | **Exam Marks** | **Exam Duration** | **Course Type** |
| --- | --- | --- | --- | --- | --- |
| **21ISI62** | **3:0:2:0** | **3** | **CIE:50 SEE:50** | **3 hours** | **IC** |

**LIST OF PROGRAMS**

**PART A**

1. Create an application to design a Visiting Card. The Visiting card should have a company logo at the top right corner. The company name should be displayed in Capital letters, aligned to the center. Information like the name of the employee, job title, phone number, address, email, fax and the website address are to be displayed.
2. Using Android, Create a login Activity. It asks “username” and “password” from user. If username and password are valid, it displays Welcome message using new activity
3. Create Implicit Intents.
4. “Happy Birth Day” App using Text View and Image View.
5. Set and retrieve shared preferences.

**PART B**

* 1. Develop an android app which displays a form to get following information from user. 1) Username 2) Password 3) Email Address 4) Phone Number 5) Country Form should be followed by a Button with label “Submit”. When user clicks the button, a message should be displayed to user describing the information entered. Utilize suitable UI controls (i.e. widgets). [When user enters country in Auto Complete Text View, list of states should be displayed in Spinner automatically]
  2. The Simple Calculator app has two edit texts and four buttons. When you enter two numbers and click a button, the app performs the calculation for that button and displays the result.
  3. Develop an android app for Text to Speech.
  4. Create the MP3 player like application with service.

Course Outcomes

On completion of this course, students will be able to:

1. Comprehend the basic features of Android Platform and Create Activities in Android.
2. Demonstrate the design concepts of user interface using components and views in Android.
3. Create and use databases for Android Application.
4. Implement messaging services in Android.
5. Deploy mobile applications in various market place for distribution

**CO-PO-PSO MAPPING MATRIX**

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **PO’s**  **CO’s** | **PO1** | **PO2** | **PO3** | **PO4** | **PO5** | **PO6** | **PO7** | **PO8** | **PO9** | **PO10** | **PO11** | **PO 12** | **PSO1** | **PSO2** | **PSO 3** |
| **C132.1** | 3 | 3 | 3 | - | 2 | - | - | 2 | 2 | 2 | - | 3 | - | - | - |
| **C132.2** | 3 | 3 | 3 | - | 2 | - | - | 2 | 2 | 2 | - | 3 | - | - | - |
| **C132.3** | 3 | 3 | 3 | 2 | 3 | - | - | 2 | 2 | 2 | - | 3 | 2 | - | - |
| **C132.4** | 3 | 3 | 3 | 2 | 3 | - | - | 2 | 2 | 2 | 2 | 3 | 2 | 2 | 2 |
| **C132.5** | 3 | 3 | 3 | 2 | 3 | - | - | 2 | 2 | 2 | 2 | 3 | 2 | 2 | 2 |
| **Avg** | 3 | 3 | 3 | 2 | 2.6 | - | - | 2 | 2 | 2 | 2 | 3 | 2 | 2 | 2 |

**INSTALLATION AND RUNNING PROCEDURE ON ANDROID STUDIO**

**Setup Android Studio**

Android Studio is the official IDE (Integrated Development Environment) for Android app development and it is based on JetBrains’ IntelliJ IDEA software. Android Studio provides many excellent features that enhance productivity when building Android apps, such as:



A blended environment where one can develop for all Android devices



Apply Changes to push code and resource changes to the running app without restarting the app



* flexible Gradle-based build system
* fast and feature-rich emulator



GitHub and Code template integration to assist you to develop common app features and import sample code



Extensive testing tools and frameworks



C++ and NDK support



Built-in support for Google Cloud Platform, making it easy to integrate Google Cloud Messaging and App Engine, and many more.

**System Requirements**

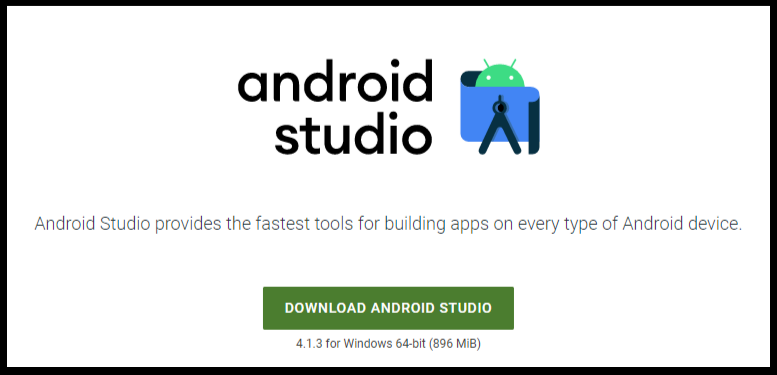


1. Microsoft Windows 7/8/10 (32-bit or 64-bit)
2. 4 GB RAM minimum, 8 GB RAM recommended (plus 1 GB for the Android Emulator)
3. 2 GB of available disk space minimum, 4 GB recommended (500 MB for IDE plus

1.5 GB for Android SDK and emulator system image) 1280 x 800 minimum screen resolution.

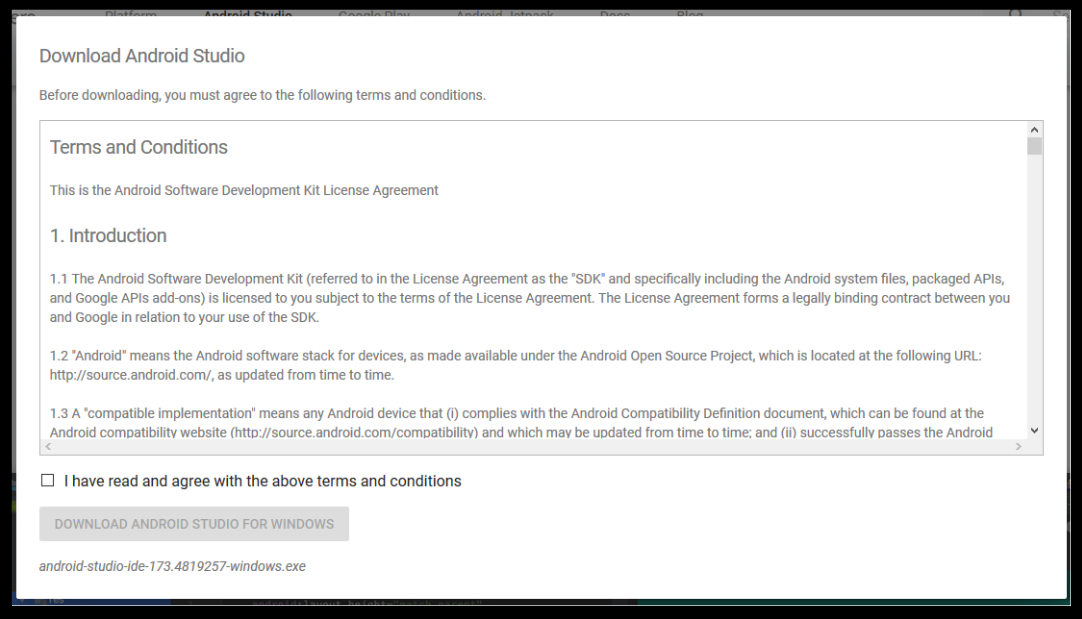
|  |  |  |
| --- | --- | --- |
| **Installation Guide** |  |  |



**Step 1. Click on the Download Android Studio Button**

**Step 2: Click on the Download Android Studio Button.**

Click on the “I have read and agree with the above terms and conditions” checkbox followed by the download button.



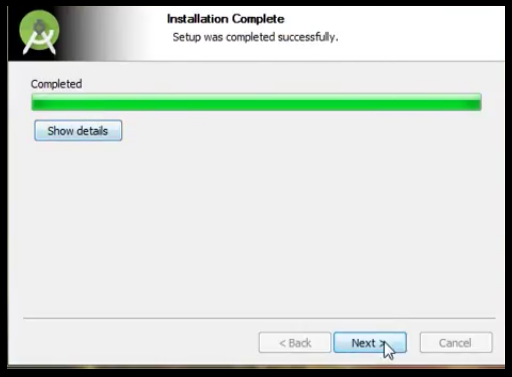
Click on the Save file button in the appeared prompt box and the file will start downloading.

**Step 3: After the downloading has finished, open the file from downloads and run it .**



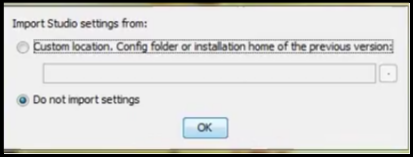
Click on next. In the next prompt, it’ll ask for a path for installation. Choose a path and hit next.

**Step 4: It will start the installation, and once it is completed, it will be like the image shown below.**



Click on next.

