



## الفصل الرابع: النسق Layout

الصفحة	العنوان
3	1. النسق Layout
5	النسق الخطي LinearLayout
8	الجاذبية Gravity
9	الوزن Weight
10	نموذج صندوق عنصر التحكم Widget Box Model
10	قياس عنصر التحكم Sizing
11	الحشوة Padding
12	الهامش Margin
13	النسق الشبكي Grid Layout
17	النسق المتداخلة Nested layout
19	النسق النسبي RelativeLayout
21	نسق الإطار FrameLayout

## الكلمات المفتاحية:

النسق الخطي LinearLayout، نموذج صندوق عنصر التحكم Widget Box Model، النسق الشبكي Grid Layout، النسق المتداخلة NestedLayout، النسق النسبي RelativeLayout، نسق الإطار .FrameLayout

## ملخص:

نعرض في هذا الفصل مختلف أنواع النسق الممكنة لإجراء عملية توضع عناصر التحكم في الواجهات.

## أهداف تعليمية:

يتعرف الطالب في هذا الفصل على:

- أنواع النسق.
- تنسيق عناصر التحكم.

## المخطط:

النسق Layout

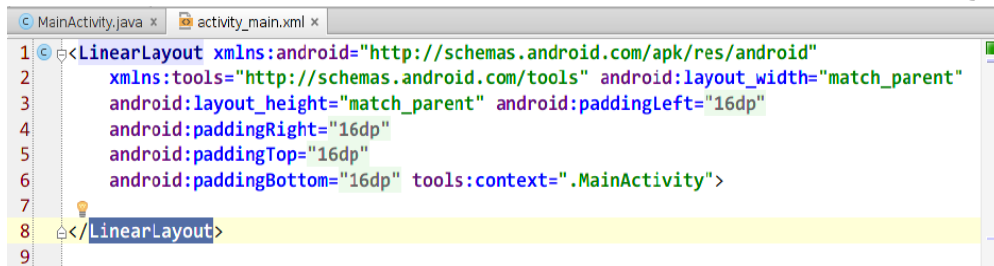
- 1 وحدة (Learning Objects)

## 1. النسق Layout

الأهداف التعليمية:

- النسق

- يُحدّد النسق طريقة توضع عناصر التحكم.
- يُمكن تغيير النسق بفتح ملف تصميم النشاط XML في طريقة عرض النص Text. ومن ثم تغيير مؤثر الفتح والإغلاق لتحديد النسق الجديد.

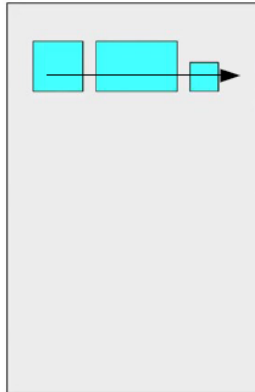


```
1 <LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
2   xmlns:tools="http://schemas.android.com/tools" android:layout_width="match_parent"
3   android:layout_height="match_parent" android:paddingLeft="16dp"
4   android:paddingRight="16dp"
5   android:paddingTop="16dp"
6   android:paddingBottom="16dp" tools:context=".MainActivity">
7
8 </LinearLayout>
9
```

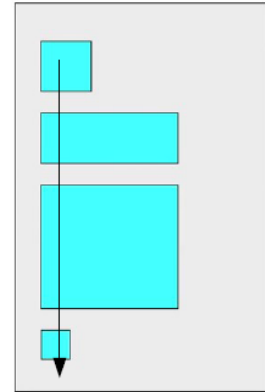
## النسق الخطي LinearLayout

- يُستخدم النسق الخطي لجعل العناصر تتوضع في خط واحد أفقي (الافتراضي) horizontal أو عمودي vertical. لاحظ أن العناصر لن تلتف wrap إذا وصلت لحافة الشاشة.

horizontal



vertical



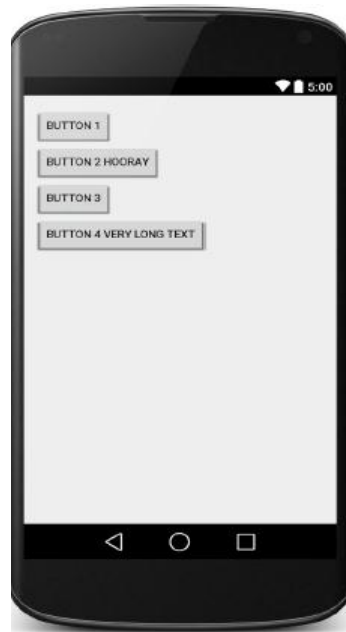
- يُبين المثال التالي استخدام النسق الخطي الأفقي:

```
<LinearLayout ...
    android:orientation="horizontal"
    tools:context=".MainActivity">
<Button ... android:text="Button 1" />
<Button ... android:text="Button 2 Hooray" />
<Button ... android:text="Button 3" />
<Button ... android:text="Button 4 Very Long Text" />
</LinearLayout>
```



• يُبين المثال التالي استخدام النسق الخطي العمودي:

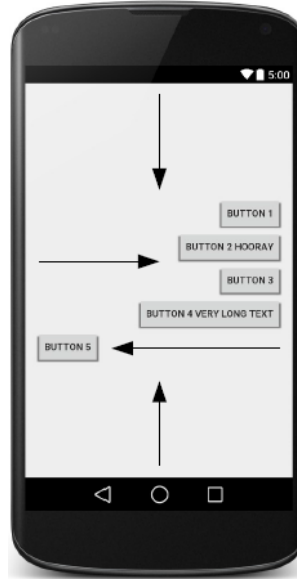
```
<LinearLayout ...  
    android:orientation="vertical"  
  
    tools:context=".MainActivity">  
  
<Button ... android:text="Button 1" />  
  
<Button ... android:text="Button 2 Hooray" />  
  
<Button ... android:text="Button 3" />  
  
<Button ... android:text="Button 4 Very Long Text" />  
  
</LinearLayout>
```





## الجاذبية Gravity

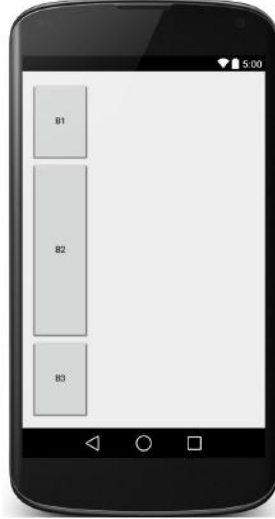
- تُحدّد الجاذبية Gravity محاذاة عناصر التحكم. والتي يُمكن أن تكون: top, bottom, left, right, center
- يُمكن اسناد أكثر من قيمة بوضع المحرف | بينهما.
- يُمكن تحديد الجاذبية على مستوى النسق فتكون لجميع عناصر التحكم أو لعناصر محدّدة كما يُبين المثال التالي:



```
<LinearLayout ...
    android:orientation="vertical"
    android:gravity="center|right">
<Button ... android:text="Button 1" />
<Button ... android:text="Button 2 Hooray" />
<Button ... android:text="Button 3" />
<Button ... android:text="Button 4 Very Long Text" />
<Button ... android:text="Button 5"
    android:layout_gravity="left" />
</LinearLayout>
```

## الوزن Weight

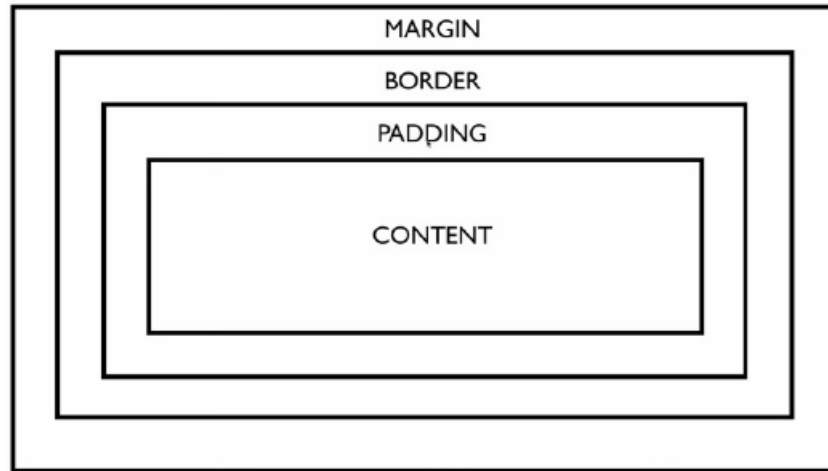
- يُعطي الوزن قياساً نسبياً لعناصر التحكم. بمعنى أنه إذا تمّ إسناد الوزن  $k$  لعنصر تحكم، فسيأخذ عنصر التحكم القياس  $k$  مقسوماً على مجموع أوزان كل العناصر. يُبين المثال التالي استخدام الوزن:



```
<LinearLayout ...  
    android:orientation="vertical">  
<Button ... android:text="B1"  
    android:layout_weight="1" />  
<Button ... android:text="B2"  
    android:layout_weight="3" />  
<Button ... android:text="B3"  
    android:layout_weight="1" />  
</LinearLayout>
```

## نموذج صندوق عنصر التحكم Widget Box Model

- يتألف نموذج صندوق عنصر التحكم من:



- المحتوى content: يكون لعنصر التحكم حجم معين (العرض width، الطول height).
- الحشوة padding: المسافة بين المحتوى والإطار.
- الإطار border: حدود عنصر التحكم.
- الهامش margin: المسافة بين عنصر التحكم وجيرانه من عناصر التحكم الأخرى.

## قياس عنصر التحكم Sizing

- يُمكن لخاصية العرض width أو الطول height أن تأخذ:
  - التفاف المحتوى wrap\_content: يكون قياس عنصر التحكم كبير كفاية لإظهار المحتوى.
  - مطابقة الأب match\_parent: يكون العرض أو الطول 100% من الشاشة.
  - قياس مُحدّد: (لا يُنصح به) مثل 64dp.
- مثال:



<Button ...

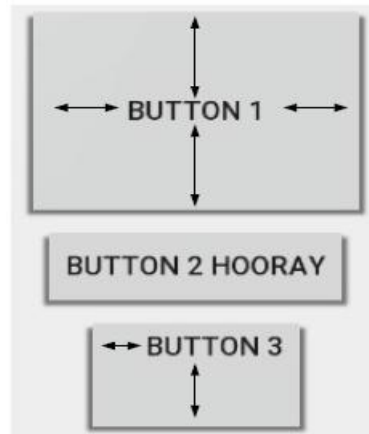
```
android:layout_width="match_parent"
```

```
android:layout_height="wrap_content" />
```

## الحشوة Padding

- يُمكن استخدام الحشوة padding لتحديد الحشوة من جميع الاتجاهات أو استخدام إحدى القيم: paddingTop, paddingBottom, paddingLeft, paddingRight
- يُبين المثال التالي استخدام الحشوة لتحديد المسافات بين المحتوى والإطار:

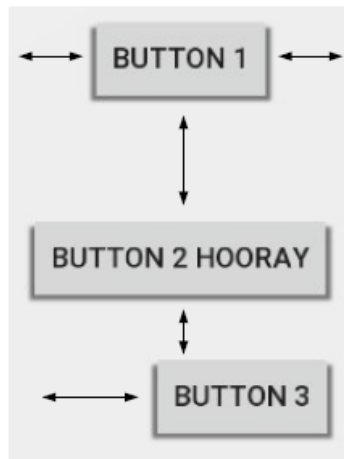
```
<LinearLayout ...  
    android:orientation="vertical">  
<Button ... android:text="Button 1"  
    android:padding="50dp" />  
<Button ... android:text="Button 2 Hooray" />  
<Button ... android:text="Button 3"  
    android:paddingLeft="30dp"  
    android:paddingBottom="40dp" />  
</LinearLayout>
```



## الهامش Margin

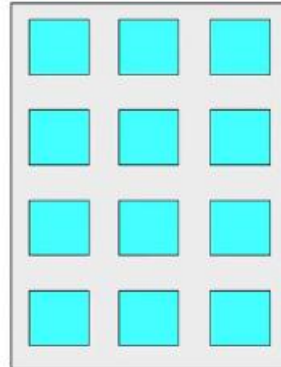
- يُمكن استخدام `layout_margin` لتحديد الهامش من جميع الاتجاهات أو استخدام إحدى القيم: `layout_marginTop`, `layout_marginBottom`, `layout_marginLeft`, `layout_marginRight`
- يُبين المثال التالي استخدام الهوامش:

```
<LinearLayout ...
    android:orientation="vertical">
<Button ... android:text="Button 1"
    android:layout_margin="50dp" />
<Button ... android:text="Button 2 Hooray" />
<Button ... android:text="Button 3"
    android:layout_marginLeft="30dp"
    android:layout_marginTop="40dp" />
</LinearLayout>
```



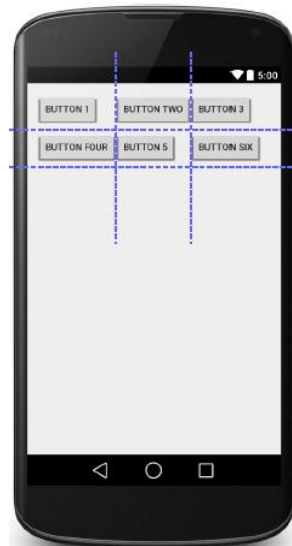
## النسق الشبكي Grid Layout

- تتوضع عناصر التحكم في خطوط من الأسطر rows والأعمدة columns.
- يوضع كل عنصر تحكم في السطر/العمود الشاغر، ما لم يتم تحديد خاصية layout\_row وخاصية layout\_column.



