

# LAB Journal for Mobile application Development

**LAB-1:-Create "Hello World" application. That will display "Hello World" in the middle of the screen in the red color with white background.**

## XML File

```
<!-- activity_main.xml -->

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

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

    android:layout_width="match_parent"

    android:layout_height="match_parent"

    android:background="#FFFFFF"> <!-- White background color -->

    <TextView

        android:id="@+id/textHelloWorld"

        android:layout_width="wrap_content"

        android:layout_height="wrap_content"

        android:text="Hello World"

        android:textColor="#FF0000" <!-- Red text color -->

        android:textSize="24sp"

        android:layout_centerInParent="true"/>

    </RelativeLayout>
```

## Java File

```
// MainActivity.java

import android.os.Bundle;

import androidx.appcompat.app.AppCompatActivity;

public class MainActivity extends AppCompatActivity {

    @Override

    protected void onCreate(Bundle savedInstanceState) {

        super.onCreate(savedInstanceState);

        setContentView(R.layout.activity_main);

    }

}
```

**Lab 2-Create My\_info application. That will display your name, qualification, contact num, email id and address with background color gray. All details must have different color. (using XML) Change color of above program using java code**

### **XML File**

```
<!-- activity_main.xml -->

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

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

    android:layout_width="match_parent"

    android:layout_height="match_parent"

    android:background="#808080"> <!-- Gray background color -->

    <TextView

        android:id="@+id/textName"

        android:layout_width="wrap_content"

        android:layout_height="wrap_content"

        android:text="Name: Your Name"

        android:textColor="#FF0000" <!-- Red text color -->

        android:textSize="18sp"

        android:layout_marginTop="20dp"

        android:layout_marginLeft="20dp"/>

    <TextView

        android:id="@+id/textQualification"

        android:layout_width="wrap_content"

        android:layout_height="wrap_content"

        android:text="Qualification: Your Qualification"

        android:textColor="#00FF00" <!-- Green text color -->

        android:textSize="18sp"

        android:layout_below="@id/textName"
```

```
android:layout_marginTop="20dp"
android:layout_marginLeft="20dp"/>
```

```
<TextView
```

```
    android:id="@+id/textContact"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="Contact: Your Contact Number"
    android:textColor="#0000FF" <!-- Blue text color -->
    android:textSize="18sp"
    android:layout_below="@id/textQualification"
    android:layout_marginTop="20dp"
    android:layout_marginLeft="20dp"/>
```

```
<TextView
```

```
    android:id="@+id/textEmail"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="Email: Your Email Address"
    android:textColor="#FF00FF" <!-- Purple text color -->
    android:textSize="18sp"
    android:layout_below="@id/textContact"
    android:layout_marginTop="20dp"
    android:layout_marginLeft="20dp"/>
```

```
<TextView
```

```
        android:id="@+id/textAddress"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Address: Your Address"
        android:textColor="#FFFF00" <!-- Yellow text color -->
        android:textSize="18sp"
        android:layout_below="@id/textEmail"
        android:layout_marginTop="20dp"
        android:layout_marginLeft="20dp"/>
</RelativeLayout>
```

\*\*\*\*\*Java Code\*\*\*\*\*

```
// MainActivity.java

import android.graphics.Color;
import android.os.Bundle;
import android.widget.TextView;

import androidx.appcompat.app.AppCompatActivity;

public class MainActivity extends AppCompatActivity {

    @Override

    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);
    }
}
```

```
// Get references to TextViews

TextView textName = findViewById(R.id.textName);

TextView textQualification = findViewById(R.id.textQualification);

TextView textContact = findViewById(R.id.textContact);

TextView textEmail = findViewById(R.id.textEmail);

TextView textAddress = findViewById(R.id.textAddress);


// Change text color dynamically

textName.setTextColor(Color.RED);

textQualification.setTextColor(Color.GREEN);

textContact.setTextColor(Color.BLUE);

textEmail.setTextColor(Color.MAGENTA);

textAddress.setTextColor(Color.YELLOW);

    }

}
```

### Lab 3 Write a program to demonstrate life cycle of activity in android

Create a new app ,give name "Life\_cycle\_Activity"

The `MainActivity` class overrides various lifecycle methods, and log messages are used to indicate when each method is called. This program is intended to be run on an Android device or emulator:

```
import android.os.Bundle;
import android.util.Log;
```

```
import androidx.annotation.NonNull;
import androidx.appcompat.app.AppCompatActivity;
```

```
public class MainActivity extends AppCompatActivity {
```

```
    private static final String TAG = "MainActivity";
```

```
    @Override
```

```
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);
```

```
//    Log.d(TAG, "onCreate: Activity created");
    Toast.makeText(this,"onCreate()",Toast.Lenght_Short).show();
}
```

```
    @Override
```

```
    protected void onStart() {
        super.onStart();
        Log.d(TAG, "onStart: Activity started");
    }
```

```
    @Override
```

```
    protected void onResume() {
        super.onResume();
        Log.d(TAG, "onResume: Activity resumed");
    }
```

```
    @Override
```

```
    protected void onPause() {
        super.onPause();
        Log.d(TAG, "onPause: Activity paused");
    }
```

```
    @Override
```

```
    protected void onStop() {
        super.onStop();
        Log.d(TAG, "onStop: Activity stopped");
    }
```

```
}

@Override
protected void onRestart() {
    super.onRestart();
    Log.d(TAG, "onRestart: Activity restarted");
}

@Override
protected void onDestroy() {
    super.onDestroy();
    Log.d(TAG, "onDestroy: Activity destroyed");
}

@Override
protected void onSaveInstanceState(@NonNull Bundle outState) {
    super.onSaveInstanceState(outState);
    Log.d(TAG, "onSaveInstanceState: Saving instance state");
}

@Override
protected void onRestoreInstanceState(@NonNull Bundle savedInstanceState) {
    super.onRestoreInstanceState(savedInstanceState);
    Log.d(TAG, "onRestoreInstanceState: Restoring instance state");
}
}
```



**LAB-4:- Create an application that designs a layout with a text box and button named submit. The user should enter the text in the text box. When the submit button is clicked then the text in the text box should be displayed in the toast.**

**Layout File(activity\_main.xml)**

```
<!-- res/layout/activity_main.xml -->
<RelativeLayout 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"
    android:paddingLeft="16dp"
    android:paddingTop="16dp"
    android:paddingRight="16dp"
    android:paddingBottom="16dp"
    tools:context=".MainActivity">

    <EditText
        android:id="@+id/editText"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:hint="Enter text"/>

    <Button
        android:id="@+id/submitButton"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_below="@id/editText"
        android:layout_marginTop="16dp"
        android:text="Submit"/>

</RelativeLayout>
```

\*\*\*\*\*Main Activity\*\*\*\*\*

```
// app/src/main/java/com.example.yourappname/MainActivity.java
package com.example.yourappname;
```

```
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 {

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
```

```
setContentView(R.layout.activity_main);

final EditText editText = findViewById(R.id.editText);
Button submitButton = findViewById(R.id.submitButton);

submitButton.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View v) {
        // Get the text from the EditText
        String enteredText = editText.getText().toString();

        // Display the text in a toast
        Toast.makeText(getApplicationContext(), "Text entered: " + enteredText,
Toast.LENGTH_SHORT).show();
    }
});
}
```

**Lab 5. Create an android application named Arithmetic\_op which perform all basic arithmetic operation like addition, subtraction, multiplication and division.**

1. Create a new Android project:  
Open Android Studio and create a new project named "Arithmetic\_op."
2. Modify the layout (activity\_main.xml):  
Open the activity\_main.xml layout file and add buttons for each operation, an input field for each operand, and a TextView to display the result. Here's an example:

