twitch.tv/dancojocar
youtube.com/dancojocar

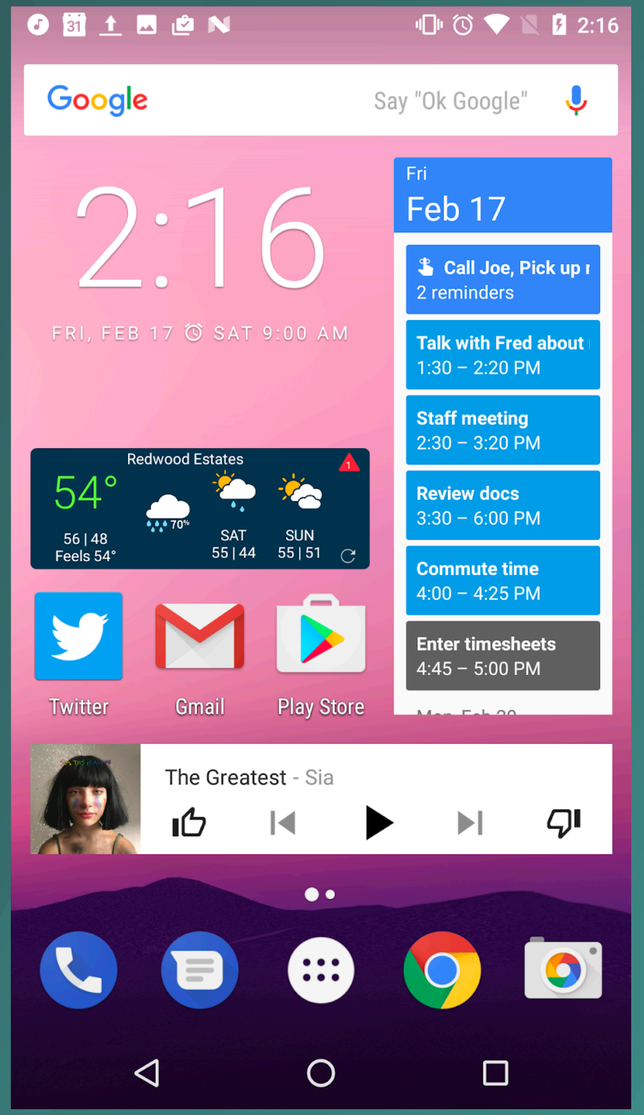
Lecture #10
Advanced Mobile Development

Mobile Applications
Fall 2023


La mulți ani!

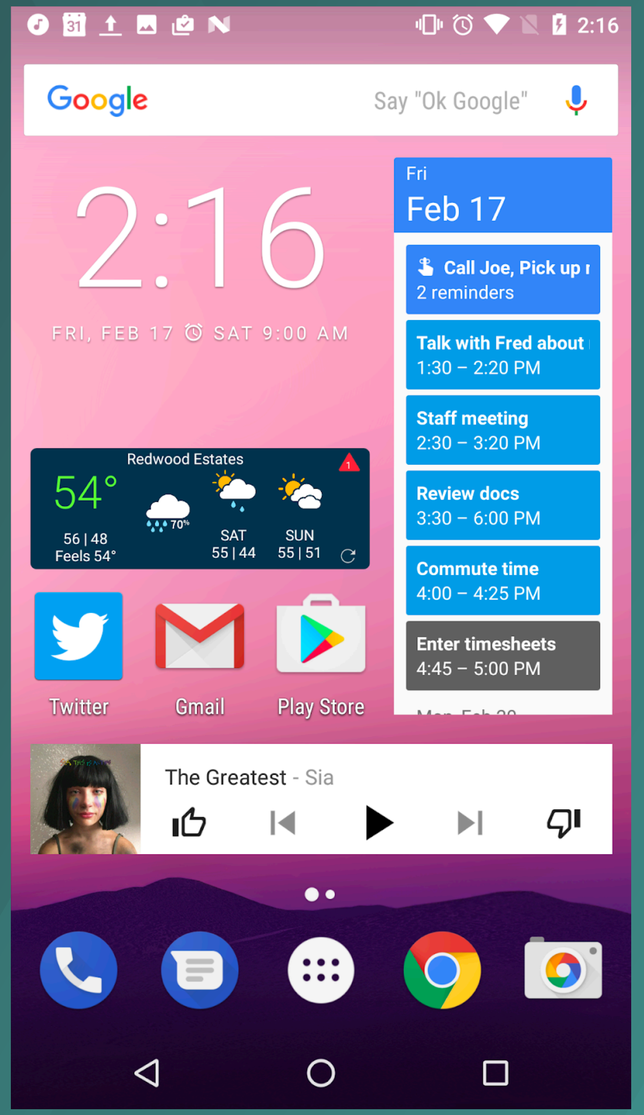
App Widgets

- A miniature app view.
- Runs on the home screen.
- Updated periodically.
- Display small amounts of information.
- Perform simple functions.



