



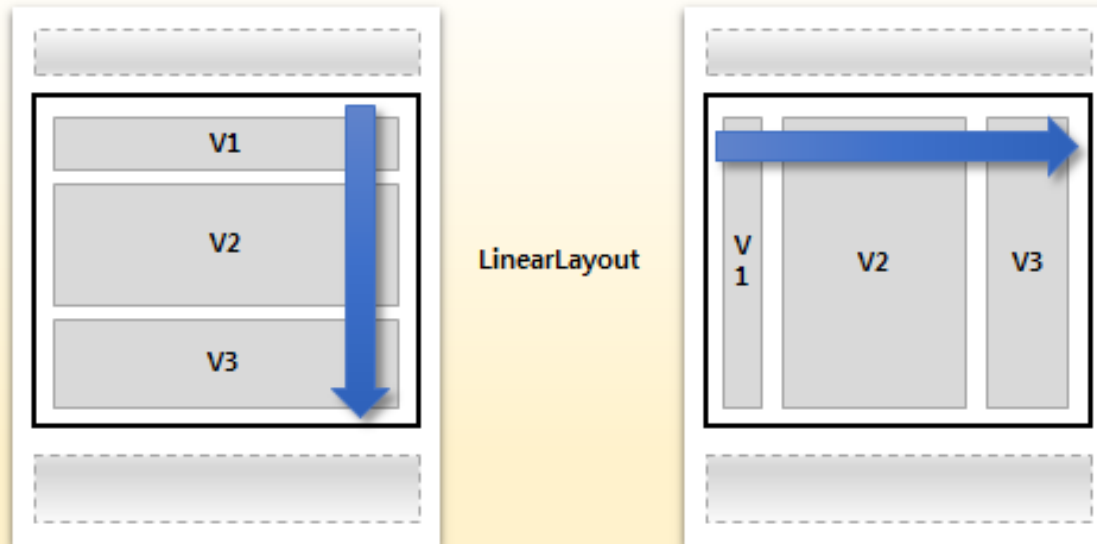
Linear Layout 실습

배 희호 교수
경북대학교
스마트IT과



LinearLayout

- LinearLayout은 Linear(선형, 선으로 된)라는 단어가 포함하는 의미대로, 여러 View(Widget)들을 **가로 또는 세로 방향으로** 나열할 때 사용하는 Layout 클래스

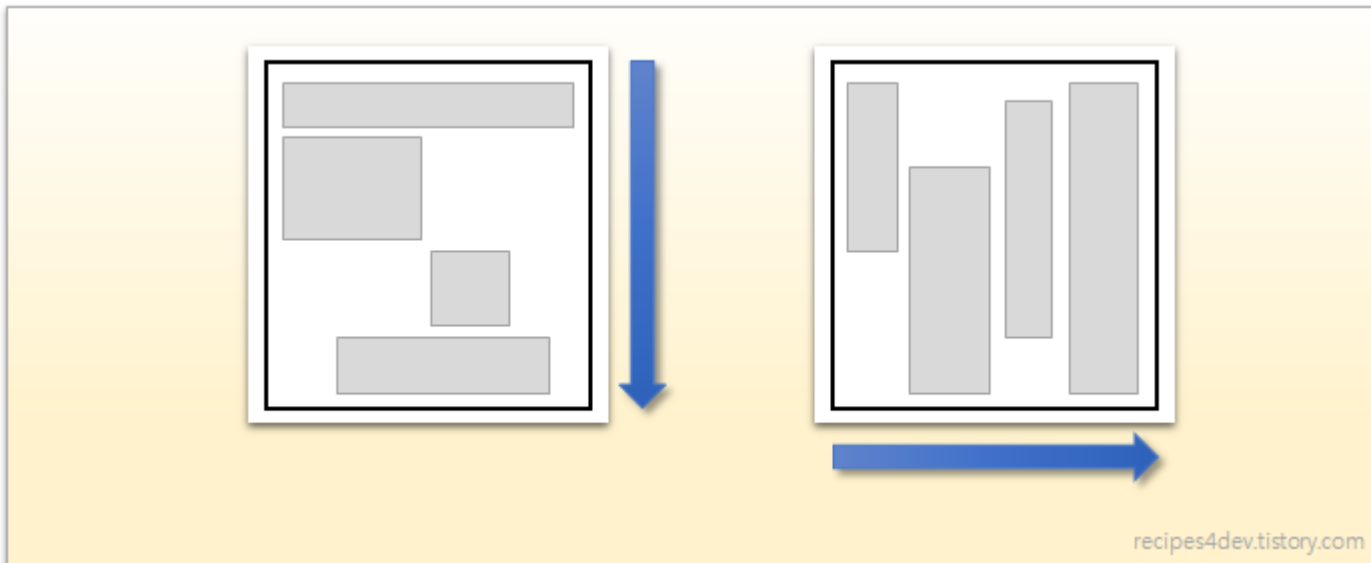


recipes4dev.tistory.com



LinearLayout

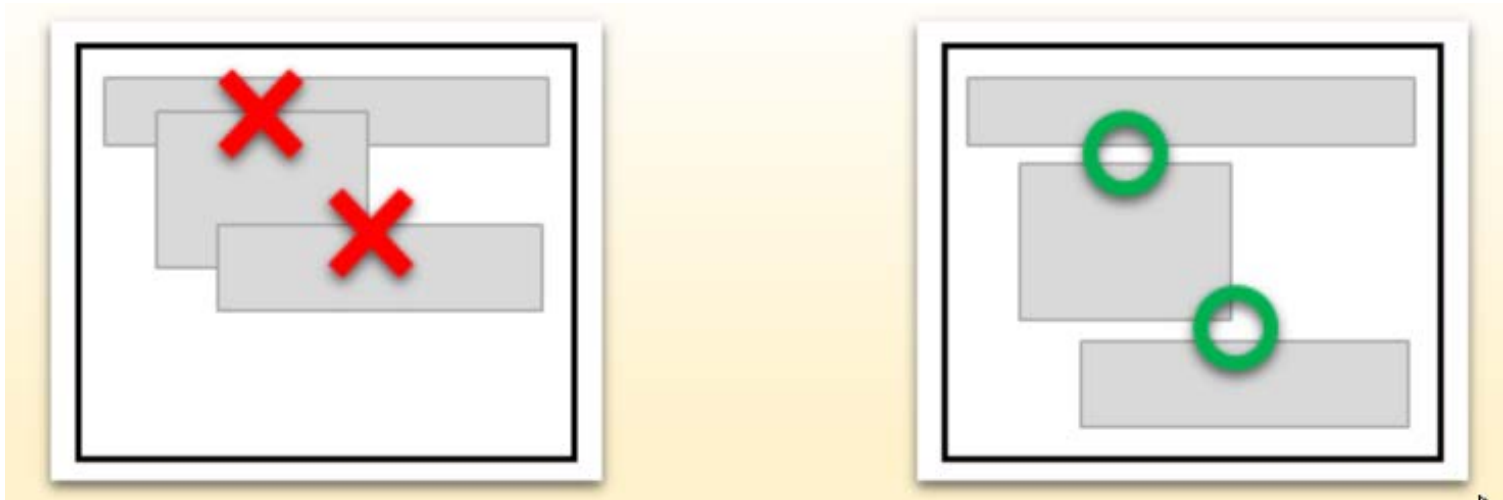
- LinearLayout의 자식(Children)으로 배치되는 View Widget 들은 오직 한 방향(가로 또는 세로)으로만 배치가 가능
-> Widget의 크기(높이 또는 너비)와 관계없이 한 줄로만 배열 함
- 아래 그림과 같이 가로 방향으로 배치될 때는 가로로 한 줄 (only one row), 세로 방향으로 배치될 때는 세로로 한 줄 (only one column)로 표시





LinearLayout

- LinearLayout의 자식들은 중첩(overlap)되지 않고, 지정한 방향으로 쌓이는(stacked) 형태로 표시됨





LinearLayout

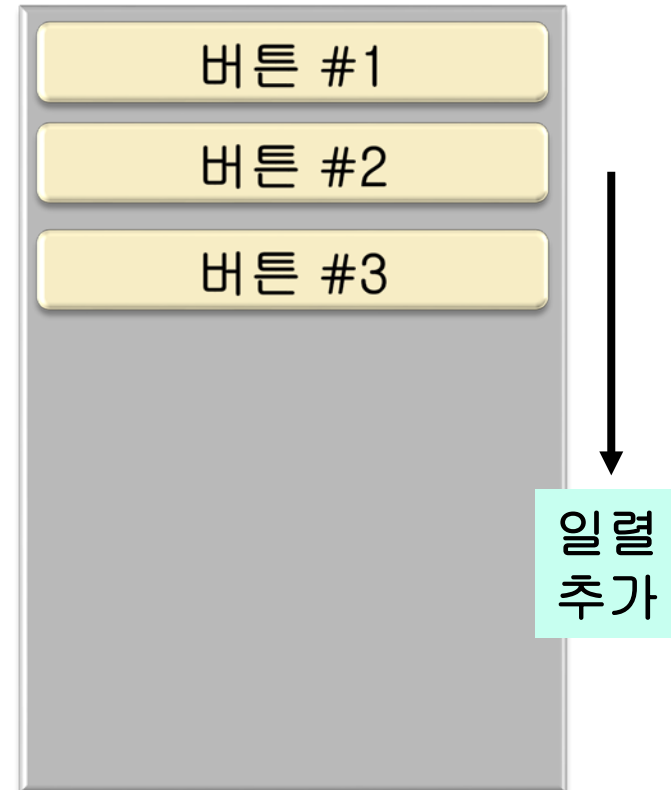
- Android 프로젝트를 생성하면 activity_main.xml 파일이 자동으로 만들어짐
- Android SDK 4.0.0이전의 버전에서는 기본적으로 LinearLayout으로 생성되었지만, 그 이후 버전부터는 RelativeLayout이 기본적으로 생성
- Child View를 수평, 수직으로 일렬 배치하는 Layout으로, 가장 단순하면서 직관적이며 사용빈도가 높음
- 기본값은 수평 정렬(Horizontal)
- LinearLayout에 붙이는 순서대로 배치
- 별도의 지정이 없는 한 View 사이의 공백은 없음

```
<LinearLayout>  
  ~내용~  
</LinearLayout>
```



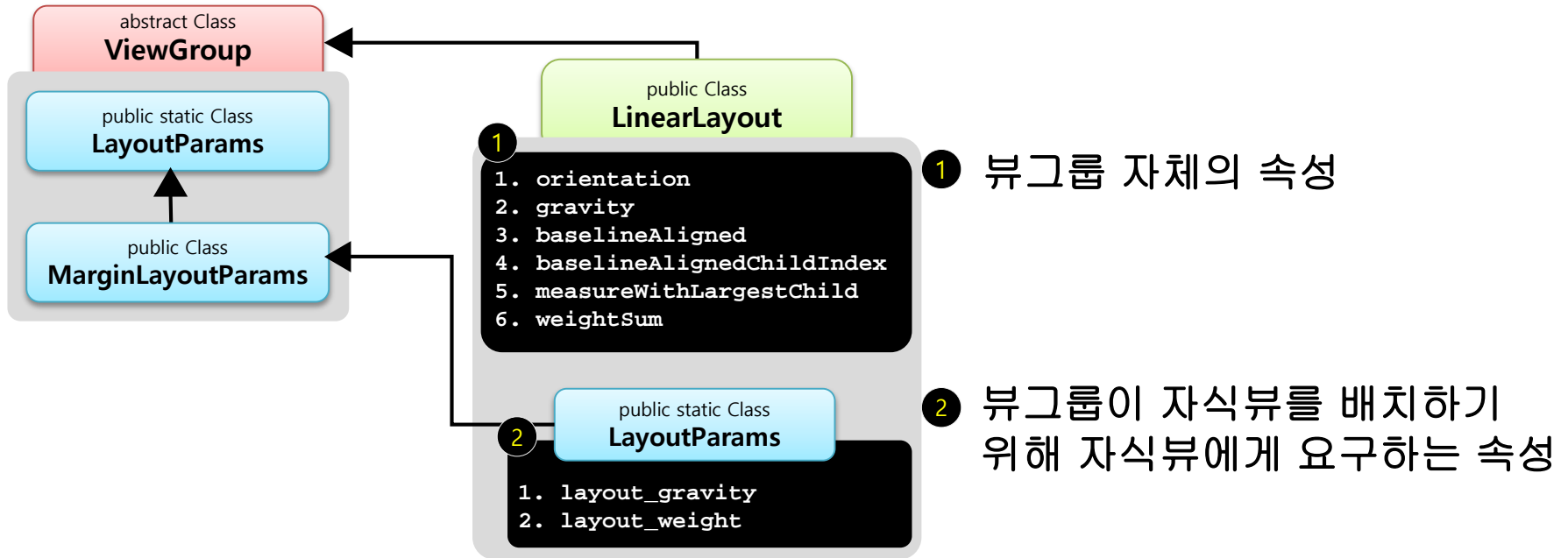
LinearLayout

- Box형 모델
- 사각형 영역들을 이용해 화면을 구성
- 표준 JAVA의 BoxLayout과 유사
- 중요한 속성으로는 방향 속성이 있음
 - 가로(Horizontal)
 - 세로(Vertical)
- 정해진 방향으로만 일렬로 추가 가능





LinearLayout 속성



뷰그룹은 자체 뷰그룹의 속성과 LayoutParams 속성을 함께 학습해야
정확한 원리를 알 수 있음



LinearLayout 속성

■ orientation

■ LinearLayout의 필수 속성 임

■ View의 배치 방향을 결정하는 속성 (디폴트는 horizontal)

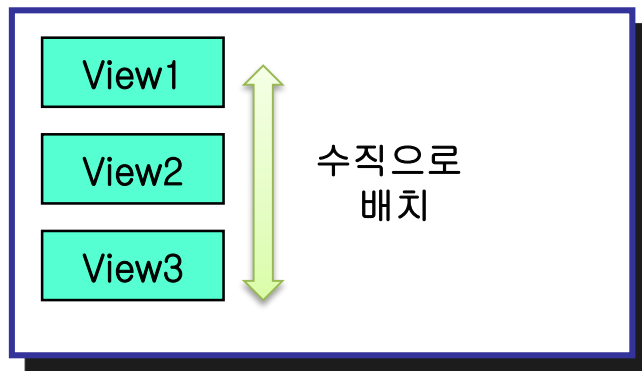
■ 속성값

■ vertical

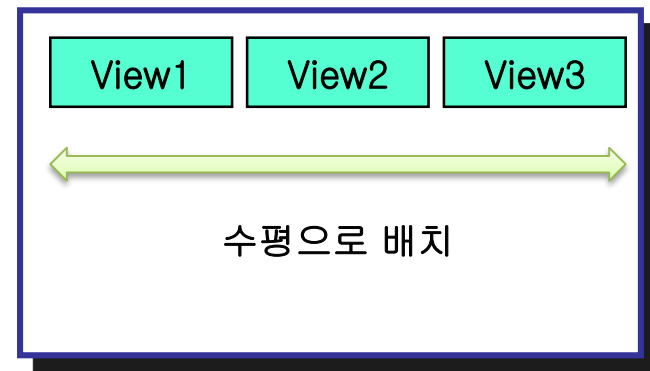
■ 자식 View를 위에서 아래로 수직으로 배열

■ horizontal

■ 자식 View를 왼쪽에서 오른쪽으로 수평 배열



[vertical]



[horizontal]



LinearLayout 속성

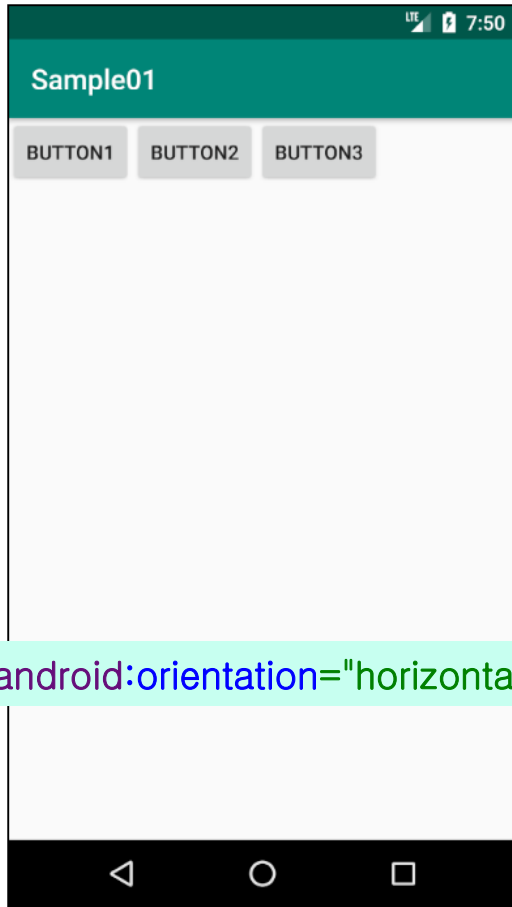
```
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:orientation="horizontal"
    tools:context=".MainActivity">
    <Button
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Button1"/>
    <Button
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Button2"/>
    <Button
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Button3"/>
</LinearLayout>
```

프로젝트를 처음 만들었을 때 만들어지는
LinearLayout은 세로 방향으로 되어 있음



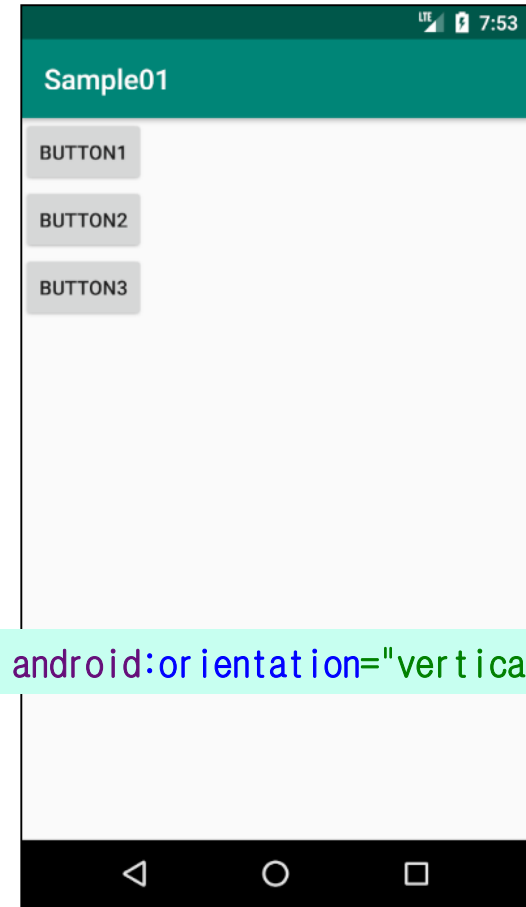
LinearLayout 속성

■ orientation



`android:orientation="horizontal"`

[가로 방향으로 설정을 바꾼 경우,
기본 설정]



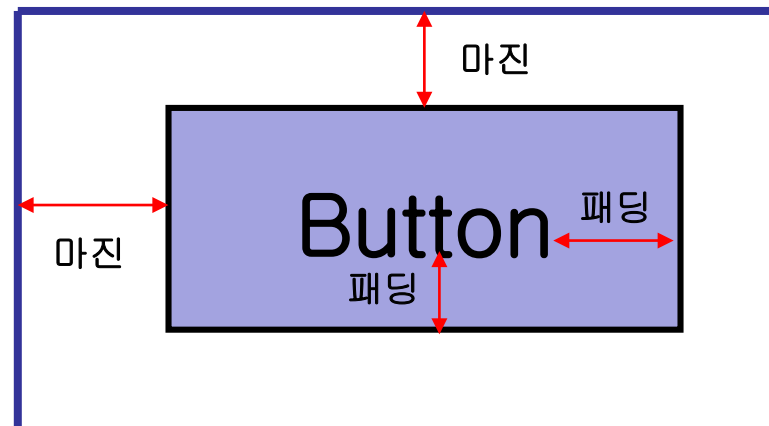
`android:orientation="vertical"`

[세로 방향으로 설정한 경우]