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

```
<RelativeLayout 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=".MainActivity">
```

```
    <EditText
```

```
        android:id="@+id/editTextOperand1"
```

```
        android:layout_width="match_parent"
```

```
        android:layout_height="wrap_content"
```

```
        android:layout_marginTop="16dp"
```

```
        android:hint="Enter operand 1"
```

```
        android:inputType="numberDecimal"/>
```

```
    <EditText
```

```
        android:id="@+id/editTextOperand2"
```

```
        android:layout_width="match_parent"
```

```
        android:layout_height="wrap_content"
```

```
        android:layout_below="@id/editTextOperand1"
```

```
android:layout_marginTop="16dp"

android:hint="Enter operand 2"

android:inputType="numberDecimal"/>
```

```
<Button
```

```
    android:id="@+id/buttonAdd"

    android:layout_width="wrap_content"

    android:layout_height="wrap_content"

    android:layout_below="@id/editTextOperand2"

    android:layout_marginTop="16dp"

    android:text="Add"/>
```

```
<Button
```

```
    android:id="@+id/buttonSubtract"

    android:layout_width="wrap_content"

    android:layout_height="wrap_content"

    android:layout_below="@id/buttonAdd"

    android:layout_marginTop="16dp"

    android:text="Subtract"/>
```

```
<Button
```

```
    android:id="@+id/buttonMultiply"

    android:layout_width="wrap_content"

    android:layout_height="wrap_content"
```

```
android:layout_below="@id/buttonSubtract"
```

```
android:layout_marginTop="16dp"
```

```
android:text="Multiply"/>
```

```
<Button
```

```
    android:id="@+id/buttonDivide"
```

```
    android:layout_width="wrap_content"
```

```
    android:layout_height="wrap_content"
```

```
    android:layout_below="@id/buttonMultiply"
```

```
    android:layout_marginTop="16dp"
```

```
    android:text="Divide"/>
```

```
<TextView
```

```
    android:id="@+id/textViewResult"
```

```
    android:layout_width="wrap_content"
```

```
    android:layout_height="wrap_content"
```

```
    android:layout_below="@id/buttonDivide"
```

```
    android:layout_marginTop="16dp"/>
```

```
</RelativeLayout>
```

3.Implement the arithmetic operations (MainActivity.java):

Open the **MainActivity.java** file and add the logic to perform arithmetic operations based on user input.

```
import android.os.Bundle;
```

```
import android.view.View;

import android.widget.Button;

import android.widget.EditText;

import android.widget.TextView;

import androidx.appcompat.app.AppCompatActivity;

public class MainActivity extends AppCompatActivity {

    private EditText editTextOperand1;

    private EditText editTextOperand2;

    private TextView textViewResult;

    @Override

    protected void onCreate(Bundle savedInstanceState) {

        super.onCreate(savedInstanceState);

        setContentView(R.layout.activity_main);

        editTextOperand1 = findViewById(R.id.editTextOperand1);

        editTextOperand2 = findViewById(R.id.editTextOperand2);

        textViewResult = findViewById(R.id.textViewResult);

        Button buttonAdd = findViewById(R.id.buttonAdd);

        Button buttonSubtract = findViewById(R.id.buttonSubtract);

        Button buttonMultiply = findViewById(R.id.buttonMultiply);

        Button buttonDivide = findViewById(R.id.buttonDivide);

        buttonAdd.setOnClickListener(new View.OnClickListener() {

            @Override

            public void onClick(View v) {
```

```
        performOperation('+');  
    }  
});
```

```
buttonSubtract.setOnClickListener(new View.OnClickListener() {  
  
    @Override  
  
    public void onClick(View v) {  
  
        performOperation('-');  
  
    }  
  
});
```

```
buttonMultiply.setOnClickListener(new View.OnClickListener() {  
  
    @Override  
  
    public void onClick(View v) {  
  
        performOperation('*');  
  
    }  
  
});
```

```
buttonDivide.setOnClickListener(new View.OnClickListener() {  
  
    @Override  
  
    public void onClick(View v) {  
  
        performOperation('/');  
  
    }  
  
});
```

```
}
```

```
private void performOperation(char operation) {  
  
    try {  
  
        double operand1 = Double.parseDouble(editTextOperand1.getText().toString());  
  
        double operand2 = Double.parseDouble(editTextOperand2.getText().toString());  
  
        double result = 0;  
  
        switch (operation) {  
  
            case '+':  
  
                result = operand1 + operand2;  
  
                break;  
  
            case '-':  
  
                result = operand1 - operand2;  
  
                break;  
  
            case '*':  
  
                result = operand1 * operand2;  
  
                break;  
  
            case '/':  
  
                if (operand2 != 0) {  
  
                    result = operand1 / operand2;  
  
                } else {  
  
                    textViewResult.setText("Cannot divide by zero");  
  
                    return;  
                }  
            }  
        }  
    }  
}
```



```
        }  
        break;  
    }  
  
    textViewResult.setText("Result: " + result);  
} catch (NumberFormatException e) {  
    textViewResult.setText("Invalid input. Please enter valid numbers.");  
}  
}  
}
```

**Lab 6:-Create simple program which show the use of auto complete text view.**

**To create a simple program that demonstrates the use of `AutoCompleteTextView` in Android, you can follow the steps below. This example assumes you are using Java for Android development.**

1) Create a new Android project:

Open Android Studio and create a new project.

2) Add `AutoCompleteTextView` to your layout:

Open your `activity_main.xml` layout file and add the following code to include an `AutoCompleteTextView`:

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

<RelativeLayout 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=".MainActivity">

    <AutoCompleteTextView

        android:id="@+id/autoCompleteTextView"

        android:layout_width="match_parent"

        android:layout_height="wrap_content"

        android:layout_margin="16dp"

        android:hint="Type here..."

    />

</RelativeLayout>
```

3. Create an array of suggestions:

In your `MainActivity.java`, define an array of suggestions that will be used for autocompletion.

```
import android.os.Bundle;
```

```

import android.widget.ArrayAdapter;

import android.widget.AutoCompleteTextView;

import androidx.appcompat.app.AppCompatActivity;

public class MainActivity extends AppCompatActivity {

    @Override

    protected void onCreate(Bundle savedInstanceState) {

        super.onCreate(savedInstanceState);

        setContentView(R.layout.activity_main);

        // Array of suggestions

        String[] suggestions = {"Apple", "Banana", "Cherry", "Date", "Fig", "Grape", "Kiwi"};

        // Creating an ArrayAdapter

        ArrayAdapter<String> adapter = new ArrayAdapter<>(this,

            android.R.layout.simple_dropdown_item_1line, suggestions);

        // Getting reference to AutoCompleteTextView

        AutoCompleteTextView autoCompleteTextView =

        findViewById(R.id.autoCompleteTextView);

        // Setting the adapter

        autoCompleteTextView.setAdapter(adapter);

    }

}

```

#### 4. Run your application:

Build and run your application on an emulator or a physical device. You should see an AutoCompleteTextView where you can start typing, and it will suggest completions based on the provided array.

This simple program demonstrates the basic usage of `AutoCompleteTextView` in Android. Users can type in the `AutoCompleteTextView`, and it will suggest options based on the predefined array.

**LAB-7:-Create sample application with login module.(Check username and password)  
On successful login, go to next screen. And on failing login, alert user using Toast. Also pass username to next screen.**

1) Create the layout for `activity_main.xml` (res/layout/activity\_main.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"
```

```
    android:layout_width="match_parent"
```

```
    android:layout_height="match_parent"
```

```
    android:padding="16dp"
```

```
    tools:context=".MainActivity">
```

```
<EditText
```

```
    android:id="@+id/editTextUsername"
```

```
    android:layout_width="match_parent"
```

```
    android:layout_height="wrap_content"
```

```
    android:hint="Username" />
```

```
<EditText
```

```
    android:id="@+id/editTextPassword"
```

```
    android:layout_width="match_parent"
```

```
    android:layout_height="wrap_content"
```

```
    android:layout_below="@id/editTextUsername"
```

```
    android:layout_marginTop="8dp"
```

```
    android:inputType="textPassword"
```

```
    android:hint="Password" />
```

```
<Button
```

```
    android:id="@+id/buttonLogin"
```

```
    android:layout_width="match_parent"
```

```
    android:layout_height="wrap_content"
```

```
    android:layout_below="@id/editTextPassword"
```

```
    android:layout_marginTop="16dp"
```

```
    android:text="Login" />
```

```
</RelativeLayout>
```

2) Create the layout for **activity\_dashboard.xml** (res/layout/activity\_dashboard.xml):

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

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

```
    android:layout_width="match_parent"
```

```
    android:layout_height="match_parent"
```

```
    android:orientation="vertical"
```

```
    android:padding="16dp">
```

```
<TextView
```

```
    android:id="@+id/textViewWelcome"
```

```
    android:layout_width="wrap_content"
```

```
    android:layout_height="wrap_content"
```

```
    android:text="Welcome, "
```

```
    android:textSize="18sp"
```

```
        android:textStyle="bold" />
</LinearLayout>
```

