

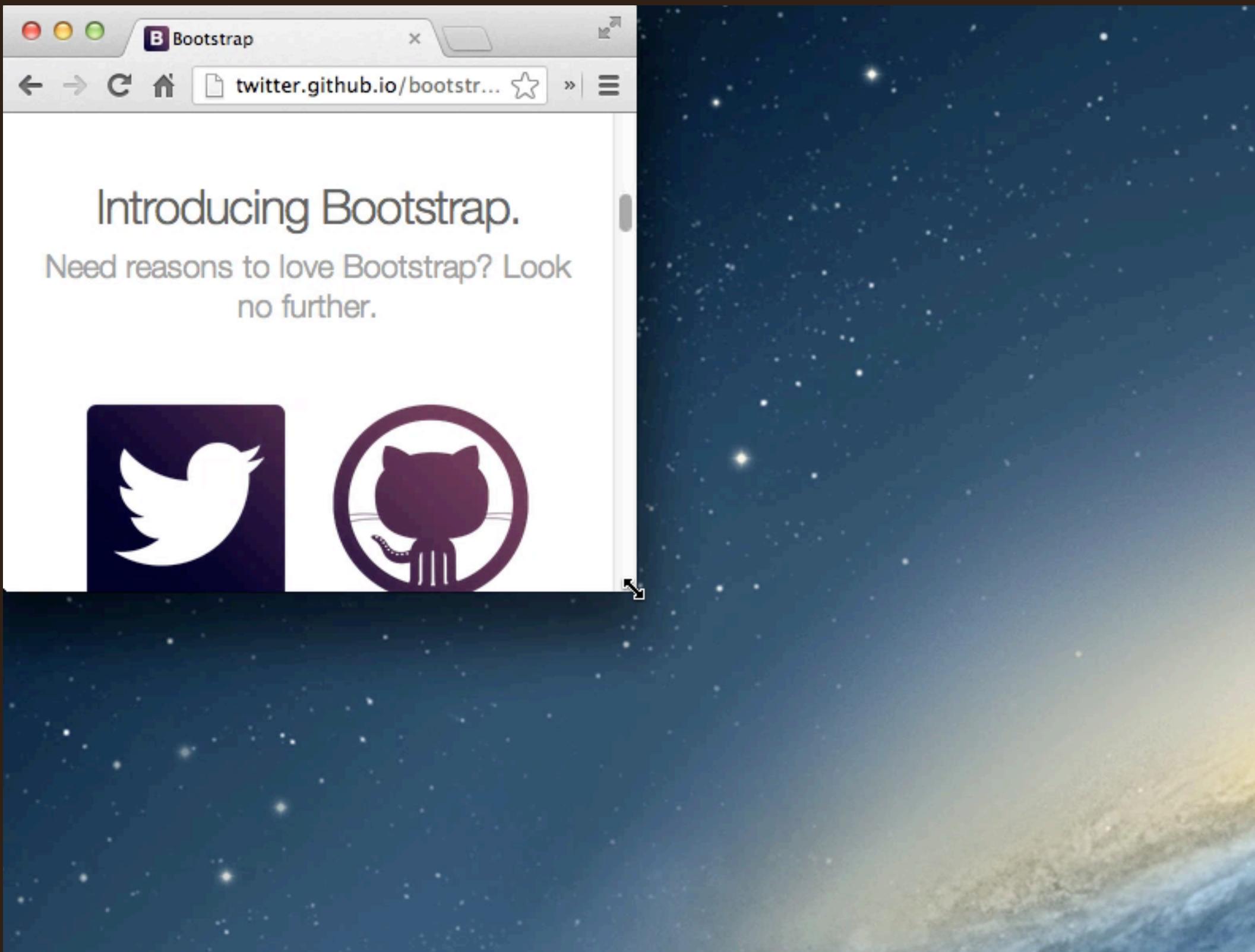
Supporting Multiple Android Device Definitions

Chiu-Ki Chan
@chiuki
bit.ly/muland

So many devices!



Infinite screen sizes!

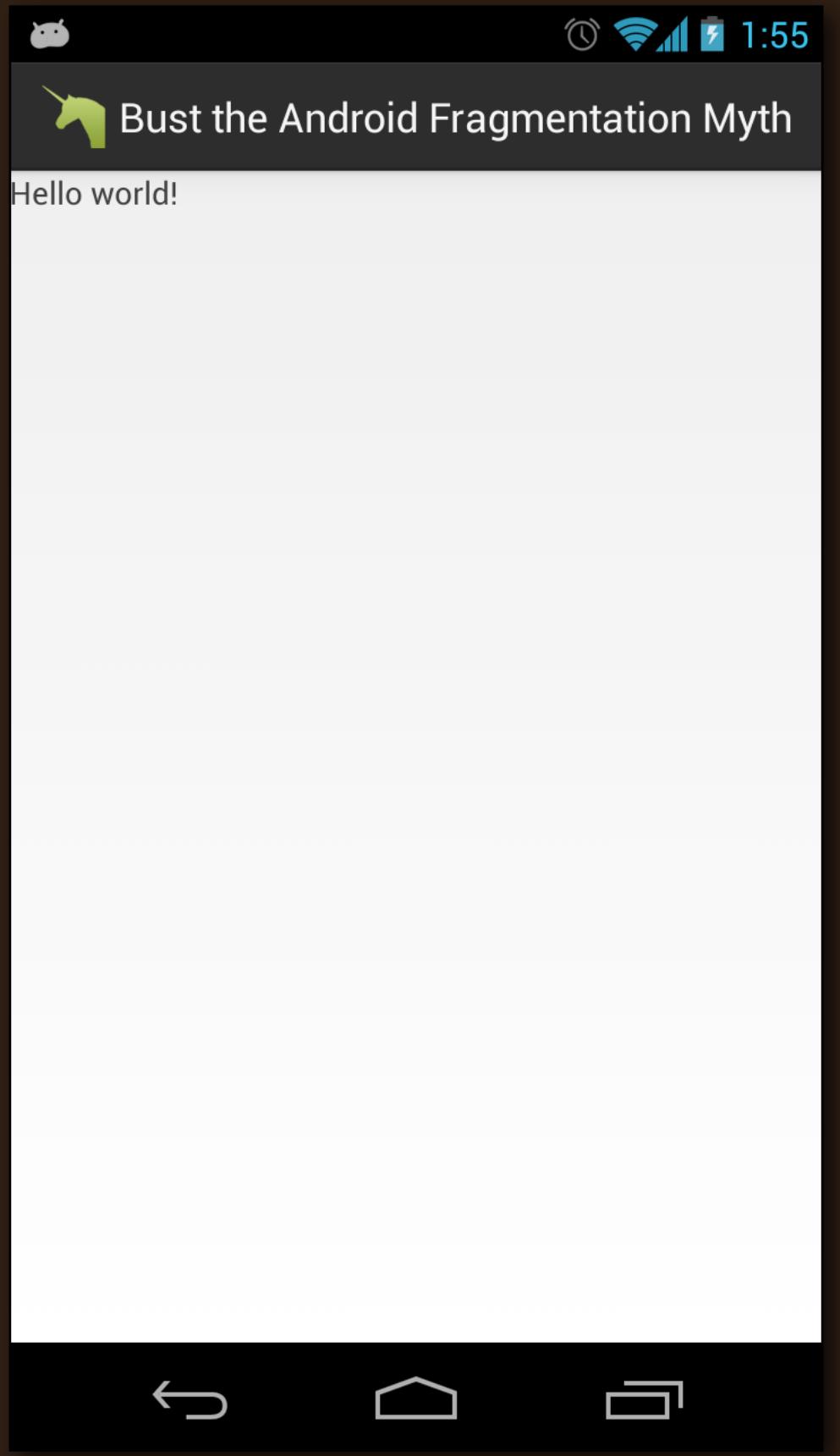


Responsive

+

Progressive

Hello World



@chiuki

The screenshot shows the Android Studio interface with the project 'bust' open. The left pane displays the project structure, and the right pane shows the code editor for `MainActivity.java`.

`MainActivity.java` content:

```
1 package com.sqisland.android.bust;
2
3 import android.os.Bundle;
4 import android.app.Activity;
5 import android.widget.TextView;
6
7 public class MainActivity extends Activity {
8     private TextView textView;
9
10    @Override
11    protected void onCreate(Bundle savedInstanceState) {
12        super.onCreate(savedInstanceState);
13        setContentView(R.layout.activity_main);
14
15        textView = (TextView) findViewById(R.id.text);
16        textView.setText(R.string.hello_world);
17    }
18}
```

The screenshot shows the Android Studio interface with the project 'bust' open. The left pane displays the project structure, and the right pane shows the code editor for `MainActivity.java`.

`MainActivity.java` content:

```
1 package com.sqisland.android.bust;
2
3 import android.os.Bundle;
4 import android.app.Activity;
5 import android.widget.TextView;
6
7 public class MainActivity extends Activity {
8     private TextView textView;
9
10    @Override
11    protected void onCreate(Bundle savedInstanceState) {
12        super.onCreate(savedInstanceState);
13        setContentView(R.layout.activity_main);
14
15        textView = (TextView) findViewById(R.id.text);
16        textView.setText(R.string.hello_world);
17    }
18}
```

The screenshot shows the Android Studio interface with the project 'bust' open. The left pane displays the project structure, and the right pane shows the code editor for `MainActivity.java`.

Project Structure:

- bust (~/projects/android-studio/bust)**
 - .idea
 - Bust
 - build
 - libs
 - src
 - main
 - java
 - com.sqisland.android.bust
 - res
 - drawable-hdpi
 - drawable-mdpi
 - drawable-xhdpi
 - drawable-xxhdpi
 - layout
 - activity_main.xml
 - values
 - strings.xml
 - styles.xml
 - values-v11
 - values-v14
 - build.gradle
 - Bust.iml
 - gradle
 - build.gradle
 - bust.iml
 - gradlew
 - gradlew.bat

MainActivity.java Code:

```
1 package com.sqisland.android.bust;
2
3 import android.os.Bundle;
4 import android.app.Activity;
5 import android.widget.TextView;
6
7 public class MainActivity extends Activity {
8     private TextView textView;
9
10    @Override
11    protected void onCreate(Bundle savedInstanceState) {
12        super.onCreate(savedInstanceState);
13        setContentView(R.layout.activity_main);
14
15        textView = (TextView) findViewById(R.id.text);
16        textView.setText(R.string.hello_world);
17    }
18}
```

The screenshot shows the Android Studio interface with the following details:

- Project View:** On the left, the project structure for the "bust" application is displayed. It includes the .idea folder, the Bust module with its build, libs, and src directories containing main/java (with MainActivity.java), res (drawable-hdpi, drawable-mdpi, drawable-xhdpi, drawable-xxhdpi, layout, values), and build.gradle files.
- MainActivity.java:** The code editor shows the Java class for the main activity. It contains a single TextView element with the following XML attributes:

```
<TextView  
    xmlns:android="http://schemas.android.com/apk/res/android"  
    android:id="@+id/text"  
    android:layout_width="wrap_content"  
    android:layout_height="wrap_content" />
```
- activity_main.xml:** The XML layout file for the main activity is shown in the code editor, defining a single TextView element with the specified attributes.

@chiuki

The screenshot shows the Android Studio interface with the following details:

- Project View:** On the left, the project structure for the "bust" application is displayed. It includes the .idea folder, the Bust module with its build, libs, and src directories containing main/java and res (with drawable-hdpi, drawable-mdpi, drawable-xhdpi, drawable-xxhdpi, layout, values, and AndroidManifest.xml), and build.gradle and Bust.iml files.
- Code Editor:** On the right, the code editor has two tabs: "MainActivity.java" and "activity_main.xml". The "activity_main.xml" tab is currently active, showing XML code for a TextView:

```
<TextView  
    xmlns:android="http://schemas.android.com/apk/res/android"  
    android:id="@+id/text"  
    android:layout_width="wrap_content"  
    android:layout_height="wrap_content" />
```

The line `android:id="@+id/text"` is highlighted with a red box.