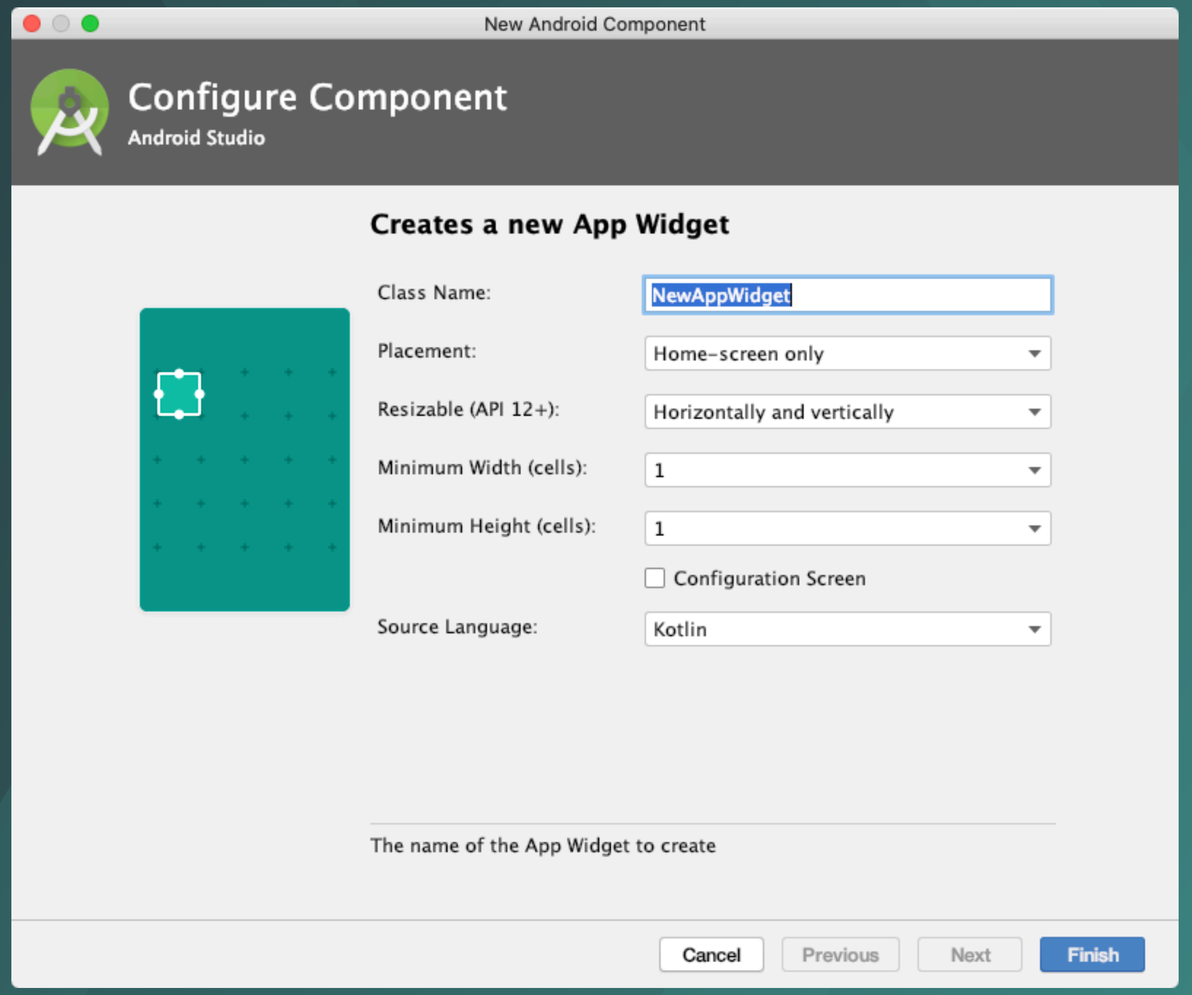
App Widgets

- Add-ons for an existing app.
- An app can have multiple widgets.
- Not available without an app.
- The default action is to start the app.

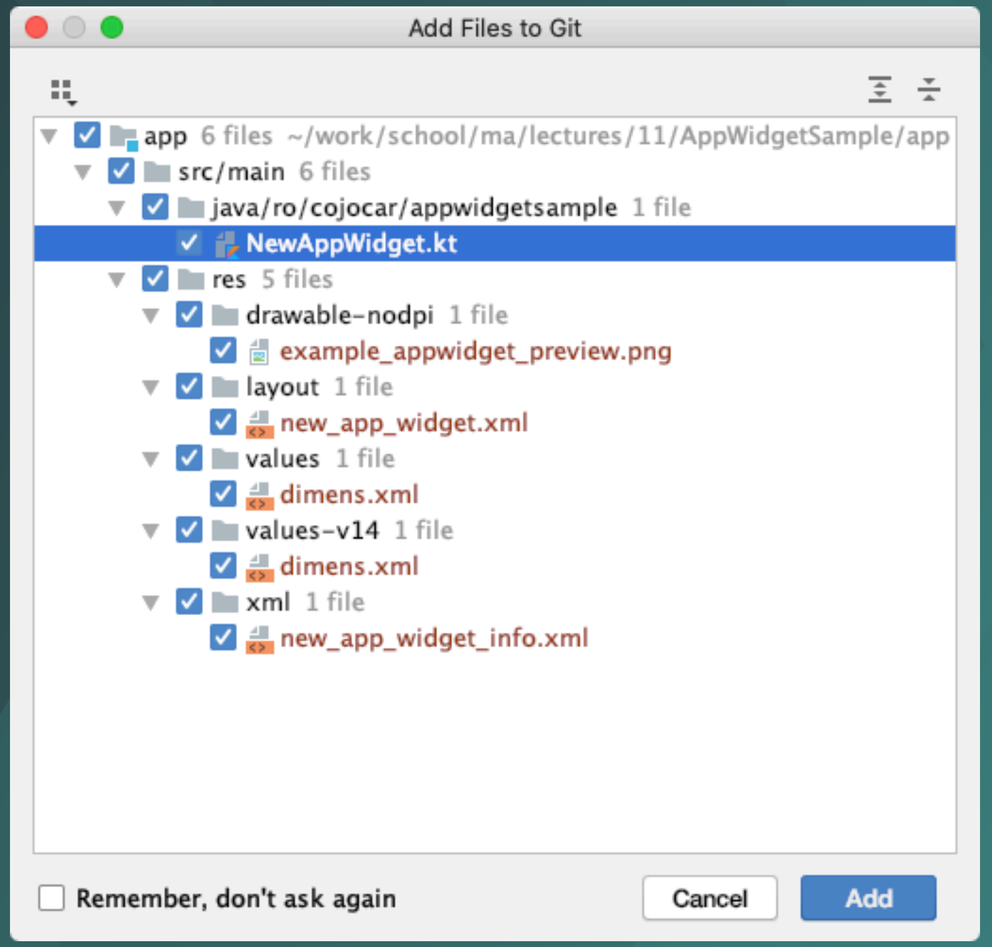


Set up the app widget project

- File > New > Widget > AppWidget.