3) Create **MainActivity.java** (src/com/example/loginexample/MainActivity.java):

```
package com.example.loginexample;

import android.content.Intent;

import android.os.Bundle;

import android.view.View;

import android.widget.Button;

import android.widget.EditText;

import android.widget.Toast;

import androidx.appcompat.app.AppCompatActivity;

public class MainActivity extends AppCompatActivity {

    private EditText editTextUsername, EditText editTextPassword;

    @Override

    protected void onCreate(Bundle savedInstanceState) {

        super.onCreate(savedInstanceState);

        setContentView(R.layout.activity_main);

        editTextUsername = findViewById(R.id.editTextUsername);

        editTextPassword = findViewById(R.id.editTextPassword);

        Button buttonLogin = findViewById(R.id.buttonLogin);

        buttonLogin.setOnClickListener(new View.OnClickListener() {

            @Override

            public void onClick(View view) {
```

```

        //this us user defined method.i called this method

        login();

    }

});

}

private void login() {

    String username = editTextUsername.getText().toString();

    String password = editTextPassword.getText().toString();

    // For simplicity, checking if both username and password are "admin"
    if (username.equals("admin") && password.equals("admin")) {

        // Successful login, go to DashboardActivity

        Intent intent = new Intent(MainActivity.this, DashboardActivity.class);

        intent.putExtra("username", username);

        startActivity(intent);

    } else {

        // Failed login, show a Toast

        Toast.makeText(this, "Login failed. Check username and password.",
        Toast.LENGTH_SHORT).show();

    }

}

}

```

4) Create **DashboardActivity.java** (src/com/example/loginexample/DashboardActivity.java):

```

package com.example.loginexample;

import android.os.Bundle;

```



```

import android.widget.TextView;

import androidx.appcompat.app.AppCompatActivity;

public class DashboardActivity extends AppCompatActivity {

    @Override

    protected void onCreate(Bundle savedInstanceState) {

        super.onCreate(savedInstanceState);

        setContentView(R.layout.activity_dashboard);

        TextView textViewWelcome = findViewById(R.id.textViewWelcome);

        // Get the username passed from MainActivity

        String username = getIntent().getStringExtra("username");

        // Display a welcome message with the username

        //  textViewWelcome.setText(getString(R.string.welcome_message, username));

        textViewWelcome.setText("Welcome"+username);

    }

}

```

5) textViewWelcome.setText(getString(R.string.welcome\_message, username)); if you writing the above line then you need to write bellow code in

Add a string resource in [res/values/strings.xml](#) for the welcome message:

```

<resources>

    <string name="app_name">LoginExample</string>

    <string name="welcome_message">Welcome, %1$s!</string>

</resources>

```

6) Finally if required, update the [AndroidManifest.xml](#) to include both activities:

```

<application

```

```
    android:allowBackup="true"

    android:icon="@mipmap/ic_launcher"

    android:label="@string/app_name"

    android:roundIcon="@mipmap/ic_launcher_round"

    android:supportRtl="true"

    android:theme="@style/AppTheme">

    <activity android:name=".MainActivity">

        <intent-filter>

            <action android:name="android.intent.action.MAIN" />

            <category android:name="android.intent.category.LAUNCHER" />

        </intent-filter>

    </activity>

    <activity android:name=".DashboardActivity" />

</application>
```

**LAB-8 Create login application where you will have to validate Email ID (UserName). Till the user name and password is not validated, login button should remain disabled.**

## Activity\_main.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"

    android:layout_width="match_parent"

    android:layout_height="match_parent"

    android:padding="16dp"

    tools:context=".MainActivity">

    <EditText

        android:id="@+id/editTextEmail"

        android:layout_width="match_parent"

        android:layout_height="wrap_content"

        android:hint="Email"

        android:inputType="textEmailAddress"

        android:layout_marginBottom="8dp"/>

    <EditText

        android:id="@+id/editTextPassword"

        android:layout_width="match_parent"

        android:layout_height="wrap_content"

        android:hint="Password"

        android:inputType="textPassword"

        android:layout_below="@id/editTextEmail"
```

```
    android:layout_marginTop="8dp"

    android:layout_marginBottom="16dp"/>
```

```
<Button
```

```
    android:id="@+id/btnLogin"

    android:layout_width="match_parent"

    android:layout_height="wrap_content"

    android:text="Login"

    android:layout_below="@id/editTextPassword"

    android:layout_marginTop="16dp"

    android:enabled="false"/>
```

```
</RelativeLayout>
```

ActivityMain.java

```
import android.os.Bundle;

import android.text.Editable;

import android.text.TextUtils;

import android.text.TextWatcher;

import android.view.View;

import android.widget.Button;

import android.widget.EditText;


import androidx.appcompat.app.AppCompatActivity;

public class MainActivity extends AppCompatActivity {
```

```

private EditText editTextEmail, editTextPassword;

private Button btnLogin;

@Override

protected void onCreate(Bundle savedInstanceState) {

    super.onCreate(savedInstanceState);

    setContentView(R.layout.activity_main);

    editTextEmail = findViewById(R.id.editTextEmail);

    editTextPassword = findViewById(R.id.editTextPassword);

    btnLogin = findViewById(R.id.btnLogin);

    // Add text change listener to email EditText

    editTextEmail.addTextChangedListener(new TextWatcher() {

        @Override

        public void beforeTextChanged(CharSequence charSequence, int start, int before, int
count) {

            }

        @Override

        public void onTextChanged(CharSequence charSequence, int start, int before, int
count) {

            // Enable the login button only if the email is not empty and is a valid email
address

            btnLogin.setEnabled(!TextUtils.isEmpty(charSequence) &&
android.util.Patterns.EMAIL_ADDRESS.matcher(charSequence).matches());

        }

        @Override

        public void afterTextChanged(Editable editable) {

```

```
    }

});

// Add click listener to login button

btnLogin.setOnClickListener(new View.OnClickListener() {

    @Override

    public void onClick(View view) {

        // Implement your login logic here

    }

});

}

}
```

**Lab-9:-Create an application that will pass username and password on the next screen.**

**Activity\_main.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"
```

```
    android:layout_width="match_parent"
```

```
    android:layout_height="match_parent"
```

```
    android:padding="16dp"
```

```
    tools:context=".MainActivity">
```

```
    <EditText
```

```
        android:id="@+id/editTextUsername"
```

```
        android:layout_width="match_parent"
```

```
        android:layout_height="wrap_content"
```

```
        android:hint="Username"
```

```
        android:inputType="text"/>
```

```
    <EditText
```

```
        android:id="@+id/editTextPassword"
```

```
        android:layout_width="match_parent"
```

```
        android:layout_height="wrap_content"
```

```
        android:layout_below="@id/editTextUsername"
```

```
        android:layout_marginTop="8dp"
```

```
        android:hint="Password"
```

```
        android:inputType="textPassword"
```

```
        android:layout_marginBottom="16dp"/>
```

```
<Button

    android:id="@+id/btnNext"

    android:layout_width="match_parent"

    android:layout_height="wrap_content"

    android:text="Next"

    android:layout_below="@id/editTextPassword"

    android:layout_marginTop="16dp"/>
```

```
</RelativeLayout>
```

### **MainActivity.java**

```
import android.content.Intent;

import android.os.Bundle;

import android.view.View;

import android.widget.Button;

import android.widget.EditText;


import androidx.appcompat.app.AppCompatActivity;


public class MainActivity extends AppCompatActivity {


    private EditText editTextUsername, editTextPassword;

    private Button btnNext;


    @Override
```



```

protected void onCreate(Bundle savedInstanceState) {

    super.onCreate(savedInstanceState);

    setContentView(R.layout.activity_main);


    editTextUsername = findViewById(R.id.editTextUsername);

    editTextPassword = findViewById(R.id.editTextPassword);

    btnNext = findViewById(R.id.btnNext);

    btnNext.setOnClickListener(new View.OnClickListener() {

        @Override

        public void onClick(View view) {

            // Get username and password from EditTexts

            String username = editTextUsername.getText().toString();

            String password = editTextPassword.getText().toString();

            // Create an Intent to start the next activity

            Intent intent = new Intent(MainActivity.this, SecondActivity.class);

            // Pass username and password as extras to the next activity

            intent.putExtra("USERNAME", username);

            intent.putExtra("PASSWORD", password);

            // Start the next activity

            startActivity(intent);

        }

    });
}

