

Lập trình Android

Biên soạn: ZendVN

Hướng dẫn: Lưu Trường Hải Lân

Hỗ trợ Skype: zendvn.support

Thắc mắc !?

- Cần chuẩn bị kiến thức gì để tham gia vào khóa học này ?
 - Đã tham gia khóa học Lập trình Java 2015 do ZendVN thực hiện
 - Hoặc có kiến thức tương đương
- Không có điện thoại di động chạy trên hệ điều hành Android thì tôi có thể tham gia vào khóa học được hay không ?
 - Hoàn toàn được !

Chương 01

Nhập môn lập trình Android

1. *Lập trình Android*
2. *Cài đặt môi trường lập trình*
3. *Cấu trúc thư mục của ứng dụng*
4. *Mối liên hệ giữa Activity và Layout*
5. *Sử dụng Log và Toast*
6. *Vòng đời của một ứng dụng*

Lập trình Android (1)

- Các hệ điều hành trên di động phổ biến: iOS, BlackBerry, Windows Mobile, Android, ...
Android vẫn đang là hệ điều hành mobile được sử dụng rộng rãi nhất
- *Android là hệ điều hành dành cho các thiết bị di động được phát triển bởi Google và dựa trên nền tảng Linux*
- *Android SDK cung cấp các công cụ và API cần thiết cho việc lập trình trên Android sử dụng ngôn ngữ lập trình Java*

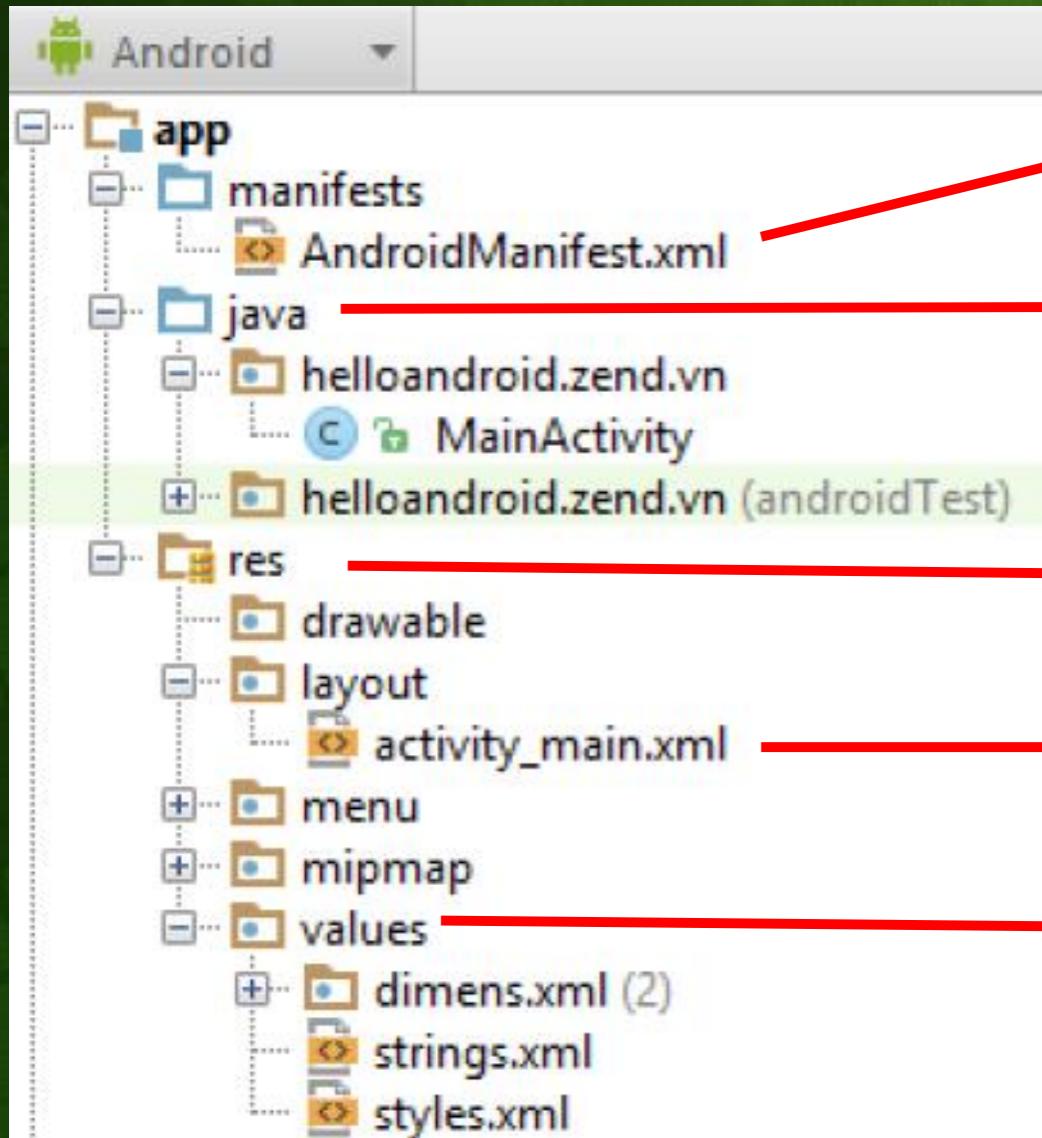
Lập trình Android (2)

- Các version Android đã phát hành (Google: Android version)
- Các thành phần chính Activities, Intent, Services, Content Provider, Broadcast Receivers, Widgets, Notifications
- Gói ứng dụng Android (Android Application package) là một file có phần mở rộng *.apk được cài đặt trên máy ảo hoặc máy thật (đóng gói tất cả mã nguồn java và tài nguyên cần thiết cho ứng dụng)

Môi trường lập trình và phát triển ứng dụng

- Cài đặt JDK (*video <https://www.youtube.com/watch?v=e6x1NWuVh-4>*)
- Cài đặt và sử dụng phần mềm Genymotion (*pdf + video*)
- Cài đặt phần mềm Android Studio (*pdf*)
- Tạo project trong Android Studio (*pdf*)
- Một số thủ thuật trong phần mềm Android Studio (*video*)
- Cài đặt plugin Genymotion trong Android Studio (*pdf + video*)
- Thực thi ứng dụng trên thiết bị thật (*video*)

Cấu trúc ứng dụng Android cơ bản (1)



Tập tin cấu hình ứng dụng

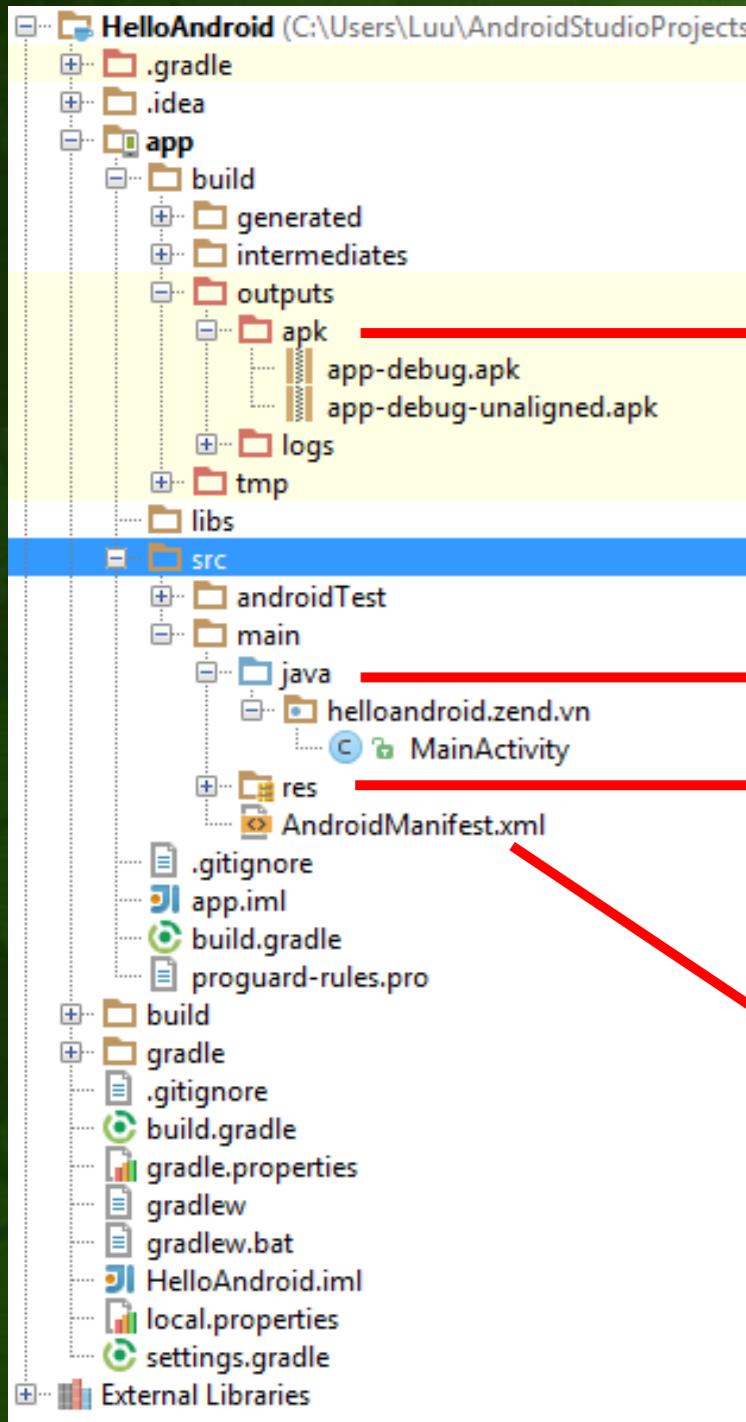
Folder chứa các mã nguồn Java

Folder chứa tài nguyên của ứng dụng

Tập tin thiết lập phần giao diện người dùng

Folder chứa các tài nguyên về string, color, dimension, ...

Cấu trúc ứng dụng Android cơ bản (2)



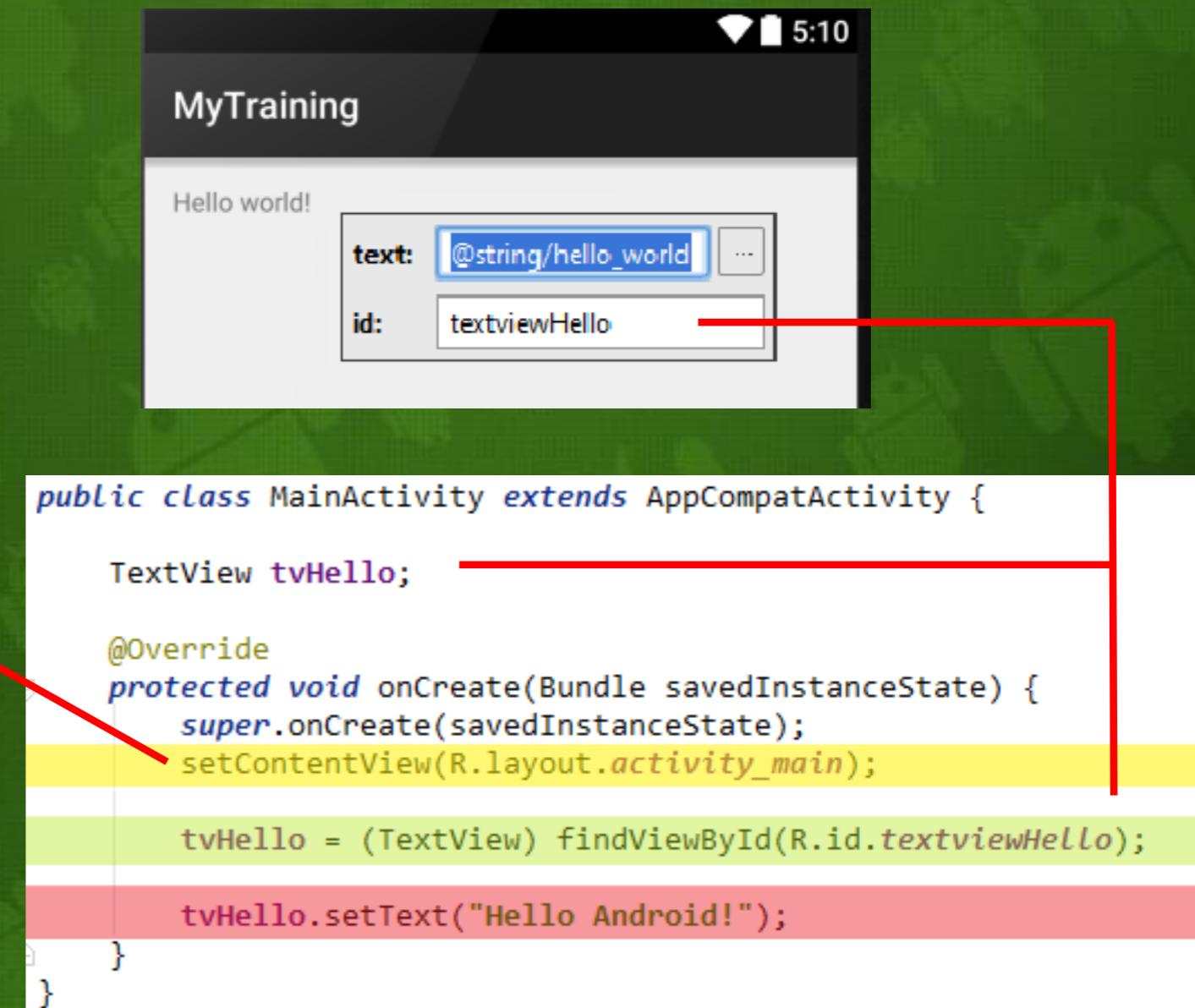
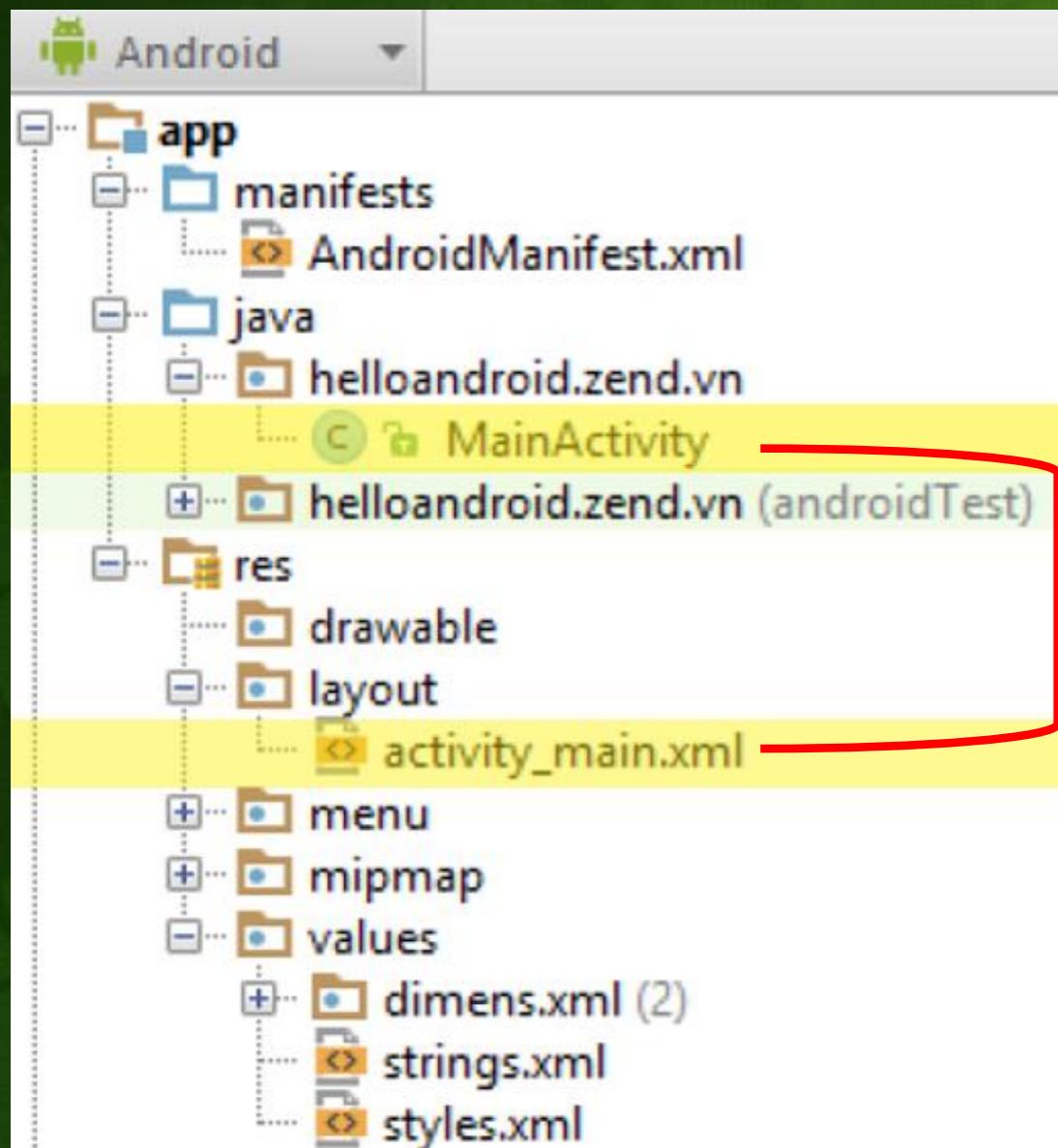
Folder chứa các file *.apk

Folder chứa các mã nguồn Java

Folder chứa tài nguyên của ứng dụng

Tập tin cấu hình ứng dụng

Mối liên hệ giữa Activity và Layout



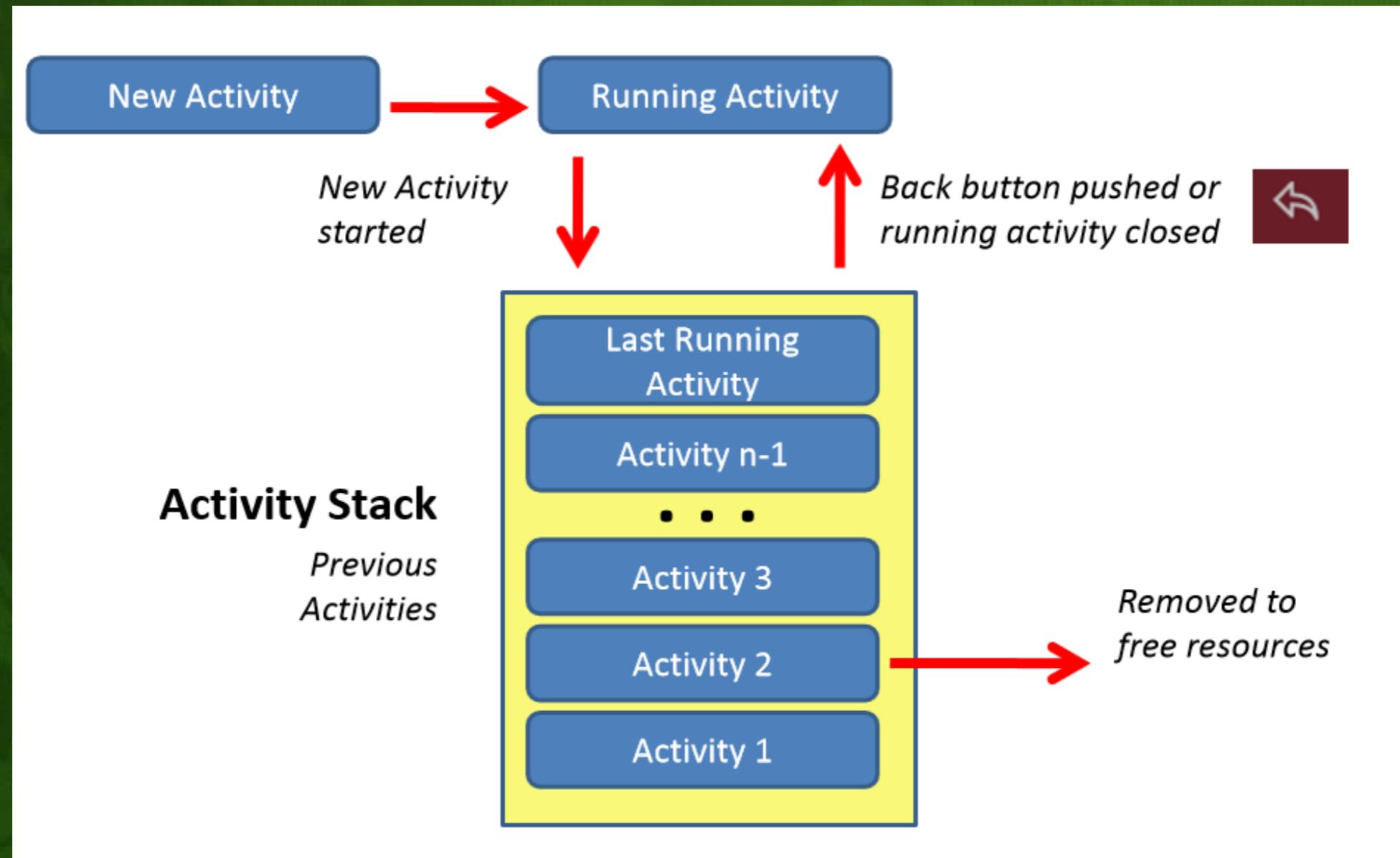
Sử dụng Log và Toast

- Khi thực thi ứng dụng muốn kiểm tra xem đã xảy ra lỗi gì hay hệ thống có cảnh báo gì, hoặc chỉ là các thông tin thực thi bình thường chúng ta có thể sử dụng *LogCat*
- Trong quá trình xây dựng ứng dụng, nếu muốn in ra giá trị của một biến đơn nào đó tại một thời điểm bất kỳ chúng ta có thể sử dụng:
 - *Log*
 - *Toast*

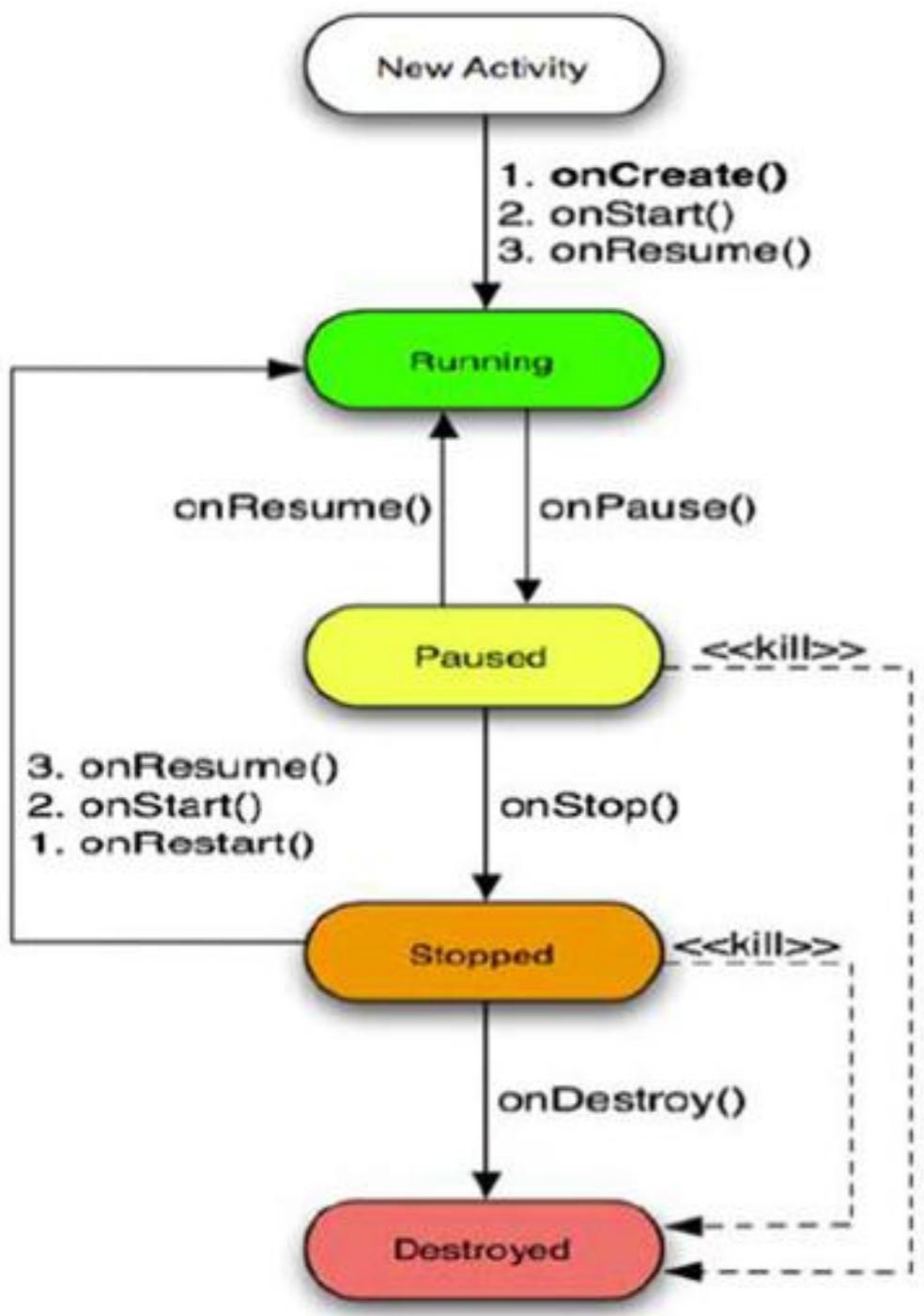
Activity Life Cycle

- **Application:** Android Project khi được biên dịch thành công sẽ được đóng gói thành tập tin *.apk, tập tin này được gọi là một Application
- Một Application có thể có một hoặc nhiều **Activity**, mỗi Activity sẽ có một vòng đời riêng độc lập hoàn toàn với các Activity khác (Ngôi nhà và các cửa sổ)
- Khi **tạo mới một Activity** nào đó cần phải khai báo, và trong nhiều Activity sẽ có một Activity chạy chính (Activity xuất hiện đầu tiên). Các phần khai báo này được thực hiện trong *AndroidManifest.xml*

Activity Stack

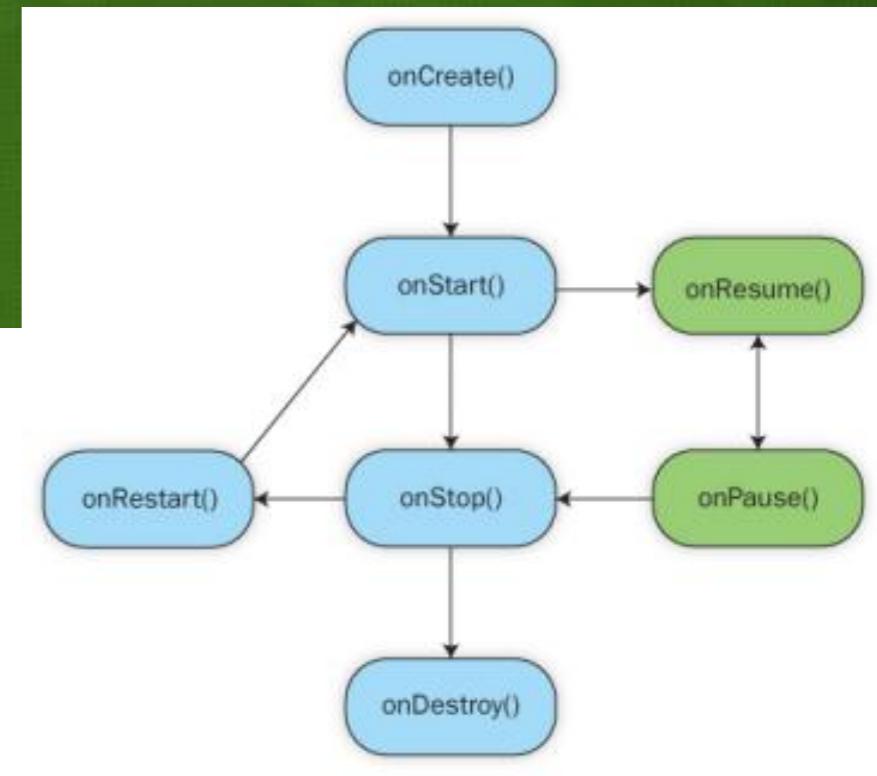
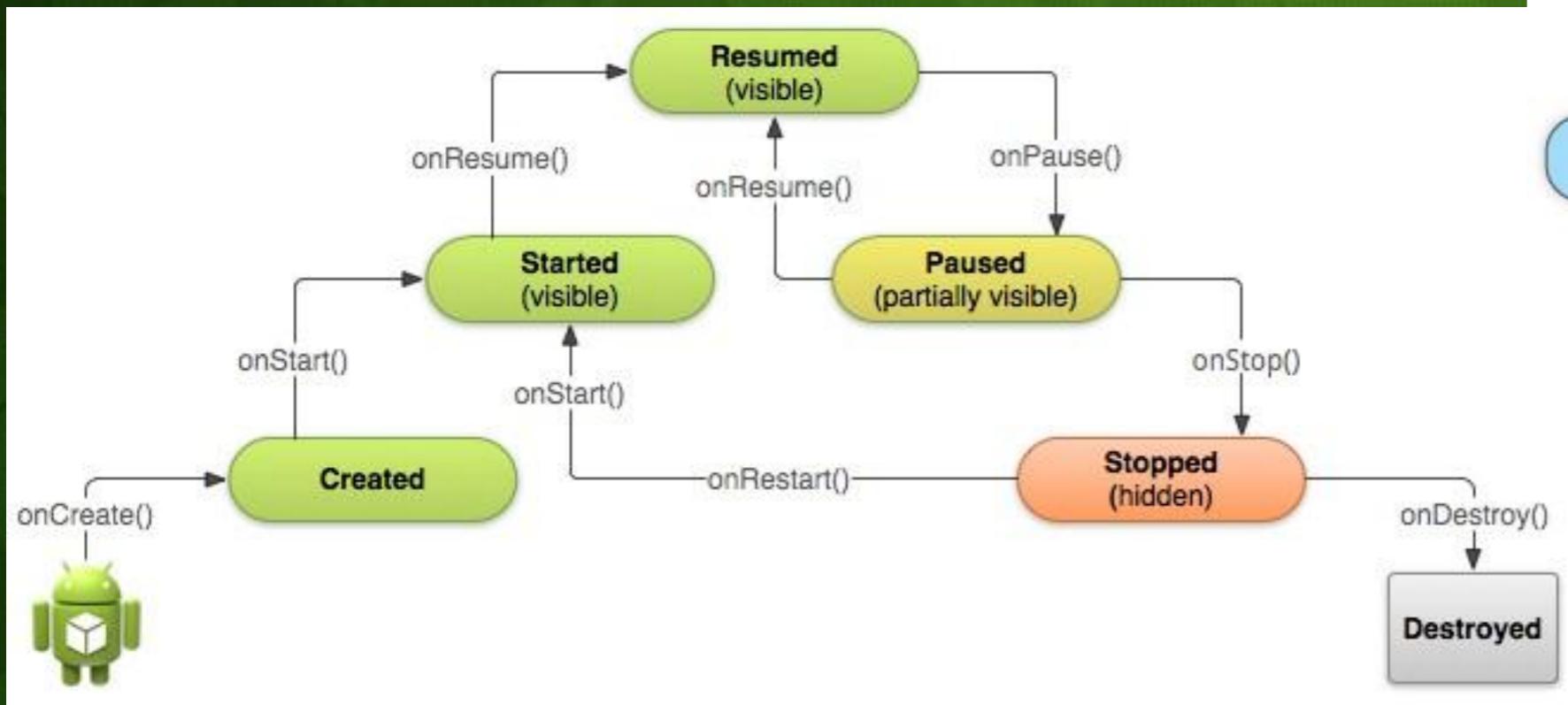


Life Cycle States



- 1. Running** Activity nằm trên cùng ứng dụng và cho phép người sử dụng tương tác
- 2. Paused** Activity bị mất focus nhưng mà vẫn nhìn thấy được Activity này
- 3. Stopped** Activity không nhìn thấy được Paused hay Stopped đều có khả năng bị Destroyed (hủy) khi bộ nhớ cần cho việc khác ưu tiên hơn.

Life Cycle States

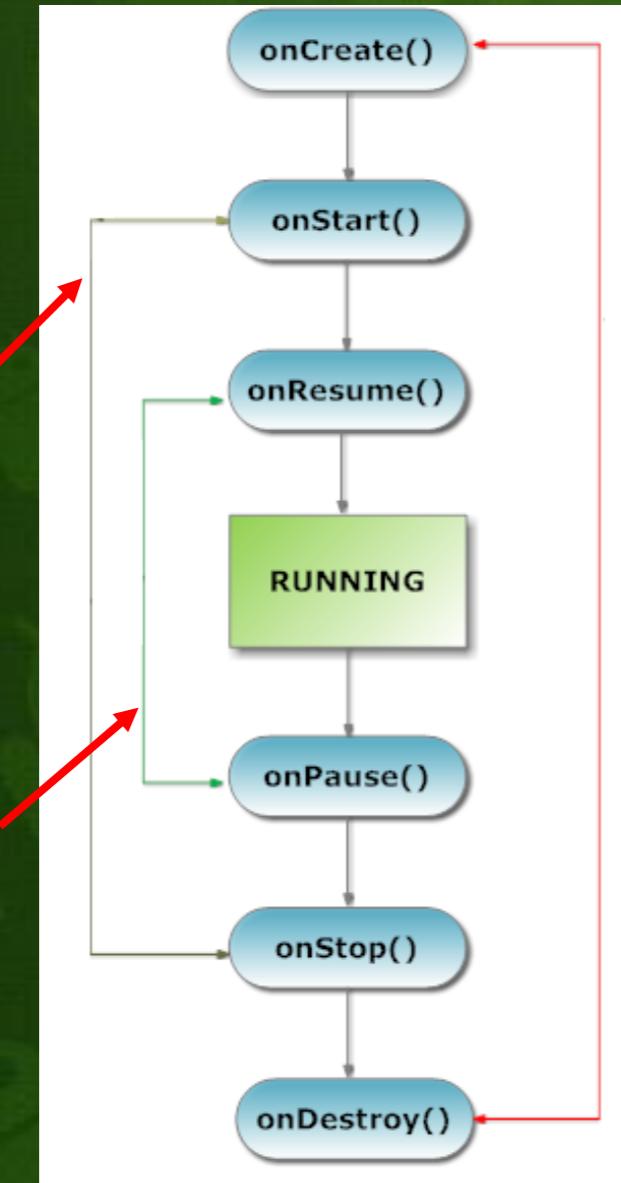


<http://developer.android.com/training/basics/activity-lifecycle/startling.html>

Visible Lifetime & Foreground Lifetime

Visible
Lifetime

Foreground
Lifetime



- **Visible Lifetime** xảy ra từ khi `onStart()` được gọi cho đến khi `onStop()` được gọi
- **Foreground Lifetime:** xảy ra từ khi `onResume()` được gọi cho đến khi `onPause()` được gọi

Thay đổi một số thông tin của ứng dụng



- *Thay đổi tên của ứng dụng*
- *Thay đổi hình ảnh của ứng dụng*

Hướng dẫn Download và cài đặt Genymotion

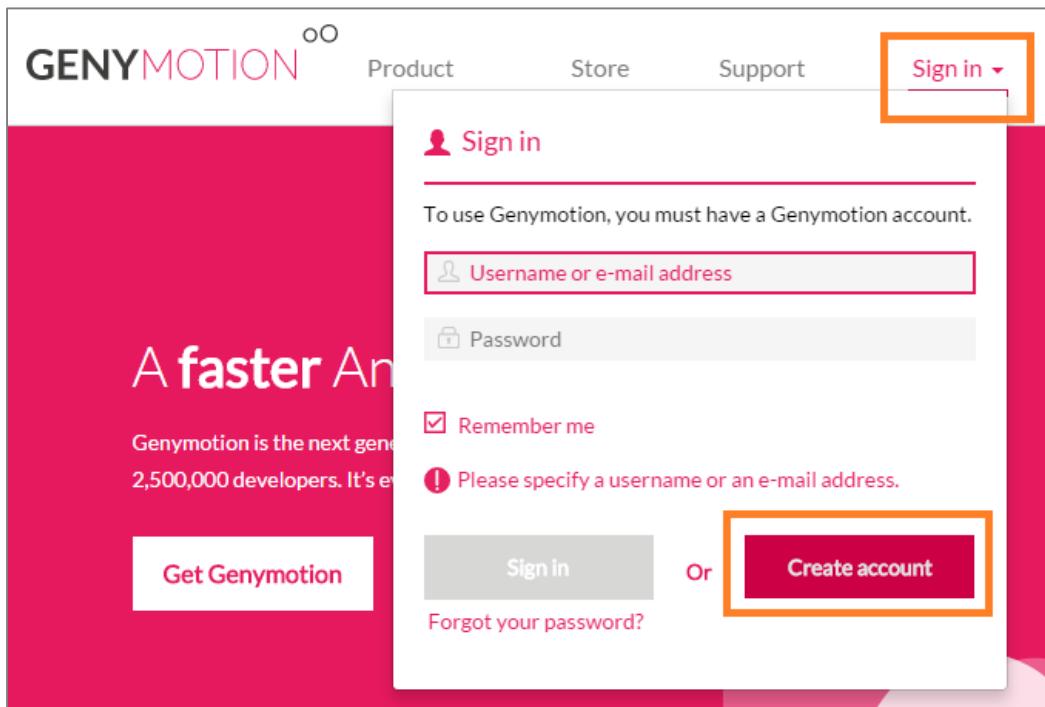
A. Genymotion là gì ?

- Genymotion là phần mềm giả lập hệ điều hành Android, phần mềm này giúp chúng ta trải nghiệm các ứng dụng hoặc game Android trên máy tính PC, đồng thời Genymotion cũng cho phép chúng ta cài đặt và sử dụng các ứng dụng từ Google Play một cách nhanh chóng và dễ dàng.
- Ngoài ra với việc định dạng rõ từng dòng máy Android và nhiều chức năng tiện ích khác, Genymotion ngày càng được sử dụng rộng rãi bởi những nhà phát triển ứng dụng chuyên nghiệp.
- Trang chủ <https://www.genymotion.com>

B. Đăng ký tài khoản tại Genymotion

Tuy Genymotion là một phần mềm miễn phí nhưng để download được phần mềm này thì chúng ta phải có một tài khoản tại Genymotion. Như vậy công việc đầu tiên thì chúng ta sẽ tiến hành đăng ký một tài khoản tại Genymotion

- **Bước 01:** Truy cập vào trang chủ của Genymotion tại địa chỉ <https://www.genymotion.com>
- **Bước 02:** Truy cập chức năng **Create Account** thông qua menu “**Sign in**”



- **Bước 03:** Điền đầy đủ thông tin vào form đăng ký

- Lưu ý địa chỉ email ở đây chúng ta phải cung cấp chính xác, bởi vì email này được dùng để kích hoạt tài khoản và thực hiện các giao dịch khác với Genymotion

Account creation

Profile information

hailan

Ithlan54@gmail.com

Your profile (optional)

Company size: Personal use

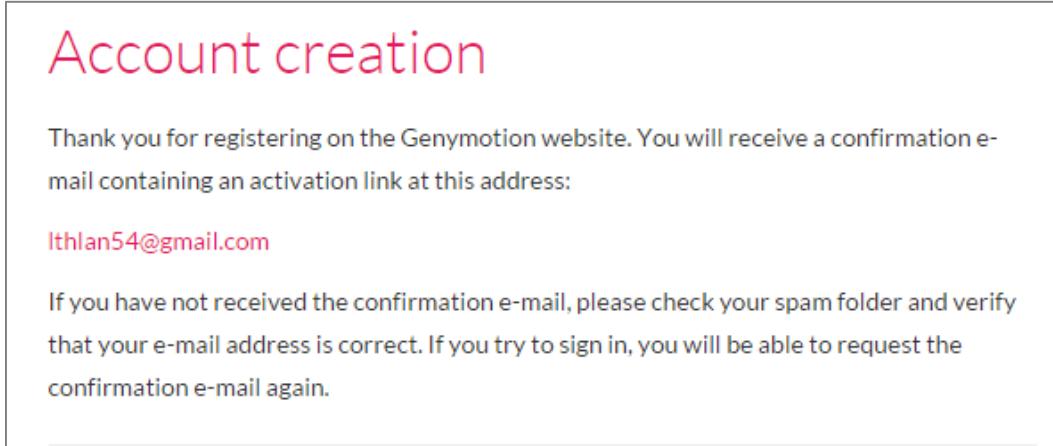
Usage type: Development

Allow Genymotion to send me e-mails about new releases

I accept terms of the [privacy statement](#)

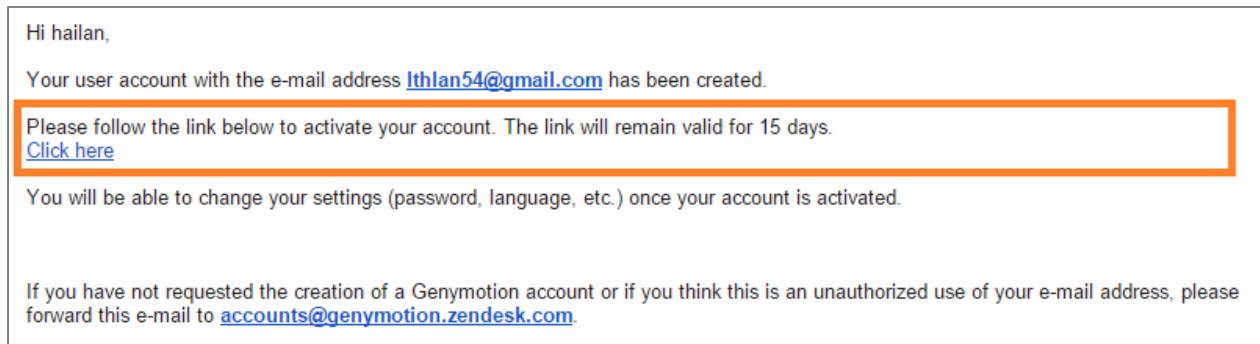
Create account

- Thông báo đăng ký thành công sẽ hiển thị như hình vẽ bên dưới



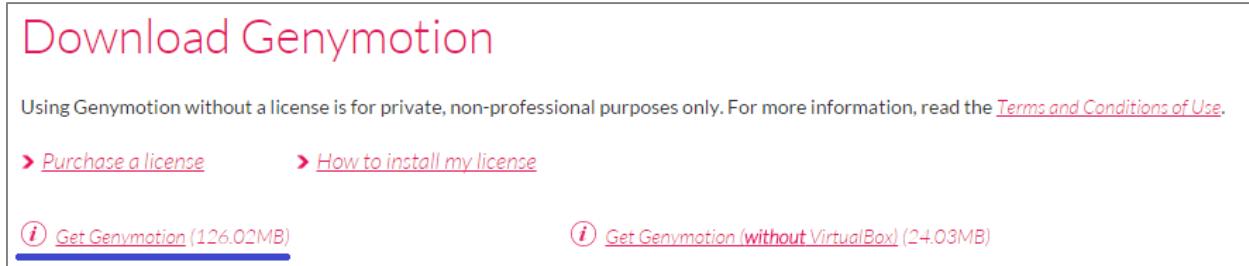
▪ **Bước 04: Kích hoạt tài khoản**

- Truy cập vào **hộp mail vừa đăng ký ở Bước 3** để kiểm tra, Genymotion sẽ gửi một email kèm một liên kết để chúng ta kích hoạt tài khoản. Chúng ta click vào liên kết “**Click here**” để tiến hành kích hoạt tài khoản vừa đăng ký

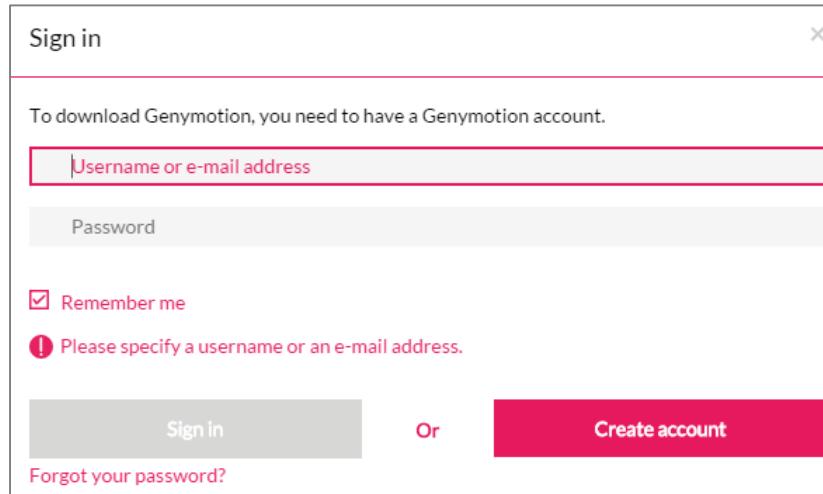


C. Download phần mềm

- **Bước 01:** Truy cập trang download của Genymotion <https://www.genymotion.com/#!/download> và chọn vào liên kết **Get Genymotion** (dung lượng khoảng 126.02MB và đính kèm luôn VirtualBox)



- **Bước 02:** Điền thông tin tài khoản vừa đăng ký ở Mục B để đăng nhập và download phần mềm

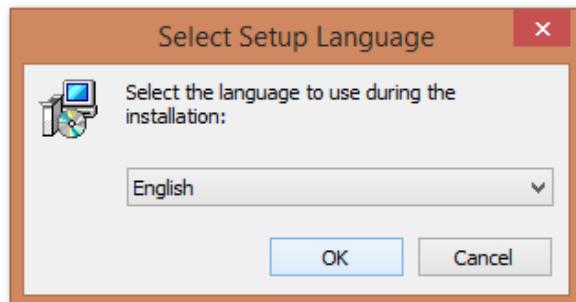


D. Cài đặt phần mềm

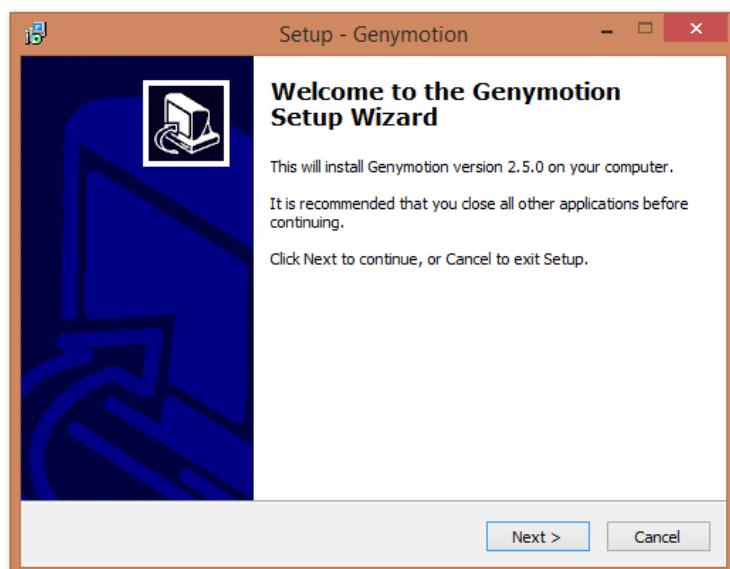
Sau khi download thành công phần mềm Genymotion, chúng ta sẽ được một tập tin với tên “**genymotion-2.5.0-vbox.exe**” với dung lượng gần 130mb (Tên của tập tin có thể thay đổi tùy vào phiên bản Genymotion mà chúng ta download được về máy)

Name	Type	Size	Date modified
genymotion-2.5.0-vbox.exe	Application	129,050 KB	7/13/2015 3:18 PM

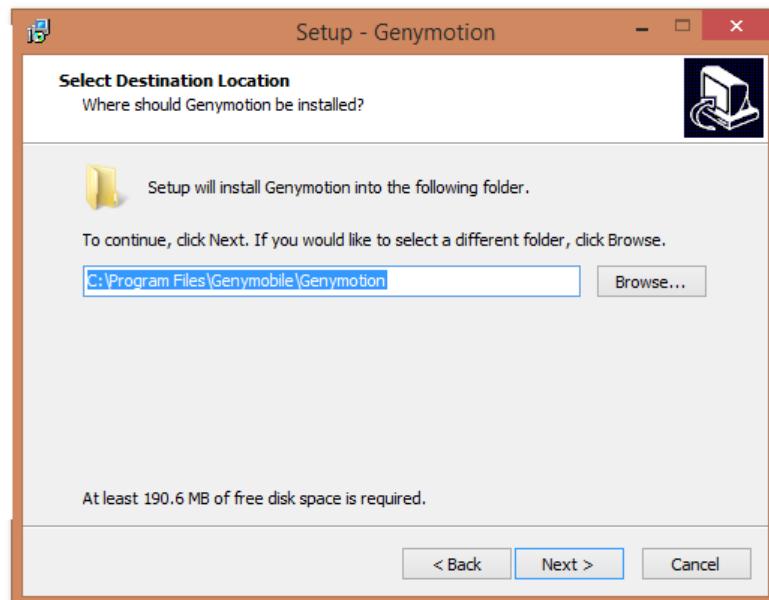
Double click vào tập tin (hoặc **Right Click + Open** hoặc **Right Click + Run as administrator**) để tiến hành cài đặt. Chúng ta nên để giá trị mặc định là ngôn ngữ English, sau đó nhấn OK



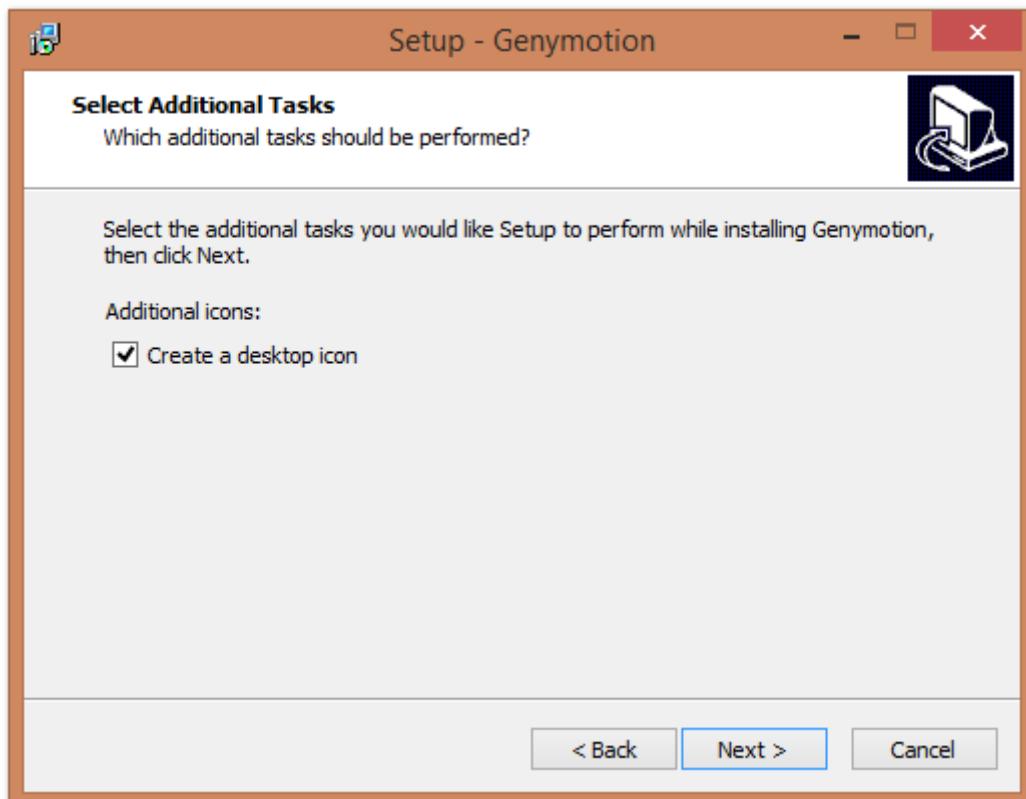
Tiếp tục click **Next**



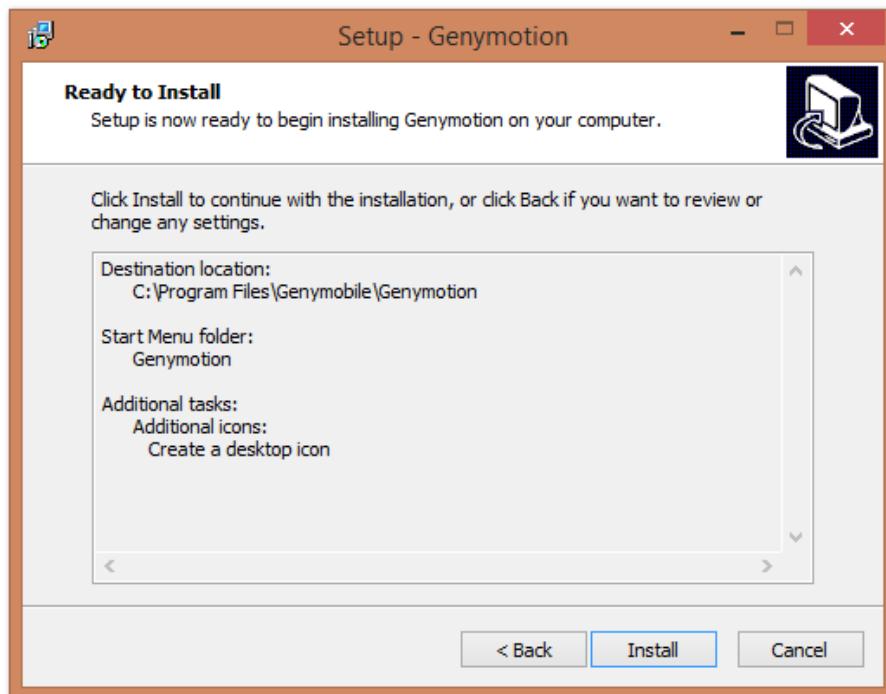
Thiếp lập thư mục chứa Genymotion khi cài đặt, ở đây chúng ta sẽ để giá trị mặc định “C:\Program Files\Genymobile\Genymotion”. Tiếp tục Click Next



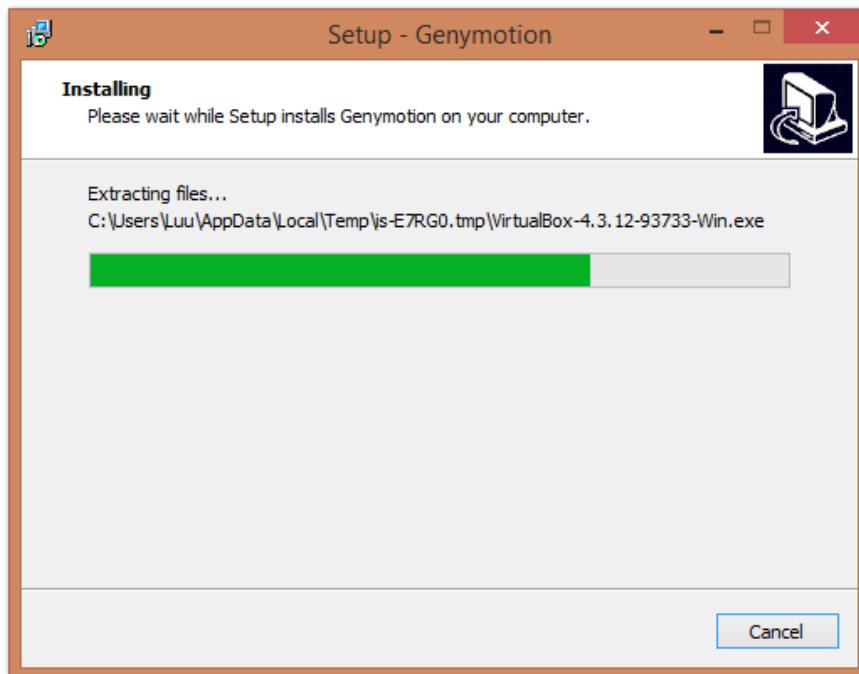
Tiếp tục click Next để tạo icon ngoài Desktop



Click **Install** để bắt đầu quá trình cài đặt



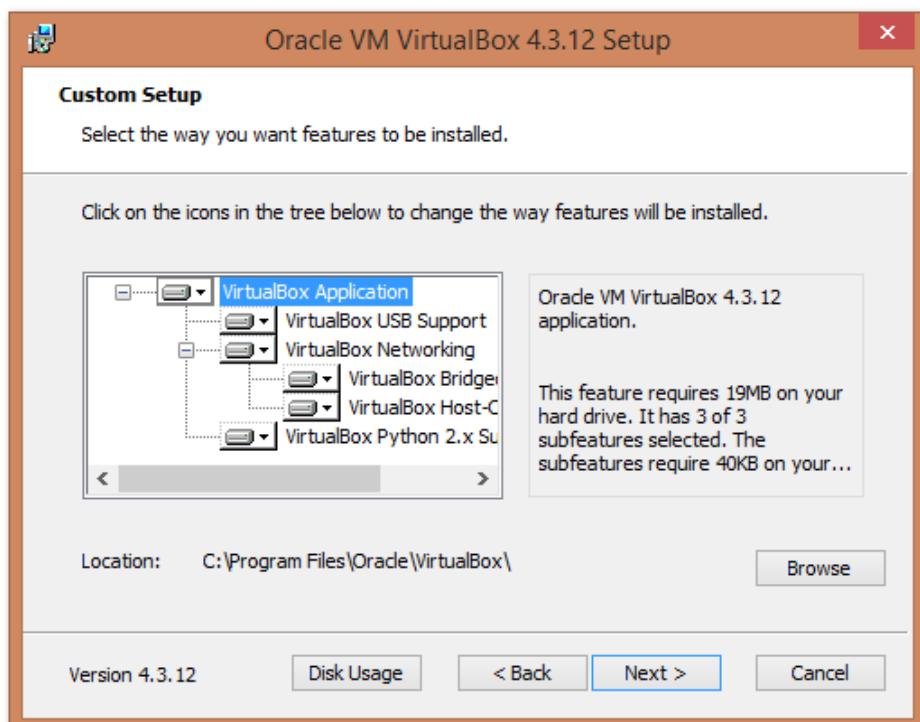
Quá trình cài đặt đang diễn ra



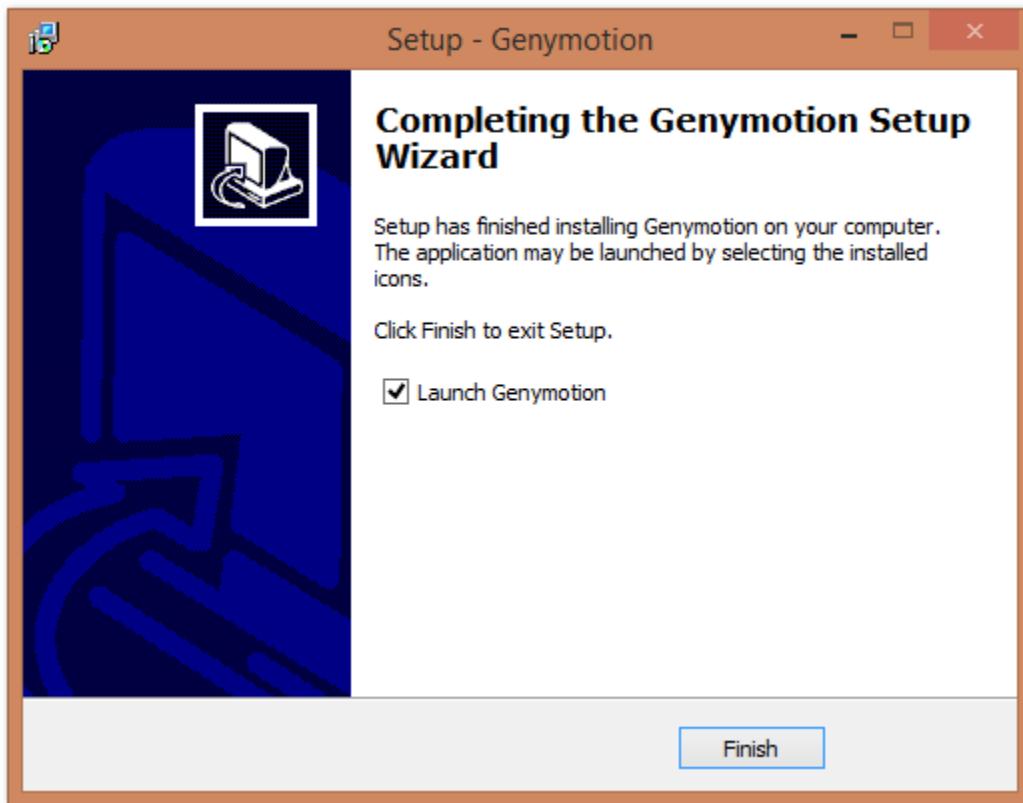
Trong quá trình cài đặt, Genymotion sẽ **cài đặt thêm VirtualBox**, đây là phần mềm đã được tích hợp sẵn trong file cài đặt, chúng ta **bắt buộc** phải cài đặt phần mềm này để sử dụng tốt các chức năng của Genymotion.



Giữ các giá trị mặc định và tiếp tục click **Next**



Quá trình cài đặt Genymotion thành công.



Tiếp tục nhấn Finish để khởi động Genymotion. Sau khi cài đặt thành công chúng ta sẽ thấy ở màn hình desktop xuất hiện 3 icon sau



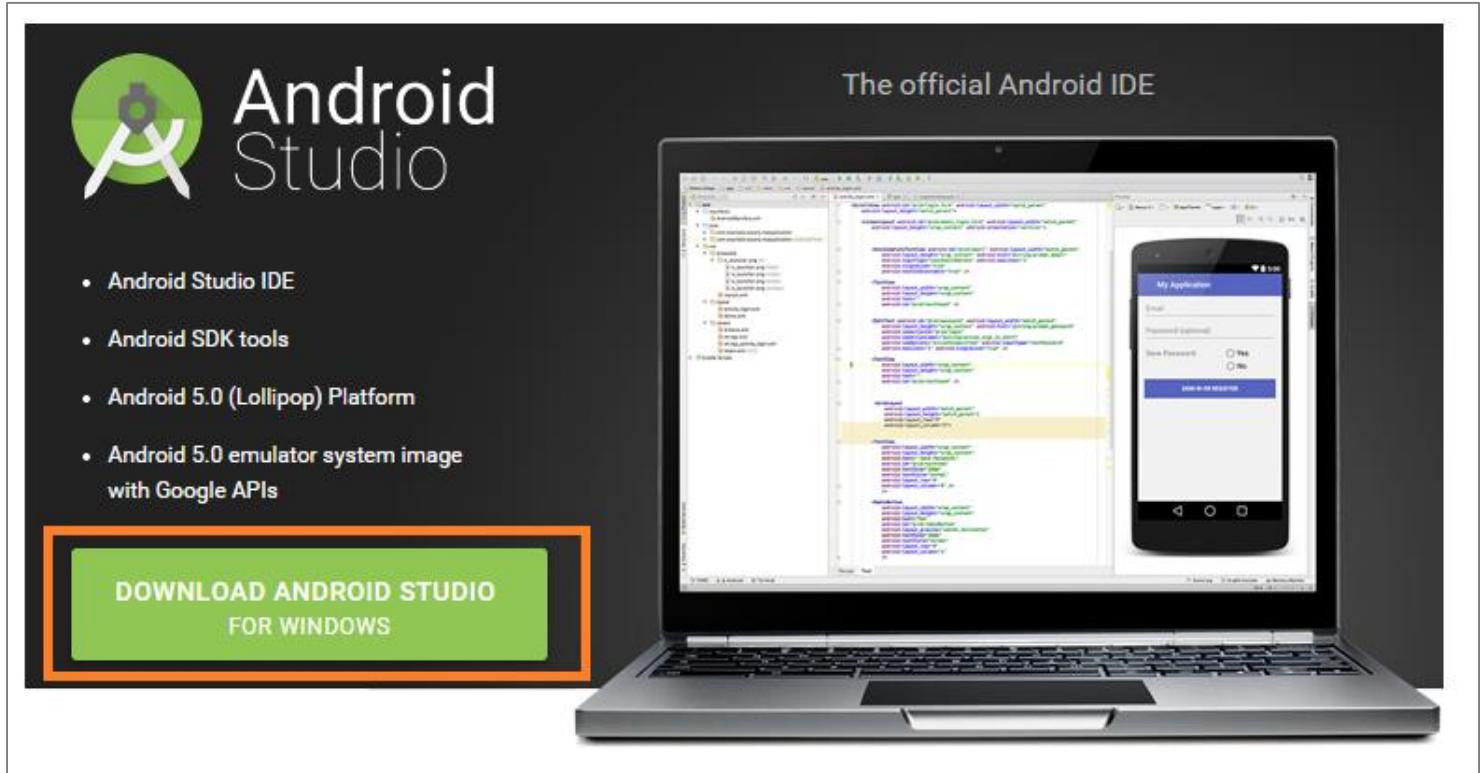
Hướng dẫn Download và cài đặt Android Studio

A. Android Studio là gì ?

- Android Studio là một phần mềm miễn phí (IDE) được phát triển bởi Google, giúp chúng ta phát triển các ứng dụng Android một cách nhanh chóng và hiệu quả nhất.
- Trang chủ <https://developer.android.com/sdk/index.html>

B. Download phần mềm

- **Bước 01:** Truy cập trang download của phần mềm Android Studio tại địa chỉ <https://developer.android.com/sdk/index.html> và chọn vào liên kết **Download Android Studio For Windows**



- **Bước 02:** Đọc các điều khoản và check vào tùy chọn “**I have read ... and conditions**”, tiếp tục click vào **Download Android Studio For Windows**

Download

Before installing Android Studio or the standalone SDK tools, you must agree to the following terms and conditions.

1.2 "Android" means the Android software stack for devices, as made available under the Android Open Source Project, which is located at the following URL: <http://source.android.com/>, as updated from time to time.

1.3 "Google" means Google Inc., a Delaware corporation with principal place of business at 1600 Amphitheatre Parkway, Mountain View, CA 94043, United States.

2. Accepting this License Agreement

2.1 In order to use the SDK, you must first agree to this License Agreement. You may not use the SDK if you do not accept this License Agreement.

