



Royal University of Bhutan

ROYAL UNIVERSITY OF BHUTAN
COLLEGE OF SCIENCE AND TECHNOLOGY
PHUENTSHOLING: BHUTAN



PLAGIARISM DECLARATION FORM

Student Name: Divyash Chhetri

Student No: 02200174

Module No and Title of the module: CTE306 - Mobile Application Development

Assignment no and Title of the Assignment: Lab Work 02

Section H2 of the Royal University of Bhutan's *Wheel of Academic Law* provides the following definition of academic dishonesty:

"Academic dishonesty may be defined as any attempt by a student to gain an unfair advantage in any assessment. It may be demonstrated by one of the following:

Collusion: the representation of a piece of unauthorized group work as the work of a single candidate.

Commissioning: submitting an assignment done by another person as the student's own work.

Duplication: the inclusion in coursework of material identical or substantially similar to material which has already been submitted for any other assessment within the University.

False declaration: making a false declaration in order to receive special consideration by an Examination Board or to obtain extensions to deadlines or exemption from work.

Falsification of data: presentation of data in laboratory reports, projects, etc., based on work purported to have been carried out by the student, which have been invented, altered or copied by the student.

Plagiarism: the unacknowledged use of another's work as if it were one's own.

Examples are:

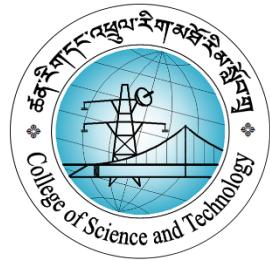
- verbatim copying of another's work without acknowledgement
- paraphrasing of another's work by simply changing a few words or altering the order of presentation, without acknowledgement
- ideas or intellectual data in any form presented as one's own without acknowledging the source(s)
- making significant use of unattributed digital images such as graphs, tables, photographs, etc. taken from test books, articles, films, plays, handouts, internet, or any other source, whether published or unpublished
- submission of a piece of work which has previously been assessed for a different award or module or at a different institution as if it were new work
- use of any material without prior permission of copyright from appropriate authority or owner of the materials used"

Student Declaration

I confirm that I have read and understood the above definitions of academic dishonesty. I declare that I have not committed any academic dishonesty when completing the attached piece of work.

Signature of Student:

Date: 09 Aug 2022



Lab 02

CTE306 – Mobile Application Development

Date – 9th August 2020

Divyash Chhetri
02200174
BE 3 IT

Module Tutor: Mr. Pema Galey

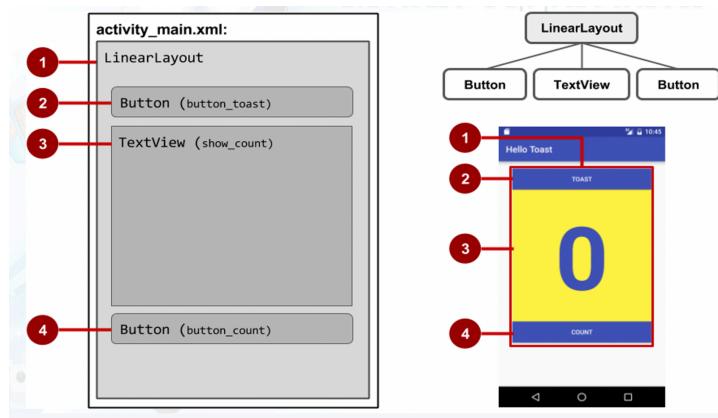
Department of Information Technology
College of Science and Technology