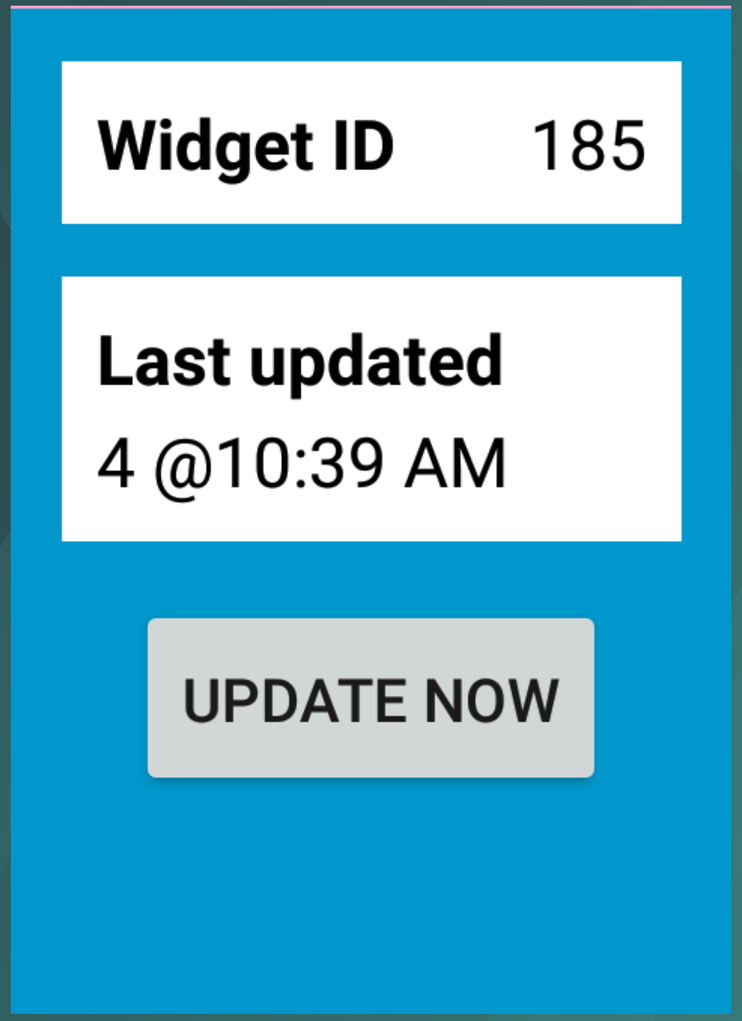


Generated Files



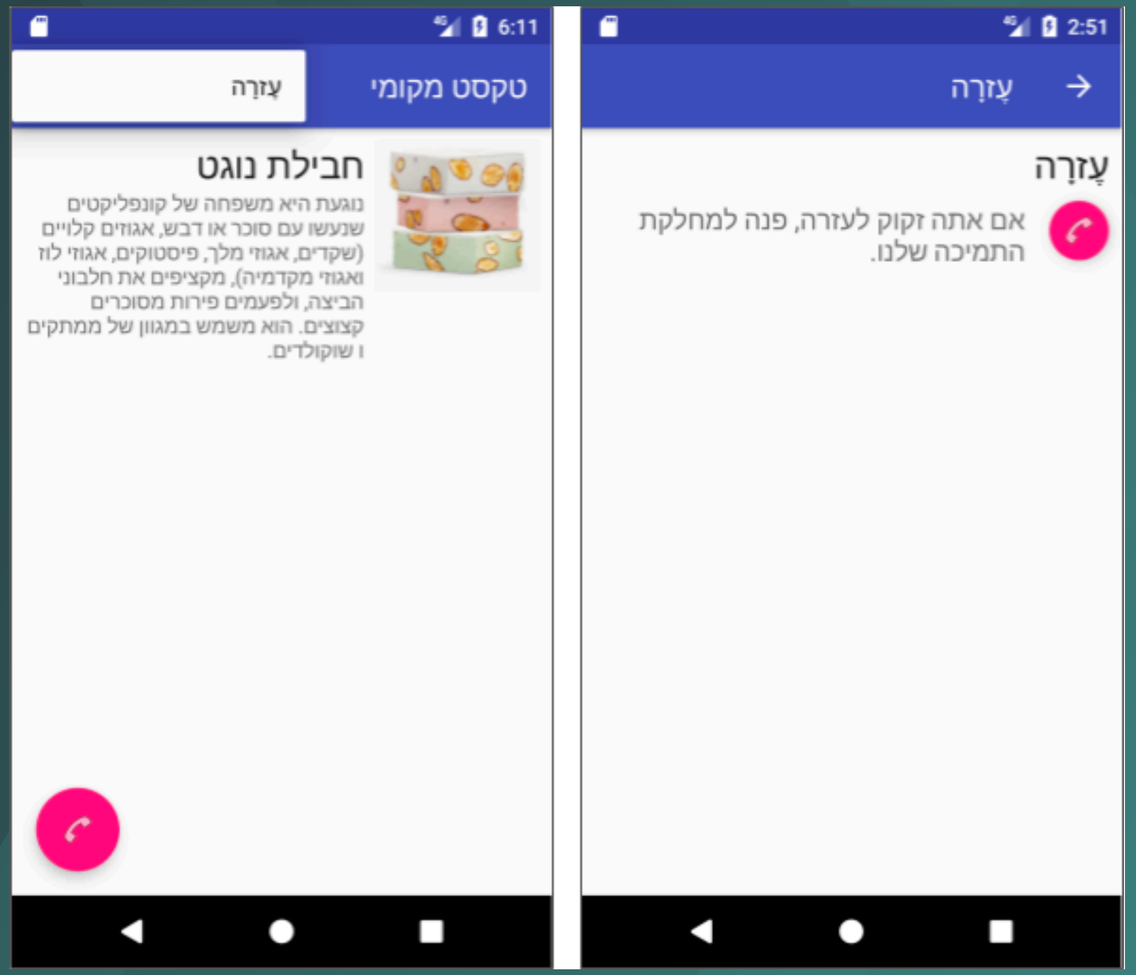
Customize the widget

DEMO

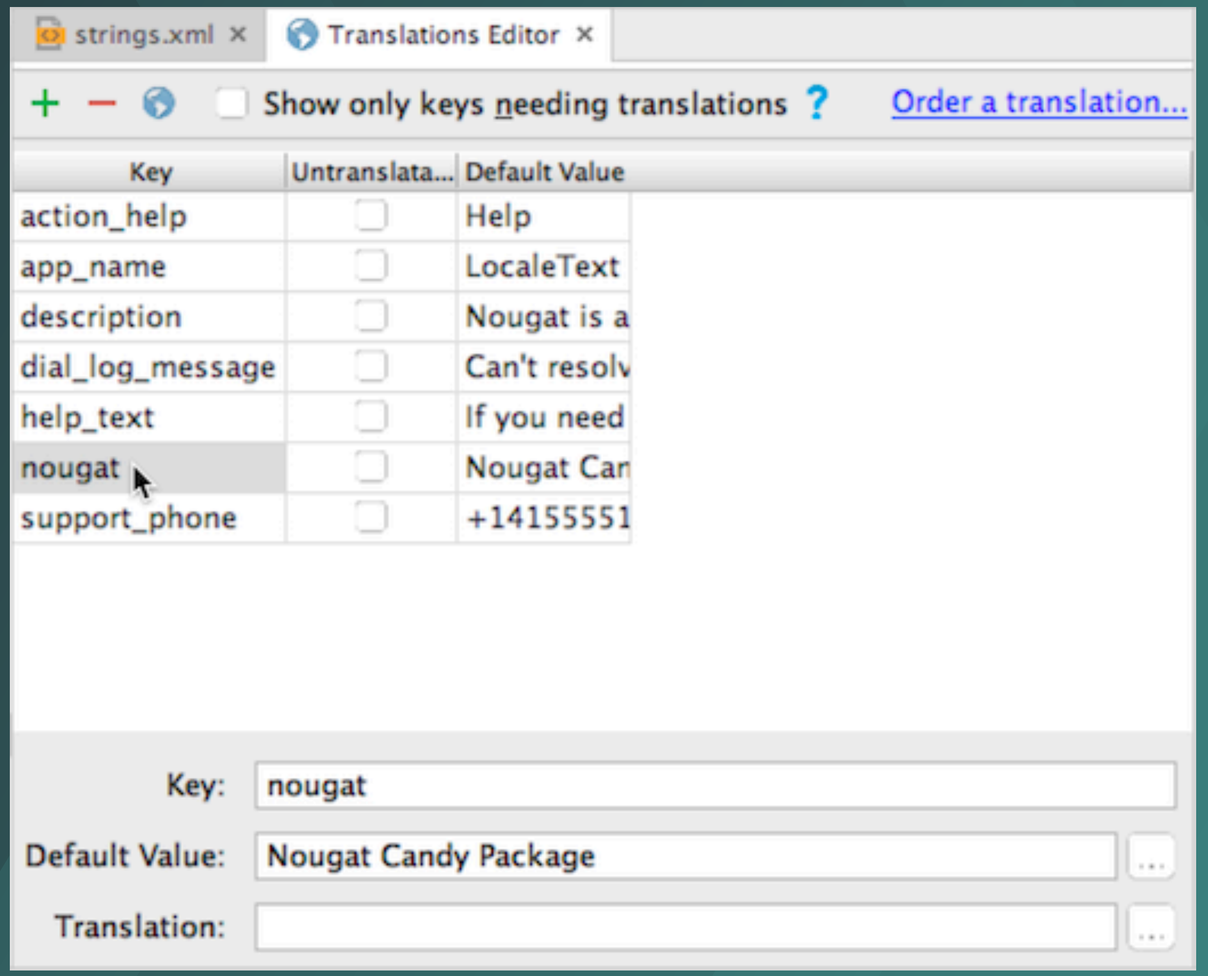


