MMI 2: Mobile Human-Computer Interaction

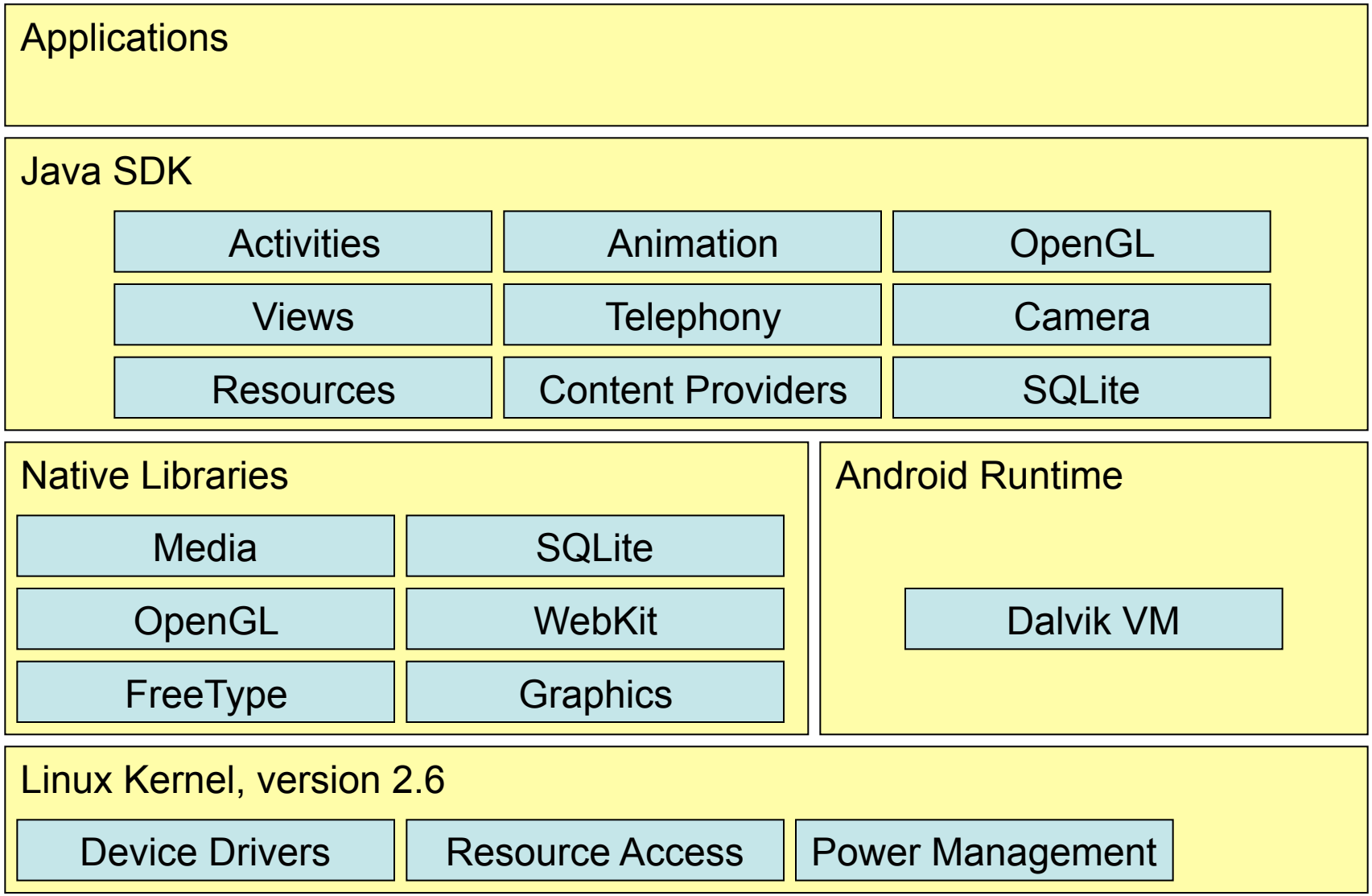
Android

Prof. Dr. Michael Rohs

michael.rohs@ifi.lmu.de

Mobile Interaction Lab, LMU München

Android Software Stack



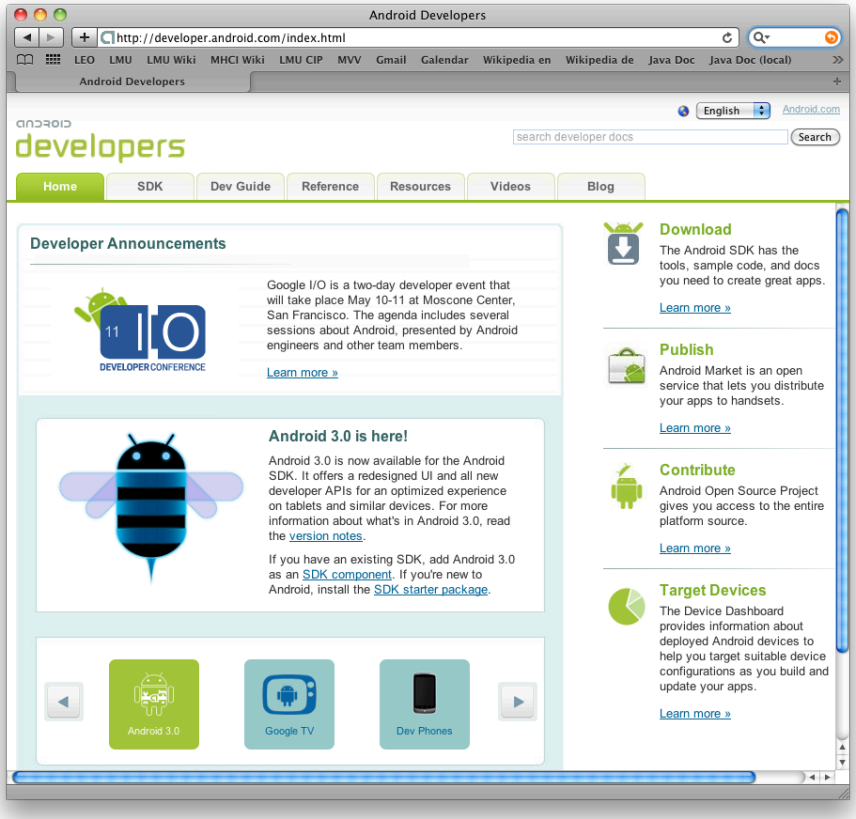
Android Characteristics

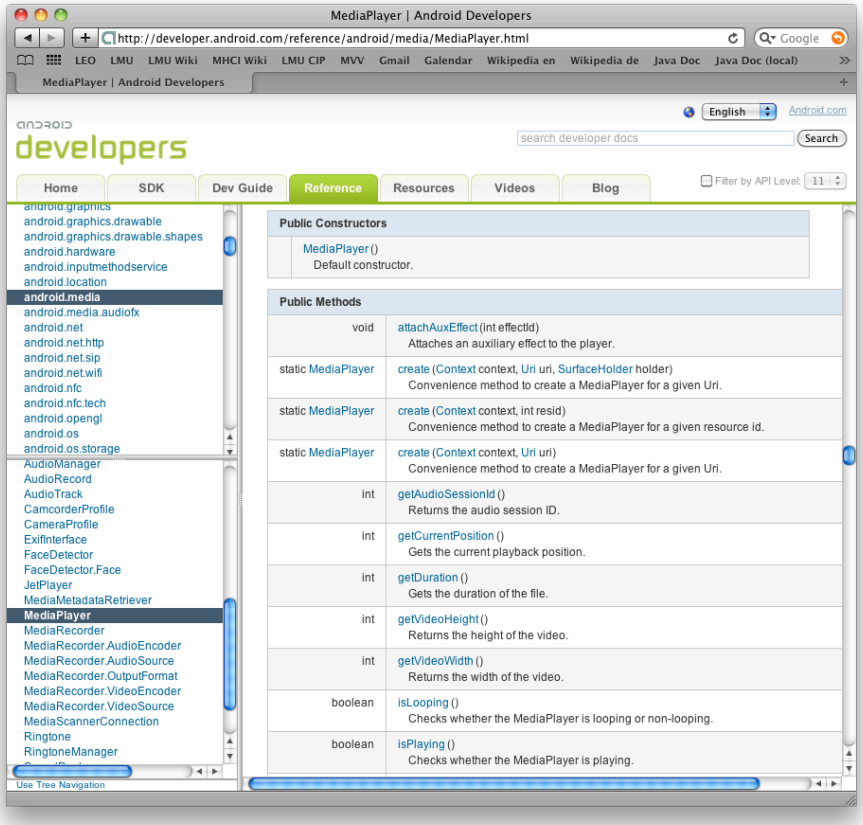
- Activity
  - Activities are the components of an application
  - Represent a logical unit of user action
  - Typically represented by a screen containing views
  - Can be invoked externally
- Declarative UI definition
  - XML files specify user interface resources
  - Resources (layout definitions, strings, bitmaps)
  - Separation of code and user interface
- “Teachable”
