Unit II:
Event Handling and TextVIew

CTE308- AS2025

Royal University of Bhutan

Tutor: Pema Galey

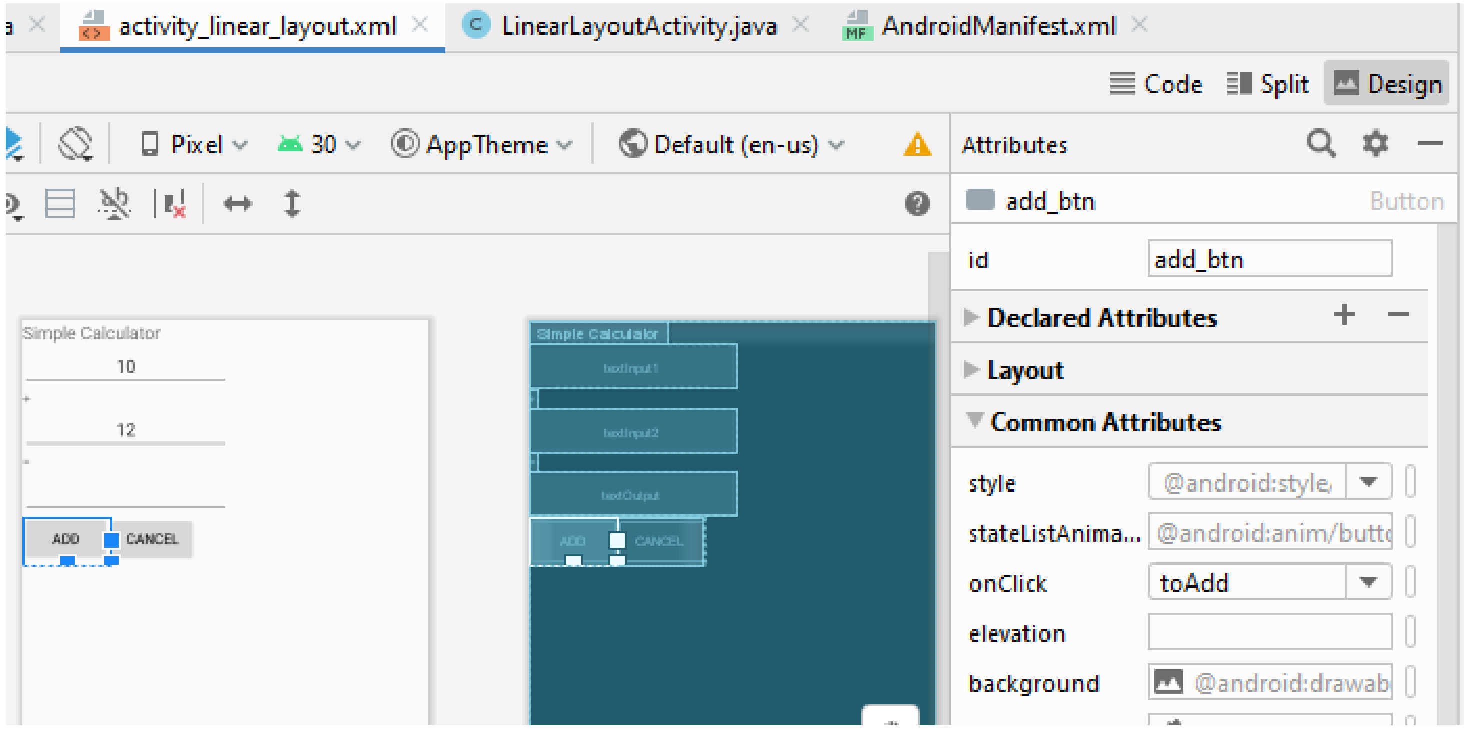
#17682761

Outline

- Button and Event in XML (onClick())
- Button and Event Handling using View.OnClickListener Interface
- Button creation and Event Handling during Run Time
- TextView, EditText & ScrollView

Button and Event in XML onClick() Attribute

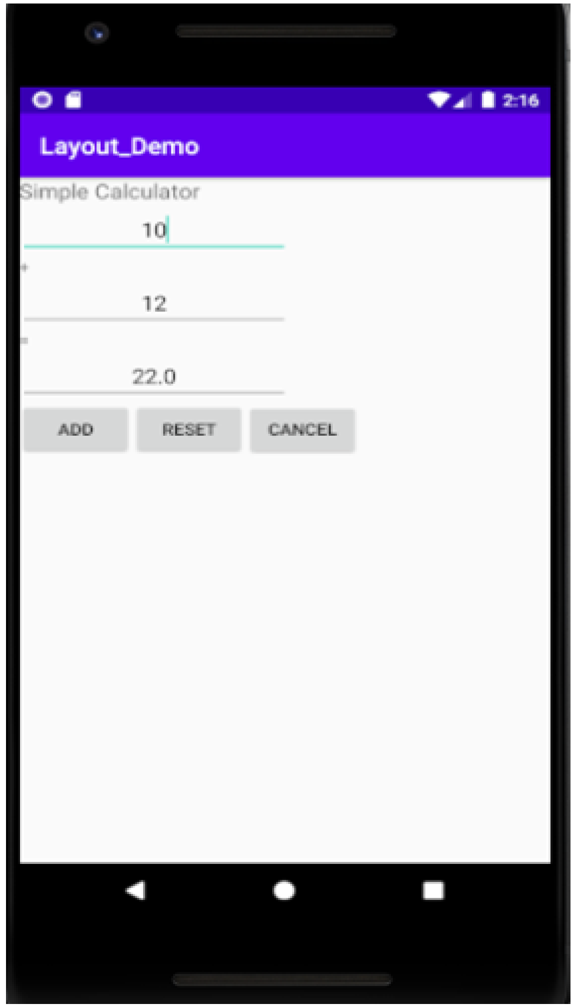
- Button creation in Visual Editor (XML)
- Set Button **onClick** attribute in XML layout file (Eg. **toAdd**).
- Event Handling Code in Java code in **toAdd()**



add_btn Button
id: add_btn
style: @android:style/
stateListAnima...: @android:anim/butto
onClick: toAdd
elevation:
background: @android:drawab