developer.android.com/guide/topics/appwidgets/index.html

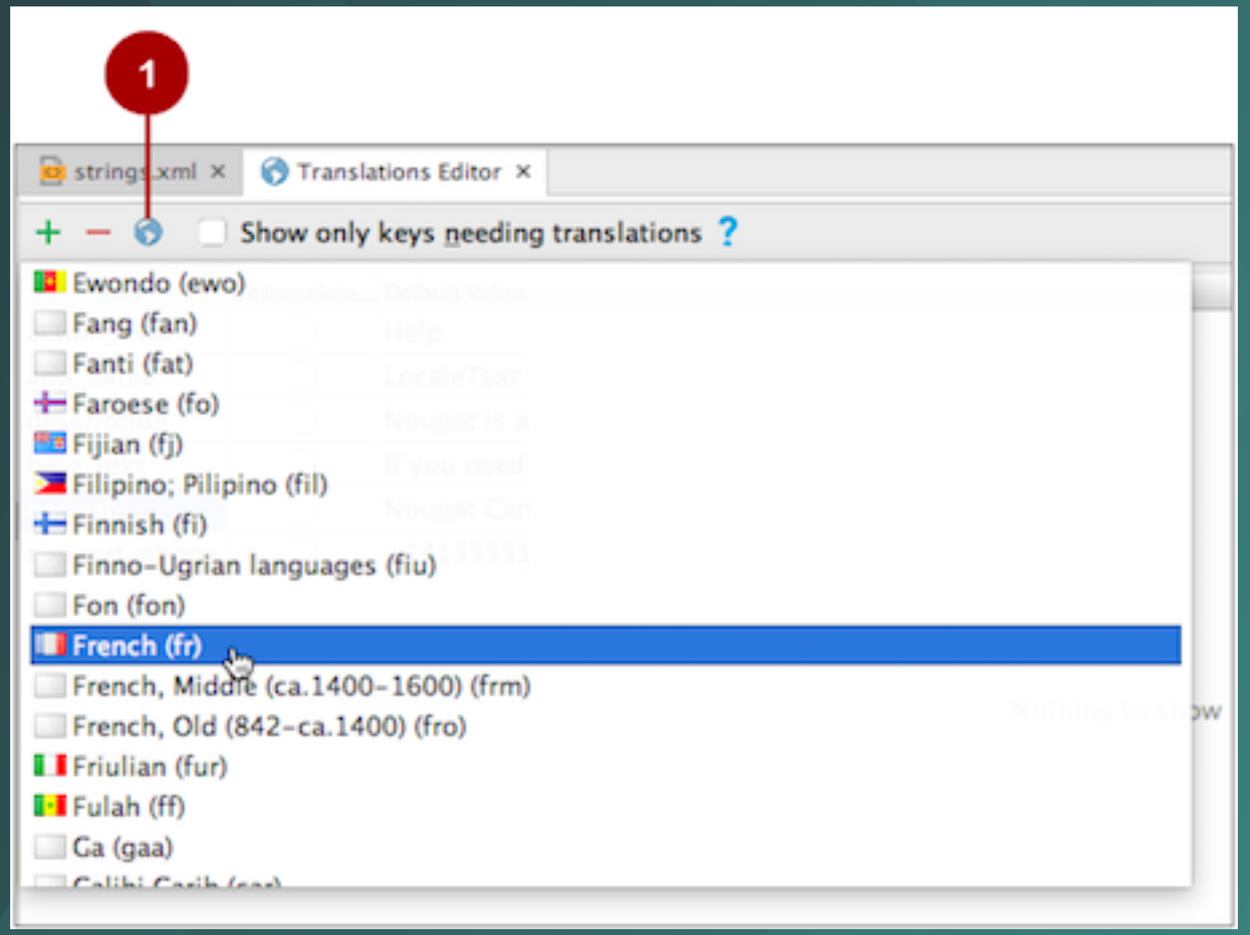
Language Support



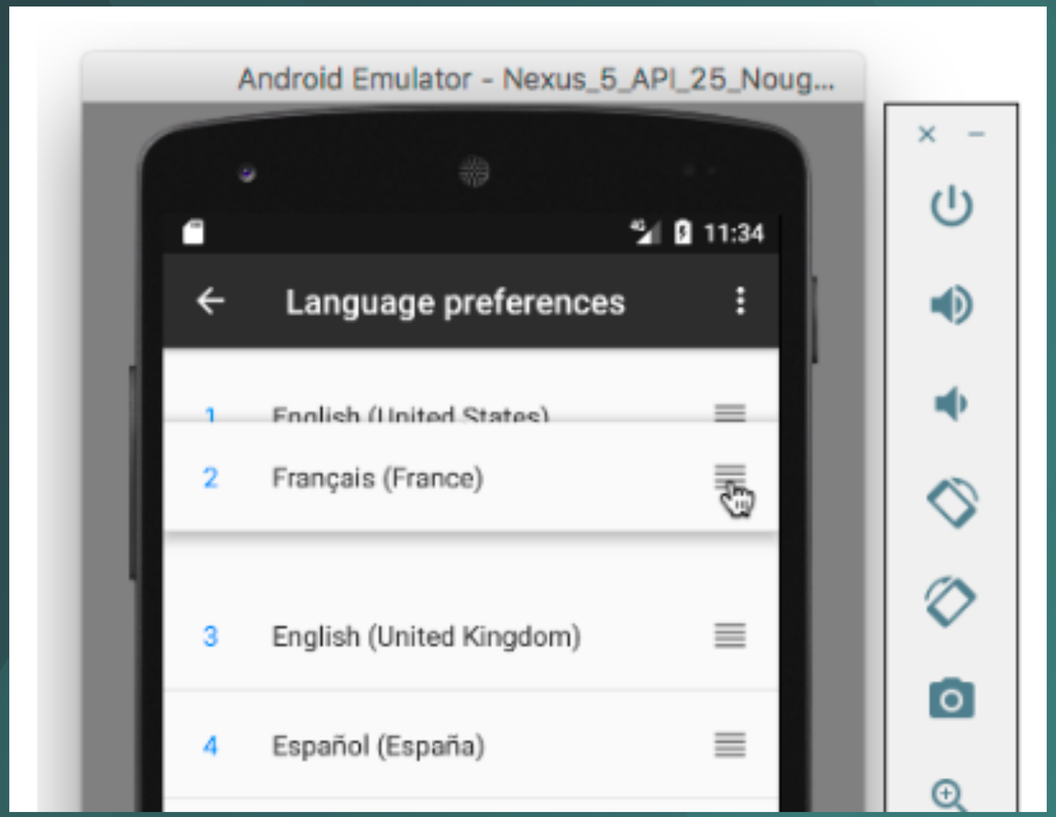
Add another language resource to the app



