1. Android là gì?

Android là 1 hệ điều hành mã nguồn mở dựa trên nền tảng linux và dùng để phát triển ứng dụng trên các thiết bị thông minh(Điện thoại, Tivi, Đồng hồ...)

1. Em có biết lịch sử hình thành của Android?

* Công ty Android được thành lập 10/2003 tại Palo Alto-California, dưới sự trợ cấp vốn của google.
* 8/2005 do khó khăn tài chính nên google mua lại công ty android và các thành viên của cty vẫn làm việc cho google.
* Năm 2007 liên minh mã nguồn mở ra đời gồm 84 công ty lớn trong lĩnh vực phần cứng, phần mềm, nhà mạng , và cùng năm đó SDK 1.0 ra đời.
* 10/2008 chiếc điện thoại đầu tiên chạy android T-Mobile G1(HTC Dream) ra đời

1. Android viết bằng ngôn ngữ nào? Mối liên hệ giữa chúng.

* Android có thể viết bằng Java,C++,Kotlin..
* Java là ngôn ngữ hướng đối tượng dễ học dễ viết. Giúp chúng ta dễ dàng tiếp cận Android.
* C/C++ gần với ngôn ngữ máy và dùng để tương tác trực tiếp với phần cứng.
* Kotlin ra đời năm 2011 và 5/2017 đc google chính thức công bố là ngôn ngữ lập trình cho Android.

Java và C++ tương tác với nhau thông qua JNI.

Kotlin ra đời để trợ giúp Java. Nó viết nhanh gọn hơn.. Java và Kotlin cũng có thể gọi lẫn nhau

1. SDK là gì?

SDK – Software Development Kit là một tập hợp công cụ hỗ trợ cho việc phát triển phần mềm thông qua một nền tảng nào đó.

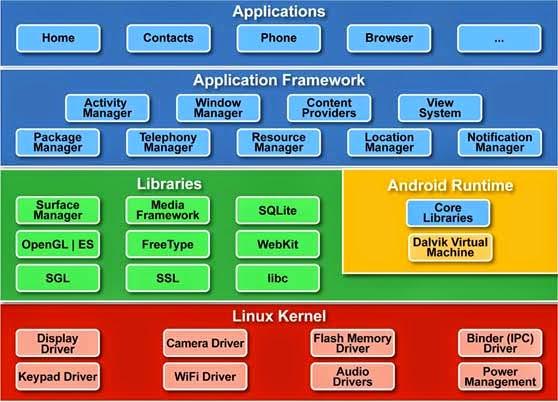
https://dammecode.wordpress.com/2018/03/16/sdk-la-gi-2/

1. Các Version của Android.?

1.0 ->8.0 các tên của nó đều mang tên các loại bánh và google có 1 phòng trưng bày các loại bánh tượng trưng cho các version. Mỗi version của android đều mang đến những cập nhật và tính năng mới đáng kể và hữu ích cho lập trình viên,

1. Kiến trúc của HĐH Android như thế nào?

Android gồm 4 tầng chính và chia thành 5 phần:



* Nhân Linux: Đây là nhân nền tảng mà hệ điều hành Android dựa vào nó để phát triển. Đâu là lớp chứa tất cả các thiết bị giao tiếp ở mức thấp dùng để điều khiển các phần cứng khác trên thiết bị Android.
* Thư viện: Chứa tất cả các mã cái mà cung cấp cấp những tính năng chính của hệ điều hành Android, đôi với ví dụ này thì SQLite là thư viện cung cấp việc hộ trợ làm việc với database dùng để chứa dữ liệu. Hoặc Webkit là thư viện cung cấp những tính năng cho trình duyệt Web.
* Android runtime: Là tầng cùng với lớp thư viện Android runtime cung cấp một tập các thư viện cốt lỗi để cho phép các lập trình viên phát triển viết ứng dụng bằng việc sử dụng ngôn ngữ lập trình Java. Android Runtime bao gốm máy ảo Dalvik(ở các version < 4.4, hiện tài là phiên bản máy ảo ART được cho là mạnh mẽ hơn trong việc xử lý biên dịch). Là cái để điều khiển mọi hoạt động của ứng dụng Android chạy trên nó(máy ảo Dalvik sẽ biên dịch ứng dụng để nó có thể chạy(thực thi) được , tương tự như các ứng dụng được biên dịch trên máy ảo Java vậy). Ngoài ra máy ảo còn giúp tối ưu năng lượng pin cũng như CPU của thiết bị Android
* Android framework: Là phần thể hiện các khả năng khác nhau của Android(kết nối, thông báo, truy xuất dữ liệu) cho nhà phát triển ứng dụng, chúng có thể được tạo ra để sử dụng trong các ứng dụng của họ.
* Application: Tầng ứng dụng là tầng bạn có thể tìm thấy chuyển các thiết bị Android như Contact, trình duyệt…Và mọi ứng dụng bạn viết đều nằm trên tầng này.

1. Các thành phần cơ bản của 1 ứng dụng android:

Android có 4 component chính:

* Activities : là giao diện có thể nhân tương tác từ người dùng, là thành phần quan trọng của android
* Service: là thành phần giúp ứng dụng có thể chạy ngầm mà không cần giao diện hiển thị tới người dùng
* Content provider: quản lý tập hợp các dữ liệu được chia sẽ bởi ứng dụng
* BroadCast receive: là thành phần cho phép nhân và gửi thông điệp trong ứng dụng, giữa các ứng dụng với nhau và giữa ứng dụng với system

1. Activities là gì? Có những cách nào truyền dữ liệu giữa các activity.

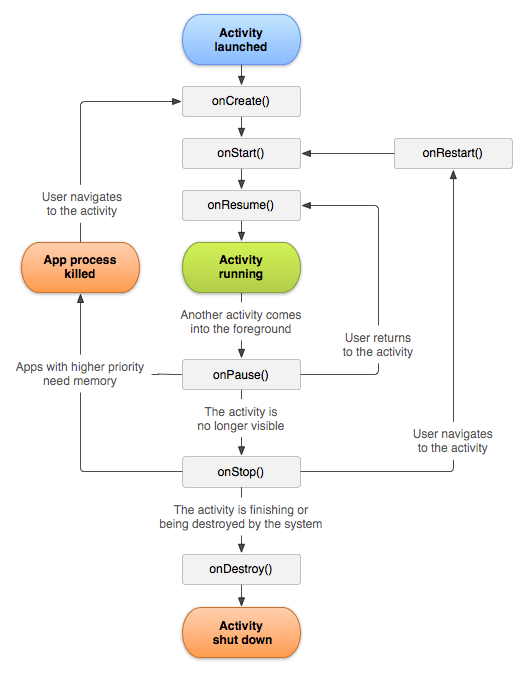
* Activity là màn hình cho phép người dùng có thể tương tác được.
* Có rất nhiều cách để chia sẻ dữ liệu giữa các activity ví dụ như:

Intent, ApplicationContext, interface,sharepreference...vv

Lưu ý: muốn truyền đối tượng qua bằng intent thì đối tượng đó phải implement serializable hoặc parcelable

1. Vòng đời của activities như thế nào? Giải thích

Có 7 method chính sẽ được gọi từ khi activity bắt đầu đến kết thúc:

* OnCreate: được gọi đầu tiên, đây là nơi khởi tạo các thành phần và dữ liệu activity
* Onstart: phương thức này được gọi khi activity đamg hiển thị tới người dùng và chưa thể tương tác được
* OnResume:Được gọi khi activity bắt đầu nhân tương tác với người dùng.
* OnPause: phương thức này được gọi khi system mở lại activity trước đó. Phương thức này cũng lưu dữ liệu chưa thay đổi, stop animation
* OnStop:được gọi khi activity không hiển thị cho người dùng thấy trong 1 thời gian dài.
* OnDestroy: Call khi hệ thống hủy activity.
* OnRestart:được gọi khi activity được hiển thị lại từ trạng thái stop.

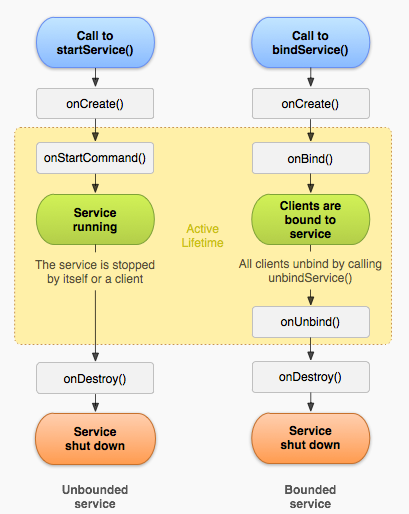
Có 3 vòng lặp chính mà bạn cần nắm rõ:

**Entire lifetime**: xảy ra giữa onCreate() và onDestroy(). 1 activity sẽ cài đặt các trạng thái trong onCreate() và giải phóng toàn bộ tài nguyên trong onDestroy(). Ví dụ có 1 luồng chạy ngầm tải dữ liệu từ trên mạng, ta có thể tạo luồng tại onCreate() và kết thúc luồng tại onDestroy().

**Visible lifetime**: xảy ra giữa onStart() và onStop(). Trong giai đoạn này, người dùng có thể nhìn thấy activity trên màn hình, tuy nhiên nó không ở trên đầu ngăn xếp và không thể tương tác với người dùng. Giữa 2 phương thức này người dùng có thể lưu trữ được tài nguyên cần thiết hiển thị lên activity cho người dùng. Ví dụ ta cần quan sát thay đổi ảnh hưởng tới giao diện, thì ta có thể đặt Broadcast Receiver tại onStart() và loại bỏ tại onStop().

**Foreground lifetime**: xảy ra giữa onResume() và onPause(). Trong giai đoạn này activity ở trên đầu ngăn xếp và có thể tương tác với người dùng.

1. Service là gì ? có mấy loại ?

Service là 1 component chính của android, nó có thể chạy ngầm các tác vụ mà không cần giao diện. Service cũng có vòng đời riêng

Có 2 loại Service

* boundService: cung cấp 1 interface cho phép tương tác với service
* started service: start và không cho phép tương tác

1. Khác nhau giữa: START\_STICKY, START\_NOT\_STICKY, START\_REDELIVER\_INTENT?

* START\_STICKY: KHi return cờ này khi service bị kill nó sẽ khởi động lại với intent null
* START\_NOT\_STICKY: Khi return cờ này thì service sẽ không khởi tạo lại khi nó bị kill
* START\_REDELIVER\_INTENT: service sẽ khởi động lại khi bị kill với intent gần nhất đc truyền vào

1. Broadcast là gì? Có mấy loại?

Broadcast là 1 thành phần chính của android, nó cho phép ứng dụng gửi và nhận thông điệp trong, giữa các ứng dụng và system.

Có 2 loại broadcast:

* Broadcast: cho phép nhân gửi thông điệp giữa các ứng dụng
* Localbroadcast: chỉ nhận và gửi thông điệp trong ứng dụng mà thôi

1. Làm thế nào để bảo vệ broadcast khỏi những kẻ tấn công, ví dụ mình có ứng dụng lắng nghe action “SCREEN\_OFF” để hiện khóa màn hình. Bây giờ ô A có thể viết 1 ứng dụng dùng for send 1000 action “SCREEN\_OFF” thì ứng dụng của mình làm thế nào để ngăn chặn.

Trường hợp 2, mình có action SEND\_MONEY gửi từ ứng dụng A tới ứng dụng B của mình. Làm thế nào để bảo vệ broadcast send\_money không cho ứng dụng nào khác lắng nghe action đó.

TL: Chúng ta có nhiều trường hợp.

· Nếu broadcast của chúng ta muốn lắng nghe chỉ send từ ứng dụng của mình thì mình có thể sử dụng localbroadcast hoặc set exprort=”false” khi khai báo broadcast/

· Nếu broadcast của chúng ta cần lắng nghe từ tất cả mọi chỗ thì mình có thể dùng add permission to broadcast.. chỉ những ứng dụng có quyền thì mới send đc hoặc nhận được broadcast của mình, khi send broadcast thì chúng ta cũng chỉ rõ ứng dụng được nhận broadcast. Thao khảo tại<https://www.youtube.com/watch?v=prueLyTvOwI>

1. Content provider là gì.

Content provider là 1 thành phần chính của android, nó cho phép chia sẻ và truy cập data dùng chung giữa các ứng dụng.

1. Intent là gì? có mấy loại ?

* Intent là 1 thành phần chung gian giữa các component chính của android và cho phép truyền thông điệp giữa các thành phần ứng dụng.