```

**LAB-10:-Create simple Application which show the use of List view.**

**XML file**

```

<?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=".ListViewLab_10">

    <ListView
        android:id="@+id/listView"
        android:layout_width="0dp"
        android:layout_height="0dp"
        android:layout_marginStart="1dp"
        android:layout_marginTop="1dp"
        android:layout_marginEnd="1dp"
        android:layout_marginBottom="1dp"
        app:layout_constraintBottom_toBottomOf="parent"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toTopOf="parent" />
</androidx.constraintlayout.widget.ConstraintLayout>

```

### **MainActivity.java**

**LAB-11:- Create simple Application which show the use of Radio button, take 3 radio button.**

**When radio button is selected we have to show the text of radio button using Toast.**

### **XML file**

```
<?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=".RadioButtonLab_11">

    <RadioGroup
        android:id="@+id/radioGroup"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:orientation="vertical"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toTopOf="parent"/>

    <RadioButton
        android:id="@+id/radioButtonTomato"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_marginTop="110dp"
        android:text="Tomato"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toTopOf="parent"
```

```
tools:ignore="MissingConstraints" />
```

```
<RadioButton
```

```
    android:id="@+id/radioButtonBrinjal"
```

```
    android:layout_width="wrap_content"
```

```
    android:layout_height="wrap_content"
```

```
    android:layout_marginTop="48dp"
```

```
    android:text="Brinjal"
```

```
    app:layout_constraintStart_toStartOf="@+id/radioButtonTomato"
```

```
    app:layout_constraintTop_toBottomOf="@+id/radioButtonTomato"
```

```
    tools:ignore="MissingConstraints" />
```

```
<RadioButton
```

```
    android:id="@+id/radioButtonPotato"
```

```
    android:layout_width="wrap_content"
```

```
    android:layout_height="wrap_content"
```

```
    android:layout_marginTop="51dp"
```

```
    android:text="Potato"
```

```
    app:layout_constraintStart_toStartOf="@+id/radioButtonBrinjal"
```

```
    app:layout_constraintTop_toBottomOf="@+id/radioButtonBrinjal"
```

```
    tools:ignore="MissingConstraints" />
```

```
/>
```

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

## **MainActivity.java**

```
package com.example.welcome;
```

```
import androidx.appcompat.app.AppCompatActivity;
```

```
import android.os.Bundle;
import android.widget.RadioButton;
import android.widget.RadioGroup;
import android.widget.Toast;
```

```
public class RadioButtonLab_11 extends AppCompatActivity {
```

```
    RadioGroup radioGroup;
```

```
    @Override
```

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

```
        super.onCreate(savedInstanceState);
```

```
        setContentView(R.layout.activity_radio_button_lab11);
```

//A radio button is a widget that allows users to select a single option from a set of mutually exclusive choices.

//It's commonly used in scenarios where users need to choose one option from multiple options presented in a list.

```
        radioGroup = findViewById(R.id.radioGroup);
```

```
        radioGroup.setOnCheckedChangeListener(new
RadioGroup.OnCheckedChangeListener() {
```

```
            @Override
```

```
            public void onCheckedChanged(RadioGroup group, int checkedId) {
```

//here no need to pass the radioButton id which is given in xml file. R.id. job is doing by checkedId

```
                RadioButton radioButton = findViewById(checkedId);
```

```
                if (radioButton.isChecked()){
```

```
                    Toast.makeText(getApplicationContext(),"You
selected"+radioButton.getText().toString(),Toast.LENGTH_LONG).show();
```

```
                } else {
```

```
                    Toast.makeText(getApplicationContext(),"Not
selected",Toast.LENGTH_LONG).show();
```

```
        }  
    }  
};  
}
```

**LAB-12:- Create simple Application which show the use of Checkbox component, take 3 checkbox and 1 button when you check the checkboxes and click on button then you have to show which checkbox is check with text of checkbox using Toast.**

#### XML file

```
<?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"
    android:padding="16dp"
    tools:context=".CheckBoxLab_12">

    <CheckBox
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:id="@+id/checkbox1"
        android:text="CheckBoxOne"
        android:layout_marginTop="16dp"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintEnd_toStartOf="parent"
        app:layout_constraintTop_toTopOf="parent" />

    <CheckBox
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:id="@+id/checkbox2"
        android:text="CheckBoxTwo"
        android:layout_marginTop="36dp"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintEnd_toStartOf="parent"
        app:layout_constraintTop_toTopOf="parent" />
```

```
<CheckBox
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:id="@+id/checkbox3"
    android:text="CheckBoxThree"
    android:layout_marginTop="60dp"
    app:layout_constraintEnd_toEndOf="parent"
    app:layout_constraintEnd_toStartOf="parent"
    app:layout_constraintTop_toTopOf="parent" />
```

```
<Button
    android:id="@+id/showButton"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="Show Checked Checkboxes"
    android:layout_marginTop="200dp"
    app:layout_constraintStart_toStartOf="parent"
    app:layout_constraintEnd_toEndOf="parent"
    app:layout_constraintTop_toTopOf="parent" />
```

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

### **MainActivity.java**

```
package com.example.welcome;

import static android.os.Build.VERSION_CODES.R;

import androidx.appcompat.app.AppCompatActivity;

import android.os.Bundle;
import android.view.View;
import android.widget.Button;
import android.widget.CheckBox;
```



```
import android.widget.Toast;
```

```
public class CheckBoxLab_12 extends AppCompatActivity {  
    CheckBox checkBox1,checkBox2,checkBox3;  
    Button btnCheck;  
    @Override  
    protected void onCreate(Bundle savedInstanceState) {  
        super.onCreate(savedInstanceState);  
        setContentView(R.layout.activity_check_box_lab12);  
  
        checkBox1 = findViewById(R.id.checkbox1);  
        checkBox2 = findViewById(R.id.checkbox2);  
        checkBox3 = findViewById(R.id.checkbox3);  
        btnCheck =findViewById(R.id.showButton);  
  
        btnCheck.setOnClickListener(new View.OnClickListener() {  
            @Override  
            public void onClick(View v) {  
                showCheckedCheckboxes(); //it is user defined method  
            }  
        });  
    }  
    private void showCheckedCheckboxes(){  
        StringBuilder checkedCheckBoxes = new StringBuilder("Checked Checkboxes : ");  
        if (checkBox1.isChecked()){  
            checkedCheckBoxes.append("check box 1").append( checkBox1.getText()).append(",");  
        }  
        if (checkBox2.isChecked()){  
            checkedCheckBoxes.append("check box 2").append( checkBox2.getText()).append(",");  
        }  
    }  
}
```

```
if (checkBox3.isChecked()){  
    checkedCheckBoxes.append("check box 3").append( checkBox3.getText()).append(",");  
}  
  
Toast.makeText(getApplicationContext(),checkedCheckBoxes.toString(),Toast.LENGTH_SHORT).show()  
;  
}  
}
```

**LAB-13:- Create simple Application which show the use of CheckBox component, take 3 checkbox when you check the checkboxes then you have to show which checkbox is check with text of checkbox using Toast.**

#### **XML file**

```
<?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=".CheckBoxLab_13">

    <CheckBox
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:id="@+id/checkbox1"
        android:text="CheckBoxOne"
        android:layout_marginTop="16dp"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintEnd_toStartOf="parent"
        app:layout_constraintTop_toTopOf="parent" />

    <CheckBox
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:id="@+id/checkbox2"
        android:text="CheckBoxTwo"
        android:layout_marginTop="36dp"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintEnd_toStartOf="parent"
        app:layout_constraintTop_toTopOf="parent" />

    <CheckBox
```

```

        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:id="@+id/checkbox3"
        android:text="CheckBoxThree"
        android:layout_marginTop="60dp"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintEnd_toStartOf="parent"
        app:layout_constraintTop_toTopOf="parent" />

</androidx.constraintlayout.widget.ConstraintLayout>

```

### **MainActivity.java**

```

package com.example.welcome;

import androidx.appcompat.app.AppCompatActivity;

import android.os.Bundle;
import android.widget.CheckBox;
import android.widget.CompoundButton;
import android.widget.Toast;

public class CheckBoxLab_13 extends AppCompatActivity {
    CheckBox checkBox1,checkBox2,checkBox3;
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_check_box_lab13);
        checkBox1 = findViewById(R.id.checkbox1);
        checkBox2 = findViewById(R.id.checkbox2);
        checkBox3 = findViewById(R.id.checkbox3);
    }
}

