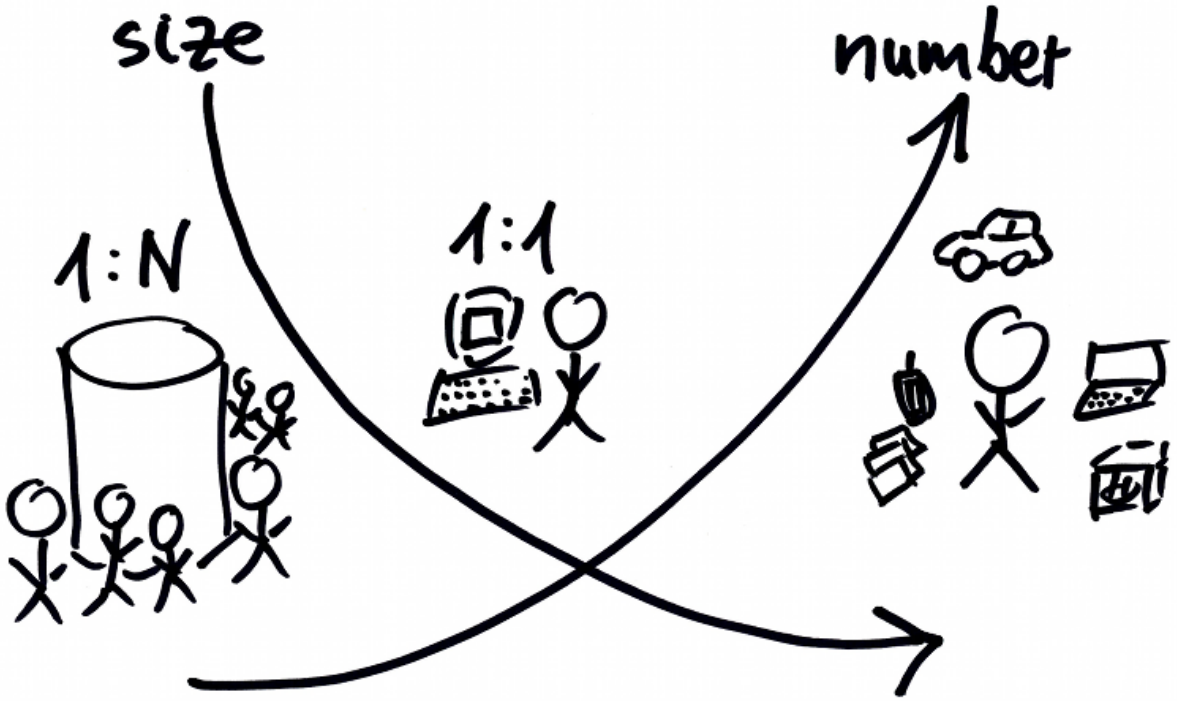
Android Workshop

Probestudium Informatik 2012
30.3.-4.4.2012

Prof. Dr. Michael Rohs
michael.rohs@ifi.lmu.de
Dipl.-Inform. Sven Kratz
sven.kratz@ifi.lmu.de
Mobile Interaction Lab, LMU München

Ubiquitous Computing



- Computers embedded in everyday things
- Technology moves into the background
- Computers in the world, instead of world in the computer


INFORMATION

Zeitplanung

	Freitag,30.03.12	**Montag, 2.04.121**	**Dienstag, 3.04.11**	**Mittwoch, 5.04.121**
Vormittag	Raum B 001 (=großer Hörsaal) 10.00 Uhr Registrierung 10.15 Uhr Begrüßung Einführung in die Projektthemen: WS I: CIP-Pool WS II: CIP-Pool WS III: Raum 115 WS IV: Raum 123	Raum B 001 Vorlesung 1 Vorlesung 2	Raum B 001 Vorlesung 3 Vertiefung der Projektthemen	Oettingenstraße Gruppenarbeit WS I: CIP-Pool WS II: CIP-Pool WS III: Raum 115 WS IV: Raum 123
Nachmittag	Gruppenarbeit WS I: CIP-Pool WS II: CIP-Pool WS III: Raum 115 WS IV: Raum 123	Gruppenarbeit WS I: CIP-Pool WS II: CIP-Pool WS III: Raum 115 WS IV: Raum 123	Gruppenarbeit WS I: CIP-Pool WS II: CIP-Pool WS III: Raum 115 WS IV: Raum 123	Raum B 001 Präsentation der Ergebnisse Vorstellung der (besten) Projekte im Plenum

Zeitplanung

Fr Vormittag	Begrüßung, Einführung Workshop
Fr Nachmittag	Workshop
Mo Vormittag	Probevorlesungen
Mo Nachmittag	Workshop
Di Vormittag	Vertiefungsvorlesung
Di Nachmittag	Workshop
Mi Vormittag	Workshop
Mi Nachmittag	Präsentationen

Ziel: Ein Orts-basiertes Quiz

- Was macht der Benutzer? (→ „Szenario“)

- „Jan ist zu Besuch in München. Er möchte mehr über die Stadt erfahren und lädt sich das neue mobile Quiz auf sein Handy. Er geht durch die Stadt. Wenn er sich einer Sehenswürdigkeit nähert, vibriert sein Handy und zeigt ein Bild und eine Erklärung dazu. Unter dem Bild sind eine Frage und vier mögliche Antworten zu sehen. Durch Antippen der richtigen Antwort bekommt er Pluspunkte. Wählt er eine falsche Antwort aus, handelt er sich Minuspunkte ein. Wenn er genügend Punkte ergattert hat, bekommt er eine günstigere Eintrittskarte für das Museum.“

Realisierung

- Ortsinformation verarbeiten
- Bild und Text auf dem Display anzeigen
- Auswahl erkennen, Punkte zählen
- zwischen Bildschirmen hin- und herschalten

- erster Teil: Hintergrundwissen erarbeiten
- zweiter Teil: Projekt in 4er-Gruppen erstellen

- Wer hat schon ein Android-Handy?

- Problem: unterschiedliches Vorwissen, bitte Fragen!!!

Exercise:

- Form groups of 4

JAVA

Klassen, Variablen, Methoden…

- Klassen sind „Pläne“ für Objekte
  - Daten: Attribute
  - Funktionalität: Methoden
- Objekte sind Instanzen von Klassen
  - werden explizit erzeugt
  - belegen Speicherplatz
- Attribut stellt Zustand von Objekt dar
  - Referenz auf anderes Objekt
  - primitiver Datentyp (int, boolean, double, float, byte, char, long)
- Methoden stellen Funktionalität des Objektes dar
  - Berechnungen
  - Ausgaben
  - etc.

Beispiel: Klasse „Haus“

```
package de.lmu.mobilehci.test;

public class Haus {

        protected double laenge;
        protected double breite;

        protected Tuer haustuer;
        protected Fenster[] fenster;

        public Haus(double laenge, double breite) {
                this.laenge = laenge;
                this.breite = breite;
        }
```

Beispiel: Klasse „Haus“

```
    public double getLaenge() {
            return laenge;
    }

    public double getBreite() {
            return breite;
    }

    public double getFlaeche() {
            return laenge * breite;
    }

    public String toString() {
            return "[Haus: " + laenge + "m lang, " +
                    breite + "m breit]";
    }
}
```

„Baumhaus“ abgeleitet von „Haus“

```
package de.lmu.mobilehci.test;

public class Baumhaus extends Haus {
    private Baum baum;
    private double distanzVomBoden;

    public Baumhaus(double laenge, double breite, double distanzVomBoden) {
        super(laenge, breite);
        this.distanzVomBoden = distanzVomBoden;
    }

    public double getDistanzVomBoden() {
        return distanzVomBoden;
    }

    public String toString() {
        return "[Baumhaus: " + laenge + "m lang, " +
                breite + "m breit, " + distanzVomBoden + "m Fallhöhe]";
    }
}
```

Interfaces

- Interfaces definieren eine Menge von Methoden
- Interfaces haben keine Implementierung
- Interfaces haben kein Zustand
- Klassen implementieren Interfaces

- Beispiel:

```
public interface OnClickListener {
    public void onClick(View v);
}
```

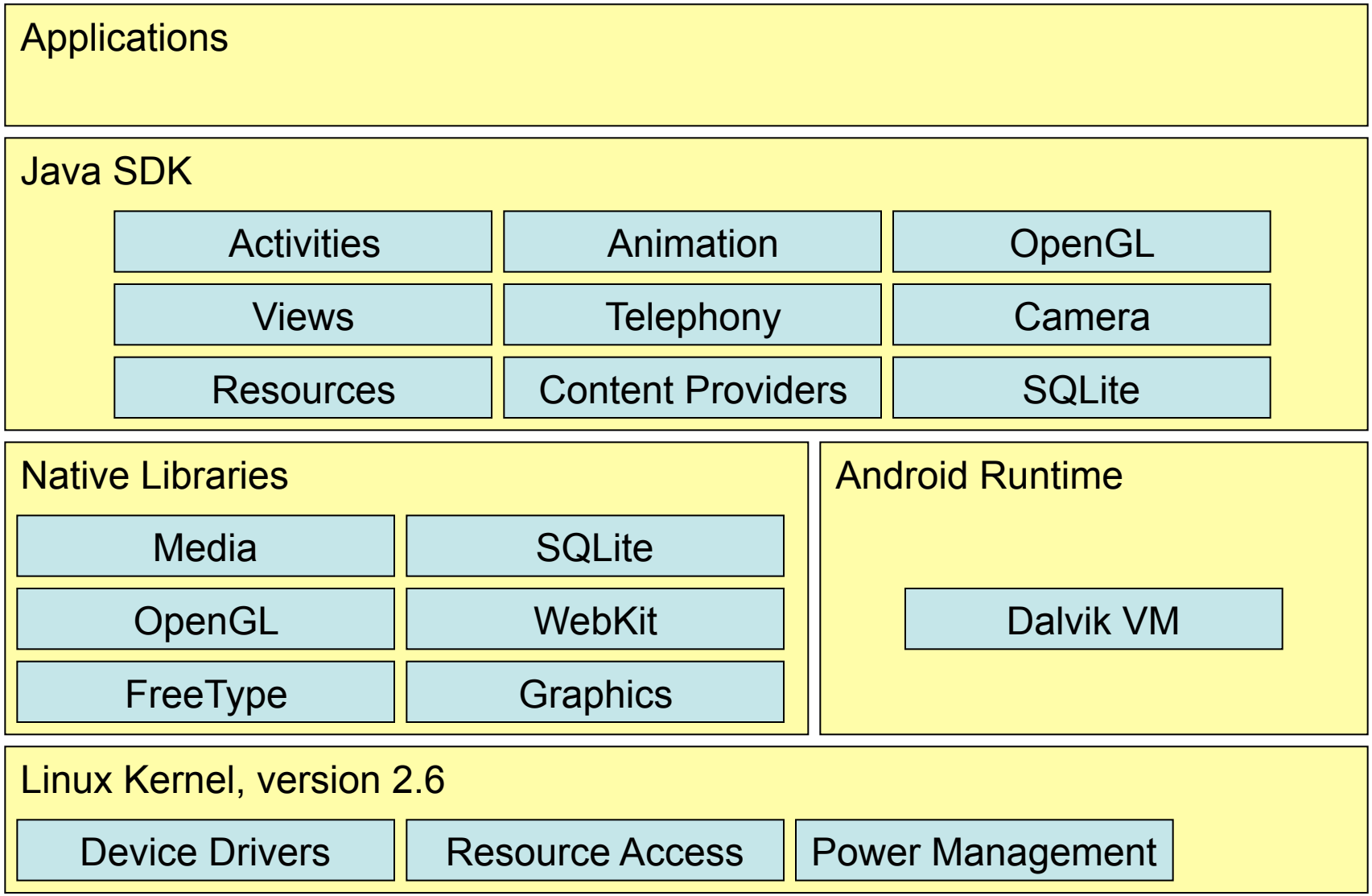
Class MyActivity implements OnClickListener

- Beispiel

```
public class MyActivity implements OnClickListener {

    public void onClick(View v) {
      System.out.println(v.toString());
    }

    // weitere Methoden der Klasse MyActivity

}
```

ANDROID

Android Software Stack

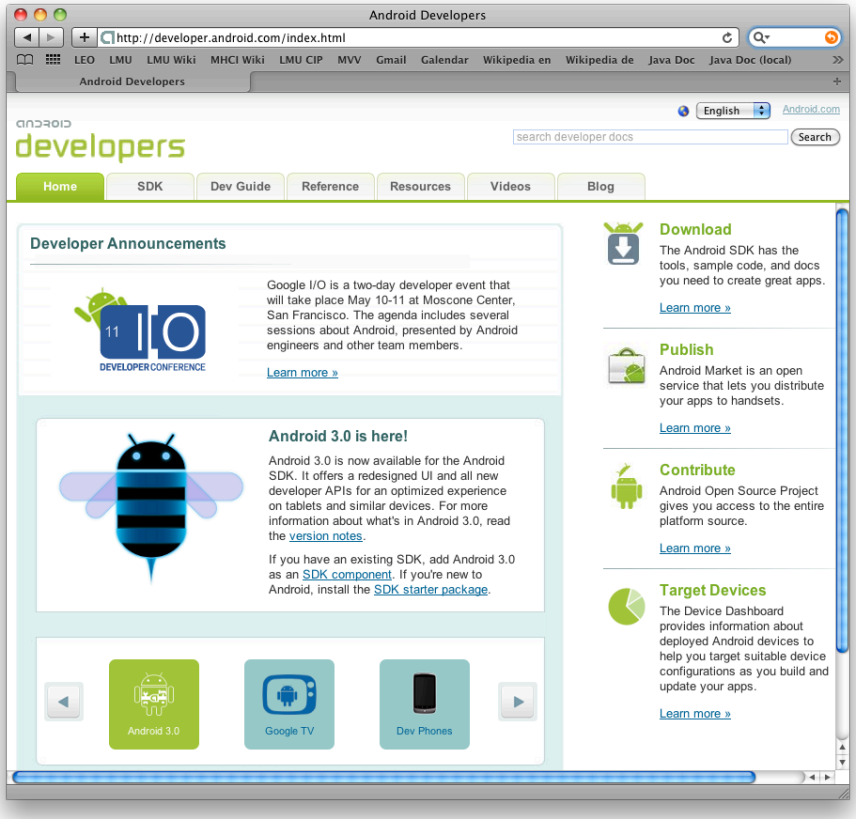


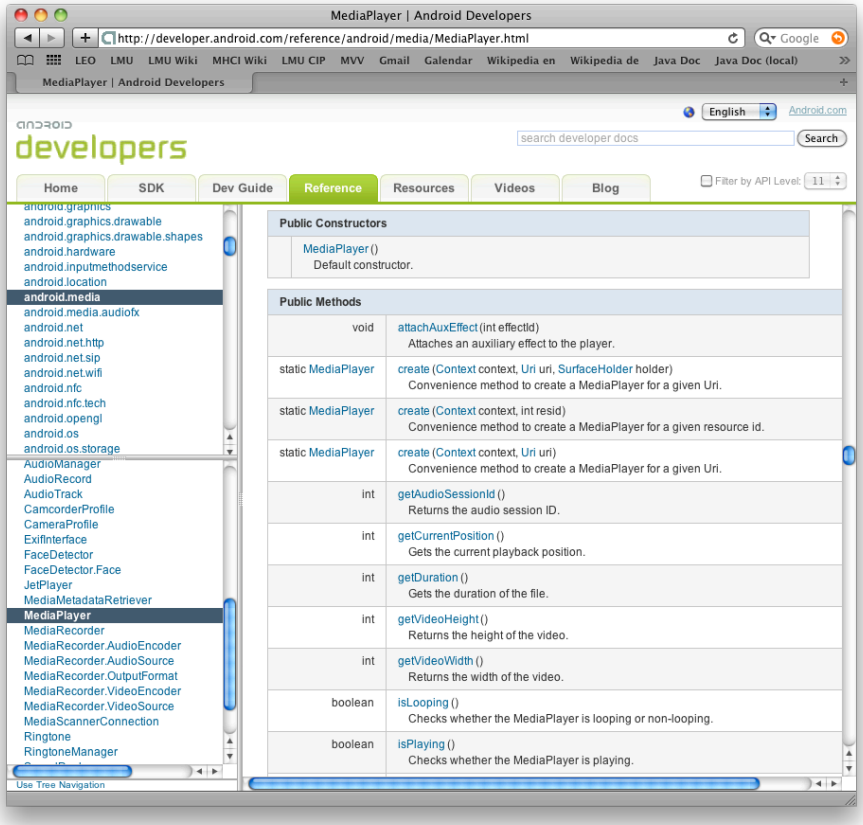
Android Characteristics

- Activity
  - Activities are the components of an application
  - Represent a logical unit of user action
  - Typically represented by a screen containing views
  - Can be invoked externally
- Declarative UI definition
  - XML files specify user interface resources
  - Resources (layout definitions, strings, bitmaps)
  - Separation of code and user interface
- User events handled programmatically

Android Resources

- Android developer pages (platform documentation)
  - http://developer.android.com





INSTALLING ANDROID

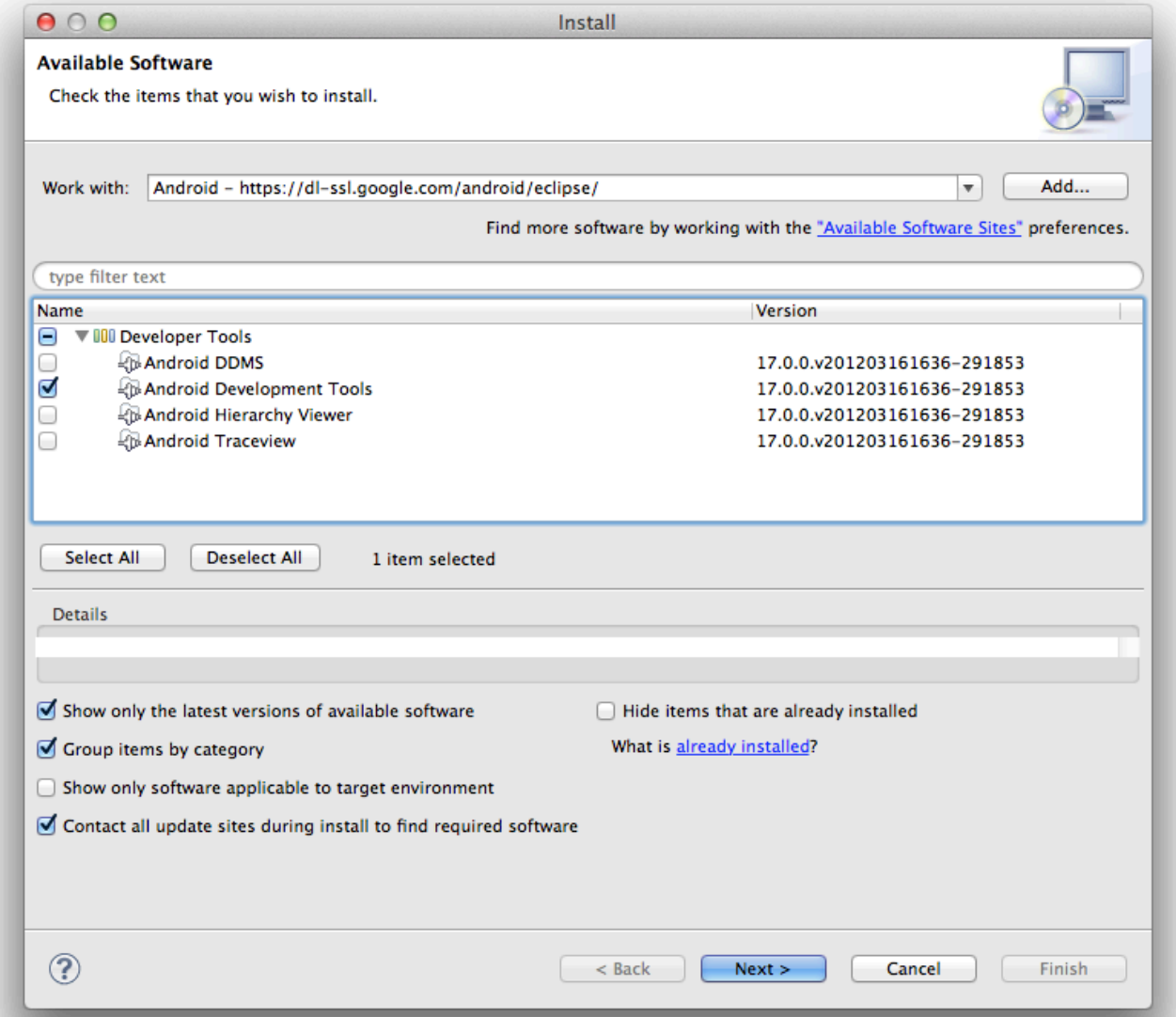
Installation (Overview)

- Start Eclipse
  - Alt-F2: “eclipse-ide-3.6” eintippen (← Eclipse-Version „Helios“)
- In Eclipse: Install Android SDK
  - Menu: Help, Install New Software…
  - https://dl-ssl.google.com/android/eclipse/
- Point Eclipse to the Android SDK starter package
  - Menu: Window, preferences, Android, SDK Location
  - /soft/IFI/lang/android-sdk-r17/iX86-unknown-linux
- In Eclipse: Android SDK and AVD Manager
  - Window / Android SDK and AVD Manager
  - New... / Virtual Devices / 2.3.3 mit Google API
- Mobile Phone
  - Anwendungen, Entwicklung: USB-Debugging, ...

In Eclipse: Install New Software...

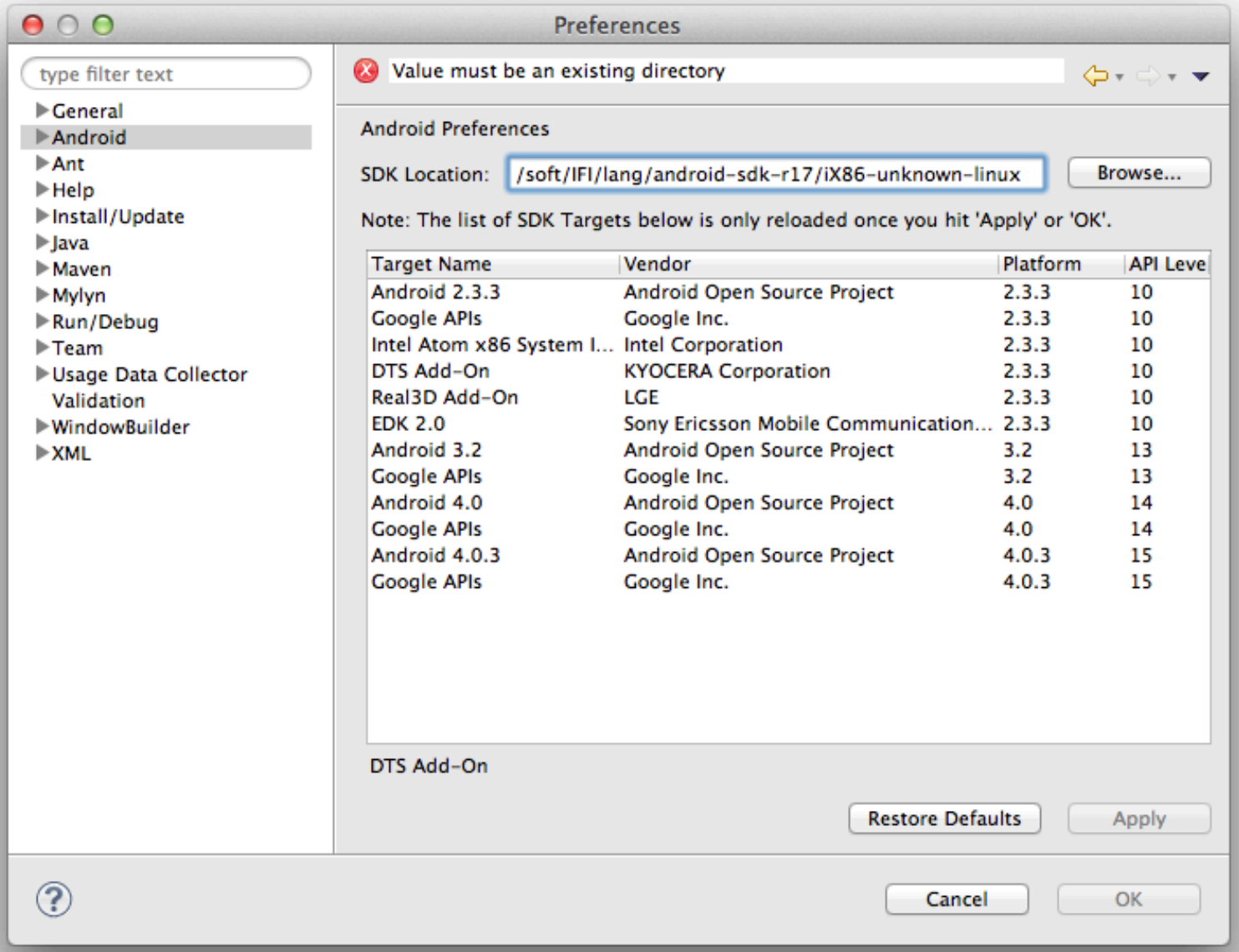
Android Plugin – https://dl-ssl.google.com/android/eclipse/

Only “Android Development Tools”

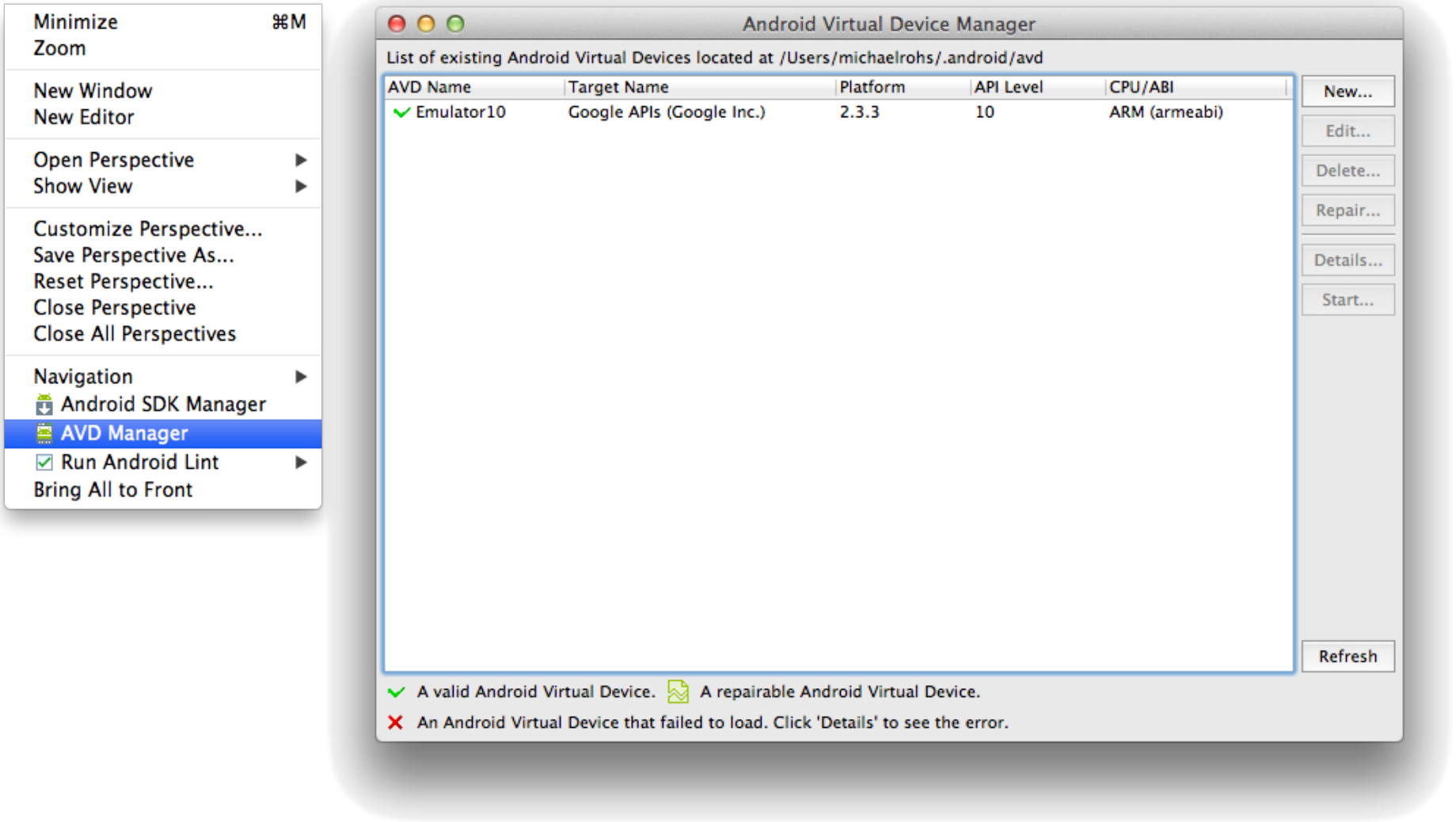


Set Path to Android SDK Starter Package

Preferences, Android: /soft/IFI/lang/android-sdk-r17/iX86-unknown-linux



Define Android Virtual Device



Entwicklung auf dem Handy

- Menü → Einstellungen → Apps
  - Unbekannte Herkunft (auswählen)
  - Entwicklung
    - USB-Debugging (auswählen)
    - Falsche Standorte (auswählen)

