|  |
| --- |
| **BATCH AND ROLL NO: Q-7 42430** |
| **EXPERIMENT NO.5** |
| **TITLE:** Design a mobile application to create different dialog boxes and menu ( popup, option ,context) |
| **DATE OF PERFORMANCE:** |
| **DATE OF SUBMISSION:** |

**Title:** Design a mobile application to create different dialog boxes and menu (popup, option, context)

**Requirements:**

1.Android Studio

**Theory:**

**Introduction**

In the ever-evolving field of mobile application development, the user interface plays a crucial role in shaping the user experience. Dialog boxes and menus are integral components that enhance user interactions within an application. This lab focuses on the design and implementation of a mobile application featuring different types of dialog boxes, including Popup Dialogs, Option Menus, and Context Menus.

**Objective of the Lab:** The primary goal of this lab is to guide you through the process of designing a mobile application with versatile user interaction components. Specifically, you will learn how to incorporate Popup Dialogs to display crucial information or prompt user actions, Option Menus for providing a set of actions within the app, and Context Menus to offer context-specific options based on user interactions.

**Components of the Application:**

1. **Popup Dialogs:**
   * Popup Dialogs are temporary overlay windows that appear on top of the current activity.
   * They are commonly used for alerts, confirmations, or presenting additional information without navigating to a new screen.
   * Popup Dialogs can be employed to offer contextual choices, providing users with quick access to specific actions.
2. **Option Menus:**
   * Option Menus provide a set of actions that users can access within the application.
   * They typically appear at the top of the screen and offer a range of options related to the current context.
   * Option Menus are ideal for presenting a concise list of actions that users may need at any point in the application.
3. **Context Menus:**
   * Context Menus are dynamic menus that appear when a user long-presses on a specific UI element, providing context-specific actions.
   * They are useful for offering relevant options based on the user's current interaction.

**Lab Prerequisites:**

* Basic understanding of mobile application development concepts.
* Familiarity with the chosen development environment (e.g., Android Studio).
* Prior knowledge of programming languages such as Java (for Android).

**Steps:**

### **Step 1: Set Up Your Development Environment**

* Ensure that you have Android Studio installed and configured on your machine.

### **Step 2: Create a New Project**

* Open Android Studio and create a new project.
* Choose an appropriate project template, such as "Empty Activity" or "Basic Activity."

### **Step 3: Design the Main Activity Layout**

* Open the XML layout file associated with your main activity (e.g., activity\_main.xml).
* Design the layout with relevant UI elements for triggering different types of dialog boxes and Popup Menus.

### **Step 4: Implement the Java Code**

* Open the Java file associated with your main activity (e.g., MainActivity.java).
* Implement the logic for creating and showing Popup Dialogs, Option Menus, and Context Menus in response to user interactions.

### **Step 5: Implement Popup Dialogs**

* Create methods for showing Popup Dialogs with different functionalities (e.g., alerts, confirmations).
* Utilize the AlertDialog.Builder class to build and display Popup Dialogs.

### **Step 6: Implement Option Menus**

* Override the onCreateOptionsMenu method in your activity to create the Option Menu.
* Inflate the menu resource file with relevant menu items.
* Handle item selections in the onOptionsItemSelected method.

### **Step 7: Implement Context Menus**

* Register the view or views for which you want to show the Context Menu using registerForContextMenu.
* Override the onCreateContextMenu method to define the items in the Context Menu.
* Handle item selections in the onContextItemSelected method.

### **Step 8: Test Your Application**

* Run your application on an emulator or a physical device.
* Test the functionality of Popup Dialogs, Option Menus, and Context Menus by interacting with the UI elements triggering these components.