**Step 5: Once “Finish” is clicked**, it will ask whether the previous settings need to be imported [if the android studio had been installed earlier], or not. It is better to choose the ‘Don’t import Settings option’.



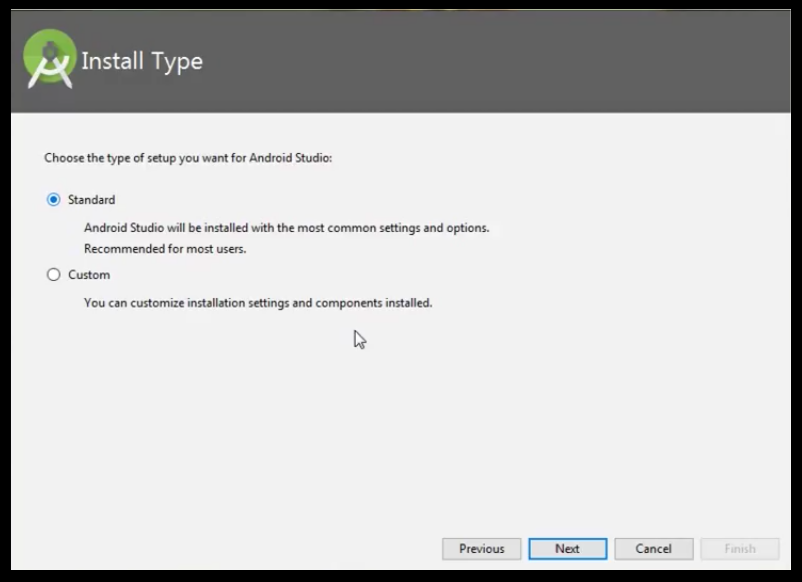
Click the OK button.

**Step 6: This will start the Android Studio.**

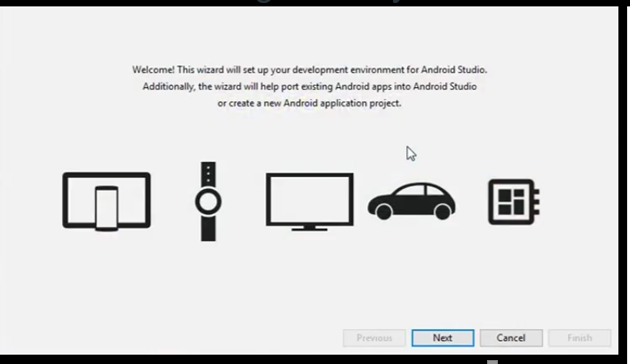


Meanwhile, it will be finding the available SDK components.

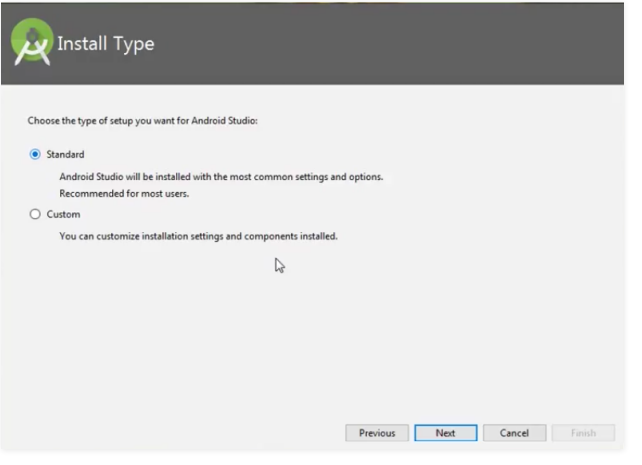


**Step 7: After it has found the SDK components, it will redirect to the Welcome dialog box.**

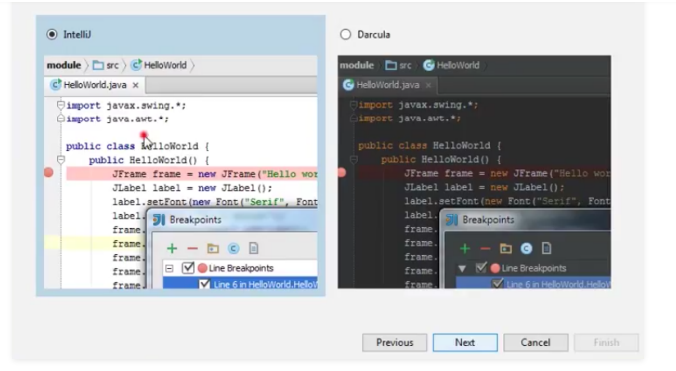
Meanwhile, it will be finding the available SDK components.



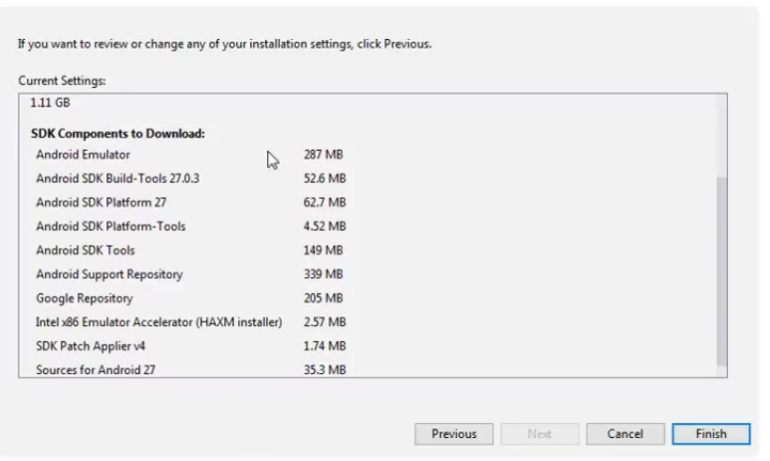
**Step 8: Choose Standard and click on Next.** Now choose the theme, whether the Light theme or the Dark one. The light one is called the IntelliJ theme whereas the dark theme is called Dracula. Choose as required.



**Click on the Next button**

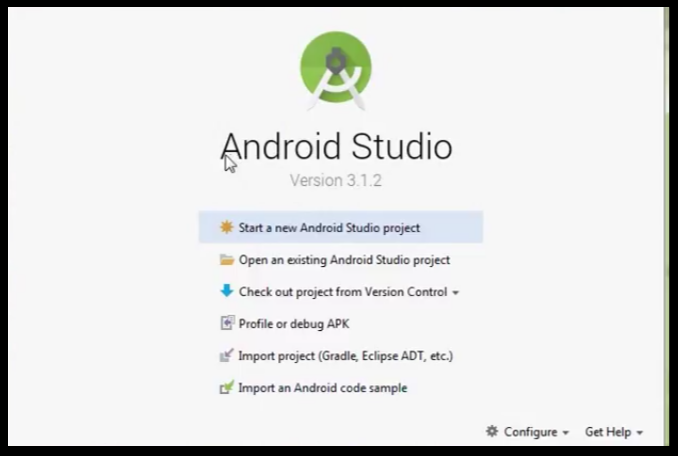


**Step 9: Click on Finish. Components begin to download let it complete.**



The Android Studio has been successfully configured. Now it’s time to launch and build apps. Click on the Finish button to launch it.

**Step 10: Click on Start a new Android Studio project to build a new app.**



[To run your first android app in Android Studio](https://www.geeksforgeeks.org/android-running-your-first-android-app/)

1. **Create an application to design a Visiting Card. The Visiting card should have a company logo at the top right corner. The company name should be displayed in Capital letters, aligned to the center. Information like the name of the employee, job title, phone number, address, email, fax and the website address are to be displayed.**

**activity\_mail.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">

<TextView

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:text=" Nagarjuna College of Engineering and Technology"

android:textColor="#DC2417"

android:textSize="30sp"

app:layout\_constraintBottom\_toBottomOf="parent"

app:layout\_constraintEnd\_toEndOf="parent"

app:layout\_constraintHorizontal\_bias="0.0"

app:layout\_constraintStart\_toStartOf="parent"

app:layout\_constraintTop\_toTopOf="parent"

app:layout\_constraintVertical\_bias="0.259" />

<TextView

android:id="@+id/textView"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:text="Student Name"

android:textColor="#354CBD"

android:textSize="24sp"

app:layout\_constraintBottom\_toBottomOf="parent"

app:layout\_constraintEnd\_toEndOf="parent"

app:layout\_constraintHorizontal\_bias="0.468"

app:layout\_constraintStart\_toStartOf="parent"

app:layout\_constraintTop\_toTopOf="parent"

app:layout\_constraintVertical\_bias="0.374" />

<TextView

android:id="@+id/textView2"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:text="4th Sem"

android:textColor="#2D6813"

android:textSize="24sp"