2.2 By clicking to accept, you hereby agree to the terms of this License Agreement.

2.3 You may not use the SDK and may not accept the License Agreement if you are a person barred from receiving

I have read and agree with the above terms and conditions

DOWNLOAD ANDROID STUDIO FOR WINDOWS

C. Cài đặt phần mềm

Lưu ý: Để cài đặt được phần mềm Android Studio yêu cầu máy tính đã cài đặt JDK, và phần hướng dẫn cài đặt JDK chúng ta đã được hướng dẫn trong khóa học “Lập trình Java” nên tôi sẽ không trình bày lại, nếu bạn nào chưa cài đặt được JDK có thể tham khảo video sau <https://www.youtube.com/watch?v=e6x1NWuVh-4>

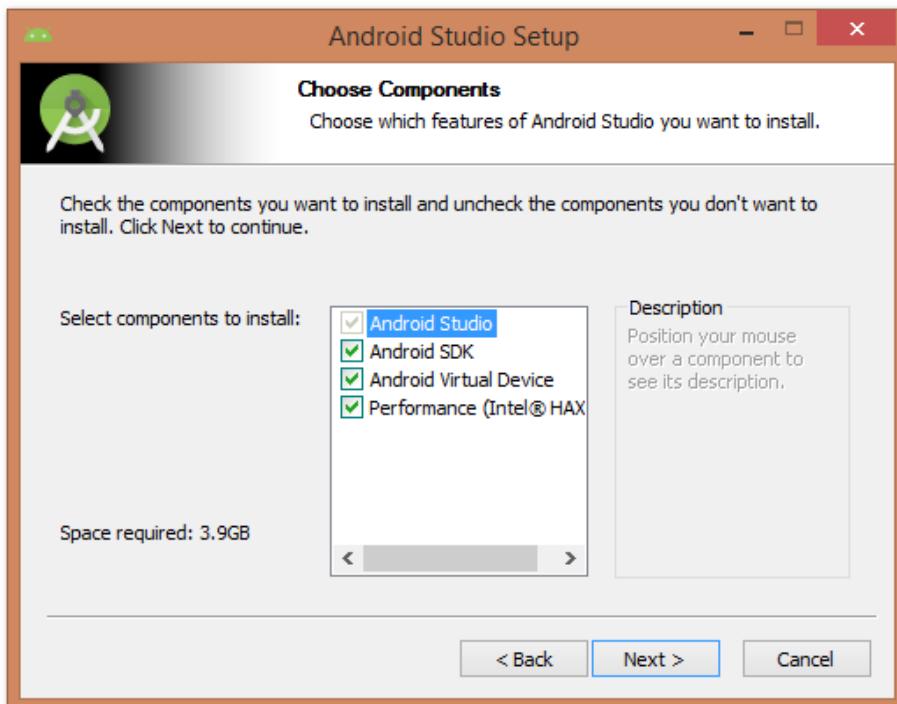
Sau khi download thành công phần mềm Android Studio, chúng ta sẽ được tập tin “**android-studio-bundle-141.1980579-windows.exe**” với dung lượng gần 910mb (Tên của tập tin có thể thay đổi tùy vào phiên bản phần mềm mà chúng ta download được về máy)

Name	Type	Size	Date modified
 android-studio-bundle-141.1980579-windows.exe	Application	908,650 KB	7/9/2015 2:11 PM

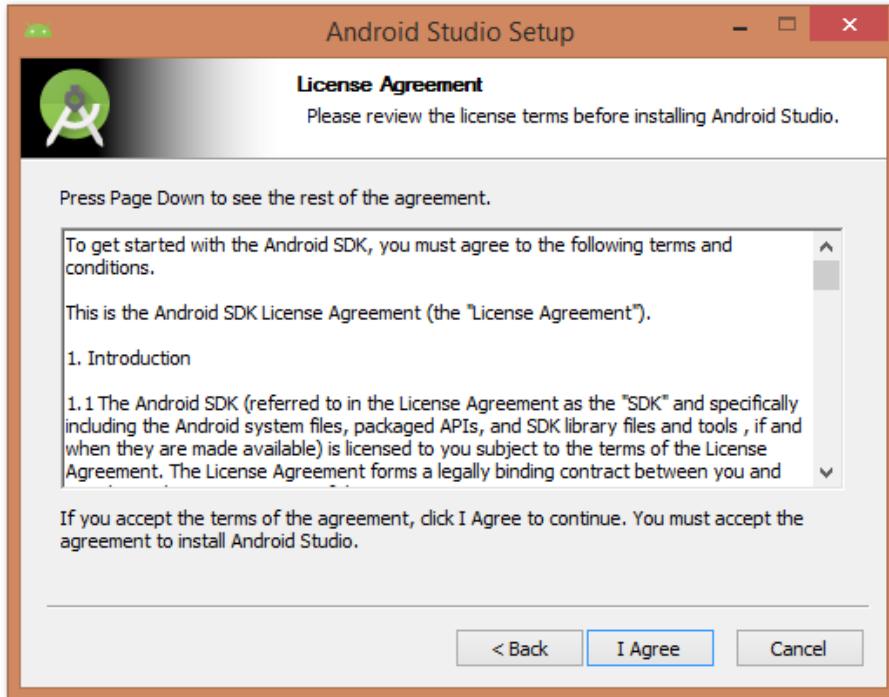
Double click vào tập tin (hoặc Right Click + Open hoặc Right Click + Run as administrator) để tiến hành cài đặt. Chúng ta nên để giá trị mặc định là ngôn ngữ English, sau đó nhấn OK và hộp thoại **Welcome to Setup Android Studio** sẽ xuất hiện:



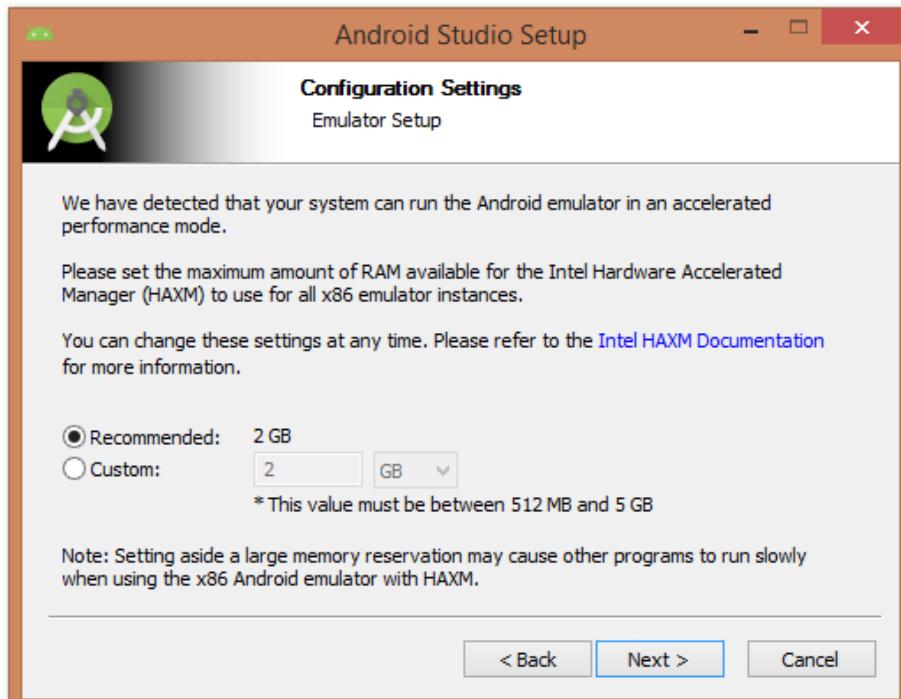
Tiếp tục click **Next**



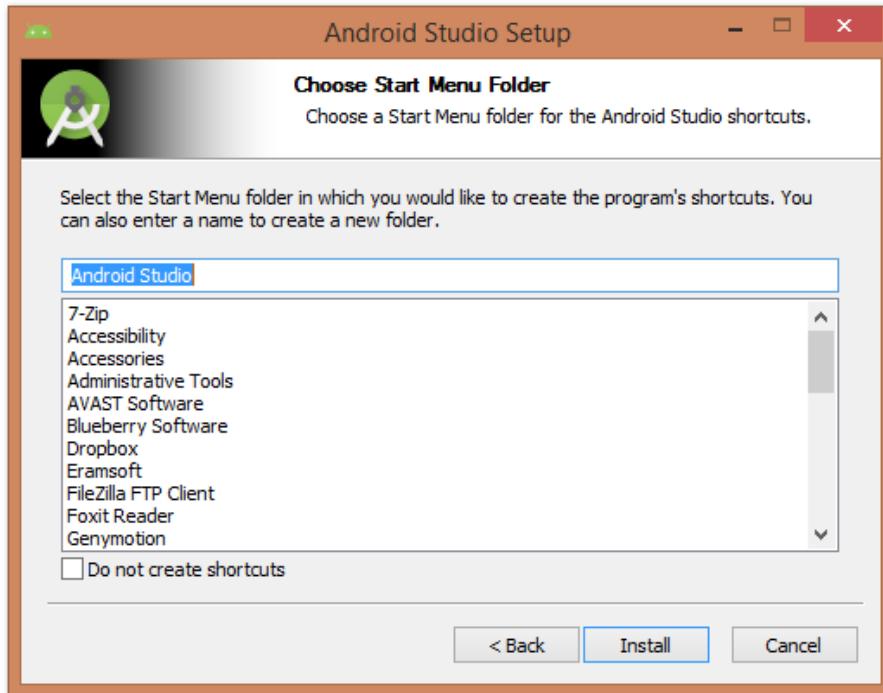
Chúng ta nên giữ nguyên các tùy chọn này sau đó nhấn **Next** để tiếp tục. Hộp loại thông báo các điều khoản và một số lưu ý khi sử dụng Android Studio xuất hiện, chúng ta tiếp tục click **I Agree**



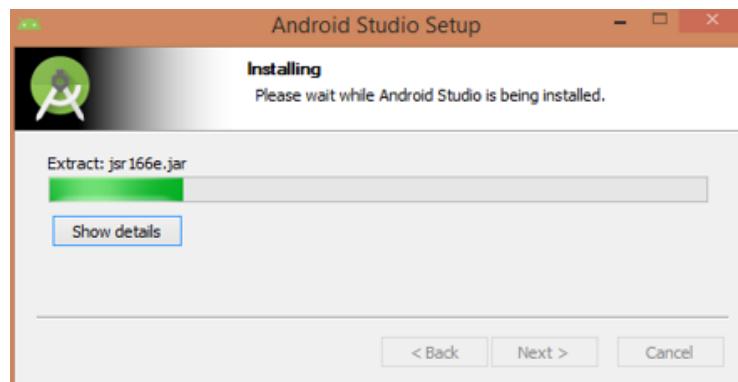
Phần cấu hình yêu cầu cài đặt chúng ta nên để giá trị mặc định, sau đó nhấn **Next**



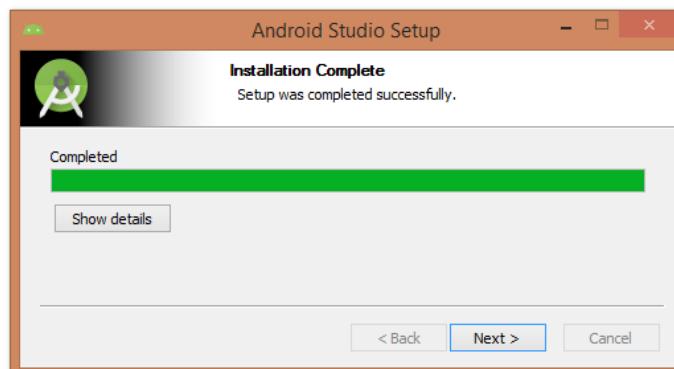
Click **Install** để bắt đầu quá trình cài đặt



Quá trình cài đặt đang diễn ra

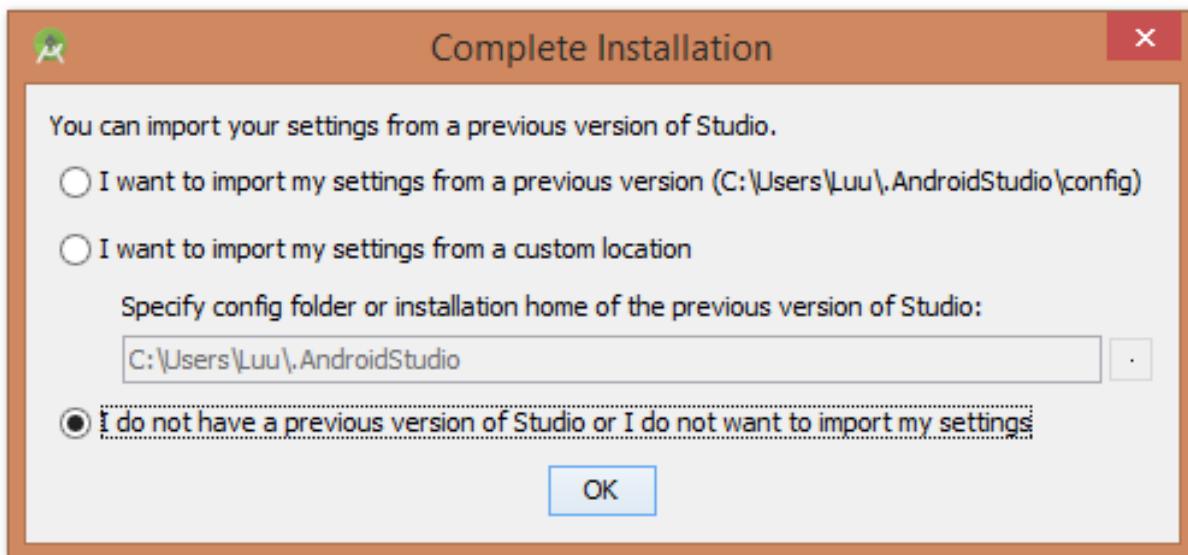


Quá trình cài đặt Android Studio thành công

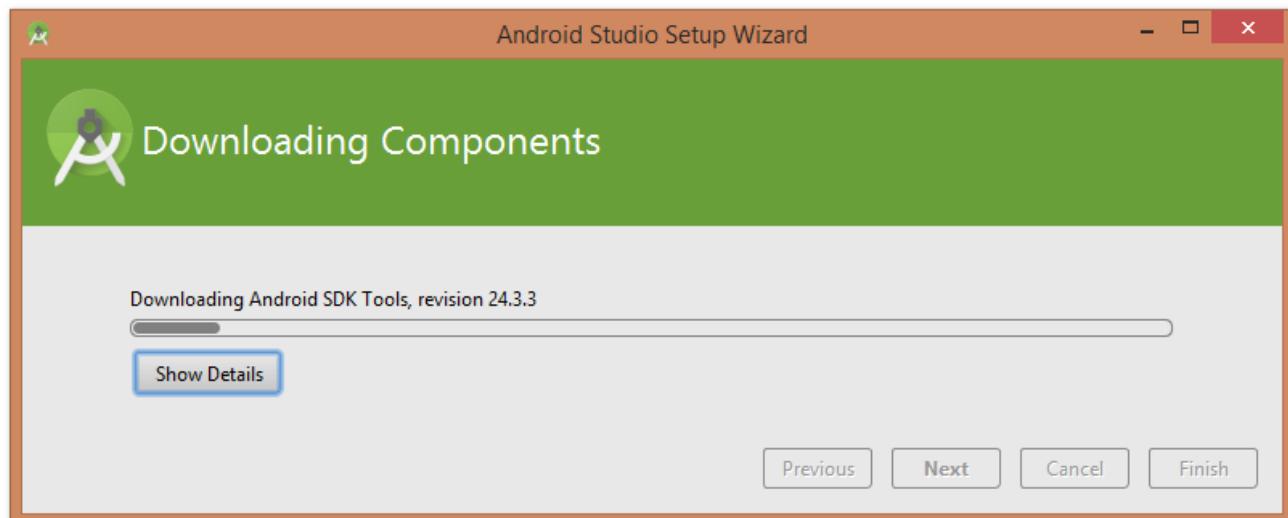




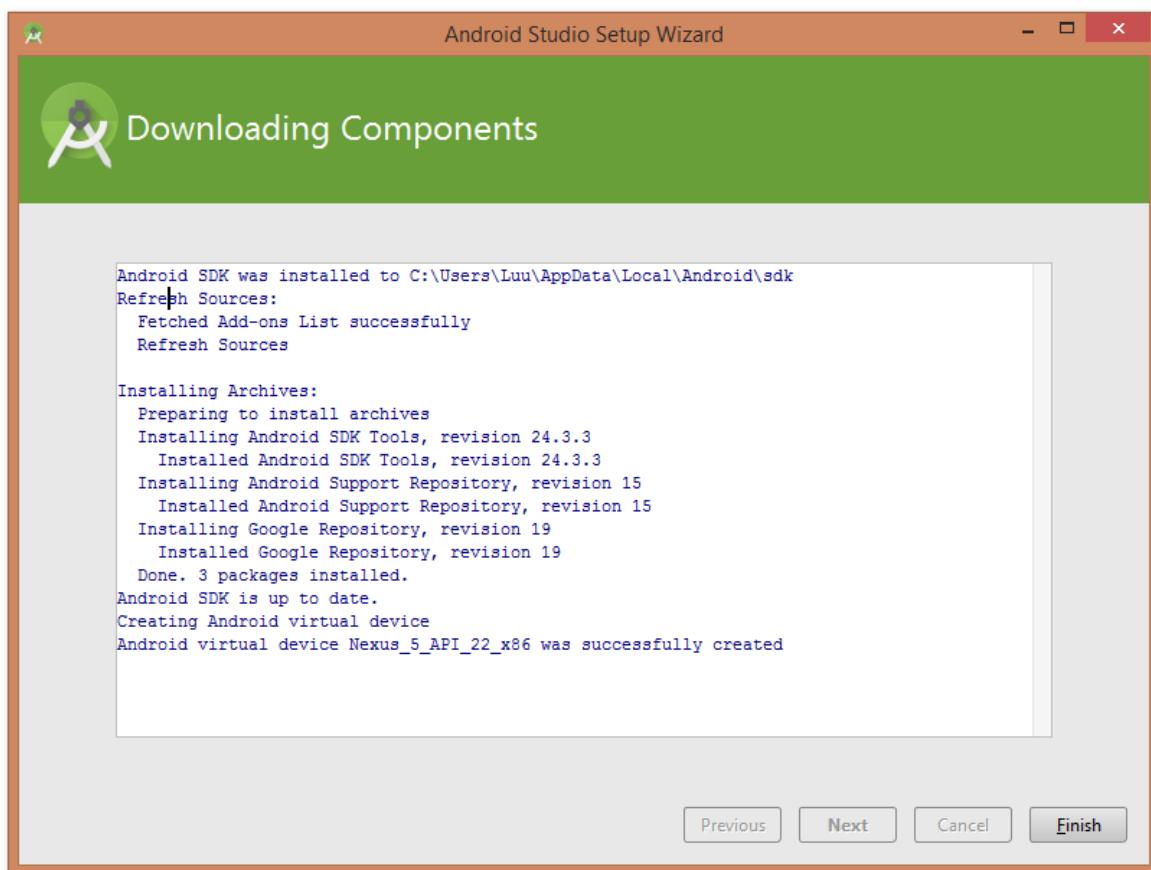
Tiếp tục click **Finish** để khởi động Android Studio. Ở lần khởi động đầu tiên, một hộp thoại import bản Android Studio cũ sẽ xuất hiện, vì đây là lần cài đặt và sử dụng Android Studio đầu tiên nên chúng ta sẽ chọn như hình dưới.



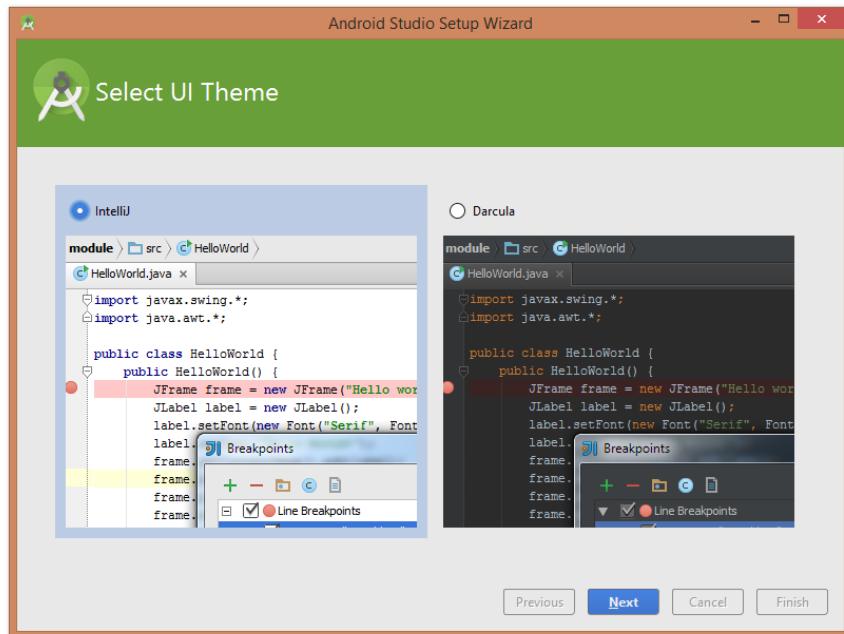
Ở lần đầu tiên này, chúng ta sẽ phải update một vài thứ cho quá trình làm việc sau này. Vui lòng chờ đợi cho đến khi hoàn tất.



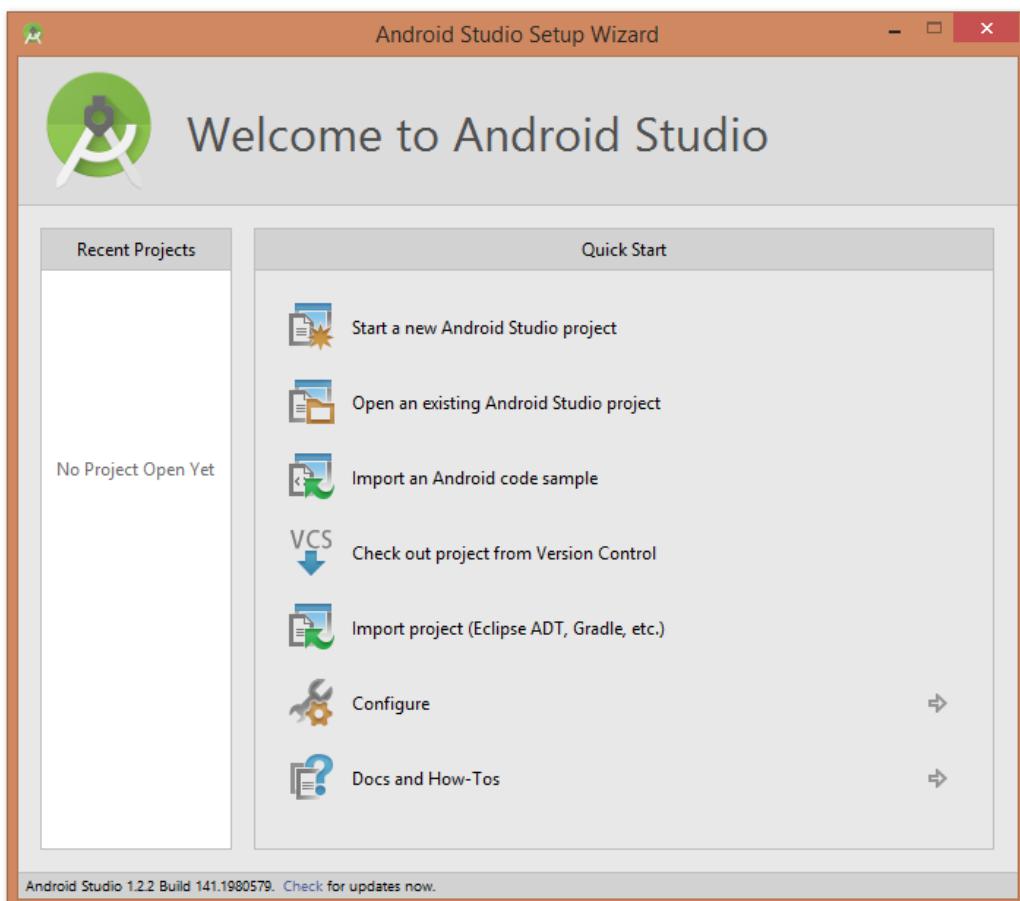
Quá trình update thành công



Chọn giao diện làm việc với Android Studio, ở đây tôi chọn theme IntelliJ, sau đó tiếp tục click **Next**

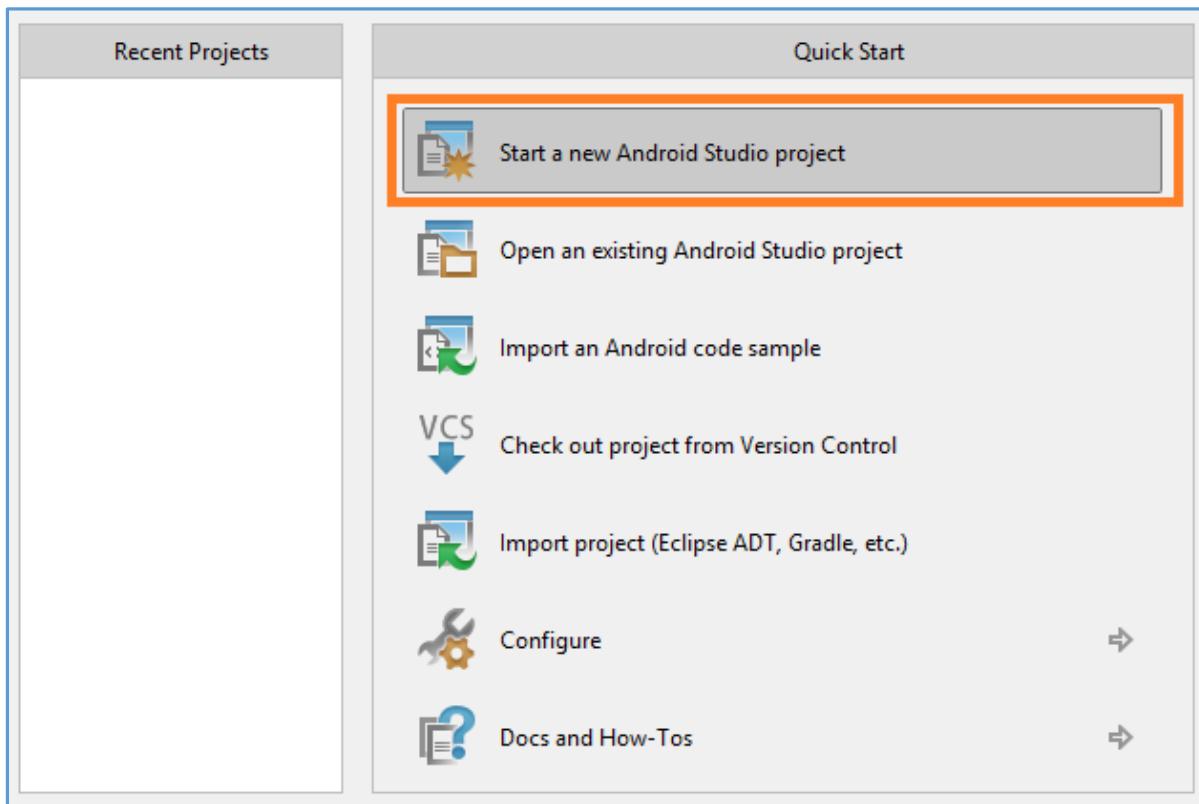


Màn hình khởi động xuất hiện chứng tỏ quá trình cài đặt đã thành công



Hướng dẫn tạo Project trong Android Studio

- Khởi động phần mềm Android Studio, tại giao diện của màn hình Welcome chúng ta tiếp tục chọn “**Start a new Android Studio project**” để tiến hành tạo mới một Project



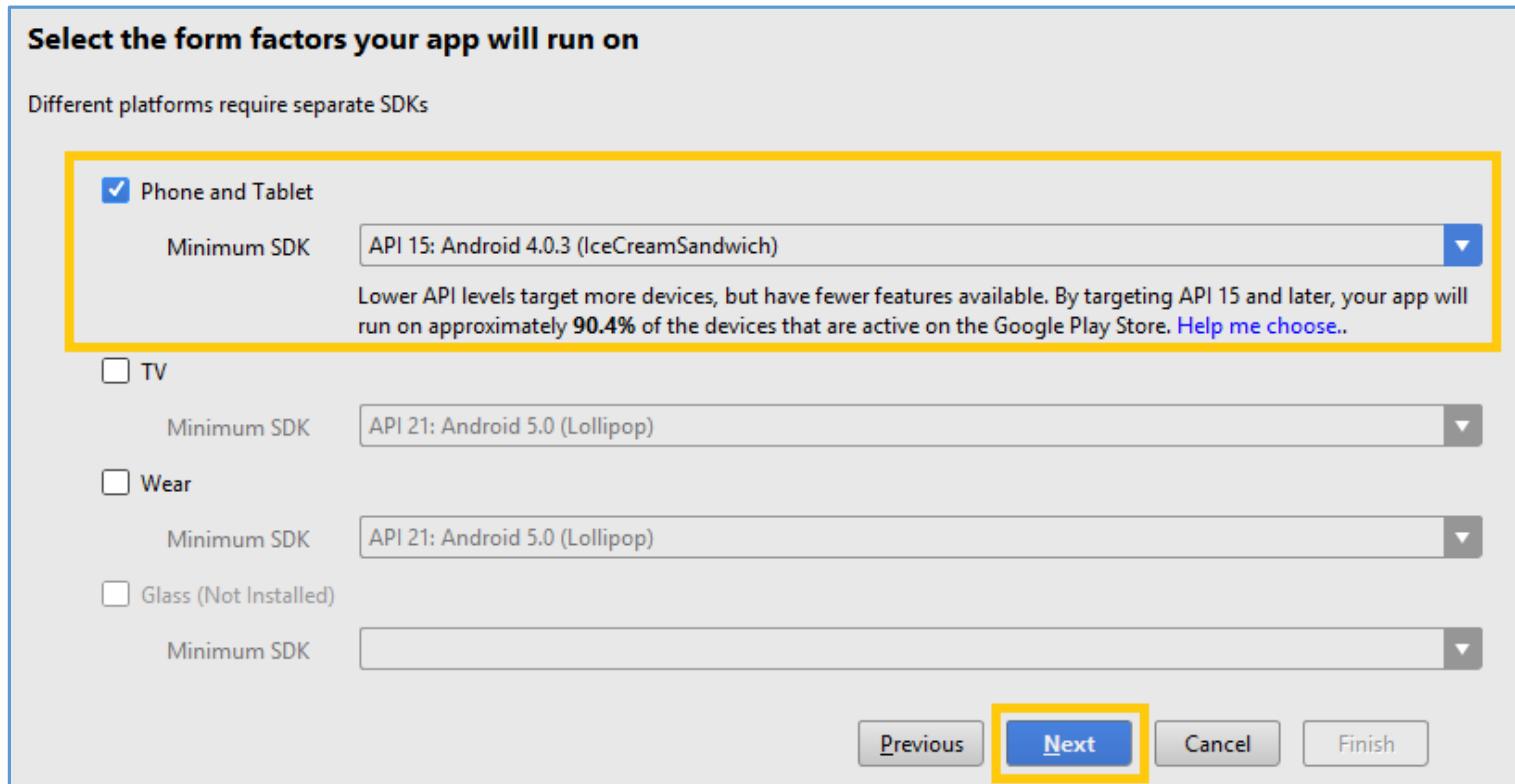
- Cửa sổ Cấu hình Project sẽ hiển thị như hình bên dưới, chúng ta lưu ý các giá trị sau:

The dialog box is titled "Configure your new project". It contains the following fields:

- Application name:** MyAndroid
- Company Domain:** zend.vn
- Package name:** myandroid.zend.vn [Edit](#)
- Project location:** C:\Users\Luu\AndroidStudioProjects\MyAndroid

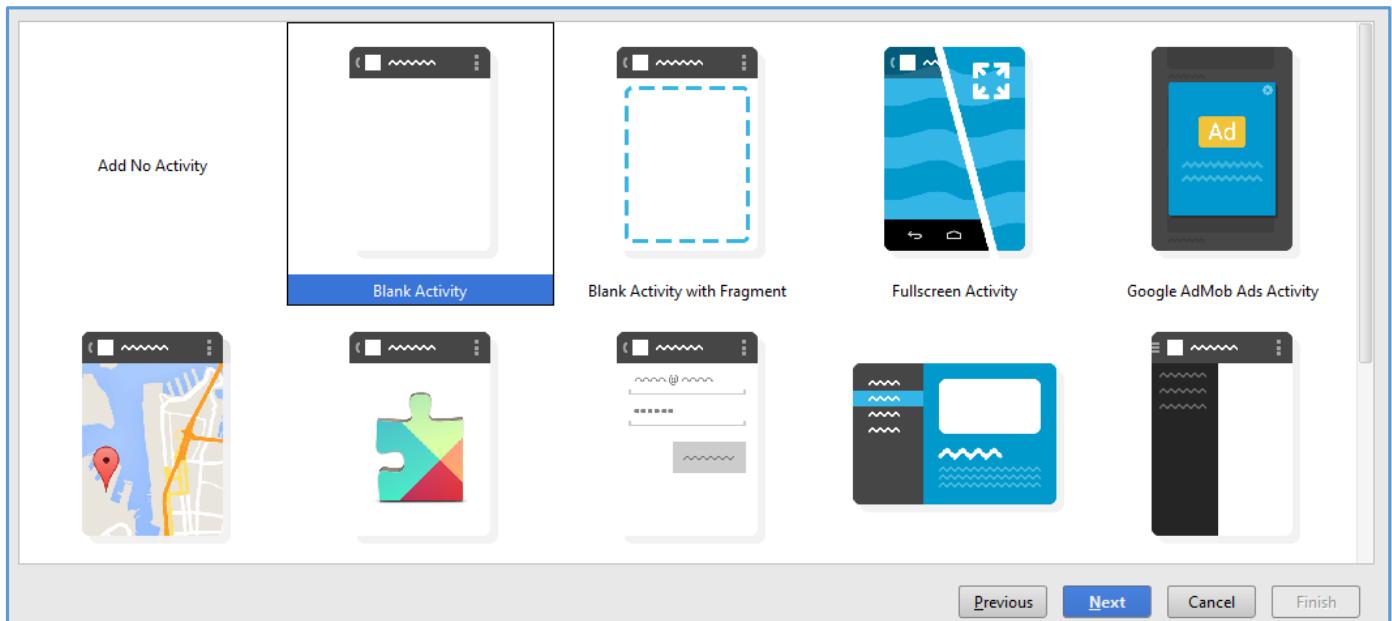
At the bottom are four buttons: "Previous", "Next" (highlighted in blue), "Cancel", and "Finish".

- **Application name:** tên của ứng dụng
- **Company Domain:** tên miền của công ty hoặc tổ chức bạn đang công tác (giá trị này sẽ được tự động thêm vào cho phần **Package name** ở phía dưới)
- **Package name:** mỗi ứng dụng nên có Package name phân biệt và duy nhất (Quy tắc đặt tên tương tự như cách đặt tên các package trong JAVA)
- **Project location:** lựa chọn nơi lưu trữ các mã nguồn của ứng dụng. Chúng ta nên lưu ý về đường dẫn này vì đây sẽ là nơi chứa tất cả các mã nguồn liên quan đến ứng dụng. Trong trường hợp muốn thay đổi thư mục này thì bạn chỉ cần click vào **Button “...”** và chọn vào folder mà bạn muốn chứa mã nguồn
- Sau khi điền đầy đủ thông tin đối với phần cấu hình Project, chúng ta tiếp tục click vào **nút Next**

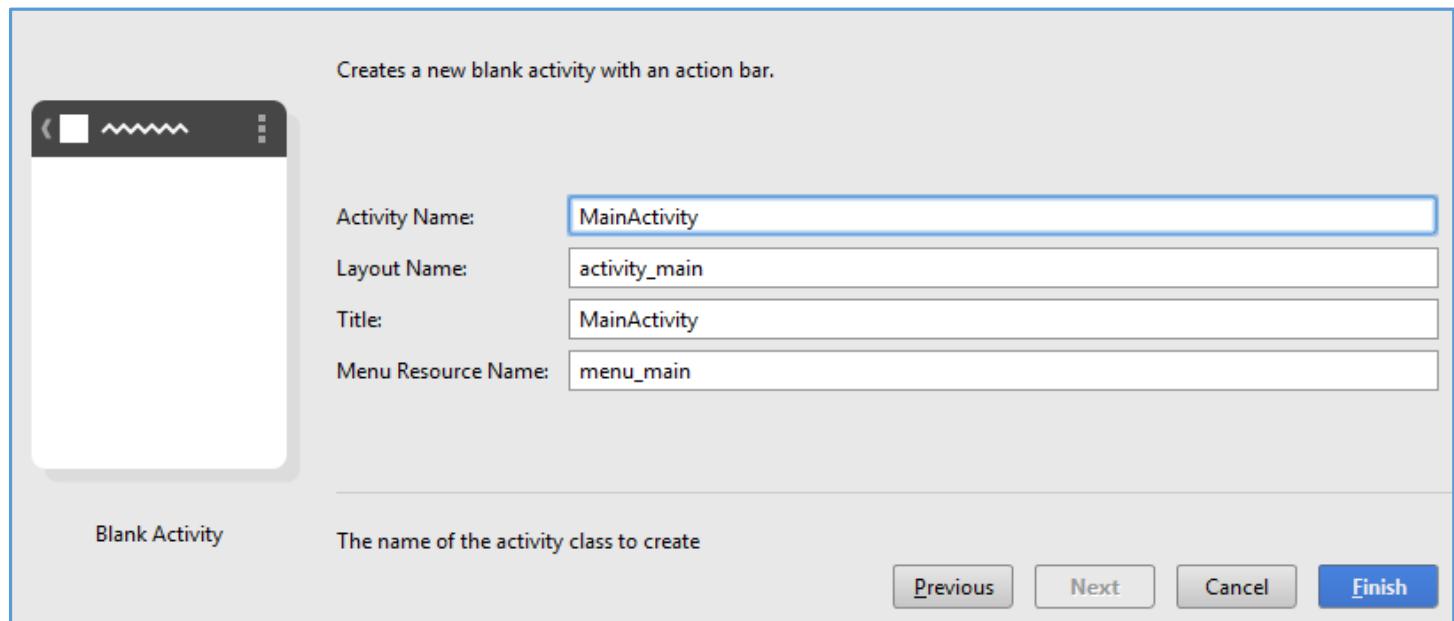


- **Lựa chọn loại thiết bị để xây dựng ứng dụng:** ở đây chúng ta xây dựng ứng dụng cho các thiết bị di động do đó chúng ta sẽ chọn vào tùy chọn **Phone and Tablet**
- **Lựa chọn Minimum SDK:** Chọn giới hạn API thấp nhất mà ứng dụng có thể cài đặt trên API đó
- Sau khi chọn đầy đủ các thông tin trên chúng ta click **Next** để tiếp tục

- Cửa sổ lựa chọn các loại giao diện sẽ xuất hiện, ở đây chúng ta chỉ mới chập chững tìm hiểu về Lập trình Android do đó chúng ta sẽ chọn vào loại giao diện đơn giản nhất “**Blank Activity**”, và tiếp tục click **Next**



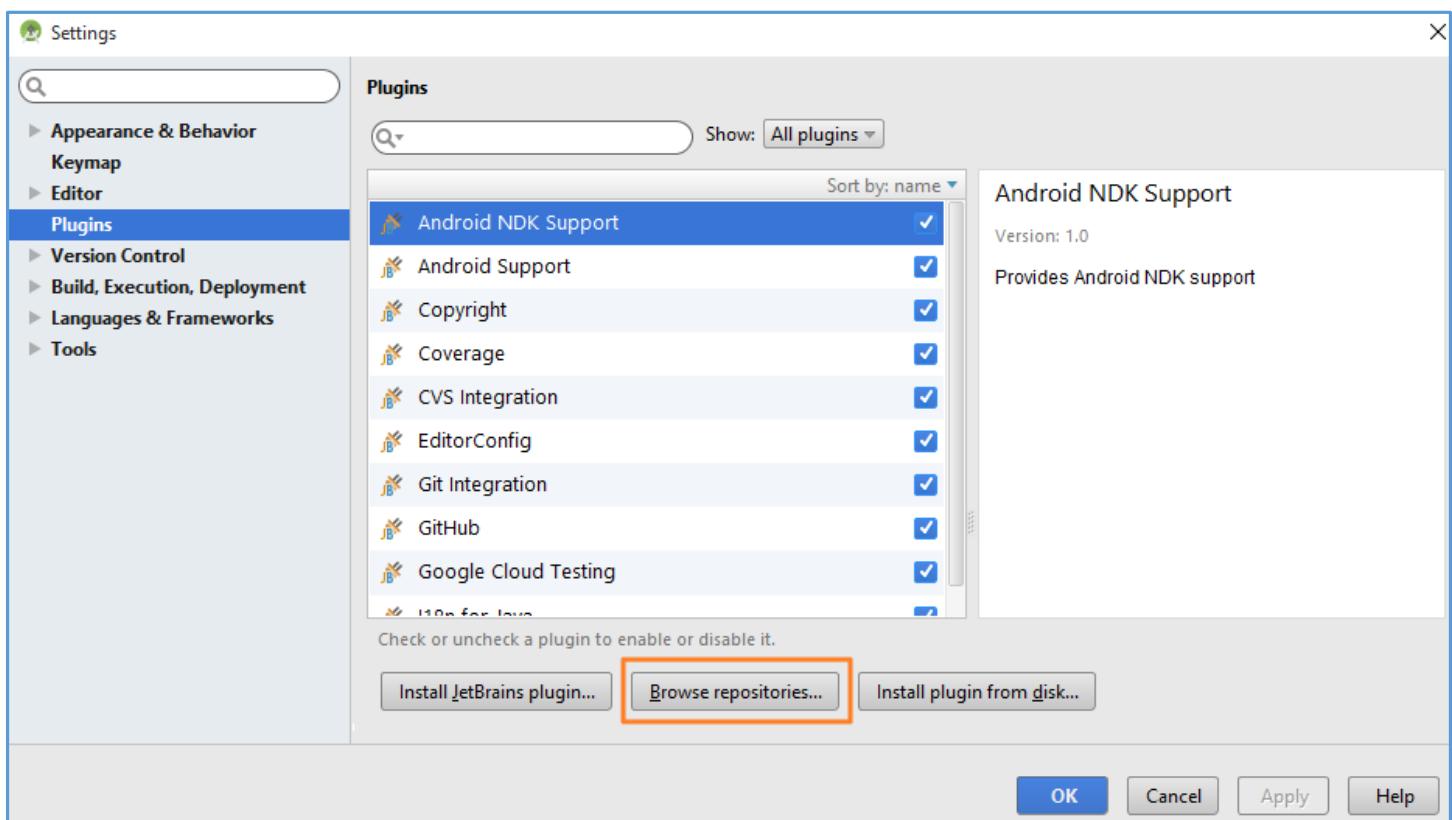
- Ở cửa sổ mới chúng ta sẽ tiến hành tạo mới một Activity cho ứng dụng (khái niệm Activity tạm thời chúng ta xem như là một cửa sổ giao diện), các thông tin này chúng ta sẽ để mặc định và tiếp tục click vào **Finish** để hoàn thành việc tạo ra một project trong Android Studio



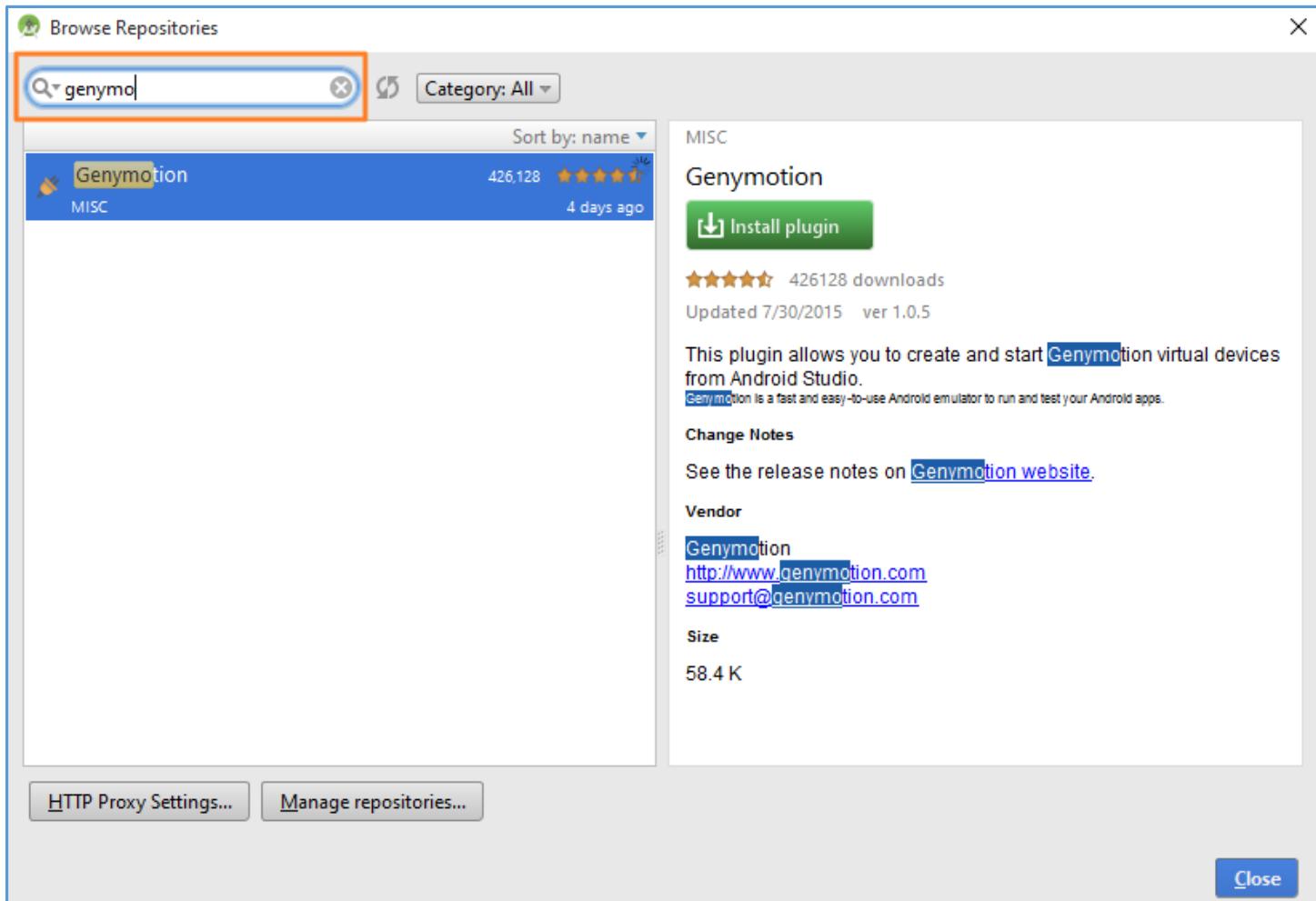
Cài đặt và cấu hình Plugin Genymotion trong Android Studio

A. Cài đặt Plugin Genymotion trong Android Studio

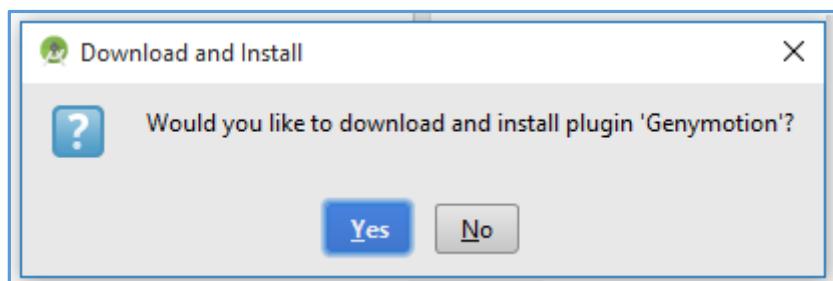
- Khởi động phần mềm Android Studio
- Vào menu **File → Setting** (hoặc nhấn tổ hợp phím **Ctrl + Alt + S**)
- Tiếp tục chọn **menu Plugins**, bạn sẽ thấy một cửa sổ giao diện xuất hiện như hình bên dưới



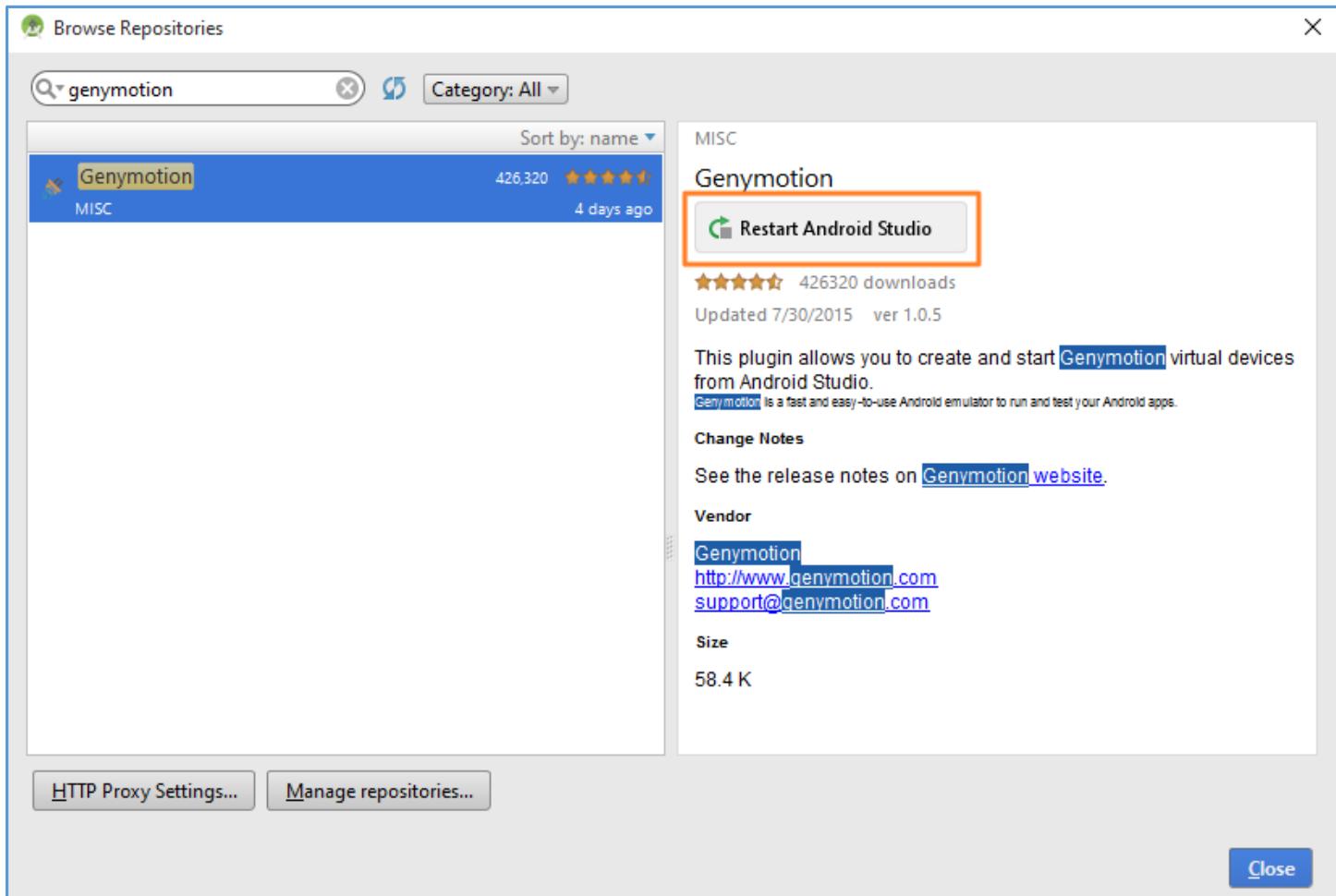
- Tiếp tục chọn vào button “**Browse repositories...**” để tìm kiếm các plugins. Sau khi click vào button này một cửa sổ mới sẽ xuất hiện



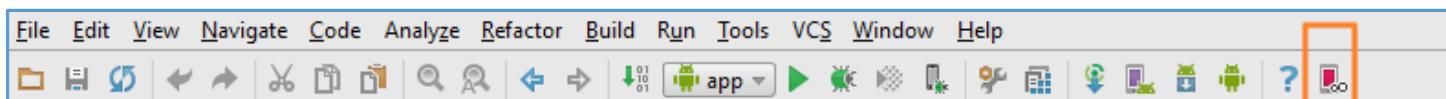
- Tại cửa sổ này chúng ta nhập vào ô tìm kiếm **từ khóa Genymotion** để tìm kiếm plugin Genymotion.
- Plugin Genymotion** sẽ xuất hiện trong khung kết quả, chúng ta chỉ việc click chọn vào nó và tiếp tục click **Install plugin** để bắt đầu quá trình cài đặt



- Tiếp tục **click Yes** để xác định lại chúng ta muốn download và cài đặt plugin Genymotion

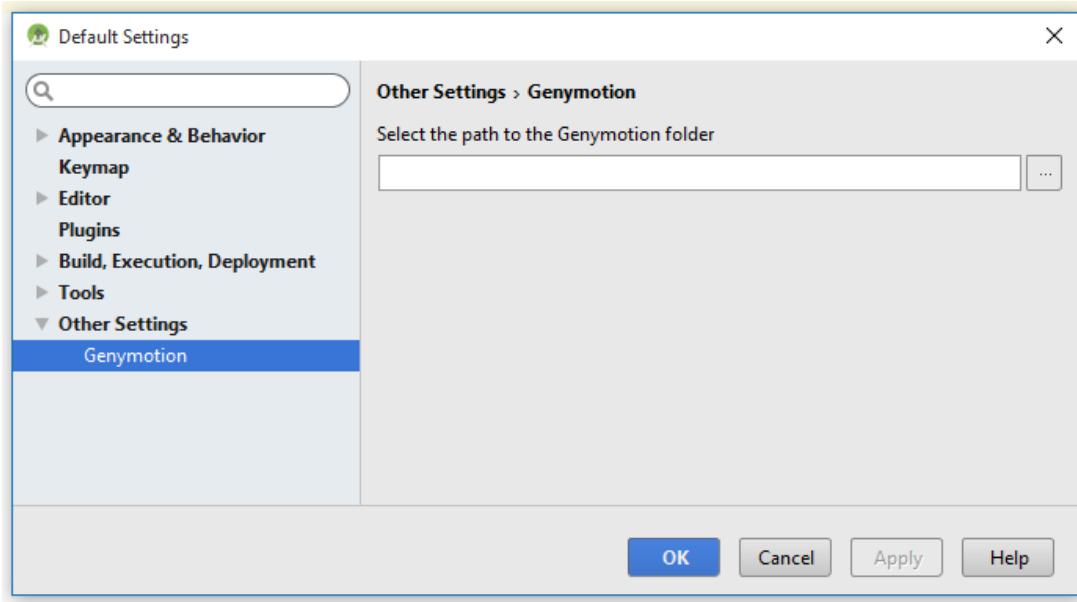


- Sau khi cài đặt xong Plugin Genymotion, chúng ta click vào “**Restart Android Studio**” để khởi động lại phần mềm Android Studio
- Sau khi khởi động lại Android Studio, chúng ta thấy icon của plugin Genymotion sẽ xuất hiện trên thanh Toolbar, điều này chứng tỏ Plugin Genymotion đã được cài đặt thành công

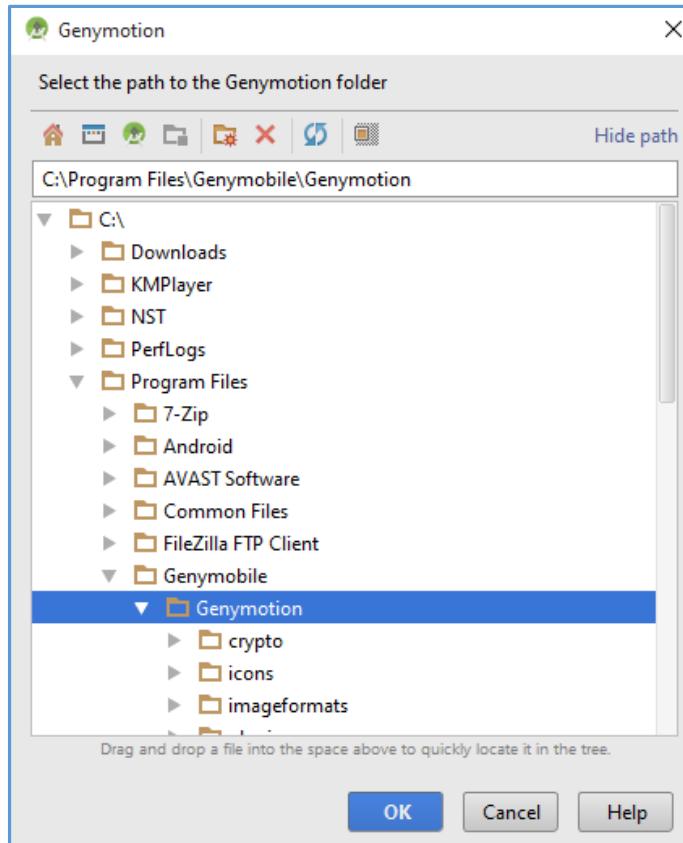


B. Cấu hình Plugin Genymotion trong Android Studio

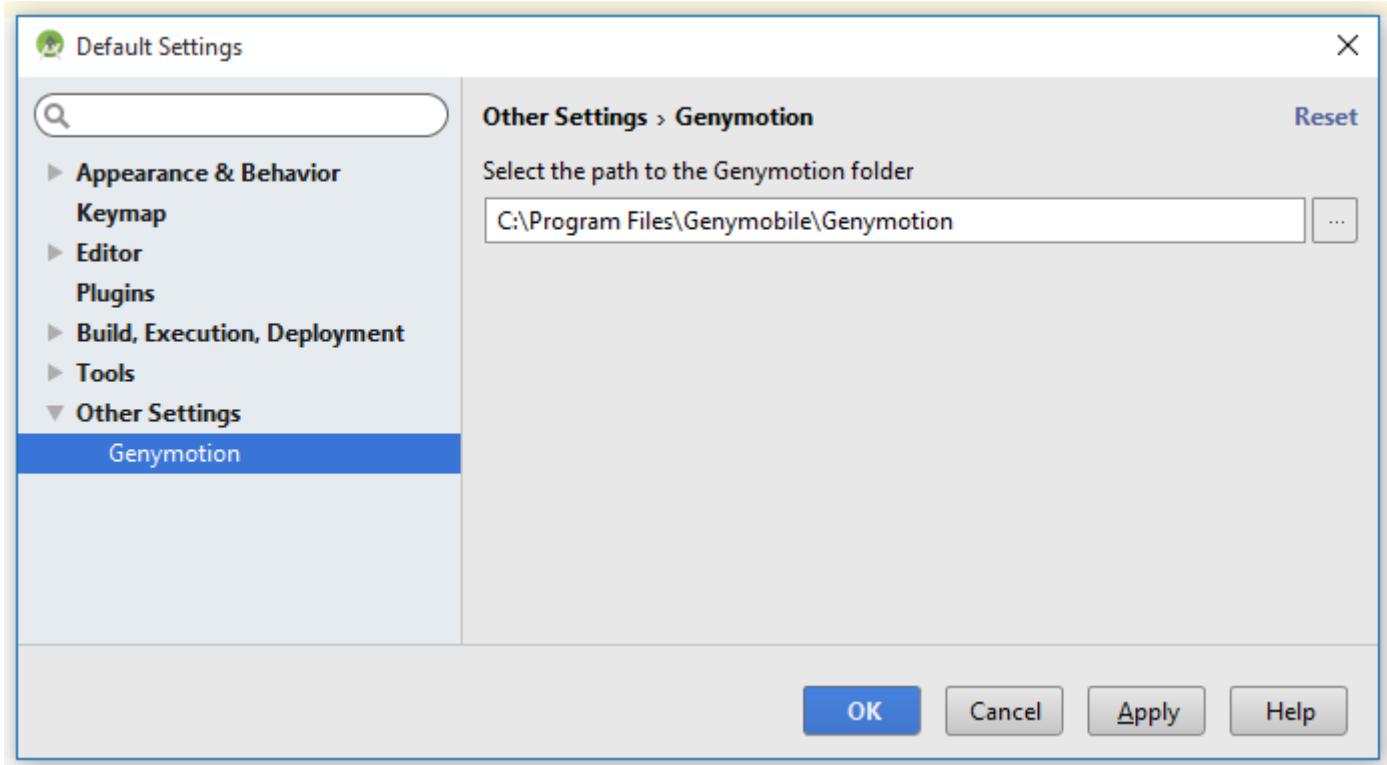
- Click vào icon của plugin Genymotion trên thanh toolbar, một cửa sổ giao diện sẽ xuất hiện như sau



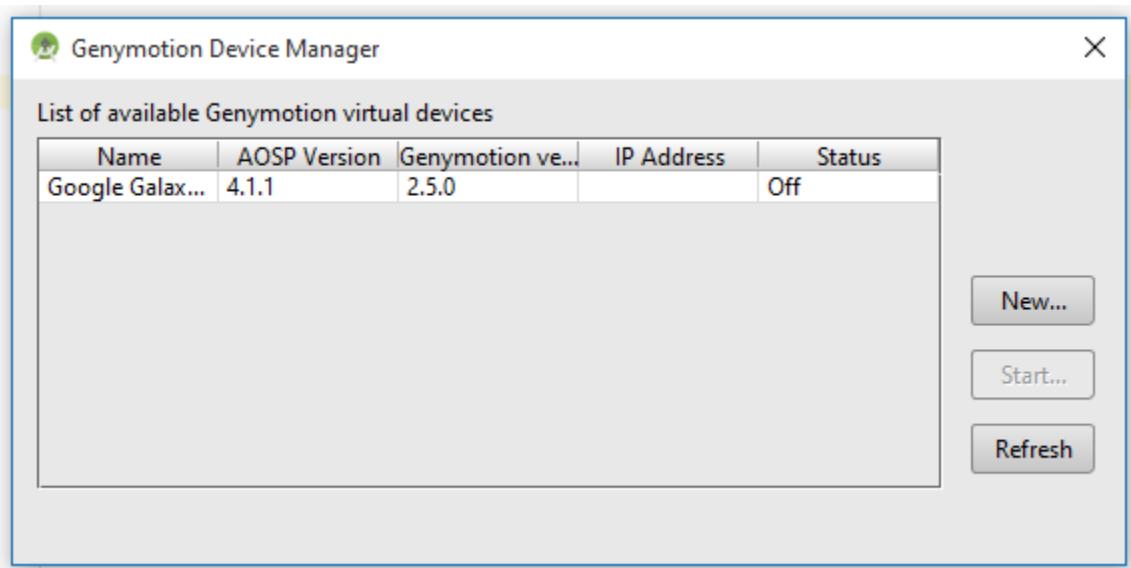
- Tiếp tục click vào dấu “...” để chọn folder chứa Genymotion đã cài đặt và **nhấn OK**



- Lúc này đường dẫn đến folder cài đặt Genymotion đã xuất hiện, chúng ta tiếp tục click **OK** để hoàn tất việc cấu hình plugin



- Bây giờ chúng ta sẽ chọn lại vào icon của plugin Genymotion trên thanh Toolbar, một cửa sổ mới xuất hiện chứng tỏ chúng ta đã cấu hình thành công Genymotion và có thể bắt đầu sử dụng nó để test các ứng dụng



Chương 02

Xây dựng bố cục giao diện với Android Layout

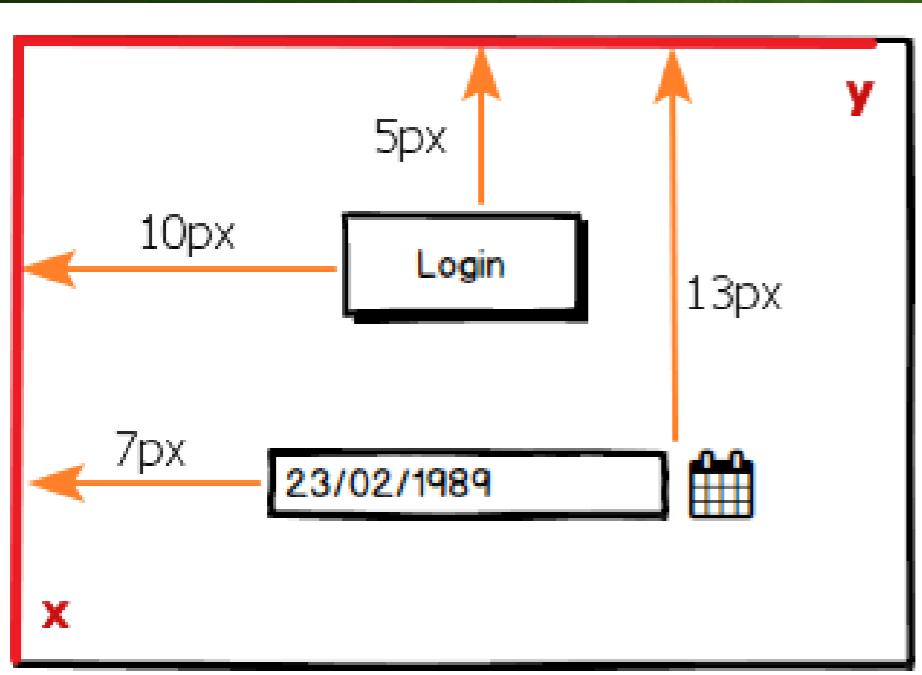
1. Sử dụng *AbsoluteLayout*
2. Sử dụng *FrameLayout*
3. Sử dụng *LinearLayout*
4. So sánh *wrap_content* và *match_parent*
5. So sánh *gravity* và *layout_gravity*
6. Sử dụng *weightSum* và *layout_weight*
7. Xây dựng giao diện với *LinearLayout*
8. So sánh *padding* và *margin*
9. Xây dựng giao diện với *RelativeLayout*
10. Sử dụng *TableLayout*
11. *Android Screen Orientation*

Android Layout

Layout định nghĩa cấu trúc và bố trí giao diện cho người dùng. Các layout phổ biến:

- *AbsoluteLayout* (không còn được sử dụng)
- *FrameLayout*
- *LinearLayout*
- *RelativeLayout*
- *TableLayout*

AbsoluteLayout



```
<AbsoluteLayout  
    xmlns:android="http://schemas.android.com/apk/res/android"  
    android:layout_width="match_parent"  
    android:layout_height="match_parent">  
    <Button  
        android:id="@+id/btnBack"  
        android:text="Back"  
        android:layout_x="10px"  
        android:layout_y="5px"  
        android:layout_width="wrap_content"  
        android:layout_height="wrap_content" />  
    </AbsoluteLayout>
```

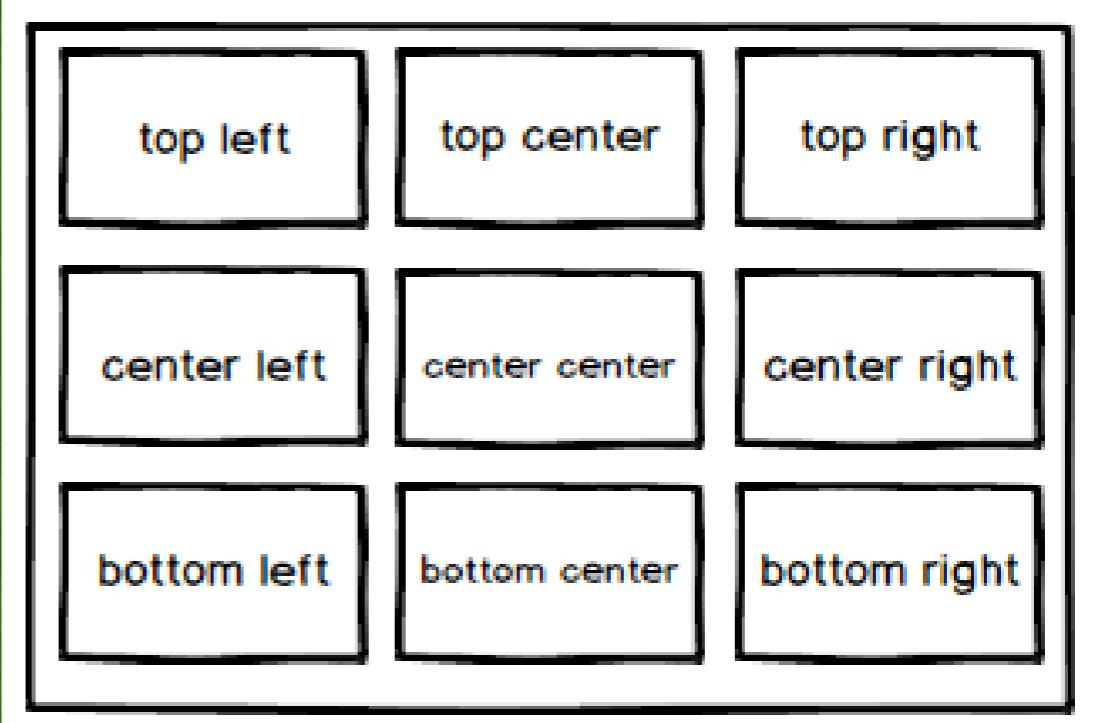
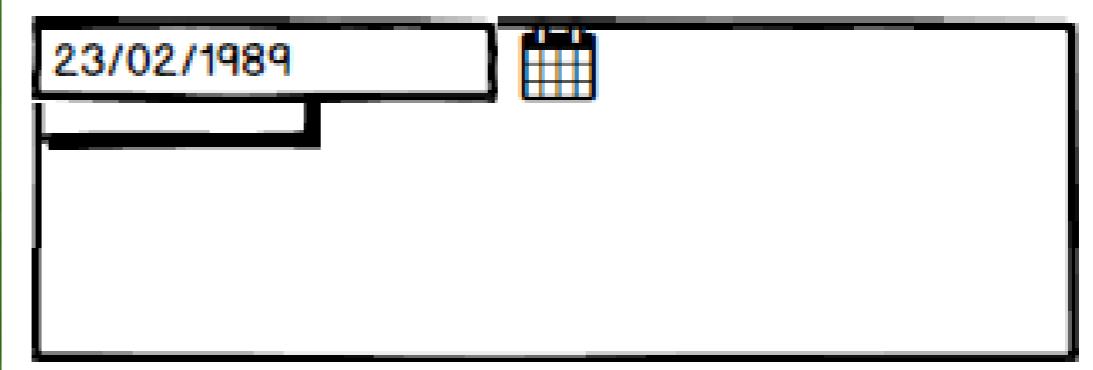
- Thiết lập các control theo vị trí tùy thích. Phải xác định rõ vị trí tọa độ x và y trên màn hình làm cho việc xây dựng giao diện rất “cứng” và không còn linh hoạt
- Đang dần dần bị “deprecated”

FrameLayout

- Phù hợp với giao diện có 1 control duy nhất, nếu có nhiều control các control sẽ nằm đè lên nhau tại góc trái màn hình

- Sử dụng thuộc tính `layout_gravity` để định lại vị trí các control

`android:layout_gravity="center"`

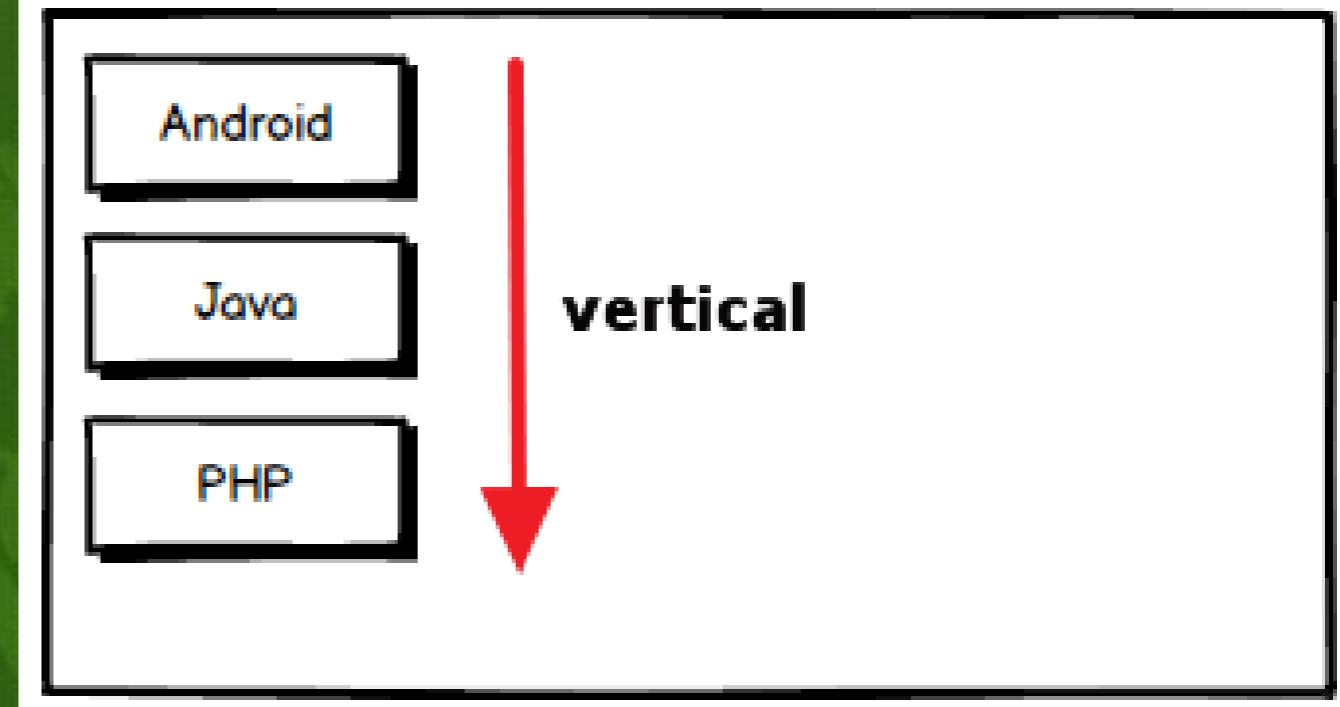
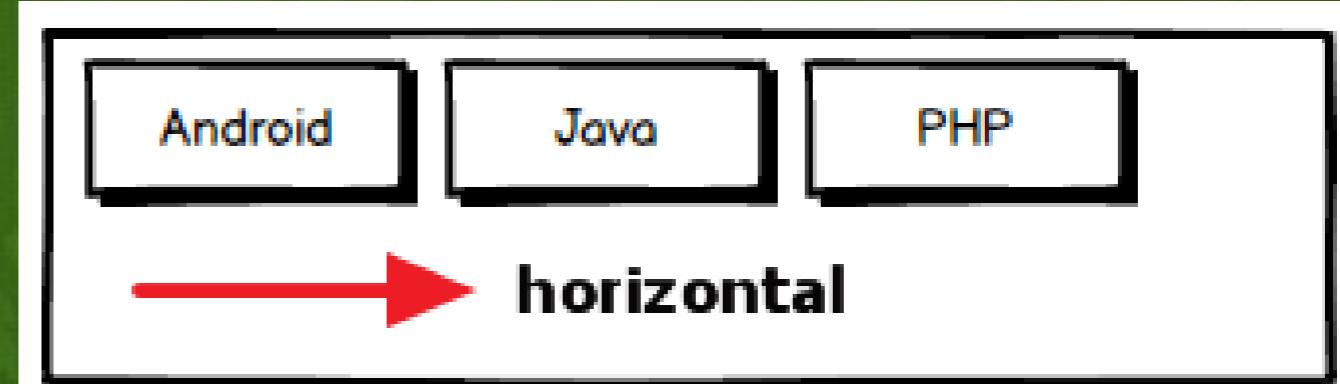


LinearLayout

Sắp xếp các control theo 2 hướng:

- *Horizontal*: Hướng từ trái qua phải (default)
- *Vertical*: Hướng từ trên xuống dưới

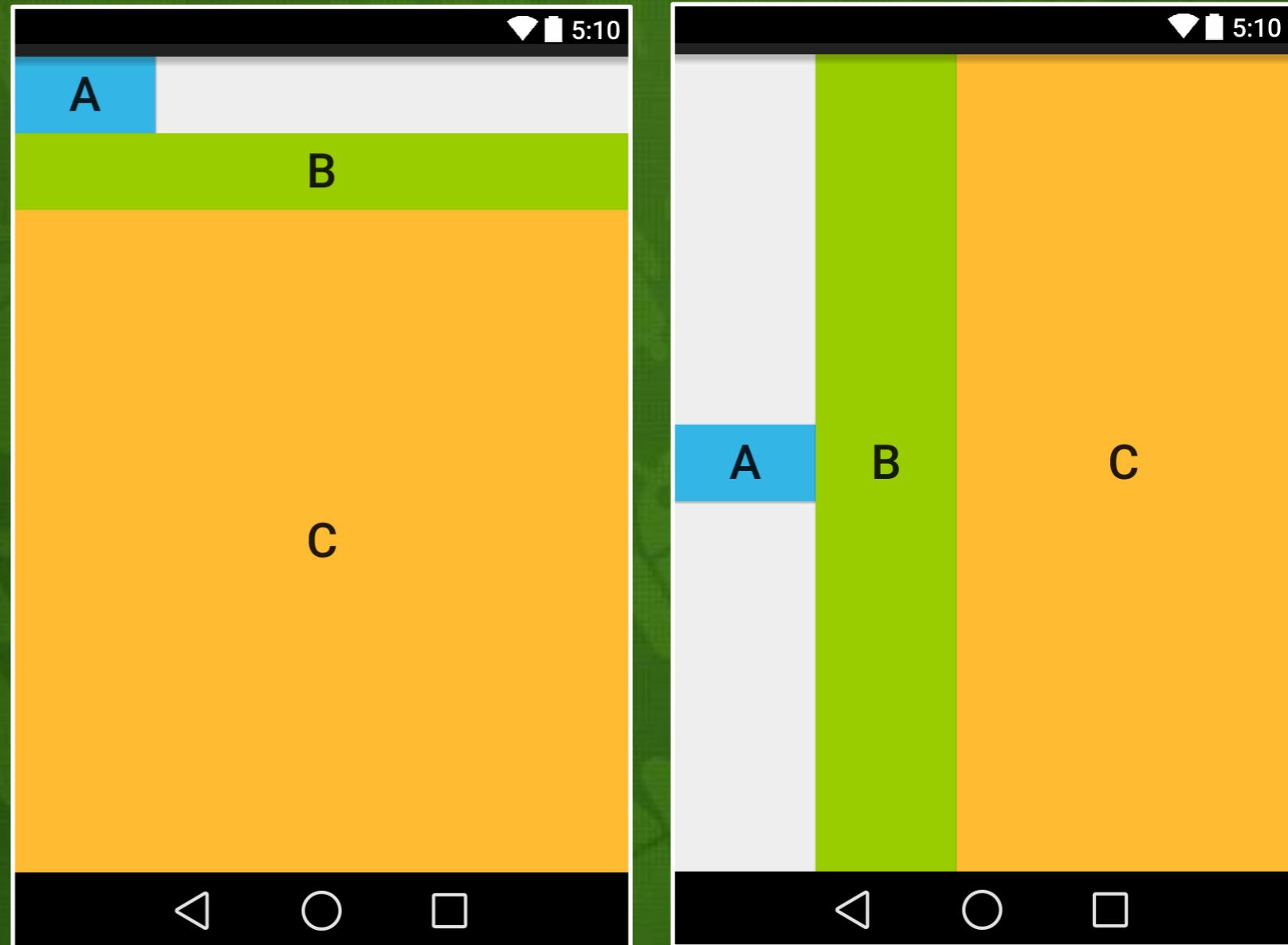
Trong một giao diện chúng ta có thể kết hợp nhiều loại layout với nhau



wrap_content vs match_parent

match_parent và fill_parent
thiết lập kích thước (width
và height) của một view
giống như view cha của nó

*wrap_content thiết lập kích
thước của một view đúng
như phần nội dung mà nó
chứa*