**XML Code:**

1. **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">**

**<androidx.appcompat.widget.Toolbar**

**android:id="@+id/main\_toolbar"**

**android:layout\_width="0dp"**

**android:layout\_height="wrap\_content"**

**android:background="#8844EA"**

**app:layout\_constraintEnd\_toEndOf="parent"**

**app:layout\_constraintStart\_toStartOf="parent"**

**app:layout\_constraintTop\_toTopOf="parent"**

**tools:ignore="MissingConstraints">**

**</androidx.appcompat.widget.Toolbar>**

**<TextView**

**android:id="@+id/textview"**

**android:layout\_width="wrap\_content"**

**android:layout\_height="wrap\_content"**

**android:text="Long Press me"**

**app:layout\_constraintBottom\_toBottomOf="parent"**

**app:layout\_constraintEnd\_toEndOf="parent"**

**app:layout\_constraintHorizontal\_bias="0.498"**

**app:layout\_constraintStart\_toStartOf="parent"**

**app:layout\_constraintTop\_toTopOf="parent"**

**app:layout\_constraintVertical\_bias="0.161" />**

**<Button**

**android:id="@+id/button"**

**android:layout\_width="wrap\_content"**

**android:layout\_height="wrap\_content"**

**android:layout\_marginStart="7dp"**

**android:layout\_marginTop="64dp"**

**android:text="Pop Up "**

**app:layout\_constraintEnd\_toEndOf="parent"**

**app:layout\_constraintHorizontal\_bias="0.477"**

**app:layout\_constraintStart\_toStartOf="parent"**

**app:layout\_constraintTop\_toBottomOf="@+id/textview" />**

**<Button**

**android:id="@+id/button2"**

**android:layout\_width="wrap\_content"**

**android:layout\_height="wrap\_content"**

**android:layout\_marginTop="60dp"**

**android:text="Dialog"**

**app:layout\_constraintEnd\_toEndOf="parent"**

**app:layout\_constraintHorizontal\_bias="0.498"**

**app:layout\_constraintStart\_toStartOf="parent"**

**app:layout\_constraintTop\_toBottomOf="@+id/button" />**

**<TextView**

**android:id="@+id/textView4"**

**android:layout\_width="wrap\_content"**

**android:layout\_height="wrap\_content"**

**android:text=""**

**app:layout\_constraintBottom\_toBottomOf="parent"**

**app:layout\_constraintEnd\_toEndOf="parent"**

**app:layout\_constraintHorizontal\_bias="0.498"**

**app:layout\_constraintStart\_toStartOf="parent"**

**app:layout\_constraintTop\_toTopOf="parent"**

**app:layout\_constraintVertical\_bias="0.588" />**

**<TextView**

**android:id="@+id/textView5"**

**android:layout\_width="wrap\_content"**

**android:layout\_height="wrap\_content"**

**android:text=""**

**app:layout\_constraintBottom\_toBottomOf="parent"**

**app:layout\_constraintEnd\_toEndOf="parent"**

**app:layout\_constraintHorizontal\_bias="0.498"**

**app:layout\_constraintStart\_toStartOf="parent"**

**app:layout\_constraintTop\_toBottomOf="@+id/textView4"**

**app:layout\_constraintVertical\_bias="0.156" />**

**</androidx.constraintlayout.widget.ConstraintLayout>**

1. **dialog\_layout.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">**