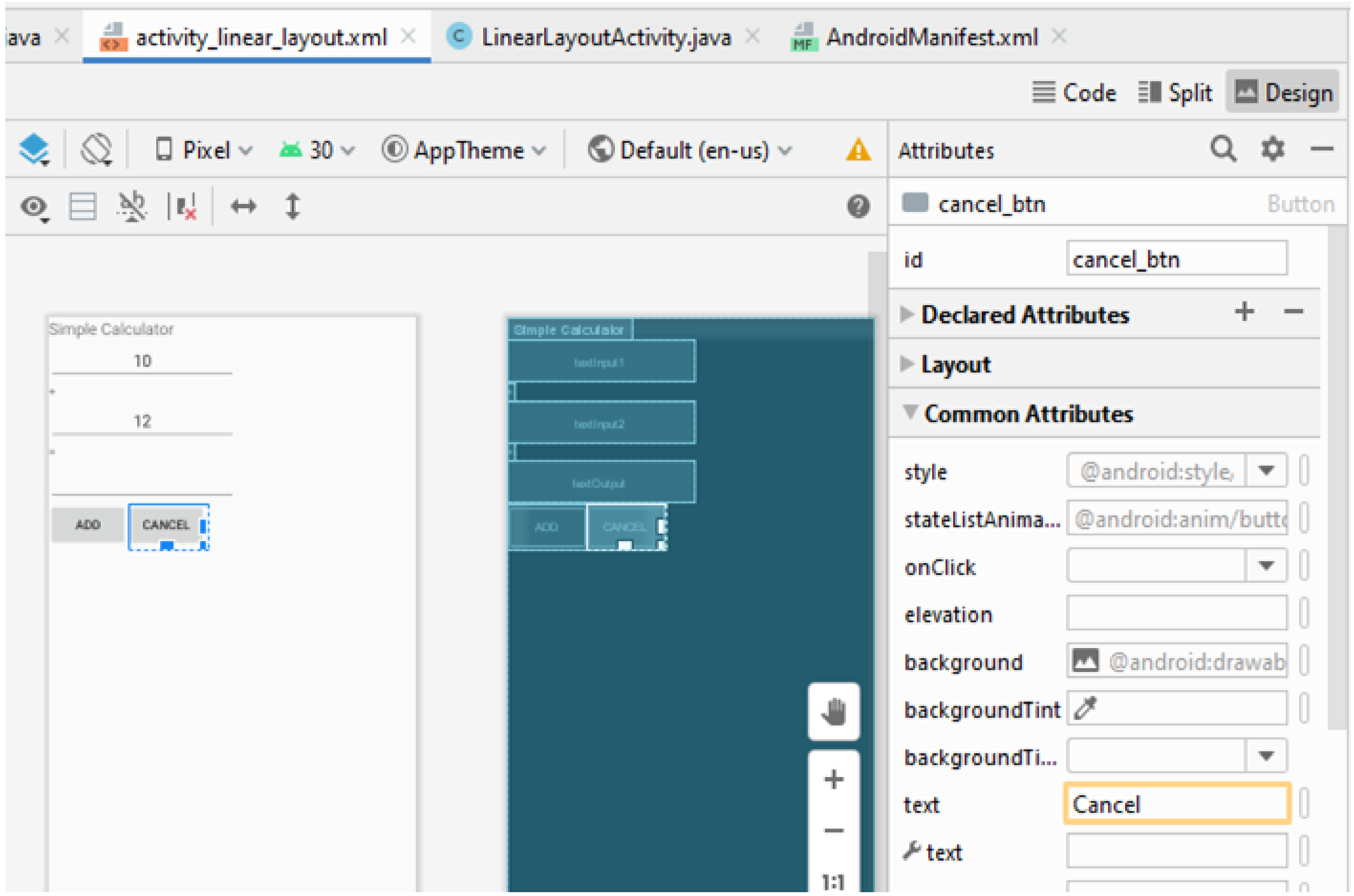
Java Code to handle button click(toAdd)

```
public class LinearLayoutActivity extends AppCompatActivity {
    EditText input1,input2, output;
    Button add, cancel,reset;
    Float sum;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_linear_layout);
        input1=(EditText) findViewById(R.id.textInput1);
        input2=(EditText)findViewById(R.id.textInput2);
        output=(EditText)findViewById(R.id.textOutput);
        add=(Button) findViewById(R.id.add_btn);
}
public void toAdd(View v){
  sum=Float.valueOf(input1.getText().toString()) +
                           Float.valueOf(input2.getText().toString());
  output.setText(String.valueOf(sum));
  Toast.makeText(this,String.valueOf(sum), Toast.LENGTH_LONG).show();
}
```



Button and Event Handling using View.OnClickListener Interface

- Button creation in Visual Editor XML Design (Eg, Cancel button)
- Event Handling Code in Java **onClick()**



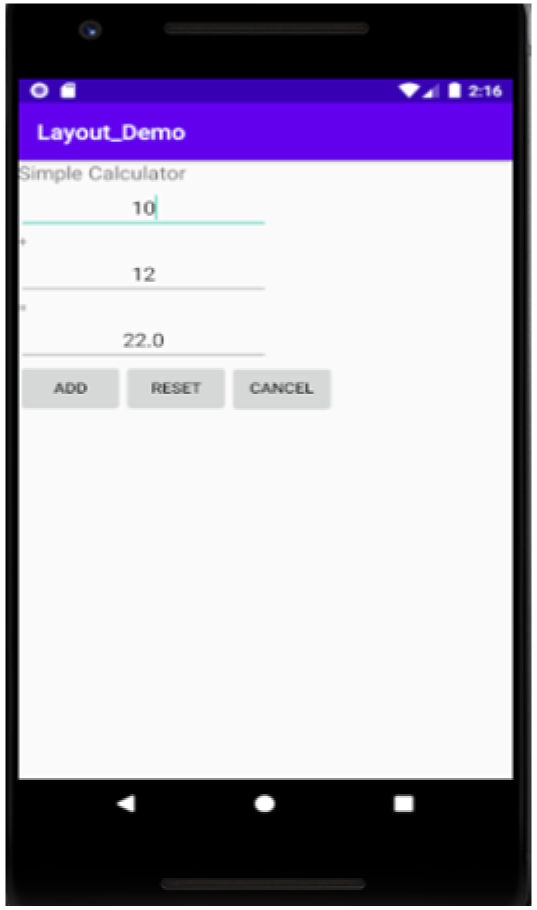
Attributes

cancel_btn Button

id	cancel_btn
Declared Attributes	
Layout	
Common Attributes	
style	@android:style/
stateListAnima...	@android:anim/butt
onClick	
elevation	
background	@android:drawab
backgroundTint	
backgroundTi...	
text	Cancel
text	