LinearLayout 속성

■ padding, margin



■ padding

- View와 내용물간의 간격 지정
- View의 입장에서 안쪽 여백을 뜻하며 View 자체 속성

■ margin

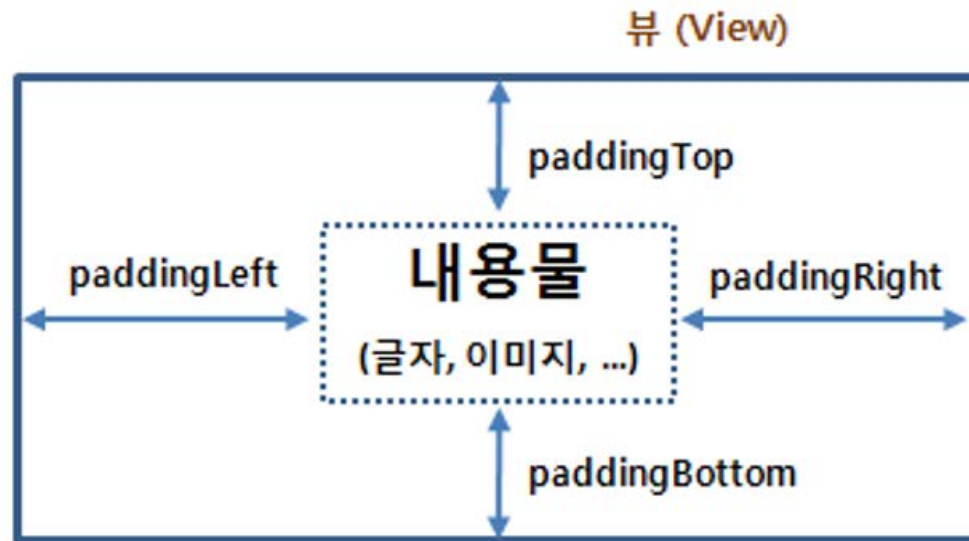
- View와 부모와의 간격을 지정하며, 근처에 형제 View가 있으면 동일한 간격으로 여백 생성
- View의 입장에서 바깥 여백을 뜻하며 Layout의 속성



LinearLayout 속성

■ padding

- View 안의 내용물인 텍스트나 이미지와 View 안의 영역 사이의 여백을 줄 수 있는 방법
- 4면 모두 동일한 여백이 지정되며, paddingLeft, paddingTop, paddingRight, paddingBottom으로 4면의 개별 여백 지정 가능



[padding을 이용한 뷰 내부의 여백 주기]



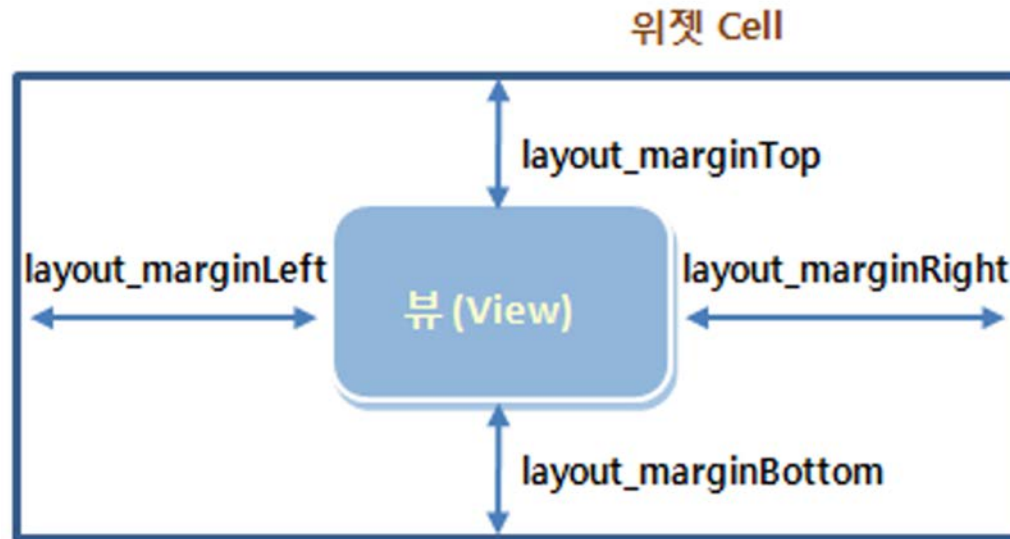
LinearLayout 속성

■ margin

- 부모 Container의 여유 공간과 View 사이의 여백을 줄 수 있는 방법

■ Widget Cell

- Widget이나 View들은 부모 컨테이너로부터 할당된 공간을 차지하게 되며 이를 'Widget Cell'이라고 부름



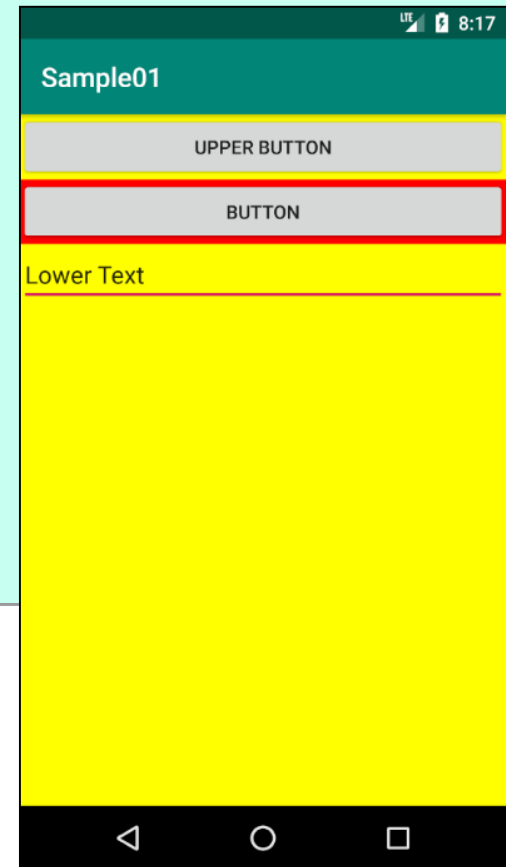
[layout_margin을 이용한 부모 여유공간과의 여백 주기]



LinearLayout 속성

```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:orientation="vertical"
    android:background="#ffFF00"
    tools:context=".MainActivity">

    <Button
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:text="Upper Button" />
```





LinearLayout 속성



```
<LinearLayout
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:background="#ff0000">

    <Button
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:text="Button" />
</LinearLayout>

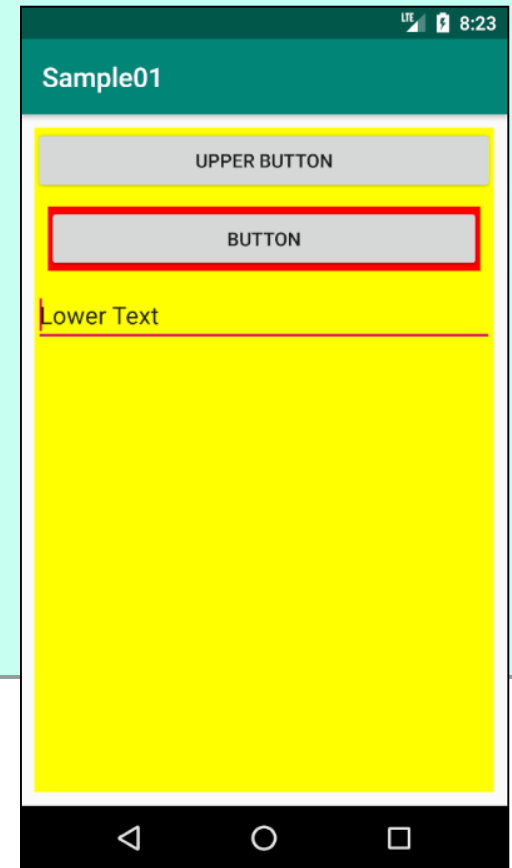
<EditText
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:text="Lower Text" />
</LinearLayout>
```



LinearLayout 속성

```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:orientation="vertical"
    android:background="#ffFF00"
    android:layout_margin="10dp"
    tools:context=".MainActivity">

    <Button
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:text="Upper Button" />
```





LinearLayout 속성



```
<LinearLayout
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:background="#ff0000"
    android:layout_margin="10dp">

    <Button
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:text="Button" />

</LinearLayout>

<EditText
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:text="Lower Text" />

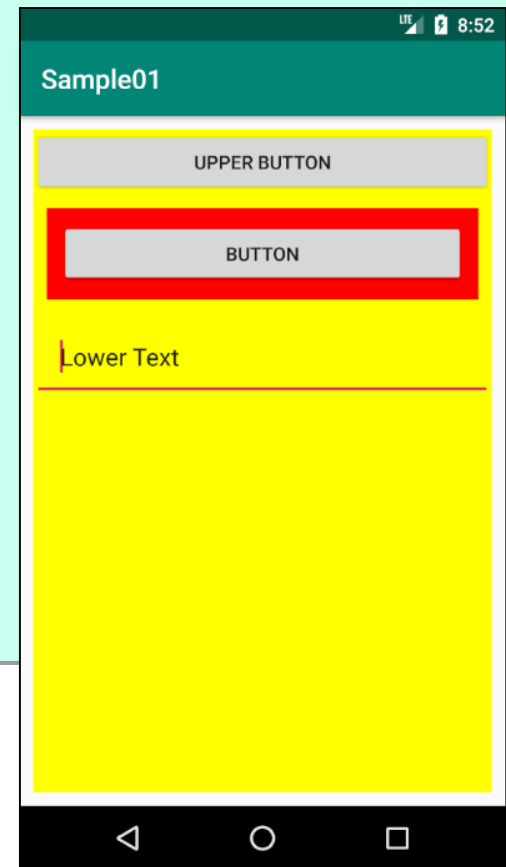
</LinearLayout>
```



LinearLayout 속성

```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:orientation="vertical"
    android:background="#ffFF00"
    android:layout_margin="10dp"
    tools:context=".MainActivity">

    <Button
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:text="Upper Button" />
```





LinearLayout 속성



```
<LinearLayout
```

```
    android:layout_width="match_parent"  
    android:layout_height="wrap_content"  
    android:background="#ff0000"  
    android:layout_margin="10dp "  
    android:padding="10dp">
```

```
    <Button
```

```
        android:layout_width="match_parent"  
        android:layout_height="wrap_content"  
        android:text="Button" />
```

```
</LinearLayout>
```

```
<EditText
```

```
    android:layout_width="match_parent"  
    android:layout_height="wrap_content "  
    android:padding="20dp "  
    android:text="Lower Text" />
```

```
</LinearLayout>
```



LinearLayout 속성

■ gravity

- Layout 내부의 Widget들을 어디에 배치할 것인지를 결정
- View에 존재하는 내용물의 수평, 수직 방향에 대해 각각 정렬 방식을 지정
- gravity 속성을 사용하기 위해서는 Container와 컨트롤들의 layout_height, layout_width 속성이 **wrap_content**가 **되야 함**

■ 속성값

- left, right, center, top, bottom 등을 사용할 수 있음
- 두 속성을 같이 지정할 때는 “|” **연산자를 이용**하며, 이 때 연산자 양쪽으로 공백이 전혀 없어야 함
- 예) 두 개를 조합한 ‘right|bottom’은 오른쪽 아래에 정렬한다는 의미



LinearLayout 속성

■ gravity 속성값

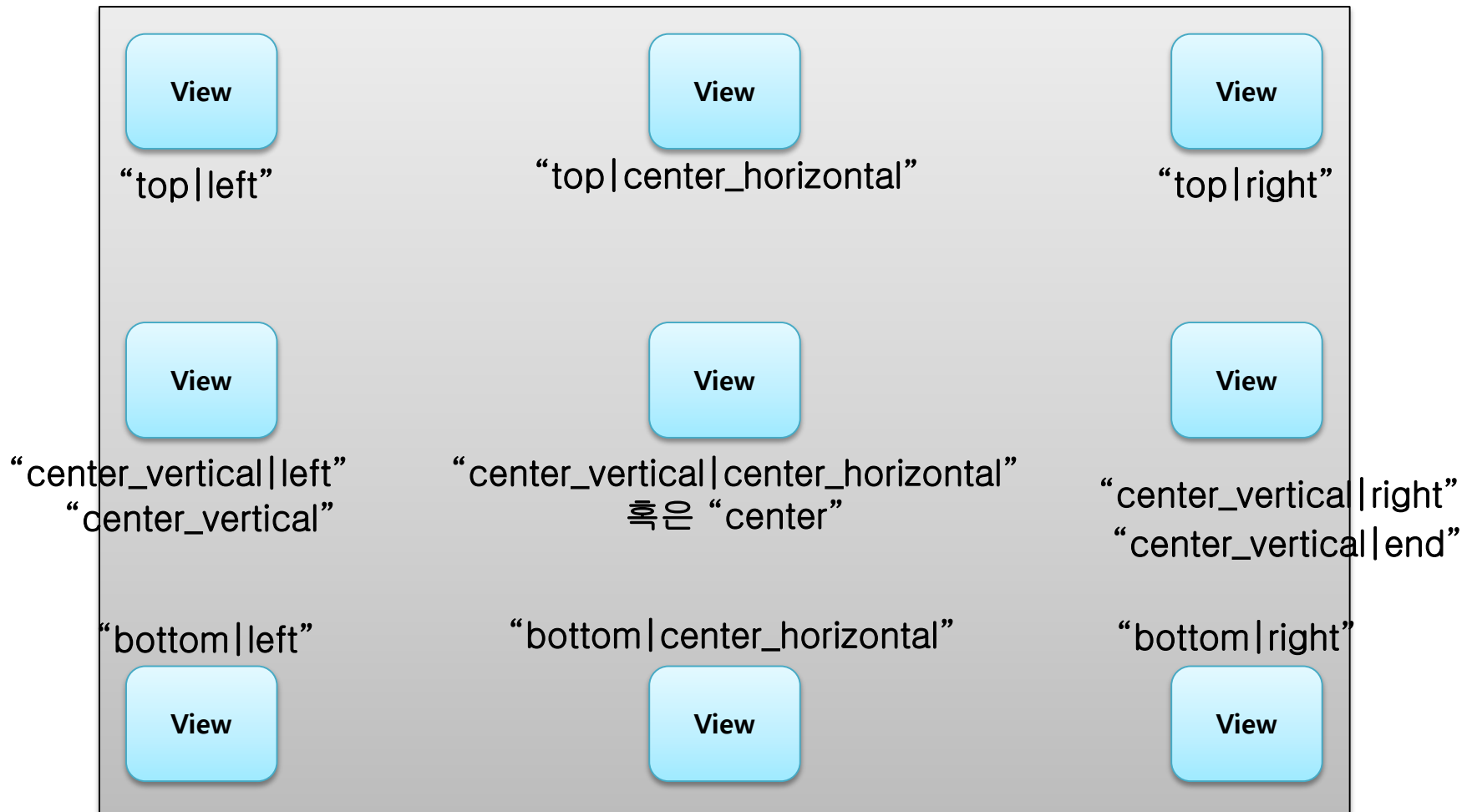
■ 각 정렬 방식은 비트 필드로 정의

상수	설명
center_horizontal	수평으로 중앙에 배치
left	컨테이너의 왼쪽에 배치하며, 크기는 바뀌지 않음
right	컨테이너의 오른쪽에 배치
fill_horizontal	수평 방향으로 가득 채움
center_vertical	수직으로 중앙에 배치
top	컨테이너의 상단에 배치하며, 크기는 바뀌지 않음
bottom	컨테이너의 하단에 배치
fill_vertical	수직 방향으로 가득 채움
center	수평으로나 수직으로 중앙에 배치
fill	컨테이너에 가득 채우도록 수직, 수평 크기를 확장



LinearLayout 속성

■ gravity 속성값





LinearLayout 속성

```
<?xml version="1.0" encoding="utf-8" ?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:orientation="vertical"
    tools:context=".MainActivity">

    <TextView
        android:layout_width="match_parent"
        android:layout_height="match_parent"
        android:text="정렬 테스트"
        android:textColor="#00ff00"
        android:textSize="40dp" />

</LinearLayout>
```





LinearLayout 속성

```
<?xml version="1.0" encoding="utf-8" ?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:orientation="vertical"
    tools:context=".MainActivity">

    <TextView
        android:layout_width="match_parent"
        android:layout_height="match_parent"
        android:text="정렬 테스트"
        android:gravity="center"
        android:textColor="#00ff00"
        android:textSize="40dp" />

</LinearLayout>
```





LinearLayout 속성

```
<?xml version="1.0" encoding="utf-8" ?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:orientation="vertical"
    tools:context=".MainActivity">

    <TextView
        android:layout_width="match_parent"
        android:layout_height="match_parent"
        android:text="정렬 테스트"
        android:gravity="center_vertical"
        android:textColor="#00ff00"
        android:textSize="40dp" />

</LinearLayout>
```



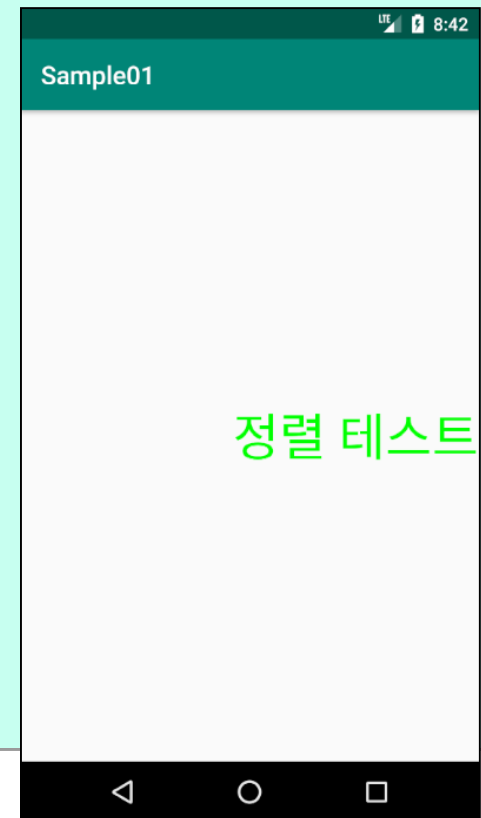


LinearLayout 속성

```
<?xml version="1.0" encoding="utf-8" ?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:orientation="vertical"
    tools:context=".MainActivity">

    <TextView
        android:layout_width="match_parent"
        android:layout_height="match_parent"
        android:text="정렬 테스트"
        android:gravity="center_vertical|right"
        android:textColor="#00ff00"
        android:textSize="40dp" />

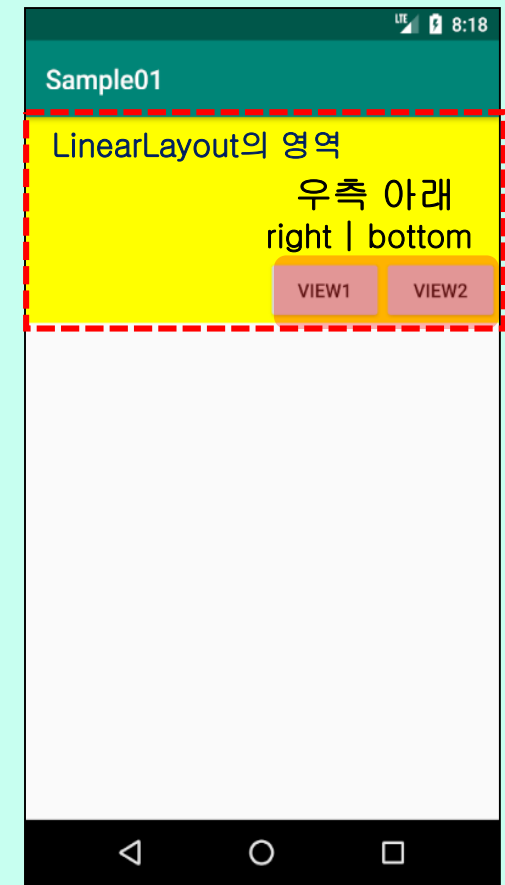
</LinearLayout>
```





LinearLayout 속성

```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="150dp"
    android:orientation="horizontal"
    android:background="#FFFF00"
    android:gravity="bottom|right"
    tools:context=".MainActivity">
    <Button
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="View1"/>
    <Button
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="View2"/>
</LinearLayout>
```