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="0.447" />

<TextView

android:id="@+id/textView3"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:text="9911223344"

android:textColor="#FF5722"

android:textSize="24sp"

app:layout\_constraintBottom\_toBottomOf="parent"

app:layout\_constraintEnd\_toEndOf="parent"

app:layout\_constraintStart\_toStartOf="parent"

app:layout\_constraintTop\_toTopOf="parent"

app:layout\_constraintVertical\_bias="0.515" />

<TextView

android:id="@+id/textView4"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:text="xyz@gmail.com"

android:textColor="#C00DDF"

android:textSize="24sp"

app:layout\_constraintBottom\_toBottomOf="parent"

app:layout\_constraintEnd\_toEndOf="parent"

app:layout\_constraintStart\_toStartOf="parent"

app:layout\_constraintTop\_toTopOf="parent"

app:layout\_constraintVertical\_bias="0.583" />

<ImageView

android:id="@+id/imageView"

android:layout\_width="117dp"

android:layout\_height="124dp"

app:layout\_constraintBottom\_toBottomOf="parent"

app:layout\_constraintEnd\_toEndOf="parent"

app:layout\_constraintHorizontal\_bias="0.931"

app:layout\_constraintStart\_toStartOf="parent"

app:layout\_constraintTop\_toTopOf="parent"

app:layout\_constraintVertical\_bias="0.026"

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

</androidx.constraintlayout.widget.ConstraintLayout>

**MainActivity.java**

package com.example.application\_1;

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);

}

}

**Output**



**2.Using Android, Create a login Activity. It asks “username” and “password” from user. If username and password are valid, it displays Welcome message using new activity**

**activity.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/etName"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:autofillHints=""

android:ems="10"

android:hint="USER NAME"

android:inputType="textPersonName"

app:layout\_constraintTop\_toBottomOf="@id/etName"

app:layout\_constraintBottom\_toBottomOf="parent"

app:layout\_constraintEnd\_toEndOf="parent"

app:layout\_constraintStart\_toStartOf="parent"

app:layout\_constraintTop\_toTopOf="parent"

app:layout\_constraintVertical\_bias="0.326" />

<EditText

android:id="@+id/etPassword"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:autofillHints=""

android:ems="10"

android:hint="PASSWORD"

android:inputType="numberPassword"

app:layout\_constraintBottom\_toBottomOf="parent"

app:layout\_constraintEnd\_toEndOf="parent"

app:layout\_constraintStart\_toStartOf="parent"

app:layout\_constraintTop\_toBottomOf="@+id/etName"

app:layout\_constraintVertical\_bias="0.064" />

<Button

android:onClick="Login"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:text="SIGN IN"

app:layout\_constraintBottom\_toBottomOf="parent"

app:layout\_constraintEnd\_toEndOf="parent"

app:layout\_constraintHorizontal\_bias="0.498"

app:layout\_constraintStart\_toStartOf="parent"

app:layout\_constraintTop\_toBottomOf="@+id/etPassword"

app:layout\_constraintVertical\_bias="0.098" />

</androidx.constraintlayout.widget.ConstraintLayout>

**MainActivity.java**

package com.example.loginform;  
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.TextView;  
import android.widget.Toast;  
  
public class MainActivity extends AppCompatActivity {  
 EditText uname,password;  
  
 protected void onCreate(Bundle savedInstanceState) {  
 super.onCreate(savedInstanceState);  
 setContentView(R.layout.*activity\_main*);  
 uname= (EditText) findViewById(R.id.*etName*) ;  
 password=(EditText) findViewById(R.id.*etPassword*);  
  
 }  
 public void Login(View view) {  
 if(uname.getText().toString().equals("Admin")&&password.getText().toString().equals("1234"))  
 {  
 startActivity(new Intent(MainActivity.this,Second.class));  
  
 }  
 else  
 {  
 Toast.*makeText*(this, "Failure", Toast.*LENGTH\_SHORT*).show();  
 }  
  
 }  
}

**Second.java**

package com.example.loginform;  
import android.os.Bundle;  
import androidx.appcompat.app.AppCompatActivity;  
  
public class Second extends AppCompatActivity {  
 @Override  
 protected void onCreate(Bundle savedInstanceState) {  
 super.onCreate(savedInstanceState);  
 setContentView(R.layout.*show*);  
 }  
}

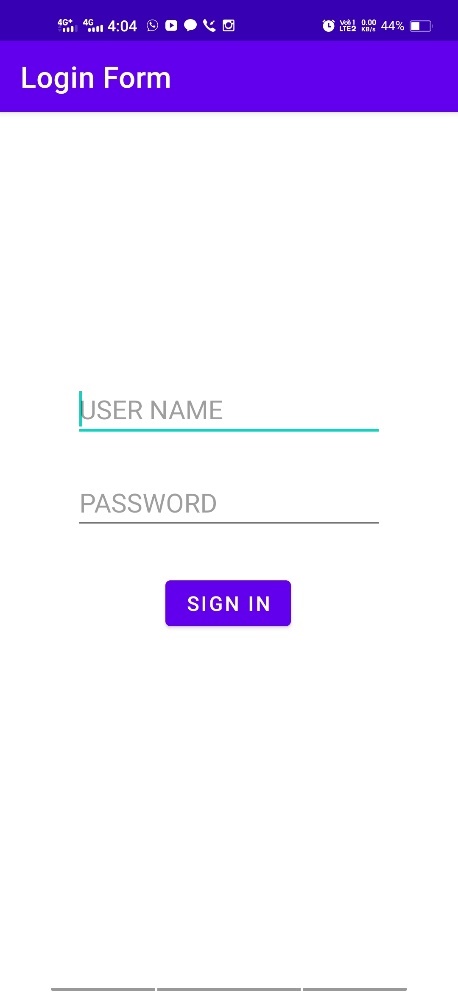
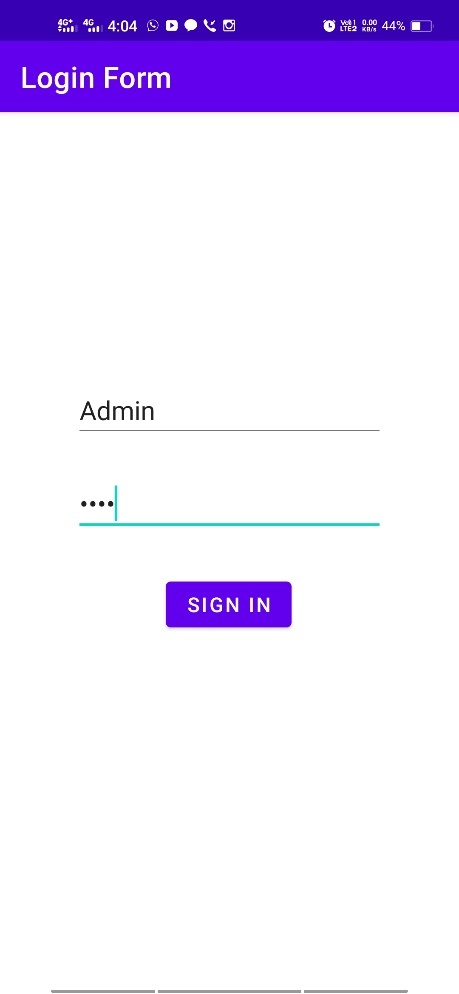
**show.xml**

<?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">  
  
 <TextView  
 android:id="@+id/textView"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:text="WELCOME"  
 tools:layout\_editor\_absoluteX="146dp"  
 tools:layout\_editor\_absoluteY="196dp"  
 tools:ignore="MissingConstraints" />  
</androidx.constraintlayout.widget.ConstraintLayout>

**Note\***

**Create activity.xml and mainactivity.java file then go to Project java folder🡪 Right click🡪new->Activity🡪Empty activity🡪 give file name as second.java and show.xml**

**OUTPUT**

**  **

**3.Create Implicit Intents**

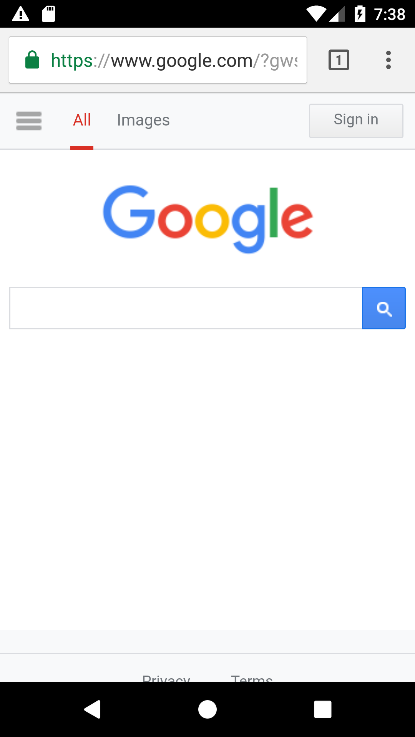
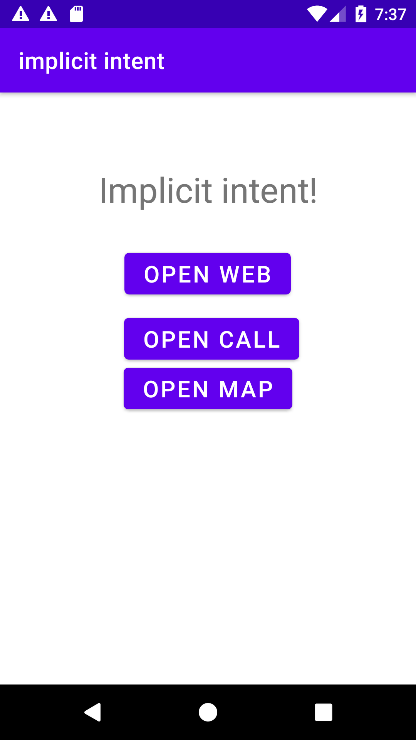
**activity\_main.xml**