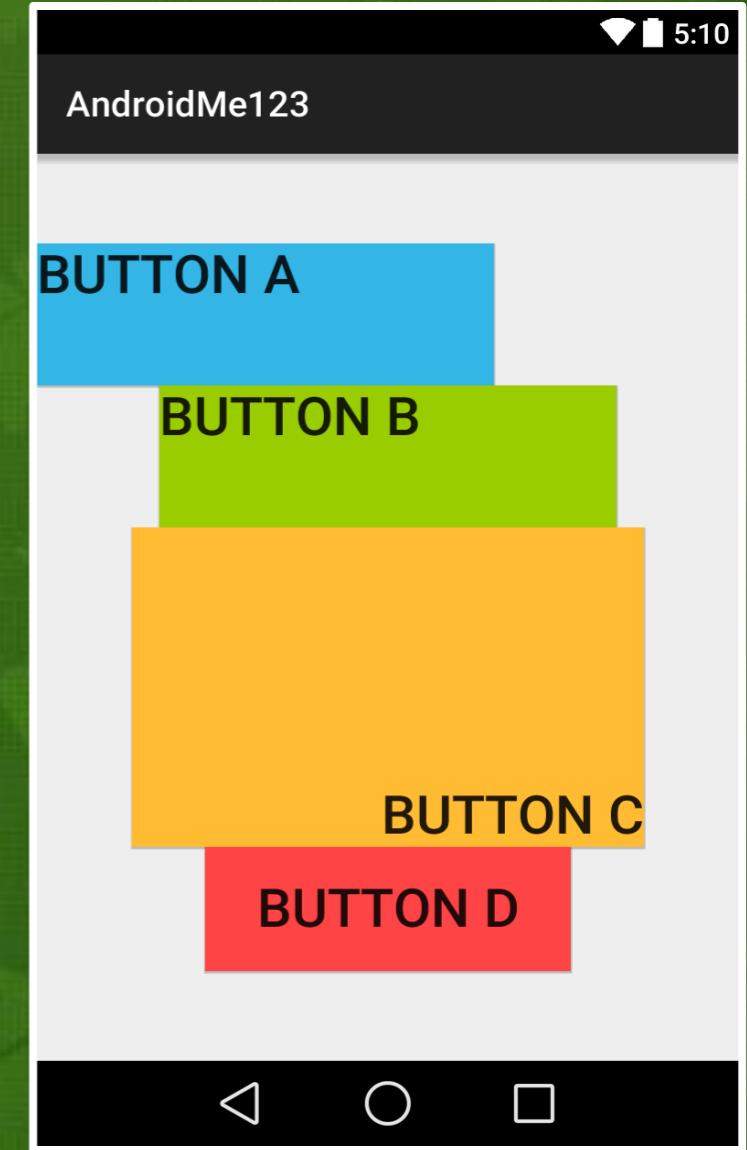
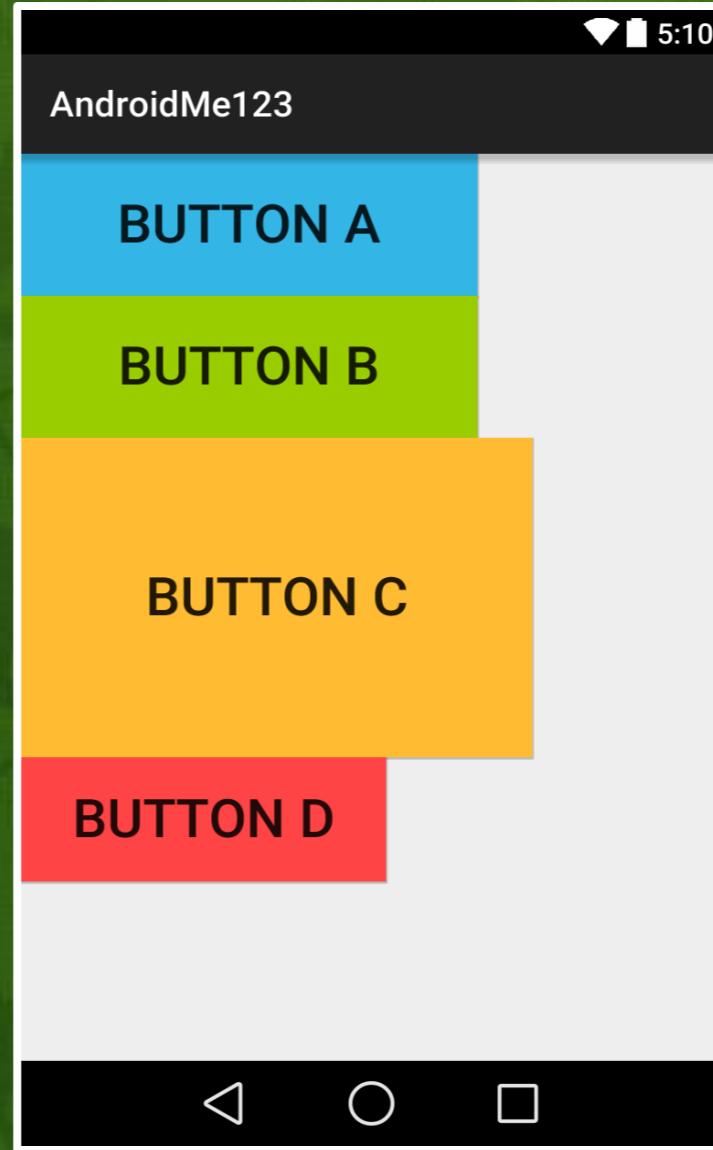
gravity vs layout_gravity

gravity

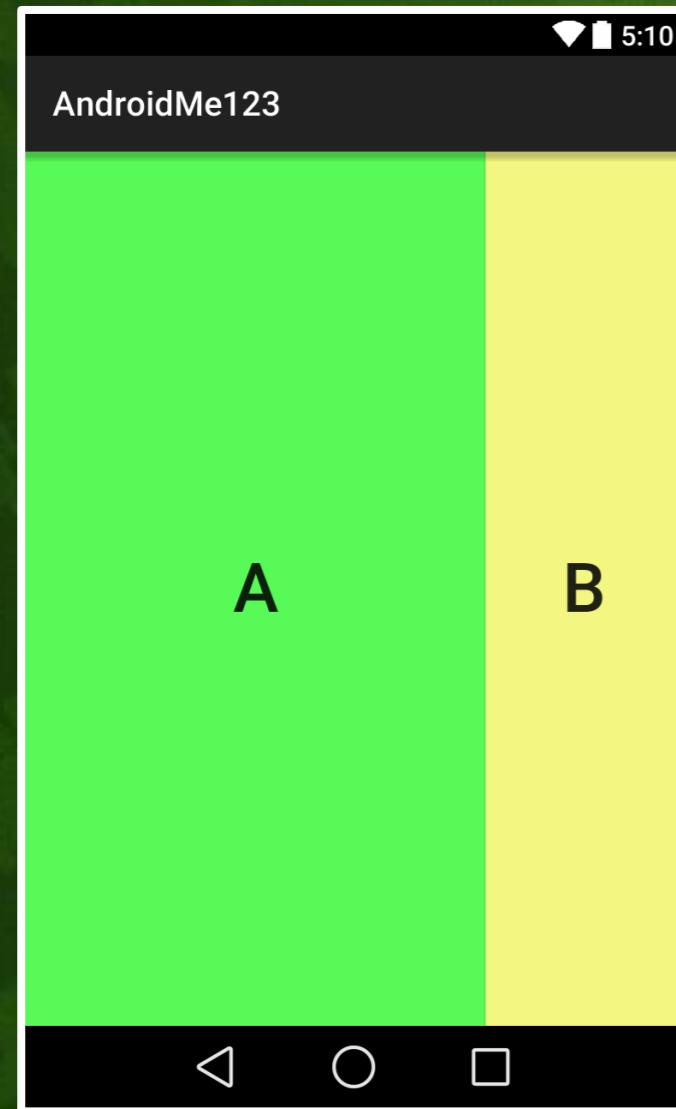
*Inside gravity of the View
(text-align in CSS)*

layout_gravity

*Outside gravity of the
View (float in CSS)*

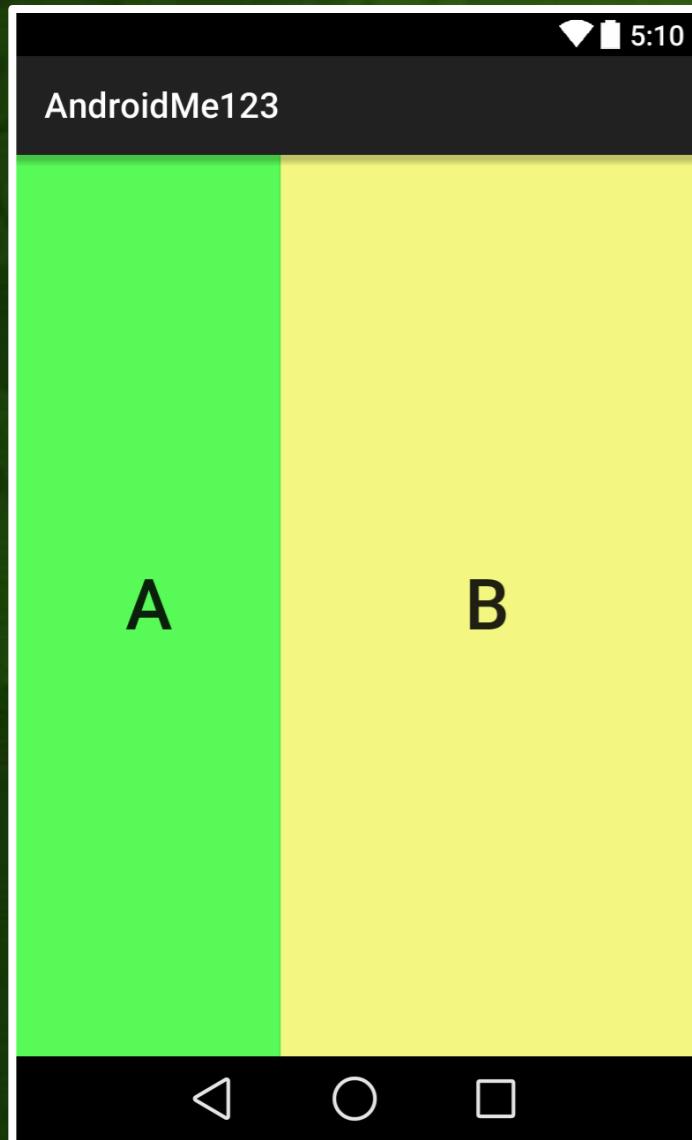


weightSum & layout_weight (1)



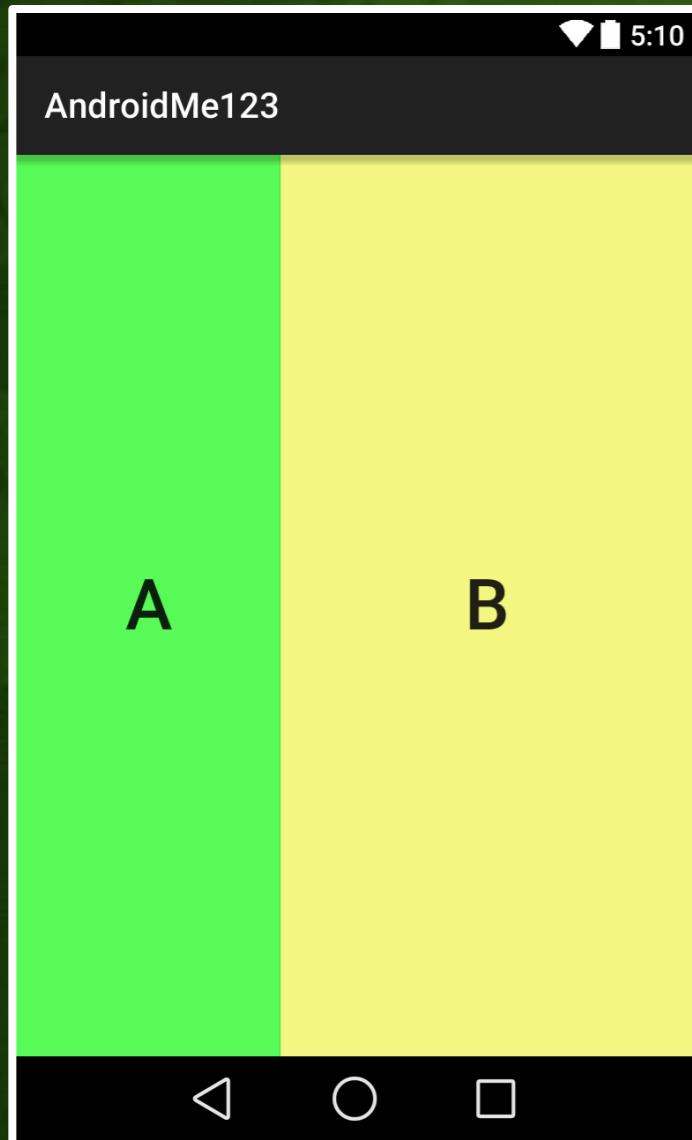
```
<LinearLayout  
    android:layout_width="match_parent"  
    android:layout_height="match_parent"  
    android:weightSum="100"  
    >  
    <Button  
        android:layout_weight="30"  
        android:layout_width="match_parent"  
        android:layout_height="match_parent"  
    />  
  
    <Button  
        android:layout_weight="70"  
        android:layout_width="match_parent"  
        android:layout_height="match_parent"  
    />  
</LinearLayout>
```

weightSum & layout_weight (2)



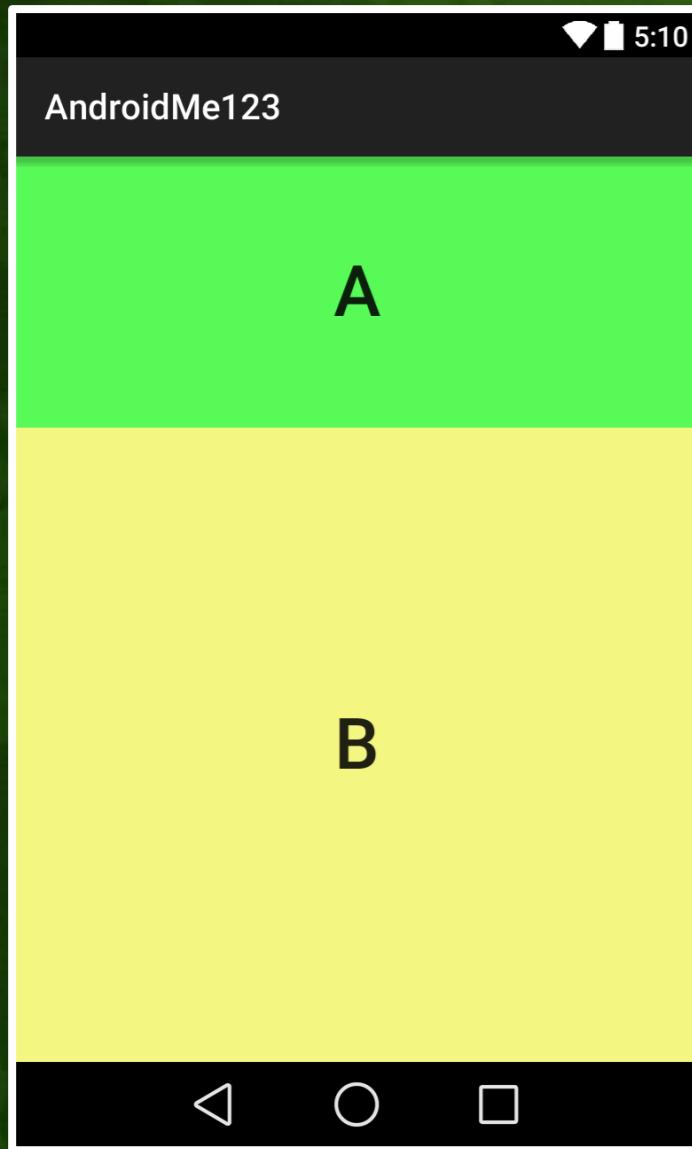
```
<LinearLayout  
    android:layout_width="match_parent"  
    android:layout_height="match_parent"  
    android:orientation="horizontal"  
    android:weightSum="100"  
    >  
    <Button  
        android:layout_weight="30"  
        android:layout_width="wrap_content"  
        android:layout_height="match_parent" />  
  
    <Button  
        android:layout_weight="70"  
        android:layout_width="wrap_content"  
        android:layout_height="match_parent"/>/  
    </LinearLayout>
```

weightSum & layout_weight (3)



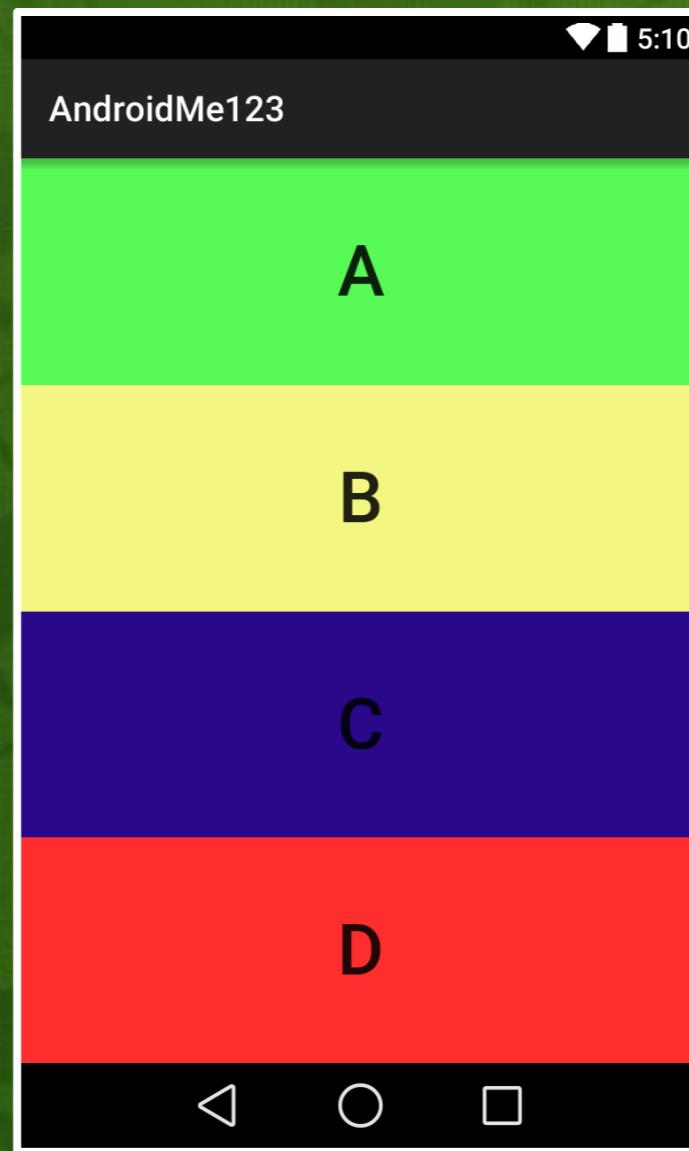
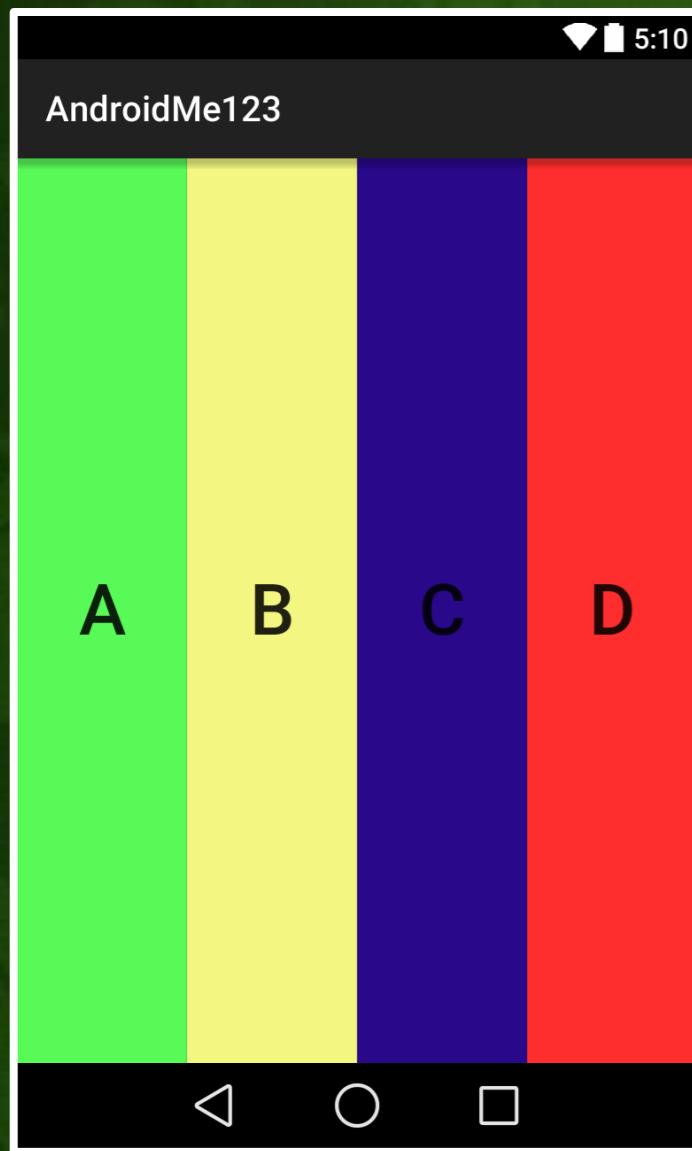
```
<LinearLayout  
    android:layout_width="match_parent"  
    android:layout_height="match_parent"  
    android:orientation="horizontal"  
    android:weightSum="100"  
    >  
    <Button  
        android:layout_weight="30"  
        android:layout_width="0dp"  
        android:layout_height="match_parent" />  
  
    <Button  
        android:layout_weight="70"  
        android:layout_width="0dp"  
        android:layout_height="match_parent" />  
</LinearLayout>
```

weightSum & layout_weight (4)



```
<LinearLayout  
    android:layout_width="match_parent"  
    android:layout_height="match_parent"  
    android:orientation="vertical"  
    android:weightSum="100" >  
    <Button  
        android:layout_weight="30"  
        android:layout_width="match_parent"  
        android:layout_height="0dp"/>  
    <Button  
        android:layout_weight="70"  
        android:layout_width="match_parent"  
        android:layout_height="0dp" />  
</LinearLayout>
```

Thực hành LinearLayout (1)



Thực hành LinearLayout (2)

LinearLayout
vertical
weightSum = 6



LinearLayout (H) = B + B

Button

LinearLayout (H) = B + B

LinearLayout (H)

- LinearLayout (V) = B + B
- LinearLayout (V) = B + B

Thực hành LinearLayout (3)

LinearLayout
vertical
weightSum = 6



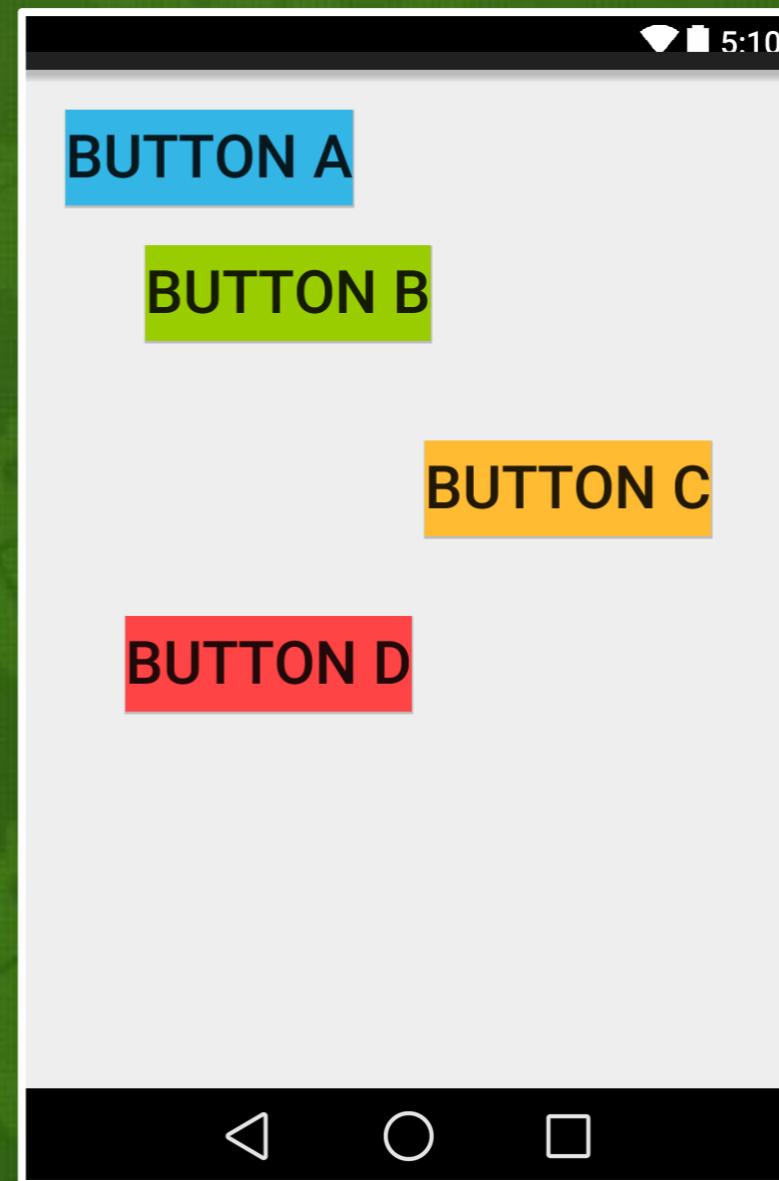
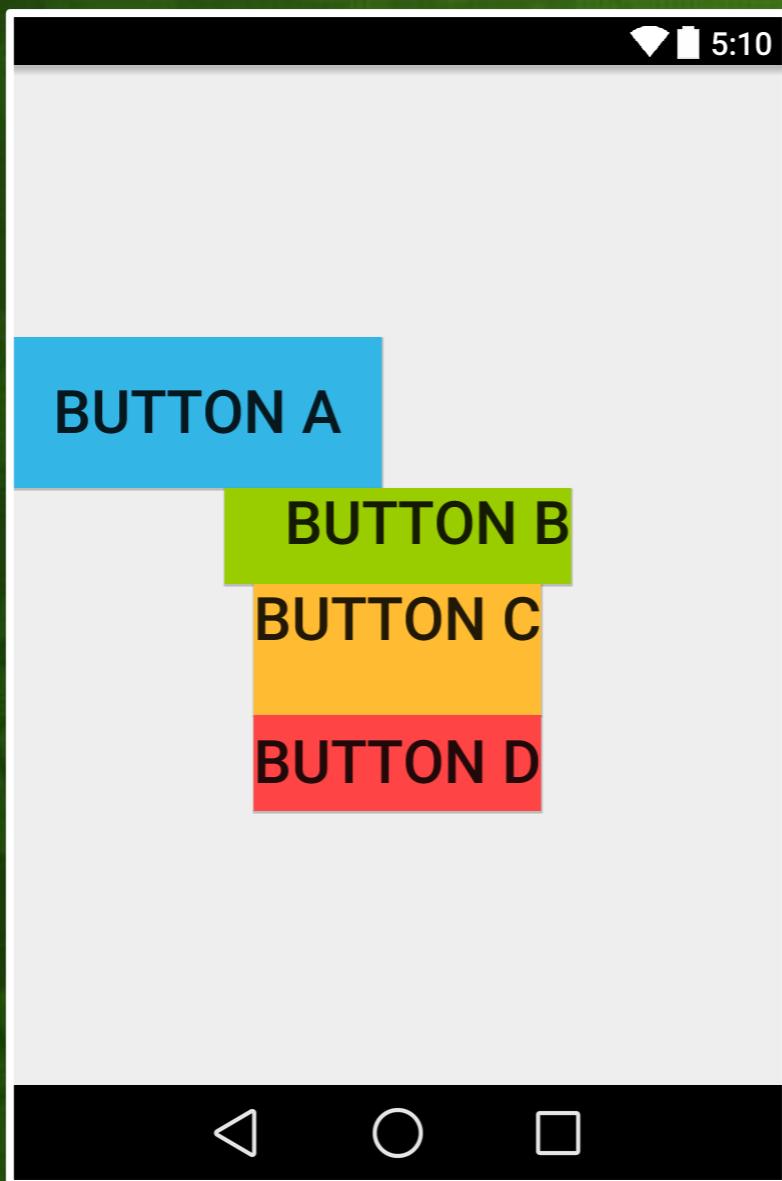
$$\text{LinearLayout (H)} = B + B$$

Button

$$\text{LinearLayout (H)}$$

- $\text{LinearLayout (V)} = B + B + B$
- $\text{LinearLayout (V)} = B + B + B$

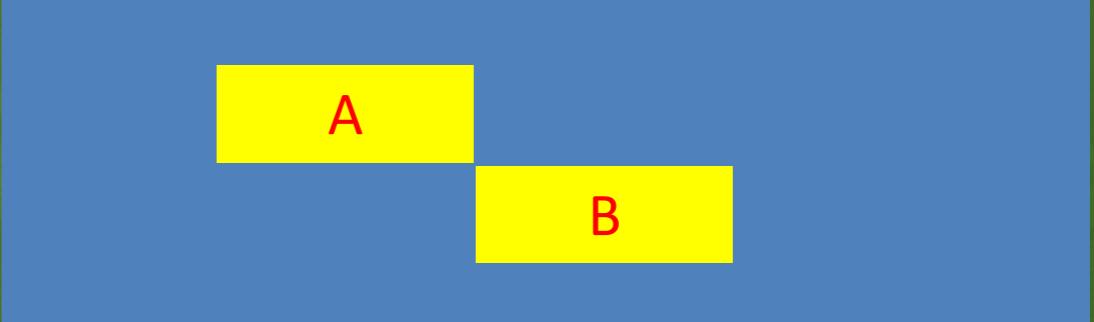
Margin vs Padding



RelativeLayout

Sắp xếp các View theo những vị trí tương đối với nhau. Vị trí mỗi View được sắp xếp dựa vào 2 yếu tố:

- Vị trí so với anh em của nó
- Vị trí so với thành phần chứa nó

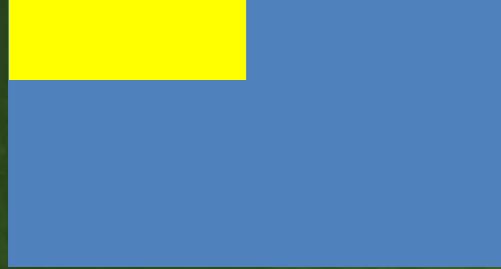


A định vị trí so với B (anh em)

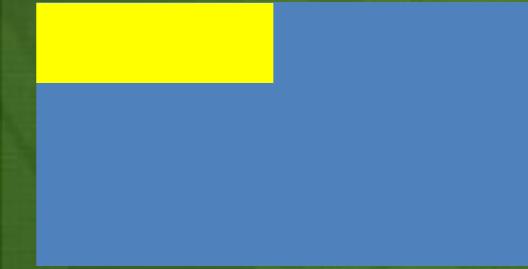


A định vị trí so với khối chứa nó

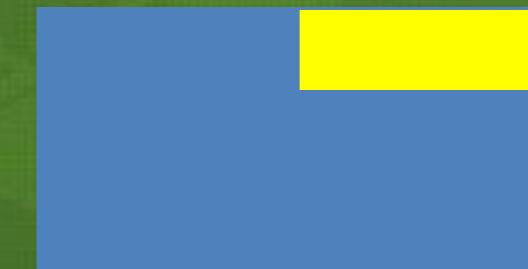
RelativeLayout - Vị trí trong mối quan hệ cha con



layout_alignParentLeft



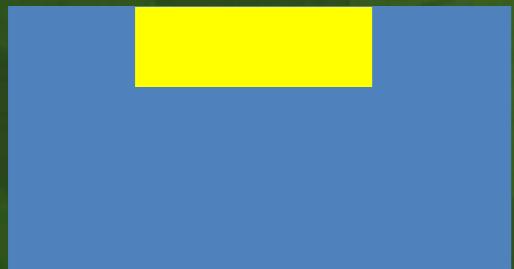
layout_alignParentTop



layout_alignParentRight



layout_alignParentBottom



layout_centerHorizontal



layout_centerVertical

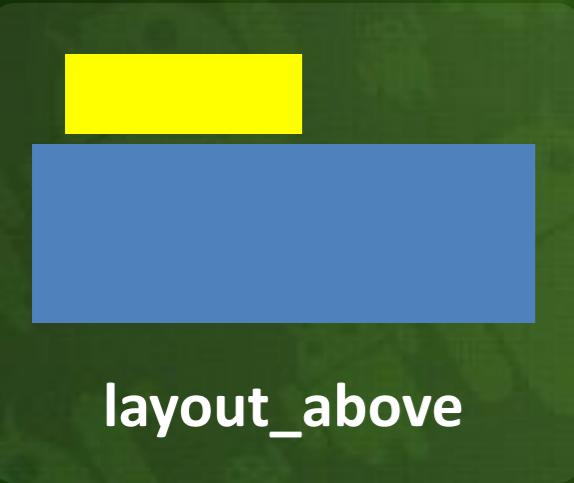


layout_centerInParent

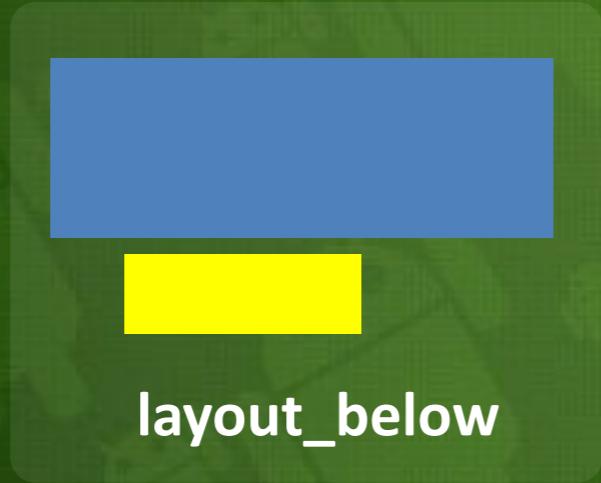


layout_alignBaseline

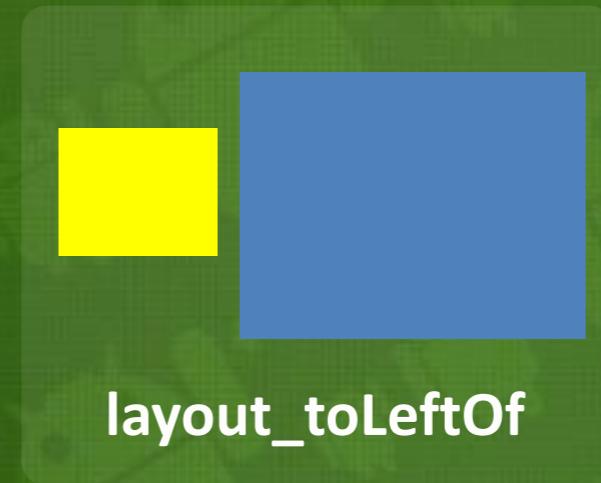
RelativeLayout – Vị trí trong mối quan hệ anh em



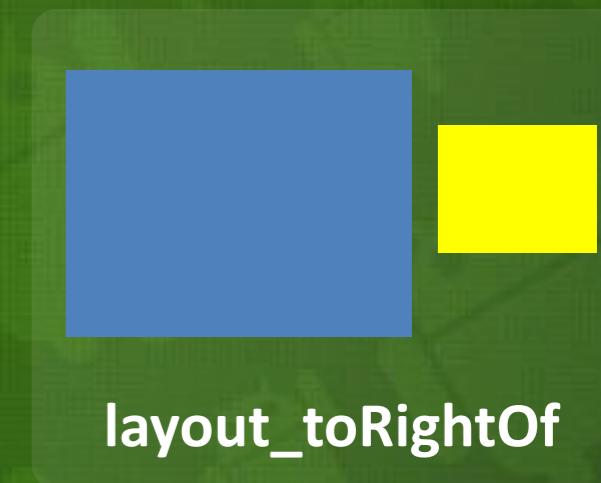
layout_above



layout_below



layout_toLeftOf



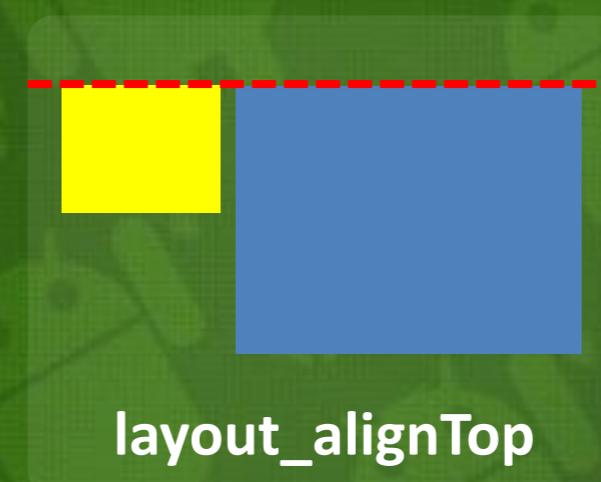
layout_toRightOf



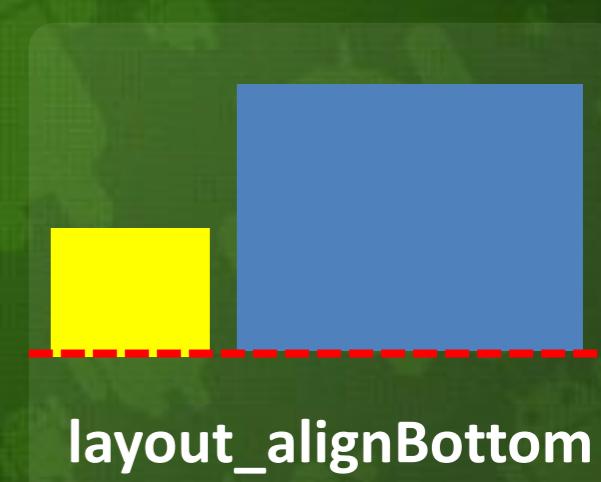
layout_alignLeft



layout_alignRight

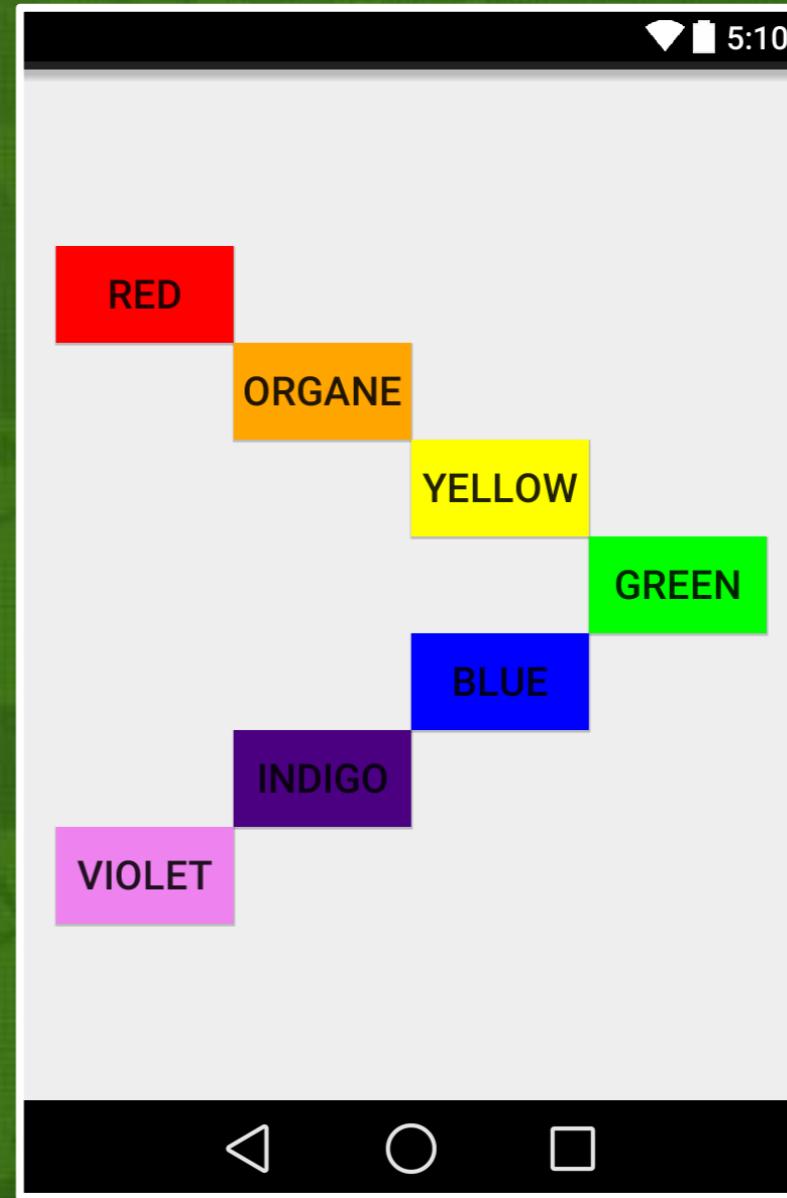
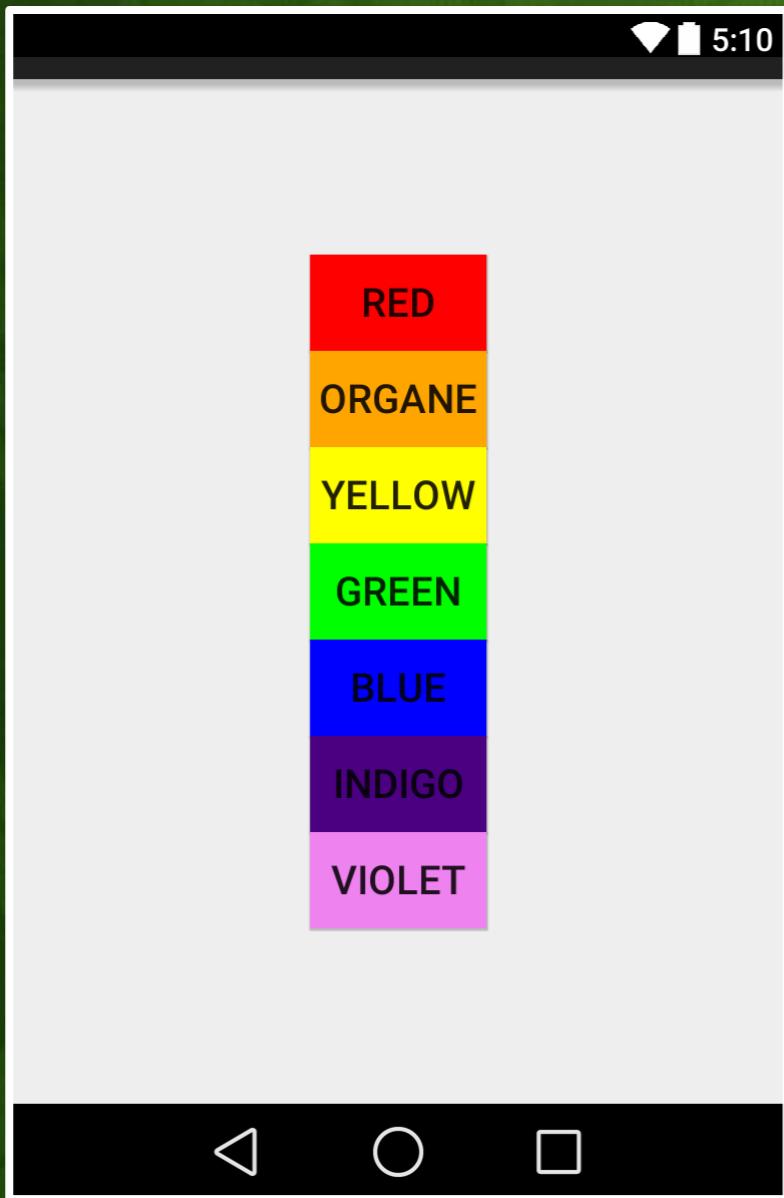


layout_alignTop

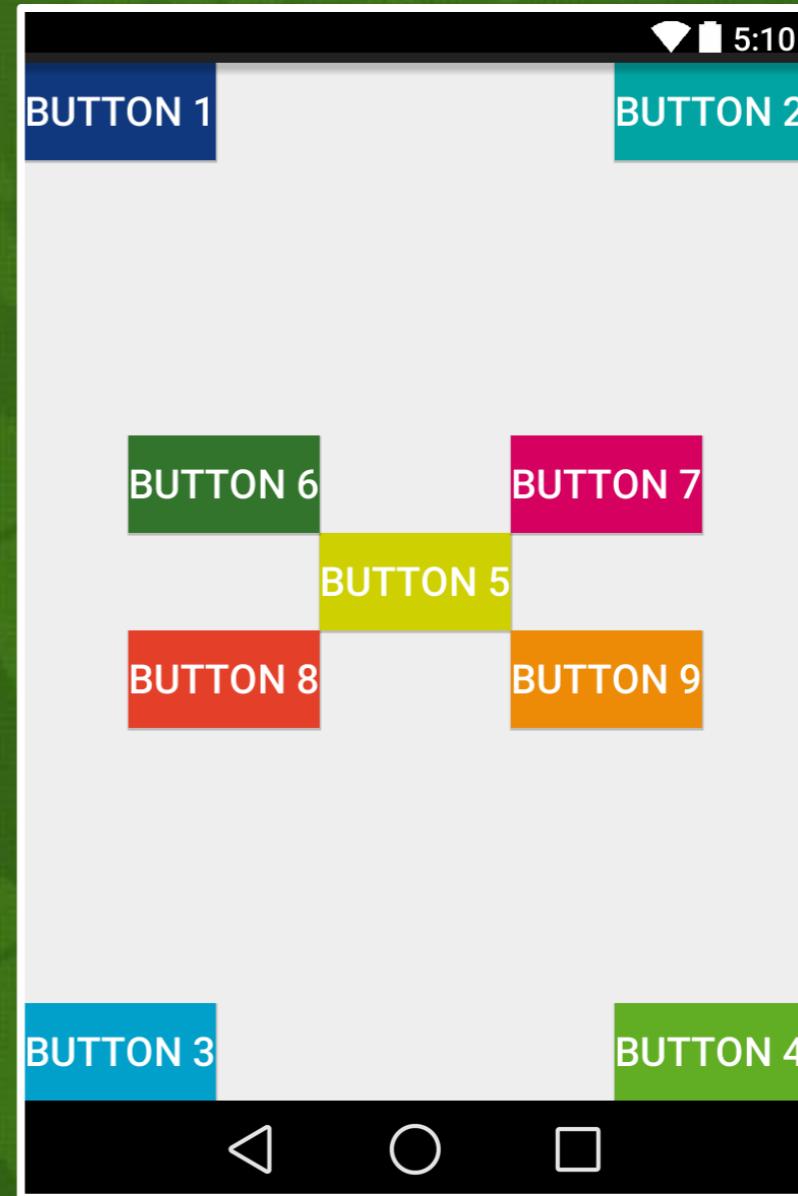
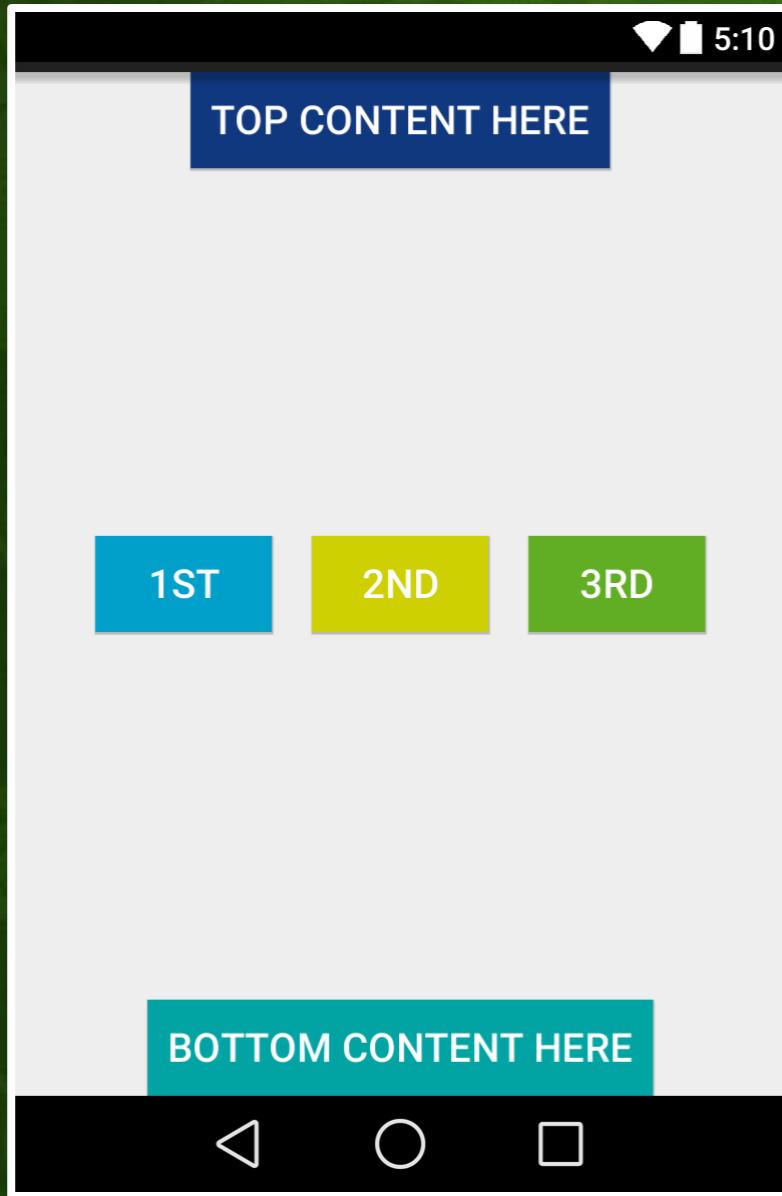


layout_alignBottom

RelativeLayout - Exercise (1)



RelativeLayout - Exercise (2)



TableLayout

- *Sắp xếp các view theo hàng (row) và cột (column)*
- *Sử dụng <TableRow> để định nghĩa một hàng nào đó*
- *Số View nhiều nhất trong TableRow dùng để định nghĩa số cột cho toàn table*
- *Xây dựng ví dụ và quan sát các thuộc tính width, height của TableRow và View*

R1 - C1

R2 - C1

R1 - C2 R1 - C3

R2 - C2

TableLayout - stretchColumns

```
<TableLayout  
    android:stretchColumns="1">
```

R1 - C1	R1 - C2	R1 - C3
R2 - C1	R2 - C2	

```
<TableLayout  
    android:stretchColumns="0, 2">
```

R1 - C1	R1 - C2	R1 - C3
R2 - C1	R2 - C2	

```
<TableLayout  
    android:stretchColumns = "*">
```

R1 - C1	R1 - C2	R1 - C3
R2 - C1	R2 - C2	

TableLayout - layout_span & layout_column

R1 - C1	R1 - C2	R1 - C3
R2 - C1	Row 2 - Column 2	

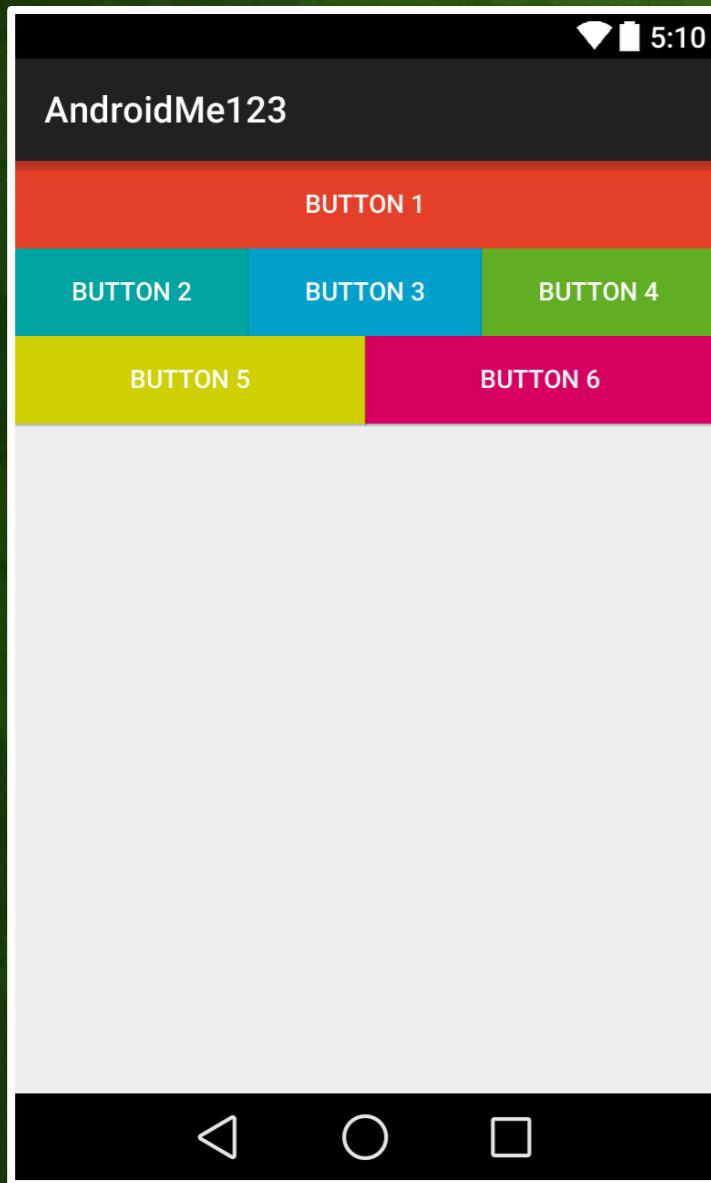
android:layout_column="2">

R1 - C1	R1 - C2	R1 - C3
R2 - C1		Row 2 - Column 2

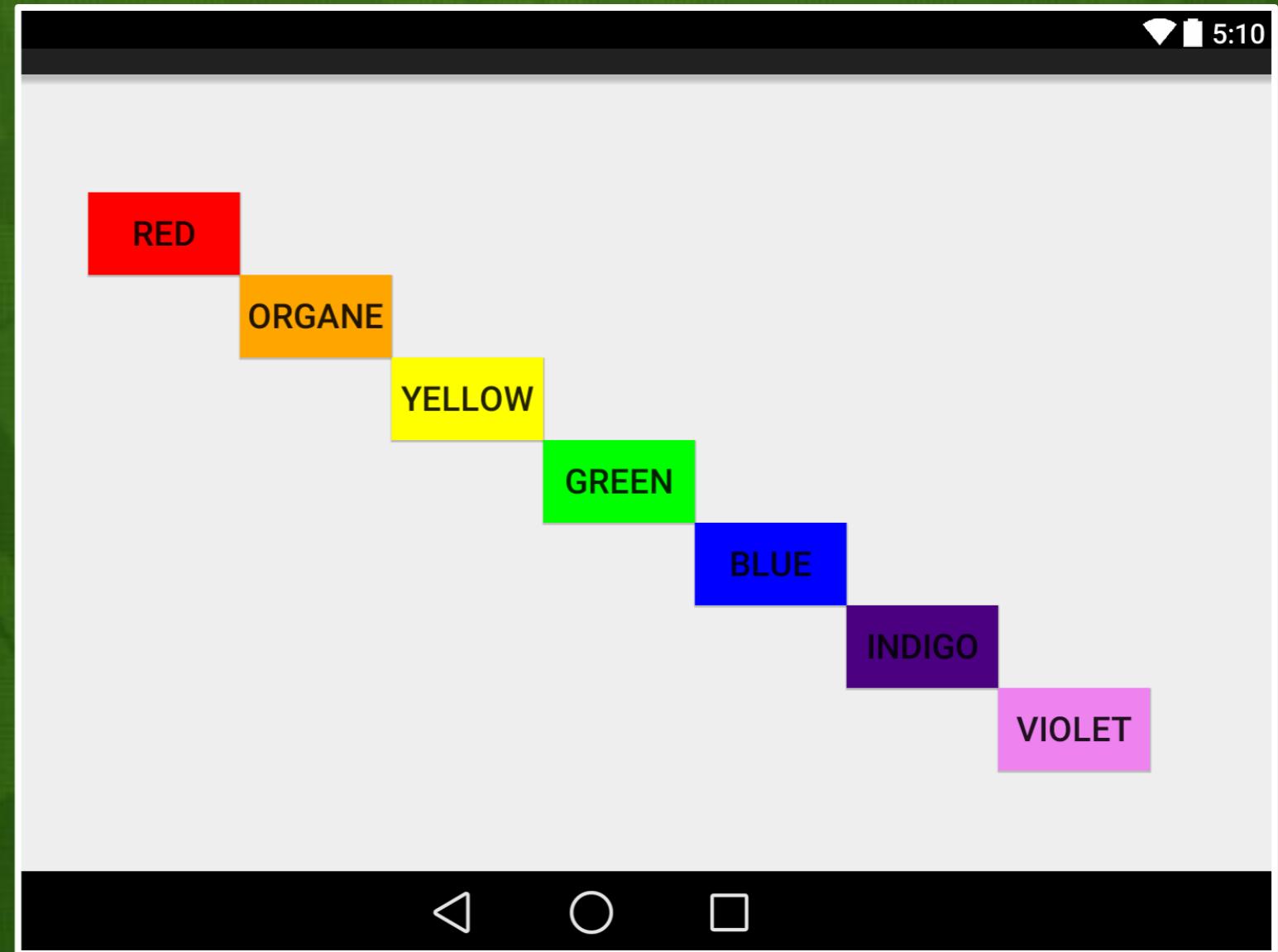
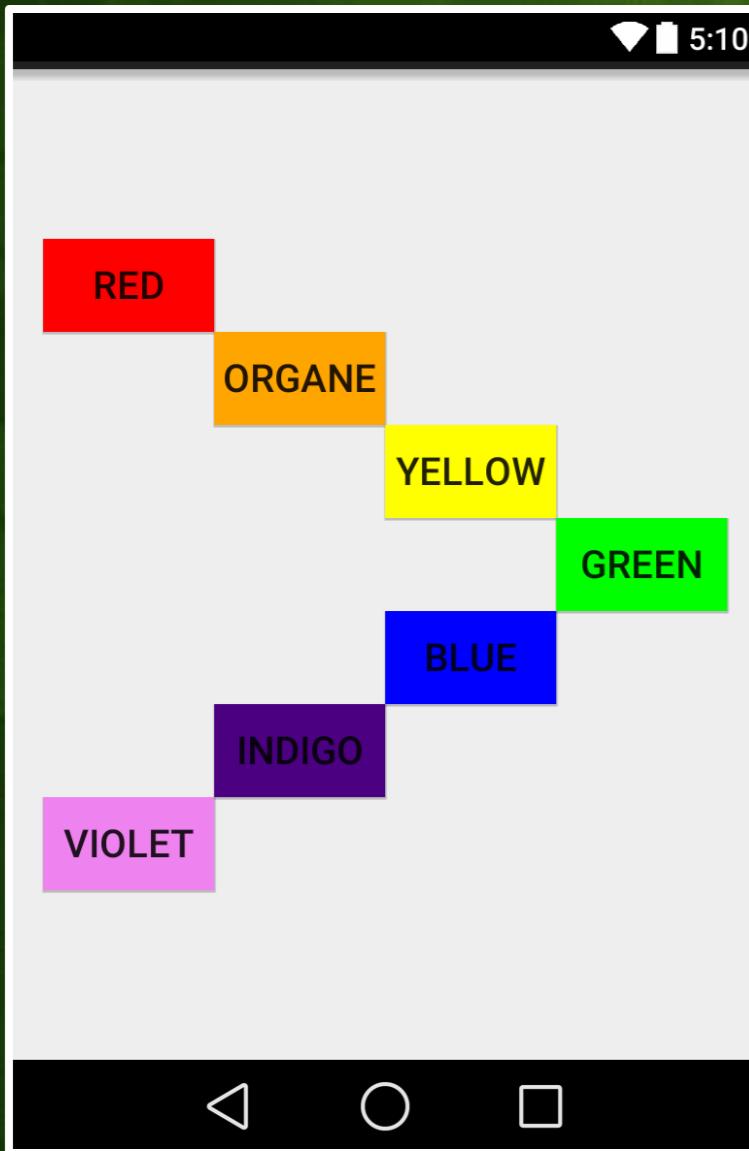
android:layout_span="2">

R1 - C1	R1 - C2	R1 - C3
R2 - C1	Row 2 - Column 2	

TableLayout - Exercise



Android Screen Orientation



Chương 3: Android Form Widgets

1. Sử dụng *TextView*, *EditText*, *Button* và *ToggleButton*
2. Sử dụng *Android Resource cơ bản*
3. *Event Listener*
4. Xây dựng app “*Lời hay ý đẹp*”
5. *Android's virtual keyboard*
6. Xây dựng app “*Máy tính điện tử*”
7. Xây dựng “*Chiếc cân vui vẻ*”
8. Xây dựng app “*Tìm số bí mật*”
9. *ImageView* và các thao tác xử lý
10. Xây dựng game “*Card*”
11. Xây dựng app “*Random Number*” với *Checkbox* và *RadioButton*

Chương 3: Android Form Widgets

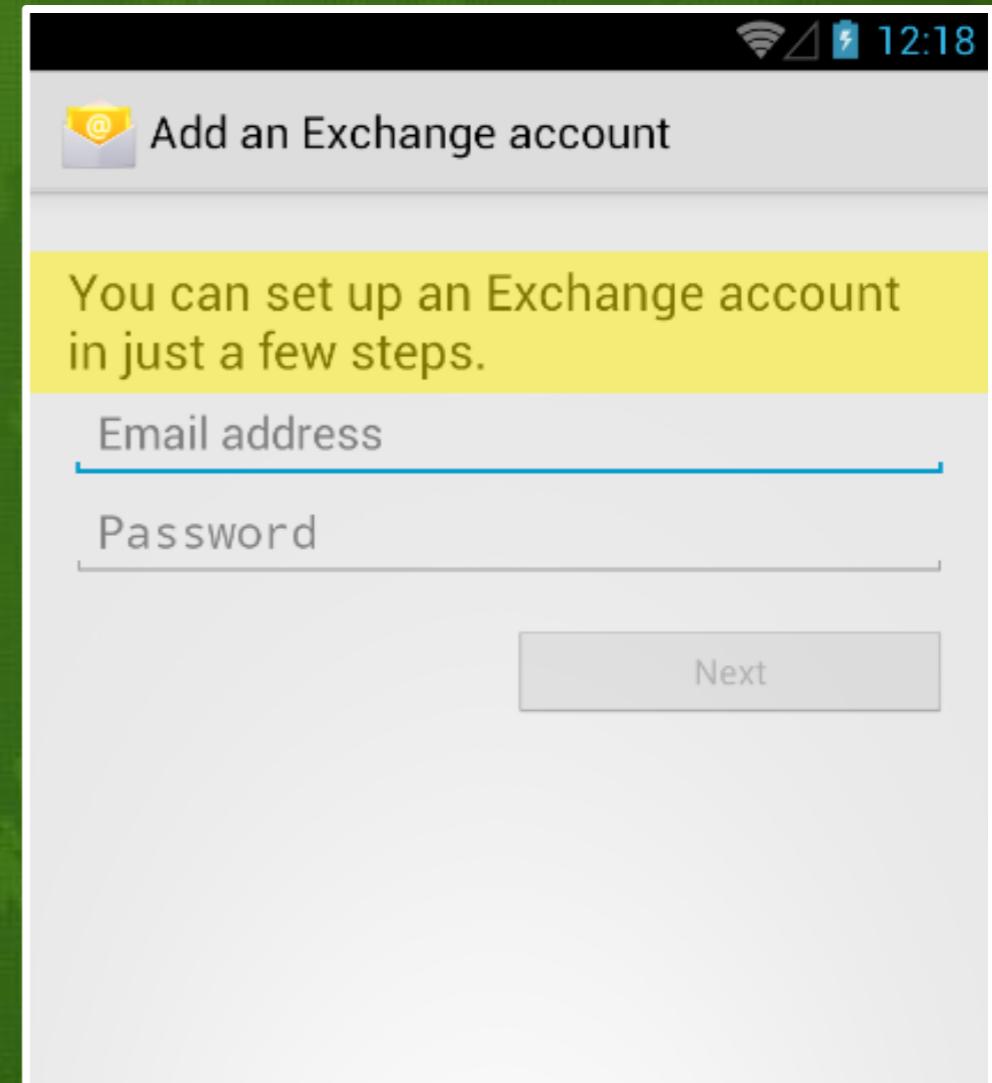
1. Sử dụng *TextView*, *EditText*, *Button* và *ToggleButton*
2. Sử dụng *Android Resource* cơ bản
3. *Event Listener*
4. Xây dựng app “*Lời hay ý đẹp*”
5. *Android's virtual keyboard*
6. Xây dựng app “*Máy tính điện tử*”
7. Xây dựng “*Chiếc cân vui vẻ*”
8. Xây dựng app “*Tìm số bí mật*”
9. *ImageView* và các thao tác xử lý
10. Xây dựng game “*Card*”
11. Xây dựng app “*Random Number*” với *Checkbox* và *RadioButton*

TextView

Chức năng: Hiển thị nội dung văn bản để miêu tả một vấn đề nào đó cho người sử dụng

Các thuộc tính thông dụng: *id, text, textAllCaps, textColor, textSize, textStyle, drawablePadding, drawableLeft, drawableRight*

Thiết lập thuộc tính: trong XML và Java code

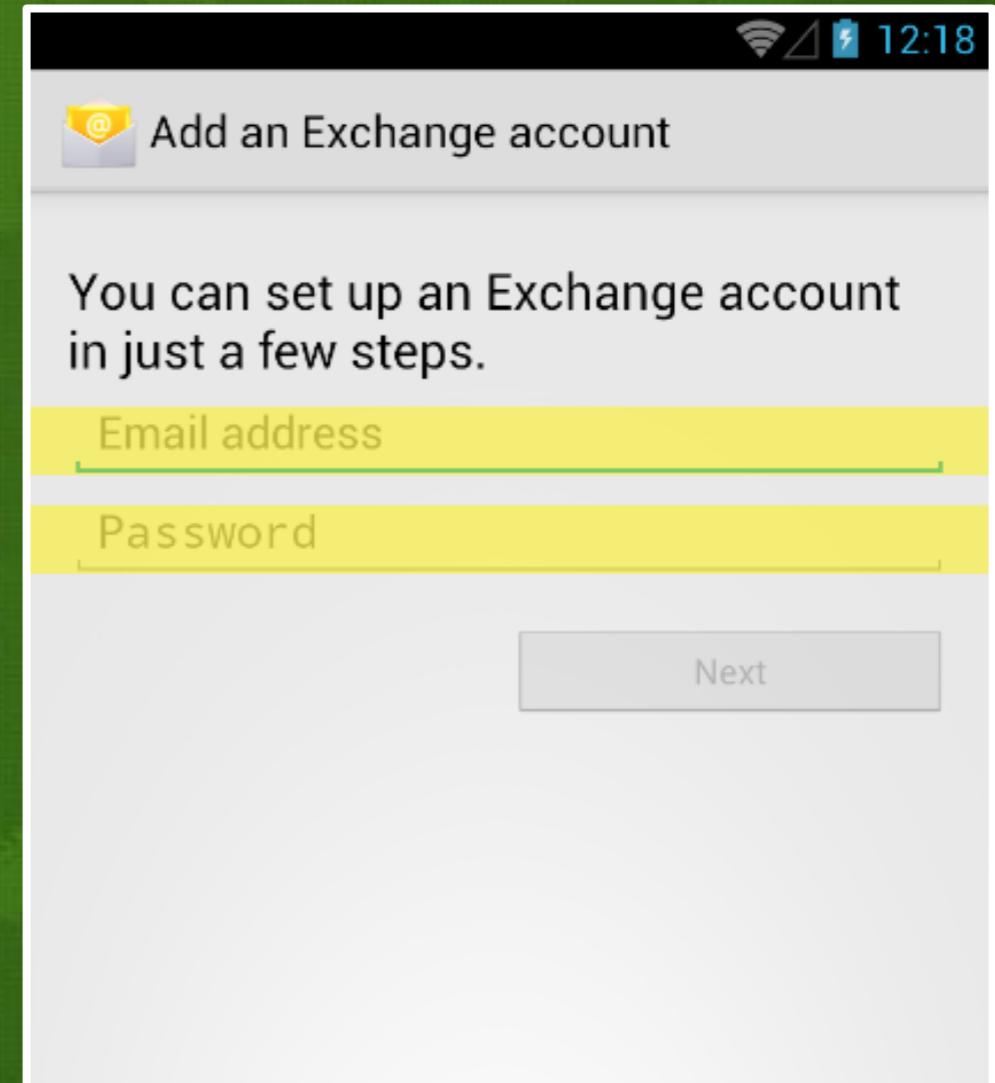


EditText

Chức năng: nhận giá trị nhập vào từ phía người sử dụng (kế thừa từ TextView)

Các thuộc tính thông dụng: id, text, ... , inputType, hint, textColorHint, editable, singleLine, password

Thiết lập thuộc tính: trong XML và Java code

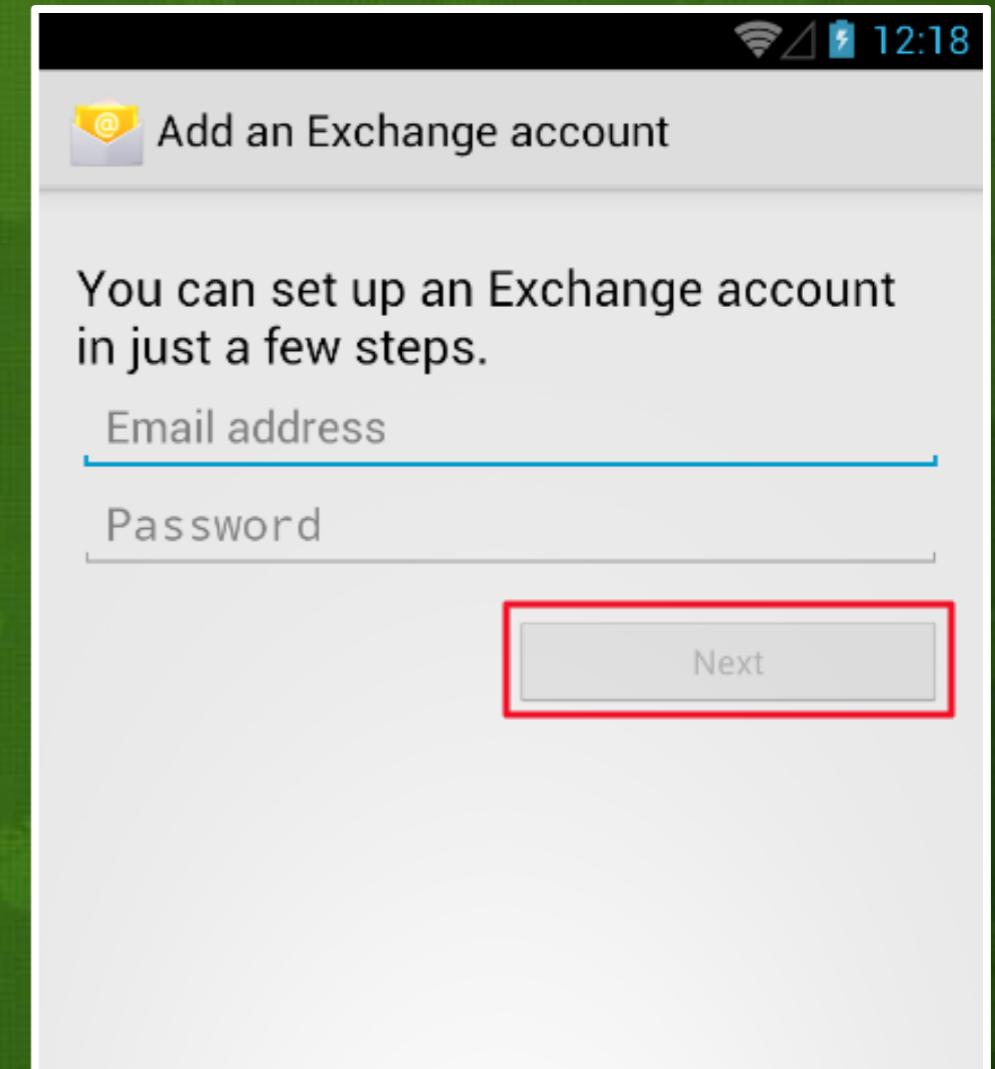


Button

Chức năng: thiết lập sự kiện khi người sử dụng chọn lựa một hành động nào đó

Các thuộc tính thông dụng: id, text, textAllCaps, textColor, textSize, textStyle, onClick, clickable

Thiết lập thuộc tính: trong XML và Java code



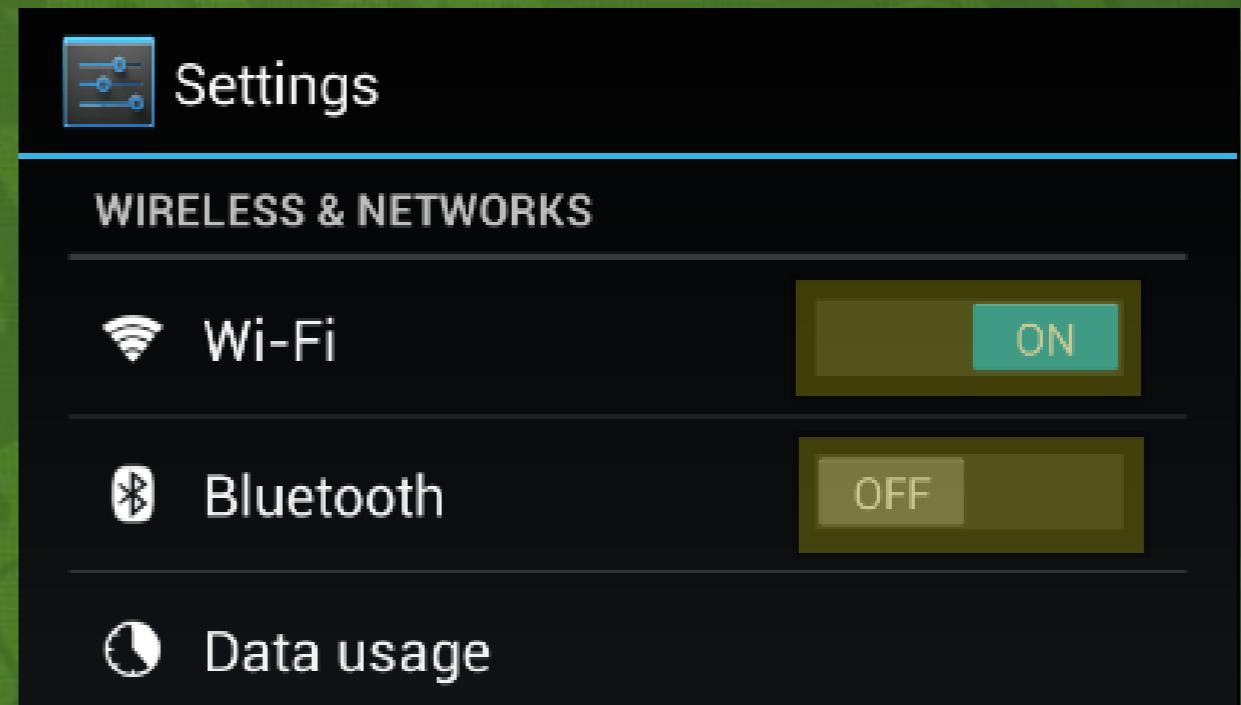
ToggleButton

Chức năng: thiết lập sự kiện khi người sử dụng chọn lựa một hành động nào đó

Các thuộc tính thông dụng: checked, textOff, textOn

Tìm hiểu:

- *setOnCheckedChangeListener*
- *isChecked*



Chương 3: Android Form Widgets

1. Sử dụng *TextView*, *EditText*, *Button* và *ToggleButton*
2. Sử dụng *Android Resource cơ bản*
3. *Event Listener*
4. Xây dựng app “*Lời hay ý đẹp*”
5. *Android's virtual keyboard*
6. Xây dựng app “*Máy tính điện tử*”
7. Xây dựng “*Chiếc cân vui vẻ*”
8. Xây dựng app “*Tìm số bí mật*”
9. *ImageView* và các thao tác xử lý
10. Xây dựng game “*Card*”
11. Xây dựng app “*Random Number*” với *Checkbox* và *RadioButton*

Sử dụng Android Resource cơ bản

Resource cung cấp các tài nguyên hình ảnh, audio, video, giao diện, string, ... góp phần tạo nên một ứng dụng hoàn chỉnh. Tài nguyên được lưu trữ trong **folder res** của ứng dụng. Các loại tài nguyên thường dùng:

- /anim – /animator *chứa tài nguyên Animation*
 - /drawable *chứa tài nguyên hình ảnh*
 - /layout *chứa tài nguyên về giao diện, layout*
 - /values *chứa tài nguyên về string, color, style, dimension*
 - /menu *chứa tài nguyên về menu, context menu, ...*
 - /mipmap *chứa tài nguyên về icon của ứng dụng*
 - ...