Aim

Task 1

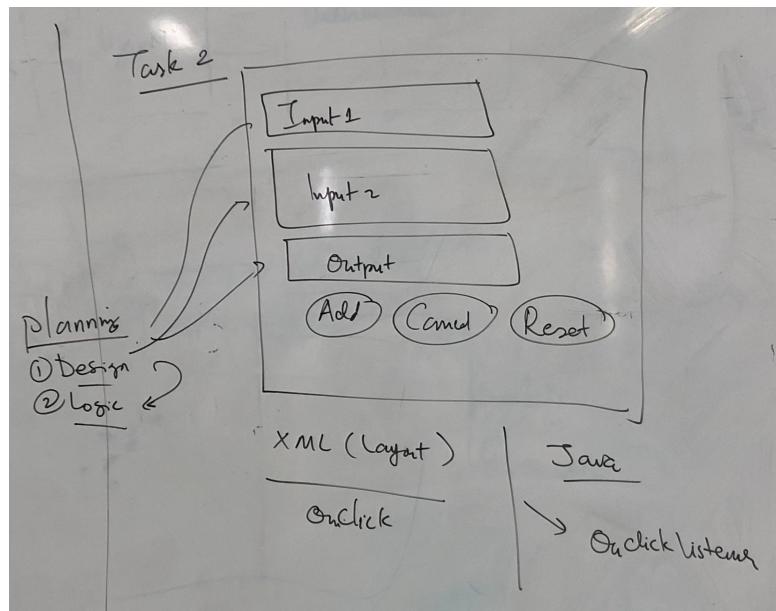
Design the app for following Template:

1. First Button show the Toast message
2. TextView to display the count number
3. Second Button to increase the count number



Task 2

Design a simple Calculator



Tutorial Link - <https://www.youtube.com/watch?v=W-2fOVxYPSA>

Theory

AndroidManifest.xml and activity_main.xml files can be edited to make a simple application with two buttons and a textView. Toast Button when clicked will display a message and the other button is the count button to keep track of the number of clicks the user made. The count shall be displayed in textView right above.

LinearLayout can be defined as a non-complex layout that is utilized in android to design layouts. In this sort of layout, all the elements present will be displayed in a linear manner. Linear manner meaning that the child of the parents will be displayed based on it's orientation. There can be two values for the orientation; vertical and horizontal. As mentioned in the name, these orientations are utilized to arrange one child after another, which can be in a vertical or horizontal manner.

Program Code

Task 1

MainActivity.java

```
package com.example.lab02task01;

import androidx.appcompat.app.AppCompatActivity;
import android.annotation.SuppressLint;
import android.os.Bundle;
import android.view.View;
import android.widget.TextView;
import android.widget.Toast;

public class MainActivity extends AppCompatActivity {

    int numberOfClicks = 0;
    TextView count_number;

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

```

public void toast(View view){
    String text = "Let's get counting!";
    Toast toast= Toast.makeText(this, text, Toast.LENGTH_LONG);
    toast.show();
}

@SuppressLint("SetTextI18n")
public void increment(View view) {
    numberOfClicks++;
    count_number.setText(Integer.toString(numberOfClicks));
}
}

```

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">

    <LinearLayout
        android:layout_width="409dp"
        android:layout_height="729dp"
        android:layout_marginStart="30dp"
        android:layout_marginTop="200dp"
        android:layout_marginEnd="30dp"
        android:orientation="vertical"
        app:layout_constraintBottom_toBottomOf="parent"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toTopOf="parent">

        <Button
            android:id="@+id/toastButton"
            android:layout_width="match_parent"
            android:layout_height="65dp"
            android:backgroundTint="#50586C"
            android:onClick="toast"
            android:text="TOAST"
            android:textColor="#DCE2F0"
            android:textSize="25sp"

```

```

        tools:ignore="HardcodedText" />

    <TextView
        android:id="@+id/countView"
        android:layout_width="match_parent"
        android:layout_height="254dp"
        android:background="#DCE2F0"
        android:gravity="center_vertical|center_horizontal"
        android:text="0"
        android:textAlignment="center"
        android:textColor="#50586C"
        android:textSize="150sp"
        android:textStyle="bold"
        tools:ignore="HardcodedText" />

    <Button
        android:id="@+id/countButton"
        android:layout_width="match_parent"
        android:layout_height="65dp"
        android:backgroundTint="#50586C"
        android:onClick="increment"
        android:text="COUNT"
        android:textColor="#DCE2F0"
        android:textSize="25sp"
        tools:ignore="HardcodedText" />
    </LinearLayout>
</androidx.constraintlayout.widget.ConstraintLayout>
```