  - Clear semantics of Java, clear design & concepts, good emulator

Installing Android

Android Resources

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





Required Software

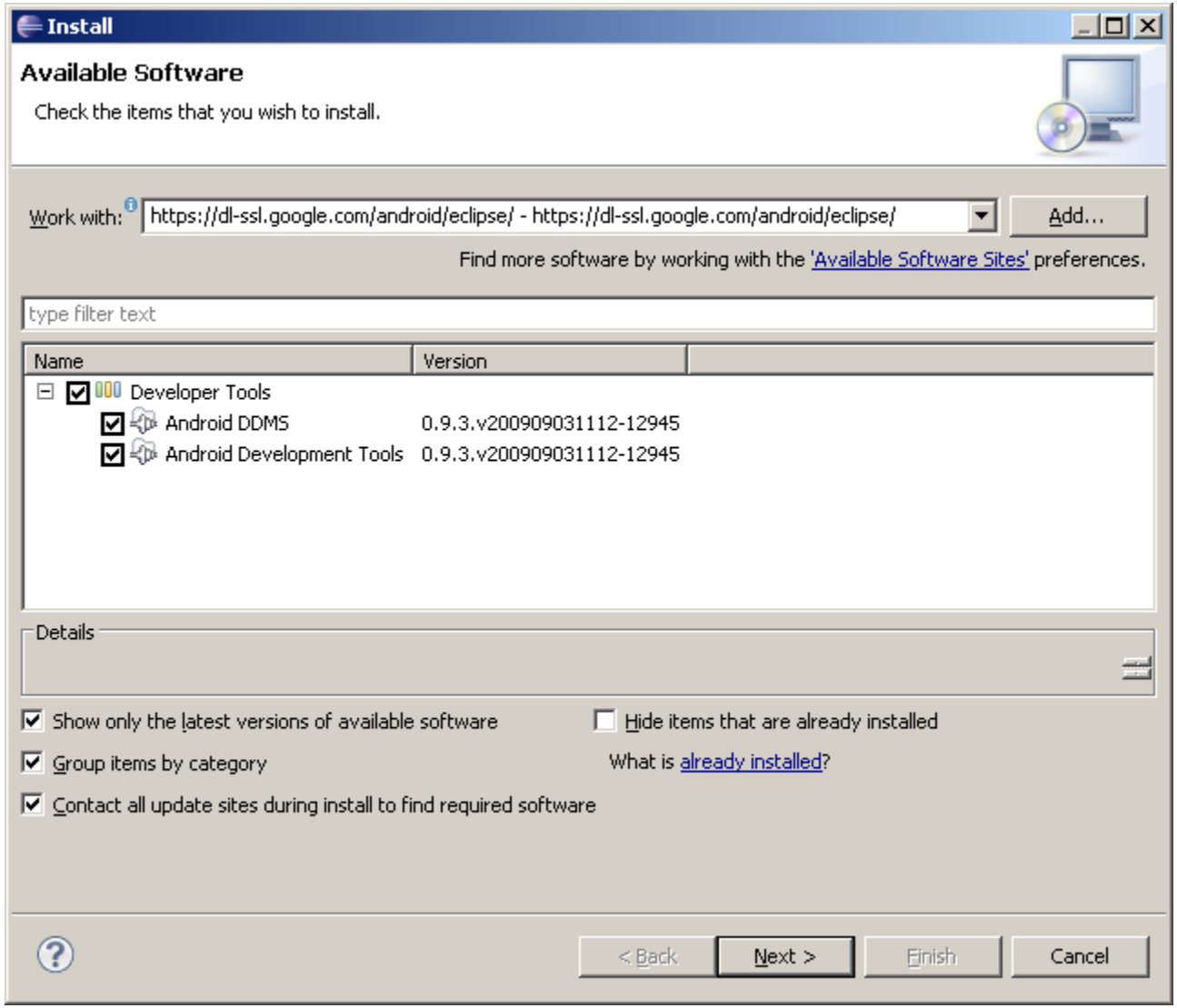
- Java JDK 6, Standard Edition (not only JRE)
  - http://java.sun.com/javase/downloads/index.jsp
- Eclipse IDE (3.4 or newer)
  - http://www.eclipse.org/downloads/
  - Eclipse IDE for Java Developers
- Android SDK starter package (depending on your platform)
  - http://dl.google.com/android/android-sdk_r08-windows.zip
  - http://dl.google.com/android/android-sdk_r08-mac_86.zip
  - http://dl.google.com/android/android-sdk_r08-linux_86.tgz
- See also: “Quick Steps”
  - http://developer.android.com/sdk/index.html