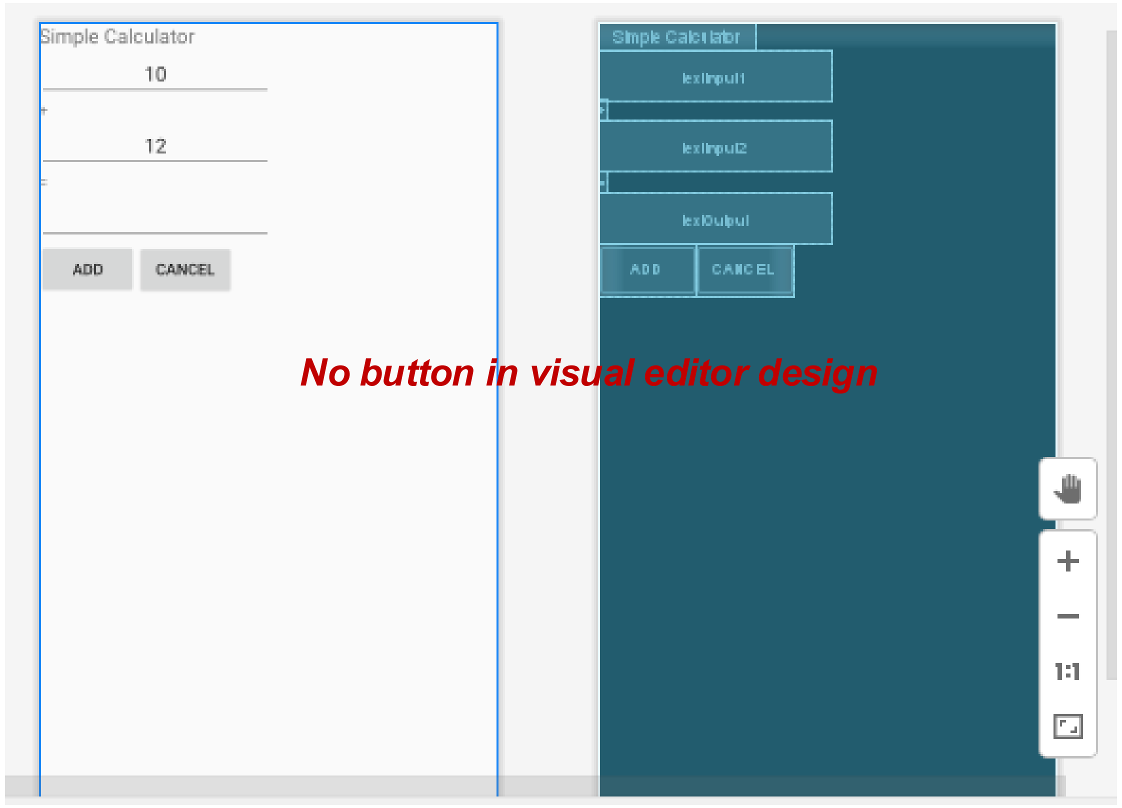
Java Code to handle button click

```
public class LinearLayoutActivity extends AppCompatActivity implements View.OnClickListener {
    EditText input1,input2, output;
    Button add, cancel, reset;
    Float sum;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_linear_layout);
        input1=(EditText) findViewById(R.id.textInput1);
        input2=(EditText)findViewById(R.id.textInput2);
        output=(EditText)findViewById(R.id.textOutput);
        add=(Button) findViewById(R.id.add_btn);

        cancel=(Button) findViewById(R.id.cancel_btn);
        cancel.setOnClickListener(this);

     @Override
     public void onClick(View v) {
        Toast.makeText(this,"Resetting……",Toast.LENGTH_LONG).show();
        output.setText("");

}
```

