Experiment No : 3

Aim : WAA to design a Form using GUI components

Theory :

- A login form is created using the <EditText /> XML element.
- The processing of data is designed in the Java file of the project.
- Clearning of data on pressing the 'Reset' button is done from Java and displaying error message using the built-in error method 'setError()' is done using the Java.android package.

- Forms in XML file as Java :

1) Logout
- UI component are defined using XML files in logout directory.
- The logout XML file defines the structure and apperance of the user interface.
- Here the defined EditText fields for name, email, password along with buttons are present in XML logout.

2) Java Code
- Main Activity class extends AppcompatActivity, which is the base class for activities in Android.

- Initialize Edittext fields and buttons using findViewById() method, which finds the view by the specified Id defined in the XML logout.
- Set click defined listners for proceed and reset buttons using setOnclickListener() method.
- Inside the click listener, we handle the button clicks :-
- For the proceed button, we retreive text from EditText field, process the data as needed, and display it in a toast message using Toast.
- For reset button, we clear all the EditText fields.

3) Event Handling
⇒ User interaction such as button clicks triggers event that are handled by event listeners.
- SetOnClickListeners() method to set the event listeners for button clicks.
- Inside the event listeners, we define actions to be performed when the corresponding button is clicked.

- Conclusion :
The experiment illustrates how to create a simple login form in android studio using event handlers.

Code:

(MainActivity.java)

```java
package com.example.mcc_exp3;

import android.os.Bundle;

import android.view.View;

import android.widget.Button;

import android.widget.EditText;

import android.widget.Toast;

import androidx.appcompat.app.AppCompatActivity;


public class MainActivity extends AppCompatActivity {

    private EditText editTextFirstName, editTextLastName,
editTextEmail, editTextPassword;


    @Override

    protected void onCreate(Bundle savedInstanceState) {

        super.onCreate(savedInstanceState);

        setContentView(R.layout.activity_main);


        editTextFirstName = findViewById(R.id.editTextFirstName);

        editTextLastName = findViewById(R.id.editTextLastName);

        editTextEmail = findViewById(R.id.editTextEmail);

        editTextPassword = findViewById(R.id.editTextPassword);


        Button buttonProceed = findViewById(R.id.buttonProceed);
```

```java
        Button buttonReset = findViewById(R.id.buttonReset);


        buttonProceed.setOnClickListener(new View.OnClickListener() {

            @Override

            public void onClick(View v) {

                String firstName =
editTextFirstName.getText().toString();

                String lastName =
editTextLastName.getText().toString();

                String email = editTextEmail.getText().toString();

                String password =
editTextPassword.getText().toString();


                String message = "First Name: " + firstName + "nLast
Name: " + lastName +

                        "nEmail: " + email + "nPassword: " + password;


                Toast.makeText(MainActivity.this, message,
Toast.LENGTH_SHORT).show();

            }

        });


        buttonReset.setOnClickListener(new View.OnClickListener() {

            @Override

            public void onClick(View v) {
```

```java
            editTextFirstName.setText("");

            editTextLastName.setText("");

            editTextEmail.setText("");

            editTextPassword.setText("");

            }

        });

    }

}
```

(activity_main.xml)

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

    <EditText
        android:id="@+id/editTextFirstName"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:hint="First Name"/>

    <EditText
        android:id="@+id/editTextLastName"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_below="@id/editTextFirstName"
        android:layout_marginTop="16dp"
        android:hint="Last Name"/>

    <EditText
        android:id="@+id/editTextEmail"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
```

```xml
        android:layout_below="@id/editTextLastName"
        android:layout_marginTop="16dp"
        android:inputType="textEmailAddress"
        android:hint="Email"/>

    <EditText
        android:id="@+id/editTextPassword"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_below="@id/editTextEmail"
        android:layout_marginTop="16dp"
        android:inputType="textPassword"
        android:hint="Password"/>

    <Button
        android:id="@+id/buttonProceed"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_below="@id/editTextPassword"
        android:layout_marginTop="16dp"
        android:text="Proceed"/>

    <Button
        android:id="@+id/buttonReset"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_below="@id/buttonProceed"
        android:layout_marginTop="16dp"
        android:text="Reset"/>
</RelativeLayout>
```

Output: