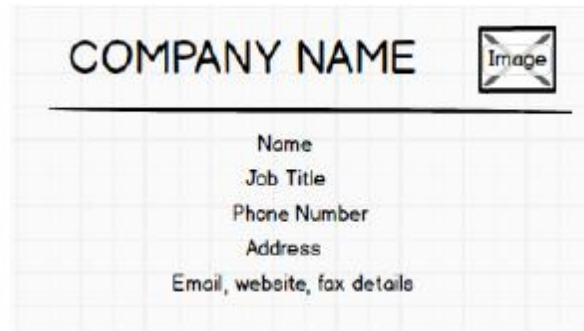


PART – A

Experiment-No: 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 is to be displayed. Insert a horizontal line between the job title and the phone number.



Components Used:

- TextView: To display texts
- ImageView: To display image(Logo)
(Add ImageView to design and in type choose "IC_LAUNCHER_FOREGROUND")
 - Download the logo & copy the same in res->drawable folder
 - In xml code of imageView change srcCompat="@drawable/logo")
- View: To display horizontal line

XML Code:

```
<?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="#070707"
android:scrollbarSize="45dp"
tools:context=".MainActivity">

<TextView
    android:id="@+id/textView"
    android:layout_width="253dp"
    android:layout_height="178dp"
    android:fontFamily="@font/aladin"
```

```
android:text="MERN stack web developer"
android:textAlignment="center"
android:textColor="#E91E63"
android:textSize="45sp"
android:textStyle="bold"
app:layout_constraintBottom_toBottomOf="parent"
app:layout_constraintEnd_toEndOf="parent"
app:layout_constraintHorizontal_bias="0.031"
app:layout_constraintStart_toStartOf="parent"
app:layout_constraintTop_toTopOf="parent"
app:layout_constraintVertical_bias="0.045" />
```

<ImageView

```
android:id="@+id/imageView2"
android:layout_width="148dp"
android:layout_height="180dp"
app:layout_constraintBottom_toBottomOf="parent"
app:layout_constraintEnd_toEndOf="parent"
app:layout_constraintHorizontal_bias="0.937"
app:layout_constraintStart_toStartOf="parent"
app:layout_constraintTop_toTopOf="parent"
app:layout_constraintVertical_bias="0.043"
app:srcCompat="@drawable/imagep" />
```

<View

```
android:id="@+id/view"
android:layout_width="370dp"
android:layout_height="10dp"
android:background="#E91E63"
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="0.331" />
```

<TextView

```
android:id="@+id/name"
android:layout_width="327dp"
android:layout_height="84dp"
android:fontFamily="@font/aladin"
android:text="Vignesh"
android:textAlignment="center"
android:textColor="#E91E63"
android:textSize="45sp"
app:layout_constraintBottom_toBottomOf="parent"
app:layout_constraintEnd_toEndOf="parent"
```

```

        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toTopOf="parent"
        app:layout_constraintVertical_bias="0.429" />

<TextView
    android:id="@+id/textView3"
    android:layout_width="252dp"
    android:layout_height="84dp"
    android:fontFamily="@font/aladin"
    android:text="Student"
    android:textAlignment="center"
    android:textColor="#E91E63"
    android:textSize="45sp"
    app:layout_constraintBottom_toBottomOf="parent"
    app:layout_constraintEnd_toEndOf="parent"
    app:layout_constraintHorizontal_bias="0.496"
    app:layout_constraintStart_toStartOf="parent"
    app:layout_constraintTop_toTopOf="parent"
    app:layout_constraintVertical_bias="0.593" />

<TextView
    android:id="@+id/textView4"
    android:layout_width="346dp"
    android:layout_height="125dp"
    android:fontFamily="@font/aladin"
    android:text="vigneshkundar125@gmail.com /n 9071077931"
    android:textAlignment="center"
    android:textColor="#E91E63"
    android:textSize="30sp"
    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="0.817" />
</androidx.constraintlayout.widget.ConstraintLayout>

```

Java Code:

```

package com.example.program1;

import androidx.appcompat.app.AppCompatActivity;

import android.os.Bundle;
import android.widget.TextView;

public class MainActivity extends AppCompatActivity {

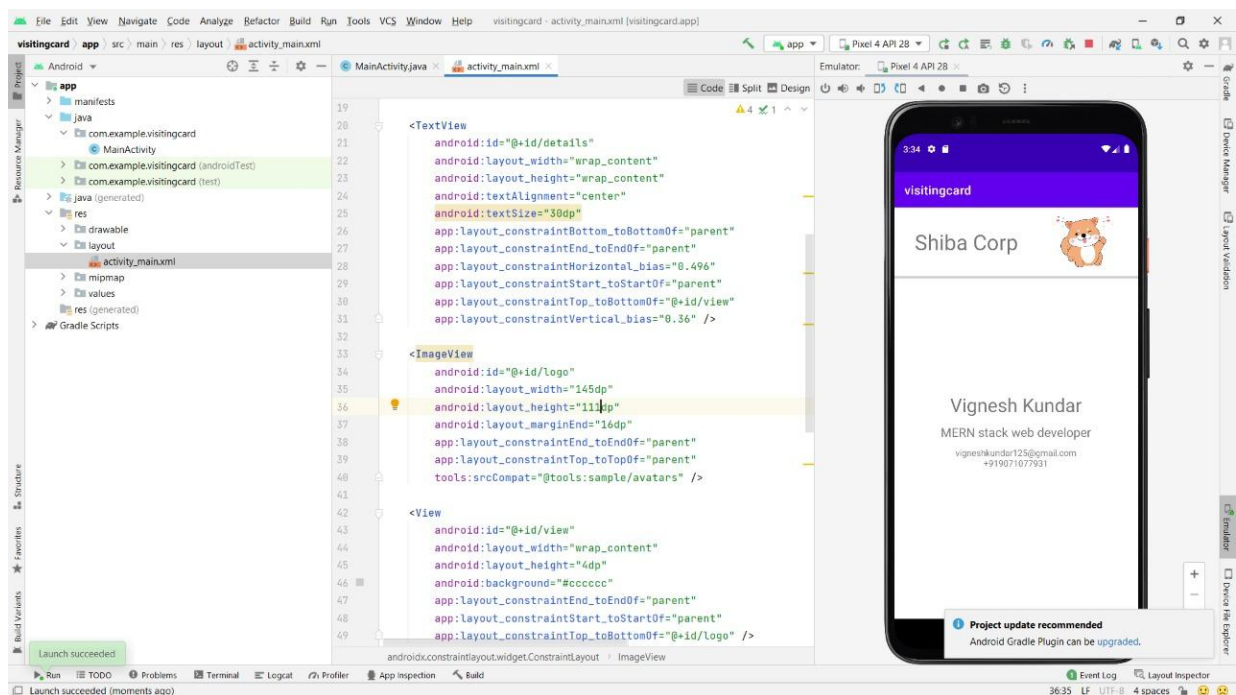
```

```

TextView name;
@Override
protected void onCreate(Bundle savedInstanceState) {
    super.onCreate(savedInstanceState);
    setContentView(R.layout.activity_main);
    name = findViewById(R.id.name);
    name.setText("Vignesh Kunder \n MERN stack web developer \n
Vigneshkunder125@gmail.com \n +919071077931");
}
}

```

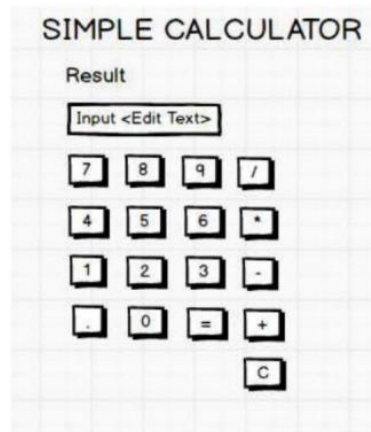
Sample Output:



PART – A

Experiment-No: 2

Develop an Android application using controls like Button, TextView, EditText for designing a calculator having basic functionality like Addition, Subtraction, Multiplication, and Division.



Components Used:

- Button: To display numbers, operations
- EditText: To display Result
- EditText: To display text(Calculator)

XML Code:

```
<?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/linearLayout"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:background="#0C0C0C"
    tools:context=".MainActivity">

    <TextView
        android:id="@+id/textView"
        android:layout_width="407dp"
        android:layout_height="166dp"
        android:text="Simple Calculator"
        android:textAlignment="center"
        android:textColor="#E91E63"
        android:textSize="60sp"
        app:layout_constraintBottom_toBottomOf="parent"
        app:layout_constraintTop_toTopOf="parent"
```

```
app:layout_constraintVertical_bias="0.0"  
tools:layout_editor_absoluteX="0dp" />
```

```
<EditText
```

```
    android:id="@+id/editTextTextPersonName3"  
    android:layout_width="213dp"  
    android:layout_height="39dp"  
    android:ems="10"  
    android:hint="Enter the first number"  
    android:inputType="numberDecimal"  
    android:minHeight="48dp"  
    android:textColor="#F3EFEF"  
    android:textColorHint="#F8F5F5"  
    app:layout_constraintBottom_toBottomOf="parent"  
    app:layout_constraintEnd_toEndOf="parent"  
    app:layout_constraintStart_toStartOf="parent"  
    app:layout_constraintTop_toTopOf="parent"  
    app:layout_constraintVertical_bias="0.275" />
```

```
<EditText
```

```
    android:id="@+id/editTextTextPersonName4"  
    android:layout_width="210dp"  
    android:layout_height="wrap_content"  
    android:ems="10"  
    android:hint="Enter the second number"  
    android:inputType="numberDecimal"  
    android:minHeight="48dp"  
    android:textColor="#F8F5F5"  
    android:textColorHint="#FAF7F7"  
    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="0.364" />
```

```
<TextView
```

```
    android:id="@+id/textView2"  
    android:layout_width="211dp"  
    android:layout_height="80dp"  
    android:text="0"  
    android:textAlignment="center"  
    android:textColor="#2196F3"  
    android:textSize="34sp"  
    app:layout_constraintBottom_toBottomOf="parent"  
    app:layout_constraintEnd_toEndOf="parent"  
    app:layout_constraintHorizontal_bias="0.495"
```

```
app:layout_constraintStart_toStartOf="parent"
app:layout_constraintTop_toTopOf="parent"
app:layout_constraintVertical_bias="0.937" />
```

```
<Button
    android:id="@+id/button"
    android:layout_width="124dp"
    android:layout_height="66dp"
    android:onClick="doAdd"
    android:text="SUM"
    app:layout_constraintBottom_toBottomOf="parent"
    app:layout_constraintEnd_toEndOf="parent"
    app:layout_constraintHorizontal_bias="0.836"
    app:layout_constraintStart_toStartOf="parent"
    app:layout_constraintTop_toTopOf="@+id/textView"
    app:layout_constraintVertical_bias="0.52" />
```

```
<Button
    android:id="@+id/button2"
    android:layout_width="127dp"
    android:layout_height="62dp"
    android:onClick="doSub"
    android:text="SUB"
    app:layout_constraintBottom_toBottomOf="parent"
    app:layout_constraintEnd_toEndOf="parent"
    app:layout_constraintHorizontal_bias="0.154"
    app:layout_constraintStart_toStartOf="parent"
    app:layout_constraintTop_toTopOf="@+id/textView"
    app:layout_constraintVertical_bias="0.517" />
```

```
<Button
    android:id="@+id/button3"
    android:layout_width="126dp"
    android:layout_height="65dp"
    android:onClick="doMul"
    android:text="MUL"
    app:layout_constraintBottom_toBottomOf="parent"
    app:layout_constraintEnd_toEndOf="parent"
    app:layout_constraintHorizontal_bias="0.154"
    app:layout_constraintStart_toStartOf="parent"
    app:layout_constraintTop_toTopOf="@+id/textView"
    app:layout_constraintVertical_bias="0.672" />
```

```
<Button
    android:id="@+id/button4"
    android:layout_width="127dp"
    android:layout_height="68dp"
```

```

        android:onClick="doDiv"
        android:text="DIV"
        app:layout_constraintBottom_toBottomOf="parent"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintHorizontal_bias="0.845"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toTopOf="parent"
        app:layout_constraintVertical_bias="0.675" />

<Button
    android:id="@+id/button5"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:onClick="clrTxt"
    android:text="Clear"
    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="@+id/textView"
    app:layout_constraintVertical_bias="0.799" />
</androidx.constraintlayout.widget.ConstraintLayout>

