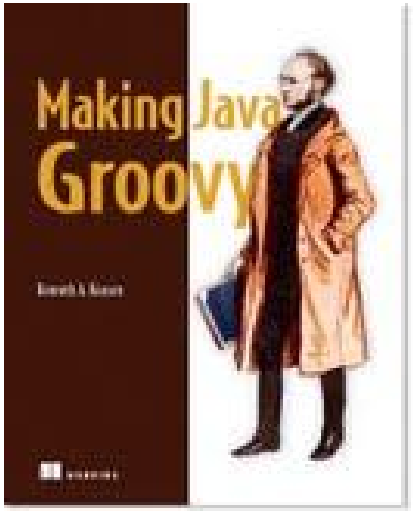
Android Development with Gradle

Using the Android Plugin for Gradle

Contact Info

Ken Kousen
ken.kousen@kousenit.com
@kenkousen



Making Java Groovy

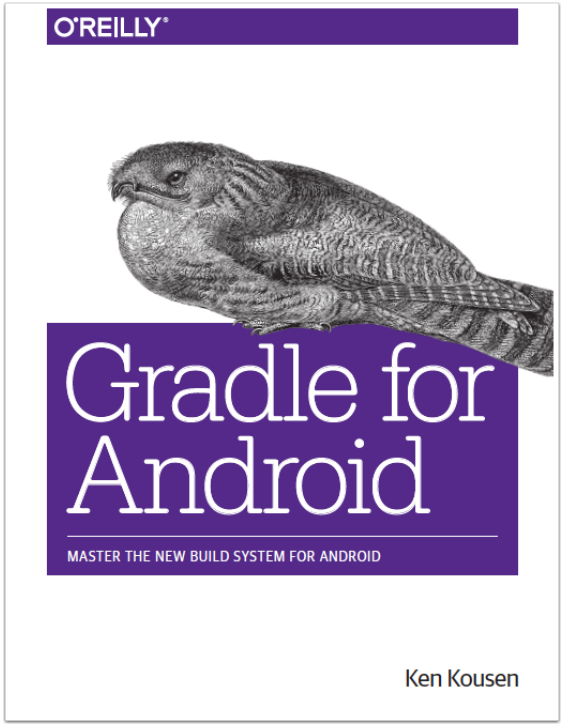
http://manning.com/kousen

Upcoming Book

Gradle for Android

O'Reilly

Part of the Gradle series



Android Home Page

Developer home page

http://developer.android.com

SDK Bundle

https://developer.android.com/sdk/index.html

Eclipse + ADT tools

Android SDK tools

Android Platform tools

Latest Android SDK + emulator

Eclipse ADT

Does not support Gradle directly

Generate gradle build file

- maps default structure to standard layout
- no longer needed → can import project

Gradle

Why Gradle?

Gradle

Because reasons

Gradle

Because reasons

(This is a Gr8 conference.
You already know why)

Android Studio

https://developer.android.com/sdk/installing/studio.html

- Early Access Preview

Android Studio

https://developer.android.com/sdk/installing/studio.html

- Early Access Preview
- Based on IntelliJ IDEA

Android Studio

https://developer.android.com/sdk/installing/studio.html

- Early Access Preview
- Based on IntelliJ IDEA
- Uses Gradle for builds

Android Studio

https://developer.android.com/sdk/installing/studio.html

- Early Access Preview
- Based on IntelliJ IDEA
- Uses Gradle for builds
  - *Beta → some bugs and unsupported features, but pretty solid so far*

Android Studio

Actually, latest version* is very useful

*As of Gr8conf.us, 0.8.4

Creating an application

Must select unique package name

`com.example.myapp`

(com.example reserved for samples)

Used in Google Play store

Creating an application

Choose min SDK level

Choose target SDK level

Manifest

AndroidManifest.xml

```
<uses-sdk … />
<application>
   <activity>... </activity>
   …
</application>
```

Change for Gradle

Now `minSdkVersion`, `targetSdkVersion` set in `build.gradle`

Overrides AndroidManifest.xml

Manifest

All activities must be declared

List permissions

Intent filters

Services

Content providers

…

Activities

Represents a screen

- Extend `android.app.Activity`

Full of callback methods

Activities

Each activity has an XML layout

- `activity_main.xml`
- `activity_welcome.xml`

XML tags with many attributes

Activities

Callback methods:

```
onCreate, onDestroy
onStart,  onStop
onPause,  onResume
```

… many others …

res

Resources folder contains subfolders

- drawable
- layout
- menu
- values
- ...