Có 2 loại intent là Implicit and Explicit.

* Inplicit intent là 1 intent hỗ trợ call những activity của hệ thống. Ta có thể truyền tên action và dữ liệu ta muốn hướng đến, ví dụ như view 1 trang web ta vọi action View và 1 uri thì hệ thống sẽ mở trình duyệt vs uri ta truyền.
* Explicit intent là intent rõ ràng, chỉ định nơi đi và nơi đến, cũng như có thể truyền dữ liệu đi cùng... thường để gọi startActivity, send broadcast, startservice..vv

1. Có những đơn vị đo nào trong android.

Px,inch, dp, dip, sp, dip.

Trong android ta thường dùng dp và sp. Dp và sp cơ bản là giống nhau nhưng sp có thể co giãn theo font nên thường dùng để set kích thước cho text. Còn lại ta dùng dp

<https://kipalog.com/posts/Android-tu-co-ban-cho-den-nang-cao---Supporting-Multiple-Screens--p1>

1. Các folder,file chính của 1 project Android.

Java: chứa các class java là Activity, broadcast, service...vv để thực hiện xử lý sự kiện và logic chương trình

Res: chứa các folder resource của chương trình.

* Drawable,minmap: folder chứa ảnh và icon của ứng dụng
* Layout: chứa các file layout của các màn hình
* Values: chứa các file để lưu chữ các màu, string, style, dimen của ứng dụng
* Ngoài ra còn rất nhiều folder khác như menu...vv

Mainfiest: Chứa mainifest file là file liệt kê các thành phần của ứng dụng, khai báo quyền.vvv File mainifest định nghĩa activity chạy đầu tiên của ứng dụng.

1. Assets folder để làm gì.

Assets là folder chứa tài nguyên của chương trình gồm những file .txt .mp3 .db ...vv và các file được lưu trong folder assets chỉ có thể đọc, và tốc độ đọc nhanh hơn.

1. Làm thế nào để thiết kế giao diện đa màn hình.

Trong android chúng ta có thể chia các folder drawable, dimens theo nhiều kích thước màn hình khác nhau.. và khi chạy ứng dụng ở device nào ứng dụng sẽ lấy giá trị đúng với kích thước màn hình. Và trong LinearLayout chúng ta có thuộc tính weight và weightsum để chia theo tỉ lệ

1. ANR là gì?

ANR viết tắt của Application Not Responding. Nếu ứng dụng thực hiện quá nhiều task trên main thread và không phản hồi trong một thời gian dài(>=5s), Android system sẽ hiển thị là dialog này.

1. WebView trong Android là gì?

Một WebView là một thành phần giao diện người dùng Android dùng để hiển thị trang web

1. Các loại layout(ViewGroup) trong android?

Android cung cấp các viewgroup là thành phần để chứa các view. Và mỗi viewgroup sẽ có 1 cách bố trí các view con của nó khác nhau.

1 số view group trong anroid:

* LinearLayout: Sẽ sắp xếp các view con theo 2 chiều ngang và dọc tương ứng với horizontal và vertical
* FrameLayout: Cho phép các con của nó đè lên nhau. View nào đc add sau sẽ hiển thị phía trên
* RelativeLayout: Cho phép các view có ràng buộc với nhau và ràng buộc với chính layout cha
* Table layout: Cho phép sắp các control theo dạng lưới (dòng và cột)

Ngoài ra còn 1 số layout khác như: gridlayout, constraint layout, absolutelayout...vv

1. View trong android là gì?

Android định nghĩa ra các view để đảm nhiệm các chức năng khác nhau và nó đc đặt trong các viewgroup.

1 số view cơ bản: TextView, EditText,Button, Spinner, ImageView,ImageButton...vv tất cả các thành phần này đều extent từ view.

Ngoài các thành phần được xây dựng sẵn thì ta cũng có thể tự định nghĩa các view riêng của mình(customview)

1. Fragment là gì? Có những cách nào để truyền dữ liệu cho fragment

Fragment là 1 activity thu nhỏ.. 1 activity có thể chứa nhiều fragment và nó giúp cho activity có thể thay đổi chức năng mà không cần khởi tạo activity mới.

