

いまさら聞けない  
ActivityやFragment  
の  
おはなし

Android Bazaar and Conference  
2016 Autumn

# mokelab

- ・ Androidアプリを開発したり
- ・ iOSアプリを開発したり
- ・ Webアプリ(TypeScript)を開発したり
- ・ 技術情報を発信したり

★ Twitter @mokelab

★ <https://tech.mokelab.com>



# グッズあるよ！



クイズを  
16問用意しました  
全問正解できたら。。。。

何もありません

Activity



# Activity

- ・ コンポーネントの1つだよ
- ・ UIを提供するよ
- ・ ライフサイクルがあるよ

コンポーネント





# 問 1

Androidには、何種類のコンポーネントがありますか？



- 4 種類

- ◆ Activity

- ◆ Service

- ◆ BroadcastReceiver

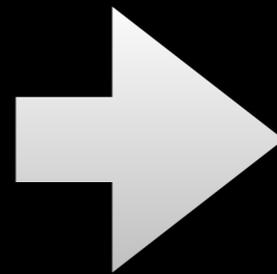
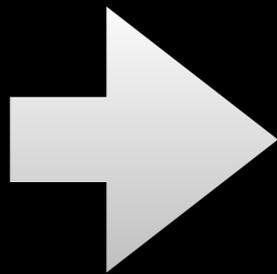
- ◆ ContentProvider

コンポーネントとは  
システムが必要に応じて  
オブジェクトを作って  
仕事させるもの

Activityの  
ライフサイクル



# ライフサイクル



※

※<http://yubin-nenga.jp/products/plane.html>

mokelab

# Javaオブジェクトの ライフサイクル

```
public void make() {  
    // 誕生  
    Date d = new Date();  
  
    // 誰からも参照されなくなったら  
    // 破棄される  
}
```



# 問2

Activityのライフサイクルメソッドを  
全部言えますか？



# 답2

- onCreate()
- onRestart()
- onStart()
- onResume()
- onPause()
- onStop()
- onDestroy()



ライフサイクルメソッドは  
なんとなくで  
呼ばれているものではない！



# onCreate()

- △ Activityが起動したとき
- ○ Activityオブジェクトが作られたとき



# onRestart()

- Activityが停止後、もういちど開始するとき
- 一度停止していないと呼ばれない



# 問3

onStart()と  
onResume()の違いは何でしょうか？



# onStart()

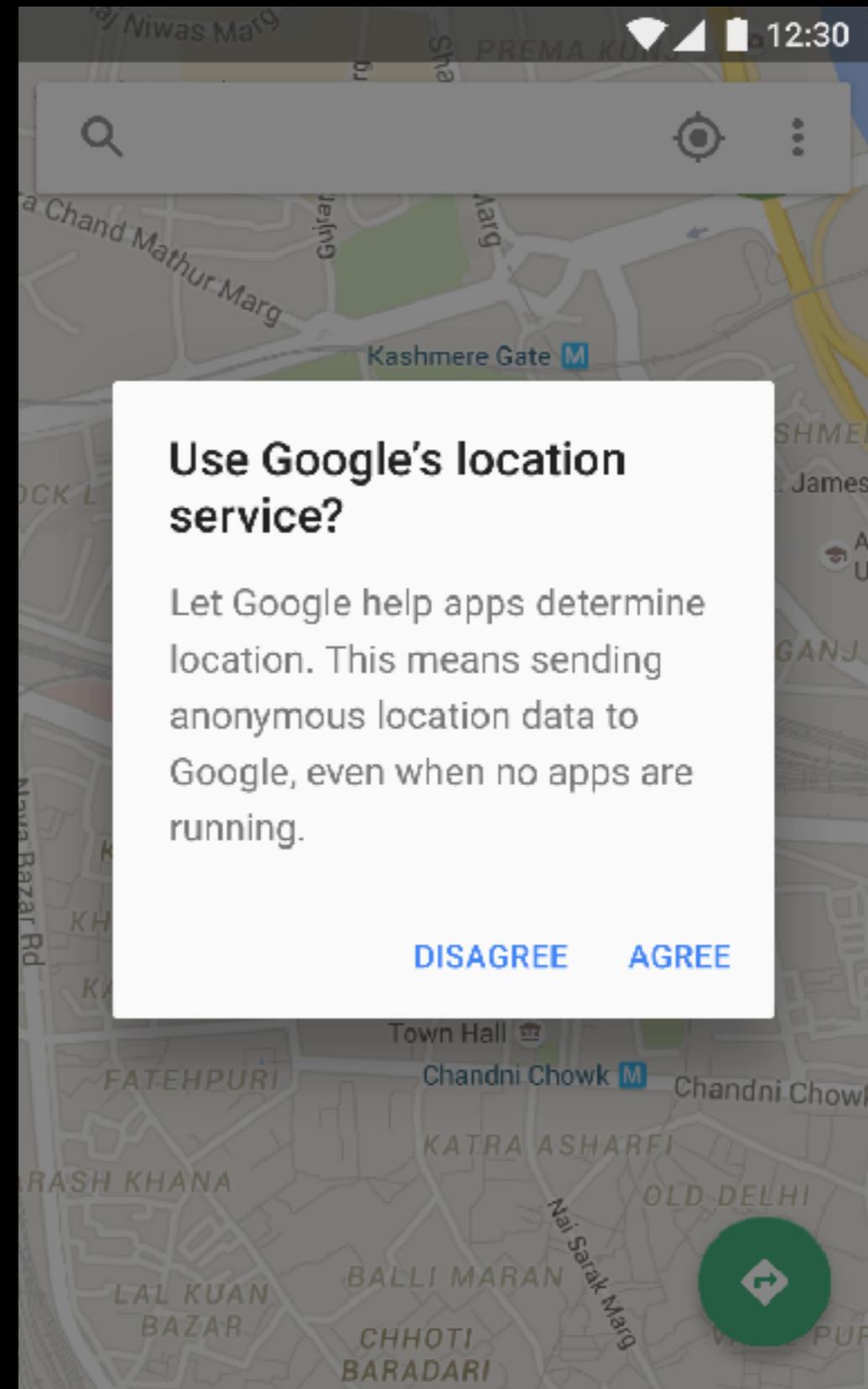
- Activityが表示されたタイミングでよばれる
- この時点では、ユーザーの操作は受け付けない



# onResume()

- ・ Activityがユーザーとの対話を始めたタイミングでよばれる
- ・ ユーザーは、ボタンタップなどができる

onStart()は  
呼ばれるが  
onResume()は  
呼ばれない例





# onPause()

- ・ Activityがユーザーとの対話をとめるときによばれる
  - ★ 別Activityを起動しようとした
  - ★ ダイアログを表示しようとした



# onStop()

- ・ Activityがみえなくなるタイミングでよばれる
  - ★ 別Activityが（全画面で）表示された
  - ★ ホームボタンを押した



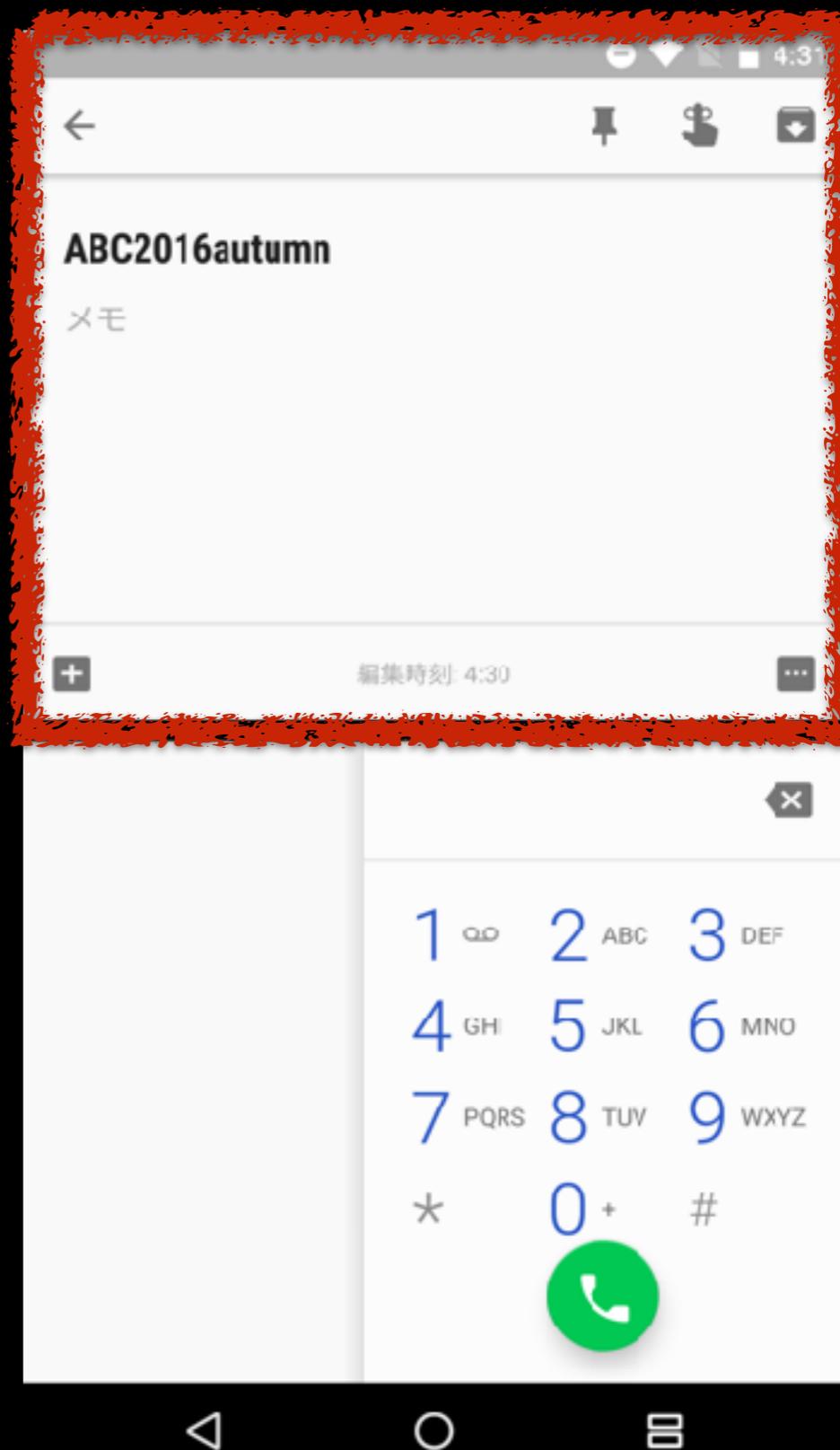
# onDestroy()

