

Mobile Application Development Lab

Course Outcome 1

Submitted By: Jomin K Mathew

20MCA321

1. Design a Login Form with username and password using LinearLayout and toast valid credentials

Activity_main.xml:

```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:orientation="vertical"
    tools:context=".MainActivity" >
    <TextView
        android:id="@+id/textView"
        android:layout_width="match_parent"
        android:layout_height="72dp"
        android:fontFamily="@font/alfa_slab_one"
        android:text="Login Page"
        android:textColor="#733B3B"
        android:textColorHighlight="#9F9494"
        android:textColorHint="#59A677"
        android:textColorLink="#5A4040"
        android:textSize="34sp" />
    <EditText
        android:id="@+id/name"
        android:layout_width="match_parent"
        android:layout_height="124dp"
        android:ems="10"
        android:hint="Name"
        android:inputType="textPersonName"
        android:textSize="20sp" />
    <EditText
        android:id="@+id/password"
        android:layout_width="match_parent"
        android:layout_height="131dp"
        android:ems="10"
        android:hint="Password"
        android:inputType="textPassword"
        android:textSize="20sp" />
    <Button
        android:id="@+id/login"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:background="#2F7A6E"
        android:backgroundTint="#651E1E"
```

```
android:checkable="false"
android:fontFamily="@font/alfa_slab_one"
android:foregroundTint="#8A2828"
android:text="Login"
android:textSize="16sp"
app:iconTint="#7E3131"
app:rippleColor="#9F2525" />
</LinearLayout>
```

MainActivity.java:

```
package com.example.login;
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;
public class MainActivity extends AppCompatActivity {
    EditText t1,t2;
    Button b1;
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);
        t1=(EditText)findViewById(R.id.name);
        t2=(EditText)findViewById(R.id.password);
        b1=(Button)findViewById(R.id.login);
    }
    public void onClick(View view){
        String a =t1.getText().toString();
        String b =t2.getText().toString();
        if((a.equals("admin"))&&(b.equals("admin")))
        {
            Toast.makeText(this,"Succesfully Logined",Toast.LENGTH_LONG).show();
        }
        else{
            Toast.makeText(this,"Check username & password",Toast.LENGTH_LONG).show();
        }
    }
}
```

Output:



2. Write a program that demonstrates Activity Lifecycle.

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="Hello World!"
app:layout_constraintBottom_toBottomOf="parent"
app:layout_constraintLeft_toLeftOf="parent"
app:layout_constraintRight_toRightOf="parent"
app:layout_constraintTop_toTopOf="parent" />
</androidx.constraintlayout.widget.ConstraintLayout>
```

MainActivity.java:

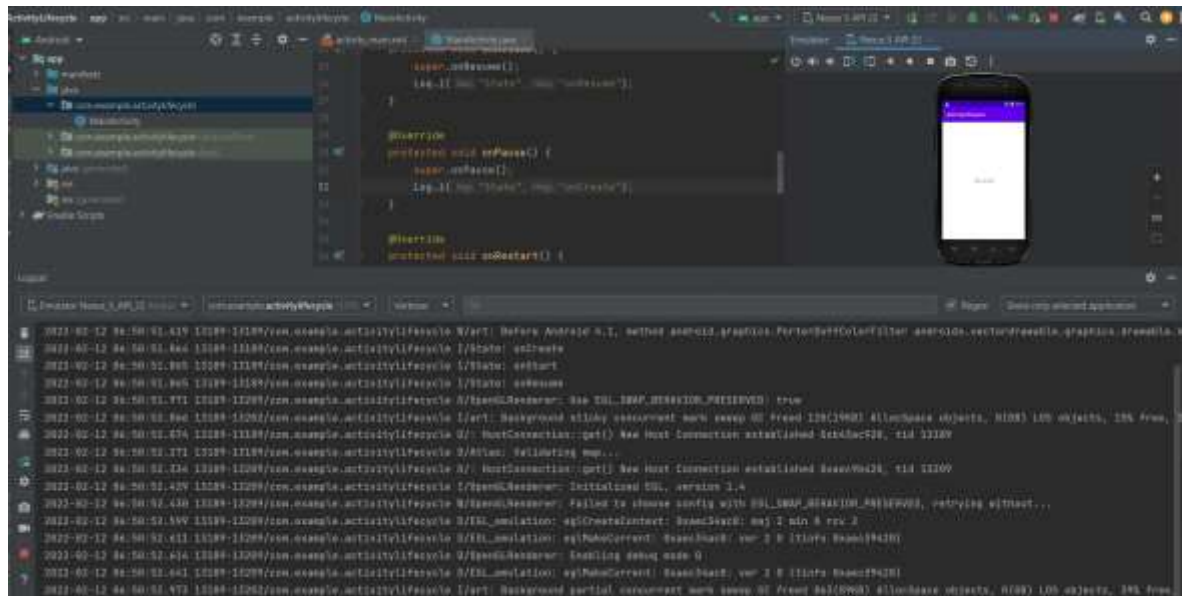
```
package com.example.activitylifecycle;
import androidx.appcompat.app.AppCompatActivity;
import android.os.Bundle;
import android.util.Log;
public class MainActivity extends AppCompatActivity {
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);
        Log.i("State", "onCreate");
    }
    @Override
    protected void onStart() {
        super.onStart();
        Log.i("State", "onStart");
    }
    @Override
    protected void onResume() {
        super.onResume();
        Log.i("State", "onResume");
    }
}
```

