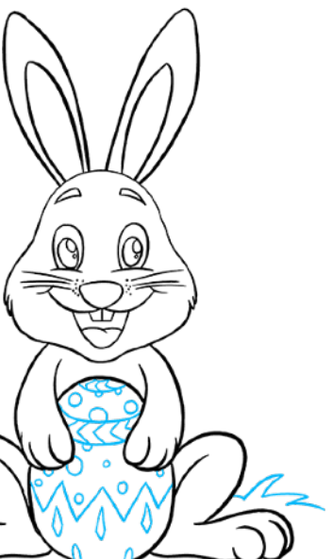
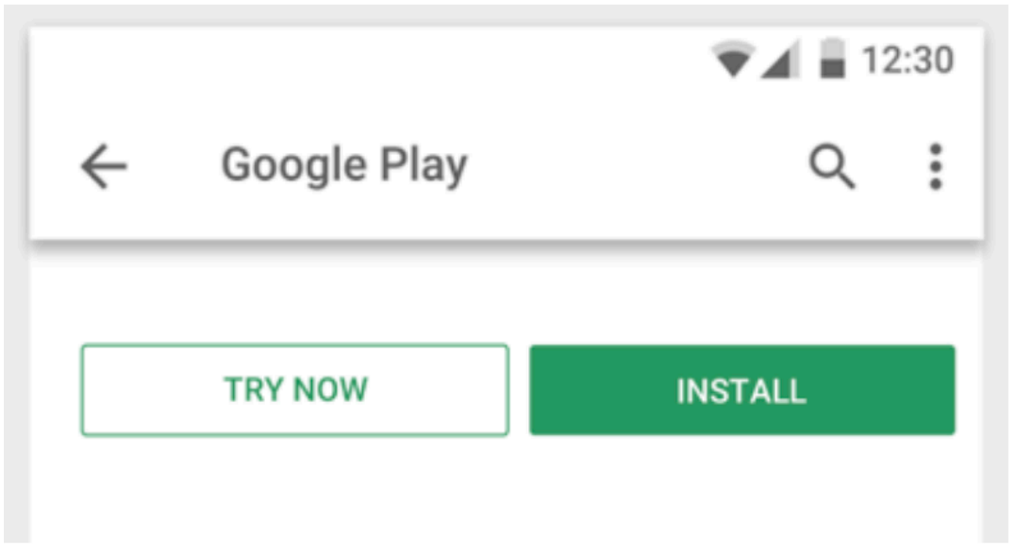
Lecture #9
Instant Apps

Android Things 2019

May 9th

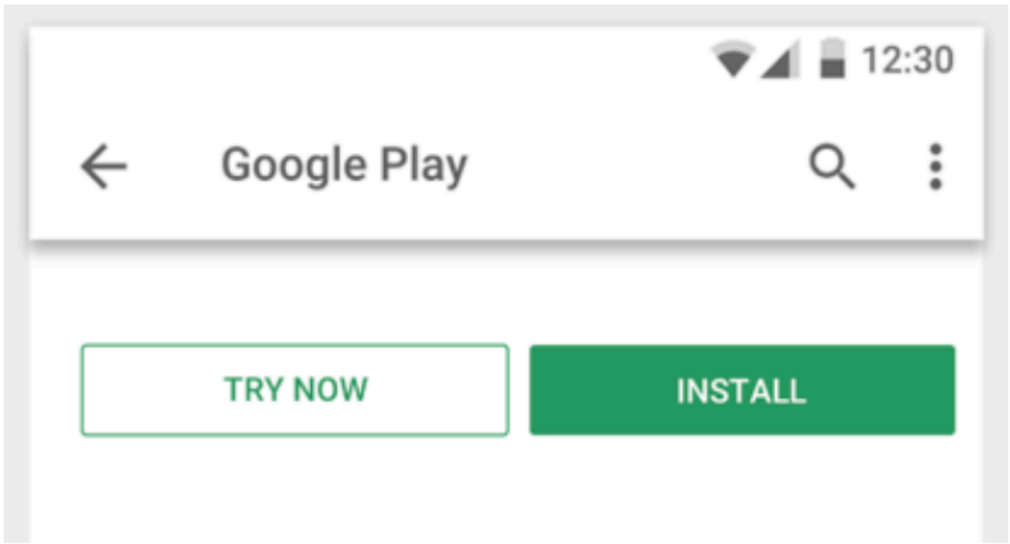


Instant Apps



Instant Apps

- Basic Instant Experience



Instant Apps

- Basic Instant Experience
- Enhanced Instant Experience

Search

Ads

Social

Beacons

Messages

Email

QR codes

NFC

App Size Limits

Category	Basic experience size limit	Enhanced experience size limit
App	10 MB	4 MB
Game	10 MB	10 MB

Benefits of reducing the size

Reduce the size of your app

Enable the instant experience: **Try Now** button and web banners

This functionality allows you to gather data and see business impact

Enable enhanced discovery surfaces: Search, ads, messages, and more

Continue to modularize and reduce your app's size

Leads to more installations, engagement, and business success

Best Practices

- Use the same codebase for both app types.
- Design for multiple feature modules.
- Don't focus on the feature module size limit at the beginning.