Các dạng Android Resource cơ bản

- *Tài nguyên mặc định*: không quan tâm đến cấu hình của thiết bị hoặc không có tài nguyên để lựa chọn
- *Tài nguyên đặc trưng*: được sử dụng trên thiết bị riêng biệt thông qua các từ hạn định và đường dẫn
- *res/layout* folder chứa tài nguyên mặc định gồm các layout của ứng dụng
- *res/layout-land* folder chứa tài nguyên đặc trưng khi app đang được xem với màn hình nằm ngang
- *res/values-vi* folder chứa tài nguyên string dành riêng cho tiếng Việt

Từ hạn định cho cấu hình

- MCC and MNC (mobile country code – mobile network code) (*mcc452, mcc452-mnc04*)
- Language and region (*en, fr, vi*)
- Screen size (*small < 3.2 inch, normal 3.2 -> 3.7 inch, large 3.7 -> 7 inch, xlarge 7 -> 11 inch*)
- Screen aspect (*long 16:9, notlong 4:3*)
- Screen Orientation (*port đứng, land ngang*)
- Dock mode (*desk để bàn, car trên xe*)
- Night mode (*night 18h -> 6h, notnight*)
- Screen pixel density (*ldpi 240x320, mdpi 480x640, hdpi 480x800, xhdpi 540x960 or 720x1280, nodpi, Xxhdpi*)
- Touchscreen type (*notouch không cảm ứng, stylus cảm ứng bút, finger cảm ứng bằng tay*)
- Keyboard availability (*keysexposed, keyshidden, keyssoft*)
- Primary text input method (*nokeys cảm ứng, qwerty bàn phím cảm ứng, 12key bàn phím số*)
- Navigation key availability (*navexposed phím điều hướng trên màn hình, navhidden phím điều hướng ẩn*)
- Primary non-touch navigation method (*nonav, dpad, trackball, wheel*)
- Platform version (API level) (*v3, v4, v7*)

Quy tắc đặt tên thư mục resource với từ hạn định

- Có thể dùng nhiều từ hạn định trong thư mục lựa chọn, ngăn cách bởi dấu “-”
→ *drawable-en-land*
- Tuân theo quyền ưu tiên từ trên xuống dưới.
→ *drawable-hdpi-land* (không được phép: *drawable-land-hdpi*)
- Thư mục tài nguyên được lựa chọn không được lồng vào nhau
→ Không được phép */res/drawable/drawable-en*
- Chỉ một giá trị cho loại từ hạn định được hỗ trợ cho một thư mục
→ Không được phép *drawable-land-port*

Truy cập tài nguyên

Trong JAVA code:

- Cú pháp: **R.<resource_type>.<resource_name>**
- *Ví dụ: R.drawable.background, R.id.tvResult, R.string.app_name*

Trong XML:

- Cú pháp: @**resource_type**/**resource_name**
- @drawable/background, @id/tvResult, @string/app_name

String Resources

- **Sử dụng:** Cung cấp định nghĩa chuỗi văn bản
- **Vị trí:** /res/values/filename.xml
- **Cú pháp**
 - `<string name="string_name">string_value</string>`
- **Tham chiếu tài nguyên:**
 - Java: R.string.string_name
 - XML: @string/string_name

Color Resources

- **Sử dụng:** Cung cấp định nghĩa màu sắc
- **Vị trí:** /res/values/filename.xml
- **Cú pháp**
 - `<color name="color_name">color_value</color>`
- **Tham chiếu tài nguyên:**
 - Java: R.color.color_name
 - XML: @color/color_name

Dimension Resources

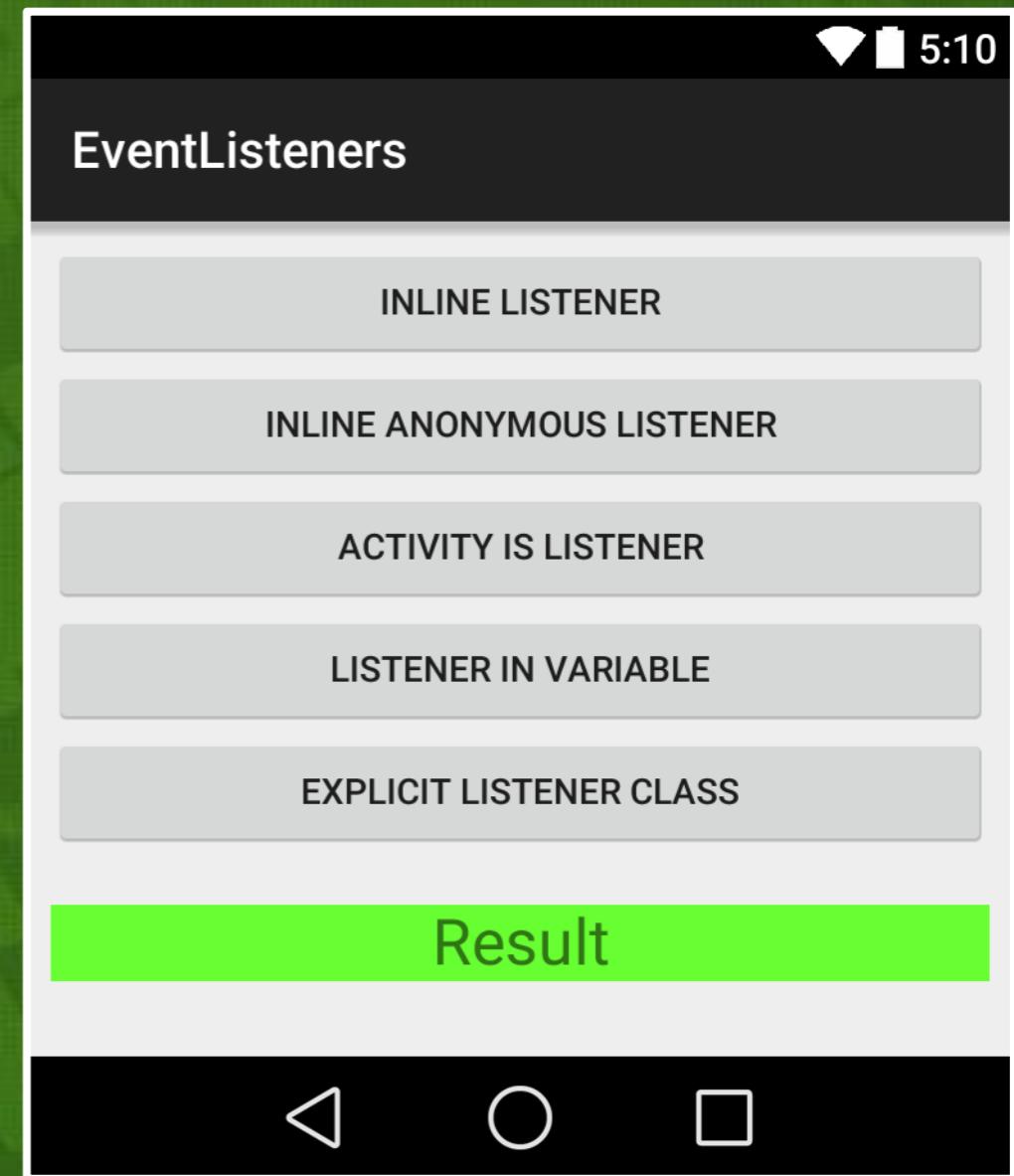
- **Sử dụng:** định kích thước cho các thành phần theo nhiều đơn vị, pixel (px), inch (in), point (pt), density-independent pixel (dp), scale-independent pixel (sp)
- **Vị trí:** /res/values/filename.xml
- **Cú pháp**
 - `<dimen name="dimen_name">dimen_value</color>`
- **Tham chiếu tài nguyên:**
 - Java: `R.dimen.dimen_name`
 - XML: `@dimen/dimen_name`

Chương 3: Android Form Widgets

1. Sử dụng *TextView*, *EditText*, *Button* và *ToggleButton*
2. Sử dụng *Android Resource cơ bản*
3. ***Event Listener***
4. Xây dựng app “*Lời hay ý đẹp*”
5. *Android's virtual keyboard*
6. Xây dựng app “*Máy tính điện tử*”
7. Xây dựng “*Chiếc cân vui vẻ*”
8. Xây dựng app “*Tìm số bí mật*”
9. *ImageView* và các thao tác xử lý
10. Xây dựng game “*Card*”
11. Xây dựng app “*Random Number*” với *Checkbox* và *RadioButton*

Event Listeners

- *Inline listener*
- *Inline anonymous listener*
- *Activity is listener*
- *Listener is variable*
- *Explicit listener class*
- *View Subclassing*



Chương 3: Android Form Widgets

1. Sử dụng *TextView*, *EditText*, *Button* và *ToggleButton*
2. Sử dụng *Android Resource cơ bản*
3. *Event Listener*
4. **Xây dựng app “Lời hay ý đẹp”**
5. *Android's virtual keyboard*
6. Xây dựng app “Máy tính điện tử”
7. Xây dựng “Chiếc cân vui vẻ”
8. Xây dựng app “Tìm số bí mật”
9. *ImageView* và các thao tác xử lý
10. Xây dựng game “Card”
11. Xây dựng app “Random Number” với *Checkbox* và *RadioButton*

SHOW QUOTATIONS

I have learned more from my failures
than from my success



Quotations

Xây dựng app Quotations (*Lời hay ý đẹp*)

Thiết kế đúng giao diện (tùy ý sử dụng loại layout)

Quotation hiển thị ngẫu nhiên: nội dung, vị trí, màu sắc và kích thước

Quotation tự động thay đổi nội dung cứ sau một khoảng thời gian nào đó

Chương 3: Android Form Widgets

1. Sử dụng *TextView*, *EditText*, *Button* và *ToggleButton*
2. Sử dụng *Android Resource cơ bản*
3. *Event Listener*
4. Xây dựng app “*Lời hay ý đẹp*”
5. *Android's virtual keyboard*
6. Xây dựng app “*Máy tính điện tử*”
7. Xây dựng “*Chiếc cân vui vẻ*”
8. Xây dựng app “*Tìm số bí mật*”
9. *ImageView* và các thao tác xử lý
10. Xây dựng game “*Card*”
11. Xây dựng app “*Random Number*” với *Checkbox* và *RadioButton*

Android's virtual keyboard

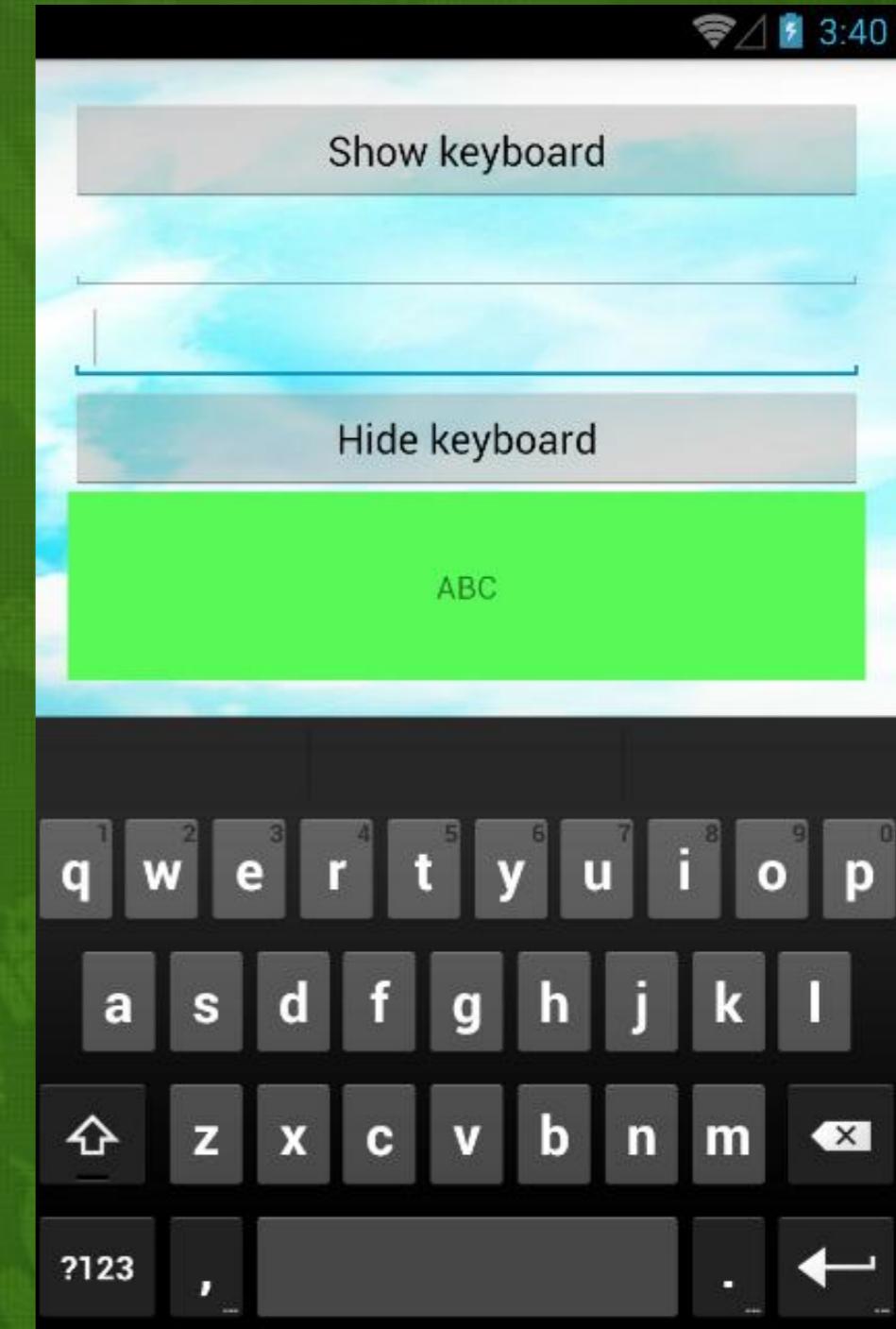
setSoftInputMode

- *AndroidManifest.xml*
- *Activity*

requestFocus

InputMethodManager

- *showSoftInput*
- *hideSoftInputFromWindow*



Chương 3: Android Form Widgets

1. Sử dụng *TextView*, *EditText*, *Button* và *ToggleButton*
2. Sử dụng *Android Resource cơ bản*
3. *Event Listener*
4. Xây dựng app “*Lời hay ý đẹp*”
5. *Android's virtual keyboard*
6. Xây dựng app “*Máy tính điện tử*”
7. Xây dựng “*Chiếc cân vui vẻ*”
8. Xây dựng app “*Tìm số bí mật*”
9. *ImageView* và các thao tác xử lý
10. Xây dựng game “*Card*”
11. Xây dựng app “*Random Number*” với *Checkbox* và *RadioButton*

Number One

Number Two

+

-

*

/

$$15 + 20 = 35$$



Caculator

Xây dựng chương trình máy tính đơn giản

Thiết kế đúng giao diện (tùy ý sử dụng loại layout)

Thực hiện các phép toán cộng, trừ, nhân, chia trên hai số nguyên với nhau

Kết quả hiển thị có dạng:

$$15 + 20 = 35$$

Chương 3: Android Form Widgets

1. Sử dụng *TextView*, *EditText*, *Button* và *ToggleButton*
2. Sử dụng *Android Resource cơ bản*
3. *Event Listener*
4. Xây dựng app “*Lời hay ý đẹp*”
5. *Android's virtual keyboard*
6. Xây dựng app “*Máy tính điện tử*”
7. Xây dựng “*Chiếc cân vui vẻ*”
8. Xây dựng app “*Tìm số bí mật*”
9. *ImageView* và các thao tác xử lý
10. Xây dựng game “*Card*”
11. Xây dựng app “*Random Number*” với *Checkbox* và *RadioButton*

Chiều cao

Cân nặng

XEM KẾT QUẢ

Chỉ số BMI của bạn là 25.5 - Bạn bị béo phì độ I

Where do you stand?



BMI

Xây dựng chương trình “Chiếc cân vui vẻ”

Thiết kế đúng giao diện (tùy ý sử dụng loại layout)

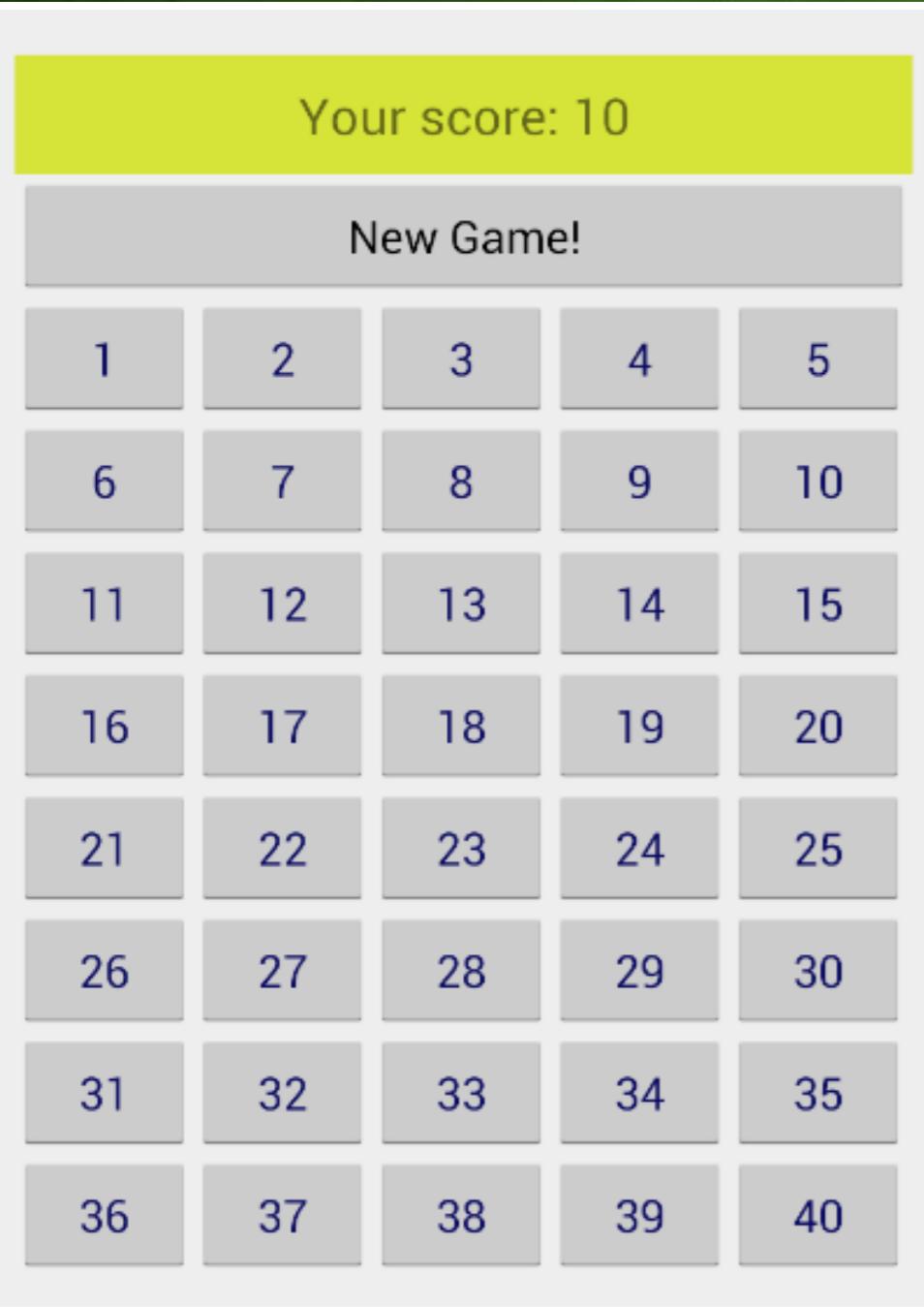
Tính chỉ số BMI và đưa ra các thông báo về thể trọng của người sử dụng

$$BMI = (\text{weight}) / (\text{height} * \text{height})$$

- Weight có đơn vị kg – Height có đơn vị là m

Chương 3: Android Form Widgets

1. Sử dụng *TextView*, *EditText*, *Button* và *ToggleButton*
2. Sử dụng *Android Resource cơ bản*
3. *Event Listener*
4. Xây dựng app “*Lời hay ý đẹp*”
5. *Android's virtual keyboard*
6. Xây dựng app “*Máy tính điện tử*”
7. Xây dựng “*Chiếc cân vui vẻ*”
8. **Xây dựng app “*Tìm số bí mật*”**
9. *ImageView và các thao tác xử lý*
10. Xây dựng game “*Card*”
11. Xây dựng app “*Random Number*” với *Checkbox* và *RadioButton*



FindMe

Xây dựng chương trình “Tìm số bí mật”

Thiết kế đúng giao diện (tùy ý sử dụng loại layout)

Phát sinh một số bí mật, nếu người dùng chọn đúng
số bí mật sẽ chiến thắng

Người dùng có thể chơi lại nếu thất bại ở lần chơi
trước

Chương 3: Android Form Widgets

1. Sử dụng *TextView*, *EditText*, *Button* và *ToggleButton*
2. Sử dụng *Android Resource cơ bản*
3. *Event Listener*
4. Xây dựng app “*Lời hay ý đẹp*”
5. *Android's virtual keyboard*
6. Xây dựng app “*Máy tính điện tử*”
7. Xây dựng “*Chiếc cân vui vẻ*”
8. Xây dựng app “*Tìm số bí mật*”
9. *ImageView và các thao tác xử lý*
10. Xây dựng game “*Card*”
11. Xây dựng app “*Random Number*” với *Checkbox* và *RadioButton*

ImageView và các thao tác xử lý



Tìm hiểu về đối tượng ImageView

Cách thiết lập hình ảnh cho ImageView thông qua XML và Java code

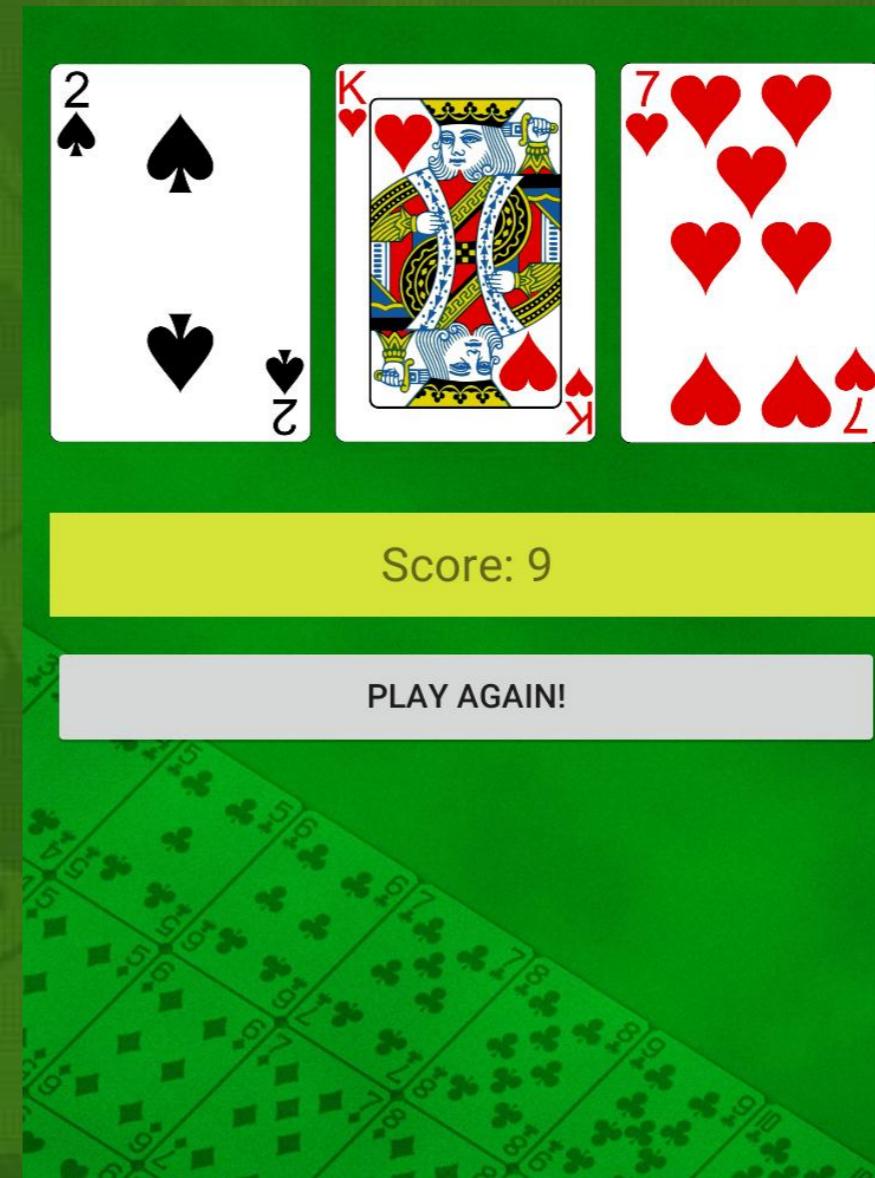
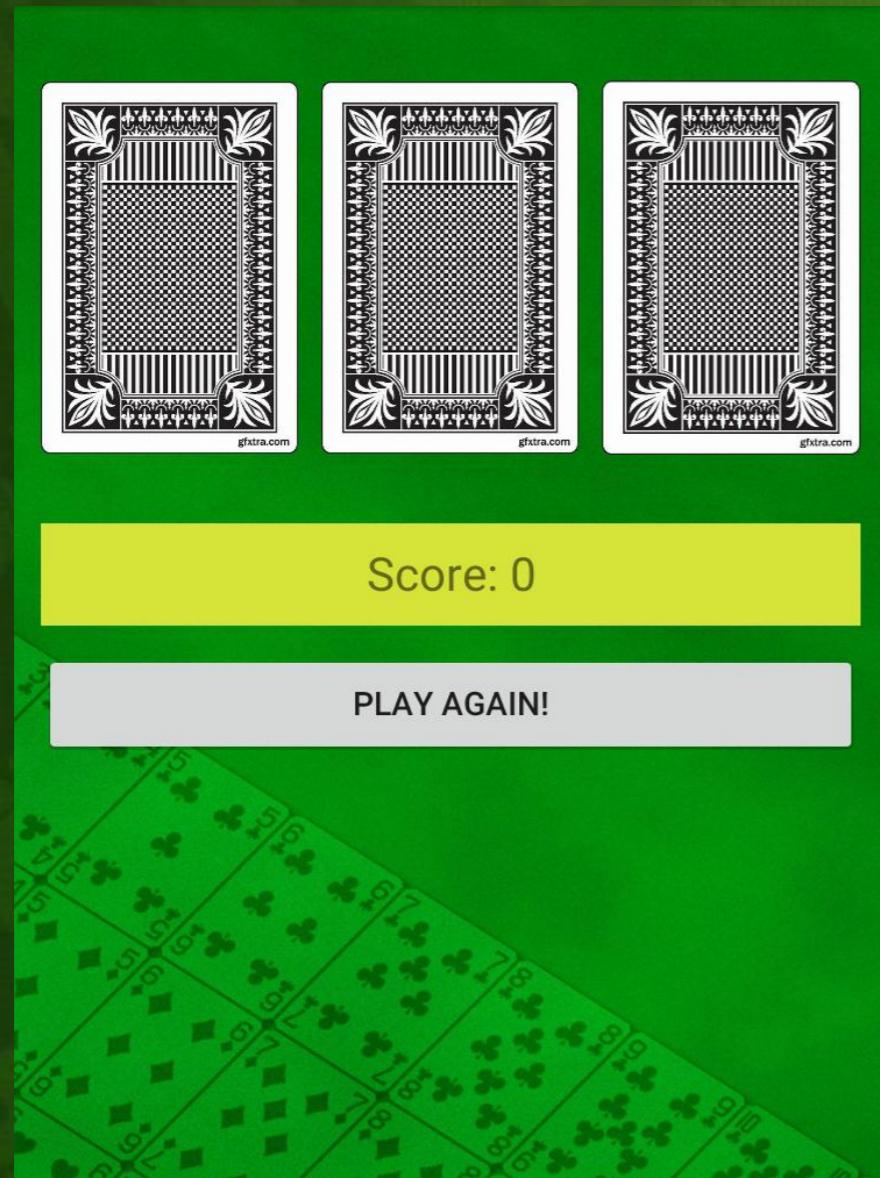
Thay đổi hình ảnh ngẫu nhiên cho ImageView

Thay đổi background ngẫu nhiên cho ứng dụng

Chương 3: Android Form Widgets

1. Sử dụng *TextView*, *EditText*, *Button* và *ToggleButton*
2. Sử dụng *Android Resource cơ bản*
3. *Event Listener*
4. Xây dựng app “*Lời hay ý đẹp*”
5. *Android's virtual keyboard*
6. Xây dựng app “*Máy tính điện tử*”
7. Xây dựng “*Chiếc cân vui vẻ*”
8. Xây dựng app “*Tìm số bí mật*”
9. *ImageView* và các thao tác xử lý
- 10. Xây dựng game “*Card*”**
11. Xây dựng app “*Random Number*” với *Checkbox* và *RadioButton*

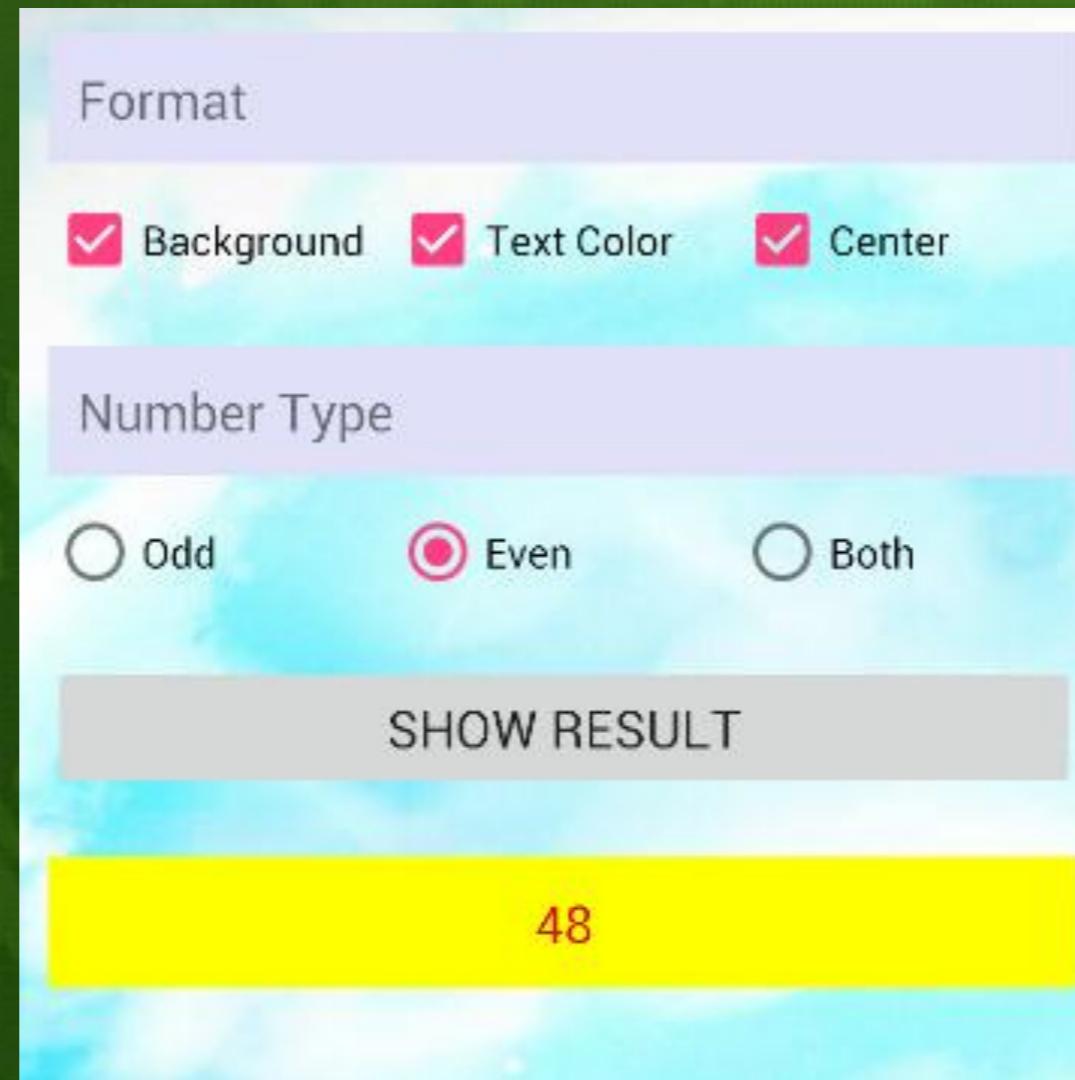
Ứng dụng ImageView với game Card



Chương 3: Android Form Widgets

1. Sử dụng *TextView*, *EditText*, *Button* và *ToggleButton*
2. Sử dụng *Android Resource cơ bản*
3. *Event Listener*
4. Xây dựng app “*Lời hay ý đẹp*”
5. *Android's virtual keyboard*
6. Xây dựng app “*Máy tính điện tử*”
7. Xây dựng “*Chiếc cân vui vẻ*”
8. Xây dựng app “*Tìm số bí mật*”
9. *ImageView* và các thao tác xử lý
10. Xây dựng game “*Card*”
11. Xây dựng app “*Random Number*” với *Checkbox* và *RadioButton*

Xây dựng app “Random Number” với Checkbox và RadioButton



Chương 04

Menu – ContextMenu – DiaLog

1. Thiết lập và sử dụng Menu
2. Thiết lập và sử dụng Popup Menu
3. Làm việc với ContextMenu
4. Sử dụng Dialog
5. Custom giao diện cho Dialog
6. DatePickerDialog và TimePickerDialog

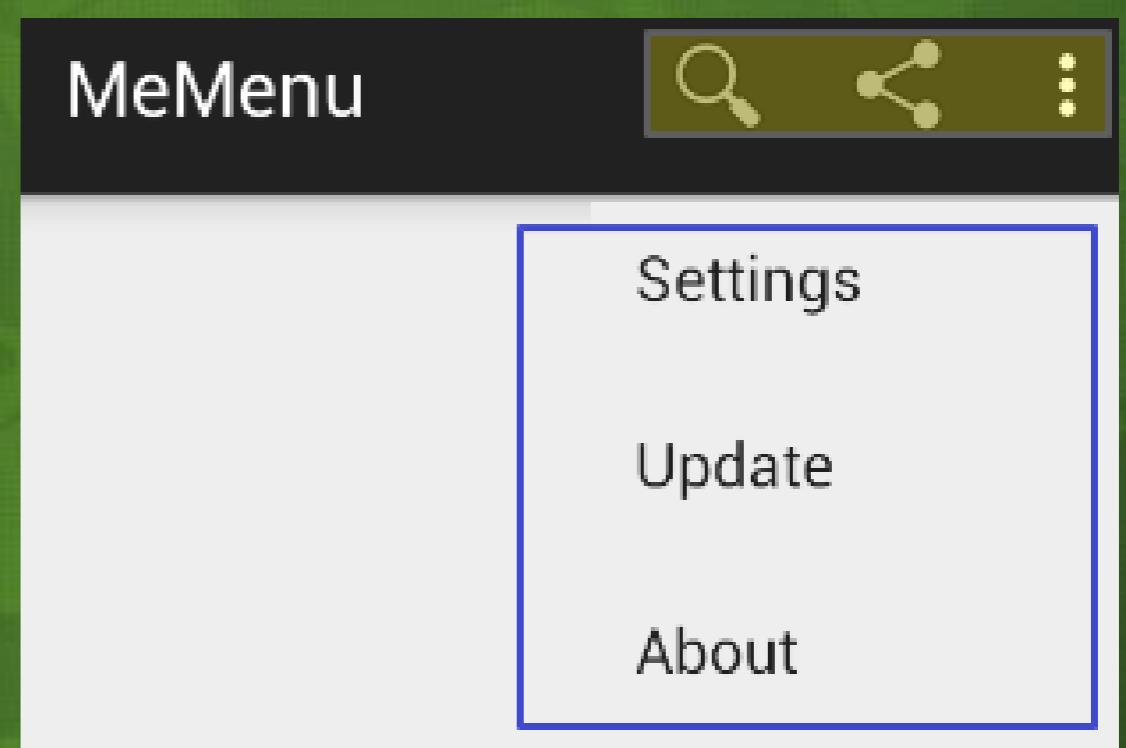
Chương 04

Menu – ContextMenu – DiaLog

1. *Thiết lập và sử dụng Menu*
2. *Thiết lập và sử dụng Popup Menu*
3. *Làm việc với ContextMenu*
4. *Sử dụng Dialog*
5. *Custom giao diện cho Dialog*
6. *DatePickerDialog và TimePickerDialog*

Thiết lập và sử dụng Menu

- *Tạo menu trong tập tin XML*
- *Tạo menu trong Java code*
- *Bắt sự kiện khi người dùng chọn vào một menu nào đó*
- *Keyword:* `showAsAction`, `orderInCategory`, `onCreateOptionsMenu`, `onOptionsItemSelected`



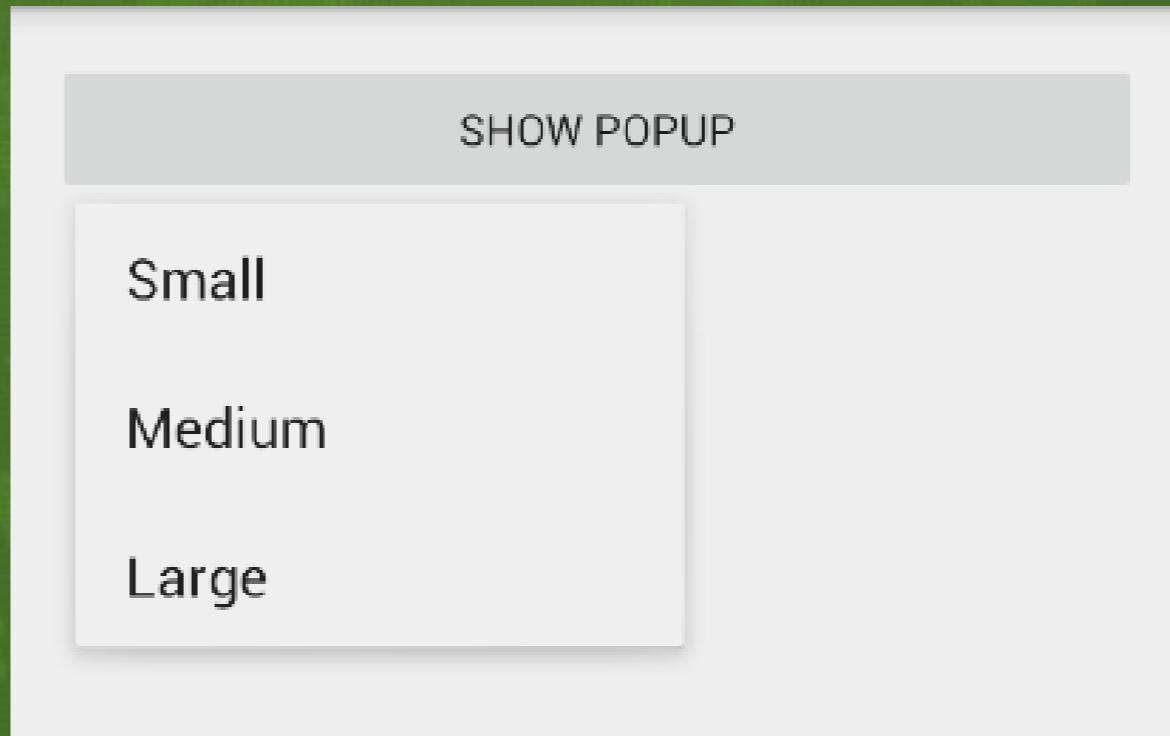
Chương 04

Menu – ContextMenu – DiaLog

1. Thiết lập và sử dụng Menu
2. Thiết lập và sử dụng Popup Menu
3. Làm việc với ContextMenu
4. Sử dụng Dialog
5. Custom giao diện cho Dialog
6. DatePickerDialog và TimePickerDialog

Thiết lập và sử dụng Popup Menu

- *Tạo menu trong tập tin XML*
- *Tạo menu trong Java code*
- *Sự kiện khi người dùng chọn vào menu*



Keyword: `PopupMenu`, `setOnMenuItemClickListener`,

`PopupMenu.OnMenuItemClickListener`

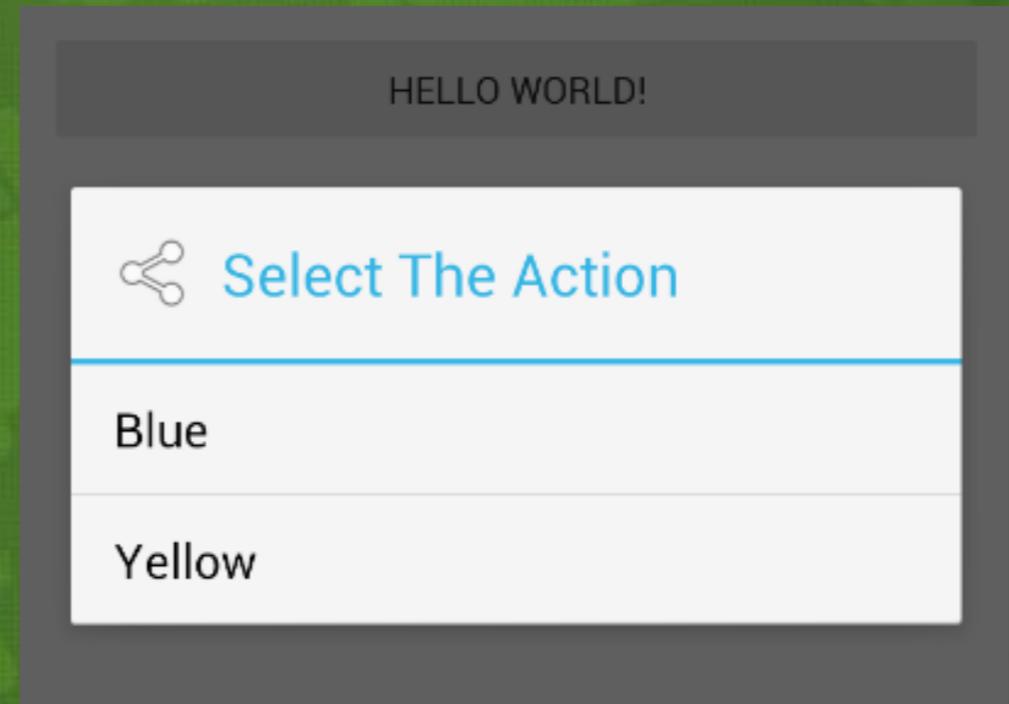
Chương 04

Menu – ContextMenu – DiaLog

1. Thiết lập và sử dụng Menu
2. Thiết lập và sử dụng Popup Menu
3. **Làm việc với ContextMenu**
4. Sử dụng Dialog
5. Custom giao diện cho Dialog
6. DatePickerDialog và TimePickerDialog

Làm việc với Context Menu

- Xảy ra khi người dùng nhấn và giữ đối với một thành phần điều khiển nào đó
- Tạo menu trong tập tin XML
- Tạo menu trong Java code
- Sự kiện khi người dùng chọn vào menu



Keyword: onCreateContextMenu, onContextItemSelected

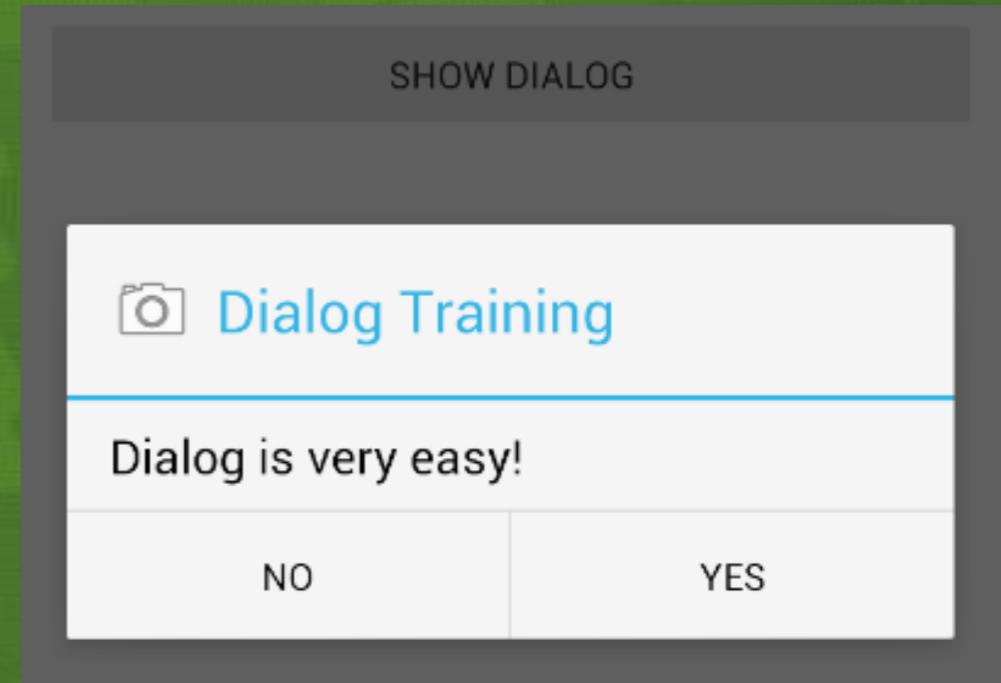
Chương 04

Menu – ContextMenu – DiaLog

- 1. Thiết lập và sử dụng Menu
- 2. Thiết lập và sử dụng Popup Menu
- 3. Làm việc với ContextMenu
- 4. Sử dụng Dialog
- 5. Custom giao diện cho Dialog
- 6. DatePickerDialog và TimePickerDialog

Sử dụng Dialog

- *Hộp thoại thông báo và yêu cầu xác nhận từ phía người sử dụng*
- *Xây dựng giao diện AlertDialog đơn giản*
- *Xây dựng giao diện AlertDialog với nhiều lựa chọn*
- *Xử lý sự kiện đối với các button trên AlertDialog*



Keyword: `AlertDialog`, `setCanceledOnTouchOutside`, `setPositiveButton`, `setNegativeButton`, `show`, `setIcon`, `setMessage`, `setTitle`

Chương 04

Menu – ContextMenu – DiaLog

1. Thiết lập và sử dụng Menu
2. Thiết lập và sử dụng Popup Menu
3. Làm việc với ContextMenu
4. Sử dụng Dialog
5. *Custom giao diện cho Dialog*
6. DatePickerDialog và TimePickerDialog

Custom giao diện cho Dialog

SHOW DIALOG

Your Character



Chương 04

Menu – ContextMenu – DiaLog

1. Thiết lập và sử dụng Menu
2. Thiết lập và sử dụng Popup Menu
3. Làm việc với ContextMenu
4. Sử dụng Dialog
5. Custom giao diện cho Dialog
6. **DatePickerDialog** và **TimePickerDialog**

DatePickerDialog

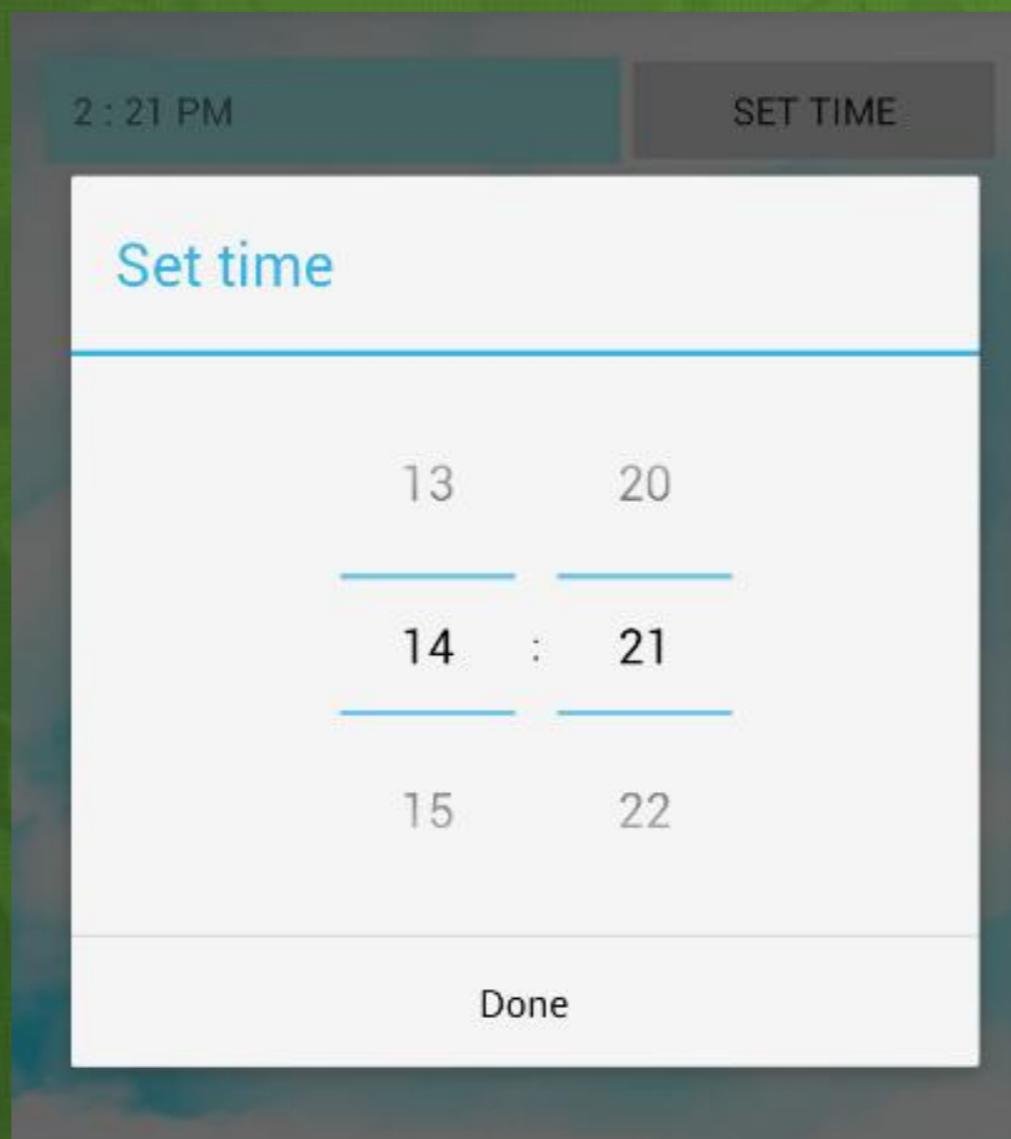
07-10-2015

09-10-2015

SHOW DISTANCE

Khoảng cách giữa 2 mốc thời gian là: 2 ngày

TimePickerDialog



Chương 05

Sử dụng Intent trong Android

1. *Chuyển đổi giữa các màn hình*
2. *Gửi và nhận dữ liệu giữa các Activity – Basic*
3. *Implicitly Intent vs Explicitly Intent*
4. *Gửi và nhận dữ liệu giữa các Activity – Advanced*
5. *Sử dụng Intent Filters*
6. *Xây dựng ứng dụng “Your memory”*

Chương 05

Sử dụng Intent trong Android

1. *Chuyển đổi giữa các màn hình*
2. *Gửi và nhận dữ liệu giữa các Activity – Basic*
3. *Implicitly Intent vs Explicitly Intent*
4. *Gửi và nhận dữ liệu giữa các Activity – Advanced*
5. *Sử dụng Intent Filters*
6. *Xây dựng ứng dụng “Your memory”*

Những điều cần biết về Intent

Intent được xem như bộ máy truyền thông điệp giữa các thành phần trong một ứng dụng, hoặc giữa các ứng dụng với nhau

Được dùng cho các trường hợp

- *Truyền thông điệp đến Activity hay Service*
- *Truyền các gói xử lý đến các thành phần ứng dụng*
- *Kích hoạt một Activity hay Service*
- ...

startActivity()

```
Intent myIntent = new Intent(MainActivity.this,  
                             ChildActivity.class);  
  
startActivity(myIntent);
```



Chương 05

Sử dụng Intent trong Android

1. *Chuyển đổi giữa các màn hình*
2. *Gửi và nhận dữ liệu giữa các Activity - Basic*
3. *Implicitly Intent vs Explicitly Intent*
4. *Gửi và nhận dữ liệu giữa các Activity - Advanced*
5. *Sử dụng Intent Filters*
6. *Xây dựng ứng dụng “Your memory”*

Gửi thông điệp kèm dữ liệu với Intent (1)

```
Intent myIntent = new Intent(FirstActivity.this,  
                                SecondActivity.class);  
  
myIntent.putExtra("param_1", "Android");  
  
myIntent.putExtra("param_2", 123);  
  
startActivity(myIntent);
```

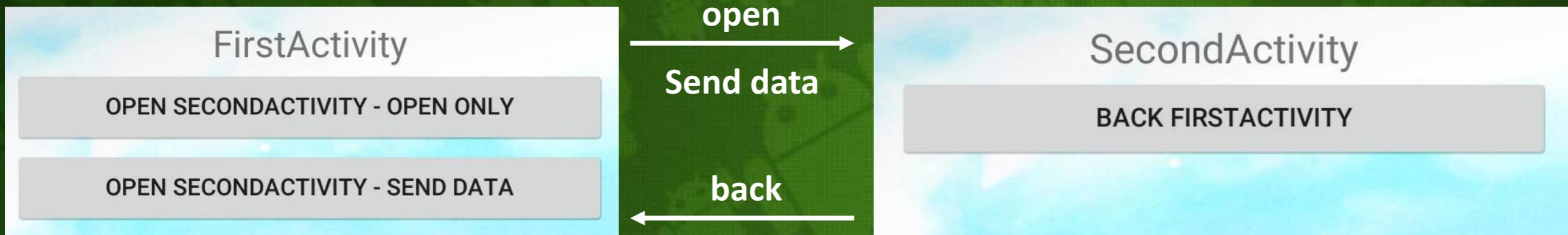


Gửi thông điệp kèm dữ liệu với Intent (2)

Truyền dữ liệu thuộc kiểu dữ liệu cơ bản: int, float, string

Truyền dữ liệu thuộc kiểu dữ liệu object

Cách nhận, đọc và xử lý dữ liệu từ Intent



Gửi thông điệp kèm dữ liệu với Intent (3)

```
Bundle bundle = new Bundle();
```

```
...
```

```
myIntent.putExtra("MyData", bundle);
```

```
startActivity(myIntent);
```



Chương 05

Sử dụng Intent trong Android

1. *Chuyển đổi giữa các màn hình*
2. *Gửi và nhận dữ liệu giữa các Activity – Basic*
3. *Implicitly Intent vs Explicitly Intent*
4. *Gửi và nhận dữ liệu giữa các Activity – Advanced*
5. *Sử dụng Intent Filters*
6. *Xây dựng ứng dụng “Your memory”*

Implicitly Intent vs Explicitly Intent (1)

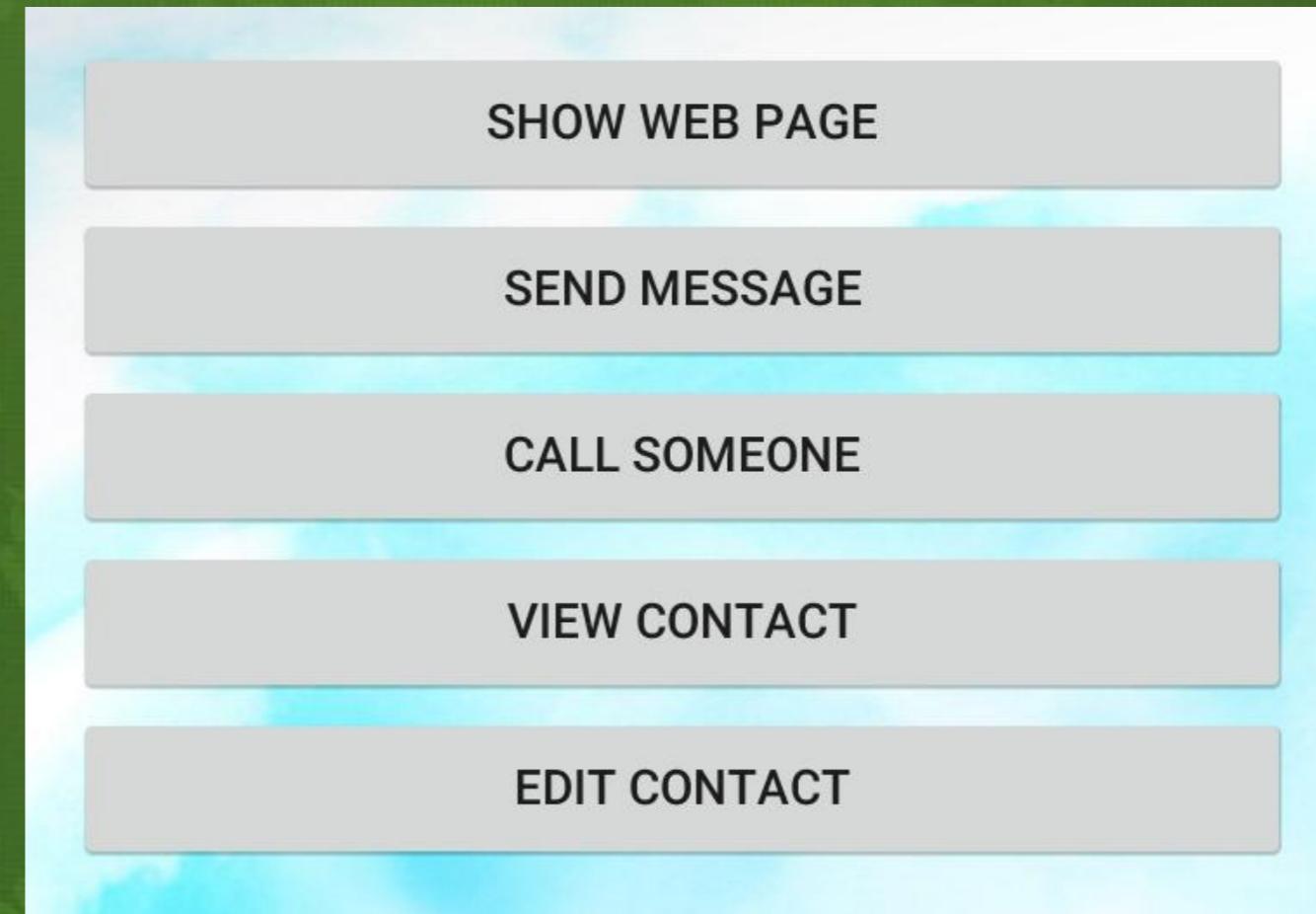
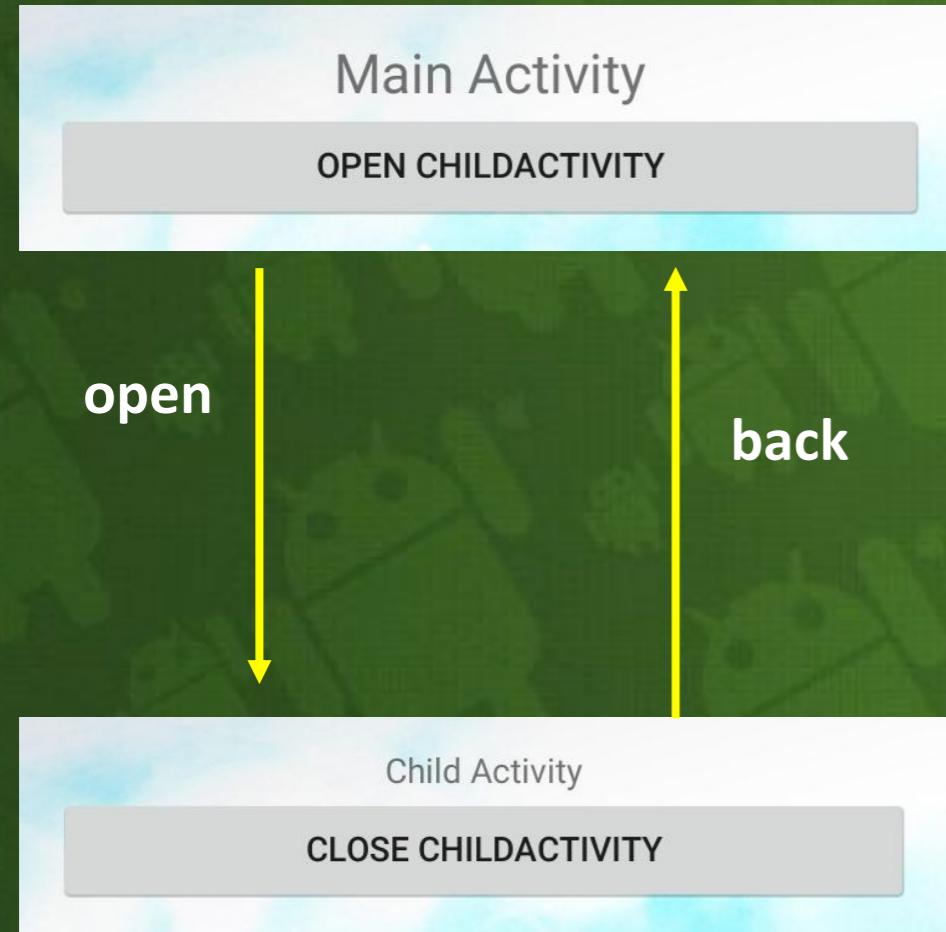
Implicitly Intent (Không xác định)

- Không chỉ rõ component sẽ xử lý công việc được chỉ định
- Sử dụng: Show web, Call, Edit Contact, Map, ...

Explicitly Intent (Xác định)

- Chỉ rõ component sẽ xử lý công việc được chỉ định
- Sử dụng: mở một Activity trong ứng dụng

Implicitly Intent vs Explicitly Intent (2)

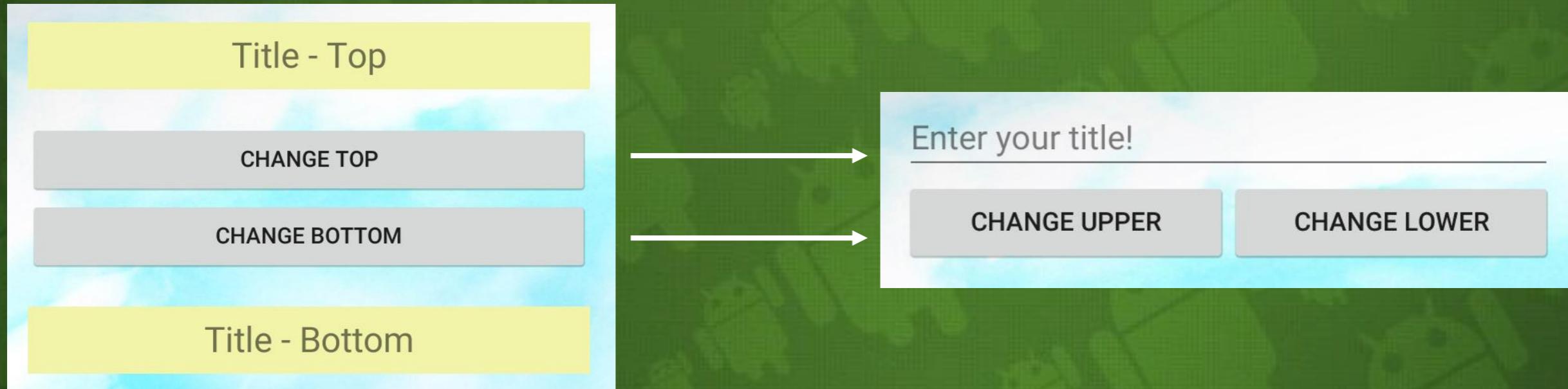


Chương 05

Sử dụng Intent trong Android

1. *Chuyển đổi giữa các màn hình*
2. *Gửi và nhận dữ liệu giữa các Activity - Basic*
3. *Implicitly Intent vs Explicitly Intent*
4. *Gửi và nhận dữ liệu giữa các Activity - Advanced*
5. *Sử dụng Intent Filters*
6. *Xây dựng ứng dụng “Your memory”*

startActivityForResult()



Keyword: startActivityForResult, onActivityResult

Chương 05

Sử dụng Intent trong Android

1. *Chuyển đổi giữa các màn hình*
2. *Gửi và nhận dữ liệu giữa các Activity – Basic*
3. *Implicitly Intent vs Explicitly Intent*
4. *Gửi và nhận dữ liệu giữa các Activity – Advanced*
5. **Sử dụng Intent Filters**
6. *Xây dựng ứng dụng “Your memory”*

Intent filter

<intent-filter> nhận các thông điệp từ bên ngoài ứng dụng

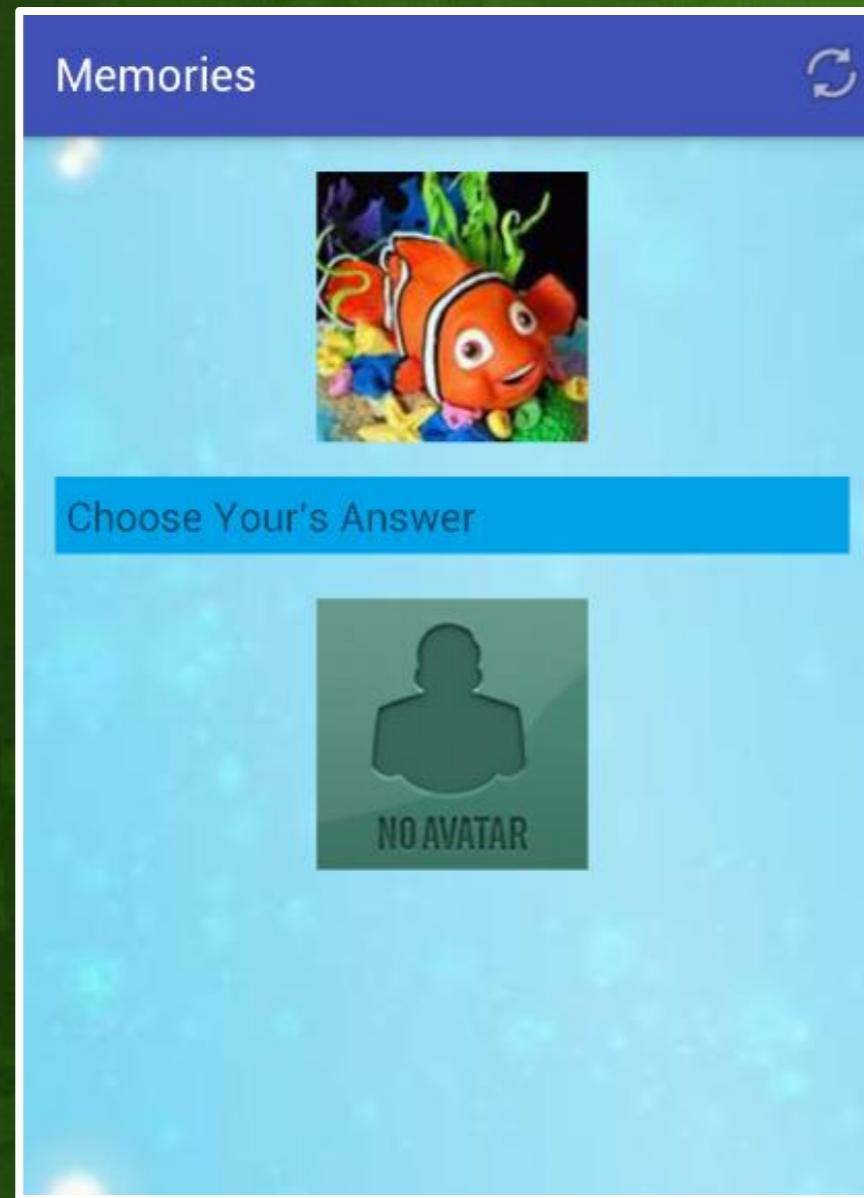
```
<activity android:name=".FilterActivity" >  
    <intent-filter>  
        <action android:name="android.intent.action.VIEW"/>  
        <category android:name="android.intent.category.DEFAULT"/>  
        <data android:scheme="http"/>  
    </intent-filter>  
    <intent-filter>...</intent-filter>  
</activity>
```

Chương 05

Sử dụng Intent trong Android

1. *Chuyển đổi giữa các màn hình*
2. *Gửi và nhận dữ liệu giữa các Activity – Basic*
3. *Implicitly Intent vs Explicitly Intent*
4. *Gửi và nhận dữ liệu giữa các Activity – Advanced*
5. *Sử dụng Intent Filters*
6. **Xây dựng ứng dụng “Your memory”**

Xây dựng ứng dụng Game “Your Memory”



Chương 06

Drawable Resource và xây dựng giao diện

1. *Bitmap File*
2. *Layer List Drawable*
3. *Shape Drawable*
4. *State Drawable*
5. *Level List*
6. *Transition Drawable*
7. *Clip Drawable*
8. *ProgressBar và tùy biến giao diện*
9. *SeekBar và tùy biến giao diện*
10. *RatingBar và tùy biến giao diện*
11. *Xây dựng tính năng multi language*
12. *Xây dựng giao diện trên nhiều thiết bị*
13. *Sử dụng style trong định nghĩa giao diện*
14. *Tổng hợp kiến thức Drawable Resource*

Chương 06

Drawable Resource và xây dựng giao diện

1. *Bitmap File*
2. *Layer List Drawable*
3. *Shape Drawable*
4. *State Drawable*
5. *Level List*
6. *Transition Drawable*
7. *Clip Drawable*
8. *ProgressBar và tùy biến giao diện*
9. *SeekBar và tùy biến giao diện*
10. *RatingBar và tùy biến giao diện*
11. *Xây dựng tính năng multi language*
12. *Xây dựng giao diện trên nhiều thiết bị*
13. *Sử dụng style trong định nghĩa giao diện*
14. *Tổng hợp kiến thức Drawable Resource*

Drawable Resource (1)

Giới thiệu: Chúng ta đã được tiếp cận các Resource cơ bản như String, Color, Dimension, ... và Drawable Resource cũng nằm trong nhóm các Adndroid Resource cơ bản này.