- △ Activityが終了するときによばれる
- ○ Activityオブジェクトが破棄されるときによばれる



# 問4

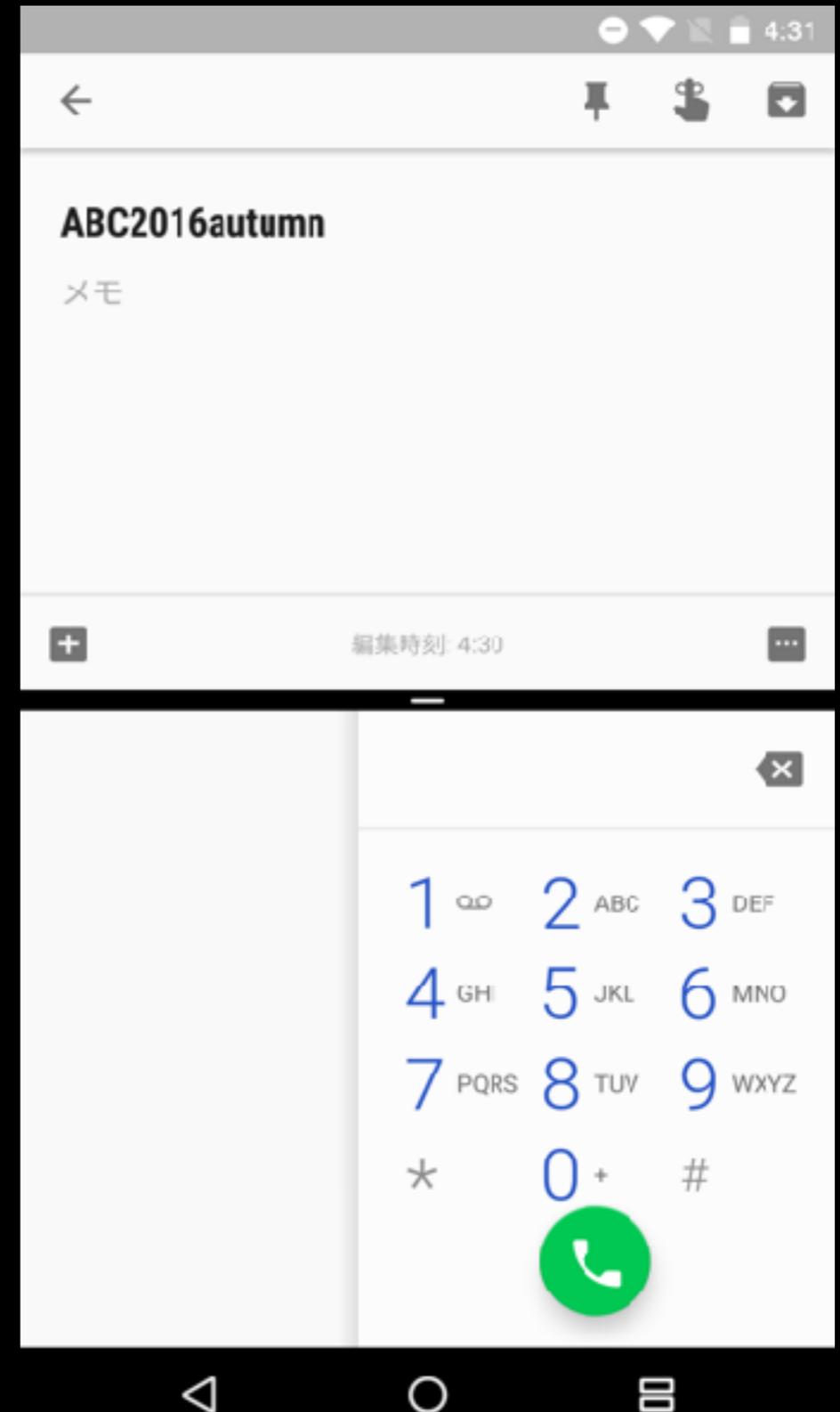
- ・ 上Activityがアクティブのとき  
下Activityをタップすると  
ライフサイクルメソッドは  
どのように呼ばれますか？





# 答4

- ・ 上Activity  
onPause()がよばれる  
(onStop()はよばれない)
- ・ 下Activity  
onResume()が呼ばれる



Activityオブジェクトの  
ライフサイクルは  
これでだいじょうぶ

タスク





# 問5

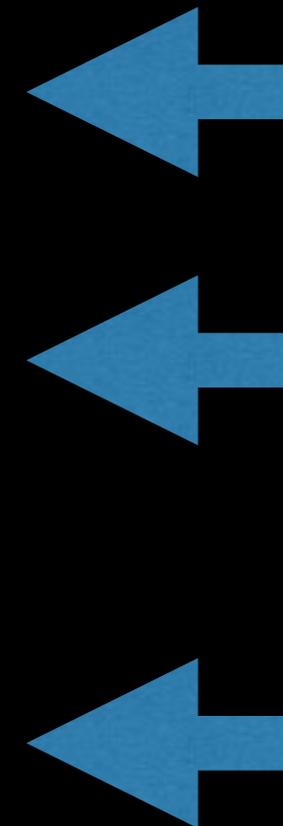
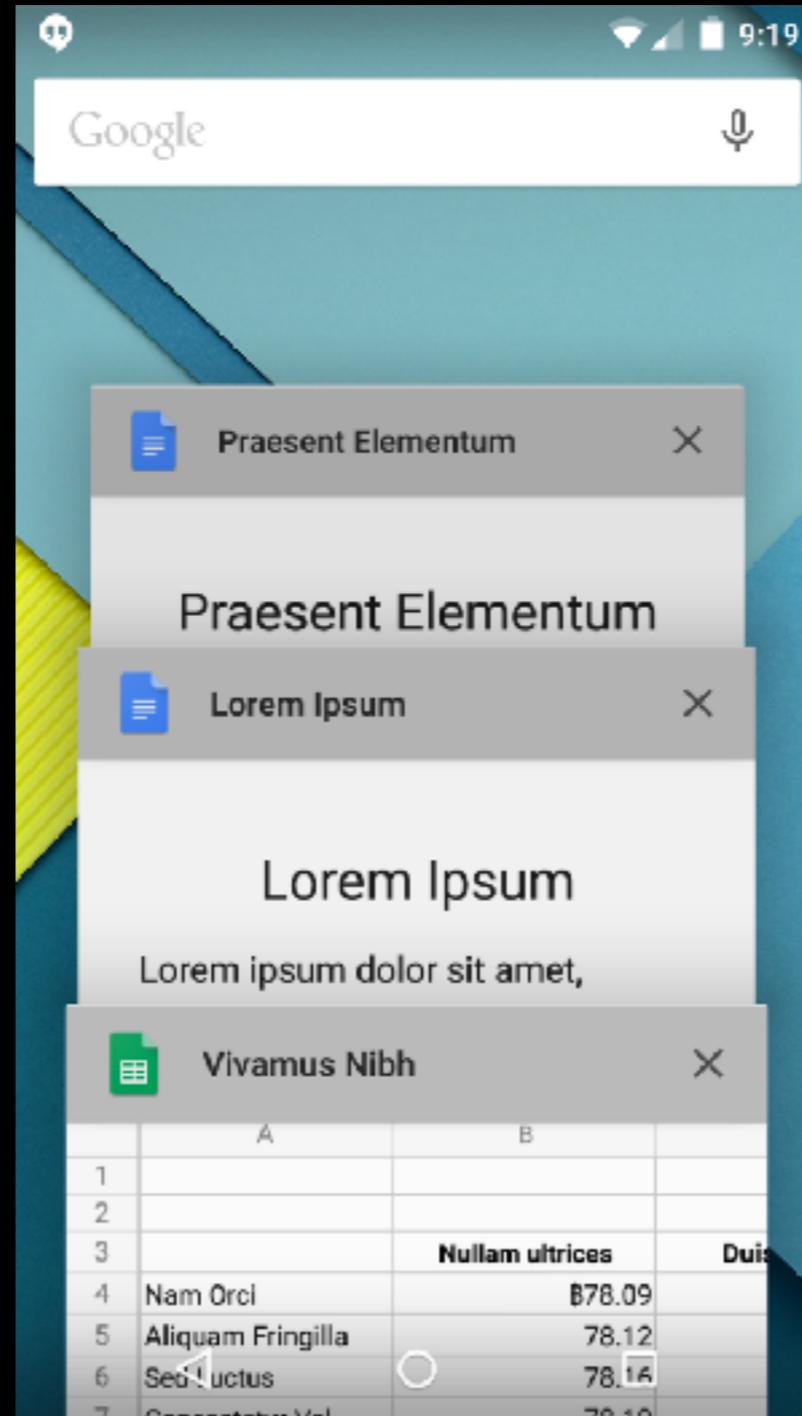
タスクとは  
ユーザーにはどのように見えるか  
説明できますか？