```

Java Code:

```

package com.example.calci;

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 {
    EditText e1,e2;
    TextView tv1;
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);

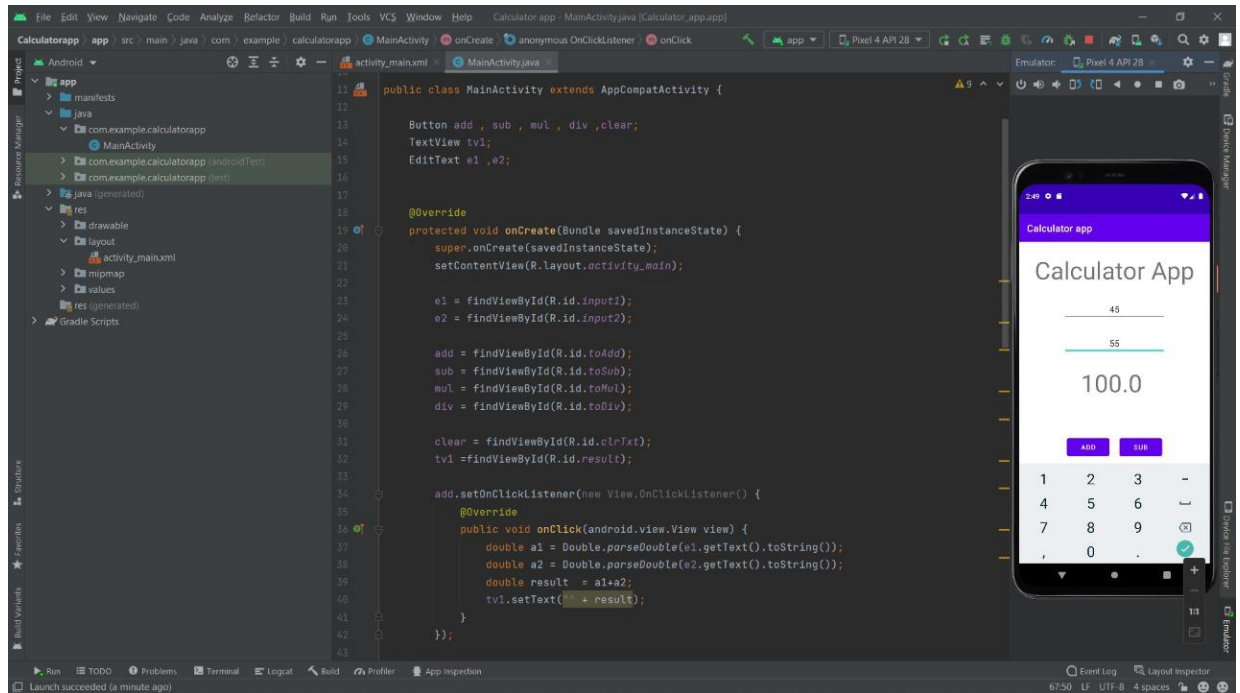
        e1 = (EditText) findViewById(R.id.editTextTextPersonName3);
        e2 = (EditText) findViewById(R.id.editTextTextPersonName4);
        tv1 = (TextView) findViewById(R.id.textView2);
    }
}

```



```
public void doAdd(View V){
    int a1 = Integer.parseInt(e1.getText().toString());
    int a2 = Integer.parseInt(e2.getText().toString());
    int result = a1+a2;
    tv1.setText(""+result);
}
public void doSub(View V){
    int a1 = Integer.parseInt(e1.getText().toString());
    int a2 = Integer.parseInt(e2.getText().toString());
    int result = a1-a2;
    tv1.setText(""+result);
}
public void doMul(View V){
    int a1 = Integer.parseInt(e1.getText().toString());
    int a2 = Integer.parseInt(e2.getText().toString());
    int result = a1*a2;
    tv1.setText(""+result);
}
public void doDiv(View V){
    int a1 = Integer.parseInt(e1.getText().toString());
    int a2 = Integer.parseInt(e2.getText().toString());
    int result = a1/a2;
    tv1.setText(""+result);
}

public void clrTxt(View v)
{
    tv1.setText("0");
    e1.setText(null);
    e2.setText(null);
}
}
```

Sample Output:

PART – A

Experiment-No: 3

Create a SIGN UP activity with Username and Password. Validation of password should happen based on the following rules:

- Password should contain uppercase and lowercase letters.
- Password should contain letters and numbers.
- Password should contain special characters.
- Minimum length of the password (the default value is 8).

On successful SIGN UP proceed to the next Login activity. Here the user should SIGN IN using the Username and Password created during signup activity. If the Username and Password are matched then navigate to the next activity which displays a message saying “Successful Login” or else display a toast message saying “Login Failed”. The user is given only two attempts and after that display a toast message saying “Failed Login Attempts” and disable the SIGN IN button. Use Bundle to transfer information from one activity to other.

The diagram illustrates the UI layout for two activities. On the left, the 'SIGNUP ACTIVITY' features a 'Username:' label followed by a text input field, a 'Password:' label followed by a text input field, and a 'SIGN UP' button at the bottom. On the right, the 'LOGIN ACTIVITY' features a 'Username:' label followed by a text input field, a 'Password:' label followed by a text input field, and a 'SIGN IN' button at the bottom. Both activities are shown on a light gray grid background.

Components Used:

- TextView: To Add text
- EditText: To enter Username and Password
- Button: For Login and Signup

Functions and Methods Used:

- Pattern - For Password validation.
- Bundle - To store and transfer data to another activity.
- Intent - For controlling and starting the activity.

XML Code:

activity_main.xml Code:

```
<?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="#74A6CD"
tools:context=".MainActivity">
```

```
<TextView
    android:id="@+id/textView"
    android:layout_width="320dp"
    android:layout_height="120dp"
    android:fontFamily="@font/akaya_telivigala"
    android:text="SignUp Activity"
    android:textAlignment="center"
    android:textColor="#F8F6F6"
    android:textSize="46sp"
    android:textStyle="bold"
    app:layout_constraintBottom_toBottomOf="parent"
    app:layout_constraintEnd_toEndOf="parent"
    app:layout_constraintStart_toStartOf="parent"
    app:layout_constraintTop_toTopOf="parent"
    app:layout_constraintVertical_bias="0.132" />
```

```
<EditText
    android:id="@+id/uname"
    android:layout_width="282dp"
    android:layout_height="67dp"
    android:ems="10"
    android:hint="Username"
    android:inputType="textPersonName"
    android:textColor="#E60808"
    android:textStyle="bold"
    app:layout_constraintBottom_toBottomOf="parent"
    app:layout_constraintEnd_toEndOf="parent"
    app:layout_constraintStart_toStartOf="parent"
    app:layout_constraintTop_toTopOf="parent" />
```

```
<EditText
    android:id="@+id/pwd"
    android:layout_width="273dp"
    android:layout_height="65dp"
    android:ems="10"
    android:hint="Password"
    android:inputType="textPassword"
```

```

        android:textColor="#BC0F0F"
        android:textStyle="bold"
        app:layout_constraintBottom_toBottomOf="parent"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toTopOf="parent"
        app:layout_constraintVertical_bias="0.662" />

<Button
    android:id="@+id/submit"
    android:layout_width="139dp"
    android:layout_height="85dp"
    android:background="#FF5722"
    android:backgroundTint="#FF5722"
    android:backgroundTintMode="add"
    android:text="SIGNUP"
    android:textColor="#FAF7F6"
    app:layout_constraintBottom_toBottomOf="parent"
    app:layout_constraintEnd_toEndOf="parent"
    app:layout_constraintStart_toStartOf="parent"
    app:layout_constraintTop_toTopOf="parent"
    app:layout_constraintVertical_bias="0.891" />
</androidx.constraintlayout.widget.ConstraintLayout>

```

activity_login.xml Code:

```

<?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=".login">

    <TextView
        android:id="@+id/textView2"
        android:layout_width="377dp"
        android:layout_height="147dp"
        android:background="#74A6CD"
        android:fontFamily="@font/akaya_telivigala"
        android:paddingTop="20sp"
        android:text="LogIn Activity"
        android:textAlignment="center"
        android:textColor="#0A1B7C"
        android:textSize="56sp"
        android:textStyle="bold"

