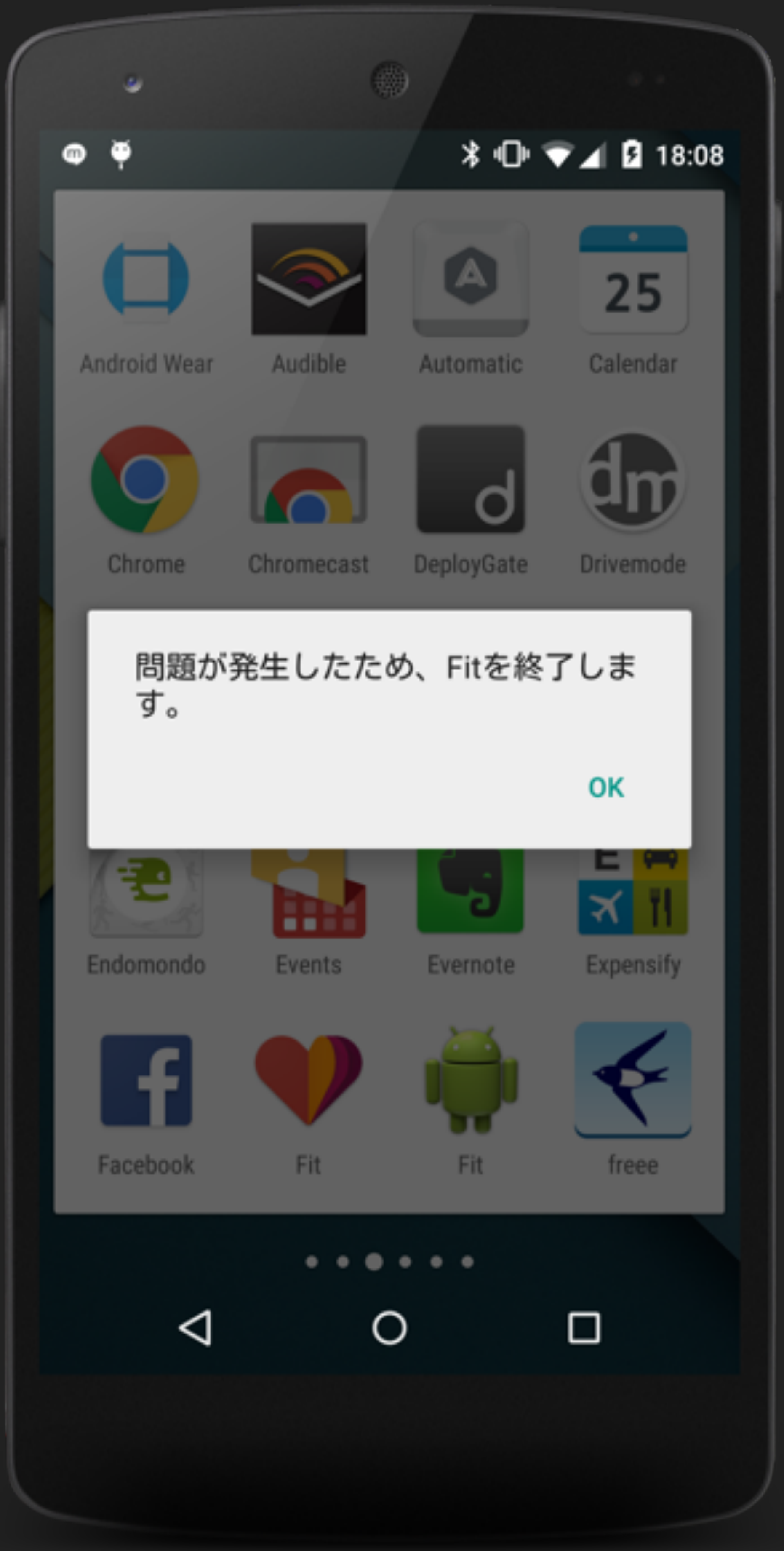
Null, the Abyss

@KeithYokoma - Drivemode, Inc.