答5

アプリ！

# タスク



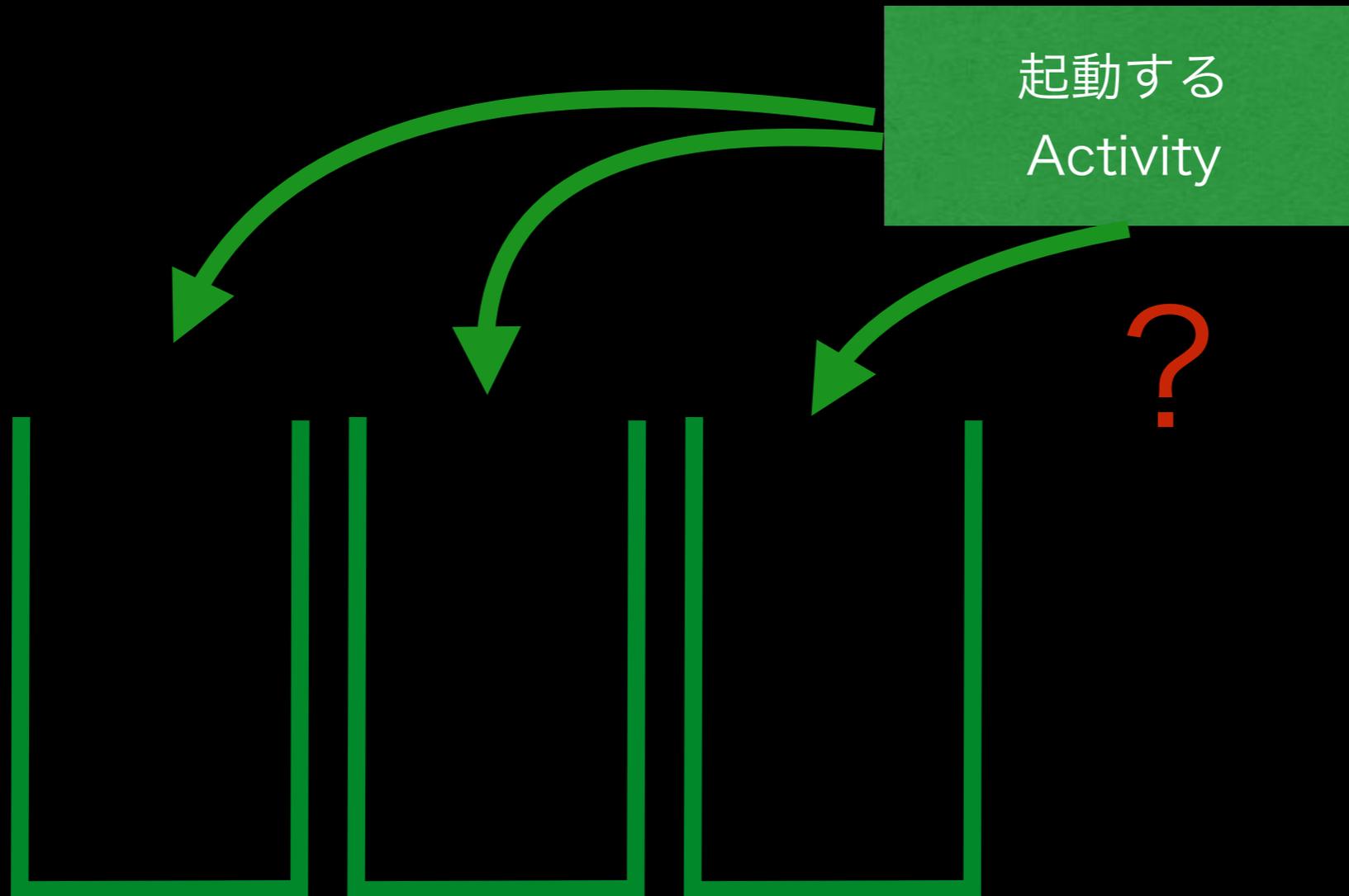
# タスクはスタック

Twitter投稿  
Activity

詳細Activity

一覧Activity

# Activityの起動





# 問6

Activityの起動モードは  
何種類ありますか？



• 4種類

★ Standard

★ SingleTop

★ SingleTask

★ SingleInstance



# Standard

- ・ 指定なしはこれで起動する
- ・ Intentを受け取ったら、インスタンスを作る
- ・ Intentを投げたActivityと同じタスクに入る



# SingleTop

- Intentを投げたActivityと同じタスクに入る
- ただし、自分自身に対してIntentを投げた場合は、新たにインスタンスを作らず、onNewIntent()が呼ばれるだけ