@chiuki

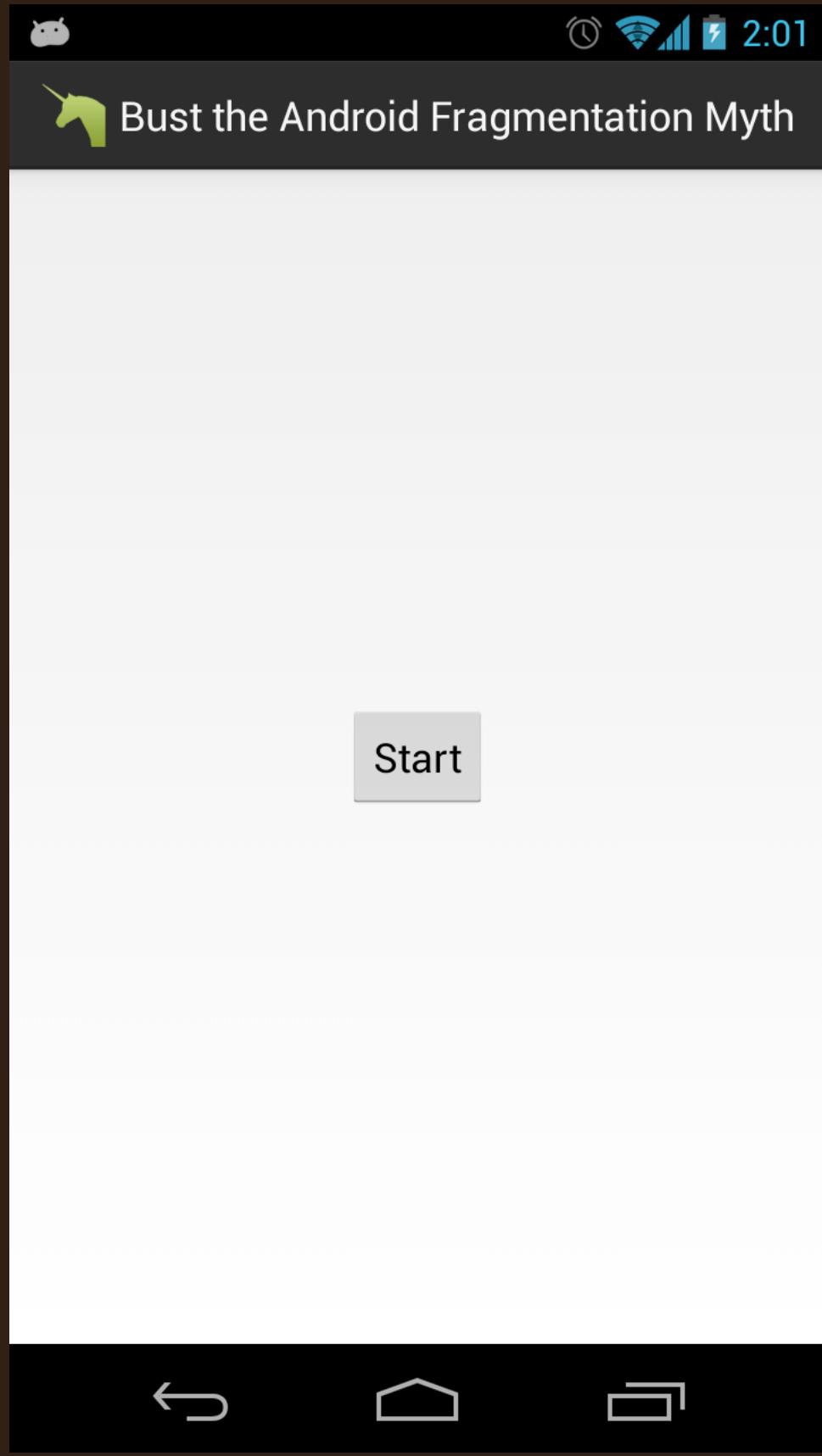
The screenshot shows the Android Studio interface with the project 'bust' open. The left pane displays the project structure, including the .idea folder, Bust module (with build, libs, and src folders), and various resource and configuration files like build.gradle, Bust.iml, gradle, and gradlew. The right pane shows the code editor with two tabs: MainActivity.java and activity_main.xml. The MainActivity.java code is as follows:

```
1 package com.sqisland.android.bust;
2
3 import android.os.Bundle;
4 import android.app.Activity;
5 import android.widget.TextView;
6
7 public class MainActivity extends Activity {
8     private TextView textView;
9
10    @Override
11    protected void onCreate(Bundle savedInstanceState) {
12        super.onCreate(savedInstanceState);
13        setContentView(R.layout.activity_main);
14
15        textView = (TextView) findViewById(R.id.text);
16        textView.setText(R.string.hello_world);
17    }
18}
```

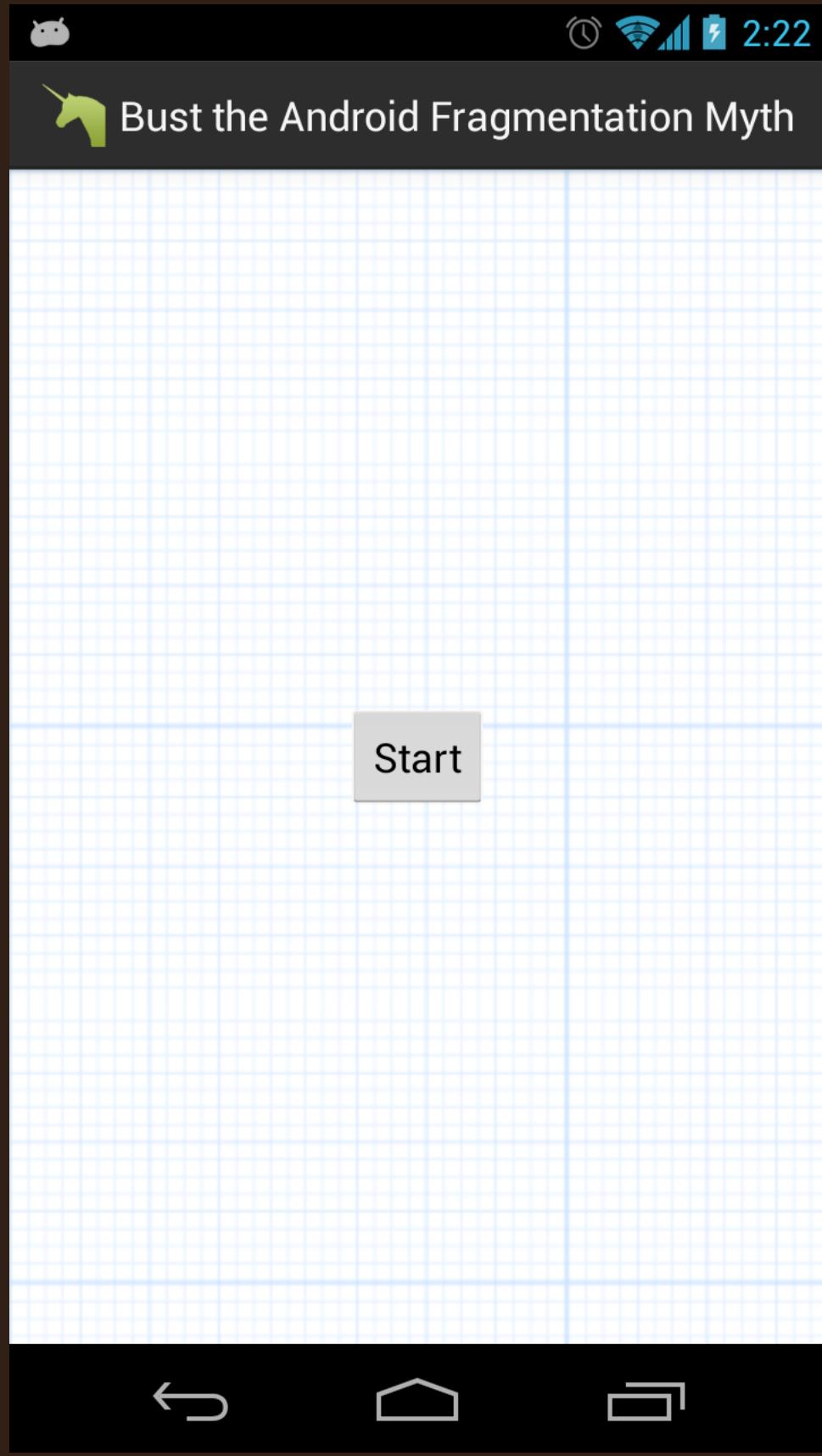
The line `textView = (TextView) findViewById(R.id.text);` is highlighted with a red rectangle.