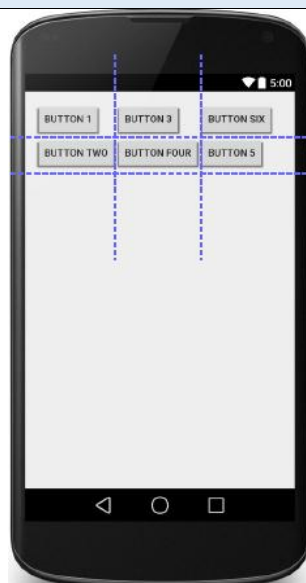
- مثال 1:

```
<GridLayout ...  
    android:rowCount="2"  
    android:columnCount="3"  
    tools:context=".MainActivity">  
<Button ... android:text="Button 1" />  
<Button ... android:text="Button Two" />  
<Button ... android:text="Button 3" />  
<Button ... android:text="Button Four" />  
<Button ... android:text="Button 5" />  
<Button ... android:text="Button Six" />  
</GridLayout>
```



مثال 2:

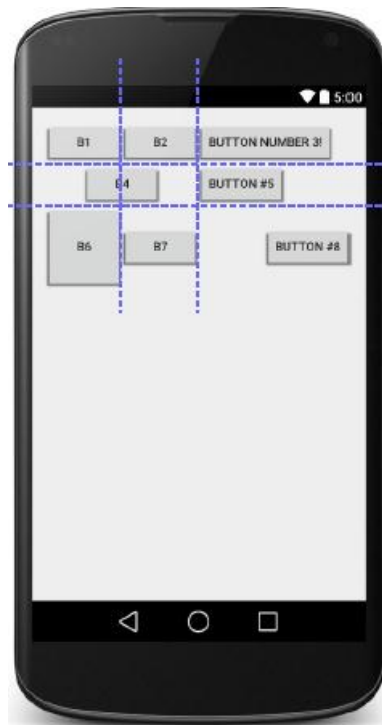
```
<GridLayout ...  
    android:rowCount="2"  
    android:columnCount="3"  
    android:orientation="vertical">  
<Button ... android:text="Button 1" />  
<Button ... android:text="Button Two" />  
<Button ... android:text="Button 3" />  
<Button ... android:text="Button Four" />  
<Button ... android:text="Button 5"  
    android:layout_row="1"  
    android:layout_column="2" />  
<Button ... android:text="Button Six"  
    android:layout_row="0"  
    android:layout_column="2" />  
</RelativeLayout>
```





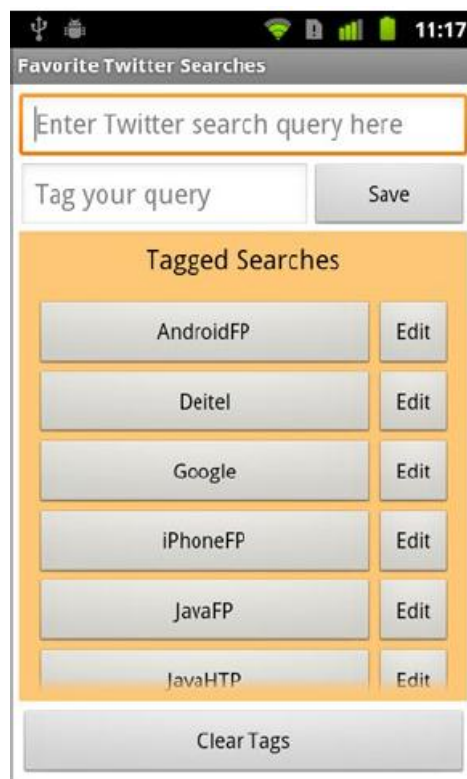
مثال 3:

```
<GridLayout ...  
    android:rowCount="2"  
    android:columnCount="3">  
<Button ... android:text="B1" />  
<Button ... android:text="B2" />  
<Button ... android:text="Button Number 3!" />  
<Button ... android:text="B4"  
    android:layout_columnSpan="2"  
    android:layout_gravity="center" />  
<Button ... android:text="B5" />  
<Button ... android:text="B6"  
    android:layout_paddingTop="40dp"  
    android:layout_paddingBottom="40dp" />  
<Button ... android:text="B7" />  
<Button ... android:text="Button #8"  
    android:layout_gravity="right" />  
</RelativeLayout>
```