# SingleTask

- どれか1つのタスクにしか所属できない
- なので、インスタンスは1つしかできない



# SingleTask

- ・ ケース1 起動するActivityが  
どのタスクにもいない
- 新しくタスクを作り、フォアグラウンドに  
もってくる



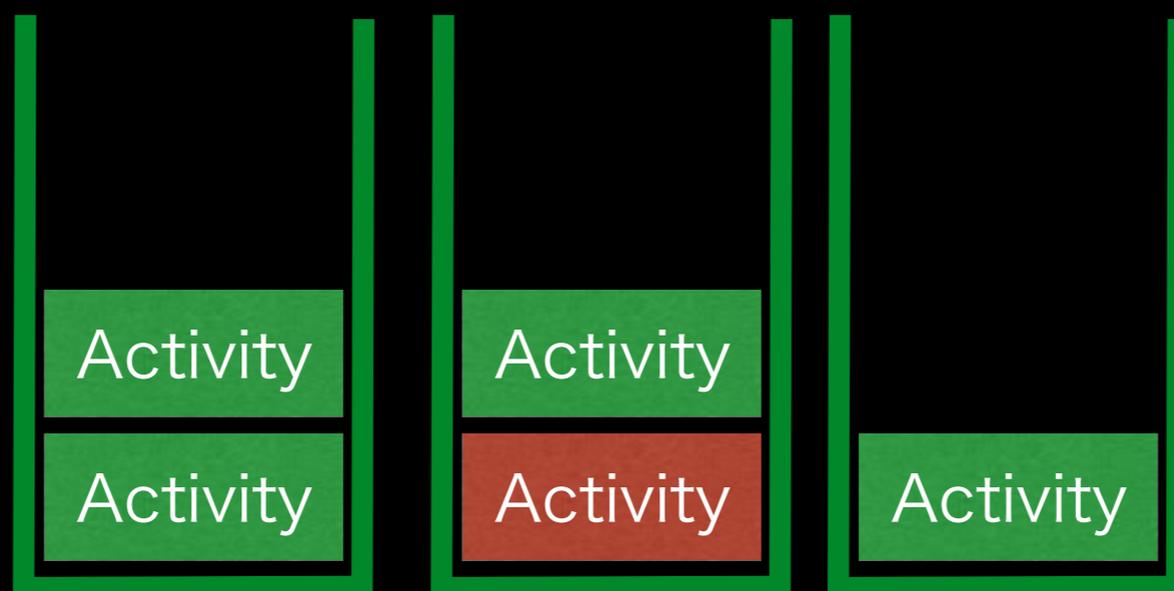
# SingleTask

- ・ ケース2 起動するActivityがすでに別タスクにいる
  - > 図で説明するよ



# SingleTask

1. 該当Activityのいるタスクを  
フォアグラウンドにする

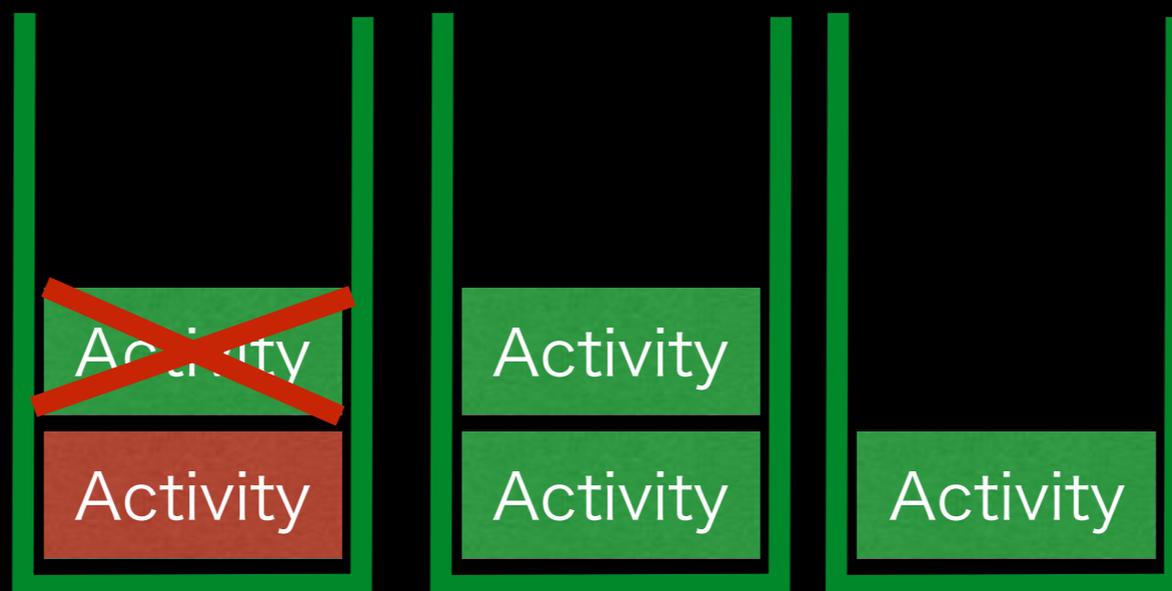


フォアグラウンド

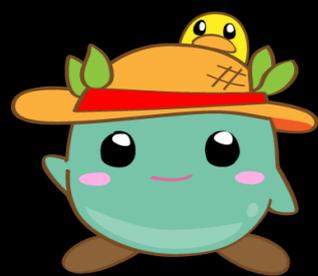


# SingleTask

2. 該当Activityよりスタックで上にいるActivityを全部終了させる！

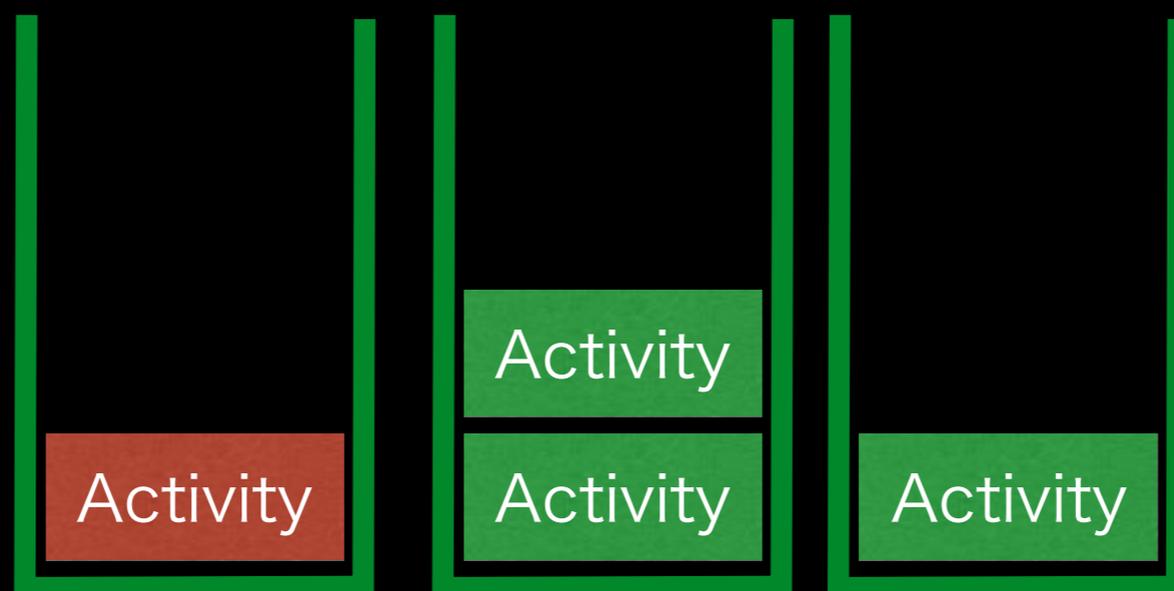


フォアグラウンド



# SingleTask

3. 該当ActivityのonNewIntent()が呼ばれる



フォアグラウンド

# 例：OAuthのコールバック

1. アプリActivityがブラウザに対してIntent投げる
2. ブラウザのタスクがフォアグラウンドになる
3. コールバックを契機に、アプリActivityにIntentが投げられる
4. アプリActivityのタスクがフォアグラウンドになる  
(新しくインスタンスは作られない)



# SingleInstance

- どれか1つのタスクにしか所属できない
  - そのタスクに別Activityを含めることはできない
- ★ SingleTaskだと含めることができるよ



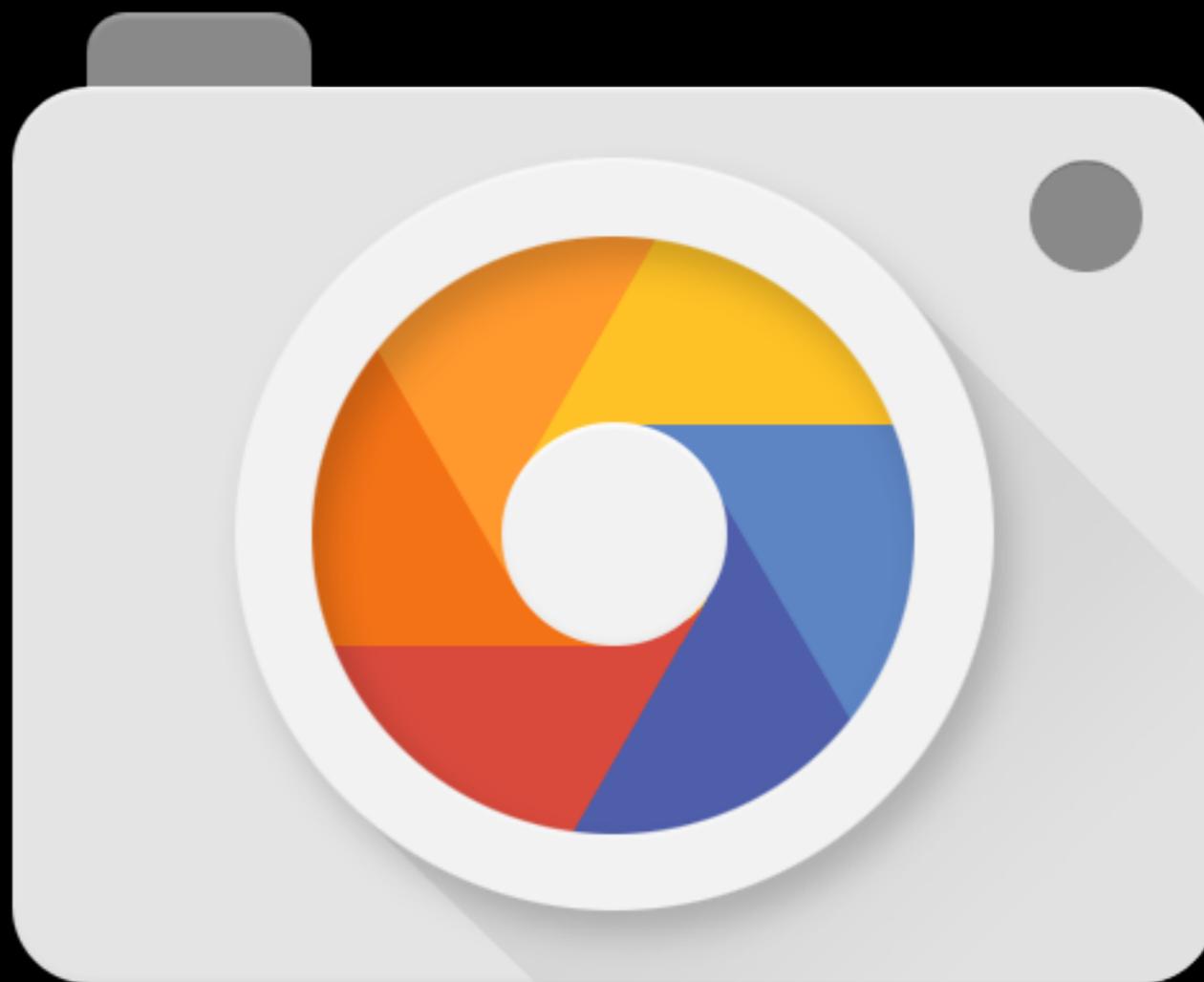
# 問7

Androidの標準アプリで  
SingleTaskが使われているのは  
どれでしょうか？



答7

カメラ！





# 答7

- ・ ホームでカメラアイコンをタップしたら  
すぐ撮影Activityが始まってほしい
- ・ 編集Activityなどがあるので、上にActivityが  
のっかるのはOKにしたい
- ★ 編集を中断して、オーバービューから  
再開はしたい



# 問8

Androidの標準アプリで  
SingleInstanceが使われているのは  
どれでしょうか？



答8

ブラウザ！





# 答8

はSingleTaskでした  
す、すいません。。。。





# 答8

ほんとの答えは  
通話中Activity





# 答8

- ・ 通話中Activityが他のActivityの上に乗っているとユーザーは通話中Activityに戻りにくくなる
  - ★ オーバービューではルートのアプリ名だから
- ・ 通話中Activityの上に別のActivityが乗っていると  
いざ通話をやめたいときに、バックボタンの連打が必要になってしまう



# 問8おわび

- ・ 単なる動作ではSingleTaskとSingleInstanceの違いを見分けるのは困難でした。。。
- ・ 通話ActivityがSingleInstanceかどうかは下記URLで確認できます(InCallActivity)  
<https://android.googlesource.com/platform/packages/apps/Dialer/+master/AndroidManifest.xml>