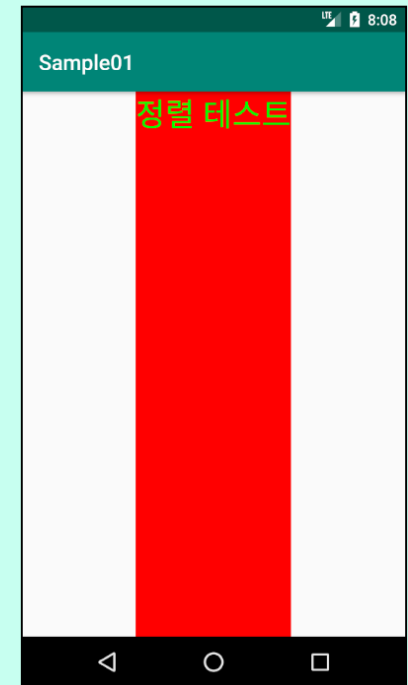


LinearLayout 속성

```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:gravity="center"
    tools:context=".MainActivity">
```

```
    <TextView
        android:layout_width="wrap_content"
        android:layout_height="match_parent"
        android:background="#ff0000"
        android:text="정렬 테스트"
        android:textColor="#00ff00"
        android:textSize="30dp" />
```

```
</LinearLayout>
```



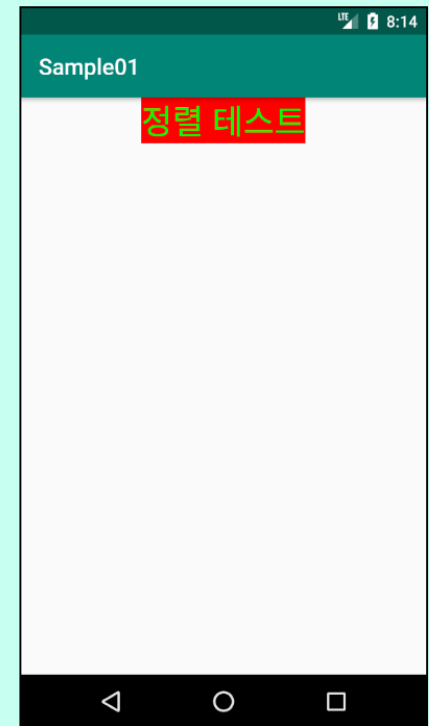


LinearLayout 속성

```
<?xml version="1.0" encoding="utf-8" ?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:gravity="center"
    tools:context=".MainActivity">

    <TextView
        android:layout_width="wrap_content"
        android:layout_height="match_parent"
        android:background="#ff0000"
        android:text="정렬 테스트"
        android:textColor="#00ff00"
        android:textSize="30dp" />

</LinearLayout>
```



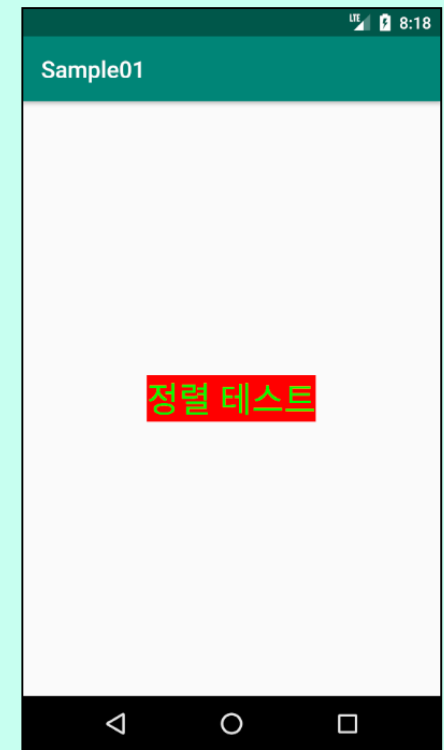


LinearLayout 속성

```
<?xml version="1.0" encoding="utf-8" ?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:gravity="center"
    tools:context=".MainActivity">

    <TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:background="#ff0000"
        android:text="정렬 테스트"
        android:textColor="#00ff00"
        android:textSize="30dp" />

</LinearLayout>
```





LinearLayout 속성

```
<?xml version="1.0" encoding="utf-8" ?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:gravity="center"
    android:orientation="vertical"
    tools:context=".MainActivity">
    <TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:background="#ff0000"
        android:text="정렬 테스트"
        android:textColor="#00ff00"
        android:textSize="30dp" />
    <Button
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="버튼이다." />
</LinearLayout>
```



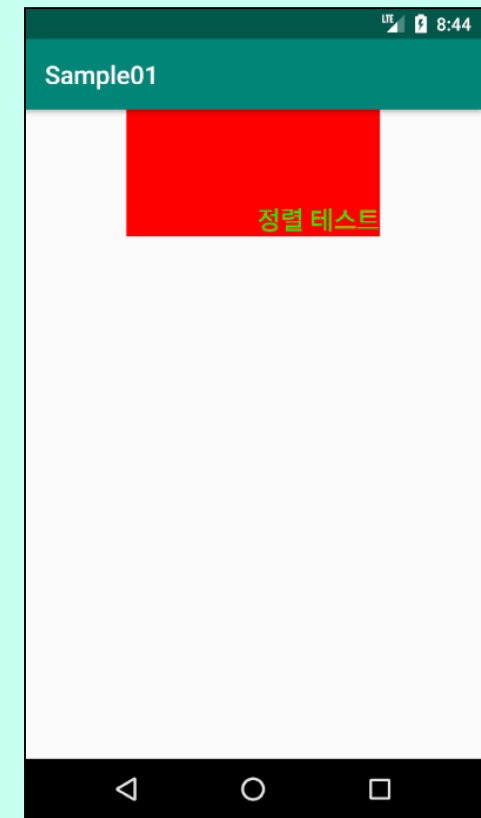


LinearLayout 속성

```
<?xml version="1.0" encoding="utf-8" ?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:orientation="vertical"
    tools:context=".MainActivity">

    <TextView
        android:layout_width="200dp"
        android:layout_height="100dp"
        android:layout_gravity="center_horizontal"
        android:background="#ff0000"
        android:gravity="right|bottom"
        android:text="정렬 테스트"
        android:textColor="#00ff00"
        android:textSize="20dp" />

</LinearLayout>
```





LinearLayout 속성

```
<?xml version="1.0" encoding="utf-8" ?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:gravity="center"
    android:orientation="vertical"
    tools:context=".MainActivity">
    <TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:background="#ff0000"
        android:text="정렬 테스트"
        android:textColor="#00ff00"
        android:textSize="30dp" />
    <Button
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="버튼이다." />
</LinearLayout>
```

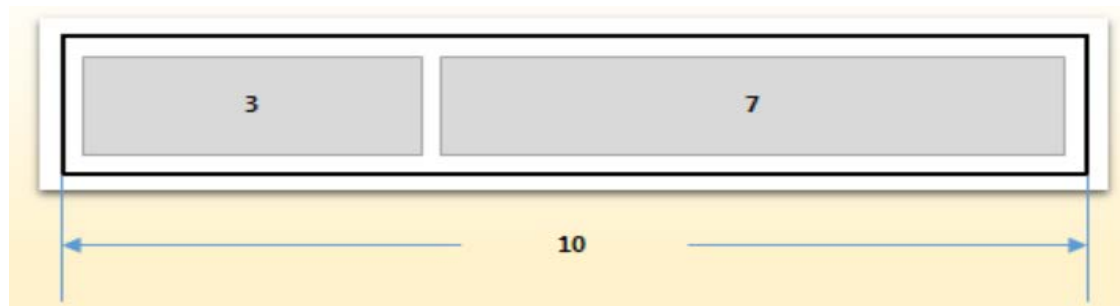




LinearLayout 속성

■ layout_weight

- View가 차지하는 공간의 가중치 값을 설정
- LinearLayout은 자식(Children)들이 배치될 때, 각 View Widget들이 차지하는 영역을 dp 등의 고정 치수 단위 뿐만 아니라, 전체 영역 대비 비율의 개념으로 지정할 수 있는 weight(가중치) 설정 기능을 제공
- weight가 0이면 자신의 고유한 크기만큼, 1이상이면 형제 View의 비율에 따라 부모의 영역을 균등하게 배분
- 예) 전체 크기를 10으로 본다면, 첫 번째 Widget은 3, 두 번째 Widget은 7의 영역을 차지하도록 만들 수 있음





LinearLayout 속성



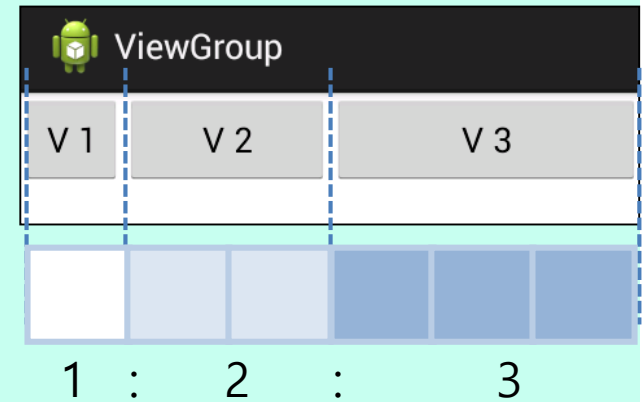
■ weight

- weight는 비율을 주는 속성인데 width나 height중 비율을 주고 싶은 속성에 0dp를 할당해야 함
- 설정하지 않으면 디폴트로 0으로 설정
- weight가 개별 View Widget의 가중치를 설정하는 값이라 할지라도, 그 값이 전체 영역 대비 %(Percentage)를 의미하거나, 전체 영역이 10 또는 100으로 고정된 것은 아니라는 것에 주의
- 자식(Children)들 간의 상대적인 값으로 적용된다는 것
- 예) 두 개의 View Widget이 각각 1, 2의 Weight 값을 가지면, 첫 번째는 1/3만큼, 두 번째는 2/3만큼의 영역을 차지하게 됨



LinearLayout 속성

```
<LinearLayout xmlns:android="..."
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:orientation="horizontal">
    <Button android:layout_width="0dp"
        android:layout_height="wrap_content"
        android:layout_weight="1"
        android:text="V 1"/>
    <Button android:layout_width="0dp"
        android:layout_height="wrap_content"
        android:layout_weight="2"
        android:text="V 2" />
    <Button android:layout_width="0dp"
        android:layout_height="wrap_content"
        android:layout_weight="3"
        android:text="V 3" />
</LinearLayout>
```





LinearLayout 속성

```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:orientation="vertical"
    tools:context=".MainActivity">
```

<Button

```
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_weight="0"
    android:text="위쪽 버튼"/>
```

<EditText

```
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_weight="1"
    android:hint="가운데 에디트"/>
```





LinearLayout 속성



```
<Button  
    android:layout_width="match_parent"  
    android:layout_height="wrap_content"  
    android:layout_weight="0"  
    android:text="아래 버튼"/>  
</LinearLayout>
```



LinearLayout 속성

```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:orientation="vertical"
    tools:context=".MainActivity">
```

<Button

```
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_weight= "1"
    android:text="위쪽 버튼"/>
```

<EditText

```
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_weight= "3"
    android:hint="가운데 에디트"/>
```





LinearLayout 속성

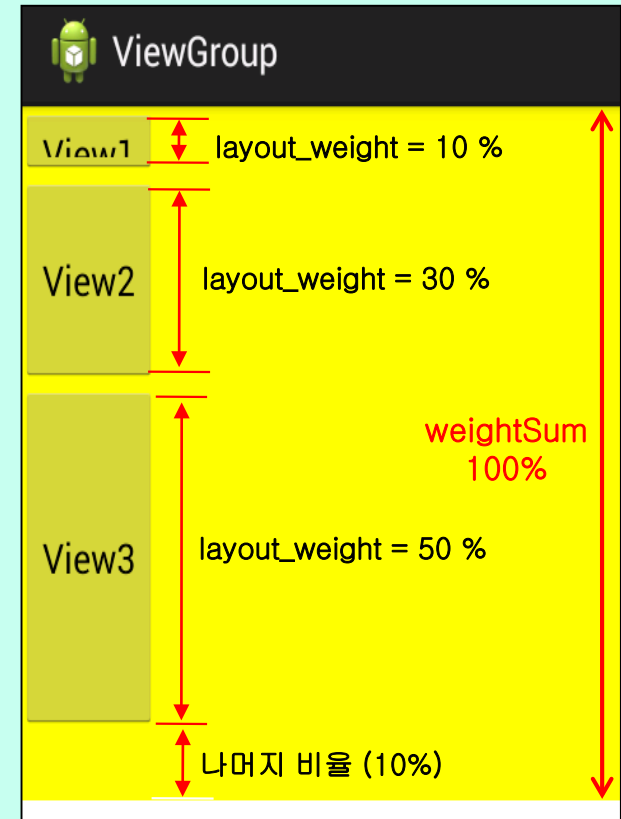


```
<Button  
    android:layout_width="match_parent"  
    android:layout_height="wrap_content"  
    android:layout_weight="1"  
    android:text="아래 버튼"/>  
</LinearLayout>
```



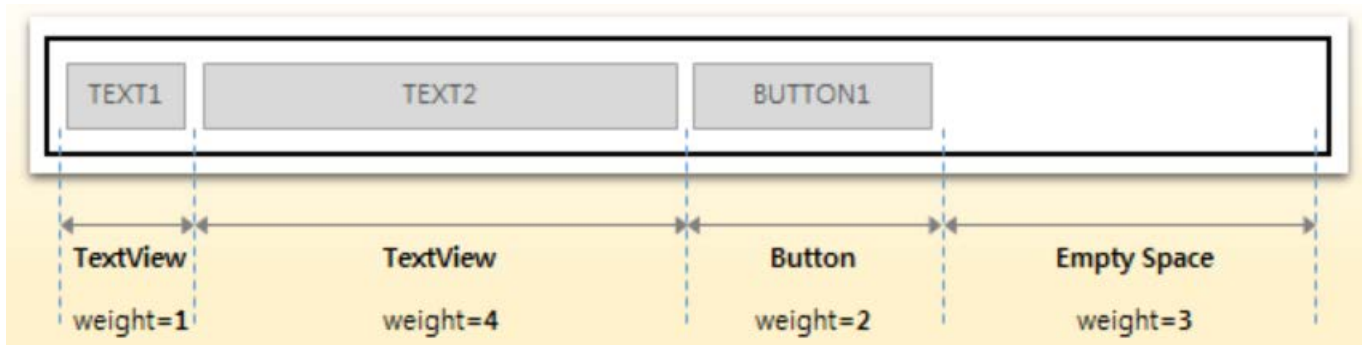

LinearLayout 속성

```
<LinearLayout xmlns:android="..."
    android:layout_width="match_parent"
    android:layout_height="300dp"
    android:background="#FF0"
    android:orientation="vertical"
    android:weightSum="100">
    <Button
        android:layout_width="wrap_content"
        android:layout_height="0dp"
        android:layout_weight="10"
        android:text="View1"/>
    <Button
        android:layout_width="wrap_content"
        android:layout_height="0dp"
        android:layout_weight="30"
        android:text="View2"/>
    <Button
        android:layout_width="wrap_content"
        android:layout_height="0dp"
        android:layout_weight="50"
        android:text="View3"/>
</LinearLayout>
```





LinearLayout 속성

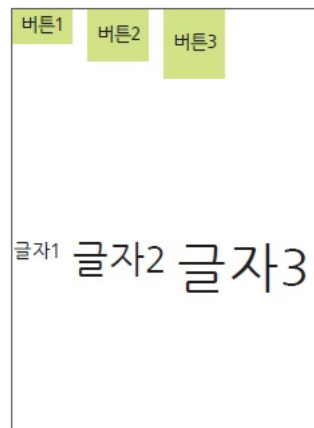




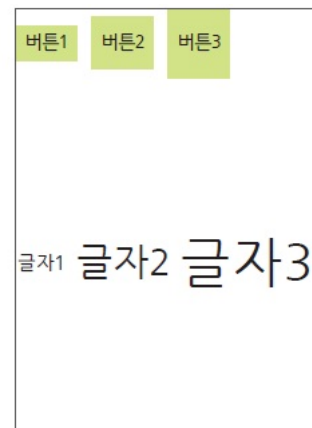
LinearLayout 속성

■ baselineAligned

- baselineAligned 속성은 크기가 다른 Widget들을 보기 좋게 정렬함
- 크기가 다른 TextView들이 수평으로 배치되었을 때는 텍스트의 **하단을 기준으로 정렬하는 것이 보기가 좋기 때문**
- 높이가 다른 자식 뷰를 수평으로 정렬 시 하단 정렬 지정 (디폴트 true)
- true와 false 값을 가질 수 있음



(a) false로 지정



(b) 생략하거나 true로 지정

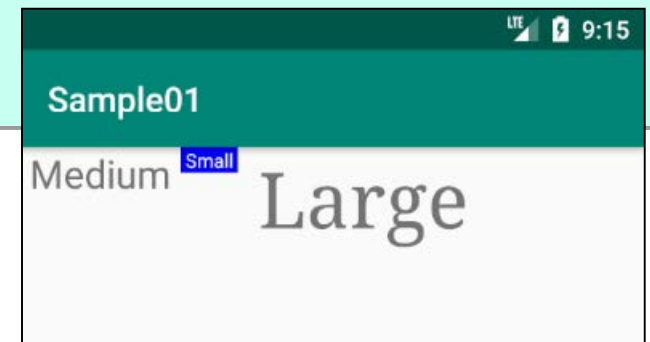
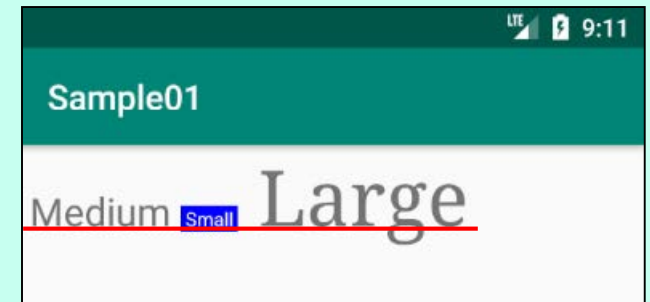
그림 5-3 baselineAligned 속성



LinearLayout 속성

```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:orientation="horizontal"
    android:baselineAligned="false"
    tools:context=".MainActivity">

    <TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text=" Medium "
        android:textSize="10pt" />
```





LinearLayout 속성



```
<TextView
```

```
    android:layout_width="wrap_content"  
    android:layout_height="wrap_content"  
    android:background="#0000ff"  
    android:text=" Small "  
    android:textColor="#FFFFFF"  
    android:textSize="5pt" />
```

```
<TextView
```

```
    android:layout_width="wrap_content"  
    android:layout_height="wrap_content"  
    android:text=" Large "  
    android:textSize="20pt"  
    android:typeface="serif" />
```

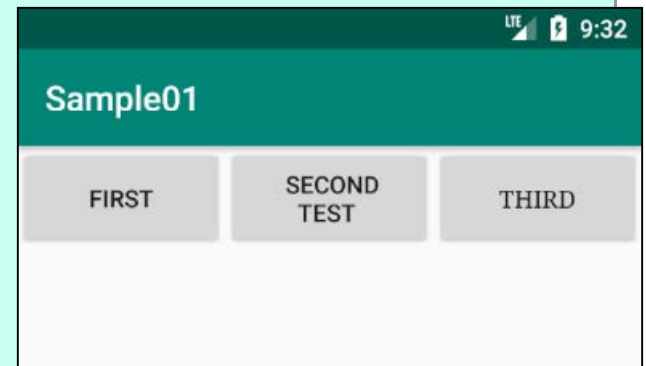
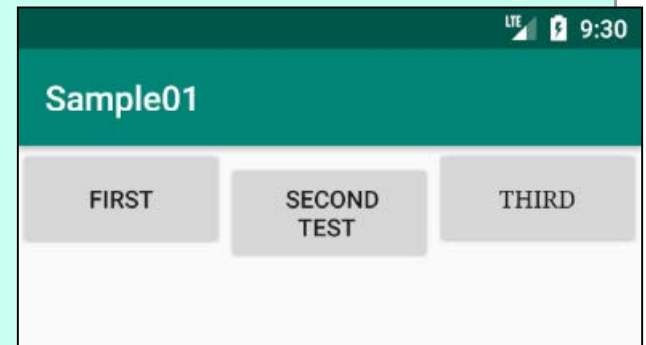
```
</LinearLayout>
```



LinearLayout 속성

```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:orientation="horizontal"
    android:baselineAligned="false"
    tools:context=".MainActivity">
```

```
    <Button
        android:layout_width="0dp"
        android:layout_height="60dp"
        android:layout_weight="1"
        android:text=" First " />
    <Button
        android:layout_width="0dp"
        android:layout_height="60dp"
        android:layout_weight="1"
        android:text=" SecondWnTest " />>
```





LinearLayout 속성



```
<Button  
    android:layout_width="0dp"  
    android:layout_height="60dp"  
    android:layout_weight="1"  
    android:text=" Third "  
    android:typeface="serif" />  
</LinearLayout>
```



LinearLayout 속성

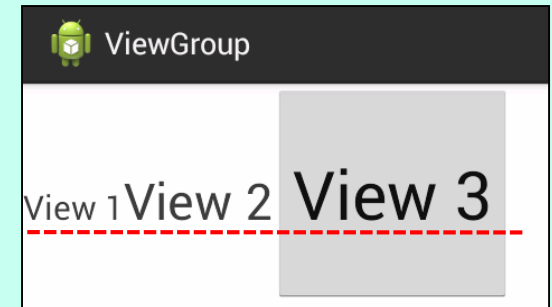
```
<LinearLayout xmlns:android="..."  
    android:layout_width="match_parent"  
    android:layout_height="130dp"  
    android:orientation="horizontal"  
    android:baselineAligned="true">
```

```
<TextView android:layout_width="wrap_content"  
    android:layout_height="wrap_content"  
    android:textSize="20dp"  
    android:text="View 1"/>
```

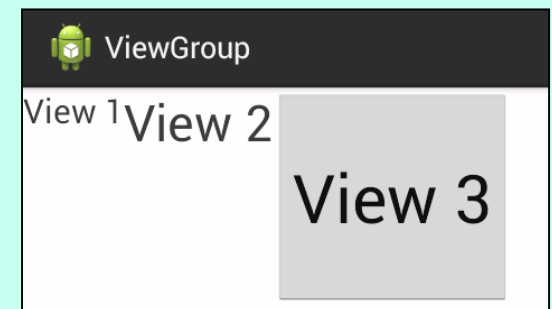
```
<TextView android:layout_width="wrap_content"  
    android:layout_height="wrap_content"  
    android:textSize="30dp"  
    android:text="View 2" />
```

```
<Button android:layout_width="wrap_content"  
    android:layout_height="130dp"  
    android:textSize="40dp"  
    android:text="View 3" />
```

```
</LinearLayout>
```



baselineAligned = *"true"*



baselineAligned = *"false"*



LinearLayout 속성

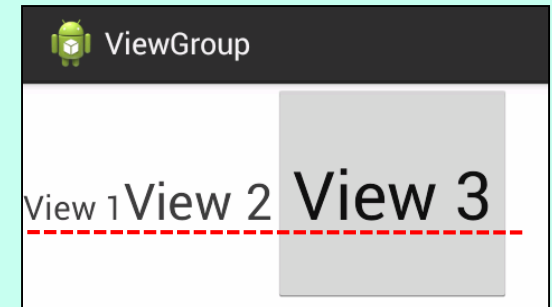
```
<LinearLayout xmlns:android="..."  
    android:layout_width="match_parent"  
    android:layout_height="130dp"  
    android:orientation="horizontal"  
    android:baselineAligned="true">
```

```
<TextView android:layout_width="wrap_content"  
    android:layout_height="wrap_content"  
    android:textSize="20dp"  
    android:text="View 1"/>
```

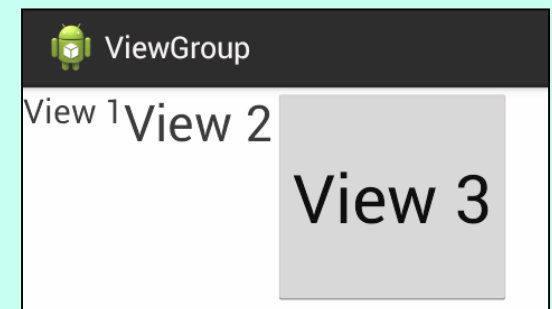
```
<TextView android:layout_width="wrap_content"  
    android:layout_height="wrap_content"  
    android:textSize="30dp"  
    android:text="View 2" />
```

```
<Button android:layout_width="wrap_content"  
    android:layout_height="130dp"  
    android:textSize="40dp"  
    android:text="View 3" />
```

```
</LinearLayout>
```



baselineAligned = *"true"*



baselineAligned = *"false"*



LinearLayout 속성



- baselineAlignedChildIndex

- 하위 메뉴의 위젯 중 몇 번째 위젯에 정렬을 맞출 지 결정
- 0은 첫 번째, 1은 2 번째...



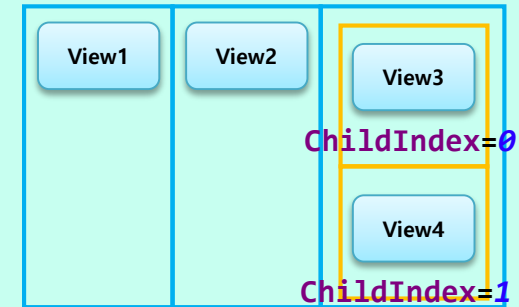
LinearLayout 속성

■ baselineAligned와 baselineAlignedChildIndex

```
<LinearLayout xmlns:android="....."  
    android:layout_width="match_parent"  
    android:layout_height="160dp"  
    android:orientation="horizontal"  
    android:baselineAligned="true">
```

```
<Button  
    android:layout_width="wrap_content"  
    android:layout_height="40dp"  
    android:text="View1"/>
```

```
<Button  
    android:layout_width="wrap_content"  
    android:layout_height="60dp"  
    android:text="View2" />
```





LinearLayout 속성

■ baselineAligned와 baselineAlignedChildIndex

<LinearLayout

```
android:layout_width="wrap_content"  
android:layout_height="match_parent"  
android:orientation="vertical"  
android:baselineAlignedChildIndex="1">
```

<Button

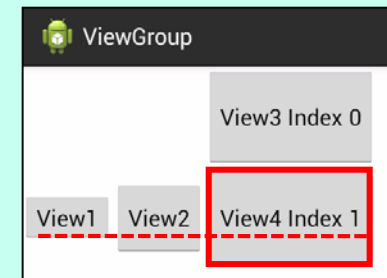
```
android:layout_width="wrap_content"  
android:layout_height="80dp"  
android:text="View3 Index 0" />
```

<Button

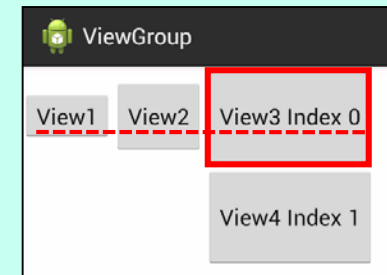
```
android:layout_width="wrap_content"  
android:layout_height="80dp"  
android:text="View4 Index 1" />
```

</LinearLayout>

</LinearLayout>



baselineAlignedChildIndex="1"



baselineAlignedChildIndex="0"



LinearLayout 속성



■ weightSum

- Android는 디바이스 해상도가 통일되어 있지 않은 관계로 절대 좌표를 사용하여 Widget을 배치할 수가 없음
- 화면을 구성하는 layout을 만들 때 상대적인 크기를 가지고 배치해야 할 경우가 많은데, 이 경우 weightSum 속성을 이용하면 편리
- LinearLayout의 최대 가중치(weight) 값 지정
- 속성값
 - 소수점 단위 사용 가능 (예) 1.2)
 - 지정되지 않을 시, 합산된(sum) 가중치는 자식 View들의 모든 layout_weight값을 더하여 계산
- weightSum은 LinearLayout 또는 FrameLayout에서만 사용할 수 있으며 RelativeLayout에서는 사용할 수 없음



LinearLayout 속성



■ measureWithLargestChild

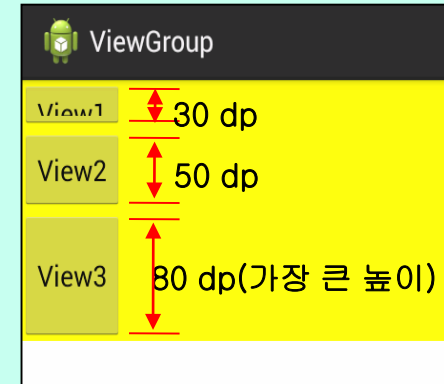
- 수평 배치 경우 길이 조정, 수직 배치 경우 높이 조정
- 가장 큰 넓이의 자식 뷰로 크기를 맞춤
- 속성값
 - true: 크기 맞추기
 - false: 크기 맞추지 않기

■ 예) android:measureWithLargestChild="false"

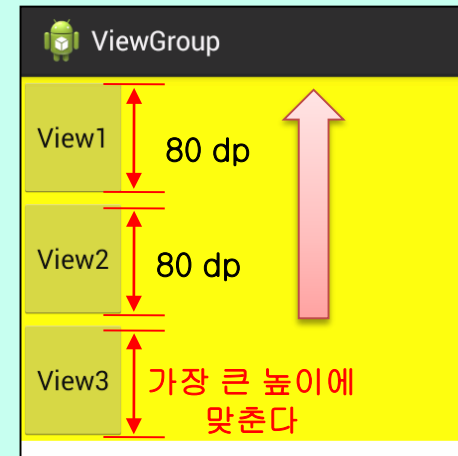


LinearLayout 속성

```
<LinearLayout xmlns:android=""  
    android:layout_width="match_parent"  
    android:layout_height="wrap_content"  
    android:orientation="vertical"  
    android:background="#FF0"  
    android:measureWithLargestChild="false">  
    <Button android:layout_width="wrap_content"  
        android:layout_height="30dp"  
        android:layout_weight="1"  
        android:text="View1"/>  
    <Button android:layout_width="wrap_content"  
        android:layout_height="50dp"  
        android:layout_weight="1"  
        android:text="View2"/>  
    <Button android:layout_width="wrap_content"  
        android:layout_height="80dp"  
        android:layout_weight="1"  
        android:text="View3"/>  
</LinearLayout>
```



android:measureWithLargestChild="false"

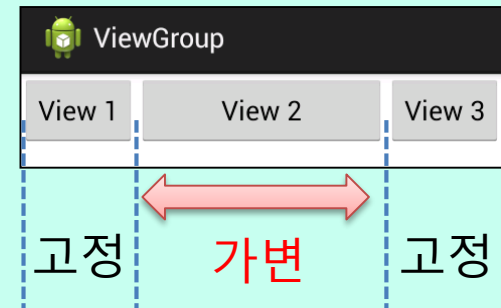


android:measureWithLargestChild="true"



LinearLayout 속성

```
<LinearLayout xmlns:android="..."
    android:layout_width="match_parent"
    android:layout_height="130dp"
    android:orientation="horizontal">
    <Button android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="View 1"/>
    <Button android:layout_width="0dp"
        android:layout_height="wrap_content"
        android:layout_weight="1"
        android:text="View 2"/>
    <Button android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="View 3"/>
</LinearLayout>
```



단말기를 가로로 회전했을 때



LinearLayout LayoutParams의 layout_weight 속성은 특정 자식뷰의 크기를 가변적으로 조절할 수 있기 때문에 다양한 화면 크기의 단말에서 유연하게 레이아웃을 유지할 수 있음



LinearLayout 속성

■ 두 가지 정렬 속성

정렬 속성	설 명
layout_gravity	[외부] 부모 컨테이너의 여유 공간에 뷰가 모두 채워지지 않아 여유 공간 안에서 뷰를 정렬할 때 뷰의 layout_width나 layout_height 속성이 match_parent가 아닐 경우에 같이 사용할 수 있음
gravity	[내부] 뷰에서 화면에 표시하는 내용물을 정렬할 때 (텍스트 뷰의 경우, 내용물은 글자가 되고 이미지 뷰의 경우 내용물은 이미지가 됨)



LinearLayout 속성



■ layout_gravity

- 뷰 자체를 부모 뷰의 어디에 둘 것인지를 결정
- layout_gravity는 자신의 위치를 부모의 어디쯤에 위치시킬지를 결정
- layout_gravity는 위젯에 많이 지정함
- 두 속성을 같이 지정할 때는“|”연산자를 이용하며, 이 때 연산자 양쪽으로 공백이 전혀 없어야 함



LinearLayout 속성



■ layout_gravity

■ 각 정렬 방식은 비트 필드로 정의

상수	설명
center_horizontal	수평으로 중앙에 배치
left	컨테이너의 왼쪽에 배치하며, 크기는 바뀌지 않음
right	컨테이너의 오른쪽에 배치
fill_horizontal	수평 방향으로 가득 채움
center_vertical	수직으로 중앙에 배치
top	컨테이너의 상단에 배치하며, 크기는 바뀌지 않음
bottom	컨테이너의 하단에 배치
fill_vertical	수직 방향으로 가득 채움
center	수평으로나 수직으로 중앙에 배치
fill	컨테이너에 가득 채우도록 수직, 수평 크기를 확장



LinearLayout 속성

res/layout/linearlayout_layout_gravity.xml

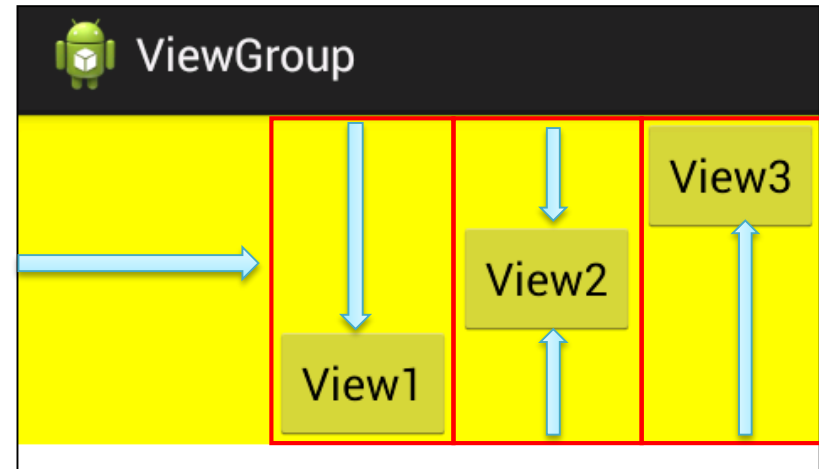
```
<LinearLayout xmlns:android="..."
    android:layout_width="match_parent"
    android:layout_height="130dp"
    android:orientation="horizontal"
    android:background="#FF0"
    android:gravity="right">

    <Button android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_gravity="bottom"
        android:text="View1"/>

    <Button android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_gravity="center_vertical"
        android:text="View2" />

    <Button android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_gravity="top"
        android:text="View3" />

</LinearLayout>
```

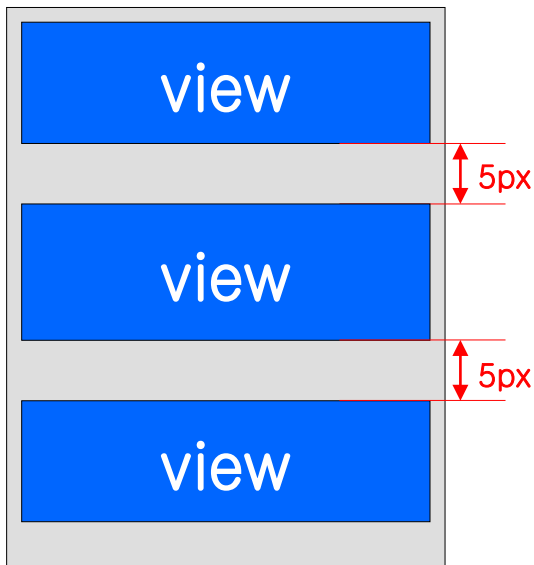




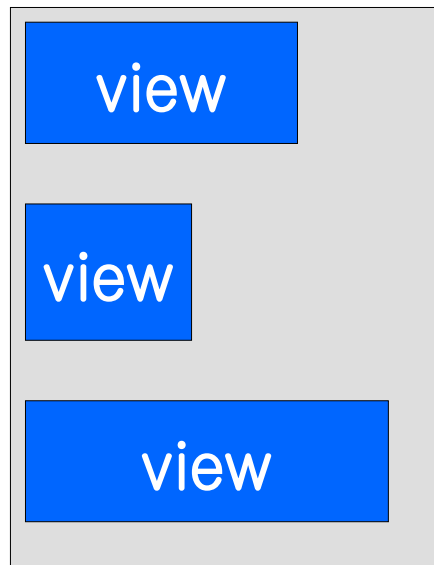
LinearLayout

- 모든 하위 구성 요소를 수직(vertical) 또는 수평(Horizontal) 형태 중 하나로 정렬하여 그리는 방식

viewgroup : [LinearLayout](#)



margin 설정 : 5px



gravity 설정 : 좌우정렬

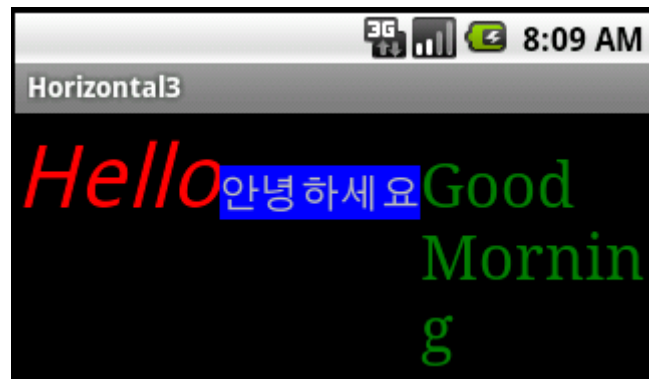
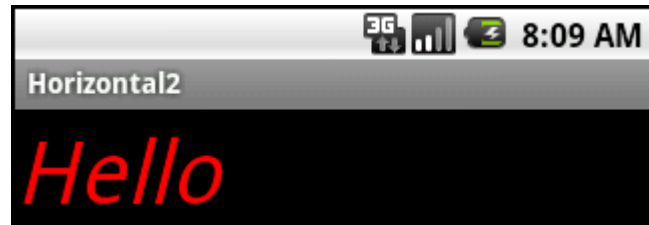
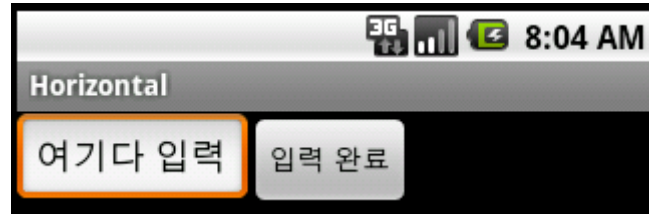


weight 설정 : 가중치 값



LinearLayout 예제

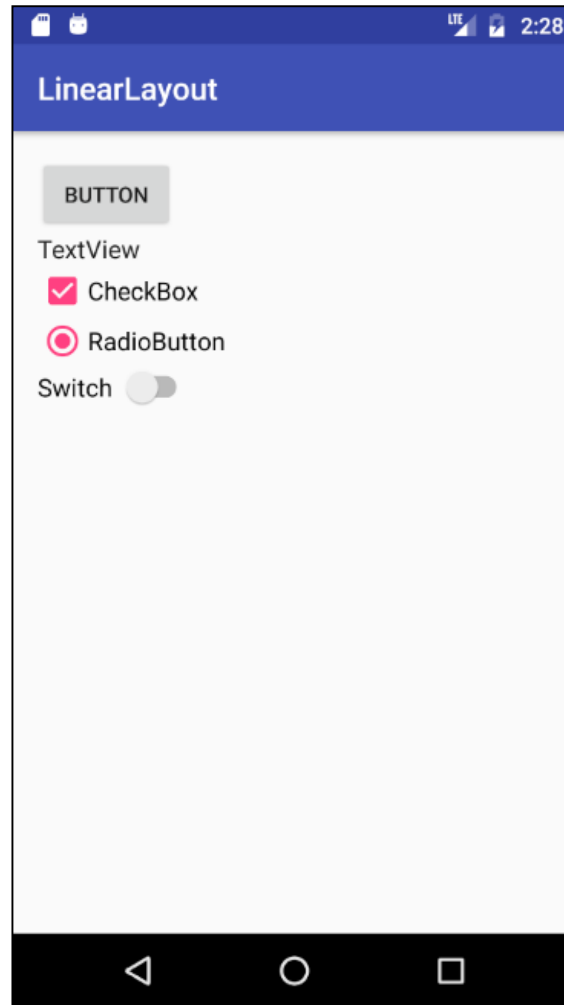
■ LinearLayout





LinearLayout 예제

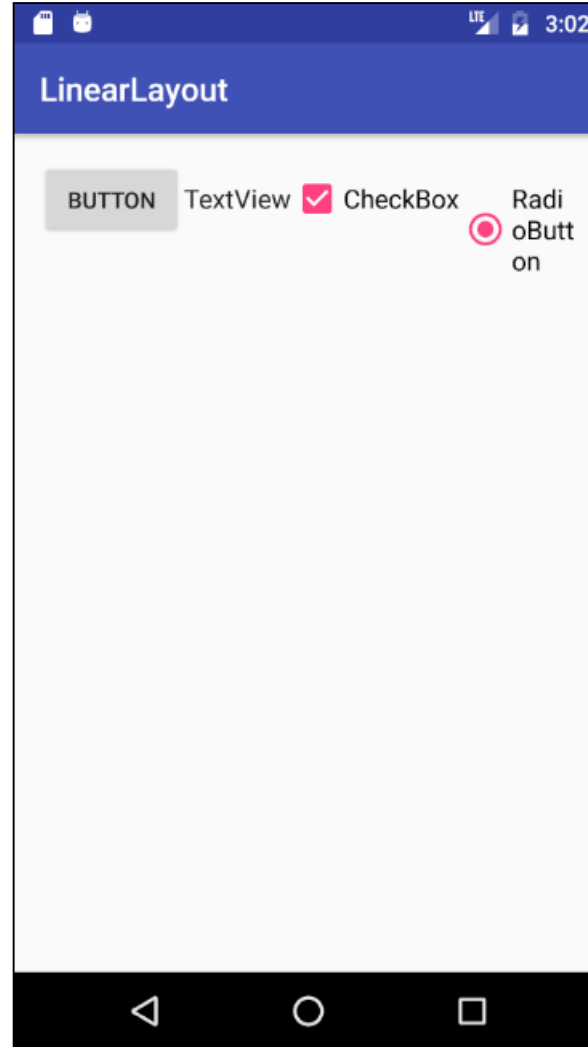
- orientation 속성이 'vertical' 값인 XML 코드





LinearLayout 예제

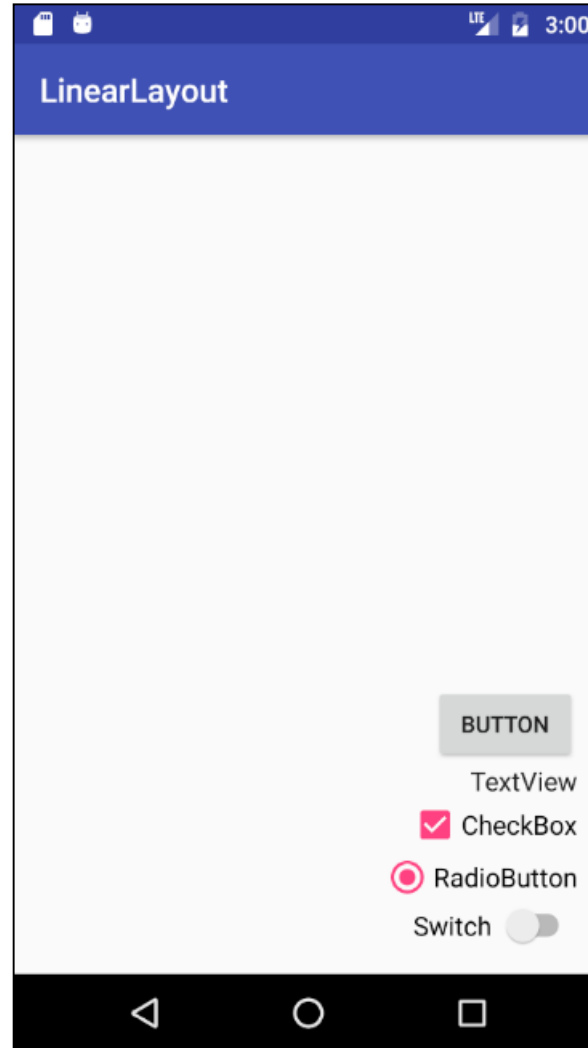
- orientation 속성이 'horizontal' 값인 XML 코드





LinearLayout 예제

- gravity 속성이 'right|bottom' 값인 XML





LinearLayout 예제

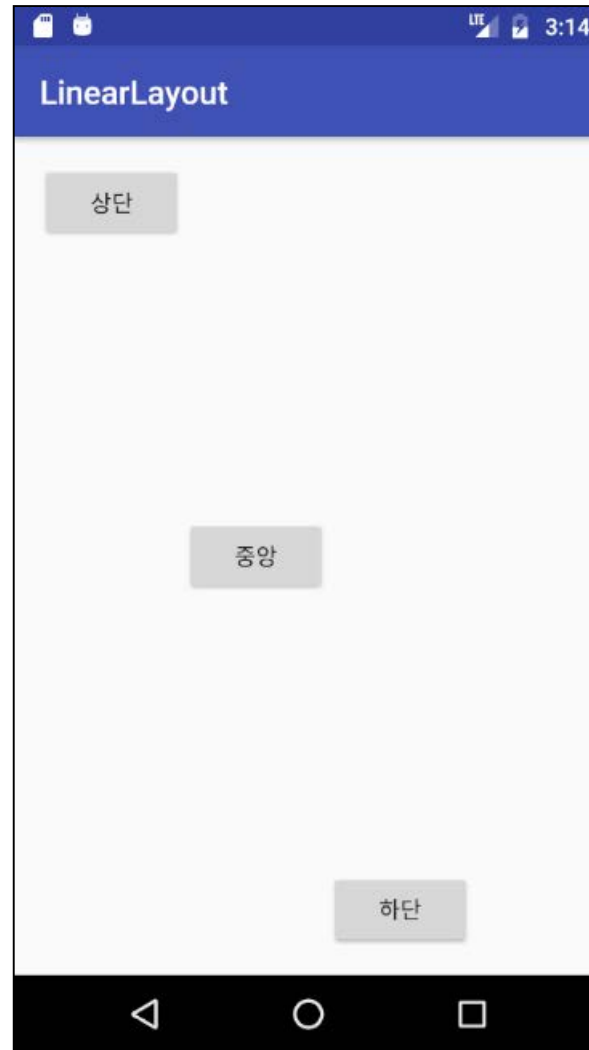
- layout_gravity 속성 (orientation = “vertical”인 경우)





LinearLayout 예제

- layout_gravity 속성 (orientation = “horizontal”인 경우)





LinearLayout 예제

■ baselineAligned



[true]



[false]

Layout/base1.xml ~ base2.xml

```
<?xml version="1.0" encoding="utf-8">
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:orientation="horizontal"
    android:layout_width="fill_parent"
    android:layout_height="fill_parent"
    android:baselineAligned="ture" >
    <TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Medium"
        android:textSize="10pt"/>
    <TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Small"
        android:textSize="5pt"
        android:background="#0000ff"/>
    <TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Large"
        android:textSize="20pt"
        android:typeface="serif"/>
</LinearLayout>
```



리니어 레이아웃의 기본 속성

- 여유 공간 (외부/내부) : `layout_margin`, `padding`
 - 뷰의 여유 공간을 어떻게 할 것인지 설정
- 공간가중치 (분할) : `layout_weight`
 - 공간가중치는 같은 부모 뷰에 포함되어 있는 뷰들이 여유 공간을 얼마나 차지할 수 있는지를 비율로 지정한 것



리니어 레이아웃-공간 가중치 설정

<TextView

```
android:id="@+id/button01"  
android:layout_width="wrap_content"  
android:layout_height="wrap_content"  
android:background="#ffffff00"  
android:text="텍스트"  
android:textColor="#ffff0000"  
android:textSize="16dp"  
android:layout_weight="1"
```

/>

공간 가중치 설정





중복 리니어레이아웃

- 화면 전체를 구성하는 리니어레이아웃 내부에 위젯이 아닌 또 다른 리니어레이아웃이 중복으로 들어가는 경우

```
<LinearLayout>
```

```
    <LinearLayout>
```

```
        위젯들...
```

```
    </LinearLayout>
```

```
    <LinearLayout>
```

```
        위젯들...
```

```
    </LinearLayout>
```

```
    <LinearLayout>
```

```
        위젯들...
```

```
    </LinearLayout>
```

```
</LinearLayout>
```



그림 5-4 다양한 배치의 레이아웃



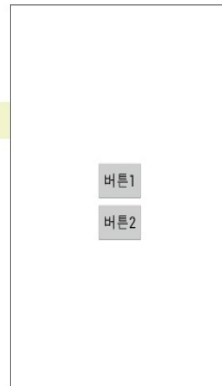
중복 리니어레이아웃

■ layout_weight 속성

- 리니어레이아웃을 여러 개 사용할 경우 각 레이아웃의 크기를 지정할 때 사용

예제 5-5 3개의 레이아웃으로 구분한 XML 코드

```
1 <LinearLayout
2     android:layout_width="match_parent"
3     android:layout_height="match_parent"
4     android:orientation="vertical" >
5     <LinearLayout
6
7         android:layout_width="match_parent"
8         android:layout_height="match_parent"
9         android:gravity="center"
10        android:orientation="vertical" >
11         <Button
12             android:text="버튼1" />
13         <Button
14             android:text="버튼2" />
15     </LinearLayout>
16     <LinearLayout
17         android:layout_width="match_parent"
18         android:layout_height="match_parent"
19         android:background="#00FF00"
20         android:gravity="center"
21         android:orientation="horizontal" >
```



```
21         <Button
22             android:text="버튼3" />
23         <Button
24             android:text="버튼4" />
25     </LinearLayout>
26     <LinearLayout
27         android:layout_width="match_parent"
28         android:layout_height="match_parent"
29         android:background="#0000FF"
30         android:gravity="center"
31         android:orientation="vertical" >
32         <Button
33             android:text="버튼5" />
34         <Button
35             android:text="버튼6" />
36     </LinearLayout>
37 </LinearLayout>
```




중복 리니어레이아웃

■ layout_height를 wrap_content로 변경

예제 5-6 layout_height를 wrap_content로 변경

```
1 <LinearLayout
2     android:orientation="vertical" >
3     <LinearLayout
4         android:layout_width="match_parent"
5         android:layout_height="wrap_content"
6         ~~~~ 중간 생략 ~~~~
7     </LinearLayout>
8     <LinearLayout
9         android:layout_width="match_parent"
10        android:layout_height="wrap_content"
11        ~~~~ 중간 생략 ~~~~
12    </LinearLayout>
13    <LinearLayout
14        android:layout_width="match_parent"
15        android:layout_height="wrap_content"
16        ~~~~ 중간 생략 ~~~~
17    </LinearLayout>
18 </LinearLayout>
```



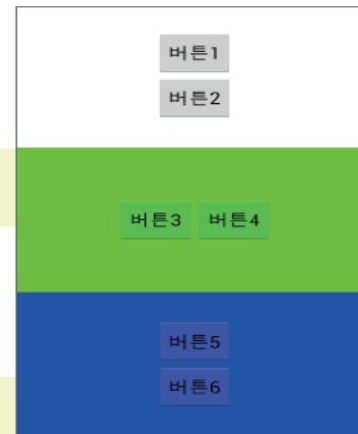


중복 리니어레이아웃

■ layout_weight를 1로 지정

예제 5-7 layout_weight를 1로 지정

```
1 <LinearLayout
2     android:orientation="vertical" >
3     <LinearLayout
4         android:layout_width="match_parent"
5         android:layout_height="match_parent"
6         android:layout_weight="1"
7         ~~~~ 중간 생략 ~~~~
8     </LinearLayout>
9     <LinearLayout
10        android:layout_width="match_parent"
11        android:layout_height="match_parent"
12        android:layout_weight="1"
13        ~~~~ 중간 생략 ~~~~
14    </LinearLayout>
15    <LinearLayout
16        android:layout_width="match_parent"
17        android:layout_height="match_parent"
18        android:layout_weight="1"
19        ~~~~ 중간 생략 ~~~~
20    </LinearLayout>
21 </LinearLayout>
```





리니어 레이아웃 - 정렬 설정하기

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

```
    android:orientation="vertical" ① 리니어 레이아웃 방향 설정
```

```
    android:layout_width="match_parent"
```

```
    android:layout_height="match_parent"
```

```
>
```

```
<Button
```

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

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

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

```
    android:layout_gravity="left" ② 첫 번째 버튼 정렬
```

```
    android:text="left"
```

```
/>
```





리니어 레이아웃 - 글자 아랫줄 정렬

```
<LinearLayout
```

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

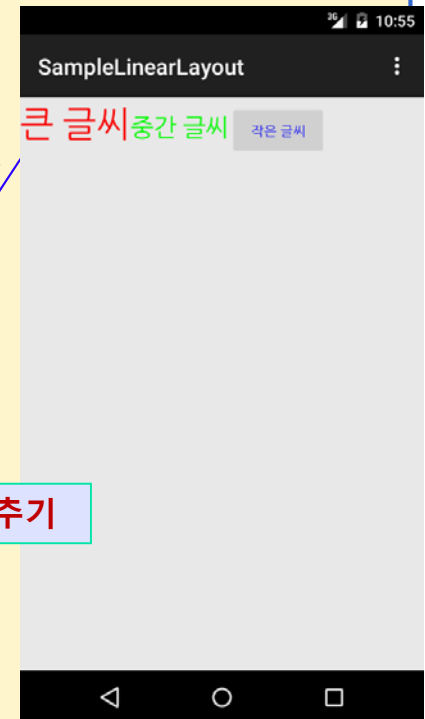
```
  android:orientation="horizontal"
```

```
  android:layout_width="match_parent"
```

```
  android:layout_height="match_parent"
```

```
  android:baselineAligned="true" ] ① 글자의 아랫줄 맞추기
```

```
>
```





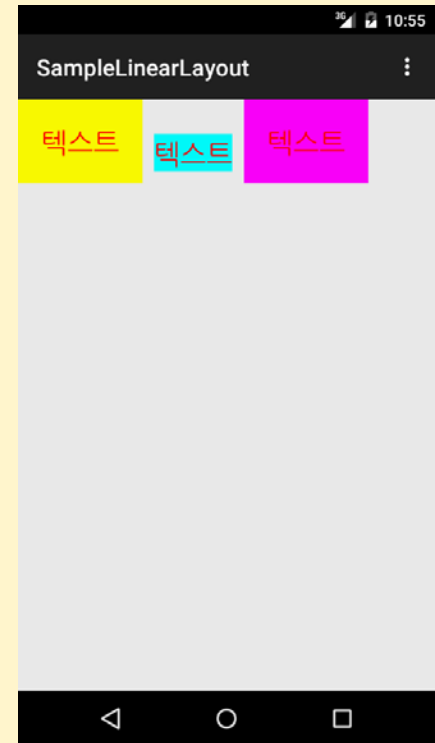
리니어 레이아웃-여유공간 설정하기

<TextView

```
android:id="@+id/button01"  
android:layout_width="wrap_content"  
android:layout_height="wrap_content"  
android:background="#ffffff00"  
android:text="텍스트"  
android:textColor="#ffff0000"  
android:textSize="24dp"  
android:padding="20dp"
```

/>

위젯 내부 여백 설정





레이아웃 중복

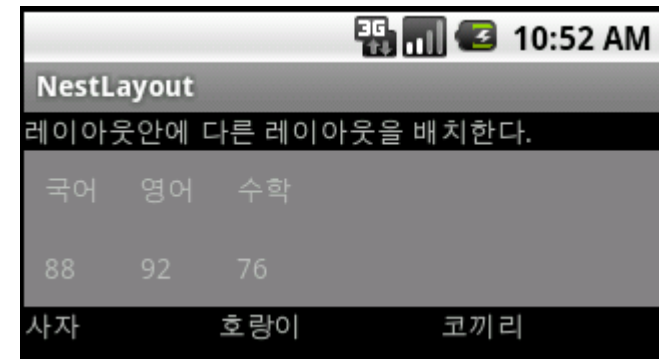
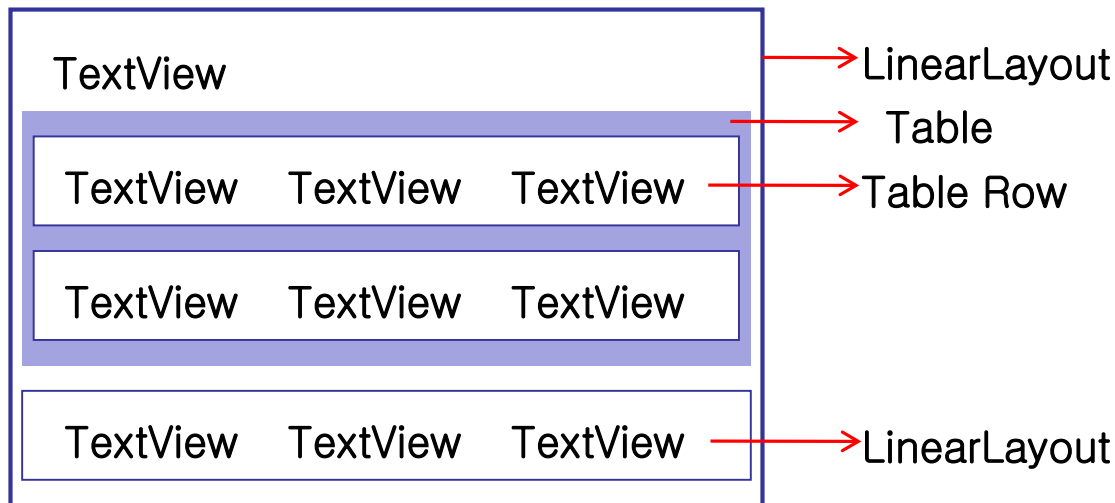


- Layout은 View의 Container이므로 View로부터 파생된 모든 객체를 Layout 안에 놓을 수 있음
- Layout 자체도 View의 파생 클래스이므로 Layout끼리 중복시킬 수 있음



레이아웃 중복 예제

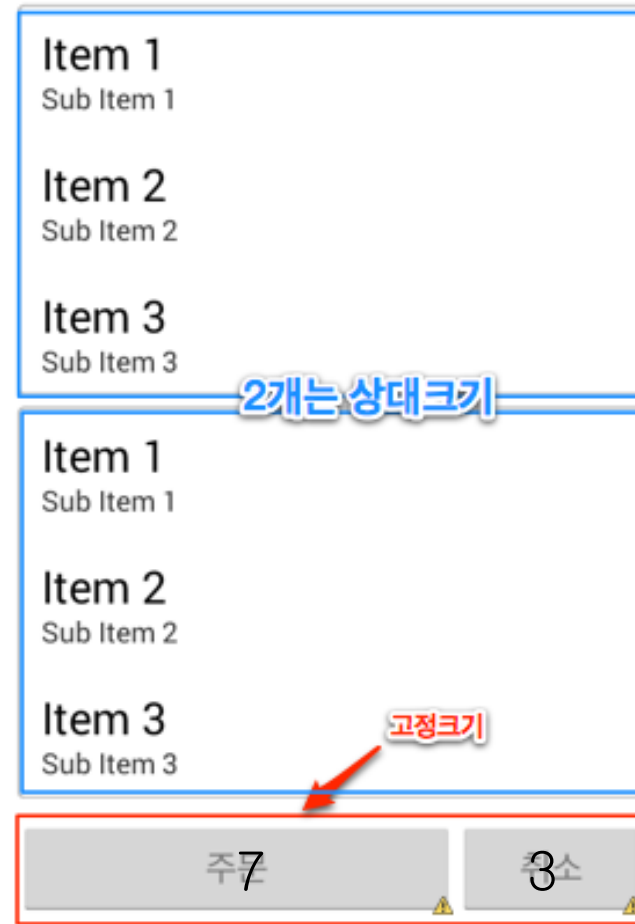
- 제일 바깥에 LinearLayout, 전체는 수직으로 뷰가 배열
- 수직 LinearLayout 안에 TextView, Table, 수평 LinearLayout이 일차 자식 뷰로 배치
- Table은 다시 2행 3열의 셀로 분할되고, 각 셀에는 Textview가 들어 있음
- 수평 LinearLayout안에는 3개의 TextView가 배치되어 있음