potatotips #17

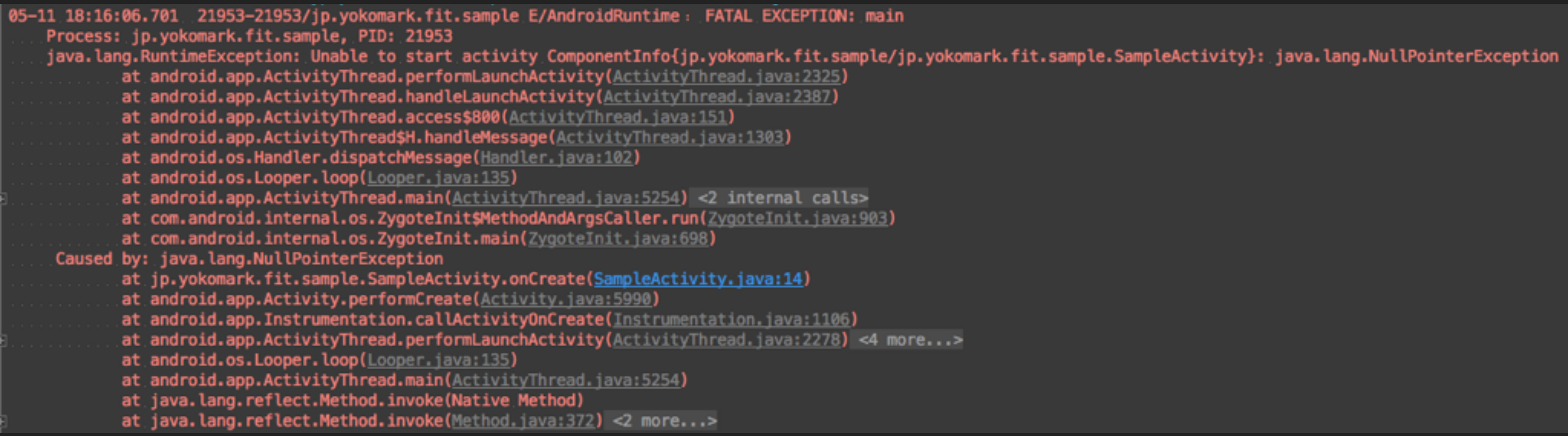
KeithYokoma

Keishin Yokomaku
Drivemode, Inc.
Android Engineer
GitHub: https://github.com/KeithYokoma
Qiita: http://qiita.com/KeithYokoma
e-Book: http://amzn.to/1mZNydv

