

# Module 3 : UI Design and Data storage

## Designing Your User interface Using Views

Various view groups are :

Basic views>>> Commonly used views such as the TextView, EditText, and Button views

Picker views>>> Views that enable users to select from a list, such as the TimePicker and DatePicker views

List views>>> Views that display a long list of items, such as the ListView and the SpinnerView views

## Basic Views

To get started, let's explore some of the basic views that you can use to design the UI of your Android applications:

- ❖ TextView
- ❖ EditText
- ❖ Button
- ❖ ImageButton
- ❖ CheckBox
- ❖ ToggleButton
- ❖ RadioButton
- ❖ RadioGroup

These basic views enable you to display text information, as well as perform some basic selection.

## Textview View

When you create a new Android project, Eclipse always creates the main.xml file (located in the res/ layout folder), which contains a <TextView> element:

```
<?xmlversion="1.0"encoding="utf-8"?>
<LinearLayoutxmlns:android="http://schemas.android.com/apk/res/android"
android:orientation="vertical"
android:layout_width="fill_parent"
android:layout_height="fill_parent" >
<TextView
```

```
android:layout_width="fill_parent"
android:layout_height="wrap_content"
android:text="@string/hello" />
</LinearLayout>
```

The TextView view is used to display text to the user. This is the most basic view and one that you will frequently use when you develop Android applications. If you need to allow users to edit the text displayed, you should use the subclass of TextView, EditText.

NOTE In some other platforms, the TextView is commonly known as the label view. Its sole purpose is to display text on the screen.

### **Button, ImageButton, editText, checkBox, toggleButton, radioButton, and radiogroup views**

Besides the TextView view, which you will likely use the most often, there are some other basic controls that you will find yourself frequently using: Button, ImageButton, EditText, CheckBox, ToggleButton, RadioButton, and RadioGroup:

- Button — Represents a push-button widget
- ImageButton — Similar to the Button view, except that it also displays an image
- EditText — A subclass of the TextView view, except that it allows users to edit its text content
- CheckBox — A special type of button that has two states: checked or unchecked
- 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.
- ToggleButton — Displays checked/unchecked states using a light indicator.

```
<?xmlversion="1.0"encoding="utf-8"?>
<LinearLayoutxmlns:android="http://schemas.android.com/apk/res/android"
android:orientation="vertical"
android:layout_width="fill_parent"
android:layout_height="fill_parent">
```

```
<Button
    android:id="@+id/btnSave"
    android:layout_width="fill_parent"
    android:layout_height="wrap_content"
    android:text="Save"
/>
<Button
    android:id="@+id/btnOpen"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="Open"
/>
<ImageButton
    android:id="@+id/btnImg1"
    android:layout_width="fill_parent"
    android:layout_height="wrap_content"
    android:src="@drawable/icon"
/>
<EditText
    android:id="@+id/txtName"
    android:layout_width="fill_parent"
    android:layout_height="wrap_content"
/>
<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
    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
    android:id="@+id/toggle1"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content" />
</LinearLayout>
```



FIGURE 4-2