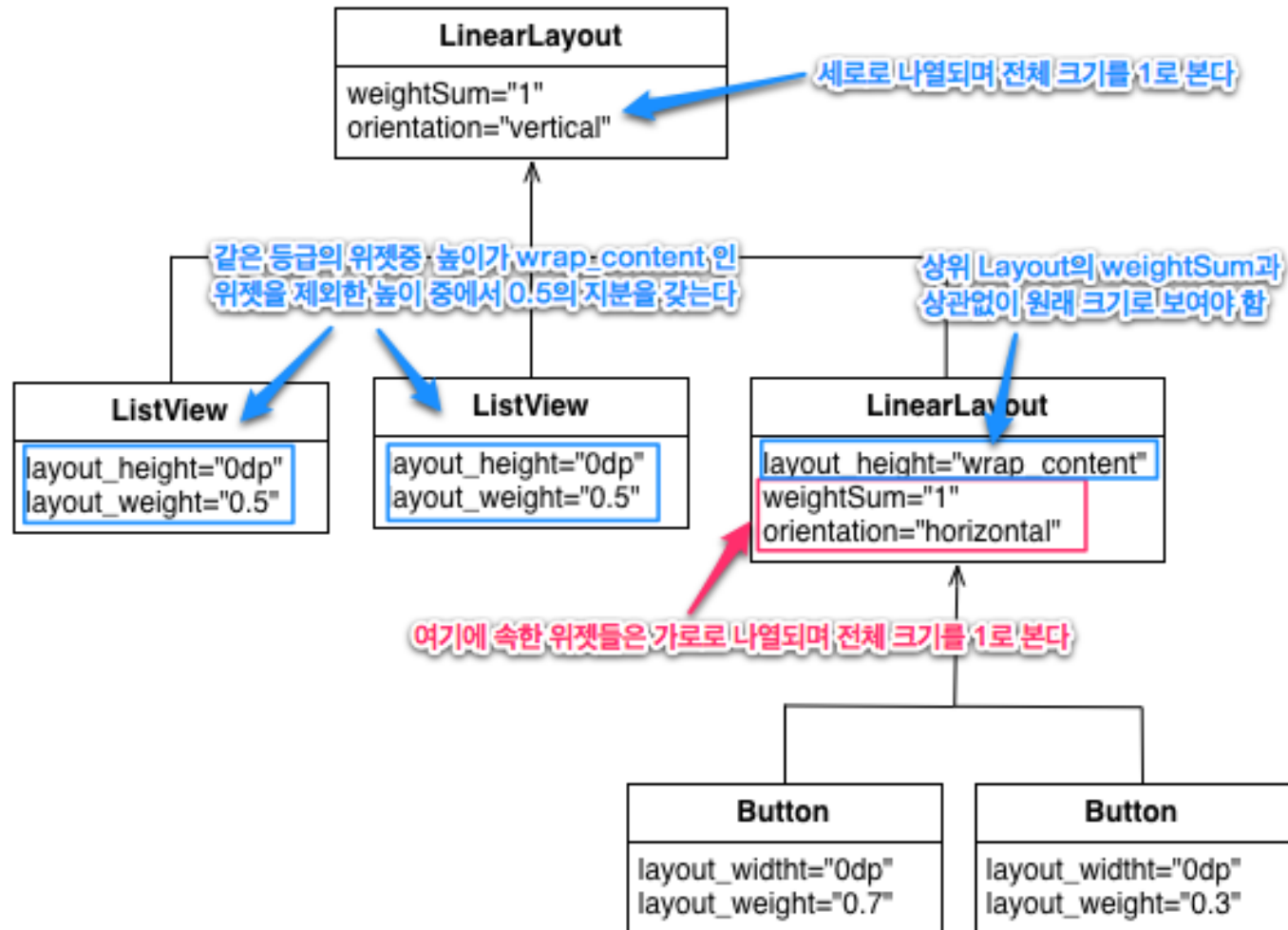
레이아웃 중복 예제(상대 크기)

- 맨 아래의 Button 영역은 원래 Button 높이를 갖고, 위의 ListView 2개는 Button 영역을 제외한 높이를 반으로 나눔





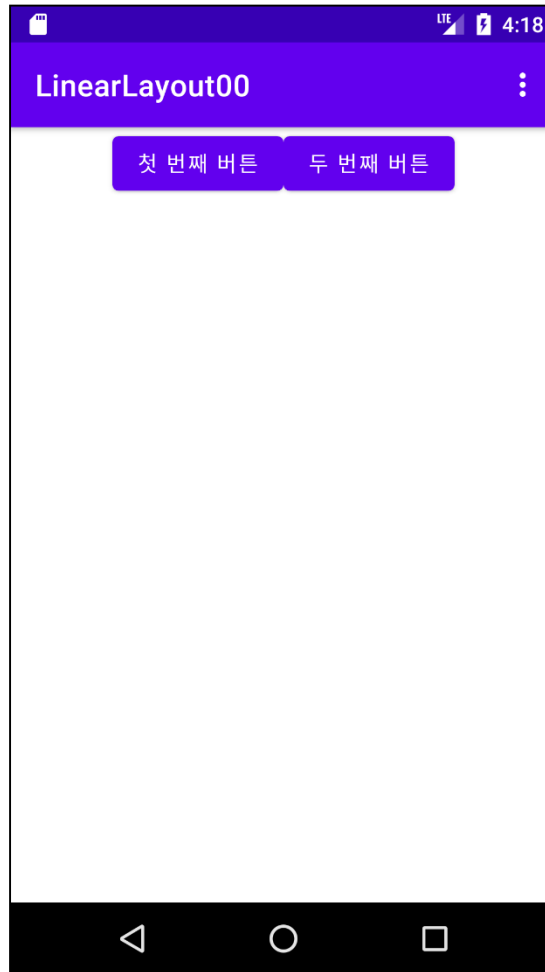
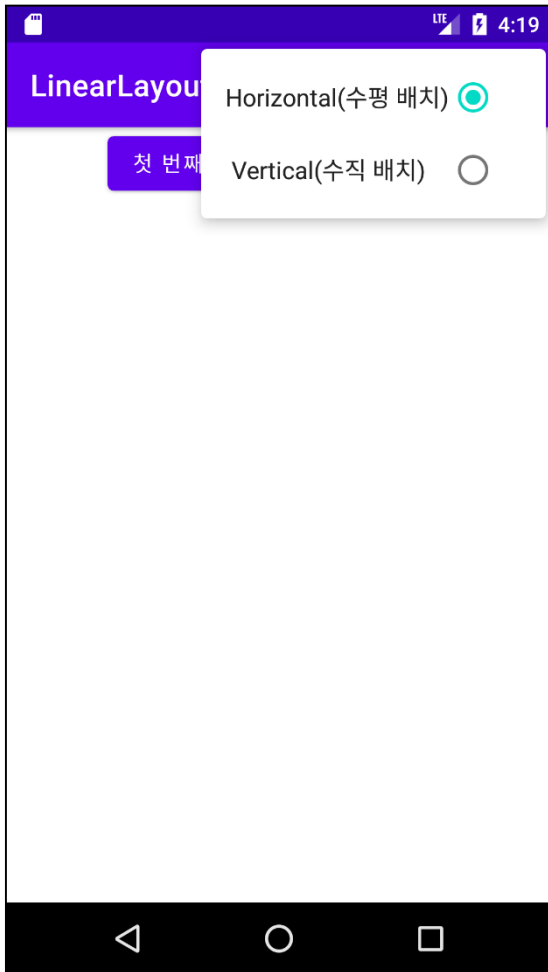
레이아웃 중복 예제





LinearLayout 예제 0

■ 다음과 같은 LinearLayout을 만들어보자





LinearLayout 예제 0

■ 사용자 인터페이스

```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:tools="http://schemas.android.com/tools"
    android:id="@+id/layout"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:gravity="center|top"
    tools:context=".MainActivity">
    <Button
        android:id="@+id/button1"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="첫 번째 버튼" />
    <Button
        android:id="@+id/button2"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="두 번째 버튼" />
</LinearLayout>
```



LinearLayout 예제 0

■ menu.xml

```
<?xml version="1.0" encoding="utf-8" ?>
<menu xmlns:android="http://schemas.android.com/apk/res/android">
    <group
        android:checkableBehavior="single"
        android:enabled="true">
        <item
            android:id="@+id/item1"
            android:checked="true"
            android:enabled="true"
            android:title="Horizontal(수평 배치)" />
        <item
            android:id="@+id/item2"
            android:enabled="true"
            android:title="Vertical(수직 배치)" />
        </group>
    </menu>
```



LinearLayout 예제 0

■ MainActivity.JAVA

```
public class MainActivity extends AppCompatActivity {  
    int index = 1;  
  
    @Override  
    protected void onCreate(Bundle savedInstanceState) {  
        super.onCreate(savedInstanceState);  
        setContentView(R.layout.activity_main);  
    }  
  
    @Override  
    protected void onStart() {  
        super.onStart();  
        if (index == 1) {  
            setContentView(R.layout.activity_main);  
        } else {  
            setContentView(R.layout.activity_main);  
            LinearLayout layout = findViewById(R.id.layout);  
            layout.setOrientation(LinearLayout.VERTICAL);  
        }  
    }  
}
```



LinearLayout 예제 0

@Override

```
public boolean onCreateOptionsMenu(Menu menu) {  
    getMenuInflater().inflate(R.menu.menu, menu);  
    return super.onCreateOptionsMenu(menu);  
}
```

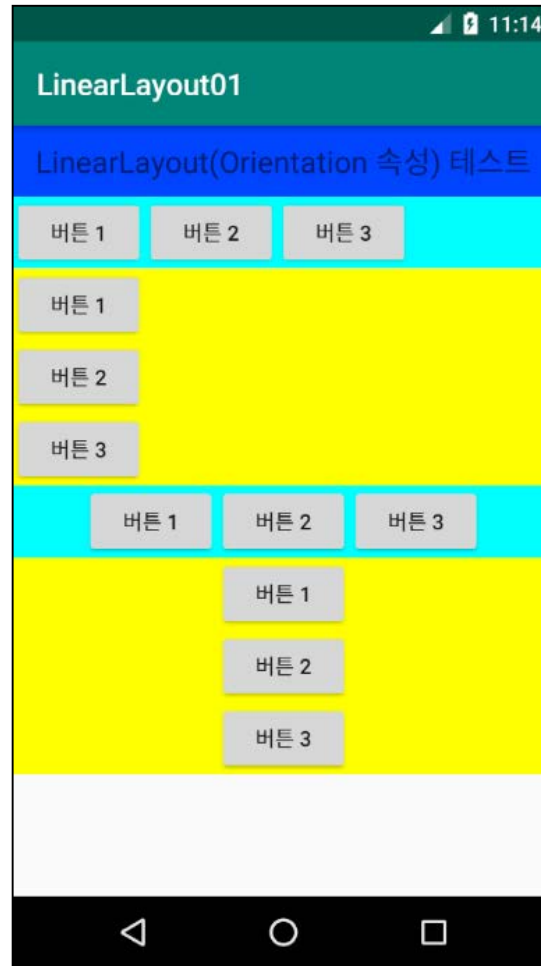
@Override

```
public boolean onOptionsItemSelected(MenuItem item) {  
    switch (item.getItemId()) {  
        case R.id.item1:  
            index = 1;  
            item.setChecked(true);  
            break;  
        case R.id.item2:  
            index = 2;  
            item.setChecked(true);  
    }  
    onStart();  
    return true;  
}
```



LinearLayout 예제 1

- 아래와 같은 화면 구성을 만들어 보자





LinearLayout 예제 1

■ 사용자 인터페이스

```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:orientation="vertical"
    tools:context=".MainActivity">

    <TextView
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:gravity="center"
        android:padding="10dp"
        android:text="LinearLayout(Orientation 속성) 테스트"
        android:background="#0044FF"
        android:textSize="20sp" />
```




LinearLayout 예제 1

■ 사용자 인터페이스

```
<LinearLayout
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:background="#00FFFF">
    <Button
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="버튼 1"/>
    <Button
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="버튼 2" />
    <Button
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="버튼 3"/>
</LinearLayout>
```



LinearLayout 예제 1

■ 사용자 인터페이스

```
<LinearLayout
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:orientation="vertical"
    android:background="#FFFF00">
    <Button
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="버튼 1"/>
    <Button
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="버튼 2" />
    <Button
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="버튼 3"/>
</LinearLayout>
```



LinearLayout 예제 1

■ 사용자 인터페이스

```
<LinearLayout
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:gravity="center"
    android:background="#00FFFF">
    <Button
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="버튼 1"/>
    <Button
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="버튼 2" />
    <Button
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="버튼 3"/>
</LinearLayout>
```



LinearLayout 예제 1



```
<LinearLayout
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:orientation="vertical"
    android:gravity="center"
    android:background="#FFFF00">
    <Button
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="버튼 1"/>
    <Button
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="버튼 2" />
    <Button
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="버튼 3"/>
</LinearLayout>
</LinearLayout>
```



LinearLayout 예제 2

- 아래와 같은 화면 구성을 만들어 보자





LinearLayout 예제 2

■ 사용자 인터페이스

```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:layout_margin="10dp"
    android:orientation="vertical"
    android:background="#FFFF00"
    tools:context=".MainActivity">

    <TextView
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:gravity="center"
        android:padding="10dp"
        android:text="LinearLayout\n(Margin, Padding 속성) 테스트"
        android:background="#0044FF"
        android:textSize="20sp" />
```



LinearLayout 예제 2

■ 사용자 인터페이스

```
<LinearLayout
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:background="#ff0000"
    android:layout_margin="10dp"
    android:gravity="center_horizontal"
    android:orientation="vertical" >

    <Button
        android:id="@+id/button01"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_margin="2dp"
        android:padding="2dp"
        android:text="마진 =2 패딩 =2" />

</LinearLayout>
```



LinearLayout 예제 2

■ 사용자 인터페이스

```
<LinearLayout
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:background="#00FF00"
    android:gravity="center_horizontal"
    android:layout_margin="10dp"
    android:orientation="vertical" >

    <Button
        android:id="@+id/button02"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_margin="8dp"
        android:padding="8dp"
        android:text="마진=8 패딩=8" />

</LinearLayout>
```




LinearLayout 예제 2

■ 사용자 인터페이스

```
<LinearLayout
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:background="#0000ff"
    android:gravity="center_horizontal"
    android:layout_margin="10dp"
    android:orientation="vertical" >

    <Button
        android:id="@+id/button03"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_margin="12dp"
        android:padding="12dp"
        android:text="마진=12 패딩=8" />

</LinearLayout>
</LinearLayout>
```



LinearLayout 예제 3

- 아래와 같은 화면 구성을 만들어 보자





LinearLayout 예제 3

■ 사용자 인터페이스

```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:orientation="vertical"
    tools:context=".MainActivity">

    <TextView
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:gravity="center"
        android:padding="10dp"
        android:text="LinearLayout\n(Margin, Padding 속성) 테스트"
        android:background="#0044FF"
        android:textSize="20sp" />
```



LinearLayout 예제 3

■ 사용자 인터페이스

```
<LinearLayout
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:background="#00FFFF"
    android:orientation="vertical">

    <Button
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:text="첫번째 버튼" />

    <Button
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:text="두번째 버튼" />
</LinearLayout>
```



LinearLayout 예제 3

■ 사용자 인터페이스

```
<LinearLayout
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_margin="20dp"
    android:background="#00FFFF"
    android:orientation="vertical">

    <Button
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:text="첫번째 버튼" />

    <Button
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:text="두번째 버튼" />
</LinearLayout>
```



LinearLayout 예제 3

■ 사용자 인터페이스

```
<LinearLayout
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:background="#FFFF00"
    android:orientation="vertical">

    <Button
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_margin="20dp"
        android:text="첫번째 버튼" />

    <Button
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_marginLeft="20dp"
        android:text="두번째 버튼" />
```



LinearLayout 예제 3

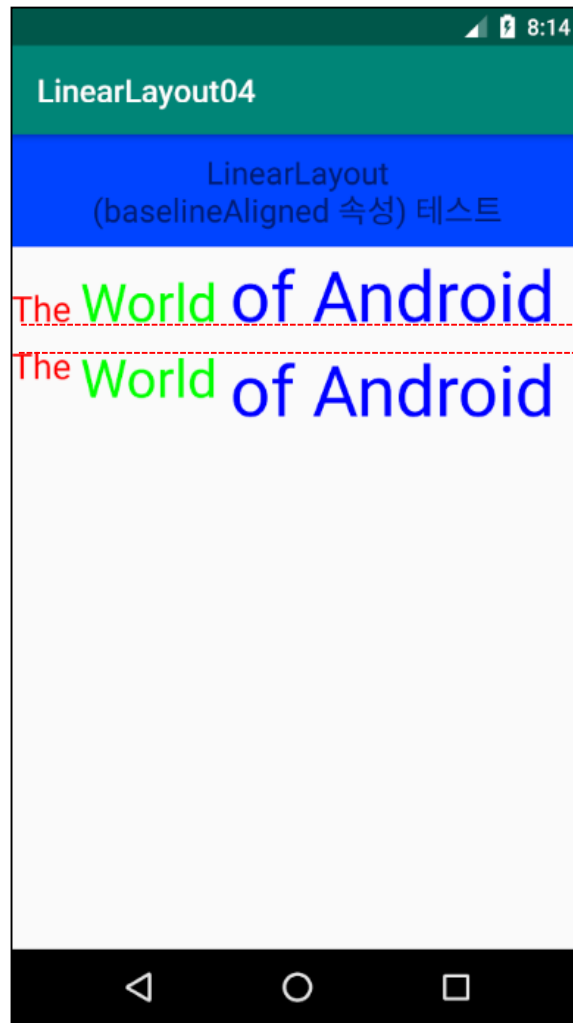
■ 사용자 인터페이스

```
<Button
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_marginRight="20dp"
    android:text="세번째 버튼" />
</LinearLayout>
</LinearLayout>
```



LinearLayout 예제 4

- 아래와 같은 화면 구성을 만들어 보자





LinearLayout 예제 4

■ 사용자 인터페이스

```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:orientation="vertical"
    tools:context=".MainActivity">

    <TextView
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:gravity="center"
        android:padding="10dp"
        android:text="LinearLayout\n(baselineAligned 속성) 테스트"
        android:background="#0044FF"
        android:textSize="20sp" />
```



LinearLayout 예제 4

```
<LinearLayout
    android:layout_width="match_parent"
    android:layout_height="wrap_content">
    <TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="The "
        android:textColor="#ff0000"
        android:textSize="10pt" />
    <TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="World "
        android:textColor="#00ff00"
        android:textSize="15pt" />
    <TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="of Android"
        android:textColor="#0000ff"
        android:textSize="20pt" />
</LinearLayout>
```



LinearLayout 예제 4

```
<LinearLayout
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:baselineAligned="false">
    <TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="The "
        android:textColor="#ff0000"
        android:textSize="10pt" />
    <TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="World "
        android:textColor="#00ff00"
        android:textSize="15pt" />
    <TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="of Android"
        android:textColor="#0000ff"
        android:textSize="20pt" />
</LinearLayout>
</LinearLayout>
```



LinearLayout 예제 5

- 아래와 같은 화면 구성을 만들어 보자





LinearLayout 예제 5

■ 사용자 인터페이스

```
<?xml version="1.0" encoding="utf-8" ?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:orientation="vertical"
    tools:context=".MainActivity">

    <TextView
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:gravity="center"
        android:padding="10dp"
        android:text="LinearLayout\n(gravity, layout_gravity 속성) 테스트"
        android:background="#0044FF"
        android:textSize="20sp" />
```