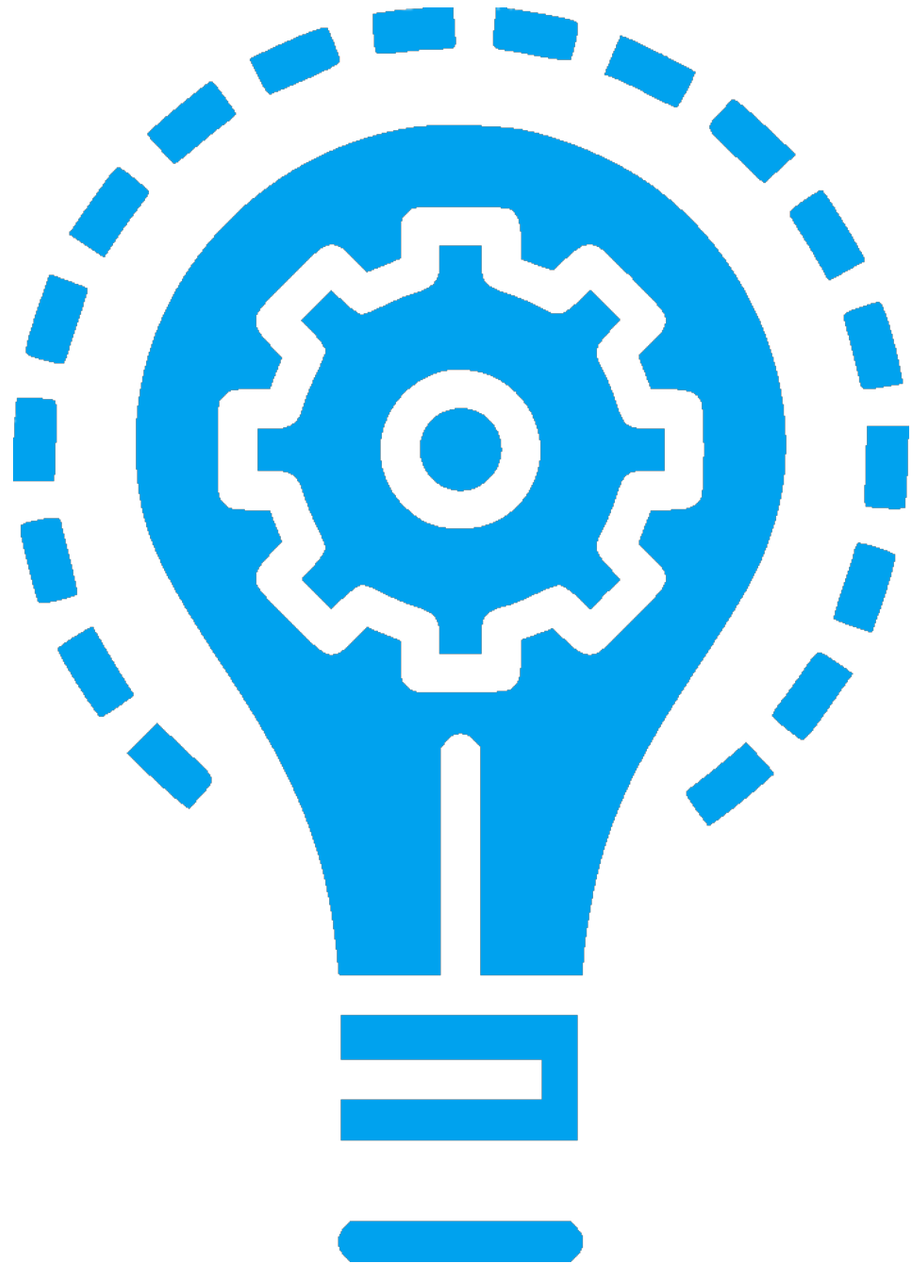


Image source: prosafe.org

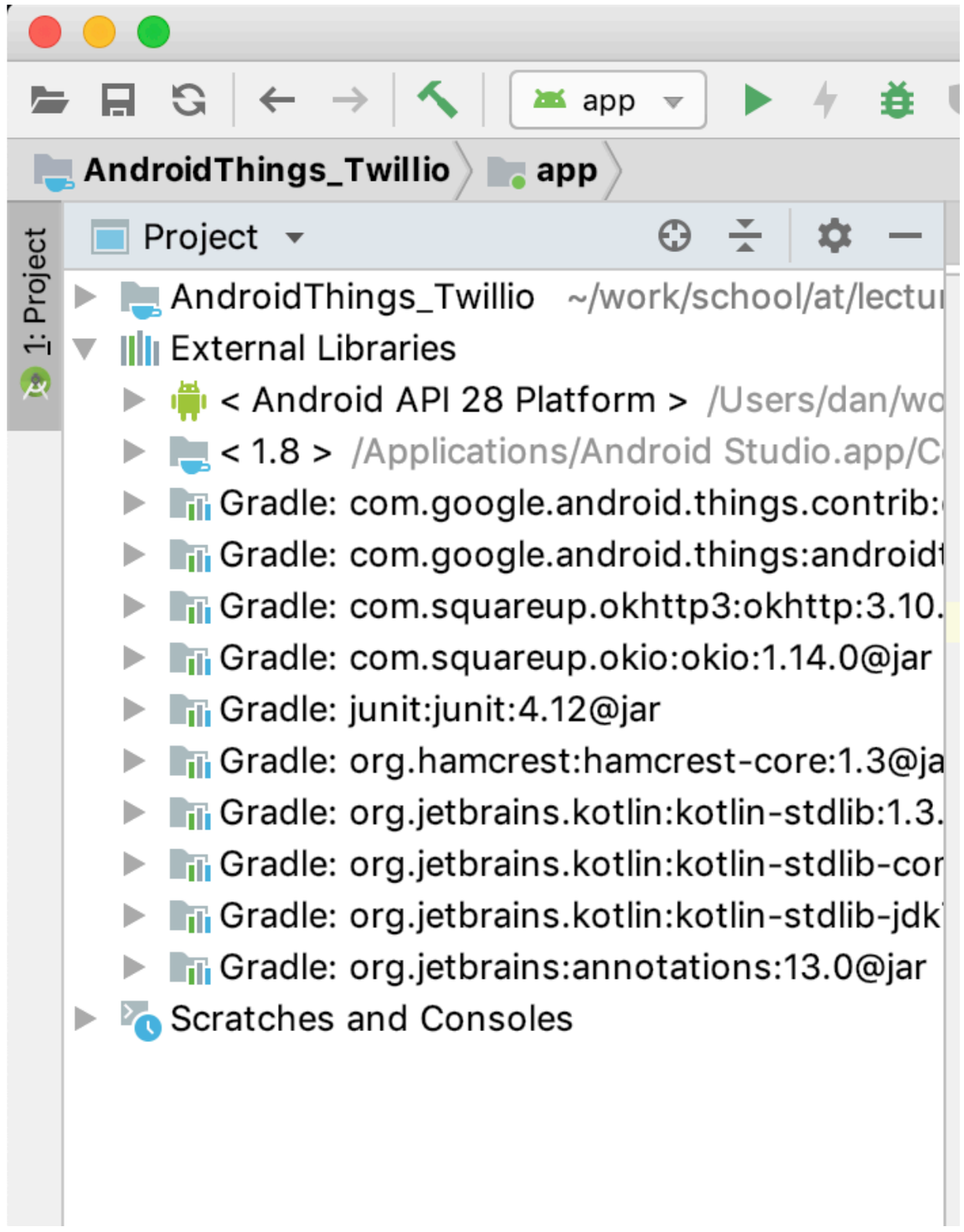
Update App Resources

- Reduce image file size.
- Remove extra languages.
- Remove extra files.

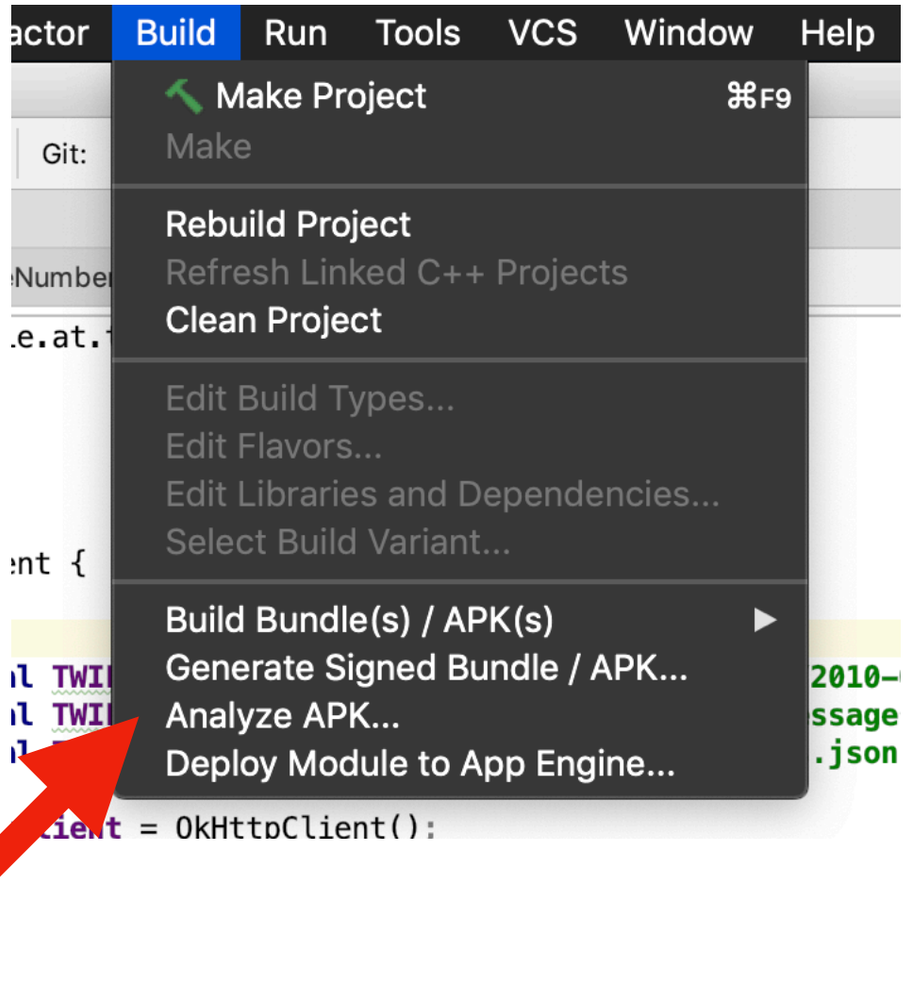
Image source: whatsnewlaporte.com

Remove Unused Libraries

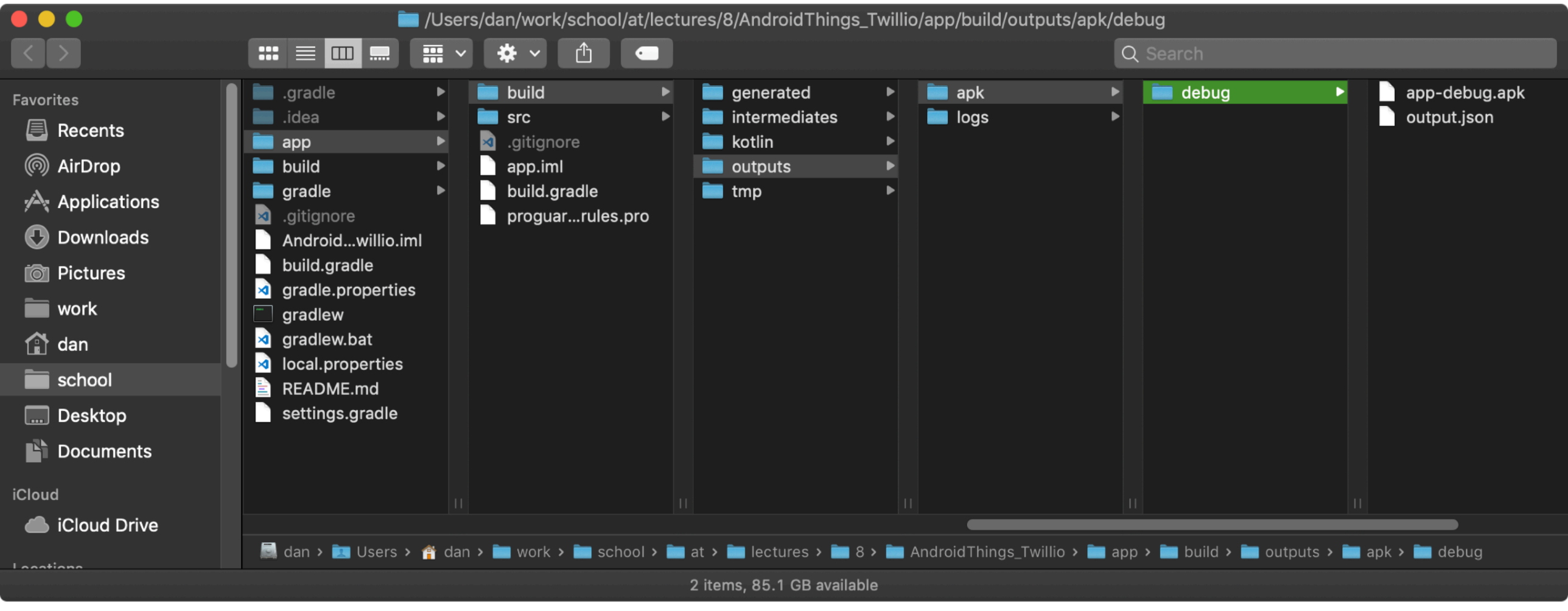
- Project view:
  - External libraries.



