

# ITW202: Mobile Application

## Unit IV: Developing for Android

Ms. Sonam Wangmo

Gyalpozhing College of Information Technology  
Royal University of Bhutan

March 4, 2021

# Text and scrolling views



# Text and scrolling views



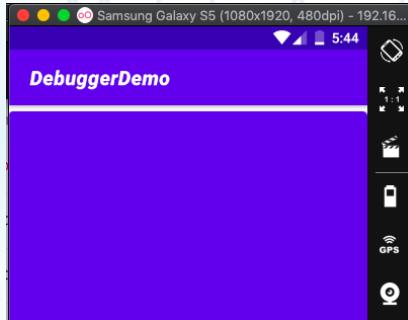
# TextView for text

- TextView is View subclass for single and multi-line text
- EditText is TextView subclass with editable text
- Controlled with layout attributes
- Set text:
  - Statically from string resource in XML
  - Dynamically from Java code and any source

# Formatting text in string resource

- Use `<b>` and `<i>` HTML tags for bold and italics
- All other HTML tags are ignored
- String resources: one unbroken line = one paragraph
- `\n` starts a new a line or paragraph
- Escape apostrophes and quotes with backslash (`\`" , `\`') )
- Escape any non-ASCII characters with backslash (`\`)

# Formatting text in string resource



```
answer_layout.xml x TheAnswer.java x gradle-wrapper.properties x strings.xml x
Edit translations for all locales in the translations editor. Open ed...
1 <resources>
2   <string name="app_name"><b> <i>DebuggerDemo</i></b></string>
3 </resources>
```

# Creating TextView in XML

Mobile Application Development

```
<TextView android:id="@+id/textview"  
    android:layout_width="match_parent"  
    android:layout_height="wrap_content"  
    android:text="@string/my_story"/>
```

# 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



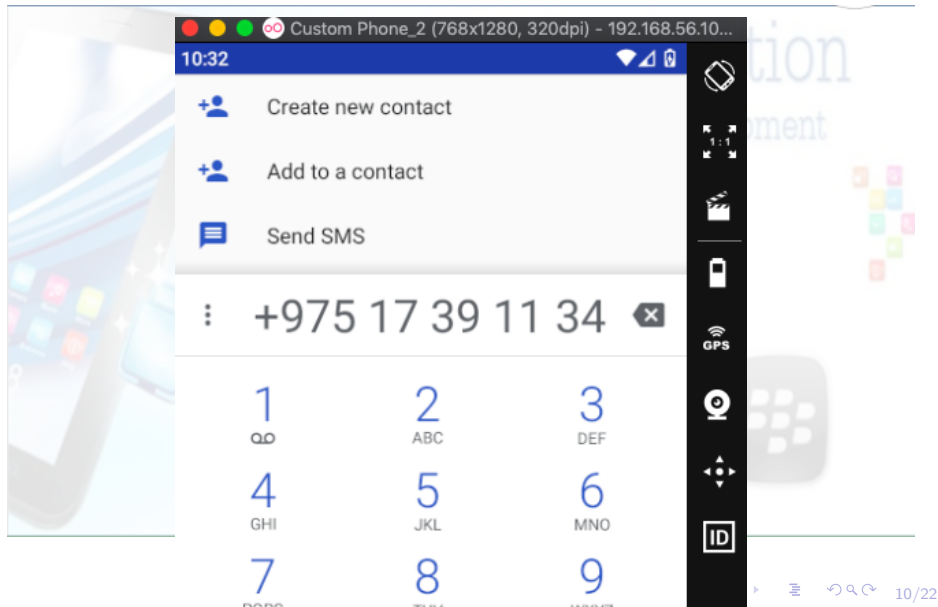
# Formatting active web links

```
1 <resources>
2   <string name="app_name"><b> <i>DebuggerDemo</i></b></string>
3   <string name="article_text">+97517391134</string>
4 </resources>
```