Sử dụng: Cung cấp định nghĩa liên quan đến hình ảnh cho ứng dụng

Vị trí: /res/drawable/filename.xml

Tham chiếu tài nguyên:

- Java: R.drawable.drawable_name
- XML: @drawable/drawable_name

Drawable Resource (2)

- *Bitmap File (jpg, png, gif)*
- *Nine-Patch File*
- *State List Drawable*
- *Layer List Drawable*
- *Level List*
- *Transition Drawable*
- *Shape Drawable*
- *Inset Drawable*
- *Clip Drawable*
- *Scale Drawable*
- ...

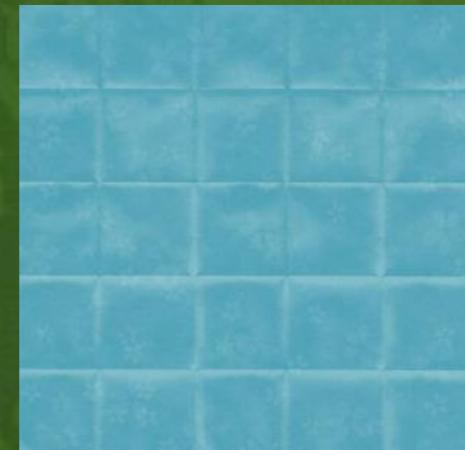
Chương 06

Drawable Resource và xây dựng giao diện

1. *Bitmap File*
2. ***Layer List Drawable***
3. *Shape Drawable*
4. *State Drawable*
5. *Level List*
6. *Transition Drawable*
7. *Clip Drawable*
8. *ProgressBar và tùy biến giao diện*
9. *SeekBar và tùy biến giao diện*
10. *RatingBar và tùy biến giao diện*
11. *Xây dựng tính năng multi language*
12. *Xây dựng giao diện trên nhiều thiết bị*
13. *Sử dụng style trong định nghĩa giao diện*
14. *Tổng hợp kiến thức Drawable Resource*

Layer List

Layer List chứa một loạt các item nằm chồng lên nhau. Item được khai báo sau sẽ nằm đè lên item được khai báo trước



=



Chương 06

Drawable Resource và xây dựng giao diện

1. *Bitmap File*
2. *Layer List Drawable*
3. ***Shape Drawable***
4. *State Drawable*
5. *Level List*
6. *Transition Drawable*
7. *Clip Drawable*
8. *ProgressBar và tùy biến giao diện*
9. *SeekBar và tùy biến giao diện*
10. *RatingBar và tùy biến giao diện*
11. *Xây dựng tính năng multi language*
12. *Xây dựng giao diện trên nhiều thiết bị*
13. *Sử dụng style trong định nghĩa giao diện*
14. *Tổng hợp kiến thức Drawable Resource*

Shape Drawable

ShapeDrawable định nghĩa các đối tượng hình học với phần định dạng màu nền, đường viền, padding, gradient, ...

Thuộc tính *android:shape* quy định dạng hình học
oval, *rectangle*, *ring*, *line*

Các thẻ con của *<shape>* cần quan tâm: *corner*,
gradient, *padding*, *size*, *solid*, *stroke*

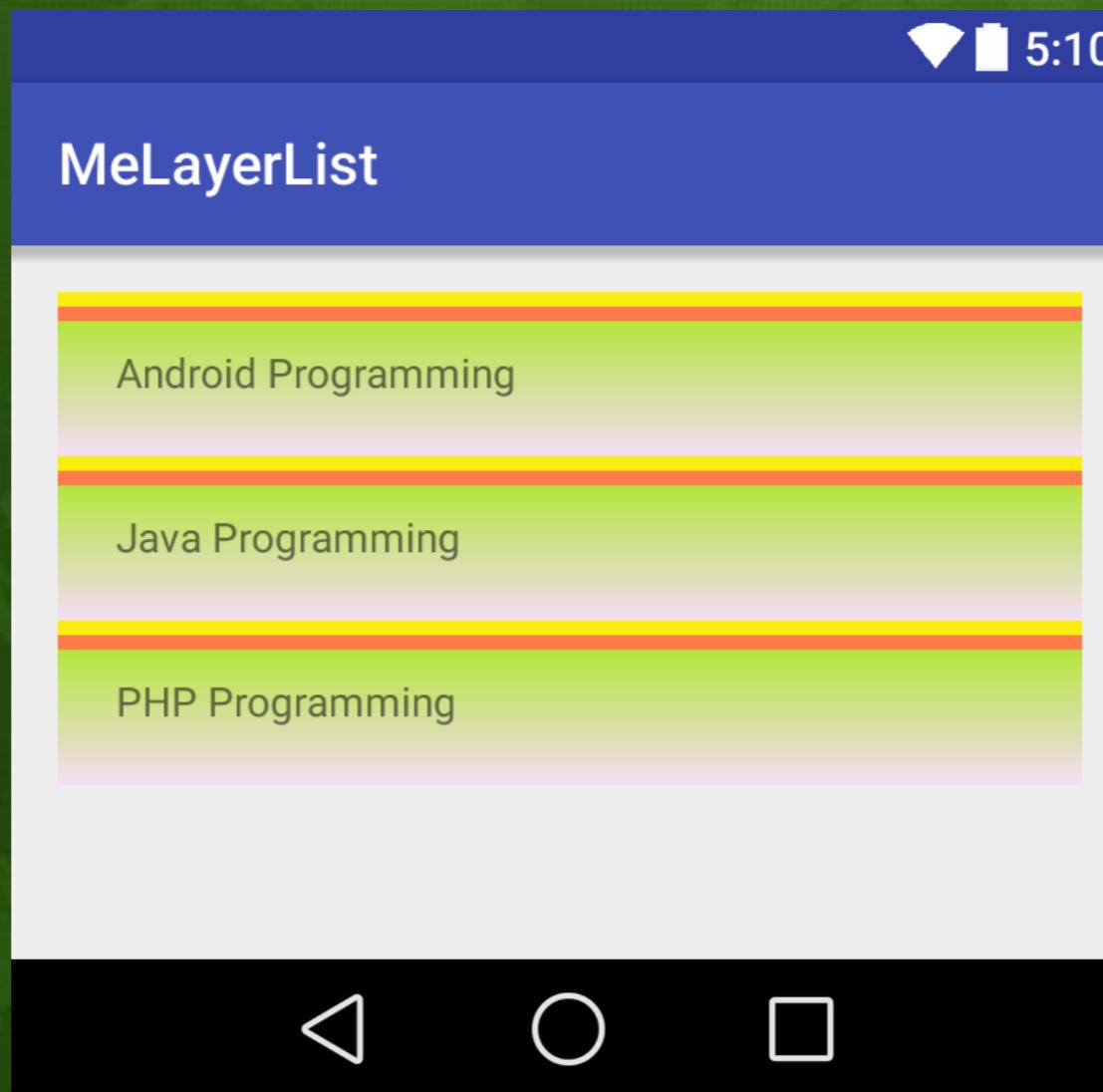
Shape Drawable

Shape Drawable

Android

Android

Tình huống xây dựng giao diện



Chương 06

Drawable Resource và xây dựng giao diện

1. *Bitmap File*
2. *Layer List Drawable*
3. *Shape Drawable*
- 4. State Drawable**
5. *Level List*
6. *Transition Drawable*
7. *Clip Drawable*
8. *ProgressBar và tùy biến giao diện*
9. *SeekBar và tùy biến giao diện*
10. *RatingBar và tùy biến giao diện*
11. *Xây dựng tính năng multi language*
12. *Xây dựng giao diện trên nhiều thiết bị*
13. *Sử dụng style trong định nghĩa giao diện*
14. *Tổng hợp kiến thức Drawable Resource*

Color State List

Color State List mô tả các trạng thái của một điều khiển nào đó bởi những màu sắc khác nhau

Ví dụ: thay đổi trạng thái của button, checkbox, ...

Các trạng thái được mô tả trong thẻ item, các trạng thái thường gặp: *pressed, focused, selected, checkable, checked, enabled, active, window_focused*

BUTTON

BUTTON

BUTTON

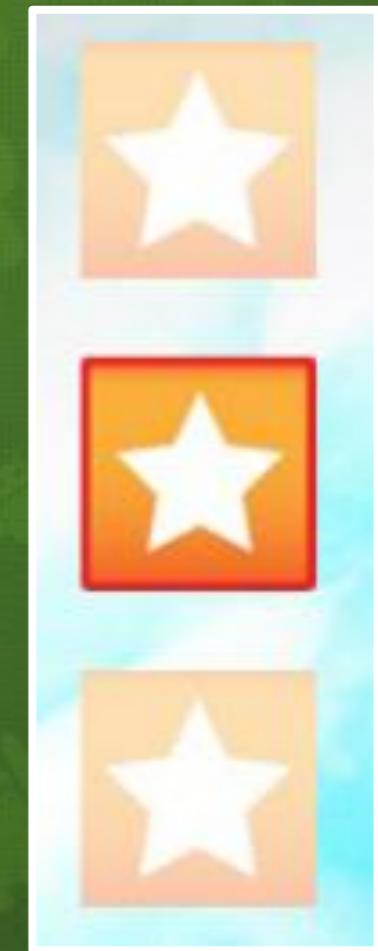
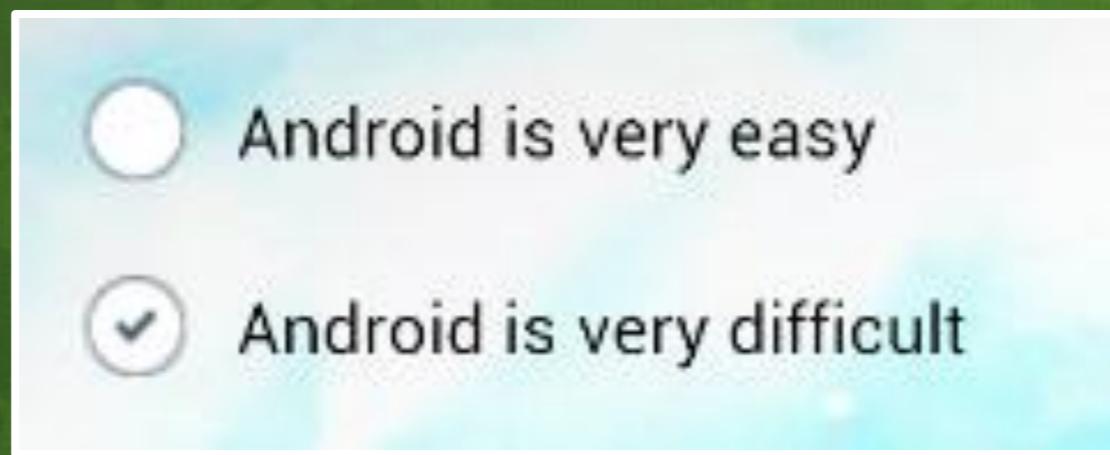
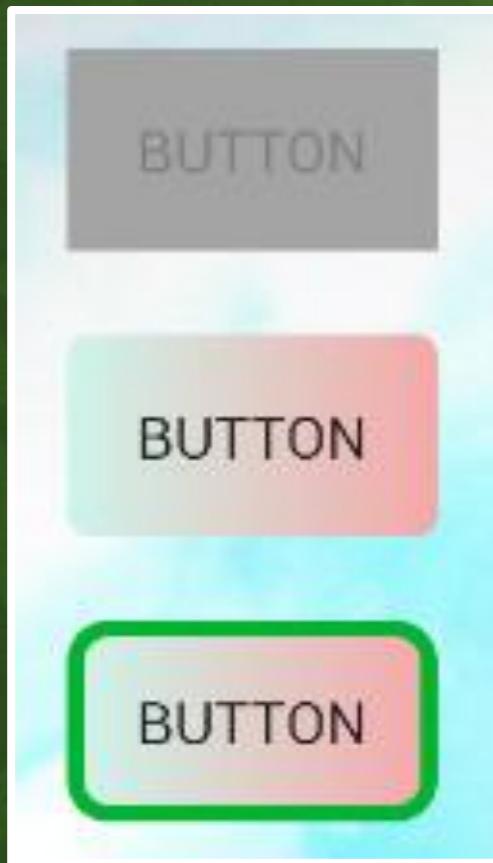
State List Drawable (1)

StateListDrawable mô tả các trạng thái của một điều khiển nào đó bởi những giao diện khác nhau

Ví dụ: thay đổi trạng thái của button, checkbox, ...

Các trạng thái được mô tả trong thẻ item, các trạng thái thường gặp: pressed, focused, selected, checkable, checked, enabled, active, window_focused

State List Drawable (2)



Chương 06

Drawable Resource và xây dựng giao diện

1. *Bitmap File*
2. *Layer List Drawable*
3. *Shape Drawable*
4. *State Drawable*
5. ***Level List***
6. *Transition Drawable*
7. *Clip Drawable*
8. *ProgressBar và tùy biến giao diện*
9. *SeekBar và tùy biến giao diện*
10. *RatingBar và tùy biến giao diện*
11. *Xây dựng tính năng multi language*
12. *Xây dựng giao diện trên nhiều thiết bị*
13. *Sử dụng style trong định nghĩa giao diện*
14. *Tổng hợp kiến thức Drawable Resource*

Level List Drawable (1)

LevelListDrawable cung cấp các drawable resource theo level mà nó được thiết lập

Keyword: *minLevel*, *maxLevel*, *setLevel*,

```
<?xml version="1.0" encoding="utf-8"?>
<level-list
    xmlns:android="http://schemas.android.com/apk/res/android">
    <item android:drawable="@drawable/image_one"
          android:minLevel="0"
          android:maxLevel="2"/>
</level-list>
```

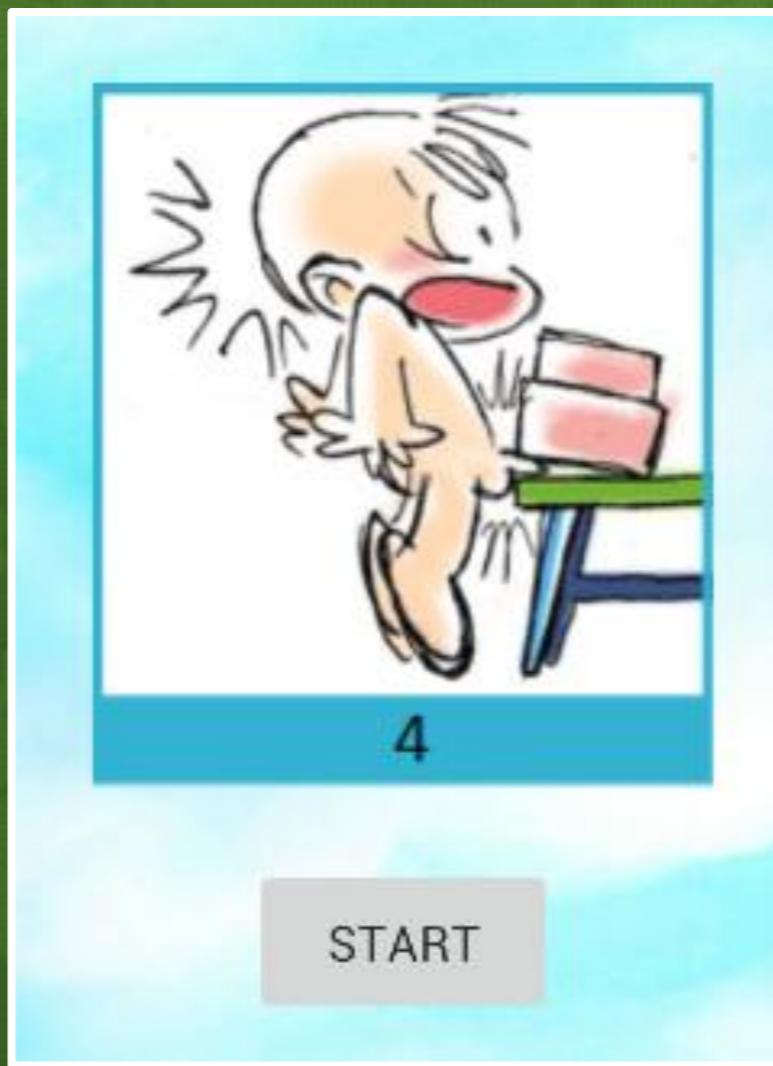
Level List Drawable (2)



PREVIOUS

NEXT

Level List Drawable (2)



Chương 06

Drawable Resource và xây dựng giao diện

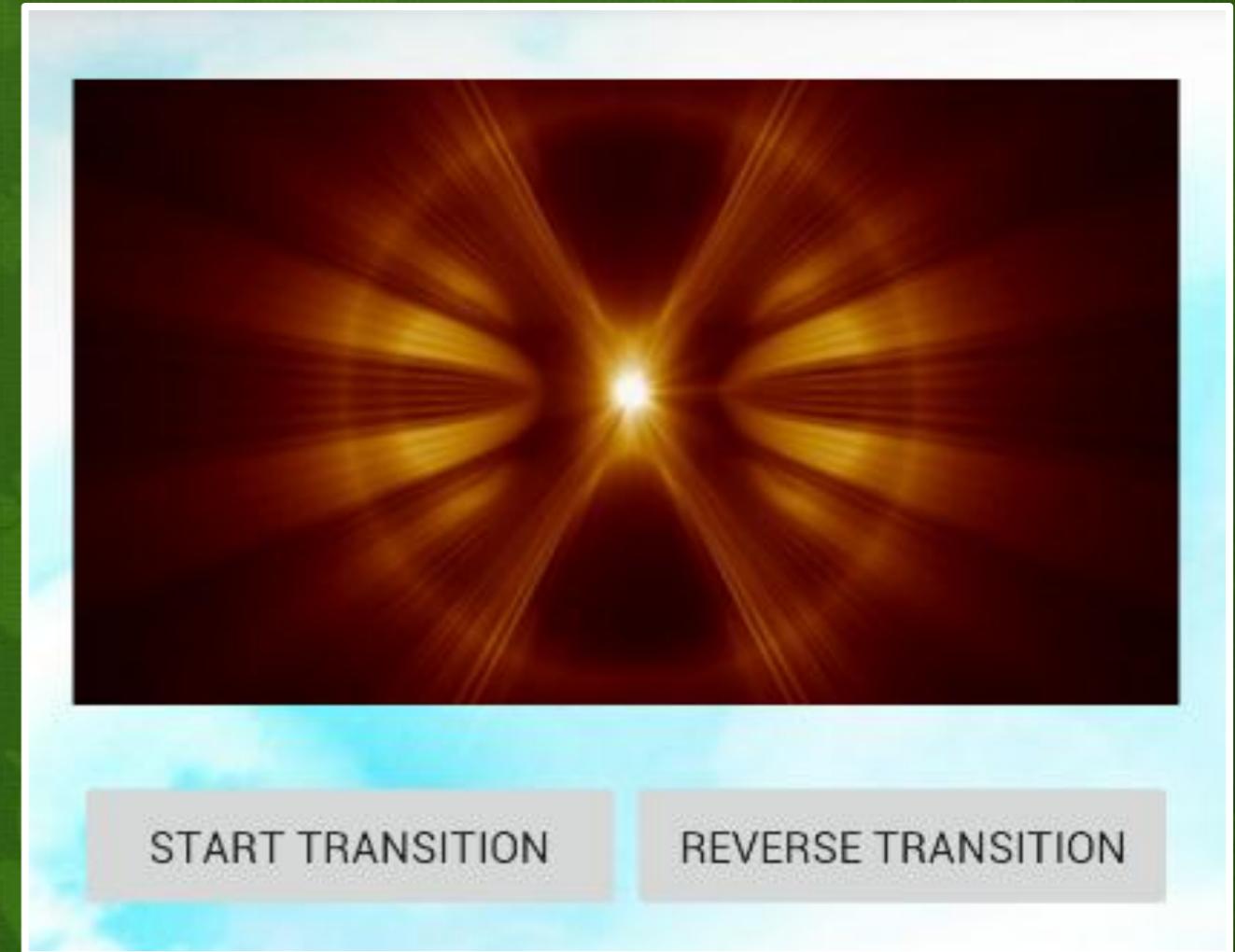
1. *Bitmap File*
2. *Layer List Drawable*
3. *Shape Drawable*
4. *State Drawable*
5. *Level List*
6. ***Transition Drawable***
7. *Clip Drawable*
8. *ProgressBar và tùy biến giao diện*
9. *SeekBar và tùy biến giao diện*
10. *RatingBar và tùy biến giao diện*
11. *Xây dựng tính năng multi language*
12. *Xây dựng giao diện trên nhiều thiết bị*
13. *Sử dụng style trong định nghĩa giao diện*
14. *Tổng hợp kiến thức Drawable Resource*

Transition Drawable

Transition Drawable cung cấp khả năng chuyển đổi giữa 2 item được khai báo

Keyword:

- *startTransition()*
- *reverseTransition()*



Chương 06

Drawable Resource và xây dựng giao diện

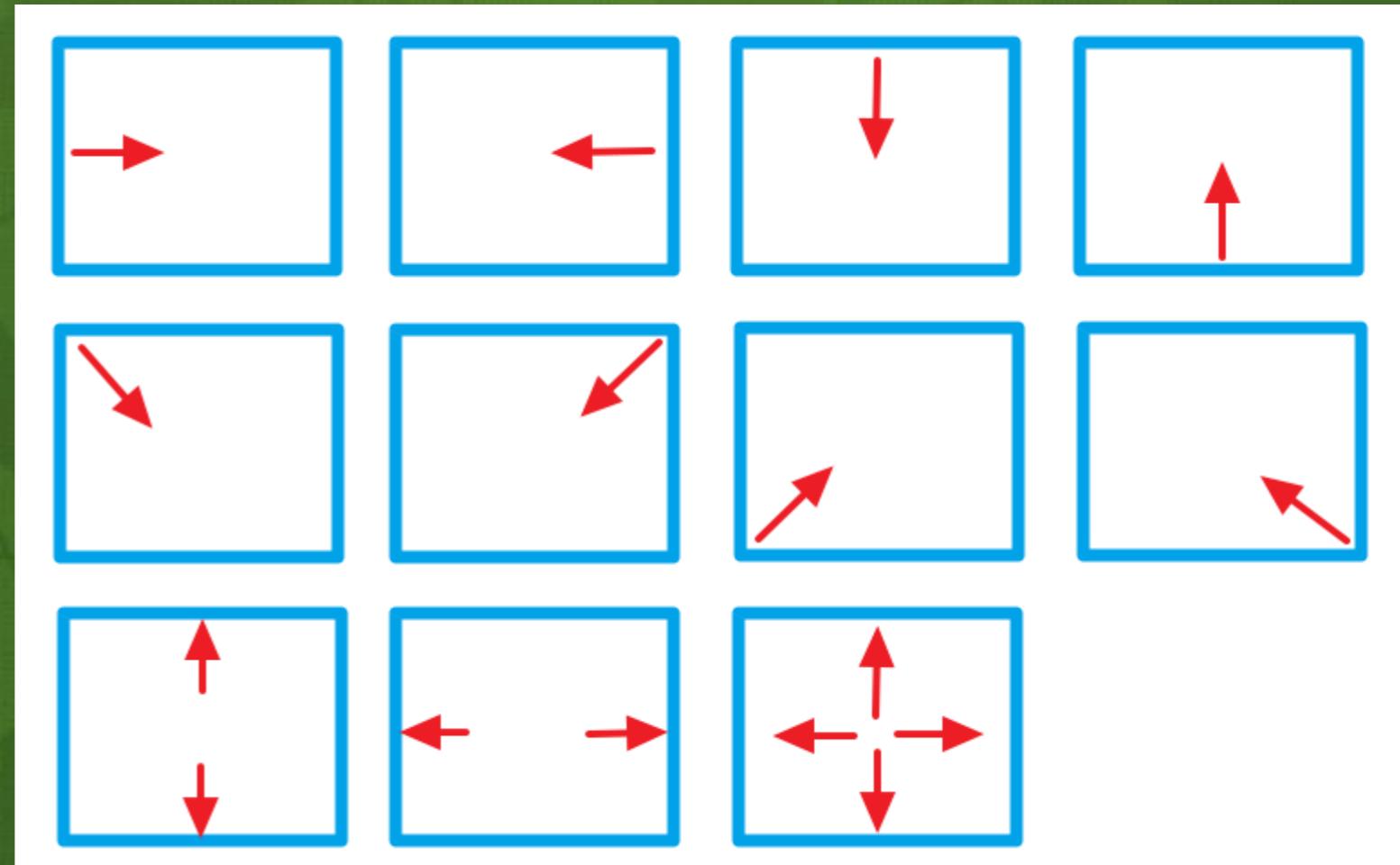
1. *Bitmap File*
2. *Layer List Drawable*
3. *Shape Drawable*
4. *State Drawable*
5. *Level List*
6. *Transition Drawable*
7. *Clip Drawable*
8. *ProgressBar và tùy biến giao diện*
9. *SeekBar và tùy biến giao diện*
10. *RatingBar và tùy biến giao diện*
11. *Xây dựng tính năng multi language*
12. *Xây dựng giao diện trên nhiều thiết bị*
13. *Sử dụng style trong định nghĩa giao diện*
14. *Tổng hợp kiến thức Drawable Resource*

Clip Drawable

Clip Drawable cung cấp khả năng cắt một drawable dựa trên level của drawable đó

Keyword:

- *clipOrientation, gravity*
- *setLevel()*
- *getLevel()*
- *10.000*



Chương 06

Drawable Resource và xây dựng giao diện

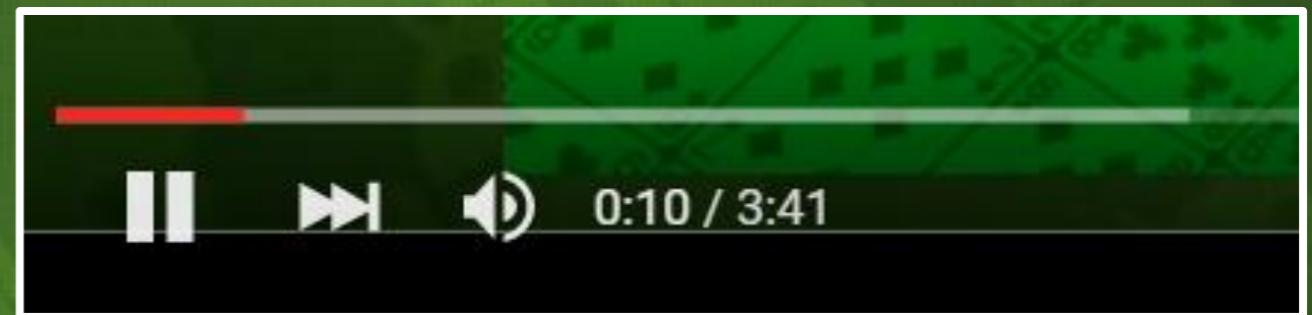
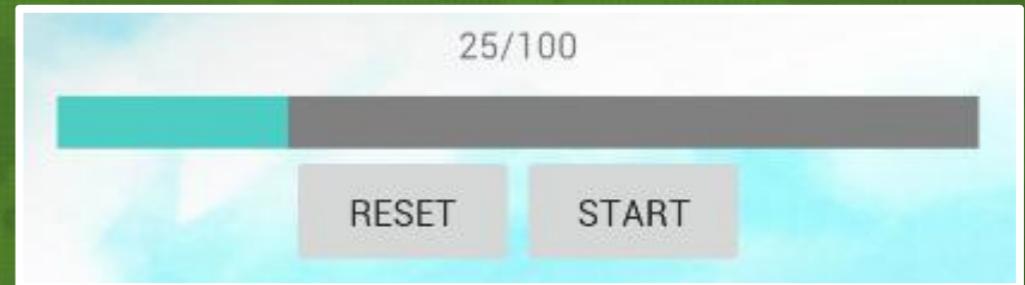
1. *Bitmap File*
2. *Layer List Drawable*
3. *Shape Drawable*
4. *State Drawable*
5. *Level List*
6. *Transition Drawable*
7. *Clip Drawable*
8. *ProgressBar và tùy biến giao diện*
9. *SeekBar và tùy biến giao diện*
10. *RatingBar và tùy biến giao diện*
11. *Xây dựng tính năng multi language*
12. *Xây dựng giao diện trên nhiều thiết bị*
13. *Sử dụng style trong định nghĩa giao diện*
14. *Tổng hợp kiến thức Drawable Resource*

ProgressBar (1)

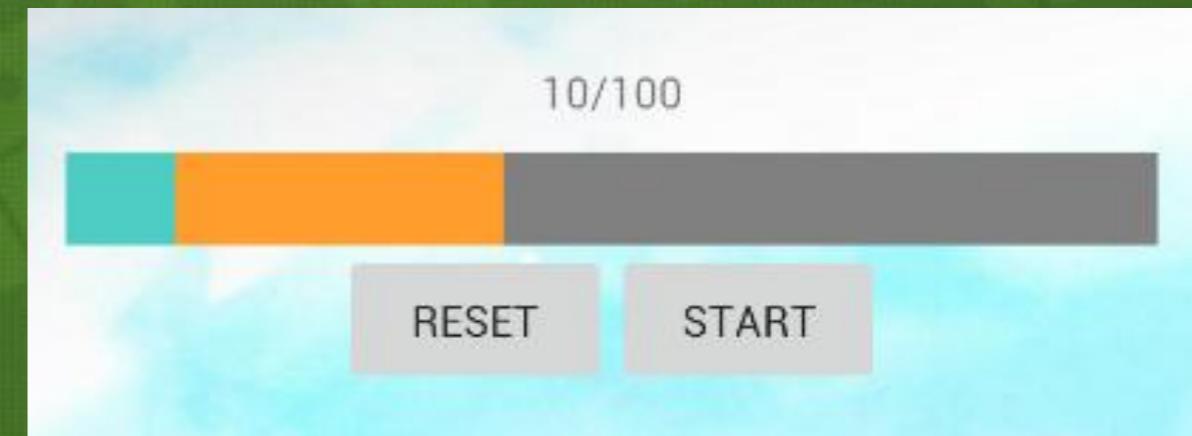
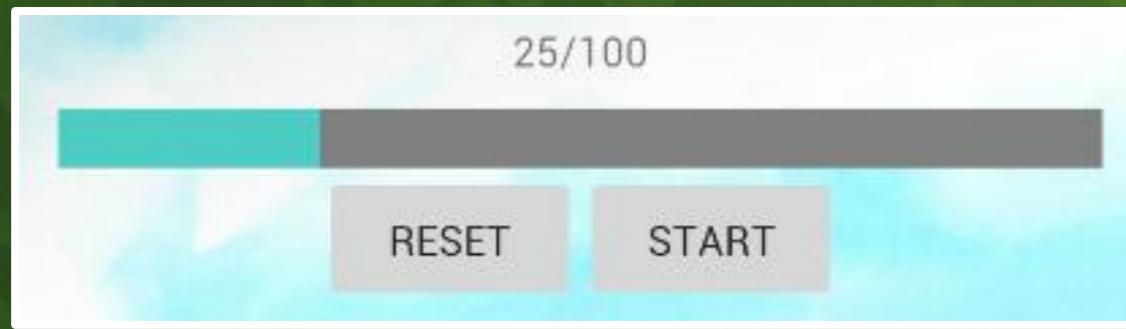
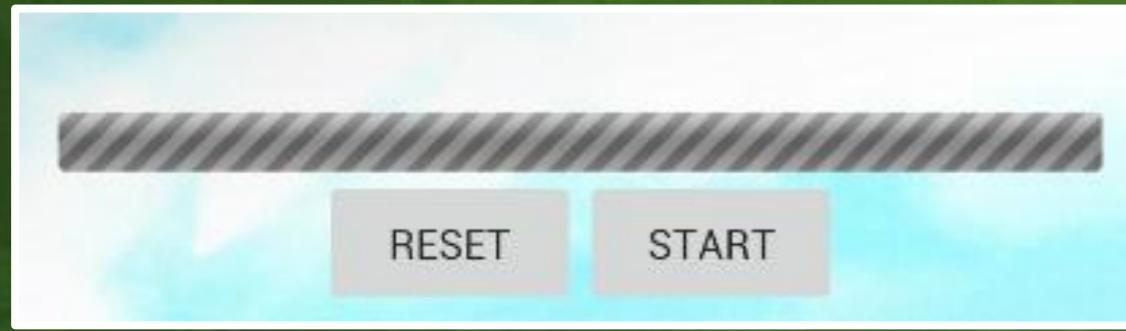
ProgressBar thường dùng để biểu diễn mức độ hoàn thành công việc của một hành động nào đó.

Keyword

- *style, customStyle, secondaryProgress, indeterminate, progress, max, ...*
- *setProgress(), setIndeterminate()*



ProgressBar (2)



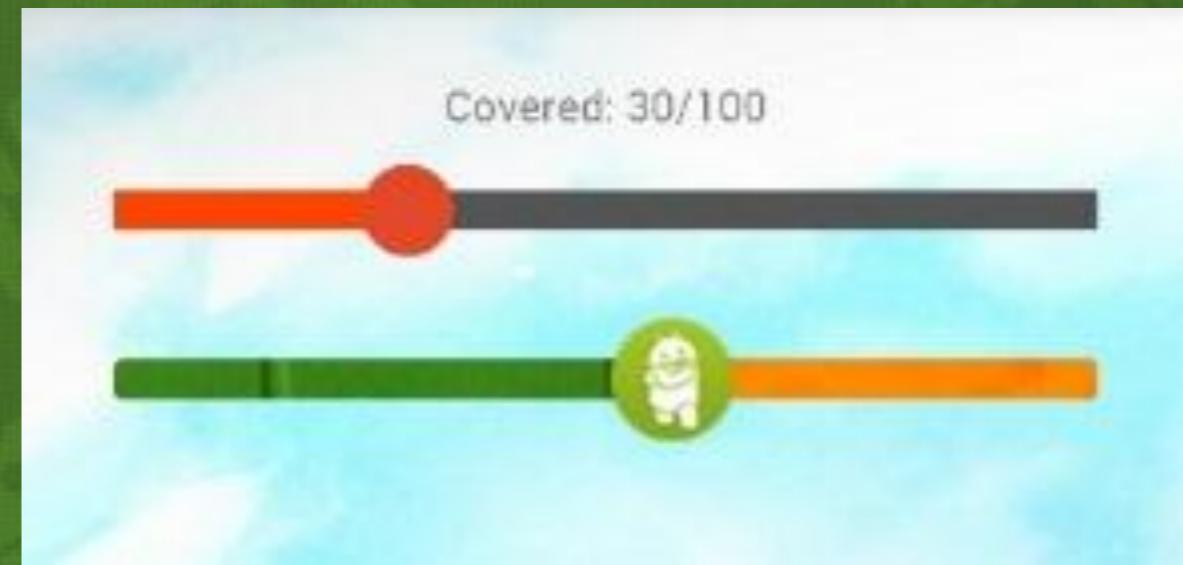
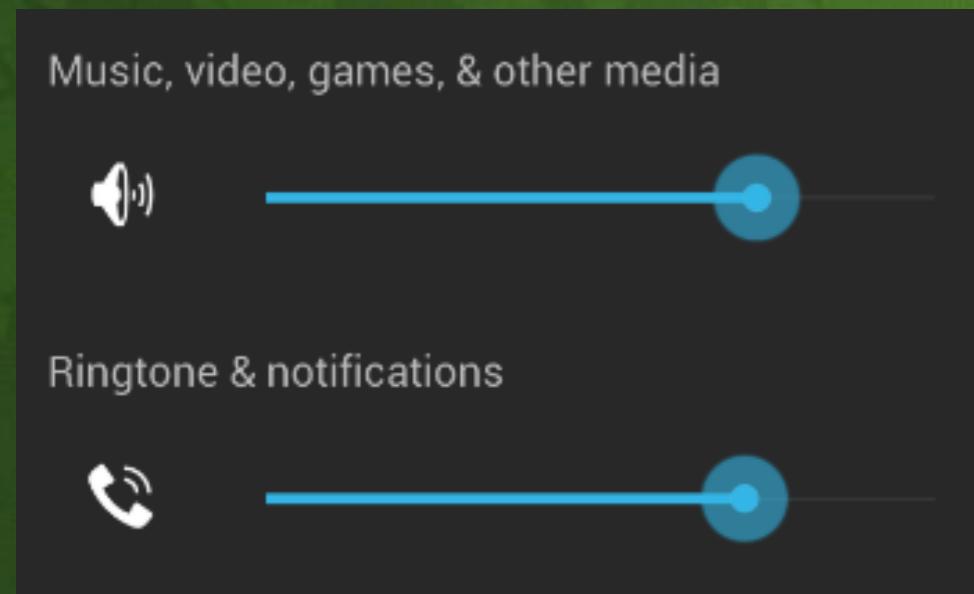
Chương 06

Drawable Resource và xây dựng giao diện

1. *Bitmap File*
2. *Layer List Drawable*
3. *Shape Drawable*
4. *State Drawable*
5. *Level List*
6. *Transition Drawable*
7. *Clip Drawable*
8. *ProgressBar và tùy biến giao diện*
9. ***SeekBar và tùy biến giao diện***
10. *RatingBar và tùy biến giao diện*
11. *Xây dựng tính năng multi language*
12. *Xây dựng giao diện trên nhiều thiết bị*
13. *Sử dụng style trong định nghĩa giao diện*
14. *Tổng hợp kiến thức Drawable Resource*

SeekBar

SeekBar là một control mở rộng của *ProgressBar* cho phép người sử dụng lựa chọn một giá trị số dựa vào giao diện kéo trượt



Chương 06

Drawable Resource và xây dựng giao diện

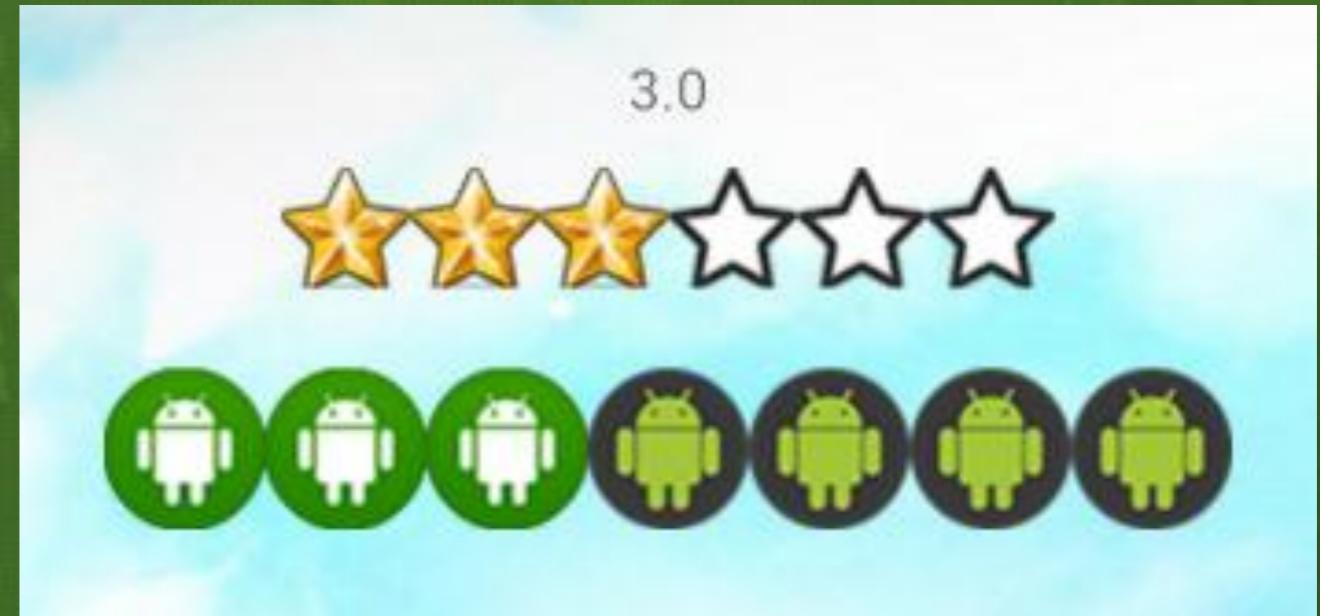
1. *Bitmap File*
2. *Layer List Drawable*
3. *Shape Drawable*
4. *State Drawable*
5. *Level List*
6. *Transition Drawable*
7. *Clip Drawable*
8. *ProgressBar và tùy biến giao diện*
9. *SeekBar và tùy biến giao diện*
- 10. *RatingBar và tùy biến giao diện***
11. *Xây dựng tính năng multi language*
12. *Xây dựng giao diện trên nhiều thiết bị*
13. *Sử dụng style trong định nghĩa giao diện*
14. *Tổng hợp kiến thức Drawable Resource*

RatingBar

RatingBar là một control mở rộng của *SeekBar* và *ProgressBar* cho phép người sử dụng lựa chọn một giá trị trong khoảng các giá trị cho trước

Keyword

- *rating, numStars*
- *setOnRatingBarChangeListener*



Chương 06

Drawable Resource và xây dựng giao diện

1. *Bitmap File*
2. *Layer List Drawable*
3. *Shape Drawable*
4. *State Drawable*
5. *Level List*
6. *Transition Drawable*
7. *Clip Drawable*
8. *ProgressBar và tùy biến giao diện*
9. *SeekBar và tùy biến giao diện*
10. *RatingBar và tùy biến giao diện*
- 11. Xây dựng tính năng multi language**
12. *Xây dựng giao diện trên nhiều thiết bị*
13. *Sử dụng style trong định nghĩa giao diện*
14. *Tổng hợp kiến thức Drawable Resource*

Xây dựng chức năng đa ngôn ngữ

Tiếng Việt

Chiều cao (cm)

Cân nặng (kg)

XEM KẾT QUẢ

Nhập chiều cao (cm) và cân nặng
của bạn (kg) để xem kết quả !"

Tiếng Anh

Height (cm)

Weight (kg)

SEE THE RESULT

Enter your height (cm) and weight
(kg) to see the result

Các dạng Android Resource cơ bản

- *Tài nguyên mặc định*: không quan tâm đến cấu hình của thiết bị hoặc không có tài nguyên để lựa chọn
- *Tài nguyên đặc trưng*: được sử dụng trên thiết bị riêng biệt thông qua các từ hạn định và đường dẫn
- *res/layout* folder chứa tài nguyên mặc định gồm các layout của ứng dụng
- *res/layout-land* folder chứa tài nguyên đặc trưng khi app đang được xem với màn hình nằm ngang
- *res/values-vi* folder chứa tài nguyên string dành riêng cho tiếng Việt

Chương 06

Drawable Resource và xây dựng giao diện

1. *Bitmap File*
2. *Layer List Drawable*
3. *Shape Drawable*
4. *State Drawable*
5. *Level List*
6. *Transition Drawable*
7. *Clip Drawable*
8. *ProgressBar và tùy biến giao diện*
9. *SeekBar và tùy biến giao diện*
10. *RatingBar và tùy biến giao diện*
11. *Xây dựng tính năng multi language*
12. *Xây dựng giao diện trên nhiều thiết bị*
13. *Sử dụng style trong định nghĩa giao diện*
14. *Tổng hợp kiến thức Drawable Resource*

Sử dụng style trong định nghĩa giao diện

The image shows a user interface for a login form. The entire form is enclosed in a yellow border. At the top left is a dark grey button labeled "Username". To its right is a light blue input field with a placeholder text "Username" in a light blue-grey color. Below this is another dark grey button labeled "Password". To its right is a light blue input field with a placeholder text "Password" in a light blue-grey color. At the bottom left is a large purple button labeled "SUBMIT". At the bottom right is another large purple button labeled "CANCEL". All text is in a white sans-serif font.

Chương 06

Drawable Resource và xây dựng giao diện

1. *Bitmap File*
2. *Layer List Drawable*
3. *Shape Drawable*
4. *State Drawable*
5. *Level List*
6. *Transition Drawable*
7. *Clip Drawable*
8. *ProgressBar và tùy biến giao diện*
9. *SeekBar và tùy biến giao diện*
10. *RatingBar và tùy biến giao diện*
11. *Xây dựng tính năng multi language*
12. *Xây dựng giao diện trên nhiều thiết bị*
13. *Sử dụng style trong định nghĩa giao diện*
- 14. Tổng hợp kiến thức Drawable Resource**

Tổng hợp kiến thức Drawable Resource (1)



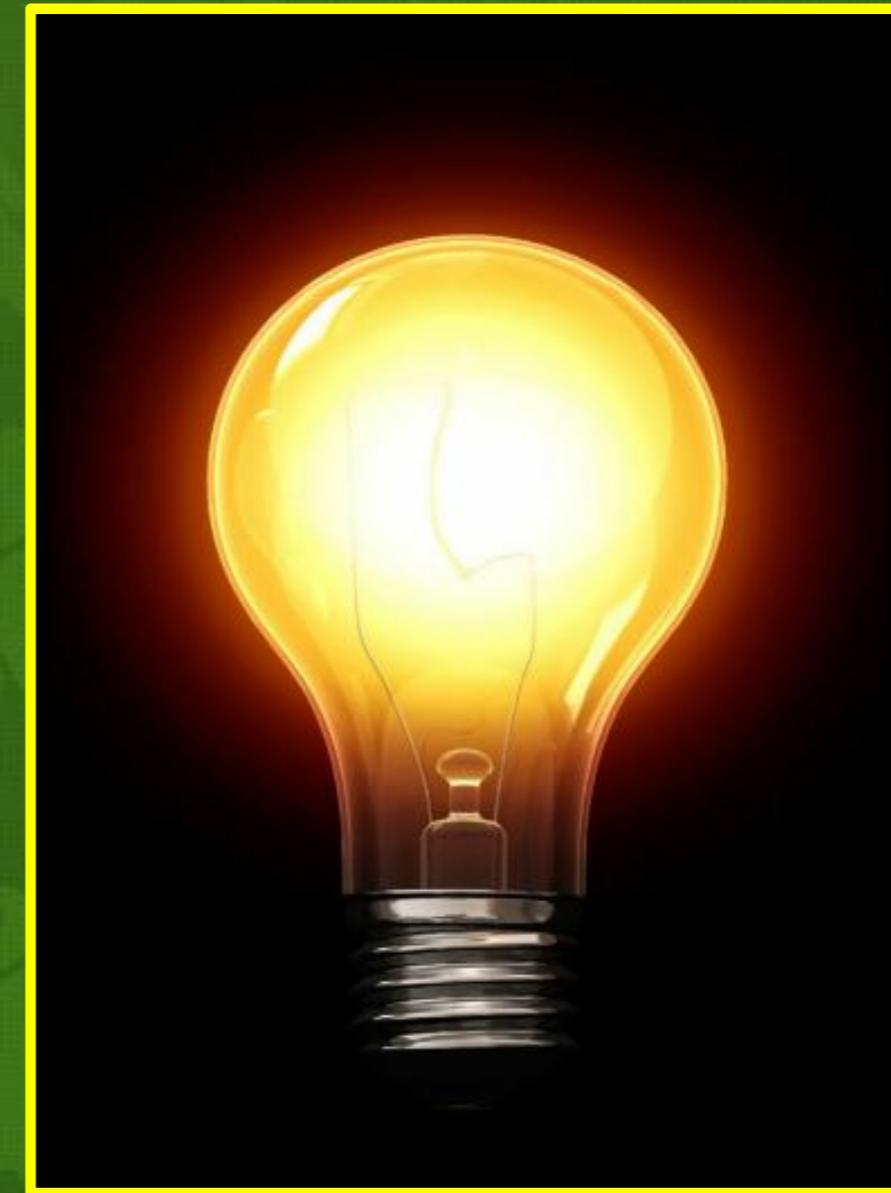


Welcom to FlashLight

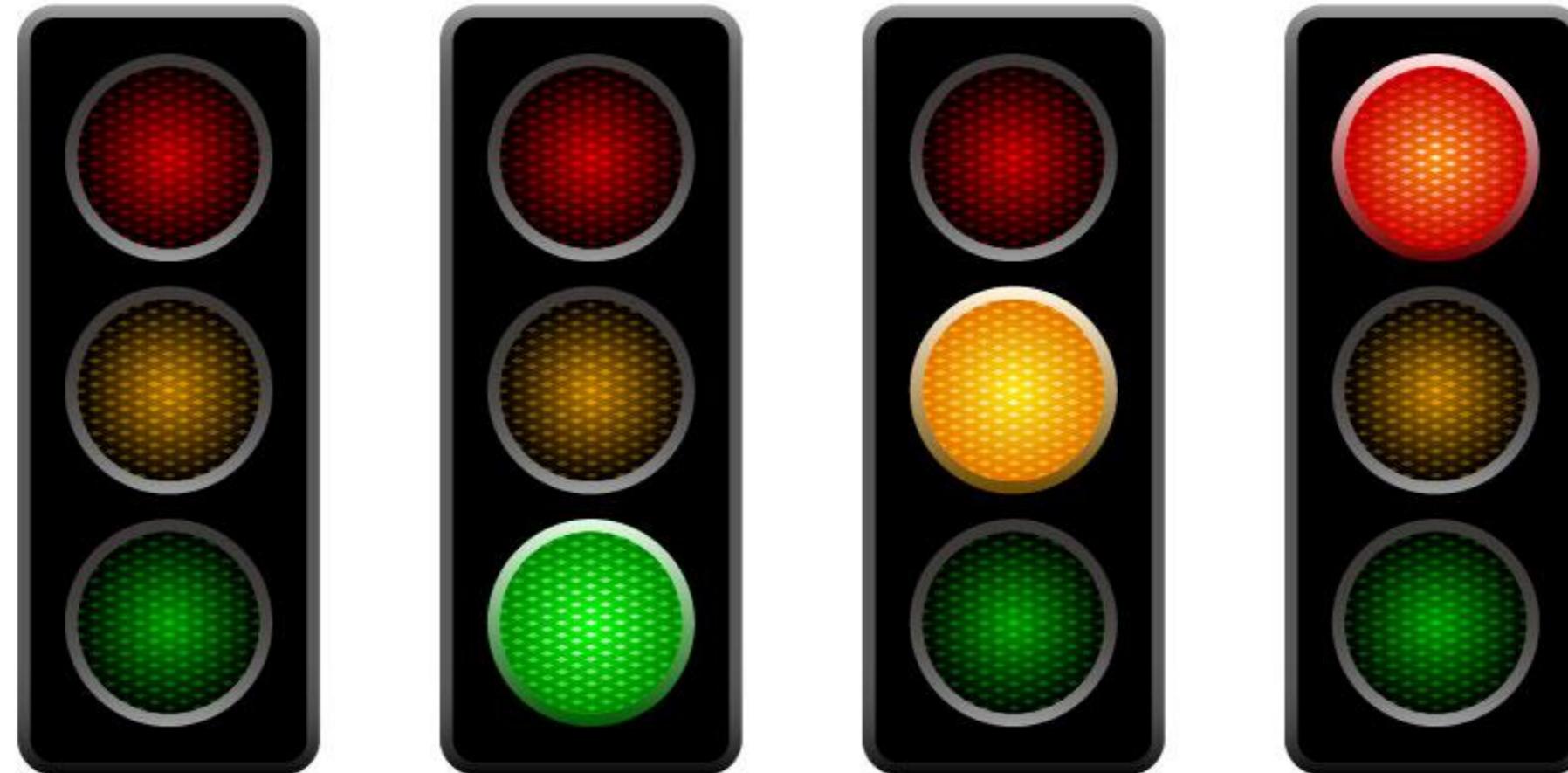
Tổng hợp kiến thức Drawable Resource (2)

- Khi ứng dụng khởi động màn hình chờ sẽ hiển thị (SplashActivity)
- Sau 2 giây thanh ProgressBar sẽ bắt đầu thay đổi, mỗi giây tăng 10%, sau khi đến 100% sẽ tự động chuyển sang MainActivity
- Ở MainActivity khi nhấn nút Back sẽ thoát khỏi ứng dụng mà không cần chuyển về SplashActivity

Tổng hợp kiến thức Drawable Resource (3)



Tổng hợp kiến thức Drawable Resource (4)



Tổng hợp kiến thức Drawable Resource (5)



Tổng hợp kiến thức Drawable Resource (6)



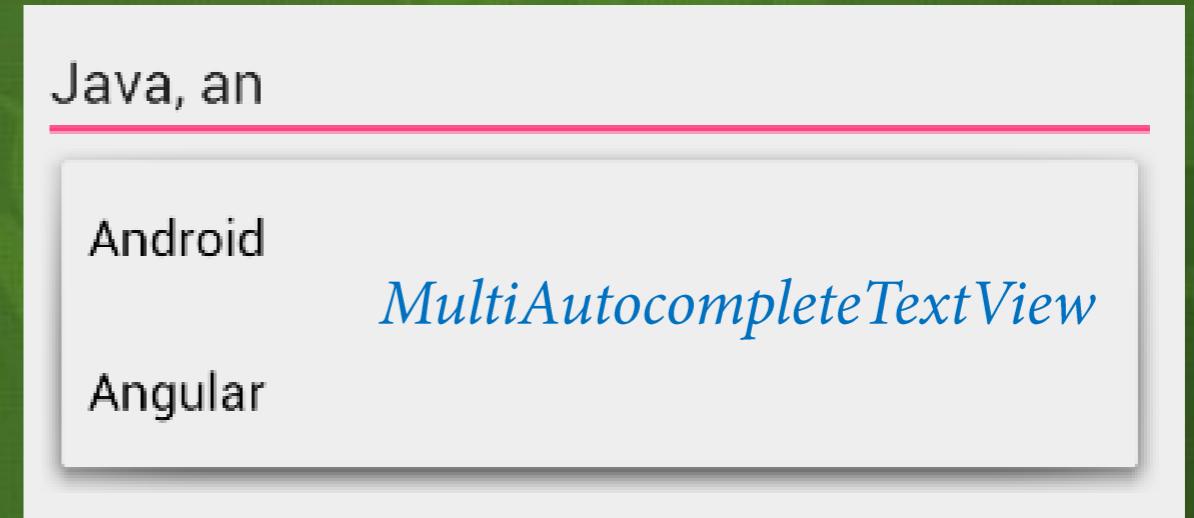
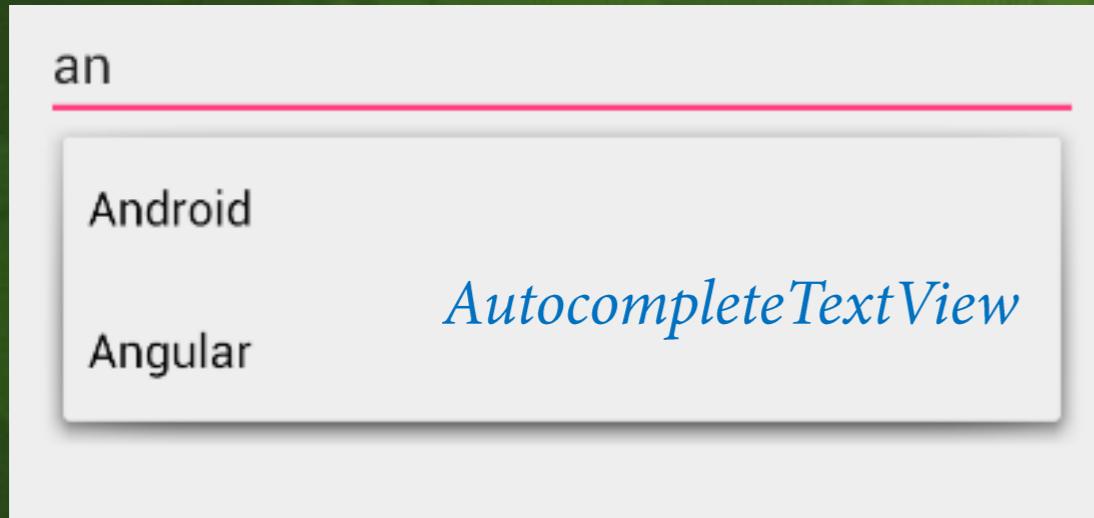
Chương 07

Android Form Widgets Advanced

Phần 01: Autocomplete - Custom Adapter - Custom Filter

Autocomplete (1)

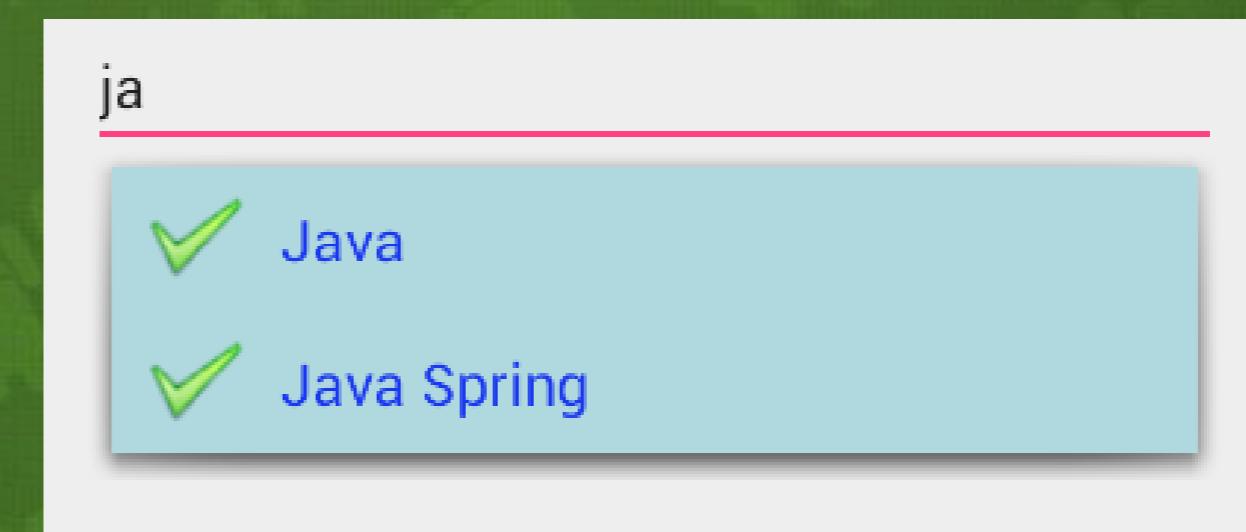
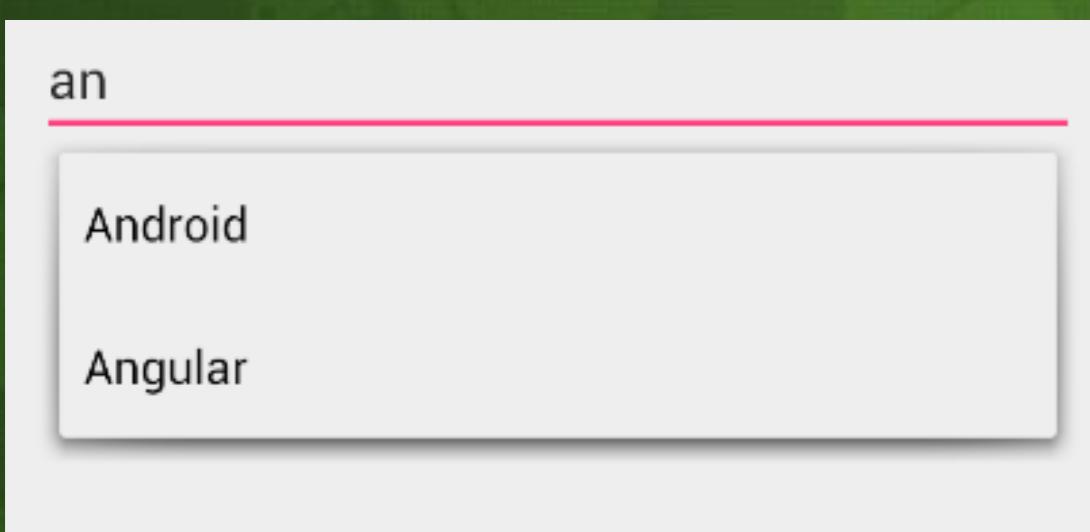
AutocompleteTextView và *MultiAutocompleteTextView* hỗ trợ nhập dữ liệu nhanh cho người sử dụng



Keyword: *completionThreshold*, *setThreshold*, *ArrayAdapter*, *setAdapter*, *setTokenizer*, *CommaTokenizer*

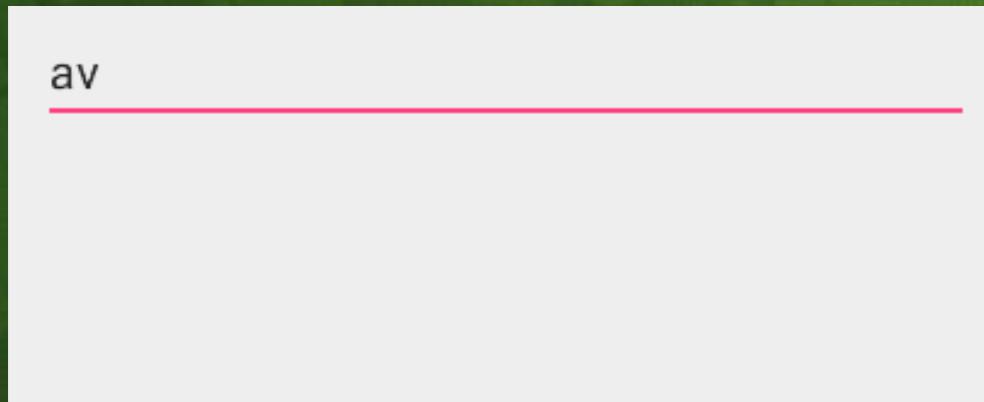
Autocomplete (2)

Custom giao diện cho Autocomplete

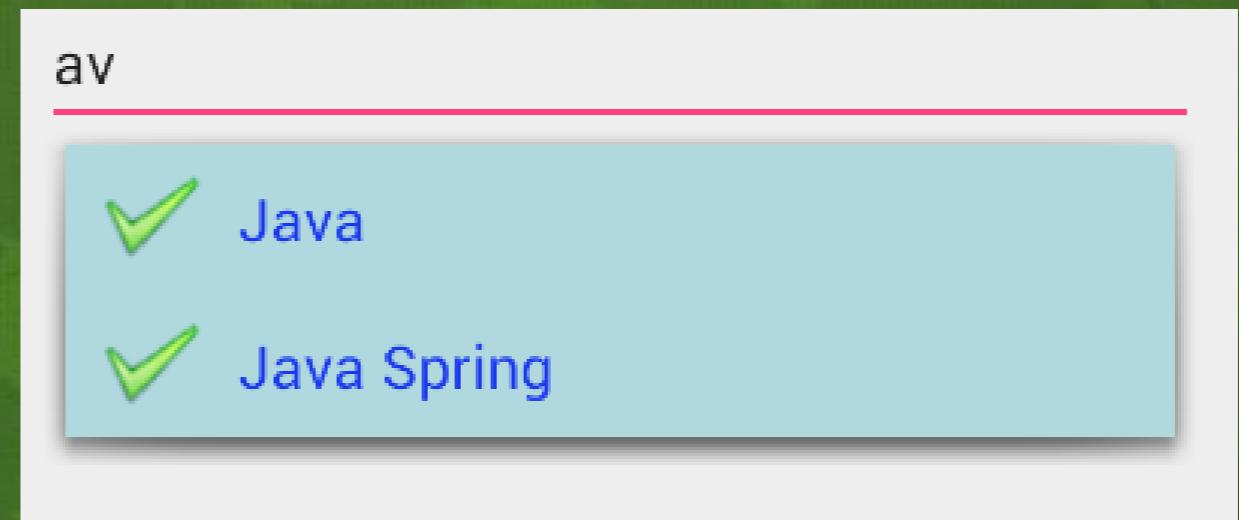


Autocomplete (3)

Custom filter cho Autocomplete



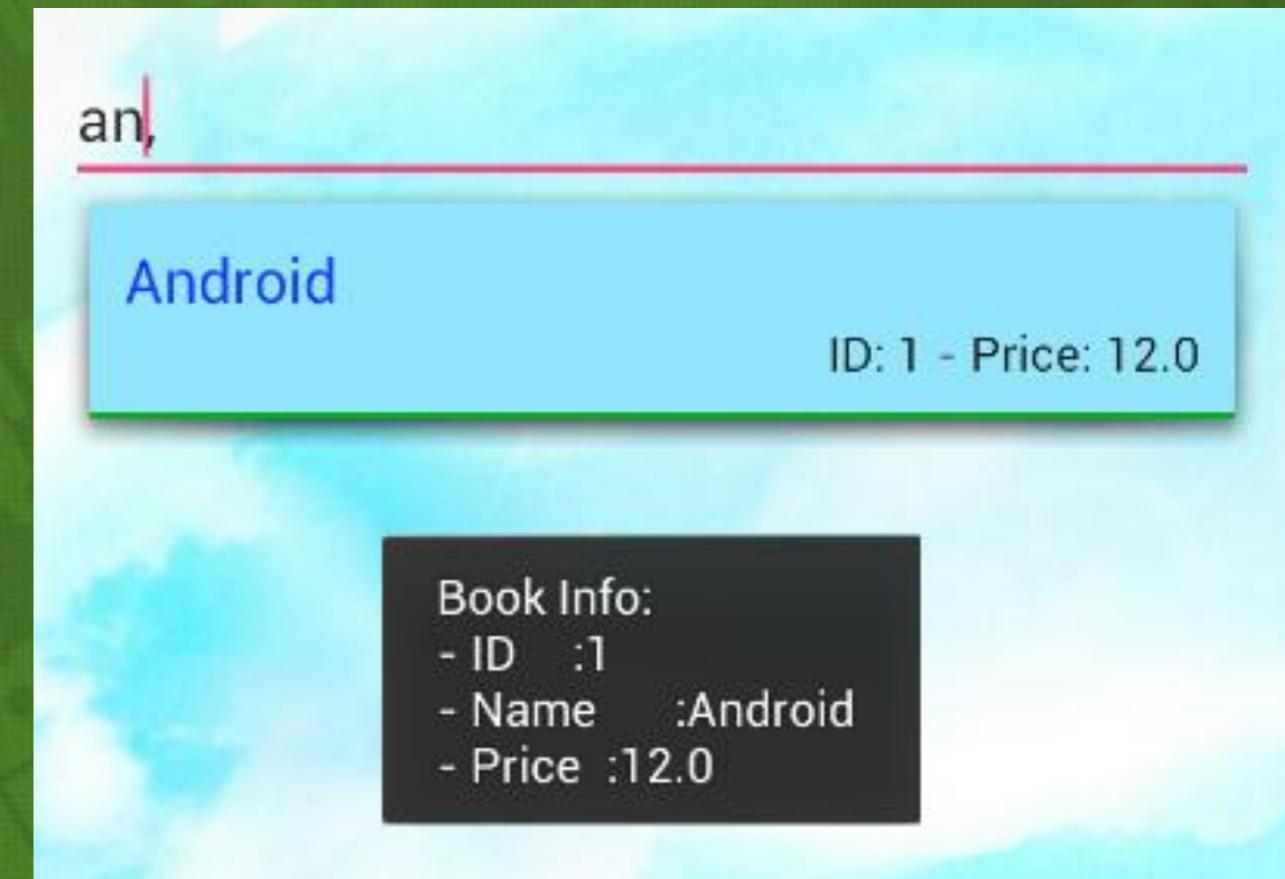
Android, Java, IOS, Java Spring



Android, Java, IOS, Java Spring

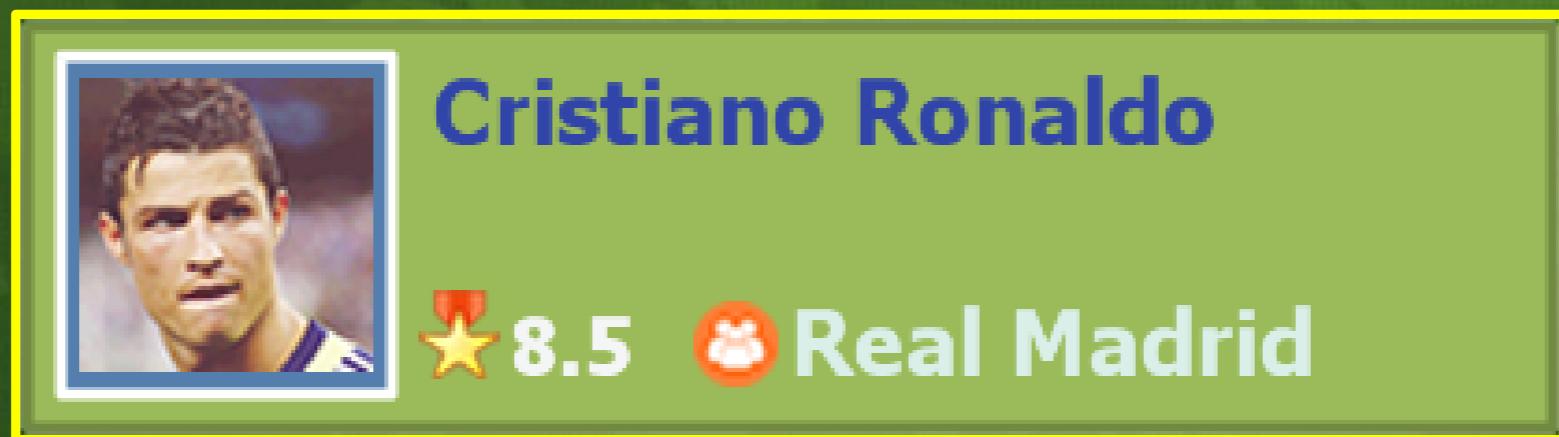
Autocomplete (4)

Phân tử của Autocomplete là một Object



Exercise Autocomplete (1)

- Mỗi phần tử trong Autocomplete là một đối tượng Player: tên cầu thủ, điểm đánh giá, hình ảnh cầu thủ và câu lạc bộ mà cầu thủ này đang thi đấu
→ Class Player (*name, score, avatar, club*)
- Custom giao diện cho Autocomplete tương đương với hình vẽ sau đây



Exercise Autocomplete (2)

- Sau khi người dùng nhập vào 2 ký tự sẽ bắt đầu việc tìm kiếm và hiển thị các kết quả tương đồng
- Yêu cầu filter dữ liệu theo tên cầu thủ và tên câu lạc bộ mà cầu thủ này đang thi đấu, ví dụ nhập **on** → sẽ tìm xem cầu thủ nào có tên hoặc tên câu lạc bộ có chứa **on** sẽ hiển thị



Cristiano Ronaldo

⭐ 8.5 ⚽ Real Madrid

Chương 07

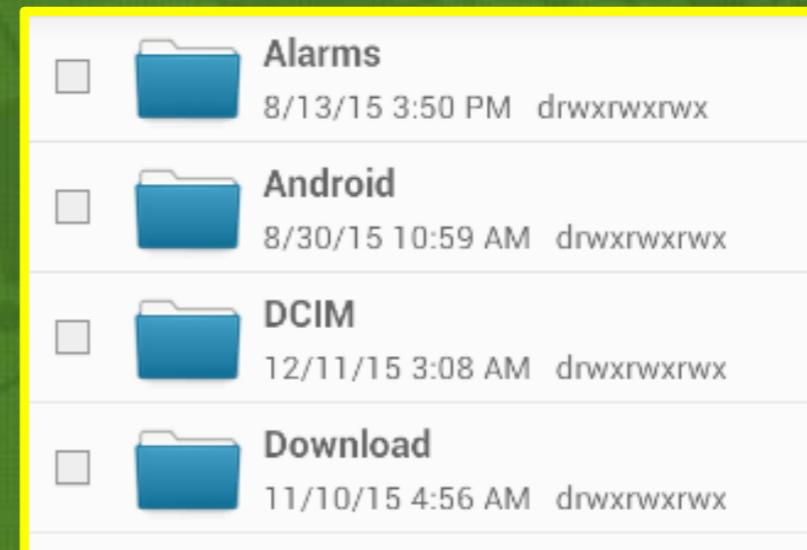
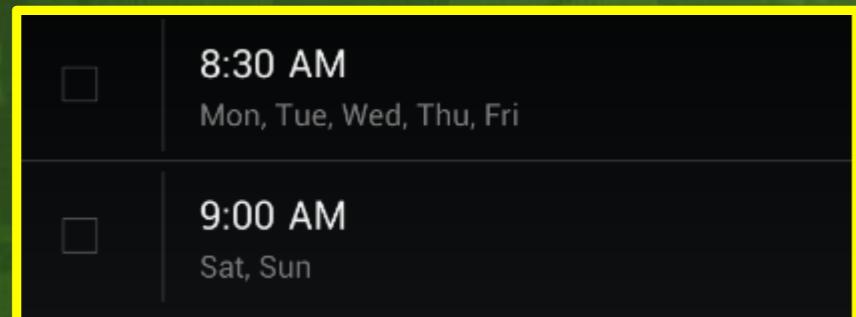
Android Form Widgets Advanced

Phần 02: Sử dụng ListView cơ bản đến nâng cao

Sử dụng ListView (1)

ListView thường dùng để hiển thị danh sách các phần tử được sắp xếp theo chiều dọc (mỗi phần tử có thể chứa nhiều widgets khác như *TextView*, *Button*, *ImageView*, ...)

