Mobile UI resources- Layout, UI elements, Draw-able

Android - UI Layouts

The basic building block for user interface is a **View** object which is created from the View class and occupies a rectangular area on the screen and is responsible for drawing and event handling. View is the base class for widgets, which are used to create interactive UI components like buttons, text fields, etc.

The **ViewGroup** is a subclass of **View** and provides invisible container that hold other Views or other ViewGroups and define their layout properties.

At third level we have different layouts which are subclasses of ViewGroup class and a typical layout defines the visual structure for an Android user interface and can be created either at run time using **View/ViewGroup** objects or you can declare your layout using simple XML file **main_layout.xml** which is located in the res/layout folder of your project.

Android Layout Types

There are number of Layouts provided by Android which you will use in almost all the Android applications to provide different view, look and feel.

Sr.No	Layout & Description
1	Linear Layout LinearLayout is a view group that aligns all children in a single direction, vertically or horizontally.
2	Relative Layout

	RelativeLayout is a view group that displays child views in relative positions.	
3	Table Layout TableLayout is a view that groups views into rows and columns.	
4	Absolute Layout AbsoluteLayout enables you to specify the exact location of its children.	
5	Frame Layout The FrameLayout is a placeholder on screen that you can use to display a single view.	
6	List View ListView is a view group that displays a list of scrollable items.	
7	Grid View GridView is a ViewGroup that displays items in a two-dimensional, scrollable grid.	

Layout Attributes

Each layout has a set of attributes which define the visual properties of that layout. There are few common attributes among all the layouts and their are other attributes which are specific to that layout. Following are common attributes and will be applied to all the layouts:

Sr.No	Attribute & Description
1	**android:id** This is the ID which uniquely identifies the view.
2	**android:layout_width** This is the width of the layout.
3	**android:layout_height** This is the height of the layout

4	**android:layout_marginTop** This is the extra space on the top side of the layout.
5	**android:layout_marginBottom** This is the extra space on the bottom side of the layout.
6	**android:layout_marginLeft** This is the extra space on the left side of the layout.
7	**android:layout_marginRight** This is the extra space on the right side of the layout.
8	**android:layout_gravity** This specifies how child Views are positioned.
9	**android:layout_weight** This specifies how much of the extra space in the layout should be allocated to the View.
10	**android:layout_x** This specifies the x-coordinate of the layout.
11	**android:layout_y** This specifies the y-coordinate of the layout.
12	**android:layout_width** This is the width of the layout.
13	**android:layout_width** This is the width of the layout.
14	**android:paddingLeft** This is the left padding filled for the layout.
15	**android:paddingRight**

		This is the right padding filled for the layout.
16	**android:paddingTop**	
	This is the top padding filled for the layout.	
17	**android:paddingBottom**	
	This is the bottom padding filled for the layout.	