**<EditText**

**android:id="@+id/editTextTextPassword"**

**android:layout\_width="wrap\_content"**

**android:layout\_height="wrap\_content"**

**android:ems="10"**

**android:hint="Password"**

**android:inputType="textPassword"**

**app:layout\_constraintBottom\_toBottomOf="parent"**

**app:layout\_constraintEnd\_toEndOf="parent"**

**app:layout\_constraintHorizontal\_bias="0.497"**

**app:layout\_constraintStart\_toStartOf="parent"**

**app:layout\_constraintTop\_toTopOf="parent"**

**app:layout\_constraintVertical\_bias="0.38" />**

**<EditText**

**android:id="@+id/editTextText"**

**android:layout\_width="wrap\_content"**

**android:layout\_height="wrap\_content"**

**android:ems="10"**

**android:hint="Username"**

**android:inputType="text"**

**app:layout\_constraintBottom\_toTopOf="@+id/editTextTextPassword"**

**app:layout\_constraintEnd\_toEndOf="parent"**

**app:layout\_constraintHorizontal\_bias="0.497"**

**app:layout\_constraintStart\_toStartOf="parent"**

**app:layout\_constraintTop\_toTopOf="parent"**

**app:layout\_constraintVertical\_bias="0.675" />**

**</androidx.constraintlayout.widget.ConstraintLayout>**

1. **contextmenu.xml**

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

**<menu xmlns:android="http://schemas.android.com/apk/res/android">**

**<item**

**android:id="@+id/item1"**

**android:title="Share"/>**

**<item**

**android:id="@+id/item2"**

**android:title="Delete"/>**

**</menu>**

1. **optionmenu.xml**

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

**<menu xmlns:android="http://schemas.android.com/apk/res/android">**

**<item android:id="@+id/optionitem1"**

**android:title="View Profile"/>**

**<item android:id="@+id/optionitem2"**

**android:title="Chat Theme"/>**

**<item android:id="@+id/optionitem3"**

**android:title="Search"/>**

**<item android:id="@+id/optionitem4"**

**android:title="New group"/>**

**<item android:id="@+id/optionitem5"**

**android:title="Linked Devices"/>**

**<item android:id="@+id/optionitem6"**

**android:title="Settings"/>**

**</menu>**

1. **popupmenu.xml**

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

**<menu xmlns:android="http://schemas.android.com/apk/res/android">**

**<item**

**android:id="@+id/java"**

**android:title="Java" />**

**<item**

**android:id="@+id/kotlin"**

**android:title="Kotlin" />**

**<item**

**android:id="@+id/android"**

**android:title="Android" />**

**<item**

**android:id="@+id/react\_native"**

**android:title="React Native" />**

**</menu>**

**Java Code:**

* 1. **Example\_Dialog.java**