Các phần tử trong *ListView* được quản lý thông qua *Adapter*, Adapter xem như thành phần trung gian để đổ dữ liệu vào *ListView*, có các loại adapter như: *BaseAdapter*, *ArrayAdapter*, *CursorAdapter* (cơ sở dữ liệu)



Sử dụng ListView (2)

- *Listview* *đỗ dữ liệu* từ *String*
- *Listview onClick* – *Listview onLongClick*
- *Listview* *đỗ dữ liệu* từ *XML*
- *Listview* *đỗ dữ liệu* từ *ArrayList*
- *Listview* *đỗ dữ liệu* từ *ArrayList* và *mỗi dòng là một đối tượng*
- *Listview CustomLayout*

String Array

Thư mục tài nguyên /res/values/filename.xml

Cú pháp khai báo:

```
<resources>
    <string-array name="array_name">
        <item>text_string</item>
        ...
    </string-array>
</resources>
```

Tham chiếu tài nguyên: getResources().getStringArray(R.array.array_name);

Sử dụng ListView (3)

Normal

- Data source
- MainActivity extends **Activity**
- list = (ListView)
`findViewById(R.id myList)`
- onItemClickListern

ListActivity

- Data source
- MainActivity extends **ListActivity**
- list = `getListView();`
- `onListItemClick`

Sử dụng ListView (3)

Android

Window Phone

Java

iOS

Swift

Swift



Android



Window Phone



Java



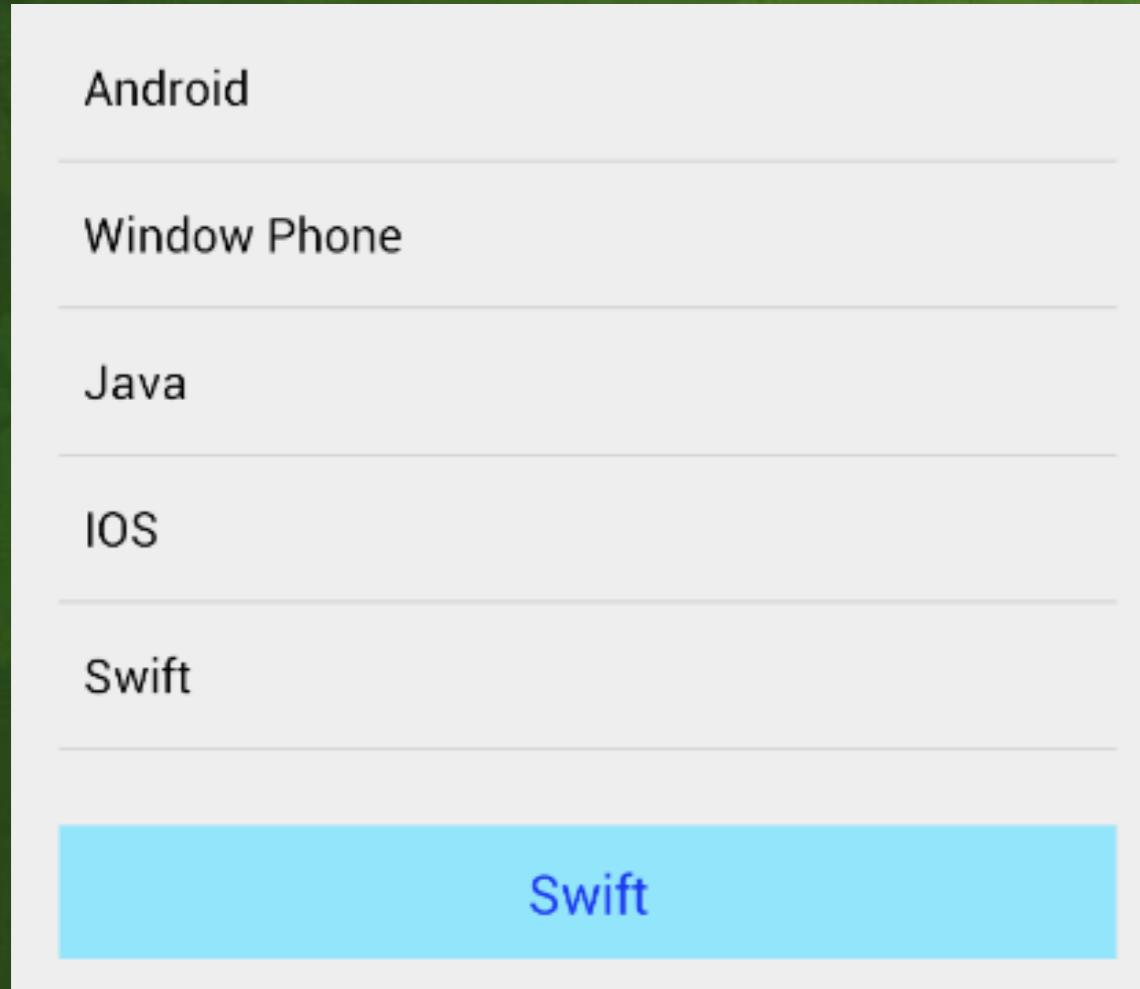
iOS



Swift

Swift

Sử dụng ListView (4)



Sử dụng ListView (5)

Tối ưu ListView với ViewHolder: giảm thiểu lại số lần gọi findViewById không cần thiết, giúp nó chạy mượt mà hơn

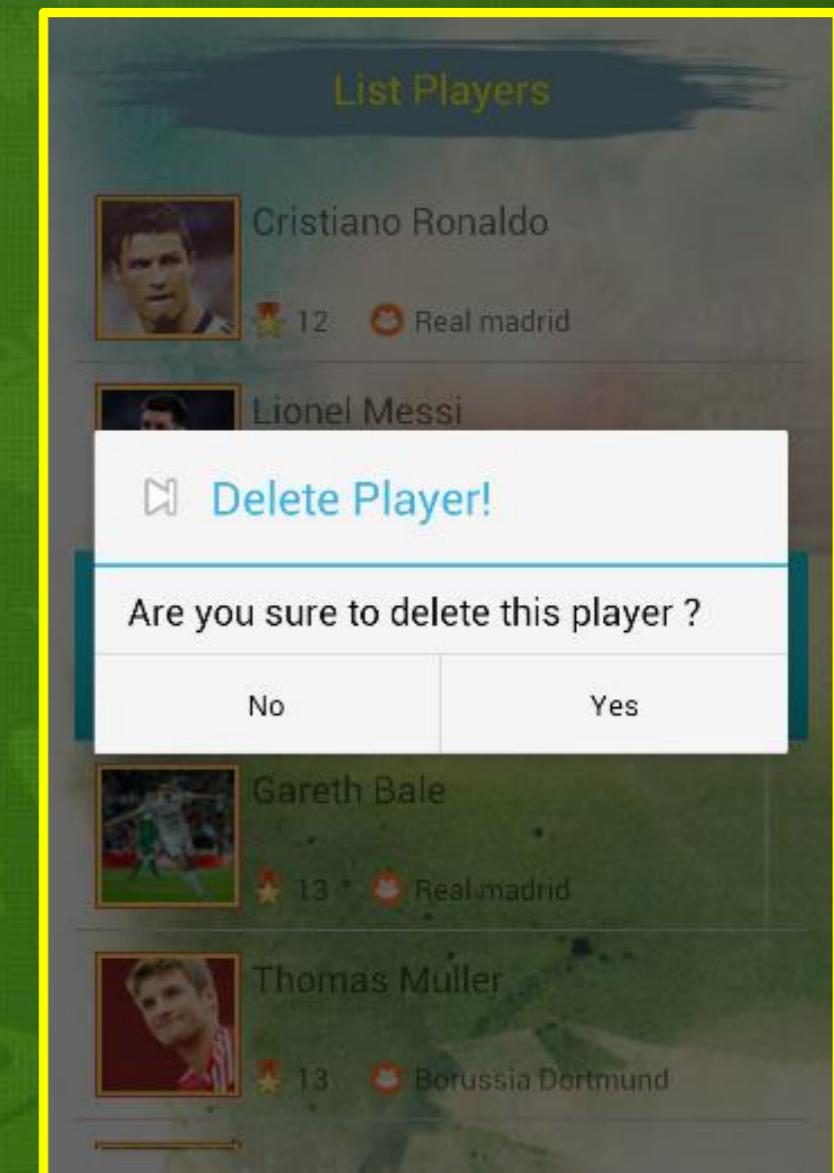
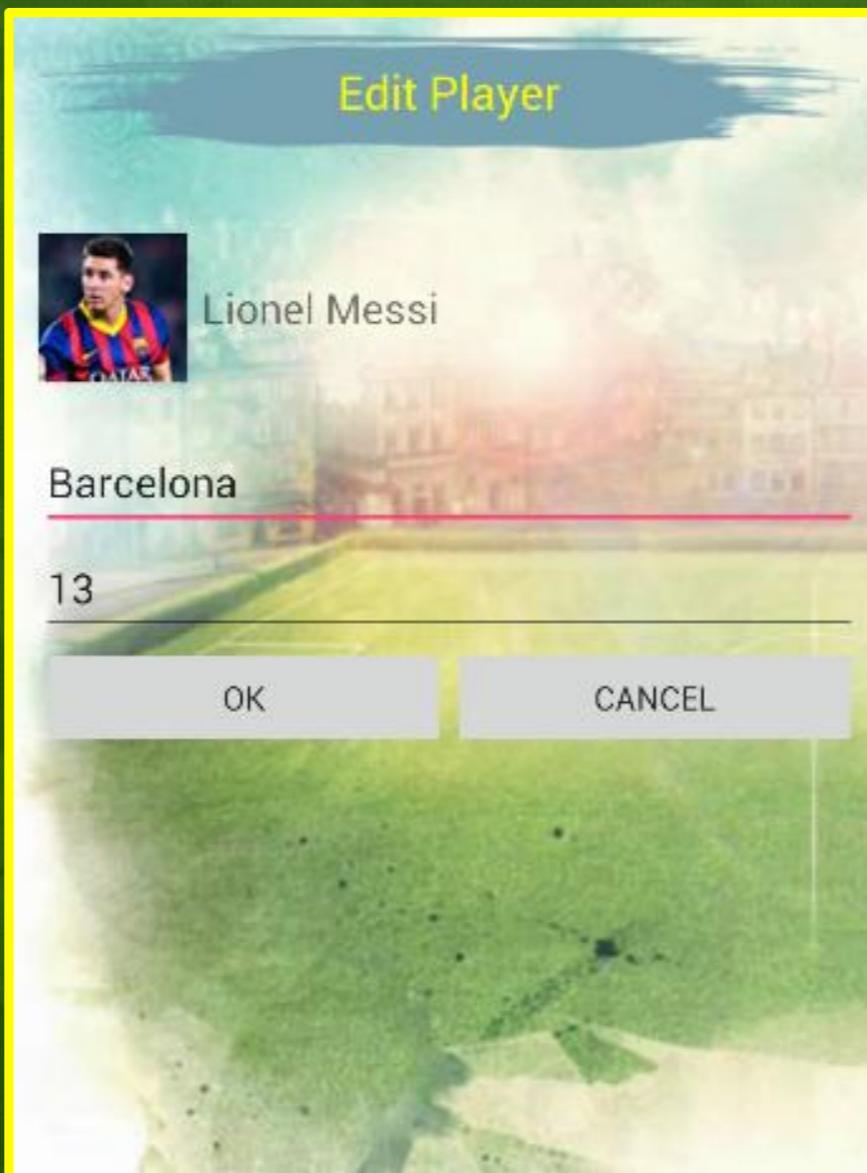
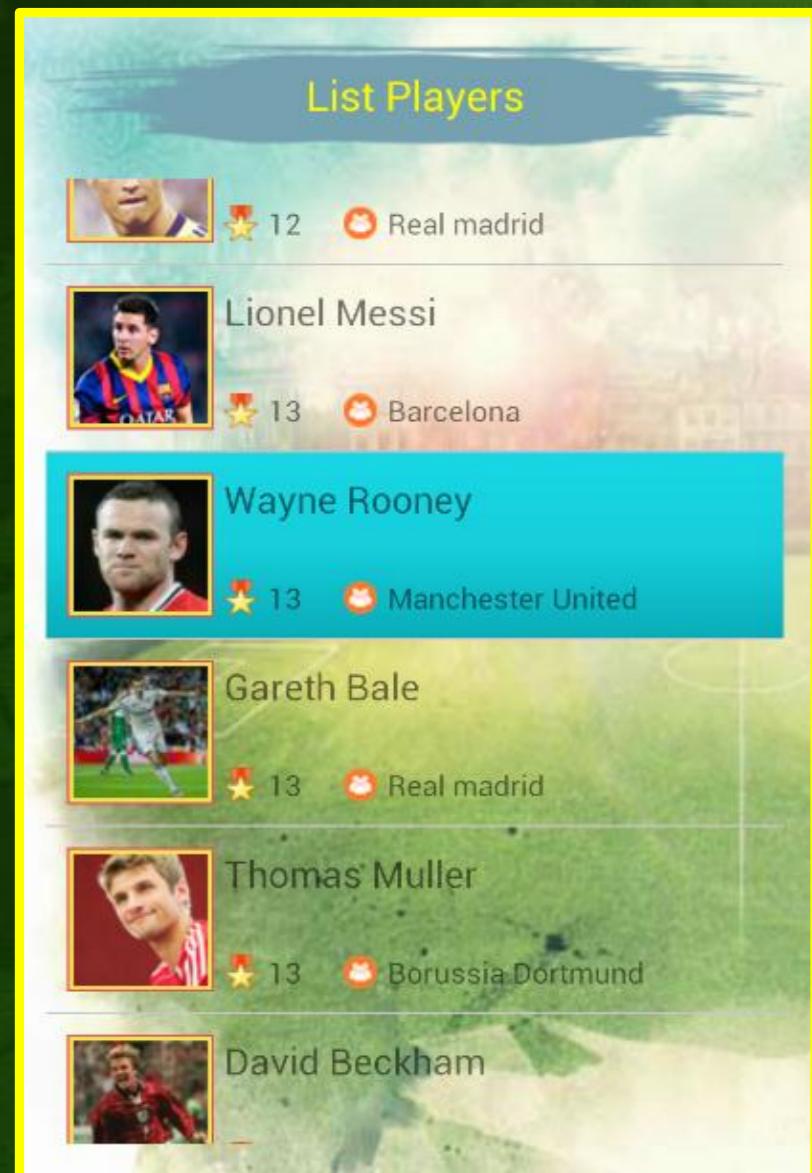
Keyword: ViewHolder, setTag, getTag, ...

Chương 07

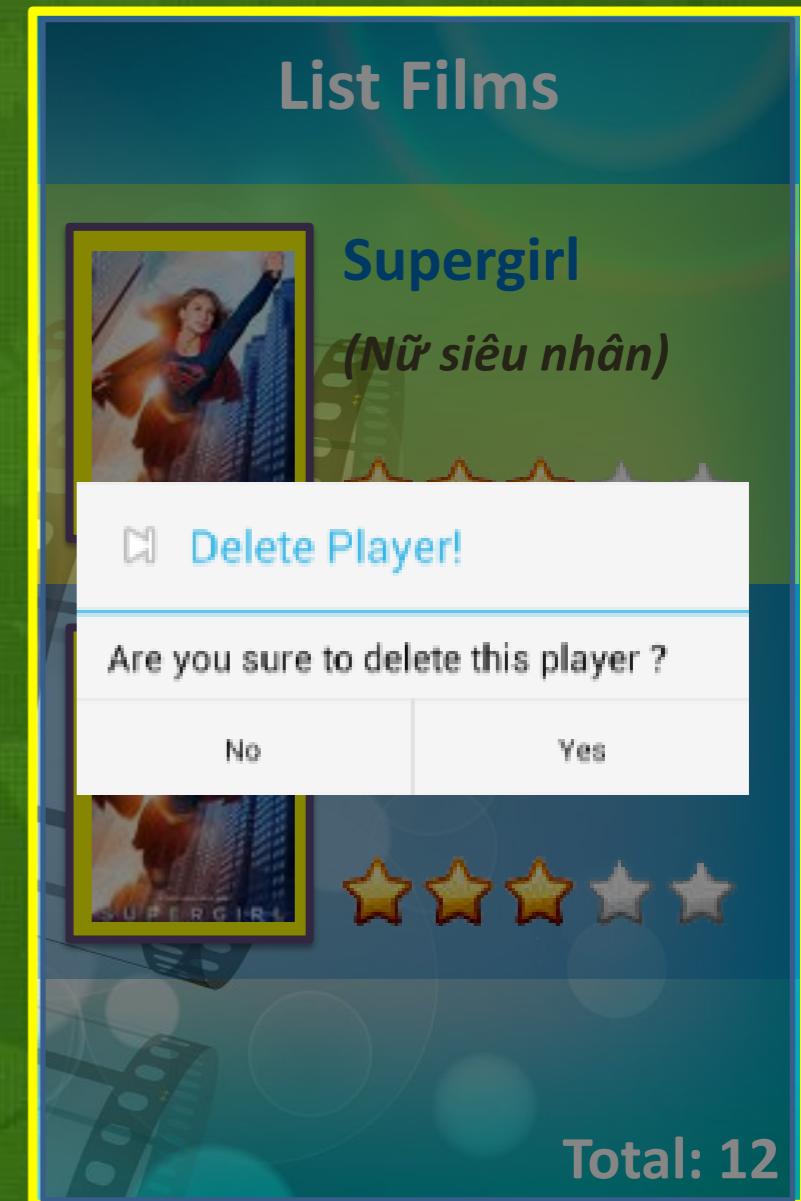
Android Form Widgets Advanced

Phần 03: Xây dựng Ứng dụng quản lý cầu thủ

Xây dựng App List Players



Xây dựng App List Films



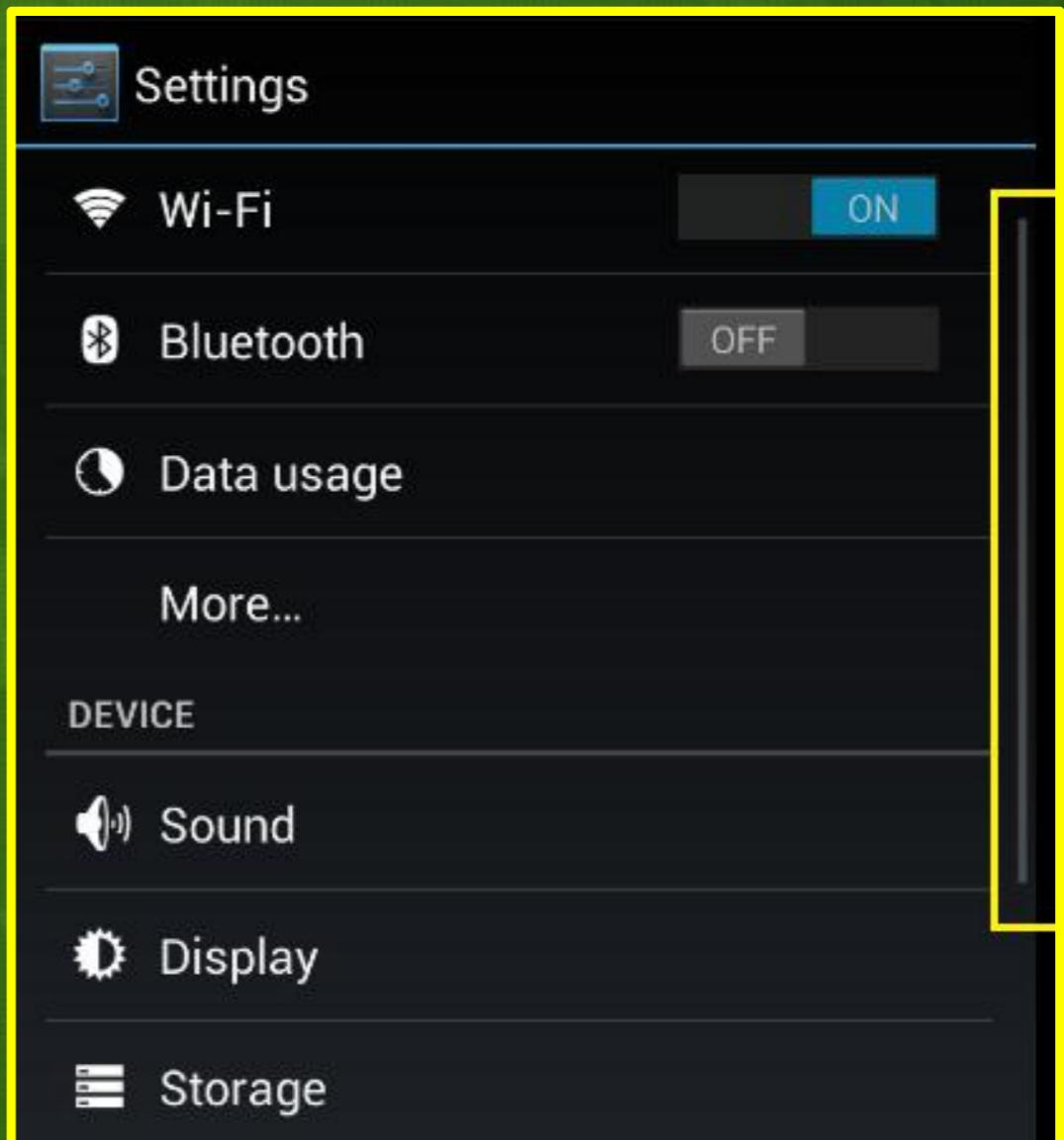
Chương 07

Android Form Widgets Advanced

Phần 04: Sử dụng ScrollView

ScrollView

- *ScrollView* thường dùng để hiển thị các nội dung chiếm quá diện tích cho phép, bên trong *ScrollView* chỉ chứa một thành phần control duy nhất.
- Scroll theo chiều ngang: *HorizontalScrollView*,
- Scroll theo chiều dọc *ScrollView*
- Keyword: *fillViewPort*, *getViewTreeObserver*, *addOnScrollChangedListener*, *fullScroll*



Chương 07

Android Form Widgets Advanced

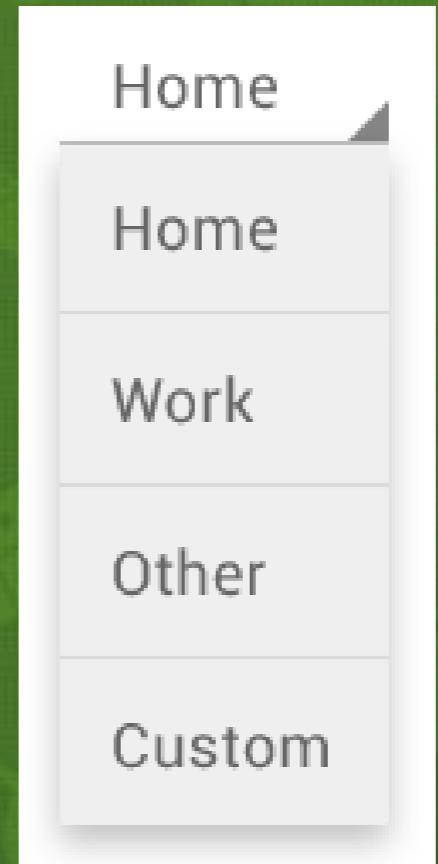
Phần 05: Sử dụng Spinner và tùy biến giao diện

Sử dụng Spinner (1)

Spinner cung cấp khả năng chọn một giá trị nào đó trong tập hợp giá trị cho trước

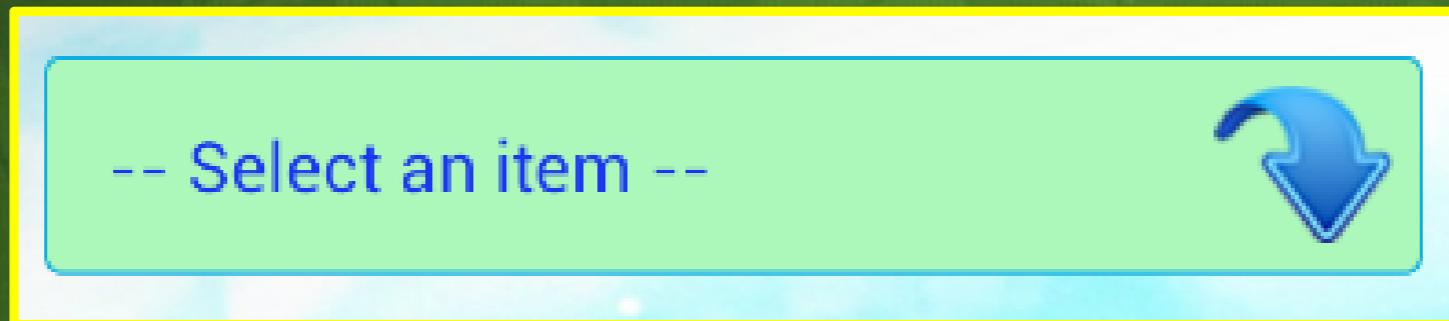
Keyword: *Android Spinner*

- Đổ dữ liệu vào *Spinner* với *String Array*
- Thiết lập thuộc tính: *prompt*, *spinnerMode*
- Phương thức: *setSelection*
- *OnItemSelectedListener*



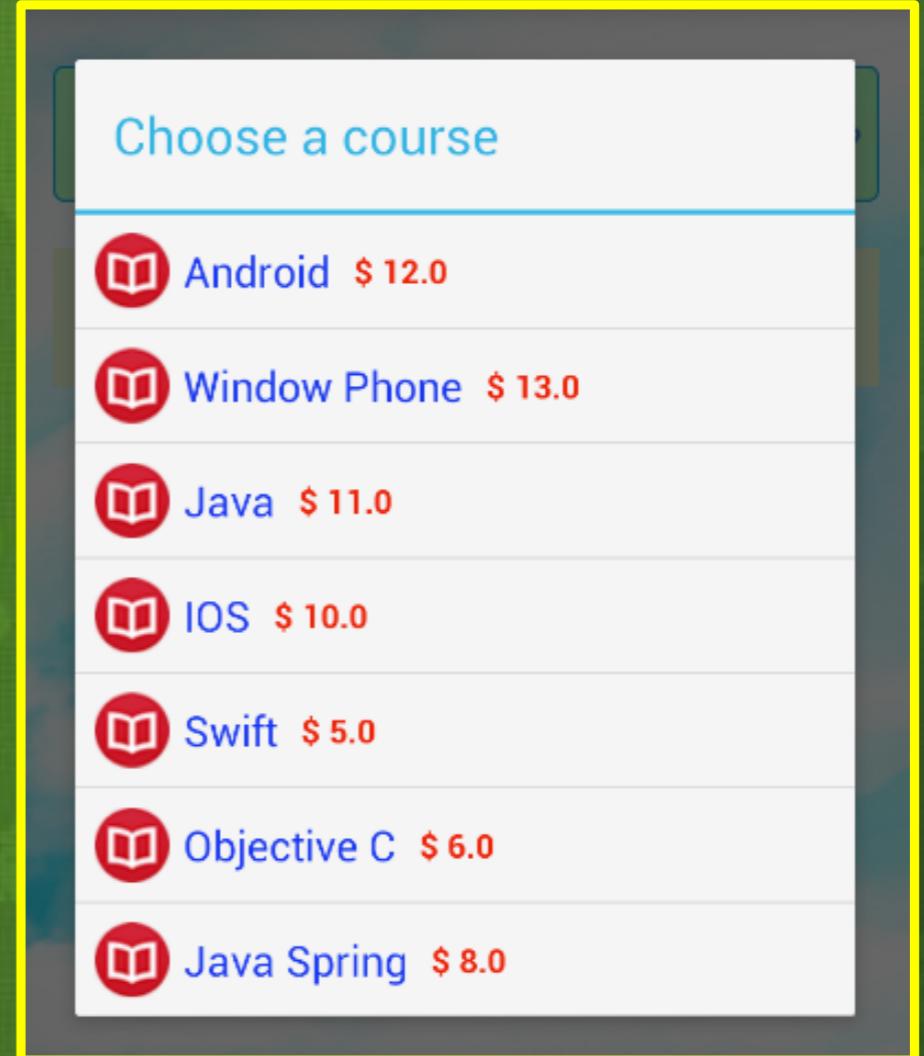
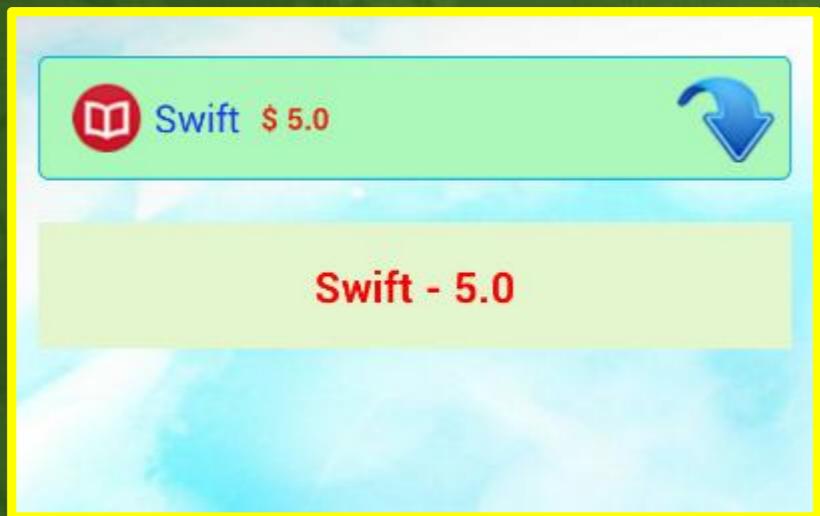
Sử dụng Spinner (2)

- Đổ dữ liệu vào Spinner với ArrayList String
- Xây dựng background cho Spinner



Sử dụng Spinner (3)

- Đổ dữ liệu vào Spinner với ArrayList Book
- Xây dựng SpinnerCustomAdapter để tùy chỉnh giao diện cho Spinner

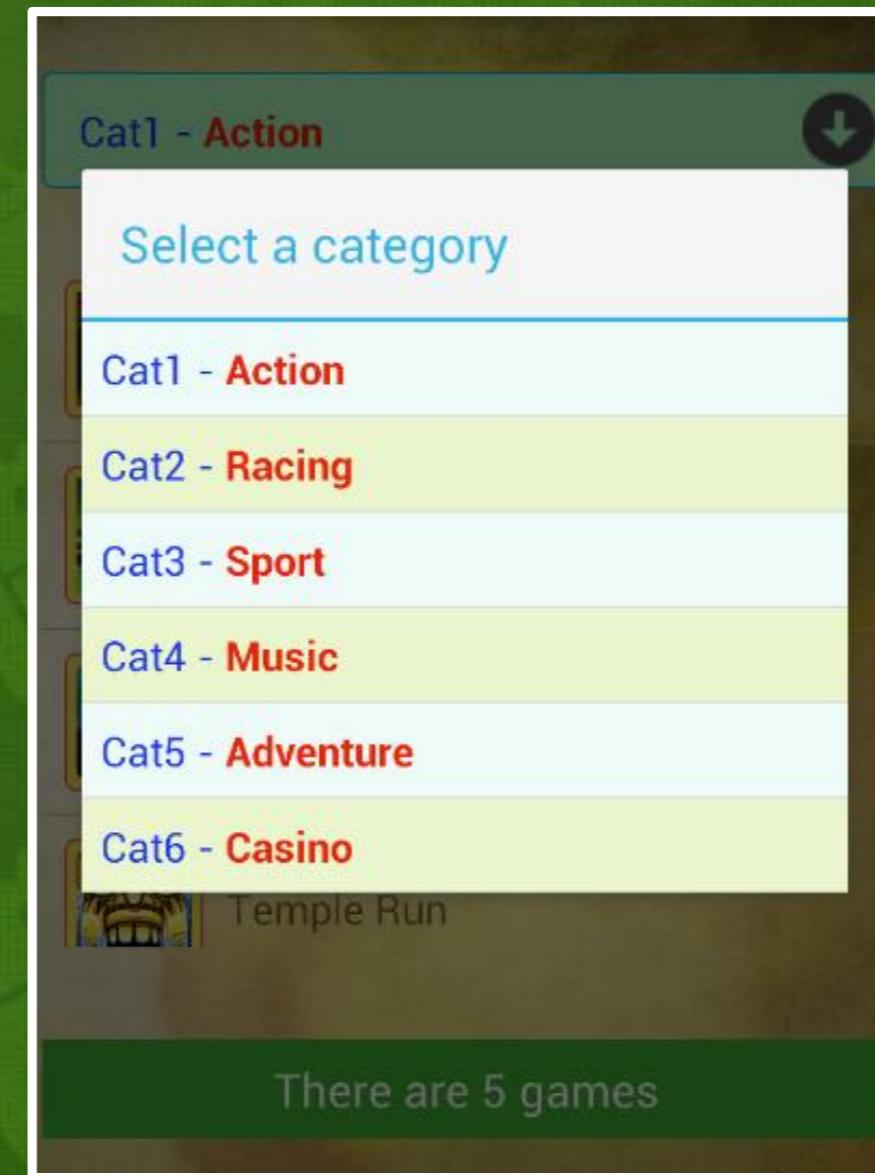
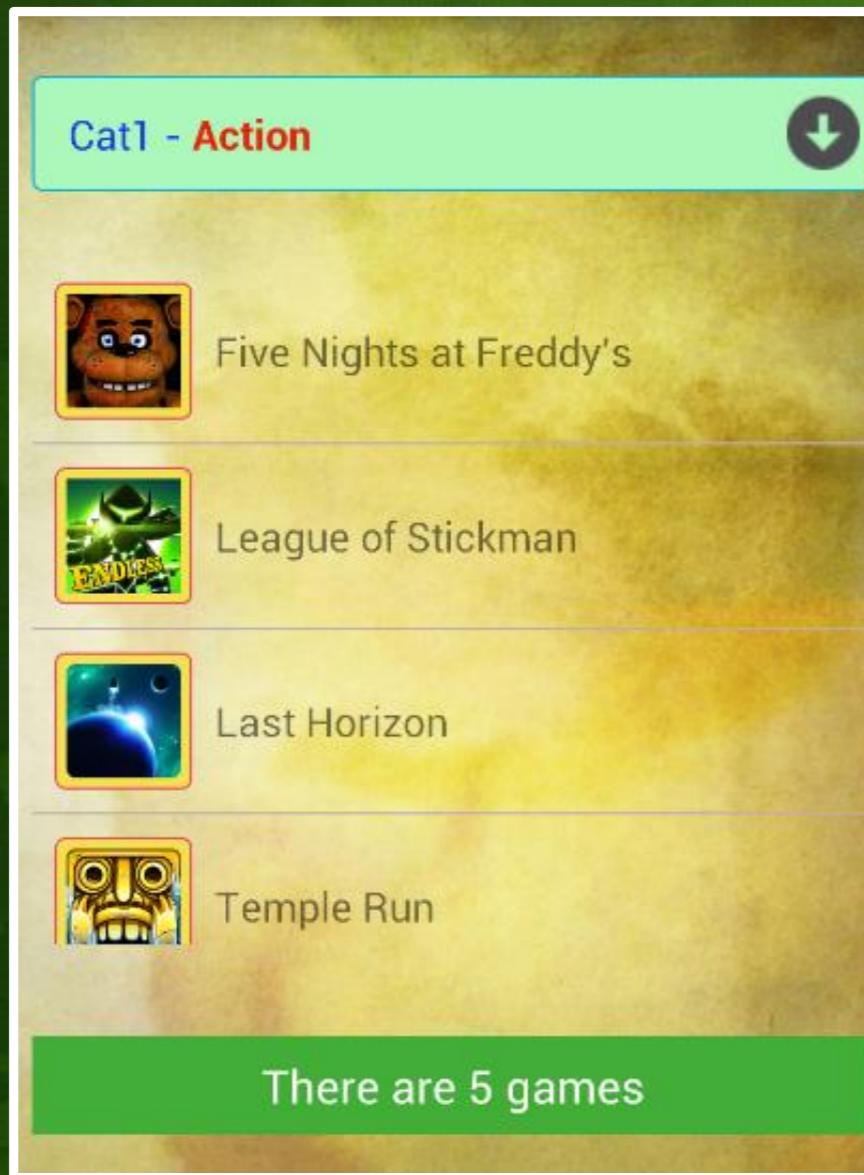


Chương 07

Android Form Widgets Advanced

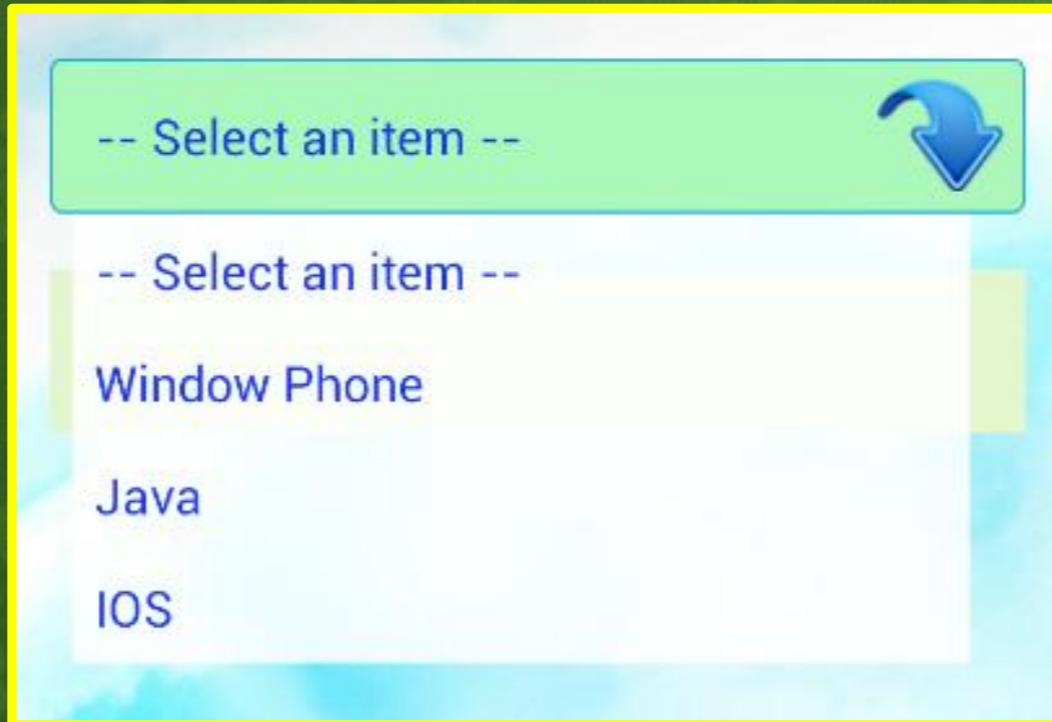
Phần 06: Xây dựng ứng dụng kết hợp Spinner và ListView

Sử dụng Spinner (4) - Kết hợp Spinner và ListView



Sử dụng Spinner (5)

Sử dụng spinnerMode=“dropdown” và thêm giá trị miêu tả “Select an item”, “Select a category”, “Select a book”



Chương 07

Android Form Widgets Advanced

Phần 07: GridView xây dựng ứng dụng My Keyboard

Sử dụng GridView (1)

GridView thường dùng để hiển thị danh sách các phần tử và danh sách này được bố trí ở dạng lưới (phân bố chiều dọc và chiều ngang)

Keyword: `columnWidth`, `horizontalSpacing`,
`verticalSpacing`, `numColumns`,



Exercise GridView (1)

Bổ sung TextView hiển thị một chuỗi ngẫu nhiên có chiều dài 6 ký tự nằm trong tập hợp [a-y]

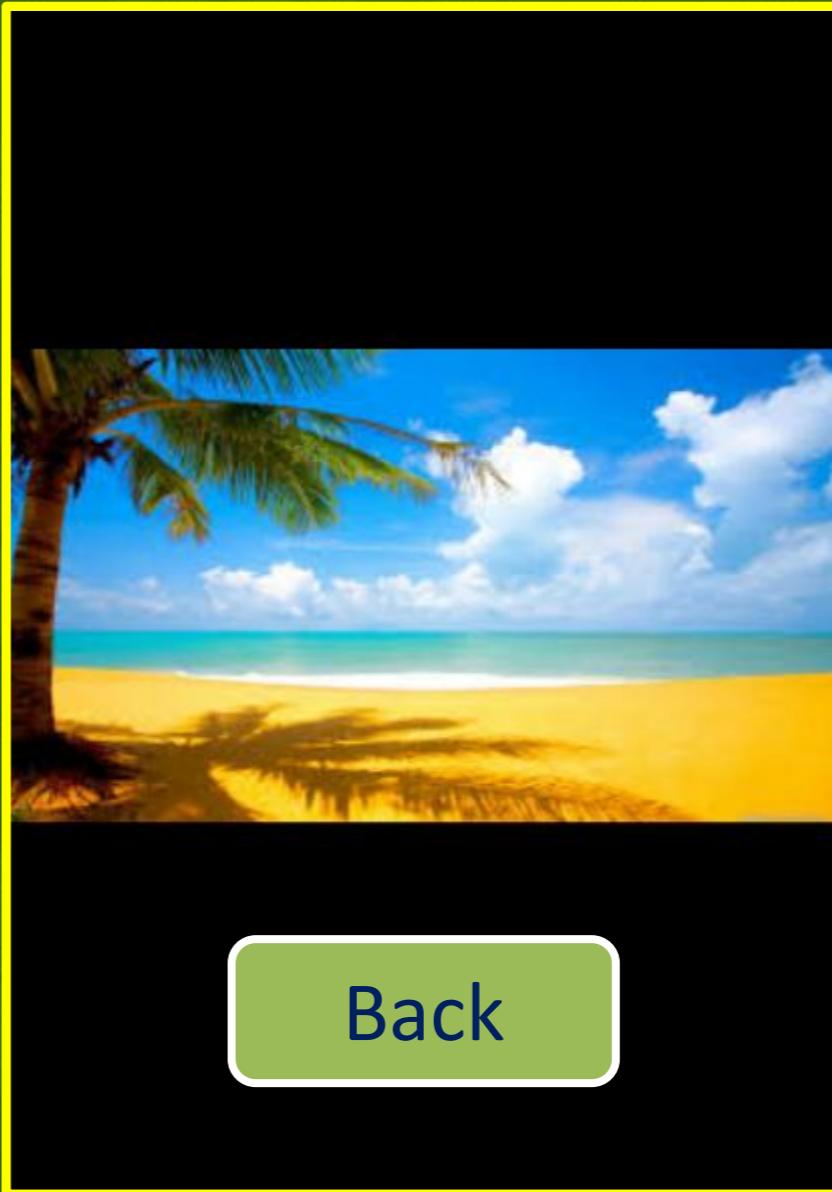
- Ví dụ: abcdef, gfhbnc, uhnduj, ...

Người chơi cần lựa chọn các ô chữ sau cho nó trùng khớp với ô chữ ngẫu nhiên cho trước

Tạo thêm một Progressbar thiết lập thời gian chơi



Exercise GridView (2)



Back

Exercise GridView (3)

List Players

	★ 12	Real madrid
	Lionel Messi	
	★ 13	Barcelona
	Wayne Rooney	
	★ 13	Manchester United
	Gareth Bale	
	★ 13	Real madrid
	Thomas Muller	
	★ 13	Borussia Dortmund
	David Beckham	

Edit Player

Lionel Messi

Barcelona

13

OK CANCEL

List Players

	Cristiano Ronaldo	
	Lionel Messi	

Delete Player!

Are you sure to delete this player ?

No Yes

Gareth Bale
★ 13 Real madrid

Thomas Muller
★ 13 Borussia Dortmund

Chương 07

Android Form Widgets Advanced

Phần 08: Sử dụng ViewFlipper

Sử dụng View Flipper (1)

ViewFlipper chuyển đổi giữa hai hay nhiều View khác nhau (Slide)

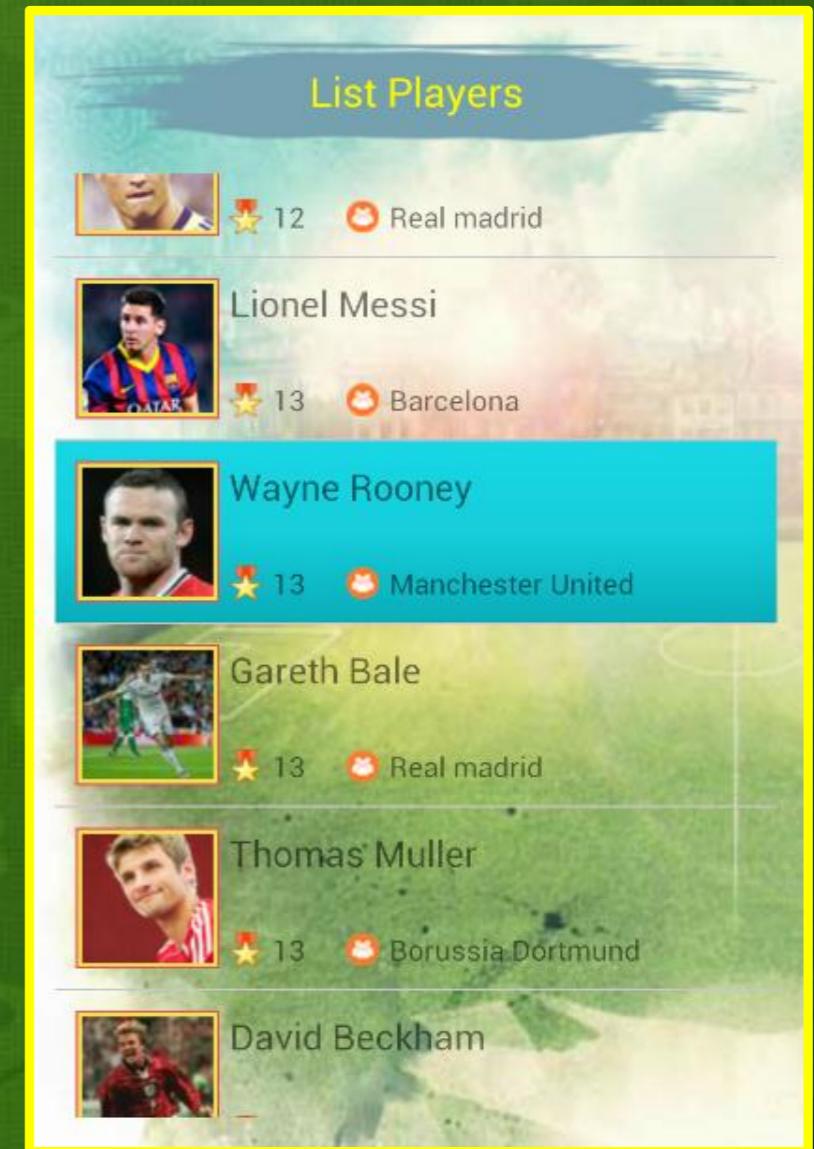
Keyword: *autoStart*, *flipInterval*, *setAutoStart*,
setFlipInterval, *showNext*, *showPrevious*



Sử dụng ViewFlipper (2)

Add Player

- Người dùng nhập Name, Club và Score của Player
- Sử dụng ViewFlipper để chọn hình ảnh của Player

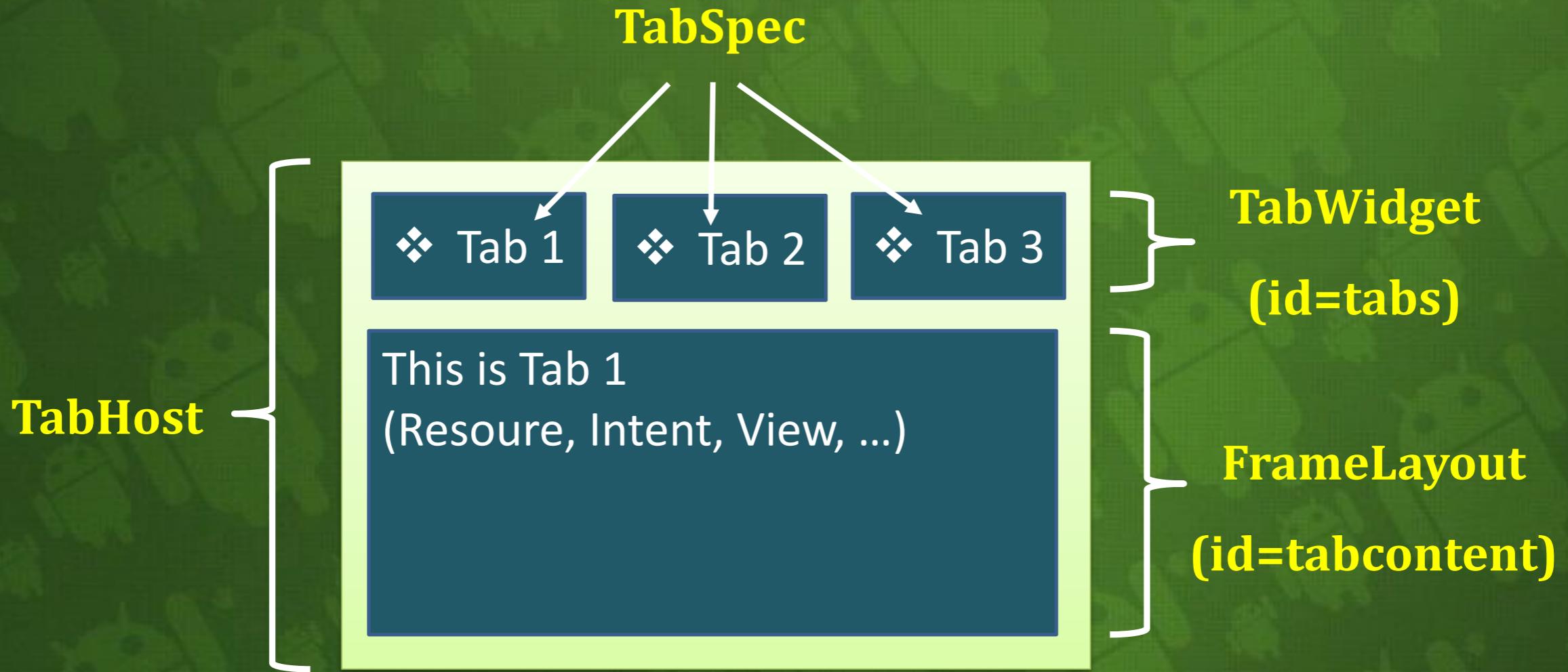


Chương 07

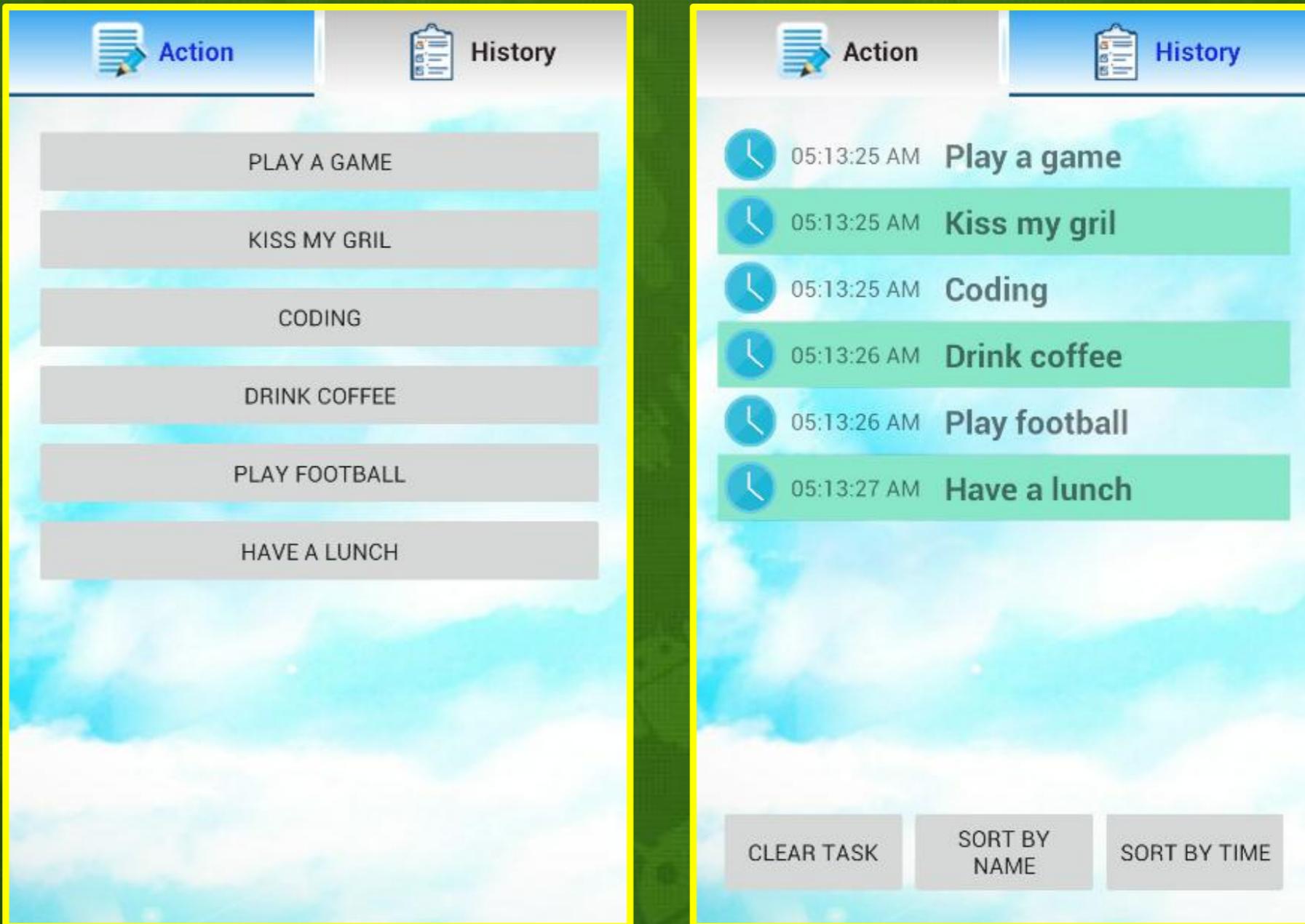
Android Form Widgets Advanced

Phần 09: Sử dụng TabHost

Sử dụng TabHost



Sử dụng TabHost (2)



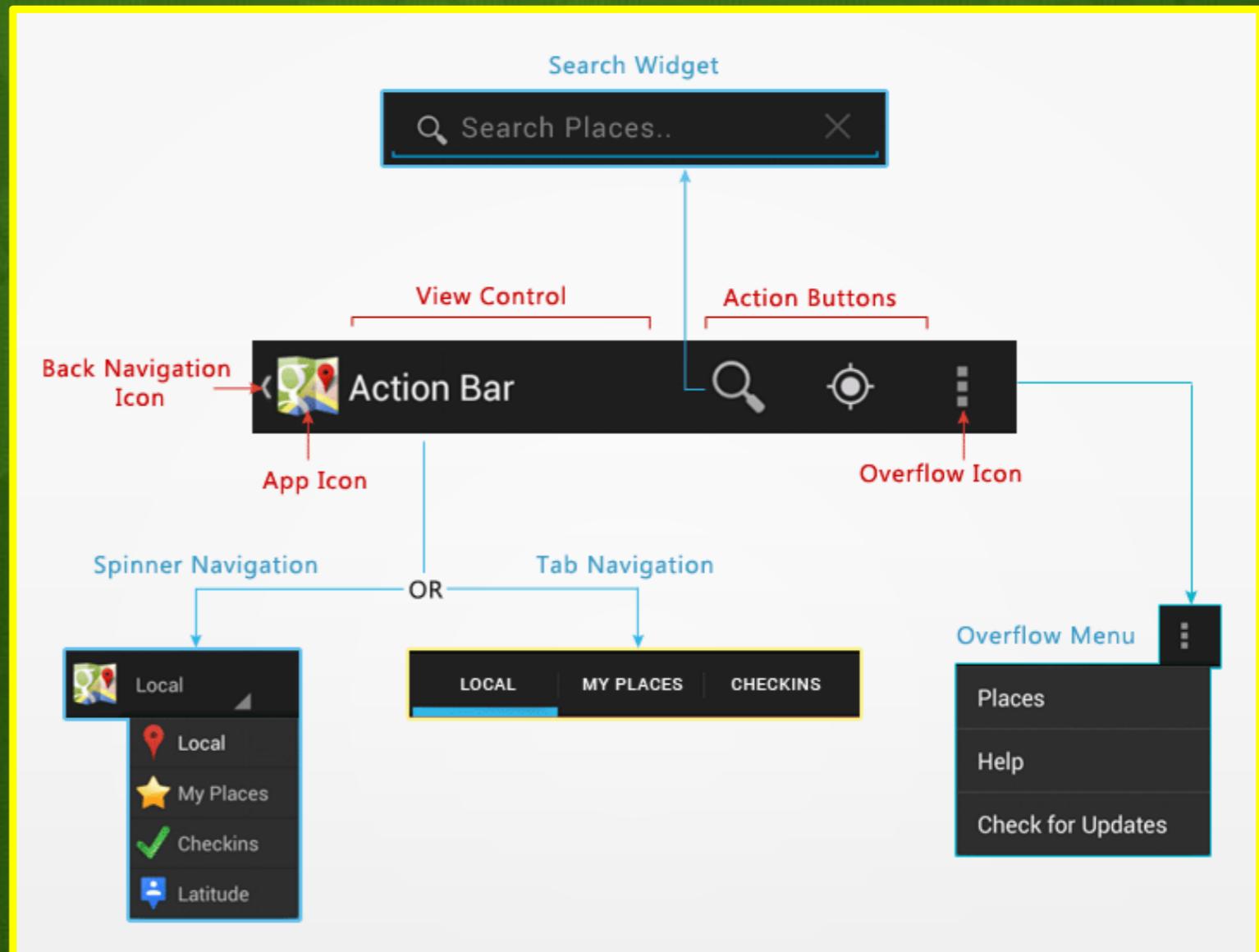
Chương 07

Android Form Widgets Advanced

Phần 10: Sử dụng AppBar

Sử dụng AppBar (1)

AppBar (ActionBar) tạm gọi “thanh ứng dụng” cung cấp cái nhìn trực quan về ứng dụng, dễ dàng bố trí các tính năng thường dùng của ứng dụng, ...



Sử dụng AppBar (2)

Từ Android 3.0 (API Level 11) các activity đều sử dụng theme mặc định và có *ActionBar* như là một “thanh ứng dụng”. Qua các phiên bản Android khác nhau thì *ActionBar* ngày càng được bổ sung các chức năng, làm cho *ActionBar* sẽ chạy khác nhau đối với các version khác nhau của *Android*

Ngược lại các tính năng mới nhất được thêm vào thư viện hỗ trợ *ToolBar* (có sẵn trên các thiết bị) → sử dụng thư viện hỗ trợ của *ToolBar* để implement *AppBar* cho ứng dụng (*ActionBar* sẽ chạy được trên nhiều thiết bị khác nhau)

Nguồn: <http://developer.android.com/training/appbar/setting-up.html>

Sử dụng AppBar (3)

Thêm ToolBar vào Activity

1. *Thêm thư viện hỗ trợ v7 appcompat vào trong Project*
2. *Activity kế thừa class AppCompatActivity*
3. *Sử dụng theme NoActionBar*
4. *Thêm Toolbar trong layout*
5. *setSupportActionBar() trong Activity*

Chương 07

Android Form Widgets Advanced

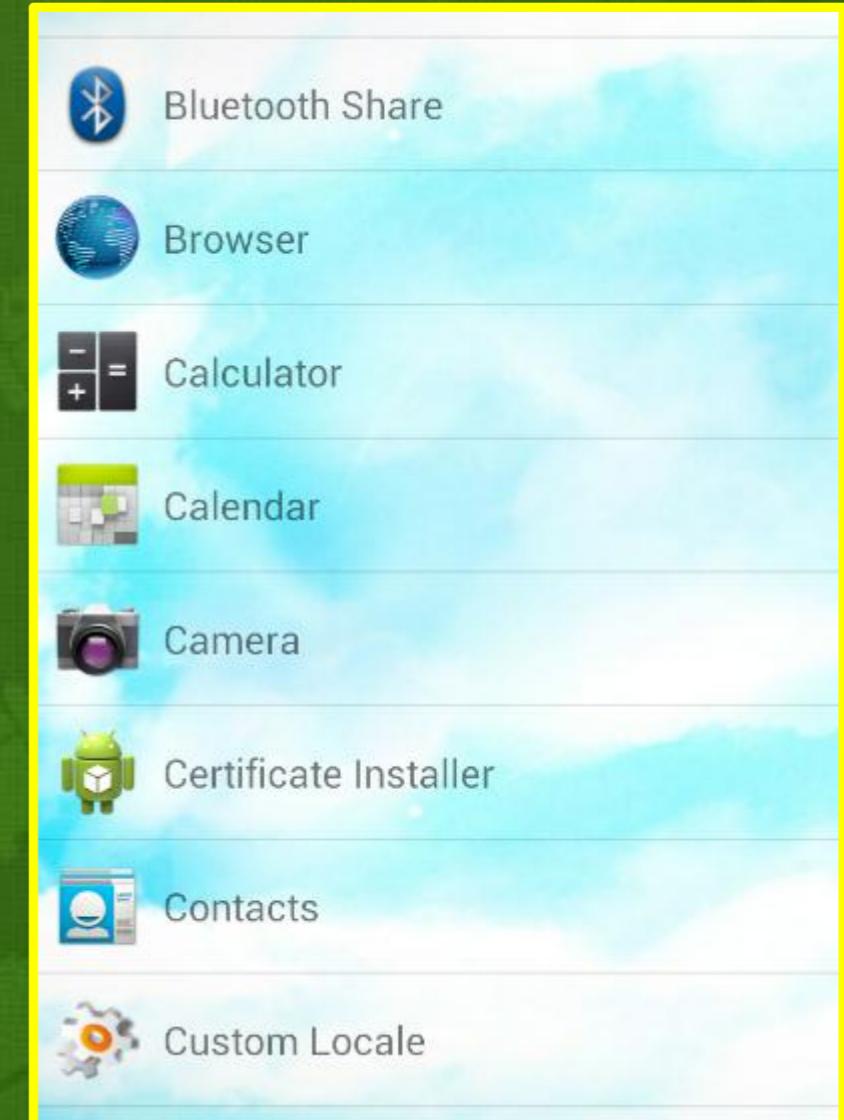
Phần 11: Truy cập dữ liệu hệ thống

Truy xuất các ứng dụng được cài đặt

Sử dụng class *PackageManager* để truy xuất các loại thông tin khác nhau liên quan đến các gói ứng dụng được cài đặt hiện trên thiết bị.

Truy xuất thông tin ứng dụng và thông tin tất cả các ứng dụng đang được cài đặt trên thiết bị

Keyword: android get list application, PackageManager, getInstalledApplications, ApplicationInfo, ...



Truy xuất các Contact trong danh bạ

Sử dụng class *ContentResolver* để “accessing the database from another app”

Permission:

android.permission.READ_CONTACTS

Keyword: *android* *get* *list* *contact*,
ContentResolver, ...

	Ronaldo	(091) 929-3949
	Bale	(098) 989-88
	Messi	01683308983

Chương 08

Xây dựng ứng dụng TodoList

TodoList - Các chức năng cần thực hiện (1)

Quản lý các công việc cần thực hiện trong ngày: name, icon, type, time, rating

- Class Task quản lý thông tin công việc
- Class Category quản lý nhóm công việc
- Tạo dữ liệu mẫu
- Custom giao diện

	Play Football	Home
		02:09 AM
	Kiss my girl	Home
		02:09 AM
	Play bida	Home
		02:09 AM
	ABC	Company
		12:19 PM

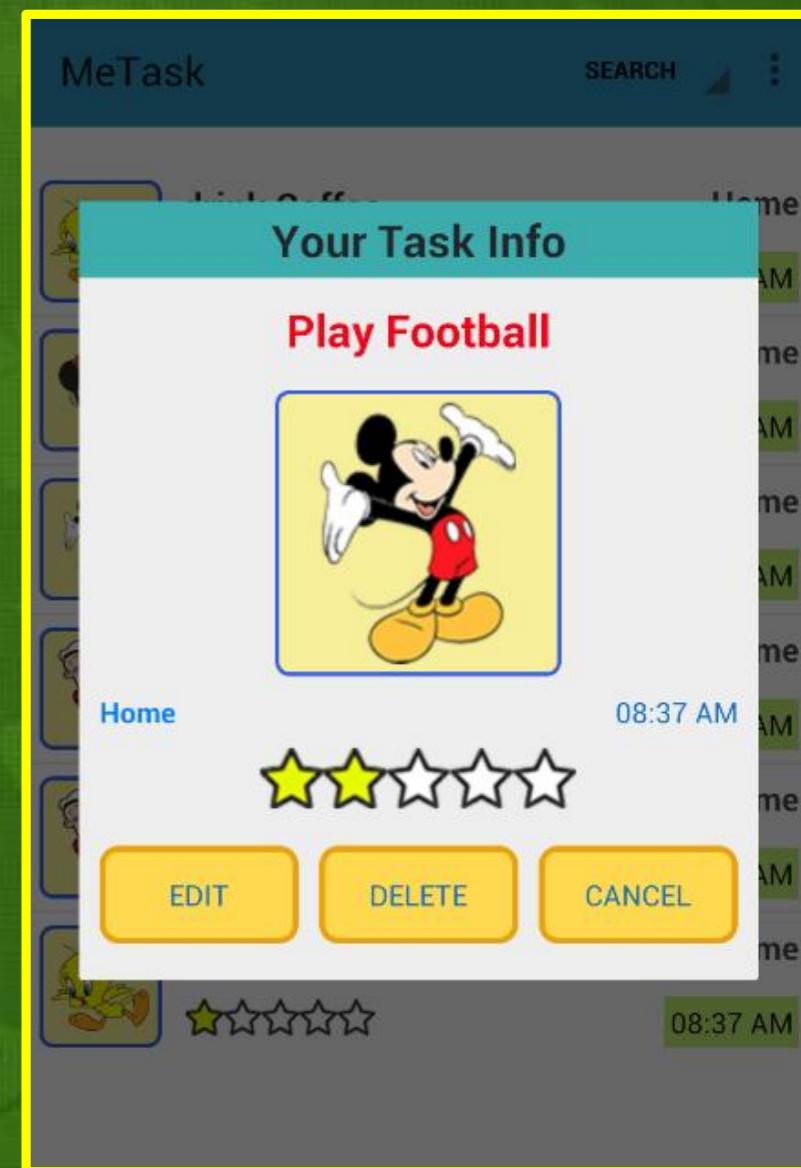
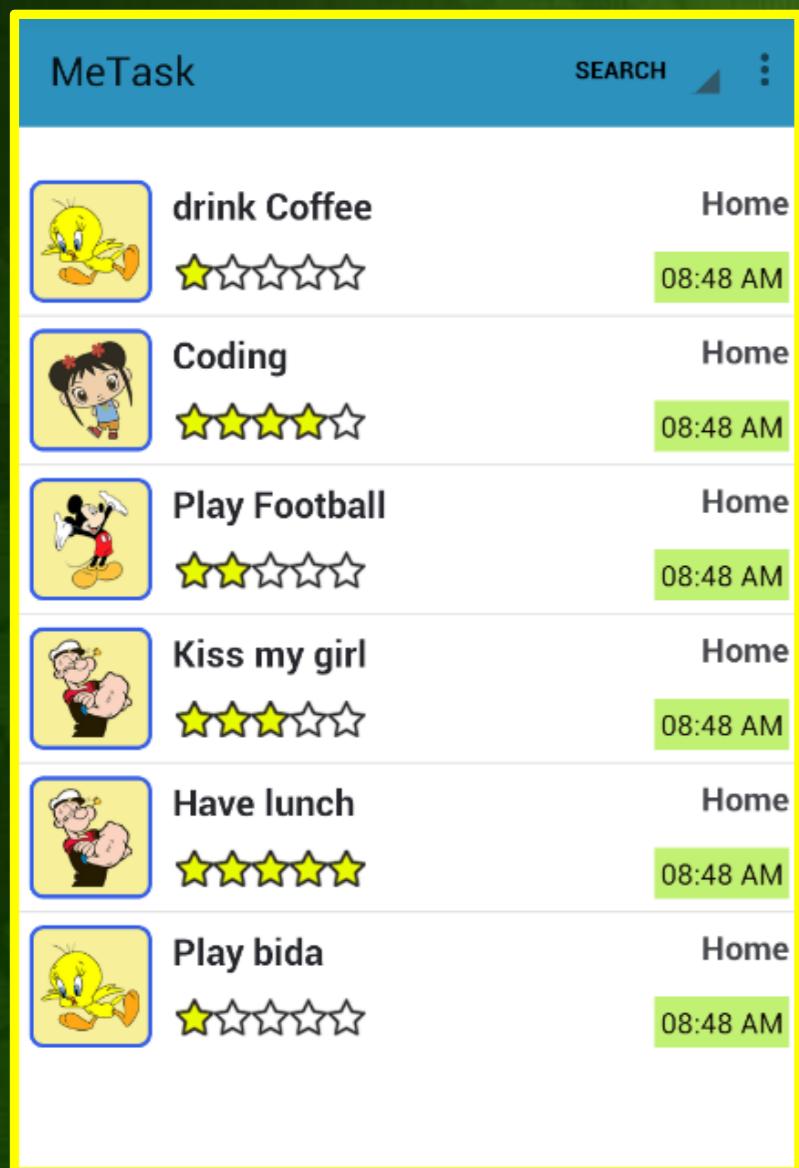
TodoList - Các chức năng cần thực hiện (2)

Sắp xếp danh sách công việc theo

- Tên của công việc ($A \rightarrow Z$)
- Tên của công việc ($Z \rightarrow A$)
- Độ ưu tiên của công việc đó
- ...

	Play Football	Home
	Kiss my girl	Home
	Play bida	Home
	ABC	Company

TodoList - Xem thông tin chi tiết công việc



TodoList - TÌM KIẾM CÔNG VIỆC

The image displays three screenshots of a mobile application interface for managing tasks, titled "MeTask".

Screenshot 1 (Left): Shows the main task list. Each task includes an icon, name, location, time, and rating.

Icon	Name	Location	Time	Rating
	drink Coffee	Home	08:48 AM	★★★★★
	Coding	Home	08:48 AM	★★★★★
	Play Football	Home	08:48 AM	★★★★★
	Kiss my girl	Home	08:48 AM	★★★★★
	Have lunch	Home	08:48 AM	★★★★★
	Play bida	Home	08:48 AM	★★★★★

Screenshot 2 (Middle): Shows the search results for the query "drink Coffee". It lists the task details from the first screenshot.

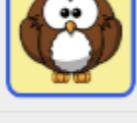
Icon	Name	Location	Time	Rating
	drink Coffee	Home	08:48 AM	★★★★★
	Coding	Home	08:48 AM	★★★★★
	Play Football	Home	08:48 AM	★★★★★
	Kiss my girl	Home	08:48 AM	★★★★★
	Play bida	Home	08:48 AM	★★★★★

Screenshot 3 (Right): Shows the search results for the query "in". It lists the task details from the first screenshot.