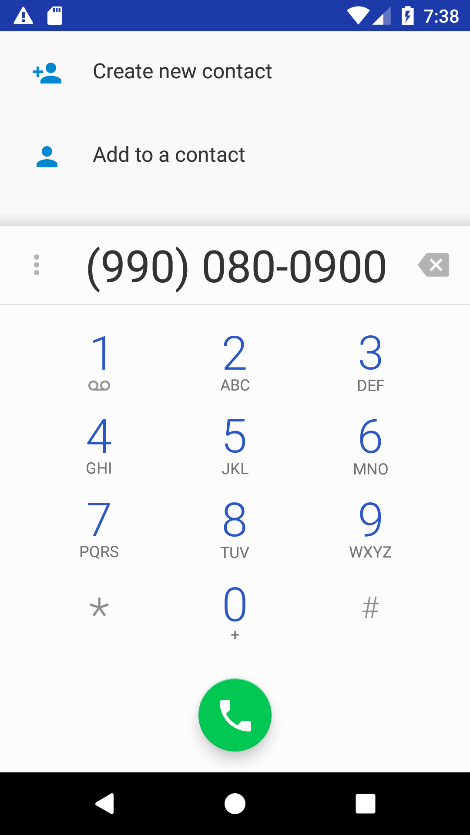
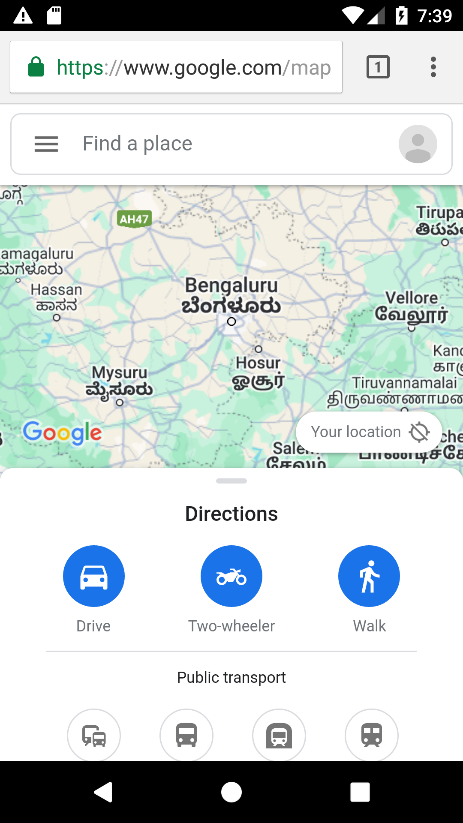
*<?*xml version="1.0" encoding="utf-8"*?>*<androidx.constraintlayout.widget.ConstraintLayout xmlns:android="http://schemas.android.com/apk/res/android"  
 xmlns:app="http://schemas.android.com/apk/res-auto"  
 xmlns:tools="http://schemas.android.com/tools"  
 android:layout\_width="match\_parent"  
 android:layout\_height="match\_parent"  
 tools:context=".MainActivity">  
  
 <TextView  
 android:id="@+id/textView"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:text="Implicit intent!"  
 android:textSize="30sp"  
 app:layout\_constraintBottom\_toBottomOf="parent"  
 app:layout\_constraintEnd\_toEndOf="parent"  
 app:layout\_constraintStart\_toStartOf="parent"  
 app:layout\_constraintTop\_toTopOf="parent"  
 app:layout\_constraintVertical\_bias="0.134" />  
  
 <Button  
 android:id="@+id/b1"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:text="open web"  
 android:onClick="dosomething"  
 android:textSize="20sp"  
 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="0.286" />  
  
 <Button  
 android:id="@+id/b2"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:text="open call"  
 android:onClick="dosomething"  
 android:textSize="20sp"  
 app:layout\_constraintBottom\_toBottomOf="parent"  
 app:layout\_constraintEnd\_toEndOf="parent"  
 app:layout\_constraintHorizontal\_bias="0.515"  
 app:layout\_constraintStart\_toStartOf="parent"  
 app:layout\_constraintTop\_toTopOf="parent"  
 app:layout\_constraintVertical\_bias="0.407" />  
  
 <Button  
 android:id="@+id/b3"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:onClick="dosomething"  
 android:text="open map"  
 android:textSize="20sp"  
 app:layout\_constraintBottom\_toBottomOf="parent"  
 app:layout\_constraintEnd\_toEndOf="parent"  
 app:layout\_constraintHorizontal\_bias="0.5"  
 app:layout\_constraintStart\_toStartOf="parent"  
 app:layout\_constraintTop\_toTopOf="parent" />  
  
</androidx.constraintlayout.widget.ConstraintLayout>

**MainActivity.java**

package com.example.implicitintent;  
  
import androidx.appcompat.app.AppCompatActivity;  
  
import android.content.Intent;  
import android.net.Uri;  
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 dosomething(View view) {  
 switch(view.getId()) {  
 case R.id.*b1*:  
 Intent i1=new Intent(Intent.*ACTION\_VIEW*, Uri.*parse*("http://www.google.com"));  
 startActivity(i1);  
 break;  
 case R.id.*b2*:  
 Intent i2=new Intent(Intent.*ACTION\_VIEW*, Uri.*parse*("tel:9900800900"));  
 startActivity(i2);  
 break;  
 case R.id.*b3*:  
 Intent i3=new Intent(Intent.*ACTION\_VIEW*, Uri.*parse*("geo:20.5937,78.9629"));  
 startActivity(i3);  
 break;  
 }  
 }  
}

**OUTPUT**



**4.“Happy Birth Day” App using Text View and Image View**

<?xmlversion="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:layout\_width="268dp"  
 android:layout\_height="84dp"  
 android:text="HappyBirthDay"  
 android:textSize="20sp"  
 app:layout\_constraintBottom\_toBottomOf="parent"  
 app:layout\_constraintEnd\_toEndOf="parent"  
 app:layout\_constraintHorizontal\_bias="0.372"  
 app:layout\_constraintStart\_toStartOf="parent"  
 app:layout\_constraintTop\_toTopOf="parent"  
 app:layout\_constraintVertical\_bias="0.141"  
 android:gravity="center"/>  
  
 <ImageView  
 android:id="@+id/imageView2"  
 android:layout\_width="136dp"  
 android:layout\_height="337dp"  
 app:srcCompat="@drawable/image" app:layout\_constraintBottom\_toBottomOf="parent"  
 app:layout\_constraintEnd\_toEndOf="parent"  
 app:layout\_constraintHorizontal\_bias="0.454"  
 app:layout\_constraintStart\_toStartOf="parent"  
 app:layout\_constraintTop\_toTopOf="parent"  
 app:layout\_constraintVertical\_bias="0.467"

tools:ignore="MissingConstraints"/>  
</androidx.constraintlayout.widget.ConstraintLayout>

**MainActivity.java**

packagecom.example.bday;  
importandroidx.appcompat.app.AppCompatActivity;  
importandroid.os.Bundle;

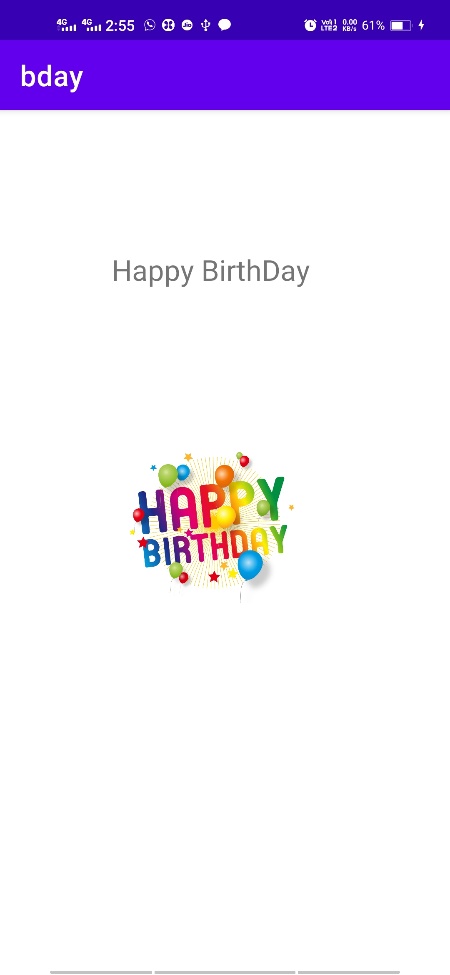
publicclassMainActivityextendsAppCompatActivity