**package com.example.expt5AD;**

**import android.app.AlertDialog;**

**import android.app.Dialog;**

**import android.content.Context;**

**import android.content.DialogInterface;**

**import android.os.Bundle;**

**import android.view.LayoutInflater;**

**import android.view.View;**

**import android.widget.EditText;**

**import androidx.annotation.NonNull;**

**import androidx.annotation.Nullable;**

**import androidx.appcompat.app.AppCompatDialogFragment;**

**public class Example\_Dialog extends AppCompatDialogFragment {**

**private EditText Username;**

**private EditText Password;**

**private ExampleDialogListener listener;**

**@NonNull**

**@Override**

**public Dialog onCreateDialog(@Nullable Bundle savedInstanceState) {**

**AlertDialog.Builder builder = new AlertDialog.Builder(getActivity());**

**LayoutInflater inflater = getActivity().getLayoutInflater();**

**View view = inflater.inflate(R.layout.dialog\_layout,null);**

**builder.setView(view)**

**.setTitle("Login")**

**.setNegativeButton("cancel", new DialogInterface.OnClickListener() {**

**@Override**

**public void onClick(DialogInterface dialog, int which) {**

**}**

**}).setPositiveButton("OK", new DialogInterface.OnClickListener() {**

**@Override**

**public void onClick(DialogInterface dialog, int which) {**

**String username = Username.getText().toString();**

**String password = Password.getText().toString();**

**listener.applyTexts(username,password);**

**}**

**});**

**Username = view.findViewById(R.id.editTextText);**

**Password = view.findViewById(R.id.editTextTextPassword);**

**return builder.create();**

**}**

**@Override**

**public void onAttach(@NonNull Context context) {**

**super.onAttach(context);**

**listener =(ExampleDialogListener) context;**

**}**

**public interface ExampleDialogListener{**

**void applyTexts(String username,String password);**

**}**

**}**

* 1. **MainActivity.java**

**package com.example.expt5AD;**

**import androidx.annotation.NonNull;**

**import androidx.appcompat.app.AppCompatActivity;**

**import android.app.Dialog;**

**import android.os.Bundle;**

**import android.view.ContextMenu;**

**import android.view.Menu;**

**import android.view.MenuInflater;**

**import android.view.MenuItem;**

**import android.view.View;**

**import android.widget.Button;**

**import android.widget.PopupMenu;**

**import android.widget.TextView;**

**import android.widget.Toast;**

**public class MainActivity extends AppCompatActivity implements Example\_Dialog.ExampleDialogListener{**

**Button btn,dialogbtn;**

**Dialog mydialog;**

**TextView textView1,textView2;**

**@Override**

**protected void onCreate(Bundle savedInstanceState) {**

**super.onCreate(savedInstanceState);**

**setContentView(R.layout.activity\_main);**

**setSupportActionBar(findViewById(R.id.main\_toolbar));**

**TextView textView = findViewById(R.id.textview);**

**textView1 = findViewById(R.id.textView4);**

**textView2 = findViewById(R.id.textView5);**

**mydialog = new Dialog(this);**

**btn = findViewById(R.id.button);**

**btn.setOnClickListener(new View.OnClickListener() {**

**@Override**

**public void onClick(View v) {**

**PopupMenu popupMenu = new PopupMenu(MainActivity.this, btn);**

**// Inflating popup menu from popup\_menu.xml file**