タスク  
アフィニティ





# 問9

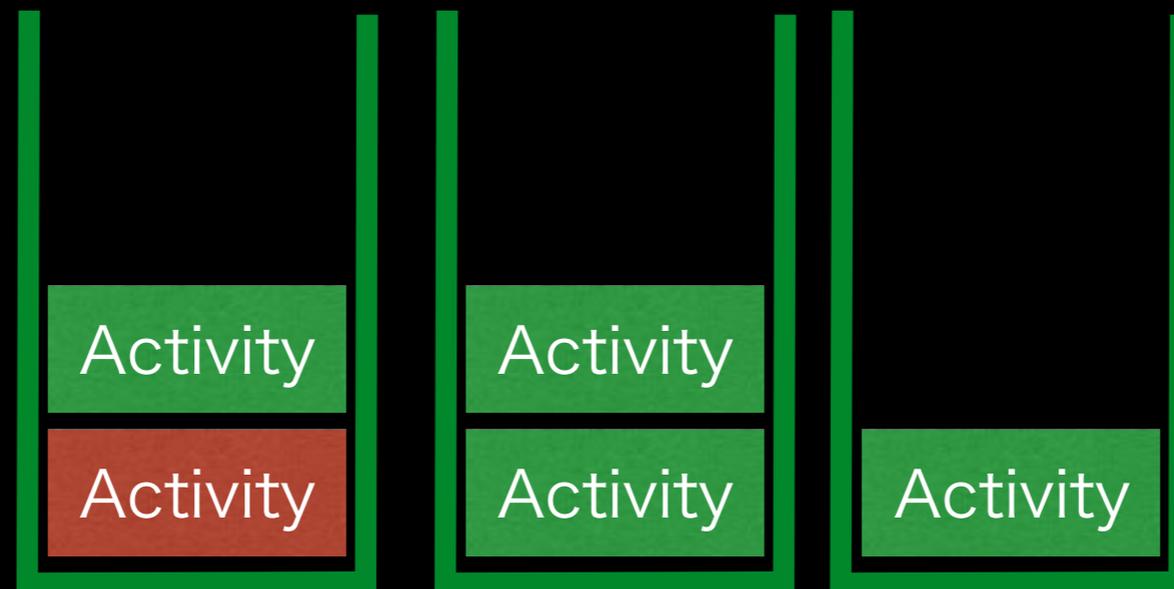
TaskAffinityということばを  
聞いたことがありますか？

# タスクアフィニティ

- ・ 「親和性」
- ・ ホームでアプリのアイコンをタップしたとき  
どういう動きをすべきか
- ★ FLAG\_ACTIVITY\_NEW\_TASK
- ★ 新しいタスクを作り続けるわけにはいかない

# タスクアフィニティ

タスクには名前がある＝アフィニティ



タスクA タスクB タスクC

# タスクアフィニティ

- ・ ホームでアプリアイコンをタップした
  - ・ 同じアフィニティなタスクがない場合
    - ・ タスクを作ってそれをフォアグラウンドにする
  - ・ 同じアフィニティなタスクがすでにある場合
    - ・ そのタスクをフォアグラウンドにする



# 問10

TaskAffinityは  
どんな時に指定すべきでしょうか？



# 答10

1つのapkにLAUNCHERなActivityが  
2つ以上ある場合！

デバッグ用Activityをいれている時に有効

# タスクアフィニティを 設定する

```
<activity
  android:name=".EditActivity"
  android:taskAffinity="com.mokelab.app.debug">
  <intent-filter>
    <action android:name="android.intent.action.EDIT"/>
    <category
  android:name="android.intent.category.DEFAULT"/>
  </intent-filter>
</activity>
```

# タスクアフィニティを 設定する

ドット(.)が1つ以上ないとインストール時にエラー

```
Error installing split apks: com.android.ddmlib.InstallException:  
Failed to finalize session :  
INSTALL_PARSE_FAILED_MANIFEST_MALFORMED: Failed  
parse during installPackageLI: /data/app/vmdl231441147.tmp/  
base.apk (at Binary XML file line #26): Invalid taskAffinity name  
com in package com.mokelab.affinitydemo: must have at least  
one '.' separator
```



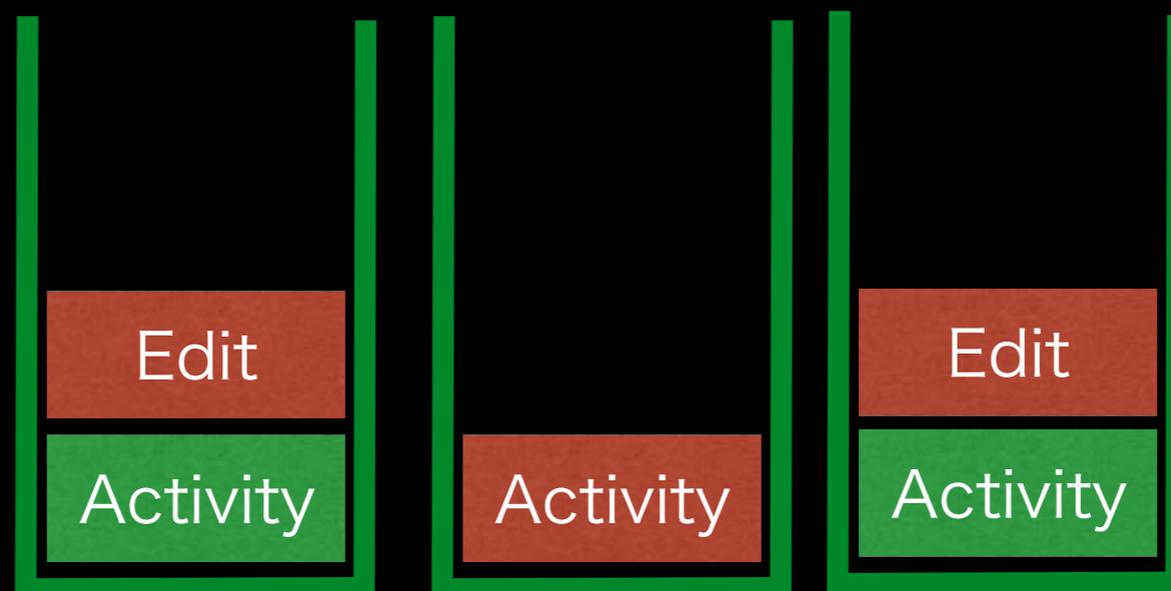
# 問11

`android:allowTaskReparenting="true"`  
は、どんな動きをするか理解していますか？



# 答11

EditActivityに  
android:allowTaskReparenting="true"をつけた  
(赤枠は同じAffinityを持つActivity)

