Experiment 04
Write a program to implement login window using UI controls.

Learning Objective: Students should be able to write a program to implement a login window using UI controls.

Tools: Android Studio

Theory:

Android Studio:

Android Studio is the official integrated development environment (IDE) for Android application development. It is based on the IntelliJ IDEA, a Java integrated development environment for software, and incorporates its code editing and developer tools. Every project in Android Studio has one or more modalities with source code and resource files. These modalities include Android app modules, Library modules, and Google App Engine modules.

The various components used in Android studio are:

1. View: The view is the component which Android Studio provides us to design the layouts of the app. So, we can understand the view as a rectangular area which is going to contain some element inside it.
   - TextView: To add some text in your application.
   - EditText: This is used when you want to take some input from the users.
   - ImageView: To add some image in the application.
   - ProgressBar: To show the progress to something. For example, the loading screen.
   - Button: Buttons are used to trigger some action on the click of the button. It can be starting a new activity or something else.
   - ImageButton: It is used to make a clickable image.
   - CheckBox: CheckBox is used to select some options out of many available options.
   - DatePicker: To select some particular date.

   Customizations available in a View are Width and Height of a View
   i. wrap_content: This will wrap the content of a view. For example, if you are using LinearLayout and the LinearLayout consists of two buttons and you are using the height of the LinearLayout as wrap_content, then the height of the LinearLayout will be equal to the length under which both the buttons are present i.e. the LinearLayout is wrapping the contents present in it.
   ii. match_parent: If a view is using match_parent, then it will get the same size as its parent is having. For example, if LinearLayout is the parent view and the height of LinearLayout is 80dp, then if we are using Button inside the LinearLayout with height as match_parent, then the height of the Button will also be 80dp i.e. matching the parent.

2. Resources: Resources is a folder which contains the images or the layout files or some string values or something else. Following are the folders that are present inside the res folder:
   - Drawable: here, we put all our graphics, vector images, custom drawings.
   - Layout - here, we put all our screen layout files. Example: activity_main.xml

- Mipmap - here, we put the images which are used to make the logo of the application.
- Values - this folder contains four default files. colors.xml, dimens.xml, strings.xml, style.xml.

Implementation:

MainActivity.java

```java
package com.example.ps_exp_4;

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.TextView;

public class MainActivity extends AppCompatActivity {
    private EditText Name;
    private EditText Password;
    private TextView Info;
    private Button Login;
    private int counter=5;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);
        Name = (EditText) findViewById(R.id.editTextTextPersonName);
        Password = (EditText) findViewById(R.id.editTextTextPersonName2);
        Login = (Button) findViewById(R.id.button);
        Info = (TextView) findViewById(R.id.textView6);
        Info.setText("Remaining Attemps:5");
        Login.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View view) {
                validate(Name.getText().toString(),Password.getText().toString());
            }
        });
    }
    private void validate(String name,String pass){
        if((name.equals("admin"))&&(pass.equals("admin"))){
            Intent intent=new Intent(MainActivity.this,Post_Login.class);
            startActivity(intent);
        }
        else{
            counter--;
            Info.setText("Remaining Attempts: "+String.valueOf(counter));
            if(counter==0){
                Login.setEnabled(false);
            }
        }
    }
}
```

