Activity main.xml

<?xml version="1.0" encoding="utf-8"?>

<RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"

xmlns:app="http://schemas.android.com/apk/res-auto"

xmlns:tools="http://schemas.android.com/tools"

android:layout\_width="match\_parent"

android:layout\_height="match\_parent"

tools:context=".MainActivity">

<EditText

android:id="@+id/username"

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:hint="username"

android:minHeight="48dp" />

<EditText

android:id="@+id/password"

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:layout\_below="@id/username"

android:hint="Password"

android:minHeight="48dp" />

<EditText

android:id="@+id/email"

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:layout\_below="@id/password"

android:hint="email"

android:minHeight="48dp" />

<EditText

android:id="@+id/phone"

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:layout\_below="@id/email"

android:hint="phone number"

android:minHeight="48dp" />

<Button

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:text="submit"

android:id="@+id/submit"

android:layout\_below="@+id/phone"

/>

</RelativeLayout>

Mainactivity.java

package com.example.reg\_form\_relative;

import androidx.appcompat.app.AppCompatActivity;

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 EditText t2, t3, t4, t5;

private Button submit;

private static final Pattern password\_pattern =

Pattern.compile("^" +

"(?=.\*[@#$%^&+=])" +

"(?=\\s+$)" +

".{4,}" +

"$");

@Override

protected void onCreate(Bundle savedInstanceState) {

super.onCreate(savedInstanceState);

setContentView(R.layout.activity\_main);

t2 = findViewById(R.id.username);

t3 = findViewById(R.id.password);

t4 = findViewById(R.id.email);

t5 = findViewById(R.id.phone);

submit = findViewById(R.id.submit);

submit.setOnClickListener(new View.OnClickListener() {

@Override

public void onClick(View view) {

String s2 = t2.getText().toString();

String s3 = t3.getText().toString();

String s4 = t4.getText().toString();

String s5 = t5.getText().toString();

if (s2.isEmpty()) {

showToast("Username cannot be empty");

return;

}

if (s3.isEmpty()) {

showToast("Password cannot be empty");

return;

} else if (!password\_pattern.matcher(s3).matches()) {

showToast("Password must contain at least 4 characters and one of the special characters: @#$%^&+=");

return;

}

if (s4.isEmpty()) {

showToast("Email cannot be empty");

return;

} else if (!isValidEmail(s4)) {

showToast("Invalid email address");

return;

}

if (s5.isEmpty()) {

showToast("Phone number cannot be empty");

return;

} else if (s5.length() != 10) {

showToast("Phone number must have exactly 10 digits");

return;

}

// All validations passed, you can proceed with further actions here

}

});

}

private void showToast(String message) {

Toast.makeText(MainActivity.this, message, Toast.LENGTH\_SHORT).show();

}

private boolean isValidEmail(String email) {

// You can use a regular expression to validate email format

// This is a simple example, and you may need a more comprehensive one

String emailPattern = "[a-zA-Z0-9.\_-]+@[a-z]+\\.+[a-z]+";

return email.matches(emailPattern);

}

}