Task 2

MainActivity.java

```

package com.example.lab02_task02;

import androidx.appcompat.app.AppCompatActivity;
import android.os.Bundle;
import android.view.View;
import android.widget.Button;
import android.widget.EditText;
import android.widget.Toast;

public class MainActivity extends AppCompatActivity {

    EditText inputA, inputB, output;
    Button addB, resetB, cancelB;
    Float sum;
```

```

@Override
protected void onCreate(Bundle savedInstanceState) {
    super.onCreate(savedInstanceState);
    setContentView(R.layout.activity_main);
    inputA = (EditText) findViewById(R.id.inputA);
    inputB = (EditText) findViewById(R.id.inputB);
    output = (EditText) findViewById(R.id.output);
    addB = (Button) findViewById(R.id.addButton);
    resetB = (Button) findViewById(R.id.resetButton);
    cancelB = (Button) findViewById(R.id.cancelButton);
}

public void add(View view) {
    sum = Float.valueOf(inputA.getText().toString()) +
Float.valueOf(inputB.getText().toString());
    Toast.makeText(this, "Adding...", Toast.LENGTH_SHORT).show();
    output.setText(String.valueOf(sum));
}

public void reset(View view) {
    Toast.makeText(this, "Resetting...", Toast.LENGTH_SHORT).show();
    inputA.setText("");
    inputA.setHint("Input A");
    inputB.setText("");
    inputB.setHint("Input A");
    output.setText("");
    output.setHint("0.0");
}

public void cancel(View view) {
    Toast.makeText(this, "Cancelled", Toast.LENGTH_SHORT).show();
    inputA.setText("");
    inputA.setHint("Input A");
    inputB.setText("");
    inputB.setHint("Input A");
}
}

```

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">

    <LinearLayout
        android:layout_width="match_parent"
        android:layout_height="match_parent"
        android:layout_marginStart="30dp"
        android:layout_marginTop="100dp"
        android:layout_marginEnd="30dp"
        android:orientation="vertical"
        app:layout_constraintBottom_toBottomOf="parent"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toTopOf="parent">

        <EditText
            android:id="@+id/inputA"
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:ems="10"
            android:hint="Input 1"
            android:inputType="textPersonName"
            android:textAlignment="textStart"
            android:textSize="24sp" />

        <EditText
            android:id="@+id/inputB"
            android:layout_width="match_parent"
            android:layout_height="83dp"
            android:ems="10"
            android:hint="Input 2"
            android:inputType="textPersonName"
            android:textAlignment="textStart"
            android:textSize="24sp" />

        <EditText
            android:id="@+id/output"
            android:layout_width="match_parent"
```

```
        android:layout_height="wrap_content"
        android:ems="10"
        android:hint="0.0"
        android:inputType="textPersonName"
        android:textAlignment="textEnd"
        android:textSize="34sp" />

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

        <Button
            android:id="@+id/addButton"
            style="?android:attr/buttonBarButtonStyle"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:layout_weight="1"
            android:backgroundTint="#3F51B5"
            android:onClick="add"
            android:text="ADD"
            android:textColor="#FFFFFF"
            android:textSize="24sp" />