@chiuki

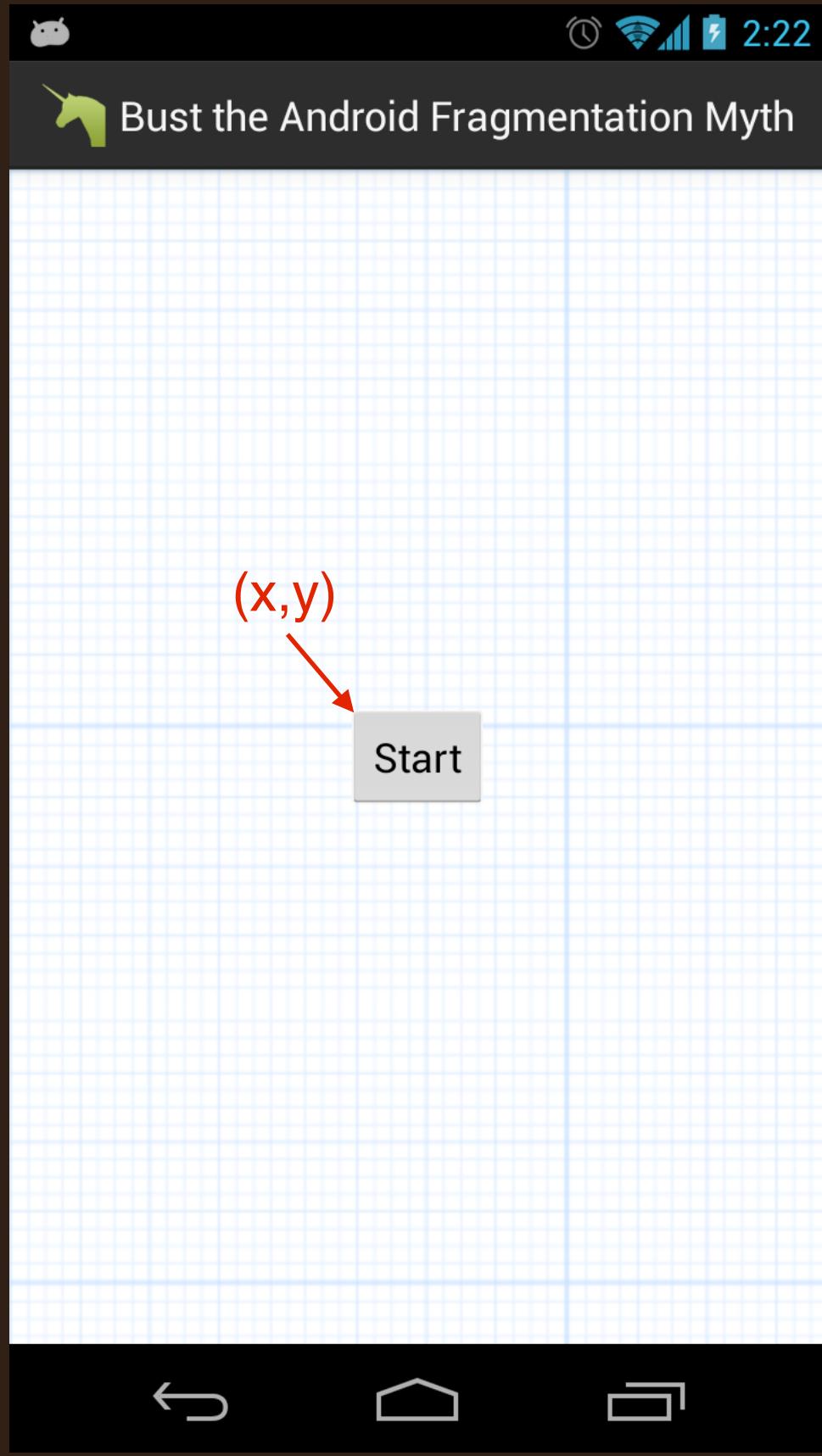
Declarative Layout



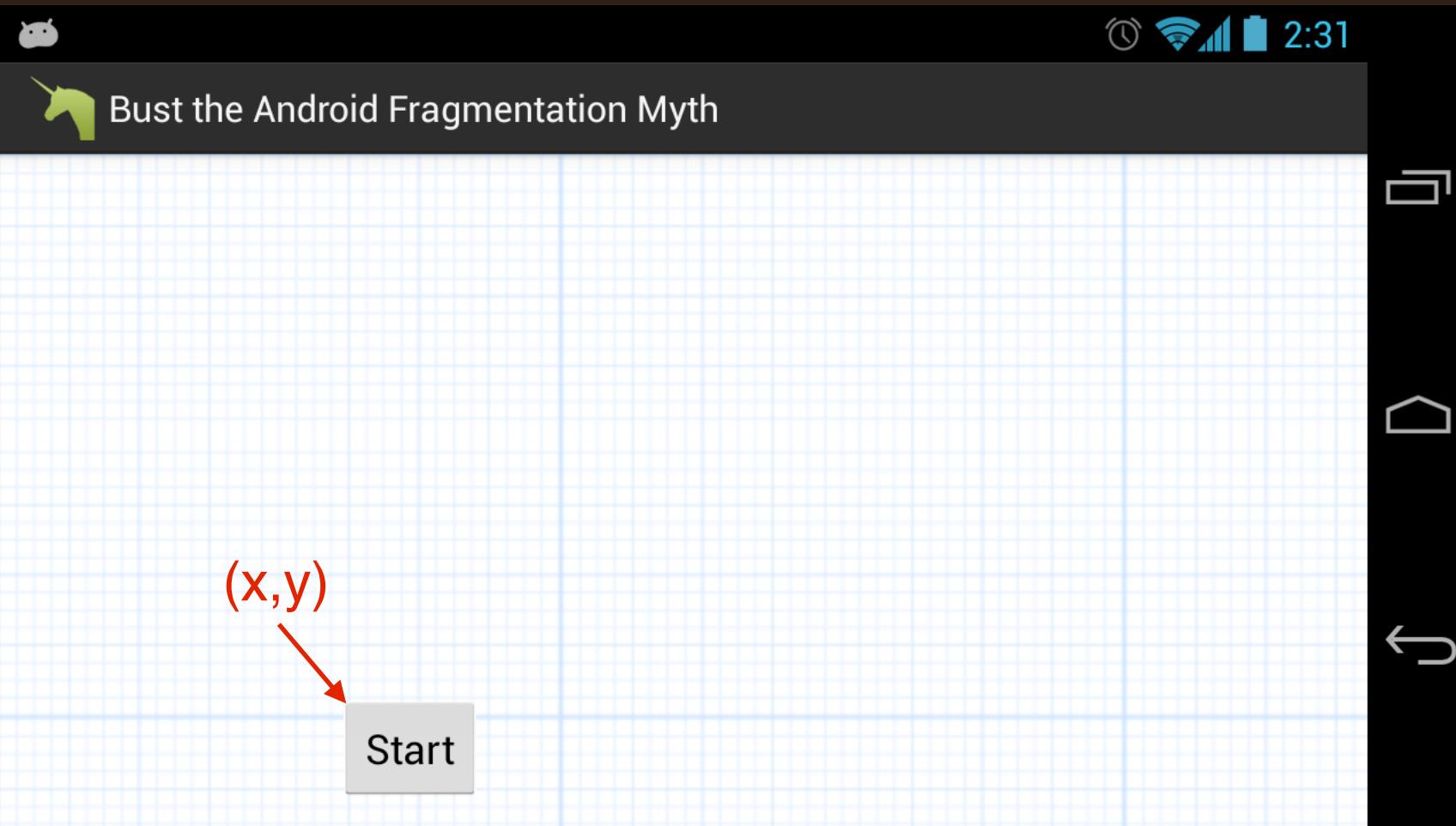
@chiuki

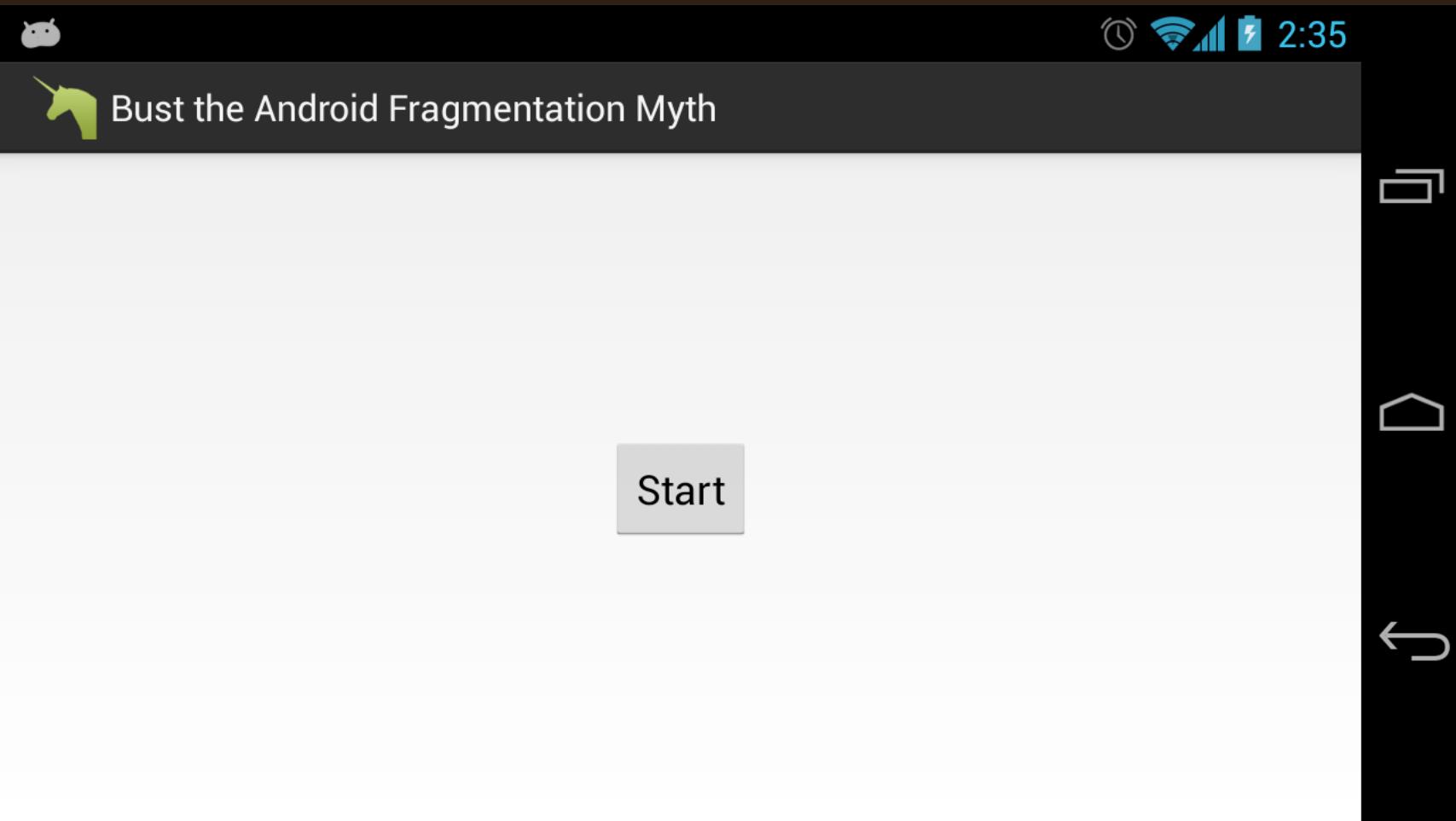


@chiuki



@chiuki

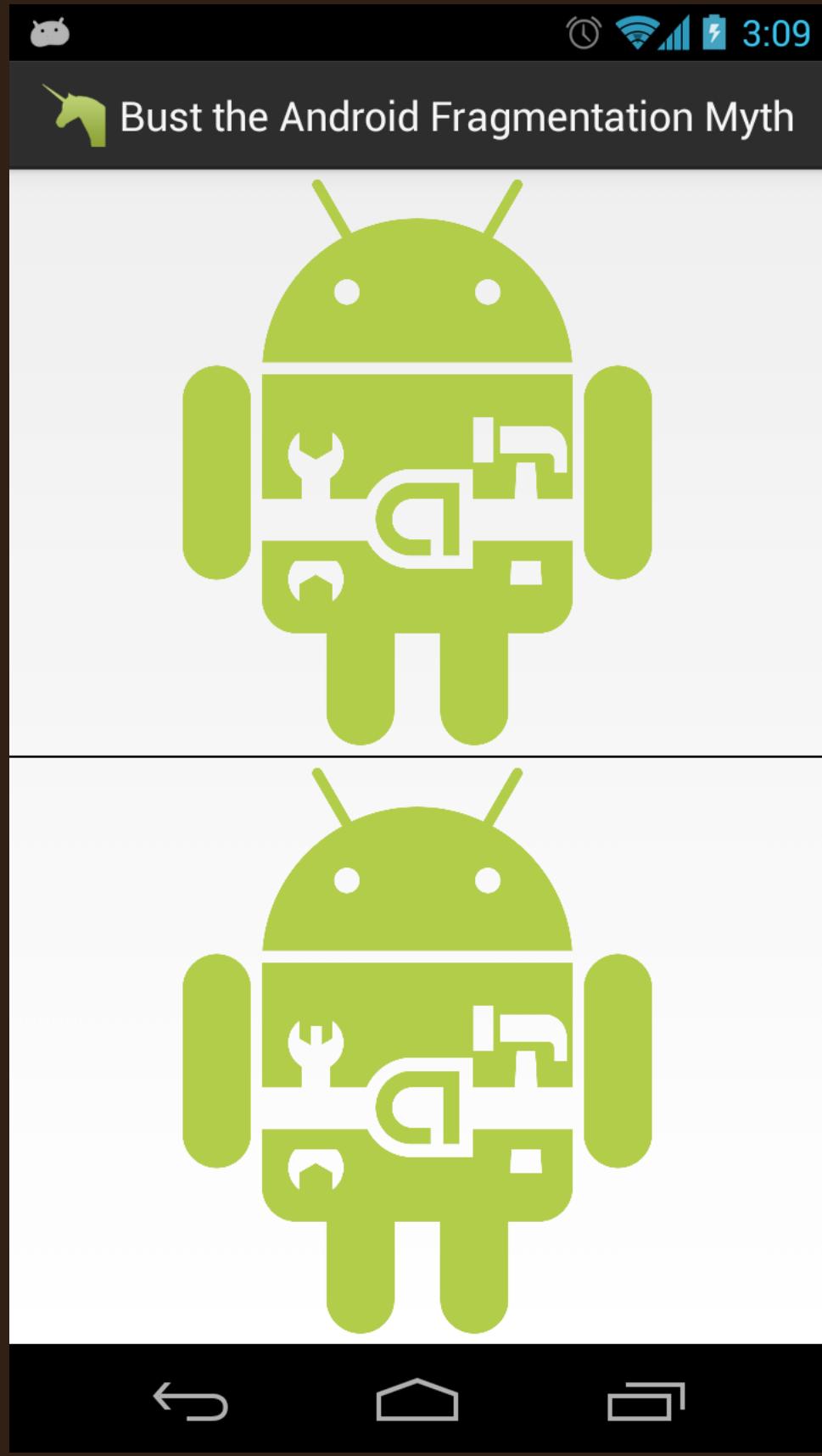




Center

Center

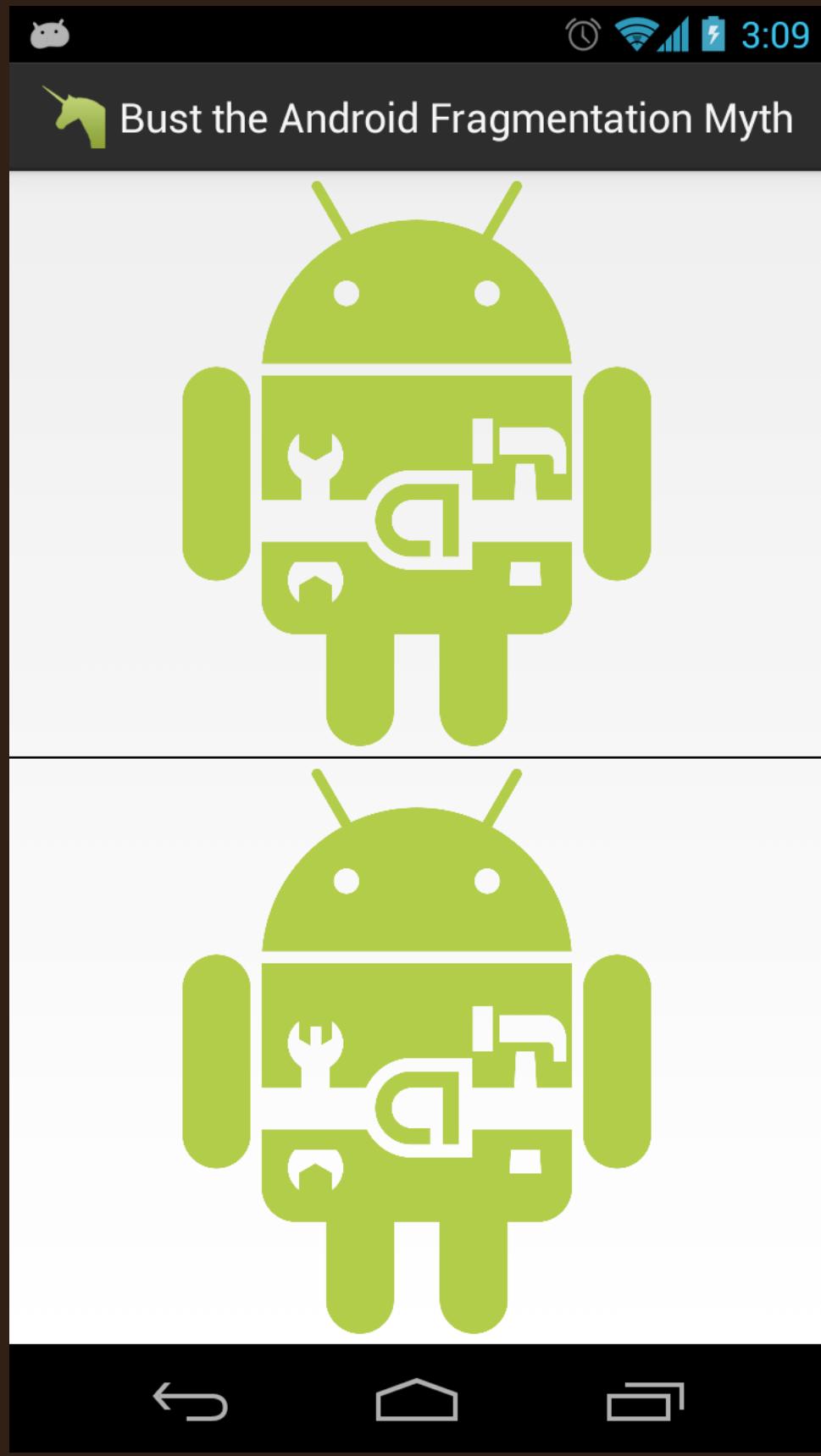
```
<Button  
    android:layout_width="wrap_content"  
    android:layout_height="wrap_content"  
    android:layout_gravity="center"  
    android:text="@string/start" />
```



@chiuki

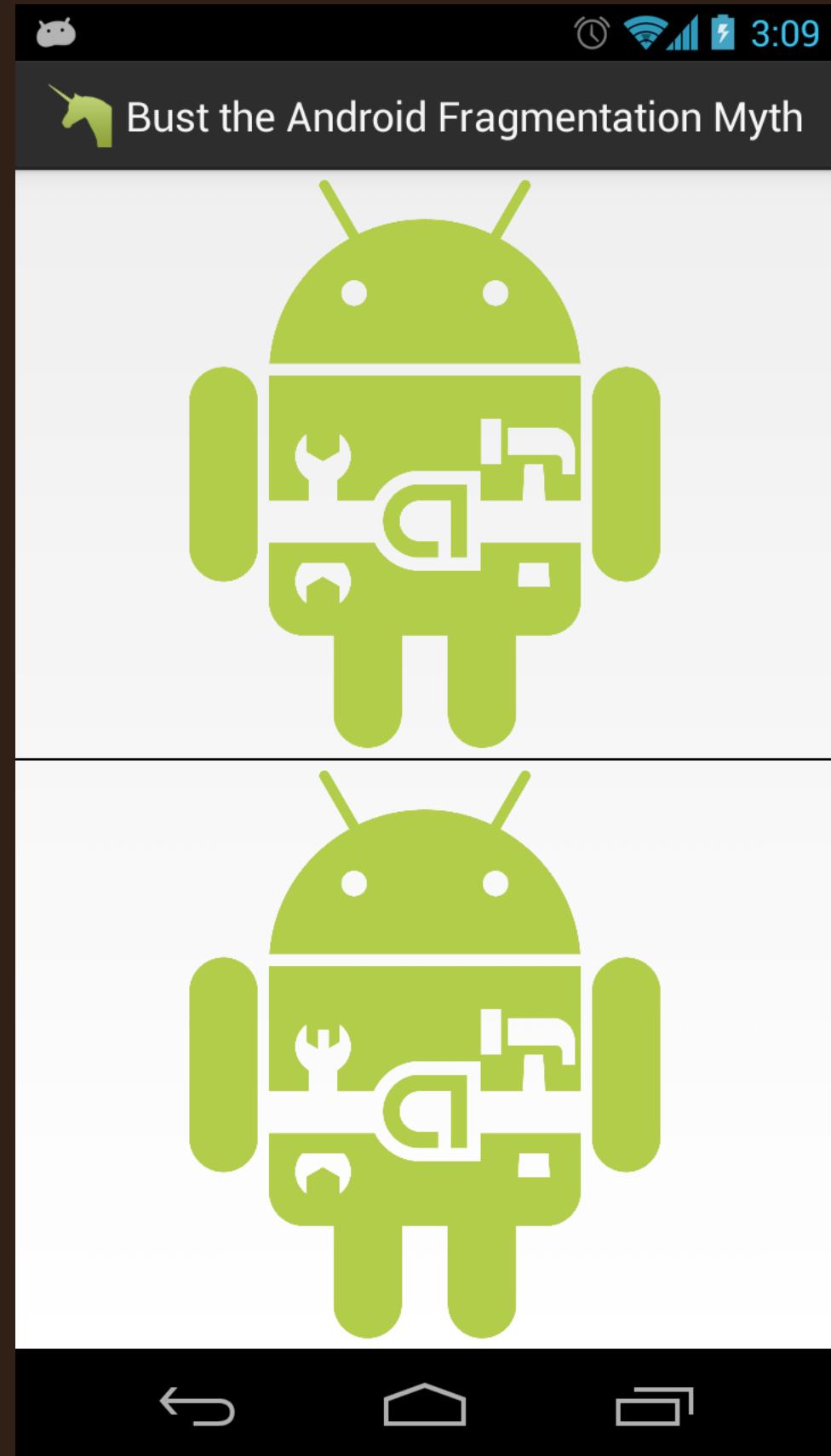
Proportional Layout

```
<LinearLayout  
    android:layout_width="match_parent"  
    android:layout_height="match_parent"  
    android:orientation="vertical" >  
    <ImageView  
        android:id="@+id/image1"  
        android:layout_width="match_parent"  
        android:layout_height="0dp"  
        android:layout_weight="1" />  
    <ImageView  
        android:id="@+id/image2"  
        android:layout_width="match_parent"  
        android:layout_height="0dp"  
        android:layout_weight="1" />  
