

## Practical 5

### Programming UI elements AppBar, Fragments, UI Components

#### AppBar:

The app bar, also known as the action bar, is one of the most important design elements in your app's activities, because it provides a visual structure and interactive elements that are familiar to users.

#### Fragments:

Android Fragment is the part of activity, it is also known as sub-activity. There can be more than one fragment in an activity. Fragments represent multiple screen inside one activity.

#### Activity\_main.xml

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

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

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

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

        android:layout_width="match_parent"
        android:layout_height="match_parent"
        android:orientation="vertical"
        tools:context=".MainActivity"
        android:background="@drawable/tech"
    >

    <ScrollView
```

```
android:layout_width="match_parent"
android:layout_height="match_parent">

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

        <android.support.v7.widget.AppCompatTextView
            android:id="@+id/my_toolbar"

android:layout_width="match_parent"

android:layout_height="?attr/actionBarSize"

android:background="?attr/colorPrimary"
            android:elevation="4dp"
            android:text="TechChallenge"
            android:textStyle="bold"
            android:textAlignment="center"
            android:textColor="#ffffff"
            android:textSize="20sp"
        />

        <Button
            android:text="Login/Register"
            android:layout_width="220dp"
            android:layout_marginTop="150dp"
            android:layout_marginLeft="80dp"

android:background="@drawable/round_button"

android:layout_height="wrap_content"
            android:id="@+id/button"/>

    </LinearLayout>
</ScrollView>

</LinearLayout>
```

Add new activity to app (LoginRegActivity)

Also, add 2 fragments (loginfrag and regfrag) to app

Fragment\_loginfrag.xml

```
<?xml version="1.0" encoding="utf-8"?>
<FrameLayout 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=".loginfrag">

    <!-- TODO: Update blank fragment layout -->
    <LinearLayout android:layout_width="match_parent"
        android:orientation="vertical"
        android:layout_height="match_parent">

        <EditText
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:inputType="textPersonName"
            android:id="@+id/editText" android:hint="User Name"/>

        <EditText
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:inputType="textPassword"
            android:id="@+id/editText2" android:hint="Password"/>

        <Button
            android:text="Login"
            android:layout_width="match_parent"
            android:layout_height="wrap_content" android:id="@+id/button2"/>

    </LinearLayout>

</FrameLayout>
```

Fragment\_regfrag.xml

```
<?xml version="1.0" encoding="utf-8"?>
<FrameLayout 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=".regfrag">

<!-- TODO: Update blank fragment layout -->
<LinearLayout android:layout_width="match_parent"
    android:orientation="vertical"
    android:layout_height="match_parent">

    <EditText
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:inputType="textPersonName"
        android:id="@+id/editText" android:hint="User Name"/>

    <EditText
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:inputType="textPassword"
        android:id="@+id/editText2" android:hint="Password"/>

    <EditText
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:inputType="phone"
        android:ems="10"
        android:id="@+id/editText3" android:hint="Contact"/>

    <EditText
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:inputType="textEmailAddress"
        android:ems="10"
        android:id="@+id/editText4" android:hint="Email ID"/>

    <Button
        android:text="Register"
        android:layout_width="match_parent"
        android:layout_height="wrap_content" android:id="@+id/button2"/>

</LinearLayout>

</FrameLayout>

```

## Activity\_loginreg.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"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    tools:context=".LoginRegActivity">

    <LinearLayout
        android:layout_marginTop="50dp"

```

```

        android:layout_width="match_parent"
    android:layout_height="match_parent"
        android:layout_alignParentStart="true"
    android:layout_alignParentTop="true">
        <fragment
            android:id="@+id/f1"
            android:layout_width="wrap_content"
    android:layout_height="match_parent"
            android:name="com.example.abc.uielements.loginfrag"
    tools:layout="@layout/fragment_loginfrag"
        />
        <fragment
            android:id="@+id/f2"
            tools:layout="@layout/fragment_regfrag"
    android:name="com.example.abc.uielements.regfrag"
            android:layout_marginLeft="60dp"
            android:layout_width="wrap_content"
    android:layout_height="match_parent"
        />
    </LinearLayout>
    <TextView
        android:text="Login"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
    android:layout_alignParentStart="true"
        android:layout_alignParentTop="true" android:id="@+id/textView"/>
    <TextView
        android:text="Register"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
    android:layout_alignParentEnd="true"
        android:layout_alignParentTop="true"
    android:layout_marginEnd="121dp" android:id="@+id/textView2"/>

</RelativeLayout>