**popupMenu.getMenuInflater().inflate(R.menu.popup, popupMenu.getMenu());**

**popupMenu.setOnMenuItemClickListener(new PopupMenu.OnMenuItemClickListener() {**

**@Override**

**public boolean onMenuItemClick(MenuItem menuItem) {**

**// Toast message on menu item clicked**

**Toast.makeText(MainActivity.this, "You Clicked " + menuItem.getTitle(), Toast.LENGTH\_SHORT).show();**

**return true;**

**}**

**});**

**// Showing the popup menu**

**popupMenu.show();**

**}**

**});**

**dialogbtn = findViewById(R.id.button2);**

**dialogbtn.setOnClickListener(new View.OnClickListener() {**

**@Override**

**public void onClick(View v) {**

**Example\_Dialog exampleDialog= new Example\_Dialog();**

**exampleDialog.show(getSupportFragmentManager(),"example dialog");**

**}**

**});**

**registerForContextMenu(textView);**

**}**

**@Override**

**public boolean onCreateOptionsMenu(Menu menu) {**

**MenuInflater inflater = getMenuInflater();**

**inflater.inflate(R.menu.optionmenu,menu);**

**return true;**

**}**

**@Override**

**public boolean onOptionsItemSelected(@NonNull MenuItem item) {**

**int id = item.getItemId();**

**if(id == R.id.optionitem1)**

**{**

**Toast.makeText(this, "View Profile selected", Toast.LENGTH\_SHORT).show();**

**return true;**

**}**

**else if(id==R.id.optionitem2) {**

**Toast.makeText(this, "Chat Theme selected", Toast.LENGTH\_SHORT).show();**

**return true;**

**}**

**else if(id==R.id.optionitem4) {**

**Toast.makeText(this, "New group selected", Toast.LENGTH\_SHORT).show();**

**return true;**

**}**

**else if(id==R.id.optionitem3) {**

**Toast.makeText(this, "Search selected", Toast.LENGTH\_SHORT).show();**

**return true;**

**}**

**else if(id==R.id.optionitem5) {**

**Toast.makeText(this, "Linked devices selected", Toast.LENGTH\_SHORT).show();**

**return true;**

**}**

**else if(id==R.id.optionitem6) {**

**Toast.makeText(this, "Settings selected", Toast.LENGTH\_SHORT).show();**

**return true;**

**}**

**else {**

**return super.onOptionsItemSelected(item);**

**}**

**}**

**@Override**

**public void onCreateContextMenu(ContextMenu menu, View v, ContextMenu.ContextMenuInfo menuInfo) {**

**super.onCreateContextMenu(menu, v, menuInfo);**

**MenuInflater inflater = getMenuInflater();**

**inflater.inflate(R.menu.contextmenu,menu);**

**}**

**@Override**

**public boolean onContextItemSelected(@NonNull MenuItem item) {**

**int id = item.getItemId();**

**if(id == R.id.item1)**

**{**

**Toast.makeText(this, "Share selected", Toast.LENGTH\_SHORT).show();**

**return true;**

**}**

**else if(id==R.id.item2) {**

**Toast.makeText(this, "Delete selected", Toast.LENGTH\_SHORT).show();**

**return true;**

**}**

**else {**

**return super.onContextItemSelected(item);**

**}**

**}**

**@Override**

**public void applyTexts(String username, String password) {**

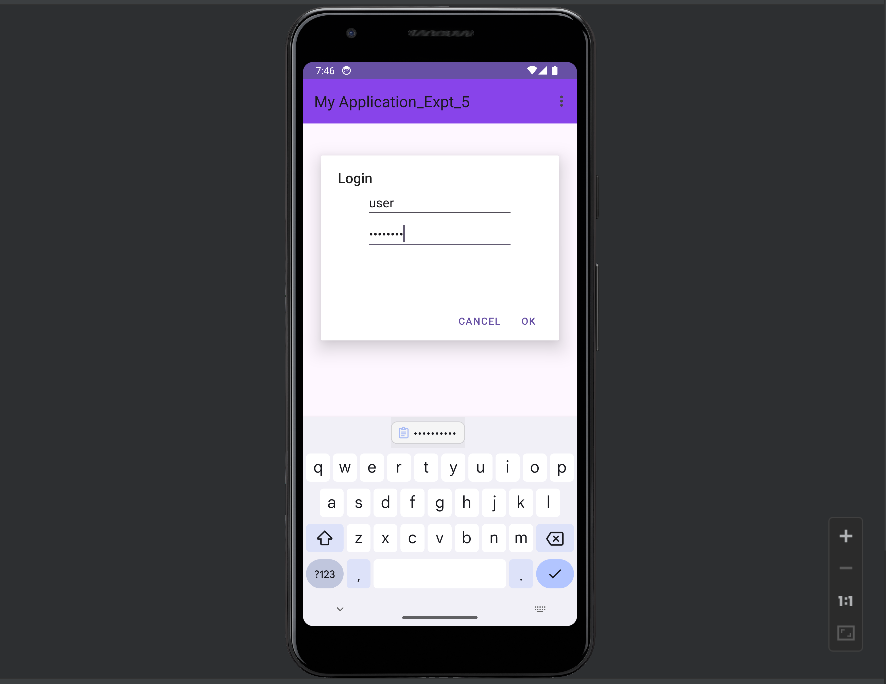
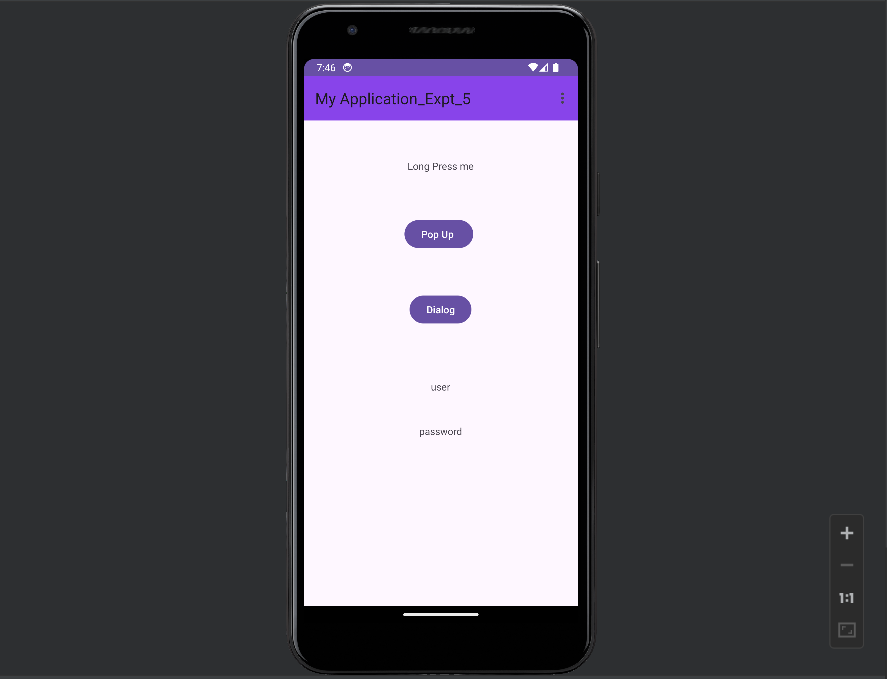
**textView1.setText(username);**