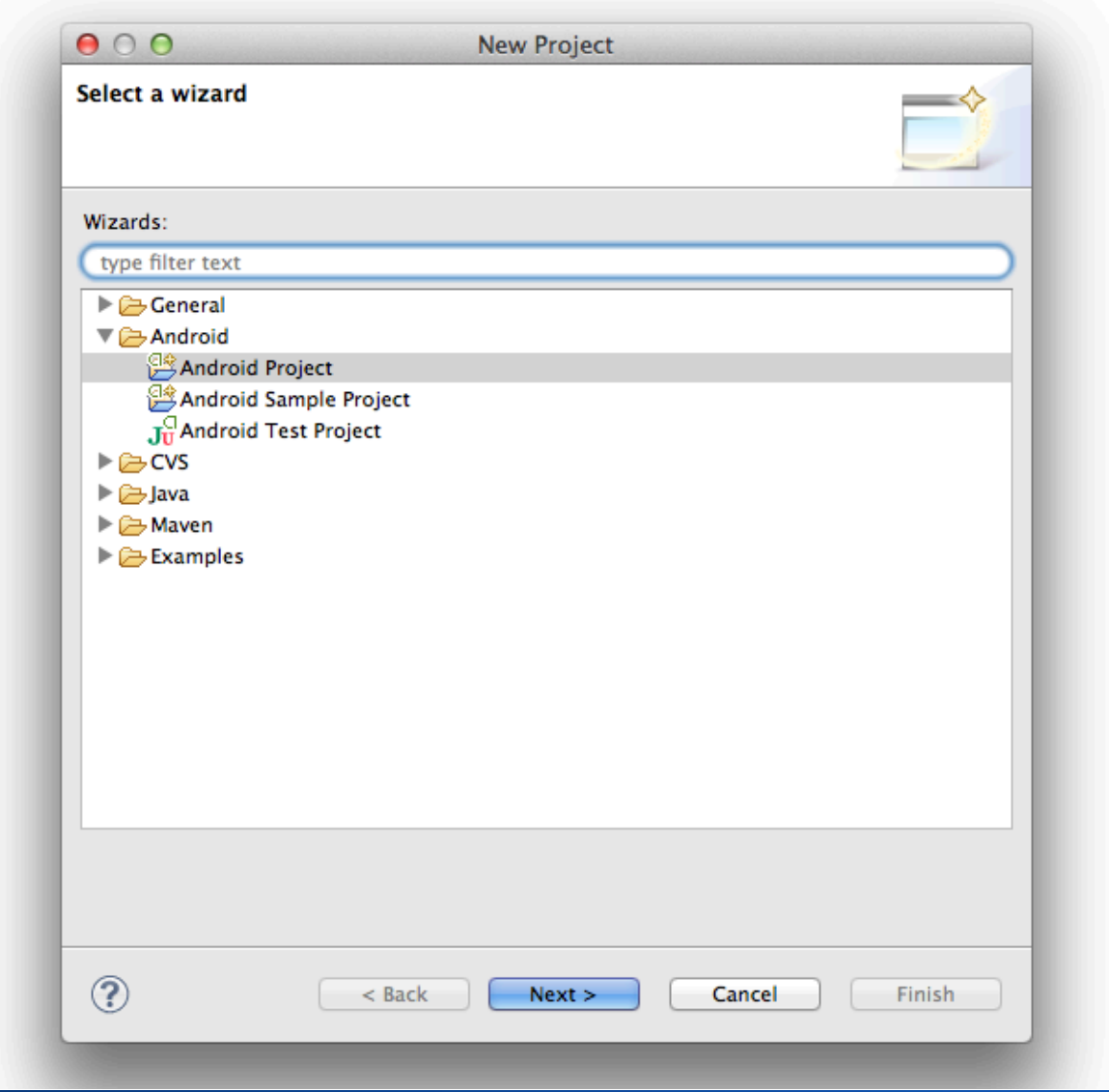
Exercise:

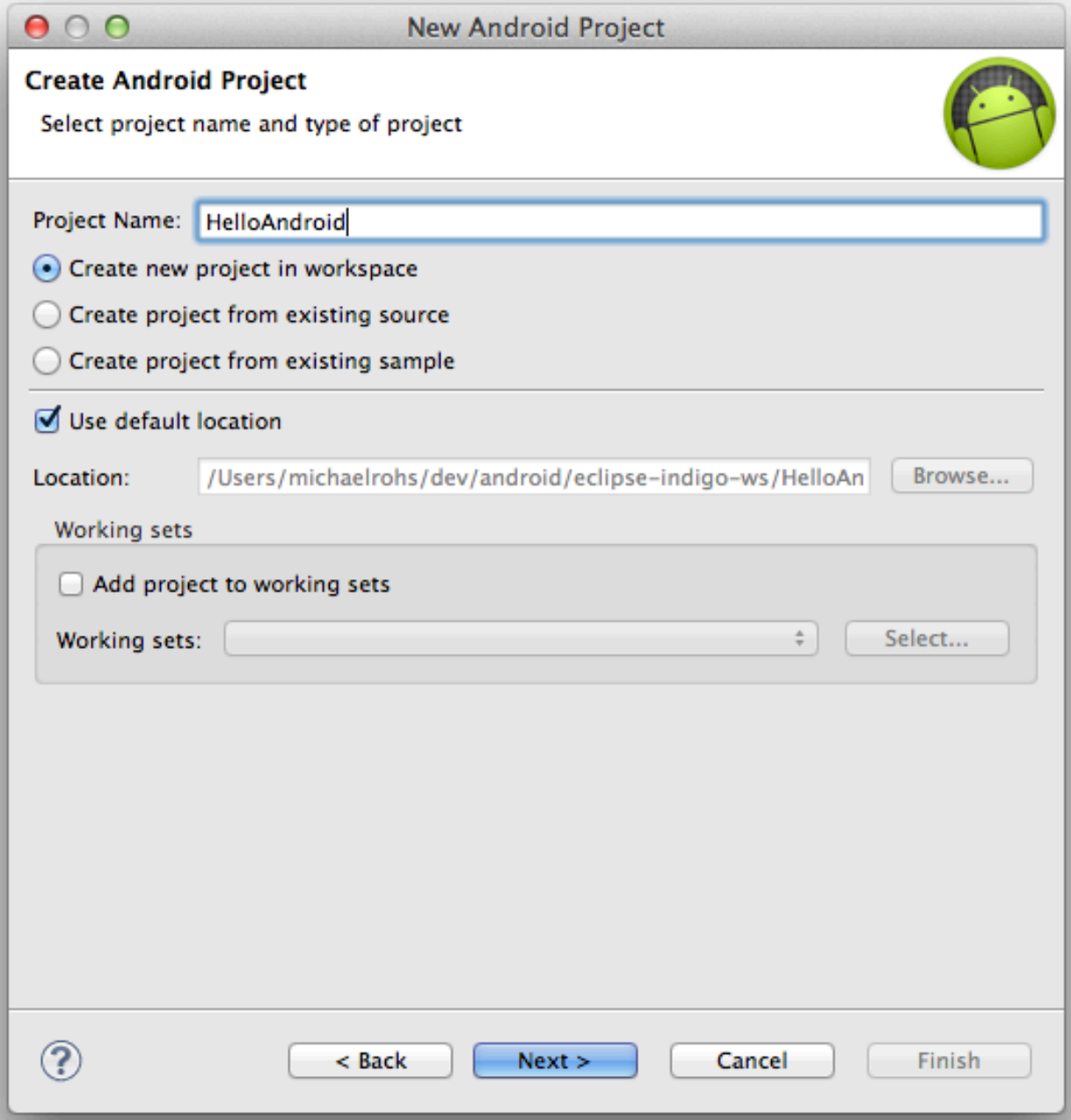
- Start Eclipse
  - Alt-F2: “eclipse-ide-3.6” eintippen
- In Eclipse: Install Android SDK
  - Menu: Help, Install New Software…
  - https://dl-ssl.google.com/android/eclipse/
- Point Eclipse to the Android SDK starter package
  - Menu: Window, preferences, Android, SDK Location
  - /soft/IFI/lang/android-sdk-r17/iX86-unknown-linux
- In Eclipse: Android SDK and AVD Manager
  - Window / Android SDK and AVD Manager
  - New... / Virtual Devices / 2.3.3 mit Google API

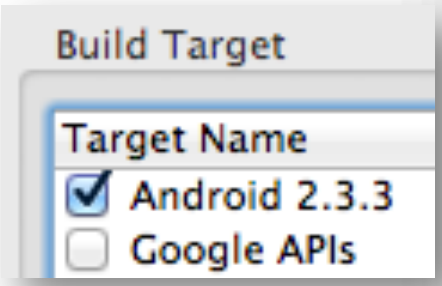
“HELLO WORLD”

Creating Your First Android Project

File → New → Project… → Android → Android Project




New Android Project
Create Android Project
Select project name and type of project
Project Name:
HelloAndroid
Create new project in workspace
Create project from existing source
Create project from existing sample
Use default location
Location:
/Users/michaelrohs/dev/android/eclipse-indigo-ws/HelloAn
Browse...
Working sets
Add project to working sets
Working sets:
Select...
< Back
Next >
Cancel
Finish



New Android Project

Select Build Target

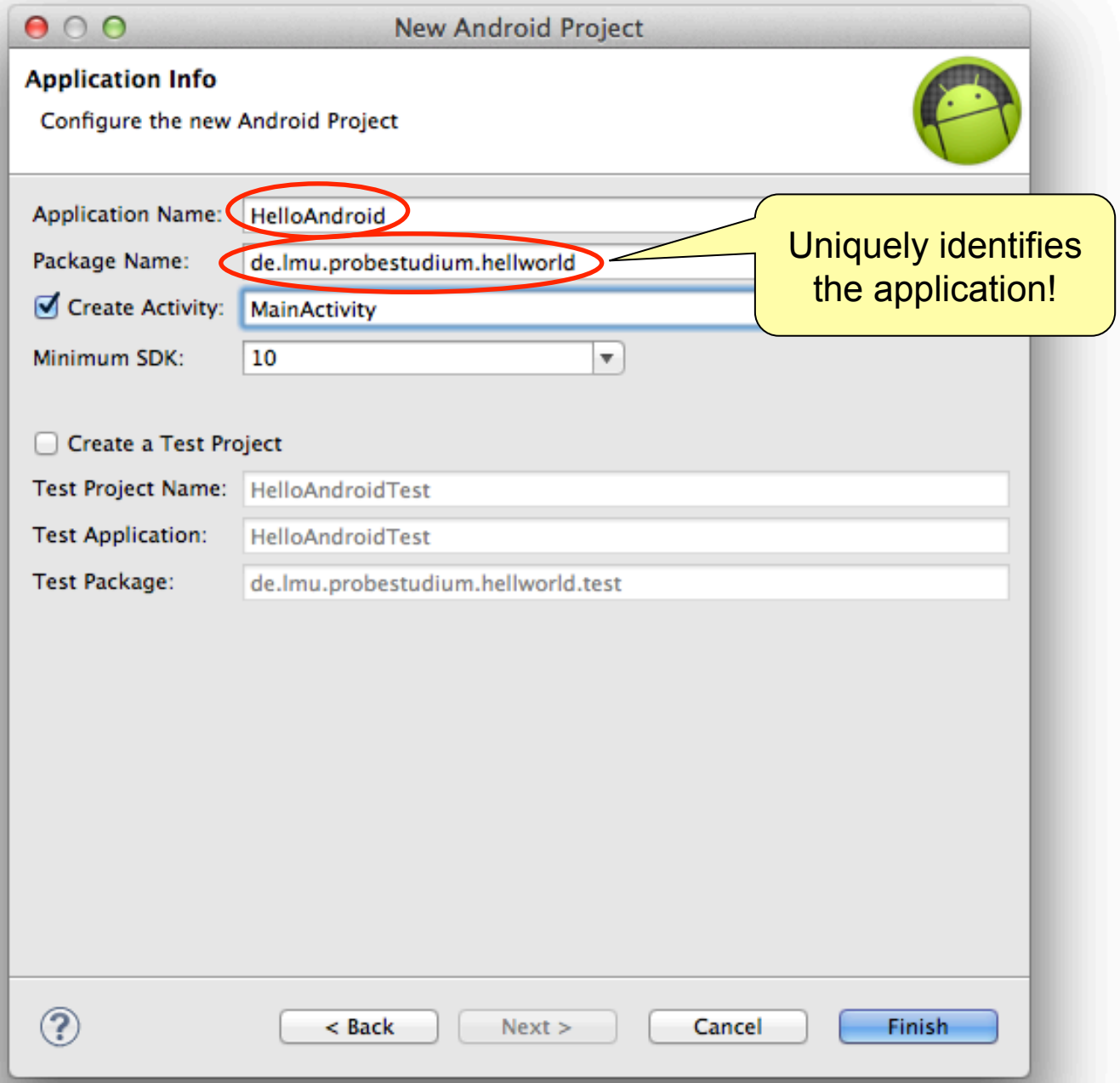
Choose an SDK to target

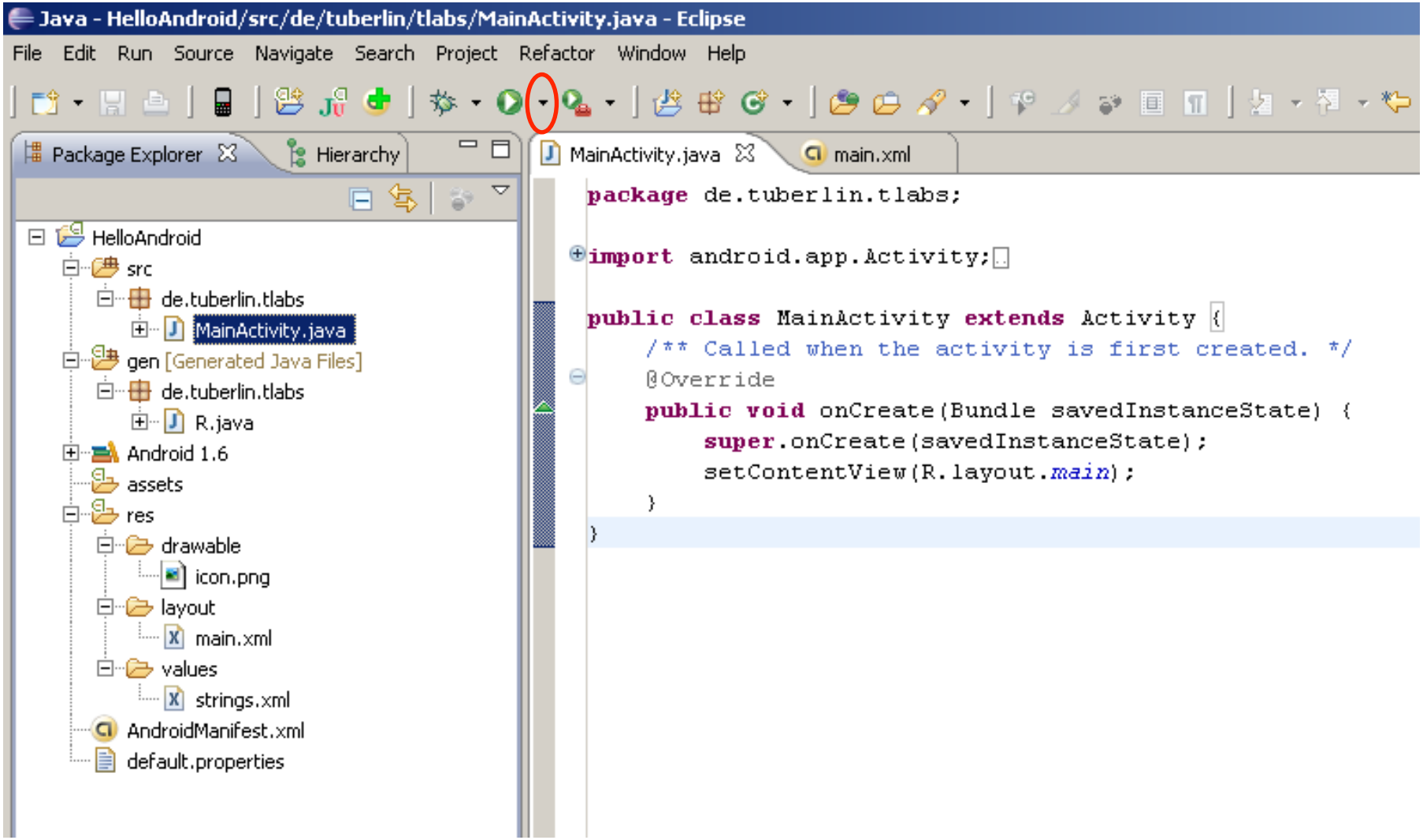
Build Target

Target Name	Vendor	Platform	API Leve
☑ Android 2.3.3	Android Open Source Project	2.3.3	10
☐ Google APIs	Google Inc.	2.3.3	10
☐ Intel Atom x86 System...	Intel Corporation	2.3.3	10
☐ DTS Add-On	KYOCERA Corporation	2.3.3	10
☐ Real3D Add-On	LGE	2.3.3	10
☐ EDK 2.0	Sony Ericsson Mobile Communicatio...	2.3.3	10
☐ Android 3.2	Android Open Source Project	3.2	13
☐ Google APIs	Google Inc.	3.2	13
☐ Android 4.0	Android Open Source Project	4.0	14
☐ Google APIs	Google Inc.	4.0	14
☐ Android 4.0.3	Android Open Source Project	4.0.3	15
☐ Google APIs	Google Inc.	4.0.3	15

Standard Android platform 4.0

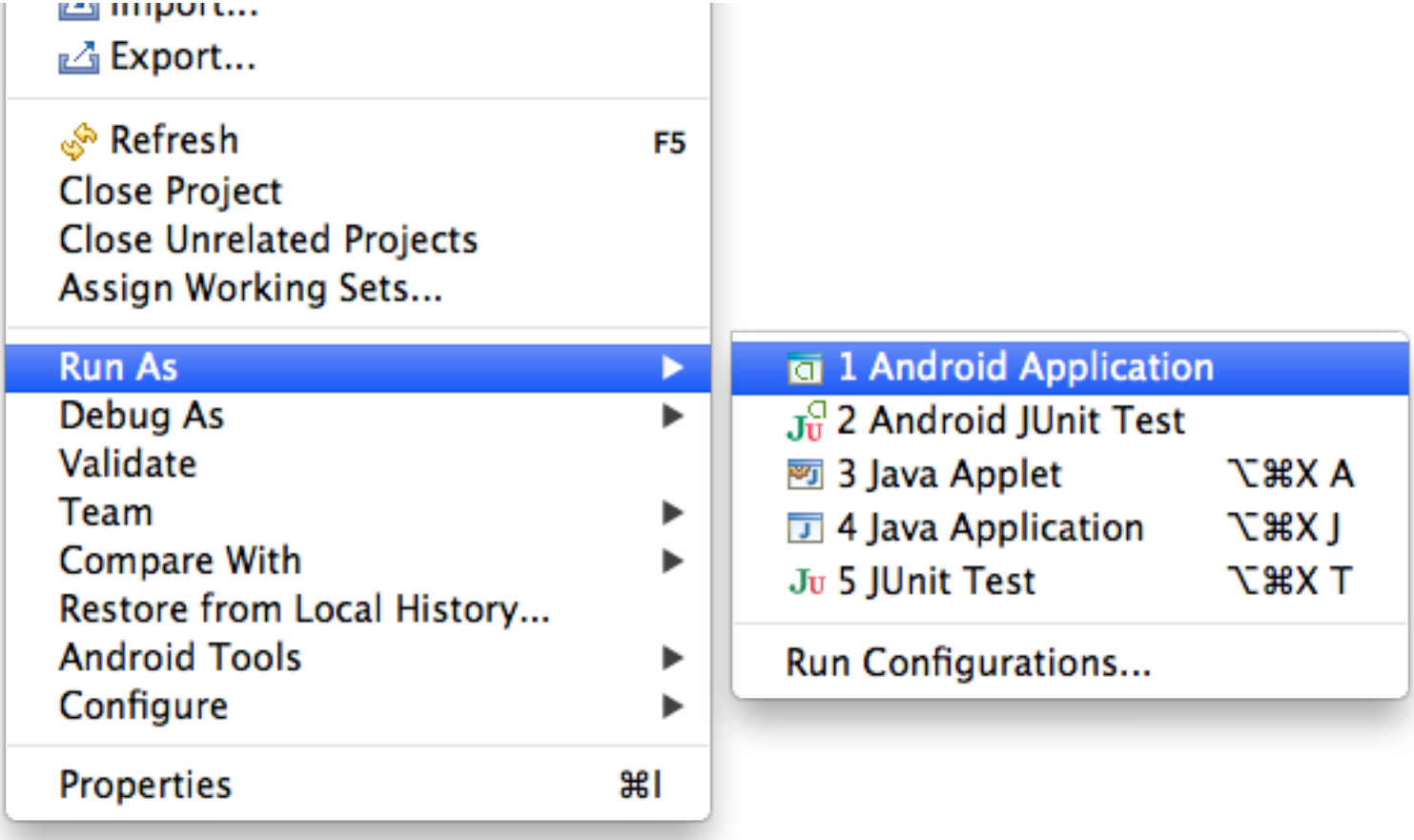
< Back | Next > | Cancel | Finish


New Android Project
Application Info
Configure the new Android Project
Application Name:
HelloAndroid
Package Name:
de.lmu.probestudium.hellworld
Create Activity:
MainActivity
Minimum SDK:
10
Create a Test Project
Test Project Name:
HelloAndroidTest
Test Application:
HelloAndroidTest
Test Package:
de.lmu.probestudium.hellworld.test
< Back
Next >
Cancel
Finish
Uniquely identifies the application!

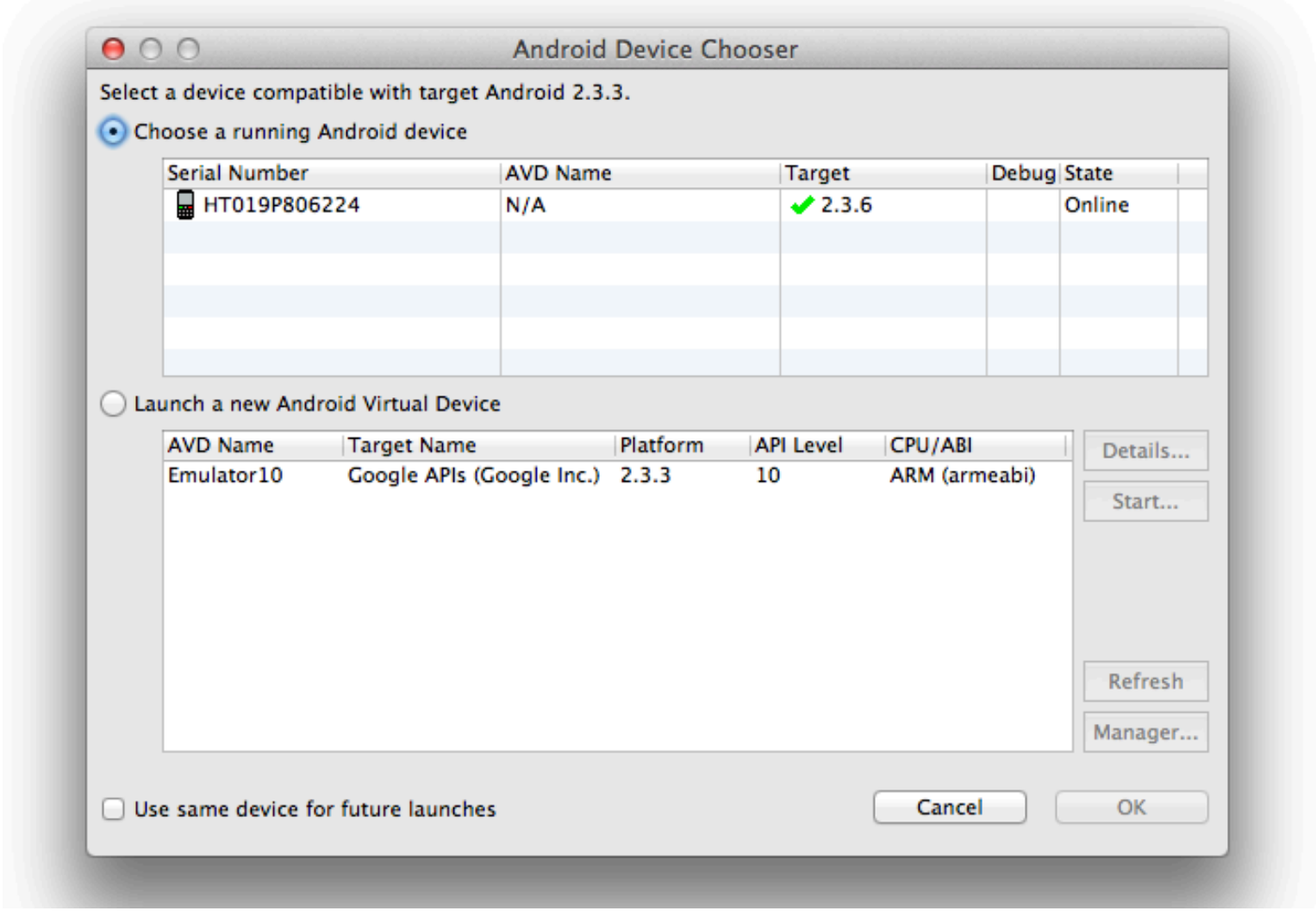

Java - HelloAndroid/src/de/tuberlin/tlabs/MainActivity.java - Eclipse
File Edit Run Source Navigate Search Project Refactor Window Help
Package Explorer
Hierarchy
HelloAndroid
src
de.tuberlin.tlabs
MainActivity.java
gen [Generated Java Files]
de.tuberlin.tlabs
R.java
Android 1.6
assets
res
drawable
icon.png
layout
main.xml
values
strings.xml
AndroidManifest.xml
default.properties
MainActivity.java
main.xml
package de.tuberlin.tlabs;
import android.app.Activity;
public class MainActivity extends Activity {
/** Called when the activity is first created. */
@Override
public void onCreate(Bundle savedInstanceState) {
super.onCreate(savedInstanceState);
setContentView(R.layout.main);
}
}

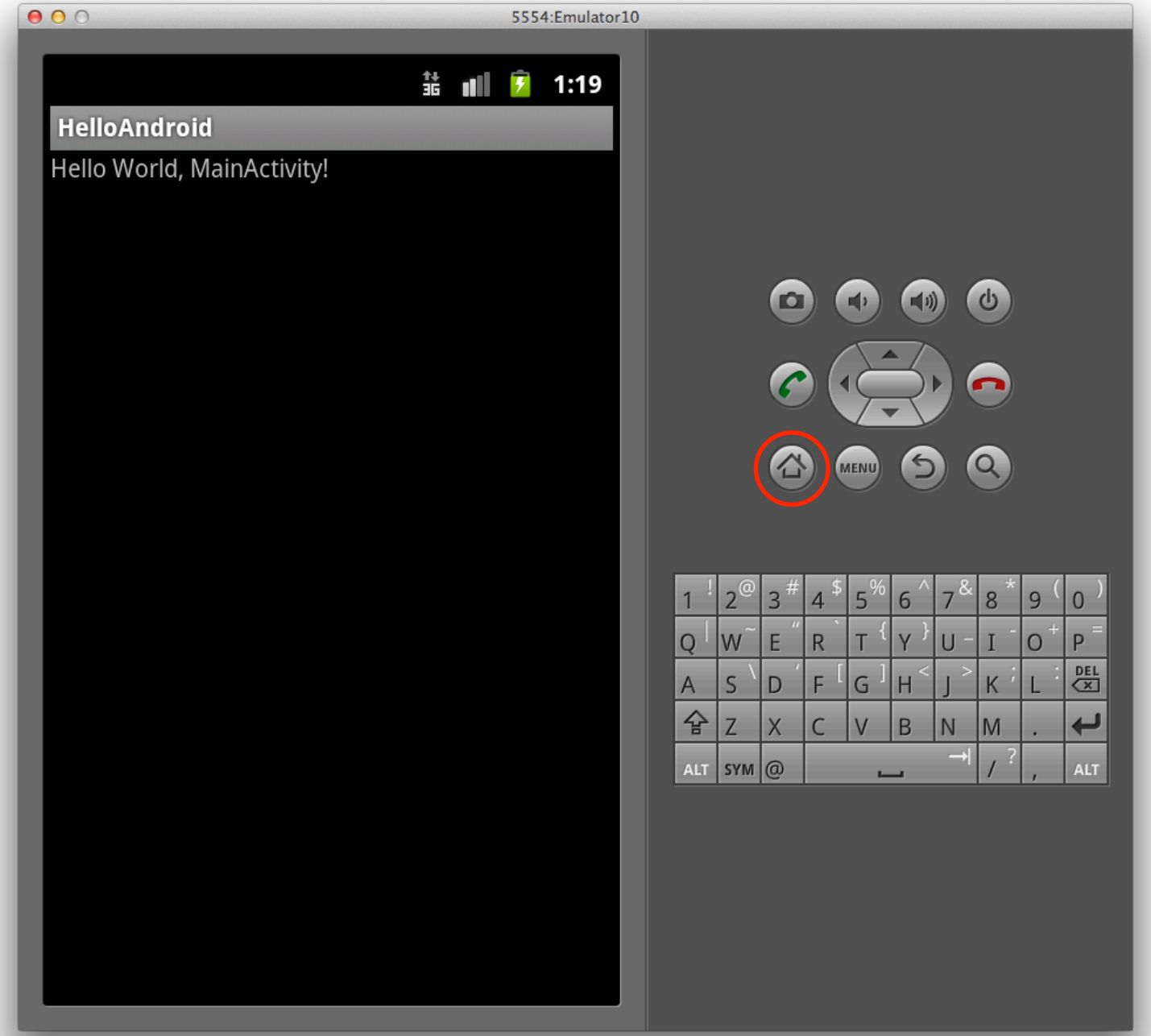
Run “HelloAndroid”

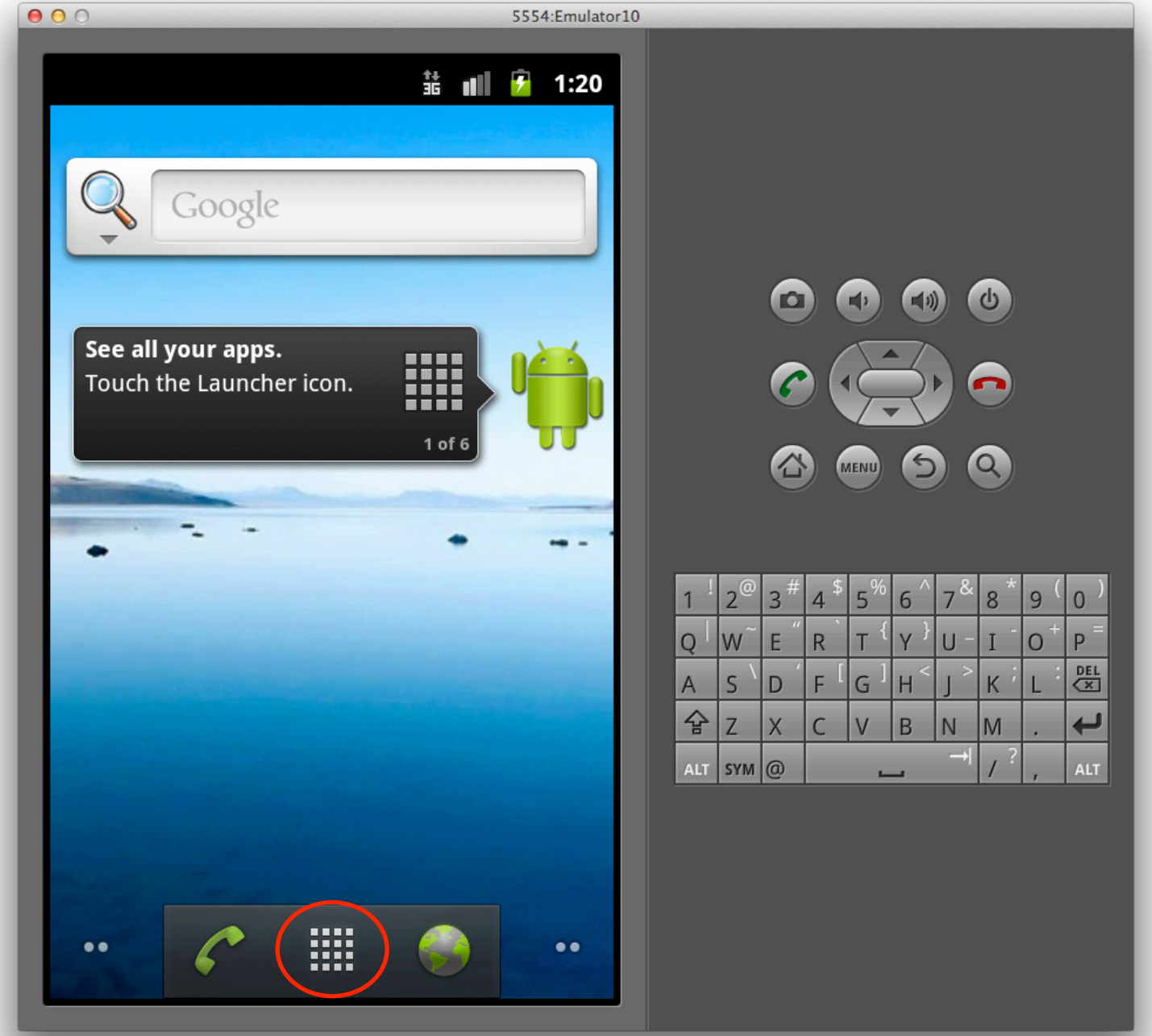
- Right-click “HelloAndroid” in Package Explorer

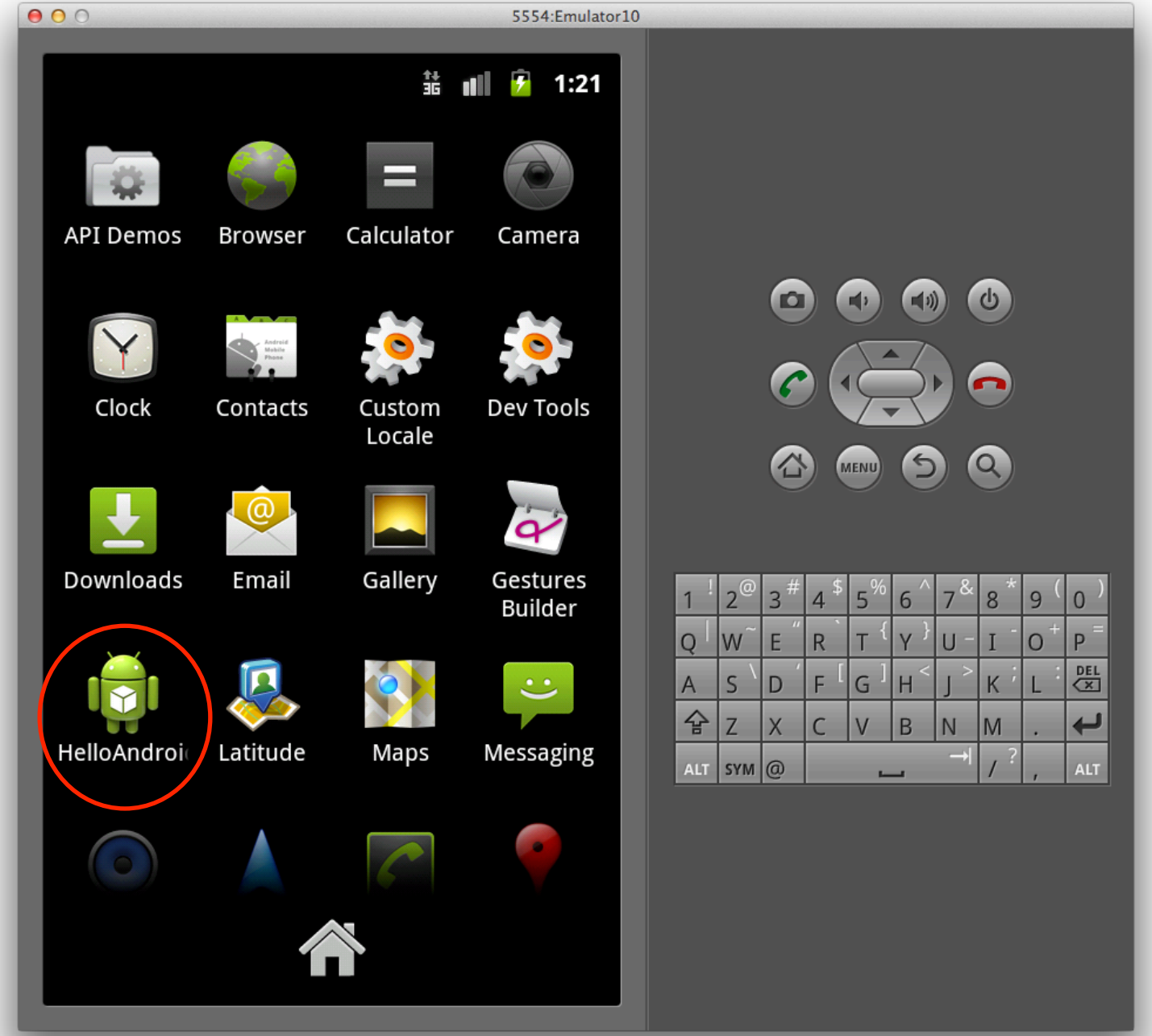


If phone plugged in and Emulator running, then choose where to run…




5554:Emulator10
1:19
HelloAndroid
Hello World, MainActivity!
MENU


5554:Emulator10
1:20
Google
See all your apps.
Touch the Launcher icon.
1 of 6
MENU
ALT
SYM
ALT


5554:Emulator10
1:21
API Demos
Browser
Calculator
Camera
Clock
Contacts
Custom Locale
Dev Tools
Downloads
Email
Gallery
Gestures Builder
HelloAndroi
Latitude
Maps
Messaging
MENU

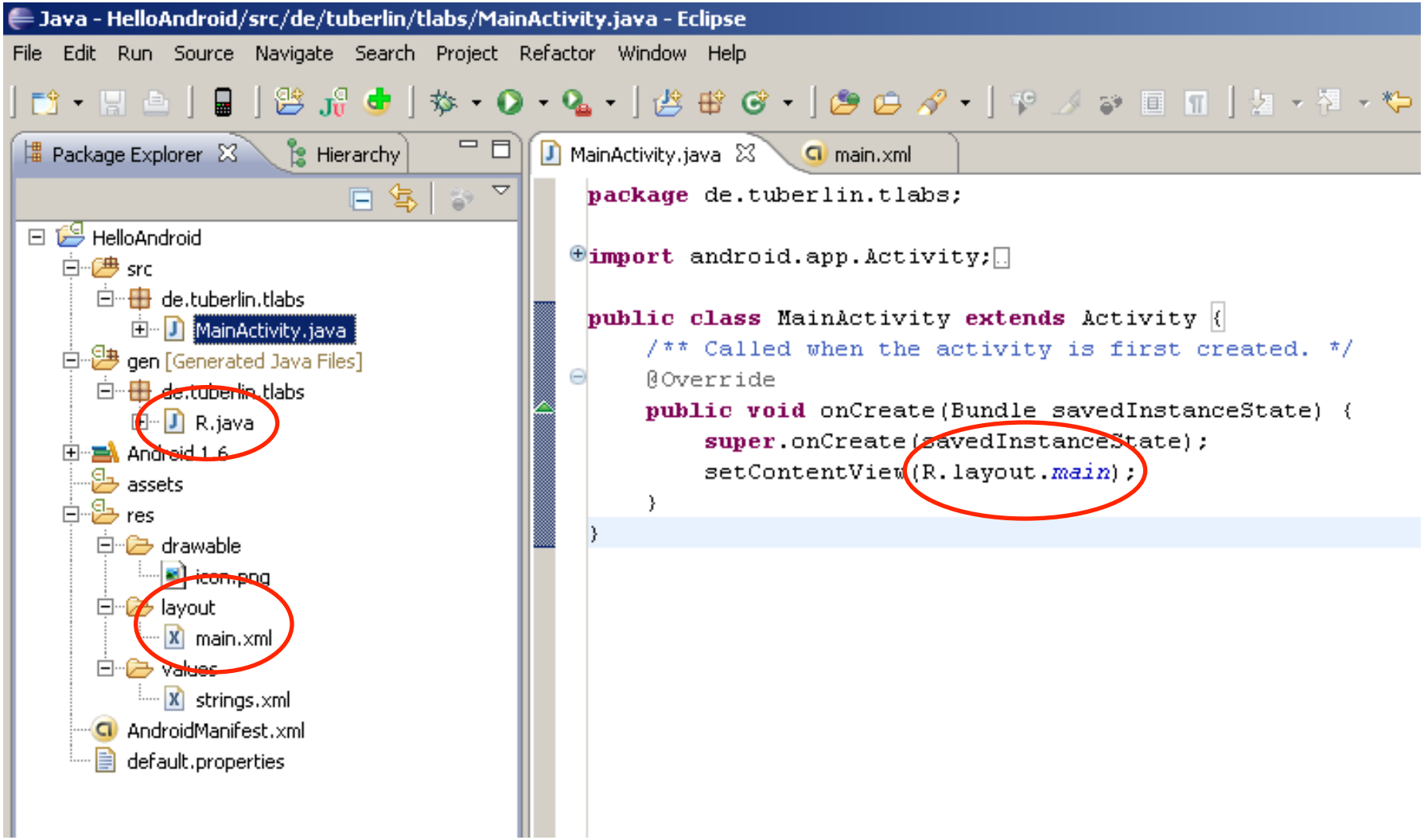
Exercise:

Create “Hello World”

Diese Folien:

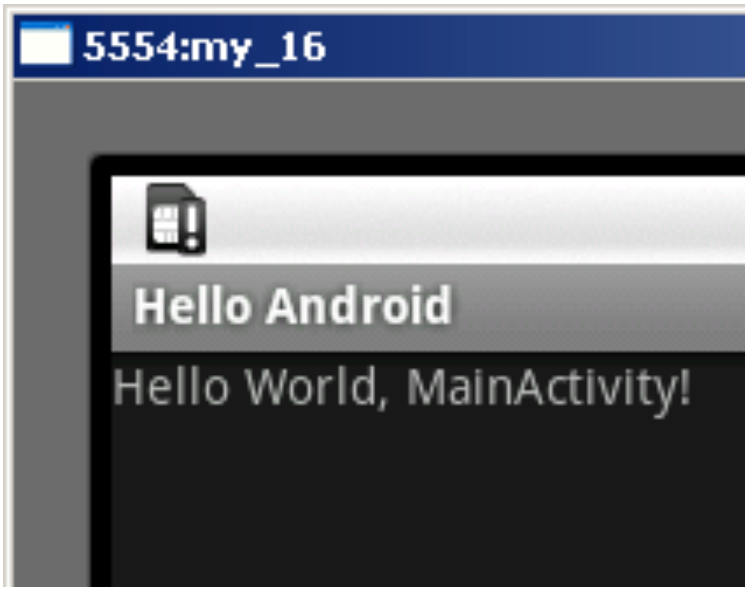
http://tiny.cc/probeandroid2012

http://www.tcs.ifi.lmu.de/schueler-an-der-uni


Java - HelloAndroid/src/de/tuberlin/tlabs/MainActivity.java - Eclipse
File Edit Run Source Navigate Search Project Refactor Window Help
Package Explorer
Hierarchy
HelloAndroid
src
de.tuberlin.tlabs
MainActivity.java
gen [Generated Java Files]
de.tuberlin.tlabs
R.java
Android 1.6
assets
res
drawable
icon.png
layout
main.xml
values
strings.xml
AndroidManifest.xml
default.properties
MainActivity.java
main.xml
package de.tuberlin.tlabs;
import android.app.Activity;
public class MainActivity extends Activity {
/** Called when the activity is first created. */
@Override
public void onCreate(Bundle savedInstanceState) {
super.onCreate(savedInstanceState);
setContentView(R.layout.main);
}
}

Declarative definition of UIs
main.xml

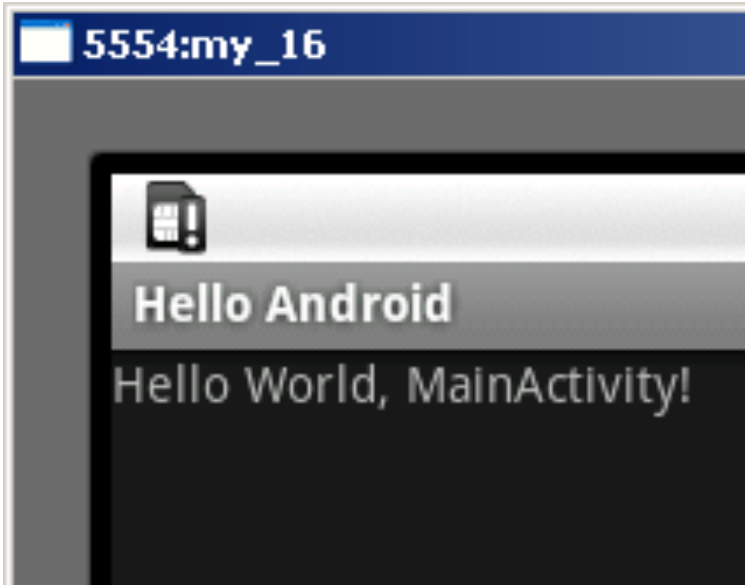
```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:orientation="vertical"
    android:layout_width="fill_parent"
    android:layout_height="fill_parent"
    >
<TextView
    android:layout_width="fill_parent"
    android:layout_height="wrap_content"
    android:text="@string/hello"
    />
</LinearLayout>
```



Separating text strings from source code strings.xml

```
<?xml version="1.0" encoding="utf-8"?>
<resources>
    <string name="hello">Hello World, MainActivity!</string>
    <string name="app_name">Hello Android</string>
</resources>
```

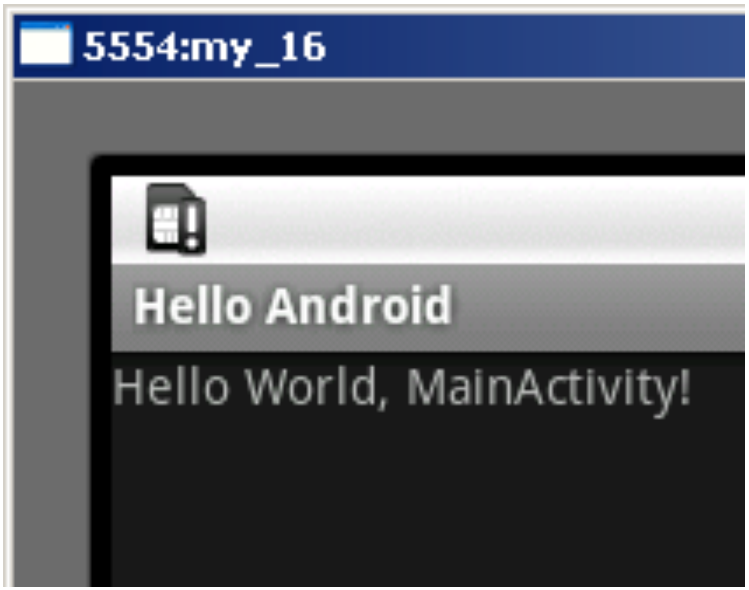
- Default language in res/values/strings.xml
- Localized languages in res/values-xx (language qualifier)
  - French in res/values-fr/strings.xml
  - Hindi in res/values-hi/strings.xml
  - etc.

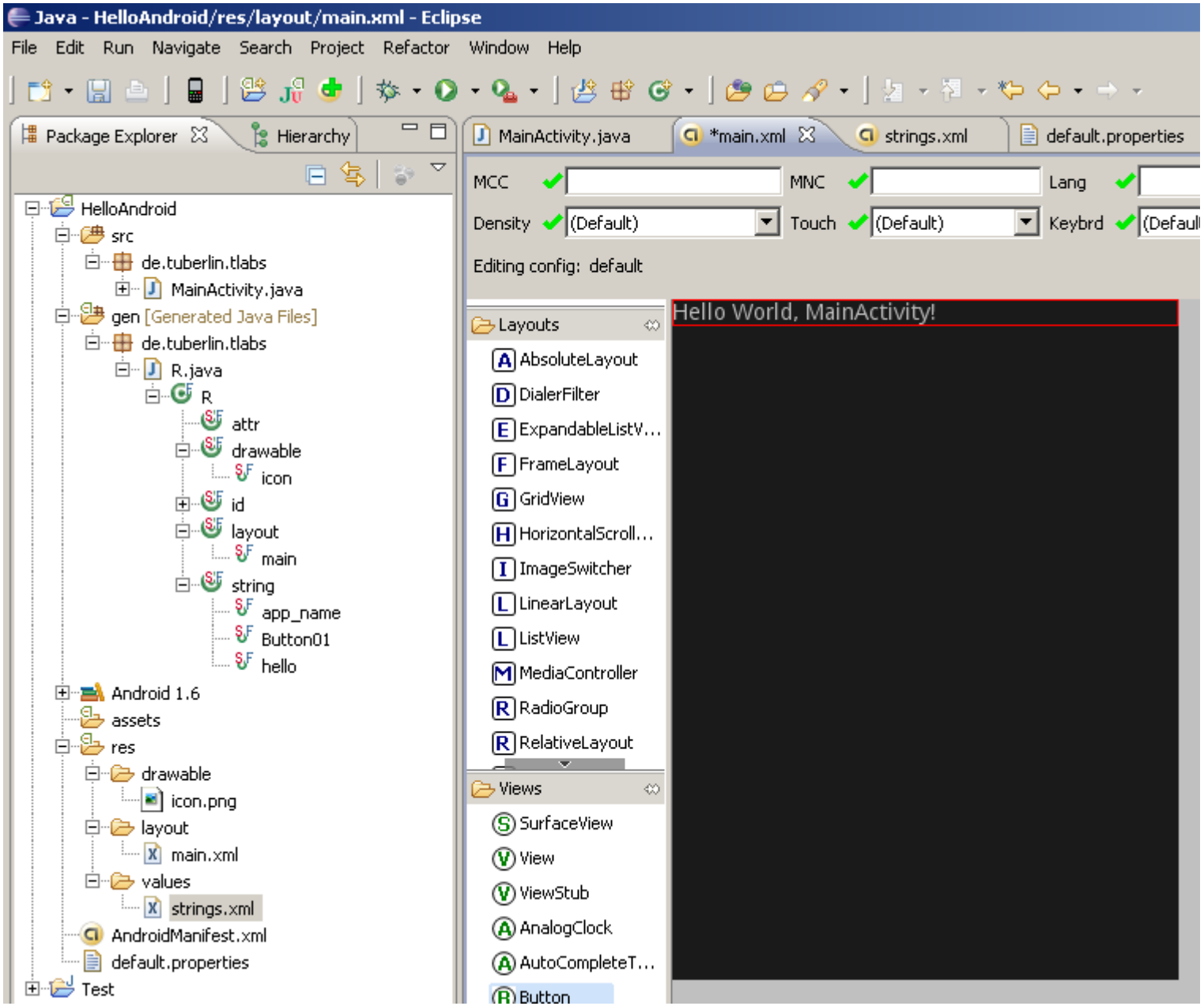


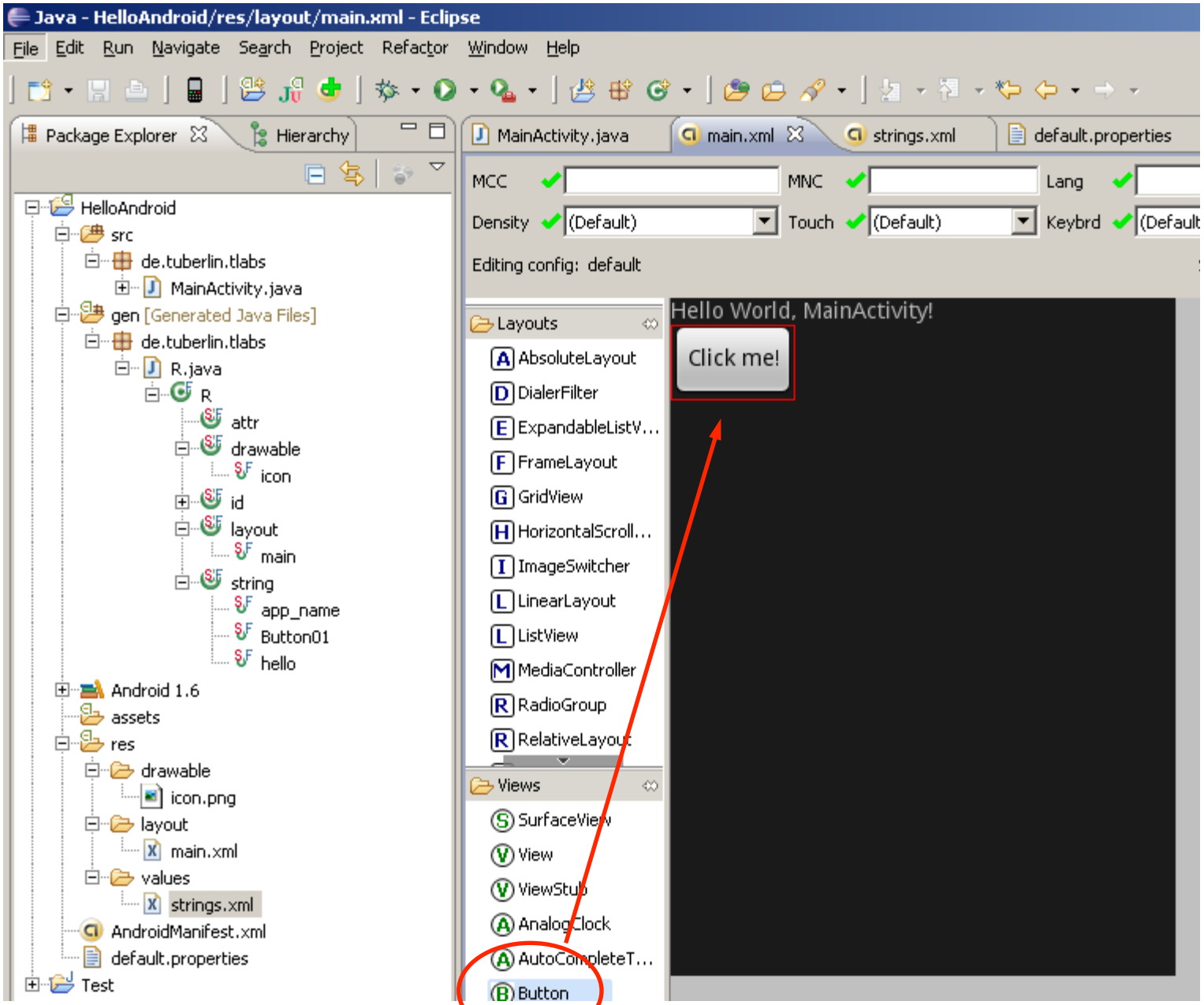
R.java

```
/* AUTO-GENERATED FILE.  DO NOT MODIFY.
 *
 * This class was automatically generated by the
 * aapt tool from the resource data it found.  It
 * should not be modified by hand.
 */

package de.tuberlin.tlabs;

public final class R {
    public static final class attr {
    }
    public static final class drawable {
        public static final int icon=0x7f020000;
    }
    public static final class id {
        public static final int Button01=0x7f050000;
    }
    public static final class layout {
        public static final int main=0x7f030000;
    }
    public static final class string {
        public static final int Button01=0x7f040002;
        public static final int app_name=0x7f040001;
        public static final int hello=0x7f040000;
    }
}
```

Never ever edit R.java!!!




Java - HelloAndroid/res/layout/main.xml - Eclipse
File Edit Run Navigate Search Project Refactor Window Help
Package Explorer
Hierarchy
HelloAndroid
src
de.tuberlin.tlabs
MainActivity.java
gen [Generated Java Files]
de.tuberlin.tlabs
R.java
R
attr
drawable
icon
id
layout
main
string
app_name
Button01
hello
Android 1.6
assets
res
drawable
icon.png
layout
main.xml
values
strings.xml
AndroidManifest.xml
default.properties
Test
MainActivity.java
*main.xml
strings.xml
default.properties
MCC
MNC
Lang
Density
(Default)
Touch
(Default)
Keybrd
Editing config: default
Layouts
AbsoluteLayout
DialerFilter
ExpandableListV...
FrameLayout
GridView
HorizontalScroll...
ImageSwitcher
LinearLayout
ListView
MediaController
RadioGroup
RelativeLayout
Views
SurfaceView
View
ViewStub
AnalogClock
AutoCompleteT...
Button
Hello World, MainActivity!

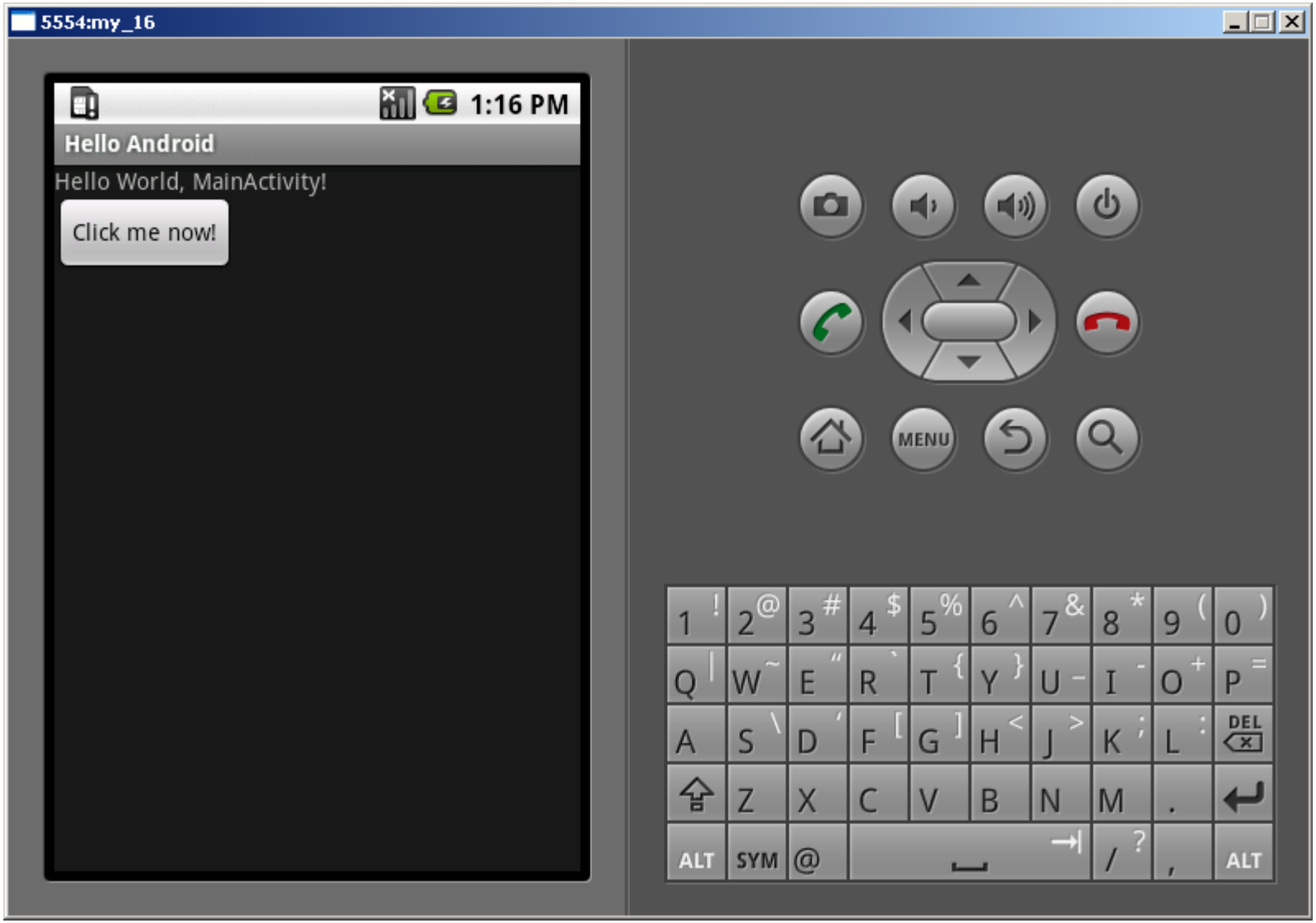

Java - HelloAndroid/res/layout/main.xml - Eclipse
File Edit Run Navigate Search Project Refactor Window Help
Package Explorer
Hierarchy
MainActivity.java
main.xml
strings.xml
default.properties
HelloAndroid
src
de.tuberlin.tlabs
MainActivity.java
gen [Generated Java Files]
de.tuberlin.tlabs
R.java
R
attr
drawable
icon
id
layout
main
string
app_name
Button01
hello
Android 1.6
assets
res
drawable
icon.png
layout
main.xml
values
strings.xml
AndroidManifest.xml
default.properties
Test
MCC
MNC
Lang
Density
(Default)
Touch
(Default)
Keybrd
Editing config: default
Layouts
AbsoluteLayout
DialerFilter
ExpandableListV...
FrameLayout
GridView
HorizontalScroll...
ImageSwitcher
LinearLayout
ListView
MediaController
RadioGroup
RelativeLayout
Views
SurfaceView
View
ViewStub
AnalogClock
AutoCompleteT...
Button
Hello World, MainActivity!
Click me!

Declarative Definition of UIs
main.xml

```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:orientation="vertical"
    android:layout_width="fill_parent"
    android:layout_height="fill_parent"
    >
<TextView
    android:layout_width="fill_parent"
    android:layout_height="wrap_content"
    android:text="@string/hello"
    />
<Button
    android:text="@string/button1"
    android:id="@+id/button1"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    />
</LinearLayout>
```

strings.xml

```
<?xml version="1.0" encoding="utf-8"?>
<resources>
    <string name="hello">Hello World, MainActivity!</string>
    <string name="app_name">Hello Android</string>
    <string name=”button1">Click me!</string>
</resources>
```


5554:my_16
1:16 PM
Hello Android
Hello World, MainActivity!
Click me now!
MENU
ALT
SYM
ALT

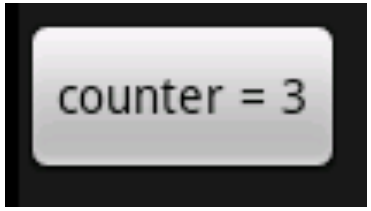
Handling Button Click Events

- XML

```
<Button android:id="@+id/button1" android:text="Click me!"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content" />
```

- Java

```
public class MainActivity extends Activity implements
                                                View.OnClickListener {
    public void onCreate(Bundle savedInstanceState) {
        ...
        Button b = (Button) findViewById(R.id.button1);
        b.setOnClickListener(this);
    }

    private int counter = 0;

    public void onClick(View v) {
        Button b = (Button)v;
        b.setText("counter = " + (++counter));
    }
}
```



Exercise:

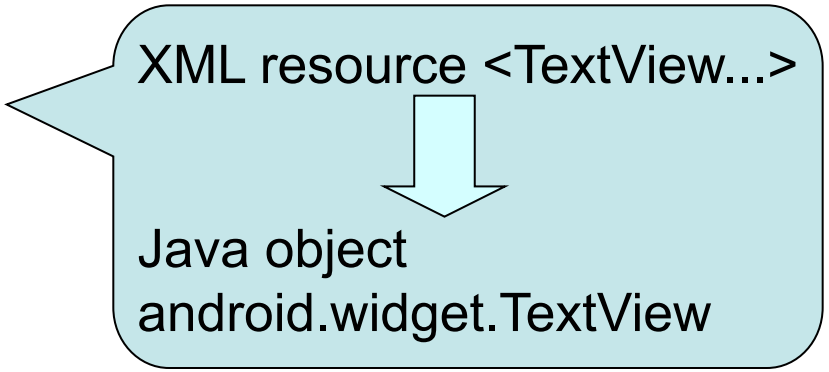
- Add a button to “Hello World”
- Handle button click events

UI from XML resources MainActivity.java

```
import android.app.Activity;
import android.os.Bundle;

public class MainActivity extends Activity {

    public void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.main);
    }
}
```

UI programmatically defined MainActivity.java

```
import android.app.Activity;
import android.os.Bundle;
import android.widget.TextView;

public class MainActivity extends Activity {

    public void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
//        setContentView(R.layout.main);
        TextView tv = new TextView(this);
        tv.setText("Hello World (TextView)!");
        setContentView(tv);
    }

}
```



Concepts so far

- Project directory structure
  - src, gen, res, AndroidManifest.xml
- Resources
  - Declarative view definitions in XML
  - Localization of string resources
  - Resource identifiers

Activities

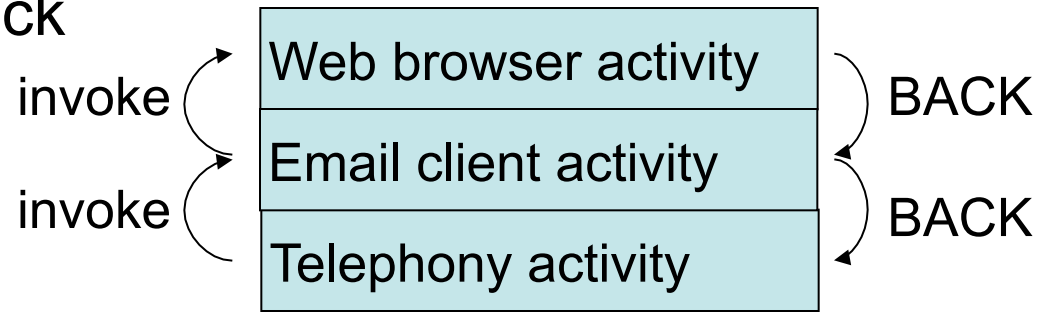
- Independent components of the application
  - Components “crash” individually
- Represent data and behavior of one View
  - Roughly: the model and controller of the MVC pattern
- Example: text messaging application
  - Activity 1 shows list of contacts
  - Activity 2 to write a message to a chosen contact
  - Activity 3 to review sent messages
- View of an Activity typically fills the screen
  - Views grouped in hierarchy
  - Parents control layout of children
  - Leaf view react to user actions
  - Associate root view with activity: activity.setContentView(view id);

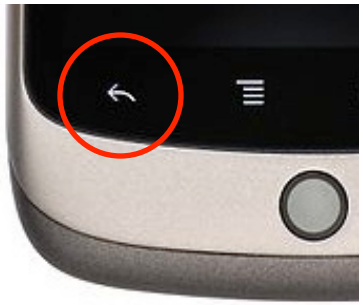
Activity Lifecycle

- Managed by system based on resources and user needs
- States
  - Running: in foreground (at top of activity stack)
  - Paused: partially visible, lost focus (e.g. dialog on top)
  - Stopped: invisible
- Lifecycle callback methods of an Activity
  - **protected void** onCreate(Bundle savedInstanceState);
  - **protected void** onStart();
  - **protected void** onRestart();
  - **protected void** onResume();
  - **protected void** onPause();
  - **protected void** onStop();
  - **protected void** onDestroy();

Tasks

- Task: what the user experiences as an “application”
  - Notion of an “application” blurry in component-based system
  - Tasks can span multiple activities and applications
- Example scenario for a task
  - User talks on the phone, looks up an email to answer a question, follows a link to a Web page with the desired information
  - Talk on phone: telephony application
  - Look up email: email client
  - Reading Web page: web browser
- Activity stack of a task:





Intents

- Intents are
  - Messages to the system
  - (Passive) representations of an operation to be performed
  - “Glue” between activities
  - Enable late runtime binding across applications
- Primary pieces: action and data
  - Example: action: ACTION_VIEW, data: URI to view
- Intents used to
  - Invoke other applications
  - Represent actions to be performed in the future
  - Register for events (→ publish-and-subscribe)

Example: Invoking an Activity

- Activity to be invoked

```
public class BasicActivity extends Activity {
    public void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.main);
}   }
```

- In AndroidManifest.xml

```
<activity android:name="BasicActivity" android:label="My Basic Activity">
    <intent-filter>
        <action android:name="de.lmu.intent.action.ShowBasicView" />
        <category android:name="android.intent.category.DEFAULT" />
    </intent-filter>
</activity>
```

- From another activity

```
Intent intent = new Intent("de.lmu.intent.action.ShowBasicView");
startActivity(intent);
```

Available Intents in Android

- Available intents
  - Browser: open a browser window
  - Dialer: calling phone numbers
  - Google Maps: open to the given location
  - Google Streetview: open to the given location

- Examples

```
Intent intent = new Intent(Intent.ACTION_VIEW);
intent.setData(Uri.parse("http://www.lmu.de"));
startActivity(intent);

Intent intent = new Intent(Intent.ACTION_VIEW);
intent.setData(Uri.parse("geo:52.5127,13.3210?z=17"));
startActivity(intent);
```

Define the contents of the application AndroidManifest.xml

```
<?xml version="1.0" encoding="utf-8"?>
<manifest xmlns:android="http://schemas.android.com/apk/res/android"
    package="de.tuberlin.tlabs"
    android:versionCode="1"
    android:versionName="1.0">
  <application android:icon="@drawable/icon" android:label="@string/app_name">
    <activity android:name=".MainActivity“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>
  <uses-sdk android:minSdkVersion="4" />
</manifest>
```

Uniquely identifies the application!

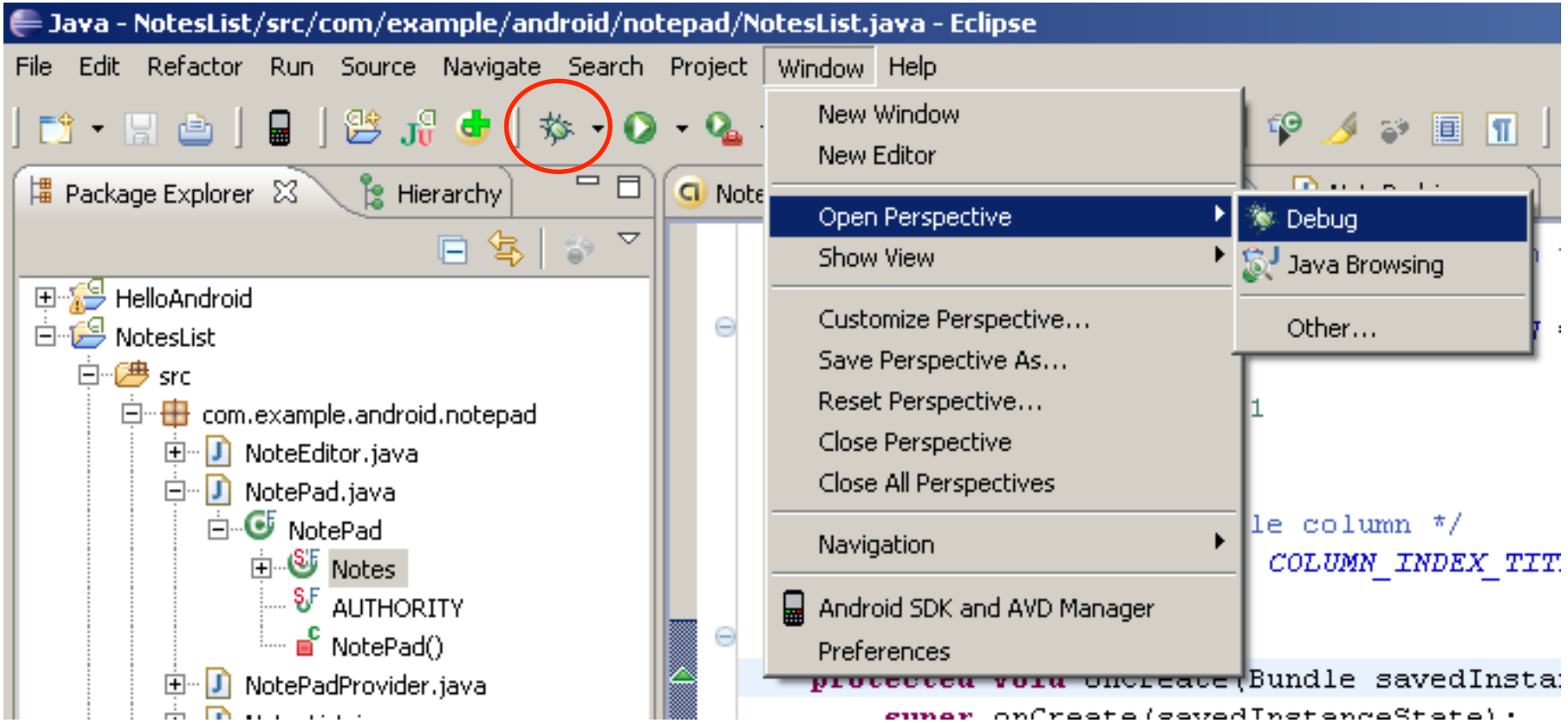
Add for android:debuggable="true" on-device debugging!

- Initial activity of application
- Listed in application launcher

Integrated Development Environment (IDE)

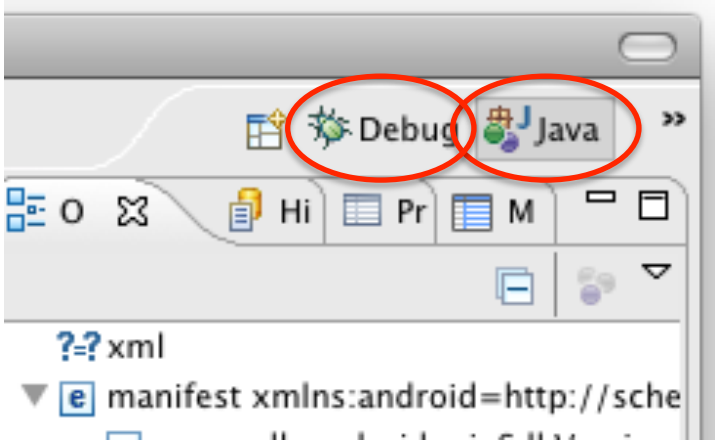
ECLIPSE

Eclipse Perspectives



Eclipse Perspectives

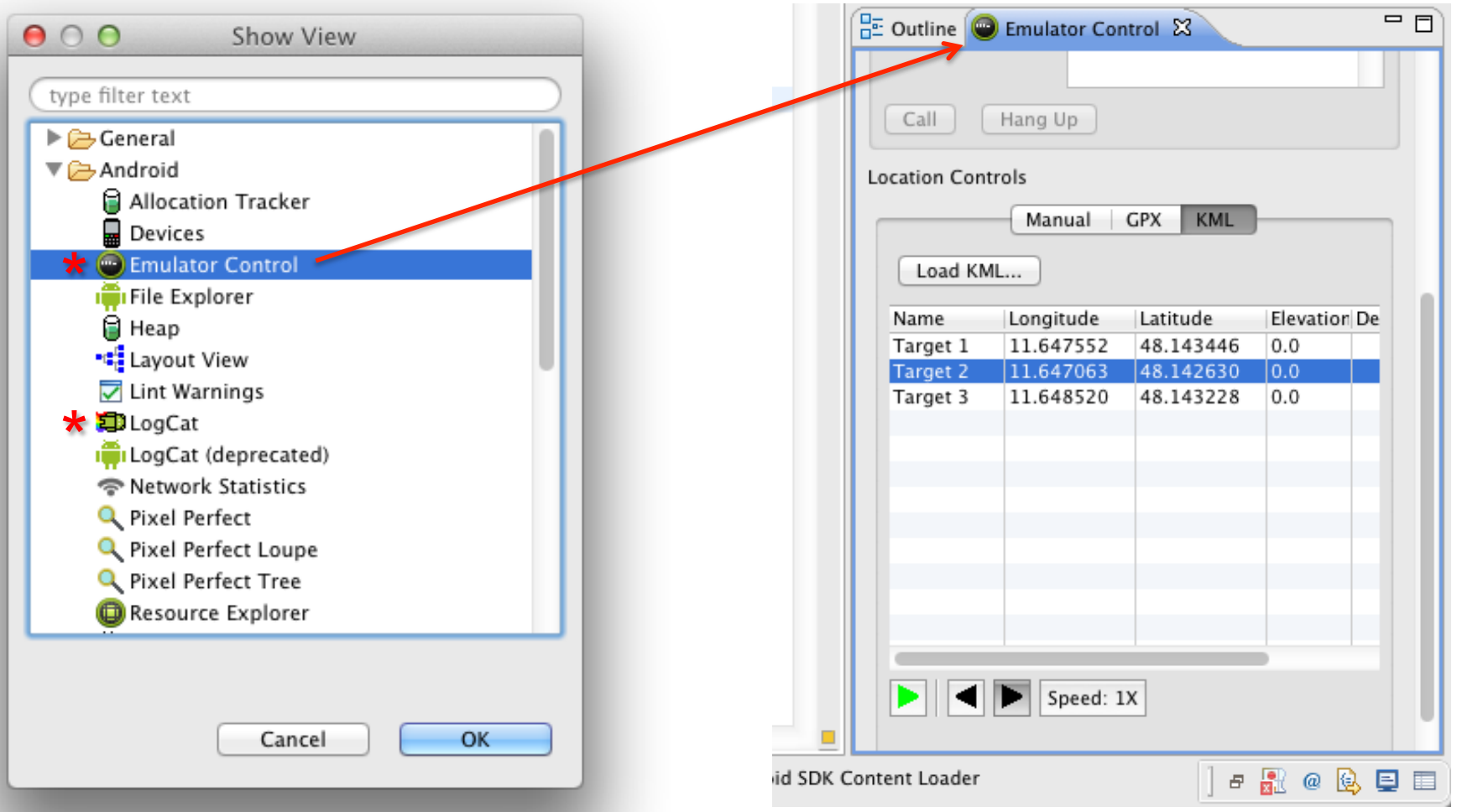
- Java Perspective
  - Writing source code
  - Adding resources
- Debug Perspective
  - Setting breakpoints
  - Inspecting variables



Eclipse tips:
Ctrl + Shift + O: organize imports
Ctrl + Space: show completions
F3: go to definition (e.g. of a class or method)

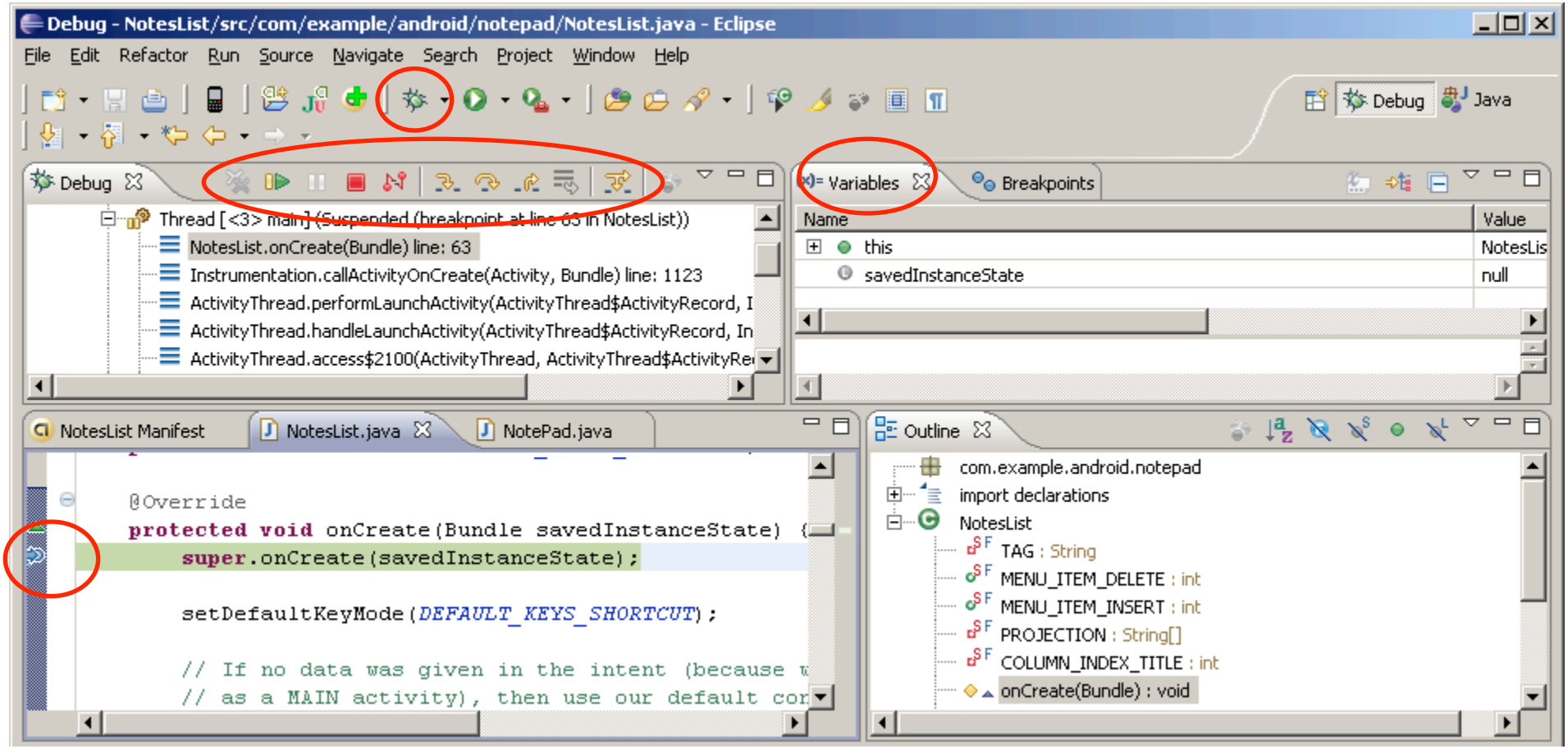
Eclipse Views

- Window → Show View → Other… → Android

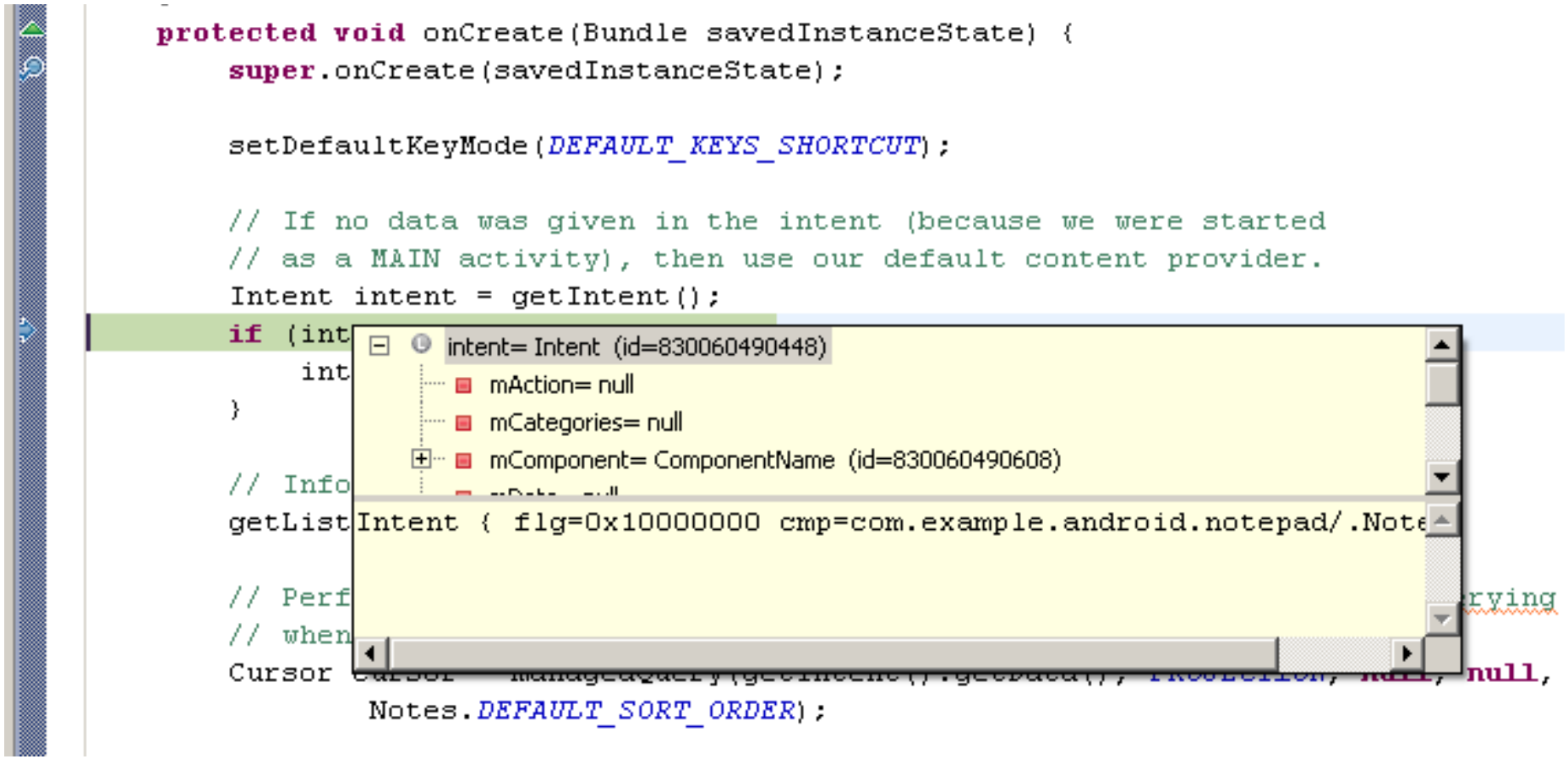


Debugging in the Emulator

- Set Breakpoint with Ctrl+Shift+B (⌘ +Shift+B)
- Step through code with F5, F6, F7 (*fn* + F5, F6, F7)



Inspecting Variables

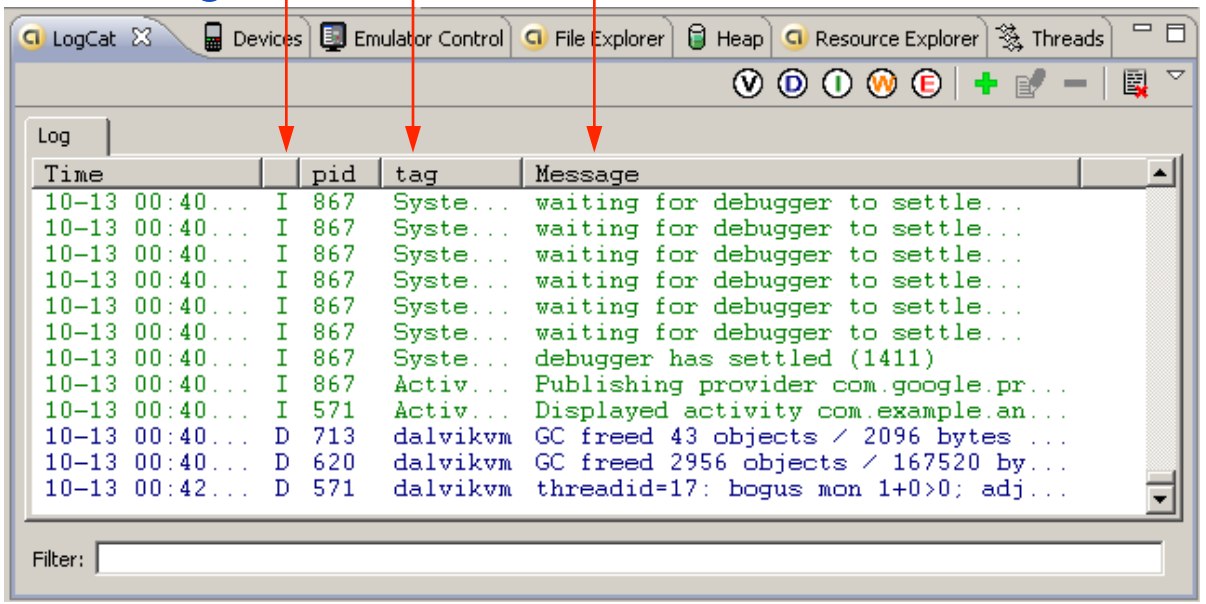


Logging and Tracing

- android.util.Log
  - informational, warning, error methods
  - Example:

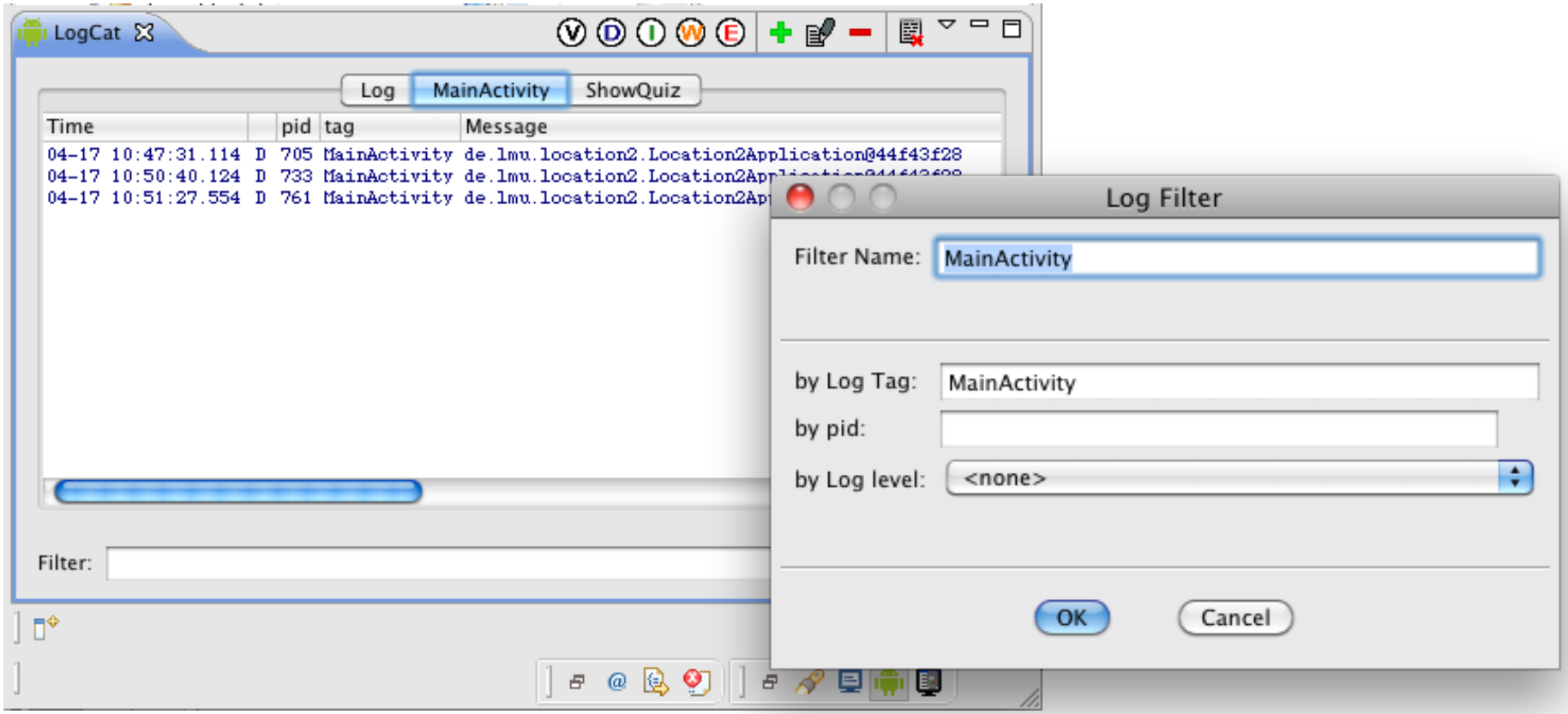
Log.d(TAG, "getAddress: " + s);

- android.os.Debug
  - Debug.startMethodTracing
  - Debug.stopMethodTracing
  - trace viewer tool
- File explorer tool to view files on the device



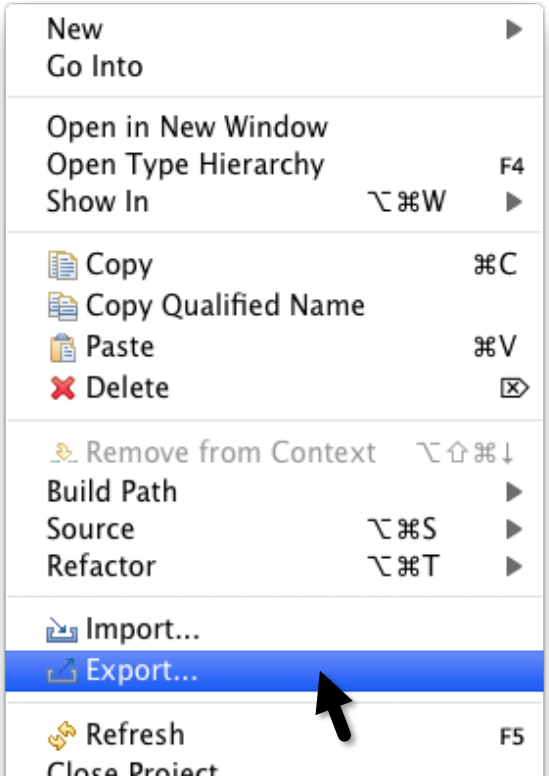
Filtering Eclipse Debug Output

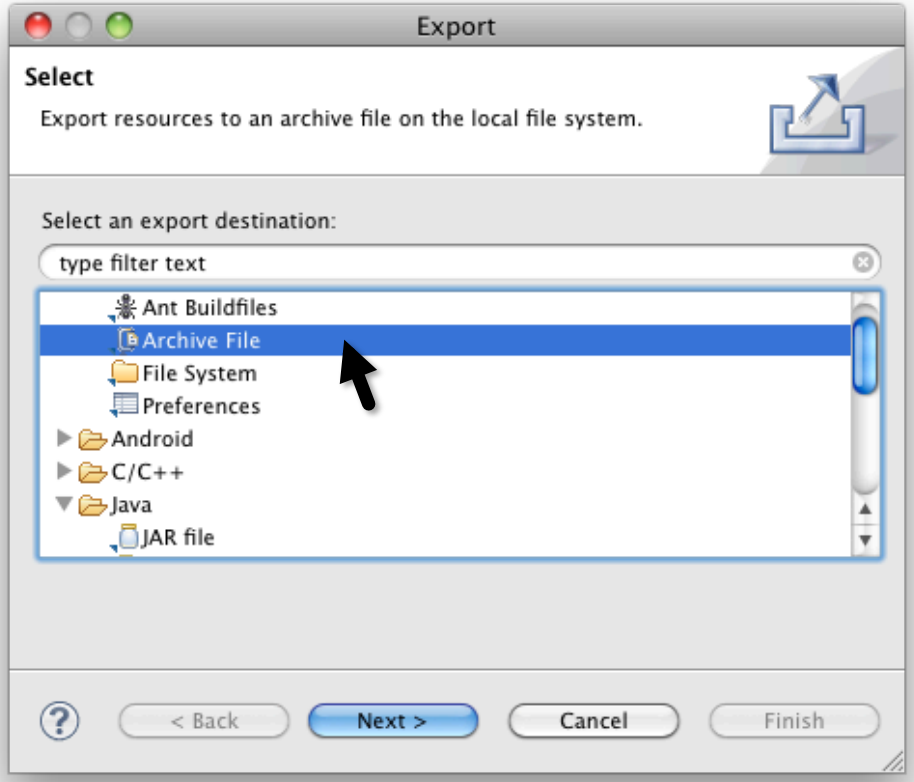
Log.*d*("MainActivity", "onCreate");



Exportieren / Importieren von Projekten

- Android-Projekte exportieren
  - Eclipse → File → Export → General → Archive File (zip)



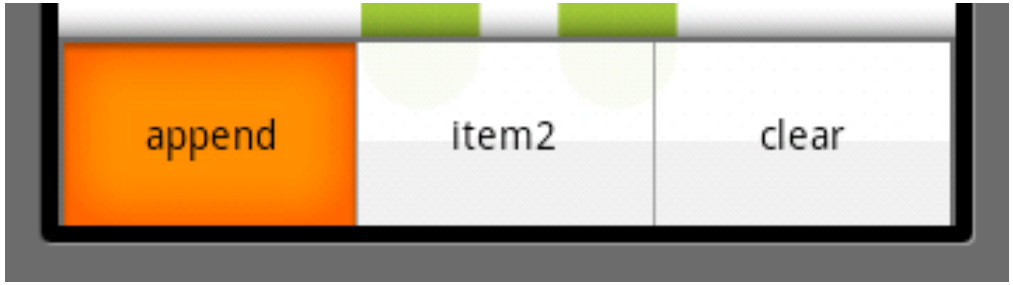


MENUS

Menus

- An activity is associated with a single menu
- Use onCreateOptionsMenu(Menu m) to populate menu
- Creating an options menu

```
public boolean onCreateOptionsMenu(Menu menu) {
    super.onCreateOptionsMenu(menu);
    menu.add(0, 1, 0, "append"); // group, id, order, title
    menu.add(0, 2, 1, "item2");
    menu.add(0, 3, 2, "clear");
    return true; // return true to enable menu
}
```



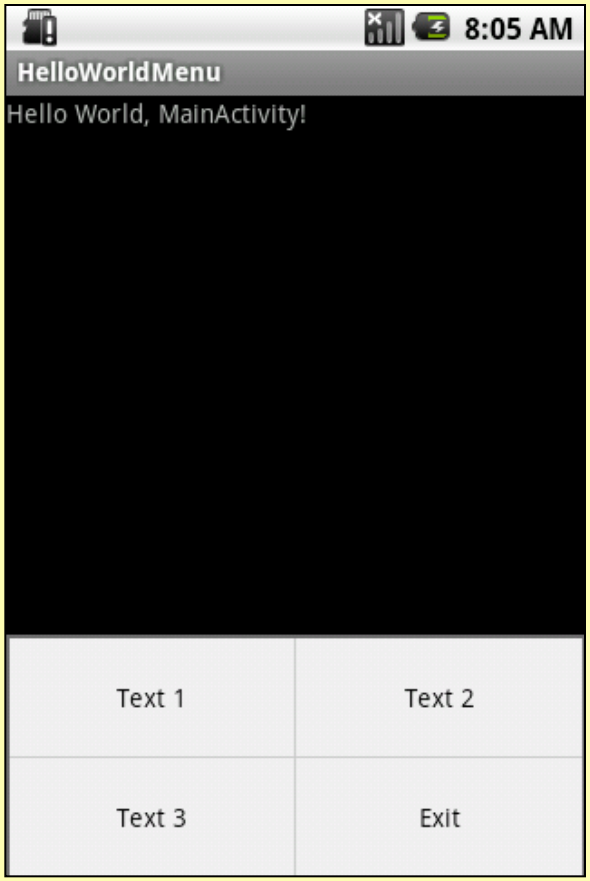
Responding to Menu Selection

- Overriding onOptionsItemSelected

```
public boolean onOptionsItemSelected(MenuItem item) {
    Log.d("MainActivity", "menu id = " + item.getItemId() +
            ", title = " + item.getTitle().toString());
    switch (item.getItemId()) {
        case X: // id of handeled item
        // handle item X
        return true;
        ...
    }
}
```

Exercise: A Menu for “Hello Android”

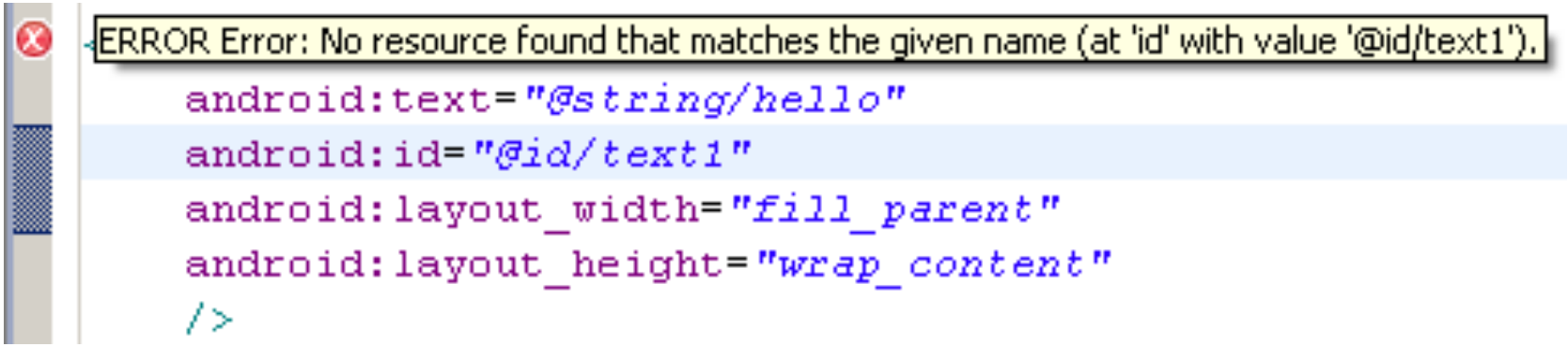
- Add a menu with four items
- Menu items 1-3 changes text shown in the top of the display
  - setText(...)
- Menu item 1 → Probestudium
- Menu item 2 → LMU
- Menu item 3 → Android
- Menu item 4: Exit the application
  - finish()



RESOURCES

Resource-Reference Syntax

- “+” Use id if it already exists, otherwise create new id
- @id/text1



- @+id/text1

```
<TextView
    android:text="@string/hello"
    android:id="@+id/text1"
    android:layout_width="fill_parent"
    android:layout_height="wrap_content"
    />
<Button
```

Image Resources

- Automatic id generation for images in /res/drawable
  - Example: /res/drawable/sample_image.jpg → R.drawable.sample_image
- Supported types: .gif, .jpg, .png

- Usage in Java

```
Button b = (Button)this.findViewById(R.id.Button01);
b.setBackgroundResource(R.drawable.sample_image);
```

- Usage in XML

```
<Button android:text="@string/Button01"
…
android:background="@drawable/sample_image" />
```

UI COMPONENTS

Core UI Component Classes

java.lang.Object
↑ android.view.View
↑ android.view.ViewGroup
↑ android.widget.LinearLayout

- android.view.View
  - Rectangular area on the screen
  - Responsible for drawing and event handling
  - Base class for widgets (buttons, text fields, etc.)
- android.view.ViewGroup
  - Is a view and contains other views (“container”)
  - Base class for layouts
- Layouts
  - Invisible containers that hold other Views
  - Define their layout properties (position, padding, size, etc.)
  - Example: LinearLayout (horizontal / vertical list of children)

Design UI in XML, Reference in Java

- Assign IDs in XML

```
<TextView android:id="@+id/nameValue" .../>
<TextView android:id="@+id/addrValue" ... />
```

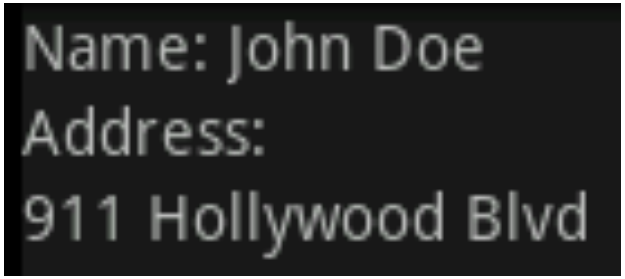
- Refer to controls using IDs

```
TextView nameValue = (TextView) findViewById(R.id.nameValue);
nameValue.setText("John Doe");
TextView addrValue = (TextView)findViewById(R.id.addrValue);
addrValue.setText("911 Hollywood Blvd.");
```

- View must have been loaded before referencing IDs

```
setContentView(R.layout.test);
```

Creating a UI in XML (/res/layout/test.xml)




```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:orientation="vertical" android:layout_width="fill_parent"
    android:layout_height="fill_parent">
  <LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
      android:orientation="horizontal" android:layout_width="fill_parent"
      android:layout_height="wrap_content">
    <TextView android:layout_width="wrap_content"
      android:layout_height="wrap_content" android:text="Name: " />
    <TextView android:layout_width="wrap_content"
      android:layout_height="wrap_content" android:text="John Doe" />
  </LinearLayout>
  <LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:orientation="vertical" android:layout_width="fill_parent"
    android:layout_height="wrap_content">
  <TextView android:layout_width="fill_parent"
    android:layout_height="wrap_content" android:text="Address:" />
  <TextView android:layout_width="fill_parent"
    android:layout_height="wrap_content" android:text="911 Hollywood Blvd." />
  </LinearLayout>
</LinearLayout>
```

Setting the XML UI in Java

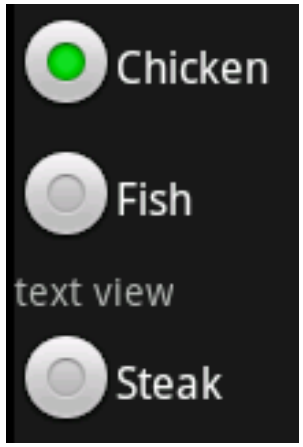
```
public class MainActivity extends Activity {
    public void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.test);
    }
}
```

Radio Button

- XML

```
<LinearLayout android:orientation="vertical"
        android:layout_width="wrap_content“
        android:layout_height="wrap_content">
    <RadioGroup android:layout_width="wrap_content“
        android:layout_height="wrap_content">
        <RadioButton android:text="Chicken"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content" />
        <RadioButton android:text="Fish"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content" />
        ...
    </RadioGroup>
</LinearLayout>
```

- Radio groups can contain arbitrary views



THE LOCATION QUIZ

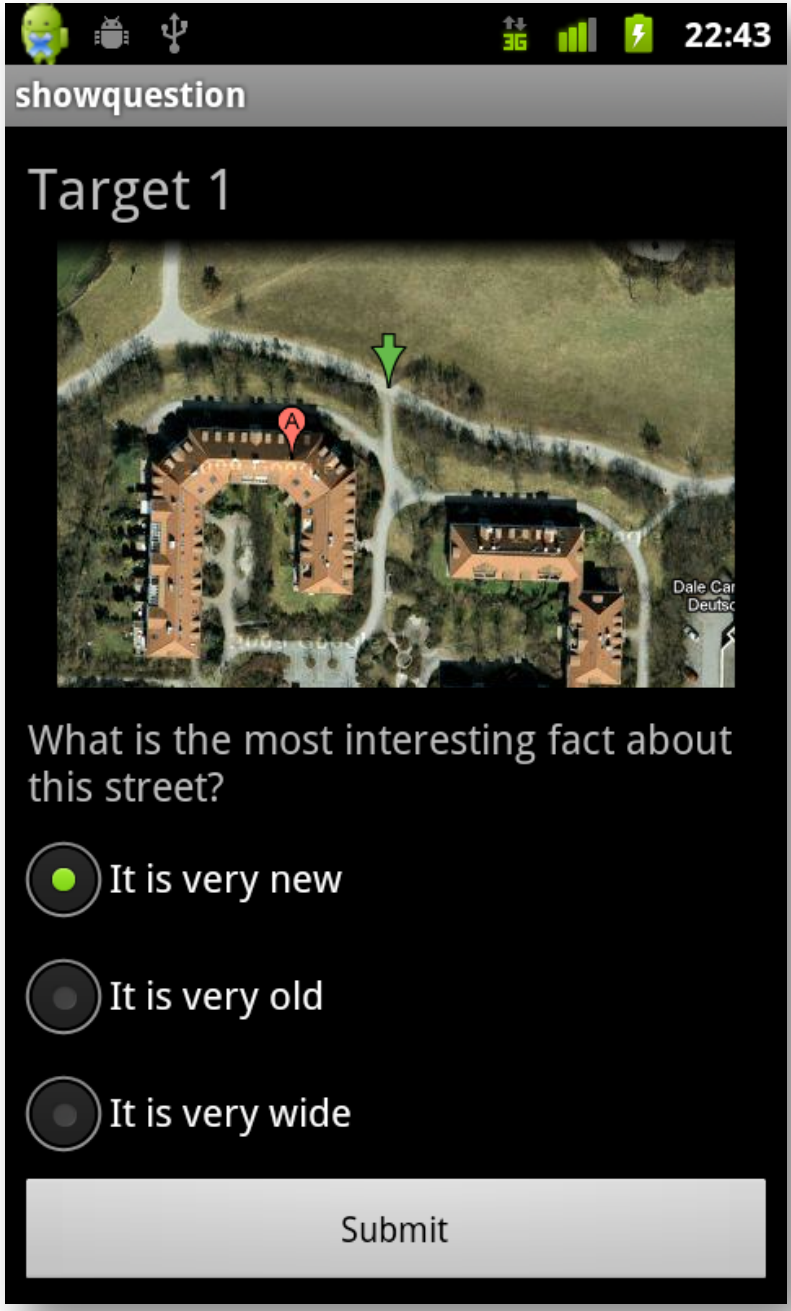
The Main Screen → MainActivity

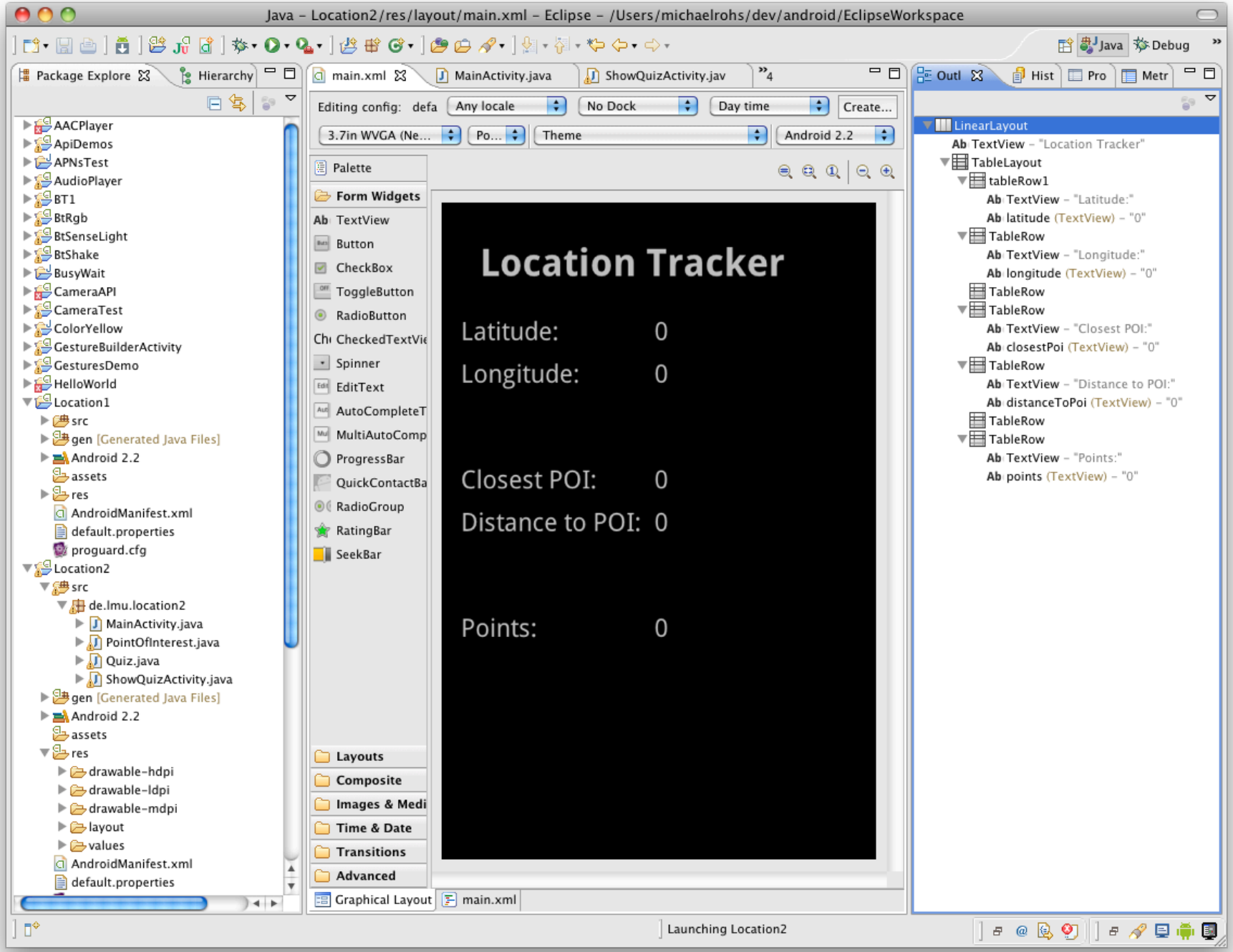
- Show current location
- Show nearest point-of-interest
- Show number of game points

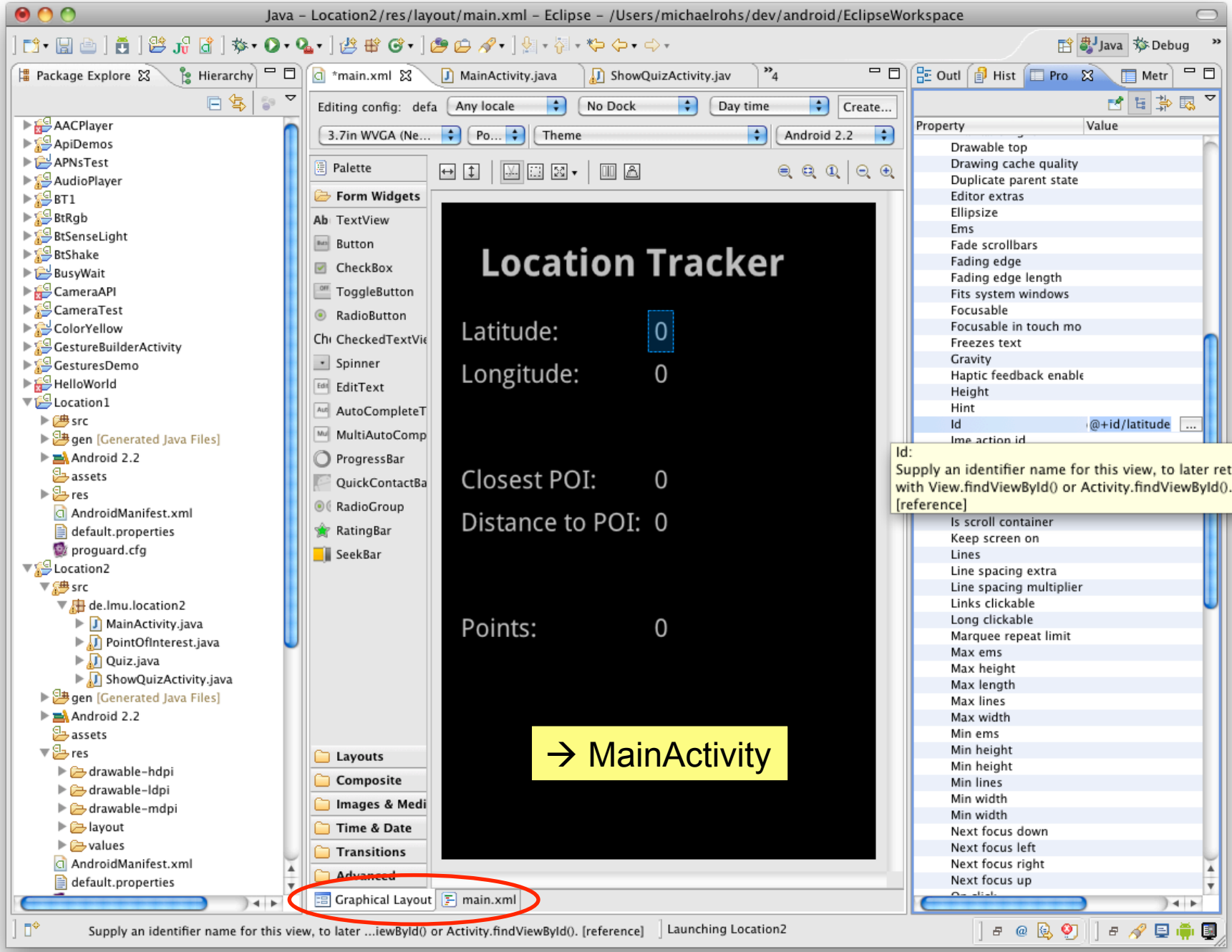
The Quiz Screen → ShowQuizActivity

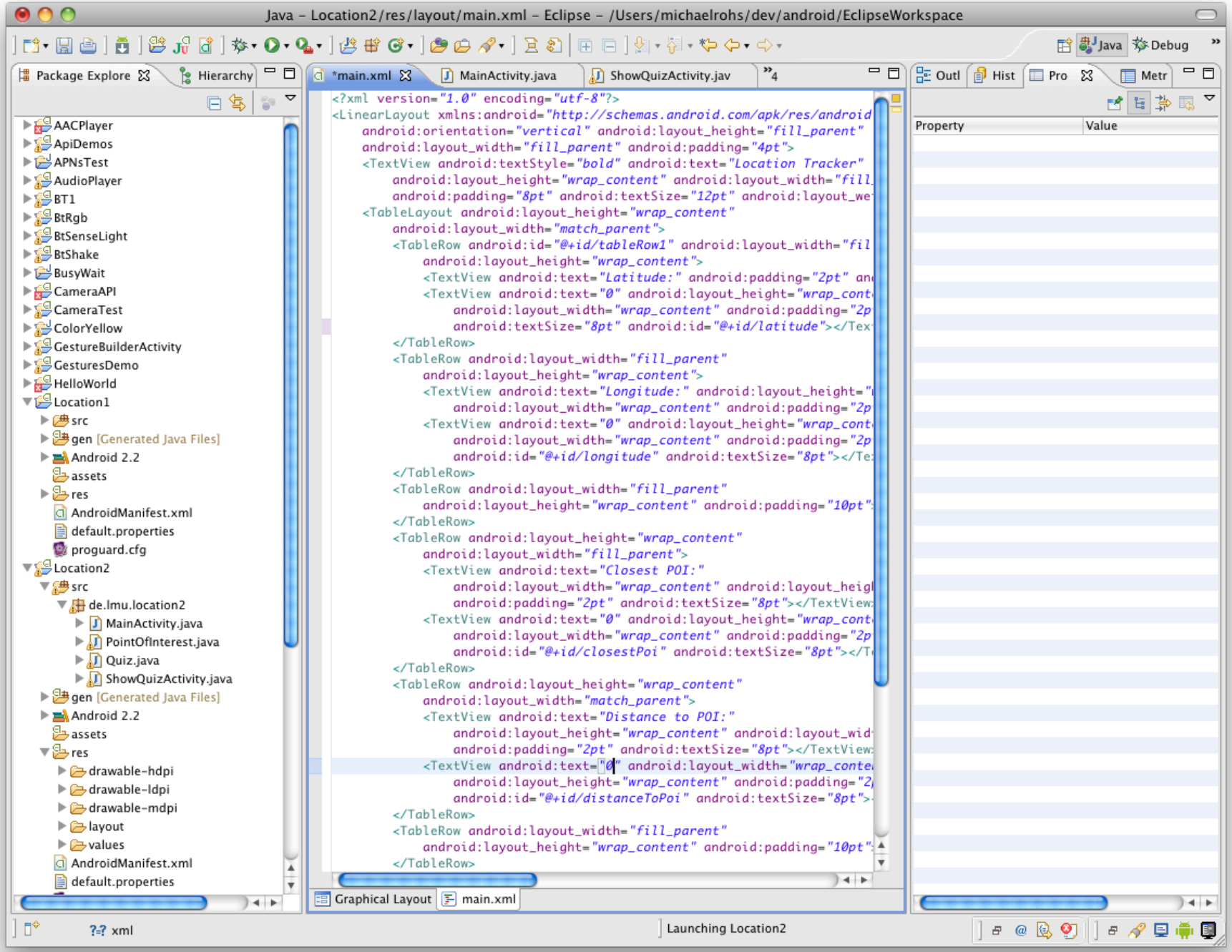
- Title of the point-of-interest
- Image of the POI
- Question
- Choices

  (the correct one gives positive, the wrong one negative points)
- Submit button



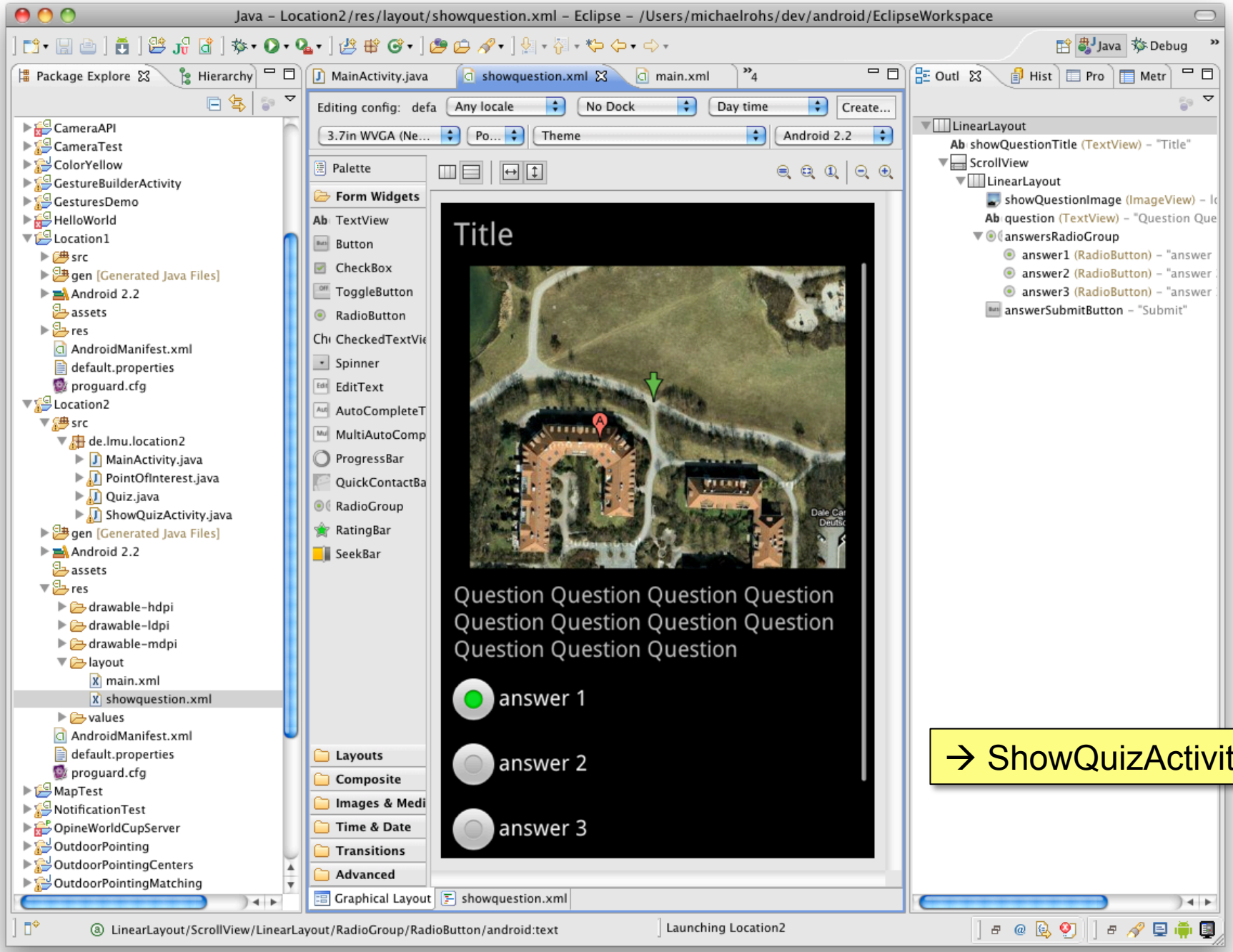

Java – Location2/res/layout/main.xml – Eclipse – /Users/michaelrohs/dev/android/EclipseWorkspace
Package Explore
Hierarchy
AACPlayer
ApiDemos
APNsTest
AudioPlayer
BT1
BtRgb
BtSenseLight
BtShake
BusyWait
CameraAPI
CameraTest
ColorYellow
GestureBuilderActivity
GesturesDemo
HelloWorld
Location1
src
gen [Generated Java Files]
Android 2.2
assets
res
AndroidManifest.xml
default.properties
proguard.cfg
Location2
src
de.lmu.location2
MainActivity.java
PointOfInterest.java
Quiz.java
ShowQuizActivity.java
gen [Generated Java Files]
Android 2.2
assets
res
drawable-hdpi
drawable-ldpi
drawable-mdpi
layout
values
AndroidManifest.xml
default.properties
main.xml
MainActivity.java
ShowQuizActivity.jav
Editing config: defa
Any locale
No Dock
Day time
Create...
3.7in WVGA (Ne...
Po...
Theme
Android 2.2
Palette
Form Widgets
TextView
Button
CheckBox
ToggleButton
RadioButton
CheckedTextVie
Spinner
EditText
AutoCompleteT
MultiAutoComp
ProgressBar
QuickContactBa
RadioGroup
RatingBar
SeekBar
Layouts
Composite
Images & Medi
Time & Date
Transitions
Advanced
Location Tracker
Latitude: 0
Longitude: 0
Closest POI: 0
Distance to POI: 0
Points: 0
Graphical Layout
main.xml
Outl
Hist
Pro
Metr
LinearLayout
TextView – "Location Tracker"
TableLayout
tableRow1
TextView – "Latitude:"
latitude (TextView) – "0"
TableRow
TextView – "Longitude:"
longitude (TextView) – "0"
TableRow
TableRow
TextView – "Closest POI:"
closestPoi (TextView) – "0"
TableRow
TextView – "Distance to POI:"
distanceToPoi (TextView) – "0"
TableRow
TableRow
TextView – "Points:"
points (TextView) – "0"
Launching Location2


Java – Location2/res/layout/main.xml – Eclipse – /Users/michaelrohs/dev/android/EclipseWorkspace
Package Explore
Hierarchy
AACPlayer
ApiDemos
APNsTest
AudioPlayer
BT1
BtRgb
BtSenseLight
BtShake
BusyWait
CameraAPI
CameraTest
ColorYellow
GestureBuilderActivity
GesturesDemo
HelloWorld
Location1
src
gen [Generated Java Files]
Android 2.2
assets
res
AndroidManifest.xml
default.properties
proguard.cfg
Location2
src
de.lmu.location2
MainActivity.java
PointOfInterest.java
Quiz.java
ShowQuizActivity.java
gen [Generated Java Files]
Android 2.2
assets
res
drawable-hdpi
drawable-ldpi
drawable-mdpi
layout
values
AndroidManifest.xml
default.properties
*main.xml
MainActivity.java
ShowQuizActivity.jav
Editing config: defa
Any locale
No Dock
Day time
Create...
3.7in WVGA (Ne...
Po...
Theme
Android 2.2
Palette
Form Widgets
TextView
Button
CheckBox
ToggleButton
RadioButton
CheckedTextVie
Spinner
EditText
AutoCompleteT
MultiAutoComp
ProgressBar
QuickContactBa
RadioGroup
RatingBar
SeekBar
Layouts
Composite
Images & Medi
Time & Date
Transitions
Advanced
Location Tracker
Latitude: 0
Longitude: 0
Closest POI: 0
Distance to POI: 0
Points: 0
→ MainActivity
Graphical Layout
main.xml
Outl
Hist
Pro
Metr
Property
Value
Drawable top
Drawing cache quality
Duplicate parent state
Editor extras
Ellipsize
Ems
Fade scrollbars
Fading edge
Fading edge length
Fits system windows
Focusable
Focusable in touch mo
Freezes text
Gravity
Haptic feedback enable
Height
Hint
Id
@+id/latitude
Ime action id
Id:
Supply an identifier name for this view, to later retrieve it with View.findViewById() or Activity.findViewById(). [reference]
Is scroll container
Keep screen on
Lines
Line spacing extra
Line spacing multiplier
Links clickable
Long clickable
Marquee repeat limit
Max ems
Max height
Max length
Max lines
Max width
Min ems
Min height
Min height
Min lines
Min width
Min width
Next focus down
Next focus left
Next focus right
Next focus up
Supply an identifier name for this view, to later ...iewById() or Activity.findViewById(). [reference]
Launching Location2


Java – Location2/res/layout/main.xml – Eclipse – /Users/michaelrohs/dev/android/EclipseWorkspace
Package Explore
Hierarchy
*main.xml
MainActivity.java
ShowQuizActivity.jav
Outl
Hist
Pro
Metr
Property
Value
AACPlayer
ApiDemos
APNsTest
AudioPlayer
BT1
BtRgb
BtSenseLight
BtShake
BusyWait
CameraAPI
CameraTest
ColorYellow
GestureBuilderActivity
GesturesDemo
HelloWorld
Location1
src
gen [Generated Java Files]
Android 2.2
assets
res
AndroidManifest.xml
default.properties
proguard.cfg
Location2
src
de.lmu.location2
MainActivity.java
PointOfInterest.java
Quiz.java
ShowQuizActivity.java
gen [Generated Java Files]
Android 2.2
assets
res
drawable-hdpi
drawable-ldpi
drawable-mdpi
layout
values
AndroidManifest.xml
default.properties
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android
android:orientation="vertical" android:layout_height="fill_parent"
android:layout_width="fill_parent" android:padding="4pt">
<TextView android:textStyle="bold" android:text="Location Tracker"
android:layout_height="wrap_content" android:layout_width="fill
android:padding="8pt" android:textSize="12pt" android:layout_we
<TableLayout android:layout_height="wrap_content"
android:layout_width="match_parent">
<TableRow android:id="@+id/tableRow1" android:layout_width="fil
android:layout_height="wrap_content">
<TextView android:text="Latitude:" android:padding="2pt" an
<TextView android:text="0" android:layout_height="wrap_cont
android:layout_width="wrap_content" android:padding="2p
android:textSize="8pt" android:id="@+id/latitude"></Tex
</TableRow>
<TableRow android:layout_width="fill_parent"
android:layout_height="wrap_content">
<TextView android:text="Longitude:" android:layout_height="
android:layout_width="wrap_content" android:padding="2p
<TextView android:text="0" android:layout_height="wrap_cont
android:layout_width="wrap_content" android:padding="2p
android:id="@+id/longitude" android:textSize="8pt"></Te
</TableRow>
<TableRow android:layout_width="fill_parent"
android:layout_height="wrap_content" android:padding="10pt"
</TableRow>
<TableRow android:layout_height="wrap_content"
android:layout_width="fill_parent">
<TextView android:text="Closest POI:"
android:layout_width="wrap_content" android:layout_heigh
android:padding="2pt" android:textSize="8pt"></TextView
<TextView android:text="0" android:layout_height="wrap_cont
android:layout_width="wrap_content" android:padding="2p
android:id="@+id/closestPoi" android:textSize="8pt"></T
</TableRow>
<TableRow android:layout_height="wrap_content"
android:layout_width="match_parent">
<TextView android:text="Distance to POI:"
android:layout_height="wrap_content" android:layout_wid
android:padding="2pt" android:textSize="8pt"></TextView
<TextView android:text="0" android:layout_width="wrap_conte
android:layout_height="wrap_content" android:padding="2
android:id="@+id/distanceToPoi" android:textSize="8pt">
</TableRow>
<TableRow android:layout_width="fill_parent"
android:layout_height="wrap_content" android:padding="10pt"
</TableRow>
Graphical Layout
main.xml
?=? xml
Launching Location2

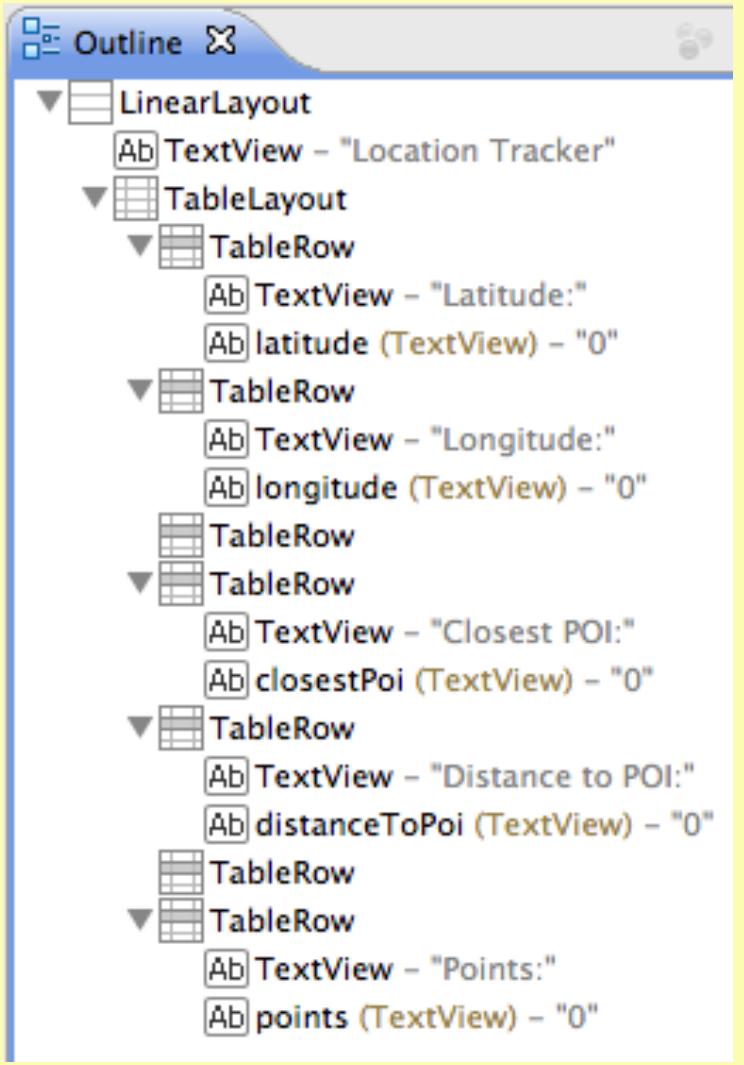
Accessing GUI Elements in Java

```
public class MainActivity extends Activity implements LocationListener {
    TextView latitudeView;
    TextView longitudeView;
    …
    public void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.main);

        latitudeView = (TextView) findViewById(R.id.latitude);
        longitudeView = (TextView) findViewById(R.id.longitude);
        closestPoiView = (TextView) findViewById(R.id.closestPoi);
      …
    }
```


Java – Location2/res/layout/showquestion.xml – Eclipse – /Users/michaelrohs/dev/android/EclipseWorkspace
Package Explore
Hierarchy
CameraAPI
CameraTest
ColorYellow
GestureBuilderActivity
GesturesDemo
HelloWorld
Location1
src
gen [Generated Java Files]
Android 2.2
assets
res
AndroidManifest.xml
default.properties
proguard.cfg
Location2
src
de.lmu.location2
MainActivity.java
PointOfInterest.java
Quiz.java
ShowQuizActivity.java
gen [Generated Java Files]
Android 2.2
assets
res
drawable-hdpi
drawable-ldpi
drawable-mdpi
layout
main.xml
showquestion.xml
values
AndroidManifest.xml
default.properties
proguard.cfg
MapTest
NotificationTest
OpineWorldCupServer
OutdoorPointing
OutdoorPointingCenters
OutdoorPointingMatching
MainActivity.java
showquestion.xml
main.xml
Editing config: defa
Any locale
No Dock
Day time
Create...
3.7in WVGA (Ne...
Po...
Theme
Android 2.2
Palette
Form Widgets
TextView
Button
CheckBox
ToggleButton
RadioButton
CheckedTextVie
Spinner
EditText
AutoCompleteT
MultiAutoComp
ProgressBar
QuickContactBa
RadioGroup
RatingBar
SeekBar
Layouts
Composite
Images & Medi
Time & Date
Transitions
Advanced
Graphical Layout
showquestion.xml
Title
Question Question Question Question
Question Question Question Question
Question Question Question
answer 1
answer 2
answer 3
Outl
Hist
Pro
Metr
LinearLayout
showQuestionTitle (TextView) – "Title"
ScrollView
LinearLayout
showQuestionImage (ImageView)
question (TextView) – "Question Que
answersRadioGroup
answer1 (RadioButton) – "answer
answer2 (RadioButton) – "answer
answer3 (RadioButton) – "answer
answerSubmitButton – "Submit"
→ ShowQuizActivity
LinearLayout/ScrollView/LinearLayout/RadioGroup/RadioButton/android:text
Launching Location2

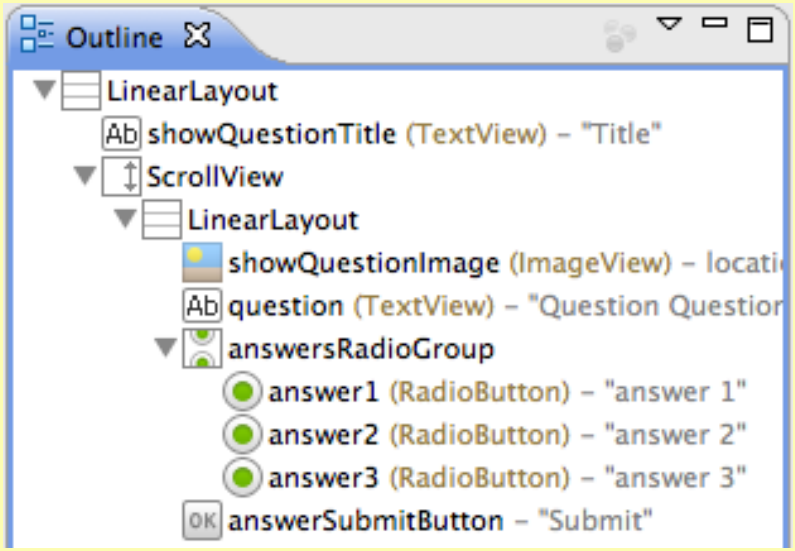
Exercise

- Create Main View
  - main view in /res/layout/main.xml
- Structure
  - LinearLayout (vertical)
    - TextView (title)
    - TableLayout
      - TableRow (latitude)
        - » TextView (label)
        - » TextView (value)
      - TableRow (longitude)
        - » …
      - TableRow (space)
      - TableRow (name of closest POI)
      - TableRow (distance to closest POI)
      - TableRow (space)
      - TableRow (number of points)



Exercise

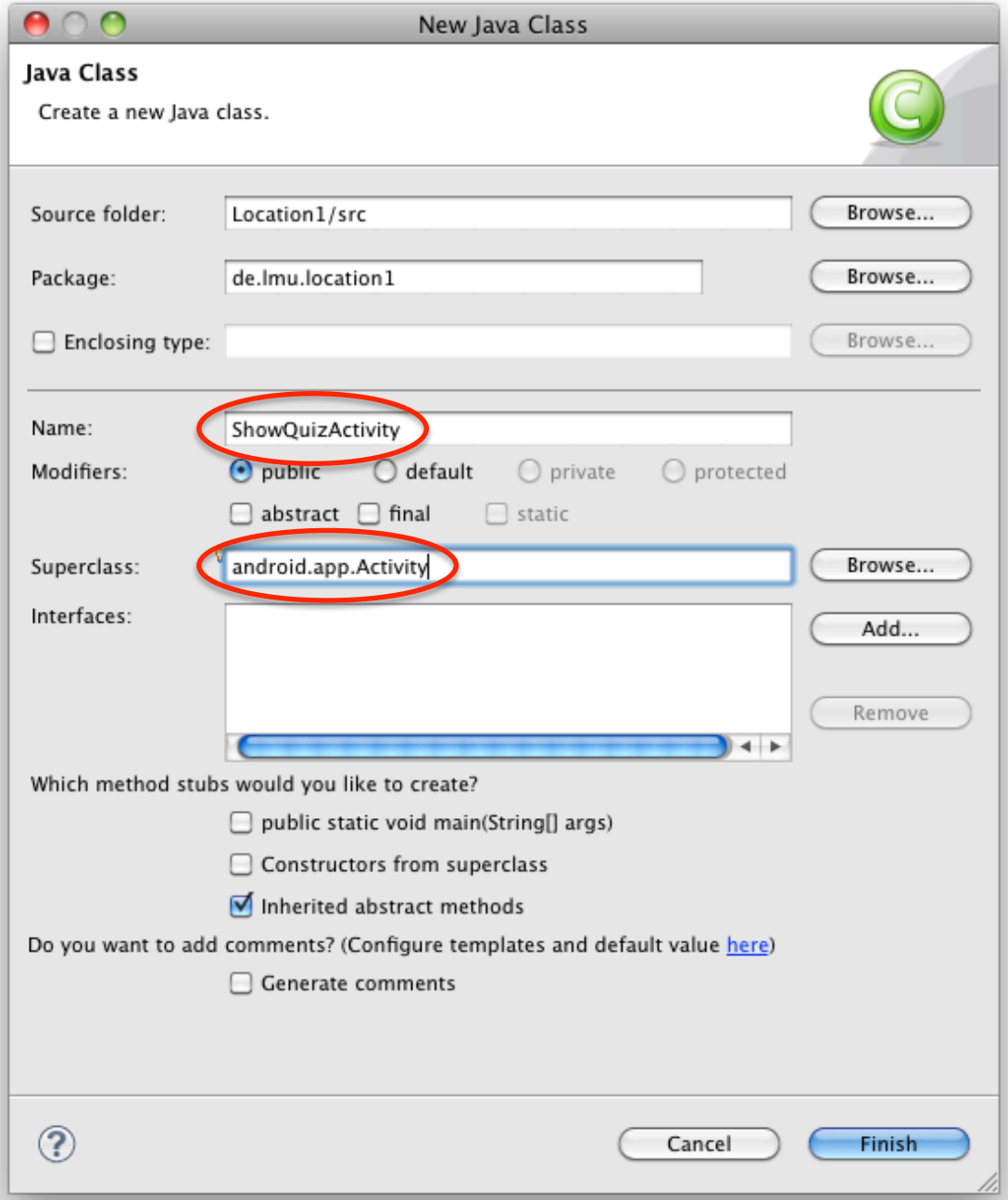
- Create Quiz View
  - quiz view in /res/layout/showquiz.xml
    (start by copying main.xml, then adapt it)
- Structure
  - LinearLayout
    - TextView (title)
    - ScrollView
      - LinearLayout (vertical)
        - » ImageView (POI image)
        - » TextView (question)
        - » RadioGroup (choices)
          - » RadioButton (choice1)
          - » RadioButton (choice2)
          - » RadioButton (choice3)
        - » Button (submit)



USING ACTIVITIES

Activities

- Create new class ShowQuizActivity
- Superclass: android.app.Activity



ShowQuizActivity → AndroidManifest.xml

- Activity class:

```
public class ShowQuizActivity extends Activity {
    public void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.showquiz);
} }
```

- AndroidManifest.xml (inside application element)

```
<activity android:name="de.lmu.quiz.ShowQuizActivity"
          android:label="showquiz"
    android:screenOrientation="portrait">
</activity>
```

How to start the new activity?

- Starting an activity:

```
Intent intent = new Intent(this, ShowQuizActivity.class);
startActivityForResult(intent, requestCode);
```

- Processing the result when the activity returns:

```
void onActivityResult(int requestCode, int resultCode, Intent data) {
    // do something with the result…
}
```

Exercise

- Create the ShowQuizActivity
- Create a menu in the MainActivity
- Start ShowQuizActivity from the menu

How to return to the previous activity?

- Set result and finish the activity

```
setResult(points);
finish();
```

Exercise

- Return from ShowQuizActivity to MainActivity
  - Set OnClickListener for submit button in ShowQuizActivity
  - When button is clicked, set result to 123 and finish the task
  - Show the result in the points view in MainActivity

How to copy data from one activity to another?

- Add “extras” to Intent objects

```
Intent intent = new Intent(this, ShowQuizActivity.class);
intent.putExtra("title", "Target 1");
intent.putExtra("image", R.drawable.location1);
startActivityForResult(intent, resultCode);
```

- Can put primitive types and Serializable types into extras
  - java.io.Serializable is just a “tagging” interface (no methods)

Exercise: Show title and image for a location

- Use a (small) image from the Web
- Name the image “location1.png” (or “location1.jpg”)
- Put the image into all “/res/drawable-*” folders
- Put title and image-id into intent extras
- Show the image in the ShowQuizActivity

How to share complex data between activities? (Possibility 1)

- In the calling activity, create a public static member (class variable) that references the shared object

```
public static PointOfInterest sharedPoi = null;
```

- Before starting the new activity, set the shared object

```
Intent intent = new Intent(this, ShowQuizActivity.class);
sharedPoi = closestPoi;
startActivity(intent);
```

- Use original shared object in called activity

```
TextView titleView = (TextView) findViewById(R.id.showQuestionTitle);
titleView.setText(MainActivity.sharedPoi.title);
```

How to share complex data between activities? (Possibility 2)

- Subclass android.app.Application, put shared data there

```
public class LocationQuiz extends Application {
    int points = 0;
    PointOfInterest currentPoi = null;
}
```

- Change AndroidManifest.xml

```
<application android:name="de.lmu.location.LocationQuiz" …>
    …
</application>
```

- Access shared data in activities

```
LocationQuiz app = (LocationQuiz) getApplication();
app.currentPoi = …;
app.points = 0;
```

LOCATION

GPS (Global Positioning System)

Exercise: Data Structures

- Need to define classes that hold data
- Which classes to define?

- Exercise: Create classes to hold the required data

class PointOfInterest

- Name
  - string
- Latitude, longitude
  - double
- Radius
  - double
- Image
  - int (Resource-ID)
- Quiz question (questions and answers for this POI)
  - class QuizQuestion

class QuizQuestion

- Question
  - String
- Answers: a fixed number of choices
  - String[n]
- Points: a fixed number of points for each choice
  - int[n]
- Choice: index of selected answer
  - int

Constructors for POI and QuizQuestion

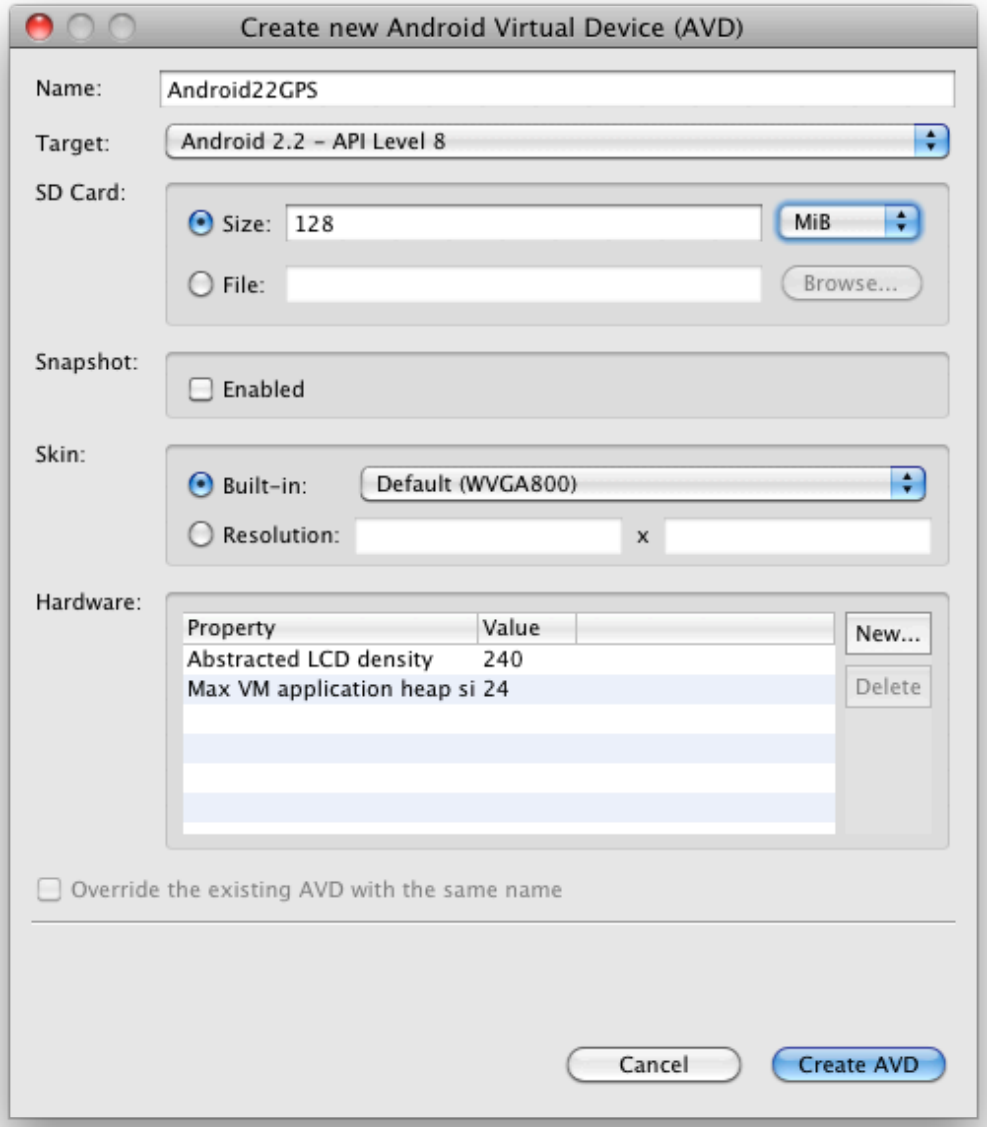
```
POI(  double lat,
      double lon,
      double radius,
      String name,
      int imageID,
      QuizQuestion q )
{
      …
}
```

```
QuizQuestion (
      String frage,
      String antwort1,
      String antwort2,
      String antwort3,
      int richtigeAntwort)
{
      …
}
```

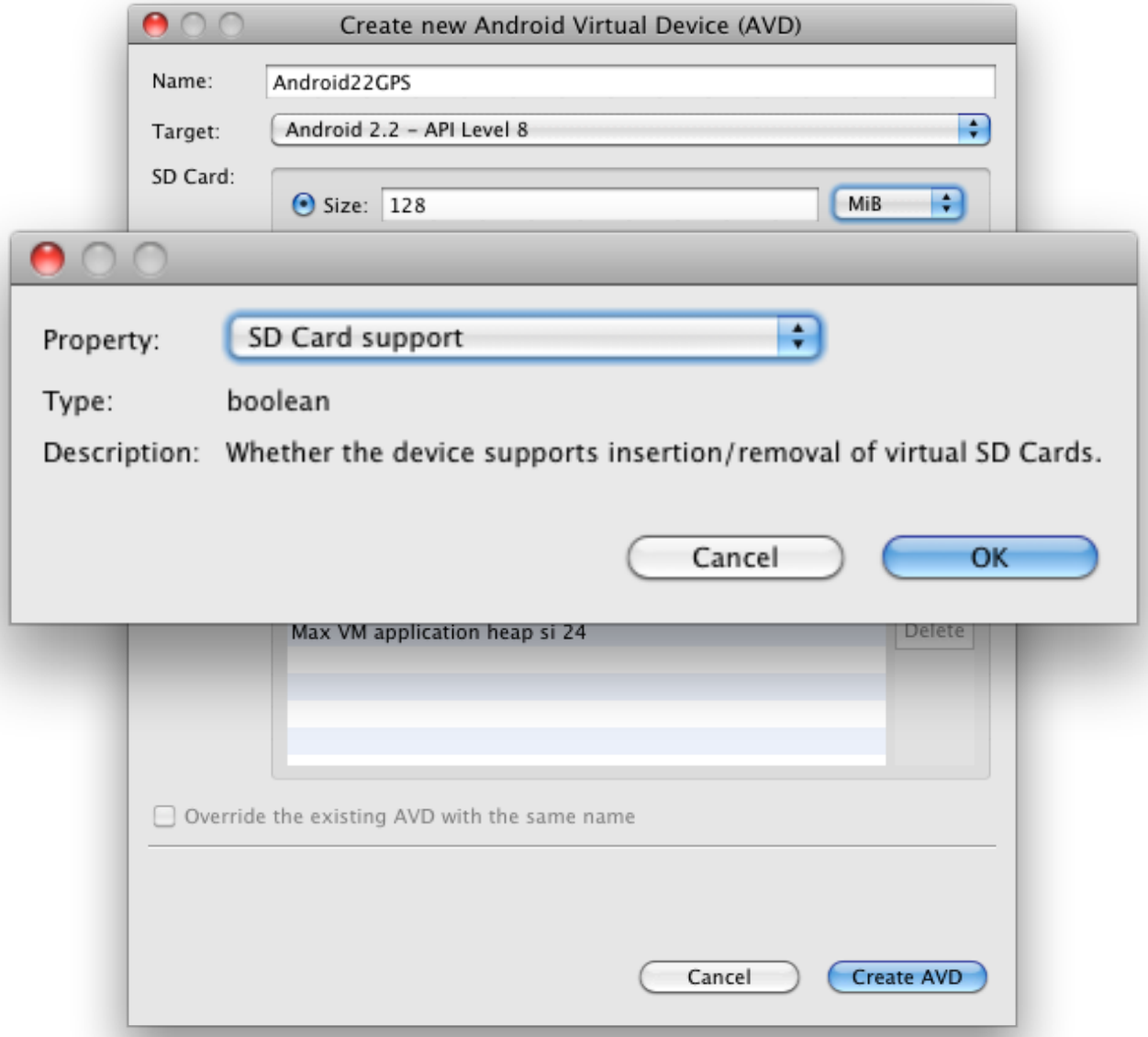
Using the Constructors

```
Vector<PointOfInterest> pois = new Vector<PointOfInterest>();

...

PointOfInterest p = new PointOfInterest(48.143446, 11.647552, 5,
                                        "Target 1", R.drawable.location1);
QuizQuestion q = new QuizQuestion("What is the most interesting
fact about this street?");
q.addAnswer("It is very new", 4);
q.addAnswer("It is very old", -4);
q.addAnswer("It is very busy", 40);
p.quiz = q;
pois.add(p);
…
```

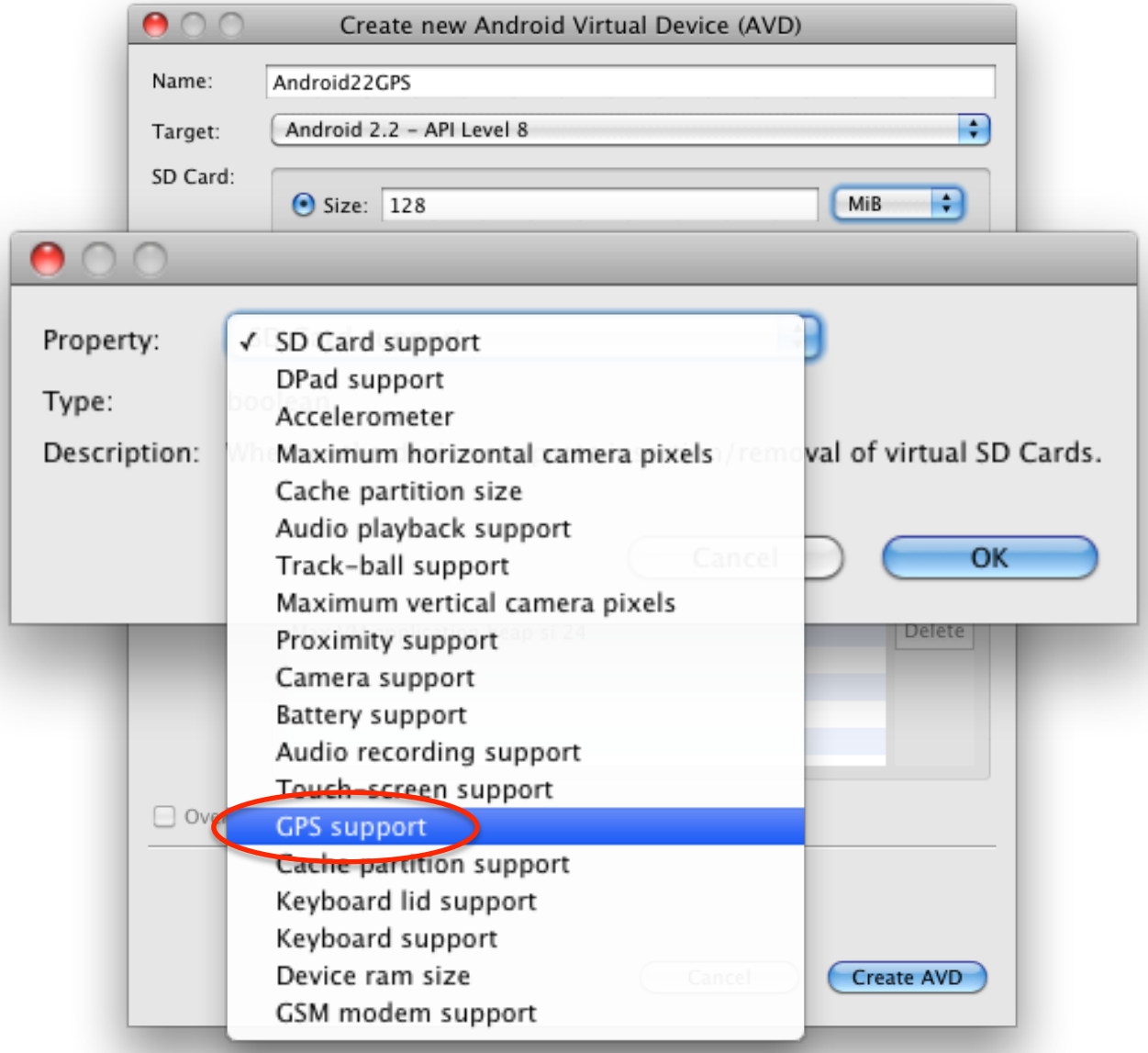
Enabling GPS on the Emulator



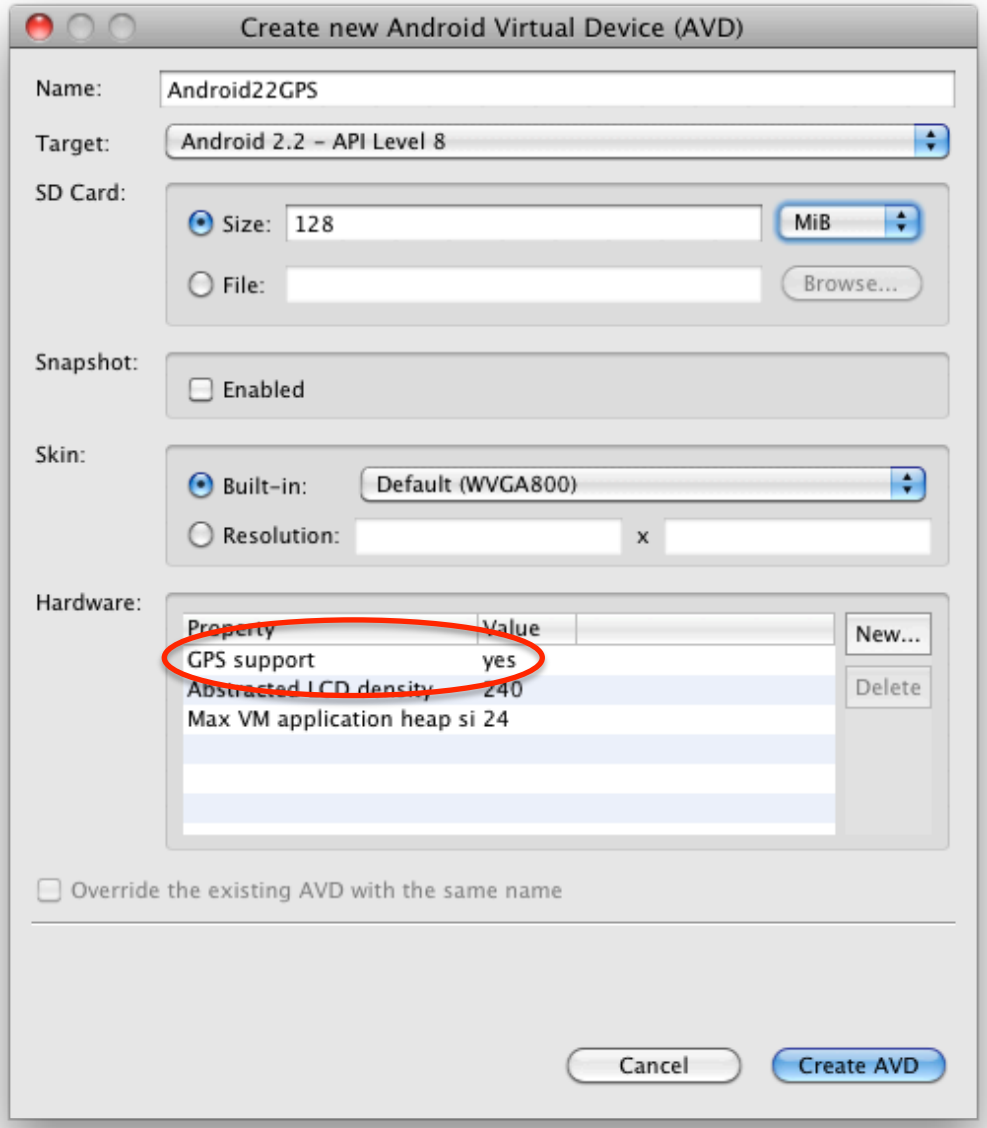
Enabling GPS on the Emulator



Enabling GPS on the Emulator

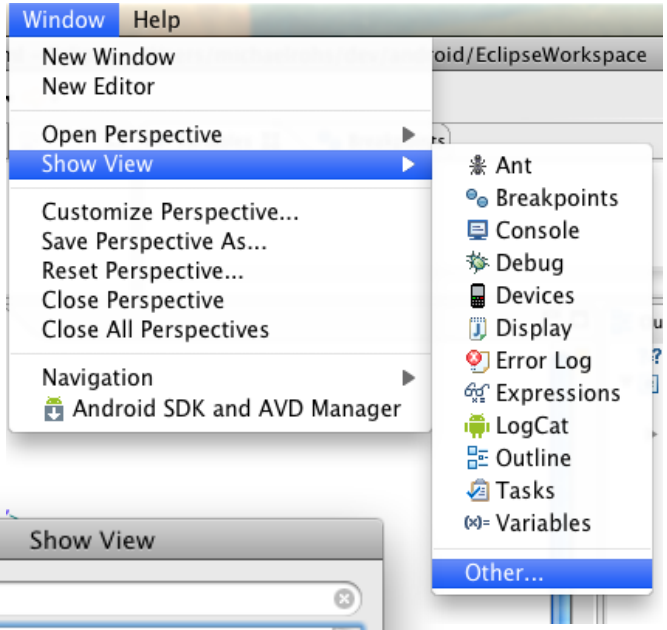


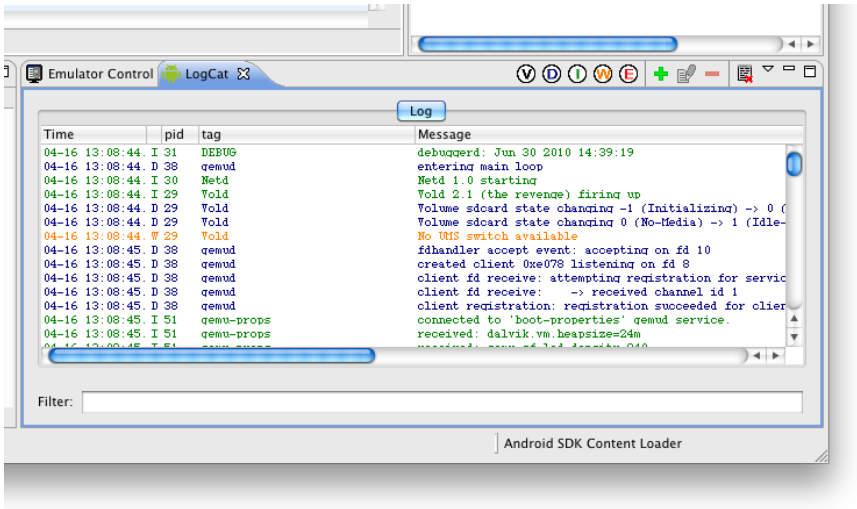
Enabling GPS on the Emulator

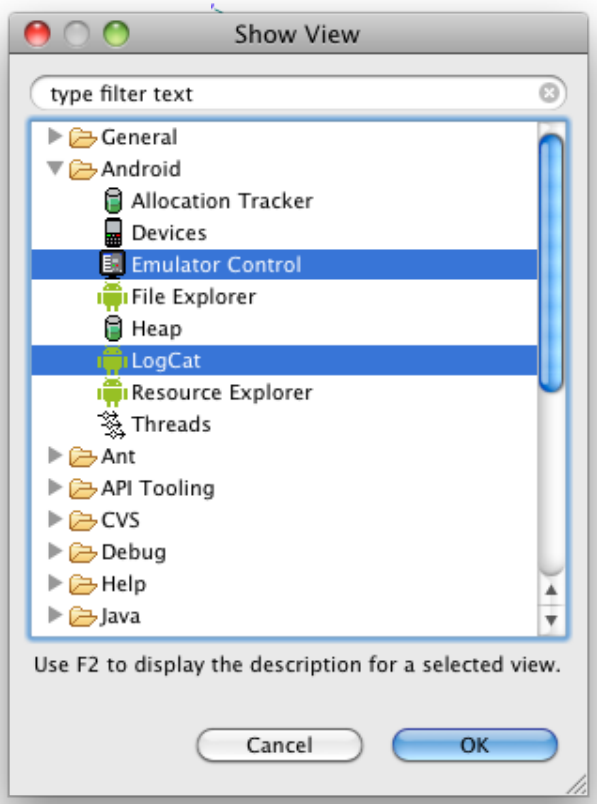


Eclipse Configuration

- LogCat View
  - Log.d output
- Emulator Control View
  - Entering locations



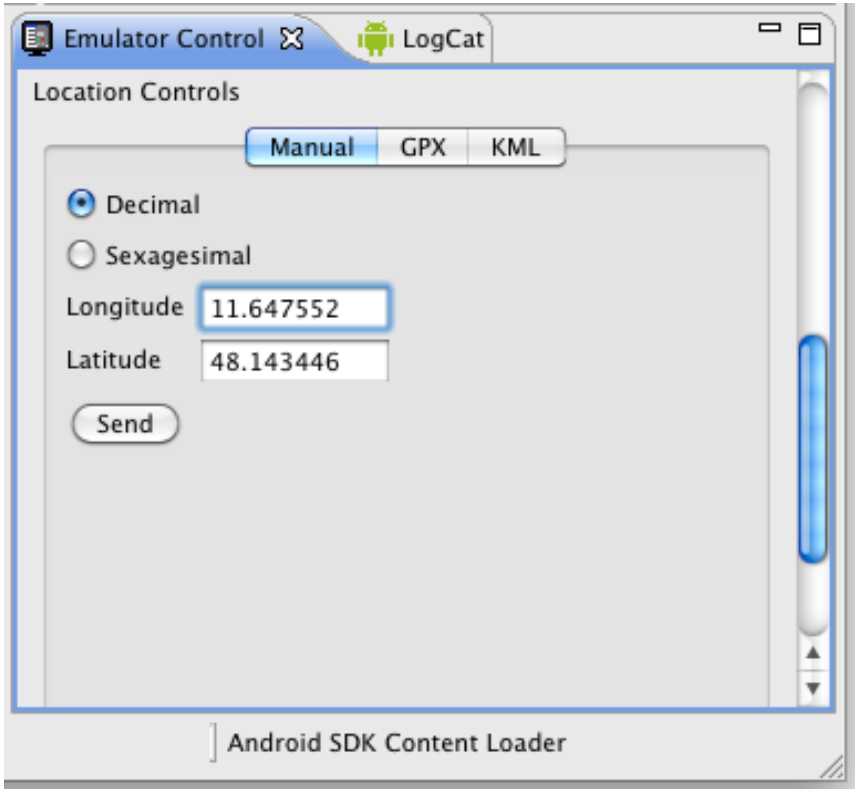




Entering Locations in Emulator Control View

- How to get latitude and longitude?

see next slides…



Keyhole Markup Language (KML)

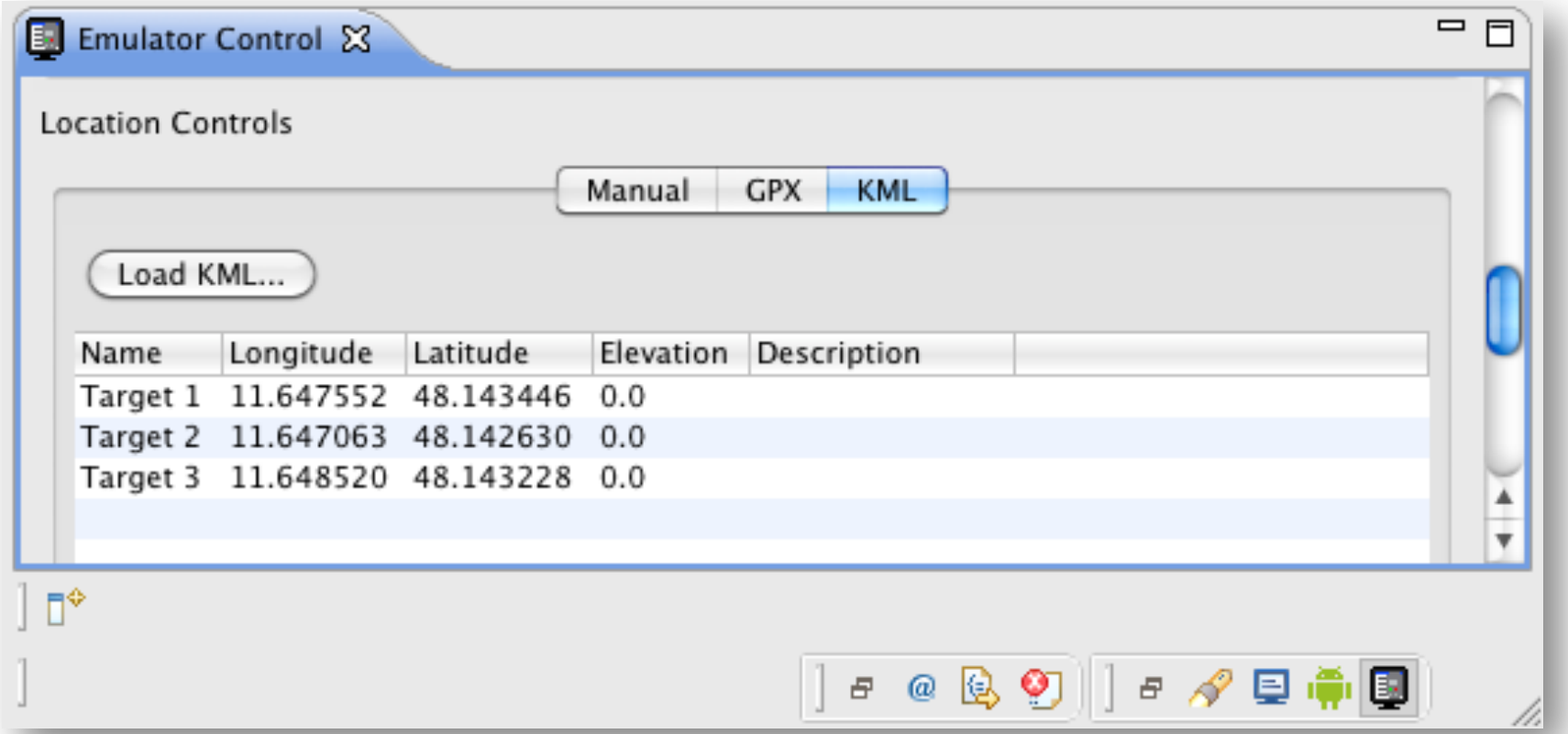
- XML-based language for expressing geographic information
  - Standardized by the Open Geospatial Consortium
  - Used in Google Maps (Mobile), Google Earth
- Example:

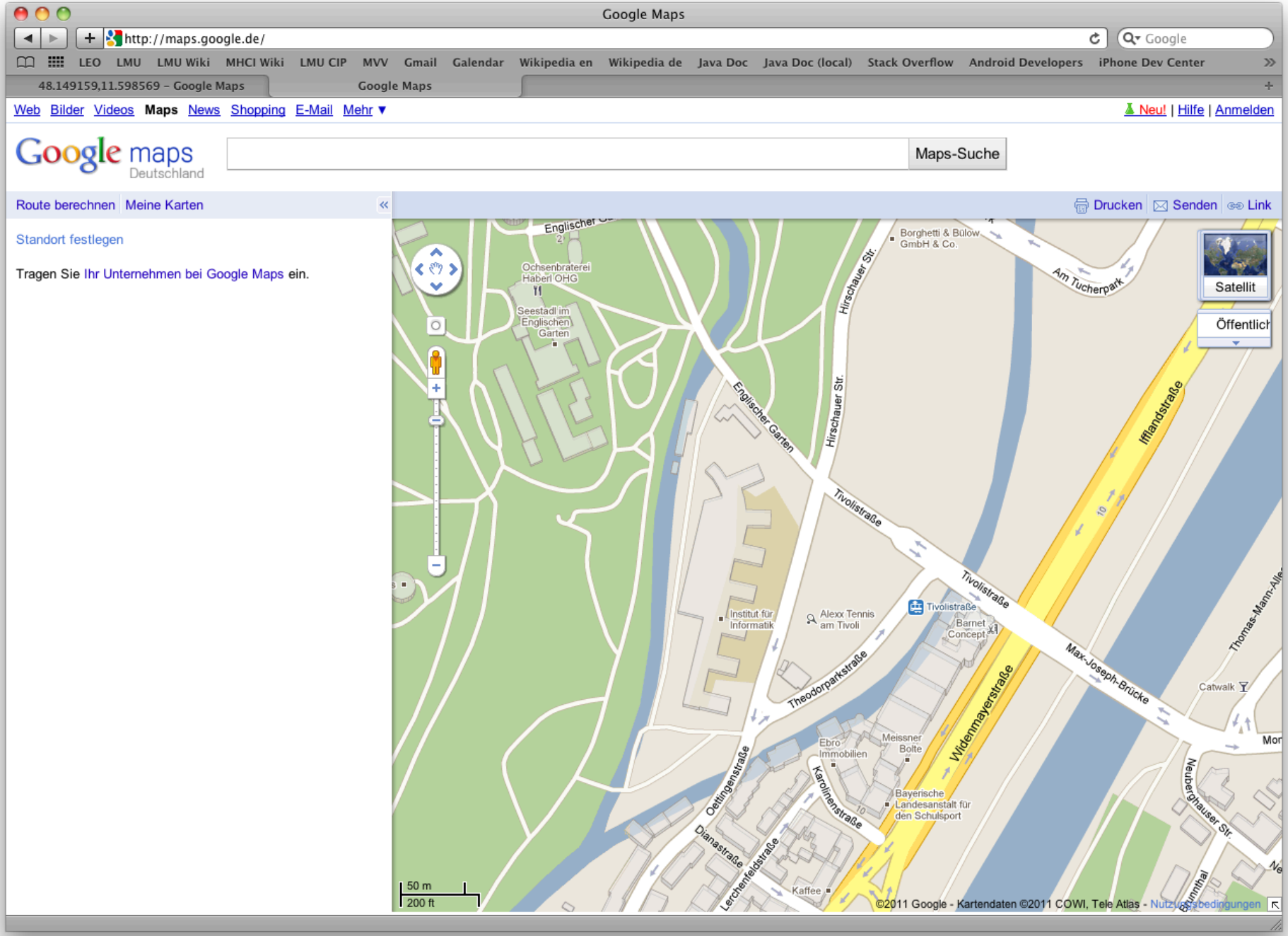
```
<?xml version="1.0" encoding="UTF-8"?>
<kml xmlns="http://earth.google.com/kml/2.2">
  <Document>
    <Placemark>
      <name>Target 1</name>
      <description>This is the first target.</description>
      <Point>
        <coordinates>11.647552,48.143446,0</coordinates> <!-- longitude, latitude, altitude -->
      </Point>
    </Placemark>
  </Document>
</kml>
```

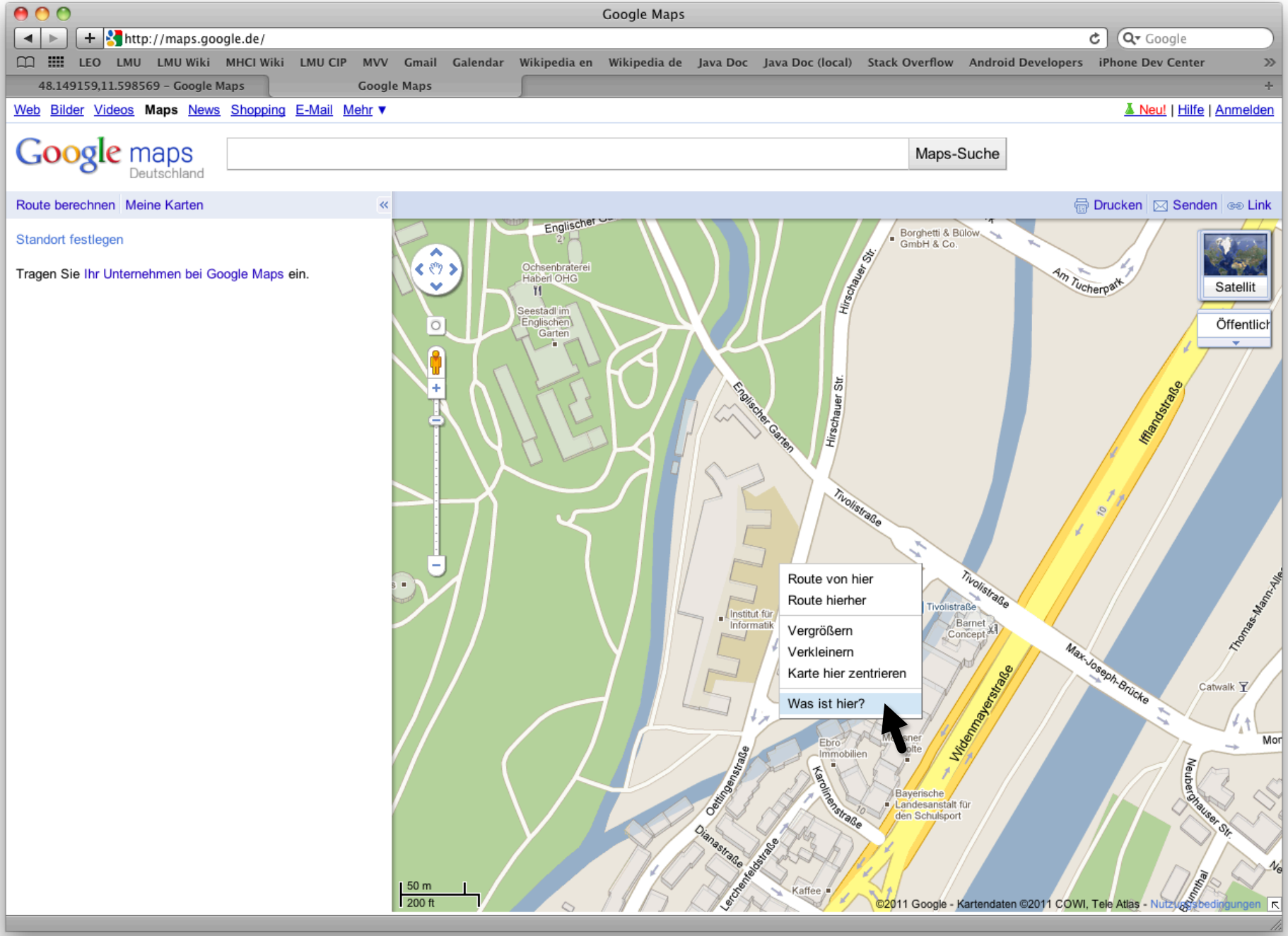
kml-samples.googlecode.com/svn/trunk/interactive/index.html

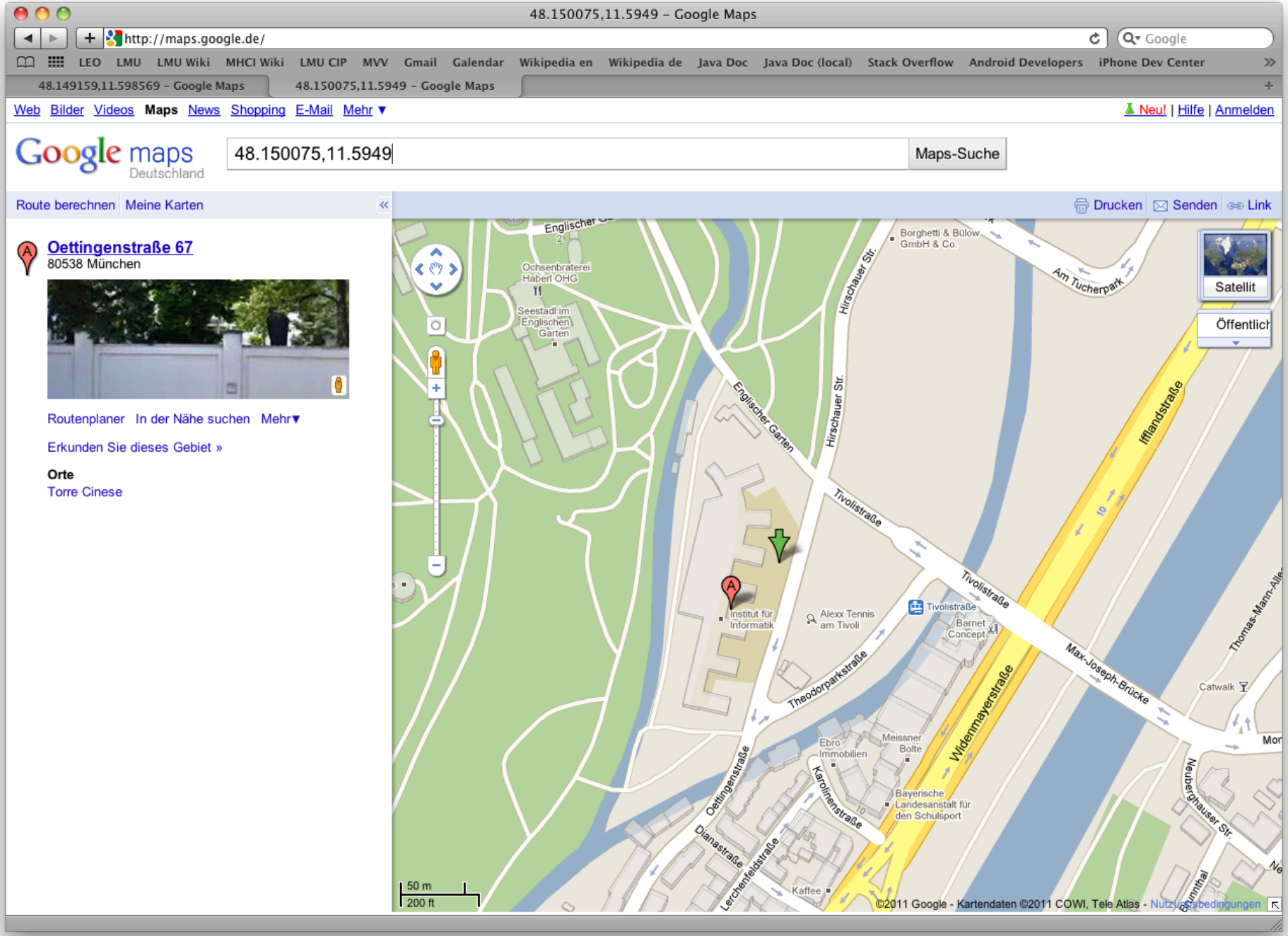
KML in the Emulator

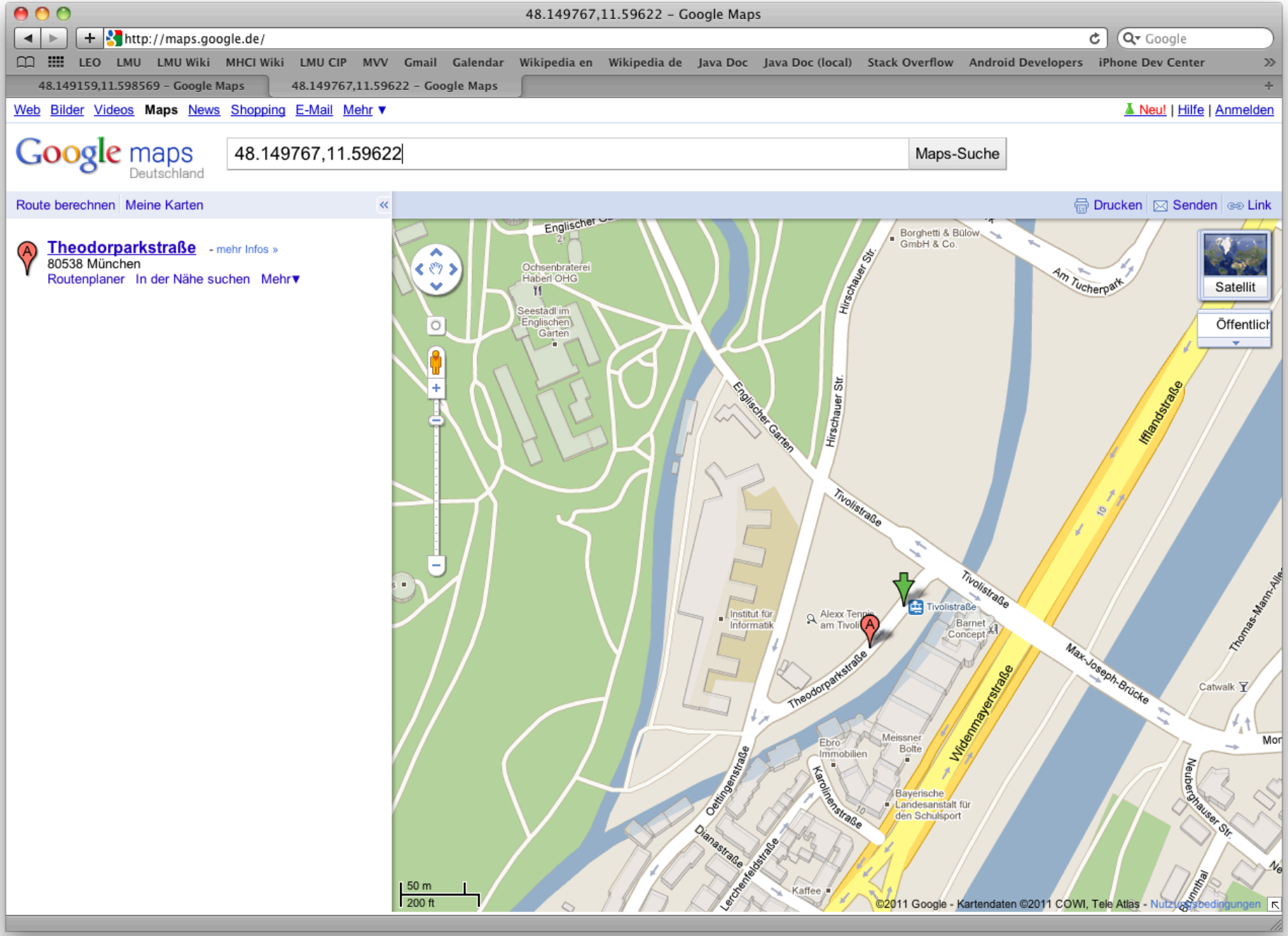
- Click a row to send location to emulator

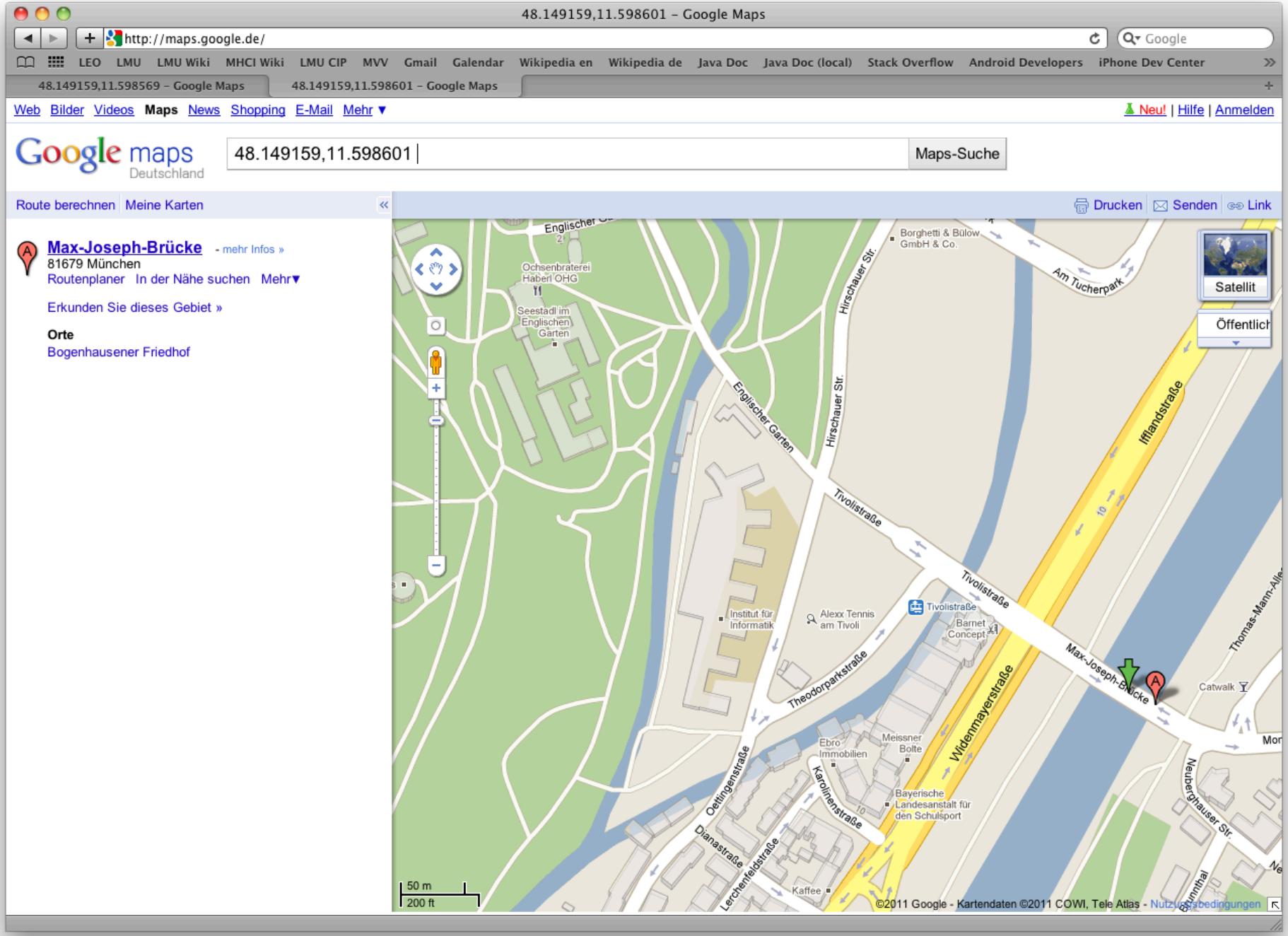


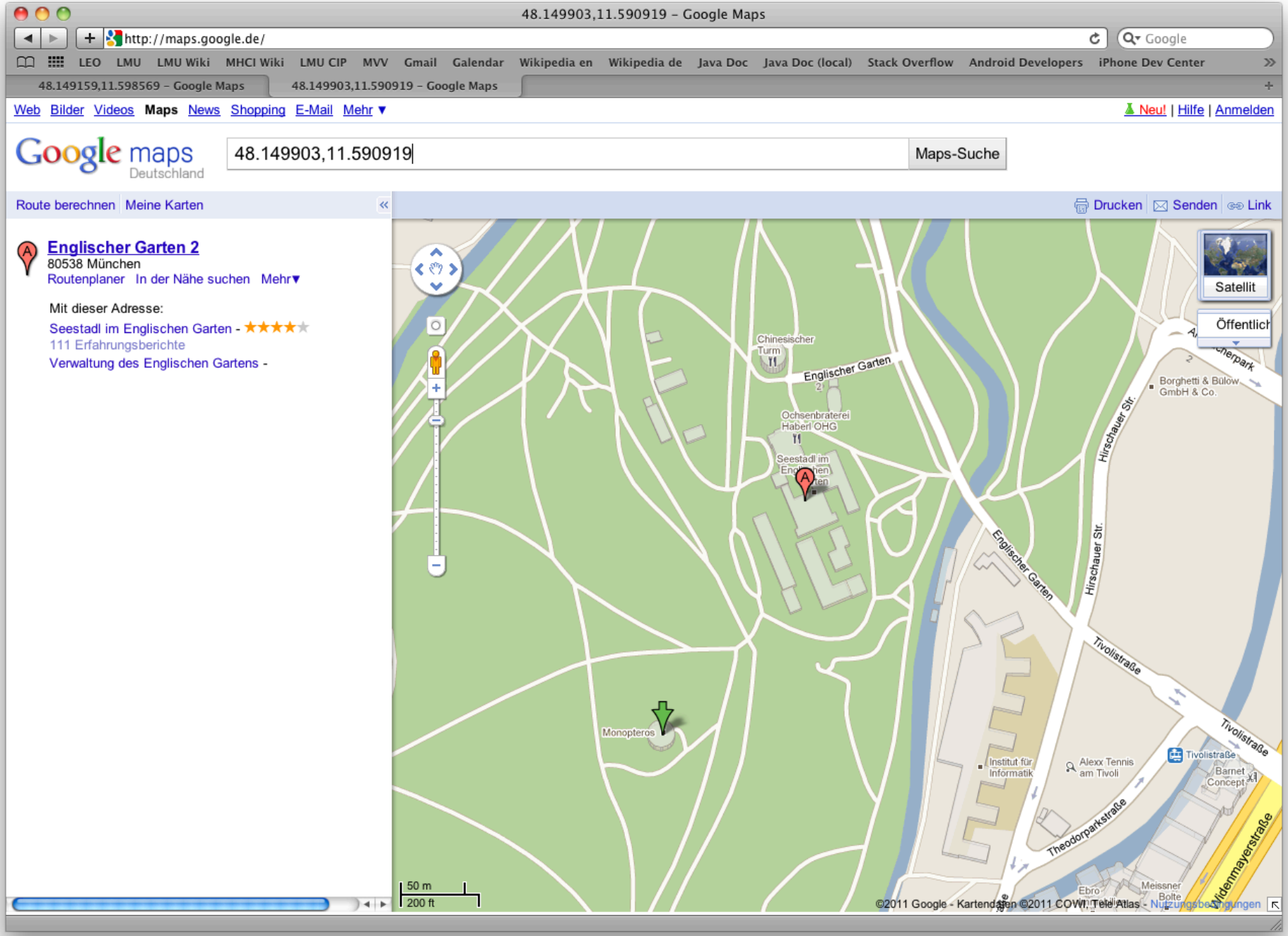

Google Maps
http://maps.google.de/
Google
LEO LMU LMU Wiki MHCI Wiki LMU CIP MVV Gmail Galendar Wikipedia en Wikipedia de Java Doc Java Doc (local) Stack Overflow Android Developers iPhone Dev Center
48.149159,11.598569 – Google Maps
Google Maps
Web Bilder Videos Maps News Shopping E-Mail Mehr ▼
Neu! | Hilfe | Anmelden
Google maps Deutschland
Maps-Suche
Route berechnen Meine Karten
Standort festlegen
Tragen Sie Ihr Unternehmen bei Google Maps ein.
Drucken Senden Link
Satellit
Öffentlich
Englischer Garten
Hirschauer Str.
Am Tucherpark
Borghetti & Bülow GmbH & Co.
Ochsenbraterei Haberl OHG
Seestadl im Englischen Garten
Tivolistraße
Ifflandstraße
Institut für Informatik
Alexx Tennis am Tivoli
Barnet Concept
Theodorparkstraße
Max-Joseph-Brücke
Thomas-Mann-Allee
Catwalk
Widenmayerstraße
Ebro Immobilien
Meissner Bolte
Oettingenstraße
Karolinenstraße
Bayerische Landesanstalt für den Schulsport
Dianastraße
Lerchenfeldstraße
Kaffee
Neuberghauser Str.
Brunnthal
50 m
200 ft
©2011 Google - Kartendaten ©2011 COWI, Tele Atlas - Nutzungsbedingungen

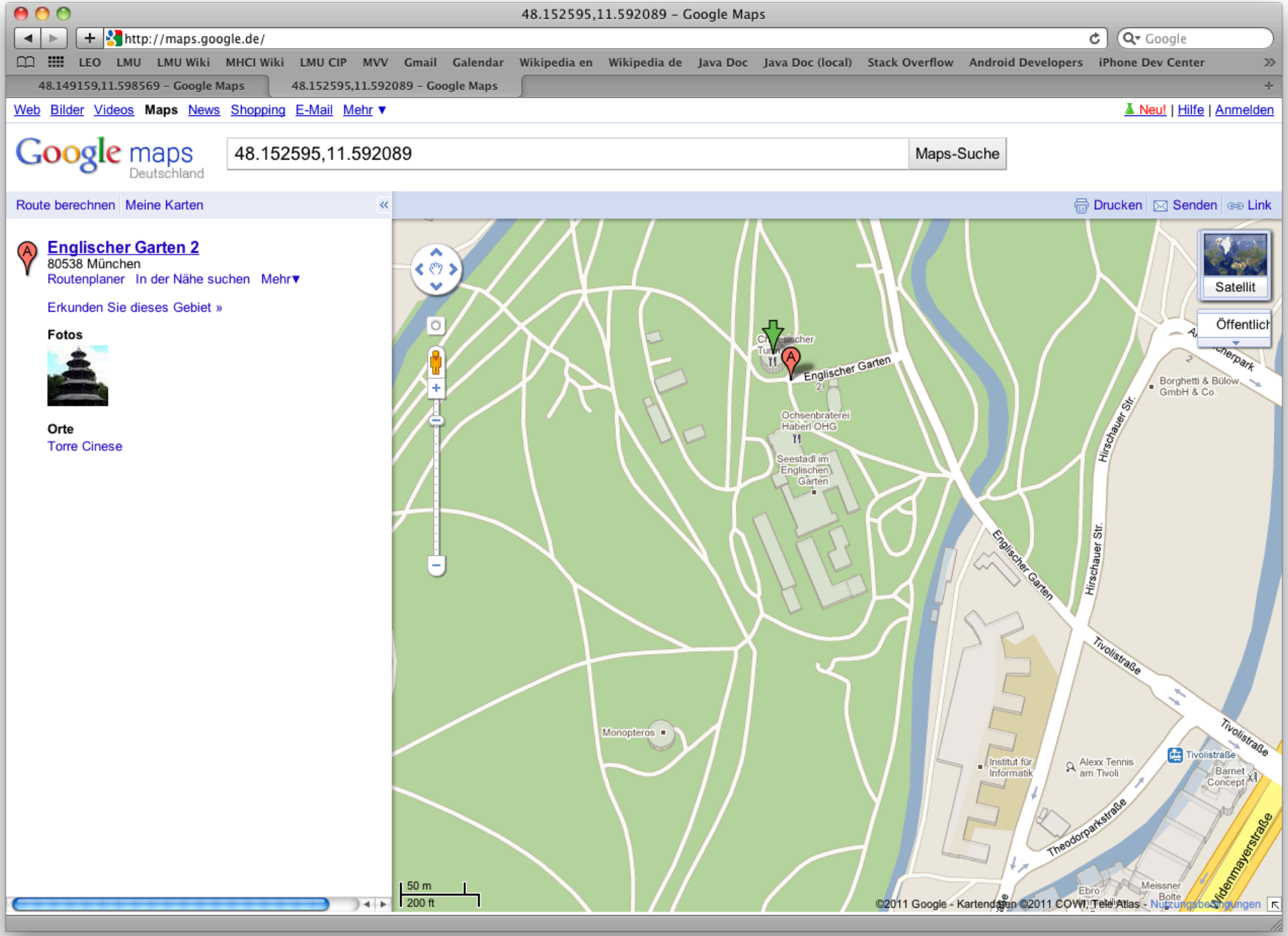

Google Maps
http://maps.google.de/
Google
LEO LMU LMU Wiki MHCI Wiki LMU CIP MVV Gmail Galendar Wikipedia en Wikipedia de Java Doc Java Doc (local) Stack Overflow Android Developers iPhone Dev Center
48.149159,11.598569 – Google Maps
Google Maps
Web Bilder Videos Maps News Shopping E-Mail Mehr ▼
Neu! | Hilfe | Anmelden
Google maps Deutschland
Maps-Suche
Route berechnen Meine Karten
Standort festlegen
Tragen Sie Ihr Unternehmen bei Google Maps ein.
Drucken Senden Link
Satellit
Öffentlich
Englischer Garten
Hirschauer Str.
Am Tucherpark
Borghetti & Bülow GmbH & Co.
Ochsenbraterei Haberl OHG
Seestadl im Englischen Garten
Tivolistraße
Ifflandstraße
Max-Joseph-Brücke
Widenmayerstraße
Thomas-Mann-Allee
Institut für Informatik
Barnet Concept
Catwalk
Ebro Immobilien
Bayerische Landesanstalt für den Schulsport
Oettingenstraße
Karolinenstraße
Dianastraße
Lerchenfeldstraße
Neuberghauser Str.
Kaffee
Route von hier
Route hierher
Vergrößern
Verkleinern
Karte hier zentrieren
Was ist hier?
50 m
200 ft
©2011 Google - Kartendaten ©2011 COWI, Tele Atlas - Nutzungsbedingungen


48.150075,11.5949 – Google Maps
http://maps.google.de/
Google
LEO LMU LMU Wiki MHCI Wiki LMU CIP MVV Gmail Galendar Wikipedia en Wikipedia de Java Doc Java Doc (local) Stack Overflow Android Developers iPhone Dev Center
48.149159,11.598569 – Google Maps
48.150075,11.5949 – Google Maps
Web Bilder Videos Maps News Shopping E-Mail Mehr ▼
Neu! | Hilfe | Anmelden
Google maps Deutschland
48.150075,11.5949
Maps-Suche
Route berechnen Meine Karten
Drucken Senden Link
Oettingenstraße 67
80538 München
Routenplaner In der Nähe suchen Mehr▼
Erkunden Sie dieses Gebiet »
Orte
Torre Cinese
Satellit
Öffentlich
Englischer Garten
Hirschauer Str.
Am Tucherpark
Ifflandstraße
Tivolistraße
Max-Joseph-Brücke
Theodorparkstraße
Widenmayerstraße
Oettingenstraße
Karolinenstraße
Dianastraße
Lerchenfeldstraße
Neuberghauser Str.
Thomas-Mann-Allee
Ochsenbraterei Haberl OHG
Seestadl im Englischen Garten
Borghetti & Bülow GmbH & Co.
Institut für Informatik
Alexx Tennis am Tivoli
Barnet Concept
Ebro Immobilien
Meissner Bolte
Bayerische Landesanstalt für den Schulsport
Catwalk
Kaffee
50 m
200 ft
©2011 Google - Kartendaten ©2011 COWI, Tele Atlas - Nutzungsbedingungen


48.149767,11.59622 – Google Maps
http://maps.google.de/
Google
LEO LMU LMU Wiki MHCI Wiki LMU CIP MVV Gmail Galendar Wikipedia en Wikipedia de Java Doc Java Doc (local) Stack Overflow Android Developers iPhone Dev Center
48.149159,11.598569 – Google Maps
48.149767,11.59622 – Google Maps
Web Bilder Videos Maps News Shopping E-Mail Mehr ▼
Neu! | Hilfe | Anmelden
Google maps Deutschland
48.149767,11.59622
Maps-Suche
Route berechnen Meine Karten
Drucken Senden Link
Theodorparkstraße - mehr Infos »
80538 München
Routenplaner In der Nähe suchen Mehr▼
Satellit
Öffentlich
Englischer Garten
Hirschauer Str.
Tivolistraße
Max-Joseph-Brücke
Theodorparkstraße
Widenmayerstraße
Ifflandstraße
Am Tucherpark
Ochsenbraterei Haberl OHG
Seestadl im Englischen Garten
Borghetti & Bülow GmbH & Co.
Institut für Informatik
Alexx Tennis am Tivoli
Barnet Concept
Ebro Immobilien
Meissner Bolte
Bayerische Landesanstalt für den Schulsport
Oettingenstraße
Dianastraße
Karolinenstraße
Lerchenfeldstraße
Neuberghauser Str.
Thomas-Mann-Alle
Catwalk
Kaffee
50 m
200 ft
©2011 Google - Kartendaten ©2011 COWI, Tele Atlas - Nutzungsbedingungen

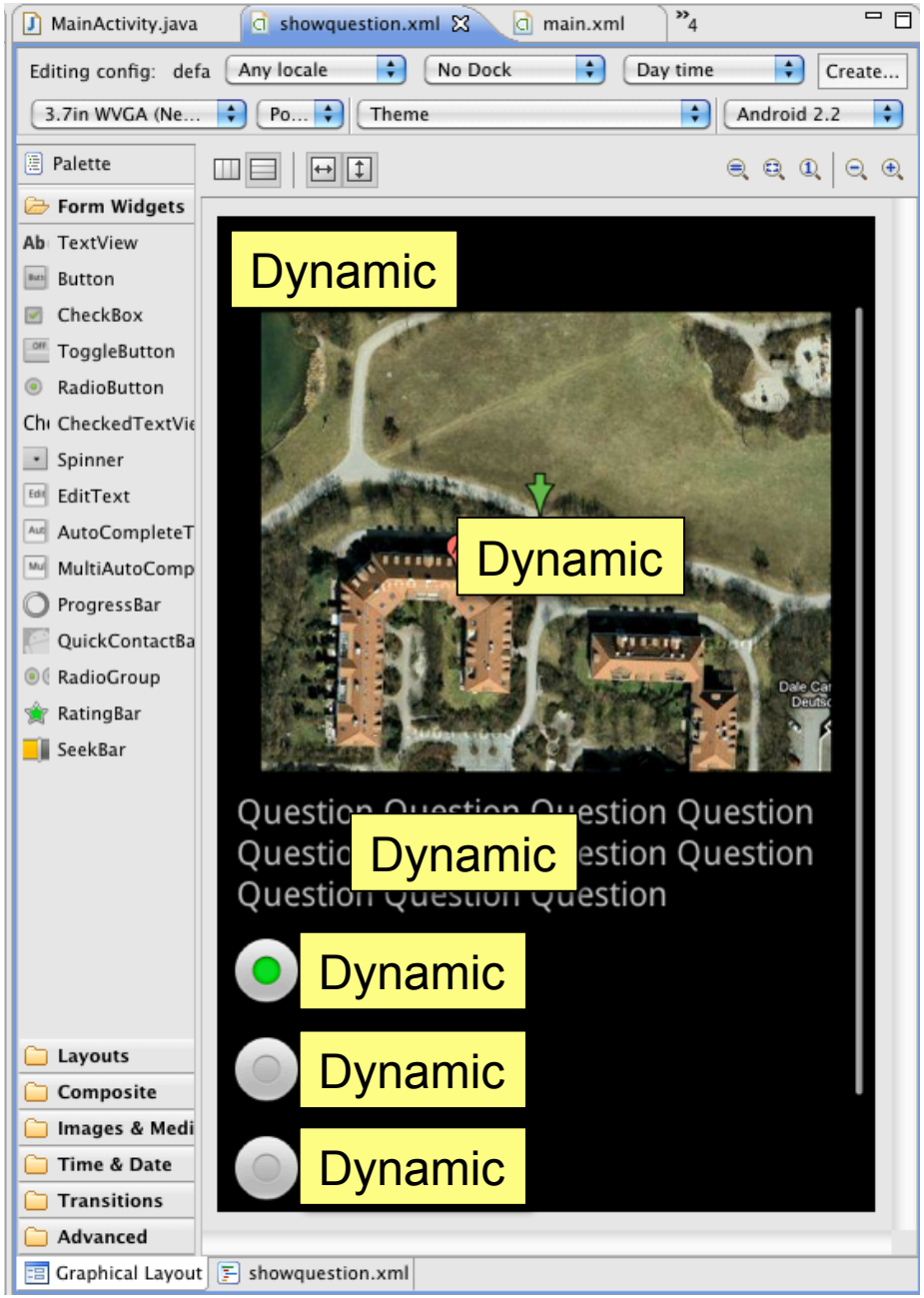

48.149159,11.598601 – Google Maps
http://maps.google.de/
Google
LEO LMU LMU Wiki MHCI Wiki LMU CIP MVV Gmail Galendar Wikipedia en Wikipedia de Java Doc Java Doc (local) Stack Overflow Android Developers iPhone Dev Center
48.149159,11.598569 – Google Maps
48.149159,11.598601 – Google Maps
Web Bilder Videos Maps News Shopping E-Mail Mehr ▼
Neu! | Hilfe | Anmelden
Google maps Deutschland
48.149159,11.598601
Maps-Suche
Route berechnen Meine Karten
Drucken Senden Link
Max-Joseph-Brücke - mehr Infos »
81679 München
Routenplaner In der Nähe suchen Mehr▼
Erkunden Sie dieses Gebiet »
Orte
Bogenhausener Friedhof
Satellit
Öffentlich
Englischer Garten
Hirschauer Str.
Am Tucherpark
Ifflandstraße
Tivolistraße
Max-Joseph-Brücke
Theodorparkstraße
Widenmayerstraße
Oettingenstraße
Dianastraße
Lerchenfeldstraße
Karolinenstraße
Neuberghauser Str.
Thomas-Mann-Allee
Ochsenbraterei Haberl OHG
Seestadl im Englischen Garten
Borghetti & Bülow GmbH & Co.
Institut für Informatik
Alexx Tennis am Tivoli
Barnet Concept
Ebro Immobilien
Meissner Bolte
Bayerische Landesanstalt für den Schulsport
Catwalk
Kaffee
50 m
200 ft
©2011 Google - Kartendaten ©2011 COWI, Tele Atlas - Nutzungsbedingungen


48.149903,11.590919 – Google Maps
http://maps.google.de/
Google
LEO LMU LMU Wiki MHCI Wiki LMU CIP MVV Gmail Galendar Wikipedia en Wikipedia de Java Doc Java Doc (local) Stack Overflow Android Developers iPhone Dev Center
48.149159,11.598569 – Google Maps
48.149903,11.590919 – Google Maps
Web Bilder Videos Maps News Shopping E-Mail Mehr ▼
Neu! | Hilfe | Anmelden
Google maps Deutschland
48.149903,11.590919
Maps-Suche
Route berechnen Meine Karten
Drucken Senden Link
Englischer Garten 2
80538 München
Routenplaner In der Nähe suchen Mehr▼
Mit dieser Adresse:
Seestadl im Englischen Garten -
111 Erfahrungsberichte
Verwaltung des Englischen Gartens -
Chinesischer Turm
Englischer Garten
Ochsenbraterei Haberl OHG
Seestadl im Englischen Garten
Monopteros
Hirschauer Str.
Borghetti & Bülow GmbH & Co.
Satellit
Öffentlich
Tivolistraße
Institut für Informatik
Alexx Tennis am Tivoli
Barnet Concept
Theodorparkstraße
Meissner Bolte
Nidermayerstraße
50 m
200 ft
©2011 Google - Kartendaten ©2011 COWI, Tele Atlas - Nutzungsbedingungen


48.152595,11.592089 – Google Maps
http://maps.google.de/
LEO LMU LMU Wiki MHCI Wiki LMU CIP MVV Gmail Galendar Wikipedia en Wikipedia de Java Doc Java Doc (local) Stack Overflow Android Developers iPhone Dev Center
48.149159,11.598569 – Google Maps
48.152595,11.592089 – Google Maps
Web Bilder Videos Maps News Shopping E-Mail Mehr ▼
Neu! | Hilfe | Anmelden
Google maps Deutschland
48.152595,11.592089
Maps-Suche
Route berechnen Meine Karten
Drucken Senden Link
Englischer Garten 2
80538 München
Routenplaner In der Nähe suchen Mehr▼
Erkunden Sie dieses Gebiet »
Fotos
Orte
Torre Cinese
Englischer Garten
Ochsenbraterei Haberl OHG
Seestadl im Englischen Garten
Monopteros
Satellit
Öffentlich
Borghetti & Bülow GmbH & Co.
Hirschauer Str.
Tivolistraße
Institut für Informatik
Alexx Tennis am Tivoli
Barnet Concept
Theodorparkstraße
Nidermayerstraße
Meissner Bolte
50 m
200 ft
©2011 Google - Kartendaten ©2011 COWI, Tele Atlas - Nutzungsbedingungen

Dynamic Content

- ShowQuizActivity controls graphical user interface
  - Replaces placeholder elements by actual content
  - Processes user input (on radio buttons, submit button)
- ShowQuizActivity will be triggered when POI nearby
  (distance < POI.radius)
- Actual content depends on POI we have entered



Algorithm for starting ShowQuizActivity

- In class MainActivity:

- onLocationChanged
  - for each known POI p
    - compute distance d of current location to POI p
    - if d < p.radius then
      - set dynamic data for POI p
      - start activity for result
      - break
- onActivityResult
  - points = points + resultCode
  - update TextView to show points

Transferring Dynamic Data to ShowQuizActivity (Alternative 1)

- Make data classes serializable (to be put into Intent)

```
public class PointOfInterest implements Serializable {…}
public class QuizQuestion implements Serializable {…}
```

- in MainActivity, onLocationChanged:

```
Intent intent = new Intent(this, ShowQuizActivity.class);
intent.putExtra("poi", closestPoi);
startActivityForResult(intent, 0);
```

- in ShowQuizActivity, onCreate or onStart:

```
PointOfInterest poi; // class member variable, outside of method
…
poi = (PointOfInterest) getIntent().getSerializableExtra("poi");
TextView titleView = (TextView) findViewById(R.id.title);
titleView.setText(poi.title);
```

Transferring Dynamic Data to ShowQuizActivity (Alternative 2)

- in MainActivity, (static) shared member variable

```
static PointOfInterest currentPoi;
```

- in MainActivity, onLocationChanged:

```
Intent intent = new Intent(this, ShowQuizActivity.class);
currentPoi = closestPoi;
startActivityForResult(intent, 0);
```

- in ShowQuizActivity, onCreate or onStart:

```
PointOfInterest poi; // class member variable, outside of method
…
poi = MainActivity.currentPoi;
TextView titleView = (TextView) findViewById(R.id.title);
titleView.setText(poi.title);
```

Exercise: Data Structures

- Exercise: Example POIs and quizzes
  - Take a few sample points-of-interest and quizzes and enter them in your data structures

Location-Based Services

- Access location data
  - GPS
  - WiFi
  - GSM
- Location Manager Service
  - Obtain device's geographical location
  - Get notification upon entering a specified location
  - Package: android.location
  - Class: LocationManager

Example: Last Known Location

```
public class LocationManagerDemoActivity extends Activity {
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);

        LocationManager locMgr = (LocationManager)
                getSystemService(Context.LOCATION_SERVICE);

        Location loc = locMgr.getLastKnownLocation(
                            LocationManager.GPS_PROVIDER);

        Toast.makeText(this, loc.toString(), 10000).show();
        Log.d("last location", loc.toString());
    }
}
```

Example: Continuous Location Updates

```
public class LocationUpdateActivity extends Activity
                                    implements LocationListener
{
   public void onCreate(Bundle savedInstanceState) {
      super.onCreate(savedInstanceState);
      LocationManager locMgr = (LocationManager)
            getSystemService(Context.LOCATION_SERVICE);
      locMgr.requestLocationUpdates(
            LocationManager.GPS_PROVIDER, 0, 0, this);
   }
   …
```

GPS_PROVIDER or *NETWORK_PROVIDER*

min time between updates [ms]

min distance between updates [m]

Example: Continuous Location Updates

```
...
   public void onLocationChanged(Location loc) {
      if (loc != null) {
         Toast.makeText(getBaseContext(),
            "New location (" + loc.getLatitude() + ", " +
            loc.getLongitude() + ")",
            Toast.LENGTH_LONG).show();
      }
   }
   public void onProviderDisabled(String provider) {}
   public void onProviderEnabled(String provider) {}
   public void onStatusChanged(String provider,
                               int status, Bundle extras) {}
}
```

Distance Between Geo-Locations

- Distance (in m) between two geolocations

```
float[] results = new float[1];
Location.distanceBetween(lat, lon, poi.latitude, poi.longitude, results);
float distance = results[0];
```

Permissions (in AndroidManifest.xml)

- Permissions for location-based services

```
<uses-permission
  android:name="android.permission.ACCESS_COARSE_LOCATION" />
<uses-permission
  android:name="android.permission.ACCESS_FINE_LOCATION" />
<uses-permission
  android:name="android.permission.ACCESS_MOCK_LOCATION" />
<uses-permission
  android:name="android.permission.ACCESS_LOCATION_EXTRA_COMMANDS" />
<uses-permission
  android:name="android.permission.INTERNET" />
```

- Overview of Android permissions
  - http://developer.android.com/reference/android/Manifest.permission.html

Example Manifest for Location

```
<?xml version="1.0" encoding="utf-8"?>
<manifest xmlns:android="http://schemas.android.com/apk/res/android"
    package="de.lmu.location"
    android:versionCode="1"
    android:versionName="1.0">
  <uses-sdk android:minSdkVersion="8" />
  <application android:icon="@drawable/icon" android:label="@string/app_name" android:debuggable="true">
    <activity android:name=".MainActivity"
              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>
  <uses-permission android:name="android.permission.ACCESS_COARSE_LOCATION" />
  <uses-permission android:name="android.permission.ACCESS_FINE_LOCATION" />
  <uses-permission android:name="android.permission.ACCESS_MOCK_LOCATION" />
  <uses-permission android:name="android.permission.ACCESS_LOCATION_EXTRA_COMMANDS" />
  <uses-permission android:name="android.permission.VIBRATE" />
</manifest>
```

Exercise: Location Updates

- Register for location updates from the location manager with an interval of 5s
  - Only register for location updates when the MainActivity is actually active (i.e. use onResume and onPause to register/ unregister updates)
- Handle location updates in the MainActivity itself
  - See next slide for template
- Start ShowQuizActivity if the user enters a POI
  - Make POI data available in ShowQuizActivity

Template for Location Updates

```
public class MainActivity extends Activity implements LocationListener {
    LocationManager locationManager = null;
    …
    public void onLocationChanged(Location location) {
        if (location != null) {
            // process location update
        }
    }
    public void onProviderDisabled(String provider) {}
    public void onProviderEnabled(String provider) {}
    public void onStatusChanged(String provider, int status, Bundle ext) {}
}
```

How to vibrate the phone?

- Java

```
Vibrator vibrator = (Vibrator) getSystemService(Context.VIBRATOR_SERVICE);
vibrator.vibrate(1000);
```

- AndroidManifest.xml

```
<uses-permission android:name="android.permission.VIBRATE" />
```

- Exercise: Make the phone vibrate when the Quiz starts (i.e. when the user enters the target area)

Exercise: Field Test

- Try out your app outdoors

- Come back and improve it
- Find test users to evaluate your program
- Improve it some more…

Prof. Dr. Michael Rohs
michael.rohs@ifi.lmu.de
Dipl.-Inform. Sven Kratz
sven.kratz@ifi.lmu.de
Mobile Interaction Lab, LMU München