Create your Gradle plugin

Nabil Hachicha

About Me

Open Source contributor

snappydb.com

Works at The App Business London

@nabil_hachicha   @nhachicha

What is a Gradle Plugin?

Motivation

- A reusable fragment of build logic

- Define the How (Tasks) & the What (DSL)

Motivation
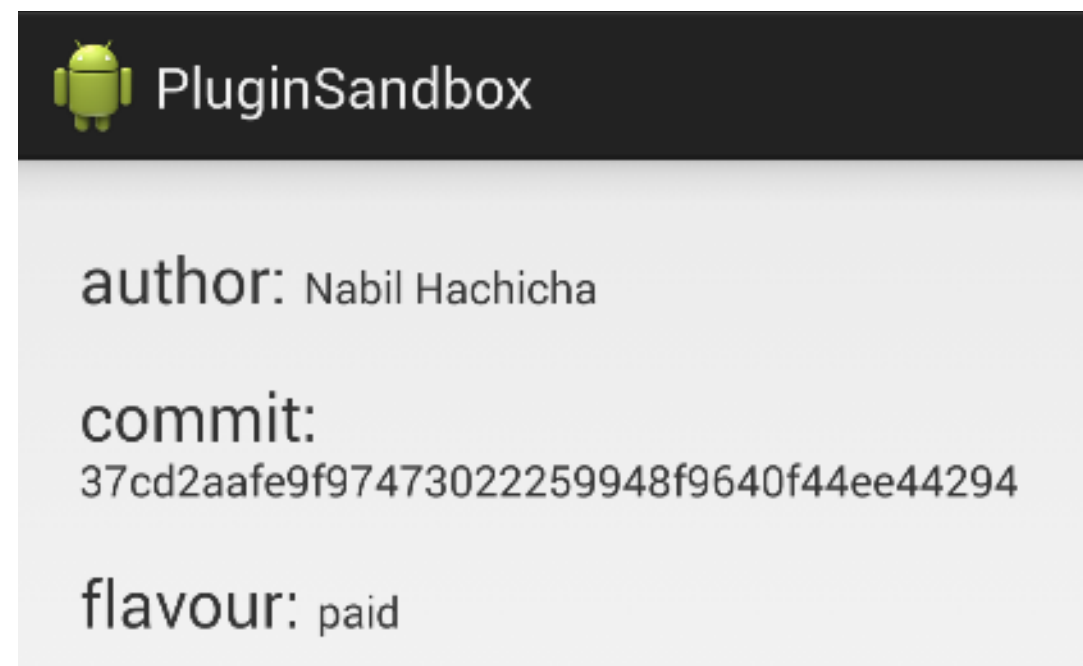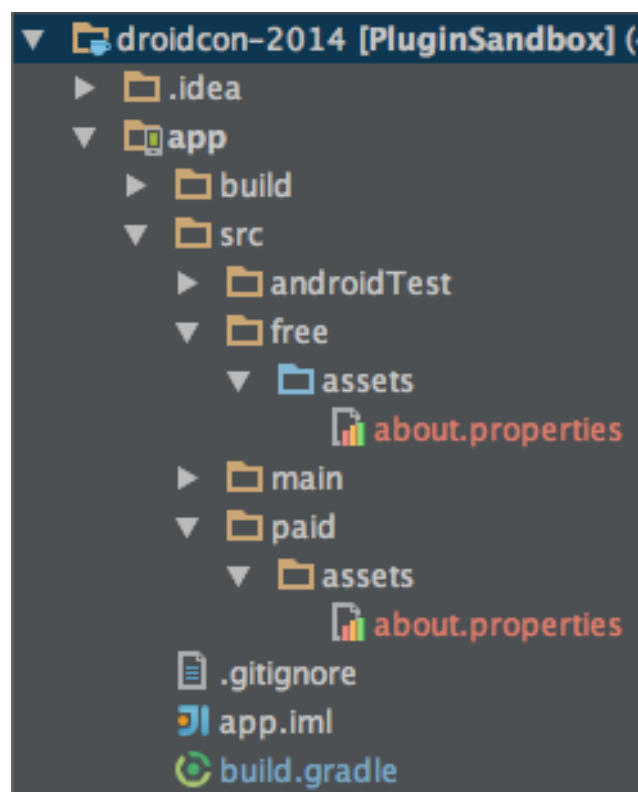
What are the benefits?