Installation

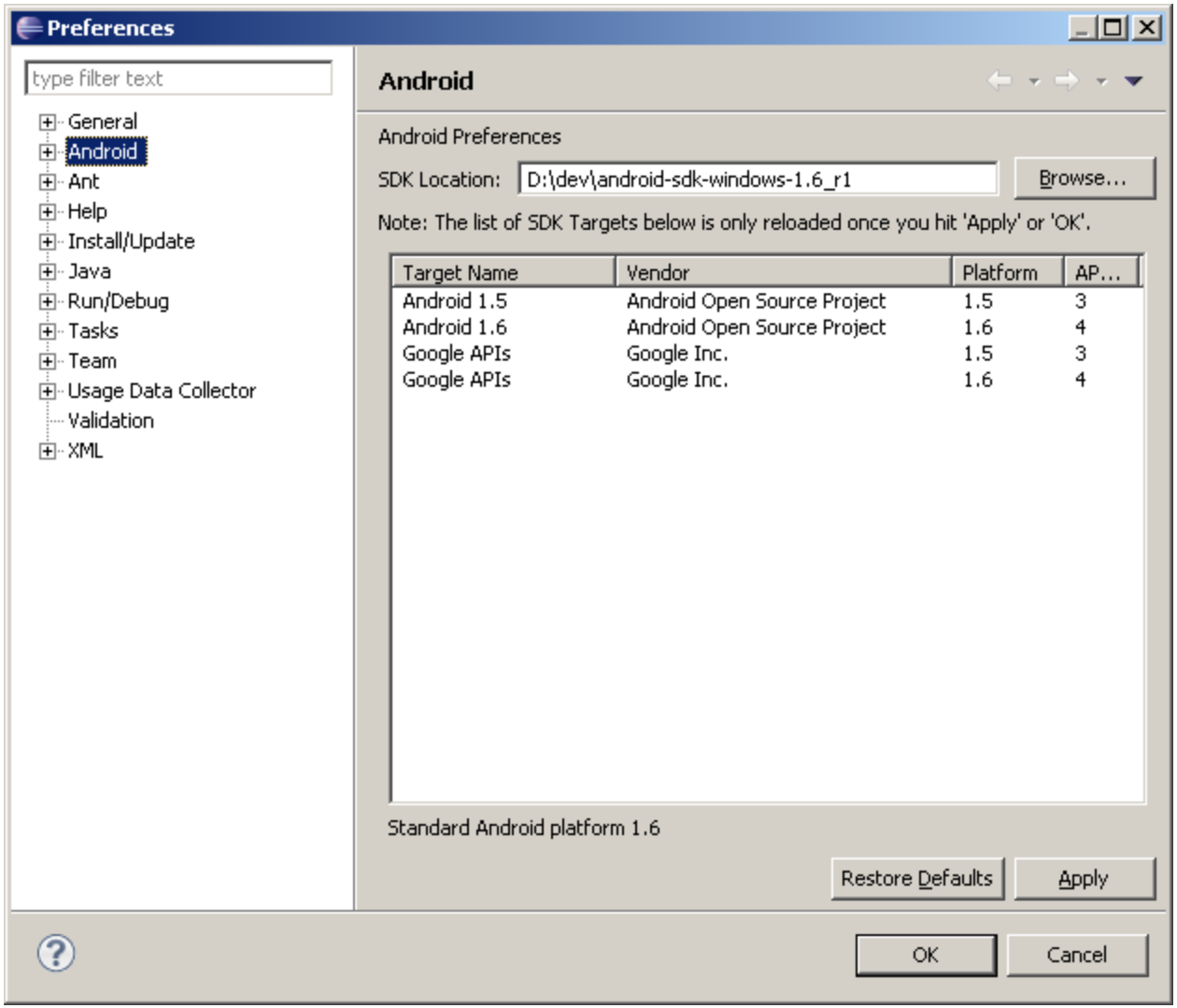
- Start Eclipse
- In Eclipse: Install Android SDK
  - Menu: Help, Install New Software…
  - https://dl-ssl.google.com/android/eclipse/
- Point Eclipse to the Android SDK starter package
  - Menu: Window, preferences, Android, SDK Location
  - /soft/IFI/lang/android-sdk-r10/iX86-unknown-linux
- In Eclipse: Android SDK and AVD Manager
  - Window / Android SDK and AVD Manager
  - New... / Virtual Devices / 2.2 (oder 1.6) mit Google API
- Mobile Phone
  - Anwendungen, Entwicklung: USB-Debugging, ...

In Eclipse: Install New Software...

