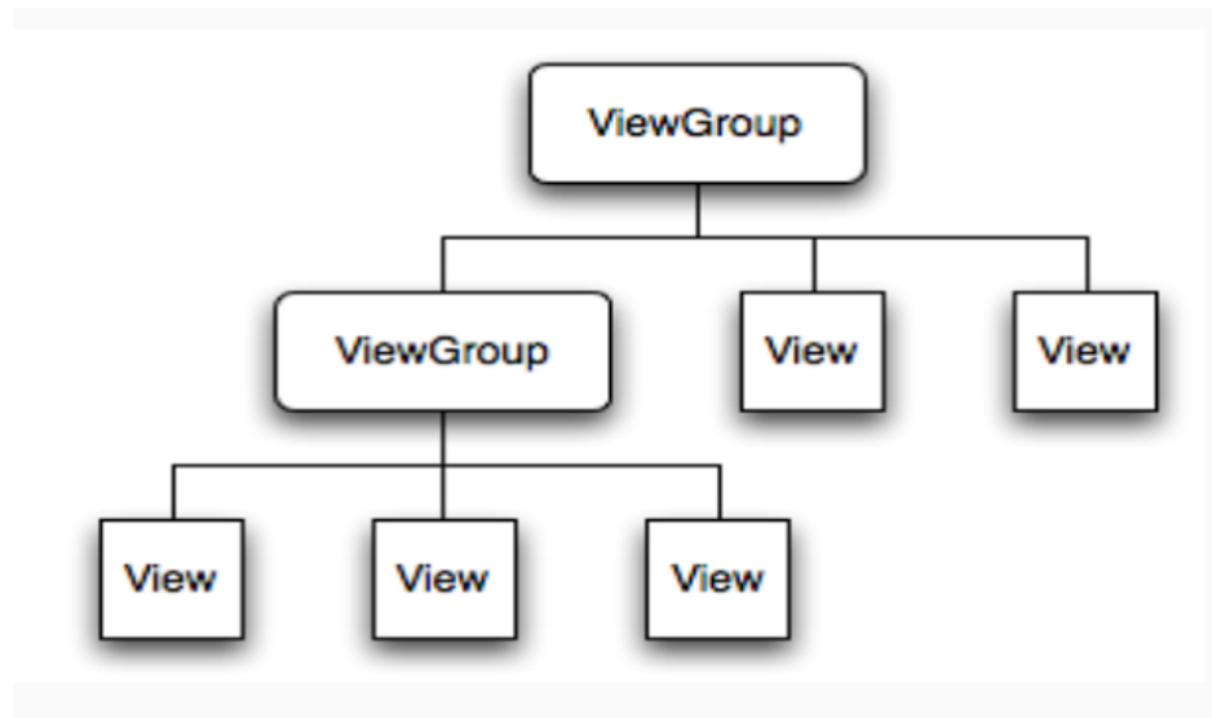


Lecture 3: Android Views and Layouts

ViewGroup



Layouts

▼ LinearLayout

- Xếp các phần tử con theo chiều hàng hoặc cột tùy setting
- Orientation: Horizontal (dạng hàng), Vertical (dạng cột)



Có thể thiết lập thuộc tính: `android:orientation="..."` trong file layout xml hoặc có thể thay đổi `setOrientation()` bằng mã khi chương trình đang chạy

- Một số thuộc tính:

- Các view trong layout này đều phải thiết lập thuộc tính kích thước `android:layout_width` và `android:layout_height`
- Weight: Xác định xem view trong layout sẽ chiếm bao nhiêu phần, giá trị có thể là 1,2,3,... mặc định là 0. Ví dụ: `android:layout_weight = 1`
- Gravity: Căn lề, mặc định là phía trên bên trái, có 2 cái có thể căn lề là `android:layout_gravity` và `android:gravity`



- Padding & Margin: Như html

▼ RelativeLayout

- **RelativeLayout** là layout mà nó hiện thị các view con nó chứa ở các vị trí trong mối liên hệ của chúng với nhau (*như View con này nằm dưới một View con khác, View con này nằm bên trái một View con khác ...*), kể cả mối liên hệ của chúng với chính phần tử cha RelativeLayout (như căn thẳng theo cạnh đáy của phần tử cha, nằm giữa phần tử cha, nằm bên trái phần tử cha ...).