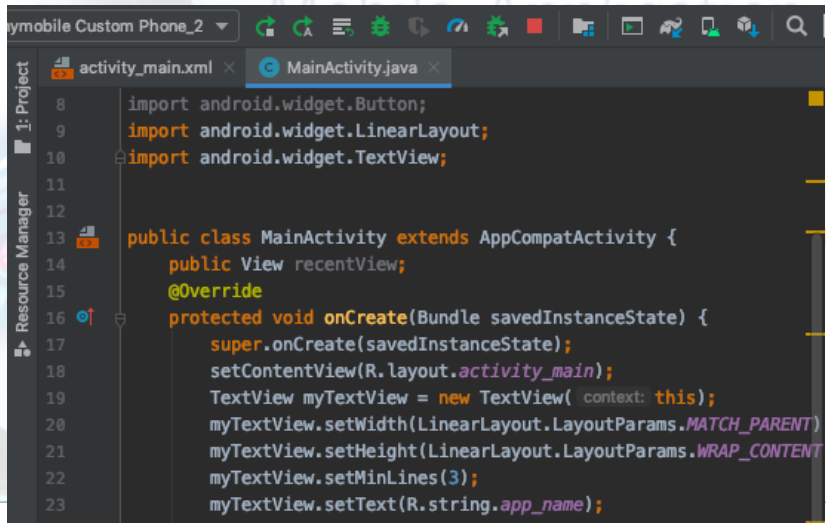
autoLink values: "web", "email", "phone", "map",  
"all"

```
answer_layout.xml x TheAnswer.java x gradle-wrapper.properties x strings.xml x
30 <Button
31   android:id="@+id/button"
32   android:layout_width="wrap_content"
33   android:layout_height="wrap_content"
34   android:layout_marginTop="124dp"
35   android:text="@string/article_text"
36   android:autoLink="phone"
37   app:layout_constraintEnd_toEndOf="parent"
38   app:layout_constraintTop_toBottomOf="@+id/answer_view" />
```

# Formatting active web links



# Creating TextView in Java code



```
1  import android.widget.Button;
2  import android.widget.LinearLayout;
3  import android.widget.TextView;
4
5  public class MainActivity extends AppCompatActivity {
6      public View recentView;
7      @Override
8      protected void onCreate(Bundle savedInstanceState) {
9          super.onCreate(savedInstanceState);
10         setContentView(R.layout.activity_main);
11         TextView myTextView = new TextView(context: this);
12         myTextView.setWidth(LinearLayout.LayoutParams.MATCH_PARENT);
13         myTextView.setHeight(LinearLayout.LayoutParams.WRAP_CONTENT);
14         myTextView.setMinLines(3);
15         myTextView.setText(R.string.app_name);
16     }
17 }
```