```

## MainActivity.kt

```

package com.example.abc.uielements

import android.content.Intent
import android.support.v7.app.AppCompatActivity
import android.os.Bundle
import android.widget.Button

class MainActivity : AppCompatActivity() {

    override fun onCreate(savedInstanceState: Bundle?) {
        super.onCreate(savedInstanceState)
        setContentView(R.layout.activity_main)

        var b=findViewById<Button>(R.id.button)

        b.setOnClickListener {

```

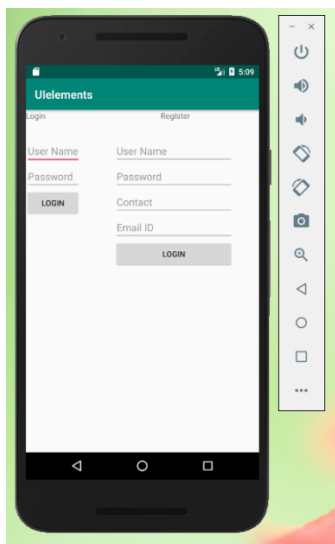
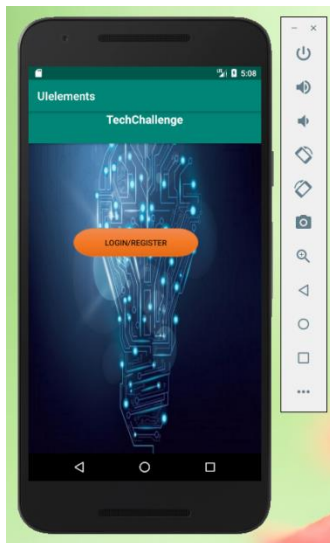
```

        var login= Intent(this,LoginRegActivity::class.java) as Intent
        startActivity(login)
    }

}
}

```

Output:



## Practical 6

### Programming menus, dialog, dialog fragments

Drop Button in activity\_main.xml

Add hexa color code for pink color in colors.xml

#### MainActivity.kt

```
package com.example.abc.dialogsandmenu

import android.app.AlertDialog
import android.content.DialogInterface
import android.support.v7.app.AppCompatActivity
import android.os.Bundle
import android.widget.Button
import kotlinx.android.synthetic.main.activity_main.*

class MainActivity : AppCompatActivity() {

    override fun onCreate(savedInstanceState: Bundle?) {
        super.onCreate(savedInstanceState)
        setContentView(R.layout.activity_main)

        var b=findViewById<Button>(R.id.button)
        val alertDialog: AlertDialog?=this?.let{
            val builder=AlertDialog.Builder(it)
            builder.apply {
                setPositiveButton("Yes", DialogInterface.OnClickListener{
                    dialog, id ->

mylayout.setBackgroundColor(resources.getColor(R.color.pcolor))
                })

                setNegativeButton("No", DialogInterface.OnClickListener{
                    dialog, id ->
                })
            }
            builder.create()
        }
        alertDialog?.setTitle("Do you want to change background color to Pink?")
        b.setOnClickListener { alertDialog?.show() }
    }
}
```

#### Creating Menus:

Add Android Resource file menu.xml to app(Resource type Menu)

```
<?xml version="1.0" encoding="utf-8"?>
<menu
xmlns:android="http://schemas.android.com/apk/
res/android">
<item android:id="@+id/m1"
        android:title="Menu 1"
        android:icon="@mipmap/ic_launcher"/>
<item android:id="@+id/m2"
        android:title="Menu 2"
        android:icon="@mipmap/ic_launcher"/>
<item android:id="@+id/m3"
        android:title="Menu 3"
        android:icon="@mipmap/ic_launcher"/>
</menu>
```

MainActivity.kt

```
override fun onCreateOptionsMenu(menu: Menu?):
Boolean {
    menuInflater.inflate(R.menu.menu, menu)
    return true
}

override fun onOptionsItemSelected(item:
MenuItem?): Boolean {
    when (item?.itemId)
    {
        R.id.m1 ->
        {Toast.makeText(this, "Menu 1 is
selected", Toast.LENGTH_LONG).show()
}
```



```
return true
}
```

```
R.id.m2 ->
{Toast.makeText(this, "Menu 2 is
selected",Toast.LENGTH_LONG).show()
return true
}
```

```
R.id.m3 ->
{Toast.makeText(this, "Menu 3 is
selected",Toast.LENGTH_LONG).show()
return true
}
else -> return
super.onOptionsItemSelected(item)
}
```

**Output:**

