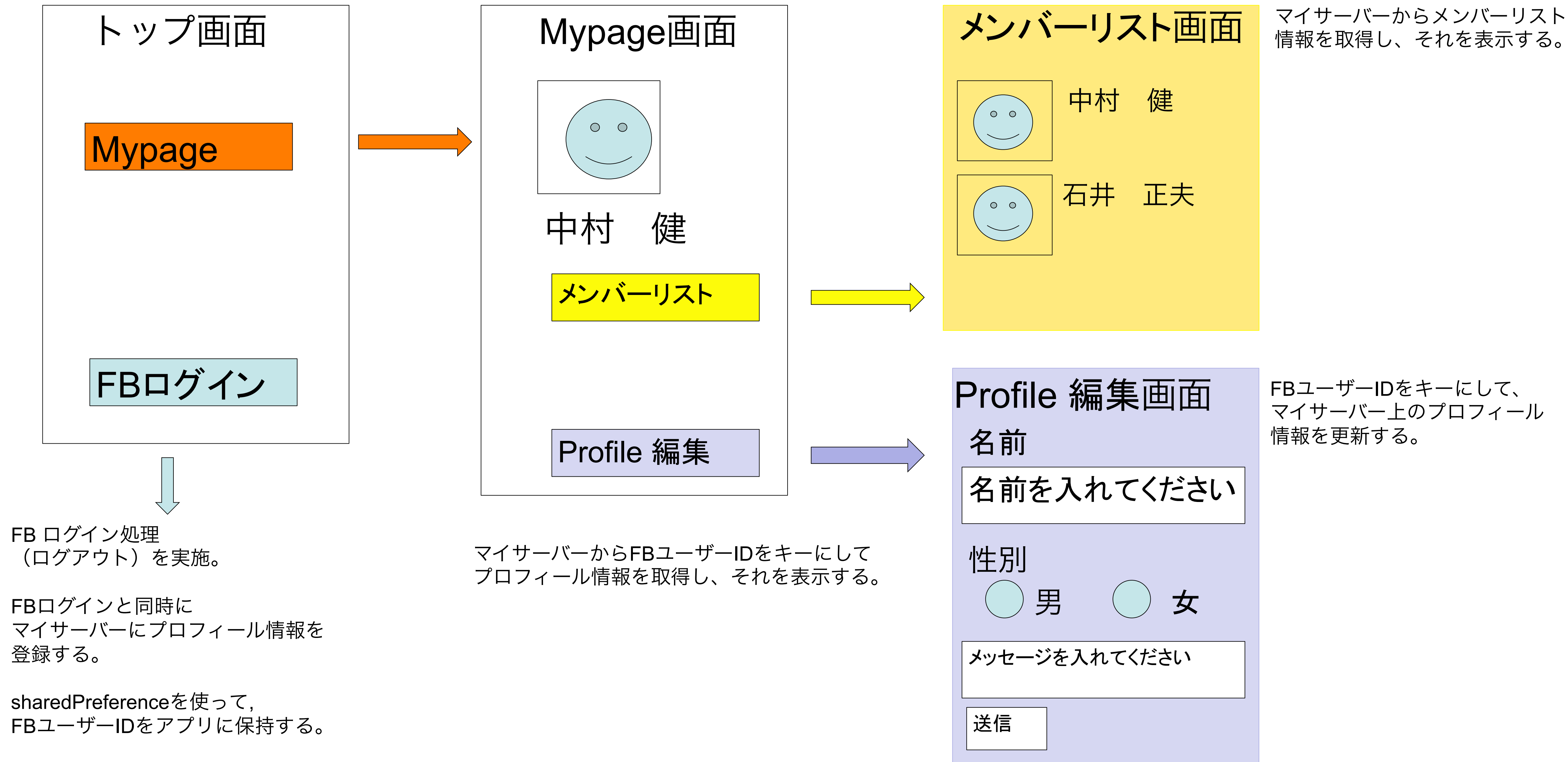




- メンバーリストアプリの作成
- Androidのデバッグについて
- リファクタリングについて
- カメラ機能と端末ギャラリーを使って写真を表示する。

# メンバーリストアプリの作成

# メンバーリストアプリの作成





# メンバーリストアプリの作成

## 各画面の画面仕様

- ・FBログインをする。
- ・取ってきたFBのプロフィールデータを下記のAPIを利用して、マイサーバー上のDBに登録する。

<http://nofiction.deca.jp/getGsproMemberAddList.php>

- ・FBのuser\_idを*SharedPreference*オブジェクトに保存する

- ・*SharedPreference*オブジェクトに保存したFBのuser\_idを取り出す。
- ・以下APIを利用して、FBuser\_idをキーにしてプロフィール情報を取得し、それを表示する。

<http://nofiction.deca.jp/getGsproProfile.php>

- ・メンバーリスト画面に遷移するボタンを用意する。
- ・プロフィール編集画面に遷移するボタンを用意する。

- ・以下のAPIを利用して、マイサーバーからメンバーリスト情報を取得し、Recyclerviewでメンバーリスト表示する。

[http://nofiction.deca.jp/getGsproMemberList.php?page=r98b\\$wk221oob](http://nofiction.deca.jp/getGsproMemberList.php?page=r98b$wk221oob)

- ・*SharedPreference*オブジェクトに保存したFBのuser\_idを取り出す。
- ・以下のAPIを利用して、user\_idをキーにして、マイサーバー上のプロフィール情報を更新する。

<http://nofiction.deca.jp/getGsproMemberUpdateList.php>

## マイサーバーのAPI一覧

//プロフィール情報新規追加API

<http://nofiction.deca.jp/getGsproMemberAddList.php>

//メンバー一覧取得API

[http://nofiction.deca.jp/getGsproMemberList.php?page=r98b\\$wk221oob](http://nofiction.deca.jp/getGsproMemberList.php?page=r98b$wk221oob)

//プロフィール情報取得API

<http://nofiction.deca.jp/getGsproProfile.php>

//プロフィール情報更新API

<http://nofiction.deca.jp/getGsproMemberUpdateList.php>

# Androidのデバッグについて

## コンパイルエラー

プログラムの中に文法上の間違いがある場合はコンパイル時に発生するエラー

## ランタイムエラー

コンパイルされた実行ファイルの実行時に発生するエラー



## エラーの例

NullPointerException

ArrayIndexOutOfBoundsException

java.lang.OutOfMemoryError:

android.view.InflateException: Binary XML file

NetworkDispatcher.run: Unhandled exception java.lang.SecurityException: Permission denied (missing INTERNET permission?)  
java.lang.SecurityException: Permission denied (missing INTERNET permission?)

java.lang.SecurityException: Permission Denial: reading com.android.providers.media.MediaProvider uri content://media/external/images/media/5503 from pid=5779, uid=10506 requires android.permission.READ\_EXTERNAL\_STORAGE