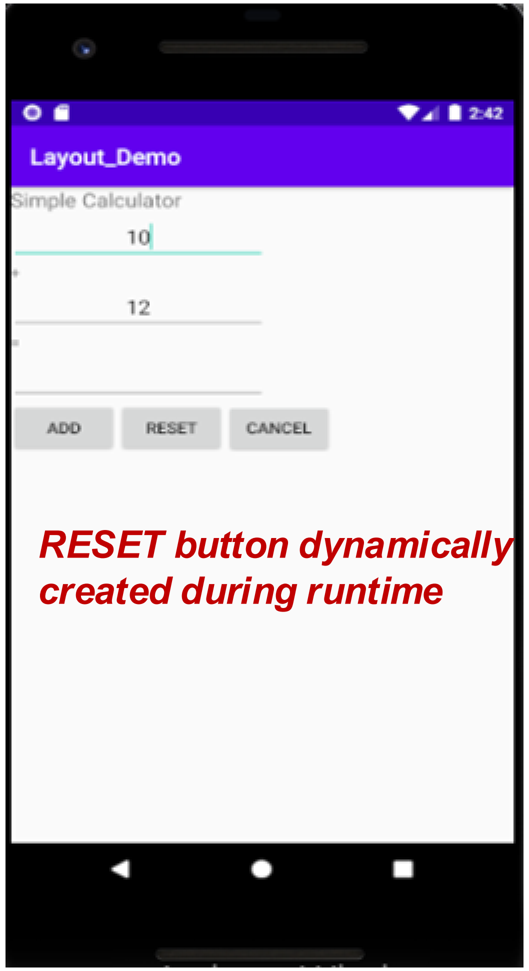


Create Button and Handle Event during Runtime

- Button creation during Runtime, no static design
- Add View to the Layout



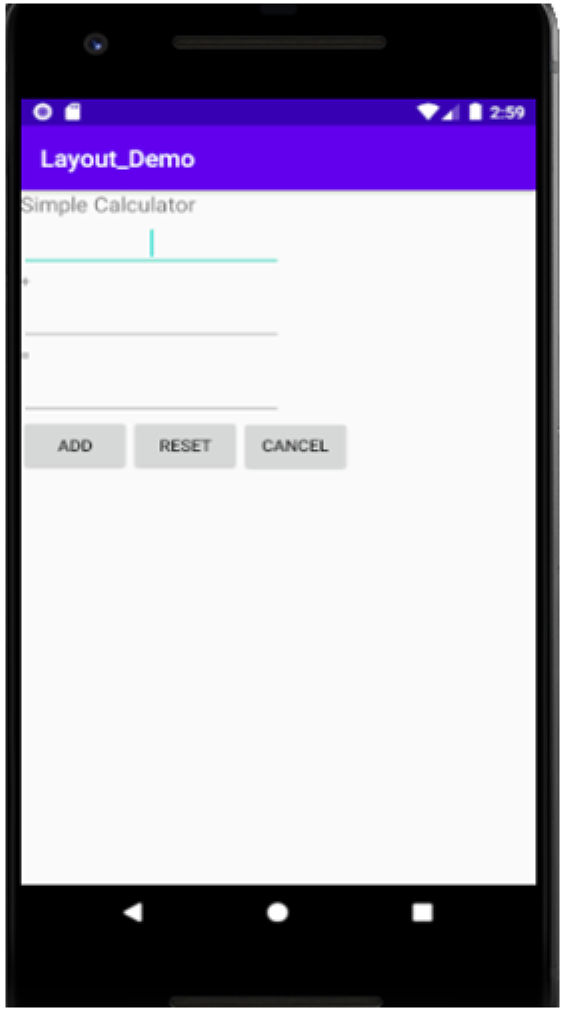
No button in visual editor design



RESET button dynamically created during runtime

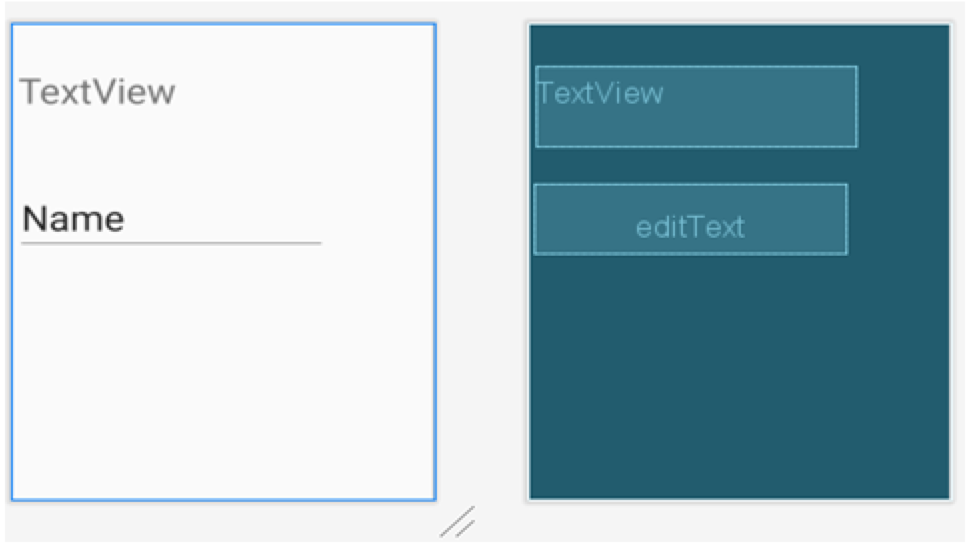
Runtime -Button Creation & event Handling

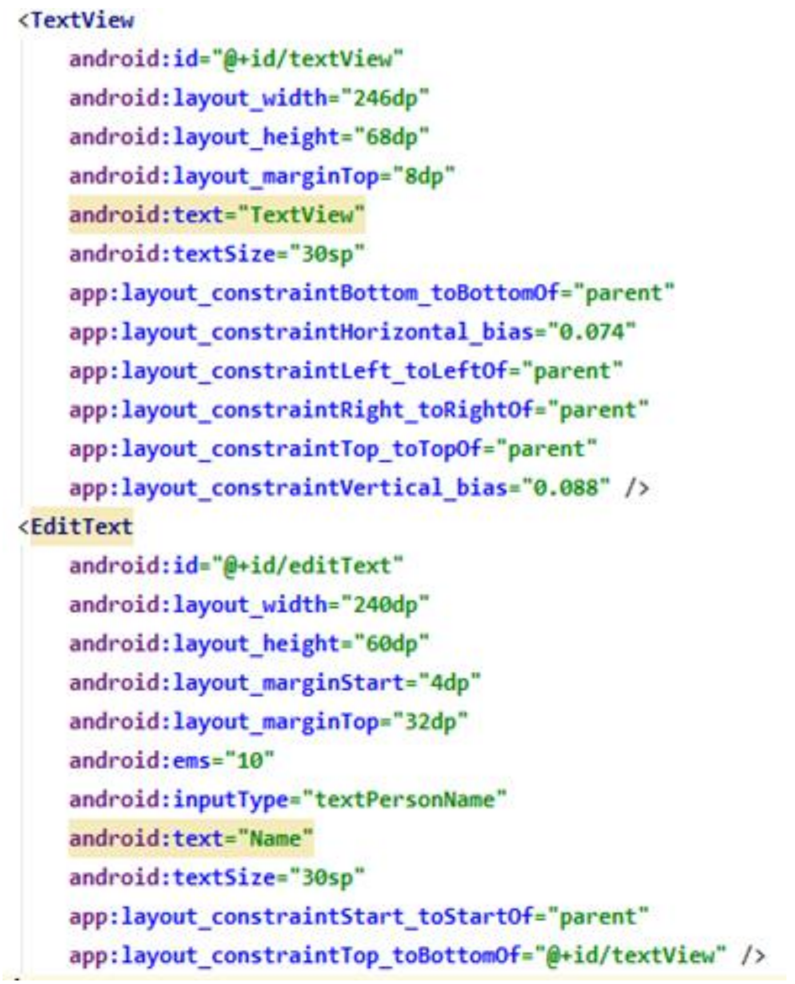
```
View.OnClickListener {
    EditText input1,input2, output;
    Button add, cancel, reset;
    Float sum;
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_linear_layout);
        input1=(EditText) findViewById(R.id.textInput1);
        input2=(EditText)findViewById(R.id.textInput2);
        output=(EditText)findViewById(R.id.textOutput);
    reset=new Button(getApplicationContext());
    reset.setText("reset");
    LinearLayout innerLL=(LinearLayout) findViewById(R.id.innerLLayout);
    LinearLayout.LayoutParams params=new LinearLayout.LayoutParams(
         RelativeLayout.LayoutParams.WRAP_CONTENT,
         RelativeLayout.LayoutParams.WRAP_CONTENT);
    //innerLL.addView(reset);
    innerLL.addView(reset,1,params);
    reset.setOnClickListener(new View.OnClickListener() {
     @Override
     public void onClick(View v) {
         input1.setText("");
         input2.setText("");
         output.setText("");
     }
 });}
```