Fragment có giao diện và vòng đời riêng

Để truyền dữ liệu từ activity vào fragment ta có thể dùng nhiều các như: setArguments, AppilcationContext,SharePreference, interface...

1. Khác nhau giữa add và replace trong Fragment.

* add(): là thêm 1 fragment vào 1 View được khai báo trước trong file xml
* replace(): là thay thế 1 fragment cũ bằng 1 fragment mới vào View được khai báo trước trong file xml. Nếu trong View chưa chứa fragment nào thì nó sẽ thêm mới fragment vào View.

Xem chi tiết hơn tại:

<https://viblo.asia/p/su-khac-nhau-giua-addfragment-va-replacefragment-trong-android-LzD5dNkdZjY>

1. method ***addToBackStack***  khi commit fragment có tác dụng gì?

Khi commit và gọi phương thức này fragment sẽ đc add to stack và khi user bấm back thì sẽ trờ về fragment trước đó.

<http://laptrinhandroid.net.vn/huong-dan-ve-fragment-trong-lap-trinh-android.html>

1. SharePreference là gì?

SharePreference là 1 thành phần giúp lưu trữ dữ liệu theo kiểu key, value trong xml.

Nó thường lưu trữ setting, account infor, high score...vv

SharePreference thường để lưu dữ liệu nhỏ. Và không lưu được trực tiếp object. Nếu muốn lưu object ta sẽ lưu từng attribute của object đó.

1. AsyncTask là gì? Các phương thức và tham số của asynctask.

Class này cho phép thực hiện những hành động và gửi kết quả tới UI thread mà không phải dùng thread và/hoặc handler. AsyncTask thì được thực hiện trên 1 backgroup thread và có thể cập nhật trạng thái đến UIThread.

Khi khởi tạo AsyncTask ta cần 3 tham số:

Tham số 1: kiểu giá trị truyền vào:

Tham số 2: kiểu dữ liệu trong lúc cập nhật.

Tham số 3: kiểu dữ liệu của kết quả trả về.

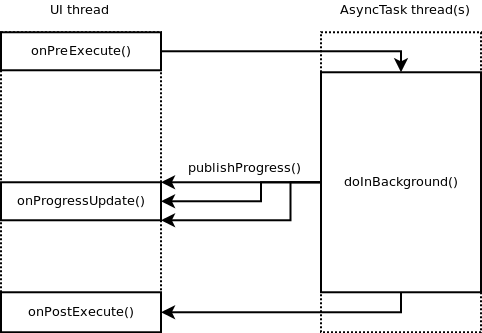
Và Asynctask có 4 phương thức chính:

OnPreExcute: Phương thức được gọi đầu tiên, ta thường khởi tạo hoặc show dialog ở đây

OnProgressUpdate: Phương thức để cập nhật tiến độ. Ví dụ update progress

doInBackgroud: phương thức chạy ngầm thực hiện công việc trên 1 thread khác

OnPostExcute: phương thức cuối cùng đc gọi, nó chứa kết quả trả về của method doinbackground



Khi 1 activity khởi tạo 1 asynctask sau đó activity destroy async task vẫn tiếp tục thực thi cho tới khi hoàn thành

1. Khi quay điện thoại thì chuyện gì sẽ xảy ra với activity đang hiển thị? Cách xử lý?

Activity sẽ call onPause,onStop, onDestroy sau đó khởi tạo lại từ đầu.

Muốn lưu lại các thông tin hiện thời ta có thể làm trong onsaveinstancestate và onrestoreinstancestate ta có thể lấy các giá trị đã lưu

1. Em sử dụng loại sql nào trong android. Tại sao lại dùng nó.

Trong android ta chỉ có thể sử dụng sqlite, sqlite là 1 bộ sql đơn giản, nó không chứa các ràng buộc giữa các table. Nó chỉ nên dùng khi lưu các bộ dữ liệu nhỏ. Vì giới hạn về phần cứng của điện thoại

1. Em biết gì về webservice, tại sao cần dùng webservice.

Webservice là 1 thành phần cung cấp tài nguyên, dữ liệu thông qua các URL, giúp cho client(mobileApp) hoặc web cố thể lấy dữ liệu và sử dụng chung dữ liệu.