LinearLayout 예제 5

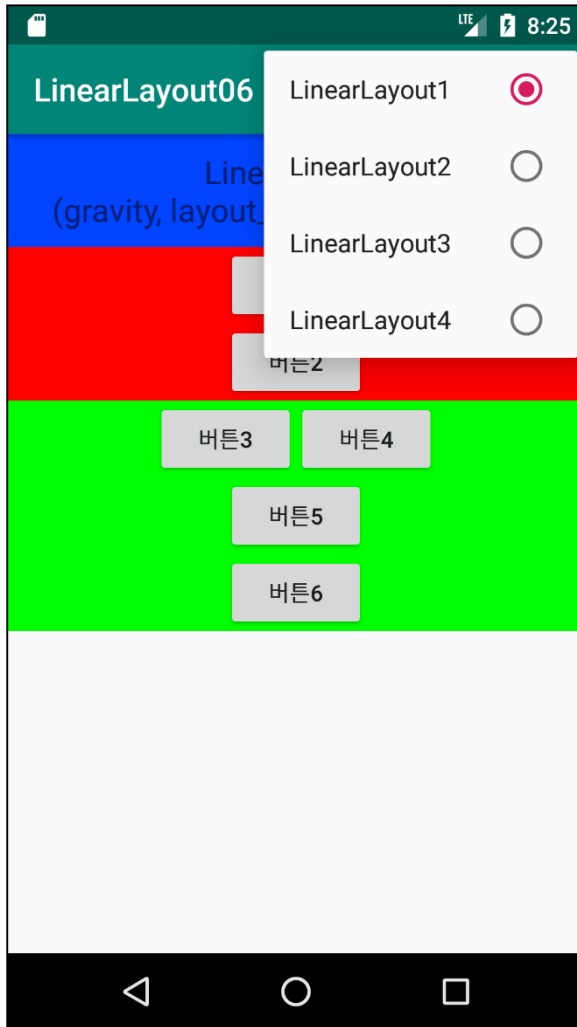


```
<LinearLayout
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:gravity="center">
    <Button
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_gravity="top"
        android:text="상단" />
    <Button
        android:text="중앙"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_gravity="center" />
    <Button
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_gravity="bottom"
        android:text="하단" />
</LinearLayout>
</LinearLayout>
```



LinearLayout 예제 6

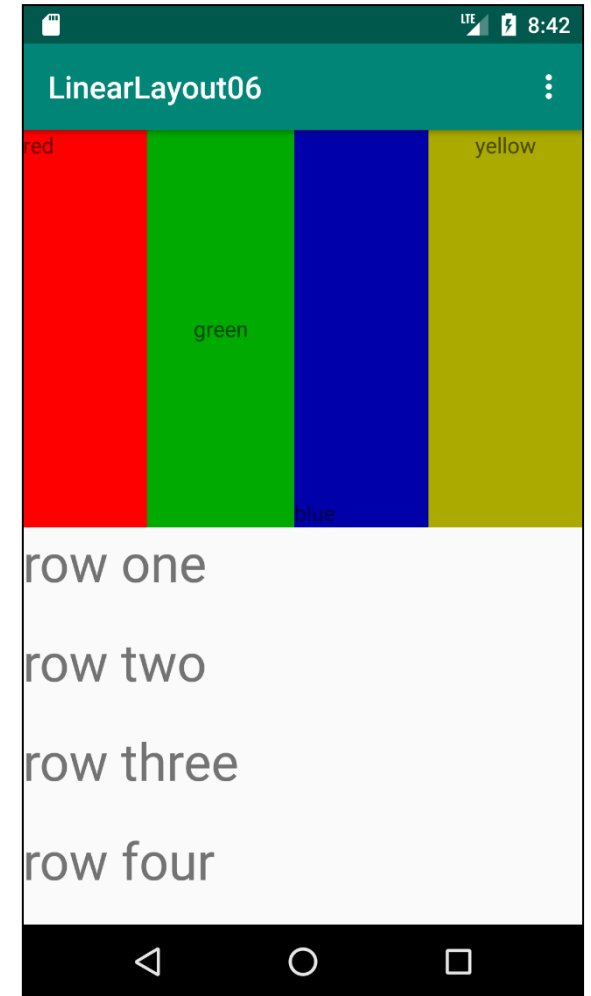
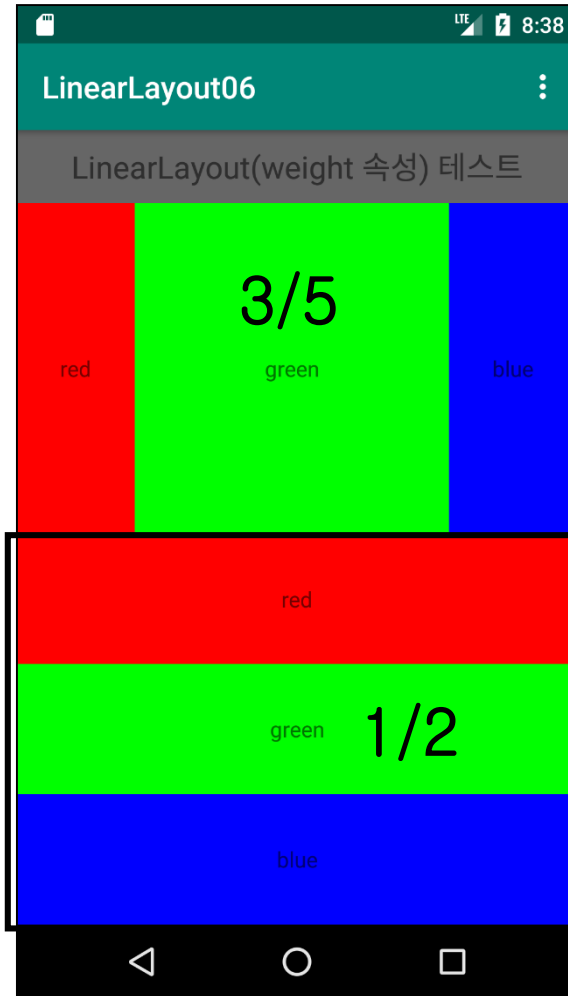
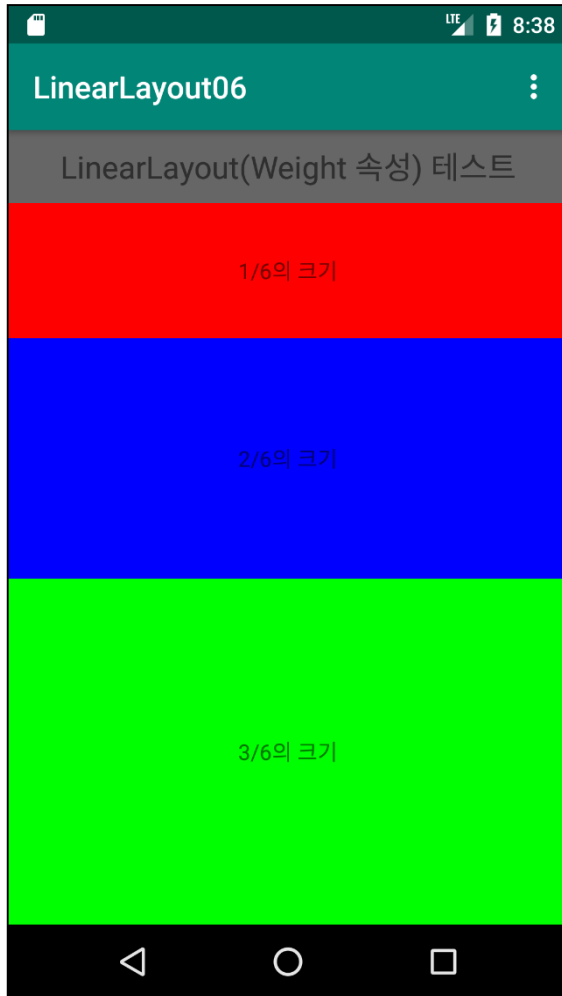
■ 아래와 같은 실행 결과를 만들어 보자





LinearLayout 예제 6

■ 실행 화면





LinearLayout 예제 6

■ menu.xml

```
<?xml version="1.0" encoding="utf-8"?>
<menu xmlns:android="http://schemas.android.com/apk/res/android">
    <group
        android:checkableBehavior="single"
        android:enabled="true">
        <item
            android:id="@+id/item1"
            android:checked="true"
            android:enabled="true"
            android:title="LinearLayout1" />
        <item
            android:id="@+id/item2"
            android:enabled="true"
            android:title="LinearLayout2" />
    </group>
</menu>
```



LinearLayout 예제 6



■ menu.xml

```
<item
    android:id="@+id/item3"
    android:enabled="true"
    android:title="LinearLayout3" />
<item
    android:id="@+id/item4"
    android:enabled="true"
    android:title="LinearLayout4" />
</group>
</menu>
```



LinearLayout 예제 6

■ activity_main1.xml

```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:orientation="vertical"
    tools:context=".MainActivity">

    <TextView
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:gravity="center"
        android:padding="10dp"
        android:text="LinearLayout\n(gravity, layout_gravity 속성) 테스트"
        android:background="#0044FF"
        android:textSize="20sp" />
```



LinearLayout 예제 6

■ activity_main1.xml

```
<LinearLayout
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:background="#ff0000"
    android:gravity="center"
    android:orientation="vertical">

    <Button
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="버튼1" />

    <Button
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="버튼2" />
</LinearLayout>
```



LinearLayout 예제 6

■ activity_main1.xml

```
<LinearLayout
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:background="#00ff00"
    android:gravity="center"
    android:orientation="vertical">

    <LinearLayout
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:gravity="center">

        <Button
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="버튼3" />
```



LinearLayout 예제 6



■ activity_main1.xml

```
<Button
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="버튼4" />
</LinearLayout>
```

```
<Button
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="버튼5" />
```

```
<Button
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="버튼6" />
```

```
</LinearLayout>
```

```
</LinearLayout>
```



LinearLayout 예제 6



■ activity_main2.xml

```
<?xml version="1.0" encoding="utf-8" ?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:orientation="vertical"
    tools:context=".MainActivity">

    <TextView
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:gravity="center"
        android:padding="10dp"
        android:text="LinearLayout(Weight 속성) 테스트"
        android:background="#0044FF"
        android:textSize="20sp" />
```



LinearLayout 예제 6



■ activity_main2.xml

```
<TextView
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_weight="1"
    android:background="#ff0000"
    android:gravity="center"
    android:text="1/6의 크기" />
```

```
<TextView
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_weight="2"
    android:background="#0000ff"
    android:gravity="center"
    android:text="2/6의 크기"/>
```




LinearLayout 예제 6

■ activity_main2.xml

```
<TextView
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_weight="3"
    android:background="#00ff00"
    android:gravity="center"
    android:text="3/6의 크기"/>
</LinearLayout>
```



LinearLayout 예제 6

■ activity_main3.xml

```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:orientation="vertical"
    tools:context=".MainActivity">

    <TextView
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:gravity="center"
        android:padding="10dp"
        android:text="LinearLayout(weight 속성) 테스트"
        android:background="#0044FF"
        android:textSize="20sp" />
```



LinearLayout 예제 6



■ activity_main3.xml

```
<LinearLayout
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_weight="3">
    <TextView
        android:layout_width="wrap_content"
        android:layout_height="match_parent"
        android:layout_weight="1"
        android:background="#ff0000"
        android:gravity="center"
        android:text="red" />
    <TextView
        android:layout_width="wrap_content"
        android:layout_height="match_parent"
        android:layout_weight="3"
        android:background="#00ff00"
        android:gravity="center"
        android:text="green" />
```



LinearLayout 예제 6



■ activity_main3.xml

```
<TextView
    android:layout_width="wrap_content"
    android:layout_height="match_parent"
    android:layout_weight="1"
    android:background="#0000ff"
    android:gravity="center"
    android:text="blue" />
```

```
</LinearLayout>
```

```
<TextView
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_weight="1"
    android:background="#ff0000"
    android:gravity="center"
    android:text="red" />
```



LinearLayout 예제 6



■ activity_main3.xml

```
<TextView  
    android:layout_width="match_parent"  
    android:layout_height="wrap_content"  
    android:layout_weight="1"  
    android:background="#00ff00"  
    android:gravity="center"  
    android:text="green" />
```

```
<TextView  
    android:layout_width="match_parent"  
    android:layout_height="wrap_content"  
    android:layout_weight="1"  
    android:background="#0000ff"  
    android:gravity="center"  
    android:text="blue" />
```

```
</LinearLayout>
```



LinearLayout 예제 6



■ activity_main4.xml

```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:orientation="vertical">

    <LinearLayout
        android:layout_width="match_parent"
        android:layout_height="match_parent"
        android:layout_weight="1">
```



LinearLayout 예제 6



■ activity_main4.xml

```
<TextView  
    android:layout_width="wrap_content"  
    android:layout_height="match_parent"  
    android:layout_weight="1"  
    android:background="@color/RED"  
    android:text="red" />
```

```
<TextView  
    android:layout_width="wrap_content"  
    android:layout_height="match_parent"  
    android:layout_weight="1"  
    android:background="#00aa00"  
    android:gravity="center"  
    android:text="@string/green" />
```



LinearLayout 예제 6



■ activity_main4.xml

```
<TextView
    android:layout_width="wrap_content"
    android:layout_height="match_parent"
    android:layout_weight="1"
    android:background="#0000aa"
    android:gravity="bottom"
    android:text="@string/blue" />
```

```
<TextView
    android:layout_width="wrap_content"
    android:layout_height="match_parent"
    android:layout_weight="1"
    android:background="#aaaa00"
    android:gravity="top|center"
    android:text="yellow" />
```

```
</LinearLayout>
```




LinearLayout 예제 6



■ activity_main4.xml

```
<LinearLayout
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:layout_weight="1"
    android:orientation="vertical">

    <TextView
        android:layout_width="match_parent"
        android:layout_height="0dp"
        android:layout_weight="1"
        android:text="row one"
        android:textSize="15pt" />

    <TextView
        android:layout_width="match_parent"
        android:layout_height="0dp"
        android:layout_weight="1"
        android:text="row two"
        android:textSize="15pt" />
```



LinearLayout 예제 6



■ activity_main4.xml

```
<TextView
    android:layout_width="match_parent"
    android:layout_height="0dp"
    android:layout_weight="1"
    android:text="row three"
    android:textSize="15pt" />
```

```
<TextView
    android:layout_width="match_parent"
    android:layout_height="0dp"
    android:layout_weight="1"
    android:text="row four"
    android:textSize="15pt" />
```

```
</LinearLayout>
```

```
</LinearLayout>
```



LinearLayout 예제 6



■ MainActivity.JAVA

```
public class MainActivity extends AppCompatActivity {  
    int index = 1;  
    @Override  
    protected void onCreate(Bundle savedInstanceState) {  
        super.onCreate(savedInstanceState);  
        setContentView(R.layout.activity_main2);  
    }  
    @Override  
    protected void onStart() {  
        super.onStart();  
        if (index == 1)  
            setContentView(R.layout.activity_main1);  
        else if (index == 2)  
            setContentView(R.layout.activity_main2);  
        else if (index == 3)  
            setContentView(R.layout.activity_main3);  
        else  
            setContentView(R.layout.activity_main4);  
    }  
}
```



LinearLayout 예제 6

■ MainActivity.JAVA

```
@Override
public boolean onCreateOptionsMenu(Menu menu) {
    getMenuInflater().inflate(R.menu.menu, menu);
    return super.onCreateOptionsMenu(menu);
}

@Override
public boolean onOptionsItemSelected(MenuItem item) {
    switch (item.getItemId()) {
        case R.id.item1:
            index = 1;
            item.setChecked(true);
            break;
        case R.id.item2:
            index = 2;
            item.setChecked(true);
            break;
    }
}
```



LinearLayout 예제 6

■ MainActivity.JAVA

```
case R.id.item3:  
    index = 3;  
    item.setChecked(true);  
    break;  
case R.id.item4:  
    index = 4;  
    item.setChecked(true);  
    break;
```

```
}
```

```
onStart();
```

```
return true;
```

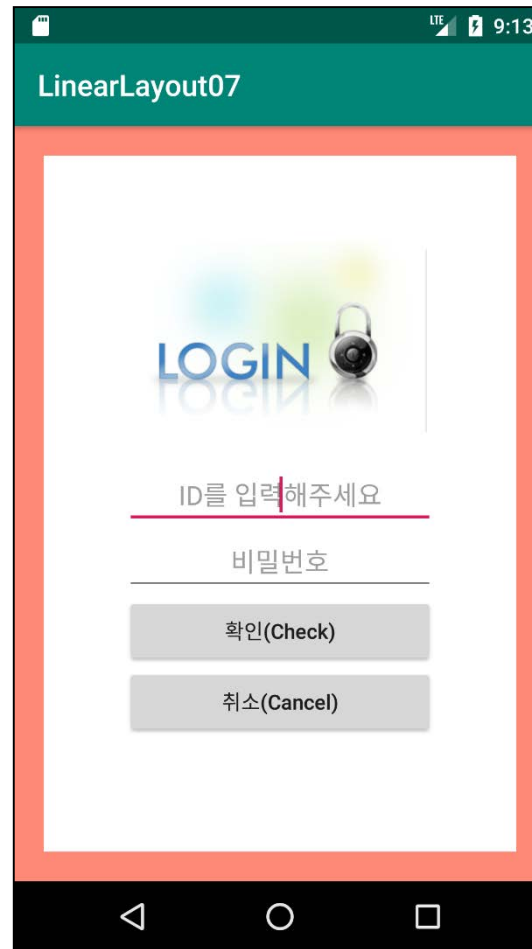
```
}
```

```
}
```



LinearLayout 예제 7

- 다음과 같은 화면을 구성하여라





LinearLayout 예제 7



■ 사용자 인터페이스

```
<?xml version="1.0" encoding="utf-8" ?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:background="#ff8976"
    android:orientation="vertical"
    tools:context="com.example.bae.report02.MainActivity">

    <LinearLayout
        android:layout_width="match_parent"
        android:layout_height="match_parent"
        android:layout_margin="20dp"
        android:background="#ffffFF"
        android:gravity="center"
        android:orientation="vertical">
```



LinearLayout 예제 7

■ 사용자 인터페이스

```
<ImageView
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:adjustViewBounds="true"
    android:maxWidth="200dp"
    android:maxHeight="200dp"
    android:src="@drawable/login" />

<LinearLayout
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_margin="20dp"
    android:gravity="center"
    android:orientation="vertical">
```




LinearLayout 예제 7

■ 사용자 인터페이스

```
<EditText
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:ems="10"
    android:gravity="center"
    android:hint="ID를 입력해주세요">
    <requestFocus/>
</EditText>
```

```
<EditText
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:ems="10"
    android:gravity="center"
    android:hint="비밀번호"
    android:inputType="textPassword" />
```



LinearLayout 예제 7

■ 사용자 인터페이스

```
<Button  
    android:layout_width="match_parent"  
    android:layout_height="wrap_content"  
    android:text="확인(Check)"  
    android:textAllCaps="false" />
```

```
<Button  
    android:layout_width="match_parent"  
    android:layout_height="wrap_content"  
    android:text="취소(Cancel)"  
    android:textAllCaps="false" />
```

```
</LinearLayout>
```

```
</LinearLayout>
```

```
</LinearLayout>
```



LinearLayout 예제 8

- 다음과 같은 화면을 구성하여라

The screenshot shows an Android application window titled "LinearLayout07". The interface is designed as a calculator. It features two input fields at the top, each with a label and a red underline: "첫번째 수" (First number) and "두번째 수" (Second number). Below these are two rows of buttons for digits 1-5 and 6-0. A section labeled "연산자" (Operator) contains four radio buttons: "더하기" (Add), "빼기" (Subtract), "곱하기" (Multiply), and "나누기" (Divide). The "더하기" option is selected. At the bottom of the calculator area is a large button labeled "계산하기" (Calculate). Below the calculator area, the text "계산결과:" (Calculation result:) is displayed in red. The entire app is running on an Android device, as indicated by the status bar at the top showing "LTE" and "9:29", and the navigation bar at the bottom.