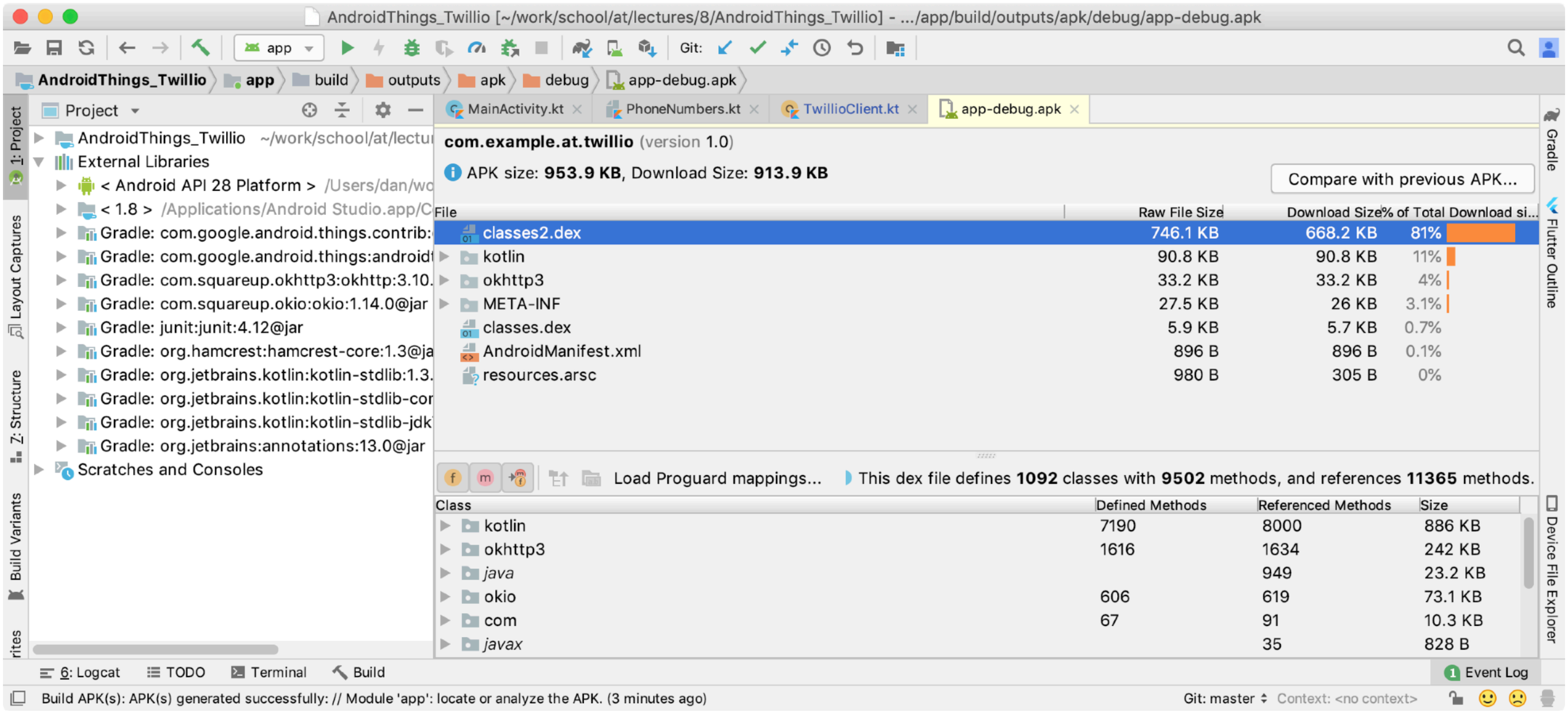
APK Analyzer



APK Analyzer



APK Analyzer

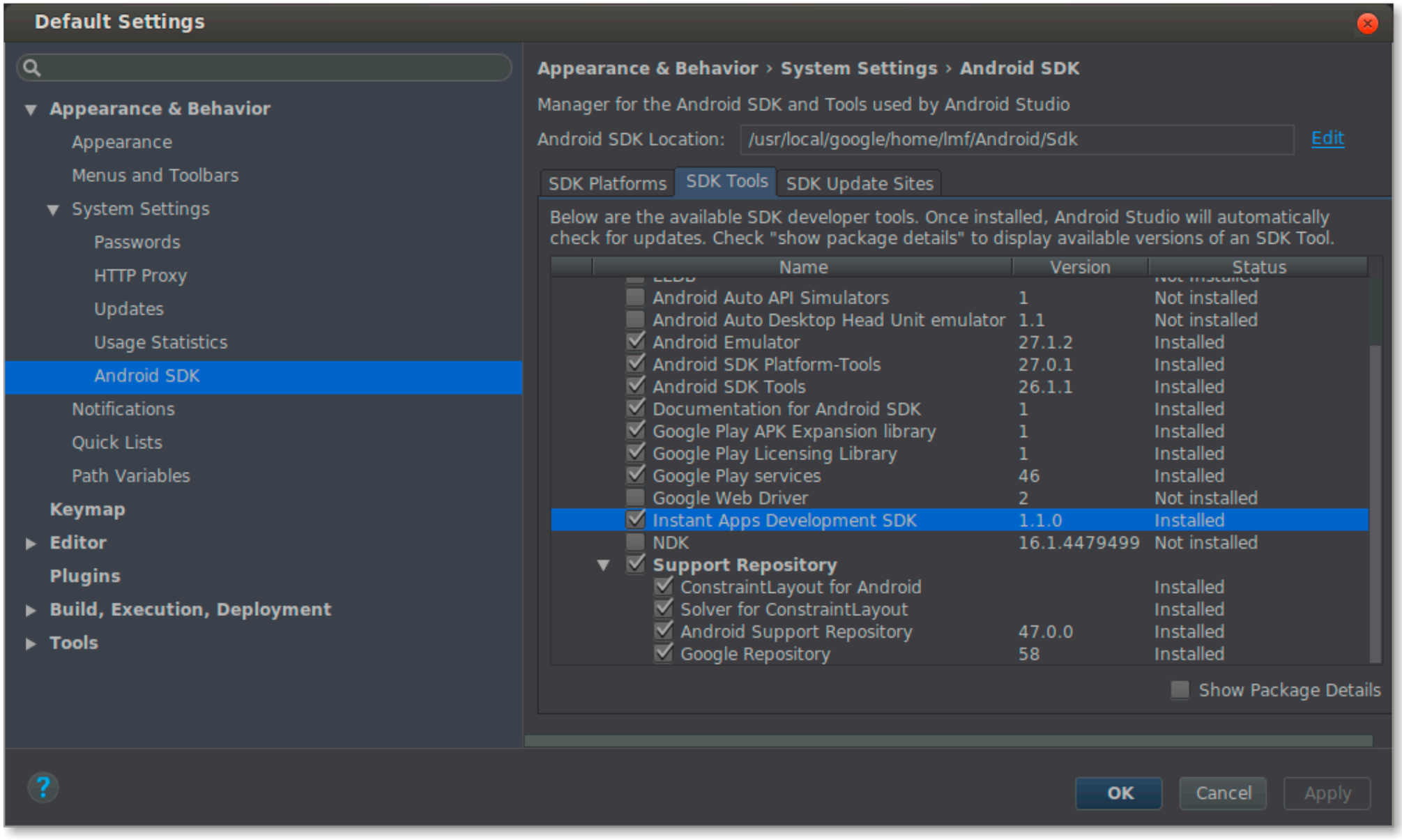


Exclude Library Parts

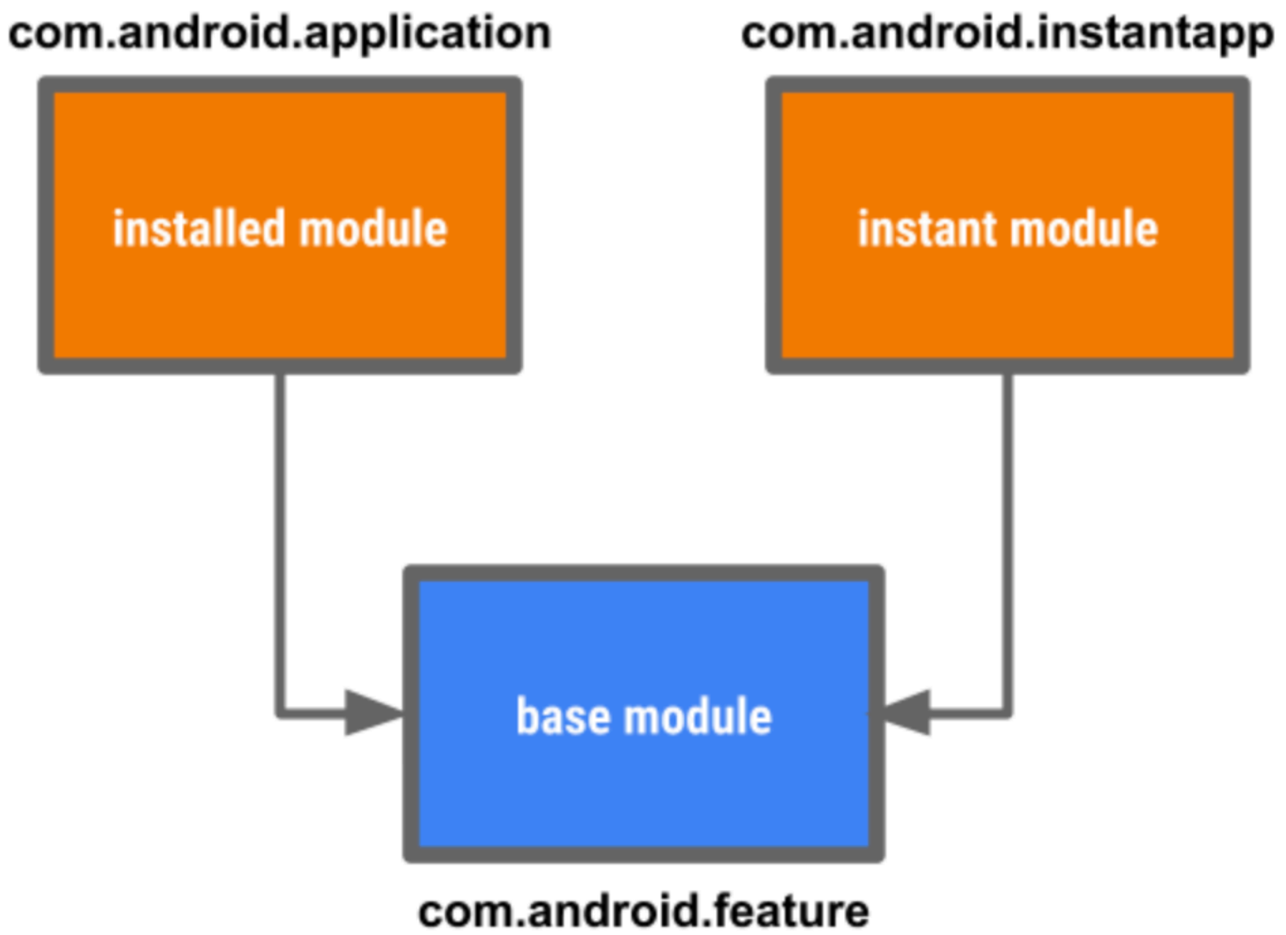
```
dependencies {
    implementation('some-important-but-large-library') {
        exclude group: 'com.example.imgtools', module: 'native'
    }
}
```

Build Your First Instant App

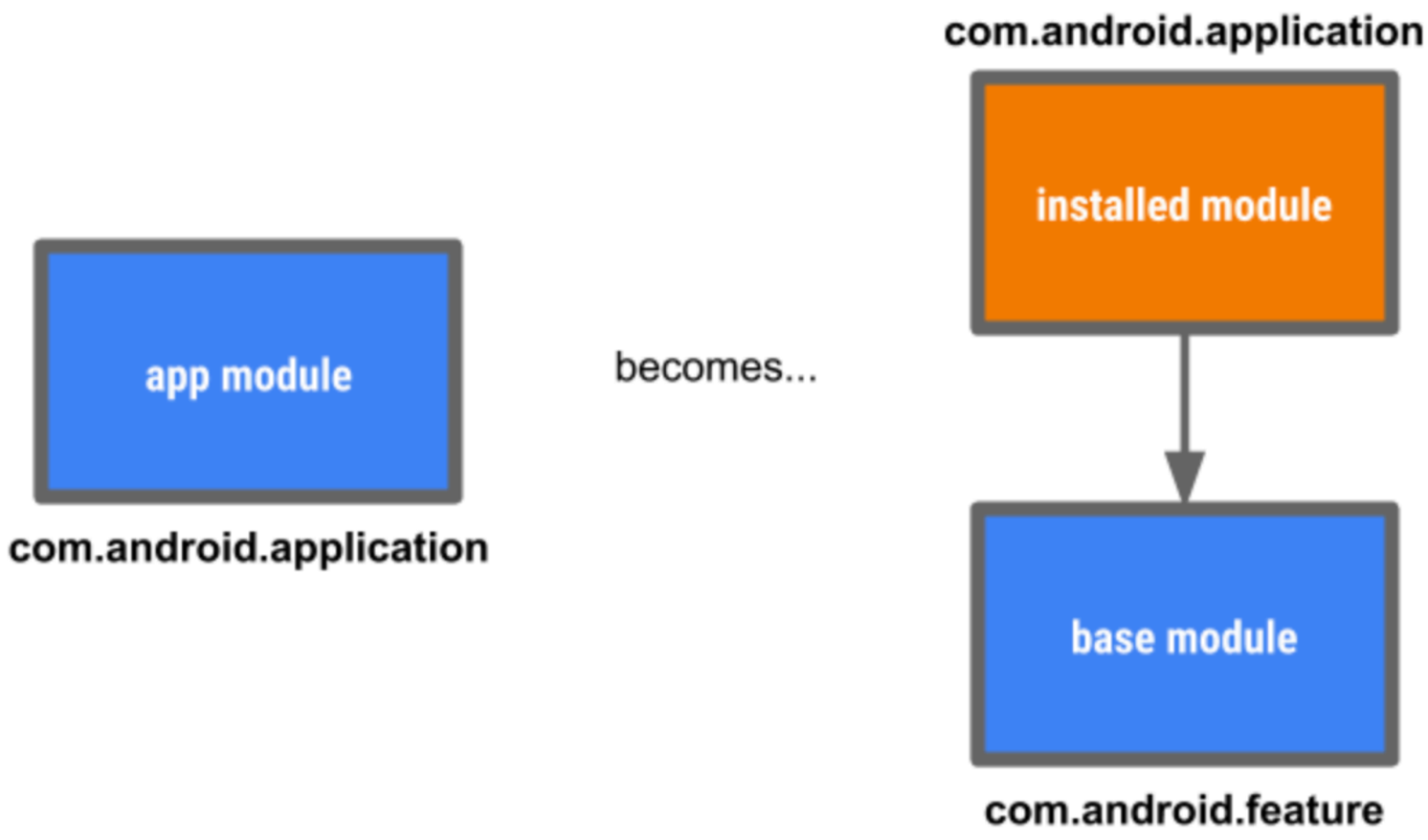
- Install Instant App SDK from: Tools -> SDK Manager