TextView for Text

- **TextView** is a View for display single and multi-line text
- **EditText** is a subclass of TextView with editable text
- Set Text ***statically from a string resource (XML***), ***dynamically (Java code)***

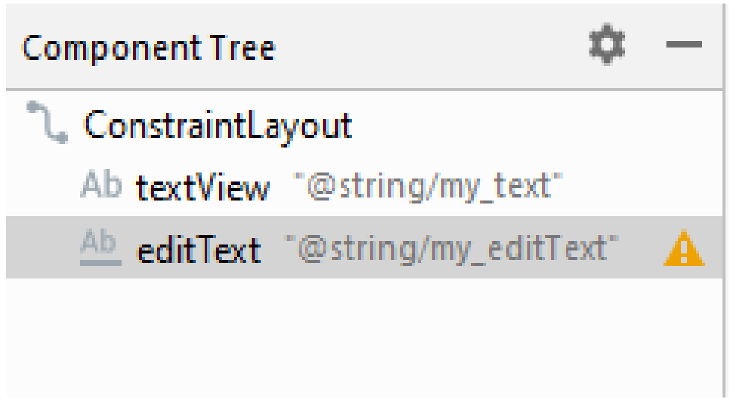




```
<TextView
    android:id="@+id/textView"
    android:layout_width="246dp"
    android:layout_height="68dp"
    android:layout_marginTop="8dp"
    android:text="TextView"
    android:textSize="30sp"
    app:layout_constraintBottom_toBottomOf="parent"
    app:layout_constraintHorizontal_bias="0.074"
    app:layout_constraintLeft_toLeftOf="parent"
    app:layout_constraintRight_toRightOf="parent"
    app:layout_constraintTop_toTopOf="parent"
    app:layout_constraintVertical_bias="0.088" />
<EditText
    android:id="@+id/editText"
    android:layout_width="240dp"
    android:layout_height="60dp"
    android:layout_marginStart="4dp"
    android:layout_marginTop="32dp"
    android:ems="10"
    android:inputType="textPersonName"
    android:text="Name"
    android:textSize="30sp"
    app:layout_constraintStart_toStartOf="parent"
    app:layout_constraintTop_toBottomOf="@+id/textView" />
```

Creating TextView XML

```
<TextView
    android:id="@+id/textView"
    android:layout_width="240dp"
    android:layout_height="70dp"
    android:text="@string/my_text"
    android:textSize="30sp" />
<EditText
    android:id="@+id/editText"
    android:layout_width="240dp"
    android:layout_height="60dp"
    android:text="@string/my_editText/>
```



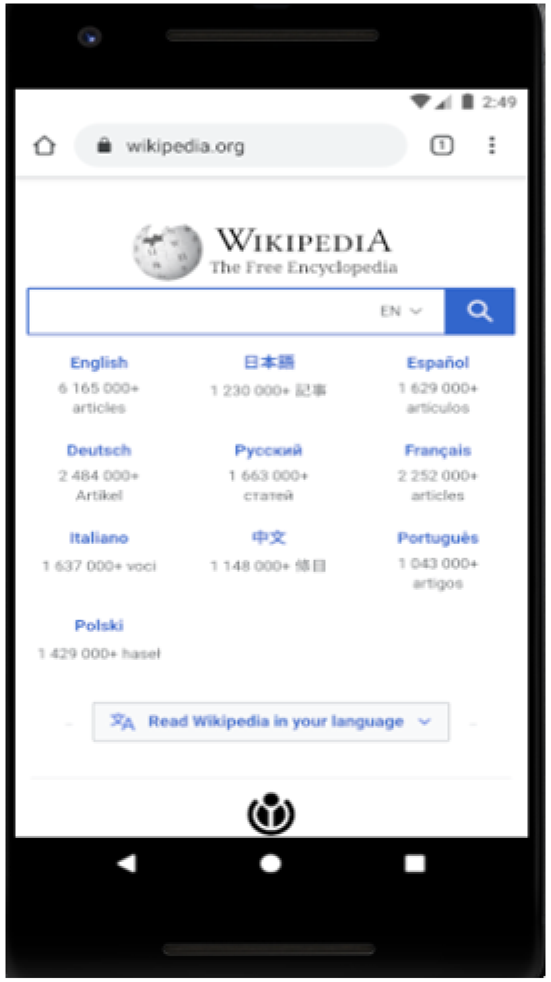
Common TextView Attributes

- android:**text**—text to display
- android:**textColor**—color of text
- android:**textAppearance**—predefined style or theme
- android:**textSize**—text size in `sp`
- android:**textStyle**—`normal`, `bold`, `italic`, or `bold|italic`
- android:**typeface**—`normal`, `sans`, `serif`, or `monospace`
- android:**lineSpacingExtra**—extra space between lines in `sp`

Formatting active web link

```
<string name="article_link">www.wikipedia.org/</string>

<TextView
    android:id="@+id/textView"
    android:layout_width="240dp"
    android:layout_height="70dp"
    android:text="@string/article_link" />
```