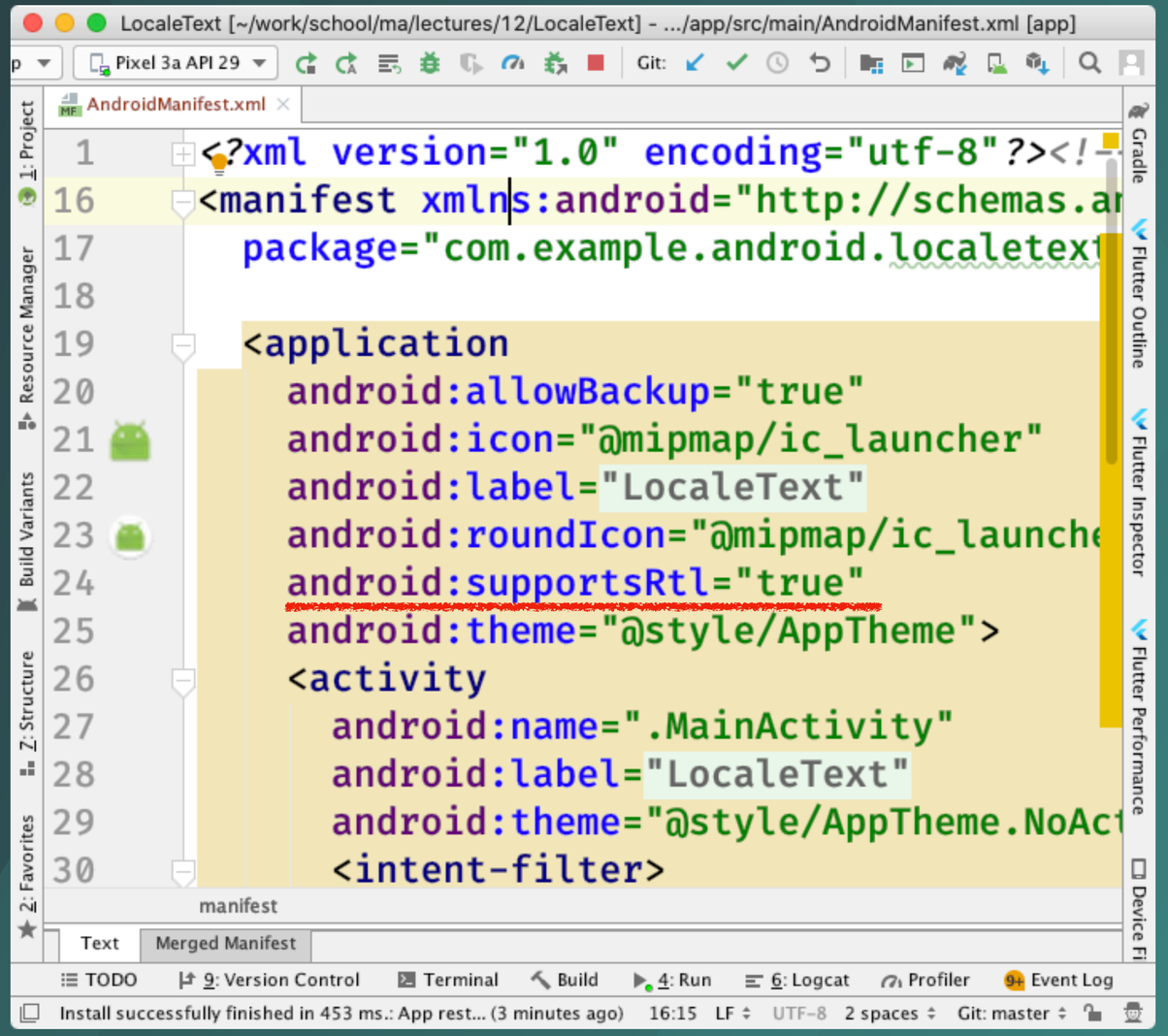
Add another language resource to the app



Run the app and switch languages

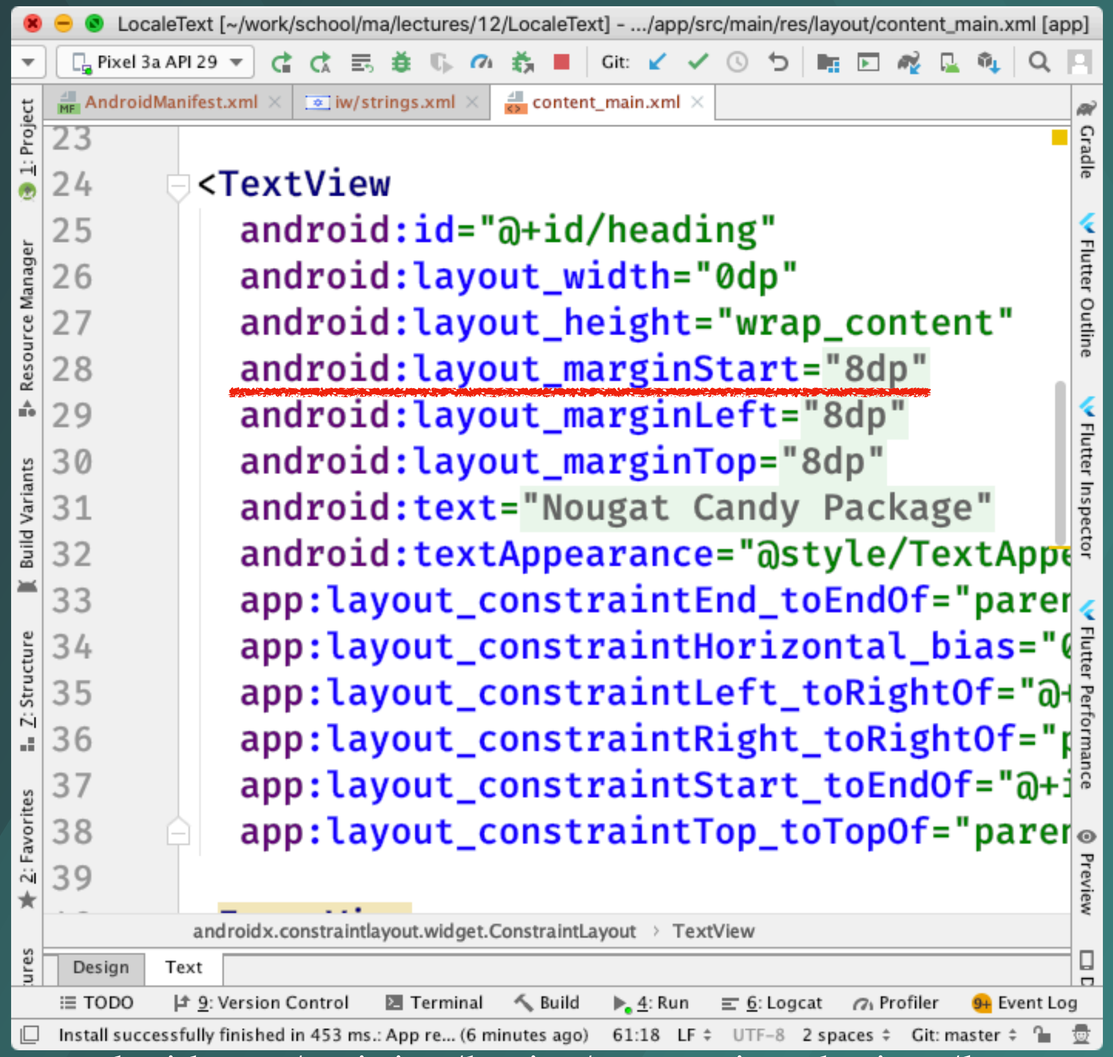


Add a right-to-left (RTL) language



Add a right-to-left (RTL) language

DEMO



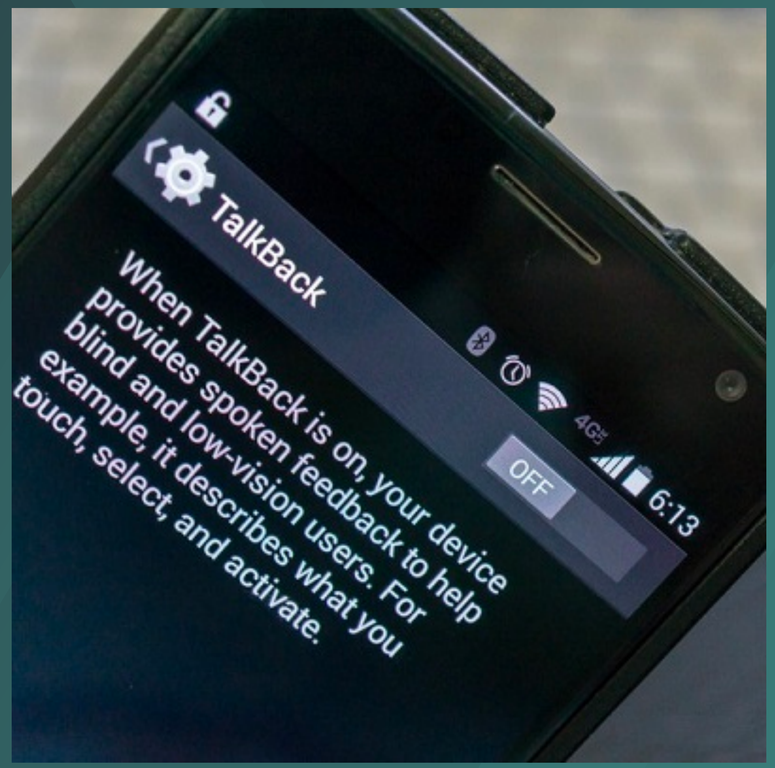
developer.android.com/training/basics/supporting-devices/languages.html

Accessibility

- Blindness
- Low vision.
- Color blindness.
- Deafness or hearing loss.
- Restricted motor skills.