Providing resources

https://developer.android.com/guide/topics/resources/providing-resources.html

- Specially named subdirectories
  - values
- Configuration qualifiers
  - values-v11
  - values-sw720dp-land

values

keys and values → layer of indirection

strings.xml:

```
<string name="hello_world">Hello world!</string>
```

Accessing resources

XML → compiled into

R.java: full of public inner classes (!)

```
(Button) findViewById(R.id.hello_button)
```

Basics

Android plugin for Gradle

- Added via `buildScript`

- Lots of customization

Basics

```
buildscript {
    repositories { mavenCentral() }  // or jcenter()
    dependencies {
        classpath 'com.android.tools.build:gradle:0.12.+'
    }
}

apply plugin: 'com.android.application'
```

Properties in build.gradle

```
android {
   versionCode …
   versionName …
}
```

or even in gradle.properties

Multiproject Builds

By default, apps in AS are multiproject builds

```
build.gradle
settings.gradle
```

Can add additional libraries, ...

Build Types

Two default build types:

- `debug`
- `release`

Configuring Build Types

Use buildTypes section of build.gradle

```
buildTypes {
   release { … }
   debug { … }
}
```

Can also add custom build types that way

Configuring Build Types

Each build type defines a source set

`src/main/ ...`

`src/debug/ ...`

`src/release/ ...`

`src/androidTest/...` (discussed below)

Configuring Build Types

Resources in build type source sets

replace their counterparts in main

Java classes *conflict*, however

Define class in each, or just in main

Generating a Release

Can't assemble a release until

you can sign it

Signing Your App

Use Java's **`keytool`** to generate cert

```
AndroidStudio/ICNDB_AS/app
▶ keytool -genkey -v -keystore ICNDB.keystore -alias ICNDB -keyalg RSA -keysize 2048 -validity 10000
Enter keystore password:
Re-enter new password:
What is your first and last name?
  [Unknown]:  Ken Kousen
What is the name of your organizational unit?
  [Unknown]:
What is the name of your organization?
  [Unknown]:  Kousen IT, Inc.
What is the name of your City or Locality?
  [Unknown]:  Marlborough
What is the name of your State or Province?
  [Unknown]:  CT
What is the two-letter country code for this unit?
  [Unknown]:  US
Is CN=Ken Kousen, OU=Unknown, O="Kousen IT, Inc.", L=Marlborough, ST=CT, C=US correct?
  [no]:  yes

Generating 2,048 bit RSA key pair and self-signed certificate (SHA256withRSA) with a validity of 10,000 days
        for: CN=Ken Kousen, OU=Unknown, O="Kousen IT, Inc.", L=Marlborough, ST=CT, C=US
Enter key password for <ICNDB>
        (RETURN if same as keystore password):
Re-enter new password:
[Storing ICNDB.keystore]
```

Signing Your App

Add signing block to `build.gradle`

```
signingConfigs {
    release {
        storeFile file('ICNDB.keystore')
        keyAlias 'ICNDB'
        storePassword 'gradle_rules'
        keyPassword 'carlos_ray_aka_chuck'
    }
}
```

Signing Your App

Passwords don't have to be in build file

- Can use system properties
- Can prompt user
- Can use external files

See docs for suggestions

Signing Your App

Add `signingConfig` to build type config

```
buildTypes {
    release {
        // …
        signingConfig signingConfigs.release
    }
}
```

Signing Your App

The **`signingReport`** task shows details

Signing Your App

Invoke **`assembleRelease`** task

Resulting apk in `build/outputs/apk` folder

Testing

Use `androidTest` source set

`src/androidTest/java`

Testing

Use `androidTest` source set

`src/androidTest/java`

`compileAndroidTest` dependencies

Testing

Use `androidTest` source set

`src/androidTest/java`

`androidTestCompile` dependencies

Run **`connectedCheck`** task

Flavors and Variants

buildTypes

debug, release

flavors → different versions of same app

free, paid

Flavors and Variants

Each flavor generates an apk

Variants combine buildTypes and flavors

free+debug, free+release,

paid+debug, paid+release

Multiple Flavors

See flavors sample

See basicMultiFlavors sample

```
flavorDimensions …
productFlavors { … }
```

REST

Instead of JSON and Apache HTTP:

Spring for Android

http://projects.spring.io/spring-android/

`RestTemplate` class

Map classes to JSON structure

References

Android new build system

http://tools.android.com/tech-docs/new-build-system

http://tools.android.com/tech-docs/new-build-system/user-guide

Developer's Guide section on Gradle

https://developer.android.com/sdk/installing/studio-build.html

Android Developers on G+

https://plus.google.com/+AndroidDevelopers/posts

References

Xavier Ducrohet on G+

https://plus.google.com/+XavierDucrohet/posts

adt-dev on Google Groups

https://groups.google.com/forum/#!forum/adt-dev

The Ultimate Android OS

Spock Testing

Shouldn't Data run on Android?

The Ultimate Android OS