

Module 3 : UI Design and Data storage

FrameLayout

The FrameLayout is a placeholder on screen that you can use to display a single view. Views that you add to a FrameLayout are always anchored to the top left of the layout. Consider the following content in main.xml:

```
<?xml version="1.0" encoding="utf-8"?>
<RelativeLayout
    android:id="@+id/RLayout"
    android:layout_width="fill_parent"
    android:layout_height="fill_parent"
    xmlns:android="http://schemas.android.com/apk/res/android"
>
<TextView
    android:id="@+id/lblComments"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="This is my lovely dog, Ookii"
    android:layout_alignParentTop="true"
    android:layout_alignParentLeft="true"
/>
<FrameLayout
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_alignLeft="@+id/lblComments"
    android:layout_below="@+id/lblComments"
    android:layout_centerHorizontal="true"
>
<ImageView
    android:src="@drawable/ookii"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
/>
```

```
</FrameLayout>
```

```
</RelativeLayout>
```

Here, you have a Frame Layout within a Relative Layout. Within the Frame Layout, you embed an ImageView. The UI is shown in Figure 3-10. NOTE This example assumes that the res/drawable-mdpi folder has an image named ookii.png. If you add another view (such as a Button view) within the FrameLayout, the view will overlap the previous view (see Figure 3-11):

```
<?xml version="1.0" encoding="utf-8"?>
```

```
<RelativeLayout
```

```
android:id="@+id/RLayout"
```

```
android:layout_width="fill_parent"
```

```
android:layout_height="fill_parent"
```

```
xmlns:android="http://schemas.android.com/apk/res/android"
```

```
>
```

```
<TextView
```

```
android:id="@+id/lblComments"
```

```
android:layout_width="wrap_content"
```

```
android:layout_height="wrap_content"
```

```
android:text="This is my lovely dog, Ookii"
```

```
android:layout_alignParentTop="true"
```

```
android:layout_alignParentLeft="true"
```

```
/>
```

```
<FrameLayout
```

```
android:layout_width="wrap_content"
```

```
android:layout_height="wrap_content"
```

```
android:layout_alignLeft="@+id/lblComments"
```

```
android:layout_below="@+id/lblComments"
```

```
android:layout_centerHorizontal="true"
```

```
>
```

```
<ImageView
```

```
android:src="@drawable/ookii"
```

```
android:layout_width="wrap_content"
```

```
android:layout_height="wrap_content"
```

/>

<Button

android:layout_width="124dp"

android:layout_height="wrap_content"

android:text="Print Picture"

/>

</FrameLayout>

</RelativeLayout>



FIGURE 3-10

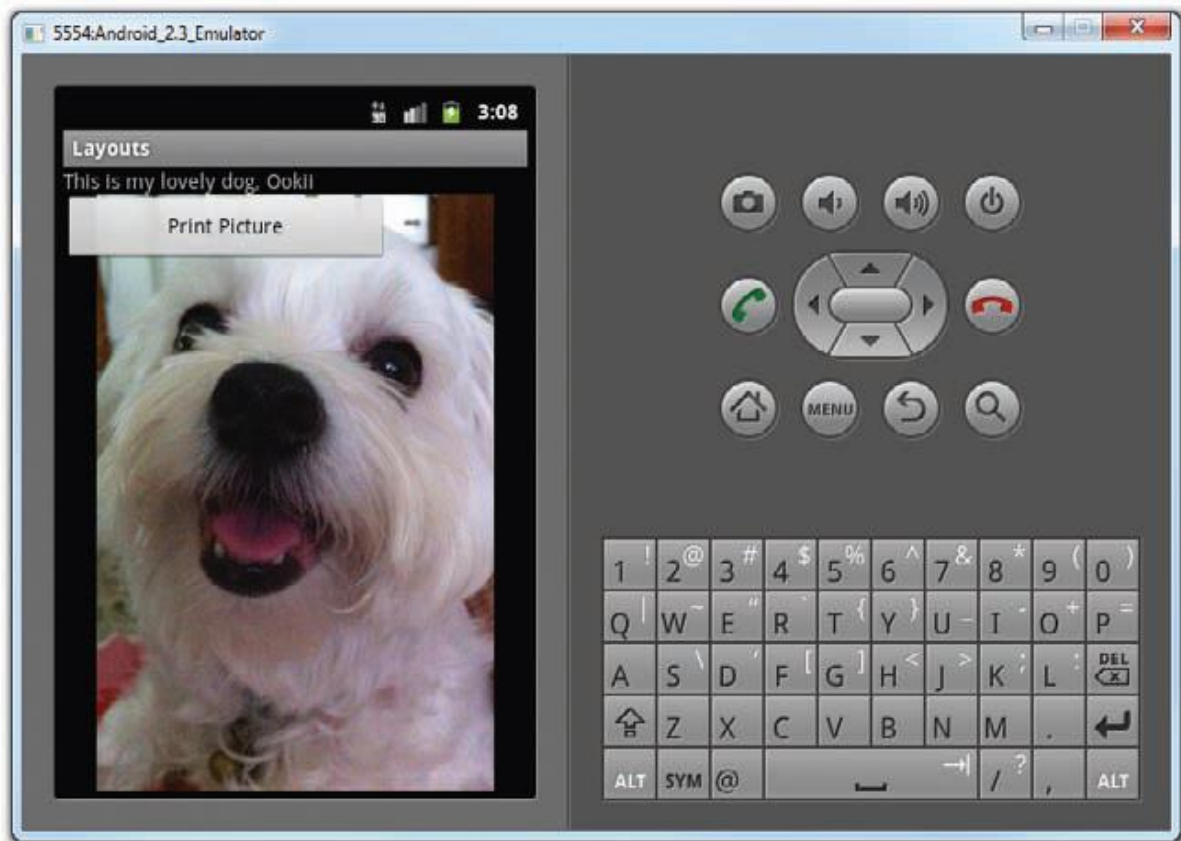


Figure 3-11

NOTE You can add multiple views to a `FrameLayout`, but each will be stacked on top of the previous one. This is useful in cases where you want to animate series of images, with only one visible at a time.

ScrollView

A `ScrollView` is a special type of `FrameLayout` in that it enables users to scroll through a list of views that occupy more space than the physical display. The `ScrollView` can contain only one child view or `ViewGroup`, which normally is a `LinearLayout`.

NOTE Do not use a `ListView` together with the `ScrollView`. The `ListView` is designed for showing a list of related information and is optimized for dealing with large lists. The following `main.xml` content shows a `ScrollView` containing a `LinearLayout`, which in turn contains some `Button` and `EditText` views:

```
<?xml version="1.0" encoding="utf-8"?>
<ScrollView
    android:layout_width="fill_parent"
    android:layout_height="fill_parent"
```

```
xmlns:android="http://schemas.android.com/apk/res/android"
```

```
>
```

```
<LinearLayout
```

```
android:layout_width="fill_parent"
```

```
android:layout_height="wrap_content"
```

```
android:orientation="vertical"
```

```
>
```

```
<Button
```

```
android:id="@+id/button1"
```

```
android:layout_width="fill_parent"
```

```
android:layout_height="wrap_content"
```

```
android:text="Button 1"
```

```
/>
```

```
<Button
```

```
android:id="@+id/button2"
```

```
android:layout_width="fill_parent"
```

```
android:layout_height="wrap_content"
```

```
android:text="Button 2"
```

```
/>
```

```
<Button
```

```
android:id="@+id/button3"
```

```
android:layout_width="fill_parent"
```

```
android:layout_height="wrap_content"
```

```
android:text="Button 3"
```

```
/>
```

```
<EditText
```

```
android:id="@+id/txt"
```

```
android:layout_width="fill_parent"
```

```
android:layout_height="300px"
```

```
/>
```

```
<Button
```

```
android:id="@+id/button4"
```

```
android:layout_width="fill_parent"
```

```
android:layout_height="wrap_content"
```

```
android:text="Button 4"  
</>  
<Button  
android:id="@+id/button5"  
android:layout_width="fill_parent"  
android:layout_height="wrap_content"  
android:text="Button 5"  
</>  
</LinearLayout>  
</ScrollView>
```

Figure 3-12 shows the ScrollView enabling the users to drag the screen upward to reveal the views located at the bottom of the screen.

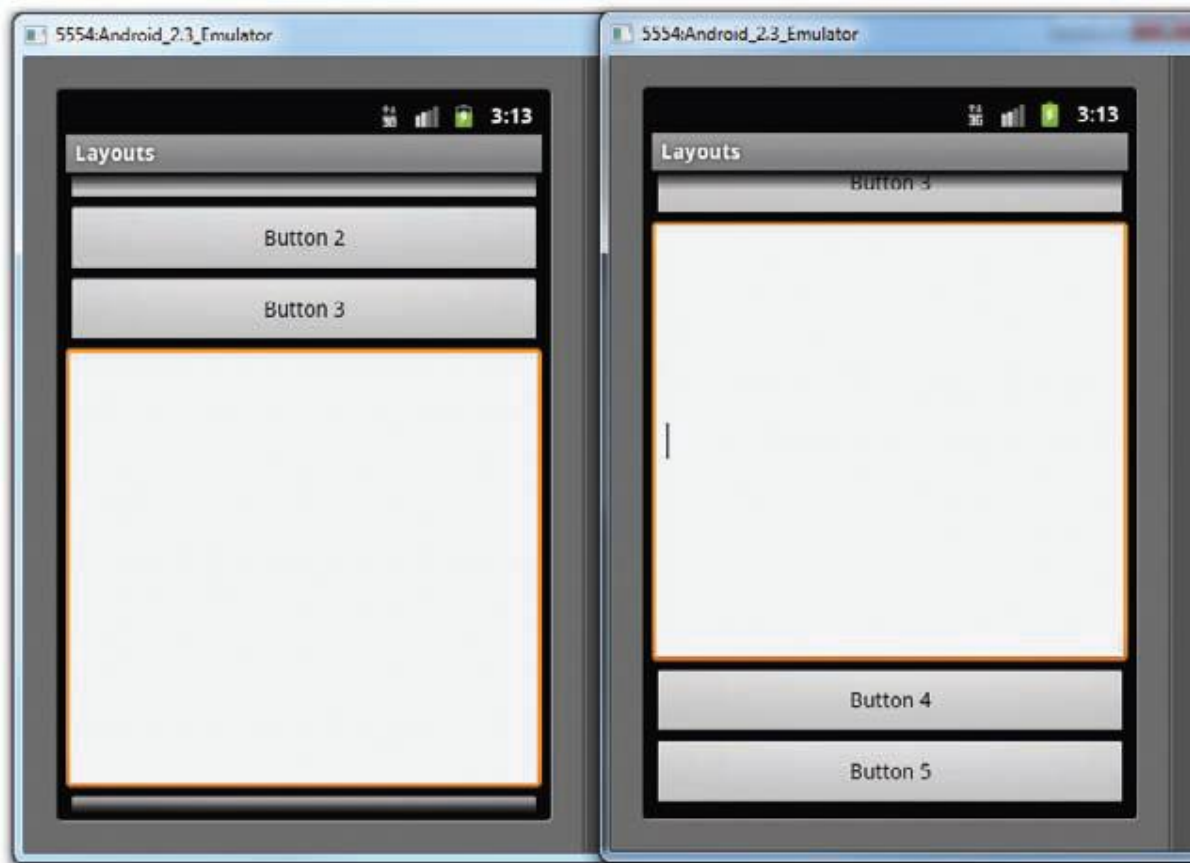


Figure 3-12