

Designing Android User Interface and Display with XML

Mr. Michaud
Marist School

XML Topics

- Tags
- Attributes
- Layouts
- Android XML Files
 - Layout
 - Manifest
 - Styles
 - Strings
- Resources Folder

What is XML?

- Markup Language designed to be both Human and Machine Readable
- HTML (Used in web pages) is a type of XML language.
- Used in Apple Preference files (plist), REAPER Music software, Web Based Layouts, and many other applications where software needs to create and read a dynamic and flexible database structure.
- Saved as a text file with the extension .xml

Android and XML

- Android uses XML to control the following
 - User Layout and Interface in non animated applications
 - Properties of the App (Manifest)
 - Store Values (Strings and Styles)

Tags: Core Element of XML

- A Tag defines an Object in XML. Any object or field can be defined with a Tag.
- Example:

```
<Button  
    android:id = "@+/buttonPlay"  
    android:layout_width = "wrap_content"  
    android:layout_height = "wrap_content"  
    android:onClick = "playSound01"  
    android:text = "Play" />
```

Tag

- Begins with a < and ends with a />
- Example
 - <Button />
- Another Example
 - <TableLayout> </TableLayout>

Tags and Attributes

- Attributes Modify or set properties of the Tag object.
- They are like the “Fields” in a class.
- Android Most Common:
 - android:id = “@+/nameOfObjectInCode”
 - android:layout_width = “match_parent”
 - android:layout_height = “match_parent”
 - android:text = “Text User Sees in Interface”
 - android:onClick = “nameOfFunction”

Example with TextView

Tag

Attributes

```
<TextView
```

```
    android:id="@+id/textSteps"  
    android:layout_width="wrap_content"  
    android:layout_height="200dp"  
    android:layout_span="4"  
    android:gravity="center_vertical|center_horizontal"  
    android:text="___"  
    android:textSize="40sp" />
```


Nested Objects

- Between the Start and End Tags, other XML objects can be nested inside.
- For example, a Linear Layout can have TextView and Button Objects 'inside' the layout.
- In Android User Interface, Objects such as text and buttons are held 'inside' layout objects such as Linear, Relative, and Table.

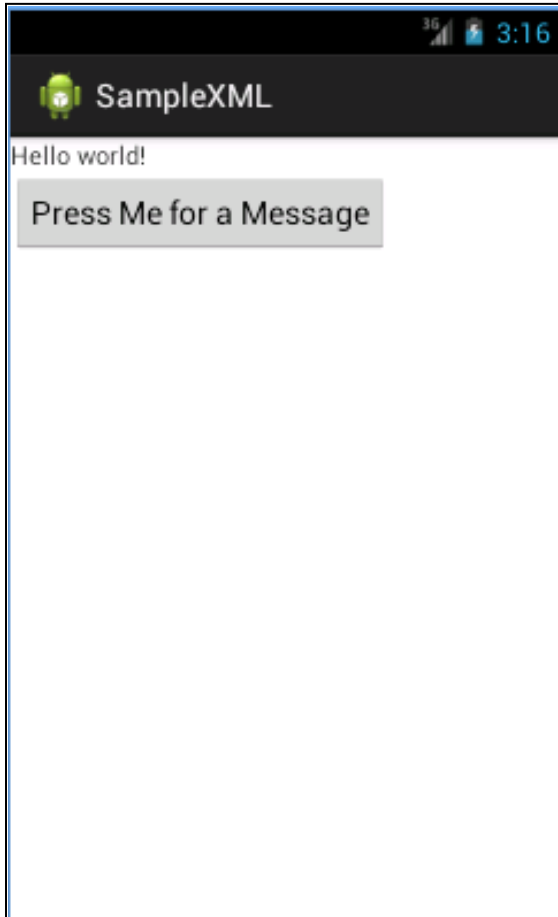
Layout Tag
Holds the Objects

Example

```
1 <LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
2   xmlns:tools="http://schemas.android.com/tools"
3   android:layout_width="match_parent"
4   android:layout_height="match_parent"
5   android:orientation="vertical"
6   tools:context=".MainActivity" >
7
8   <TextView
9     android:layout_width="wrap_content"
10    android:layout_height="wrap_content"
11    android:text="@string/hello_world" />
12
13   <Button
14     android:id = "@+id/buttonSample"
15     android:layout_width = "wrap_content"
16     android:layout_height = "wrap_content"
17     android:onClick = "displayText"
18     android:text = "Press Me for a Message" />
19
20 </LinearLayout>
21
```

Nested Objects:
TextView
Button

Example XML on Phone



```
1 <LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
2   xmlns:tools="http://schemas.android.com/tools"
3   android:layout_width="match_parent"
4   android:layout_height="match_parent"
5   android:orientation="vertical"
6   tools:context=".MainActivity" >
7
8   <TextView
9     android:layout_width="wrap_content"
10    android:layout_height="wrap_content"
11    android:text="@string/hello_world" />
12
13   <Button
14     android:id = "@+id/buttonSample"
15     android:layout_width = "wrap_content"
16     android:layout_height = "wrap_content"
17     android:onClick = "displayText"
18     android:text = "Press Me for a Message" />
19
20 </LinearLayout>
21
```