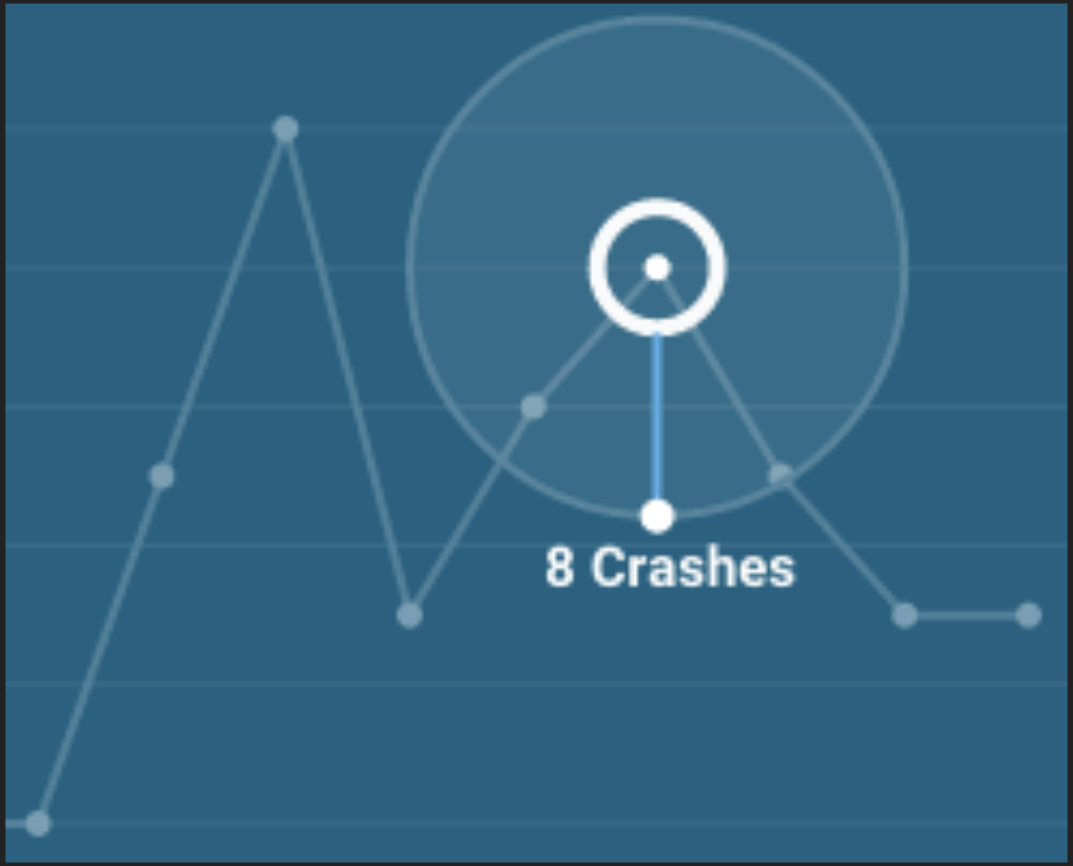


問題が発生したため、Fitを終了します。

★☆☆☆☆



```
05-11 18:16:06.701  21953-21953/jp.yokomark.fit.sample E/AndroidRuntime﹕ FATAL EXCEPTION: main
    Process: jp.yokomark.fit.sample, PID: 21953
    java.lang.RuntimeException: Unable to start activity ComponentInfo{jp.yokomark.fit.sample/jp.yokomark.fit.sample.SampleActivity}: java.lang.NullPointerException
            at android.app.ActivityThread.performLaunchActivity(ActivityThread.java:2325)
            at android.app.ActivityThread.handleLaunchActivity(ActivityThread.java:2387)
            at android.app.ActivityThread.access$800(ActivityThread.java:151)
            at android.app.ActivityThread$H.handleMessage(ActivityThread.java:1303)
            at android.os.Handler.dispatchMessage(Handler.java:102)
            at android.os.Looper.loop(Looper.java:135)
            at android.app.ActivityThread.main(ActivityThread.java:5254) <2 internal calls>
            at com.android.internal.os.ZygoteInit$MethodAndArgsCaller.run(ZygoteInit.java:903)
            at com.android.internal.os.ZygoteInit.main(ZygoteInit.java:698)
     Caused by: java.lang.NullPointerException
            at jp.yokomark.fit.sample.SampleActivity.onCreate(SampleActivity.java:14)
            at android.app.Activity.performCreate(Activity.java:5990)
            at android.app.Instrumentation.callActivityOnCreate(Instrumentation.java:1106)
            at android.app.ActivityThread.performLaunchActivity(ActivityThread.java:2278) <4 more...>
            at android.os.Looper.loop(Looper.java:135)
            at android.app.ActivityThread.main(ActivityThread.java:5254)
            at java.lang.reflect.Method.invoke(Native Method)
            at java.lang.reflect.Method.invoke(Method.java:372) <2 more...>
```


8 Crashes

★☆☆☆☆

★☆☆☆☆

★☆☆☆☆

“A lot of sadness is coming from NullPointerException”

–Someone

Avoid NullPointerException

- Null check for nullable value
- Using support annotation
- Using NullObject pattern
- Returning null value
- Null abyss in Android

Null check statement

```
public void method(Object arg) {
  if (arg == null) {
    throw new NullPointerException();
  }

  // …
}
```

Null check statement

```
public void method(Object arg) {
  if (arg == null) {
    throw new NullPointerException();
  }

  // …
}
```

Null check statement

✓ Easy and basic way to avoid NPE
✗ Too many checks is bad for performance
✗ Need documentation for nullity
✗ Failure at runtime

Support Annotations

```
public void method(@Nullable Object arg) {
  if (arg == null) {
    Log.w(“Logger”, “Nothing I can do.”)
    return;
  }

  // …
}
```

