Android Programming Lecture 2

9/7/2011

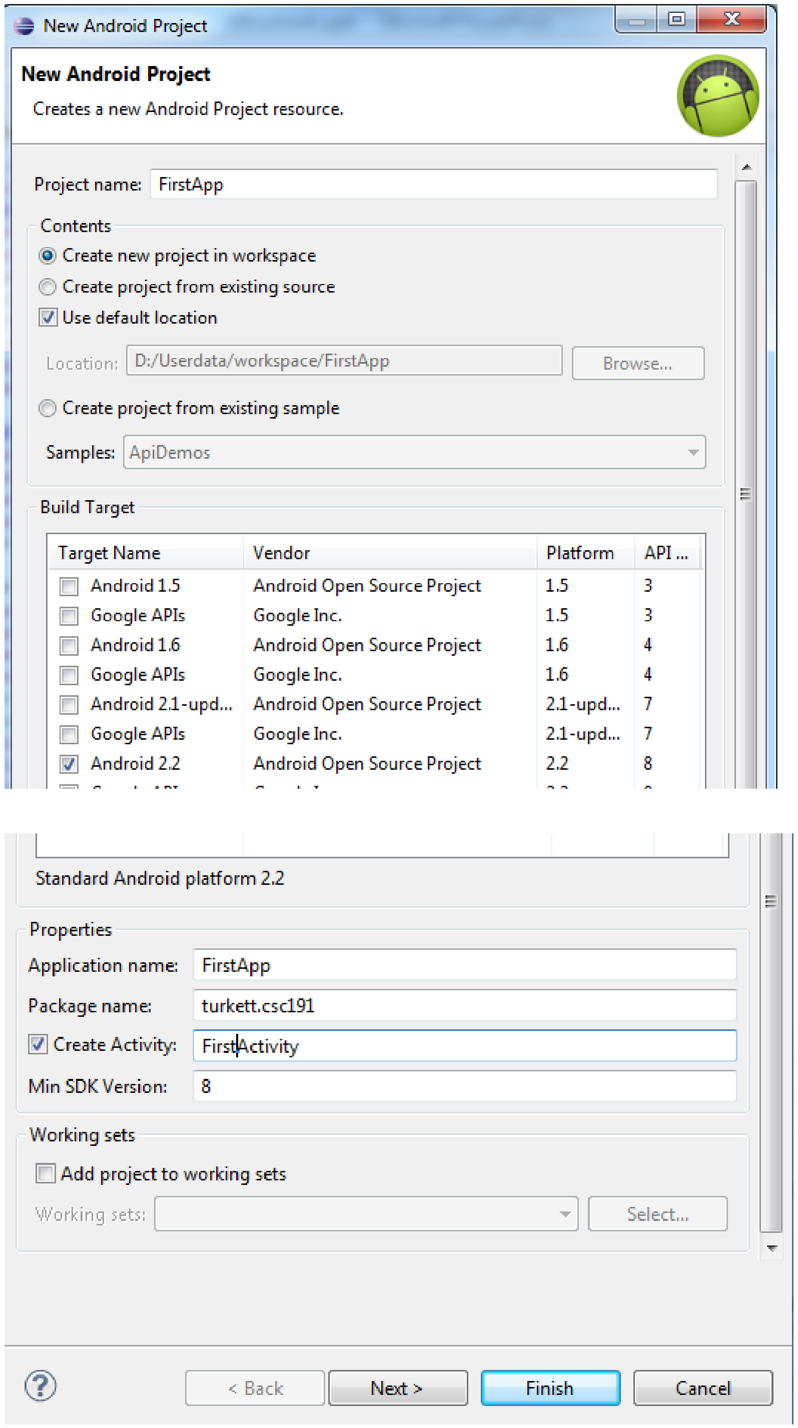
Creating a first app

1. Create a new Android project (a collection of source code and resources for the app) from the Eclipse file menu

2. Choose a project name (can be anything)

3. Application specifics:
   1. Target platform
   2. Application name
   3. Package name
   4. Initial activity to launch
   5. Absolute minimum platform

4. Finish



New Android Project

New Android Project

Creates a new Android Project resource.

Project name: FirstApp

Contents

- [✓] Create new project in workspace
- Create project from existing source
- [✓] Use default location

Location: D:/Userdata/workspace/FirstApp

- Create project from existing sample

Samples: ApiDemos

Build Target

	Target Name	Vendor	Platform	API ...
☐	Android 1.5	Android Open Source Project	1.5	3
☐	Google APIs	Google Inc.	1.5	3
☐	Android 1.6	Android Open Source Project	1.6	4
☐	Google APIs	Google Inc.	1.6	4
☐	Android 2.1-upd...	Android Open Source Project	2.1-upd...	7
☐	Google APIs	Google Inc.	2.1-upd...	7
✓	Android 2.2	Android Open Source Project	2.2	8

Standard Android platform 2.2

Properties

Application name: FirstApp

Package name: turkett.csc191

[✓] Create Activity: FirstActivity

Min SDK Version: 8

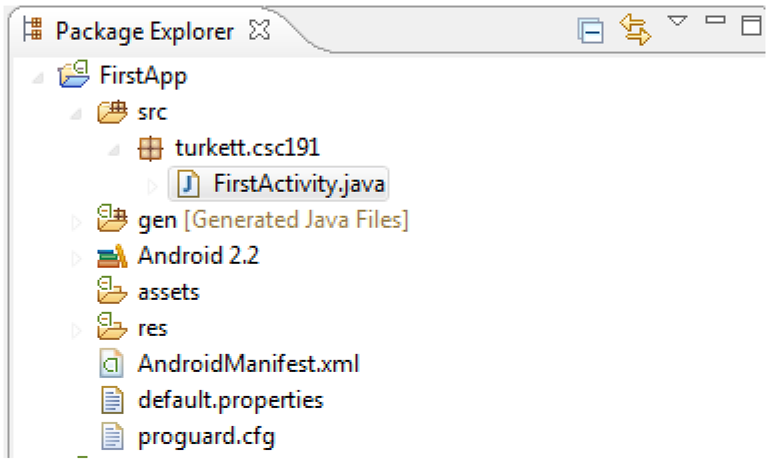
Working sets

☐ Add project to working sets

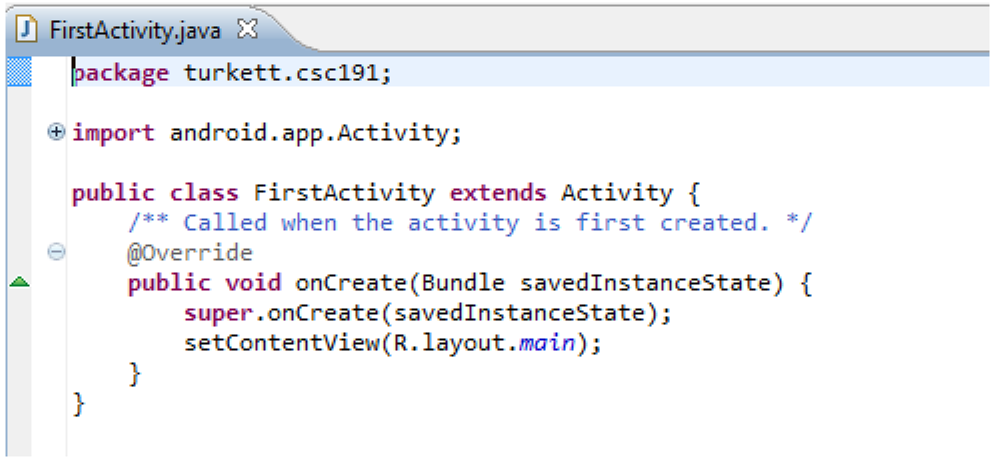
Working sets:

Creating a first app

1. Expand the project, src folder, and your chosen package



2. Choosing your Activity file will reveal a default implementation of the *onCreate* function
   1. Calls the onCreate of the Activity parent class
   2. Sets the content of this screen to be an XML specified layout *(we'll come back to this)*


FirstActivity.java

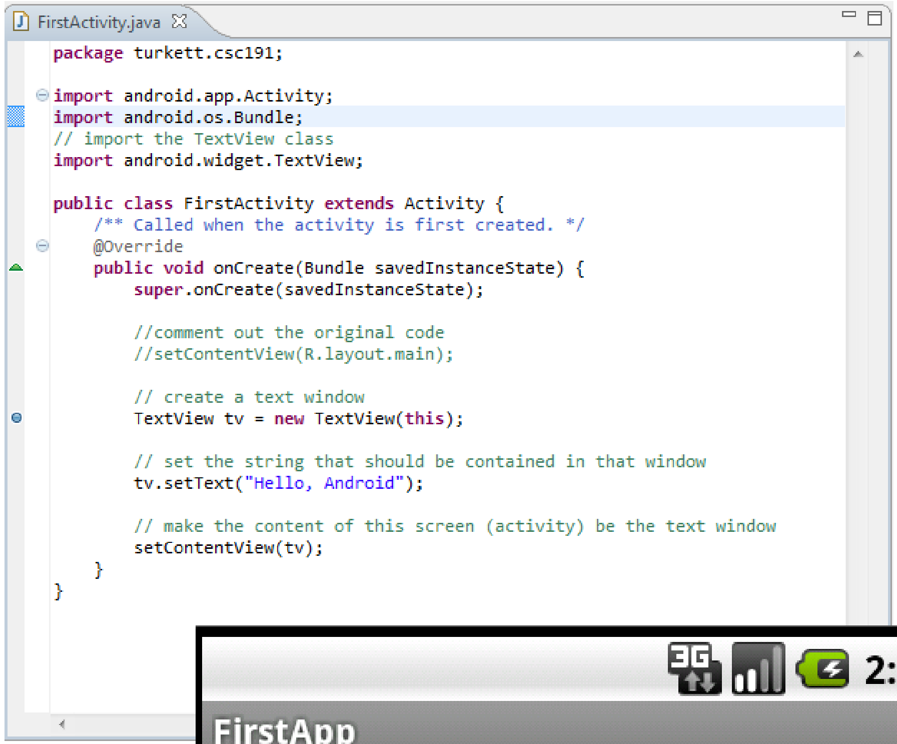
```java
package turkett.csc191;

import android.app.Activity;

public class FirstActivity extends Activity {
    /** Called when the activity is first created. */
    @Override
    public void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.main);
    }
}
```

Creating a first app

3. Replace pre-generated code with your own TextView code

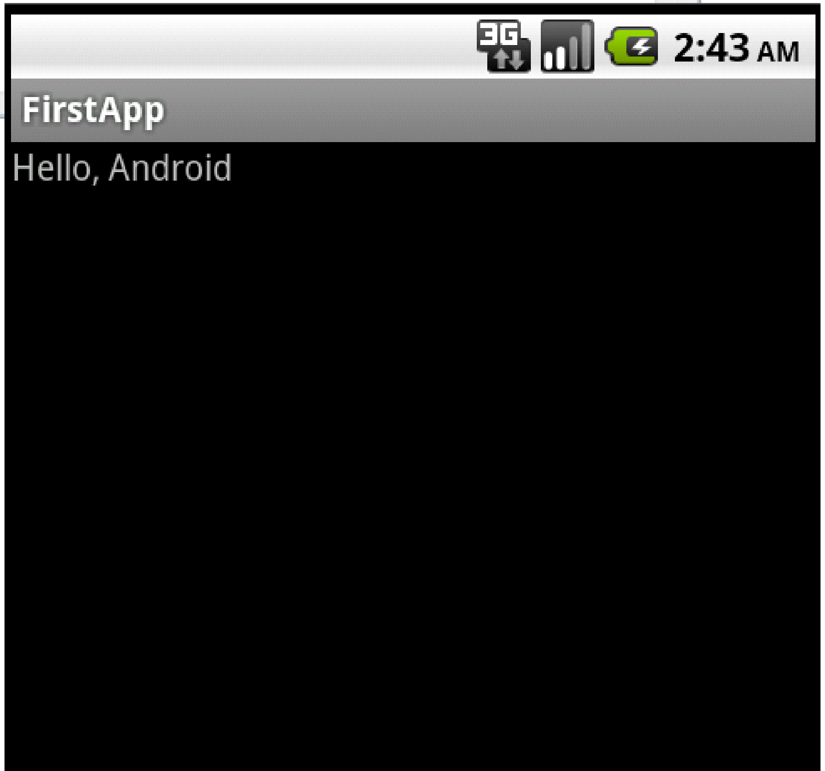
4. Run the app from Eclipse

5. Emulator should start, and open your app



FirstActivity.java

```java
package turkett.csc191;

import android.app.Activity;
import android.os.Bundle;
// import the TextView class
import android.widget.TextView;

public class FirstActivity extends Activity {
    /** Called when the activity is first created. */
    @Override
    public void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);

        //comment out the original code
        //setContentView(R.layout.main);

        // create a text window
        TextView tv = new TextView(this);

        // set the string that should be contained in that window
        tv.setText("Hello, Android");

        // make the content of this screen (activity) be the text window
        setContentView(tv);
    }
}
```



2:43 AM

FirstApp

```
Hello, Android
```

Exercise Discussion

- Discussion of exercise from last week

Debugging Android

- Traditional System.out.println is not available in the Android system
- Don’t want to debug through the app user interface:
  - Errors crash and close app

- Instead use Logging mechanisms

```
Log.v(String tag,
 String message);
```

Common:

tag ➔ app name

message ➔ debug message.