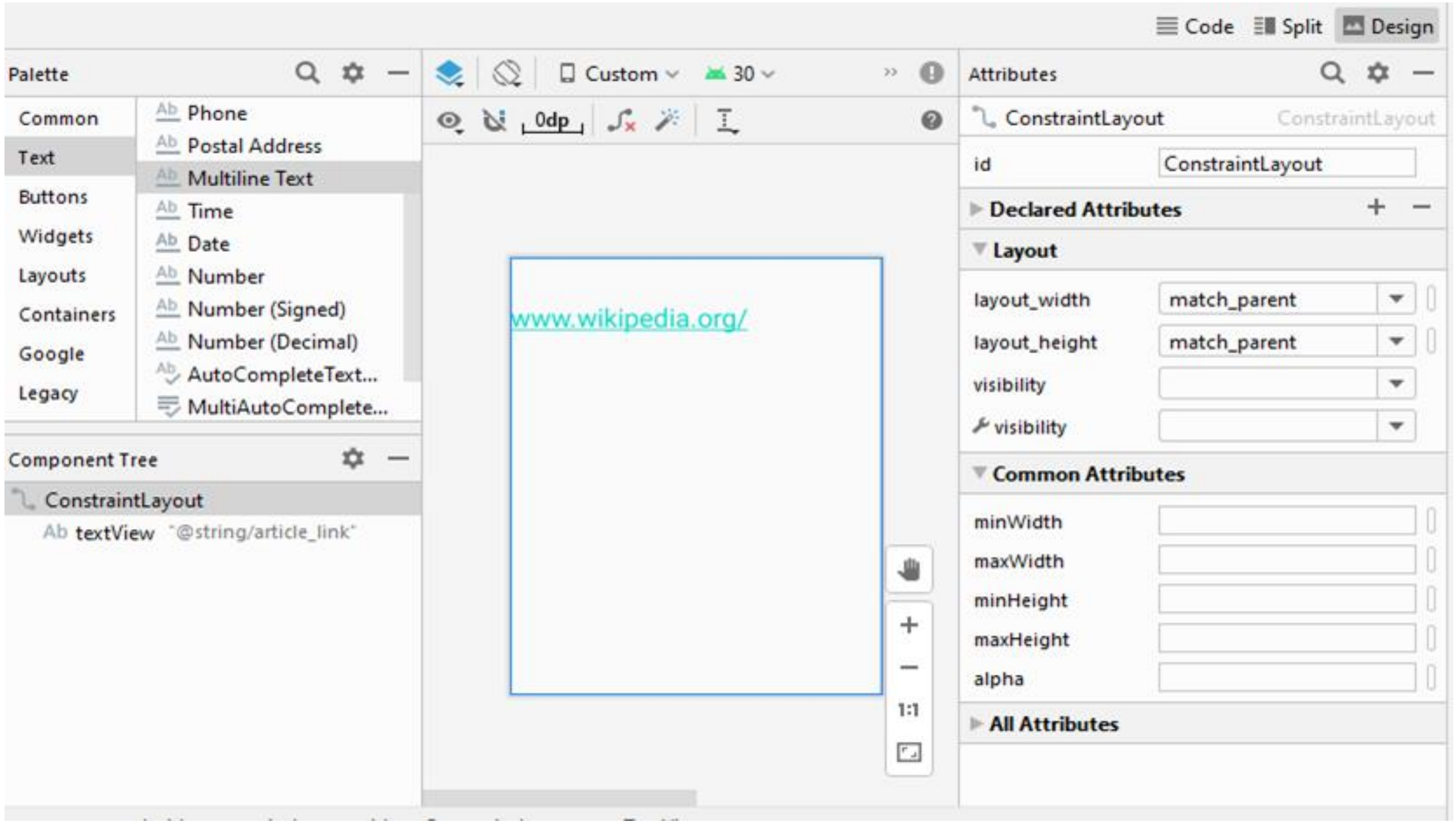
autoLink values:"web", "email", "phone", "map", "all"



Creating TextView in Java Code

```
TextView tv_java=new TextView(this);
//tv_java.setText("My Text Message");
ConstraintLayout cLayout = (ConstraintLayout) findViewById(R.id.ConstraintLayout);
ConstraintLayout.LayoutParams params = new ConstraintLayout.LayoutParams(
        ConstraintLayout.LayoutParams.WRAP_CONTENT,
        ConstraintLayout.LayoutParams.WRAP_CONTENT);
//tv_java.setMinLines(5);
tv_java.setText(R.string.my_text);
tv_java.append(" in TextView");
cLayout.addView(tv_java, params);
```

TextView (Not Design in Visual Editor Nor in XML)



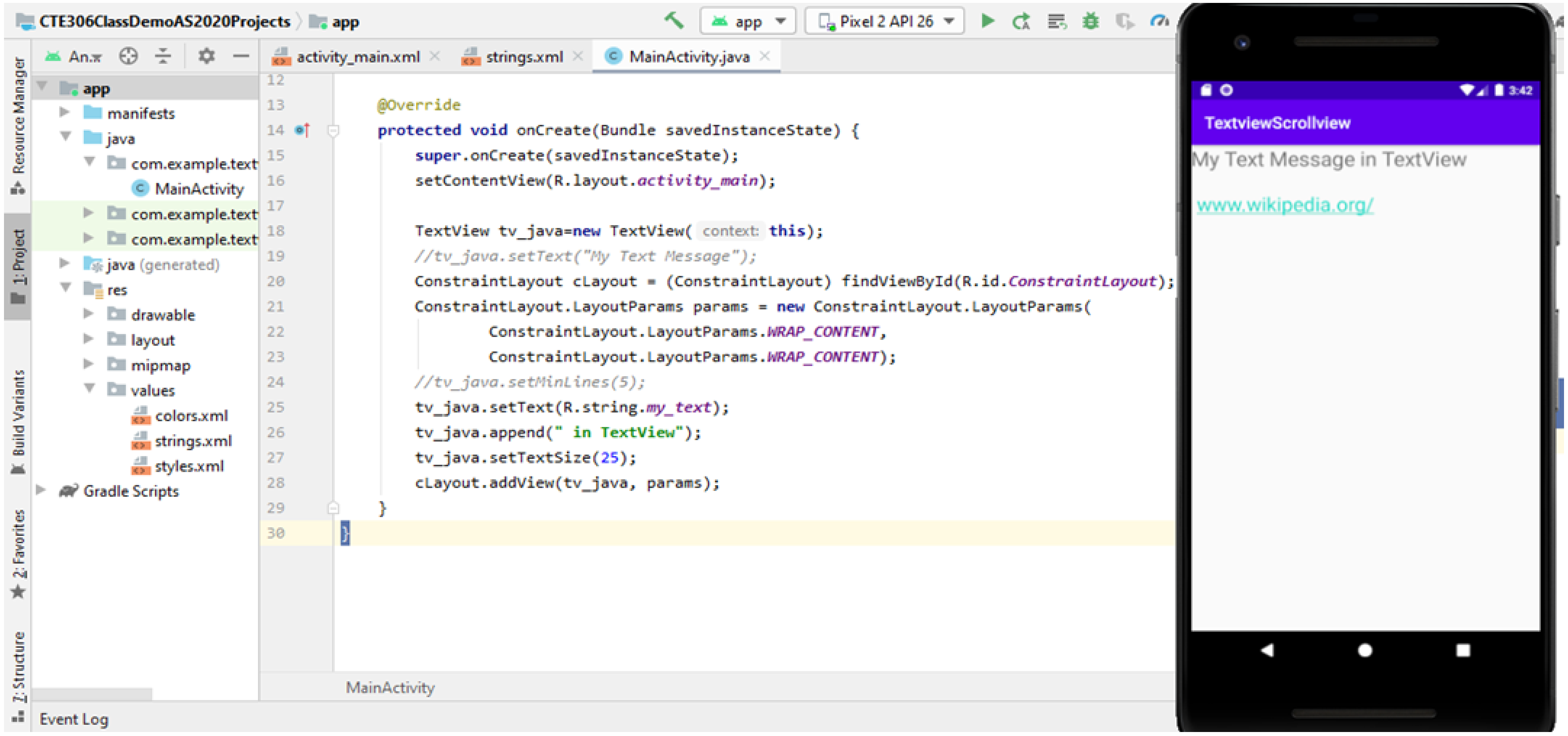
ConstraintLayout
textView "@string/article_link"