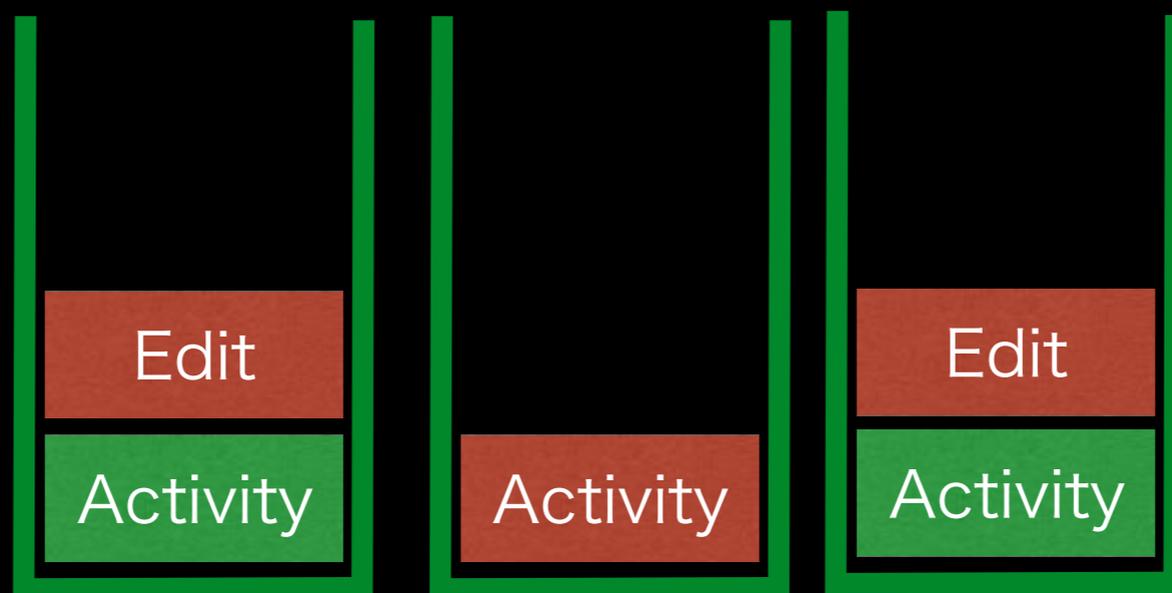


フォアグラウンド



# 答11

ホームでActivityのアイコンをタップする

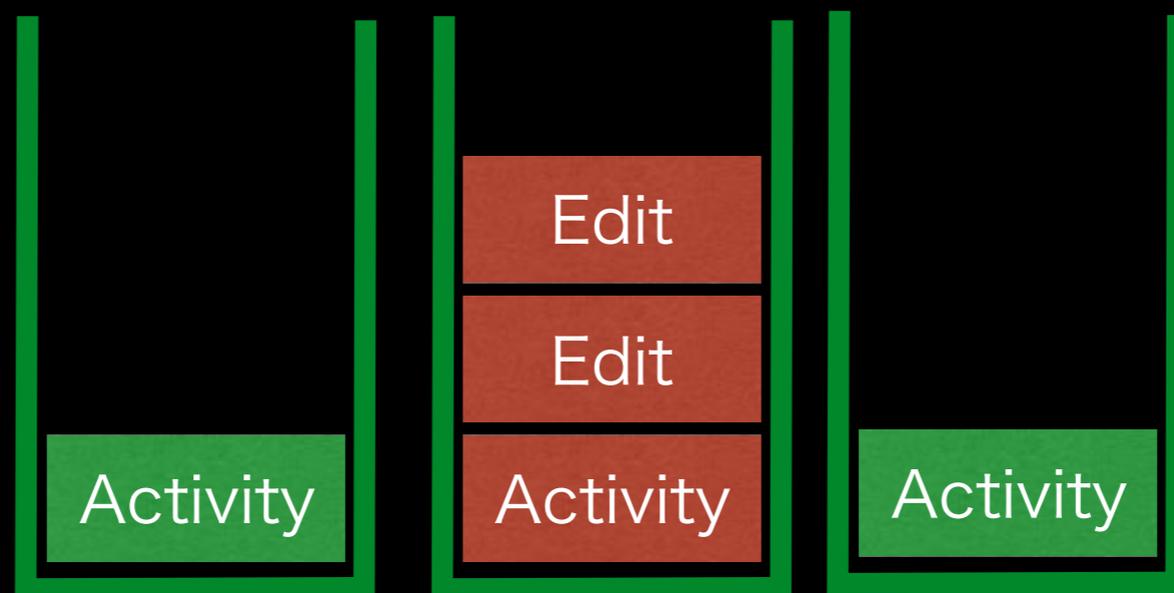


フォアグラウンド



# 答11

android:allowTaskReparenting="true"がついたEditActivityが  
集合する

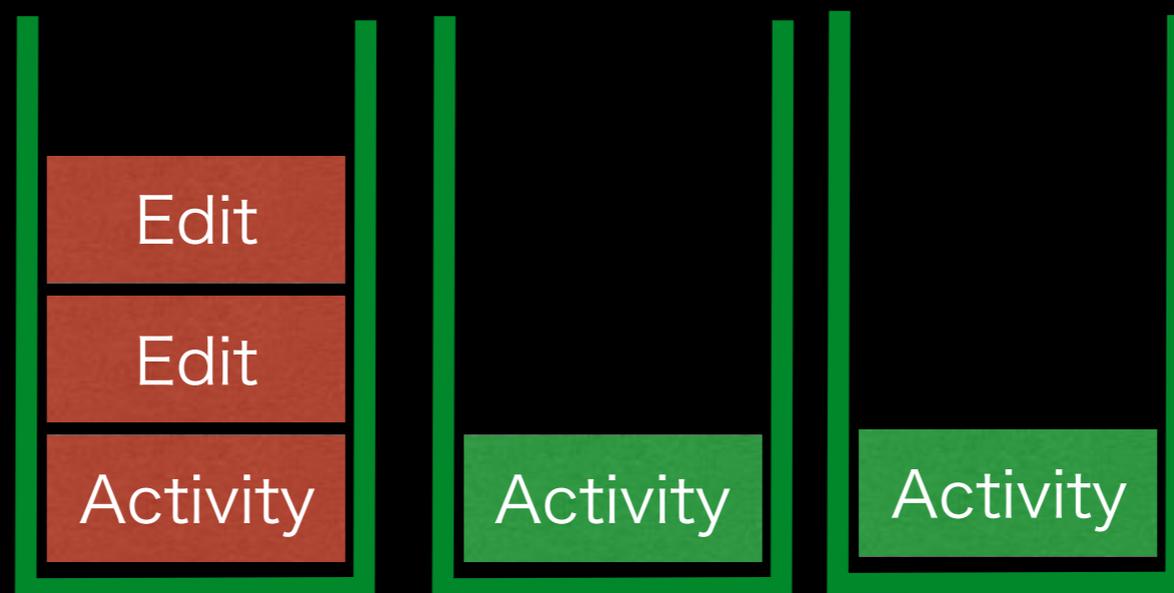


フォアグラウンド



# 答11

そして、フォアグラウンドに移動



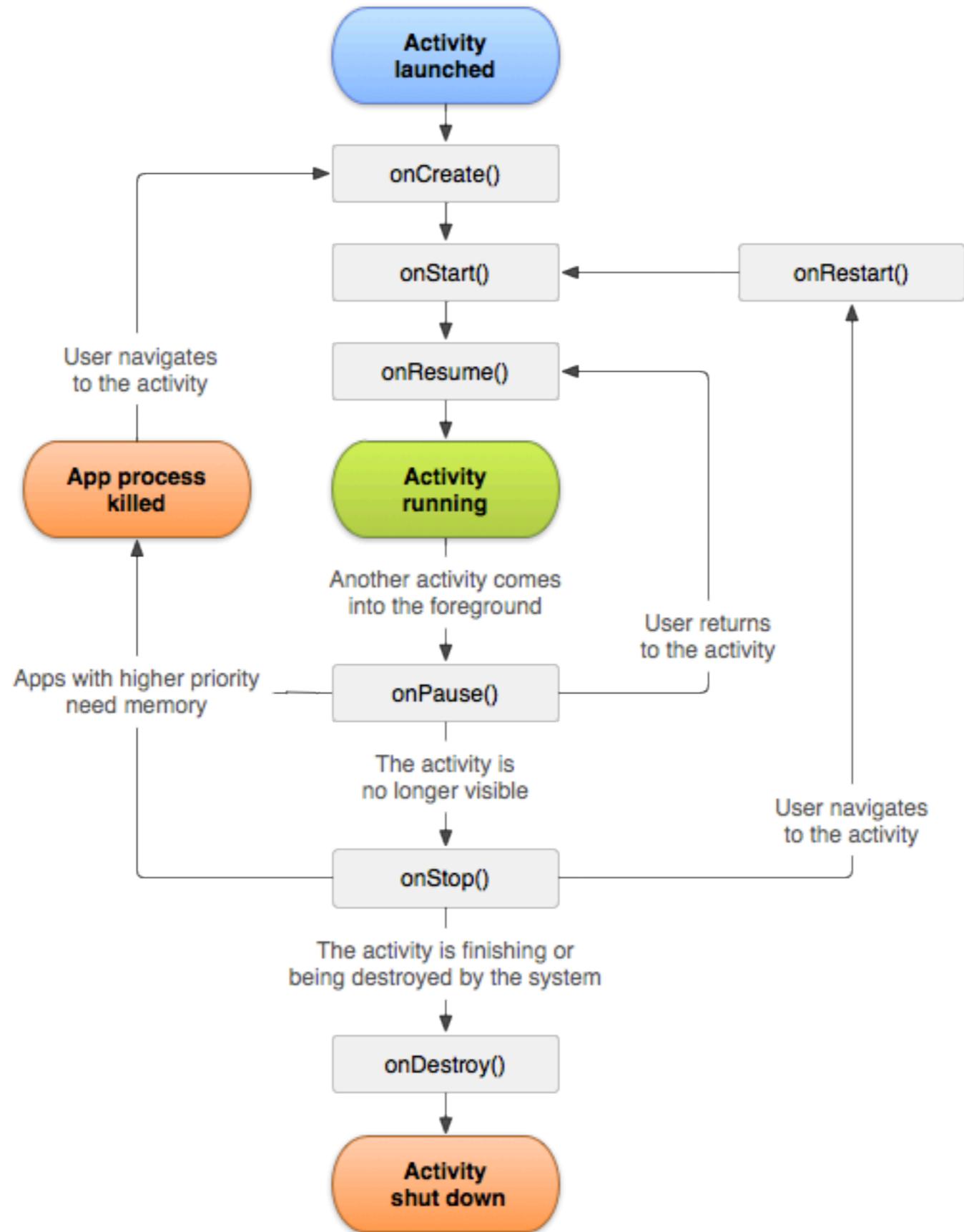
フォアグラウンド

タスクと起動モードは  
もう大丈夫ですよ？

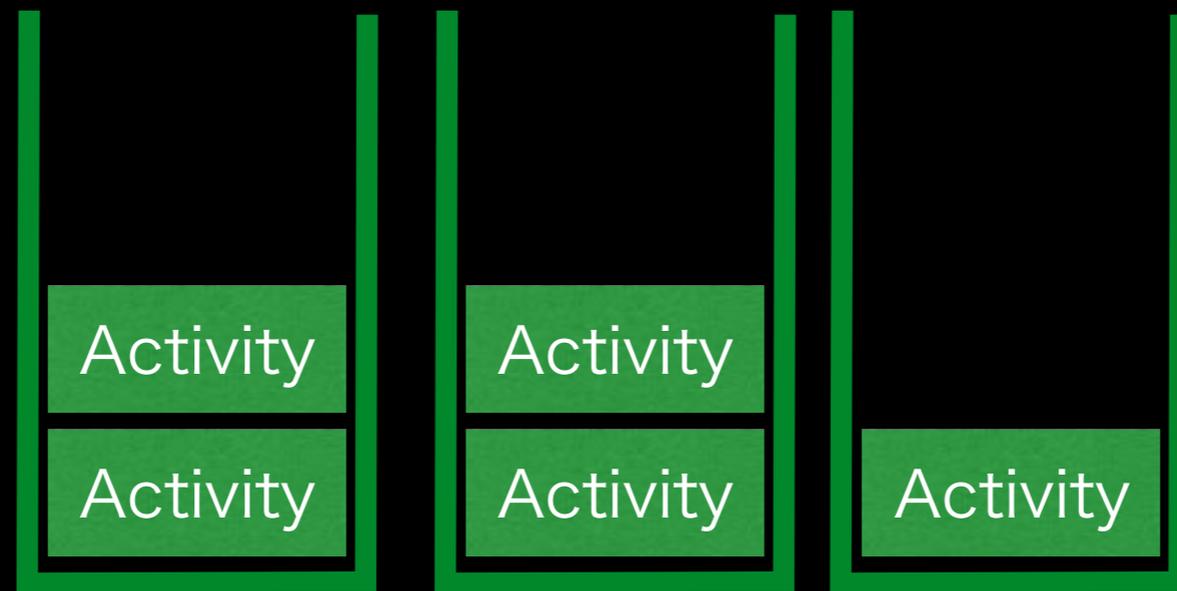
ですよね？

Activity  
の  
ライフサイクル  
(再)