```

```
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="0.027" />
```

```
<EditText
    android:id="@+id/loginuname"
    android:layout_width="352dp"
    android:layout_height="61dp"
    android:ems="10"
    android:hint="Username"
    android:inputType="textPersonName"
    android:textColor="#041051"
    app:layout_constraintBottom_toBottomOf="parent"
    app:layout_constraintEnd_toEndOf="parent"
    app:layout_constraintHorizontal_bias="0.508"
    app:layout_constraintStart_toStartOf="parent"
    app:layout_constraintTop_toTopOf="parent"
    app:layout_constraintVertical_bias="0.417" />
```

```
<EditText
    android:id="@+id/loginpwd"
    android:layout_width="352dp"
    android:layout_height="70dp"
    android:hint="Password"
    android:inputType="textPassword"
    android:textColor="#0D1B67"
    app:layout_constraintBottom_toBottomOf="parent"
    app:layout_constraintEnd_toEndOf="parent"
    app:layout_constraintHorizontal_bias="0.508"
    app:layout_constraintStart_toStartOf="parent"
    app:layout_constraintTop_toTopOf="parent"
    app:layout_constraintVertical_bias="0.611" />
```

```
<Button
    android:id="@+id/login"
    android:layout_width="166dp"
    android:layout_height="93dp"
    android:text="LOGIN"
    app:layout_constraintBottom_toBottomOf="parent"
    app:layout_constraintEnd_toEndOf="parent"
    app:layout_constraintHorizontal_bias="0.497"
    app:layout_constraintStart_toStartOf="parent"
    app:layout_constraintTop_toTopOf="parent"
```

```
        app:layout_constraintVertical_bias="0.873" />
</androidx.constraintlayout.widget.ConstraintLayout>
```

activity_success.xml Code:

```
<?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=".success">

    <TextView
        android:id="@+id/textView3"
        android:layout_width="354dp"
        android:layout_height="213dp"
        android:background="#74A6CD"
        android:fontFamily="@font/akaya_telivigala"
        android:paddingTop="40sp"
        android:text="LogIn Successful"
        android:textAlignment="center"
        android:textColor="#061A88"
        android:textSize="50sp"
        android:textStyle="bold"
        app:layout_constraintBottom_toBottomOf="parent"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintHorizontal_bias="0.609"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toTopOf="parent"
        app:layout_constraintVertical_bias="0.444" />
</androidx.constraintlayout.widget.ConstraintLayout>
```

Java Code:**MainActivity.java:**

```
package com.example.signup;

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;

import java.util.regex.Pattern;

public class MainActivity extends AppCompatActivity {
    private Button signup;
    private EditText uname,pwd;
    static String Username, Password;
    private static final String password_pattern = "^(?=.*[0-9])(?=.*[A-Z])(?=.*[a-z])(?=.*[@$#%^&+=!])(?=\S+$).{8,}$";

    Pattern psPattern = Pattern.compile(password_pattern);

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);
        signup = findViewById(R.id.submit);
        uname = findViewById(R.id.uname);
        pwd = findViewById(R.id.pwd);

        signup.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View view) {
                Username = uname.getText().toString();
                Password = pwd.getText().toString();

                if (Username.equals("") || Password.equals("")) {
                    Toast.makeText(MainActivity.this, "Username or Password cannot be empty",
                    Toast.LENGTH_SHORT).show();
                }
                else if (isValid()) {
                    Bundle bundle = new Bundle();
                    bundle.putString("Username", Username);
                    bundle.putString("Password", Password);
                    Intent intent = new Intent(getApplicationContext(), login.class);
                    intent.putExtras(bundle);

                    startActivity(intent);
                }
            }
        });
    }

    private boolean isValid() {
        if (!psPattern.matcher(Password).matches()) {

```



```

        Toast.makeText(MainActivity.this, "• Password should contain uppercase and
lowercase letters.\n" +
        "• Password should contain letters and numbers.\n" +
        "• Password should contain special characters.\n" +
        "• Minimum length of the password (the default value is 8).",
        Toast.LENGTH_LONG).show();
        return false;
    } else return true;

    }
}

```

login.java:

```

package com.example.signup;

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 login extends AppCompatActivity {
    private Button btnLogin;
    private EditText password, username;
    private String strname, strpwd;
    private int counter = 0;
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_login);
        btnLogin = findViewById(R.id.login);
        password = findViewById(R.id.loginpwd);
        username = findViewById(R.id.loginuname);

        Bundle bundle = getIntent().getExtras();
        strname = bundle.getString("Username");
        strpwd = bundle.getString("Password");

        Intent intent = new Intent(getApplicationContext(), success.class);

        btnLogin.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View view) {

```

```
        if(counter<=2){
            if (username.getText().toString().equals(strname) &&
password.getText().toString().equals(strpwd)){
                startActivity(intent);
            }
            else{
                Toast.makeText(login.this, "Invalid Username or Password",
Toast.LENGTH_SHORT).show();
                counter++;
                username.setText(null);
                password.setText(null);
            }
        }
        else{
            Toast.makeText(login.this, "Maximum attempts reached",
Toast.LENGTH_SHORT).show();
            btnLogin.setEnabled(false);
        }
    }
    });
}
```

success.java:

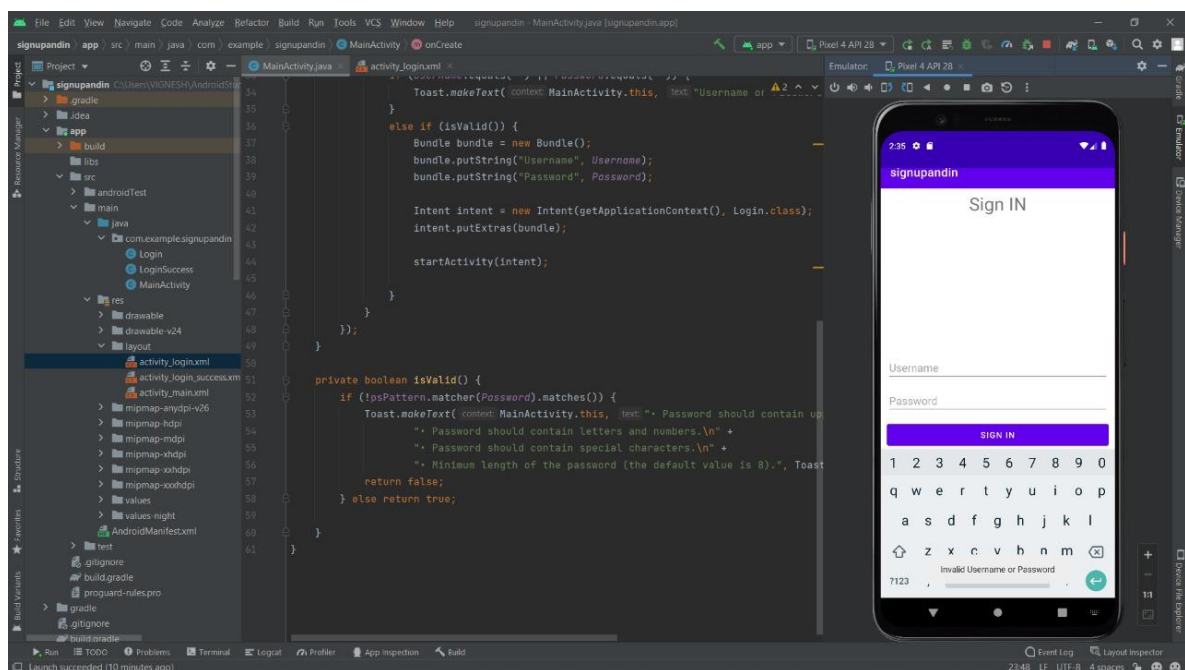
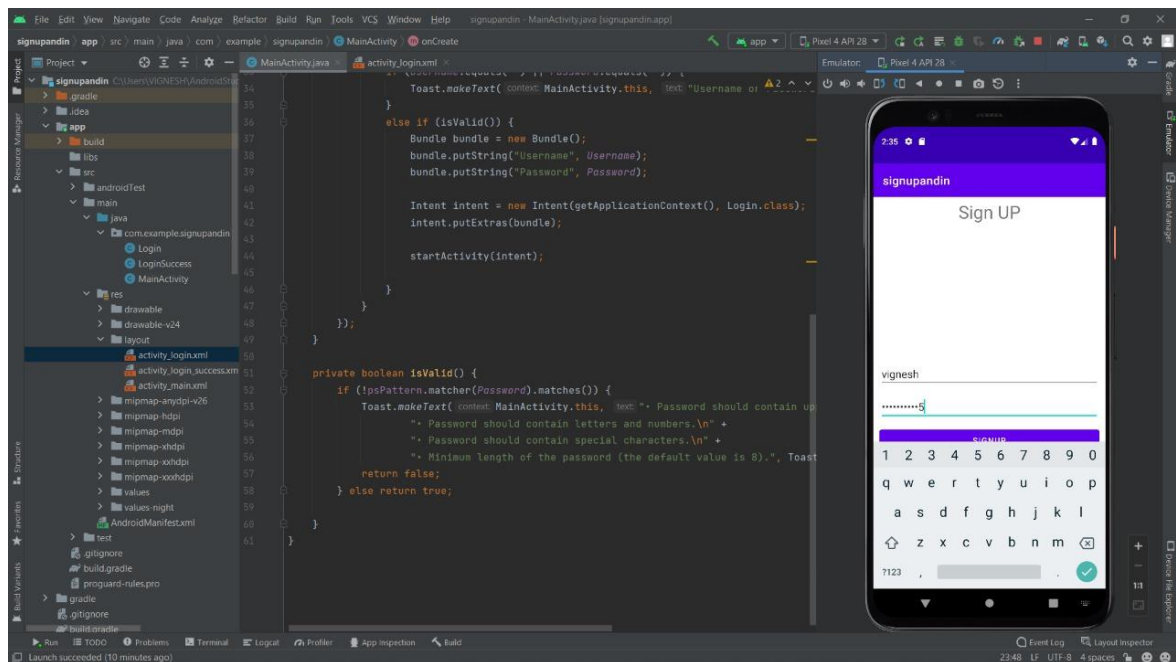
```
package com.example.signup;

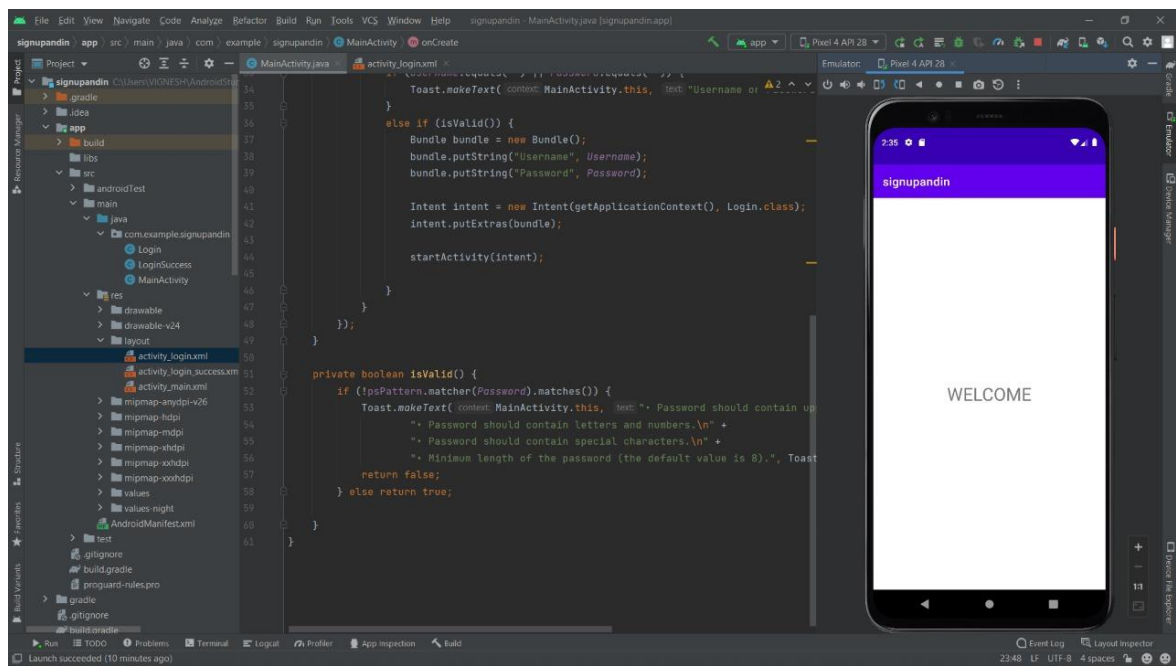
import androidx.appcompat.app.AppCompatActivity;

import android.os.Bundle;

public class success extends AppCompatActivity {

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_success);
    }
}
```