Nó giải quyết bài toán cần lưu bộ dữ liệu lớn mà điện thoại k thể đáp ứng.

1. Khác nhau giữa thread, service, asynctask

<https://androidcoban.com/sanh-thread-handler-asynctask-service-intentservice.html>

1. Vong doi cua activity anh huong ntn so voi thread va asyntask

Trong onResume ta start thread, asyntask, khi activity pause và resume lại thì:

Asyntask se khong tao moi the hien cua no khi no dc goi lại

thread sẽ tạo new thể hiện khac...

vong doi cua activity khong anh huong den asyntask va thread(start activity moi thi asyntask và thread van chay tiep). nhung khi application dung. tat ca cac thread cua no deu dung.

1. Toast là gì?

Toast là component cho phép hiển thị nội dung trong 1 thời gian ngắn trên màn hình

1. Notification là gì? Cách sử dụng? Notification có gì thay đổi ở android O

Notification là 1 thành phần giúp hiển thị các thông tin trên thanh status bar.

Để show 1 notification ta cần thông qua Notification service của hệ thống.

Từ Android O ta cần thêm channel thì mới có thể hiển thị đc.

1. Animation là gì?Có những loại animation nào?

Animation là thành phần cung cấp các hiệu ứng cho view.

Có các loại là ViewAnimation, PropertyAnimation, DrawbaleAnimation

<https://viblo.asia/p/animation-trong-android-jlA7GKk4GKZQ>

<https://medium.com/@shubham.bestfriendforu/a-beginners-guide-to-implement-android-animations-part-1-2-part-series-b5fce1fc85>

1. Có những loại drawable nào? Bitmap vs nine-patch khác nhau ntn?

***Drawable*** là khái niệm chung về graphics để chỉ những gì mà bạn có thể vẽ.

Có nhiều loại drawable trong Andriod:

* BimapDrawable
* ColorDrawable
* GradientDrawable
* ShapeDrawable
* RippleDrawable (Android 5.0)
* VectorDrawable
* AnimatedDrawable (Android 5.0)
* StateListDrawable
* 9 Paths Drawables

<http://eitguide.net/drawable-trong-android/>

1. WindowManager là gì?

Là 1 thành phần quản lý các view đc vẽ trên màn hình, chúng ta có thể sử dụng windowmanager để vẽ các view nổi trên màn hình của các ứng dụng khác. Ví dụ như view tròn messenger của facebook

1. Custom view là gì?

Android đã cung cấp 1 bộ các view để thiết kế ứng dụng, nhưng có thể nó vẫn không đáp ứng được nhu cầu của ứng dụng và đôi khi ta cần 1 số view đặc biệt, lúc đó ta phải customview chính là việc tạo ra những view mà đáp ứng nhu cầu của ứng dụng.

<https://kipalog.com/posts/Android--Hieu-sau-hon-ve-CustomView-va-Huong-dan-xay-dung-thu-vien-UI-IndicatorView>

1. Context là gì? Sử dụng nó như thế nào?

Context là bối cảnh của tình trạng hiện tại của ứng dụng hoặc đối tượng. Nó nói cho các đối tượng được tạo mới biết chuyện gì đang xảy ra. Thông thường  ta gọi nó để lấy thông tin liên quan đến những phần khác của ứng dụng (activiry và package/ application). Android có những loại context sau:

* Application context: nó là một thể hiện dạng singleton (thể hiện duy nhất) và có thể được truy cập trong activity thông qua getApplicationContext(). Nó được gắn liền với vòng đời của ứng dụng. Nó có thể được dùng khi ta cần một context với vòng đời khác so với context hiện tại hoặc khi ta truyền một context vượt qua phạm vi của activity

Ví dụ:

Nếu ta tạo một đối tượng dạng singleton cho ứng dụng và đối tượng này cần một context thì luôn luôn dùng application context. Nếu ta truyền activity context ở đây thì nó sẽ dẫn đến memory leak vì nó sẽ luôn giữ tha chiếu đến activity nên activity sẽ không bị GC thu thập dẫn đến leak memory

Ta chỉ dùng getApplicationContext() khi ta cần một context có vòng đời lâu hơn các context khác mà ta có.