LinearLayout 예제 8

■ 사용자 인터페이스

```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:orientation="vertical"
    android:padding="5dp"
    tools:context=".MainActivity">

    <LinearLayout
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_margin="5dp">
```



LinearLayout 예제 8

■ 사용자 인터페이스

```
<TextView  
    android:layout_width="wrap_content"  
    android:layout_height="wrap_content"  
    android:text="첫번째 수" />
```

```
<EditText  
    android:layout_width="wrap_content"  
    android:layout_height="wrap_content"  
    android:hint="첫번째 수를 입력하세요." />
```

```
</LinearLayout>
```

```
<LinearLayout  
    android:layout_width="match_parent"  
    android:layout_height="wrap_content"  
    android:layout_margin="5dp">
```



LinearLayout 예제 8

■ 사용자 인터페이스

```
<TextView  
    android:layout_width="wrap_content"  
    android:layout_height="wrap_content"  
    android:text="두번째 수" />
```

```
<EditText  
    android:layout_width="wrap_content"  
    android:layout_height="wrap_content"  
    android:hint="두번째 수를 입력하세요." />
```

```
</LinearLayout>
```

```
<LinearLayout  
    android:layout_width="match_parent"  
    android:layout_height="wrap_content"  
    android:layout_margin="5dp">
```



LinearLayout 예제 8

■ 사용자 인터페이스

```
<Button  
    android:layout_width="0dp"  
    android:layout_height="wrap_content"  
    android:layout_weight="1"  
    android:text="1" />
```

```
<Button  
    android:layout_width="0dp"  
    android:layout_height="wrap_content"  
    android:layout_weight="1"  
    android:text="2" />
```

```
<Button  
    android:layout_width="0dp"  
    android:layout_height="wrap_content"  
    android:layout_weight="1"  
    android:text="3" />
```



LinearLayout 예제 8

■ 사용자 인터페이스

```
<Button
    android:layout_width="0dp"
    android:layout_height="wrap_content"
    android:layout_weight="1"
    android:text="4" />

<Button
    android:layout_width="0dp"
    android:layout_height="wrap_content"
    android:layout_weight="1"
    android:text="5" />
</LinearLayout>
```




LinearLayout 예제 8

■ 사용자 인터페이스

```
<LinearLayout
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_margin="5dp">

    <Button
        android:layout_width="0dp"
        android:layout_height="wrap_content"
        android:layout_weight="1"
        android:text="6" />

    <Button
        android:layout_width="0dp"
        android:layout_height="wrap_content"
        android:layout_weight="1"
        android:text="7" />
```



LinearLayout 예제 8



■ 사용자 인터페이스

```
<Button  
    android:layout_width="0dp"  
    android:layout_height="wrap_content"  
    android:layout_weight="1"  
    android:text="8" />
```

```
<Button  
    android:layout_width="0dp"  
    android:layout_height="wrap_content"  
    android:layout_weight="1"  
    android:text="9" />
```

```
<Button  
    android:layout_width="0dp"  
    android:layout_height="wrap_content"  
    android:layout_weight="1"  
    android:text="0" />
```

```
</LinearLayout>
```



LinearLayout 예제 8

■ 사용자 인터페이스

<TextView

```
    android:layout_width="wrap_content"  
    android:layout_height="wrap_content"  
    android:layout_gravity="center"  
    android:text="연산자" />
```

<RadioGroup

```
    android:layout_width="wrap_content"  
    android:layout_height="wrap_content"  
    android:layout_gravity="center"  
    android:layout_margin="5dp">
```

<RadioButton

```
    android:layout_width="wrap_content"  
    android:layout_height="wrap_content"  
    android:checked="true"  
    android:text="더하기" />
```



LinearLayout 예제 8

■ 사용자 인터페이스

```
<RadioButton  
    android:layout_width="wrap_content"  
    android:layout_height="wrap_content"  
    android:text="빼기" />
```

```
<RadioButton  
    android:layout_width="wrap_content"  
    android:layout_height="wrap_content"  
    android:text="곱하기" />
```

```
<RadioButton  
    android:layout_width="wrap_content"  
    android:layout_height="wrap_content"  
    android:text="나누기" />
```

```
</RadioGroup>
```



LinearLayout 예제 8

■ 사용자 인터페이스

<Button

```
    android:layout_width="match_parent"  
    android:layout_height="wrap_content"  
    android:layout_margin="5dp"  
    android:text="계산하기" />
```

<TextView

```
    android:layout_width="wrap_content"  
    android:layout_height="wrap_content"  
    android:text="계산결과: "  
    android:textColor="#FF0000"  
    android:textSize="20dp" />
```

</LinearLayout>



LinearLayout 예제 9

- 다음과 같은 화면을 구성하여라

The screenshot shows a mobile application interface with a green header bar containing the title "LinearLayout09". Below the header, the text "연락처 추가" (Add Contact) is centered. The form consists of several input fields, each with a label and a placeholder text "이름을 입력하세요" (Enter name):

- 이름 (Name): Input field with placeholder "이름을 입력하세요"
- 소속 (Address): Input field with placeholder "소속을 입력하세요"
- 전화번호 (Phone Number): Input field with placeholder "전화번호를 입력하세요"
- 휴대폰 (Mobile Phone): Input field with placeholder "전화번호를 입력하세요"
- 이메일 (Email): Input field with placeholder "이메일을 입력하세요"

Below the email field, there is a section for gender selection:

성별 (Gender): ☒ 남자 (Male) ☐ 여자 (Female)

At the bottom of the form, there is a gray button labeled "추가" (Add).



LinearLayout 예제 9

■ 사용자 인터페이스

```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:orientation="vertical"
    tools:context=".MainActivity">

    <TextView
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:gravity="center"
        android:text="연락처 추가"
        android:textSize="30dp" />
```



LinearLayout 예제 9

■ 사용자 인터페이스

```
<LinearLayout
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_weight="1">

    <TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_margin="4dp"
        android:text="이름 " />

    <EditText
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:hint="이름을 입력하세요 " />
</LinearLayout>
```




LinearLayout 예제 9

■ 사용자 인터페이스

```
<LinearLayout
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_weight="1">

    <TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_margin="4dp"
        android:text="소 속 " />

    <EditText
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:hint="소속을 입력하세요 " />
</LinearLayout>
```



LinearLayout 예제 9

■ 사용자 인터페이스

```
<LinearLayout
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_weight="1">

    <TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_margin="4dp"
        android:text="전화번호 " />

    <EditText
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:hint="전화번호를 입력하세요"
        android:inputType="phone" />
</LinearLayout>
```



LinearLayout 예제 9

■ 사용자 인터페이스

```
<LinearLayout
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_weight="1">

    <TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_margin="4dp"
        android:text="휴대폰 " />

    <EditText
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:hint="전화번호를 입력하세요"
        android:inputType="phone" />
</LinearLayout>
```



LinearLayout 예제 9

■ 사용자 인터페이스

```
<LinearLayout
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_weight="1">

    <TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:paddingLeft="4dp"
        android:text="이메일 " />

    <EditText
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:hint="이메일을 입력하세요 " />
</LinearLayout>
```



LinearLayout 예제 9

■ 사용자 인터페이스

```
<LinearLayout
```

```
    android:layout_width="match_parent"  
    android:layout_height="wrap_content"  
    android:layout_weight="1">
```

```
<TextView
```

```
    android:layout_width="wrap_content"  
    android:layout_height="wrap_content"  
    android:layout_margin="4dp"  
    android:text="성 별" />
```

```
<RadioGroup
```

```
    android:layout_width="match_parent"  
    android:layout_height="wrap_content"  
    android:orientation="horizontal">
```



LinearLayout 예제 9

■ 사용자 인터페이스

```
<RadioButton  
    android:layout_width="0dp"  
    android:layout_height="wrap_content"  
    android:layout_weight="1"  
    android:text="남자" />
```

```
<RadioButton  
    android:layout_width="0dp"  
    android:layout_height="wrap_content"  
    android:layout_weight="1"  
    android:text="여자" />
```

```
</RadioGroup>
```

```
</LinearLayout>
```



LinearLayout 예제 9

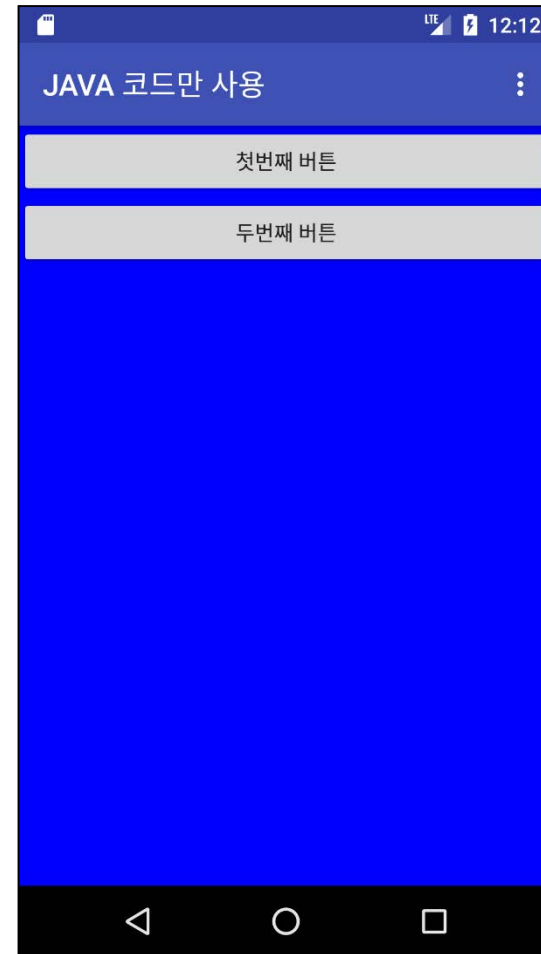
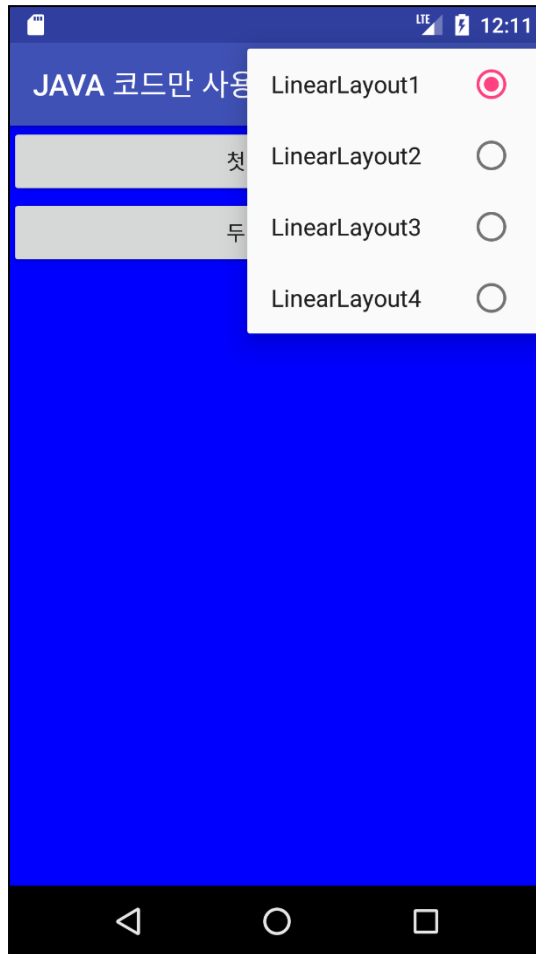
■ 사용자 인터페이스

```
<Button  
    android:layout_width="match_parent"  
    android:layout_height="wrap_content"  
    android:text="추가" />  
</LinearLayout>
```



LinearLayout 예제 10

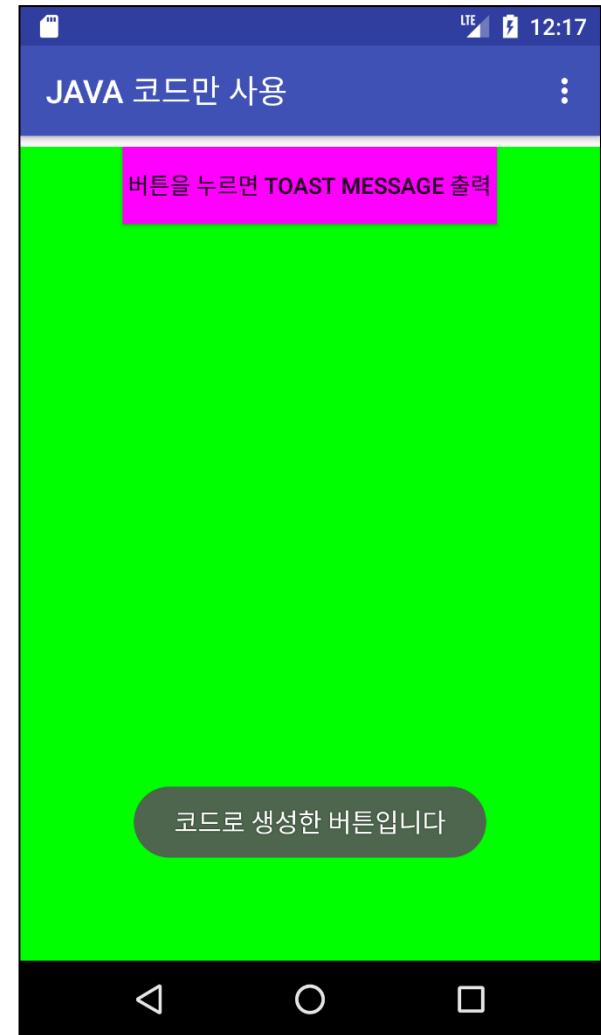
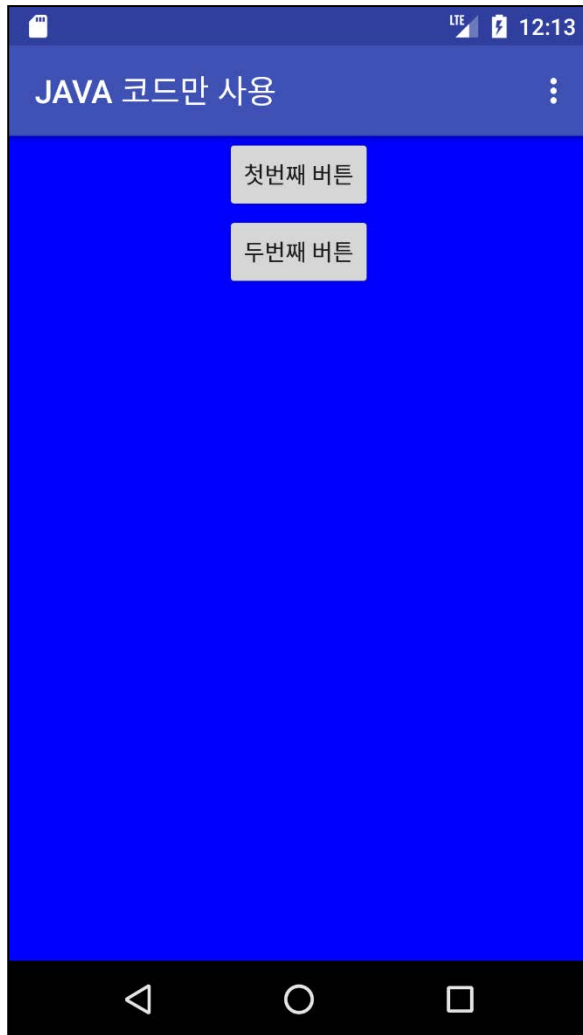
■ 다음과 같은 화면을 JAVA로만 코딩해보자





LinearLayout 예제 10

■ 실행 화면





LinearLayout 예제 10



■ menu.xml

```
<?xml version="1.0" encoding="utf-8"?>
<menu xmlns:android="http://schemas.android.com/apk/res/android">
  <group
    android:checkableBehavior="single"
    android:enabled="true">
    <item
      android:id="@+id/item1"
      android:checked="true"
      android:enabled="true"
      android:title="LinearLayout1" />
    <item
      android:id="@+id/item2"
      android:enabled="true"
      android:title="LinearLayout2" />
    <item
      android:id="@+id/item3"
      android:enabled="true"
      android:title="LinearLayout3" />
  </group>
</menu>
```



LinearLayout 예제 10

■ menu.xml

```
<item
    android:id="@+id/item4"
    android:enabled="true"
    android:title="LinearLayout4" />
</group>
</menu>
```



LinearLayout 예제 10

■ MainActivity.JAVA

```
public class MainActivity extends AppCompatActivity {  
    int index = 1;  
  
    @Override  
    protected void onCreate(Bundle savedInstanceState) {  
        super.onCreate(savedInstanceState);  
    }  
  
    @Override  
    protected void onStart() {  
        super.onStart();  
        LinearLayout layout = new LinearLayout(this);  
        layout.setBackgroundColor(Color.BLUE);  
        layout.setOrientation(LinearLayout.VERTICAL);  
        Button button1 = new Button(this);  
        button1.setText("첫번째 버튼");  
        Button button2 = new Button(this);  
        button2.setText("두번째 버튼");  
    }  
}
```



LinearLayout 예제 10

■ MainActivity.JAVA

```
if (index == 1) {  
    layout.addView(button1);  
    layout.addView(button2);  
    setContentView(layout);  
} else if (index == 2){  
    LinearLayout.LayoutParams param = new LinearLayout.LayoutParams(  
        LinearLayout.LayoutParams.WRAP_CONTENT,  
        LinearLayout.LayoutParams.WRAP_CONTENT);  
    layout.setGravity(Gravity.CENTER|Gravity.TOP);  
    layout.addView(button1, param);  
    layout.addView(button2, param);  
    setContentView(layout);  
} else if (index == 3){
```



LinearLayout 예제 10

■ MainActivity.JAVA

```
layout.setOrientation(LinearLayout.HORIZONTAL);
layout.setGravity(Gravity.CENTER|Gravity.TOP);
LinearLayout.LayoutParams param = new LinearLayout.LayoutParams(
    LinearLayout.LayoutParams.WRAP_CONTENT,
    LinearLayout.LayoutParams.WRAP_CONTENT);
layout.addView(button1, param);
layout.addView(button2, param);
setContentView(layout);
} else {
    LinearLayout.LayoutParams params = new LinearLayout.LayoutParams(
        LinearLayout.LayoutParams.MATCH_PARENT,
        LinearLayout.LayoutParams.MATCH_PARENT);
    layoutParams.topMargin = 20;
    LinearLayout linearLayout = new LinearLayout(this);
    linearLayout.setOrientation(LinearLayout.VERTICAL);
    linearLayout.setGravity(Gravity.CENTER | Gravity.TOP);
    linearLayout.setBackgroundColor(Color.rgb( 0, 255, 0));
    setContentView(linearLayout, params);
```



LinearLayout 예제 10

■ MainActivity.JAVA

```
LinearLayout.LayoutParams params1 = new LinearLayout.LayoutParams(
    LinearLayout.LayoutParams.WRAP_CONTENT,
    LinearLayout.LayoutParams.WRAP_CONTENT);
Button button = new Button(this);
button.setText("버튼을 누르면 Toast Message 출력");
button.setPadding(10, 10, 10, 10);
button.setLayoutParams(params1);
button.setBackgroundColor(Color.MAGENTA);
linearLayout.addView(button);
button.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View v) {
        Toast.makeText(getApplicationContext(), "코드로 생성한 버튼입니다",
            Toast.LENGTH_LONG).show();
    }
});
}
```



LinearLayout 예제 10

■ MainActivity.JAVA

@Override

```
public boolean onCreateOptionsMenu(Menu menu) {  
    getMenuInflater().inflate(R.menu.menu, menu);  
    return super.onCreateOptionsMenu(menu);  
}
```

@Override

```
public boolean onOptionsItemSelected(MenuItem item) {  
    switch (item.getItemId()) {  
        case R.id.item1:  
            index = 1;  
            item.setChecked(true);  
            break;  
        case R.id.item2:  
            index = 2;  
            item.setChecked(true);  
            break;  
    }
```




LinearLayout 예제 10

■ MainActivity.JAVA

```
        case R.id.item3:
            index = 3;
            item.setChecked(true);
            break;
        case R.id.item4:
            index = 4;
            item.setChecked(true);
            break;
    }
    onStart();
    return true;
}
```



Layout에 선 긋기

- Android 화면을 만들 때, View 사이에 선을 그려줘야 하는 경우가 많지만, 선 긋기라는 기능을 기본적으로 제공해 주지 않음
 - Style에 Divider라는 스타일을 설정
 - 선의 스타일이니, 필요에 맞게 설정 해주면 됨

```
<style name="Divider">  
  <item name="android:layout_width">match_parent</item>  
  <item name="android:layout_height">1 dp</item>  
  <item name="android:background">?android:attr/listDivider</item>  
</style>
```

- 해당 XML에서 설정해 놓은 Style을 적용한 View를 추가하면 됨

```
<View style="@style/Divider"/>
```