Google Play Instant



Convert Existing Module



Convert Existing Module

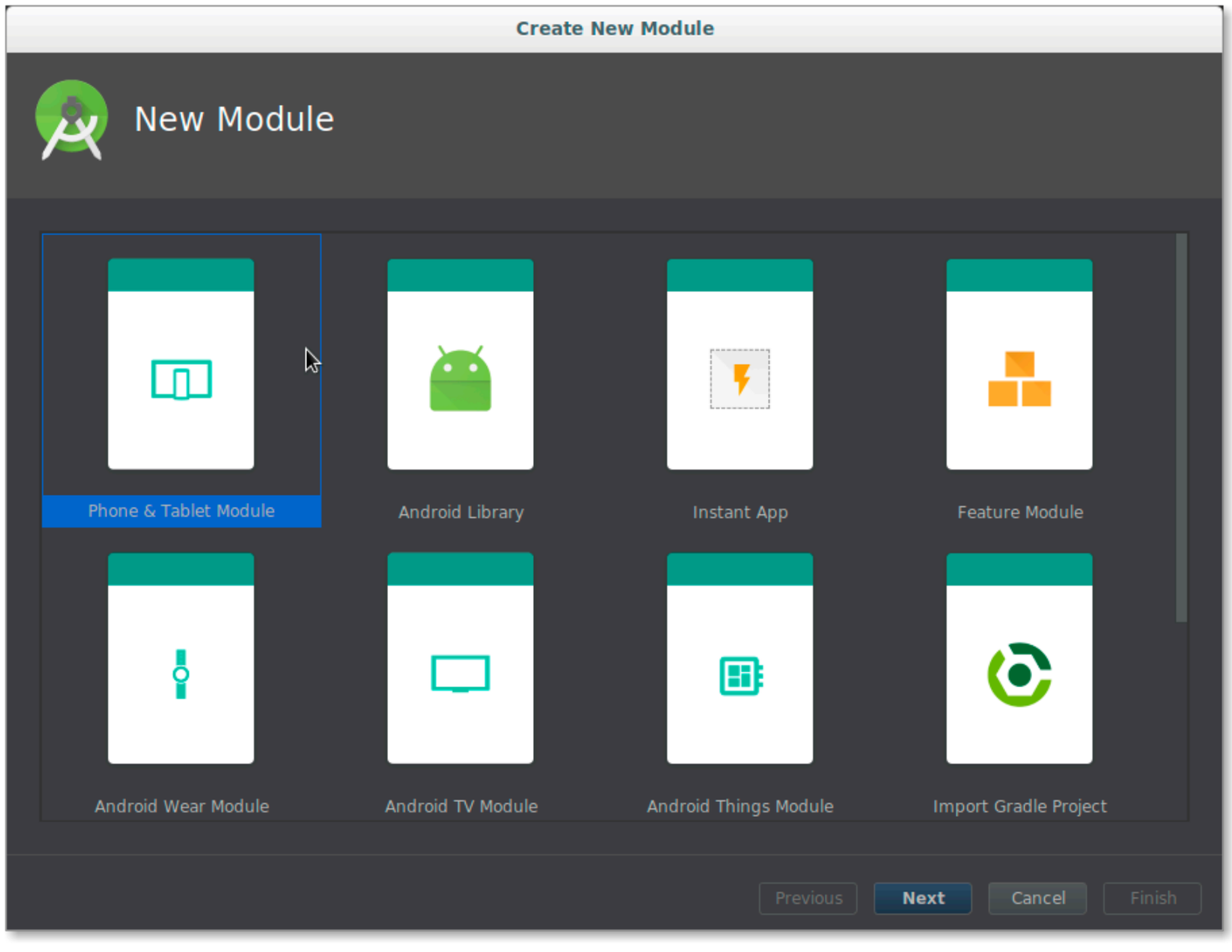
- Convert :app to a feature module.

```
// replace
// apply plugin: 'com.android.application'
// with
apply plugin: 'com.android.feature'
```

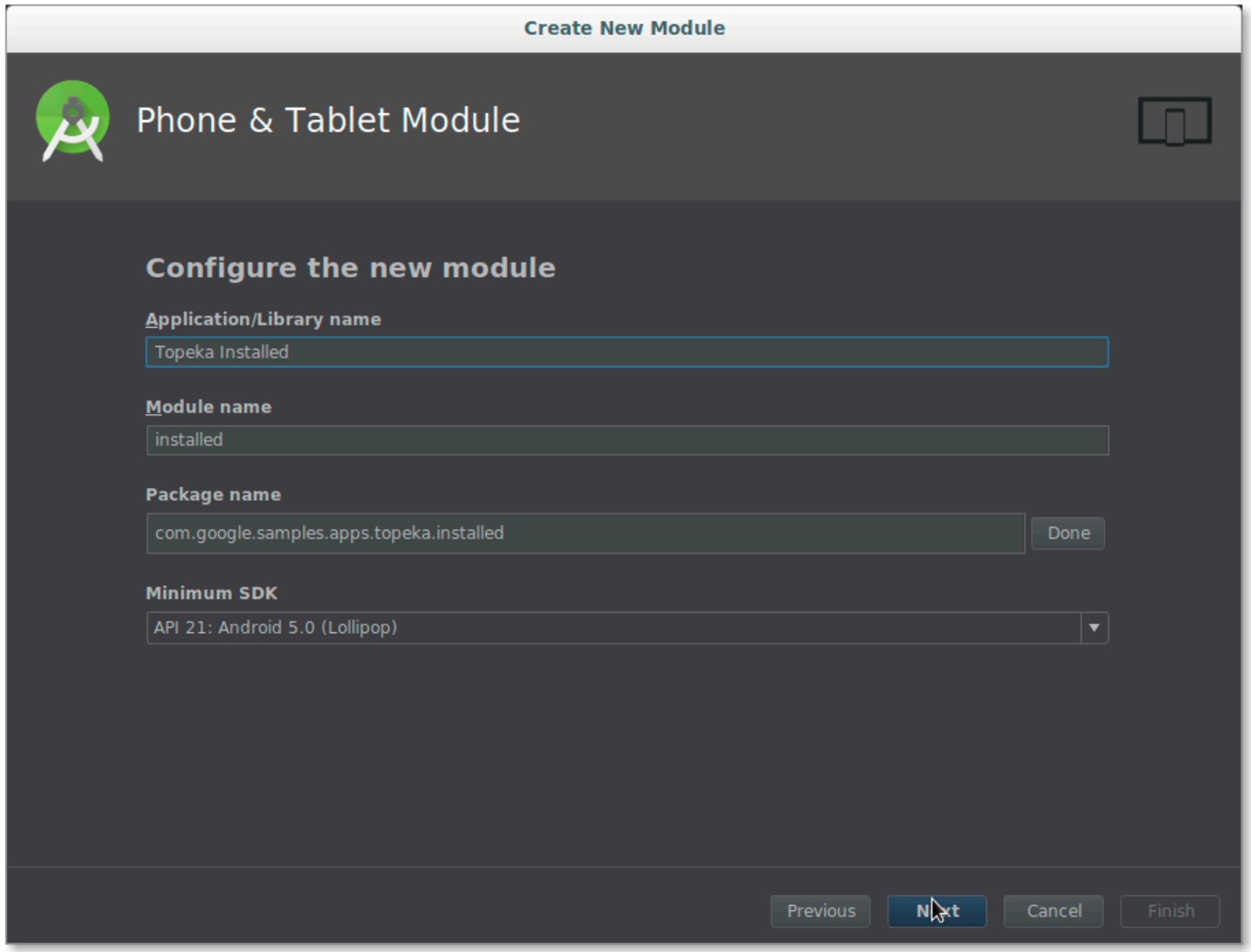
- Add baseFeature flag.

```
android {
    ...
    baseFeature true
    ...
}
```