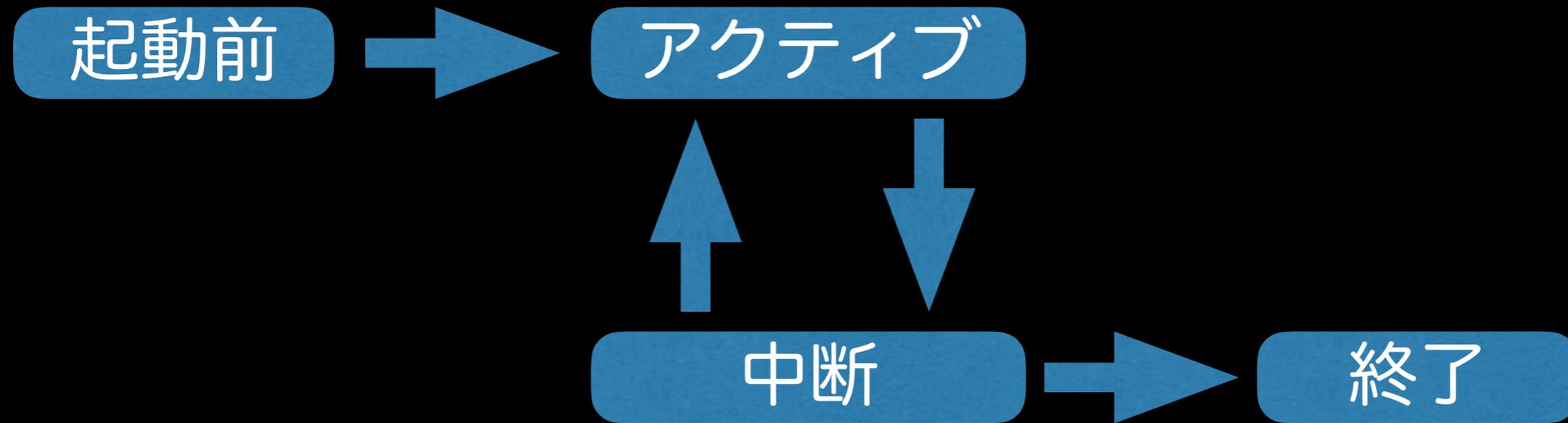


# タスクとActivity

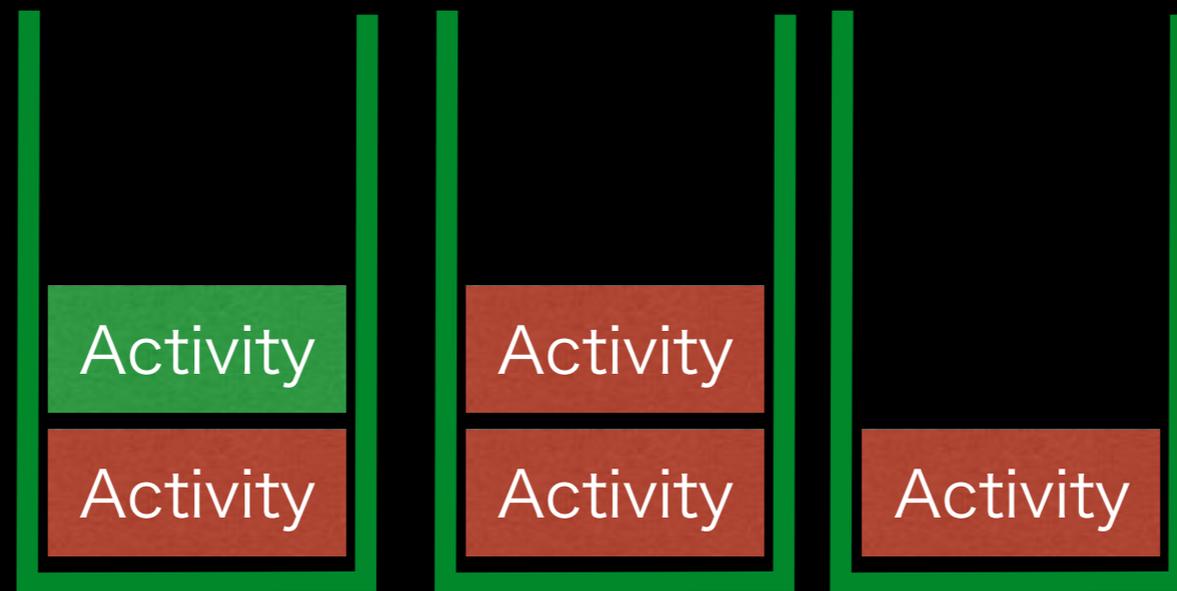


フォアグラウンド

# Activityの状態



# 中断中なActivity



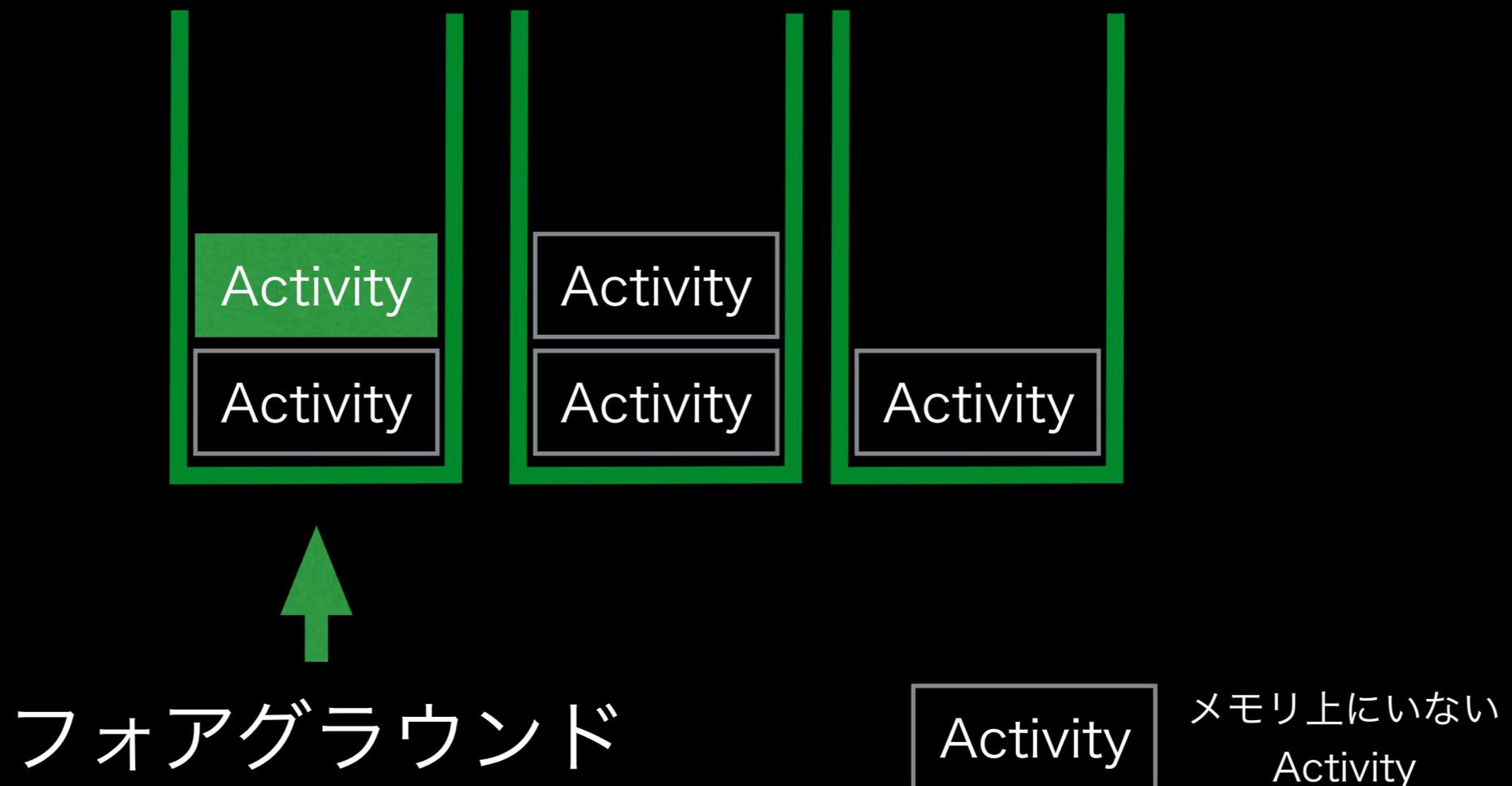
フォアグラウンド

Activity

中断状態

# 中断中なActivity

Activityオブジェクトは破棄されていることがある



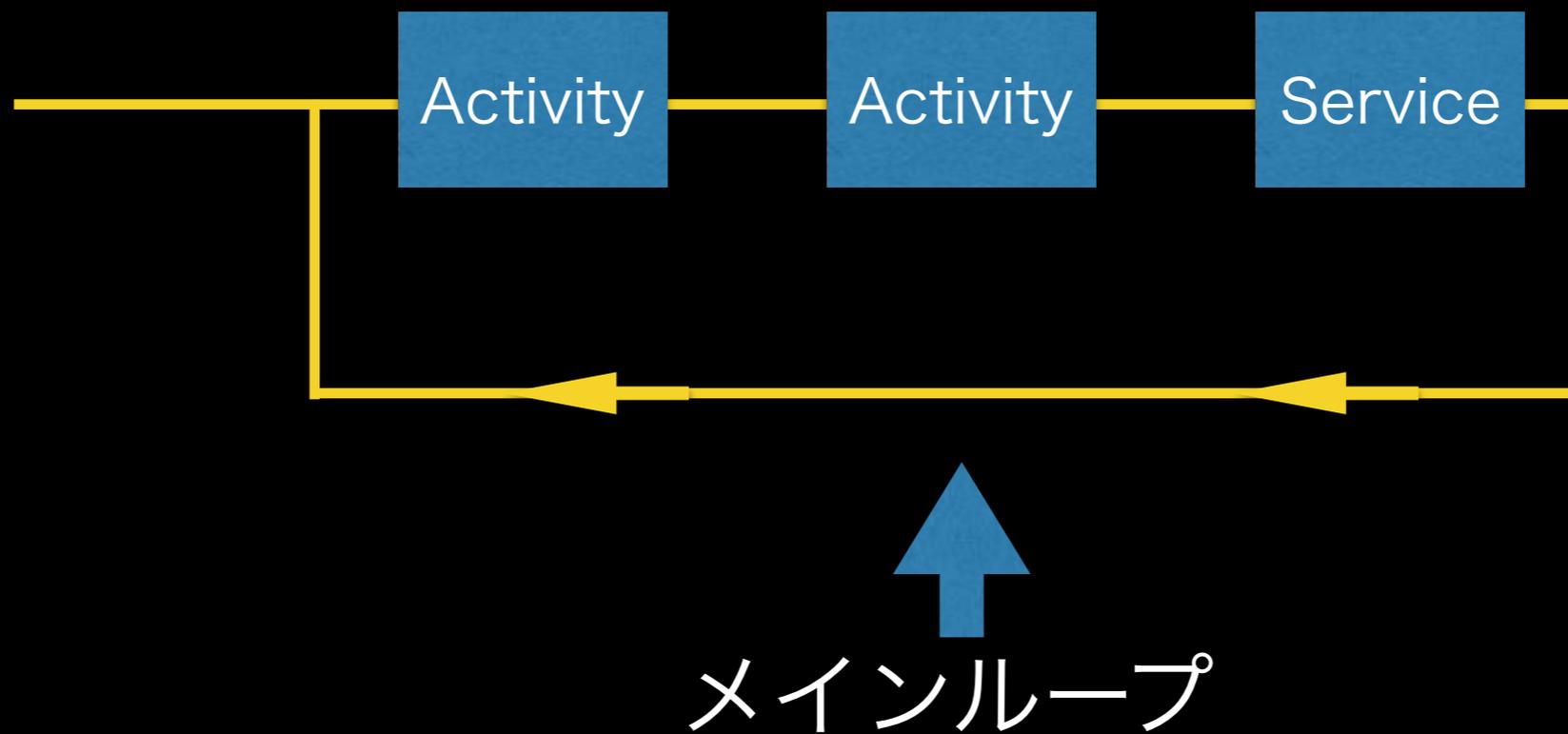
# オブジェクトの 状態保存

- ・ 中断状態の場合、オブジェクトが破棄されることがある
- ・ アクティブに戻る時、オブジェクトを作り直す
- ・ これをうまくやるのがonSaveInstanceState()

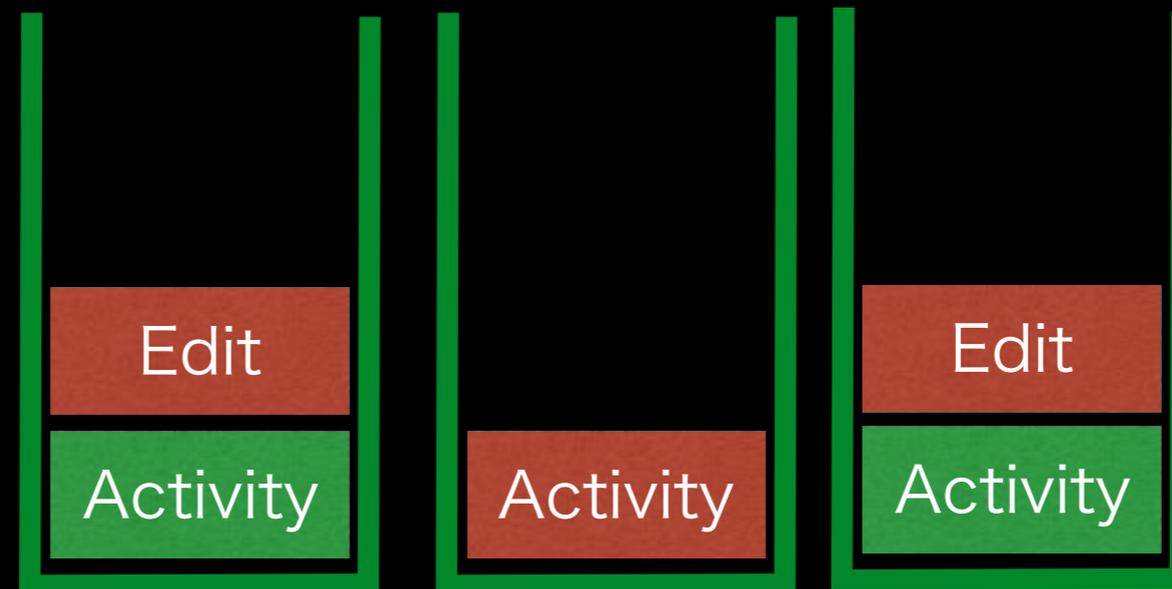
プロセス



# プロセスと コンポーネント



# 1プロセスで 複数アプリに存在



フォアグラウンド

# プロセス

- ・ 1プロセスで、複数のコンポーネントが動作する
- ・ コンポーネント (Activity) は、複数タスク内に存在することがある
- ・ static変数は、プロセス単位で保持される変数
  - ★ static変数で状態を持ったりすると。。。

# プロセス

いつ停止させられるか  
予測不能！

プロセス

停止前に

状態保存処理は

書けない！



# 問12

static変数にアクセストークンを  
保持しているアプリを  
クラッシュさせるには、どうすればよいですか？



# 答12

## アプリ

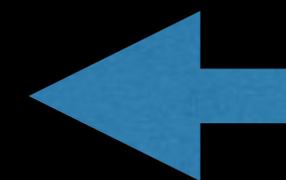
アクティビティを保持しない

ユーザーが離れたアクティビティを直ちに破棄する



バックグラウンドプロセスの上限

バックグラウンドプロセスを使用しない



# Activityまとめ

- ・ Activityオブジェクトのライフサイクルメソッドの呼ばれるタイミングを学んだよ
- ・ タスクと起動モードについて学んだよ
- ・ プロセスとタスクの関係について学んだよ

Fragment



# 主要人物

- Fragment
- FragmentManager
- バックスタック



# Fragment

- ・ レイアウトに貼って使うよ
  - ★なので、使い回ししやすいよ
- ・ 独自のレイアウトを持ってるよ
- ・ 独自のライフサイクルがあるよ



# FragmentManager

- ・ Fragmentを管理する人だよ
- ・ Fragmentのはりかえなどをやるよ
- ・ ActivityやFragmentが持ってるよ
- ★ FragmentのgetFragmentManager()は親のFragmentManagerを取ってくるよ