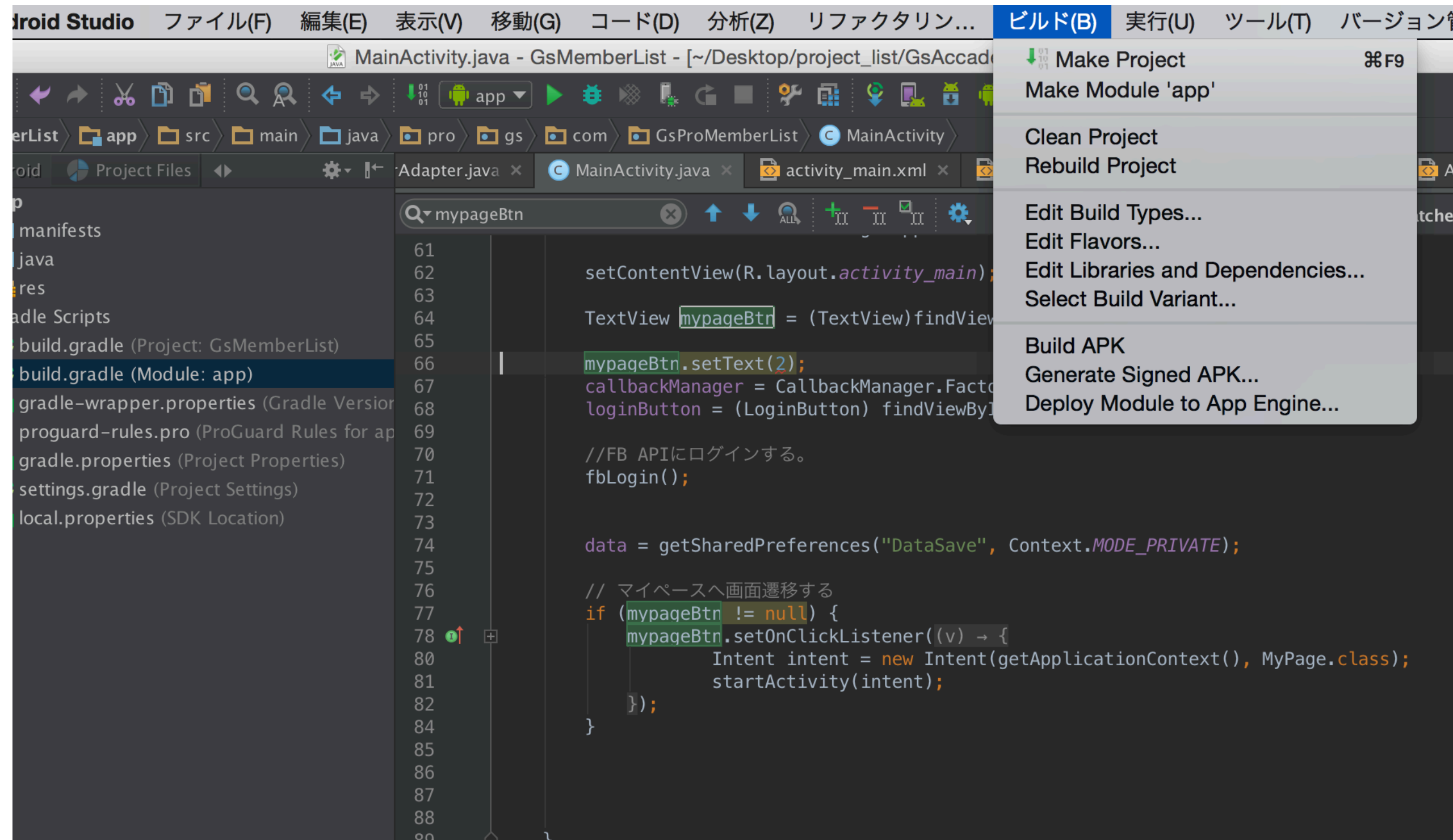
ActivityNotFoundException: Unable to find explicit activity class {pro.gs.com.fbfirebase/pro.gs.com.GsProMemberList.MyPage}; have you declared this activity in your AndroidManifest.xml?