activity_main.xml

```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">

    <EditText

android:id="@+id/editTextTextPersonName"
        android:layout_width="297dp"
        android:layout_height="51dp"
        android:layout_marginStart="57dp"
        android:layout_marginTop="43dp"
        android:ems="10"
        android:inputType="textPersonName"
        android:text="Enter Name"
        android:textColor="#314E45"
        android:textSize="20sp"

app:layout_constraintStart_toStartOf="parent"

app:layout_constraintTop_toBottomOf="@+id/textView4" />

    <EditText

android:id="@+id/editTextTextPersonName2"
        android:layout_width="297dp"
        android:layout_height="51dp"
        android:layout_marginStart="57dp"
        android:layout_marginTop="40dp"
        android:ems="10"
        android:inputType="textPersonName"
        android:text="Password"
        android:textColor="#314E45"
        android:textSize="20sp"

app:layout_constraintStart_toStartOf="parent"

app:layout_constraintTop_toBottomOf="@+id/editTextTextPersonName" />

    <Button
        android:id="@+id/button"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_marginStart="105dp"
        android:layout_marginTop="30dp"
        android:backgroundTint="#314E45"
        android:text="Login"
        android:textSize="16sp"

app:layout_constraintStart_toStartOf="parent"

app:layout_constraintTop_toBottomOf="@+id/editTextTextPersonName2" />

    <Button
        android:id="@+id/button2"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_marginStart="30dp"
        android:layout_marginTop="30dp"
        android:backgroundTint="@color/white"
        android:text="Sign Up"
        android:textColor="#314E45"
        android:textSize="16sp"

app:layout_constraintStart_toEndOf="@+id/button"

app:layout_constraintTop_toBottomOf="@+id/editTextTextPersonName2"
        app:strokeColor="#314E45"
```

```
    app:strokeWidth="5px" />

<TextView
    android:id="@+id/textView2"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_marginStart="140dp"
    android:layout_marginTop="26dp"
    android:text="Forgot Password?"
    android:textSize="16sp"

app:layout_constraintStart_toStartOf="parent"

app:layout_constraintTop_toBottomOf="@+id/b
utton" />

<TextView
    android:id="@+id/textView6"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_marginStart="129dp"
    android:layout_marginTop="28dp"
    android:text="Remaining Attempts: "
    android:textSize="16sp"

app:layout_constraintStart_toStartOf="parent"

app:layout_constraintTop_toBottomOf="@+id/t
extView2" />

<TextView
```

```
    android:id="@+id/textView3"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_marginStart="172dp"
    android:layout_marginTop="141dp"
    android:text="Welcome"
    android:textColor="#314E45"
    android:textSize="16sp"

app:layout_constraintStart_toStartOf="parent"

app:layout_constraintTop_toTopOf="parent" />

<TextView
    android:id="@+id/textView4"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_marginStart="58dp"
    android:layout_marginTop="11dp"
    android:text="Please Fill the details given
below to login"
    android:textColor="#314E45"
    android:textSize="16sp"

app:layout_constraintStart_toStartOf="parent"

app:layout_constraintTop_toBottomOf="@+id/t
extView3" />

</androidx.constraintlayout.widget.ConstraintLa
yout>
```

Post_Login.java

```java
package com.example.ps_exp_4;

import android.os.Bundle;

import androidx.appcompat.app.AppCompatActivity;

public class Post_Login extends AppCompatActivity {
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.after_login);
```

```
    }
}
```

after_login.xml

```xml
<?xml version="1.0" encoding="utf-8"?>
<androidx.constraintlayout.widget.ConstraintLayout
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">

    <TextView
        android:id="@+id/textView"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Welcome Ashutosh"
        android:textColor="#314E45"
        android:textSize="20sp"
        tools:layout_editor_absoluteX="-1dp"
        tools:layout_editor_absoluteY="54dp" />
</androidx.constraintlayout.widget.ConstraintLayout>
```
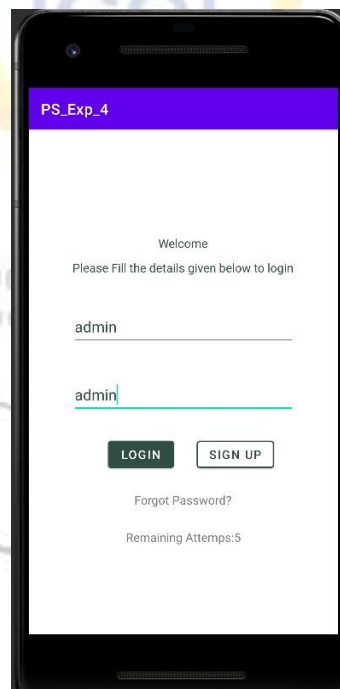
Design:

Result and Discussion: We successfully implemented a program to execute a program on Android Studio to create a Login Page which has a validation to display the next page.

Learning Outcomes: The student should have the ability to
LO1: explain how a login page can be implemented in Android Studio.
LO2: execute a simple program to create a login page with validation

Course Outcomes: Upon completion of the course students will be able to execute a simple programs illustrating login pages with validations.

Conclusion: We understood the concept of Login Page and how it can be created for an application. We also learned how to validate the login details filled by the users for the application. Several concepts related to Android Application Development were revised while performing this experiment.

For Faculty Use

Correction Parameters	Formative Assessment [40%]	Timely completion of Practical [40%]	Attendance / Learning Attitude [20%]	
Marks Obtained				