```

```

checkBox1.setOnCheckedChangeListener(new CompoundButton.OnCheckedChangeListener() {
    @Override
    public void onCheckedChanged(CompoundButton buttonView, boolean isChecked) {
        if(isChecked){
            //code to execute when checkbox is checked

            Toast.makeText(getApplicationContext(),"The check box 1 is
checked",Toast.LENGTH_SHORT).show();
        }
        else {
            Toast.makeText(getApplicationContext(),"The check box 1 is not
checked",Toast.LENGTH_SHORT).show();
        }
    }
});

```

```

checkBox2.setOnCheckedChangeListener(new CompoundButton.OnCheckedChangeListener() {
    @Override
    public void onCheckedChanged(CompoundButton buttonView, boolean isChecked) {
        if(isChecked){
            //code to execute when checkbox is checked

            Toast.makeText(getApplicationContext(),"The check box 2 is
checked",Toast.LENGTH_SHORT).show();
        }
        else {
            Toast.makeText(getApplicationContext(),"The check box 2 is not
checked",Toast.LENGTH_SHORT).show();
        }
    }
});

```

```

checkBox3.setOnCheckedChangeListener(new CompoundButton.OnCheckedChangeListener() {
    @Override

```

```
public void onCheckedChanged(CompoundButton buttonView, boolean isChecked) {  
    if(isChecked){  
        //code to execute when checkbox is checked  
        Toast.makeText(getApplicationContext(),"The check box 3 is  
checked",Toast.LENGTH_SHORT).show();  
    }  
    else {  
        Toast.makeText(getApplicationContext(),"The check box 3 is not  
checked",Toast.LENGTH_SHORT).show();  
    }  
}  
});  
  
}  
}
```

## **LAB-14:- Create simple Application which shows the use of WebView.**

### **XML file**

```
<?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=".WebViewLab_14">

    <WebView

        android:id="@+id/webView"
        android:layout_width="match_parent"
        android:layout_height="match_parent"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toTopOf="parent"
    />

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

### **MainActivity.java**

```
package com.example.welcome;

import androidx.appcompat.app.AppCompatActivity;

import android.os.Bundle;
import android.webkit.WebSettings;
import android.webkit.WebView;

public class WebViewLab_14 extends AppCompatActivity {

    WebView webView;
```

@Override

```
protected void onCreate(Bundle savedInstanceState) {  
    super.onCreate(savedInstanceState);  
    setContentView(R.layout.activity_web_view_lab14);  
    webView = findViewById(R.id.webView);  
    // Enable JavaScript (optional, depending on your use case)  
    WebSettings webSettings = webView.getSettings();  
    webSettings.getJavaScriptEnabled();  
    //load url in the webview  
    webView.loadUrl("https://www.google.com");  
}  
}
```



### **LAB-15:- Create simple Application which shows the use of SeekBar.**

#### **XML file**

```
<?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=".SeekBarLab_15">

    <SeekBar

        android:id="@+id/seekBar"
        android:layout_width="200dp"
        android:layout_height="100dp"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintTop_toTopOf="parent"
        />

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

#### **MainActivity.java**

```
package com.example.welcome;

import androidx.appcompat.app.AppCompatActivity;

import android.os.Bundle;
import android.widget.SeekBar;
import android.widget.Toast;

public class SeekBarLab_15 extends AppCompatActivity {
```

```
SeekBar seekBar;

@Override

protected void onCreate(Bundle savedInstanceState) {

    super.onCreate(savedInstanceState);

    setContentView(R.layout.activity_seek_bar_lab15);

    seekBar = findViewById(R.id.seekBar);

    seekBar.setOnSeekBarChangeListener(new SeekBar.OnSeekBarChangeListener() {

        @Override

        public void onProgressChanged(SeekBar seekBar, int progress, boolean fromUser) {

            Toast.makeText(getApplicationContext(),progress,Toast.LENGTH_LONG).show();

        }


        @Override

        public void onStartTrackingTouch(SeekBar seekBar) {

            //do something when the user starts dragging the seekbar

        }


        @Override

        public void onStopTrackingTouch(SeekBar seekBar) {

            //do something when user stop dragging the seekbar

        }

    });

}

}
```

## LAB-16:- Create simple Application which shows the use of RatingBar.

### XML file:-

```
<?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=".RatingBarLab_16">

    <RatingBar

        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:id="@+id/ratingBar"

        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintTop_toTopOf="parent"

    />

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

### MainActivity.java

```
package com.example.welcome;

import androidx.appcompat.app.AppCompatActivity;

import android.os.Bundle;
import android.widget.RatingBar;
import android.widget.Toast;
```

```
public class RatingBarLab_16 extends AppCompatActivity {  
    RatingBar ratingBar;  
  
    @Override  
    protected void onCreate(Bundle savedInstanceState) {  
        super.onCreate(savedInstanceState);  
        setContentView(R.layout.activity_rating_bar_lab16);  
        ratingBar = findViewById(R.id.ratingBar);  
        ratingBar.setOnRatingBarChangeListener(new RatingBar.OnRatingBarChangeListener() {  
            @Override  
            public void onRatingChanged(RatingBar ratingBar, float rating, boolean fromUser) {  
                Toast.makeText(getApplicationContext(),"rating "+(int) rating,Toast.LENGTH_SHORT).show();  
            }  
        });  
    }  
}
```

## **LAB-17:- Create simple Application which shows the use of Alert Dialog.**

### **XML file**

```
<?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=".AlertDialogLab_17">

    <Button

        android:id="@+id/showAlertDialog"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Alert dialog display"
        android:textSize="30dp"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toTopOf="parent"

    />

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

### **MainActiviyt.java**

```
package com.example.welcome;

import androidx.appcompat.app.AlertDialog;
import androidx.appcompat.app.AppCompatActivity;
import android.content.DialogInterface;
import android.os.Bundle;
import android.view.View;
import android.widget.Button;
```

```

public class AlertDialogLab_17 extends AppCompatActivity {

    Button btnShowAlertDialog;

    @Override

    protected void onCreate(Bundle savedInstanceState) {

        super.onCreate(savedInstanceState);

        setContentView(R.layout.activity_alert_dialog_lab17);

        btnShowAlertDialog = findViewById(R.id.showAlertDialog);

        btnShowAlertDialog.setOnClickListener(new View.OnClickListener() {

            @Override

            public void onClick(View v) {

                showAlertDialog();

            }

        });

    }

    void showAlertDialog(){

        AlertDialog.Builder builder = new AlertDialog.Builder(this);

        builder.setTitle("Oops...").

            setMessage("Request fail: Unable to restart Host"+"\\n"+"please check your connection and
try again")

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

                @Override

                public void onClick(DialogInterface dialog, int which) {

                    //Handle the positive button clicked

                    dialog.dismiss();

                }

            });

        //create and show AlertDialog

        AlertDialog alertDialog =builder.create();

        alertDialog.show();

    }

}

```

## **LAB-18:- Create simple Application which shows the use of TimePicker.**

### **XML file**

```
<?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=".TimePickerLab_18">

    <TimePicker

        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:id="@+id/timePicker"
        app:layout_constraintTop_toTopOf="parent"
        app:layout_constraintStart_toStartOf="parent"
        />

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

### **MainActivity.java**

```
package com.example.welcome;

import androidx.appcompat.app.AppCompatActivity;

import android.os.Bundle;
import android.widget.TimePicker;
import android.widget.Toast;

public class TimePickerLab_18 extends AppCompatActivity {

    TimePicker timePicker;

    @Override
```

```

protected void onCreate(Bundle savedInstanceState) {
    super.onCreate(savedInstanceState);
    setContentView(R.layout.activity_time_picker_lab18);
    timePicker = findViewById(R.id.timePicker);
    //if want to change the time when user select the specific time formate
    timePicker.setOnTimeChangedListener(new TimePicker.OnTimeChangedListener() {
        @Override
        public void onTimeChanged(TimePicker view, int hourOfDay, int minute) {
            //handle the time change
            Toast.makeText(getApplicationContext(),"Hours "+hourOfDay+"minute
"+minute,Toast.LENGTH_LONG).show();
        }
    });
}
}
}
//run the program and try to change the minute hand, you will see the time is change on up

```



## LAB-19:- Create simple Application which shows the use of DatePicker.

### XML file

```
<?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=".DatePickerLab_19">

    <DatePicker

        android:id="@+id/datePicker"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toTopOf="parent" />