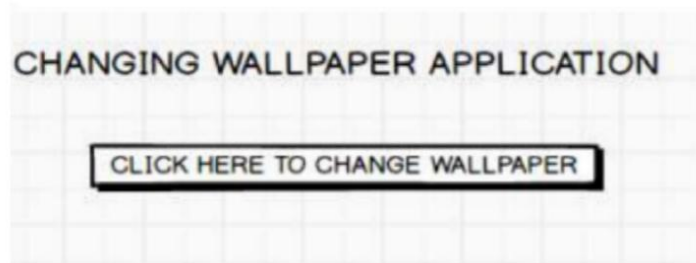
Sample Output:



PART – A

Experiment-No: 4

Develop an application to set an image as wallpaper. On click of a button, the wallpaper image should start to change randomly every 30 seconds.



Components Used:

- Button: To change Wallpaper

XML Code:

```
<?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">

<LinearLayout
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:background="#DFB8C6"
    android:orientation="vertical">

    <TextView
        android:id="@+id/textView"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_marginTop="100dp"
        android:fontFamily="cursive"
        android:text="Changing Wallpaper Application"
        android:textAlignment="center"
        android:textColor="#E91E63"
        android:textSize="50sp"
        android:textStyle="bold" />

    <Button
```

```

        android:id="@+id/button"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_marginLeft="30dp"
        android:layout_marginTop="50dp"
        android:layout_marginRight="30dp"
        android:backgroundTint="#E91E63"
        android:fontFamily="cursive"
        android:text="Click to Change"
        android:textColor="#EFEFEF" />
    </LinearLayout>
</androidx.constraintlayout.widget.ConstraintLayout>

```

Java Code:

```

package com.example.prog4;

import androidx.appcompat.app.AppCompatActivity;

import android.app.WallpaperManager;
import android.graphics.Bitmap;
import android.graphics.BitmapFactory;
import android.os.Bundle;
import android.util.DisplayMetrics;
import android.view.View;
import android.widget.Button;

public class MainActivity extends AppCompatActivity {
    private WallpaperManager wallpaperManager=null;
    Button btn;
    Bitmap wlp1;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);

        wallpaperManager=WallpaperManager.getInstance(this);

        btn=(Button)findViewById(R.id.button);
        btn.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View view) {
                try {
                    while(true) {
                        wlp1 = BitmapFactory.decodeResource(getResources(), R.drawable.img7);
                        DisplayMetrics metrics = new DisplayMetrics();

```

```

getWindowManager().getDefaultDisplay().getMetrics(metrics);
int height = metrics.heightPixels;
int width = metrics.widthPixels;
Bitmap scaledwlp1 = Bitmap.createScaledBitmap(wlp1, width, height, true);
wallpaperManager.setBitmap(scaledwlp1);
Thread.sleep(10000);

wlp1 = BitmapFactory.decodeResource(getResources(), R.drawable.img8);
scaledwlp1 = Bitmap.createScaledBitmap(wlp1, width, height, true);
wallpaperManager.setBitmap(scaledwlp1);
Thread.sleep(10000);

wlp1 = BitmapFactory.decodeResource(getResources(), R.drawable.img9);
scaledwlp1 = Bitmap.createScaledBitmap(wlp1, width, height, true);
wallpaperManager.setBitmap(scaledwlp1);
Thread.sleep(10000);

wlp1 = BitmapFactory.decodeResource(getResources(), R.drawable.img0);
scaledwlp1 = Bitmap.createScaledBitmap(wlp1, width, height, true);
wallpaperManager.setBitmap(scaledwlp1);
Thread.sleep(10000);

wlp1 = BitmapFactory.decodeResource(getResources(), R.drawable.img11);
scaledwlp1 = Bitmap.createScaledBitmap(wlp1, width, height, true);
wallpaperManager.setBitmap(scaledwlp1);
Thread.sleep(10000);
    }
} catch (Exception e) {
    e.printStackTrace();
}
}
});
}
}

```

Manifest Code:

```

<?xml version="1.0" encoding="utf-8"?>
<manifest xmlns:android="http://schemas.android.com/apk/res/android"
    package="com.example.prog4">
    <uses-permission android:name="android.permission.SET_WALLPAPER"/>

    <application
        android:allowBackup="true"
        android:icon="@mipmap/ic_launcher"
        android:label="@string/app_name"

