

Gradle- the basics

Gradle is a build system, like make, it can be very complex. Fortunately Android Studio (AS) handles most of the gradle tasks for you with some exceptions. For instance changing what APIs your device targets, or if you add new components that require adding their gradle dependencies. These often come up when importing someone else's project that uses different SDKs (like when you import my projects), or when upgrading projects. This is a short guide on dealing with those situations.

Open Project

build.gradle (for the app)

first lib or application (see plugins section in gradle file)

```
plugins {  
    id 'com.android.application'}
```

- lib means your building a library
- app means your building an application

'com.android.library' or 'com.android.application'

Check your compileSdk

in general use the latest you have installed, check SDK manager occasionally to see if there are updates

is the version installed? (red squiggles if not) If not AS will ask you if you want to install it, careful! They are >1GB. or change to one that you have (see SDK Manager)

defaultconfig

applicationId: project packagename

which versions you support min to target and all in between

buildtypes (not really relevant in this class)

used to support different project flavors for instance a freemium verses paid

minifyEnabled and proguard are security and obfuscation for release builds

dependencies

libraries you need (if possible build with jetpack, AndroidX support which makes all the V4 or V7 appcompatlibrary issues go away. If updating old project can hit Refactor → Migrate to AndroidX (bit glitchy) or when creating project on the Configure your project screen, select 'Use AndroidX Artifacts')

(see AndroidX Overview

<https://developer.android.com/jetpack/androidx>

and Migrating to AndroidX

see <https://developer.android.com/jetpack/androidx/migrate>

)

gradle.properties

Make sure the following line is NOT commented out if using AndroidX

android.useAndroidX=true

settings.gradle – whats in this project (project(s), libraries), where to look for packages (google maven central etc)

Once this is all set make sure you use the appropriate packages in your java files and in your XML files (it will not compile if you do not)

And sometimes, gradle is just not going to work well (its complex). Often the problem is that it cannot resolve to a particular package to include in the apps build.gradle dependencies section. For example see <https://developer.android.com/jetpack/androidx/releases/preference>