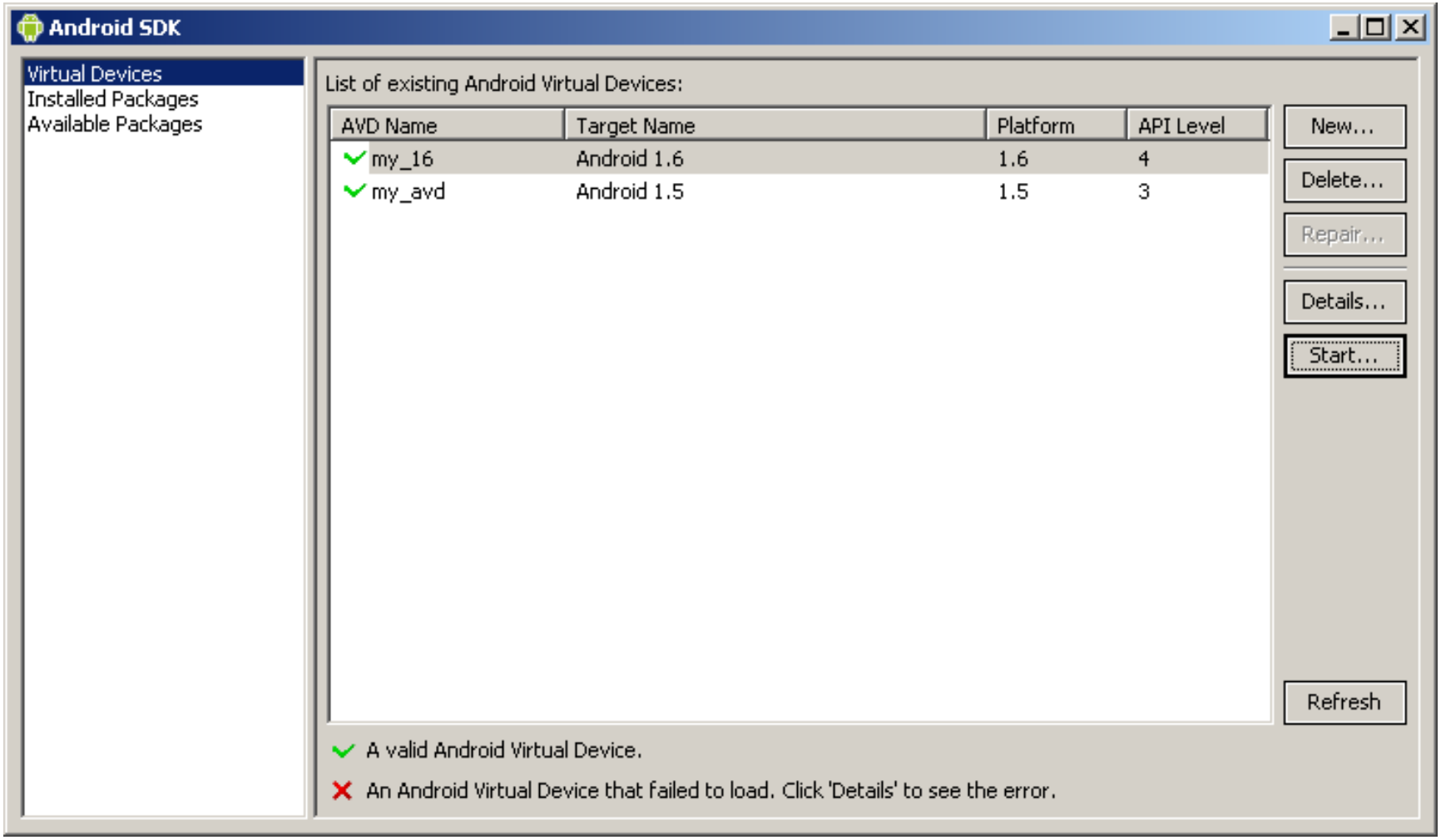
Android Plugin – https://dl-ssl.google.com/android/eclipse/