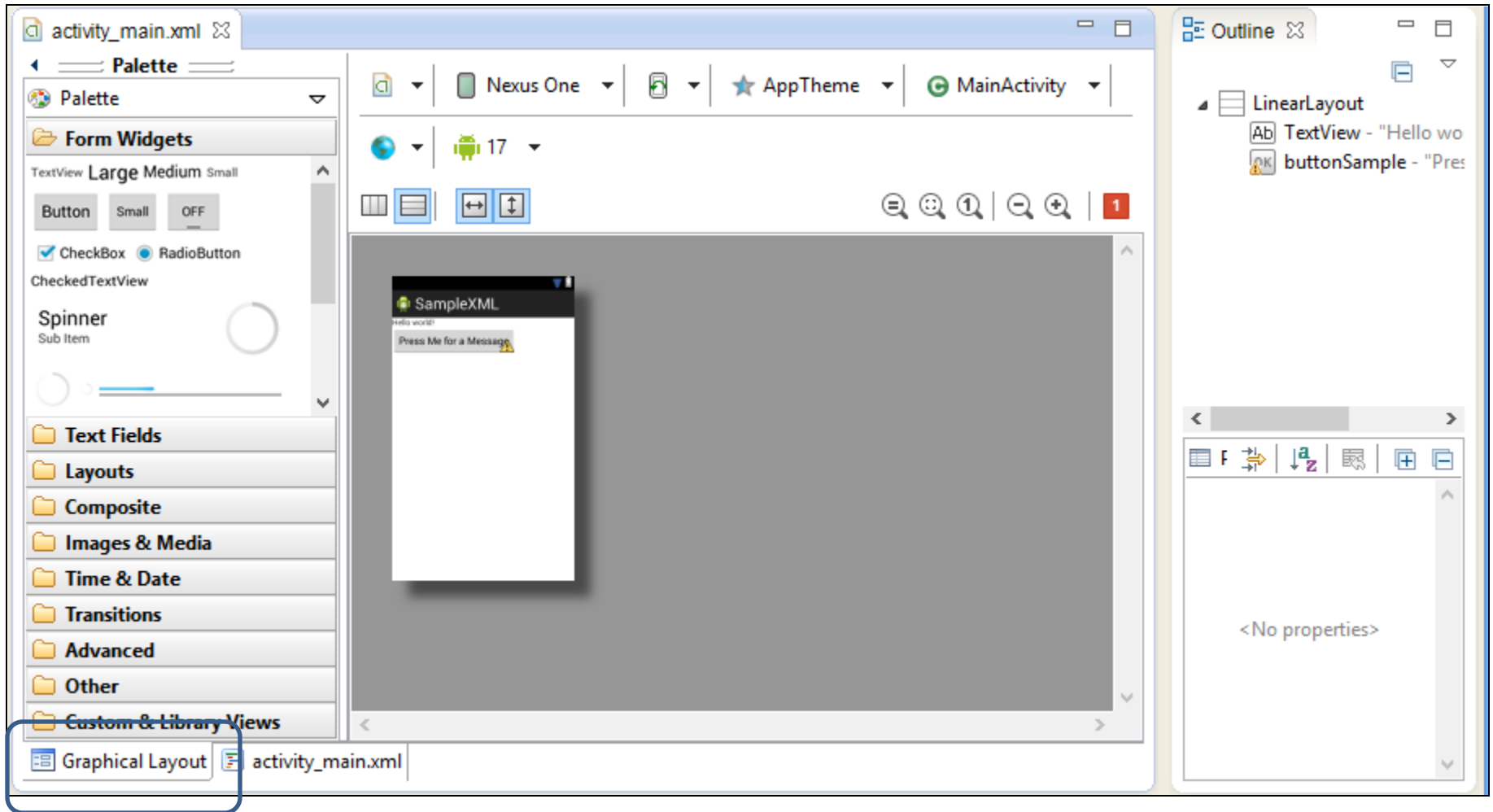
Android Layouts

- `<RelativeLayout>`
- `<LinearLayout>`
 - Horizontal
 - Vertical
- `<TableLayout>`
 - Used with nested `<TableRow>` objects

Android Objects (Not all)

- <TextView>
- <Button>
- <ImageButton>
- <EditText>
- <SeekBar>
- <View>
- <ImageView>

XML GUI Editor



XML Code Editor

The image shows a screenshot of an XML code editor. The main window displays the XML code for an activity named `activity_main.xml`. The code defines a `LinearLayout` with a vertical orientation, containing a `TextView` and a `Button`. The `Button` has an `onClick` event set to `displayText` and a text string `"Press Me for a Message"`. A warning icon is visible next to line 18, indicating a potential issue with the `android:text` attribute. The editor also features a graphical layout preview at the bottom, which is currently disabled, and an outline view on the right side showing the structure of the XML elements.

```
1 <LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
2   xmlns:tools="http://schemas.android.com/tools"
3   android:layout_width="match_parent"
4   android:layout_height="match_parent"
5   android:orientation="vertical"
6   tools:context=".MainActivity" >
7
8   <TextView
9     android:layout_width="wrap_content"
10    android:layout_height="wrap_content"
11    android:text="@string/hello_world" />
12
13   <Button
14     android:id = "@+id/buttonSample"
15     android:layout_width = "wrap_content"
16     android:layout_height = "wrap_content"
17     android:onClick = "displayText"
18     android:text = "Press Me for a Message" />
19
20 </LinearLayout>
21
22
23
```

Outline:

- LinearLayout
 - TextView: @string/he
 - Button: @+id/button