Support Annotations

```
public void method(@Nullable Object arg) {
  if (arg == null) {
    Log.w(“Logger”, “Nothing I can do.”)
    return;
  }

  // …
}
```

Support Annotations

✓ Gain Lint support for nullity

✓ Users can understand about nullity

✗ Still have possibility of null value

✗ Failure at runtime

Using NullObject pattern

✓ No possibility of null value
✓ No change to get NullPointerException
✗ Need to learn architecture

Using NullObject pattern
Bad Practice

```java
public enum NavMenu {
  HOME(R.id.home), PROFILE(R.id.profile);

  public static NavMenu get(int id) {
    for (NavMenu menu : values()) {
      if (menu.id == id) return menu;
    }
    return null;
  }
}
```

Using NullObject pattern
Good Practice

```
public enum NavMenu {
  HOME(R.id.home), PROFILE(R.id.profile), NONE(-1);

  public static NavMenu get(int id) {
    for (NavMenu menu : values()) {
      if (menu.id == id) return menu;
    }
    return NONE;
  }
}
```

Returning null value
Bad Practice

```
public List<Result> get(int count) {
  if (something went wrong) {
    return null;
  }

  //…
}
```

Returning null value
Good Practice

```
public List<Result> get(int count) {
  if (something went wrong) {
    return new ArrayList<>();
  }

  //…
}
```

Returning null value
Other Good Practice

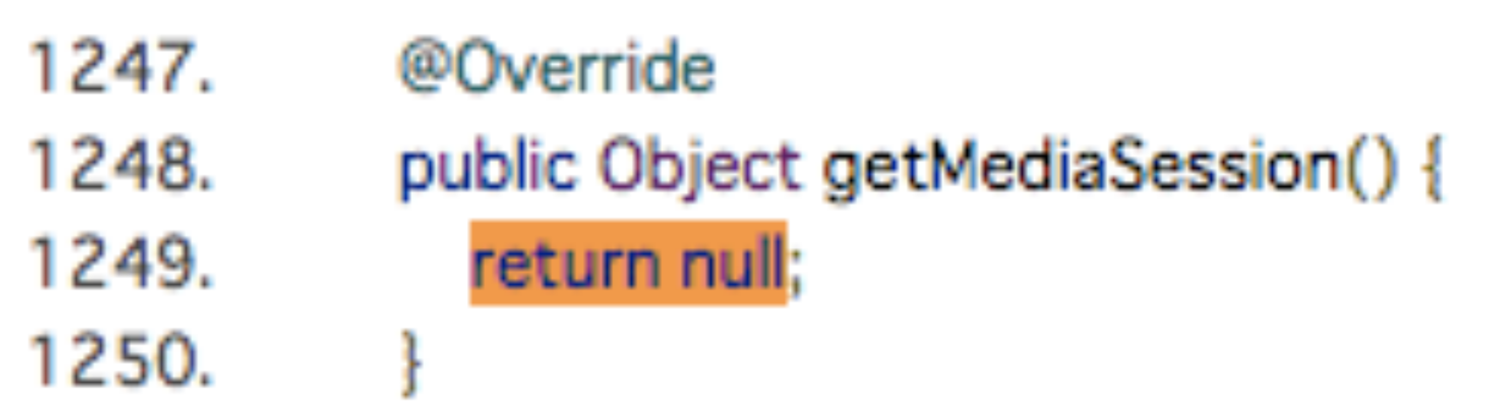
```
public List<Result> get(int count) {
  if (something went wrong) {
    throw new SomeException(“Request failed for some reason.”);
  }

  //…
}
```

Returning null value

- “null” means value is absent
- Empty collection instead of null
- Failure for throwing Exception

Null abyss in Android

- Some support API returns “null”
  - e.g. MediaSessionCompat
- You need to verify nullity for those APIs…



```
@Override
public Object getMediaSession() {
    return null;
}
```

“When you gaze into null, null gazes into you”

–Someone

For more details…

- Effective Java

Null, the Abyss

@KeithYokoma - Drivemode, Inc.

potatotips #17