Set Path to Android SDK Starter Package



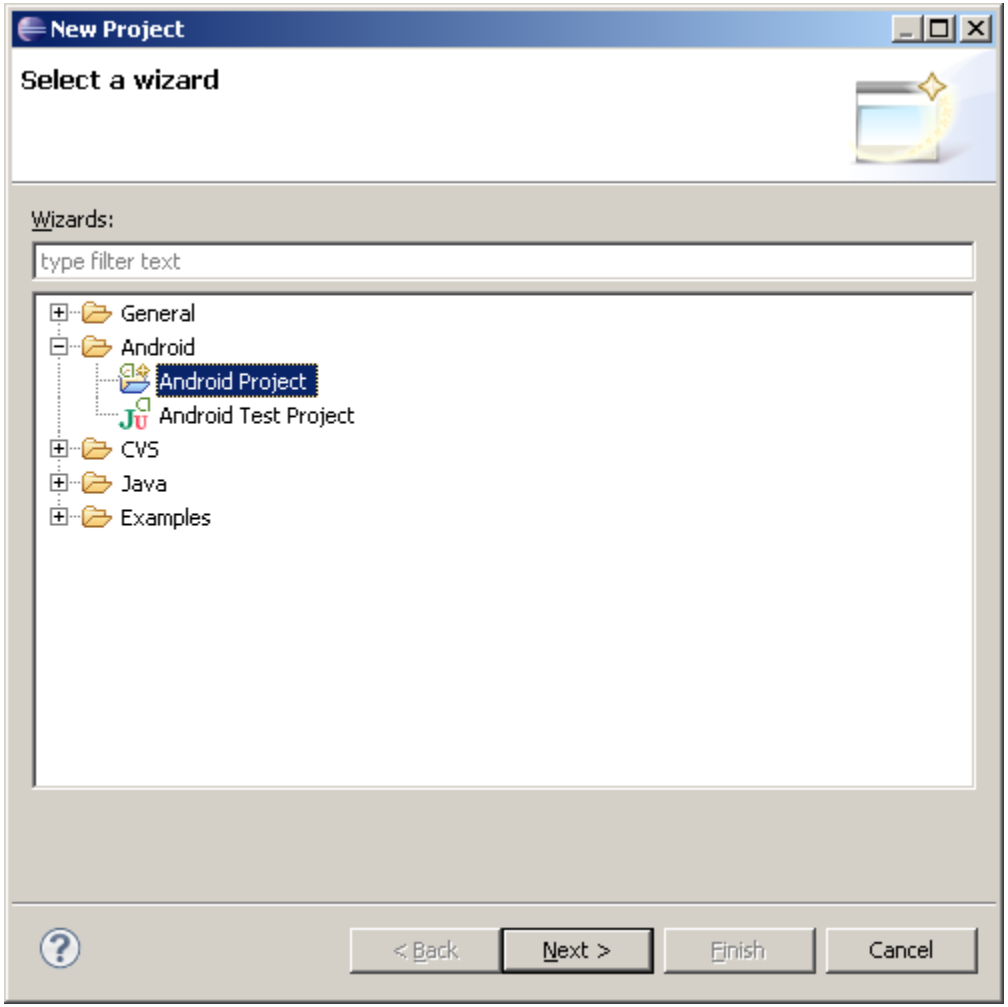
Define Android Virtual Device

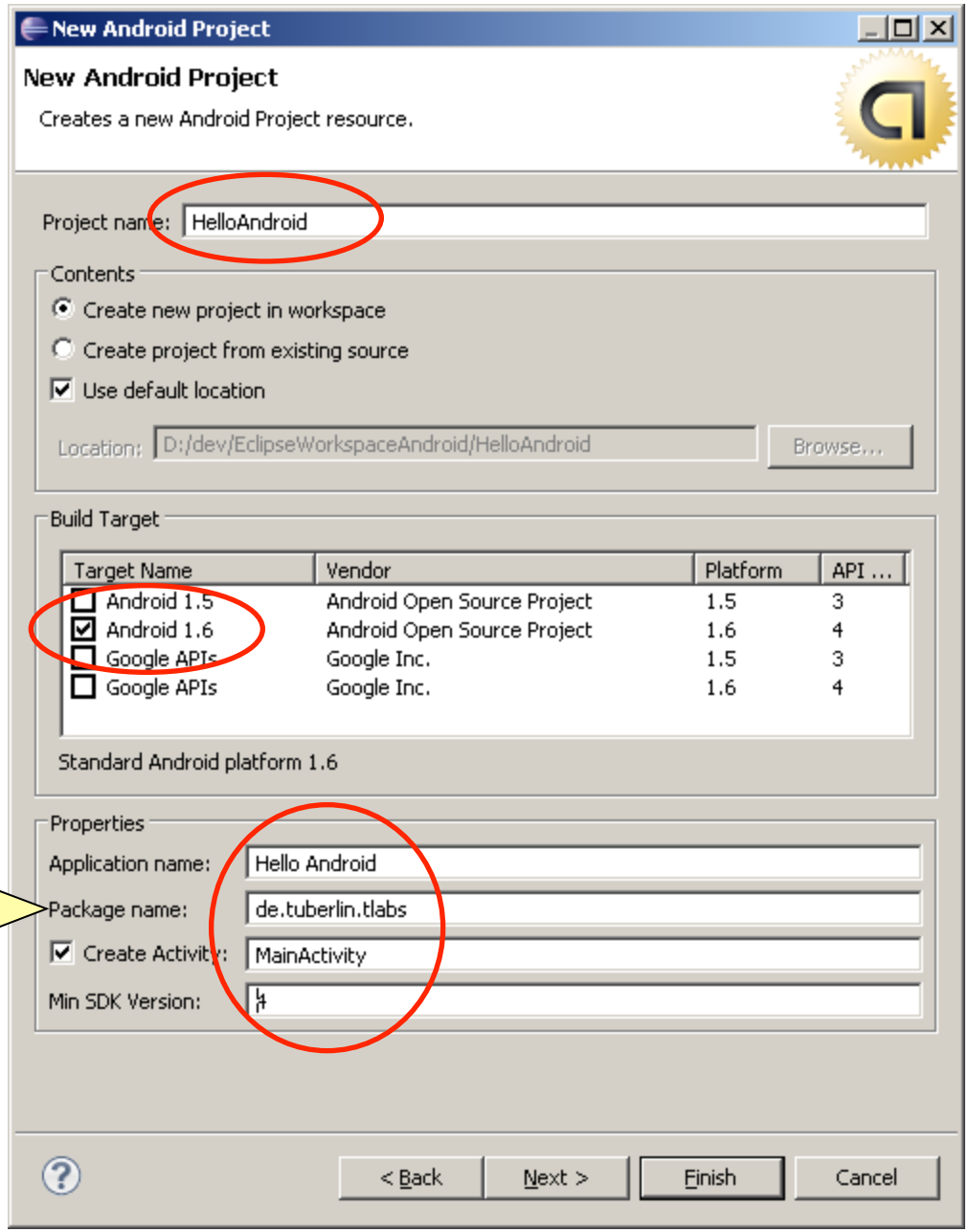


“Hello World”