Icon	Name	Location	Time	Rating
	drink Coffee	Home	08:30 AM	★★★★★
	Coding	Home	08:30 AM	★★★★★
	Play Football	Home	08:30 AM	★★★★★
	Kiss my girl	Home	08:30 AM	★★★★★
	Play bida	Home	08:30 AM	★★★★★

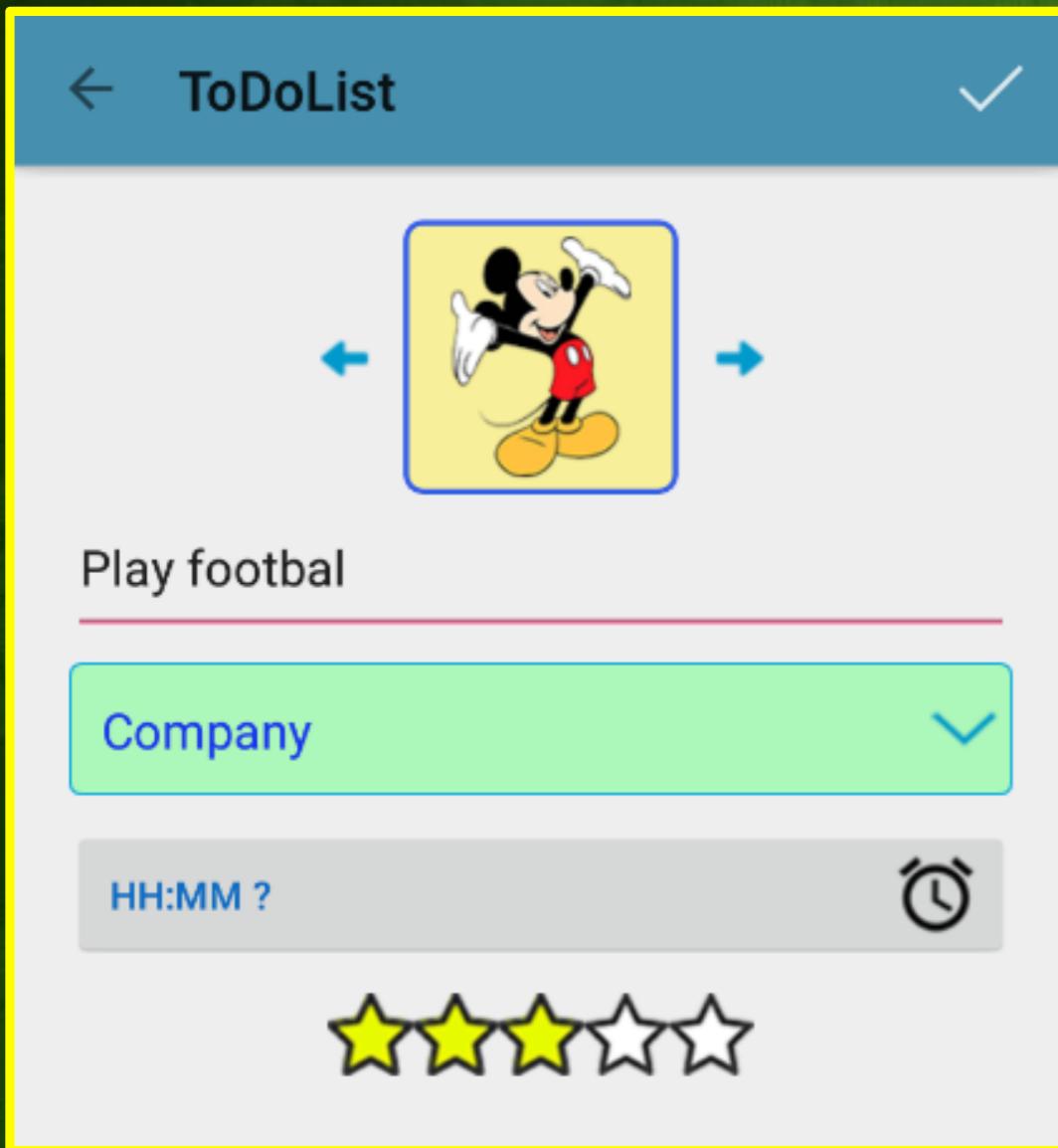
ToDoList - Multi Delete

ToDoList					
SEARCH :					
	drink coffee	★★★★★	Home	02:01 AM	
	coding	★★★★★	Company	02:01 AM	
	play football	★★★★★	Others	02:01 AM	
	kiss my girl	★★★★★	Home	02:01 AM	
	have lunch	★★★★★	Home	02:01 AM	

← 2 selected					
ToDoList					
	drink coffee	★★★★★	Home	02:01 AM	
	coding	★★★★★	Company	02:01 AM	
	play football	★★★★★	Others	02:01 AM	
	kiss my girl	★★★★★	Home	02:01 AM	
	have lunch	★★★★★	Home	02:01 AM	

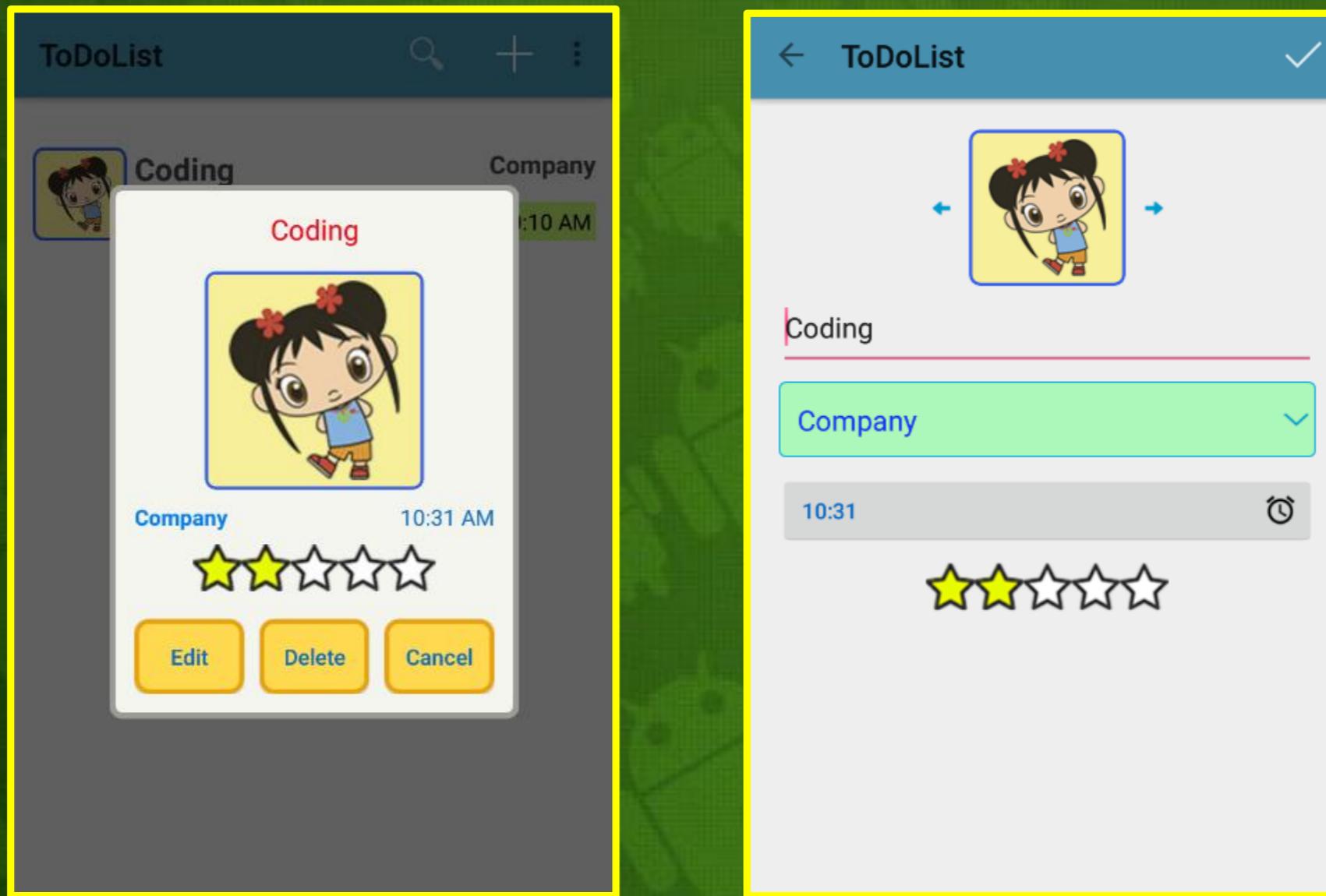
ToDoList					
SEARCH :					
	drink coffee	★★★★★	Home	02:01 AM	
	kiss my girl	★★★★★	Home	02:01 AM	
	have lunch	★★★★★	Home	02:01 AM	
Successfull					

ToDoList - Add Task



- Xây dựng giao diện
- ViewFlipper
- Spinner Category
- TimePickerDialog
- Validate → Add Task
- Load List Task

ToDoList - Edit Task



Chương 09

Sử dụng Animation Resources trong Android

1. *Animation Resource*
2. *Property Animation*
3. *Alpha Animation*
4. *Scale Animation*
5. *Rotate Animation*
6. *Translate Animation*
7. *Multi Animation*
8. *So sánh cách sử dụng Animator và Animation*
9. *Frame Animation*
10. *Interpolators và Custom interpolators*
11. *Ứng dụng Animation trong việc xây dựng giao diện*

#01 - Animation Resource

Animation Resource xây dựng các hiệu ứng chuyển đổi cho các đối tượng View

Phân loại Animation Resource:

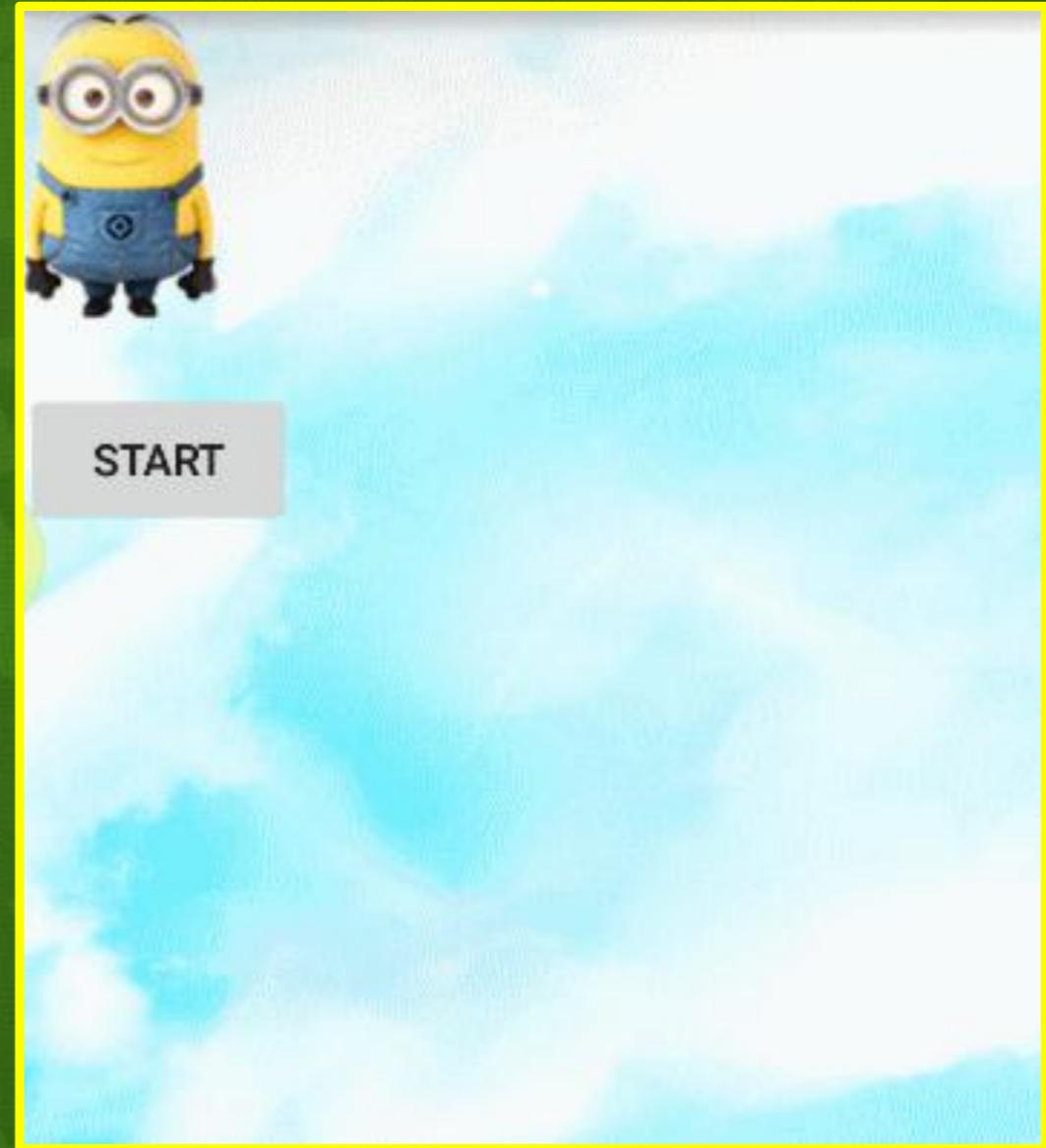
- *Property Animation*
- *View Animation*
 - *Tween Animation (Alpha, Scale, Rotate, Translate)*
 - *Frame Animation*

#02 - Property Animation

Property Animation Creates an animation by modifying an object's property values over a set period of time with an Animator

Vị trí: res/animator/fileName.xml

- Sử dụng XML
- Sử dụng Java code



#03 - Tween Animation

Tween Animation Creates an animation by performing a series of transformations on a single image with an Animation

- *Alpha Animation* *Làm mờ một đối tượng*
- *Scale Animation* *Phóng to (thu nhỏ) một đối tượng*
- *Rotate Animation* *Xoay đối tượng*
- *Translate Animation* *Di chuyển một đối tượng*

#03 - Alpha Animation

Alpha Animation tạo hiệu ứng làm mờ dần và hiển thị dần dần của một đối tượng nào đó

Vị trí: res/anim/fileName.xml

Keyword: alpha, fromAlpha, toAlpha



#04 - Scale Animation

Scale Animation tạo hiệu ứng Zoom của một đối tượng nào đó

Vị trí: res/anim/fileName.xml

Keyword: scale, fromXScale, toXScale, fromYScale, toYScale, pivotX, pivotY



#05 - Rotate Animation

Rotate Animation tạo hiệu ứng xoay của một đối tượng nào đó

Vị trí: *res/anim/fileName.xml*

Keyword: *rotate, fromDegrees, toDegrees, pivotX, pivotY*



#06 - Translate Animation (1)

Translate Animation tạo hiệu ứng di chuyển một đối tượng nào đó

Vị trí: *res/anim/fileName.xml*

Keyword: *translate, fromYDelta, toYDelta, duration, fromXDelta, toXDelta*



#06 - Translate Animation (2)

Các sự kiện bắt đầu di chuyển và kết thúc di chuyển

Keyword: `setAnimationListener`, `onAnimationStart`,
`onAnimationEnd`, `onAnimationRepeat`



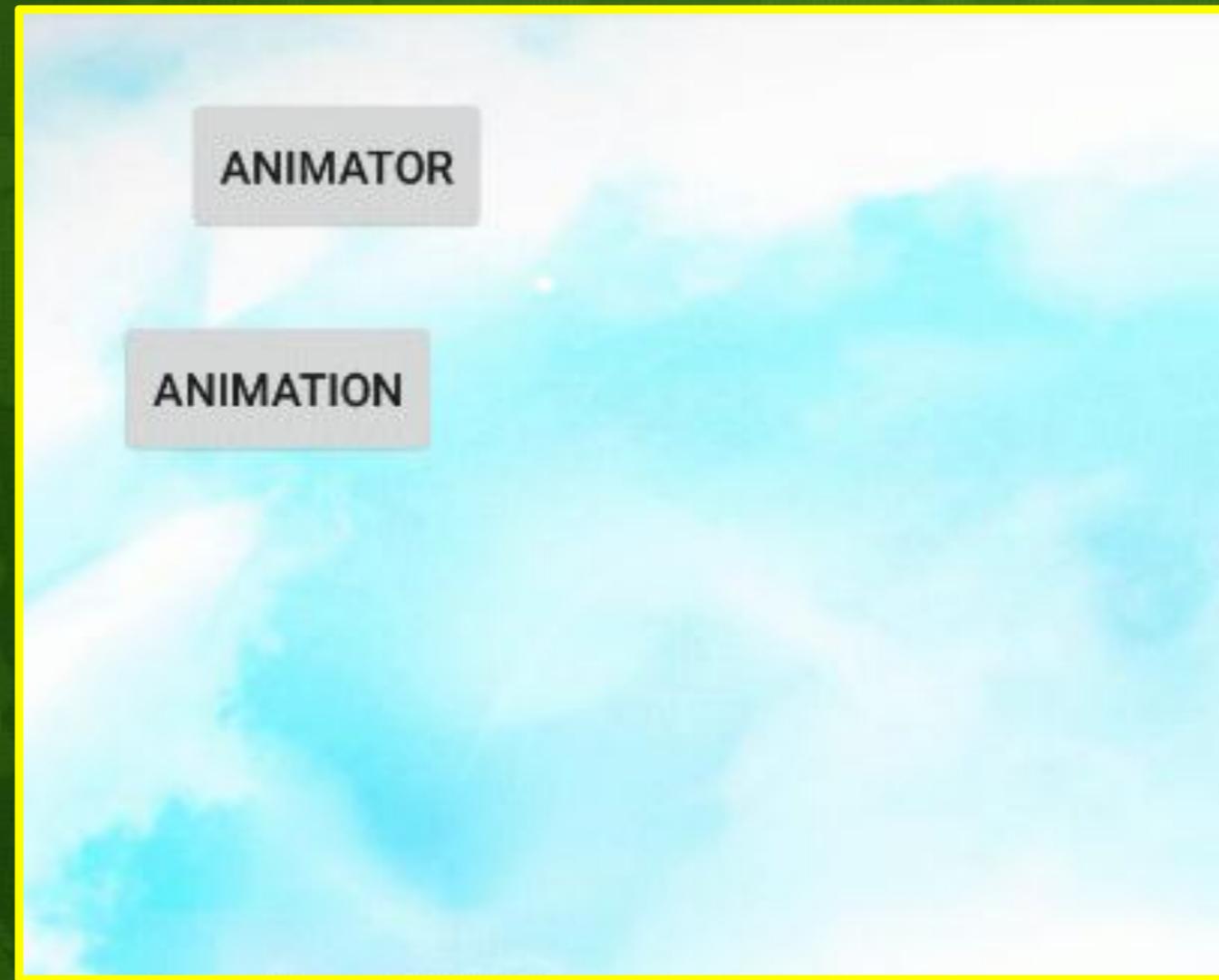
#07 - Kết hợp các Animation

Kết hợp các Tween Animation để xây dựng giao diện hoạt họa như slide hình

- *Sử dụng XML*
- *Sử dụng Java code*



#08 – So sánh Animation và Animator

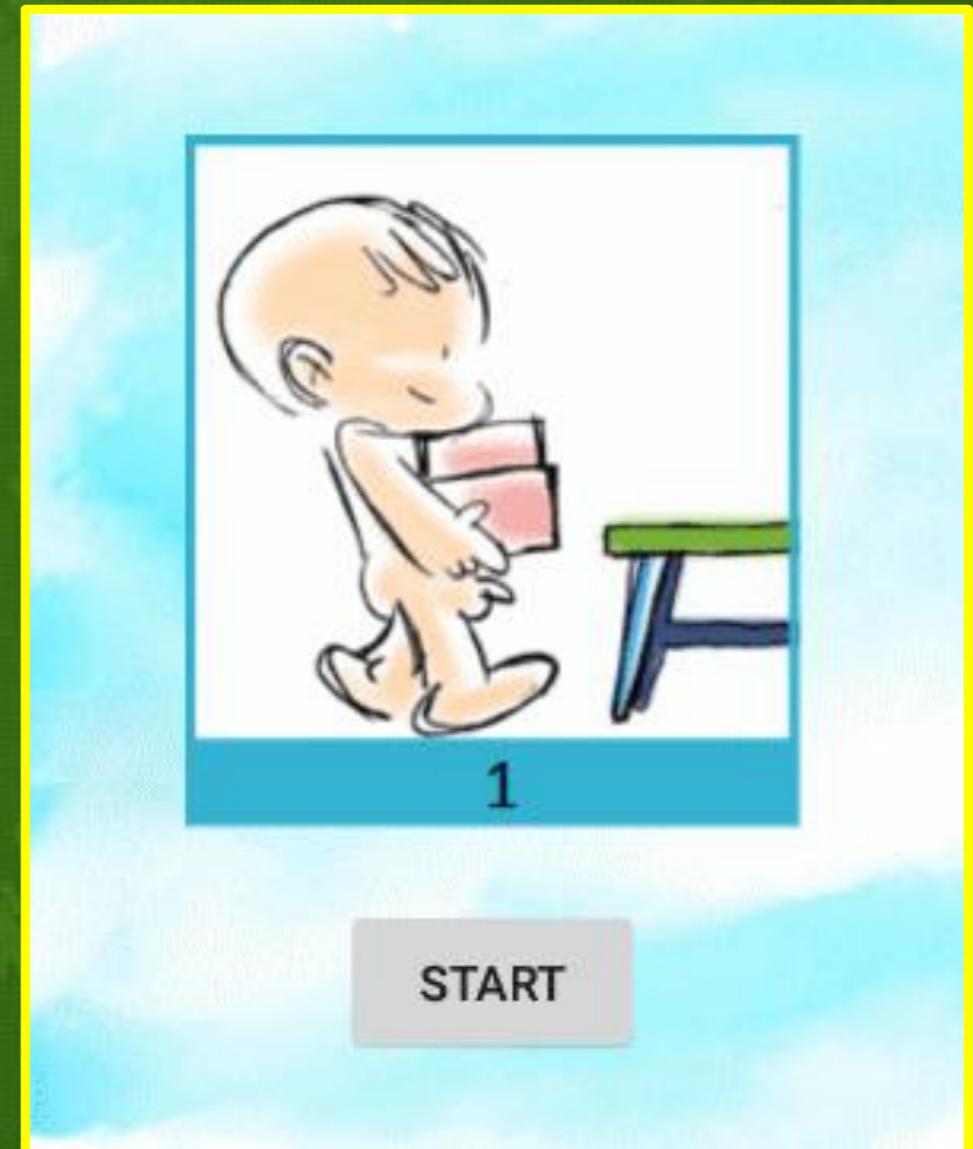


#09 - Frame Animation

Frame Animation creates an animation by showing a sequence of images in order with an AnimationDrawable

Vị trí: res/drawable/fileName.xml

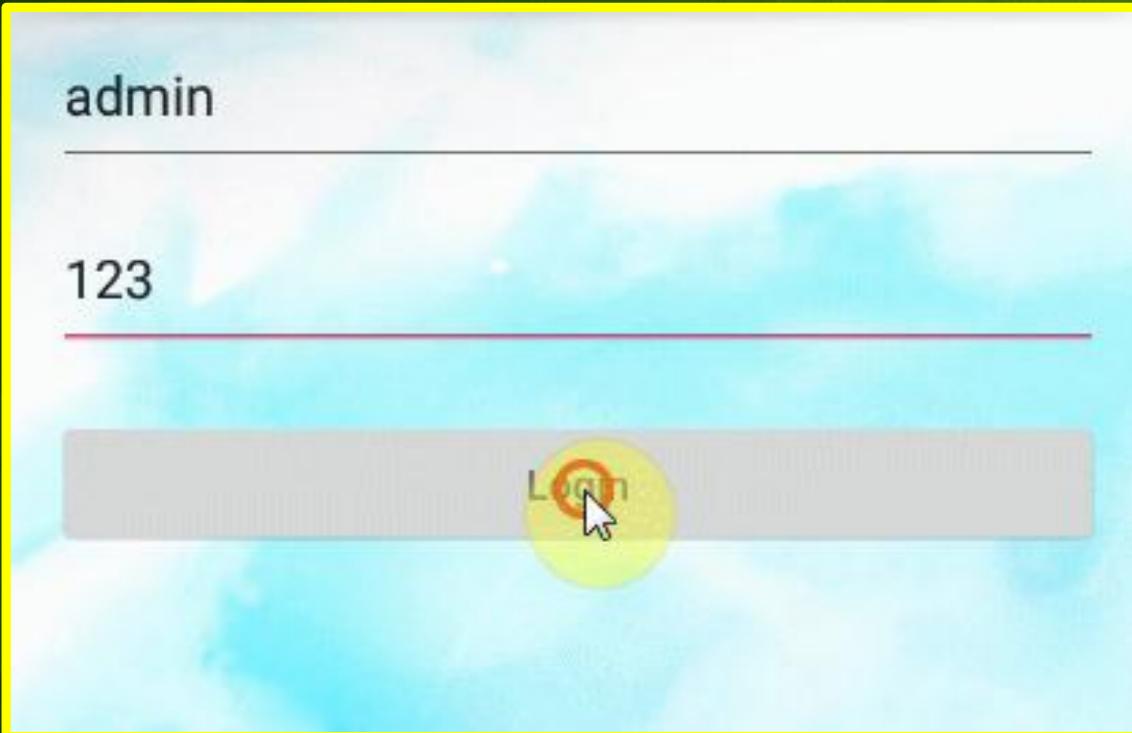
→ *Liên tưởng đến Level List*



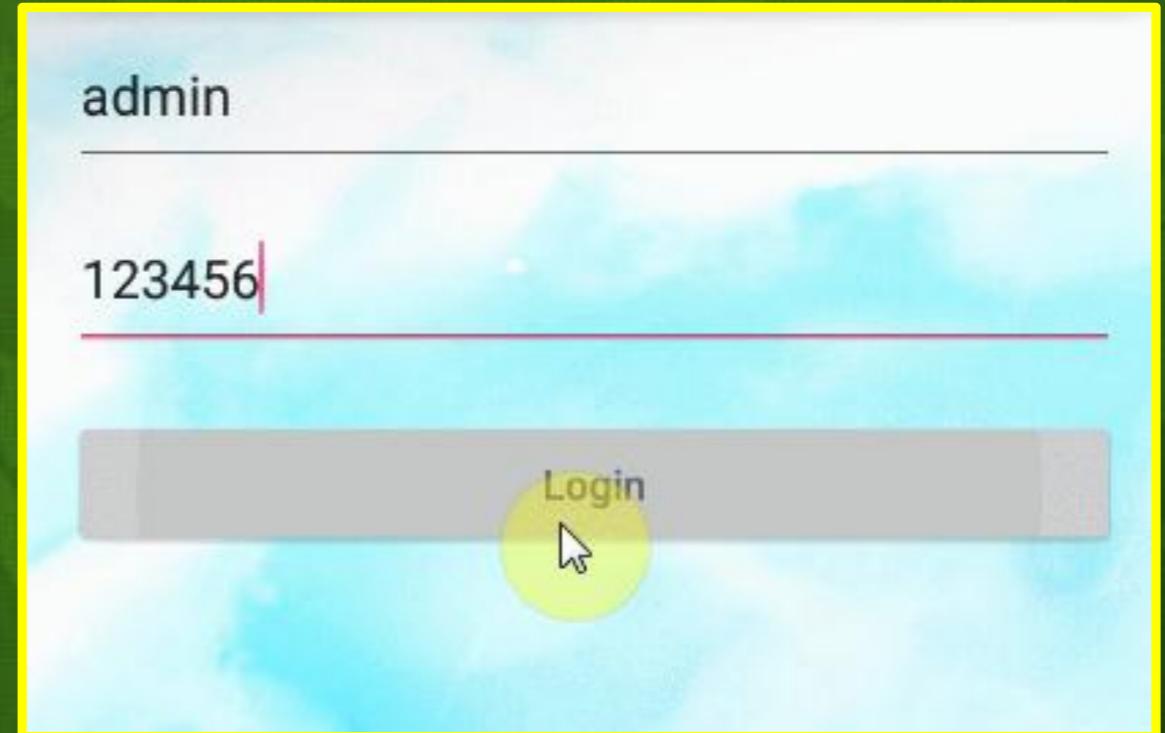
#10 - Interpolators và Custom Interpolators

- *Interpolator* An interpolator defines the rate of change of an animation. This allows the basic animation effects (alpha, scale, translate, rotate) to be accelerated, decelerated, repeated, etc.
- *Keyword: android interpolator*
- *Cách custom Interpolator*

#11 - Áp dụng Animation - Login



Hiệu ứng rung khi thông tin đăng nhập sai



Hiệu ứng chuyển Activity khi đăng nhập thành công

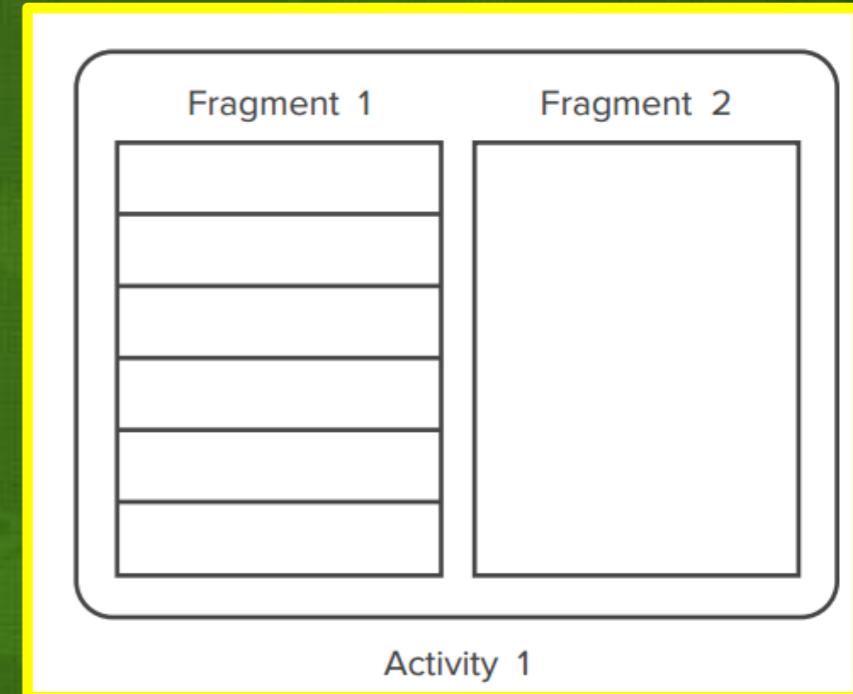
Chương 10

Xây dựng giao diện với Fragment

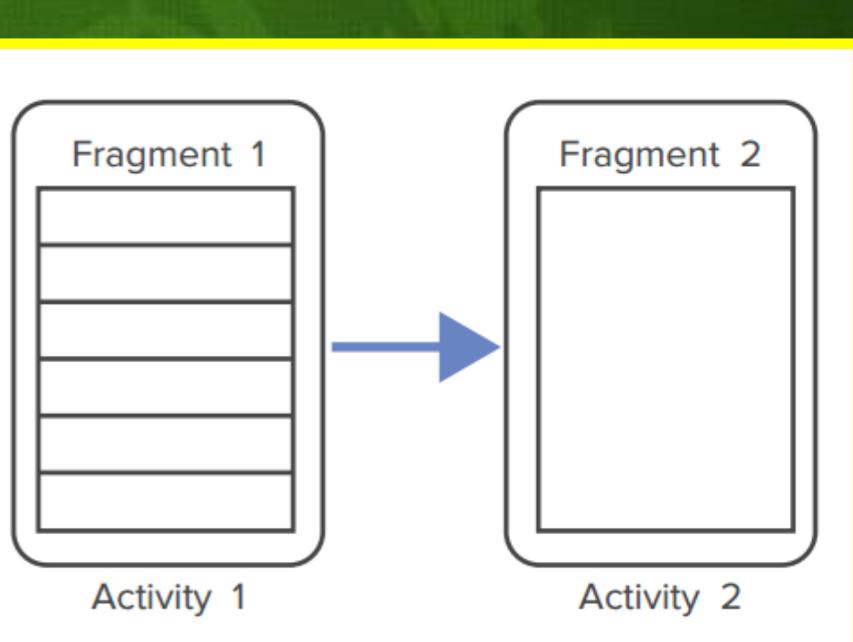
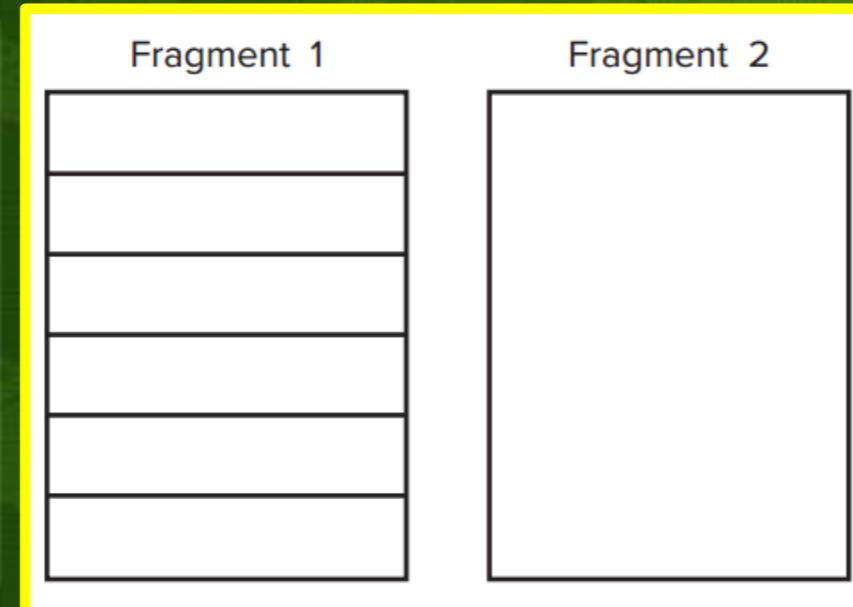
- 1. *Create Fragment*
- 2. *Fragment Life Cycle*
- 3. *Thay đổi layout theo Screen Orientation*
- 4. *Fragment Communicator*
- 5. *Fragment onSaveInstanceState*
- 6. *Fragment Backstack*
- 7. *Sử dụng ListFragment*
- 8. *Sử dụng Dialog Fragment*
- 9. *Tình huống xử lý giao diện với Fragment*

#1 Fragment (1)

Fragment là một loại sub-Activity, Fragment được chứa trong một Activity hoặc Fragment khác, mỗi Activity có thể chứa một hoặc nhiều Fragment



*Liên hệ Layout
“Master - detail”*



#1 Fragment (2)

- *Fragment có thể được xem là một Sub Activity*
- *Fragment có vòng đời riêng của nó*
- *Fragment được add hoặc remove trong khi Activity đang được thực thi*
- *Fragment được giới thiệu từ API 11, có thể sử dụng Fragment ở những API thấp hơn bằng cách sử dụng các gói thư viện hỗ trợ*

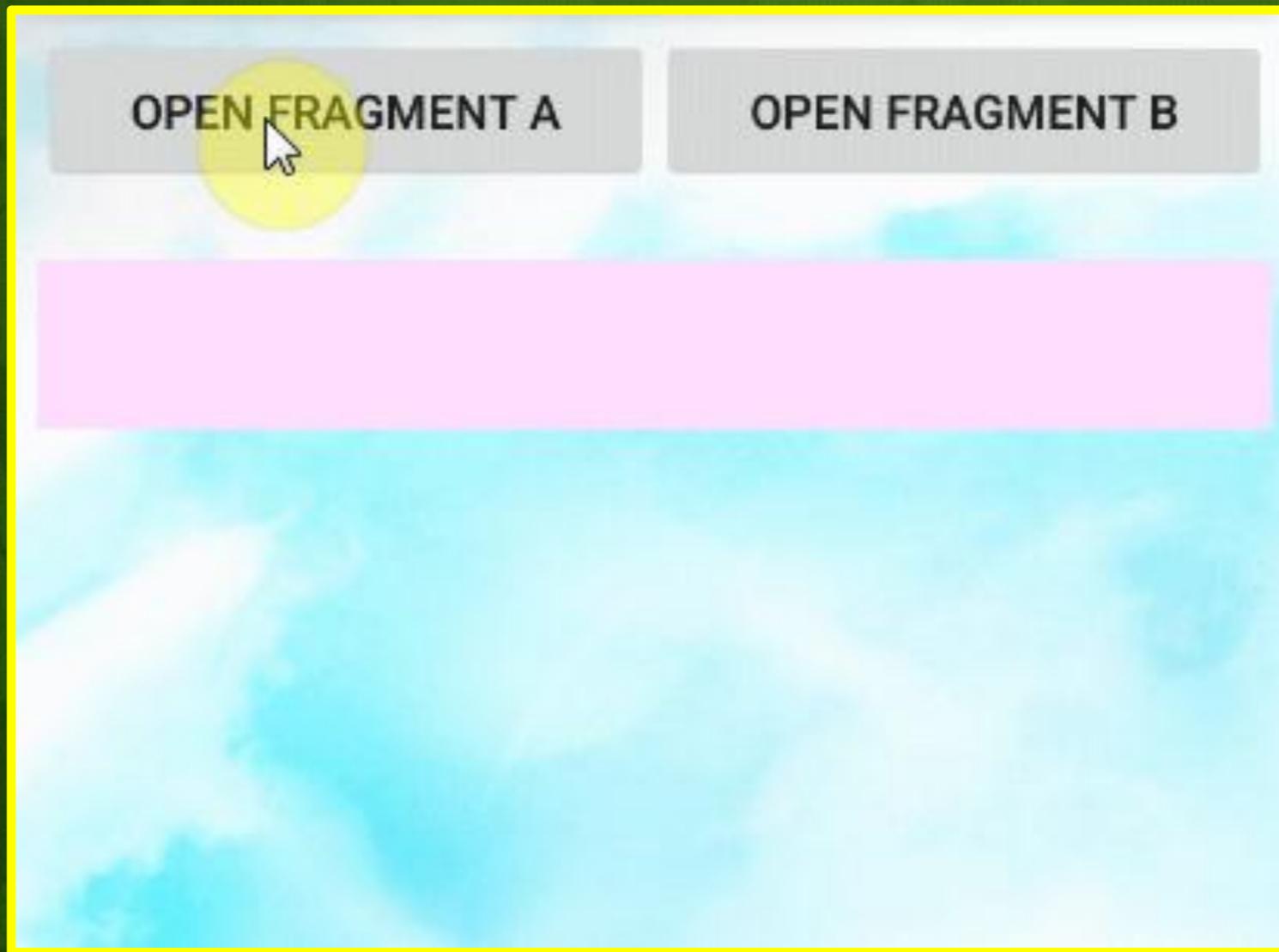
#1 Create Fragment - XML

1. *Tạo class FragmentX và extend từ Fragment (android.app.Fragment)*
2. *Tạo layout cho FragmentX*
3. *Override onCreateView trong FragmentX và chỉ định tập tin layout cần sử dụng*
4. *Tạo thẻ fragment trong tập tin layout chính và kéo vào class FragmentX thông qua android:name*

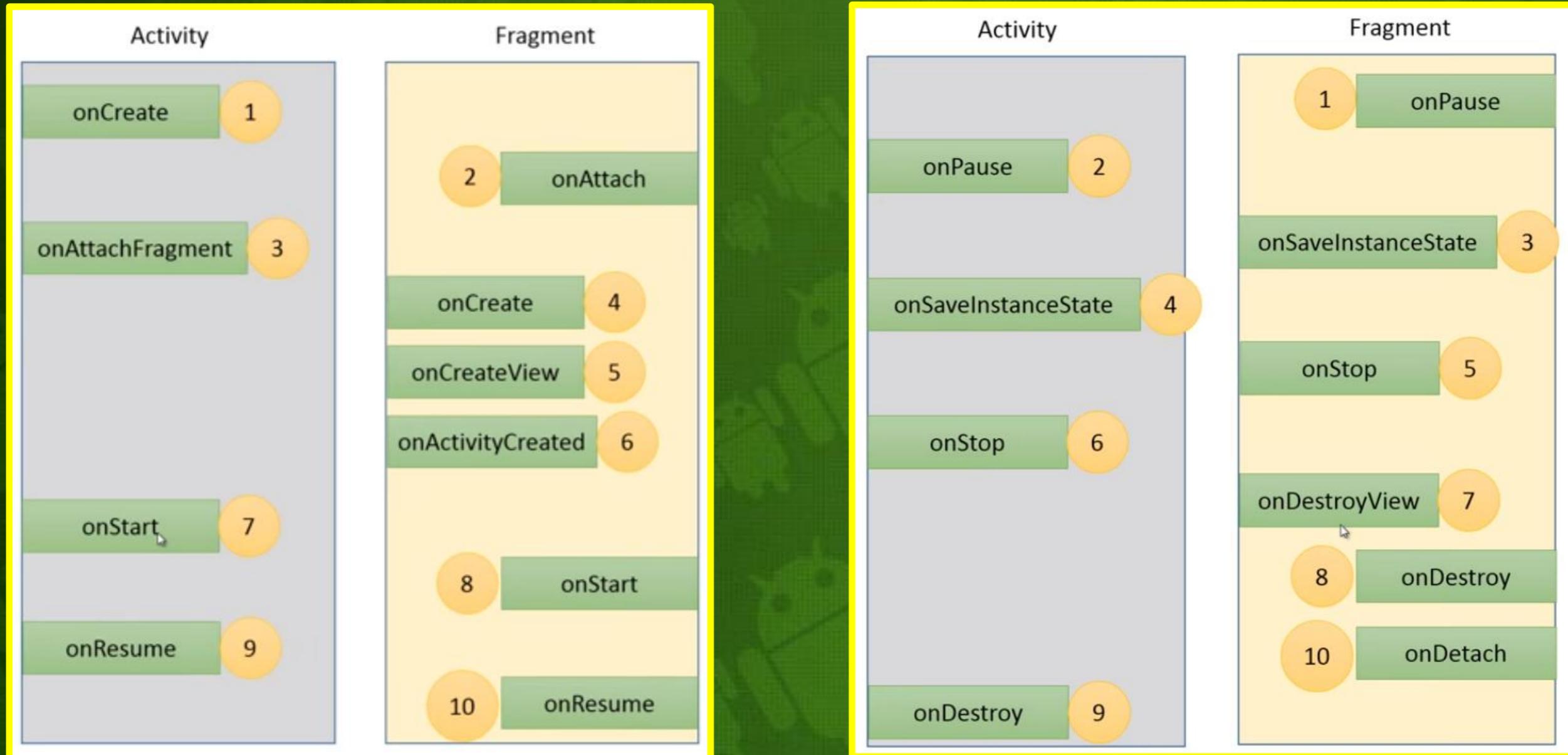
#1 Create Fragment - Java

1. *Tạo class FragmentX và extend từ Fragment (android.app.Fragment)*
2. *Tạo layout cho FragmentX*
3. *Override onCreateView trong FragmentX và chỉ định tập tin layout cần sử dụng*
4. *getFragmentManager().beginTransaction()*
5. *add()*
6. *commit()*

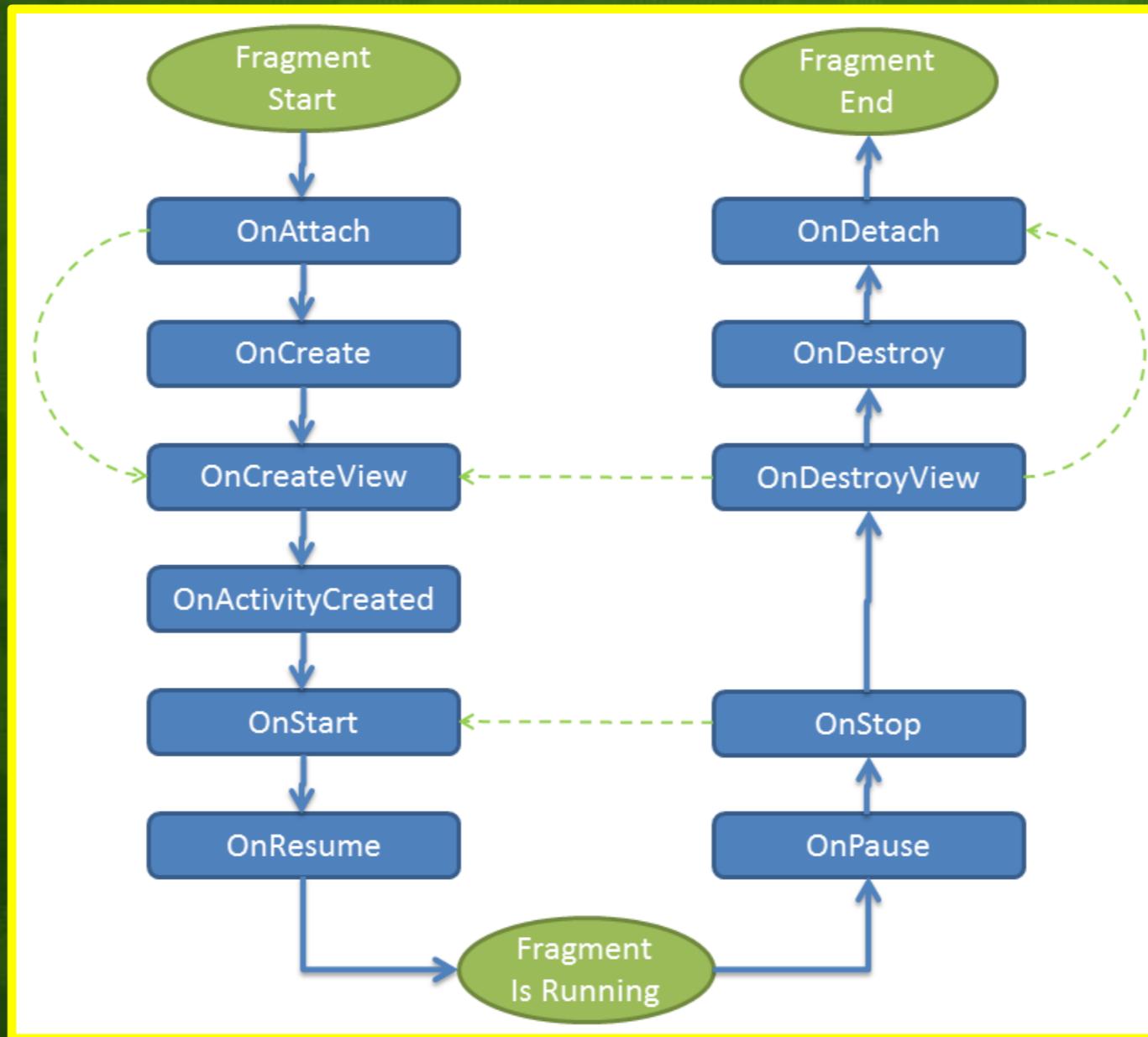
#1 Fragment - replace



#2 Fragment - Lifecycle (1)



#2 Fragment - Lifecycle (2)



#3 Fragment - Orientation

Portrait

Land

SUBMIT

#4 Fragment - Communicator (1)

Activity – Button Call Fragment

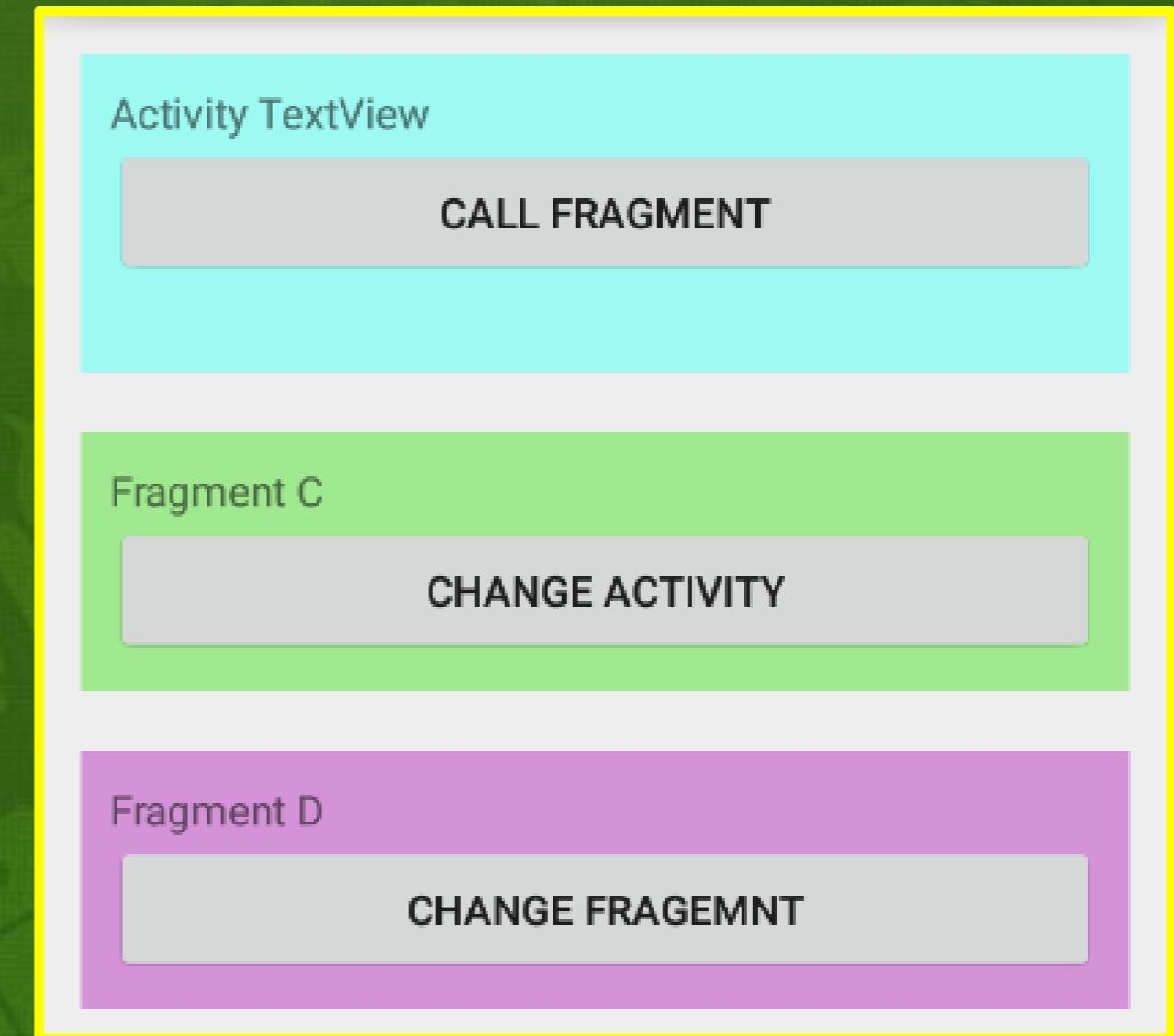
- *Changed: Fragment C, Fragment D*

Fragment C – Button Change Activity

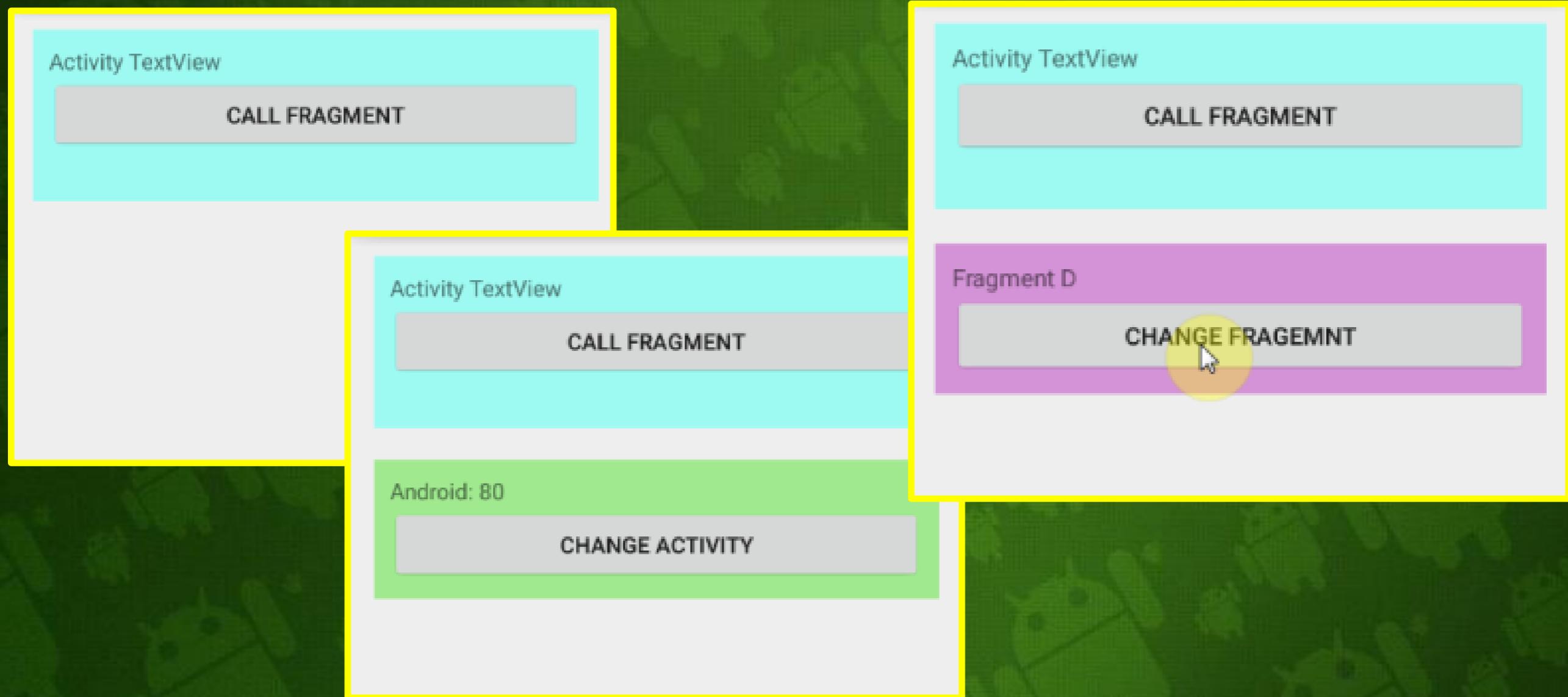
- *Activity TextView changed*

Fragment D – Button Change Fragment

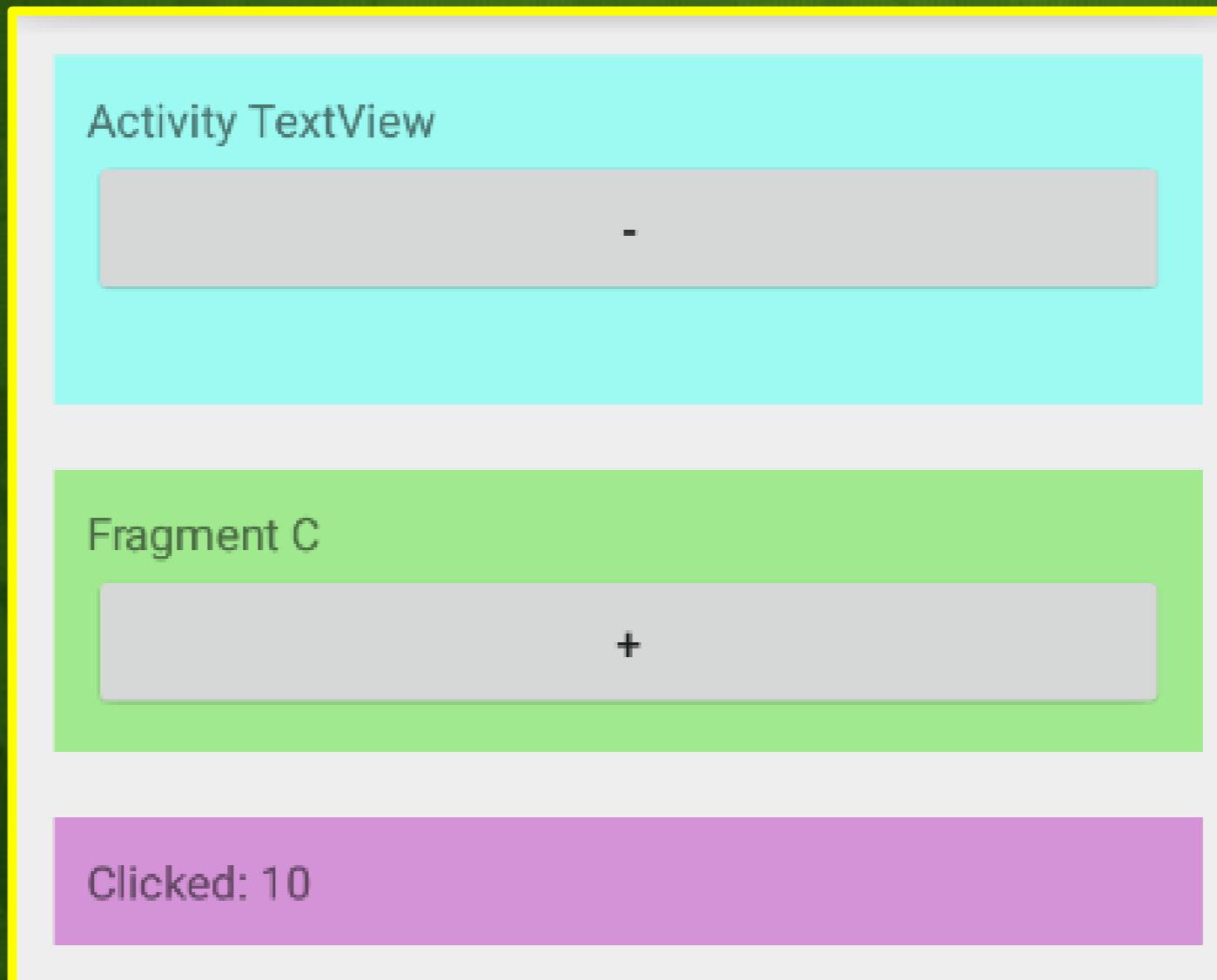
- *Fragment C changed*



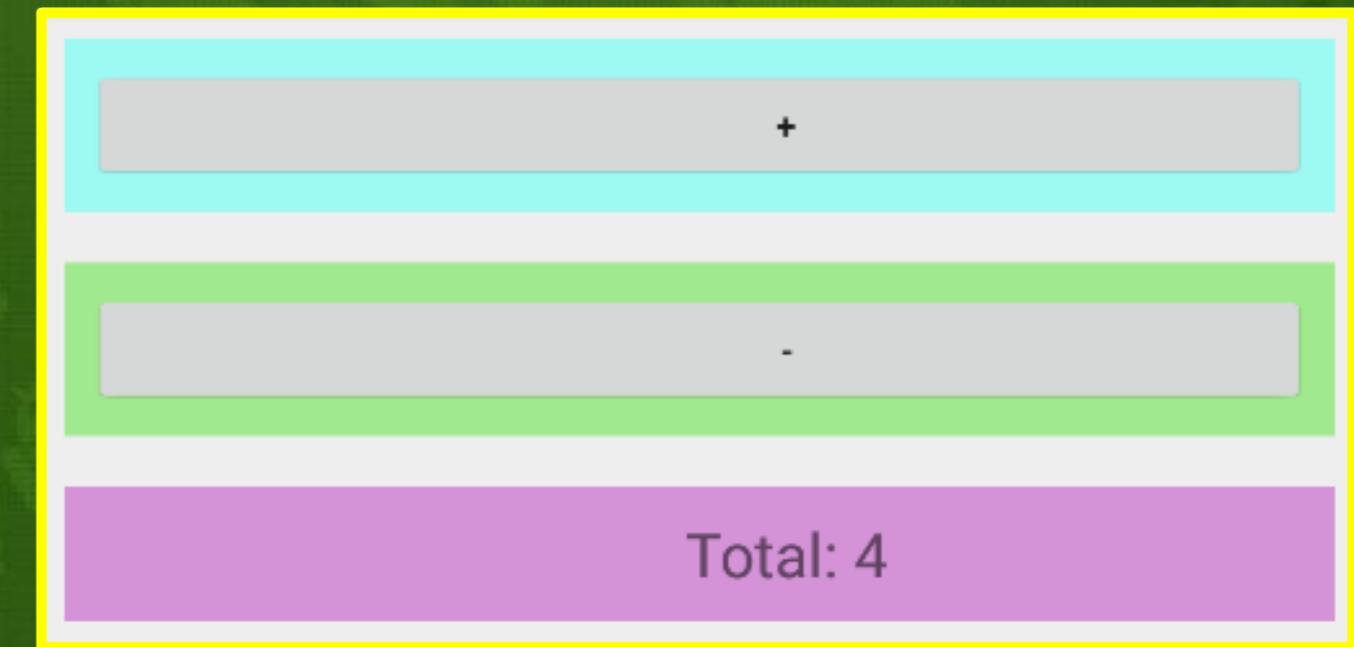
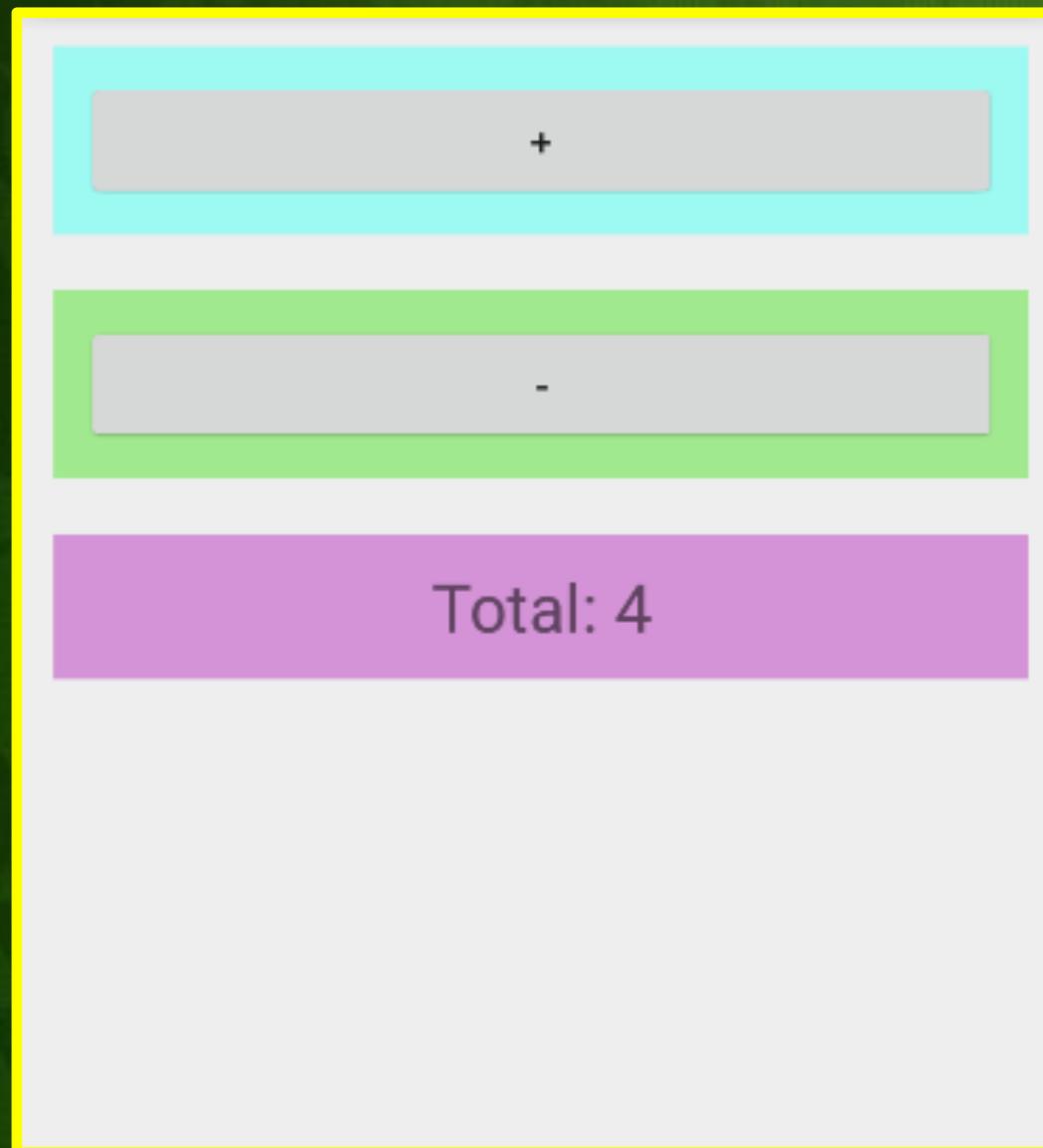
#4 Fragment - Communicator (2)



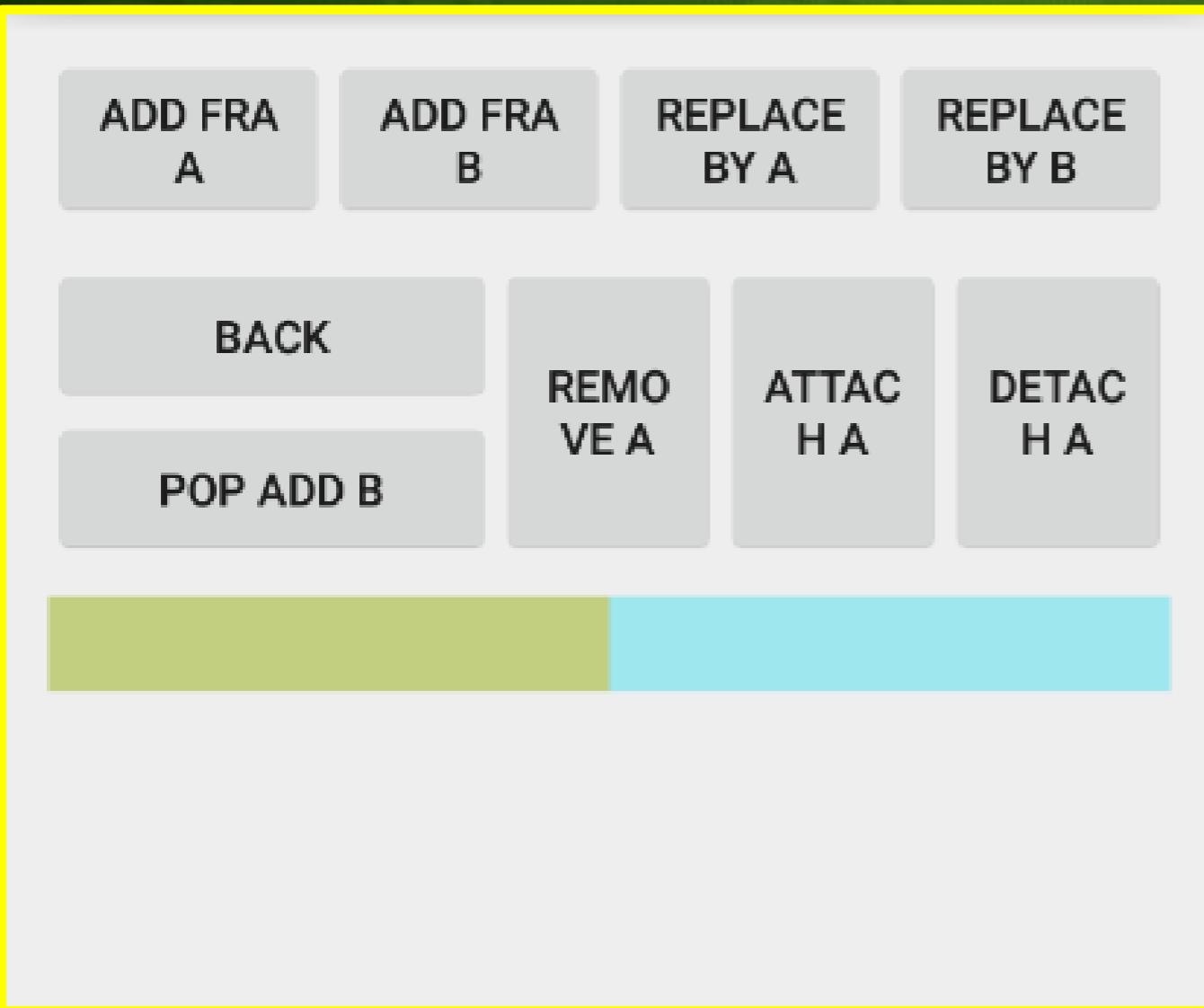
#4 Fragment - Communicator (3)



#5 Fragment - onSaveInstanceState



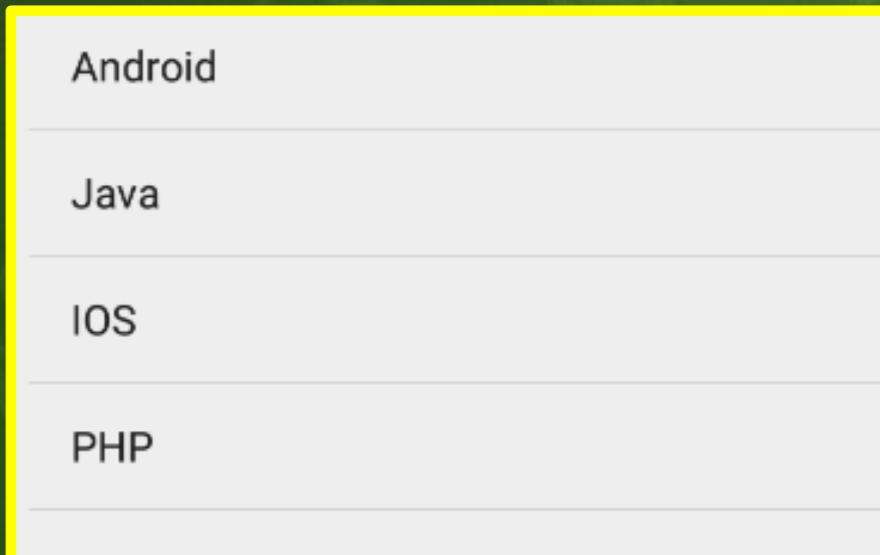
#6 Fragment - Back Stack



- *addToBackStack*
- *onBackStackChanged*
- *popBackStack*

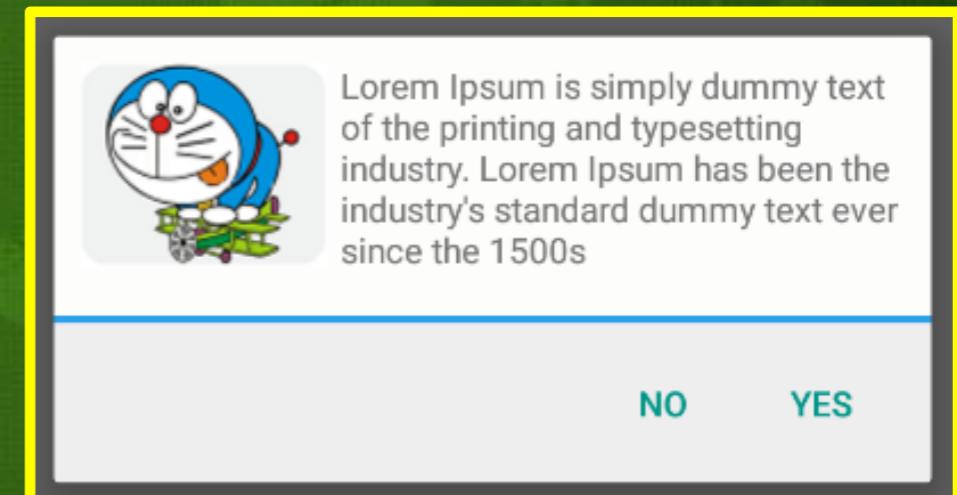
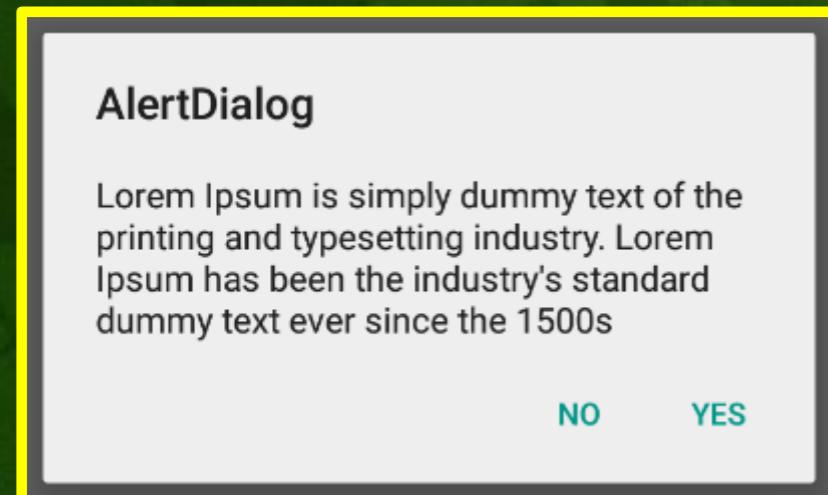
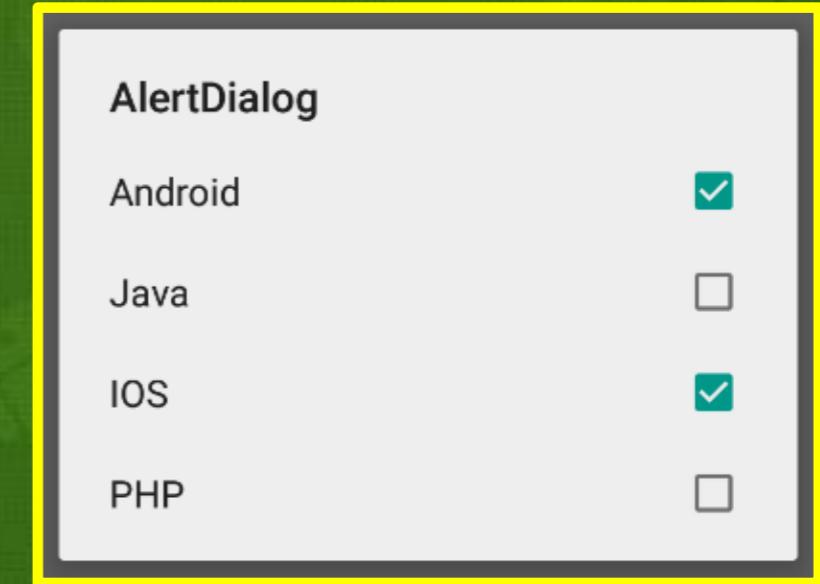
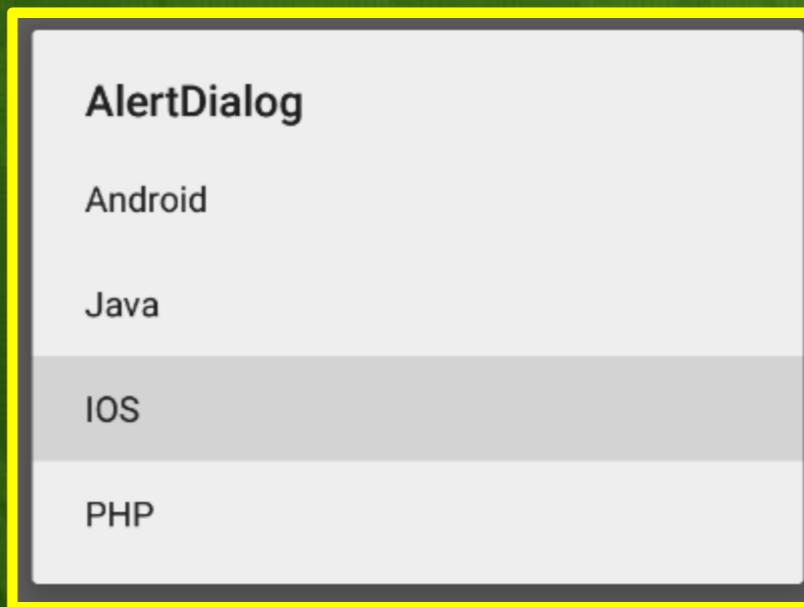
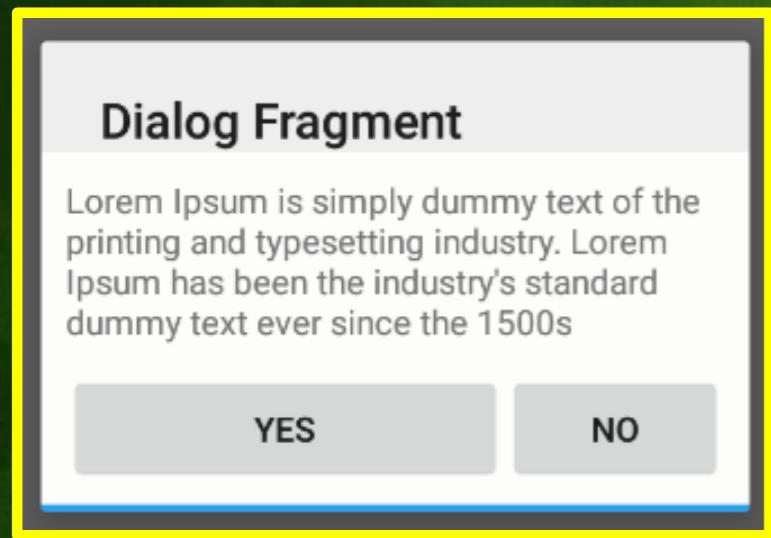
#7 Sử dụng List Fragment