TalkBack

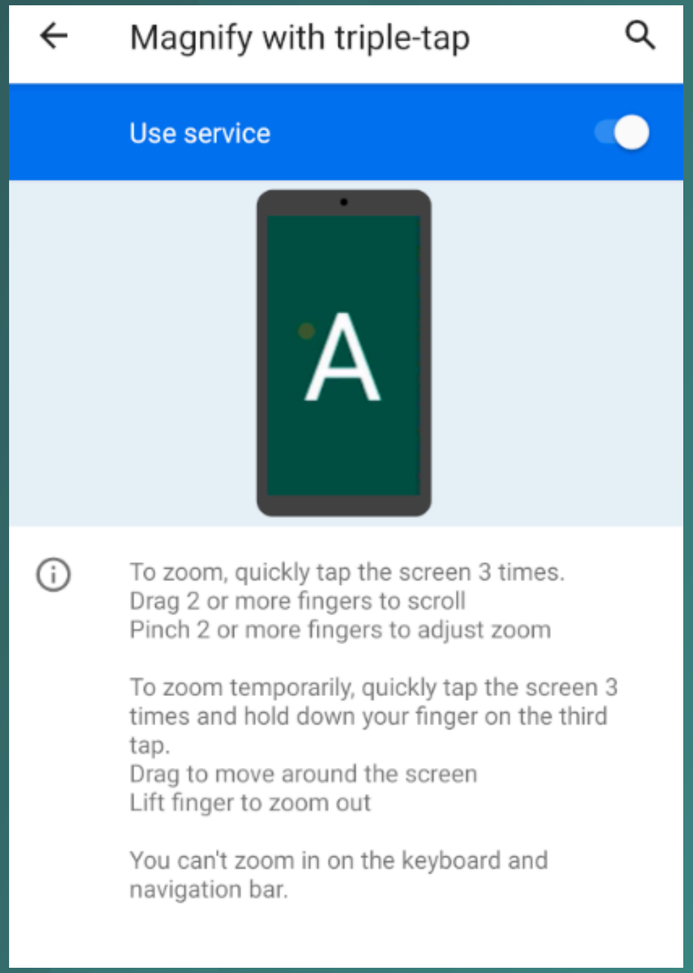
- Settings > Accessibility > TalkBack
- Settings > Accessibility > TalkBack > Settings > Launch TalkBack tutorial.



Font and Color

DEMO

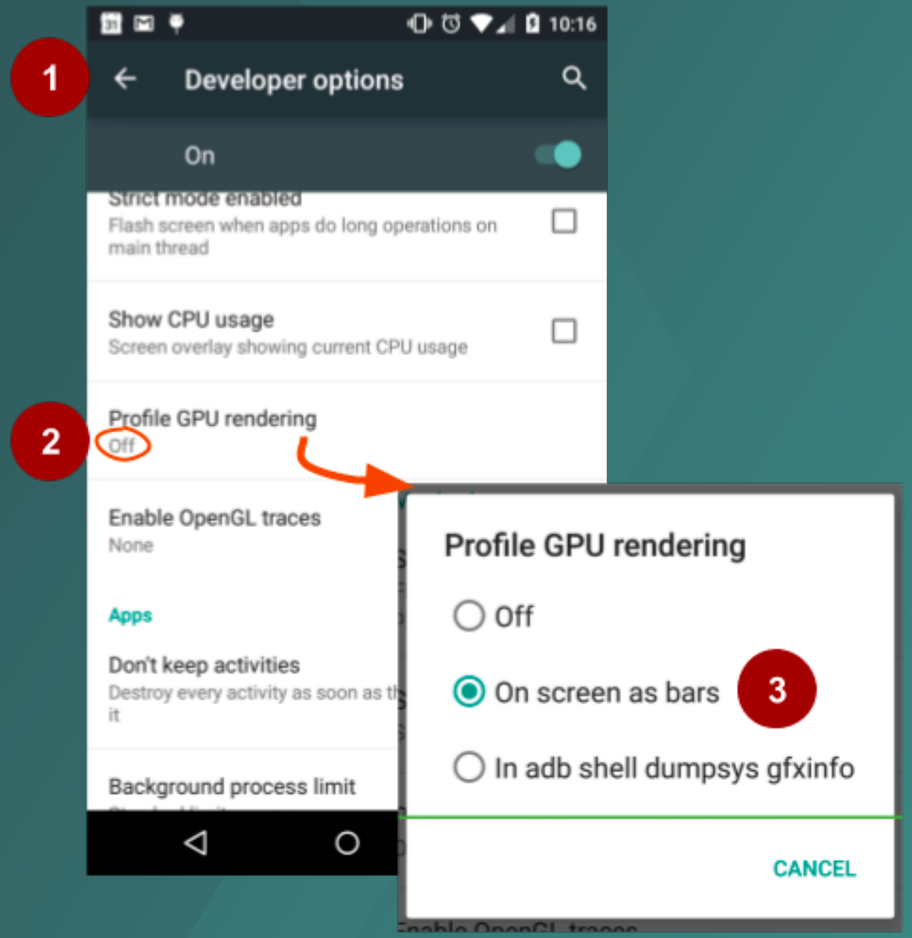
- Settings > Accessibility > Magnification gesture.
- Settings > Accessibility > Font size.
- Settings > Accessibility > High contrast text.