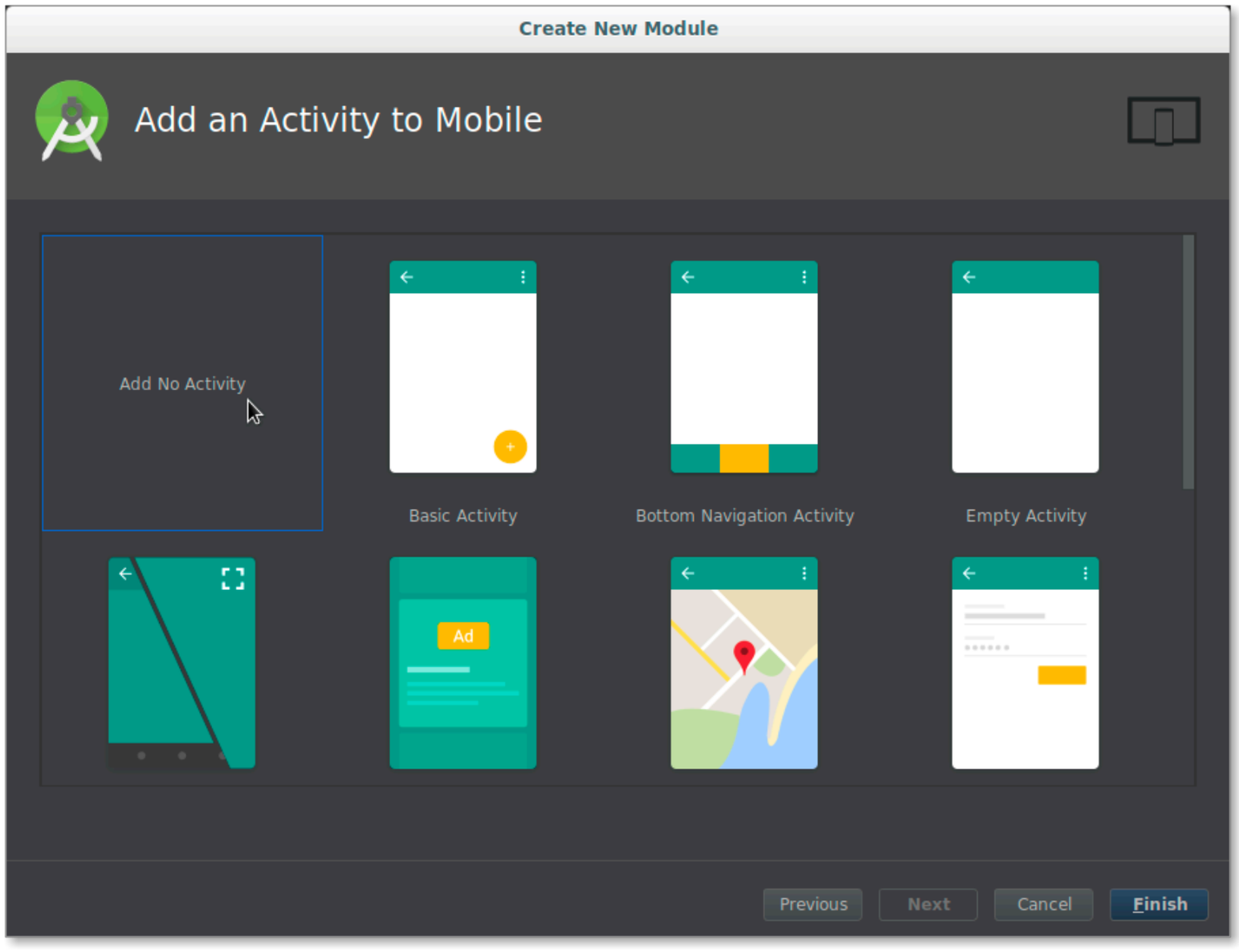
Create the new Installed Module



Create the new Installed Module



Create the new Installed Module



Convert Installed to an App Module

- Move **applicationId** from :app to :installed

```
android {
    ...
    defaultConfig {
        // replace
        // applicationId "com.google.samples.apps.installed"
        // with
        applicationId "com.google.samples.apps.topeka"
        ...

    }
    ...
}
```

- Associate :installed and :app

```
dependencies {
    implementation project(':base')
}
```

Update Manifest Files

- :installed should have a minimal manifest, like:

```
<manifest xmlns:android="http://schemas.android.com/apk/res/android"
          package="com.google.samples.apps.topeka">
</manifest>
```

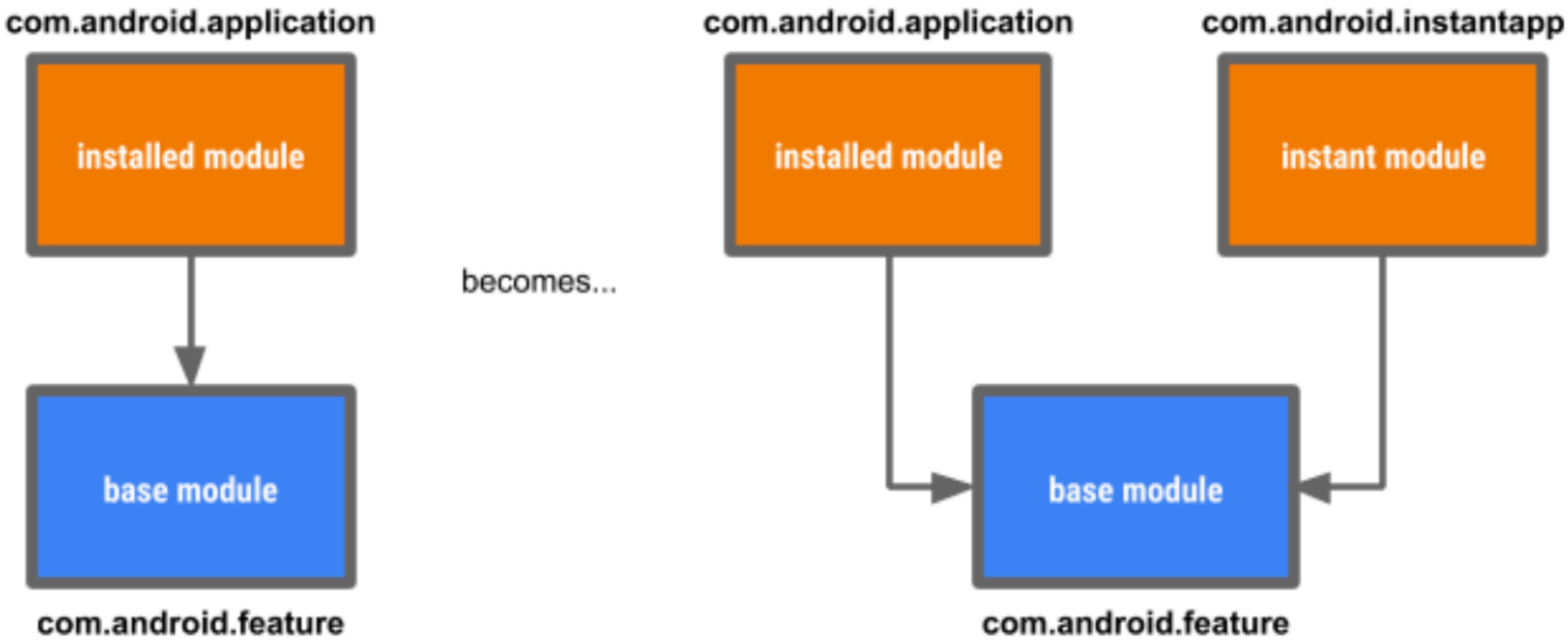
- :app should have a different package, like:

```
<manifest xmlns:android="http://schemas.android.com/apk/res/android"
          xmlns:tools="http://schemas.android.com/tools"
          package="com.google.samples.apps.topeka.base">
...
</manifest>
```