### النسق المتداخلة Nested layout

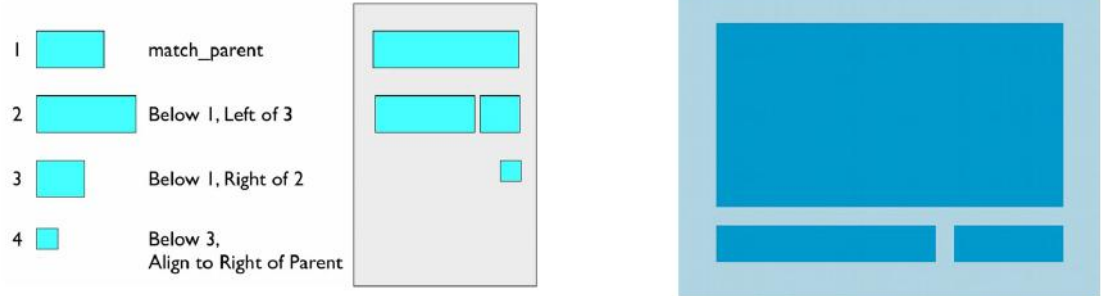
- يُمكن استخدام النسق المتداخلة للحصول على مظهر متطور أكثر.



```
<OuterLayoutType ...>  
  <InnerLayoutType ...>  
    <Widget ... />  
    <Widget ... />  
  </InnerLayoutType>  
  <InnerLayoutType ...>  
    <Widget ... />  
    <Widget ... />  
  </InnerLayoutType>  
  <Widget ... />  
  <Widget ... />  
</OuterLayoutType>
```

## النسق النسبي RelativeLayout

- يكون توضع وقياس كل عنصر نسبي إما للأب (النشاط Activity نفسه)، أو إلى عناصر تحكم أخرى.
- يُمكن استخدام النسق النسبي عوضاً عن النسق المتداخلة.



- في حال النسق النسبي مع الأب نستخدم:

`layout_alignParentTop`, `layout_alignParentBottom`, `layout_alignParentLeft`,  
`layout_alignParentRight`

`layout_centerHorizontal`, `layout_centerVertical`, `layout_centerInParent`

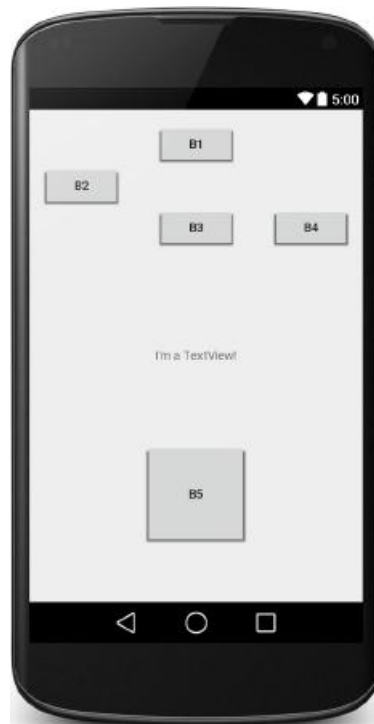
وحيث يجب وضع القيمة `true` في الخاصية لتفعيلها.

- في حال النسق النسبي مع عناصر أخرى

`layout_below`, `layout_above`, `layout_toLeftOf`, `layout_toRightOf`

وحيث تأخذ الخاصية معرف العنصر الآخر (من الشكل `@id/theID`).

مثال:



```
<RelativeLayout ... >

<Button ... android:id="@+id/b1" android:text="B1"
    android:layout_alignParentTop="true"
    android:layout_centerHorizontal="true" />

<Button ... android:id="@+id/b2" android:text="B2"
    android:layout_alignParentLeft="true"
    android:layout_below="@+id/b1" />

<Button ... android:id="@+id/b3" android:text="B3"
    android:layout_centerHorizontal="true"
    android:layout_below="@+id/b2" />

<Button ... android:id="@+id/b4" android:text="B4"
    android:layout_alignParentRight="true"
    android:layout_below="@+id/b2" />

<TextView ... android:id="@+id/tv1"
    android:text="I'm a TextView!"
    android:layout_centerInParent="true" />

<Button ... android:id="@+id/b5" android:text="B5"
    android:padding="50dp"
    android:layout_centerHorizontal="true"
    android:layout_alignParentBottom="true"
    android:layout_marginBottom="50dp" />

</RelativeLayout>
```

## نسق الإطار **FrameLayout**

- يُستخدم لوضع عنصر وحيد داخله يأخذ كامل حجم النشاط.
- يُستخدم بشكل عام مع الكتل والتي سنتعرض لها لاحقاً.

مثال:



```
<FrameLayout ... >
<ImageView
  android:src="@drawable/jellybean"
... />
</FrameLayout>
```