    <Button

        android:id="@+id/btn"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_marginEnd="93dp"
        android:layout_marginBottom="164dp"
        android:text="Display selected date"
        app:layout_constraintBottom_toBottomOf="parent"
        app:layout_constraintEnd_toEndOf="parent" />
</androidx.constraintlayout.widget.ConstraintLayout>
```

### MainActivity.java

```
package com.example.welcome;
```

```
import androidx.appcompat.app.AppCompatActivity;
```

```
import android.os.Bundle;
```

```
import android.view.View;
```

```
import android.widget.Button;
```

```
import android.widget.DatePicker;
```

```
import android.widget.Toast;
```

```
public class DatePickerLab_19 extends AppCompatActivity {
```

```
    DatePicker datePicker;
```

```
    Button btn;
```

```
    @Override
```

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

```
        super.onCreate(savedInstanceState);
```

```
        setContentView(R.layout.activity_date_picker_lab19);
```

```
        datePicker = findViewById(R.id.datePicker);
```

```
        btn = findViewById(R.id.btn);
```

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

```
            @Override
```

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

```
                String day = "day"+datePicker.getDayOfMonth();
```

```
                String month = "month"+datePicker.getMonth()+1;
```

```
                String year = "year"+ datePicker.getYear();
```

```
                Toast.makeText(getApplicationContext(),day + "\n"+ month+ "\n"
+year,Toast.LENGTH_LONG).show();
```

```
            }
```

```
        });
```

```
    }
```

```
}
```

//Note- if you get an error while displaying the months, it is because datePicker started as zero in january.

//so in line no.23 you write +1.(String month = "month"+datePicker.getMonth()+1;)

## **LAB-20:- Create simple Application which shows the use of ImageView.**

### **XML file**

```
<?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=".ImageViewLab_20">

    <ImageView

        android:id="@+id/imageView"
        android:layout_width="0dp"
        android:layout_height="0dp"
        app:layout_constraintBottom_toBottomOf="parent"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toTopOf="parent" />

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

### **MainActivity.java**

```
package com.example.welcome;

import androidx.appcompat.app.AppCompatActivity;
import android.os.Bundle;
import android.widget.ImageView;

public class ImageViewLab_20 extends AppCompatActivity {

    ImageView imageView;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_image_view_lab20);
        imageView =findViewById(R.id.imageView);
    }
}
```

```
//set an image resources programmatically  
imageView.setImageResource(R.drawable.ic_launcher_background);  
//you can put your image name in imageView.setImageResource(R.drawable.your-any-image);  
}  
}
```

**LAB-21:- Create application in which take two button start and stop service. When you press start**

**button one service should be started and when you press stop then service should be destroy.**

This project(Music Player app) will demonstrate the use of services in android. It has two button. Start and stop button. When you click on start button, automatically the default ringtone for mobile will play and if you click on button stop, it will stop the default playing ringtone.

You can see the UI for Apps

- 1)** Create a java class name as MyCustomService in package(create this class where your activity file is available)

```
package com.mastercoding.musicplayerapp;

import android.app.Service;
import android.content.Intent;
import android.media.MediaPlayer;
import android.os.IBinder;
import android.provider.Settings;
import androidx.annotation.Nullable;

//this is lab no 21

public class MyCustomService extends Service {

    // To Play music, we need a media player object
    private MediaPlayer player;

    @Override
    public int onStartCommand(Intent intent, int flags, int startId) {

        // this will play the audio of default ringtone in the device
        player = MediaPlayer.create(
            this,
            Settings.System.DEFAULT_RINGTONE_URI
        );

        // play the ringtone audio on loop (continuously)
        player.setLooping(true);
        player.start();

        return START_STICKY;
    }

    @Override
    public void onDestroy() {
        super.onDestroy();
        player.stop();
    }

    @Nullable
    @Override
    public IBinder onBind(Intent intent) {
        return null;
    }
}
```

## 2) Write bellow code in MainActivity.java

```
package com.mastercoding.musicplayerapp;

import androidx.appcompat.app.AppCompatActivity;
import android.content.Intent;
import android.os.Bundle;
import android.view.View;
import android.widget.Button;
import android.widget.TextView;

public class MainActivity extends AppCompatActivity {

    TextView txt;
    Button start_btn, stop_btn;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);
        start_btn = findViewById(R.id.start_btn);
        stop_btn = findViewById(R.id.stop_btn);
        txt = findViewById(R.id.textView);
        start_btn.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                Intent serviceIntent = new Intent(getApplicationContext(),
                    MyCustomService.class);
                startService(serviceIntent);
            }
        });
        stop_btn.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                Intent serviceIntent = new Intent(getApplicationContext(),
                    MyCustomService.class);
                stopService(serviceIntent); }
        });
    }
}
```

```
<?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"
    android:background="@drawable/back"
    tools:context=".MainActivity">

    <TextView
        android:id="@+id/textView"
        android:layout_width="0dp"
        android:layout_height="wrap_content"
        android:layout_marginTop="28dp"
        android:gravity="center"
        android:text="The Services App"
        android:textColor="@color/white"
        android:textSize="32sp"
        android:textStyle="bold"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintHorizontal_bias="0.0"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toTopOf="parent" />

    <Button
        android:id="@+id/start_btn"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_marginBottom="114dp"
        android:text="Start Service"
        app:layout_constraintBottom_toBottomOf="parent"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintStart_toStartOf="parent" />
```

```
<Button
    android:id="@+id/stop_btn"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_marginBottom="36dp"
    android:text="Stop Service"
    app:layout_constraintBottom_toBottomOf="parent"
    app:layout_constraintEnd_toEndOf="@+id/start_btn" />
</androidx.constraintlayout.widget.ConstraintLayout>
```

OutPut Screen



## The Services App



Start Service

Stop Service

**LAB-22:- Create application to create "school.db" database and create table with name "student" which contain columns (id, name, surname, mark) using SQLite database.**

**Ans-** In this example, we are not creating the UI. We will display the stored data in Log.so that why we will only write the java code.

DatabaseHelper.java

```
package com.example.welcome;
```

```
import android.content.ContentValues;
```

```
import android.content.Context;
```

```
import android.database.Cursor;
```

```
import android.database.sqlite.SQLiteDatabase;
```

```
import android.database.sqlite.SQLiteOpenHelper;
```

```
import androidx.annotation.Nullable;
```

```
public class DatabaseHelper extends SQLiteOpenHelper {
```

```
    String Database_Name="paruldatabase";
```

```
    int Database_version=1;
```

```
    //Table and column name
```

```
    String Table_name="student";
```

```
    String id="_id";
```

```
    String column_name="name";
```

```
    String column_mark ="mark";
```

```
    public DatabaseHelper(@Nullable Context context, @Nullable String name, @Nullable  
SQLiteDatabase.CursorFactory factory, int version) {
```

```
        super(context, name, factory, version);
```

```
    }
```

```
    @Override
```

```
    public void onCreate(SQLiteDatabase db) {
```

```
        // Create your table
```

```
        String createTableQuery = "CREATE TABLE " + Table_name + "(" +
```

```
        id + " INTEGER PRIMARY KEY AUTOINCREMENT, " + column_name + " TEXT, " +  
column_mark + " INTEGER);
```

```
        db.execSQL(createTableQuery);
```

```
    }
```

```
    @Override
```

```
    public void onUpgrade(SQLiteDatabase db, int oldVersion, int newVersion) {
```

```
        //we will write when we require some logic
```

```
    }
```

```
}
```

MainActivity.java

```
import android.database.Cursor;
```

```
import android.os.Bundle;
```

```
import android.util.Log;
```

```
import android.view.View;
```

```
public class MainActivity extends AppCompatActivity {
```

```
    @Override
```

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

```
        super.onCreate(savedInstanceState);
```

```
        setContentView(R.layout.activity_main);
```

```
        // Create an instance of your DBHelper
```

```
        DatabaseHelper dbHelper = new DatabaseHelper(this, "paruldatabase", null, 1);
```

```
    }
```

```
}
```