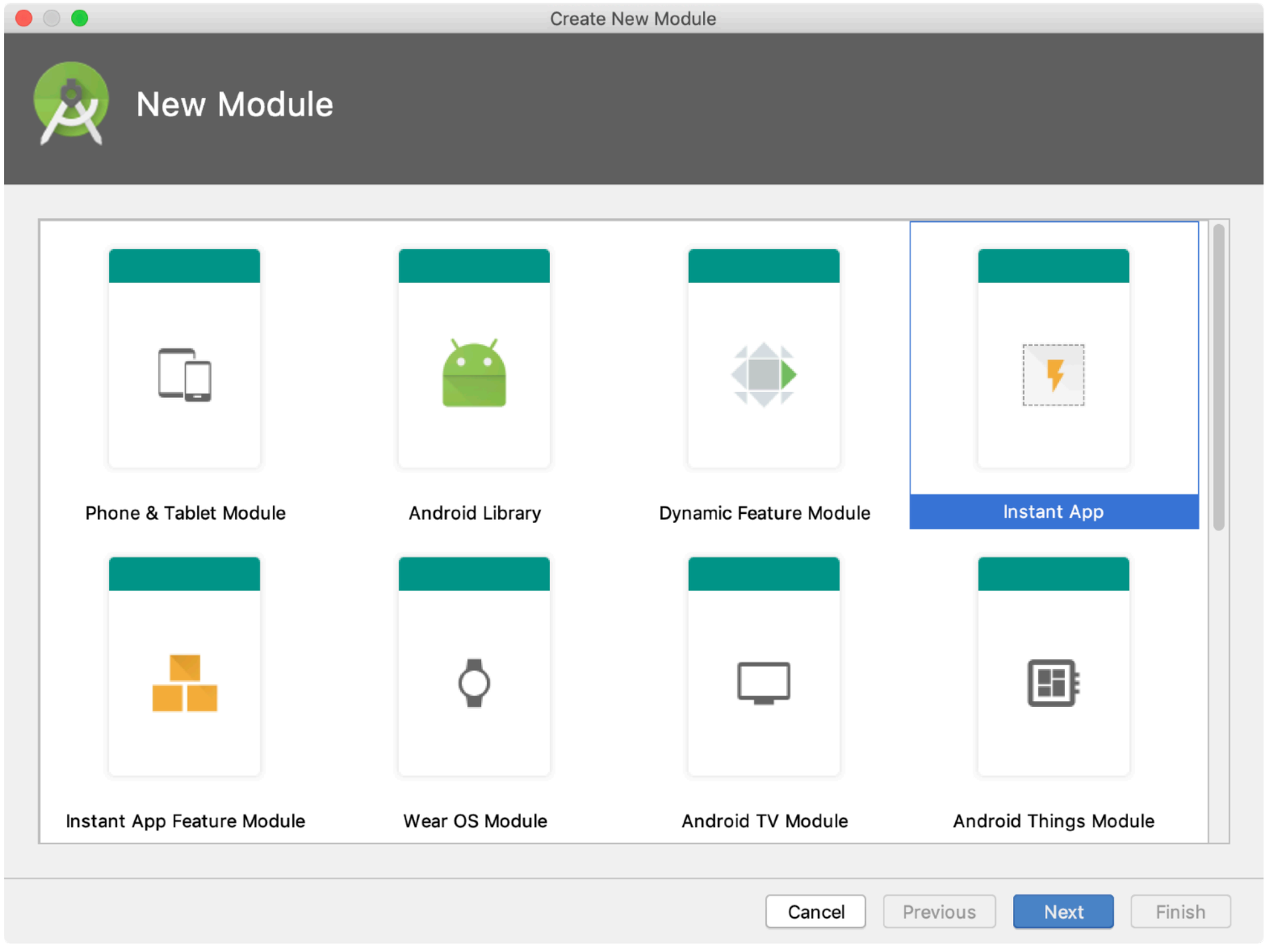
- Update the packages in :app to reflect the new package!

- Update the R imports.