{

@Override  
 protectedvoidonCreate(BundlesavedInstanceState)

{  
 super.onCreate(savedInstanceState);  
 setContentView(R.layout.*activity\_main*);  
 }  
}

**OUTPUT**

****

**5.Set and Retrieve shared preferences**

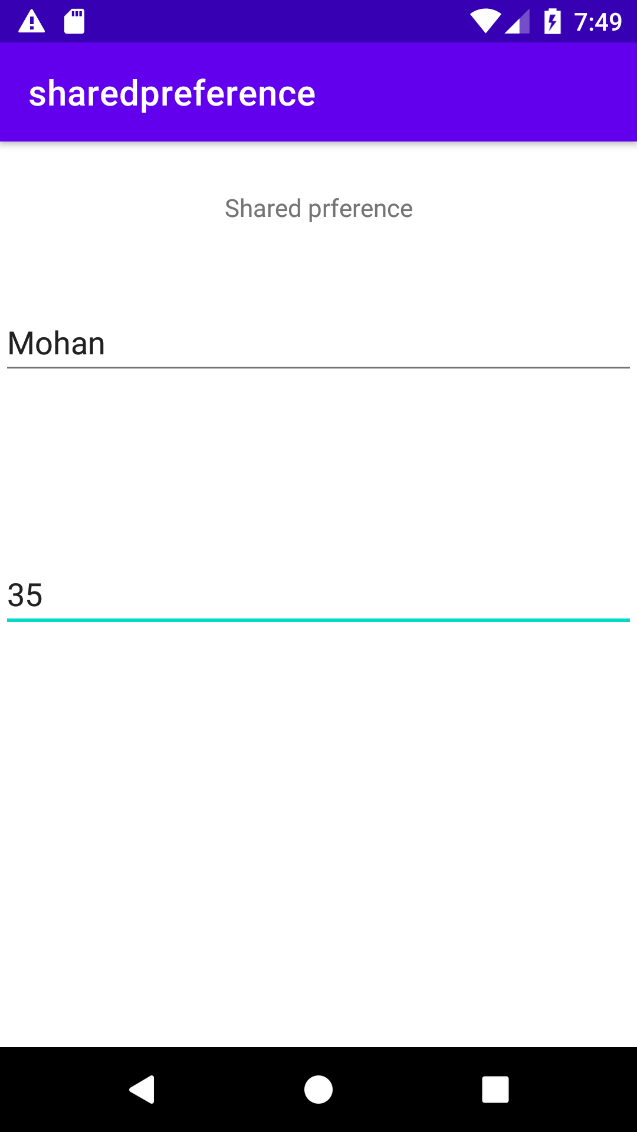
**activity\_main.xml**

*<?*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">  
  
 <TextView  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:text="Shared prference"  
 app:layout\_constraintBottom\_toBottomOf="parent"  
 app:layout\_constraintEnd\_toEndOf="parent"  
 app:layout\_constraintStart\_toStartOf="parent"  
 app:layout\_constraintTop\_toTopOf="parent"  
 app:layout\_constraintVertical\_bias="0.056" />  
  
 <EditText  
 android:id="@+id/edit1"  
 android:layout\_width="0dp"  
 android:layout\_height="wrap\_content"  
 android:ems="10"  
 android:hint="enter your name"  
 android:inputType="textPersonName"  
 app:layout\_constraintBottom\_toBottomOf="parent"  
 app:layout\_constraintEnd\_toEndOf="parent"  
 app:layout\_constraintHorizontal\_bias="1.0"  
 app:layout\_constraintStart\_toStartOf="parent"  
 app:layout\_constraintTop\_toTopOf="parent"  
 app:layout\_constraintVertical\_bias="0.195" />  
  
 <EditText  
 android:id="@+id/edit2"  
 android:layout\_width="0dp"  
 android:layout\_height="wrap\_content"  
 android:ems="10"  
 android:hint="enter your age"  
 android:inputType="number"  
 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.example.sharedpreference;  
  
import androidx.appcompat.app.AppCompatActivity;  
  
import android.content.Context;  
import android.content.SharedPreferences;  
import android.os.Bundle;  
import android.widget.EditText;  
  
public class MainActivity extends AppCompatActivity {  
EditText name,age;  
 @Override  
 protected void onCreate(Bundle savedInstanceState) {  
 super.onCreate(savedInstanceState);  
 setContentView(R.layout.*activity\_main*);  
 name=(EditText) findViewById(R.id.*edit1*);  
 age=(EditText) findViewById(R.id.*edit2*);  
 }  
  
 @Override  
 protected void onResume() {  
 super.onResume();  
 SharedPreferences sh1=getSharedPreferences("myownshared", Context.*MODE\_PRIVATE*);  
 String s1 =sh1.getString("user","");  
 int a1=sh1.getInt("age",0);  
 name.setText(s1);  
 age.setText(String.*valueOf*(a1));  
 }  
  
 @Override  
 protected void onPause() {  
 super.onPause();  
 SharedPreferences sh=getSharedPreferences("myownshared",*MODE\_PRIVATE*);  
 SharedPreferences.Editor myedit=sh.edit();  
 myedit.putString("user",name.getText().toString());  
 myedit.putInt("age",Integer.*parseInt*(age.getText().toString()));  
 myedit.commit();  
 }  
}

**OUTPUT**



**PART-B**

* 1. **Develop an android app which displays a form to get following information from user. 1) Username 2) Password 3) Email Address 4) Phone Number 5) Country Form should be followed by a Button with label “Submit”. When user clicks the button, a message should be displayed to user describing the information entered. Utilize suitable UI controls (i.e. widgets). [When user enters country in Auto Complete Text View, list of states should be displayed in Spinner automatically]**

**avtivitymain.xml**

<?xmlversion="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/username"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:layout\_marginStart="100dp"  
 android:layout\_marginTop="47dp"  
 android:layout\_marginEnd="101dp"  
 android:ems="10"  
 android:hint="Username"  
 android:inputType="textPersonName"  
 android:minHeight="48dp"  
 app:layout\_constraintEnd\_toEndOf="parent"  
 app:layout\_constraintStart\_toStartOf="parent"  
 app:layout\_constraintTop\_toTopOf="parent"/>  
  
 <EditText  
 android:id="@+id/password"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:layout\_marginStart="100dp"  
 android:layout\_marginTop="20dp"  
 android:layout\_marginEnd="101dp"  
 android:ems="10"  
 android:hint="Password"  
 android:inputType="textPassword"  
 android:minHeight="48dp"  
 app:layout\_constraintEnd\_toEndOf="parent"

app:layout\_constraintHorizontal\_bias="1.0"  
 app:layout\_constraintStart\_toStartOf="parent"  
 app:layout\_constraintTop\_toBottomOf="@+id/username"/>  
  
 <EditText  
 android:id="@+id/email"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:layout\_marginStart="100dp"  
 android:layout\_marginTop="17dp"  
 android:layout\_marginEnd="101dp"  
 android:ems="10"  
 android:hint="Email"  
 android:inputType="textEmailAddress"  
 android:minHeight="48dp"  
 app:layout\_constraintEnd\_toEndOf="parent"  
 app:layout\_constraintStart\_toStartOf="parent"  
 app:layout\_constraintTop\_toBottomOf="@+id/password"/>  
  
 <EditText  
 android:id="@+id/phone"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:layout\_marginStart="100dp"  
 android:layout\_marginTop="18dp"  
 android:layout\_marginEnd="101dp"  
 android:ems="10"  
 android:hint="PhoneNumber"  
 android:inputType="phone"  
 android:minHeight="48dp"  
 app:layout\_constraintEnd\_toEndOf="parent"  
 app:layout\_constraintStart\_toStartOf="parent"  
 app:layout\_constraintTop\_toBottomOf="@+id/email"/>  
  
 <Button  
 android:id="@+id/submit"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:layout\_marginStart="158dp"  
 android:layout\_marginTop="44dp"  
 android:layout\_marginEnd="159dp"  
 android:onClick="submitData"  
 android:text="Submit"  
 app:layout\_constraintEnd\_toEndOf="parent"  
 app:layout\_constraintHorizontal\_bias="0.666"  
 app:layout\_constraintStart\_toStartOf="parent"  
 app:layout\_constraintTop\_toBottomOf="@+id/country\_spinner"/

