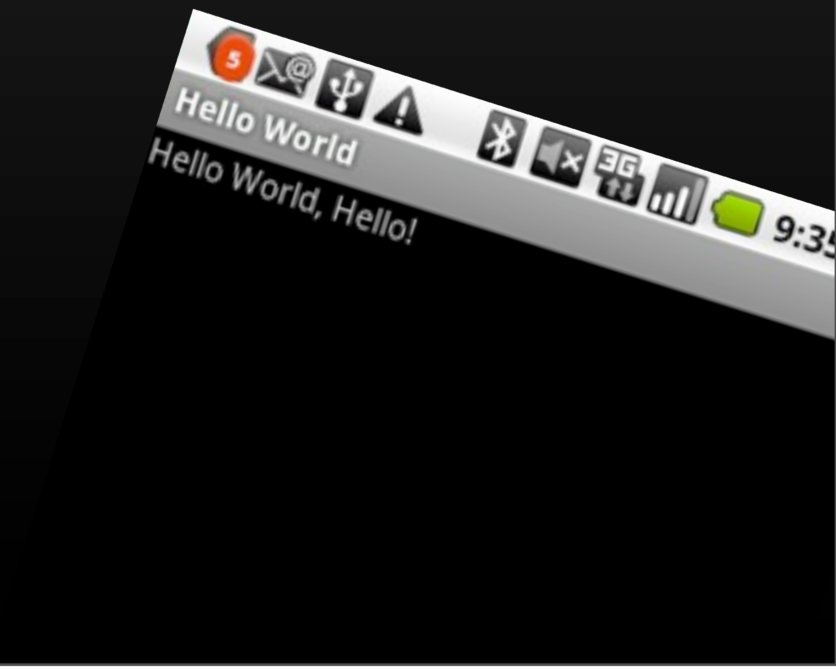
Android Developer Bootcamp

Google I/O Bootcamp 2010

Assumptions

- Familiarity with Java programming
- Installed Eclipse along with the Android SDK and the Android Developer Tools
- Have either an emulator image or a device that is set up for USB debugging


Hello World

Hello World, Hello!

Beginning Lab

In other words, if you have written

Agenda

- Building a form
- Transitioning to a new screen
- Displaying lists of data
- Taking advantage of reusable components

Importing the Project

- We’ve already created the project with the Eclipse new Android project wizard
- Expand IO2010BootCampBeginningLab.zip into your development directory
  Get at: http://bit.ly/b8Ldm3
- Use File->Import->General->Existing Project into Workspace

Project Overview

- In the Eclipse workspace, everyone should have:
  - The AndroidManifest.xml file
  - A source (src) directory
  - A resource (res) directory

The Android Manifest

```
<manifest
 android:versionCode="1"
 android:versionName="1.0"
 package="com.android.googleio2010bootcampcodelab">
 <application android:icon="@drawable/icon"
  android:label="@string/app_name">
  <activity android:name=".CollectData"
    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=".ListData"></activity>
 </application>
 <uses-sdk android:minSdkVersion="3"/>
</manifest>
```

The Android Manifest

```
<manifest
 android:versionCode="1"
 android:versionName="1.0"
 package="com.android.googleio2010bootcampcodelab">
 <application android:icon="@drawable/icon"
  android:label="@string/app_name">
  <activity android:name=".CollectData"
    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=".ListData"></activity>
 </application>
 <uses-sdk android:minSdkVersion="3"/>
</manifest>
```

The Android Manifest

```
<manifest
 android:versionCode="1"
 android:versionName="1.0"
 package="com.android.googleio2010bootcampcodelab">
 <application android:icon="@drawable/icon"
  android:label="@string/app_name">
  <activity android:name=".CollectData"
    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=".ListData"></activity>
 </application>
 <uses-sdk android:minSdkVersion="3"/>
</manifest>
```

Our Main Activity

CollectData.java

- An activity approximately represents a screen
- Android applications consist of a collection of Activities
- Android provides a powerful View system used by Activities

The Android Manifest

```
<manifest
 android:versionCode="1"
 android:versionName="1.0"
 package="com.android.googleio2010bootcampcodelab">
 <application android:icon="@drawable/icon"
  android:label="@string/app_name">
  <activity android:name=".CollectData"
    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=".ListData"></activity>
 </application>
 <uses-sdk android:minSdkVersion="3"/>
</manifest>
```