# Text and scrolling views



# What about large amounts of text?

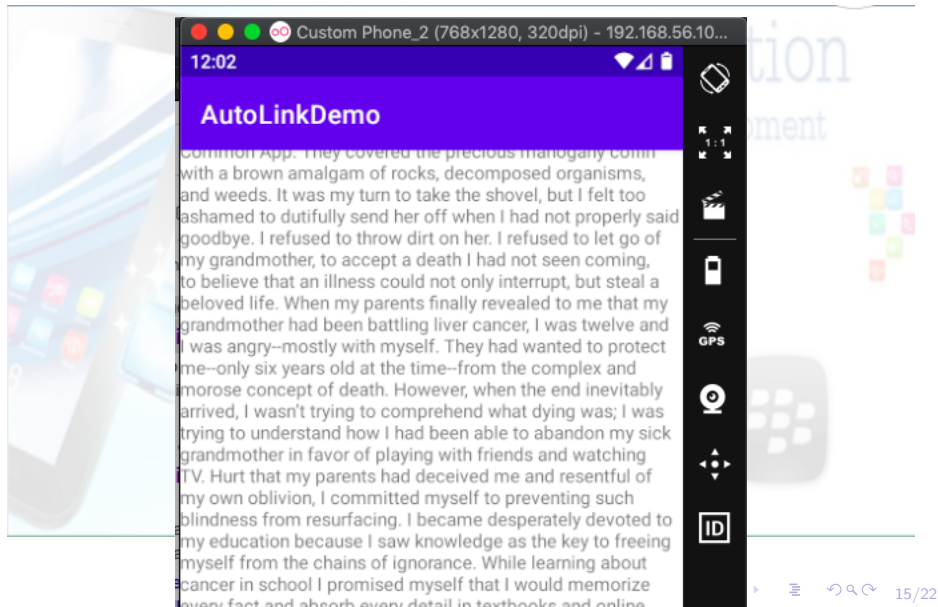
## Mobile Application Development

- News stories, articles, etc...
- To scroll a TextView, embed it in a ScrollView
- Only one View element (usually TextView) allowed in a ScrollView
- To scroll multiple elements, use one ViewGroup (such as LinearLayout) within the ScrollView

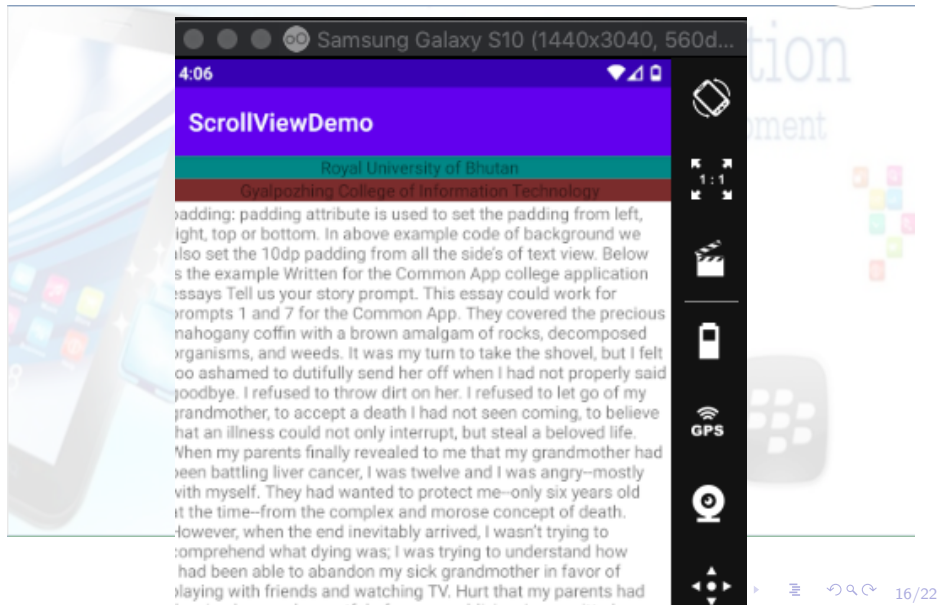
# ScrollView for scrolling content

- ScrollView is a subclass of FrameLayout
- Holds all content in memory
- Not good for long texts, complex layouts
- Do not nest multiple scrolling views
- Use HorizontalScrollView for horizontal scrolling
- Use a RecyclerView for lists