- *Hiển thị dữ liệu dạng danh sách*
- *onActivityCreated để khai báo adapter*
- *onCreateView chỉ định phần layout*



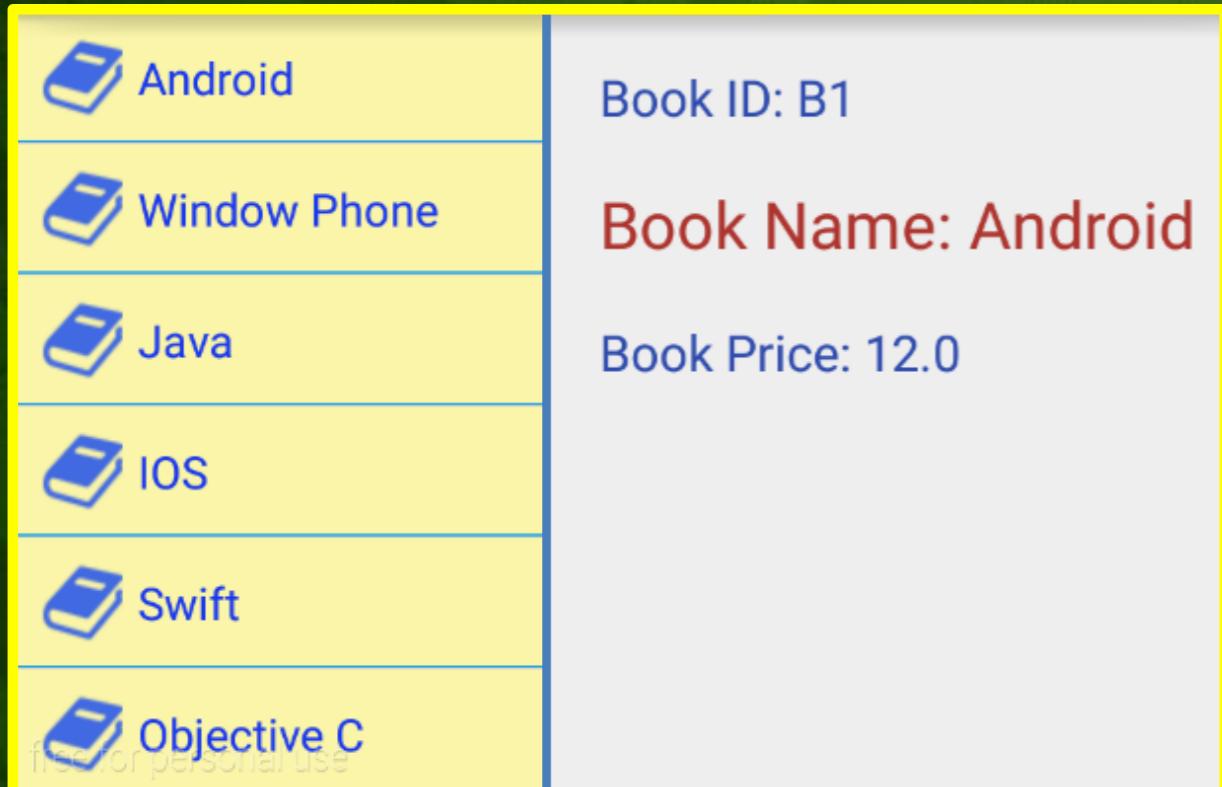
	Android	12.0
	Window Phone	13.0
	Java	11.0
	IOS	10.0
	Swift	5.0
	Objective C	6.0
<p>- ID:B5 - Name:Swift - Price:5.0</p>		

#8 Sử dụng DialogFragment



#9 Tình huống xử lý giao diện với Fragment (1)

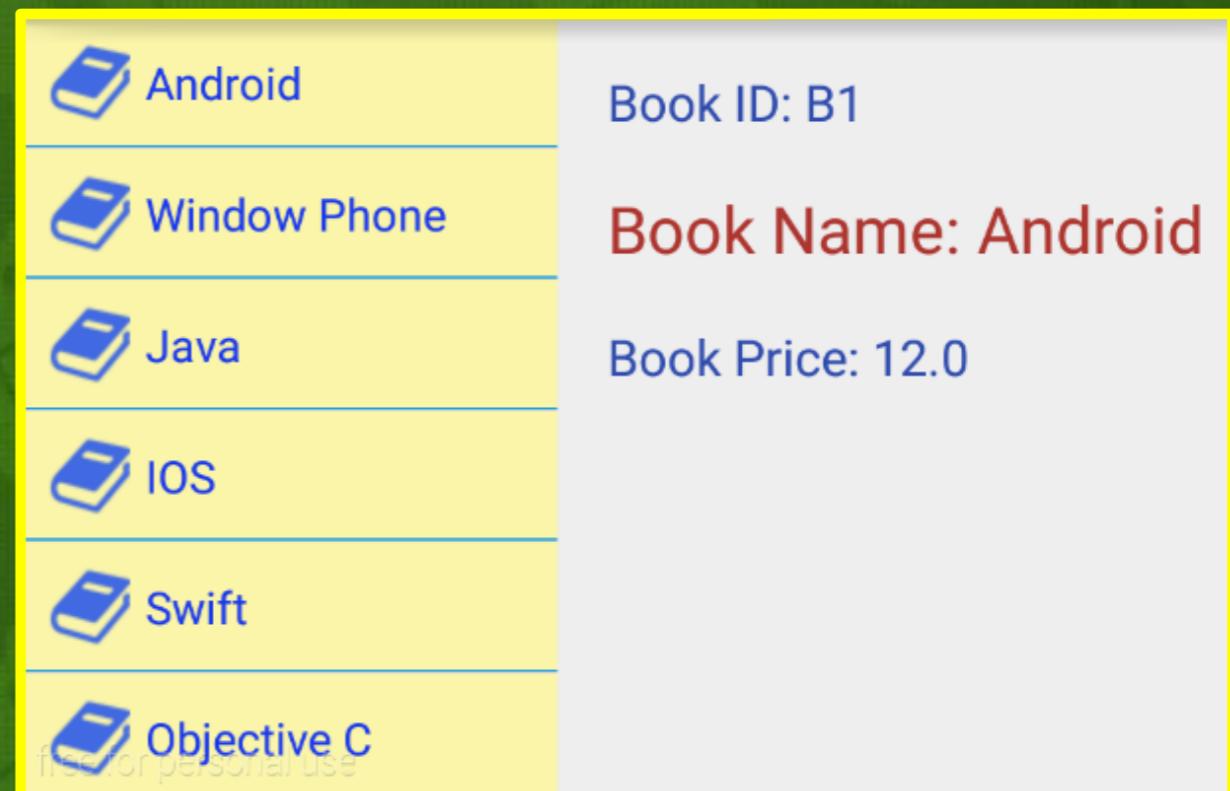
Orientation: Portrait



MainActivity

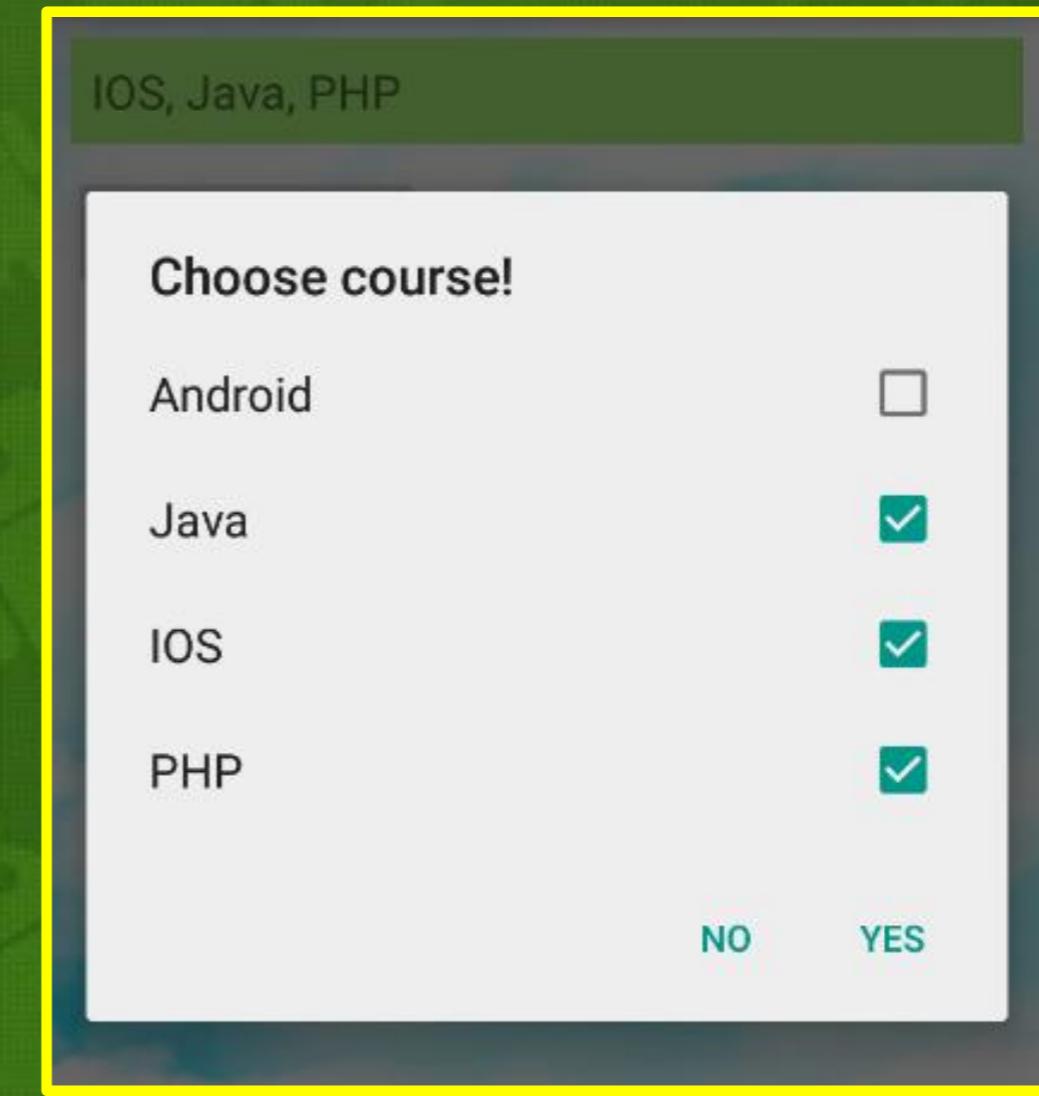
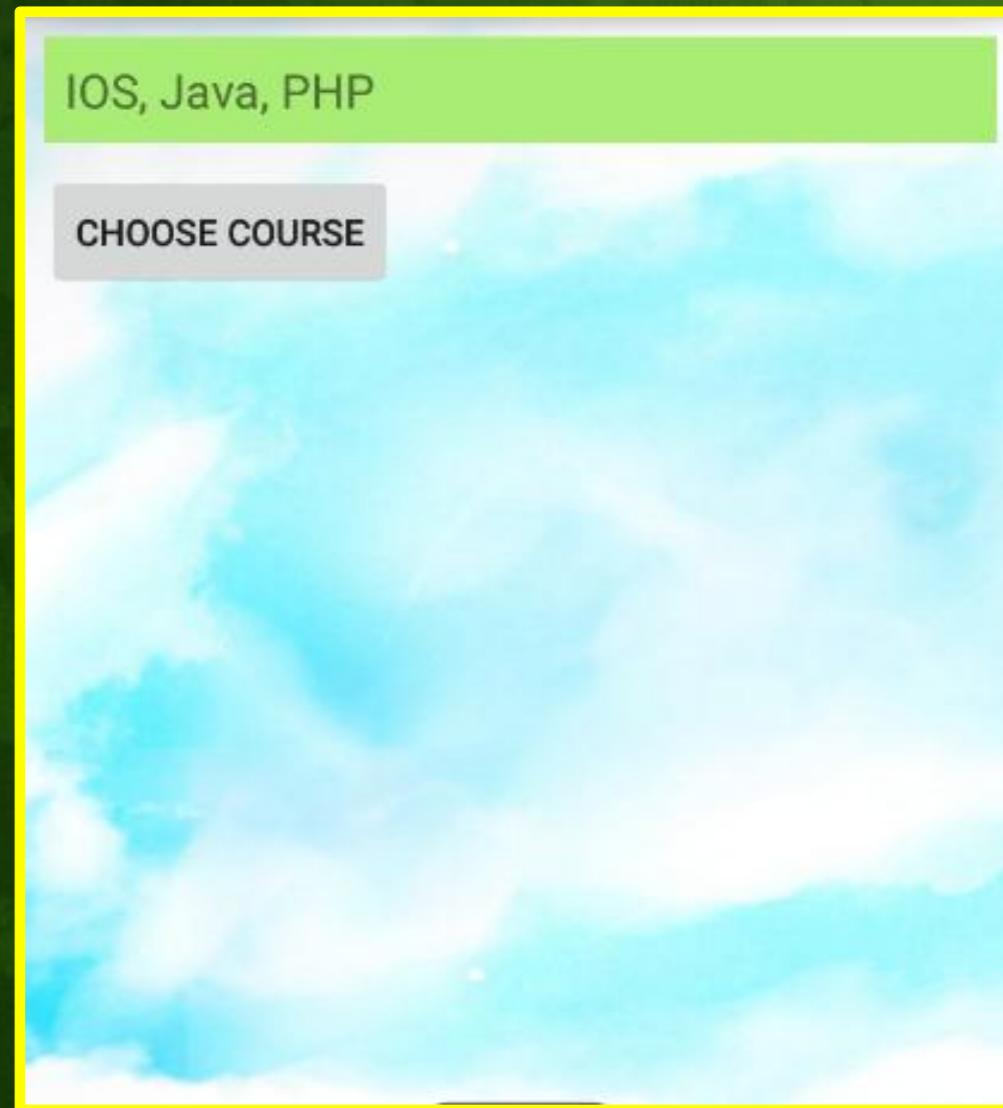
InfoActivity

Orientation: Landscape



MainActivity

#9 Tình huống xử lý giao diện với Fragment (2)



Chương 11

Thao tác với tập tin trong Android

1. *Xử lý tập tin trong Android*
2. *Internal Storage*
3. *Thao tác với các tập tin Cache*
4. *External Storage*
5. *Shared Preferences*
6. *Xây dựng ứng dụng*

#1 Xử lý tập tin trong Android (1)

Reference: <http://developer.android.com/intl/vi/training/basics/data-storage/index.html>

- *Android sử dụng File API và InputStream để đọc và ghi dữ liệu*
- *Thông thường các thiết bị Android sẽ có 2 vùng lưu trữ dữ liệu: Internal Storage và External Storage*
- *Các ứng dụng được thiết lập mặc định cài đặt ở bộ nhớ trong, thay đổi bằng thiết lập android:installLocation trong AndroidManifest*

#1 Xử lý tập tin trong Android (2)

<i>Internal Storage</i>	<i>External Storage</i>
<i>Luôn hỗ trợ sẵn</i>	<i>Không ở trạng thái sẵn sàng</i>
<i>Các tập tin được lưu ở đây được truy cập mặc định chỉ bởi ứng dụng</i>	<i>Các tập tin được lưu ở đây có thể được đọc bên ngoài tầm kiểm soát của ứng dụng</i>
<i>Uninstall → các file bị xóa</i>	<i>Uninstall → các file bị xóa nếu được đặt ở getExternalFilesDir()</i>
<i>Lưu trữ tập tin có dung lượng nhỏ</i>	<i>Lưu trữ tập tin có dung lượng lớn</i>

#1 Xử lý tập tin trong Android (3)

Các quyền hạn thường sử dụng:

<uses-permission android:name=“android.permission.XXX”

WRITE_EXTERNAL_STORAGE

quyền ghi dữ liệu với bộ nhớ ngoài

READ_EXTERNAL_STORAGE

quyền đọc dữ liệu với bộ nhớ ngoài

#1 Xử lý tập tin trong Android (4)

File file = new File(pathToFile)

file.createNewFile();

file.getParent()

file.isFile()

file.canRead()

file.isDirectory()

file.canWrite()

file.listFiles()

file.delete()

file.length()

file.exists()

#2 - Internal Storage (1)

- *Internal Storage* Khi một ứng dụng được cài đặt, hệ thống sẽ cung cấp 1 thư mục lưu trữ dữ liệu riêng biệt cho ứng dụng, khi ứng dụng bị xóa thì vùng lưu trữ này sẽ bị xóa theo!
- Vị trí `data/data/packagename/files`
- Không gian lưu trữ của Internal tương đối thấp (2Mb)
- Truy xuất các tập tin trong Internal thông qua `java.io: deleteFile(), fileList(), openFileInput(), openFileOutput(), ...`

#2 - Internal Storage (2)

/data/data/me.android.training.medata/files

WRITE

READ

CLEAR

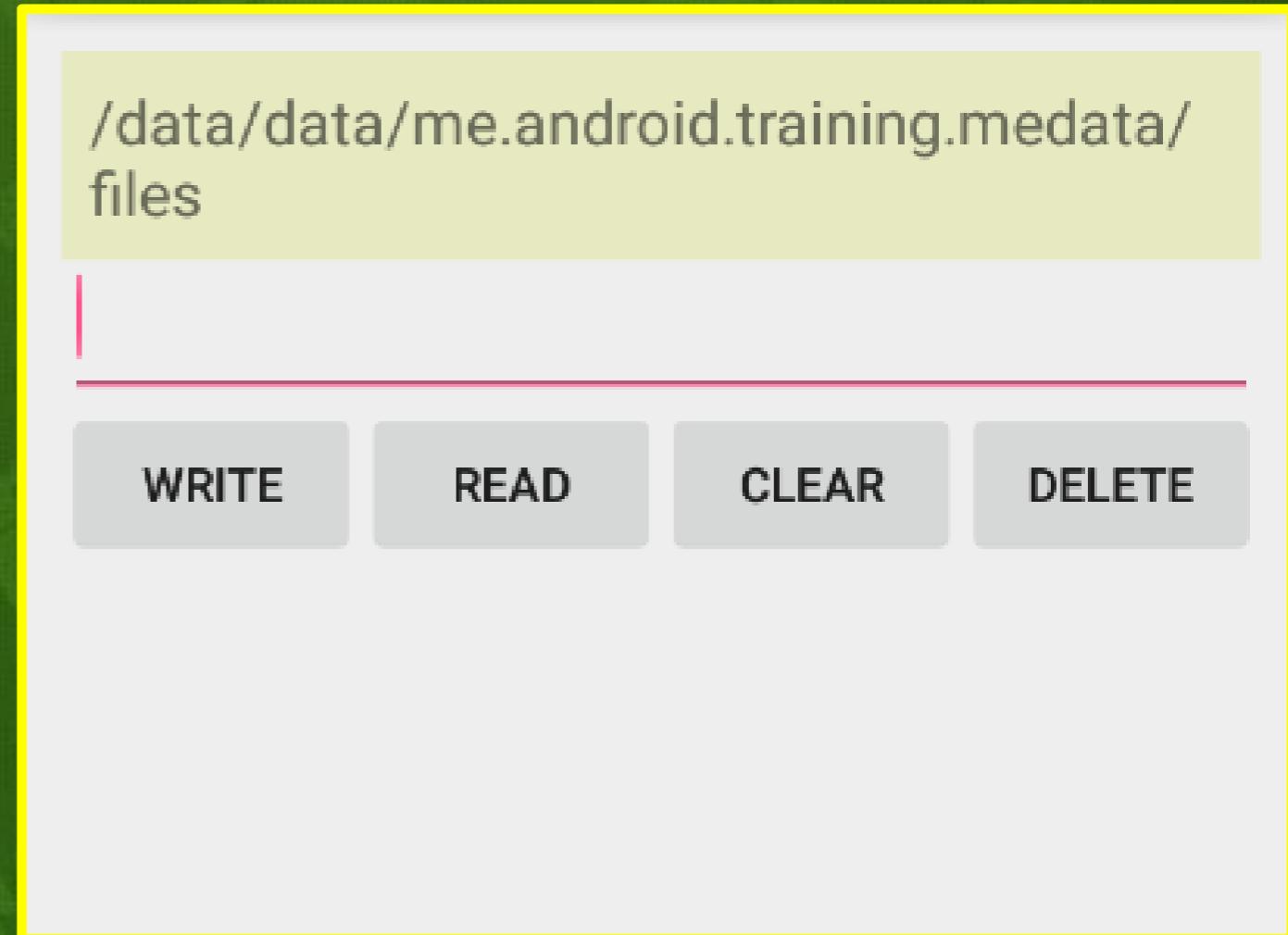
DELETE

#3 – Thao tác với các tập tin Cache

Dung lượng tối đa của file cache là
1MB

Khi gỡ bỏ ứng dụng thì các file cache sẽ
được tự động xóa

Keyword: `getCacheDir`



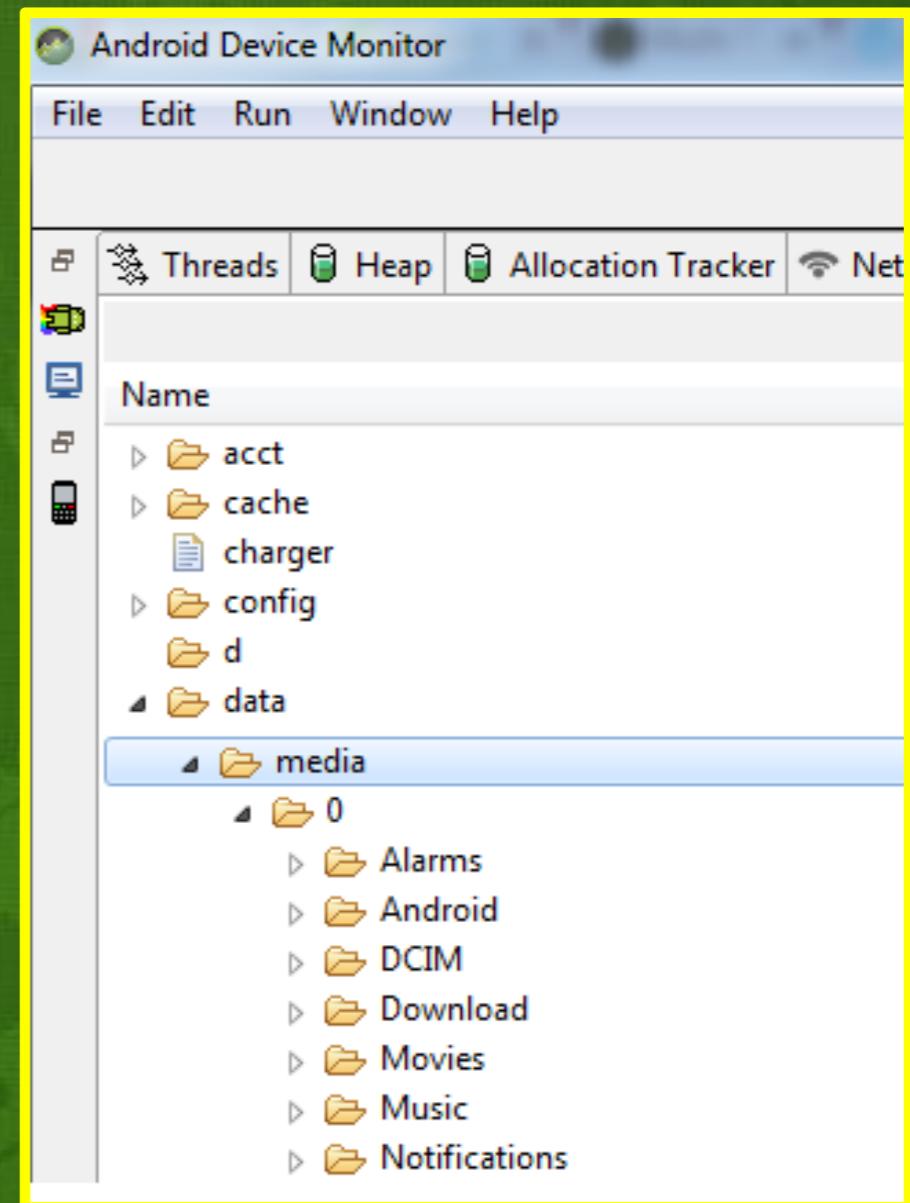
#4 - External Storage

Trước khi làm việc với bộ nhớ ngoài cần kiểm tra trạng thái của nó

Keyword: *Enviroment, getExternalStorageState(),
MEDIA_MOUNTED,
MEDIA_MOUNTED_READ_ONLY*

Xin quyền ghi dữ liệu *WRITE_EXTERNAL_STORAGE*

Xin quyền đọc dữ liệu *READ_EXTERNAL_STORAGE*



#5 SharedPreferences

SharedPreferences lưu trữ và truy xuất dữ liệu trong tập tin XML, chỉ hỗ trợ các kiểu cơ bản như boolean, float, int, long, string, ... (uses: update, credentials, settings, caching, ...)

Keyword: SharedPreferences, SharedPreferences.Editor, putXXX, getPreferences(), getXXX, getSharedPreferences(), edit(), commit(), ...

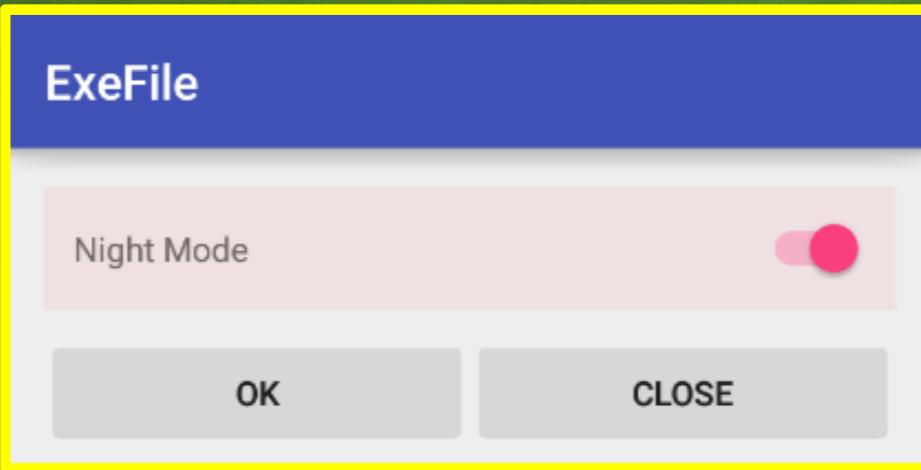
```
<?xml version='1.0' encoding='utf-8' standalone='yes' ?>
<map>
    <int name="cost" value="12" />
    <string name="name">Android</string>
</map>
```

#6 Xây dựng ứng dụng

ExeFile

Setting

A store owner was tacking a sign above his door that read Puppies for Sale. Signs have a way of attracting children, and soon a little boy appeared at the store and asked, How much are you gonna sell those puppies for? The store owner replied, Anywhere from \$30 to \$50



ExeFile

⋮

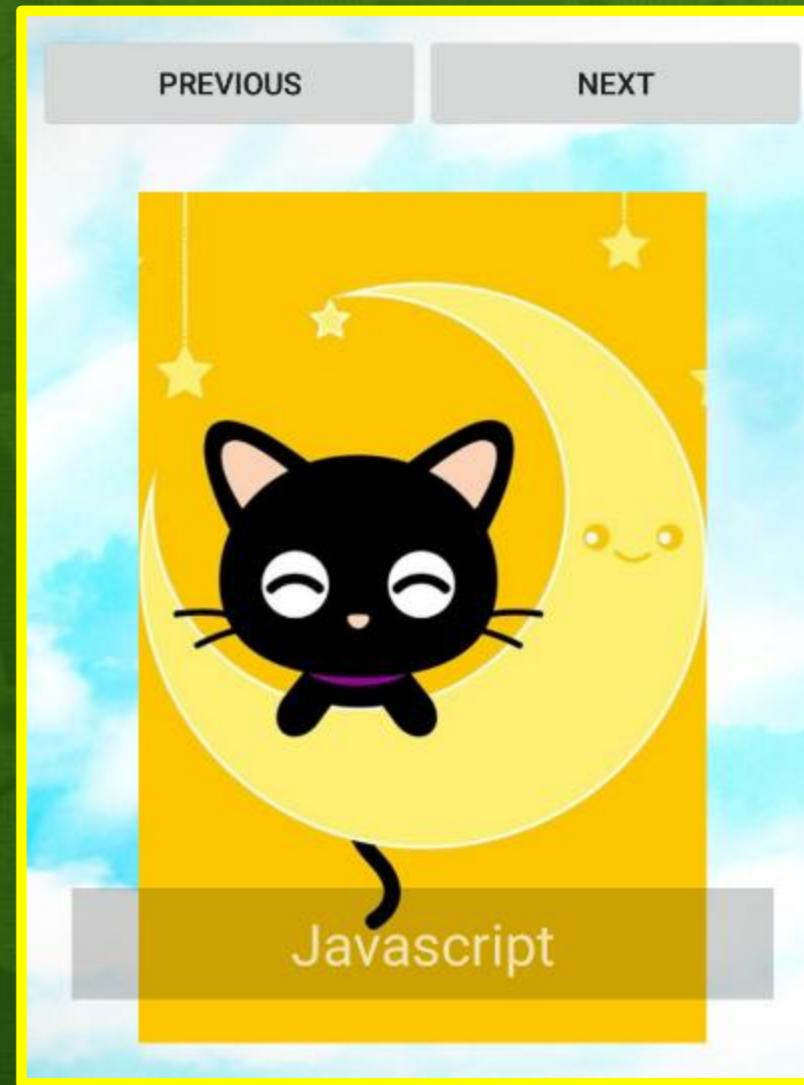
A store owner was tacking a sign above his door that read Puppies for Sale. Signs have a way of attracting children, and soon a little boy appeared at the store and asked, How much are you gonna sell those puppies for? The store owner replied, Anywhere from \$30 to \$50

Chương 12

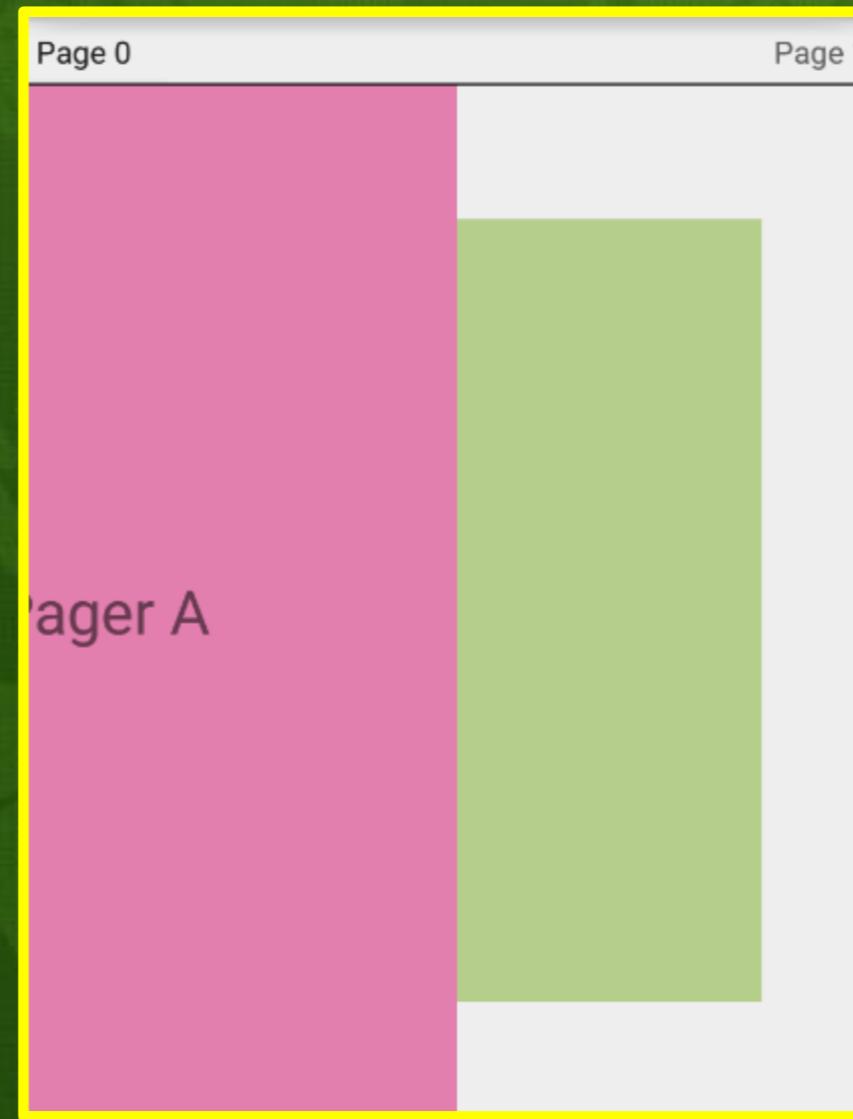
Sử dụng ViewPager

1. *ViewPager* với ứng dụng *Slider*
2. *ViewPager* kết hợp *Fragment*
3. *ViewPager* kết hợp *TabHost*
4. *Custom hiệu ứng* với *PageTransformer*

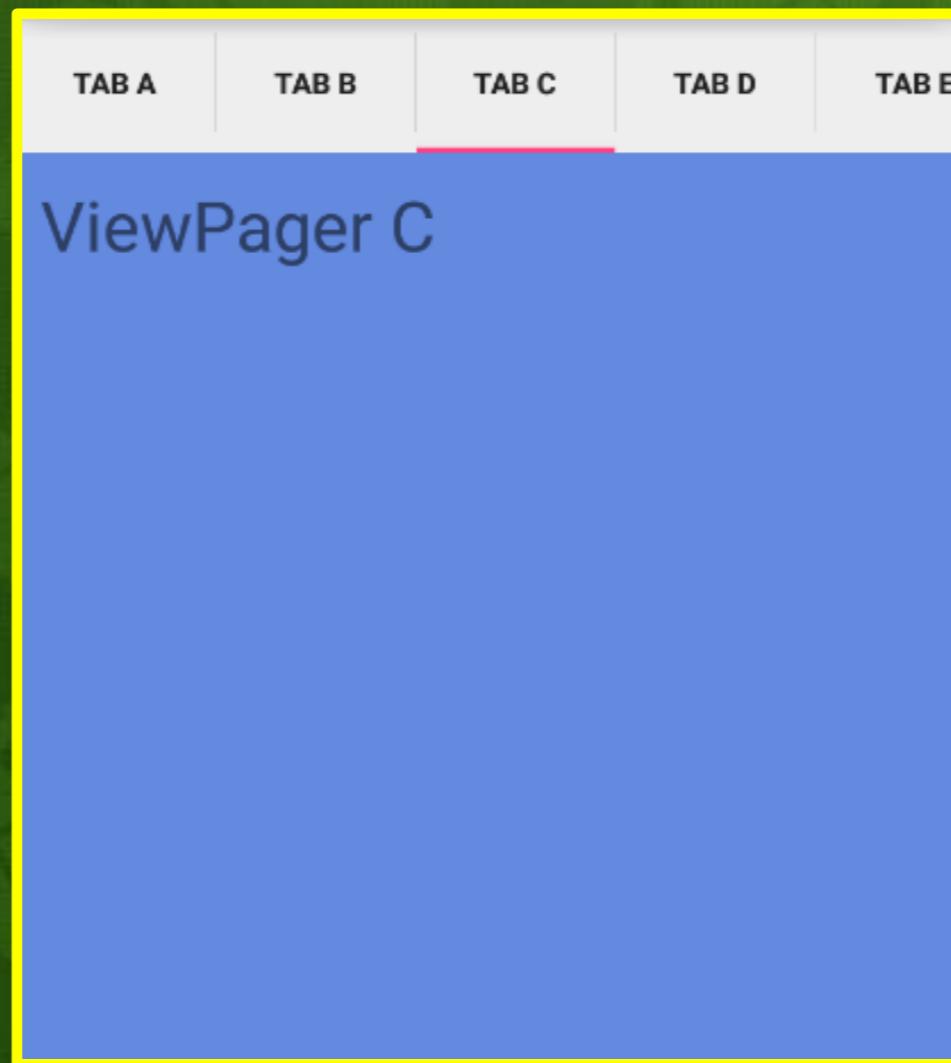
#1 ViewPager với ứng dụng Slider



#2 ViewPager kết hợp Fragment



#4 ViewPager kết hợp TabHost



#5 Custom hiệu ứng với PageTransformer

1. *implements ViewPager.PageTransformer*
2. *override transformPage(View view, float position)*
 - *position < -1 This page is way off-screen to the left*
 - *position <= 0 Use the default slide transition when moving to the left page*
 - *position <= 1 Fade the page out.*
 - *position > 1 This page is way off-screen to the right.*
3. *setPageTransformer*

Chương 13

Cơ sở dữ liệu SQLite trong Android

1. *SQLite và công cụ quản lý*
2. *SQLite CRUD*
3. *Xây dựng ứng dụng quản lý với SQLite*
4. *Master selection trong SQLite*
5. *Sử dụng Content Provider phổ biến*
6. *Xây dựng Content Provider*

#1 SQLite và công cụ quản lý (1)

SQLite là hệ thống cơ sở dữ liệu quan hệ nhỏ gọn, hoàn chỉnh, có thể cài đặt bên trong các trình ứng dụng khác (hỗ trợ cú pháp truy vấn như SQL) → xem chương 7 Ngôn ngữ SQL khóa học PHP

SQLite hỗ trợ 3 kiểu dữ liệu cơ bản: Text, Integer, Real

SQLite được tích hợp sẵn trong Android dưới dạng một tập tin và được đặt ở vị trí data/data/application_name/database/

#1 SQLite và công cụ quản lý (2)

1. *SQLite Manager (on Firefox)*

- *Download và cài đặt trên Firefox*
- *Các thao tác sử dụng cơ bản*

2. *SQLiteAdmin (on Window) <http://sqliteadmin.orbmw2k.de/>*

3. ...

#2 SQLite CRUD

- *SQLiteOpenHelper*
- *onCreate* tạo CSDL nếu như CSDL chưa tồn tại
- *onUpgrade* tạo CSDL nếu như CSDL có sự thay đổi về version
- *SQLiteDatabase insert()*, *update()*, *delete()*, *execSQL()*
- *getReadableDatabase* < *getWritableDatabase*: lấy đối tượng *SQLiteDatabase*
- *ContentValue* khai báo key/value cho các quá trình *insert*, *update* dữ liệu

#3 Xây dựng ứng dụng quản lý với SQLite

IOS

ADD

STT	Tên	Mô tả
1	Joomla 2.5	
2	Zend 2	
3	Android	
4	iOS	

#4 Master selection trong SQLite

- *rawQuery()* nhận một câu truy vấn từ SQL
- *query()* cung cấp cấu trúc để thực hiện câu truy vấn
- *SQLiteQueryBuilder*

#5 Sử dụng Content Provider (1)

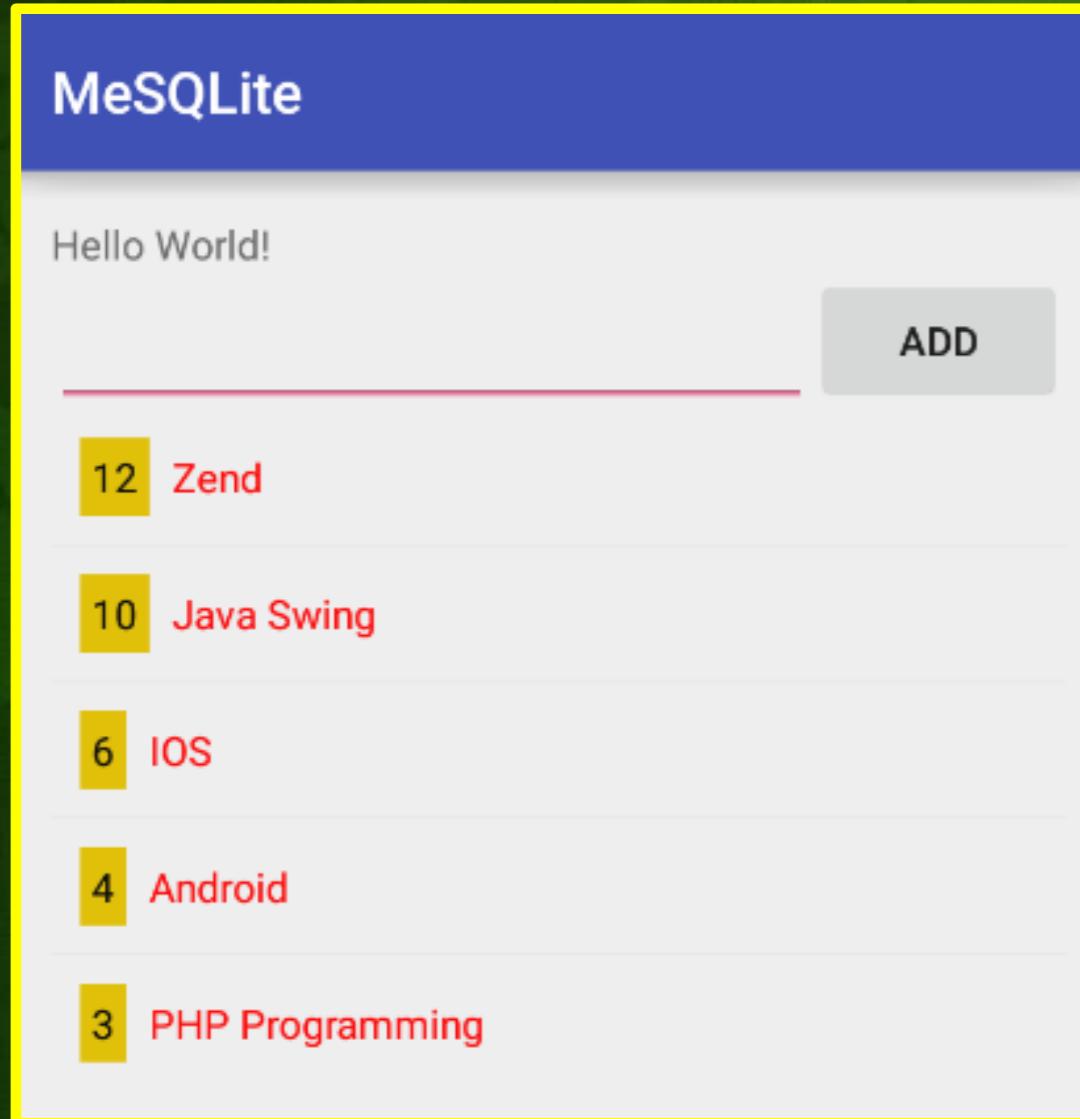
- *Content Provider cung cấp dữ liệu cho các ứng dụng bên ngoài*
- *Xây dựng ứng dụng có thể truy xuất đến danh sách contact, sms, browser, history, call log, ... → Content Provider*
- *Xây dựng ứng dụng quản lý lớp học cho phép chia sẻ thông tin lớp học đến các ứng dụng khác → Content Provider*
- *Content Resolver truy cập dữ liệu của các ứng dụng khác, Content Resolver hỗ trợ các phương thức insert(), update(), query()*

#5 Sử dụng Content Provider (2)

*Cursor cursor = getContentResolver().query(**uri**, projection, selection, selectionArgs, sortOrder), trong đó uri có dạng như sau [prefix]://[authority]/[data_type]/[id]*

- *Prefix* *content://*
- *Authority* *tên của ContentProvider cần truy xuất (package)*
- *data_type* *loại dữ liệu sẽ được truy xuất (table)*
- *Id* *chỉ định dữ liệu cần truy xuất*

#6 Xây dựng Content Provider



Chương 14

Xử lý đa luồng và khai thác dữ liệu Internet

1. Sử dụng *post* và *postDelayed* trong Handler
2. Xây dựng *ProgressBar* với Handler
3. Xây dựng Bộ đếm ngược với Handler
4. Xử lý đa luồng với *AsyncTask*
5. Download hình ảnh từ Internet
6. Đọc và sử dụng Regular Expression bóc tách nội dung từ Internet
7. Sử dụng BroadCast Receiver

#1 Multi thread (1)

Android hỗ trợ multi-thread (nhiều tiến trình chạy đồng thời) giúp ứng dụng chạy nhanh và hiệu quả hơn.

Ưu điểm: Mỗi luồng có thể dùng chung và chia sẻ nguồn tài nguyên trong quá trình chạy, nhưng có thể thực hiện một cách độc lập.

Nhược điểm: Càng nhiều luồng thì xử lý càng phức tạp, cần phát hiện tránh các luồng chết, luồng chạy mà không làm gì trong ứng

#1 Multi thread (2)

Trong Android sẽ phân thành 2 loại Thread:

- *Main Thread: tương tác được với giao diện người dùng*
- *Worker Thread: không tương tác được với giao diện người dùng*

Main Thread: Handler, AsyncTask, BroadCast Receiver, Service, ...

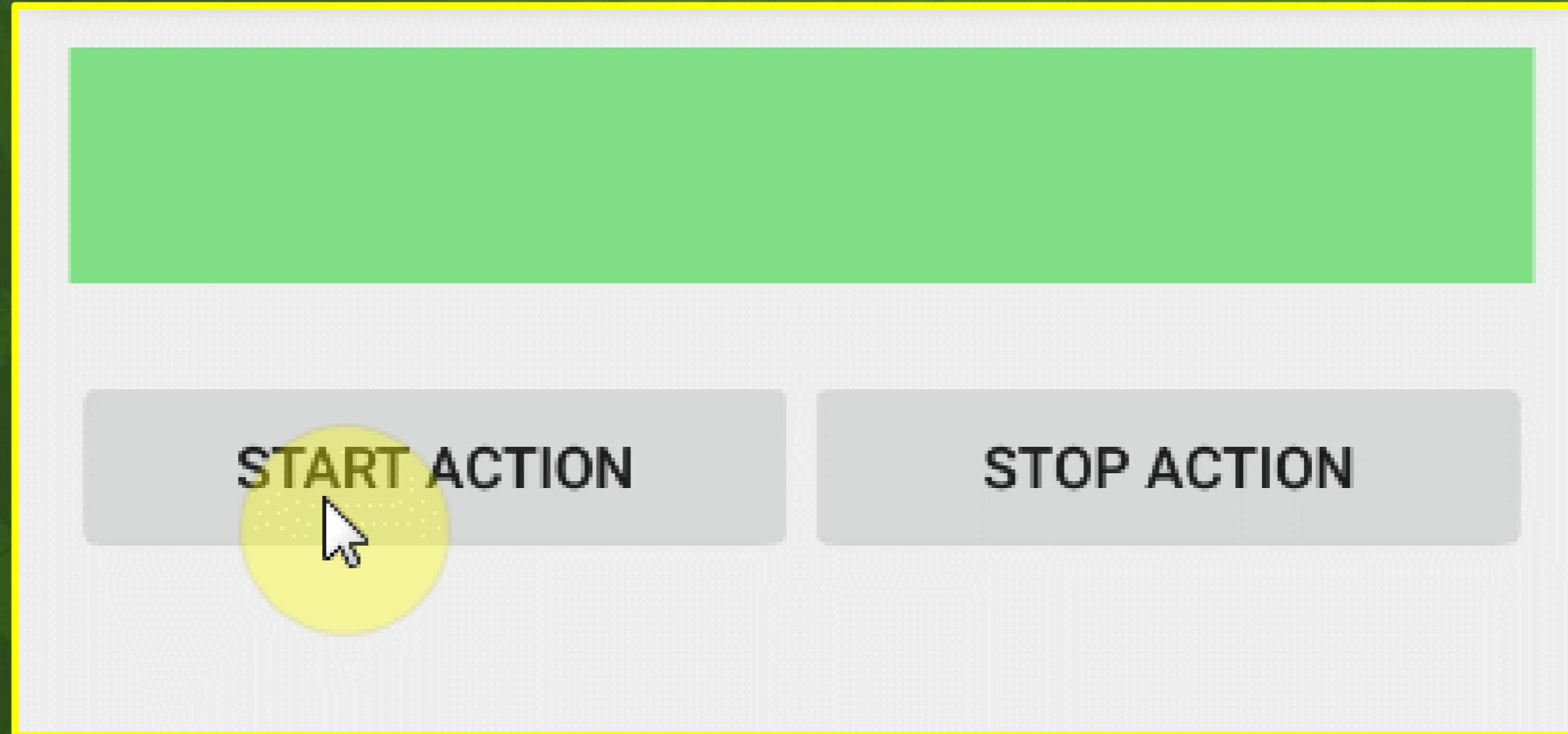
#1 Sử dụng post và postDelayed trong Handler (1)

Handler giao tiếp giữa 2 hay nhiều thread, ngoài ra Handler còn được sử dụng để xử lý 1 yêu cầu nào của người dùng sau 1 khoảng thời gian nhất định.

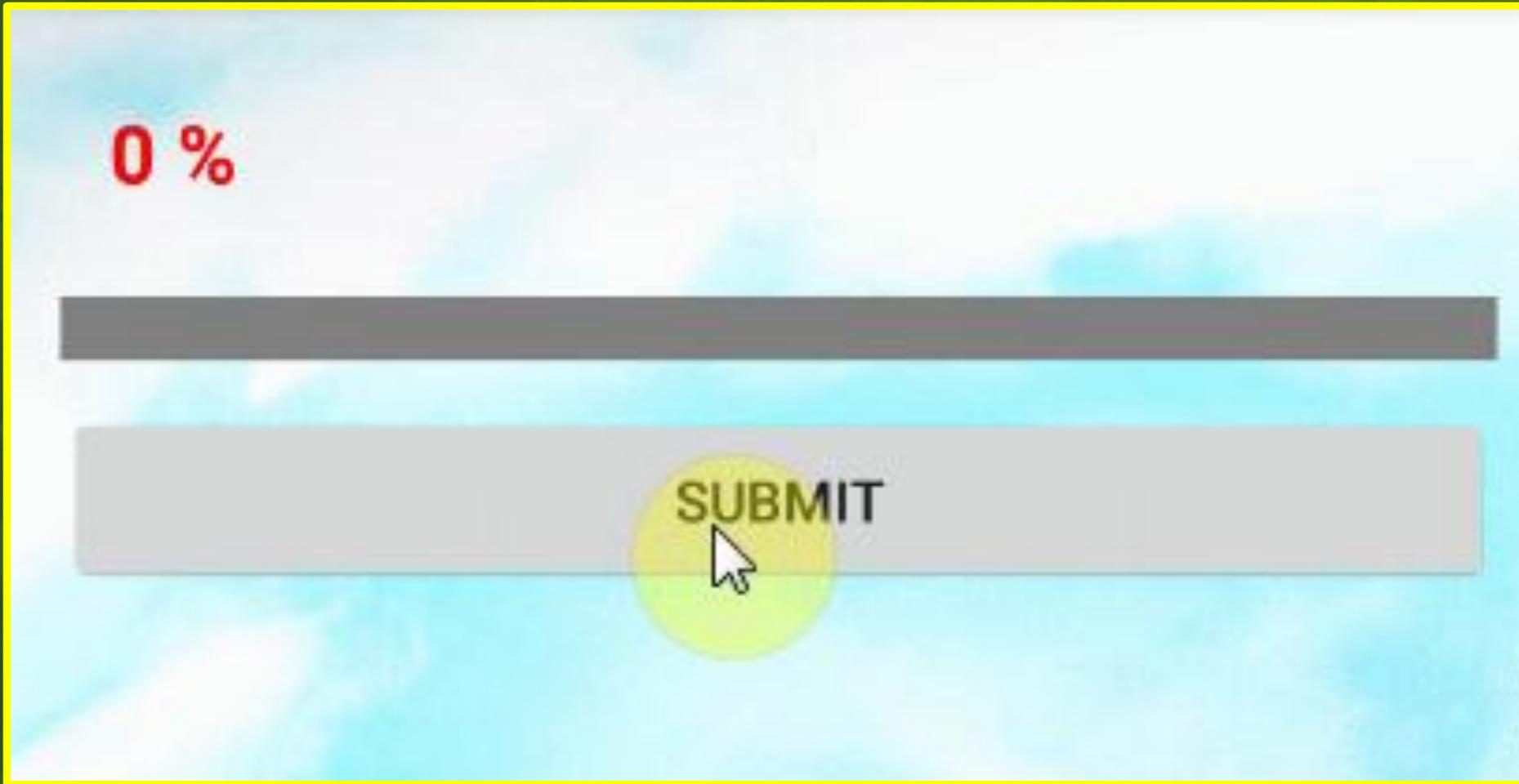
Tìm hiểu các phương thức thông dụng

- *post(Runnable)*
- *postDelay(Runnable, long)*

#1 Sử dụng post và postDelayed trong Handler (2)



#2 Xây dựng ProgressBar với Handler



#3 Xây dựng Bộ đếm ngược với Handler

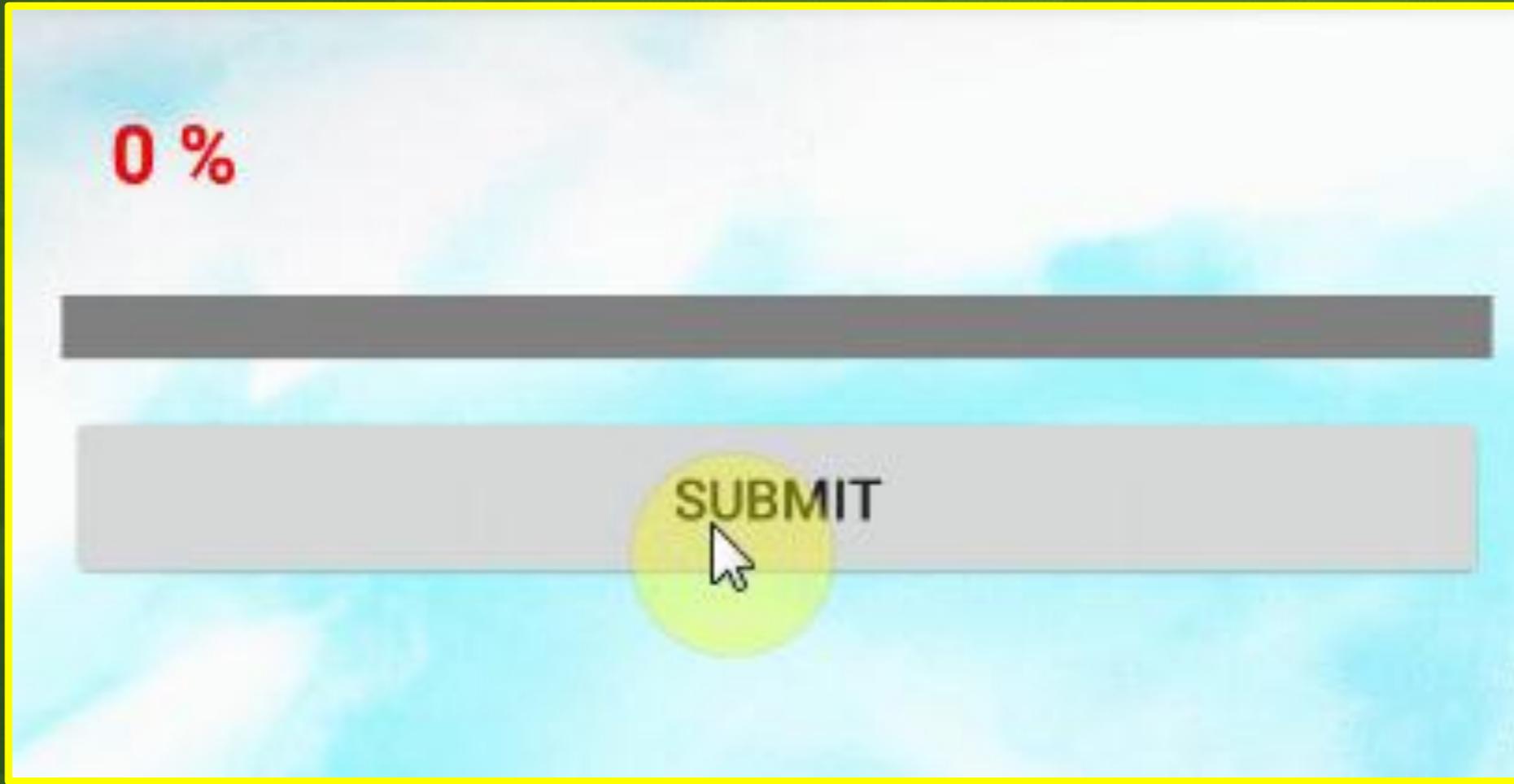


#4 Xử lý đa luồng với AsyncTask (1)

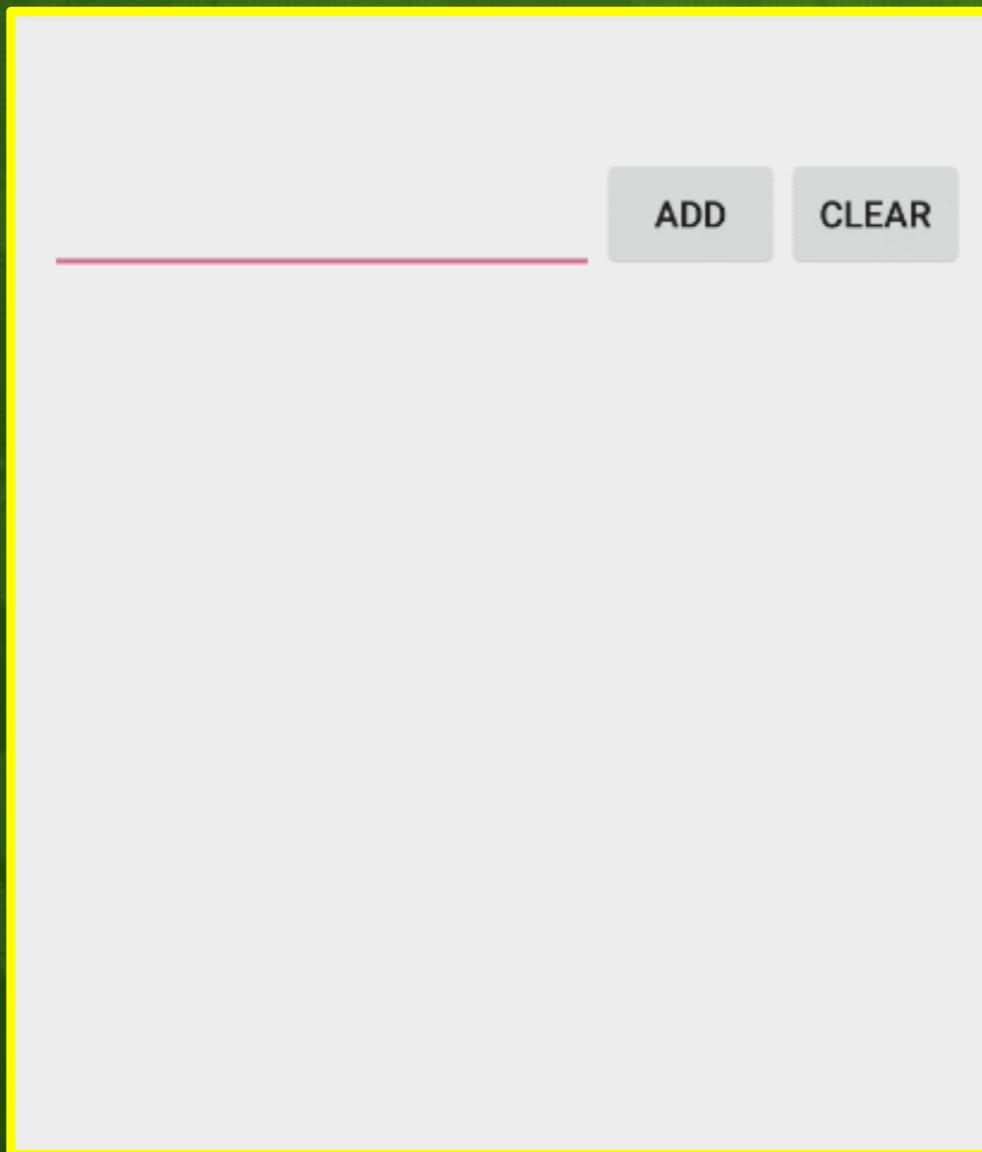
AsyncTask<Params, Progress, Result> 3 đối số là các Generic Type (AsyncTask UI Thread)

- *onPreExecute()* Tự động được gọi đầu tiên khi tiến trình được kích hoạt.
- *doInBackground(Params)* thực thi trong quá trình tiến trình chạy nền → *publishProgress* → *onProgressUpdate()* cập nhật giao diện. Note: không thể cập nhật giao diện trong *doInBackground()*.
- *onProgressUpdate (Progress)* cập nhật giao diện lúc runtime
- *onPostExecute(Result)*: xử lý kết quả trả về (xem hình vẽ)

#4 Xử lý đa luồng với AsyncTask (2)



#4 Xử lý đa luồng với AsyncTask (3)



#5 Download hình ảnh từ Internet

Xin quyền kết nối Internet

- *Android.permission.INTERNET*
- *Android.permission.ACCESS_NETWORK_STA
TE*

DOWNLOAD IMAGE

READ CONTENT



#6 Đọc và sử dụng Regular Expression bóc tách nội dung từ Internet

Xin quyền kết nối Internet

- *Android.permission.INTERNET*
- *Android.permission.ACCESS_NETWORK_ST
ATE*

100-388-580-400-468-300-250-218-313-626-456-
218-313-626-456-218-313-626-456-218-313-626-
456-218-313-626-456-218-313-626-656-300-250-
300-600-300-250-

DOWNLOAD IMAGE

READ CONTENT

#7 Sử dụng BroadCast Receiver

BroadCast Receiver nhận các lời gọi (các sự kiện, intent, ...) từ bên ngoài ứng dụng.

- *Extend BroadCastReceiver*
- *Override onReceive()*
- *Đăng ký AndroidManifest*

Exercise: Xây dựng BroadCastReceiver nhận các cuộc gọi, các tin nhắn, thay đổi ngôn ngữ và custom lại BroadCastReceiver

Chương 15

Thao tác với XML và Json trong Android

1. XML những vấn đề cần biết
2. Sử dụng XmlPullParser toàn tập
3. XmlPullParser đọc và phân tích RSS
4. Đọc, ghi và phân tách chuỗi Json
5. Khai thác JSON từ Internet
6. Sử dụng class Gson toàn tập

#1 Thao tác với tập tin XML

Kiến thức cần biết về XML: Khóa học Lập trình PHP – Chương 4 Làm việc với XML

*Các phương pháp Parse dữ liệu từ XML trong Android: DOM, SAX, XmlPullParser
(dễ sử dụng, tốn ít tài nguyên)*

#2 Thao tác với XmlPullParser (01)

```
1 <?xml version="1.0" encoding="UTF-8"?>
2
3 <book>
4   <title>Android Programming</title>
5   <author>Amazon</author>
6   <pages>500</pages>
7   <weight units="gram">400</weight>
8 </book>
```

READ

Android Programming - Amazon -
500 - 400 - gram

#2 Thao tác với XmlPullParser (02)

```
1  <?xml version="1.0" encoding="UTF-8"?>
2  <book_shop>
3      <book>
4          <title>PHP Programming</title>
5          <author>Amazon</author>
6          <pages>500</pages>
7          <weight units="gram">400</weight>
8      </book>
9
10     <book>
11         <title>Zend Framework</title>
12         <author>Aki</author>
13         <pages>400</pages>
14         <weight units="gram">800</weight>
15     </book>
16
17     <book>
18         <title>Joomla CMS</title>
19         <author>joomla.org</author>
20         <pages>800</pages>
21         <weight units="kg">1</weight>
22     </book>
23 </book_shop>
```

READ

PHP Programming

Amazon

500 pages 400 gram

Zend Framework

Aki

400 pages 800 gram

Joomla CMS

joomla.org

800 pages 1 kg

#3 XmlPullParser đọc và xử lý RSS (vnexpress.net)

```
▼<rss xmlns:slash="http://purl.org/rss/1.0/modules/slash/" version="2.0">
  ▼<channel>
    <title>Số hóa - VnExpress RSS</title>
    <description>VnExpress RSS</description>
    ▶<image>...</image>
    <pubDate>Sat, 02 Apr 2016 16:08:12 +0700</pubDate>
    <generator>VnExpress</generator>
    <link>http://vnexpress.net/rss/so-hoa.rss</link>
  ▼<item>
    ▶<title>...</title>
    ▶<description>...</description>
    <pubDate>Sat, 02 Apr 2016 09:37:45 +0700</pubDate>
    ▶<link>...</link>
    ▶<guid>...</guid>
    <slash:comments>0</slash:comments>
  </item>
  ▶<item>...</item>
  ▶<item>...</item>
  ▶<item>...</item>
  ▶<item>...</item>
```



Trò đùa Cá tháng tư của Google làm người dùng mất việc

Sat, 02 Apr 2016 09:37:45 +0700

Google sớm nay đã phải gỡ ứng dụng mang tên Mic Drop trong hòm thư Gmail vì gây phiền toái cho người sử dụng, mặc dù đây chỉ là trò đùa của hãng trong ngày Cá tháng tư.



Nokia sắp trở lại với smartphone chạy Android

Sat, 02 Apr 2016 11:00:00 +0700

Thương hiệu Phần Lan sắp kết thúc thời hạn cho Microsoft mượn thương hiệu và có thể trở lại thị trường điện thoại di động ngay năm nay với smartphone chạy Android.



Dưới 35 triệu đồng thì mua điện thoại gì?

Sat, 02 Apr 2016 10:00:00 +0700

Em được chị tặng cho khoản tiền, cộng với tiền để dành

#3 XmlPullParser đọc và xử lý RSS (vietnamnet)

```
▼<rss version="2.0">
  ▼<channel>
    <title>VietNamNet - Xã hội</title>
    <description>Chuyên mục Xã hội - Báo VietNamNet</description>
    <link>http://vietnamnet.vn</link>
    <copyright>© VietNamNet - Báo VietNamNet</copyright>
    <generator>VietNamNet.CMS.UI.Generator - version 1.5</generator>
  ▼<item>
    <title>11 đập thủy điện dồn đại họa cho ĐBSCL</title>
    ▶<description>...</description>
    ▶<link>...</link>
    <pubDate>01/04/2016 (GMT+7)</pubDate>
    ▶<image>...</image>
  </item>
  ▶<item>...</item>
  ▶<item>...</item>
  ▶<item>...</item>
```



11 đập thủy điện dồn đại họa cho ĐBSCL

01/04/2016 (GMT+7)

Trong khi hạn mặn chưa kịp khắc phục, các chuyên gia lo ngại 11 dự án thủy điện ở hạ lưu sông Mekong của Lào và Campuchia sẽ khiến ĐBSCL gặp "đại họa".



'Khắc tinh' khiến giang hồ đất Cảng rùng mình

02/04/2016 (GMT+7)

Quá trình đấu tranh với tội phạm ở đâu cũng gian nan, đặc biệt là ở đất Cảng. Đã có những 'khắc tinh' khiến giới giang hồ mới nghe đã thấy rùng mình.



Bắt 2 đối tượng xưng phóng viên tổng tiền doanh nghiệp

02/04/2016 (GMT+7)

#4 Json và phân tách chuỗi Json (1)

Kiến thức cần biết về Json: Khóa học Lập trình PHP – Chương 9 – 9.1 Json

2 class thường được sử dụng để làm việc với Json:

- *JSONObject*
- *JSONArray*

Các phương thức: `getString()`, `getInt()`, `getDouble()`, `getJSONArray()`,`getJSONObject()`

#4 Json và phân tách chuỗi Json (3)

```
[ {  
    "book": {  
        "title": "Android Programming",  
        "author": "Amazon",  
        "pages": 500,  
        "weight": {  
            "units": "gram",  
            "value": 400  
        }  
    }  
}
```

READ

**Android Programming - Amazon -
500 - 400 - gram**

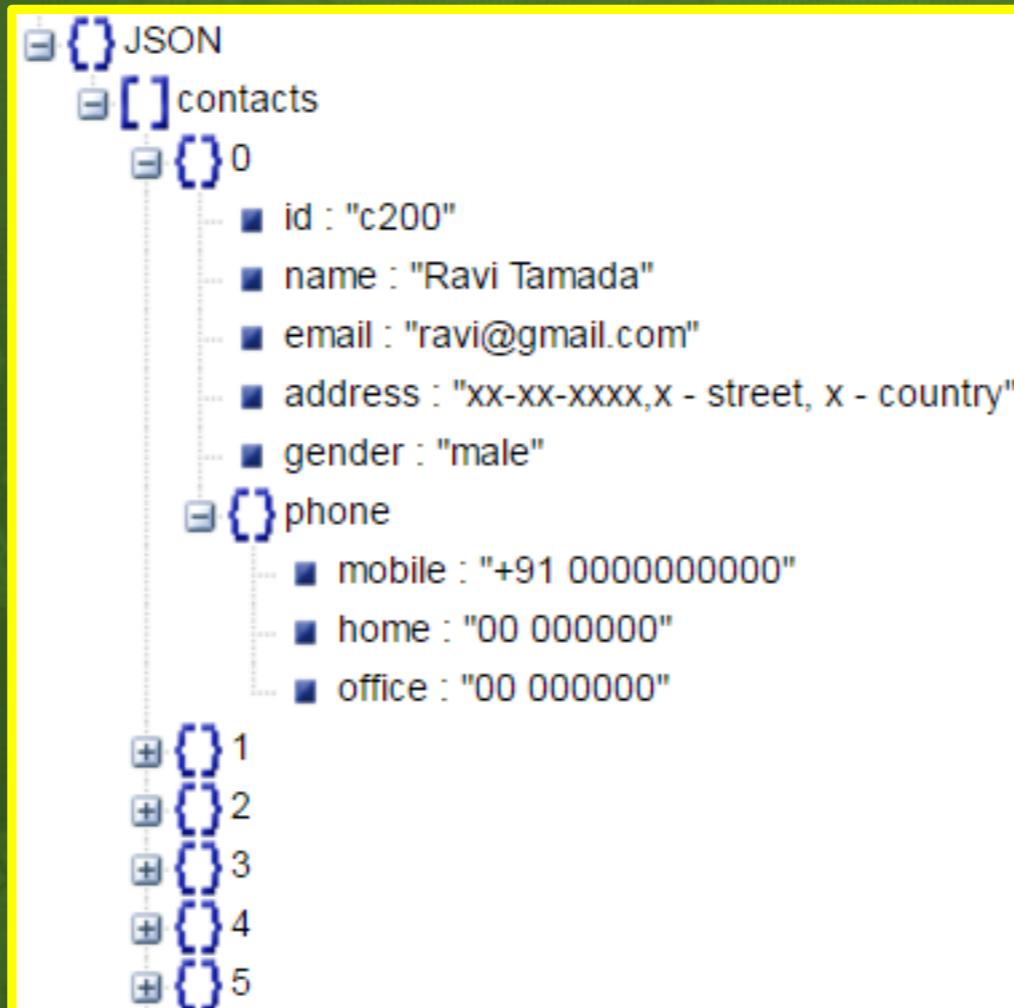
#4 Json và phân tách chuỗi Json (3)

```
{ □  
  "book": [ □  
    { □  
      "title": "PHP Programming",  
      "author": "Amazon",  
      "pages": 500,  
      "weight": { □  
        "units": "gram",  
        "value": 400  
      }  
    },  
    { □  
      "title": "Zend Framework",  
      "author": "Aki",  
      "pages": 400,  
      "weight": { □  
        "units": "gram",  
        "value": 800  
      }  
    },  
    { □ }  
  ]  
}
```

READ

PHP Programming	Amazon
500 pages	400 gram
<hr/>	
Zend Framework	Aki
400 pages	800 gram
<hr/>	
Joomla CMS	joomla.org
800 pages	1 kg

#5 Khai thác JSON từ Internet



#6 Sử dụng class Gson toàn tập