Views and Layouts

A Demonstration using RelativeLayout

- Layouts are responsible for placing and sizing views
- Layouts are also views, and can therefore contain other layouts
- RelativeLayout places widgets/views relative to both the parent layout and to siblings
- References to all resources, including layouts, are automatically generated into the R.java class

Event Handling

Getting notified when our button is clicked

- Calling setContentView will inflate the view hierarchy based upon the XML
- Use findViewById to get a reference to a view
- Use the found reference to attach an event handler

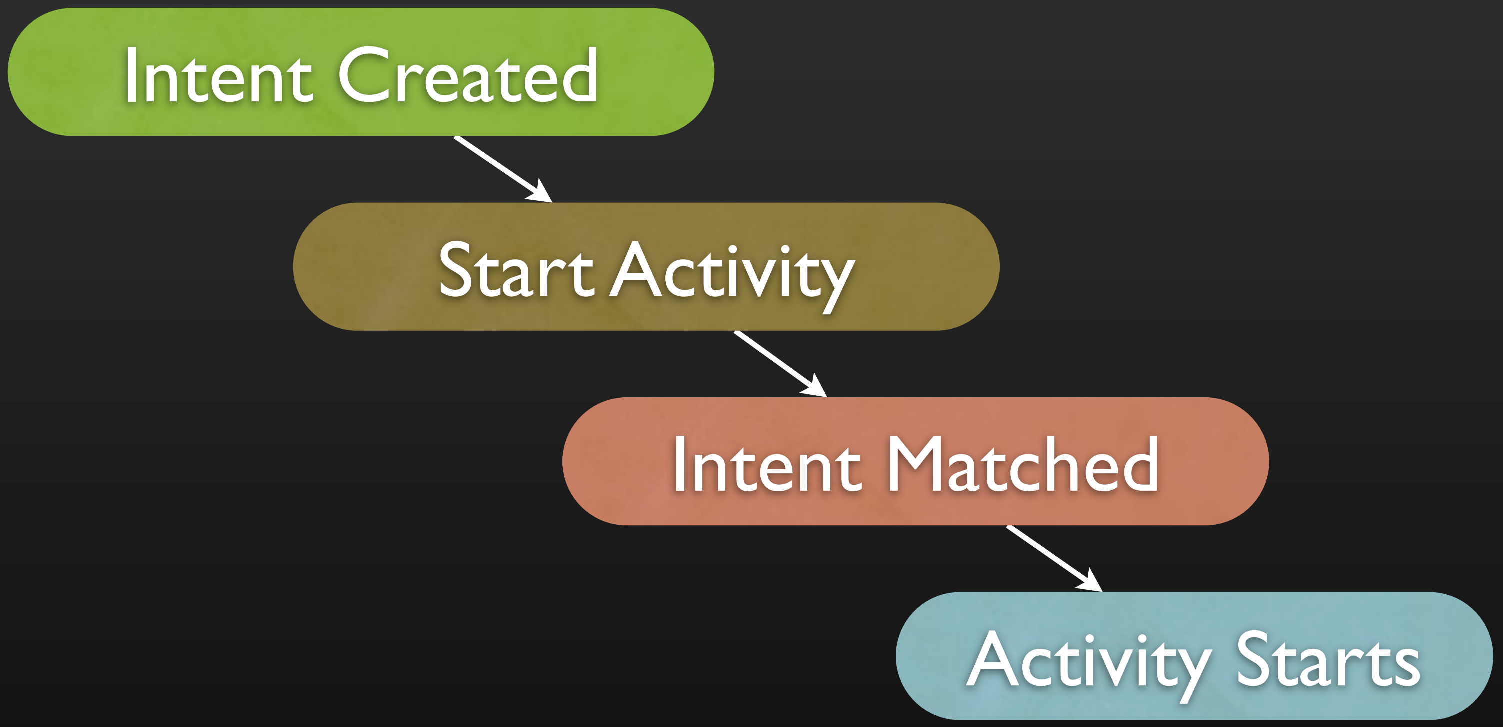
Intents

Transitioning Between Activities

- Intents are messages that tell the system the next action to take
- Intents can contain data for the receiving activity
- Activities are registered with the system in the AndroidManifest

Activity Lifecycle

How Activities Get Launched



Intent Created

Start Activity

Intent Matched

Activity Starts

The Android Manifest

```
<manifest
 android:versionCode="1"
 android:versionName="1.0"
 package="com.android.googleio2010bootcampcodelab">
 <application android:icon="@drawable/icon"
  android:label="@string/app_name">
  <activity android:name=".CollectData"
    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=".ListData"></activity>
 </application>
 <uses-sdk android:minSdkVersion="3"/>
</manifest>
```

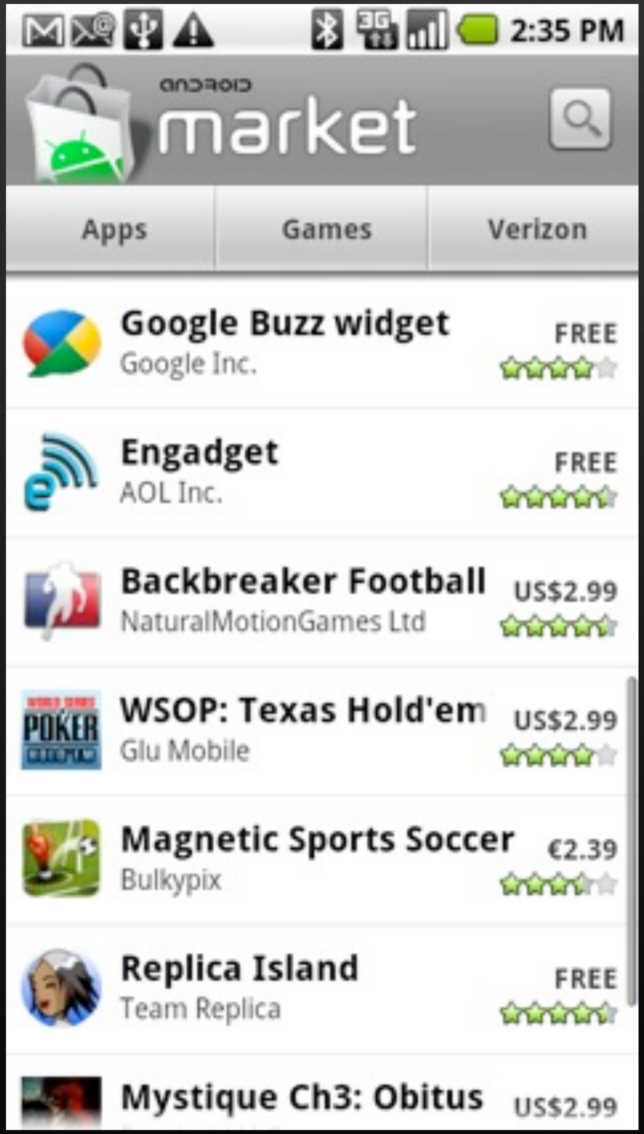
ListView

Displaying sets of data

- ListView displays data into sets of child views in a vertically scrolling list
- ListAdapter is responsible for supplying child views to a ListView and filling them with data
- Supports different types of child views with different vertical sizes for maximum flexibility
- Handles recycling of child views

ListView

Examples

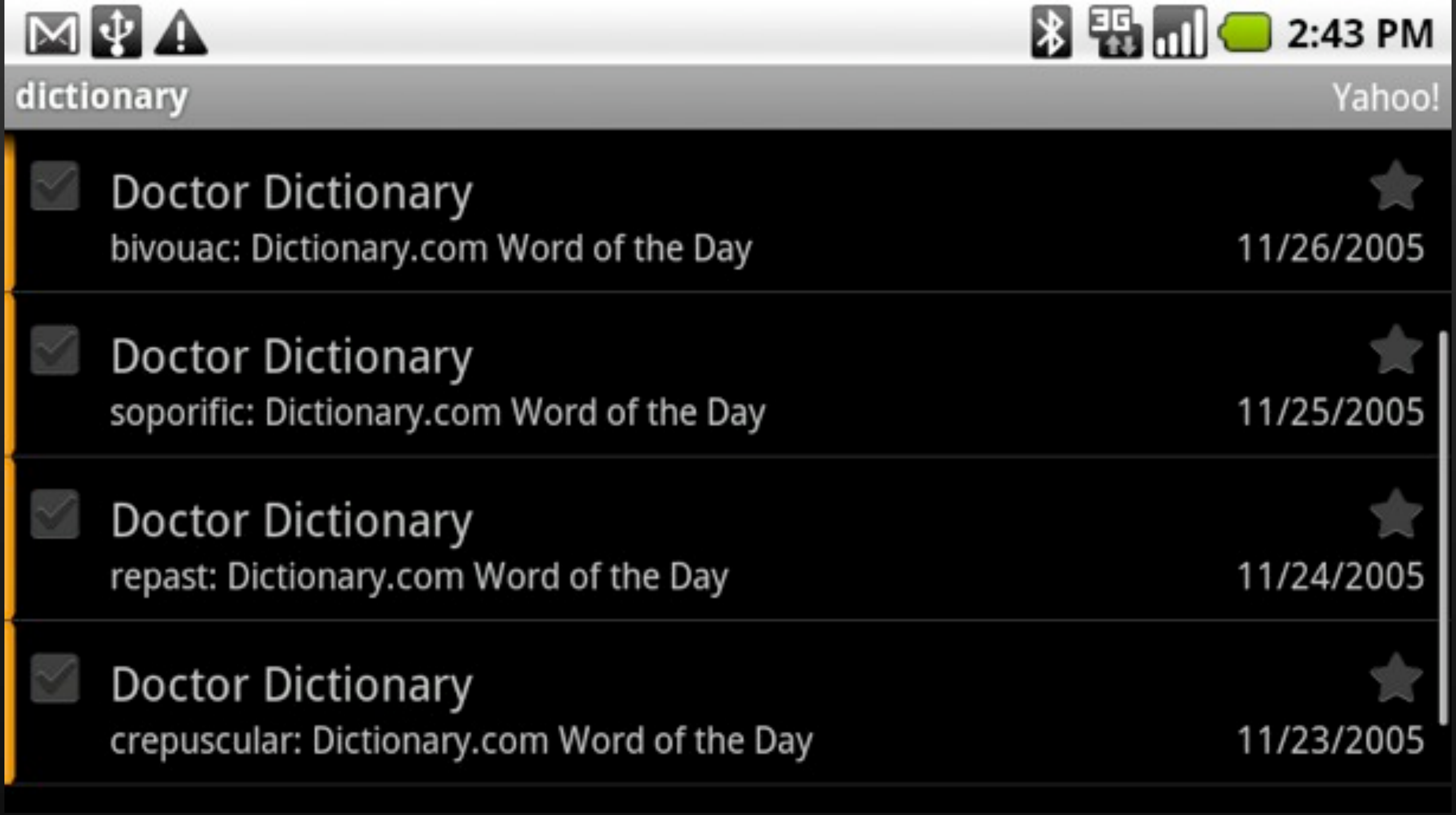


2:35 PM

market

Apps | Games | Verizon

- Google Buzz widget — Google Inc. — FREE
- Engadget — AOL Inc. — FREE
- Backbreaker Football — NaturalMotionGames Ltd — US$2.99
- WSOP: Texas Hold'em — Glu Mobile — US$2.99
- Magnetic Sports Soccer — Bulkypix — €2.39
- Replica Island — Team Replica — FREE
- Mystique Ch3: Obitus — US$2.99



2:43 PM

dictionary — Yahoo!

- Doctor Dictionary — bivouac: Dictionary.com Word of the Day — 11/26/2005
- Doctor Dictionary — soporific: Dictionary.com Word of the Day — 11/25/2005
- Doctor Dictionary — repast: Dictionary.com Word of the Day — 11/24/2005
- Doctor Dictionary — crepuscular: Dictionary.com Word of the Day — 11/23/2005

Intentionally

Applications work together

- Applications talk to each other using Intents
- IntentFilters provide a description of the generic activities that applications perform
- The Intent “description” can match up the Intent to the IntentFilter in many ways

Beyond the Lab

Extending our application

- The first bonus activity demonstrates how to create a context-sensitive menu for choosing the query to perform
- The second bonus activity demonstrates how to create an options menu that links to a preferences screen that selects the default query
- A function is provided that uses reflection to get the integer ID's from the R.id class, rather than from a lookup function

Further Beyond the Lab

Extending our application

- As an exercise, replace the included SimpleAdapter with a CursorAdapter backed by a SQLite database
- Convert the first activity into a windowless activity, make it work like a dialog, and have the second activity start it

Q & A