<Spinner  
 android:id="@+id/country\_spinner"  
 android:layout\_width="215dp"  
 android:layout\_height="48dp"  
 android:layout\_marginStart="1dp"  
 android:layout\_marginTop="28dp"  
 android:layout\_marginEnd="1dp"  
 app:layout\_constraintEnd\_toEndOf="parent"  
 app:layout\_constraintStart\_toStartOf="parent"  
 app:layout\_constraintTop\_toBottomOf="@+id/phone"  
 tools:ignore="SpeakableTextPresentCheck"/>  
  
</androidx.constraintlayout.widget.ConstraintLayout>

**Mainactivity.java**

package com.example.formvalidation;

import static android.content.ContentValues.TAG;

import androidx.appcompat.app.AlertDialog;

import androidx.appcompat.app.AppCompatActivity;

import android.os.Bundle;

import android.util.Log;

import android.view.View;

import android.widget.AdapterView;

import android.widget.ArrayAdapter;

import android.widget.Button;

import android.widget.EditText;

import android.widget.Spinner;

import android.widget.Toast;

import java.util.ArrayList;

import java.util.List;

publicclassMainActivityextendsAppCompatActivityimplements AdapterView.OnItemSelectedListener {

EditText username, password, phone, email;

Button submit;

Spinner spinner;

String allData = "", spinner\_text;

@Override

protected void onCreate(Bundle savedInstanceState) {

super.onCreate(savedInstanceState);

setContentView(R.layout.activity\_main);

//find spinner

spinner = (Spinner) findViewById(R.id.country\_spinner);

//Fetch the array of string and create adapter

ArrayAdapter<CharSequence>adapter=ArrayAdapter.createFromResource(this, R.array.locations, android.R.layout.simple\_spinner\_item);

//set drop-down spinner

adapter.setDropDownViewResource(android.R.layout.simple\_spinner\_dropdown\_item);

//set adapter

spinner.setAdapter(adapter);

spinner.setOnItemSelectedListener(this);

}

public void submitData(View view)

{

username = (EditText) findViewById(R.id.username);

password = (EditText) findViewById(R.id.password);

email = (EditText) findViewById(R.id.email);

phone = (EditText) findViewById(R.id.phone);

submit = (Button) findViewById(R.id.submit);

allData+="Name:"+username.getText().toString()+"\n"+"Email:"+email.getText().toString()+"\n"+"Password:"+password.getText().toString()+"\n"+"Phone:"+phone.getText().toString()+ "\n" + "Country: "+ spinner\_text+ "\n";

Toast.makeText(this, ""+allData, Toast.LENGTH\_SHORT).show();

allData = "";

}

@Override

public void onItemSelected(AdapterView<?> adapterView,View view, int i, long l)

{

spinner\_text = adapterView.getItemAtPosition(i).toString();

}

@Override

public void onNothingSelected(AdapterView<?> adapterView) {

}

**}**

**Strings.xml**

<resources>

<string name="app\_name">spinner</string>

<string-array name="locations">

<item>India</item>

<item>Canada</item>

<item>Japan</item>

<item>Germany</item>

<item>Switzerland</item>

<item>Australia</item>

<item>United State</item>

<item>New Zealand</item>

<item>United Kingdom</item>

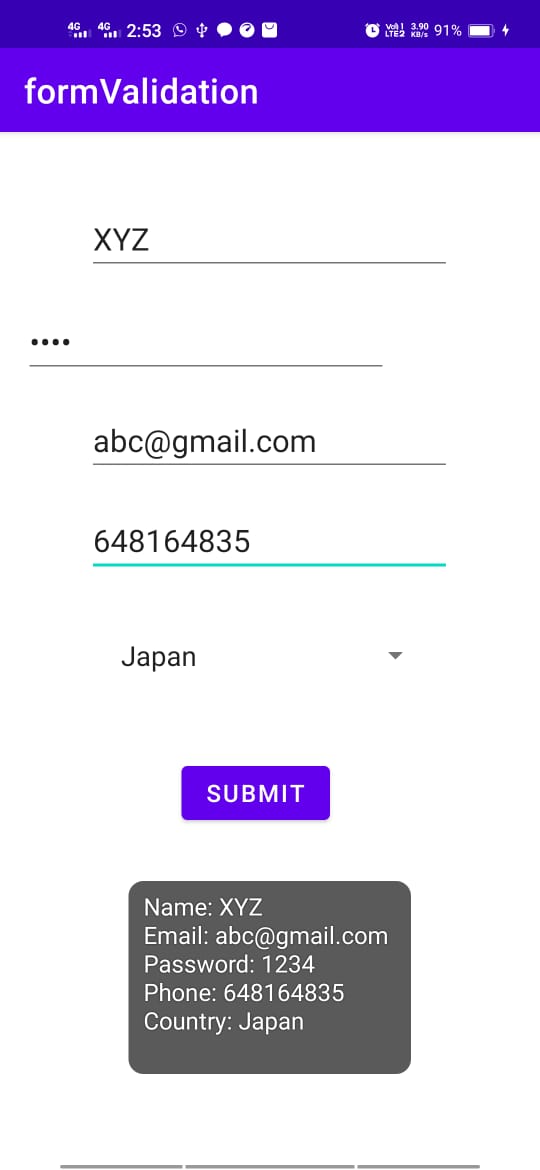
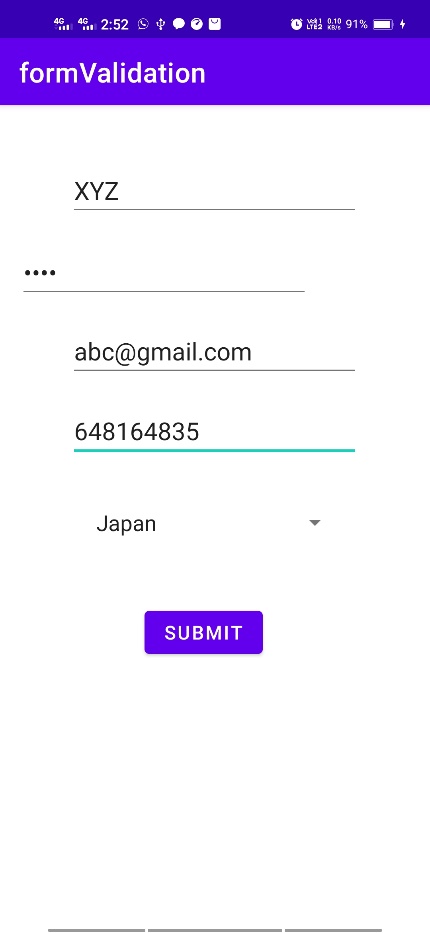
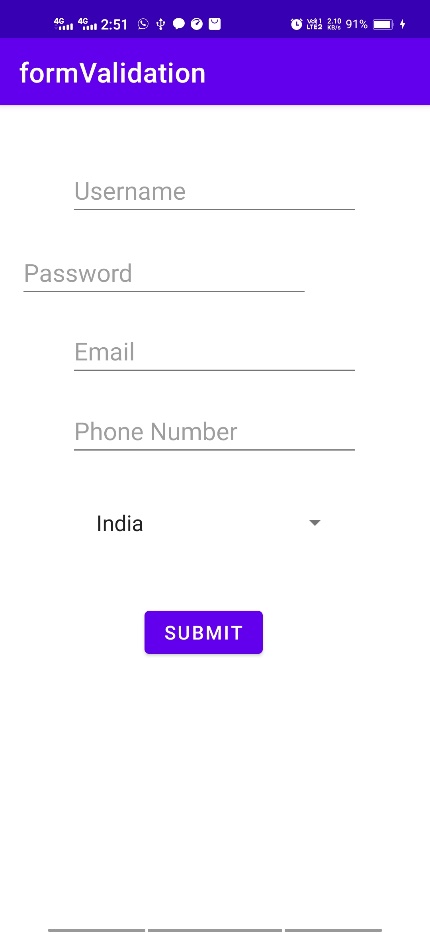
<item>Sweden</item>

</string-array>

</resources>

**Project🡪res folder🡪values folder🡪strings.xml. type the code for selecting city names**

**OUTPUT**

****

**2.The Simple Calculator app has two edit texts and four buttons. When you enter two numbers and click a button, the app performs the calculation for that button and displays the result.**

<?xmlversion="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:orientation="horizontal"  
 tools:context=".MainActivity">  
  
 <LinearLayout  
 android:id="@+id/linearLayout"  
 android:layout\_width="fill\_parent"  
 android:layout\_height="wrap\_content"  
 app:layout\_constraintEnd\_toEndOf="parent"  
 app:layout\_constraintStart\_toStartOf="parent"  
 app:layout\_constraintTop\_toTopOf="parent">  
  