Thuộc tính	Mô tả
<code>android:layout_above</code>	Cạnh dưới của view này ở trên view ID đã cho và phải là một tham chiếu tới resource khác, có dạng " <code>@[+][package:]type:name</code> ".
<code>android:layout_alignBottom</code>	Cạnh dưới của view này khớp với cạnh dưới cùng của view ID đã cho và phải là một tham chiếu tới resource khác, có dạng " <code>@ [+] [package:] type: name</code> ".
<code>android:layout_alignLeft</code>	Cạnh trái của view này khớp với cạnh trái của view ID đã cho và phải là một tham chiếu tới

	resource khác, có dạng “@ [+] [package:] type: name”
android:layout_alignParentBottom	Nếu true thì cạnh dưới của view con này khớp với cạnh dưới của view cha và ngược lại.
android:layout_alignParentLeft	Nếu true thì cạnh trái của view con này khớp với cạnh trái của view cha và ngược lại.
android:layout_alignParentRight	Nếu true thì cạnh phải của view con này khớp với cạnh phải của view cha và ngược lại.
android:layout_alignParentTop	Nếu true thì cạnh trên của view con này khớp với cạnh trên của view cha và ngược lại.
android:layout_alignRight	Cạnh phải của view này khớp với cạnh phải của view ID đã cho và phải là tham chiếu đến resource khác, có dạng “@ [+] [package:] type: name”.
android:layout_alignTop	Cạnh trên của view này khớp với cạnh trên của view ID đã cho và phải là tham chiếu đến resource khác, có dạng “@ [+] [package:] type: name”.
android:layout_below	Cạnh trên của view này dưới view ID đã cho và phải là một tham chiếu tới resource khác, có dạng “@[+][package:]type:name”.
android:layout_centerHorizontal	Nếu true thì căn chỉnh giữa view con này theo chiều ngang bên trong view cha của nó và ngược lại thì không.
android:layout_centerInParent	Nếu true căn giữa view con này theo chiều ngang và dọc bên trong view cha của nó và ngược lại thì không.
android:layout_centerVertical	Nếu true căn giữa view con này theo chiều dọc bên trong view cha của nó và ngược lại thì không.

▼ AbsoluteLayout

- **AbsoluteLayout** trong Android cho phép bạn chỉ định vị trí của các **view con bên trong nó thông qua tọa độ x/y**. Tuy nhiên AbsoluteLayout không linh hoạt và khó hỗ trợ nhiều loại kích thước màn hình. Bởi vị trí các view con được xác định là tuyệt đối.

```

<AbsoluteLayout 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"
    tools:ignore="Deprecated">
    <EditText
        android:layout_x="0px"
        android:layout_y="50px"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:hint="@string/edt_email"
    />
    <EditText
        android:layout_x="0px"
        android:layout_y="150px"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:inputType="textPassword"
        android:hint="@string/edt_password"
    />
    <Button
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="@string/btn_login"
        android:layout_x="300px"
        android:layout_y="300px"/>
</AbsoluteLayout>

```

▼ TableLayout

- TableLayout được sử dụng để hiển thị các phần tử View con trong các hàng và cột.
- Mỗi hàng là một đối tượng view TableRow, bên trong TableRow chứa các View con, mỗi View con này nằm ở vị trí một ô bảng (cell).

Thuộc tính	Chức năng
android:stretchColumns	cho phép chỉ định các cột sẽ được kéo dài (stretched) để lấp đầy không gian còn trống
android:shrinkColumns	chỉ định các cột sẽ bị co lại (shrunk) để tránh việc các View con tràn ra ngoài TableLayout
android:layout_span	áp dụng cho View con để chỉ định số ô liên tiếp trong một TableRow sẽ được hợp nhất với nhau
android:layout_column	được áp dụng cho một View con trong một TableRow để chỉ định vị trí

▼ ConstraintLayout

- Tương tự RelativeLayout nhưng hỗ trợ kéo thả nhiều hơn