</LinearLayout>
```



@chiuki

Divider



Divider

```
<LinearLayout>
    <ImageView
        android:layout_width="match_parent"
        android:layout_height="0dp"
        android:layout_weight="1" />
    <View
        android:layout_width="match_parent"
        android:layout_height="1dp"
        android:background="@android:color/black" />
    <ImageView
        android:layout_width="match_parent"
        android:layout_height="0dp"
        android:layout_weight="1" />
</LinearLayout>
```

Divider

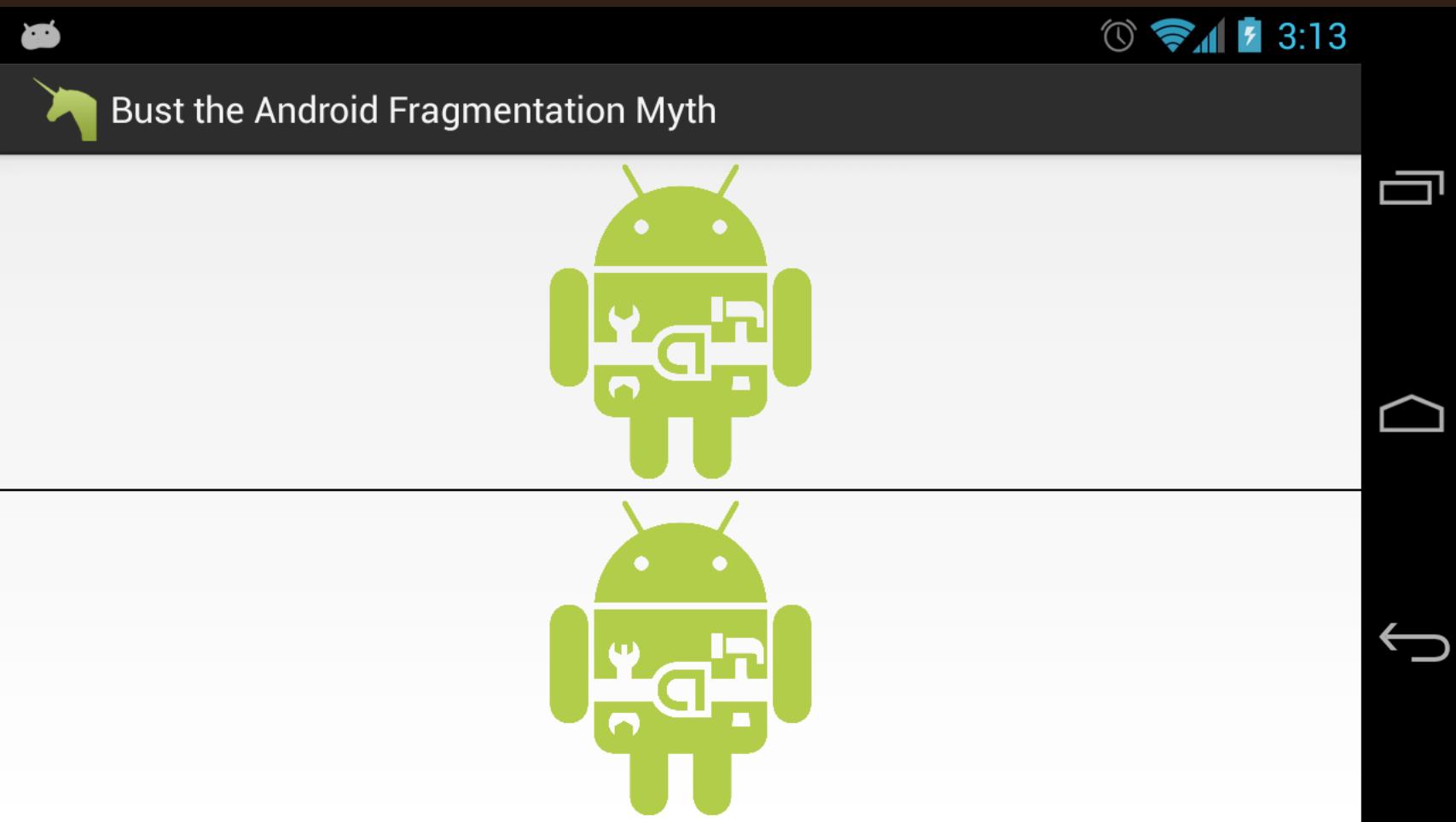
```
<LinearLayout>
    <ImageView
        android:layout_width="match_parent"
        android:layout_height="0dp"
        android:layout_weight="1" />
    <View
        android:layout_width="match_parent"
        android:layout_height="1dp"
        android:background="@android:color/black" />
    <ImageView
        android:layout_width="match_parent"
        android:layout_height="0dp"
        android:layout_weight="1" />
</LinearLayout>
```