## エラーの例

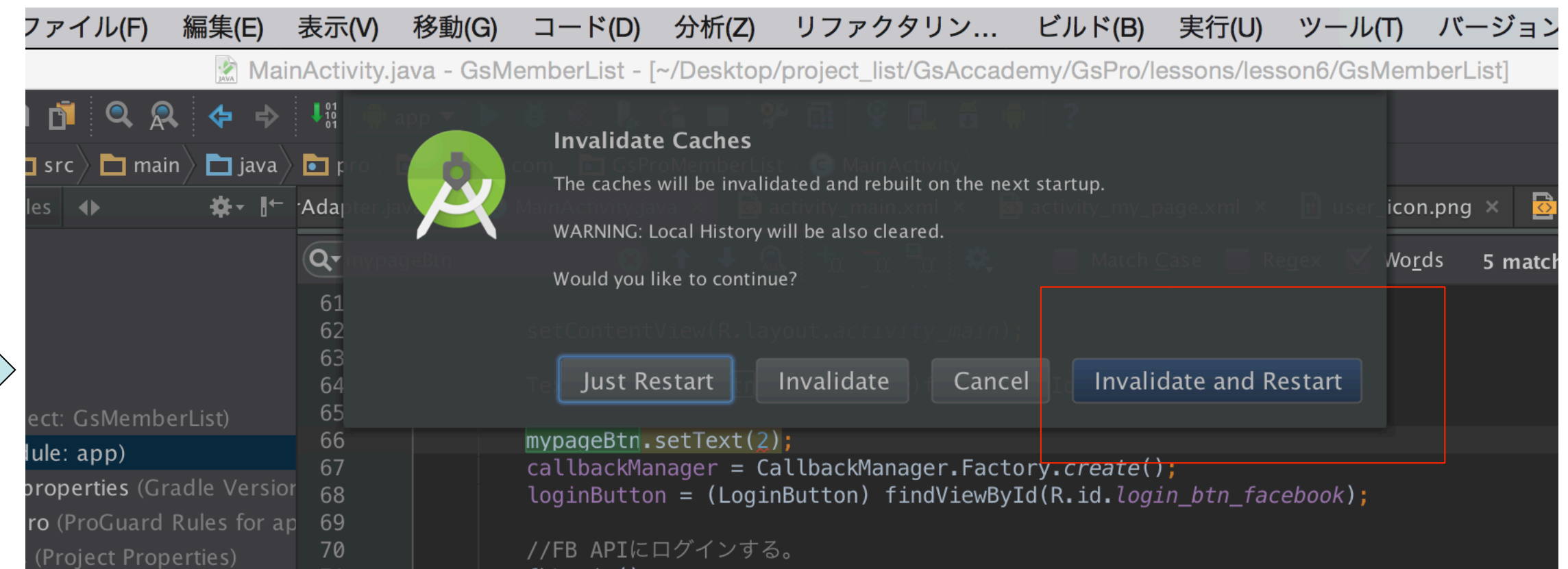
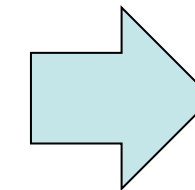
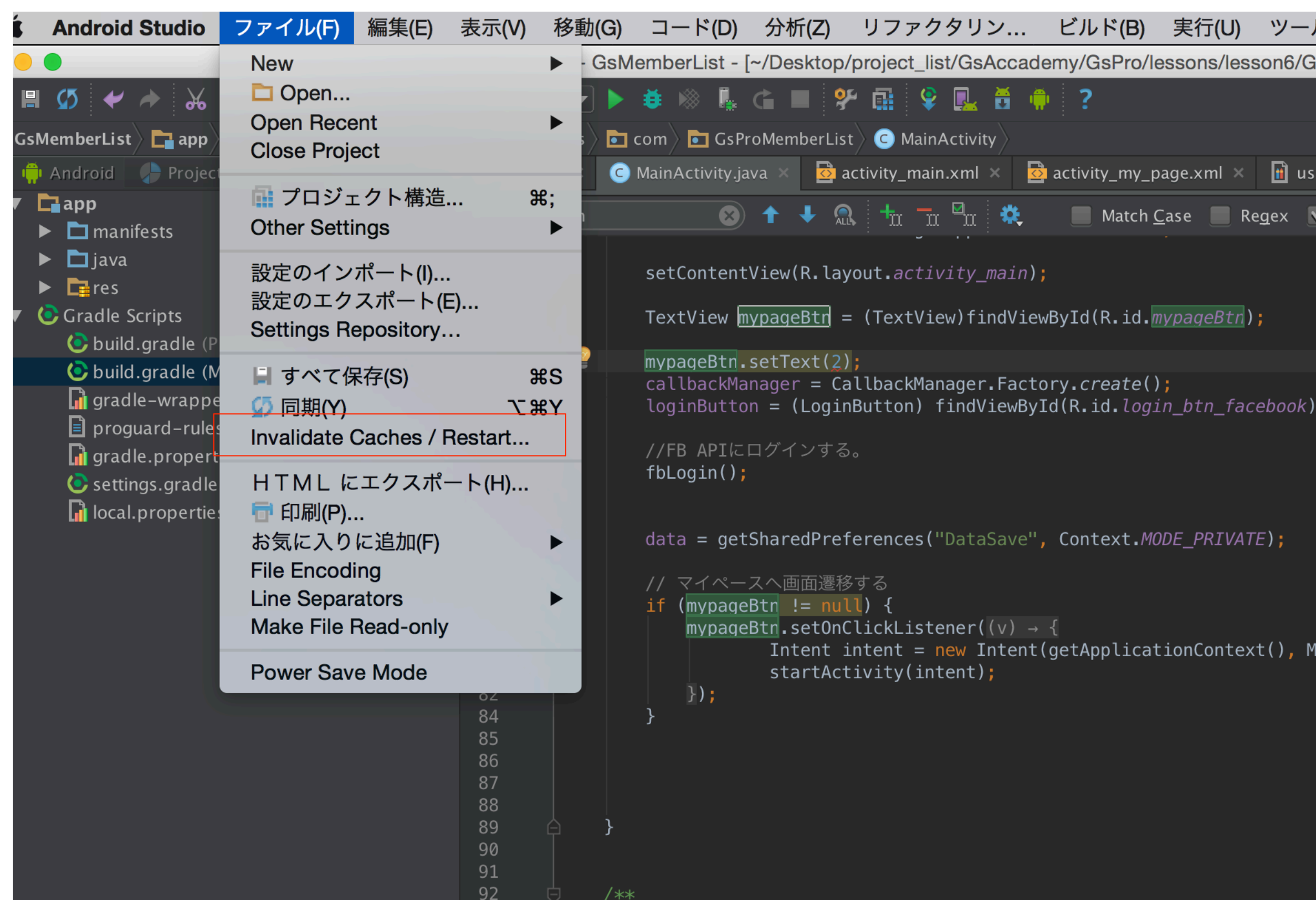
```
java.lang.RuntimeException: Unable to start activity ComponentInfo{pro.gs.com.GsProMemberList/pro.gs.com.GsProMemberList.MainActivity}:  
android.content.res.Resources$NotFoundException: String resource
```

```
int price = 1000;  
((TextView) view.findViewById(R.id.price)).setText(price);
```