Motivation

- Keeping the build simple & concise

- Reduce boilerplate build code (no copy/paste)

- Separation of concern How (Tasks) vs the What (configuration)

Use case study

A plugin that generates a properties file

Gradle Tasks

Demo

Writing a lifecycle hook

```
app:assembleFreeDebug - Assembles the Debug build for flavor Free
    app:checkFreeDebugManifest
    app:compileFreeDebugAidl
    app:compileFreeDebugJava
    app:compileFreeDebugNdk
    app:compileFreeDebugRenderscript
    app:dexFreeDebug
    app:displayRevisionFoRfree
    app:generateFreeDebugAssets
    app:generateFreeDebugBuildConfig
    app:generateFreeDebugResValues
    app:generateFreeDebugResources
    app:generateFreeDebugSources
    app:mergeFreeDebugAssets
    app:mergeFreeDebugResources
    app:packageFreeDebug
    app:preBuild
    app:preDexFreeDebug
    app:preFreeDebugBuild
    app:prepareFreeDebugDependencies
    app:processFreeDebugJavaRes
    app:processFreeDebugManifest
    app:processFreeDebugResources
    app:validateDebugSigning
    app:zipalignFreeDebug
```

Writing a lifecycle hook

```
app:assembleFreeDebug - Assembles the Debug build for flavor Free
    app:checkFreeDebugManifest
    app:compileFreeDebugAidl
    app:compileFreeDebugJava
    app:compileFreeDebugNdk
    app:compileFreeDebugRenderscript
    app:dexFreeDebug
    app:displayRevisionFoRfree
    app:generateFreeDebugAssets
    app:generateFreeDebugBuildConfig
    app:generateFreeDebugResValues
    app:generateFreeDebugResources
    app:generateFreeDebugSources
    app:mergeFreeDebugAssets
    app:mergeFreeDebugResources
    app:packageFreeDebug
    app:preBuild
    app:preDexFreeDebug
    app:preFreeDebugBuild
    app:prepareFreeDebugDependencies
    app:processFreeDebugJavaRes
    app:processFreeDebugManifest
    app:processFreeDebugResources
    app:validateDebugSigning
    app:zipalignFreeDebug
```

dependsOn → displayRevision

Writing a lifecycle hook

Demo

Extra properties

- User-defined variables

- Ex: use it to disable or enable the plugin for a specific flavour

```
android {
    compileSdkVersion 20
    buildToolsVersion "20.0.0"

    defaultConfig {...}
    buildTypes {...}

    productFlavors {
        free {
            ext.enableAboutPlugin = true;

        }
        paid

    }
}
```

Custom Task

- **Structure** your code into classes & methods

- **Maintainability** (any method is fully testable)

- **Reusability**

DSL

- **Task** add new functionality (ability **To Do** Things – they are unit of works)

- **Extension** add new build language elements **To Talk** about Things

```
android {
    compileSdkVersion 20
    buildToolsVersion "20.0.0"

    defaultConfig {...}
    buildTypes {...}

    productFlavors {...}
}

dependencies {...}

// customising our plugin

about {
    fileName = "about"
}
```

Incremental Build

- Input haven't changed & Output is still there = Skip redundant work.

  but...

- You need to tell gradle what's the Input & Output

Thanks!

Q/A

THE APP BUSINESS

The App Business is hiring!

theappbusiness.com/careers/

@nabil_hachicha   @nhachicha