```

```

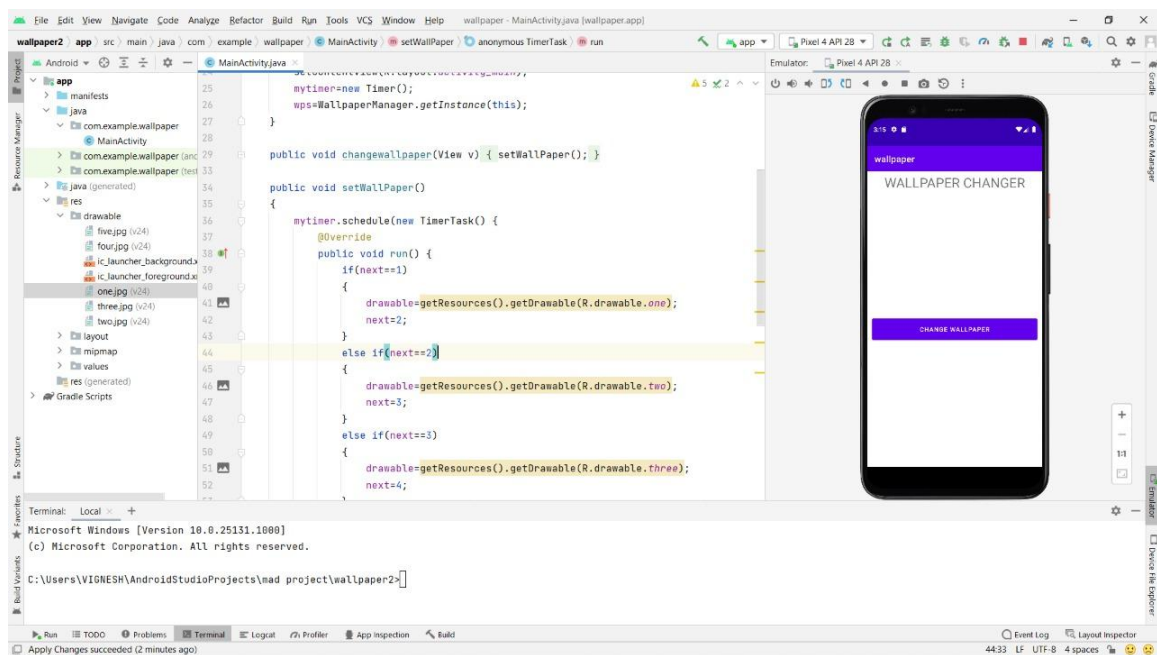
android:roundIcon="@mipmap/ic_launcher_round"
android:supportRtl="true"
android:theme="@style/Theme.Prog4">
<activity
    android:name=".MainActivity"
    android:exported="true">
    <intent-filter>
        <action android:name="android.intent.action.MAIN" />

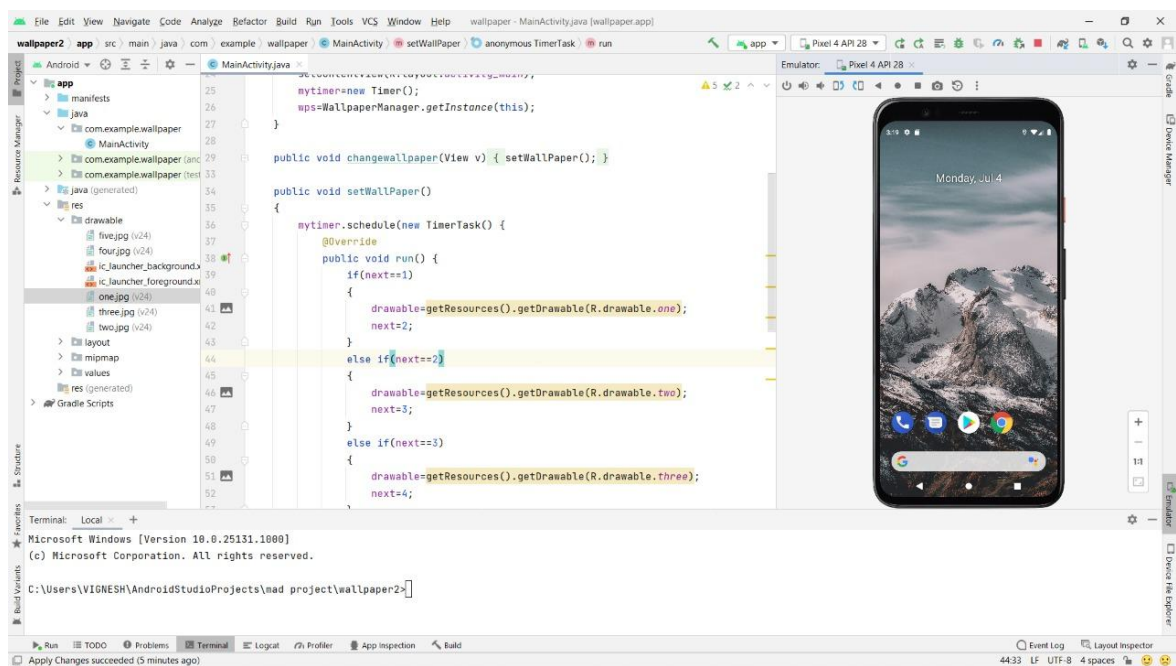
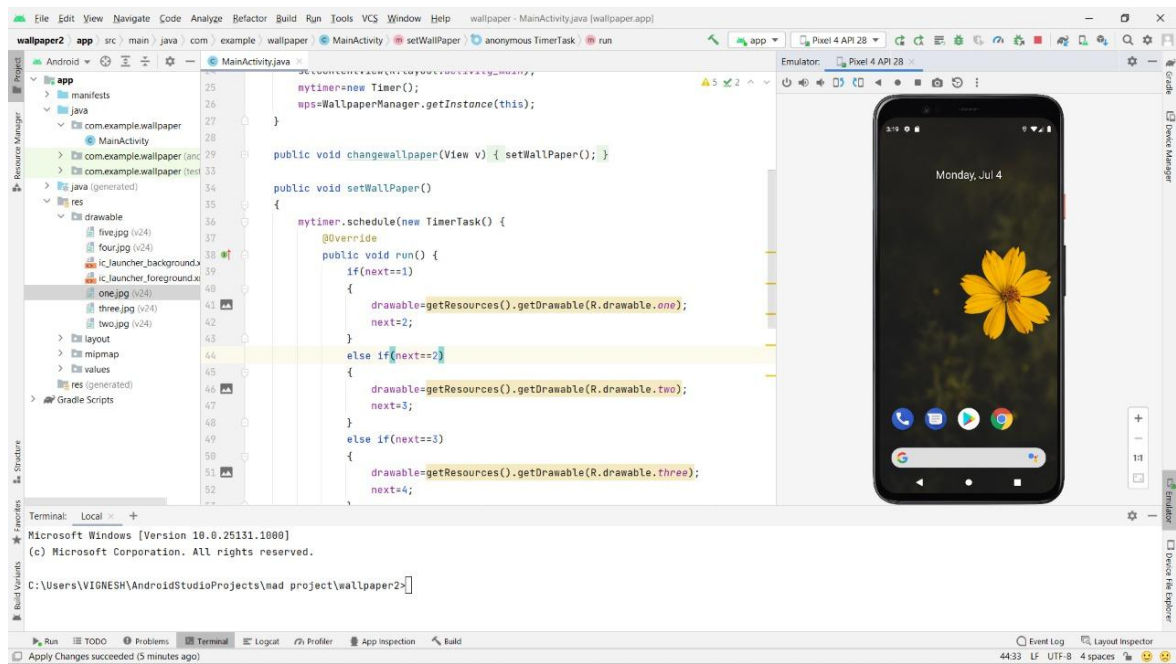
        <category android:name="android.intent.category.LAUNCHER" />
    </intent-filter>
</activity>
</application>

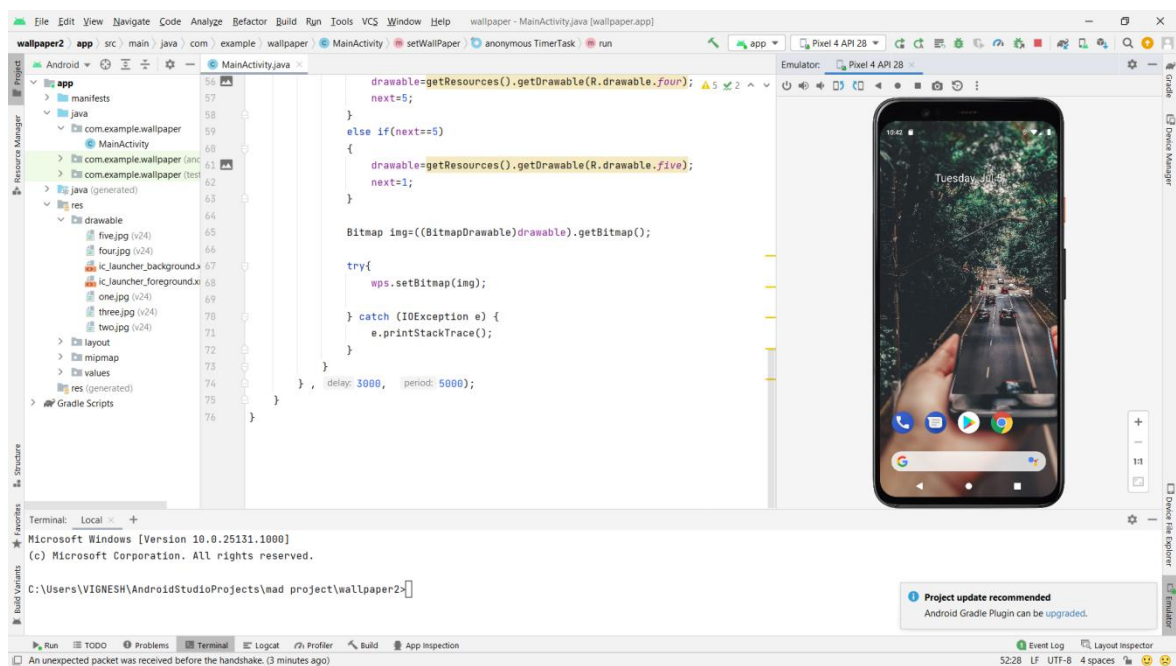
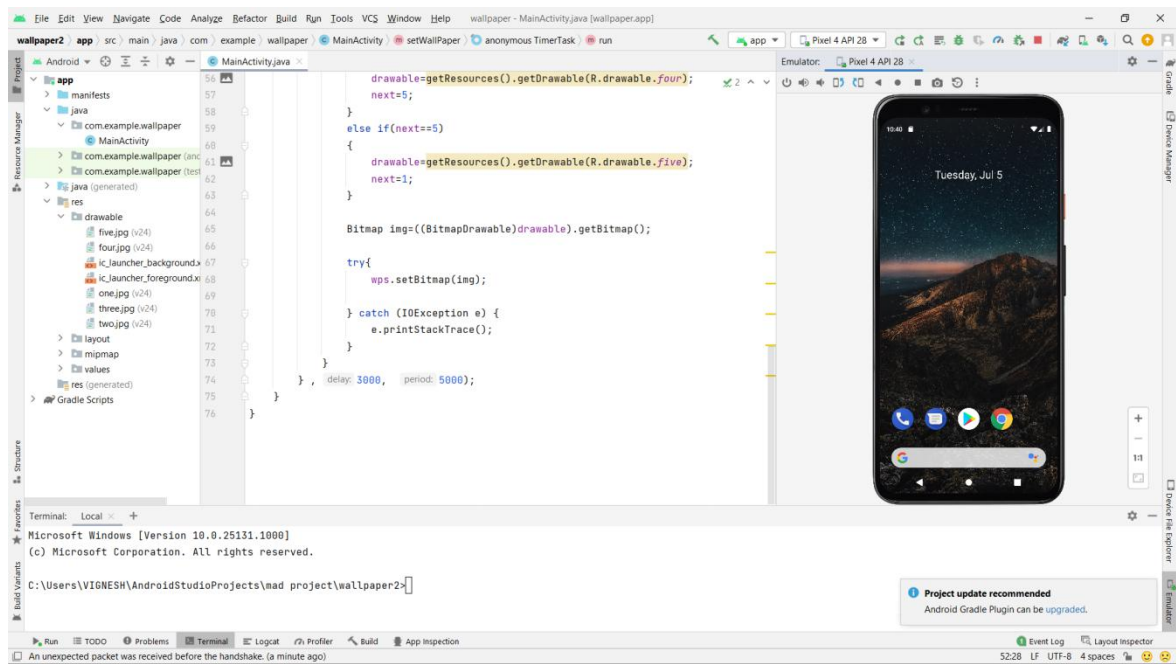
</manifest>

```

Sample Output:







PART – A

Experiment-No: 5

Write a program to create an activity with two buttons START and STOP. On pressing of the START button, the activity must start the counter by displaying the numbers from one and the counter must keep on counting until the STOP button is pressed. Display the counter value in a TextView control.



Components Used:

- TextView: To display texts
- Button: To start/stop/reset counter

XML Code:

```
<?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="#0E0E0E"
tools:context=".MainActivity">

<TextView
    android:id="@+id/textView"
    android:layout_width="match_parent"
    android:layout_height="132dp"
    android:text="0"
    android:textAlignment="center"
    android:textColor="#E91E63"
    android:textSize="60sp"
    app:layout_constraintBottom_toBottomOf="parent"
    app:layout_constraintEnd_toEndOf="parent"
    app:layout_constraintStart_toStartOf="parent"
```

```

        app:layout_constraintTop_toTopOf="parent" />

<Button
    android:id="@+id/start"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:onClick="onStrt"
    android:text="START"
    android:textColor="#E91E63"
    android:textSize="34sp"
    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="0.256" />

<Button
    android:id="@+id/stop"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:onClick="onStp"
    android:text="STOP"
    android:textColor="#E91E63"
    android:textSize="34sp"
    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="0.708" />

</androidx.constraintlayout.widget.ConstraintLayout>