```

@Override
protected void onPause() {
    super.onPause();
    Log.i("State","onCreate");
}
@Override
protected void onRestart() {
    super.onRestart();
    Log.i("State","onRestart");
}
@Override
protected void onStop() {
    super.onStop();
    Log.i("State","onStop");
}
@Override
protected void onDestroy() {
    super.onDestroy();
    Log.i("State","onDestroy");
}
}

```

Output:



3. Implementing basic arithmetic operations of a simple calculator

Activity_main.xml:

```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:orientation="vertical"
    tools:context=".MainActivity">
    <TextView
        android:id="@+id/textView4"
        android:layout_width="match_parent"
        android:layout_height="53dp"
        android:fontFamily="@font/archivo_black"
        android:text="Simple Calculator"
        android:textColor="#407B95"
        android:textSize="20sp" />
    <EditText
        android:id="@+id/num1"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:ems="10"
        android:hint="Enter the 1st number"
        android:inputType="number" />
    <EditText
        android:id="@+id/num2"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:ems="10"
        android:hint="Enter the 2nd number"
        android:inputType="number" />
    <EditText
        android:id="@+id/result"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:ems="10"
        android:hint="Result"
        android:inputType="number" />
    <Button
        android:id="@+id/add"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:onClick="add"
```

```

        android:text="ADD" />
    <Button
        android:id="@+id/sub"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:onClick="sub"
        android:text="SUB" />
    <Button
        android:id="@+id/mult"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:onClick="mult"
        android:text="Mult" />
    <Button
        android:id="@+id/div"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:onClick="div"
        android:text="Div" />
</LinearLayout>

```

MainActivity.java:

```

package com.example.calculator;
import androidx.appcompat.app.AppCompatActivity;
import android.os.Bundle;
import android.view.View;
import android.widget.EditText;
import android.widget.Button;
public class MainActivity extends AppCompatActivity {
    EditText e1,e2,e3;
    Button b1,b2,b3,b4;
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);
        e1=(EditText)findViewById(R.id.num1);
        e2=(EditText)findViewById(R.id.num2);
        e3=(EditText)findViewById(R.id.result);
        b1=(Button)findViewById(R.id.add);
        b2=(Button)findViewById(R.id.sub);
        b3=(Button)findViewById(R.id.mult);
        b4=(Button)findViewById(R.id.div);
    }
    public void add(View view)
    {

```

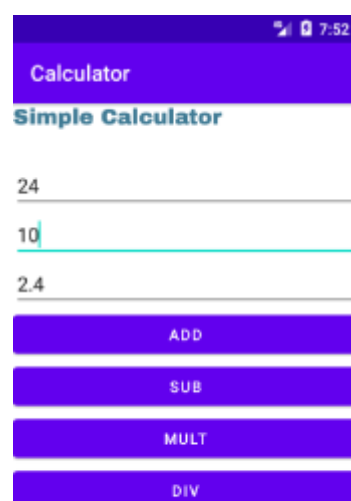


```

int a=Integer.parseInt(e1.getText().toString());
int b=Integer.parseInt(e2.getText().toString());
String c=String.valueOf(a+b);
e3.setText(c);
}
public void sub(View view)
{
int a=Integer.parseInt(e1.getText().toString());
int b=Integer.parseInt(e2.getText().toString());
String c=String.valueOf(a-b);
e3.setText(c);
}
public void mult(View view)
{
int a=Integer.parseInt(e1.getText().toString());
int b=Integer.parseInt(e2.getText().toString());
String c=String.valueOf(a*b);
e3.setText(c);
}
public void div(View view)
{
float a=Integer.parseInt(e1.getText().toString());
float b=Integer.parseInt(e2.getText().toString());
String c=String.valueOf(a/b);
e3.setText(c);
}
}

```

Output:



4. Implement validations on various UI controls

Activity_main.xml:

```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:baselineAligned="false"
    android:orientation="vertical"
    android:textAlignment="center"
    tools:context=".MainActivity">
    <TextView
        android:id="@+id/textView3"
        android:layout_width="match_parent"
        android:layout_height="48dp"
        android:fontFamily="@font/baloo_chettan"
        android:text="Validation Form"
        android:textColor="#3D2361"
        android:textSize="24sp" />
    <EditText
        android:id="@+id/name"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:ems="10"
        android:hint="Enter the Name"
        android:inputType="textPersonName" />
    <EditText
        android:id="@+id/email"
        android:layout_width="match_parent"
        android:layout_height="60dp"
        android:ems="10"
        android:hint="Enter the Email"
        android:inputType="textEmailAddress" />
    <EditText
        android:id="@+id/number"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:ems="10"
        android:hint="Enter Phone number"
        android:inputType="phone" />
    <EditText
        android:id="@+id/address"
        android:layout_width="match_parent"
```

```

android:layout_height="wrap_content"
android:ems="10"
android:hint="Enter the Address"
android:inputType="textPersonName" />
<EditText
android:id="@+id/password"
android:layout_width="match_parent"
android:layout_height="wrap_content"
android:ems="10"
android:hint="Enter the password"
android:inputType="textPassword" />
<Button
android:id="@+id/button"
android:layout_width="match_parent"
android:layout_height="wrap_content"
android:onClick="onSubmit"
android:text="Submit" />
</LinearLayout>

```

MainActivity.java:

```

package com.example.registration;
import androidx.appcompat.app.AppCompatActivity;
import android.os.Bundle;
import android.view.View;
import android.widget.EditText;
import android.widget.Toast;
public class MainActivity extends AppCompatActivity {
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);
    }
    public Boolean validateName(){
        EditText ed1 = (EditText) findViewById(R.id.name);
        String name = ed1.getText().toString();
        if (name.isEmpty()){
            ed1.setError("Name cannot be empty");
            return false;
        }
        else {
            return true;
        }
    }
    public Boolean validateAddress(){
        EditText ed5 = (EditText) findViewById(R.id.address);
    }
}

```

```

String address = ed5.getText().toString();
if (address.isEmpty()){
ed5.setError("Address cannot be empty");
return false;
}
else {
return true;
}
}

public Boolean validateEmail(){
EditText ed2 = (EditText) findViewById(R.id.email);
String email = ed2.getText().toString();
String emailPattern = "[a-zA-Z0-9._-]+@[a-z]+\\.+[a-z]+";
if (email.isEmpty()){
ed2.setError("Email cannot be empty");
return false;
}
else if (!email.matches(emailPattern)){
ed2.setError("Invalid email address");
return false;
}
else{
return true;
}
}

public Boolean validateNumber() {
EditText ed3 = (EditText) findViewById(R.id.number);
String num = ed3.getText().toString();
String numval = ".{10,}";
if (num.isEmpty()) {
ed3.setError("Phone number cannot be empty");
return false;
} else if (!num.matches(numval)) {
ed3.setError("Check it exist 10 numbers");
return false;
} else {
return true;
}
}

public Boolean validatePassword(){
EditText ed4 = (EditText) findViewById(R.id.password);
String pswd = ed4.getText().toString();
String passwordVal = "^" +
//"(?=.*[0-9])" +      //at least 1 digit
//"(?=.*[a-z])" +      //at least 1 lower case letter
//"(?=.*[A-Z])" +      //at least 1 upper case letter

```

```

"(?=.*[a-zA-Z])" + //any letter
"(?=.*[@#$%^&+=])" + //at least 1 special character
"(?=\s+$)" + //no white spaces
".{4,}" + //at least 4 characters
"$";
if (pswd.isEmpty()){
ed4.setError("Password cannot be empty");
return false;
}
else if (!pswd.matches(passwordVal)) {
ed4.setError("Password is too weak");
return false;
}
else{
return true;
}
}
}
public void onSubmit(View view){
if(validateName() && validateEmail() && validateNumber() && validateAddress() &&
validatePassword()){
Toast t = Toast.makeText(getApplicationContext(),"Succesfully
Submitted",Toast.LENGTH_LONG);
t.show();
}
}
}
}

```

Output:

The screenshot shows a mobile application interface for registration. At the top, there's a red header with the text 'Registration'. Below it, a section titled 'Validation Form' contains five input fields. The first four fields are filled with the text 'admin', 'admin@gmail.com', '2345678915', and 'admins home' respectively. The fifth field is a password field, represented by a series of dots. Below these fields is a prominent red button with the word 'SUBMIT' in white capital letters. At the very bottom of the screen, a grey toast notification bubble displays the text 'Succesfully Submitted'.