

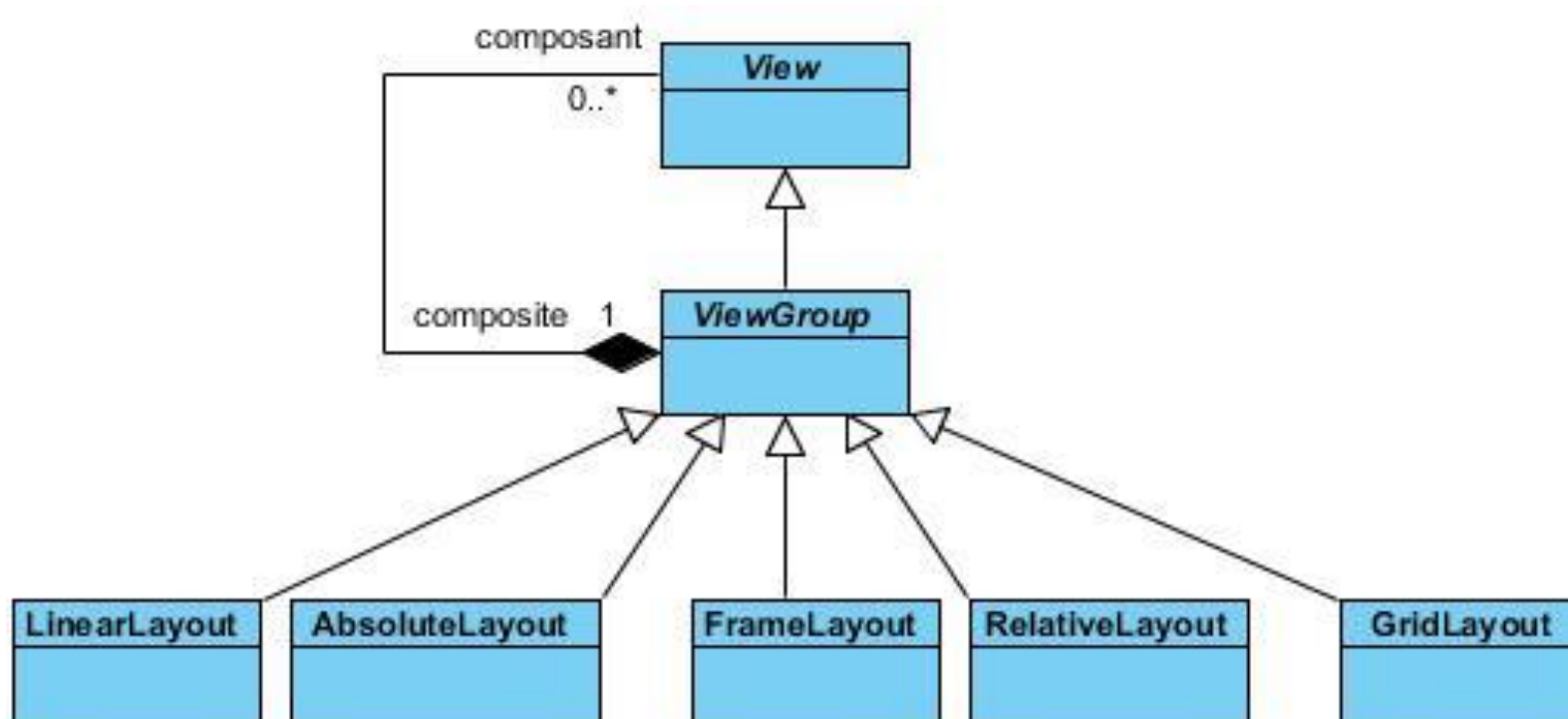
Les Layouts





ViewGroup

- ▶ Container de vues enfants qui gère leur positionnement (Composite)
- ▶ Les **Layouts** sont des **ViewGroups**





Layouts

- ▶ Ce sont des **Vues** qui gèrent la disposition de vues filles (Composants)
- ▶ Un **Layout** impose aux vues filles un ensemble de contrainte de placement
- ▶ Chaque type de **Layout** implémente une stratégie de placement des **vues composantes**
- ▶ Les **Layouts** sont définis de manière statique dans des ressources XML (rappel)



Les Différents Layouts

- ▶ **FrameLayout** : permet de superposer des composants les uns au-dessus des autres de manière à ce qu'un seul ne soit visible à la fois
- ▶ **LinearLayout** : dispose les composants les uns à la suite des autres dans le sens vertical ou bien horizontal
- ▶ **RelativeLayout** : dispose les composants relativement les uns par rapport aux autres
- ▶ **TableLayout** : disposition des composants dans une Table comme en Html
- ▶ **GridLayout** : disposition des composants dans les cellules d'une grille