 <EditText  
 android:id="@+id/firstNo"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:layout\_marginLeft="30dp"  
 android:layout\_marginTop="100dp"  
 android:gravity="center"  
 android:inputType="number"  
 android:backgroundTint="#FFBF00"  
 android:ems="7"  
 android:hint="EnterFirstNo"  
 android:minHeight="48dp"/>  
  
 <EditText  
 android:id="@+id/secondtNo"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:layout\_marginLeft="20dp"  
 android:layout\_marginTop="100dp"  
 android:backgroundTint="#FFBF00"  
 android:ems="7"  
 android:inputType="number"  
 android:gravity="center"  
 android:hint="EnterSecondNo"  
 android:minHeight="48dp"/>  
  
 </LinearLayout>  
  
 <TextView  
 android:id="@+id/output"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:layout\_marginTop="36dp"  
 android:gravity="center"  
 android:textSize="20sp"  
 app:layout\_constraintEnd\_toEndOf="parent"  
 app:layout\_constraintHorizontal\_bias="0.498"  
 app:layout\_constraintStart\_toStartOf="parent"  
 app:layout\_constraintTop\_toBottomOf="@+id/linearLayout"/>

<LinearLayout  
 android:id="@+id/linearLayout1"  
 android:layout\_width="match\_parent"  
 android:layout\_height="wrap\_content"  
 android:layout\_marginTop="30dp"  
 android:orientation="horizontal"  
 app:layout\_constraintTop\_toBottomOf="@+id/output">  
  
 <Button  
 android:onClick="add"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:text="+"  
 android:textColor="@color/black"  
 android:backgroundTint="#FFBF00"  
 android:gravity="center"  
 android:textSize="25dp"  
 android:layout\_marginLeft="10dp"

android:layout\_marginTop="10dp"/>  
  
 <Button  
 android:onClick="sub"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:textColor="@color/black"  
 android:text="-"  
 android:backgroundTint="#FFBF00"  
 android:gravity="center"  
 android:textSize="25dp"  
 android:layout\_marginLeft="10dp

android:layout\_marginTop="10dp"/>

<Button  
 android:onClick="mul"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:textColor="@color/black"  
 android:text="\*"  
 android:gravity="center"  
 android:textSize="25dp"  
 android:backgroundTint="#FFBF00"  
 android:layout\_marginLeft="10dp"

android:layout\_marginTop="10dp"/>  
  
 <Button  
 android:onClick="div"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:text="/"  
 android:gravity="center"  
 android:backgroundTint="#FFBF00"  
 android:textColor="@color/black"  
 android:layout\_marginLeft="10dp"  
 android:textSize="25dp

android:layout\_marginTop="10dp"/>  
  
 </LinearLayout>  
</androidx.constraintlayout.widget.ConstraintLayout>

**Mainactivity.java**

package com.example.simplecalculator;

import androidx.appcompat.app.AppCompatActivity;

import android.os.Bundle;

import android.view.View;

import android.widget.EditText;

import android.widget.TextView;

public class MainActivity extends AppCompatActivity

{

int number1, number2;

TextView output;

EditText no1,no2;

@Override

protected void onCreate(Bundle savedInstanceState)

{

super.onCreate(savedInstanceState);

setContentView(R.layout.activity\_main);

no1 = (EditText) findViewById(R.id.firstNo);

no2 = (EditText) findViewById(R.id.secondtNo);

output =(TextView) findViewById(R.id.output);

}

public void add(View view)

{

number1 = Integer.parseInt(no1.getText().toString());

number2 = Integer.parseInt(no2.getText().toString());

output.setText(String.valueOf(number1+number2));

}

public void sub(View view)

{

number1 = Integer.parseInt(no1.getText().toString());

number2 = Integer.parseInt(no2.getText().toString());

output.setText(String.valueOf(number1-number2));

}

public void mul(View view)

{

number1 = Integer.parseInt(no1.getText().toString());

number2 = Integer.parseInt(no2.getText().toString());

output.setText(String.valueOf(number1\*number2));

}

public void div(View view)

{

number1 = Integer.parseInt(no1.getText().toString());

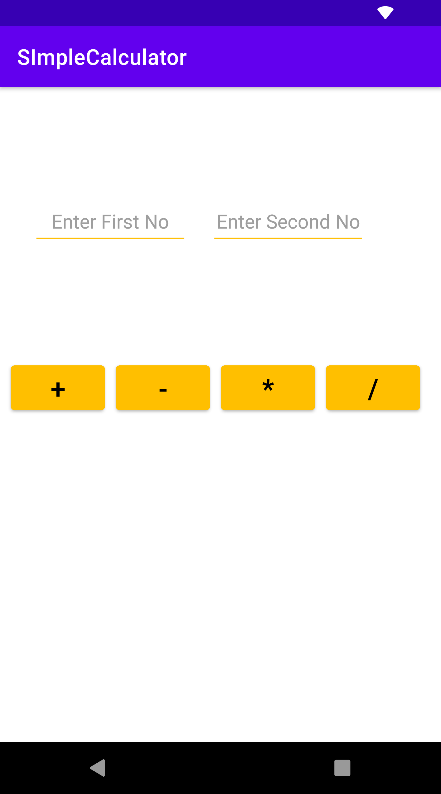
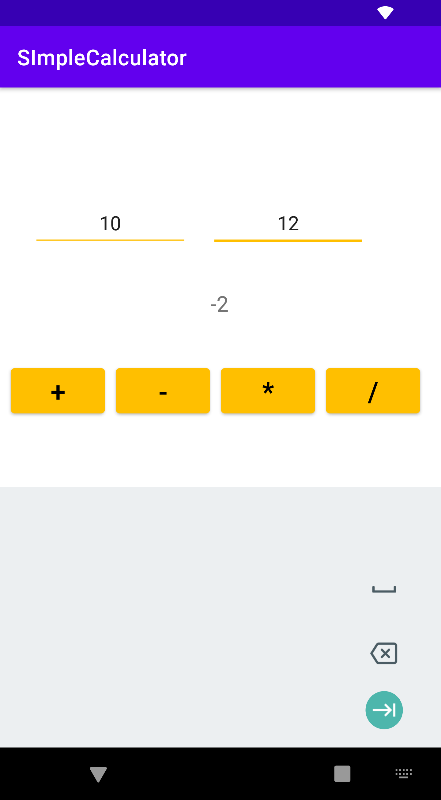
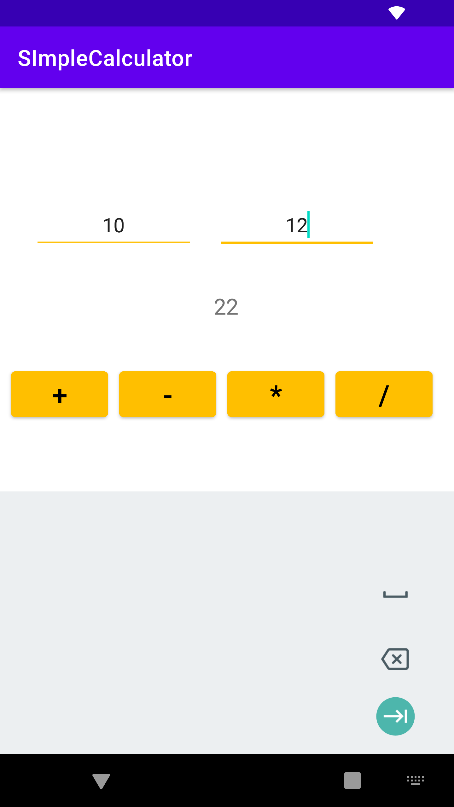
number2 = Integer.parseInt(no2.getText().toString());