# バックスタック

- ・ Fragmentのほりかえ時に、状態をスタックに積めるよ
- ・ バックボタンで、1つ前の状態 (=画面) に戻れるようにするしくみ



# 問13

Fragmentは  
android.app.Fragmentと  
サポートライブラリ版の  
どちらを使うべきでしょうか？



答13

サポートライブラリ版！

Fragmentの  
ライフサイクル



# ライフサイクル

- ・ 基本はActivityと同じ
- ・ 見えなくなったら(`onStop()`が呼ばれたら)  
Viewだけ先に捨てる



# onCreateView()

- ・ このFragmentが提供するViewを作ってほしいときに呼ばれるメソッド
- ・ Viewを作ってreturnしよう



# 問14

親Activityが  
ユーザー情報を保持しています  
このユーザー情報をFragmentで  
取得してよいタイミングは  
いつでしょうか？



答14

onActivityCreated()



# onActivityCreated()

- ・ 親となるActivityがちゃんとつくられた後に  
よばれるメソッド
  - ★ onCreate()やonCreateView()の時点では  
ちゃんと作られていないことがある
- ・ 親Activityから情報を取りたいときは  
このメソッド内でとる



# onDestroyView()

- ・ このFragmentが提供するViewを破棄するタイミングでよばれる
- ・ フィールドにViewをセットしている場合はnullをいれてちゃんと破棄されるようにしましょう

★ 特にButterKnife

引数のある  
Fragment





# 問15

Activityでは、起動するためのIntentに  
引数データがセットできました  
これと同様なことを  
Fragmentで実現するには  
どうすればよいのでしょうか？



答15

setArguments()

※setterはアンチパターン

実行結果  
のある  
Fragment





# 問16

Activityでは、次Activityの実行結果を  
onActivityResult()で受け取れました  
これと同様なことを  
Fragmentで実現するには  
どのようにすればいいのでしょうか？



# 答16

1. `setTargetFragment()`で、呼び出し元Fragmentとリクエストコードをセット
2. `getTargetFragment()`で取り出して、`onActivityResult()`を呼ぶ

※コールバックオブジェクトを渡すのはアンチパターン

# Fragmentまとめ

- Fragmentがどのようなものか学んだよ
- 引数つきFragmentのつくりかたを学んだよ
- 呼び出し元に結果を伝える方法を学んだよ

ようやく  
アプリ開発の  
スタートラインに立てた！

クラッシュするアプリが  
減りますように