```

Java Code:

```

package com.example.timer;

import androidx.appcompat.app.AppCompatActivity;

import android.os.Bundle;
import android.os.Handler;
import android.view.View;
import android.widget.Button;
import android.widget.TextView;

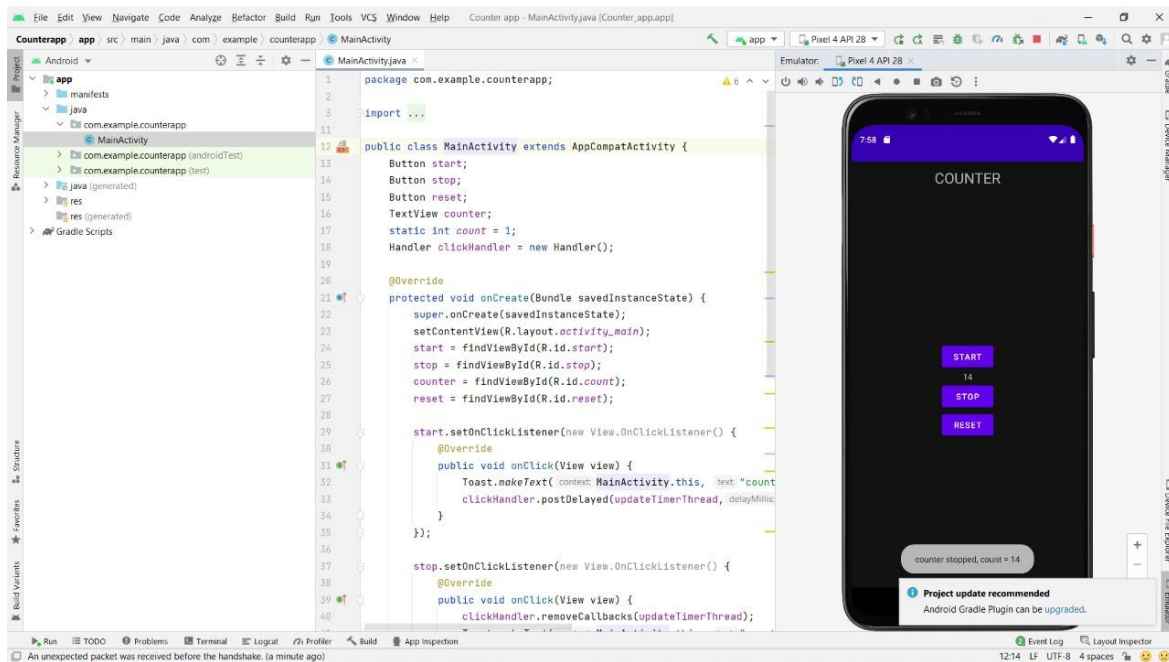
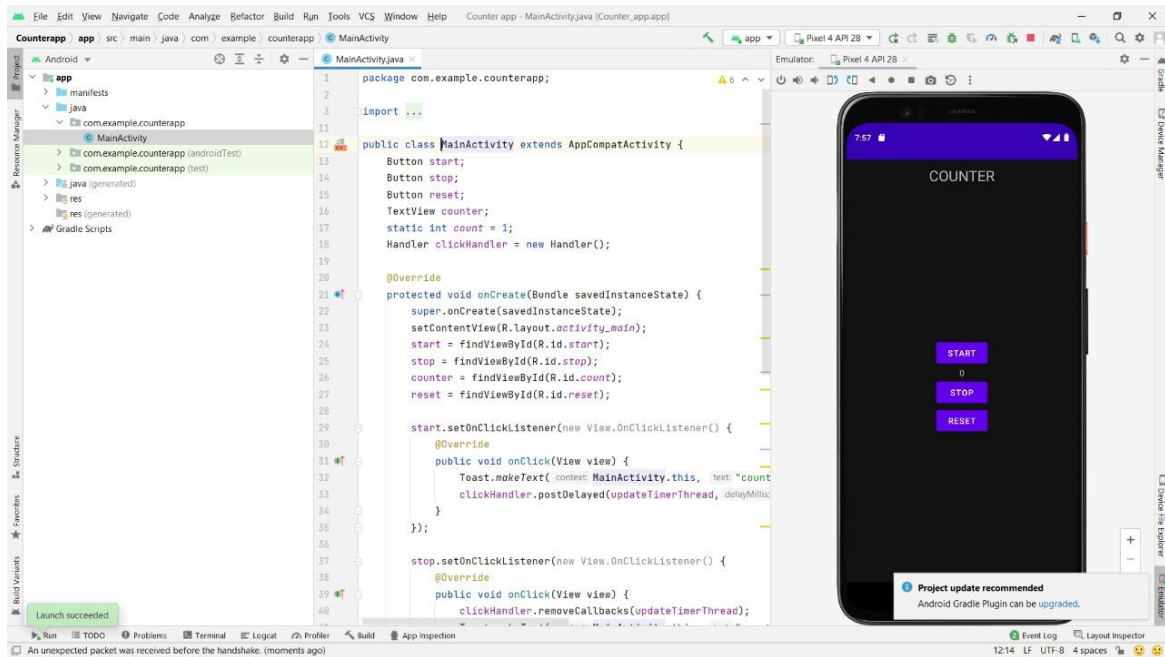
public class MainActivity extends AppCompatActivity {

```

```
static int count = 1;
Button start,stop;
TextView counter;
Handler customHandler=new Handler();

@Override
protected void onCreate(Bundle savedInstanceState) {
    super.onCreate(savedInstanceState);
    setContentView(R.layout.activity_main);
    start = findViewById(R.id.start);
    stop = findViewById(R.id.stop);
    counter = findViewById(R.id.textView);
}
private final Runnable updateTimerThread = new Runnable() {
    @Override
    public void run() {
        counter.setText(""+count);
        customHandler.postDelayed(updateTimerThread,1000);
        count++;
    }
};
public void onStart(View view)
{
    customHandler.postDelayed(updateTimerThread,0);
}

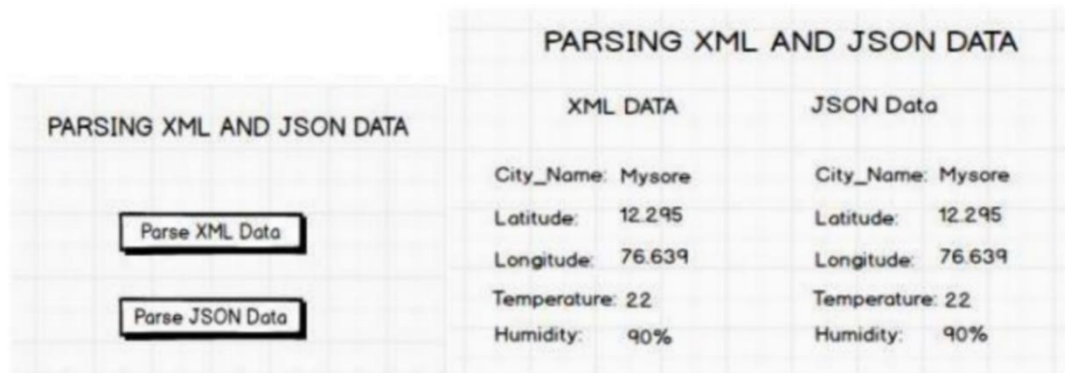
public void onStop(View view)
{
    customHandler.removeCallbacks(updateTimerThread);
    counter.setText(String.valueOf(count));
}
}
```

Sample Output:

PART – A

Experiment-No: 6

Create two files of XML and JSON type with values for City_Name, Latitude, Longitude, Temperature and Humidity. Develop an application to create an activity with two buttons to parse the XML and JSON files when clicked should display the data in their respective layouts side by side.



Components Used:

- Button: To display the data from the file on clicking
- TextView: To display data

Functions and Methods used:

- InputStream: This abstract class is the superclass of all classes representing an input stream of bytes.
- getAssets(): Returns an AssetManager instance for the application's package.
- DocumentBuilderFactory: Defines a factory API that enables applications to obtain a parser that produces DOM object trees from XML documents.
- DocumentBuilder: Defines the API to obtain DOM Document instances from an XML document.
- Document: The Document interface represents the entire HTML or XML document.
- stringBuilder: This class provides an API compatible with StringBuffer, but with no guarantee of synchronization.
- NodeList: The NodeList interface provides the abstraction of an ordered collection of nodes, without defining or constraining how this collection is implemented.
- getElementsByTagName: Returns a NodeList of all the Elements in document order with a given tag name and are contained in the document.
- Node: The Node interface is the primary datatype for the entire Document Object Model. It represents a single node in the document tree.

- Element: The Element interface represents an element in an HTML or XML document.
- read(): Reads some number of bytes from the input stream and stores them into the buffer array b.
- JSONArray: A dense indexed sequence of values.
- String(): Constructs a new String by decoding the specified array of bytes using the specified charset.
- getNodeTypes(): A code representing the type of the underlying object, as defined above.
- getString(): This method returns the value associated with the given key, or null if no mapping of the desired type exists for the given key or a null value is explicitly associated with the key.
- toString(): This method returns a string representation of the object.
- makeText(): This method makes a standard toast that just contains text.
- setText(): This method sets the text to be displayed.

Steps:

- Add Assets folder by following the given hierarchy:
- App->new->folder->Assets folder
- Inside the assets folder create new files of xml and json using the following hierarchy:
 - a. new->file->city.xml
 - b. new->file->city.json

city.xml:

```
<?xml version="1.0"?>
<records>
  <place>
    <name>Mysore</name>
    <lat>12.295</lat>
    <long>76.639</long>
    <temp>22</temp>
    <humidity>90 %</humidity>
  </place>
</records>
```

city.json:

```
[
  {
    "name": "HASSAN",
    "lat": "12.295",
    "long": "76.639",
    "temp": "22",
```



```

    "humidity": "92 %"
  }
]

```

XML Code:

```

<?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="#0E0E0E"
android:textAlignment="center"
tools:context=".MainActivity">

    <Button
        android:id="@+id/buttonXML"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_marginStart="60dp"
        android:layout_marginTop="164dp"
        android:text="Parse XML"
        app:layout_constraintEnd_toStartOf="@+id/buttonJSON"
        app:layout_constraintHorizontal_bias="0.043"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toTopOf="parent" />

    <Button
        android:id="@+id/buttonJSON"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_marginTop="164dp"
        android:layout_marginEnd="56dp"
        android:text="Parse JSON"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintTop_toTopOf="parent" />

    <TextView
        android:id="@+id/textView"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_marginStart="72dp"
        android:textColor="#E91E63"
        app:layout_constraintBottom_toBottomOf="parent"
        app:layout_constraintEnd_toStartOf="@+id/textView2"
        app:layout_constraintHorizontal_bias="0.026"

```

```
app:layout_constraintStart_toStartOf="parent"
app:layout_constraintTop_toBottomOf="@+id/buttonXML"
app:layout_constraintVertical_bias="0.194" />

<TextView
    android:id="@+id/textView2"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_marginEnd="72dp"
    android:textColor="#E91E63"
    app:layout_constraintBottom_toBottomOf="parent"
    app:layout_constraintEnd_toEndOf="parent"
    app:layout_constraintTop_toBottomOf="@+id/buttonJSON"
    app:layout_constraintVertical_bias="0.194" />

<TextView
    android:id="@+id/textView3"
    android:layout_width="389dp"
    android:layout_height="111dp"
    android:text="XML and JSON Parser"
    android:textAlignment="center"
    android:textColor="#E91E63"
    android:textSize="42sp"
    android:textStyle="bold"
    app:layout_constraintBottom_toBottomOf="parent"
    app:layout_constraintEnd_toEndOf="parent"
    app:layout_constraintHorizontal_bias="0.666"
    app:layout_constraintStart_toStartOf="parent"
    app:layout_constraintTop_toTopOf="parent"
    app:layout_constraintVertical_bias="0.01" />
</androidx.constraintlayout.widget.ConstraintLayout>
```