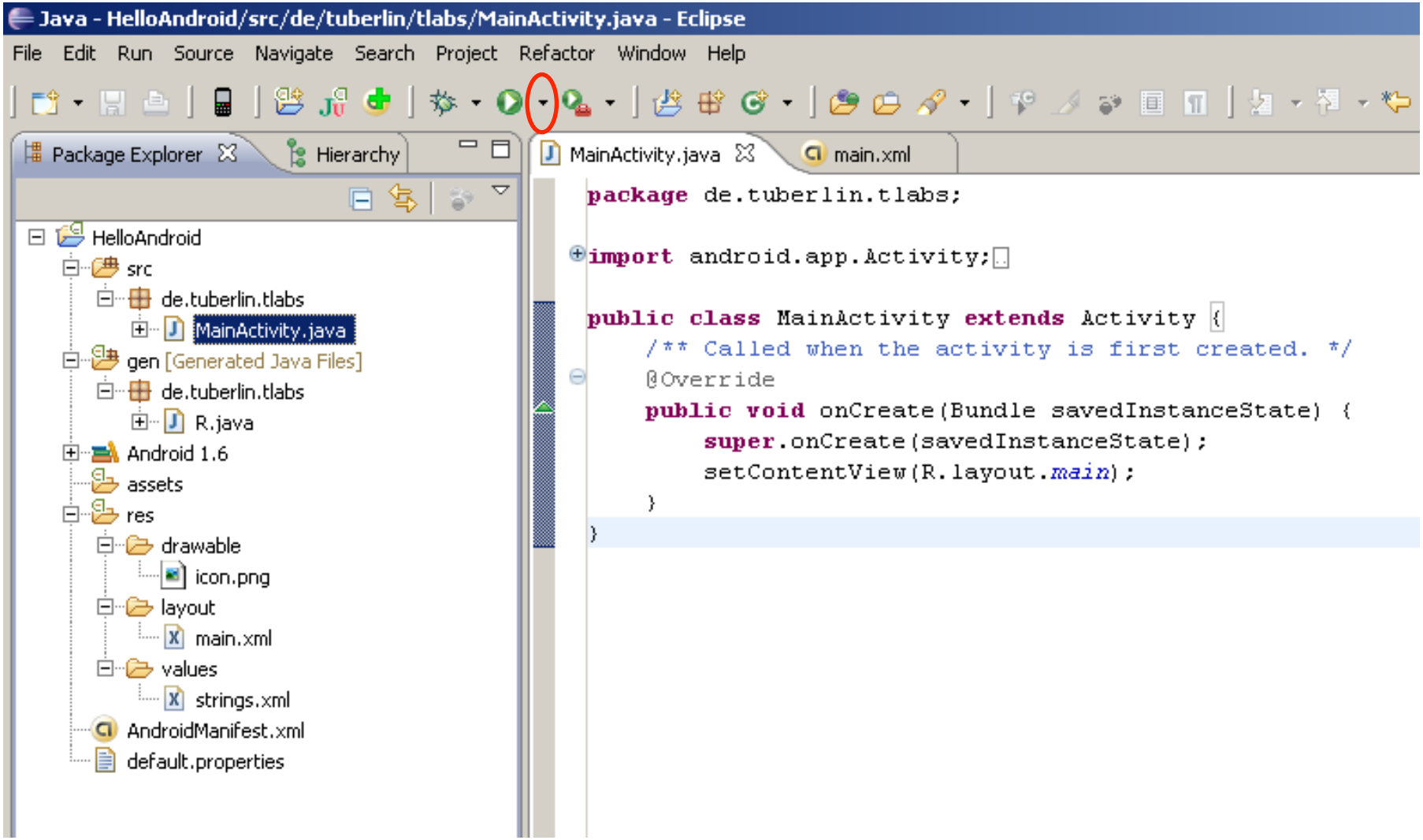
Creating Your First Android Project

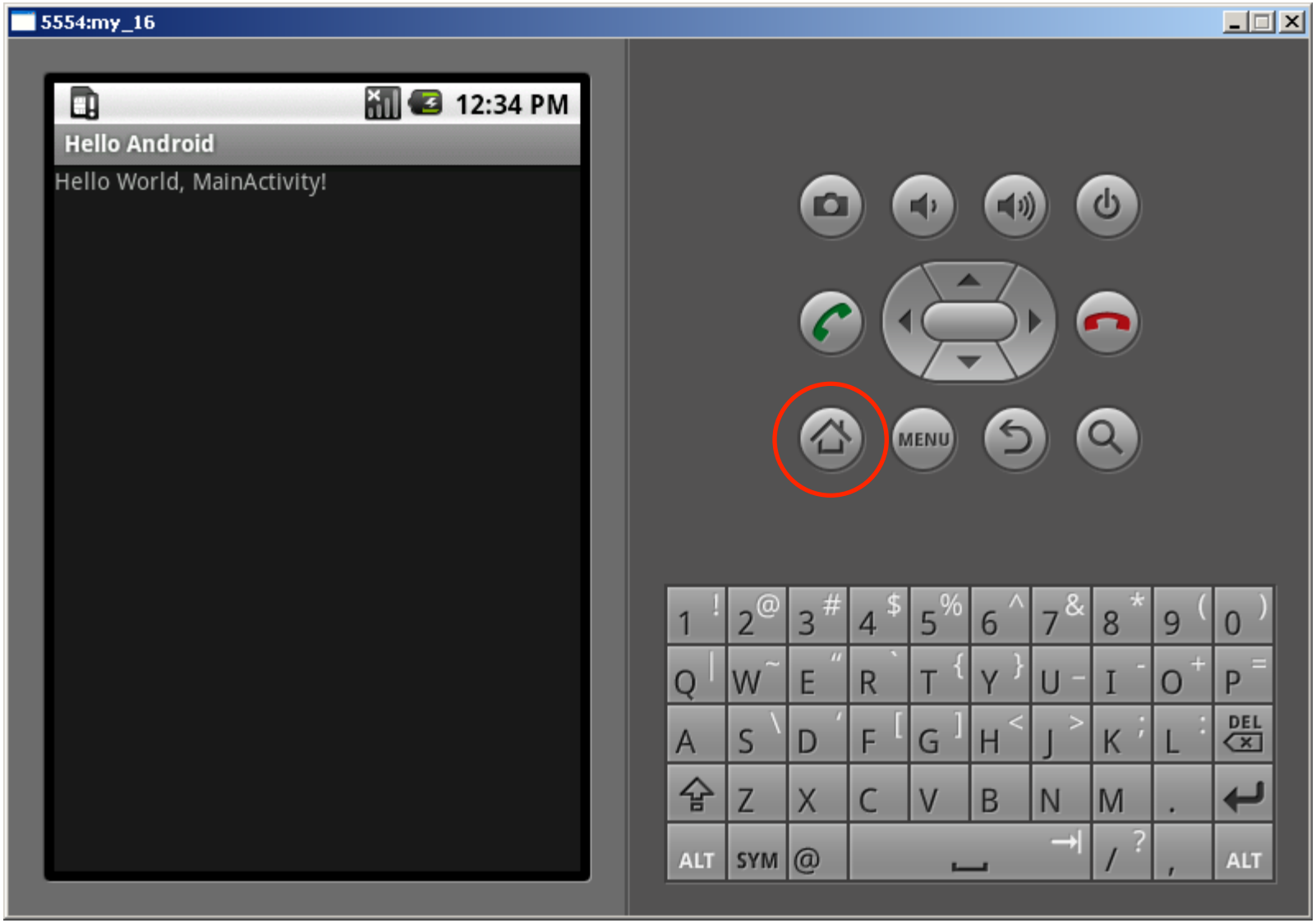
File → New Project → Android → Android Project