Le FrameLayout

- ▶ Superposition de composants les uns au-dessus des autres de manière à ce qu'un seul ne soit visible à la fois. (utilisation propriété **visibility**)
- ▶ Possibilité d'afficher plusieurs composants à la fois en les positionnant en utilisant la propriété **gravity**

```
<FrameLayout
    android:layout_width="match_parent"
    android:layout_height="match_parent">

    <ImageView
        android:id="@+id/imageView"
        android:layout_width="114dp"
        android:layout_height="160dp"
        android:layout_gravity="center_horizontal"
        app:srcCompat="@android:drawable/btn_star_big_on" />

    <TextView
        android:id="@+id/textView3"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_gravity="center"
        android:text="Une étoile"
        android:textSize="24sp" />

</FrameLayout>
```

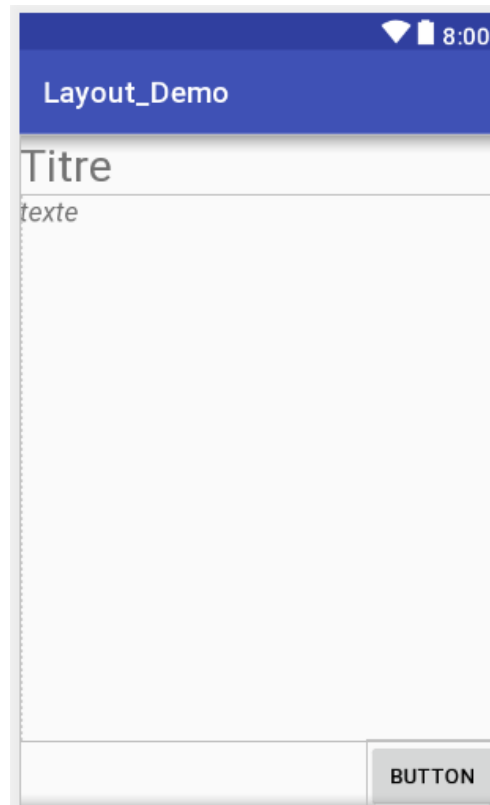




Le LinearLayout

- Placement des composants les uns à la suite des autres dans le sens vertical ou bien horizontal

```
<LinearLayout
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:orientation="vertical">
    <TextView
        android:id="@+id/titre"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:text="Titre"
        android:textSize="30sp" />
    <TextView
        android:id="@+id/texte"
        android:layout_width="match_parent"
        android:layout_height="0dp"
        android:layout_gravity="top"
        android:layout_weight="1"
        android:text="texte"
        android:textSize="18sp"
        android:textStyle="italic" />
    <Button
        android:id="@+id/button2"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_gravity="right"
        android:text="Button" />
</LinearLayout>
```



Contraintes applicables à chaque composant:

- **layout_width** et **layout_height**: décrit la manière de dimensionner sa case
 - **match_parent**: taille du parent
 - **wrap_content**: s'adapte à la taille du contenu
- **weight**: poids pour le redimensionnement valeur entière XXX
- **gravity**: placement à l'intérieur de la case (**left**, **top**, **bottom**, ..., **center**, ...)



Le RelativeLayout (1)

► Placement des composants les uns par rapport aux autres

```
<RelativeLayout
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    app:layout_constraintHorizontal_bias="0.0"
    app:layout_constraintLeft_toLeftOf="parent"
    app:layout_constraintRight_toRightOf="parent">

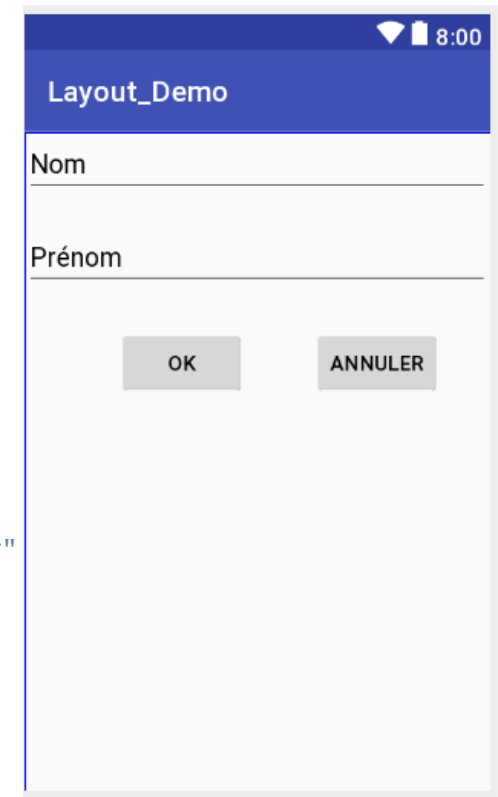
    <EditText
        android:id="@+id/txtName"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_alignParentStart="true"
        android:layout_alignParentTop="true"
        android:ems="10"
        android:inputType="textPersonName"
        android:text="Nom" />

    <EditText
        android:id="@+id/txtPrenom"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_alignParentStart="true"
        android:layout_below="@+id/txtName"
        android:layout_marginTop="19dp"
        android:ems="10"
        android:inputType="textPersonName"
        android:text="Prénom" />

    <Button
        android:id="@+id/btnAnnuler"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_alignParentEnd="true"
        android:layout_below="@+id/txtPrenom"
        android:layout_marginEnd="34dp"
        android:layout_marginTop="26dp"
        android:text="Annuler" />

    <Button
        android:id="@+id/btnOk"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_alignTop="@+id/btnAnnuler"
        android:layout_marginEnd="46dp"
        android:layout_toStartOf="@+id/btnAnnuler"
        android:text="Ok" />

</RelativeLayout>
```