www.wikipedia.org/

ConstraintLayout ConstraintLayout

id	ConstraintLayout
layout_width	match_parent
layout_height	match_parent
visibility	
visibility	
minWidth	
maxWidth	
minHeight	
maxHeight	
alpha	

Creating TextView in Java Code

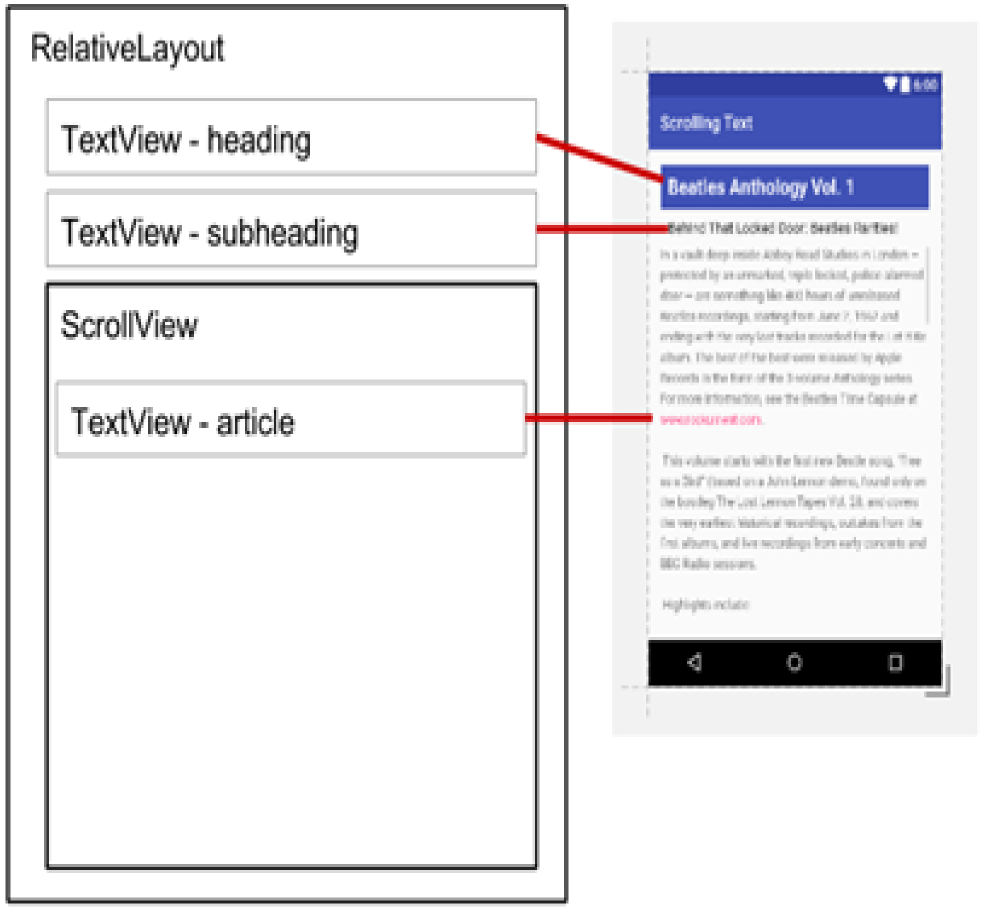


```
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);

        TextView tv_java=new TextView( context: this);
        //tv_java.setText("My Text Message");
        ConstraintLayout cLayout = (ConstraintLayout) findViewById(R.id.ConstraintLayout);
        ConstraintLayout.LayoutParams params = new ConstraintLayout.LayoutParams(
                ConstraintLayout.LayoutParams.WRAP_CONTENT,
                ConstraintLayout.LayoutParams.WRAP_CONTENT);
        //tv_java.setMinLines(5);
        tv_java.setText(R.string.my_text);
        tv_java.append(" in TextView");
        tv_java.setTextSize(25);
        cLayout.addView(tv_java, params);
    }
}
```

My Text Message in TextView

ScrollView: Large Amount of Text?

- Users may need to scroll.
  - News, Articles,...
- To allow users to scrolls a TextView, embed it in ScrollView
- Other views can also be embedded in ScrollView:
  - LinearLayout, TextView, Buttons, ...