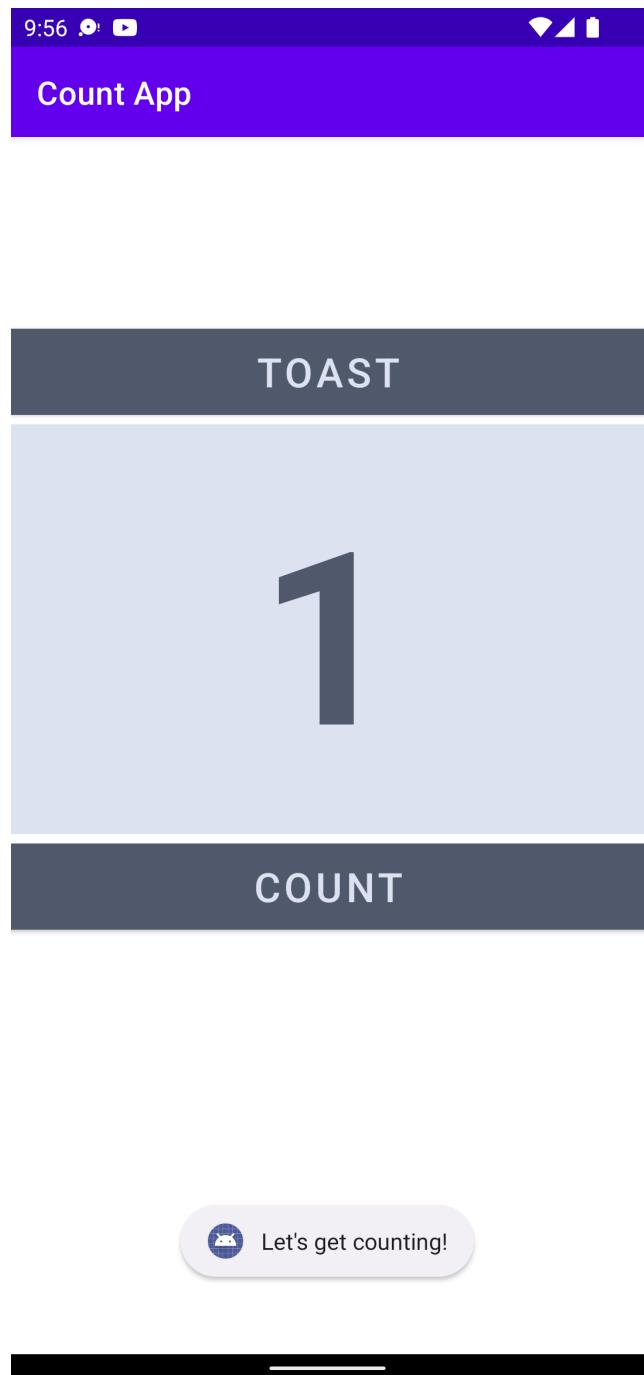
        <Button
            android:id="@+id/resetButton"
            style="?android:attr/buttonBarButtonStyle"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:layout_weight="1"
            android:backgroundTint="#3F51B5"
            android:onClick="reset"
            android:text="Reset"
            android:textColor="#FFFFFF"
            android:textSize="24sp" />

        <Button
            android:id="@+id cancelButton"
            style="?android:attr/buttonBarButtonStyle"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:layout_weight="1"
            android:backgroundTint="#3F51B5"
            android:onClick="cancel"
            android:text="CANCEL"
            android:textColor="#FFFFFF"
```

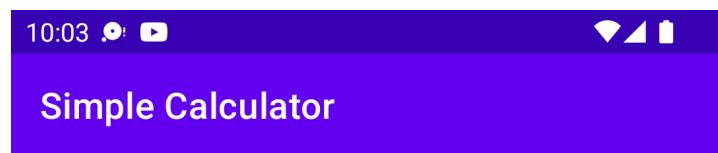
```
        android:textSize="24sp" />
    </LinearLayout>
</LinearLayout>
</androidx.constraintlayout.widget.ConstraintLayout>
```

Output

Task 1



Task 2



2

3

5.0

ADD RESET CANCEL



Adding...

Conclusion

An orientation property called Linear Layout can decide how to align or arrange the children in a single line. Depending on the value entered by the user, the single line alignment might be either vertical or horizontal. The task gave us the opportunity to explore new methods to use buttons. Two buttons were utilized in this practical. The Toast button was the first one. This is a basic button that, when pressed, shows a message or alert without interfering with the operation of an application. The Count button was the other button we created. The button's function was to utilize a textView to keep track of how many times a user had clicked on it. We also learned about how to use xml and its purpose.

References

Android. (n.d.). *Android Studio Docs*. Retrieved 08 09, 2022, from Developers Android:
<https://developer.android.com/docs>

Voytenko, V. (2019, 01 11). *Adding code to a Button Click in Android Studio (Java)*.
Retrieved 08 09, 2022, from YouTube: <https://www.youtube.com/watch?v=W-2fOVxYPSA>