dp

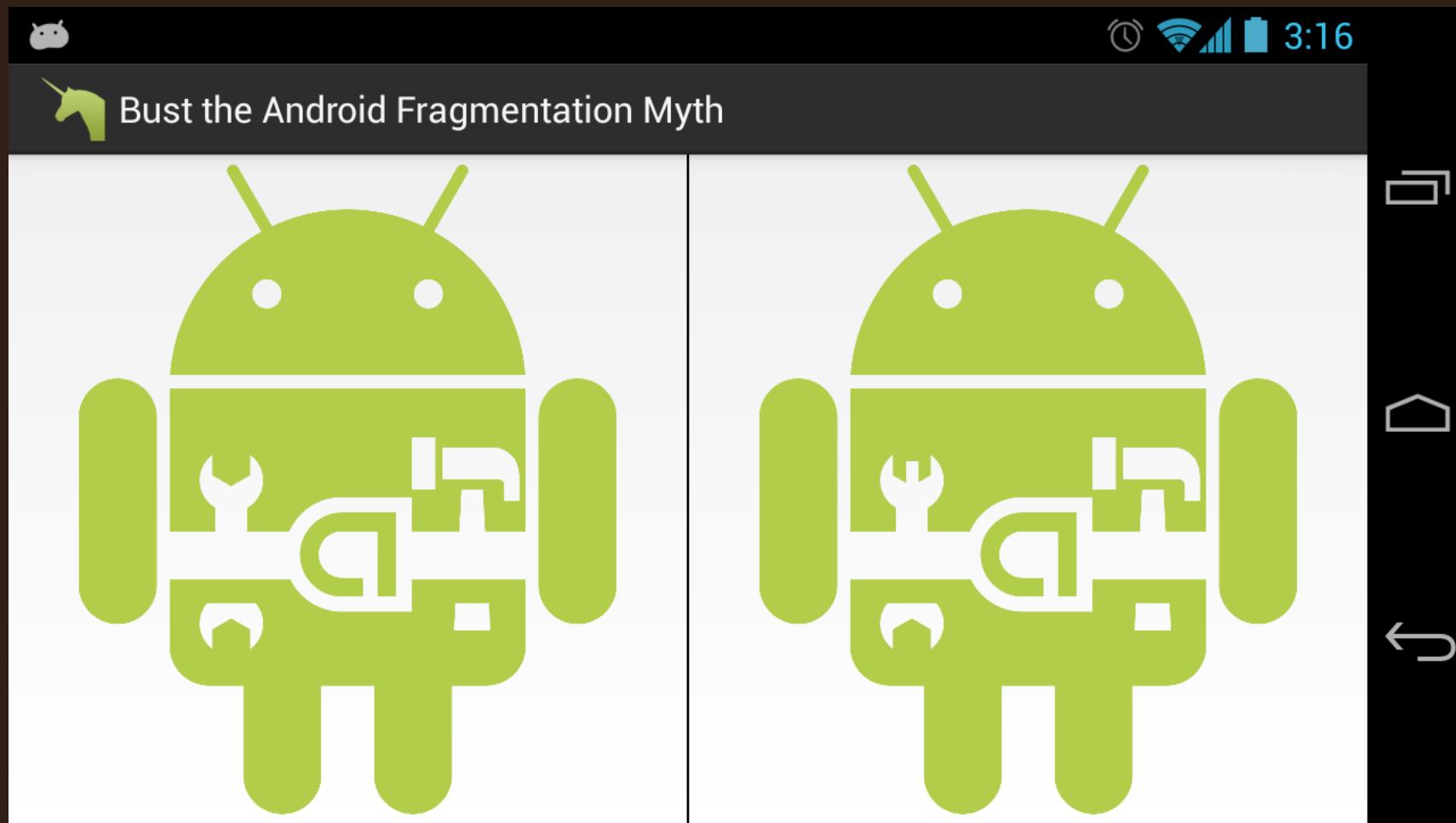
@chiuki

dp

Density-independent pixel



@chiuki



@chiuki

res/layout-land

```
<LinearLayout  
    android:layout_width="match_parent"  
    android:layout_height="match_parent"  
    android:orientation="horizontal" >  
    <ImageView  
        android:id="@+id/image1"  
        android:layout_width="0dp"  
        android:layout_height="match_parent"  
        android:layout_weight="1" />  
    <ImageView  
        android:id="@+id/image2"  
        android:layout_width="0dp"  
        android:layout_height="match_parent"  
        android:layout_weight="1" />  
</LinearLayout>
```

The screenshot shows the Android Studio interface with the project 'bust' open. The left pane displays the project structure, and the right pane shows the code editor with the MainActivity.java file.

Project Structure:

- Project: bust (~/projects/android-studio/bust)
- .idea
- Bust
- build
- libs
- src
 - main
 - java
 - com.sqisland.android.bust
 - MainActivity
 - res
 - drawable-hdpi
 - drawable-mdpi
 - drawable-xhdpi
 - drawable-xxhdpi
 - layout
 - activity_main.xml
 - layout-land
 - activity_main.xml
 - values
 - strings.xml
 - styles.xml
 - values-v11
 - values-v14
 - AndroidManifest.xml
 - build.gradle
 - Bust.iml
 - gradle
 - build.gradle
 - bust.iml

Code Editor (MainActivity.java):

```
1 package com.sqisland.android.bust;
2
3 import android.os.Bundle;
4 import android.app.Activity;
5 import android.widget.ImageView;
6
7 public class MainActivity extends Activity {
8     private ImageView image1;
9     private ImageView image2;
10
11     @Override
12     protected void onCreate(Bundle savedInstanceState) {
13         super.onCreate(savedInstanceState);
14         setContentView(R.layout.activity_main);
15
16         image1 = (ImageView) findViewById(R.id.image1);
17         image2 = (ImageView) findViewById(R.id.image2);
18
19         image1.setImageResource(R.drawable.android_tools_1);
20         image2.setImageResource(R.drawable.android_tools_2);
21     }
22 }
```

Resource Folders

Resource Folders

<http://developer.android.com/guide/topics/resources/providing-resources.html>

Type	Variation
layout	Language & Region: en, fr, fr-rCA
values	Screen size: small, large, xlarge, sw320dp, h720dp
drawable	Screen orientation: port, land
menu	Screen density: ldpi, mdpi, hdpi, xhdpi, nodpi, tvdpi
	Platform version: v4, v11, v14, v21
	UI mode: car, desk, television, appliance

Resource Folders

<http://developer.android.com/guide/topics/resources/providing-resources.html>

Type	Variation
layout	Language & Region: en, fr, fr-rCA
values	Screen size: small, large, xlarge, sw320dp, h720dp
drawable	Screen orientation: port, land
menu	Screen density: ldpi, mdpi, hdpi, xhdpi, nodpi, tvdpi
	Platform version: v4, v11, v14, v21
	UI mode: car, desk, television, appliance

res/layout-land/activity_main.xml

Resource Folders

<http://developer.android.com/guide/topics/resources/providing-resources.html>

Type	Variation
layout values drawable menu	Language & Region: en, fr, fr-rCA Screen size: small, large, xlarge, sw320dp, h720dp Screen orientation: port, land Screen density: ldpi, mdpi, hdpi , xhdpi, nodpi, tvdpi Platform version: v4, v11, v14, v21 UI mode: car, desk, television, appliance

res/drawable-hdpi/ic_launcher.png

Resource Folders

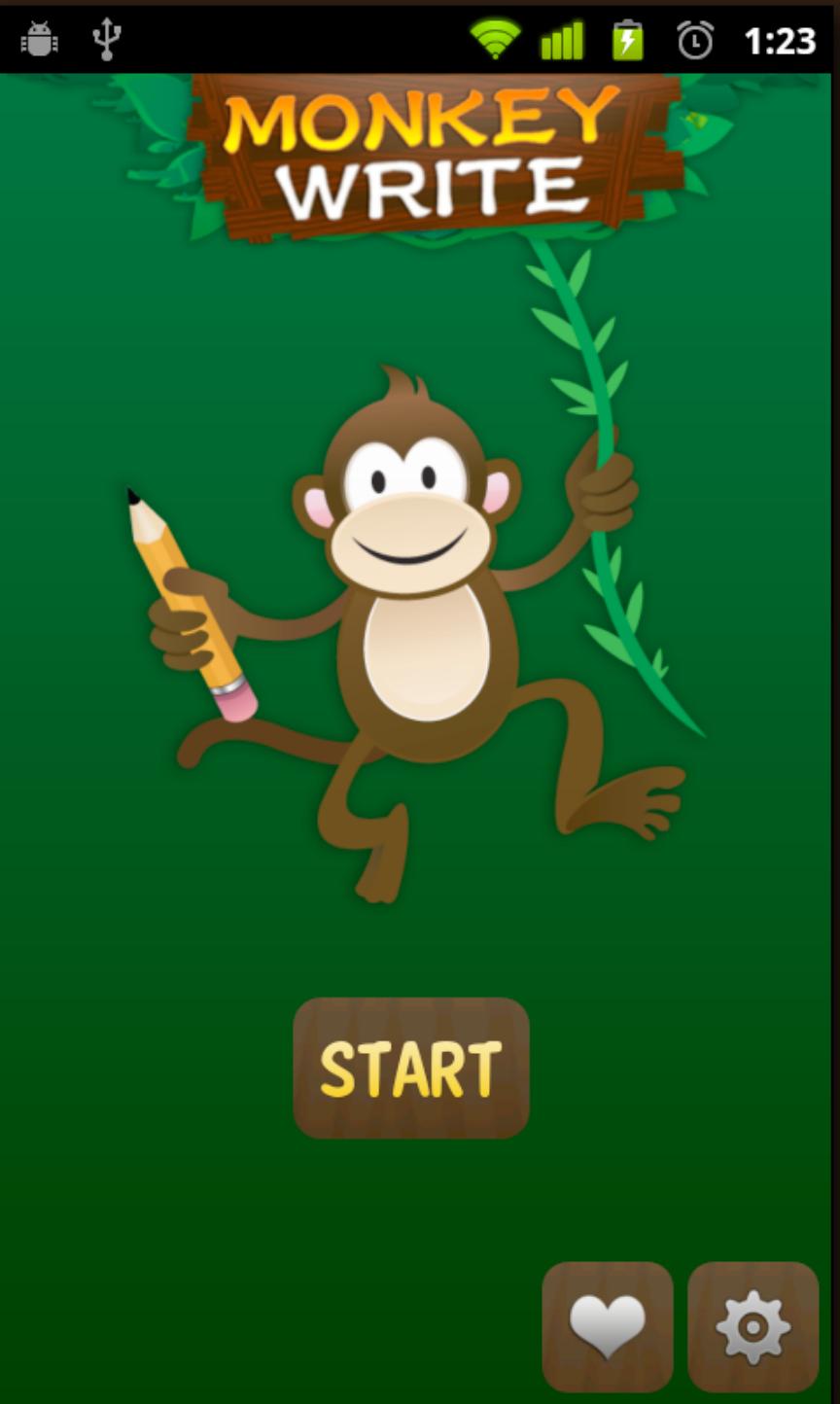
<http://developer.android.com/guide/topics/resources/providing-resources.html>

Type	Variation
layout values drawable menu	Language & Region: en, fr, fr-rCA Screen size: small, large , xlarge, sw320dp, h720dp Screen orientation: port, land Screen density: ldpi, mdpi, hdpi, xhdpi, nodpi, tvdpi Platform version: v4, v11, v14, v21 UI mode: car, desk, television, appliance