**LAB-23:- Create application to insert data into table "student" using SQLite database.**

**DatabaseHelper.java**

```
package com.example.welcome;
```

```
import android.content.ContentValues;
```

```
import android.content.Context;
```

```
import android.database.Cursor;
```

```

import android.database.sqlite.SQLiteDatabase;
import android.database.sqlite.SQLiteOpenHelper;

import androidx.annotation.Nullable;

public class DatabaseHelper extends SQLiteOpenHelper {

    String Database_Name="paruldatabase";

    int Database_version=1;

    //Table and column name
    String Table_name="student";

    String id="_id";

    String column_name="name";

    String column_mark ="mark";

    public DatabaseHelper(@Nullable Context context, @Nullable String name, @Nullable
    SQLiteDatabase.CursorFactory factory, int version) {

        super(context, name, factory, version);
    }

    @Override
    public void onCreate(SQLiteDatabase db) {

        // Create your table

        String createTableQuery = "CREATE TABLE " + Table_name + "(" +

        id + " INTEGER PRIMARY KEY AUTOINCREMENT, " + column_name + " TEXT, " + column_mark + "
INTEGER)";

        db.execSQL(createTableQuery);

    }

    @Override
    public void onUpgrade(SQLiteDatabase db, int oldVersion, int newVersion) {

        //we will write when we require some logic

    }

    //create our own method to add the data in to database
    void addData(String name, int age){

        SQLiteDatabase db =this.getWritableDatabase();

        ContentValues values =new ContentValues();

```

```

        values.put(column_name,name);
        values.put(column_mark,age);

        //insert the new row
        db.insert(Table_name,null,values);
        db.close();
    }

    // Create operation
    public long insertData(String name, int mark) {
        SQLiteDatabase db = this.getWritableDatabase();
        ContentValues values = new ContentValues();
        values.put(column_name, name);
        values.put(column_mark, mark);
        long newRowId = db.insert(Table_name, null, values);
        db.close();
        return newRowId;
    }

```

MainActivity.java

```

import android.database.Cursor;
import android.os.Bundle;
import android.util.Log;
import android.view.View;

public class MainActivity extends AppCompatActivity {

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);

        // Create an instance of your DBHelper

```

```

        DatabaseHelper dbHelper = new DatabaseHelper(this, "paruldatabase", null, 1);

        Add data to the database

        // dbHelper.addData("sam wibo", 25);
        //dbHelper.addData("John gibo", 26);
        //Log.d("MainActivity", "Newly inserted ID: ");
        long newId = dbHelper.insertData("Ajay", 25);
        Log.d("MainActivity", "Newly inserted ID: " + newId);
    }

    cursor.close();
}

}
}
}

```

**LAB-24:- . Create application which shows inserted data in table student in alert dialog box using SQLite database.**

#### **XML file**

```

<?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">

    <Button

        android:id="@+id/showAlertDialog"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Alert dialog display"
        android:textSize="30dp"
    >

```

```

        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toTopOf="parent"

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

DatabaseHelper.java

package com.example.lab_24_insertdatawithalrtdialog;

import android.content.ContentValues;
import android.content.Context;
import android.database.sqlite.SQLiteDatabase;
import android.database.sqlite.SQLiteOpenHelper;
import androidx.annotation.Nullable;

public class DatabaseHelper extends SQLiteOpenHelper {

    String Database_Name="paruldatabase";

    int Database_version=1;

    //Table and column name
    String Table_name="student";

    String id="_id";

    String column_name="name";

    String column_mark ="mark";


    public DatabaseHelper(@Nullable Context context, @Nullable String name, @Nullable
    SQLiteDatabase.CursorFactory factory, int version) {

        super(context, name, factory, version);
    }

    @Override

    public void onCreate(SQLiteDatabase db) {

        // Create your table

        String createTableQuery = "CREATE TABLE " + Table_name + "(" +

            id + " INTEGER PRIMARY KEY AUTOINCREMENT, " + column_name + " TEXT, " +
            column_mark + " INTEGER)";

        db.execSQL(createTableQuery);
    }

```

```

    }

    @Override

    public void onUpgrade(SQLiteDatabase db, int oldVersion, int newVersion) {

        //we will write when we require some logic
    }

    //create our own method to add the data in to database
    void addData(String name, int mark){

        SQLiteDatabase db =this.getWritableDatabase();

        ContentValues values =new ContentValues();

        values.put(column_name,name);

        values.put(column_mark,mark);


        //insert the new row

        db.insert(Table_name,null,values);

        db.close();

    }

}

```

### **MainActivity.java**

```

package com.example.lab_24_insertdatawithalertdialog;


import androidx.appcompat.app.AppCompatActivity;

import androidx.appcompat.app.AlertDialog;


import android.content.DialogInterface;

import android.os.Bundle;

import android.util.Log;

import android.view.View;

import android.widget.Button;public class MainActivity extends AppCompatActivity {

    Button btnShowAlertDialog;

```



```

@Override

protected void onCreate(Bundle savedInstanceState) {

    super.onCreate(savedInstanceState);

    setContentView(R.layout.activity_main);

    DatabaseHelper helper = new DatabaseHelper(this, "paruldatabase", null, 1);

    btnShowAlertDialog = findViewById(R.id.showAlertDiaglog);

    btnShowAlertDialog.setOnClickListener(new View.OnClickListener() {

        @Override

        public void onClick(View v) {

            // Add data to the database

            helper.addData("sam wibo", 25);

            helper.addData("John gibo", 26);

            // Log.d("MainActivity", "Newly inserted ID: ")

            AlertDialog.Builder builder = new AlertDialog.Builder(getApplicationContext());

            builder.setTitle("Data is inserted...").

                setMessage("Response pass:Data is inserted successfully in database")

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

                    @Override

                    public void onClick(DialogInterface dialog, int which) {

                        //Handle the positive button clicked

                        dialog.dismiss();

                    }

                });

            //create and show AlertDialog

            AlertDialog alertDialog = builder.create();

            alertDialog.show();

        }

    });

}

```

**LAB-25:- Create application which updates data in table student using SQLite database.**

**DatabaseHelper.java**

```
package com.example.welcome;

import android.content.ContentValues;
import android.content.Context;
import android.database.Cursor;
import android.database.sqlite.SQLiteDatabase;
import android.database.sqlite.SQLiteOpenHelper;
import androidx.annotation.Nullable;

public class DatabaseHelper extends SQLiteOpenHelper {

    String Database_Name="paruldatabase";

    int Database_version=1;

    //Table and column name

    String Table_name="student";

    String id="_id";

    String column_name="name";

    String column_mark ="mark";

    public DatabaseHelper(@Nullable Context context, @Nullable String name, @Nullable
    SQLiteDatabase.CursorFactory factory, int version) {

        super(context, name, factory, version);
    }

    @Override

    public void onCreate(SQLiteDatabase db) {

        // Create your table

        String createTableQuery = "CREATE TABLE " + Table_name + "(" +

        id + " INTEGER PRIMARY KEY AUTOINCREMENT, " + column_name + " TEXT, " + column_mark + "

        INTEGER)";

        db.execSQL(createTableQuery);
    }

    @Override

    public void onUpgrade(SQLiteDatabase db, int oldVersion, int newVersion) {

        //we will write when we require some logic
    }
}
```

```

}

//create our own method to add the data in to database
void addData(String name, int age){
    SQLiteDatabase db =this.getWritableDatabase();

    ContentValues values =new ContentValues();

    values.put(column_name,name);
    values.put(column_mark,age);

    //insert the new row
    db.insert(Table_name,null,values);
    db.close();
}

// Create operation
public long insertData(String name, int mark) {
    SQLiteDatabase db = this.getWritableDatabase();
    ContentValues values = new ContentValues();
    values.put(column_name, name);
    values.put(column_mark, mark);
    long newRowId = db.insert(Table_name, null, values);
    db.close();
    return newRowId;
}

// Read operation
public Cursor getAllData() {
    SQLiteDatabase db = this.getReadableDatabase();
    return db.query(Table_name, null, null, null, null, null, null);
}

// Update operation
public int updateData(long id, String newName, int newMark) {

```

```

        SQLiteDatabase db = this.getWritableDatabase();

        ContentValues values = new ContentValues();

        values.put(column_name, newName);

        values.put(column_mark, newMark);

        return db.update(Table_name, values, id + "=?", new String[]{String.valueOf(id)});

    }

}

```

### **MainActivity.java**

```

package com.example.welcome;

import androidx.appcompat.app.AppCompatActivity;

import android.database.Cursor;
import android.os.Bundle;
import android.util.Log;
import android.view.View;
import android.widget.Button;
import android.widget.EditText;
import android.widget.TextView;

public class MainActivity extends AppCompatActivity {

    // EditText editText;

    // Button btn;

    // TextView title;

    @Override
    protected void onCreate(Bundle savedInstanceState) {

        super.onCreate(savedInstanceState);

        setContentView(R.layout.activity_main);

        // Create an instance of your DBHelper
    }
}