Le RelativeLayout (2)

- ▶ Contraintes applicables à chaque composant:
 - ▶ **layout_width** et **layout_height**: décrit la manière de dimensionner sa case
 - ▶ **match_parent**: taille du parent
 - ▶ **wrap_content**: s'adapte à la taille du contenu
 - ▶ **layout_below**, **layout_above** : placement relatif à un autre composant
 - ▶ **layout_alignLeft**, **layout_alignRight** : alignement relatif entre composants
 - ▶ **layout_marginTop**, **layout_marginLeft** ... : marges autour du composant



Le TableLayout (1)

- Positionnement des composants dans des lignes les **TableRow**
- le **TableLayout** est similaire à une Table en Html

```
<TableLayout
    android:layout_width="match_parent"
    android:layout_height="match_parent">

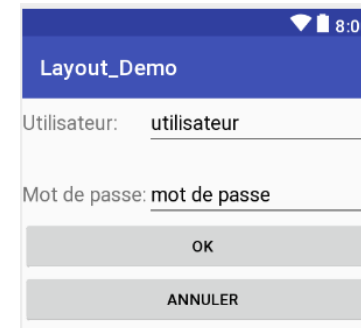
    <TableRow
        android:layout_width="match_parent"
        android:layout_height="match_parent">
        <TextView
            android:id="@+id/lblLogin"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="Utilisateur:"
            android:textSize="18sp" />
        <EditText
            android:id="@+id/txtLogin"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:ems="10"
            android:inputType="textPersonName"
            android:text="utilisateur" />
    </TableRow>

    <TableRow
        android:layout_width="match_parent"
        android:layout_height="match_parent"
        android:layout_margin="10dp" />
```

```
<TableRow
    android:layout_width="match_parent"
    android:layout_height="match_parent" >
    <TextView
        android:id="@+id/lblPassword"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Mot de passe:"
        android:textSize="18sp" />
    <EditText
        android:id="@+id/editText3"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:ems="10"
        android:inputType="textPersonName"
        android:text="mot de passe" />
</TableRow>

<TableRow
    android:layout_width="match_parent"
    android:layout_height="match_parent" >
    <Button
        android:id="@+id/btnOk"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_span="2"
        android:text="OK" />
</TableRow>
```

```
<TableRow
    android:layout_width="match_parent"
    android:layout_height="match_parent">
    <Button
        android:id="@+id/btnAnnuler"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_span="2"
        android:text="Annuler" />
    </TableRow>
</TableLayout>
```





Le TableLayout (2)

- ▶ Contraintes applicables à chaque composant:
- ▶ **Layout_column** : numéro de colonne dans la ligne
- ▶ **Layout_span** : nombre de colonnes occupées

A screenshot of an Android application titled "Layout_Demo". The interface is displayed on a screen with a blue header bar containing the title. Below the header, there are two text input fields. The first field is labeled "Utilisateur:" and contains the text "utilisateur". The second field is labeled "Mot de passe:" and contains the text "mot de passe". Below these fields are two buttons: "OK" and "ANNULER". The status bar at the top right shows a Wi-Fi icon, a battery icon, and the time "8:00".



Le GridLayout (1)

- Découpage de l'espace en Lignes et colonnes pour obtenir des cellules :

```
<GridLayout
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:columnCount="2">

    <TextView
        android:id="@+id/lblLogin2"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Utilisateur:"
        android:textSize="18sp" />

    <EditText
        android:id="@+id/editText2"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:ems="10"
        android:inputType="textPersonName"
        android:labelFor="@+id/lblPassword"
        android:text="mot de passe" />

    <TextView
        android:id="@+id/lblPassword2"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Mot de passe:"
        android:textSize="18sp" />
```

```
<EditText
    android:id="@+id/txtLogin2"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:ems="10"
    android:inputType="textPersonName"
    android:labelFor="@+id/lblLogin"
    android:text="utilisateur" />

<Button
    android:id="@+id/btnOk2"
    android:layout_width="wrap_content"
    android:layout_column="1"
    android:layout_gravity="right"
    android:freezesText="false"
    android:gravity="center"
    android:text="OK" />

</GridLayout>
```



Le GridLayout (2)

- ▶ Découpage de l'espace en Lignes et colonnes pour obtenir des cellules :
- ▶ Propriété du GridLayout : **columnCount** : nombre de colonnes par ligne
- ▶ Propriétés pour les composants:
 - ▶ **Layout_columnSpan** : nombre de colonnes occupées
 - ▶ **Layout_column** : numéro de la première cellule à occuper