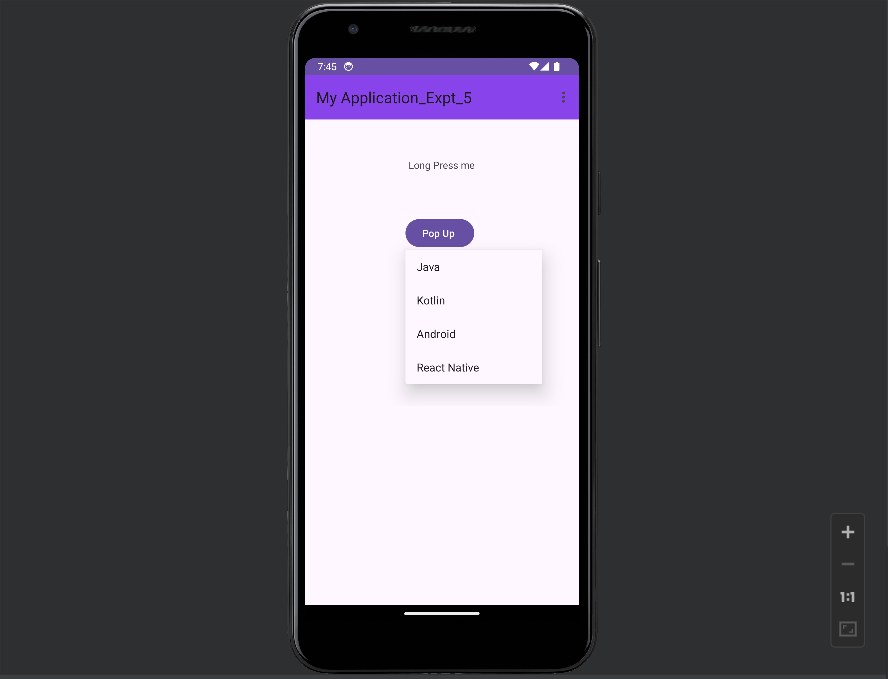
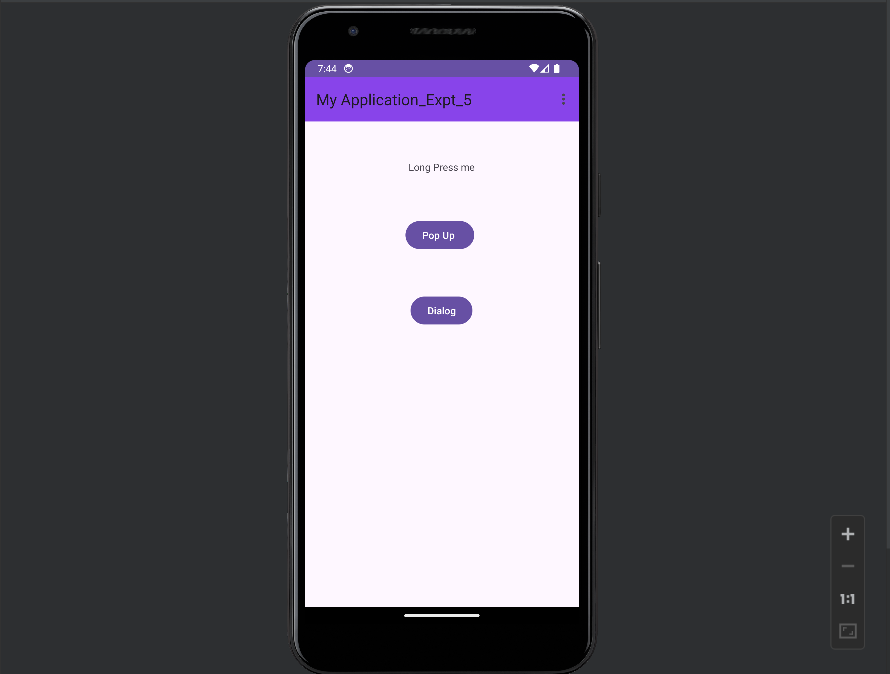
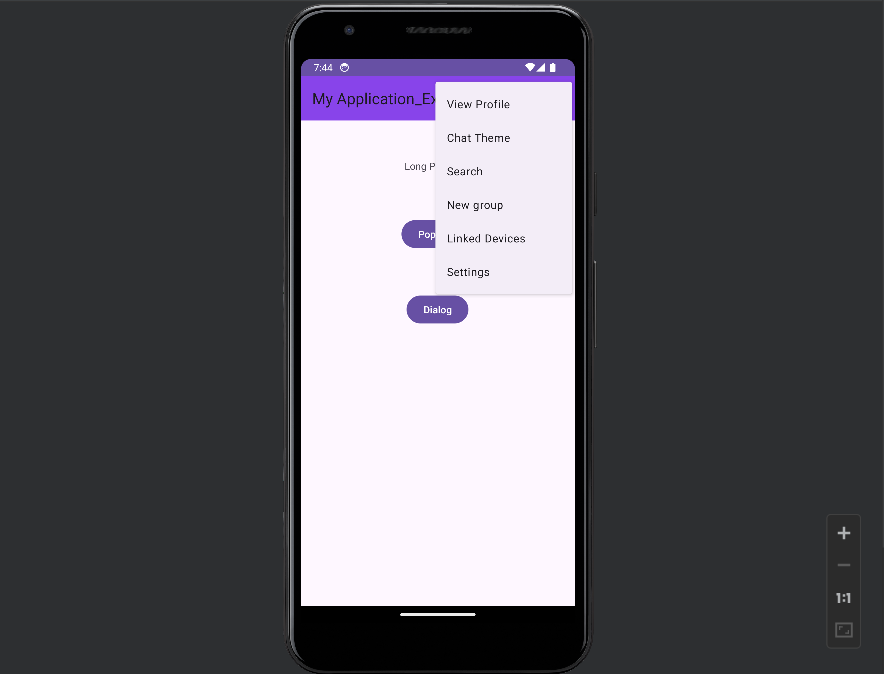
**textView2.setText(password);**

**}**

**}**

**Output:**

****

** **

#### Conclusion:

#### ………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………