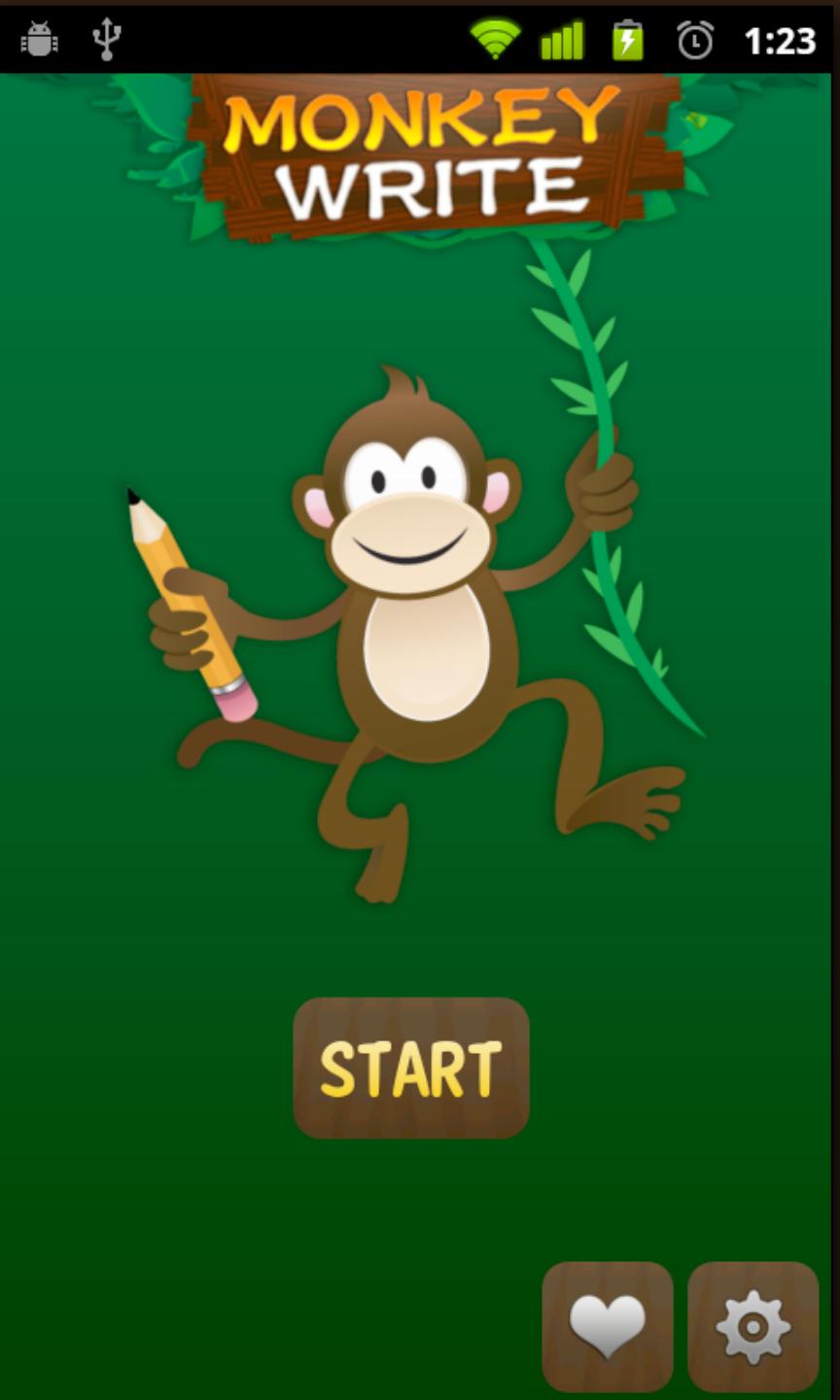
res/drawable-large-land/splash.png



res/drawable-large-land/splash.png



res/drawable/splash.xml



res/drawable/splash

xml

XML drawable

XML shape

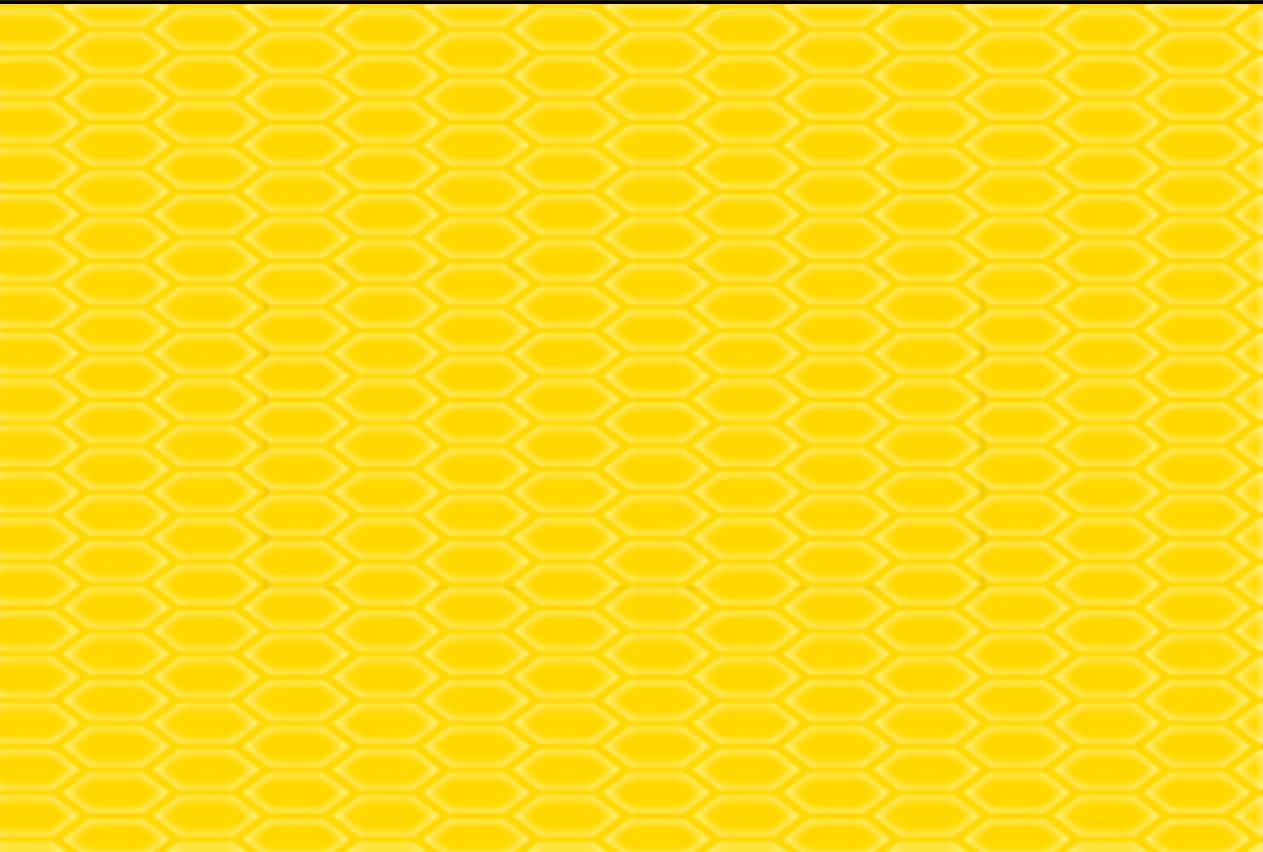
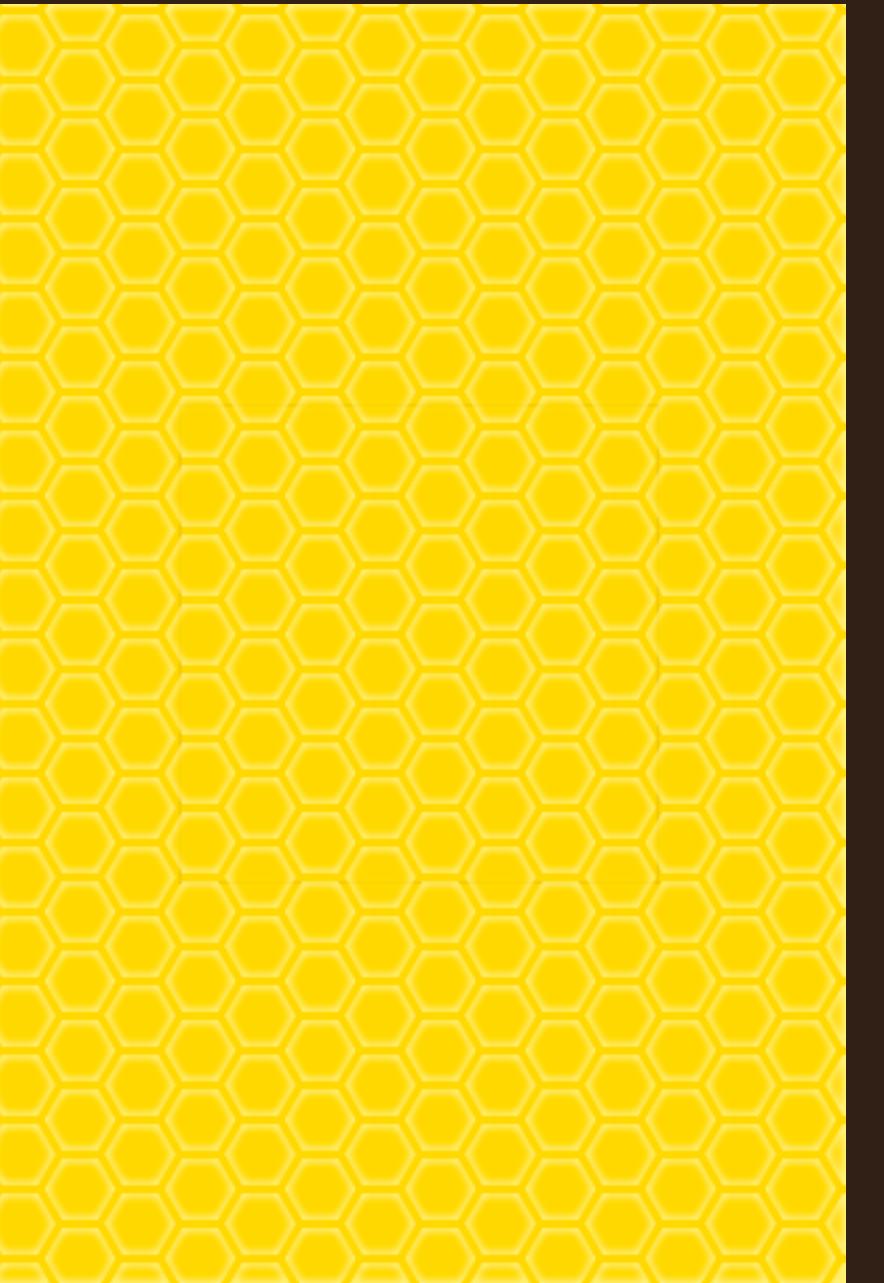
```
<shape android:shape="rectangle">  
    <gradient  
        android:startColor="#063"  
        android:endColor="#030"  
        android:angle="270" />  
</shape>
```

