

03

Views & Layouts:

LinearLayouts:

ngang hoặc dọc

- Các view trong LinearLayout đều phải thiết lập thuộc tính kích thước
 - `android:layout_width`
 - `android:layout_height`
 - Giá trị kích thước có thể là:
 - Một giá trị cụ thể - như 200 dip
 - `wrap_content` – vừa đủ nội dung bên trong
 - `Fill_parent` hoặc `match_parent` – bằng độ lớn của đối tượng cha chứa chúng
- Weight:
 - `android:layout_weight` – trọng số để xác định tỉ lệ tương ứng phần không gian còn trống dành cho đối tượng (view)
 - Giá trị có thể là 1,2,3,.. Mặc định là 0
- Gravity: được dùng để xác định cách căn lề của các đối tượng trên màn hình
 - Mặc định thì các đối tượng sẽ căn lề phía trên, bên trái
 - Khi cần căn lề, ta dùng thuộc tính XML:
 - `android:layout_gravity="..."`
 - `android:gravity="..."`
 - Giá trị có thể là: left, center, right, top, bottom, vv
- Margin & Padding:
 - `android:layout_margin="..."` : Cách lề bên ngoài
 - `android:padding="..."` : Cách lề bên trong

RelativeLayouts:

có thể để các thành phần con ở vị trí tương ứng với thành phần khác

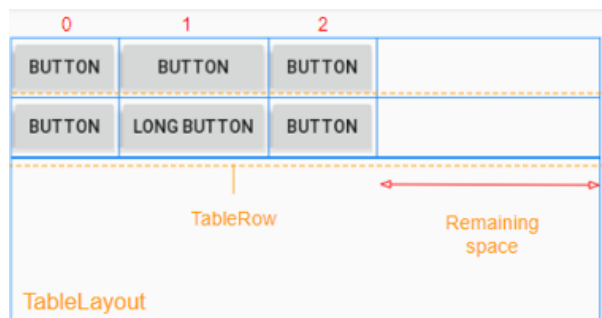
- Thuộc tính:

android:layout_above
 android:layout_alignBaseline
 android:layout_alignBottom/Left/Right/Top
 android:layout_alignParentBottom/Left/Right/Top
 android:layout_alignWithParentIfMissing
 android:layout_below
 android:layout_centerHorizontal
 android:layout_centerInParent
 android:layout_centerVertical
 android:layout_toLeftOf
 android:layout_toRightOf

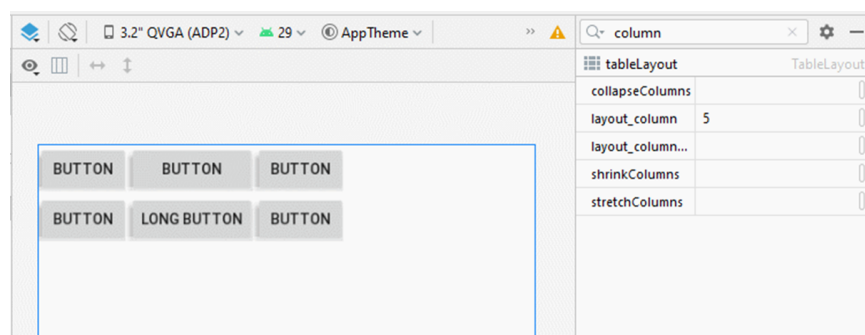
AbsoluteLayouts:

Chỉ định vị trí các view con thông qua toạ độ x y

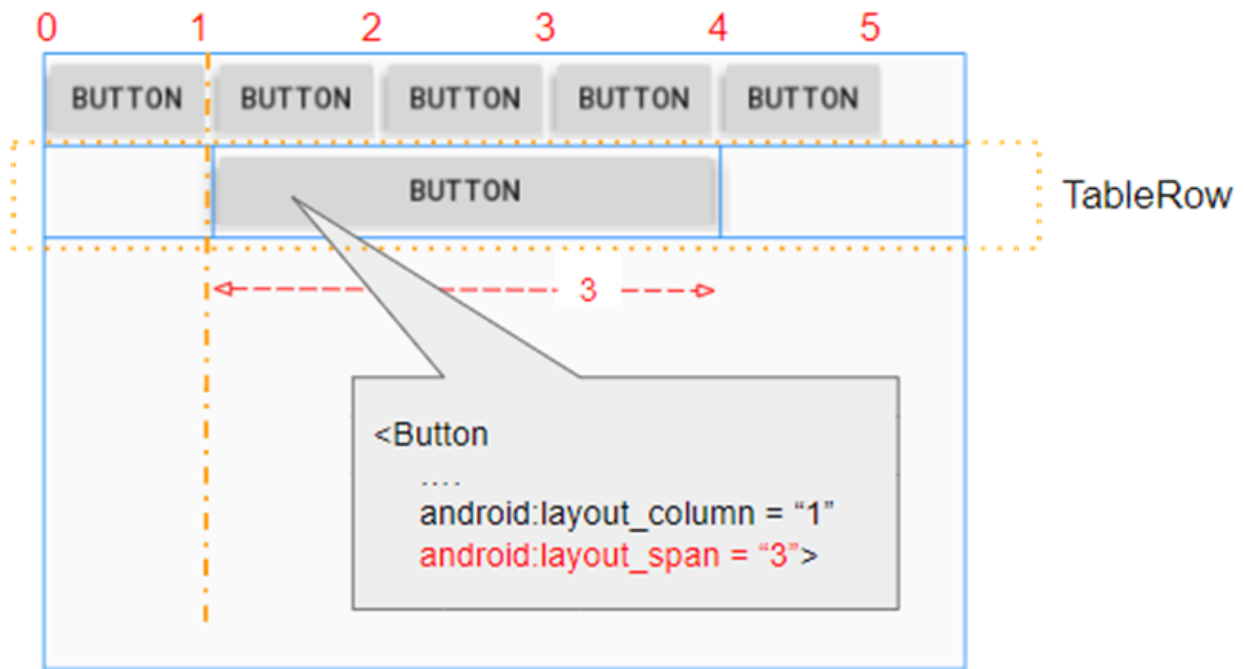
TableLayouts



- android:stretchColumns: cho phép chỉ định các cột sẽ được kéo dài để lấp đầy ko 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

hỗ trợ kéo thả