Java Code:

```
package com.example.xmlandjson;

import androidx.appcompat.app.AppCompatActivity;

import android.os.Bundle;
import android.view.View;
import android.widget.Button;
import android.widget.TextView;
import android.widget.Toast;

import org.json.JSONArray;
import org.json.JSONObject;
import org.w3c.dom.Document;
```

```

import org.w3c.dom.Element;
import org.w3c.dom.Node;
import org.w3c.dom.NodeList;

import java.io.InputStream;
import java.nio.charset.StandardCharsets;

import javax.xml.parsers.DocumentBuilder;
import javax.xml.parsers.DocumentBuilderFactory;

public class MainActivity extends AppCompatActivity {
    Button btnXML, btnJSON;
    TextView txtJSON, txtXML;
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);
        btnXML = findViewById(R.id.buttonXML);
        btnJSON = findViewById(R.id.buttonJSON);
        txtXML = findViewById(R.id.textView);
        txtJSON = findViewById(R.id.textView2);
        btnXML.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View view) {
                try {
                    InputStream inputStream = getAssets().open("city.xml");
                    DocumentBuilderFactory documentBuilderFactory =
DocumentBuilderFactory.newInstance();
                    DocumentBuilder documentBuilder =
documentBuilderFactory.newDocumentBuilder();
                    Document document = documentBuilder.parse(inputStream);
                    StringBuilder xmlData = new StringBuilder();
                    xmlData.append("Data from XML");
                    xmlData.append("\n-----");
                    NodeList nodeList = document.getElementsByTagName("place");
                    for (int i=0; i<nodeList.getLength(); i++) {
                        Node node = nodeList.item(i);
                        if (node.getNodeType() == Node.ELEMENT_NODE){
                            Element element = (Element)node;
                            xmlData.append("\nCity Name: ").append(getValue("name", element));
                            xmlData.append("\nLatitude: ").append(getValue("lat", element));
                            xmlData.append("\nLongitude: ").append(getValue("long", element));
                            xmlData.append("\nTemperature: ").append(getValue("temp", element));
                            xmlData.append("\nHumidity: ").append(getValue("humidity", element));
                            xmlData.append("\n-----");
                        }
                    }
                }
            }
        });
    }
}

```

```

        txtXML.setText(xmlData.toString());
    }catch (Exception ex){
        ex.printStackTrace();
        Toast.makeText(MainActivity.this, "Error Parsing XML Data",
Toast.LENGTH_SHORT).show();
    }
}
});

btnJSON.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View view) {
        try {

            InputStream is = getAssets().open("city.json");
            int size = is.available();
            byte[] buffer = new byte[size];
            is.read(buffer);
            String jsonString = new String(buffer, StandardCharsets.UTF_8);
            JSONArray jsonArray = new JSONArray(jsonString);
            StringBuilder jsonData = new StringBuilder();
            jsonData.append("Data from JSON");
            jsonData.append("\n-----");
            for (int i=0; i<jsonArray.length(); i++){
                JSONObject jsonObject = jsonArray.getJSONObject(i);

                jsonData.append("\nCity Name: ").append(jsonObject.getString("name"));
                jsonData.append("\nLatitude: ").append(jsonObject.getString("lat"));
                jsonData.append("\nLongitude: ").append(jsonObject.getString("long"));
                jsonData.append("\nTemperature: ").append(jsonObject.getString("temp"));
                jsonData.append("\nHumidity: ").append(jsonObject.getString("humidity"));
                jsonData.append("\n-----");
            }
            txtJSON.setText(jsonData.toString());
        }
        catch (Exception ex){
            ex.printStackTrace();
            Toast.makeText(MainActivity.this, "Error Parsing JSON Data",
Toast.LENGTH_SHORT).show();
        }
    }
});

private String getValue(String tag, Element element){
    return
    element.getElementsByTagName(tag).item(0).getChildNodes().item(0).getNodeValue();
}

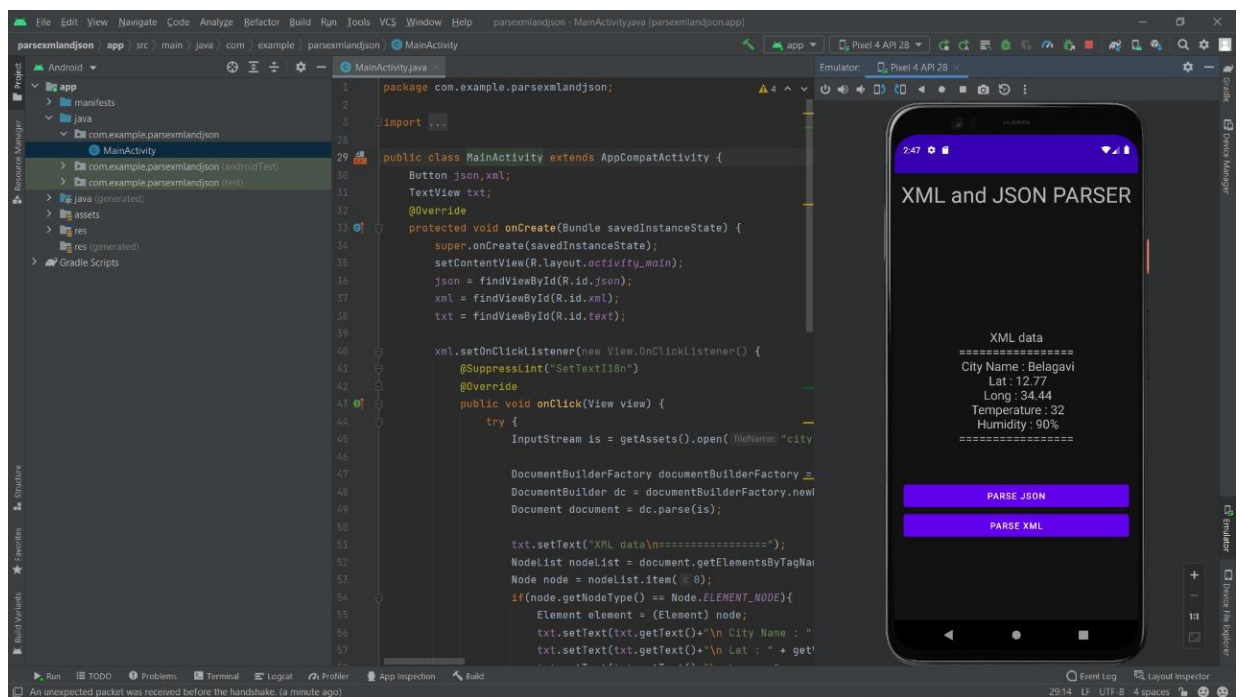
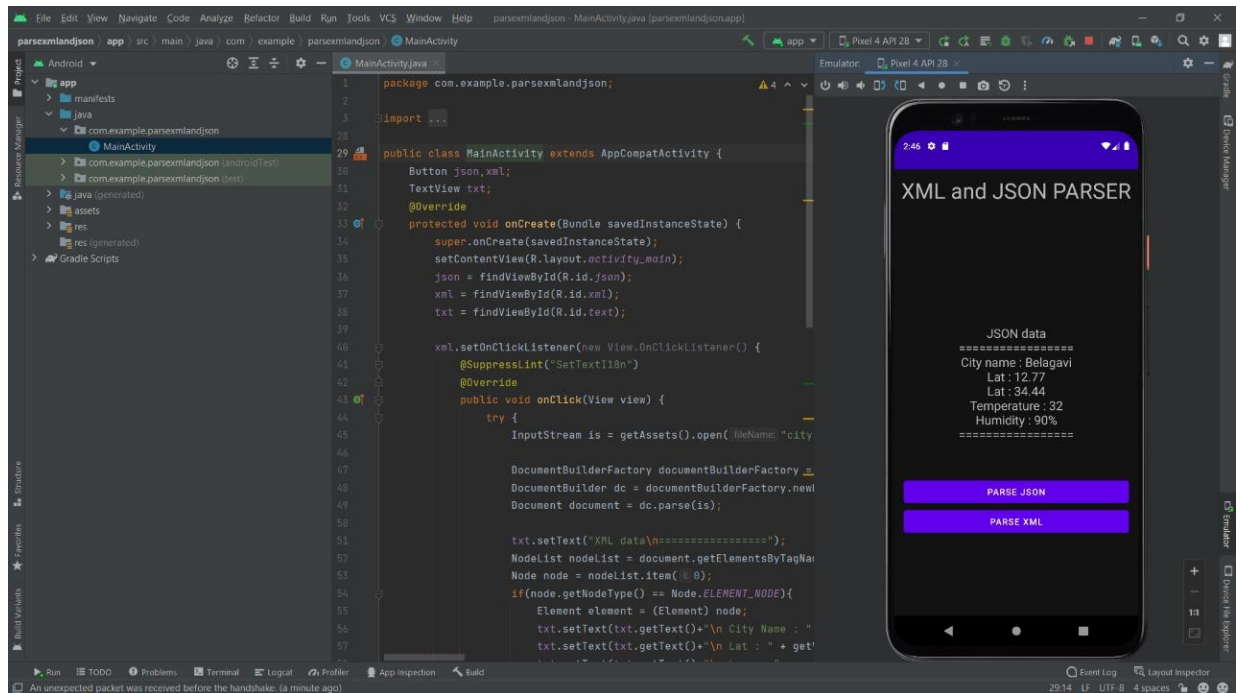
```

```

    }
}

```

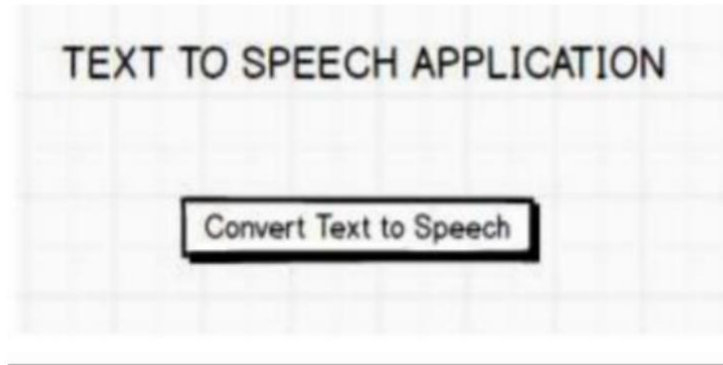
Sample Output:



PART – A

Experiment-No: 7

Develop a simple application with one EditText so that user can write some text in it. Create a button called “Convert Text To Speech” that converts the user input text into voice.



Components Used:

- EditText: To input text
- Button: To convert text to speech

Functions and Methods Used:

- onInit(): method called to signal the completion of TextToSpeech engine initialization
- TextToSpeech: class used to synthesize speech from text for immediate playback.
- speak(): method is used to convert the input text to voice. The parameters passed are input text, queuing strategy and speech parameters
- Locale(): A Locale object represents a specific geographical, political, or cultural region.