Gradient

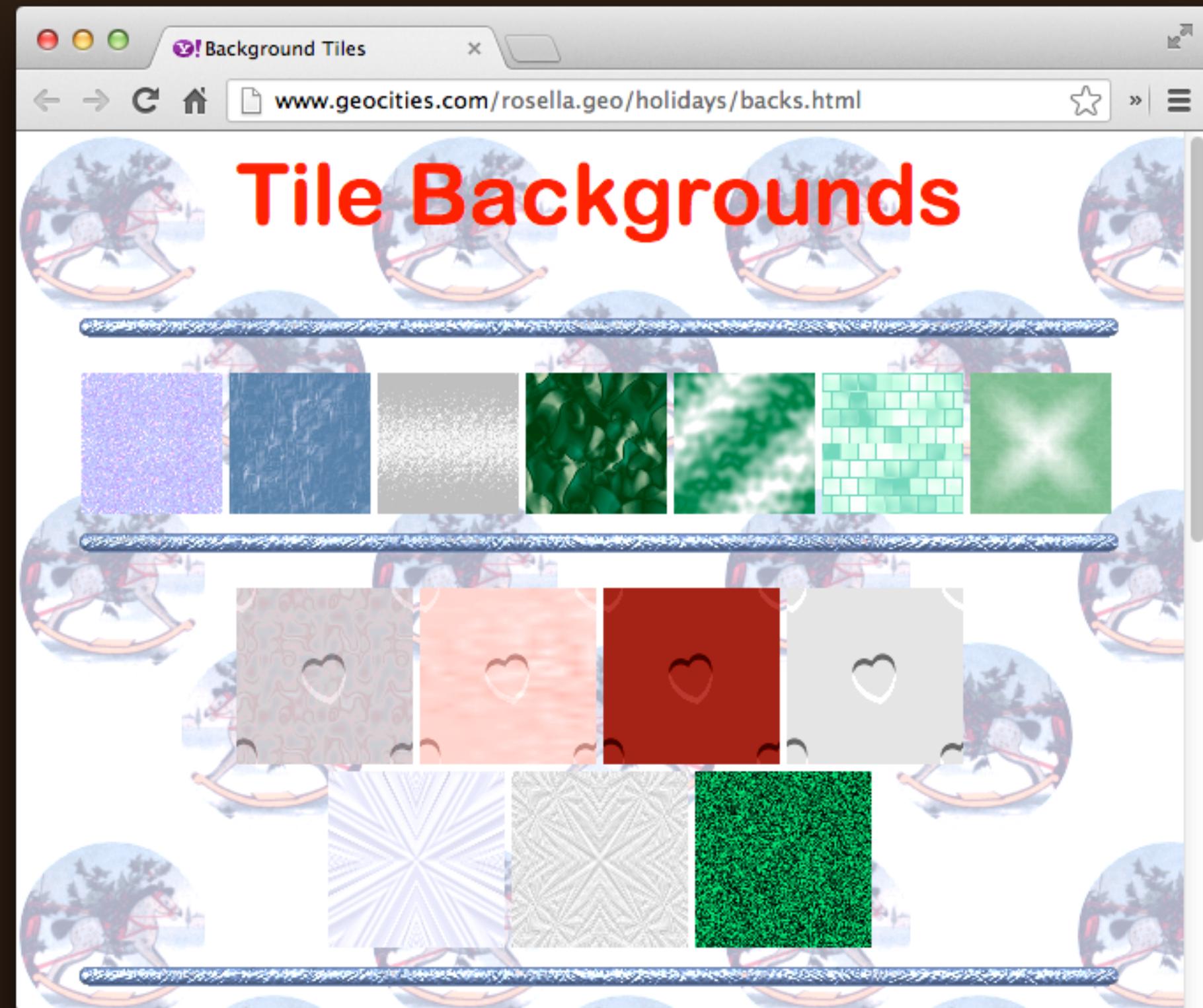


Gradient

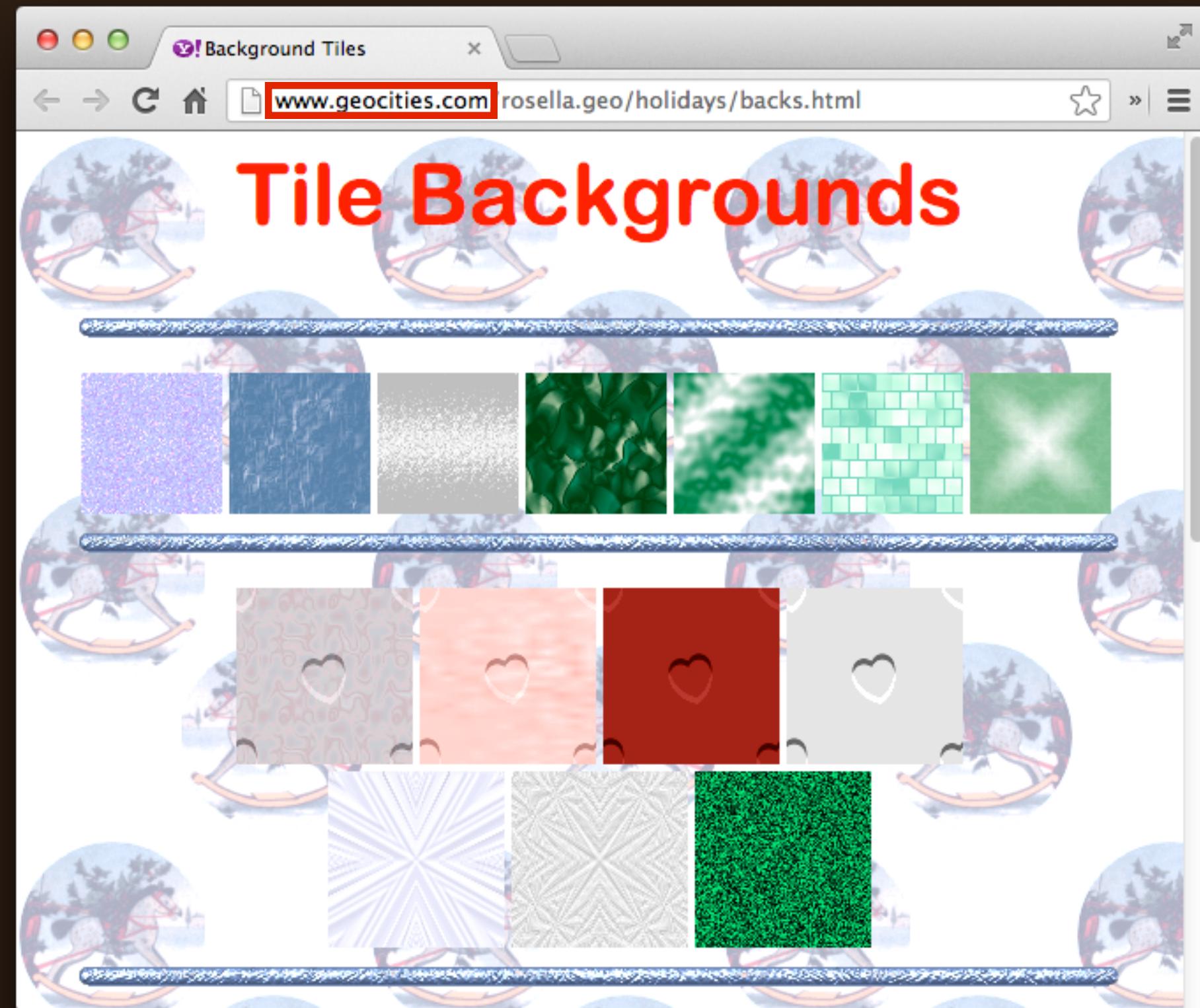
Pattern



Tiles



Tiles



Tiling background

```
<bitmap  
    android:src="@drawable/tile"  
    android:tileMode="repeat"  
    android:dither="true" />
```

res/drawable/background.xml

Tiling background

```
<bitmap  
    android:src="@drawable/tile"     tile.png  
    android:tileMode="repeat"  
    android:dither="true" />
```

res/drawable/background.xml

Tiling background

res/drawable/background.xml

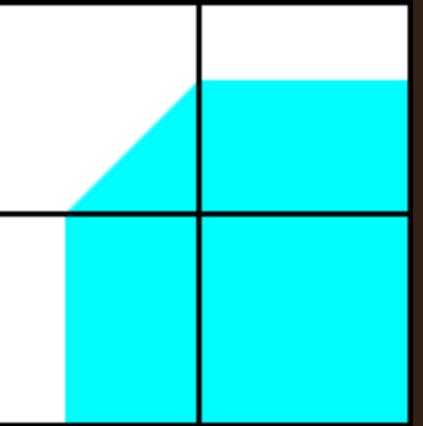
```
<View  
    android:layout_width="match_parent"  
    android:layout_height="match_parent"  
    android:background="@drawable/background" />
```

res/layout/activity_main.xml

Tile mode

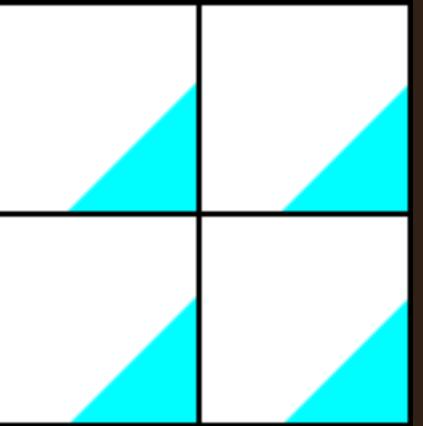
clamp

Replicates the edge color



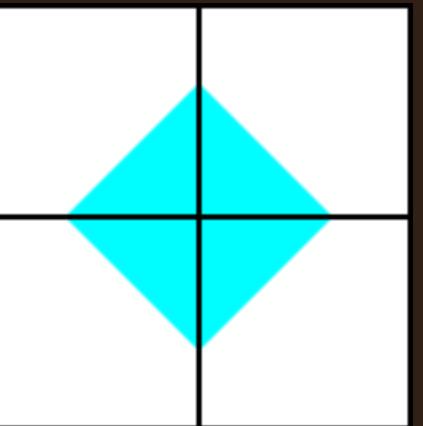
repeat

Repeats the bitmap in both direction



mirror

Repeats with alternating mirror images



GridView



Auto fit columns

```
<GridView  
    android:layout_width="match_parent"  
    android:layout_height="match_parent"  
    android:numColumns="auto_fit"  
    android:columnWidth="@dimen/workbook_column_width"  
    android:padding="@dimen/workbook_padding"  
    android:horizontalSpacing="@dimen/workbook_spacing"  
    android:verticalSpacing="@dimen/workbook_spacing"  
    android:scrollbarStyle="outsideOverlay" />
```

Auto fit columns

```
<GridView  
    android:layout_width="match_parent"  
    android:layout_height="match_parent"  
    android:numColumns="auto_fit"  
    android:columnWidth="@dimen/workbook_column_width"  
    android:padding="@dimen/workbook_padding"  
    android:horizontalSpacing="@dimen/workbook_spacing"  
    android:verticalSpacing="@dimen/workbook_spacing"  
    android:scrollbarStyle="outsideOverlay" />
```

Auto fit columns

```
<GridView  
    android:layout_width="match_parent"  
    android:layout_height="match_parent"  
    android:numColumns="auto_fit"  
    android:columnWidth="@dimen/workbook_column_width"  
    android:padding="@dimen/workbook_padding"  
    android:horizontalSpacing="@dimen/workbook_spacing"  
    android:verticalSpacing="@dimen/workbook_spacing"  
    android:scrollbarStyle="outsideOverlay" />
```

values/dimen.xml

```
<resources>
    <dimen name="workbook_padding">8dp</dimen>
    <dimen name="workbook_spacing">12dp</dimen>
    <dimen name="workbook_column_width">120dp</dimen>
<resources>
```

res/values/dimens.xml

values-large/dimen.xml

```
<resources>
    <dimen name="workbook_padding">32dp</dimen>
    <dimen name="workbook_spacing">18dp</dimen>
    <dimen name="workbook_column_width">180dp</dimen>
<resources>
```

res/values-large/dimens.xml

7-inch

Monkey's kindle 1 4:26

Settings Wi-Fi Battery

Introduction

Chinese characters are written in a specific order to give them structure. Follow the stroke numbers to learn the order.

人 口 手

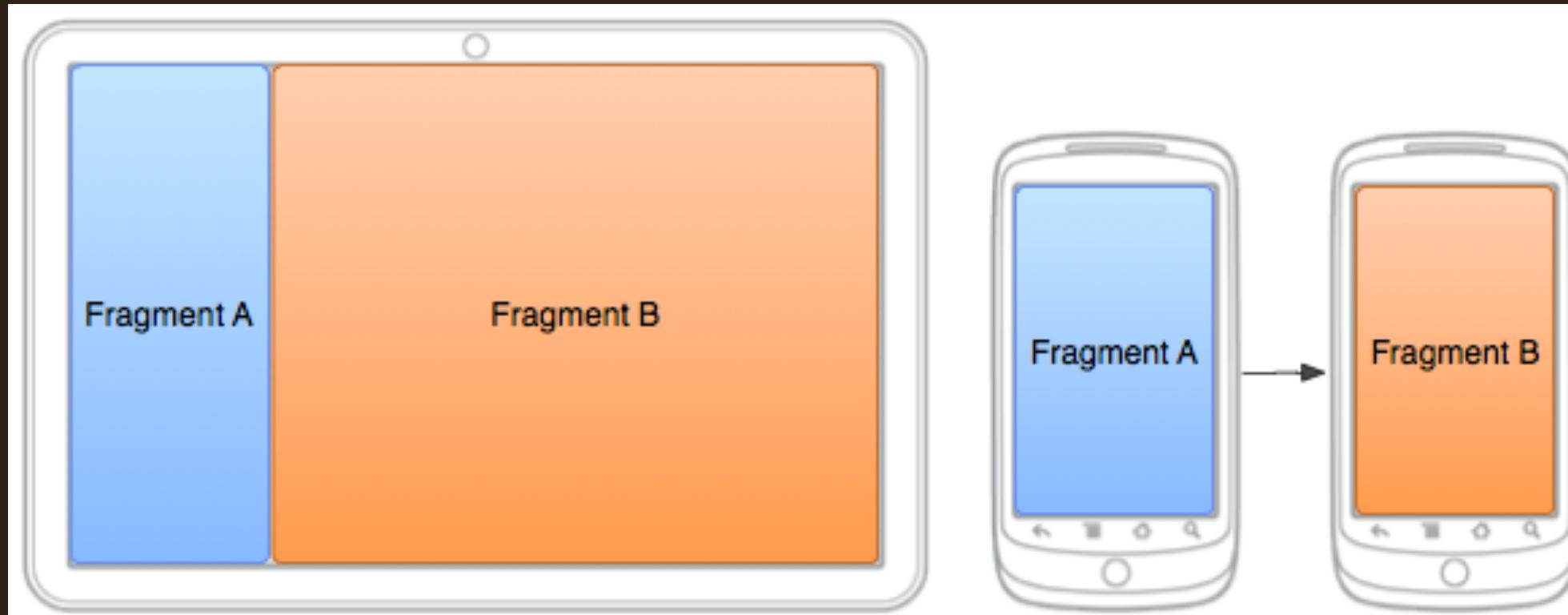
工 木 水

Home Back Search

Phone



Fragments



github.com/codepath/android-master-detail-demo

Progressive

Resource Folders

<http://developer.android.com/guide/topics/resources/providing-resources.html>

Type	Variation
layout	Language & Region: en, fr, fr-rCA
values	Screen size: small, large, xlarge, sw320dp, h720dp
drawable	Screen orientation: port, land
menu	Screen density: ldpi, mdpi, hdpi, xhdpi, nodpi, tvdpi
	Platform version: v4, v11, v14, v21
	UI mode: car, desk, television, appliance

Resource Folders

<http://developer.android.com/guide/topics/resources/providing-resources.html>

Type	Variation
layout values drawable menu	Language & Region: en, fr, fr-rCA Screen size: small, large, xlarge, sw320dp, h720dp Screen orientation: port, land Screen density: ldpi, mdpi, hdpi, xhdpi, nodpi, tvdpi Platform version: v4, v11, v14, v21 UI mode: car, desk, television, appliance

Ripples

RippleDrawable

Added in API level 21

```
<!-- A red ripple masked against an opaque rectangle. -->
<ripple android:color="#ffff0000">
    <item android:id="@+id/mask"
        android:drawable="@color/white" />
</ripple>
```

Ripples

RippleDrawable

Added in API level 21

```
<!-- A red ripple masked against an opaque rectangle. -->
<ripple android:color="#ffff0000">
    <item android:id="@+id/mask"
        android:drawable="@color/white" />
</ripple>
```

drawable-v21

```
<ripple android:color="@color/colorAccent">
    <item>
        <shape android:shape="rectangle">
            <solid android:color="@color/colorPrimary" />
        </shape>
    </item>
</ripple>
```

res/**drawable-v21**/background.xml

drawable as default

```
<selector>
    <item android:state_pressed="true"
        android:drawable="@color/colorAccent" />
    <item android:drawable="@color/colorPrimary" />
</selector>
```

res/drawable/background.xml

android:background

```
<TextView  
    android:layout_width="match_parent"  
    android:layout_height="match_parent"  
    android:background="drawable/background"  
    android:text="@string/click_me" />
```

Progressive



API level

android.com/reference/android/app/ActivityManager.html

public int getMemoryClass () Added in API level 5

Return the approximate per-application memory class of the current device. This gives you an idea of how hard a memory limit you should impose on your application to let the overall system work best. The returned value is in megabytes; the baseline Android memory class is 16 (which happens to be the Java heap limit of those devices); some device with more memory may return 24 or even higher numbers.

Detect API level

android.os.Build.VERSION.SDK_INT

Progressive

```
// Default value  
int heapSize = 16;  
  
// Detect OS version  
int sdk = Integer.parseInt(Build.VERSION.SDK_INT);  
  
// Progressive enhancement for newer devices  
if (sdk >= Build.VERSION_CODES_ECLAIR) {  
    heapSize = getHeapSize(this);  
}
```

getHeapSize

```
public int getHeapSize() {  
    ActivityManager activityManager = (ActivityManager)  
        getSystemService(Context.ACTIVITY_SERVICE);  
    return activityManager.getMemoryClass();  
}
```

Support Library

Ship new functionality with your app!

Support Library

Fragment

AppCompat

ViewPager

SlidingPaneLayout

DrawerLayout

RecyclerView

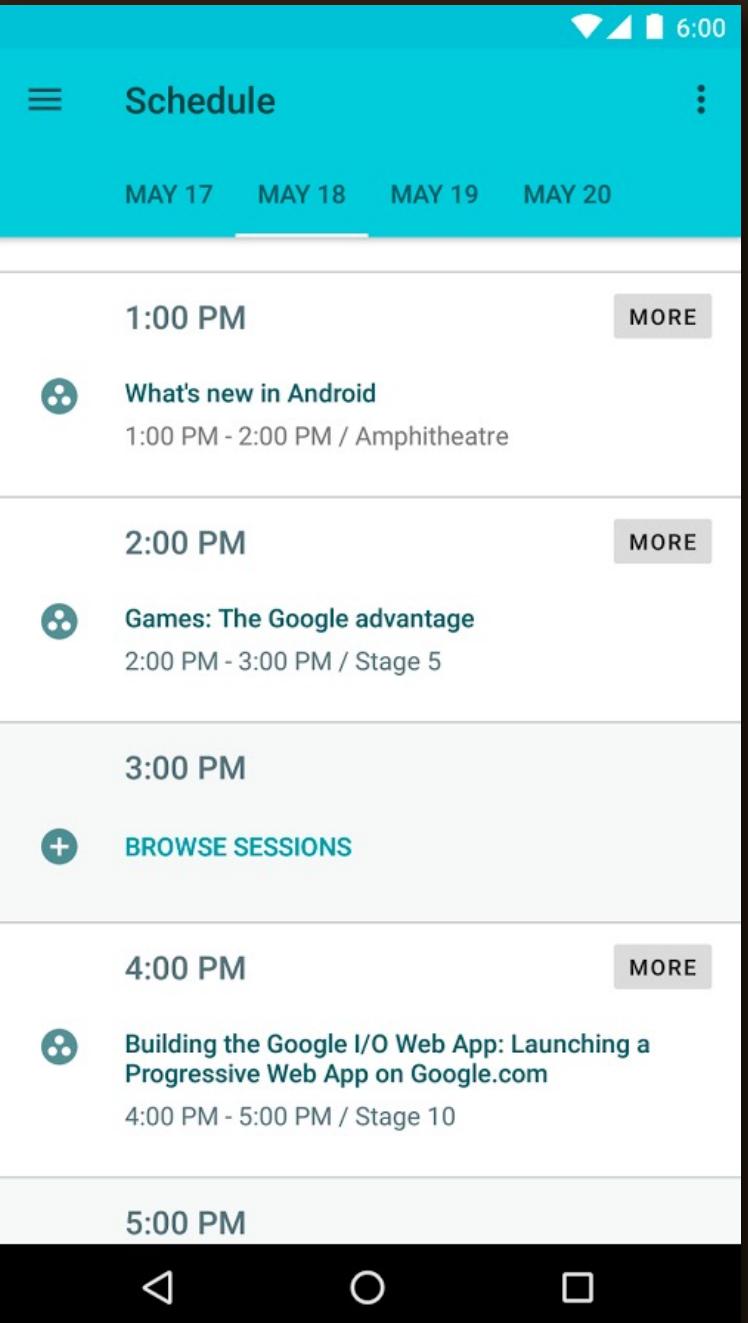
Loader

VectorDrawable

LruCache

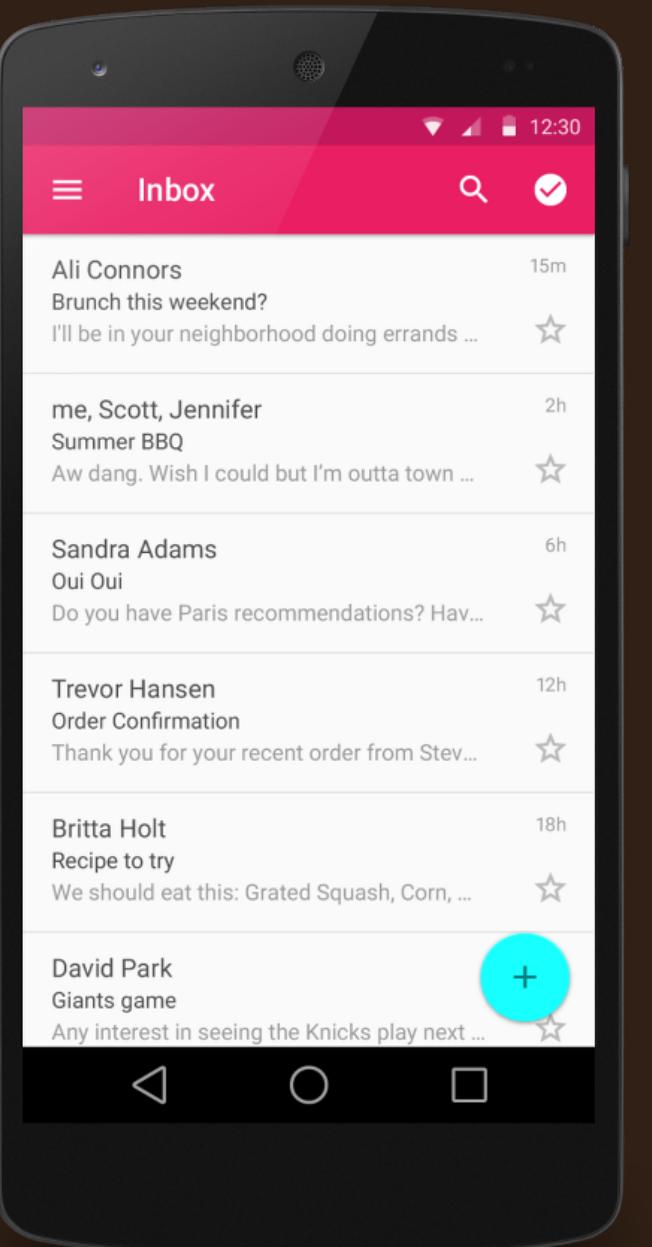
TaskStackBuilder

AppCompat



Material Design

Support library only



RecyclerView

Summary

Summary

Responsive layout

Declarative layout

Density-independent pixels (dp)

Resource folders

XML drawables

Progressive functionality

Resource folders

Version check

Support library



Thank you!

Learn more

developer.android.com/training/multiscreen

is.gd/FluidAndroidLayouts

is.gd/BeautifulAndroid

pluralsight.com/authors/chiuki-chan

Stay in touch

tinyletter.com/sqisland

blog.sqisland.com

[@chiuki](http://twitter.com/chiuki)

@chiuki



Slides

bit.ly/muland



@chiuki

360|AnDev Core Track

- Activities in the Wild: Exploring the Activity Lifecycle
- Fragments: What are they good for
- Async Task, Threads, Pools and Executors
- Measure, Layout, Draw, Repeat: Custom Views and ViewGroups
- A Practical Guide to Material Design implementation for Android Developers
- Libraries I wish I knew about when I started
- Intro to iOS for Android developers

TICKETS ON SALE NOW!



Discount Code:
DroidCommunity