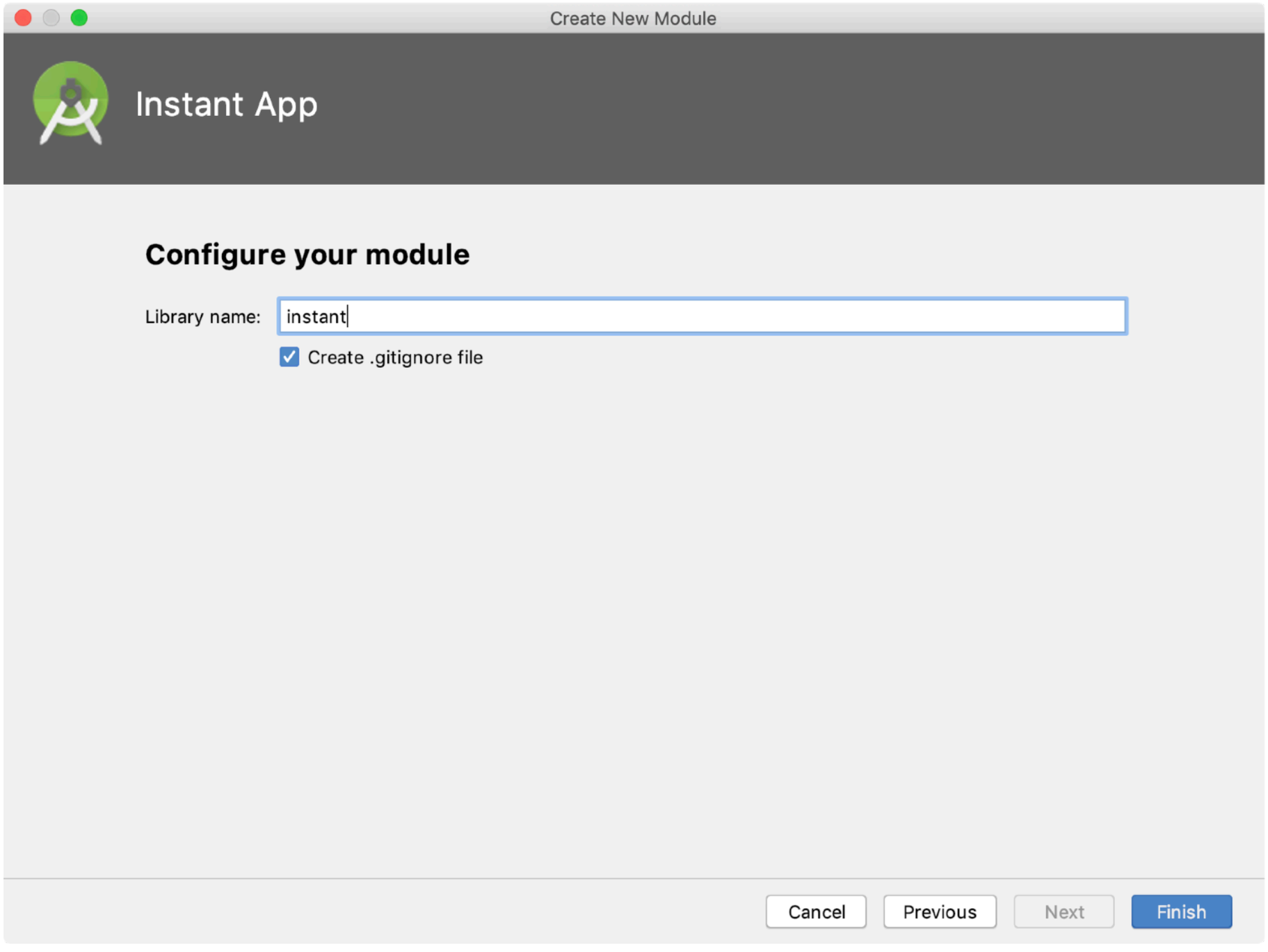
Create the Instant Module



Create the Module



Create the Module

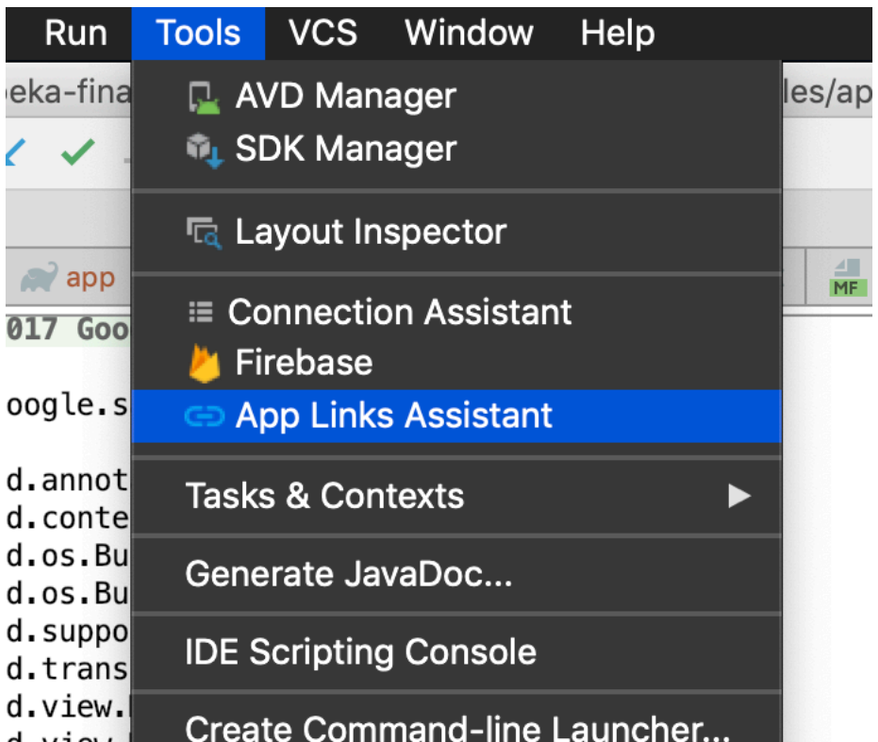


Configure the Module

Associate :instant with :app

```
dependencies {
    implementation project(':base')
}
```

Add an App Link with App Links Assistant



Create a Link

1 Android App Links Support

URL-to-Activity mappings

Use the URL Mapping table below to add, update or delete URL to Activity mappings. The URL Mapper will update your AndroidManifest.xml file to include the appropriate URL intent filters.

URL Mapping

Host	Path values	Activity	Order

Check URL Mapping

Enter a URL to chec

Add URL Mapping

Basic URL Mapping

You can add or edit your URL mapping here

Host

http://www.cs.ubbcluj.ro

Path

pathPattern | Enter a path e.g. /myPath | How it works

Activity

com.google.samples.apps.topeka.activity.SignInActivity (app)

Cancel | Show Advanced | OK

App Links Assistant

Android App Links enable your users to launch directly into your app when they click on URLs that your app supports and they can also make your app content searchable.

The App Links Assistant will walk you through how to implement Android App Links below.

1 Add URL intent filters

Use the URL Mapping editor to easily add URL intent filters to your Activities.

Open URL Mapping Editor

2 Add logic to handle the intent

When the system starts the activity through the intent filter, you can use the data provided by the intent to determine your app's response.

Select each URL-mapped activity and insert the template codes. You can then add your own logic to handle the intent as appropriate.

Select Activity

3 Associate website

Associate your app with your website through a Digital Asset Links file.

Open Digital Asset Links File Generator

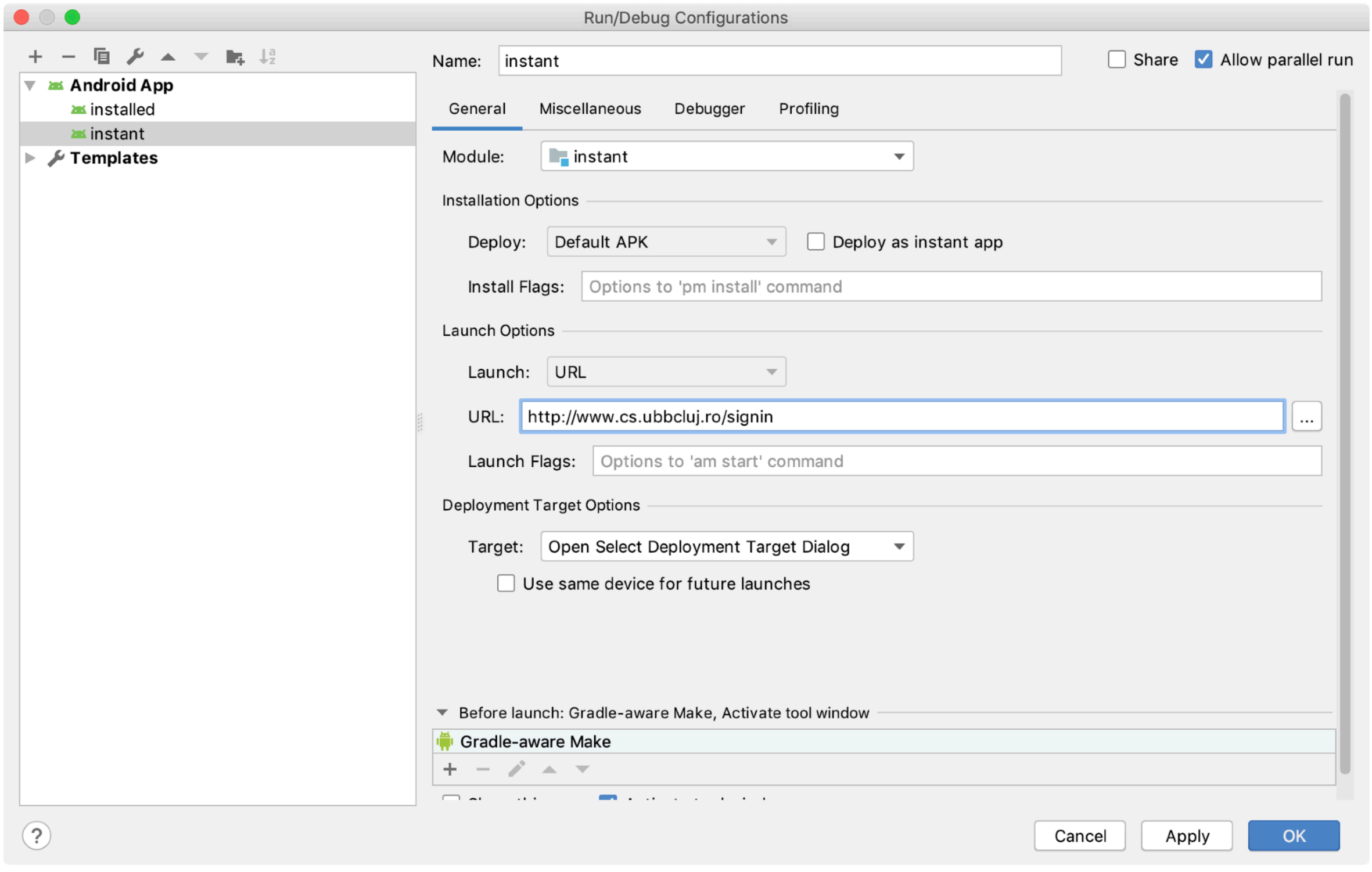
4 Test on device or emulator

Test your implementation of Android App Links by simulating launching a URL on a device or emulator.

Test App Links

DEMO

Setup & Run



Lecture outcomes

- Refactor an existing app in smaller modules.

- Create instant apps.