エラーが消えない場合  
Clean Projectを実行して、キャッシュをクリアする。



それでも不可解にエラーが消えない場合  
Invalidate Caches/Restartを実行して、全キャッシュを削除して、  
プロジェクトを立ち上げ直す。キャッシュが邪魔をして、コンパイルエラーが  
解消されないことがあります。



## Android Studioのデバッグ実行について

<http://qiita.com/FumihikoSHIROYAMA/items/e3159103ba2fba07e063>  
を参考



# リファクタリングについて

# リファクタリングについて

---

リファクタリング（refactoring）とは、  
プログラムの振る舞いを変えずにソースコードを変更すること。

カメラ機能と端末ギャラリーを使って  
写真を表示する。

# アクティビティのライフサイクル

# アクティビティのライフサイクル

ライフサイクルとは？

Activityの起動・停止・再生・終了・破棄などの流れ。

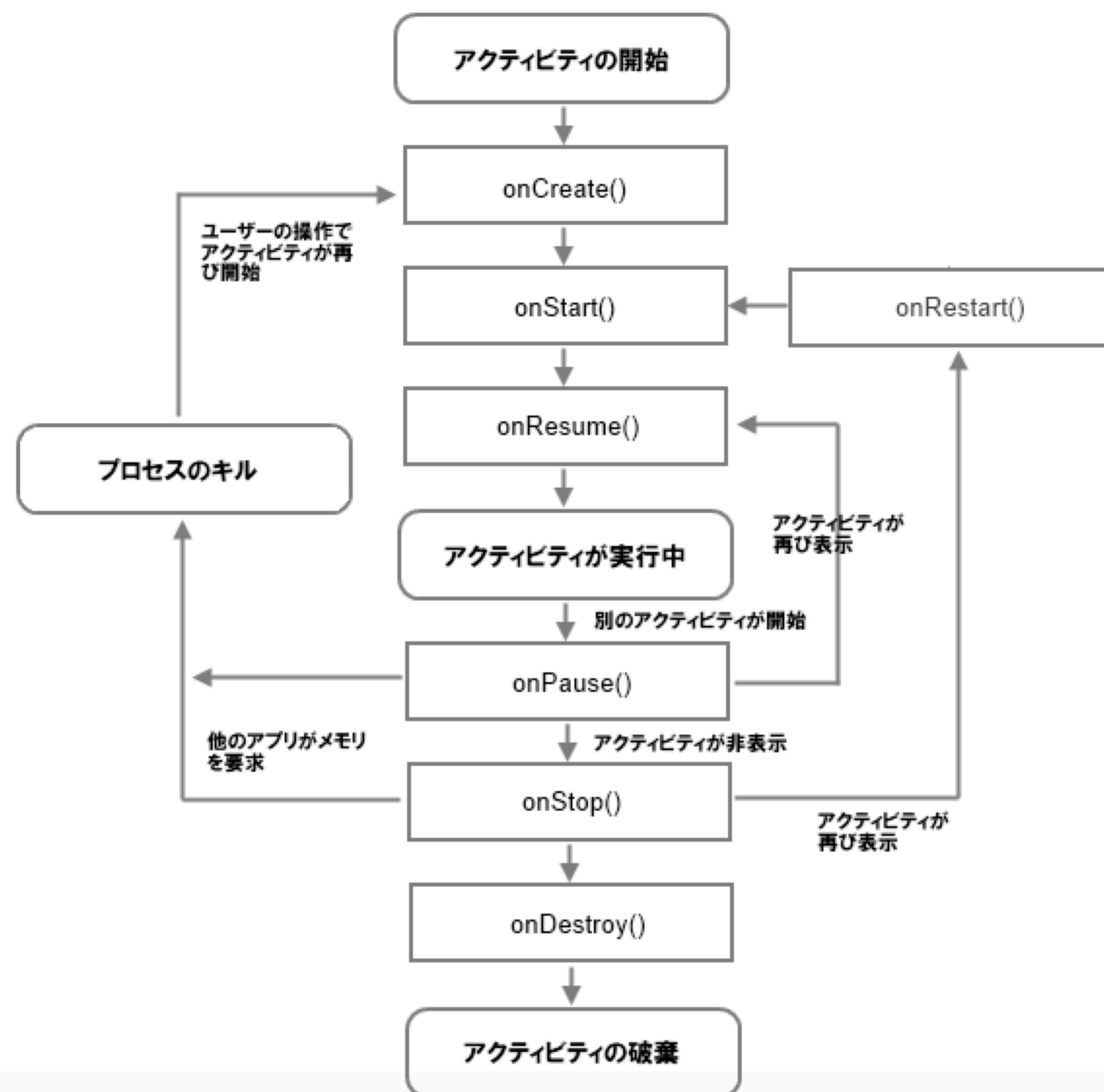
Activityの起動状況（起動・停止・再生・終了・破棄など）によって、異なるActivityのライフサイクル用メソッドが実行される。

ライフサイクル用メソッドとは、Activityがバックグラウンドとフォアグラウンドを行き交う過程の処理の為のメソッド。  
onCreate/onResume/onPause/onStopなど

Activityが持つライフサイクル用のメソッドをoverrideすることにより、起動状況毎にプログラムの処理を書くことが出来る。



# アクティビティのライフサイクル



Activityクラスが持っているライフサイクル用の各メソッドをoverrideして、起動状況毎の処理を実行する。

```

@Override
protected void onCreate(Bundle savedInstanceState) {
    super.onCreate(savedInstanceState);
}

@Override
protected void onStart() {
    super.onStart();
    Log.d("gspro_activity_life", "onStart");
}

@Override
protected void onPause() {
    super.onPause();
    Log.d("gspro_activity_life", "onPause");
}

@Override
protected void onResume() {
    super.onResume();
    Log.d("gspro_activity_life", "onResume");
}

@Override
protected void onRestart() {
    super.onRestart();
    Log.d("gspro_activity_life", "onRestart");
}

@Override
protected void onDestroy() {
    super.onDestroy();
    Log.d("gspro_activity_life", "onDestroy");
}

@Override
protected void onStop() {
    super.onStop();
    Log.d("gspro_activity_life", "onStop");
}
  
```