* Activity context: là context trong một activity. Nó gắn liền với vòng đời – lifecycle của một activity. Nó nên đươc dùng khi ta muốn truyền một context trong phạm vi của activity hoặc ta cần một context với vòng đời gắn liền với context hiện tại.

Ví dụ:

Nếu ta phải tạo một đối tượng với vòng đời gắn liền với activity thì ta có thể dùng activity context.

* GetContext() trong ContentProvider:

là một application context. Context này được truy cập trong qua method getContext()

1. Có bao nhiêu cách lưu data trong ứng dụng android?

Android cung cấp 5 cách lưu data như sau. Từ khóa private nghãi là chỉ ứng dụng của ta mới truy cập được. Từ khóa public thì có thể được truy cập bởi các ứng dụng khác

* Shared Preferences: lưu những dữ liệu cơ sở – primitive data dưới hình thức cặp key – value, nên dùng với các private data.
* Bộ nhớ trong của thiết bị – Internal Storage: nên dùng với các private data
* Bộ nhớ ngoài thiết bị (card rời,…) – External Storage: nên dùng với các public data. Thông thường video, ảnh dung lượng lớn ta nên lưu ở mục này
* SQLite Database: lưu những data có cấu trúc trong private database. Ví dụ như lưu danh bạ điện thoại, đây là dữ liệu có cấu trúc: tên, số điện thoại, email,…. Private database
* Network Connection: lưu data trên web với server do ta quản lý. Ví dụ như thông tin user cần login chẳng hạn

1. Memory leaks – rò rỉ bộ nhớ trong Android là gì?

Là một dạng rò rỉ tài nguyên xảy ra khi ứng dụng không quản lý tốt việc cấp phát bộ nhớ. Hiểu theo nghĩa bộ nhớ đã được cấp phát nhưng sau một thời gian không dùng đến nữa nhưng vẫn không được giải phóng, vẫn tồn tại gây ra tốn tài nguyên bộ nhớ cũng như có thể gây chậm performent

1. Làm thế nào để tránh memory leaks – rò rỉ bộ nhớ trong Android

Về mặc nguyên tắc, Android đã có GC để giải phóng memory nên khi code ta cần hỗ trợ tốt cho GC thì sẽ góp phần tránh memory leak. Có một số lỗi ta nên tránh khi code

* Xử lý các đối tượng UI trong background
* Dùng static view vì như vật nó sẽ luôn tồn tại và không bao giờ bị kill
* Dùng static Context
* Dùng application context và activity context mà không xác định rõ
* Quên giải phóng cho các bộ listener. Điều này sẽ giữ cho activity luôn tồn tại để bắt các sự kiện
* Dùng inner class kiểu non static
* Dùng anonymous class

1. Design pattern là gì? Em có biết design pattern nào không?

Design pattern là những mẫu chung để giải quyết các vấn đề trong quá trình develop

Chúng ta phải biết 1 số design pattern như singleton, Factory, Builder...vv ngoài ra các em có thể tham khảo tại:

<https://www.tutorialspoint.com/design_pattern/>

Ngoài ra còn cần nắm chắc các thành phần khác cũng hay được hỏi như: viewpager,adapter, các parse dữ liệu json/xml, các lib hay dùng như retrofit, Picasso, volley,Jsoup,GCM,Firebase

Các mô hình trong android: MVC, MVVM,MVP, VIPEr, CLEAN Architec

Cần nắm rõ các project các em đã làm.

44. câu hỏi liên quan đến load ảnh giống glide.

Làm thế nào để load ảnh nhanh như glide khi người dùng load nhiều ảnh và scroll liên tục.

My solution: sử dụng threadpoolexcutor để load ảnh, nó sẽ sử dụng 1 số lượng thread để load ảnh..

và mình sẽ biết đc view nào đang ở ngoài màn hình và trong màn hình để set priority ưu tiên xem thằng nào đc load trước...

đồng thời sử dụng cache(memory cache, disk cache) để load nhanh hơn.

nếu to hơn view scale ảnh đúng kích thước của view

sửu dung bitmap pool để giảm thiểu GC(trình thu dọn rác) bị gọi

Có cách nào không cần theo dõi view nào ở trong ở ngoài màn hình mà vẫn load đúng không

https://viblo.asia/p/cach-toi-uu-cua-glide-va-fresco-khi-load-image-ORNZqPw3K0n