developer.android.com/guide/topics/ui/accessibility/index.html

GPU Profiling

- Settings
  - Developer options
    - Monitoring
      - Profile GPU rendering



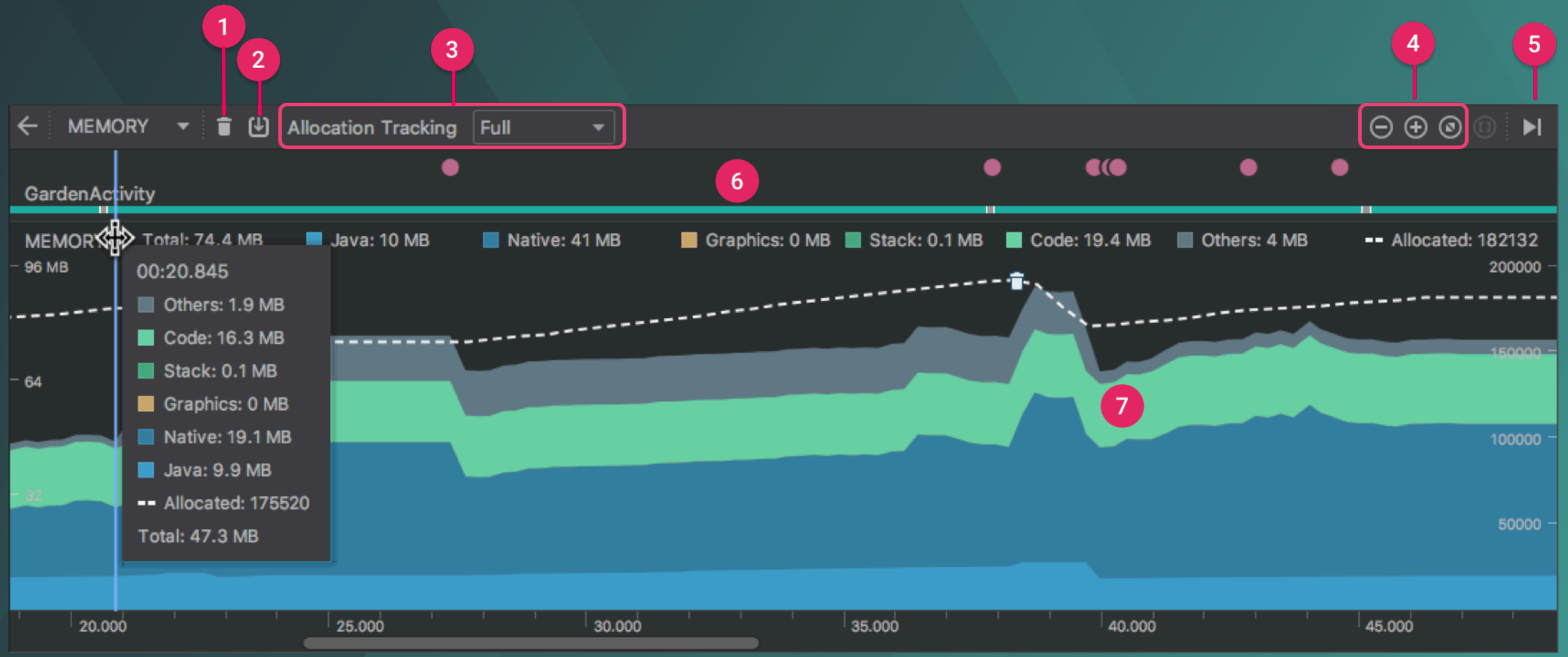
GPU Profiling

DEMO

developer.android.com/topic/performance/rendering/inspect-gpu-rendering#profile_rendering

Memory Profiling

DEMO



developer.android.com/studio/profile/memory-profiler.html#overview

ViewBinding

```
<?xml version="1.0" encoding="utf-8"?>
<RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent">

    <TextView
        android:id="@+id/mainTitle"
        tools:text="Main Title" />

    <TextView
        android:id="@+id/subTitle"
        tools:text="Main Subtitle" />
</RelativeLayout>
```

ViewBinding

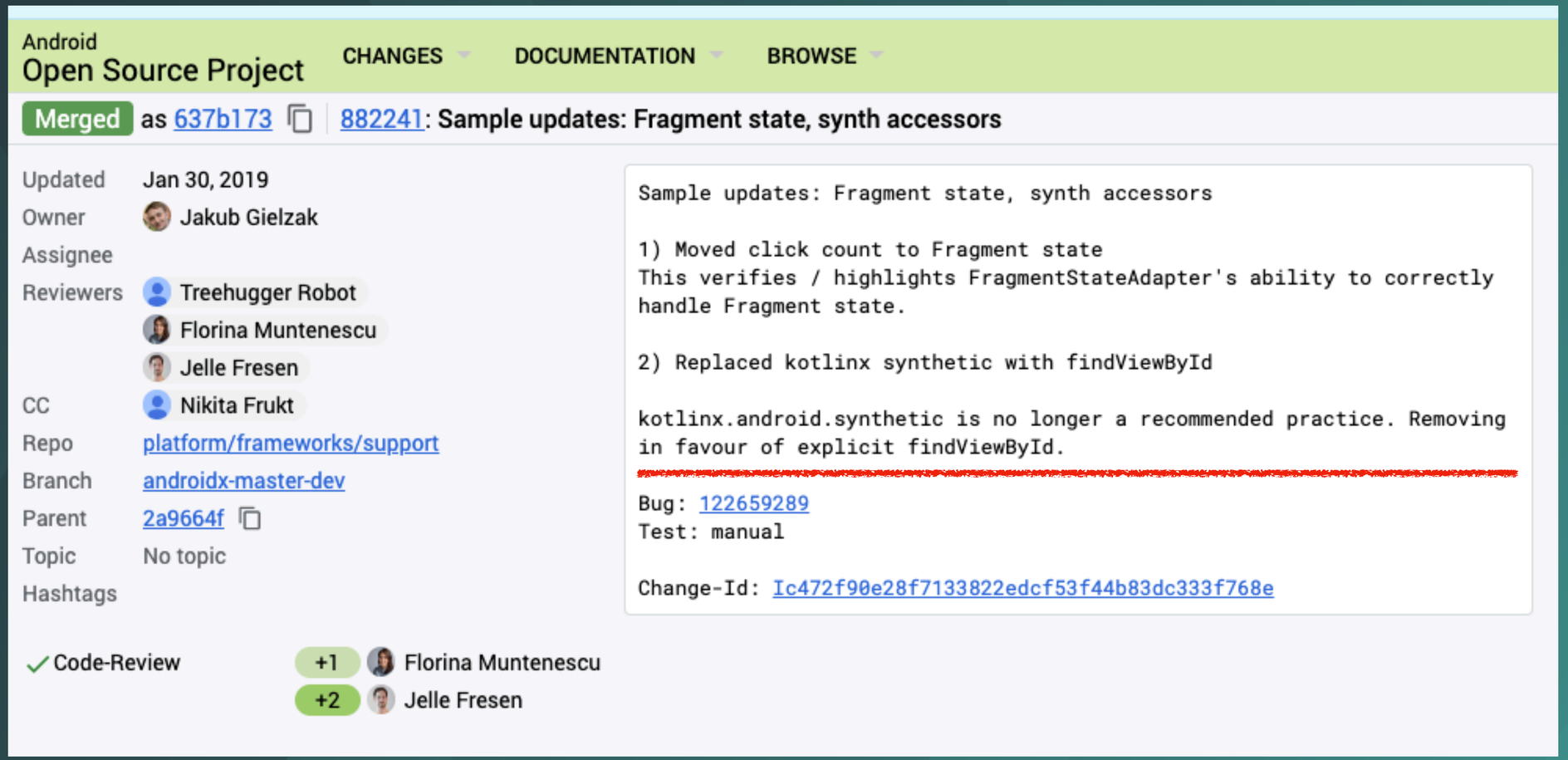
```
public class MainActivity extends AppCompatActivity {
    private TextView txtViewMainTitle;
    private TextView txtViewSubTitle;

    @Override
    protected void onCreate(@Nullable Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);

        txtViewMainTitle = findViewById(R.id.mainTitle;
        txtViewSubTitle = findViewById(R.id.subTitle);

        txtViewMainTitle.setText("This is my main title");
        txtViewSubTitle.setText("This is my subTitle");
    }
}
```

ViewBinding



android-review.googlesource.com/c/platform/frameworks/support/+/882241

ViewBinding
The Argument Over Kotlin Synthetics

- They Are Kotlin Only.
- They Don’t Expose Nullability.
- The Code Generated Is Not Guaranteed To Be Performant.
- Everything Exists In A Global Namespace.
- Typing Isn’t Guaranteed.