# ScrollView for scrolling content



# ScrollView layout with one TextView



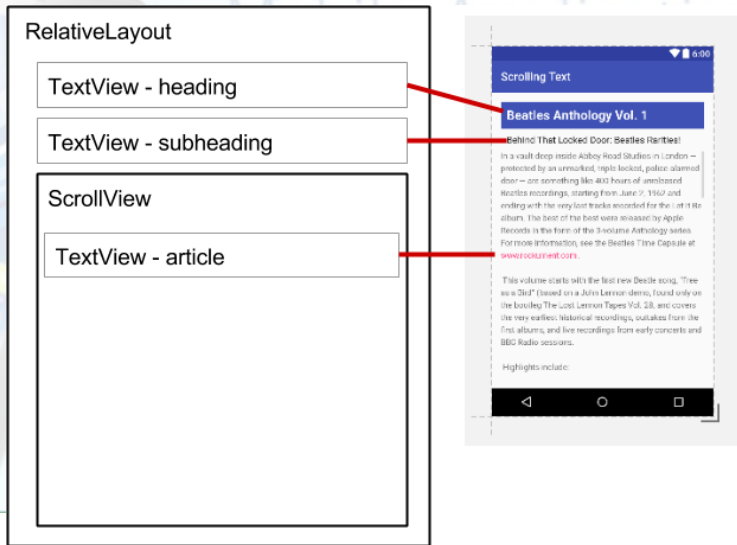


# ScrollView layout with one TextView

```
activity_main.xml x strings.xml x MainActivity.java x
Code Split Design

1 <?xml version="1.0" encoding="utf-8"?>
2 <RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"
3   xmlns:app="http://schemas.android.com/apk/res-auto"
4   xmlns:tools="http://schemas.android.com/tools"
5   android:layout_width="match_parent"
6   android:layout_height="match_parent"
7   android:orientation="vertical"
8   tools:context=".MainActivity">
9
10  <TextView
11    android:id="@+id/textView"
12    android:layout_width="match_parent"
13    android:layout_height="wrap_content"
14    android:background="@color/teal_700"
15    android:text="Royal University of Bhutan"
16    android:textAlignment="center"/>
17
18  <TextView
19    android:id="@+id/GCIT"
20    android:layout_width="match_parent"
21    android:layout_height="wrap_content"
22    android:background="#7A2C2C"
23    android:textAlignment="center"
24    android:text="Gyalpozhing College of Information Technology"
25    android:layout_below="@+id/textView"/>
26
27  <ScrollView
28    android:layout_width="wrap_content"
29    android:layout_height="wrap_content"
30    android:layout_below="@+id/GCIT">
31
```

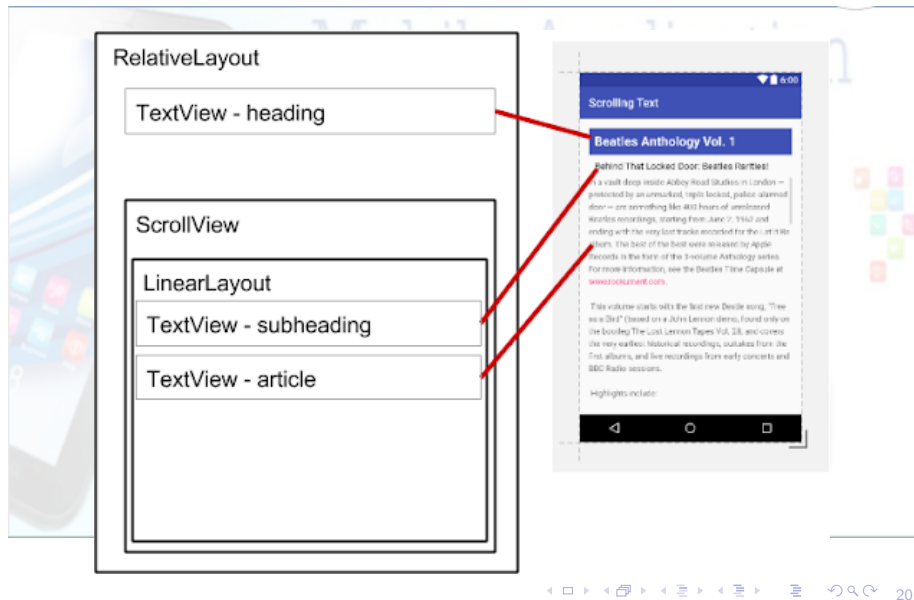
# ScrollView layout with one TextView



# ScrollView layout with a view group

```
<ScrollView ...  
    <LinearLayout  
        android:layout_width="match_parent"  
        android:layout_height="wrap_content"  
        android:orientation="vertical">  
  
        <TextView  
            android:id="@+id/article_subheading"  
            .../>  
  
        <TextView  
            android:id="@+id/article" ... />  
    </LinearLayout>  
</ScrollView>
```

# ScrollView layout with a view group



# 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

# Mobile Application Development

**THANK YOU**