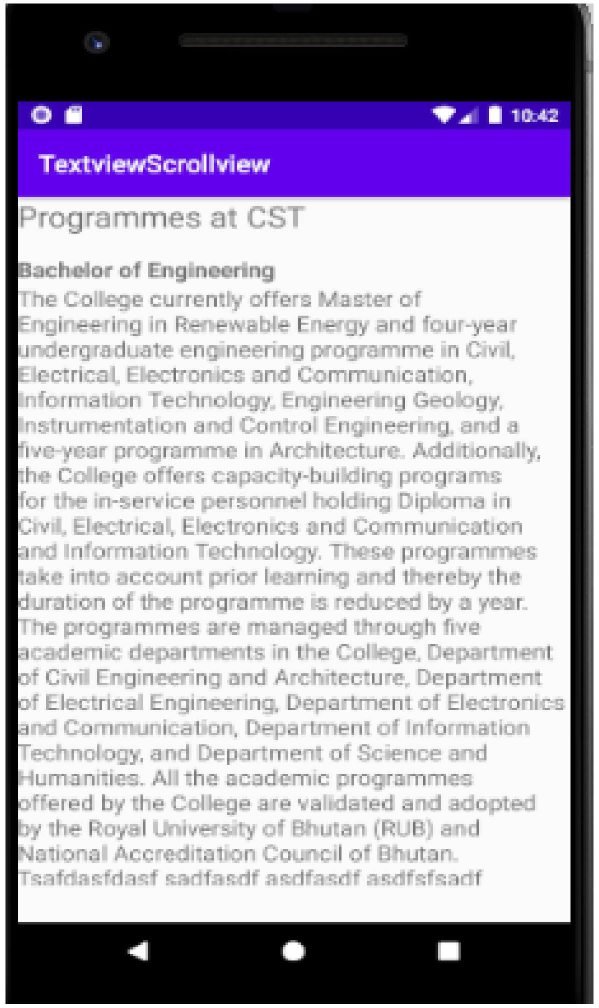


ScrollView for Scrolling the Content

- **ScrollView** is a subclass of **FrameLayout**.
- It can only hold one view.
- Holds all content in memory.
- Not good for long text, complex layout
- Do not nest multiple scrolling views
- Use HorizontalScrollView for horizontal scrolling
- Use a RecyclerView for lists

ScrollView for Scrolling the Content

```
<ScrollView
   android:layout_width="match_parent"
   android:layout_height="400dp">
   <TextView
       android:id="@+id/textView4"
       android:layout_width="match_parent"
       android:layout_height="match_parent"
       android:text="@string/txtContent"
       android:textAlignment="viewStart"
       android:textSize="18sp" />

</ScrollView>
```



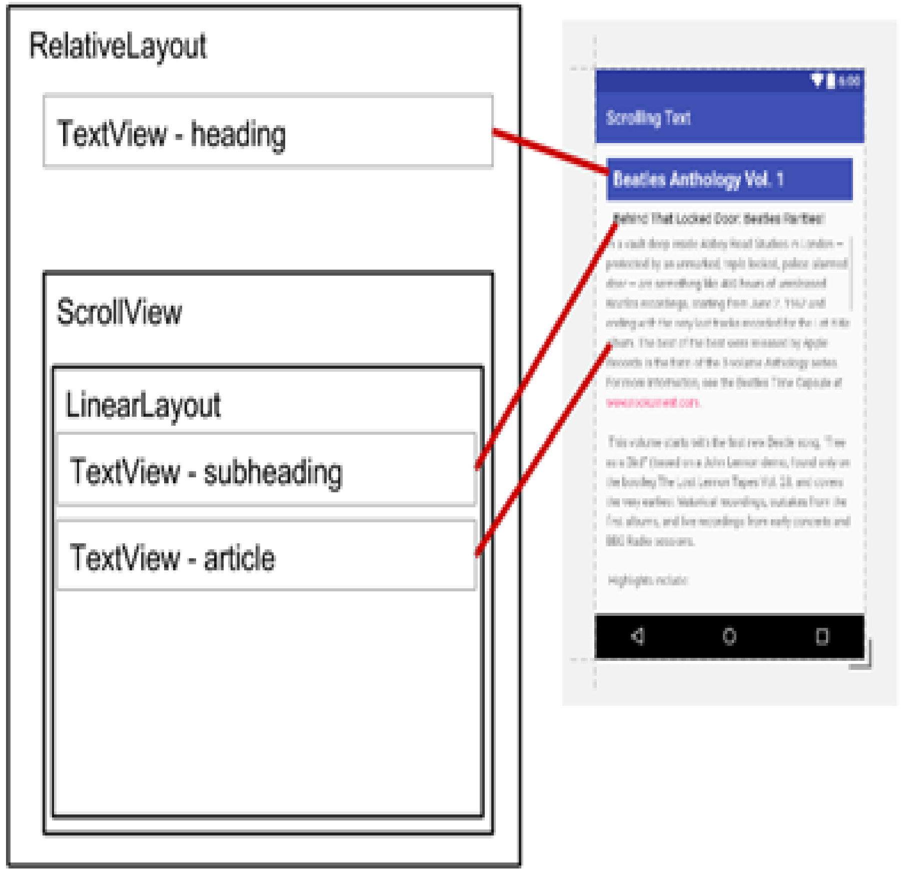
Programmes at CST

Bachelor of Engineering

The College currently offers Master of Engineering in Renewable Energy and four-year undergraduate engineering programme in Civil, Electrical, Electronics and Communication, Information Technology, Engineering Geology, Instrumentation and Control Engineering, and a five-year programme in Architecture. Additionally, the College offers capacity-building programs for the in-service personnel holding Diploma in Civil, Electrical, Electronics and Communication and Information Technology. These programmes take into account prior learning and thereby the duration of the programme is reduced by a year. The programmes are managed through five academic departments in the College, Department of Civil Engineering and Architecture, Department of Electrical Engineering, Department of Electronics and Communication, Department of Information Technology, and Department of Science and Humanities. All the academic programmes offered by the College are validated and adopted by the Royal University of Bhutan (RUB) and National Accreditation Council of Bhutan.
Tsafdasfdasf sadfasdf asdfasdf asdfsfsadf

ScrollView Layout with View Group

```
<ScrollView
   android:id="@+id/scrollView"
   android:layout_width="match_parent"
   android:layout_height="500dp">
   <LinearLayout
       android:layout_width="match_parent"
       android:layout_height="wrap_content"
       android:orientation="vertical">
       <TextView
           android:id="@+id/article_subheading"
           android:layout_width="match_parent"
           android:layout_height="wrap_content"
           android:text="Introduction"
           android:textAlignment="viewStart"
           android:textSize="18sp"
           android:textStyle="bold"
       <TextView
           android:id="@+id/article"
           android:layout_width="match_parent"
           android:layout_height="wrap_content"
           android:text="@string/txtContent" />
   </LinearLayout>
</ScrollView>
```



ScrollView with Image and Button

```
<ScrollView...>
      <LinearLayout...>
      <ImageView.../>
      <Button.../>
      <TextView.../>
      </LinearLayout>
</ScrollView>
```

One child of ScrollView which can be a layout

Children of the layout

Explore more…

- **Developer Documentation:**
  - TextView
  - ScrollView and HorizontalScrollView
  - String Resources

- **Other:**
  - Android Developers Blog: Linkify your Text!
  - Codepath: Working with a TextView

Thank you!