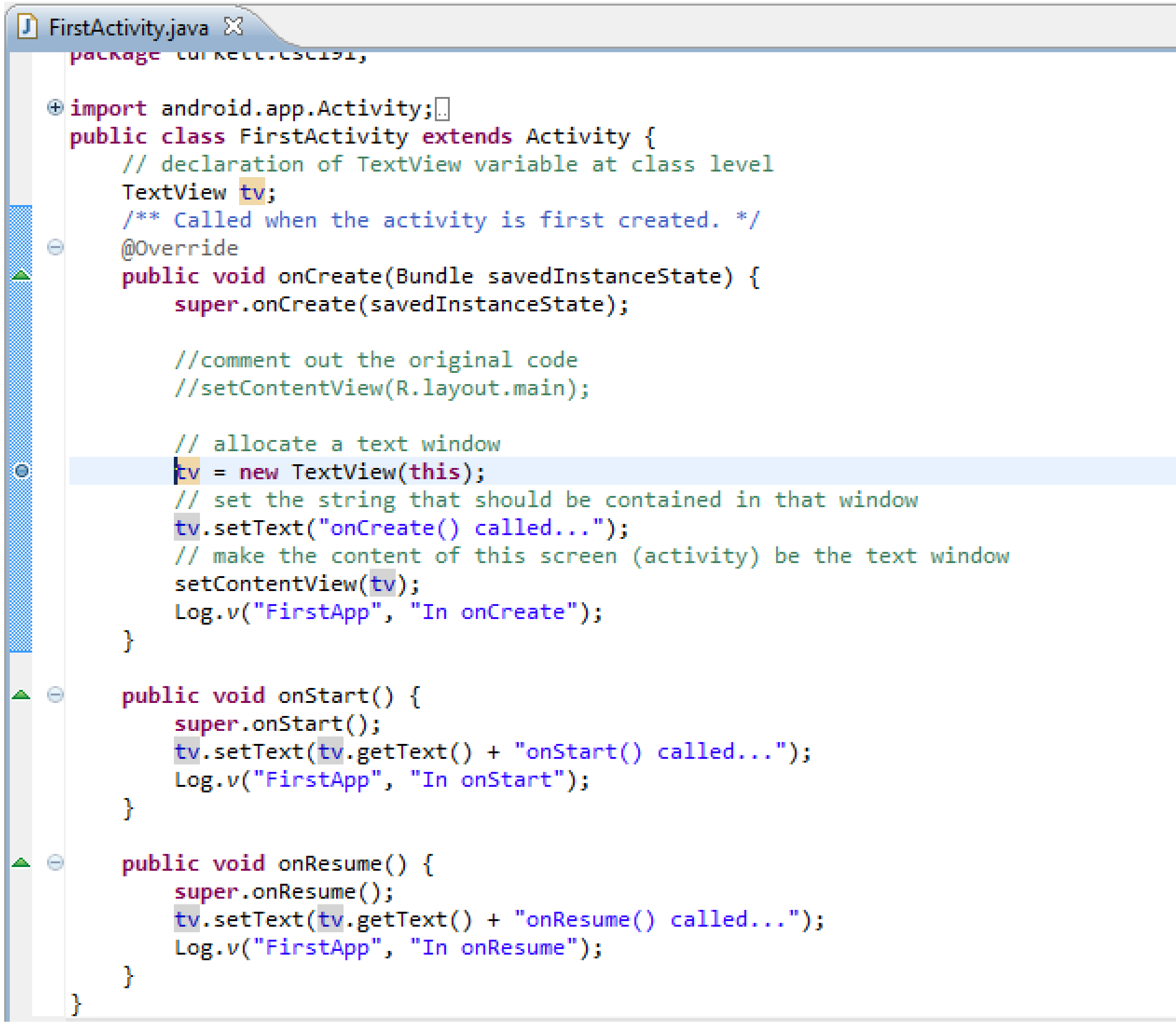
Requires importing

```
      android.util.Log;
```

Debugging Android Example

Log messages added to three startup functions

tag: `FirstApp`

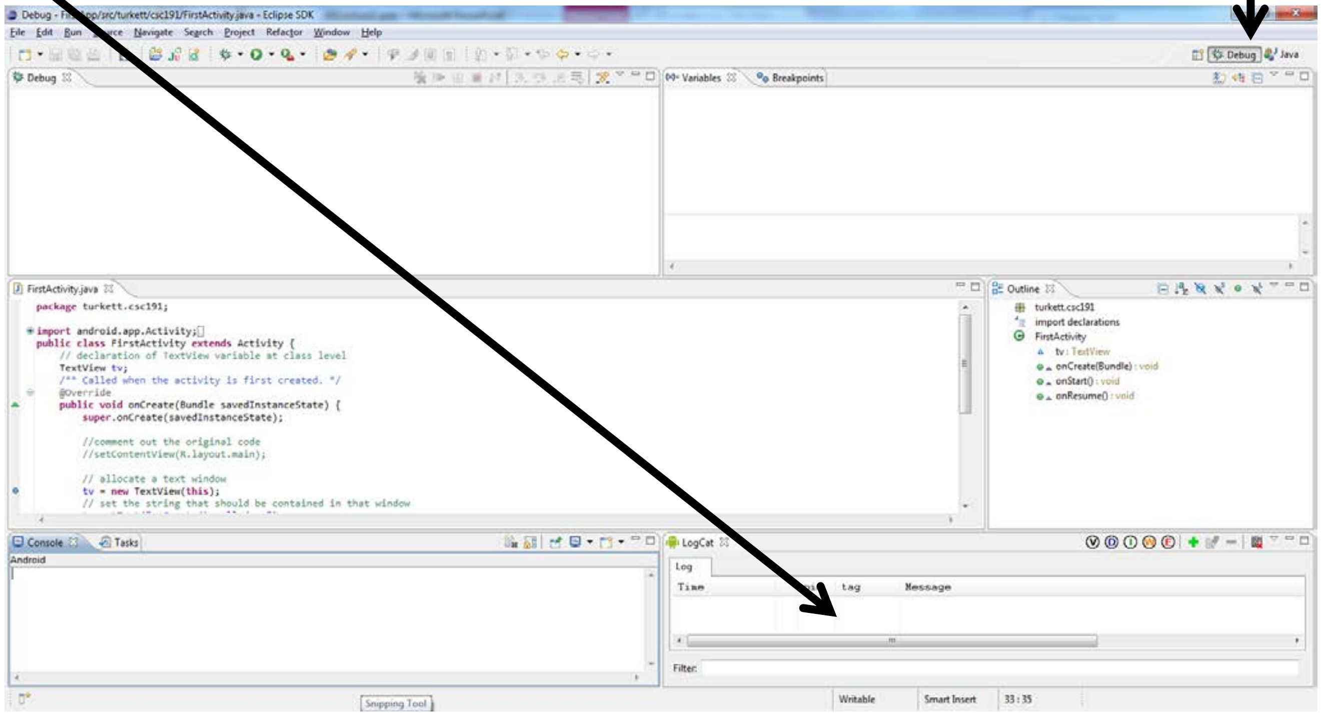


FirstActivity.java

```
import android.app.Activity;
public class FirstActivity extends Activity {
    // declaration of TextView variable at class level
    TextView tv;
    /** Called when the activity is first created. */
    @Override
    public void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);

        //comment out the original code
        //setContentView(R.layout.main);

        // allocate a text window
        tv = new TextView(this);
        // set the string that should be contained in that window
        tv.setText("onCreate() called...");
        // make the content of this screen (activity) be the text window
        setContentView(tv);
        Log.v("FirstApp", "In onCreate");
    }

    public void onStart() {
        super.onStart();
        tv.setText(tv.getText() + "onStart() called...");
        Log.v("FirstApp", "In onStart");
    }

    public void onResume() {
        super.onResume();
        tv.setText(tv.getText() + "onResume() called...");
        Log.v("FirstApp", "In onResume");
    }
}
```

Debugging Android

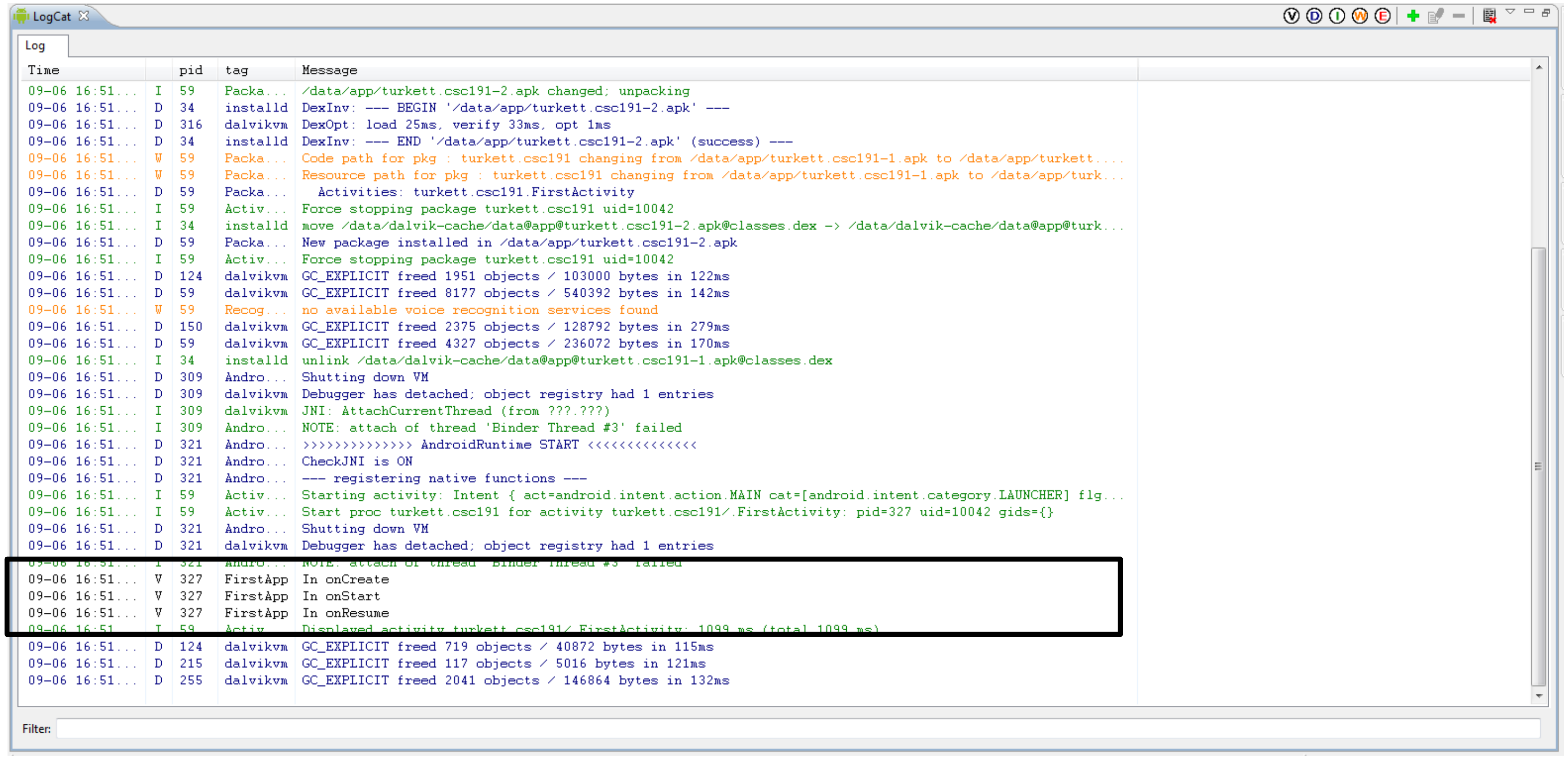
- Log messages appear in the LogCat component of the Eclipse interface
  - Change to Debug mode
  - LogCat will be one of windows in Debug mode



```
package turkett.csc191;

import android.app.Activity;
public class FirstActivity extends Activity {
    // declaration of TextView variable at class level
    TextView tv;
    /** Called when the activity is first created. */
    @Override
    public void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);

        //comment out the original code
        //setContentView(R.layout.main);

        // allocate a text window
        tv = new TextView(this);
        // set the string that should be contained in that window
```

Debugging Android

LogCat will be full of messages from the device



LogCat Log

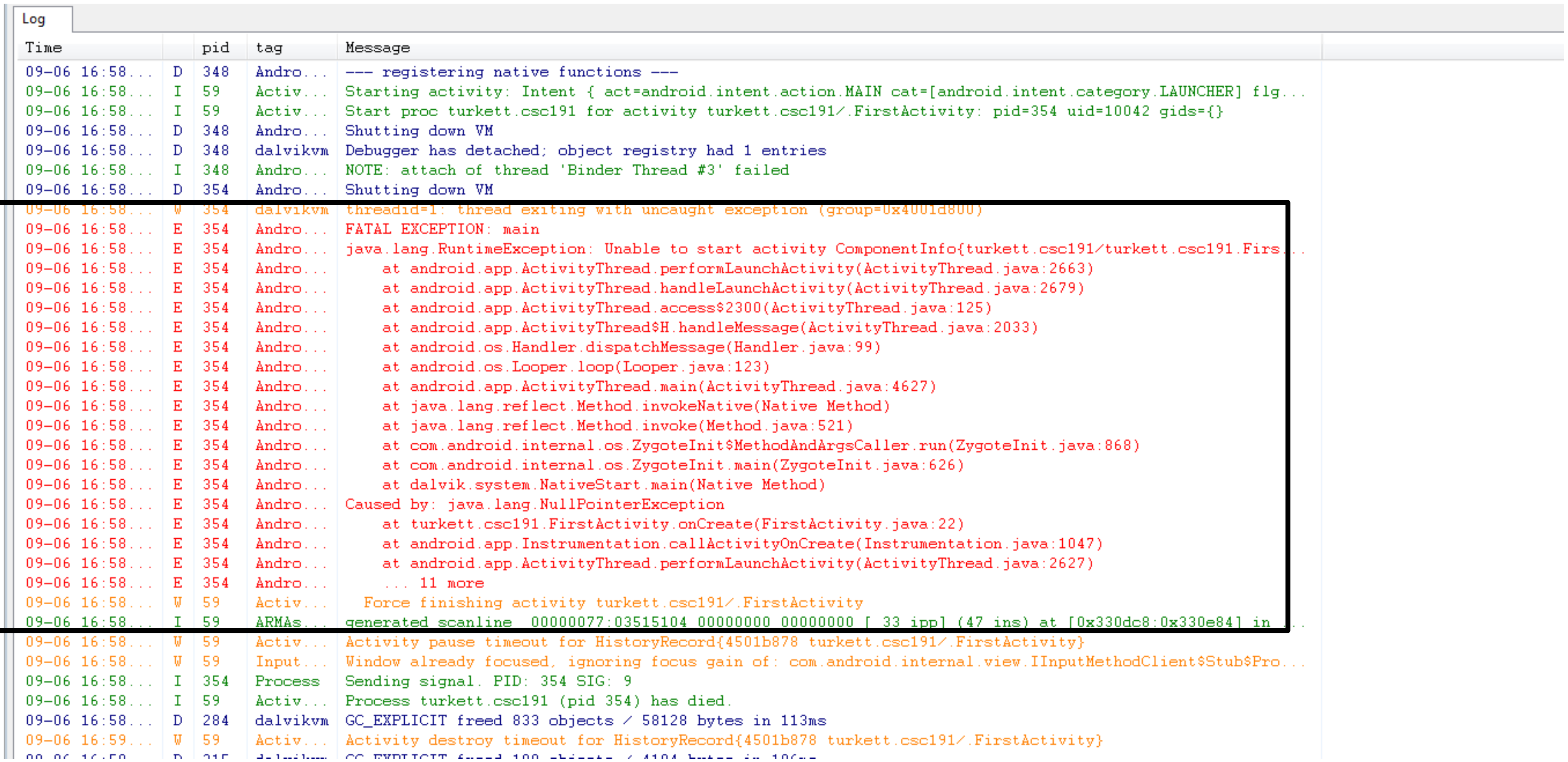
```
Time            pid  tag       Message
09-06 16:51...  I  59   Packa...  /data/app/turkett.csc191-2.apk changed; unpacking
09-06 16:51...  D  34   installd  DexInv: --- BEGIN '/data/app/turkett.csc191-2.apk' ---
09-06 16:51...  D  316  dalvikvm  DexOpt: load 25ms, verify 33ms, opt 1ms
09-06 16:51...  D  34   installd  DexInv: --- END '/data/app/turkett.csc191-2.apk' (success) ---
09-06 16:51...  W  59   Packa...  Code path for pkg : turkett.csc191 changing from /data/app/turkett.csc191-1.apk to /data/app/turkett....
09-06 16:51...  W  59   Packa...  Resource path for pkg : turkett.csc191 changing from /data/app/turkett.csc191-1.apk to /data/app/turk...
09-06 16:51...  D  59   Packa...    Activities: turkett.csc191.FirstActivity
09-06 16:51...  I  59   Activ...  Force stopping package turkett.csc191 uid=10042
09-06 16:51...  I  34   installd  move /data/dalvik-cache/data@app@turkett.csc191-2.apk@classes.dex -> /data/dalvik-cache/data@app@turk...
09-06 16:51...  D  59   Packa...  New package installed in /data/app/turkett.csc191-2.apk
09-06 16:51...  I  59   Activ...  Force stopping package turkett.csc191 uid=10042
09-06 16:51...  D  124  dalvikvm  GC_EXPLICIT freed 1951 objects / 103000 bytes in 122ms
09-06 16:51...  D  59   dalvikvm  GC_EXPLICIT freed 8177 objects / 540392 bytes in 142ms
09-06 16:51...  W  59   Recog...  no available voice recognition services found
09-06 16:51...  D  150  dalvikvm  GC_EXPLICIT freed 2375 objects / 128792 bytes in 279ms
09-06 16:51...  D  59   dalvikvm  GC_EXPLICIT freed 4327 objects / 236072 bytes in 170ms
09-06 16:51...  I  34   installd  unlink /data/dalvik-cache/data@app@turkett.csc191-1.apk@classes.dex
09-06 16:51...  D  309  Andro...  Shutting down VM
09-06 16:51...  D  309  dalvikvm  Debugger has detached; object registry had 1 entries
09-06 16:51...  I  309  dalvikvm  JNI: AttachCurrentThread (from ???.???)
09-06 16:51...  I  309  Andro...  NOTE: attach of thread 'Binder Thread #3' failed
09-06 16:51...  D  321  Andro...  >>>>>>>>>>>>>> AndroidRuntime START <<<<<<<<<<<<<<
09-06 16:51...  D  321  Andro...  CheckJNI is ON
09-06 16:51...  D  321  Andro...  --- registering native functions ---
09-06 16:51...  I  59   Activ...  Starting activity: Intent { act=android.intent.action.MAIN cat=[android.intent.category.LAUNCHER] flg...
09-06 16:51...  I  59   Activ...  Start proc turkett.csc191 for activity turkett.csc191/.FirstActivity: pid=327 uid=10042 gids={}
09-06 16:51...  D  321  Andro...  Shutting down VM
09-06 16:51...  D  321  dalvikvm  Debugger has detached; object registry had 1 entries
09-06 16:51...  I  321  Andro...  NOTE: attach of thread 'Binder Thread #3' failed
09-06 16:51...  V  327  FirstApp  In onCreate
09-06 16:51...  V  327  FirstApp  In onStart
09-06 16:51...  V  327  FirstApp  In onResume
09-06 16:51...  I  59   Activ...  Displayed activity turkett.csc191/.FirstActivity: 1099 ms (total 1099 ms)
09-06 16:51...  D  124  dalvikvm  GC_EXPLICIT freed 719 objects / 40872 bytes in 115ms
09-06 16:51...  D  215  dalvikvm  GC_EXPLICIT freed 117 objects / 5016 bytes in 121ms
09-06 16:51...  D  255  dalvikvm  GC_EXPLICIT freed 2041 objects / 146864 bytes in 132ms
```

Filter:

Can setup up filters (using the + button) on your tag

Debugging Android

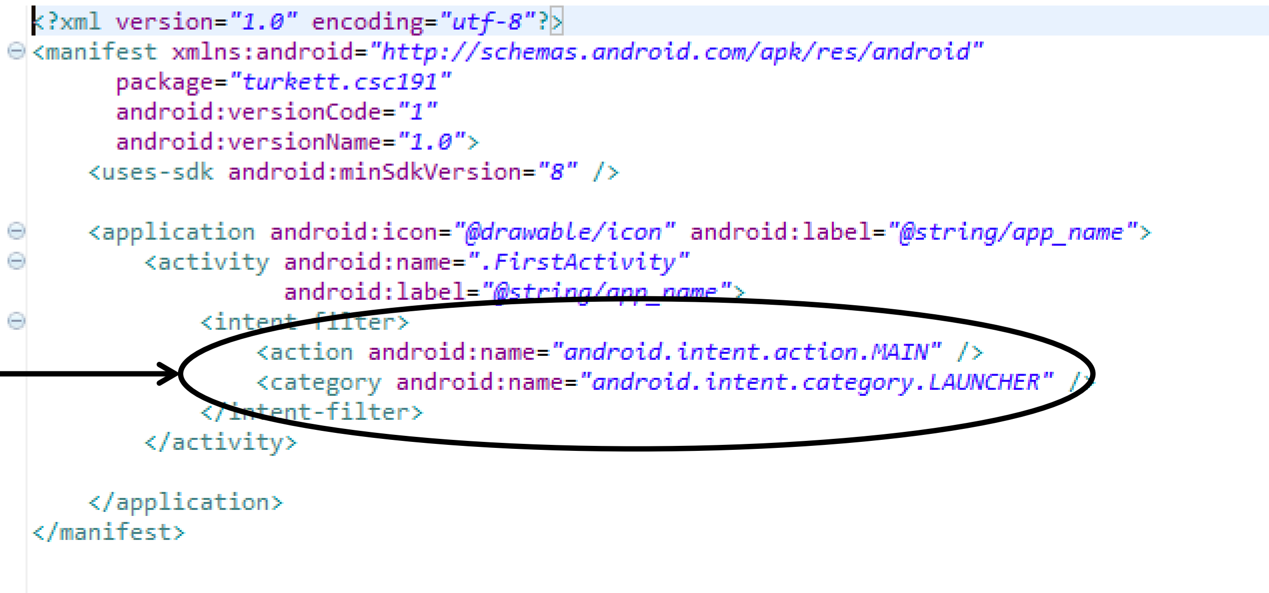
Crash stack traces show up in red



```
Log
Time                pid  tag       Message
09-06 16:58...  D  348  Andro...  --- registering native functions ---
09-06 16:58...  I  59   Activ...  Starting activity: Intent { act=android.intent.action.MAIN cat=[android.intent.category.LAUNCHER] flg...
09-06 16:58...  I  59   Activ...  Start proc turkett.csc191 for activity turkett.csc191/.FirstActivity: pid=354 uid=10042 gids={}
09-06 16:58...  D  348  Andro...  Shutting down VM
09-06 16:58...  D  348  dalvikvm  Debugger has detached; object registry had 1 entries
09-06 16:58...  I  348  Andro...  NOTE: attach of thread 'Binder Thread #3' failed
09-06 16:58...  D  354  Andro...  Shutting down VM
09-06 16:58...  W  354  dalvikvm  threadid=1: thread exiting with uncaught exception (group=0x4001d800)
09-06 16:58...  E  354  Andro...  FATAL EXCEPTION: main
09-06 16:58...  E  354  Andro...  java.lang.RuntimeException: Unable to start activity ComponentInfo{turkett.csc191/turkett.csc191.Firs...
09-06 16:58...  E  354  Andro...      at android.app.ActivityThread.performLaunchActivity(ActivityThread.java:2663)
09-06 16:58...  E  354  Andro...      at android.app.ActivityThread.handleLaunchActivity(ActivityThread.java:2679)
09-06 16:58...  E  354  Andro...      at android.app.ActivityThread.access$2300(ActivityThread.java:125)
09-06 16:58...  E  354  Andro...      at android.app.ActivityThread$H.handleMessage(ActivityThread.java:2033)
09-06 16:58...  E  354  Andro...      at android.os.Handler.dispatchMessage(Handler.java:99)
09-06 16:58...  E  354  Andro...      at android.os.Looper.loop(Looper.java:123)
09-06 16:58...  E  354  Andro...      at android.app.ActivityThread.main(ActivityThread.java:4627)
09-06 16:58...  E  354  Andro...      at java.lang.reflect.Method.invokeNative(Native Method)
09-06 16:58...  E  354  Andro...      at java.lang.reflect.Method.invoke(Method.java:521)
09-06 16:58...  E  354  Andro...      at com.android.internal.os.ZygoteInit$MethodAndArgsCaller.run(ZygoteInit.java:868)
09-06 16:58...  E  354  Andro...      at com.android.internal.os.ZygoteInit.main(ZygoteInit.java:626)
09-06 16:58...  E  354  Andro...      at dalvik.system.NativeStart.main(Native Method)
09-06 16:58...  E  354  Andro...  Caused by: java.lang.NullPointerException
09-06 16:58...  E  354  Andro...      at turkett.csc191.FirstActivity.onCreate(FirstActivity.java:22)
09-06 16:58...  E  354  Andro...      at android.app.Instrumentation.callActivityOnCreate(Instrumentation.java:1047)
09-06 16:58...  E  354  Andro...      at android.app.ActivityThread.performLaunchActivity(ActivityThread.java:2627)
09-06 16:58...  E  354  Andro...      ... 11 more
09-06 16:58...  W  59   Activ...    Force finishing activity turkett.csc191/.FirstActivity
09-06 16:58...  I  59   ARMAs...  generated scanline__00000077:03515104_00000000_00000000 [ 33 ipp] (47 ins) at [0x330dc8:0x330e84] in ...
09-06 16:58...  W  59   Activ...  Activity pause timeout for HistoryRecord{4501b878 turkett.csc191/.FirstActivity}
09-06 16:58...  W  59   Input...  Window already focused, ignoring focus gain of: com.android.internal.view.IInputMethodClient$Stub$Pro...
09-06 16:58...  I  354  Process   Sending signal. PID: 354 SIG: 9
09-06 16:58...  I  59   Activ...  Process turkett.csc191 (pid 354) has died.
09-06 16:58...  D  284  dalvikvm  GC_EXPLICIT freed 833 objects / 58128 bytes in 113ms
09-06 16:59...  W  59   Activ...  Activity destroy timeout for HistoryRecord{4501b878 turkett.csc191/.FirstActivity}
09-06 16:59...  D  315  dalvikvm  GC_EXPLICIT freed 100 objects / 4104 bytes in 106ms
```

Applications and Activities

- How does the Application know the initial Activity to call?
  - Stored in application manifest: AndroidManifest.xml
    - Managed by Eclipse for us

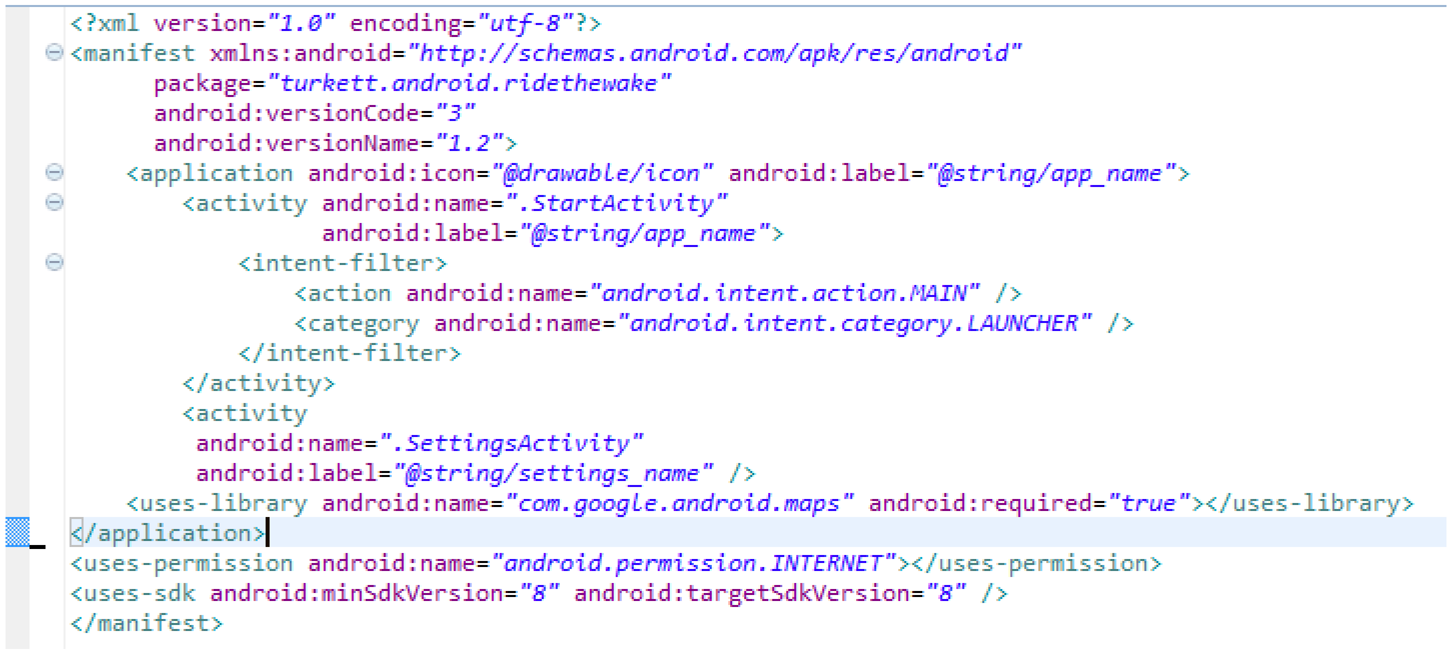


```
<?xml version="1.0" encoding="utf-8"?>
<manifest xmlns:android="http://schemas.android.com/apk/res/android"
      package="turkett.csc191"
      android:versionCode="1"
      android:versionName="1.0">
    <uses-sdk android:minSdkVersion="8" />

    <application android:icon="@drawable/icon" android:label="@string/app_name">
        <activity android:name=".FirstActivity"
                  android:label="@string/app_name">
            <intent-filter>
                <action android:name="android.intent.action.MAIN" />
                <category android:name="android.intent.category.LAUNCHER" />
            </intent-filter>
        </activity>

    </application>
</manifest>
```

Indication that the activity is the first target

Applications and Activities

- A manifest for an Application with two Activity components

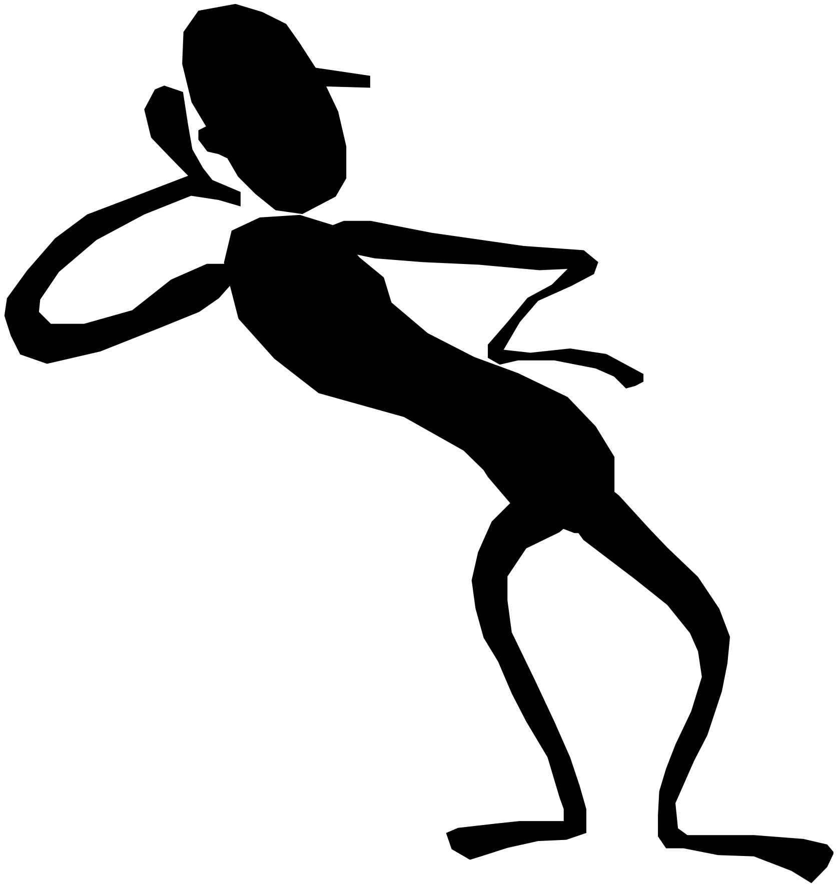


```xml
<?xml version="1.0" encoding="utf-8"?>
<manifest xmlns:android="http://schemas.android.com/apk/res/android"
      package="turkett.android.ridethewake"
      android:versionCode="3"
      android:versionName="1.2">
    <application android:icon="@drawable/icon" android:label="@string/app_name">
        <activity android:name=".StartActivity"
                  android:label="@string/app_name">
            <intent-filter>
                <action android:name="android.intent.action.MAIN" />
                <category android:name="android.intent.category.LAUNCHER" />
            </intent-filter>
        </activity>
        <activity
         android:name=".SettingsActivity"
         android:label="@string/settings_name" />
    <uses-library android:name="com.google.android.maps" android:required="true"></uses-library>
</application>
<uses-permission android:name="android.permission.INTERNET"></uses-permission>
<uses-sdk android:minSdkVersion="8" android:targetSdkVersion="8" />
</manifest>
```

Important Java Concepts

- Packages:
  - packages of classes = directories of files
    - Importing in Java
    - Your own
- Inheriting from Activity/super
- Class methods: Log.v(String x, String y)
- Becoming familiar with the Android API
  - http://developer.android.com/reference/packages.html
  - http://developer.android.com/reference/classes.html

Listener Pattern



These *onCreate*, *onPause*, etc. functions are examples of the *Listener* design pattern.

A design pattern (according to Wikipedia):

> "…a **design pattern** is a general reusable solution to a commonly occurring problem within a given context… It is a description or template for how to solve a problem that can be used in many different situations".

Listener Pattern

- Also called an "observer pattern"
- A subject
  - maintains a list of observers interested in state changes of the subject
  - automatically notifies the observers of such changes, often by calling a common method that all such observers implement

Subject

Listener #1

Listener #2

Listener Pattern

- This pattern is very common in Graphical User Interfaces
  - A component of an application may be interested in being notified when a particular software “OK” button has been pressed on the screen
  - … when a physical keyboard key or hardware button has been pressed
  - … when a new GPS location has been received by the device
  - … lots of other examples

- Typically, any class can be a listener if:
  - It subscribes to the updates
  - Implements all necessary methods that the subject may call
    - These will typically be the “onXXX” methods
    - Methods are gathered in a Java *interface*

Important Java Concepts

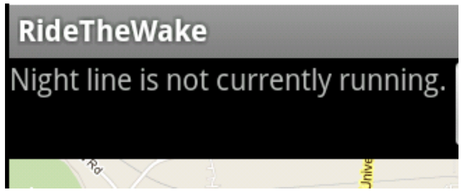
- Interface
  - A Java class specifying functions to be implemented but without an implementation
  - Must implement all functions if decide to implement the interface

Graphical User Interface Components

- Views:
  - Single widgets or controls
  - How the user interacts with your application
- ViewGroups:
  - One or more views combined together
  - Two uses:
    - Layouts: Invisible, control the flow of other widgets
    - Advanced widgets: Visible, implement complex controls

Simple View Items

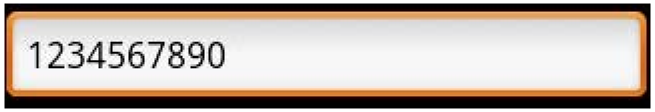
- TextView



RideTheWake

Night line is not currently running.

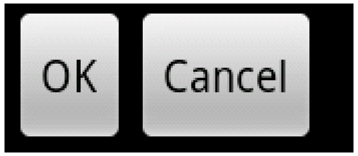
- EditText



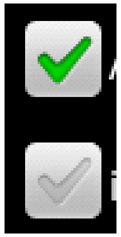
1234567890

Can also be used as a password field

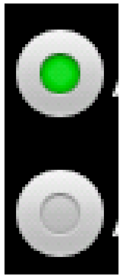
- Button:



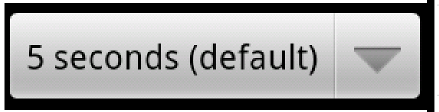
- CheckBox:



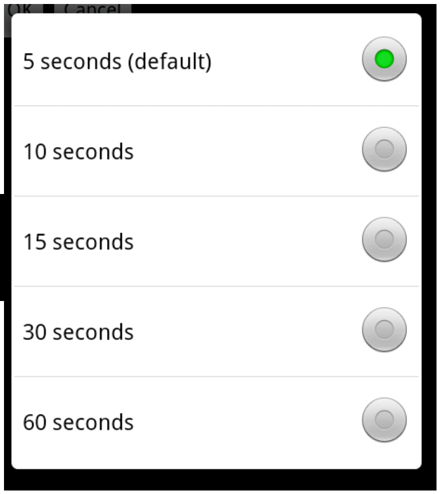
- RadioButton:



- Spinner:



5 seconds (default)

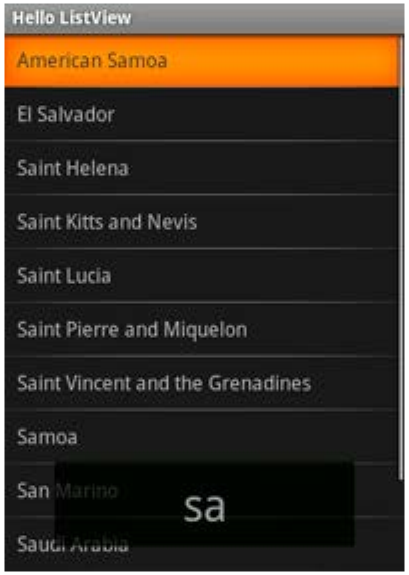


5 seconds (default) ●
10 seconds
15 seconds
30 seconds
60 seconds

More View Items

ListView (ViewGroup)

Vertical scrolling of TextViews

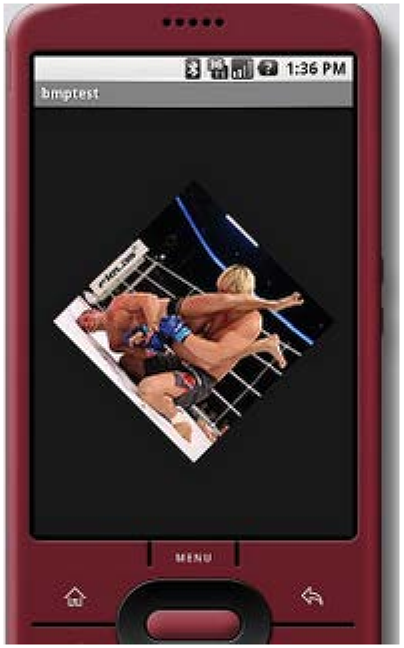


ViewFlipper

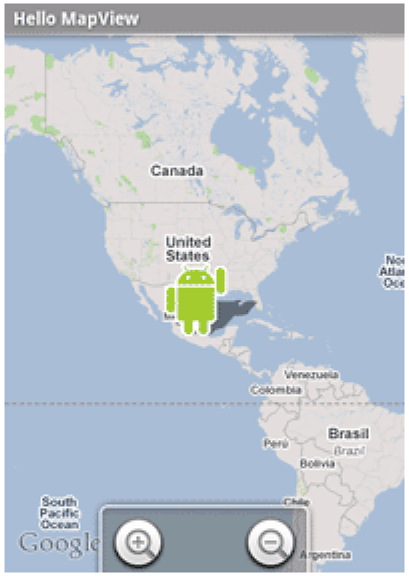
Horizontal scrolling

More View Items

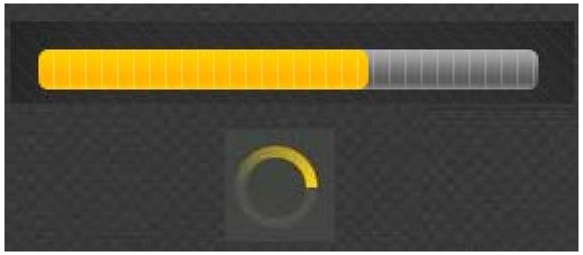
ImageView



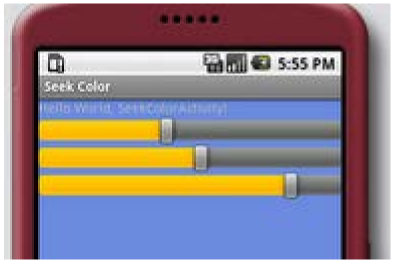
MapView



ProgressBar/Icon

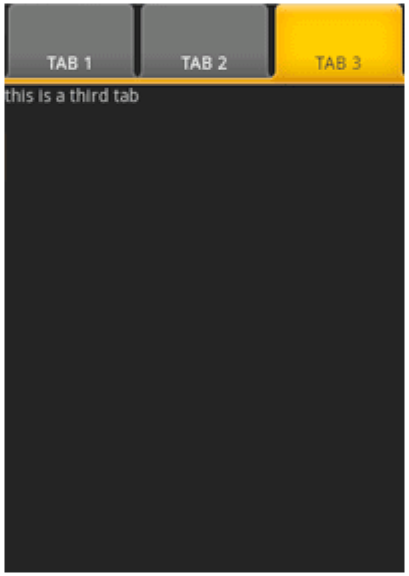


SeekBar

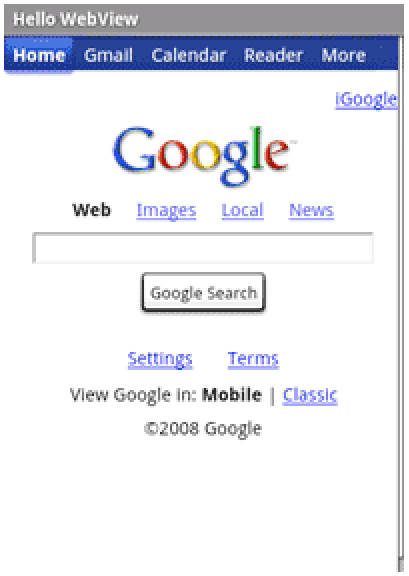


More View Items

TabHost



WebView



Even more: http://vidarvestnes.blogspot.com/2010/01/android-gui-examples.html

Assignment #2 & Friday

- Discussion of Assignment #2
  - Matching GUI components with a list of functional requirements

- Friday
  - Layouts
  - More on Views
  - XML Layouts?