Basic Views

- Button

- ImageButton

- CheckBox

- ToggleButton

- RadioButton

- RadioGroup

Button

- Represents a push-button widget

```
<Button android:id="@+id/btnOpen"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="Open" />
```

ImageButton

- Similar to the Button view, except that it also displays an image

```
<ImageButton android:id="@+id/btnImg1"
    android:layout_width="fill_parent"
    android:layout_height="wrap_content"
    android:src="@drawable/icon" />
```

CheckBox

- A special type of button that has two states: checked or unchecked

```
<CheckBox android:id="@+id/chkAutosave"
    android:layout_width="fill_parent"
    android:layout_height="wrap_content"
    android:text="Autosave" />

<CheckBox android:id="@+id/star"
    style="?android:attr/starStyle"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content" />
```

RadioGroup and RadioButton

- The RadioButton has two states: either checked or unchecked. Once a RadioButton is checked, it cannot be unchecked.

- A RadioGroup is used to group together one or more RadioButton views, thereby allowing only one RadioButton to be checked within the RadioGroup.

```xml
<RadioGroup android:id="@+id/rdbGp1"
    android:layout_width="fill_parent"
    android:layout_height="wrap_content"
    android:orientation="vertical" >
    <RadioButton android:id="@+id/rdb1"
        android:layout_width="fill_parent"
        android:layout_height="wrap_content"
        android:text="Option 1" />
    <RadioButton android:id="@+id/rdb2"
        android:layout_width="fill_parent"
        android:layout_height="wrap_content"
        android:text="Option 2" />
</RadioGroup>
```

ToggleButton

- Displays checked/unchecked states using a light indicator

```xml
<ToggleButton android:id="@+id/toggle1"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content" />
```

ProgressBar View

- Provides visual feedback of some ongoing tasks, such as when you are performing a task in the background.

- For example, you might be downloading some data from the Web and need to update the user about the status of the download.

```xml
<ProgressBar android:id="@+id/progressbar"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content" />
```

AutoCompleteTextView View

- The AutoCompleteTextView is a view that is similar to EditText (in fact it is a subclass of EditText), except that it shows a list of completion suggestions automatically while the user is typing.

```xml
<AutoCompleteTextView android:id="@+id/txtCountries"
    android:layout_width="fill_parent"
    android:layout_height="wrap_content" />
```

LECTURE – 04-a "Basic Views"

THANK YOU ☺

By: AbuBakar Ubaid