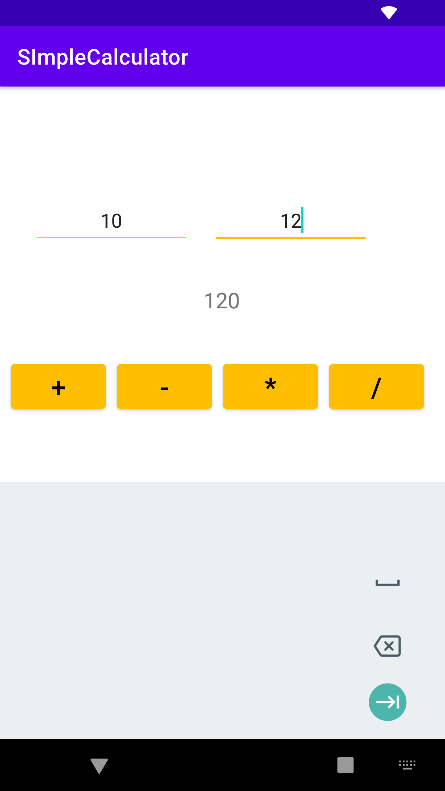
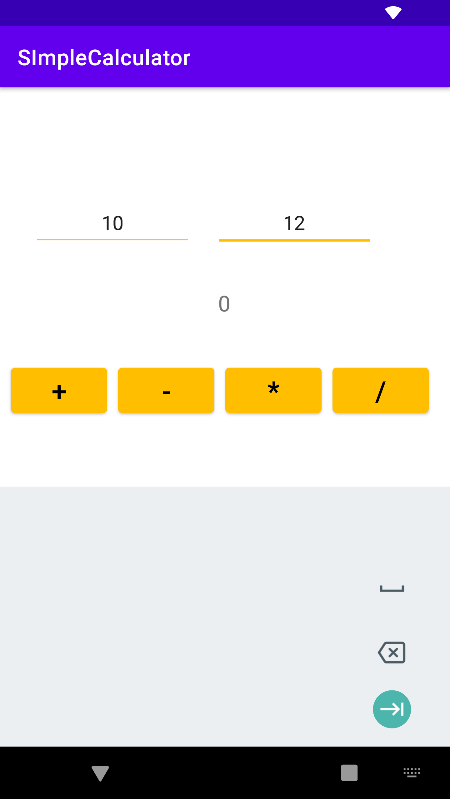
output.setText(String.valueOf(number1/number2));

}

}

**OUTPUT**

**3.Develop an android app for Text to Speech.**

**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/text"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:layout\_gravity="center"  
 android:layout\_marginTop="100dp"  
 android:ems="10"  
 android:hint="Enter your Text"  
 android:minHeight="48dp"  
 tools:ignore="MissingConstraints" />  
  
 <Button  
 android:onClick="voice"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:layout\_gravity="center"  
 android:layout\_marginTop="20dp"  
 android:text="Click Here"  
 android:backgroundTint="#FFBE1A"  
 app:layout\_constraintTop\_toBottomOf="@id/text"  
 tools:ignore="MissingConstraints" />  
</androidx.constraintlayout.widget.ConstraintLayout>

**Mainactivity.java**

package com.example.tts;

import androidx.appcompat.app.AppCompatActivity;

import android.os.Bundle;  
import android.speech.tts.TextToSpeech;  
import android.view.View;  
import android.widget.Button;  
import android.widget.EditText;  
  
public class MainActivity extends AppCompatActivity  
{  
  
 EditText textData;  
 TextToSpeech tts;  
 @Override  
 protected void onCreate(Bundle savedInstanceState)  
 {  
 super.onCreate(savedInstanceState);  
 setContentView(R.layout.*activity\_main*);  
 }  
 public void voice(View view)  
 {  
 textData = (EditText) findViewById(R.id.*text*);  
 tts = new TextToSpeech(getApplicationContext(), new TextToSpeech.OnInitListener()  
 {  
 @Override  
 public void onInit(int i)  
 {  
 if (i == TextToSpeech.*SUCCESS*)  
 {  
 tts.speak(textData.getText().toString(), TextToSpeech.*QUEUE\_ADD*, null, null);  
 }  
 }  
  
 });  
 }  
}

**OUTPUT**

****

**4.Create the MP3 player like application with service**

**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">  
  
 <TextView  
 android:id="@+id/tv"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:layout\_gravity="center"  
 android:layout\_marginTop="20px"  
 android:fontFamily="sans-serif-light"  
 android:text="NGI MUSIC PLAYER"  
 android:textSize="20dp"  
 android:textStyle="bold"  
 tools:ignore="MissingConstraints" />  
  
 <Button  
 android:id="@+id/btn1"  
 android:onClick="play"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:text="Play"  
 android:backgroundTint="#53BC09"  
 android:layout\_marginTop="20dp"  
 android:layout\_marginLeft="50dp"  
 app:layout\_constraintTop\_toBottomOf="@+id/tv"  
 tools:ignore="MissingConstraints" />  
 <Button  
 android:id="@+id/btn2"  
 android:onClick="pause"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:text="Pause"  
 app:layout\_constraintTop\_toBottomOf="@+id/btn1"  
 android:backgroundTint="#FF8400"  
 android:layout\_marginTop="20dp"  
 android:layout\_marginLeft="20dp"  
 tools:ignore="MissingConstraints" />

<Button  
 android:onClick="stop"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:text="Stop"  
 app:layout\_constraintTop\_toBottomOf="@+id/btn2"  
 android:backgroundTint="#ED1313"  
 android:layout\_marginTop="20dp"  
 android:layout\_marginLeft="50dp"  
 tools:ignore="MissingConstraints" />  
</androidx.constraintlayout.widget.ConstraintLayout>

**Main activity.java**

package com.example.musicplayer;

import androidx.appcompat.app.AppCompatActivity;

import android.media.MediaPlayer;

import android.os.Bundle;

import android.view.View;

public class MainActivity extends AppCompatActivity {

MediaPlayer player;

@Override

protected void onCreate(Bundle savedInstanceState) {

super.onCreate(savedInstanceState);

setContentView(R.layout.activity\_main);

player = MediaPlayer.create(this, R.raw.m);

}

}

public void play(View view){

player.start();

}

public void pause(View view){

player.pause();

}

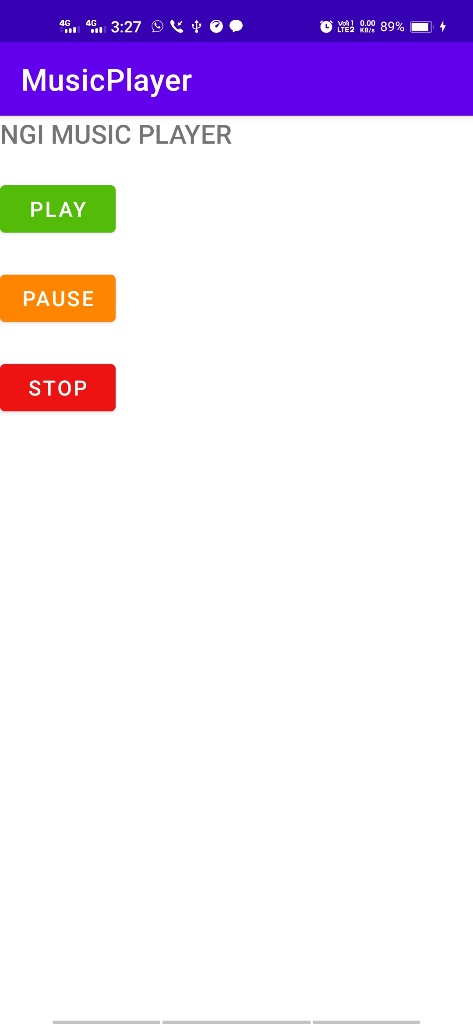
public void stop(View view){

player.stop();

}

**Note\* create new folder in res and give it name raw and paste the downloaded music file and name it as music**

**OUTPUT**

****

### **ANDROID VIVA QUESTIONS WITH ANSWERS**

### **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)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).

### **4)What is activity in Android?**

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

### **5)What are the life cycle methods of android activity?**

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

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

### **6)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

### **7)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.

### **8)Give a list of impotent folders in android**

The following folders are declared as impotent in android:

* AndroidManifest.xml
* build.xml
* bin/
* src/
* res/
* assets/

### **9)Explain the use of 'bundle' in android?**

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

### **10) 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.

### **11) 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.

### **12) Can the bytecode be written in java be run on android?**

No

### **13) List the various storages that are provided by Android.**

The various storage provided by android are:

* Shared Preferences
* Internal Storage
* External Storage
* SQLite Databases
* Network Connection

### **14) How are layouts placed in Android?**

Layouts in Android are placed as XML files.

### **15) Where are layouts placed in Android?**

Layouts in Android are placed in the layout folder.

### **16) What is the implicit intent in android?**

The Implicit intent is used to invoke the system components.

### **17) What is explicit intent in android?**

An explicit intent is used to invoke the activity class.

### **18) How to call another activity in android?**

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

### **19) 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.

### **20) What is the name of the database used in android?**

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

### **21) What is AAPT?**

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

### **22) What is a content provider?**

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

### **23) What is fragment?**

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

### **24) What is ADB?**

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

### **2) 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.

### **326) Which language does Android support to develop an application?**

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

### **27) What is ADT in Android?**

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

### **28) 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.

### **29) What is the Adapter in Android?**

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

### **30) What is nine-patch images tool in Android?**

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

### **31) Which kernel is used in Android?**

Android is a customized Linux 3.6 kernel.

### **32) What is application Widgets in Android?**

Application widgets are miniature application views that can be embedded in other applications and receive periodic updates.

### **33) 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.

### **34) Define Android Architecture?**

The Android architecture consists of 4 components:

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

### **35) Name the dialog box which is supported by Android?**

* Alert Dialog
* Progress Dialog
* Date Picker Dialog
* Time picker Dialog