```

```
DatabaseHelper dbhelper = new DatabaseHelper(this, "paruldatabase", null, 1);
```

```
// Add data to the database
```

```
// dbhelper.addData("sam wibo", 25);
```

```
//dbhelper.addData("John gibo", 26);
```

```
Log.d("MainActivity", "Newly inserted ID: ");
```

```
long newId = dbhelper.insertData("Ajay", 25);
```

```
Log.d("MainActivity", "Newly inserted ID: " + newId);
```

```
/* // Get all data operation
```

```
Cursor cursor = dbhelper.getAllData();
```

```
if (cursor != null) {
```

```
    while (cursor.moveToNext()) {
```

```
        long id = cursor.getLong(cursor.getColumnIndexOrThrow(dbhelper.id));
```

```
        String name = cursor.getString(cursor.getColumnIndexOrThrow(dbhelper.column_name));
```

```
        int age = cursor.getInt(cursor.getColumnIndexOrThrow(dbhelper.column_age));
```

```
        Log.d("MainActivity", "ID: " + id + ", Name: " + name + ", Age: " + age);
```

```
    }
```

```
    cursor.close();
```

```
}
```

```
// Update operation
```

```
int rowsAffected = dbhelper.updateData(newId, "Updated Name", 30);
```

```
Log.d("MainActivity", "Rows updated: " + rowsAffected);
```

```
// Get all data after update
```

```
cursor = dbhelper.getAllData();
```

```
if (cursor != null) {
```

```
    while (cursor.moveToNext()) {
```

```

        long id = cursor.getLong(cursor.getColumnIndexOrThrow(dbhelper.id));
        String name = cursor.getString(cursor.getColumnIndexOrThrow(dbhelper.column_name));
        int age = cursor.getInt(cursor.getColumnIndexOrThrow(dbhelper.column_age));

        Log.d("MainActivity", "ID: " + id + ", Name: " + name + ", Age: " + age);
    }
    cursor.close();
}

// Delete operation
int rowsDeleted = dbhelper.deleteData(newId);
Log.d("MainActivity", "Rows deleted: " + rowsDeleted);

// Get all data after delete
cursor = dbhelper.getAllData();
if (cursor != null) {
    while (cursor.moveToNext()) {
        long id = cursor.getLong(cursor.getColumnIndexOrThrow(dbhelper.id));
        String name = cursor.getString(cursor.getColumnIndexOrThrow(dbhelper.column_name));
        int age = cursor.getInt(cursor.getColumnIndexOrThrow(dbhelper.column_age));

        Log.d("MainActivity", "ID: " + id + ", Name: " + name + ", Age: " + age);
    }
    cursor.close();
}
dbhelper.close();*/
}
}

```

**LAB-26:- Create application which deletes record from table student using SQLite database.**

## **DatabaseHelper.java**

```
package com.example.welcome;

import android.content.ContentValues;
import android.content.Context;
import android.database.Cursor;
import android.database.sqlite.SQLiteDatabase;
import android.database.sqlite.SQLiteOpenHelper;

import androidx.annotation.Nullable;

public class DatabaseHelper extends SQLiteOpenHelper {

    String Database_Name="paruldatabase";

    int Database_version=1;

    //Table and column name

    String Table_name="student";

    String id="_id";

    String column_name="name";

    String column_mark ="mark";

    public DatabaseHelper(@Nullable Context context, @Nullable String name, @Nullable
SQLiteDatabase.CursorFactory factory, int version) {

        super(context, name, factory, version);
    }

    @Override

    public void onCreate(SQLiteDatabase db) {

        // Create your table

        String createTableQuery = "CREATE TABLE " + Table_name + "(" +

            id + " INTEGER PRIMARY KEY AUTOINCREMENT, " + column_name + " TEXT, " +
column_mark + " INTEGER)";

        db.execSQL(createTableQuery);
    }

    @Override
```

```

public void onUpgrade(SQLiteDatabase db, int oldVersion, int newVersion) {
    //we will write when we require some logic
}

//create our own method to add the data in to database
void addData(String name, int age){
    SQLiteDatabase db =this.getWritableDatabase();
    ContentValues values =new ContentValues();
    values.put(column_name,name);
    values.put(column_mark,age);

    //insert the new row
    db.insert(Table_name,null,values);
    db.close();
}

// Create operation
public long insertData(String name, int mark) {
    SQLiteDatabase db = this.getWritableDatabase();
    ContentValues values = new ContentValues();
    values.put(column_name, name);
    values.put(column_mark, mark);
    long newRowId = db.insert(Table_name, null, values);
    db.close();
    return newRowId;
}

// Read operation
public Cursor getAllData() {
    SQLiteDatabase db = this.getReadableDatabase();
    return db.query(Table_name, null, null, null, null, null, null);
}

```



```
}
```

```
// Update operation
```

```
public int updateData(long id, String newName, int newMark) {  
    SQLiteDatabase db = this.getWritableDatabase();  
    ContentValues values = new ContentValues();  
    values.put(column_name, newName);  
    values.put(column_mark, newMark);  
    return db.update(Table_name, values, id + "=?", new String[]{String.valueOf(id)});  
}
```

```
// Delete operation
```

```
public int deleteData(long id) {  
    SQLiteDatabase db = this.getWritableDatabase();  
    return db.delete(Table_name, id + "=?", new String[]{String.valueOf(id)});  
}  
}
```

### **MainActivity.java**

```
package com.example.welcome;  
  
import androidx.appcompat.app.AppCompatActivity;  
  
import android.database.Cursor;  
import android.os.Bundle;  
import android.util.Log;  
import android.view.View;  
import android.widget.Button;  
import android.widget.EditText;  
import android.widget.TextView;
```

```

public class MainActivity extends AppCompatActivity {

    // EditText editText;

    // Button btn;

    // TextView title;


    @Override

    protected void onCreate(Bundle savedInstanceState) {

        super.onCreate(savedInstanceState);

        setContentView(R.layout.activity_main);

        // Create an instance of your DBHelper

        DatabaseHelper dbHelper = new DatabaseHelper(this, "paruldatabase", null, 1);


        // Add data to the database

        // dbHelper.addData("sam wibo", 25);

        //dbHelper.addData("John gibo", 26);

        Log.d("MainActivity", "Newly inserted ID: ");


        long newId = dbHelper.insertData("Ajay", 25);

        Log.d("MainActivity", "Newly inserted ID: " + newId);


        /* // Get all data operation

        Cursor cursor = dbHelper.getAllData();

        if (cursor != null) {

            while (cursor.moveToNext()) {

                long id = cursor.getLong(cursor.getColumnIndexOrThrow(dbHelper.id));

                String name =

cursor.getString(cursor.getColumnIndexOrThrow(dbHelper.column_name));

                int age = cursor.getInt(cursor.getColumnIndexOrThrow(dbHelper.column_age));

```

```

        Log.d("MainActivity", "ID: " + id + ", Name: " + name + ", Age: " + age);
    }
    cursor.close();
}

// Update operation
int rowsAffected = dbHelper.updateData(newId, "Updated Name", 30);
Log.d("MainActivity", "Rows updated: " + rowsAffected);

// Get all data after update
cursor = dbHelper.getAllData();
if (cursor != null) {
    while (cursor.moveToNext()) {
        long id = cursor.getLong(cursor.getColumnIndexOrThrow(dbHelper.id));
        String name =
cursor.getString(cursor.getColumnIndexOrThrow(dbHelper.column_name));
        int age = cursor.getInt(cursor.getColumnIndexOrThrow(dbHelper.column_age));

        Log.d("MainActivity", "ID: " + id + ", Name: " + name + ", Age: " + age);
    }
    cursor.close();
}

// Delete operation
int rowsDeleted = dbHelper.deleteData(newId);
Log.d("MainActivity", "Rows deleted: " + rowsDeleted);

// Get all data after delete
cursor = dbHelper.getAllData();
if (cursor != null) {

```

```
while (cursor.moveToNext()) {  
    long id = cursor.getLong(cursor.getColumnIndexOrThrow(dbhelper.id));  
    String name =  
cursor.getString(cursor.getColumnIndexOrThrow(dbhelper.column_name));  
    int age = cursor.getInt(cursor.getColumnIndexOrThrow(dbhelper.column_age));  
  
    Log.d("MainActivity", "ID: " + id + ", Name: " + name + ", Age: " + age);  
}  
cursor.close();  
}  
dbhelper.close();*/  
}  
}
```