proandroiddev.com/the-argument-over-kotlin-synthetics-735305dd4ed0

ViewBinding

Android Studio 3.6 Canary 11+.

app/build.gradle:

```
android {
    ...
    viewBinding {
        enabled = true
    }
}
```

ViewBinding

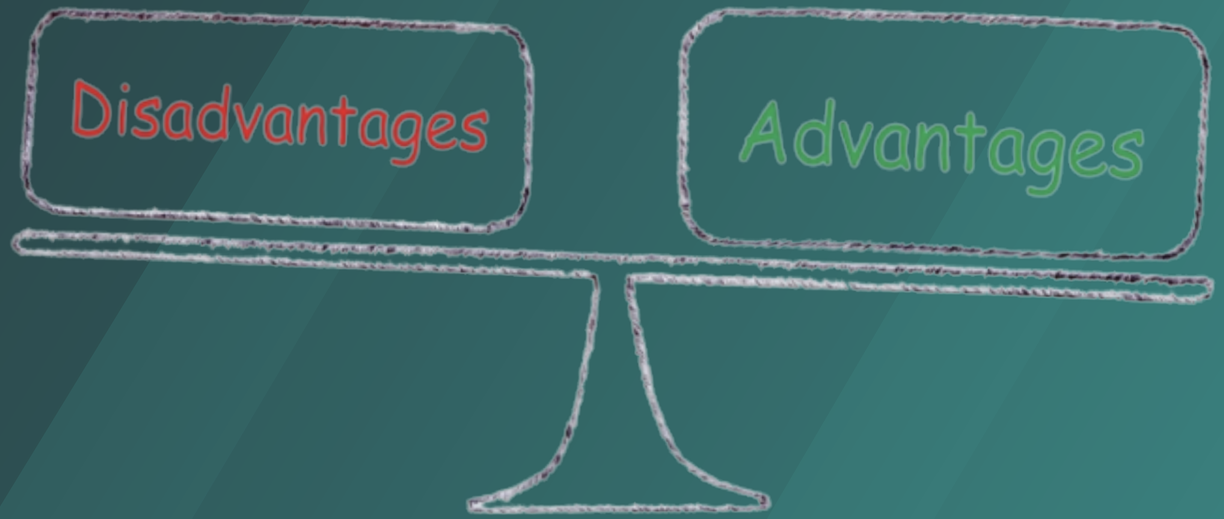
build.gradle:

```
buildscript {
    ext.kotlin_version = '1.7.20'
    repositories {
        google()
        mavenCentral()
    }
    dependencies {
        classpath “com.android.tools.build:gradle:7.3.1"
        classpath "org.jetbrains.kotlin:kotlin-gradle-plugin:$kotlin_version"
    }
}
```

ViewBinding

- The auto-generated .class increases app size.
- Harder to debug.

- Null safety.
- Type safety.
- Speed.

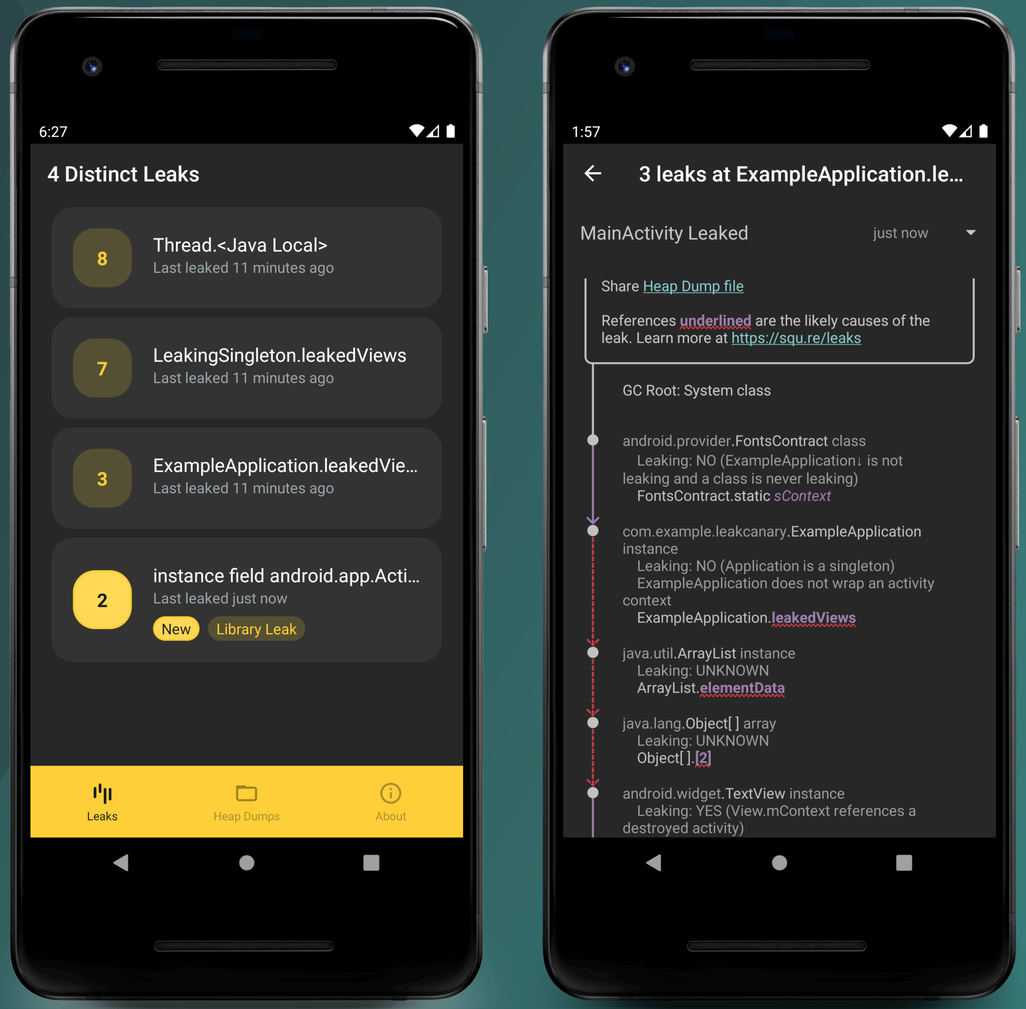


proandroiddev.com/the-argument-over-kotlin-synthetics-735305dd4ed0

Memory Leaks

“A small leak will sink a great ship.” - Benjamin Franklin

Memory Leaks



https://square.github.io/leakcanary

Memory Leaks

```
dependencies {
  // debugImplementation because LeakCanary should only run in debug builds.
  debugImplementation 'com.squareup.leakcanary:leakcanary-android:2.10'
}
```

https://square.github.io/leakcanary

Memory Leaks


```
dependencies {
  // debugImplementation because LeakCanary should only run in debug builds.
  debugImplementation 'com.squareup.leakcanary:leakcanary-android:2.10'
}
```

LeakCanary automatically detects leaks of the following objects:

- Destroyed Activity instances.
- Destroyed Fragment instances.
- Destroyed fragment View instances.
- Cleared ViewModel instances.
- Destroyed Service instance.

https://square.github.io/leakcanary

Lecture outcomes

- Identify app widgets, and understand the key parts of an app widget.
- Implement app widget actions when an element of an app widget is tapped.
- Add support for different languages.
- Test your app for accessibility in a variety of ways.
- Use the Profile GPU Rendering tool to visualize Android drawing the screen.
- Use Memory Profiler to collect data about your app.