XML Code:

```
<?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:background="#87C4DF"
    android:orientation="vertical"
    tools:context=".MainActivity">

    <TextView
        android:id="@+id/textView"
```

```

        android:layout_width="match_parent"
        android:layout_height="101dp"
        android:layout_marginTop="200dp"
        android:text="TEXT TO SPEECH CONVERTER"
        android:textAlignment="center"
        android:textColor="#E91E63"
        android:textSize="34sp"
        android:textStyle="bold"
        app:autoSizeTextType="uniform" />

<EditText
    android:id="@+id/edittext"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_marginTop="40dp"
    android:backgroundTint="#3F51B5"
    android:ems="10"
    android:hint="Enter the Text here..."
    android:inputType="textPersonName"
    android:text="Enter the Text here..."
    android:textAlignment="center" />

<Button
    android:id="@+id/button"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_marginLeft="30dp"
    android:layout_marginRight="30dp"
    android:padding="16dp"
    android:text="Convert"
    android:textSize="20sp" />
</LinearLayout>

```

Java Code:

```

package com.example.prog7;

import androidx.appcompat.app.AppCompatActivity;
import android.speech.tts.TextToSpeech;
import java.util.Locale;

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

public class MainActivity extends AppCompatActivity {

```



```

EditText Text;
Button btn;
TextToSpeech tts;

```

@Override

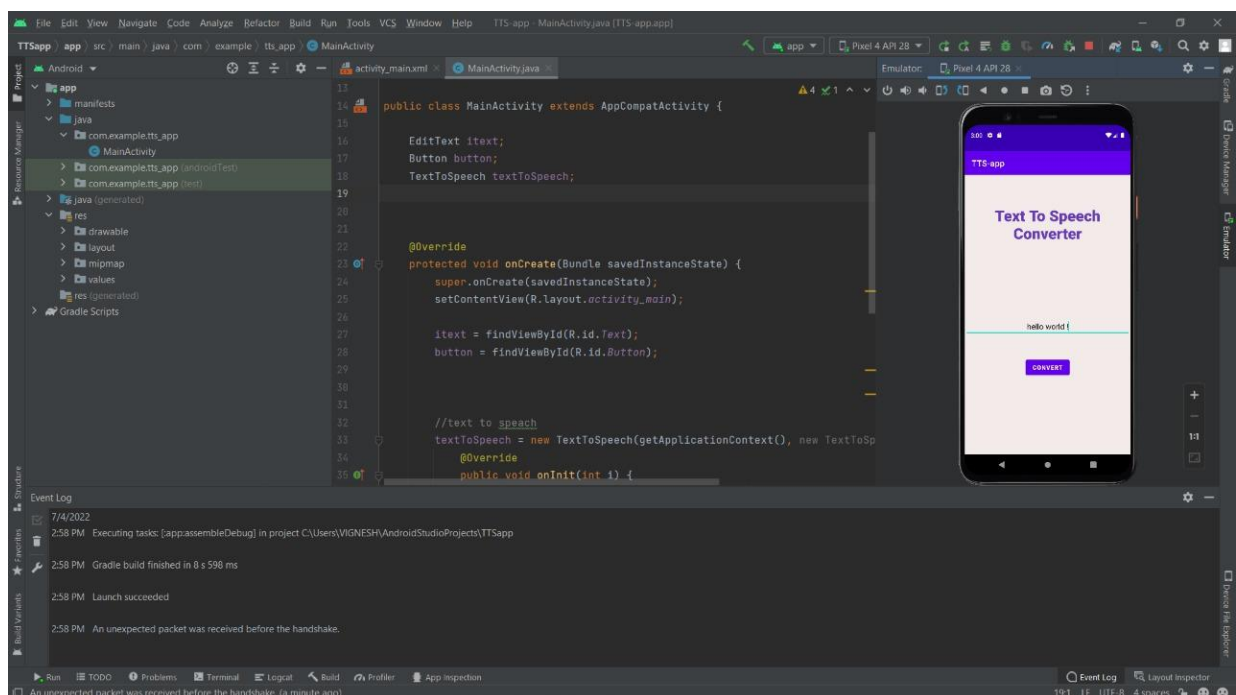
```

protected void onCreate(Bundle savedInstanceState) {
    super.onCreate(savedInstanceState);
    setContentView(R.layout.activity_main);
    Text = findViewById(R.id.edittext);
    btn = findViewById(R.id.button);

    tts = new TextToSpeech(getApplicationContext(), new TextToSpeech.OnInitListener()
{
    @Override
    public void onInit(int i) {
        if(i!=TextToSpeech.ERROR){
            tts.setLanguage(Locale.US);
        }
    }
});
btn.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View view) {
        tts.speak(Text.getText().toString(),TextToSpeech.QUEUE_FLUSH,null);
    }
});
}}

```

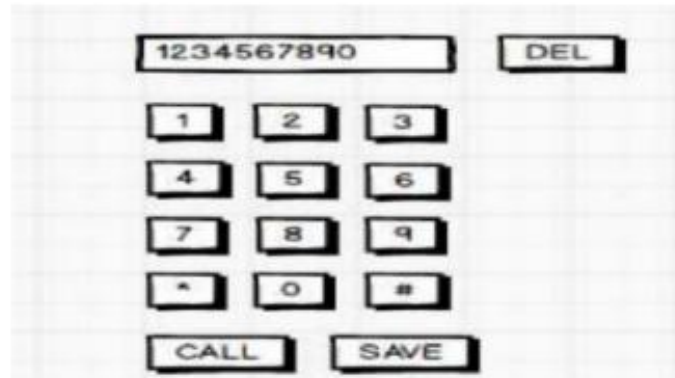
Sample Output:



PART – A

Experiment-No: 8

Create an activity like a phone dialer with CALL and SAVE buttons. On pressing the CALL button, it must call the phone number and on pressing the SAVE button it must save the number to the phone contacts.



Components Used:

- EditText: To enter Phone Number
- ImageView : To display clickable call and save icon

Functions and Methods Used:

- getText(): This method returns the text that TextView is displaying.
- toString(): This method returns a string representation of the object.
- setImageResource(): This method is used to set Image from the path specified.
- ContextCompat: This class is used when you would like to retrieve resources.
- ACTION_DIAL - Opens the dialer or phone app.
- ACTION_CALL - Places a phone call (requires the CALL_PHONE permission)
- ACTION_INSERT- To insert a new contact, specify Contacts.CONTENT_TYPE as the type, and include any known contact information in extras specified by constants in ContactsContract.Intents.Insert.

XML Code:

```
<?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="#03A9F4"
tools:context=".MainActivity">
```

```
<TextView
    android:id="@+id/textView"
    android:layout_width="358dp"
    android:layout_height="104dp"
    android:background="#8EECF8"
    android:fontFamily="sans-serif-black"
    android:text="CALL AND SAVE ACTIVITY"
    android:textAlignment="center"
    android:textColor="#03A9F4"
    android:textSize="40sp"
    android:textStyle="italic"
    app:layout_constraintBottom_toBottomOf="parent"
    app:layout_constraintEnd_toEndOf="parent"
    app:layout_constraintHorizontal_bias="0.509"
    app:layout_constraintStart_toStartOf="parent"
    app:layout_constraintTop_toTopOf="parent"
    app:layout_constraintVertical_bias="0.066" />
```

```
<EditText
    android:id="@+id/editTextPhone"
    android:layout_width="291dp"
    android:layout_height="78dp"
    android:ems="10"
    android:inputType="phone"
    app:layout_constraintBottom_toBottomOf="parent"
    app:layout_constraintEnd_toEndOf="parent"
    app:layout_constraintHorizontal_bias="0.133"
    app:layout_constraintStart_toStartOf="parent"
    app:layout_constraintTop_toTopOf="parent"
    app:layout_constraintVertical_bias="0.33" />
```

```
<Button
    android:id="@+id/bdel"
    android:layout_width="91dp"
    android:layout_height="87dp"
    android:text="DEL"
    android:textSize="24sp"
    app:layout_constraintBottom_toBottomOf="parent"
    app:layout_constraintEnd_toEndOf="parent"
    app:layout_constraintHorizontal_bias="0.981"
    app:layout_constraintStart_toStartOf="parent"
    app:layout_constraintTop_toTopOf="parent"
    app:layout_constraintVertical_bias="0.332" />
```

```
<Button
    android:id="@+id/bcall"
```

```

        android:layout_width="150dp"
        android:layout_height="107dp"
        android:text="CALL"
        android:textSize="24sp"
        app:layout_constraintBottom_toBottomOf="parent"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintHorizontal_bias="0.098"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toTopOf="parent"
        app:layout_constraintVertical_bias="0.707" />

<Button
    android:id="@+id/bsave"
    android:layout_width="154dp"
    android:layout_height="105dp"
    android:text="SAVE"
    android:textSize="24sp"
    app:layout_constraintBottom_toBottomOf="parent"
    app:layout_constraintEnd_toEndOf="parent"
    app:layout_constraintHorizontal_bias="0.898"
    app:layout_constraintStart_toStartOf="parent"
    app:layout_constraintTop_toTopOf="parent"
    app:layout_constraintVertical_bias="0.709" />
</androidx.constraintlayout.widget.ConstraintLayout>

```

Java Code:

```

package com.example.callandsave;

import androidx.appcompat.app.AppCompatActivity;

import android.content.Intent;
import android.net.Uri;
import android.os.Bundle;
import android.provider.ContactsContract;
import android.view.View;
import android.widget.Button;
import android.widget.EditText;

public class MainActivity extends AppCompatActivity {

    Button btnDel,btnCall,btnSave;
    EditText number;
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);
    }
}

```

```
btnDel = findViewById(R.id.bdel);
btnCall = findViewById(R.id.bcall);
btnSave = findViewById(R.id.bsave);
number = findViewById(R.id.editTextPhone);

btnDel.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View view) {
        number.setText("");
    }
});

btnSave.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View view) {
        String phoneNumber = number.getText().toString();
        Intent intent = new Intent(Intent.ACTION_INSERT);
        intent.setType(ContactsContract.Contacts.CONTENT_TYPE);
        intent.putExtra(ContactsContract.Intents.Insert.PHONE,phoneNumber);
        startActivity(intent);
    }
});

btnCall.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View view) {
        String phoneNumber = number.getText().toString();
        Intent intent = new Intent(Intent.ACTION_DIAL);
        intent.setData(Uri.parse("tel:"+phoneNumber));
        startActivity(intent);
    }
});
}
```

Sample Output: