

# FrameLayout

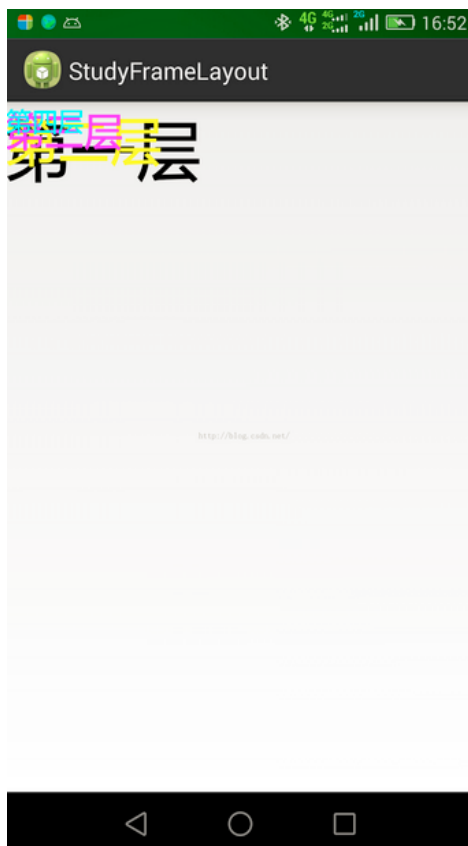
## 知识解析

FrameLayout 是所有布局中最简单的，在同一个 FrameLayout 中若有数个组件在内，则会以最上层的组件为主，若在同个 FrameLayout 当中有同大小的组件，则此时只会看到最上层的组件；若同一个 FrameLayout 当中有不同大小的组件，则会看到由下至上的组件。

FrameLayout 中所有的子元素将会固定在屏幕的左上角；你不能为 FrameLayout 中的一个子元素指定一个位置。后一个子元素将会直接在前一个子元素之上进行覆盖填充，把它们部份或全部挡住（除非后一个子元素是透明的）。



## 功能演示



## 实战操作

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

<FrameLayout

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

    android:layout_width="fill_parent"

    android:layout_height="fill_parent"

    >
```

```
<TextView
```

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

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

```
    android:textSize="50sp"
```

```
    android:textColor="#000000"
```

```
    android:text="第一层"/>
```

```
<TextView
```

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

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

```
    android:textSize="40sp"
```

```
    android:textColor="#ffff00"
```

```
    android:text="第二层"/>
```

```
<TextView
```

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

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

```
    android:textSize="30sp"
```

```
    android:textColor="#ff00ff"
```

```
    android:text="第三层"/>
```

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

```
        android:layout_height="wrap_content"

        android:textSize="20sp"

        android:textColor="#00ffff"

        android:text="第四层"/>

</FrameLayout>
```

## 职业素质

帧布局容器为每个加入起重工的组件创建了一个空白的区域，每个子组件占据一帧，这些帧都会根据 `gravity` 属性执行自动对齐。这布局的效果有点类似与 AWT 编程 的 `CardLayout`，都是把组件一个一个地叠加在一起，与 `CardLayout` 的区别在于，`CardLayout` 可以将下面的 `Card` 移动上来，但是 `FrameLayout` 则没有相对应的方法。