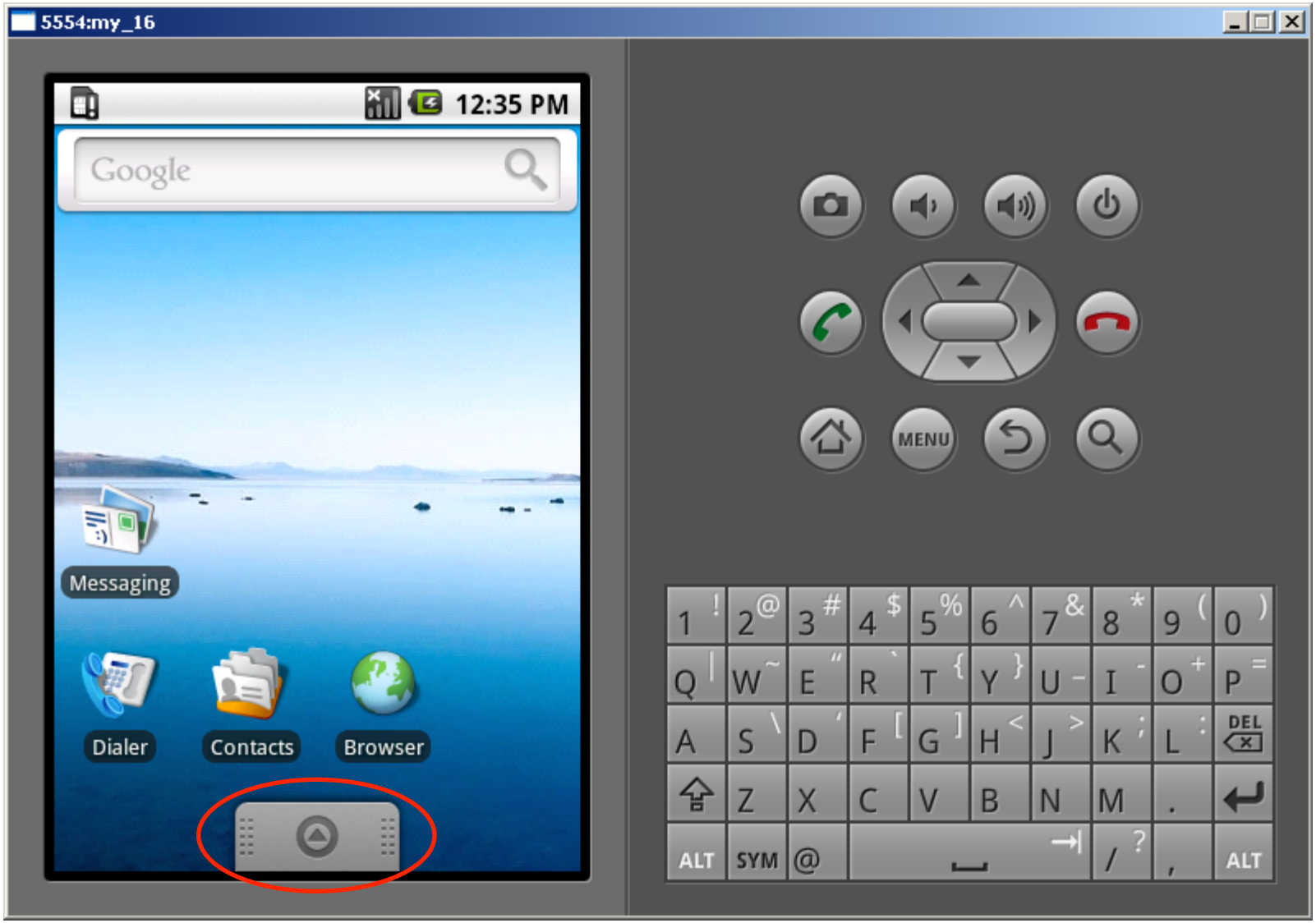


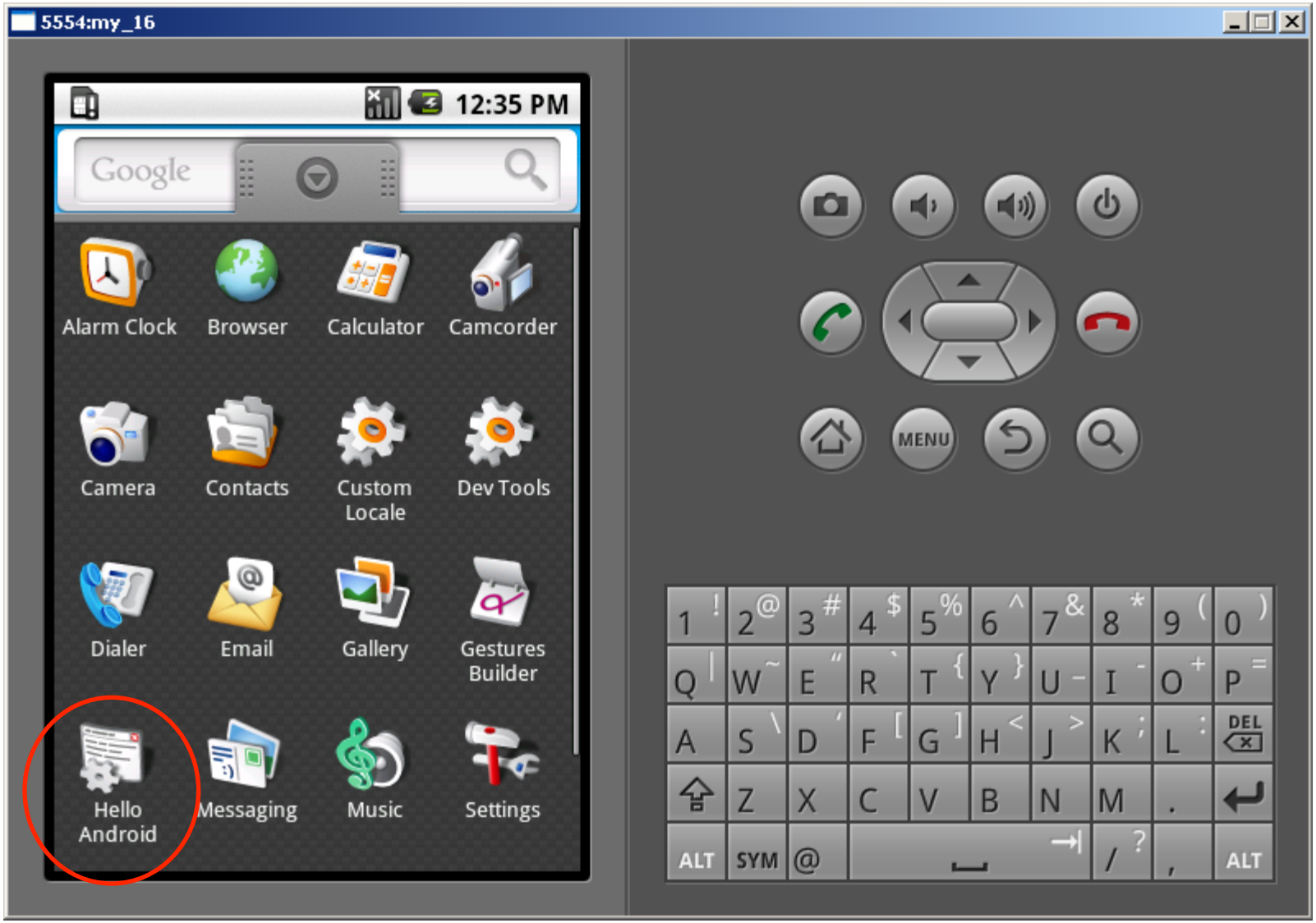


Uniquely identifies the application!


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);
}
}

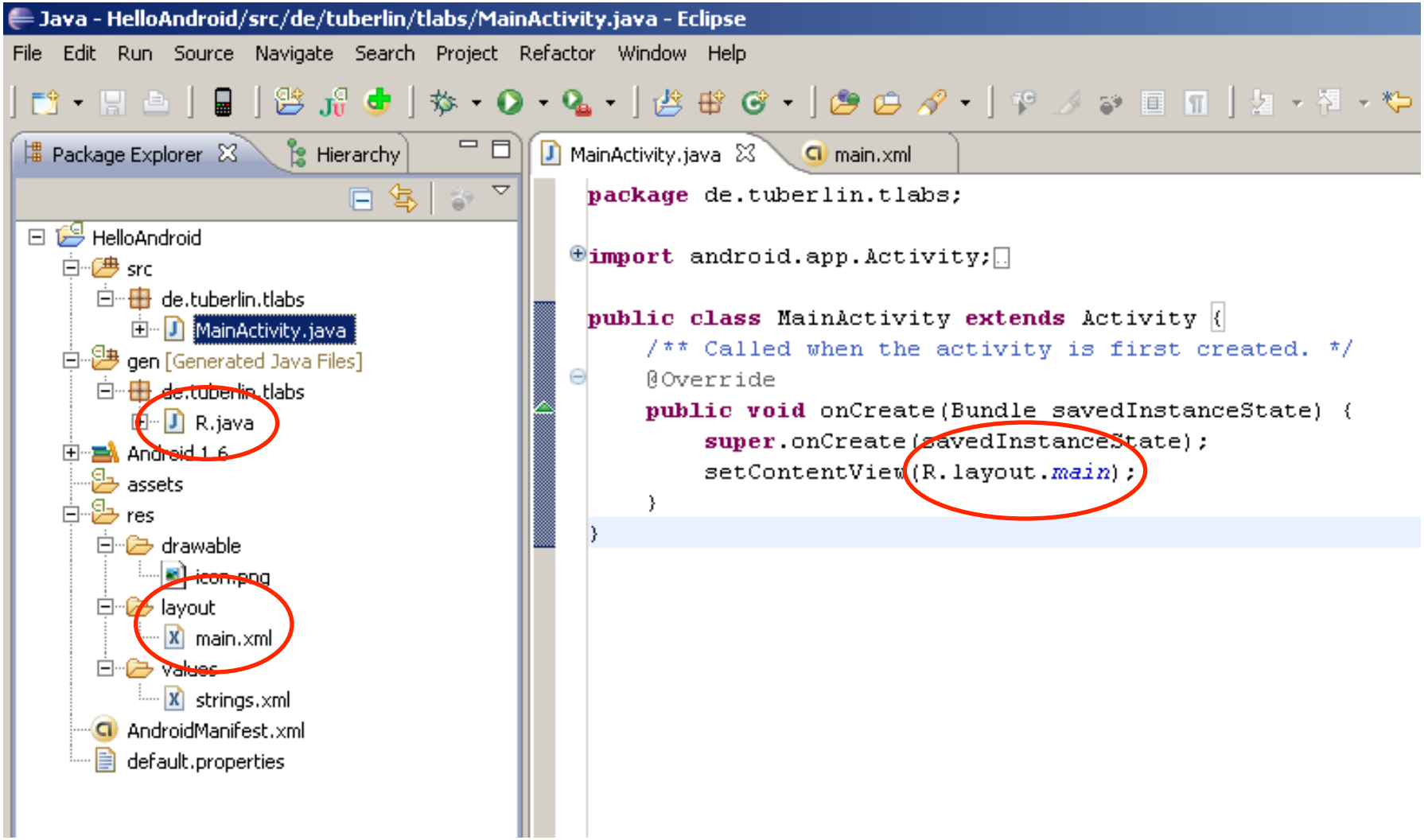

5554:my_16
12:34 PM
Hello Android
Hello World, MainActivity!
MENU


5554:my_16
12:35 PM
Google
Messaging
Dialer
Contacts
Browser
MENU
ALT
SYM
ALT


5554:my_16
12:35 PM
Google
Alarm Clock
Browser
Calculator
Camcorder
Camera
Contacts
Custom Locale
Dev Tools
Dialer
Email
Gallery
Gestures Builder
Hello Android
Messaging
Music
Settings
MENU
ALT
SYM
DEL

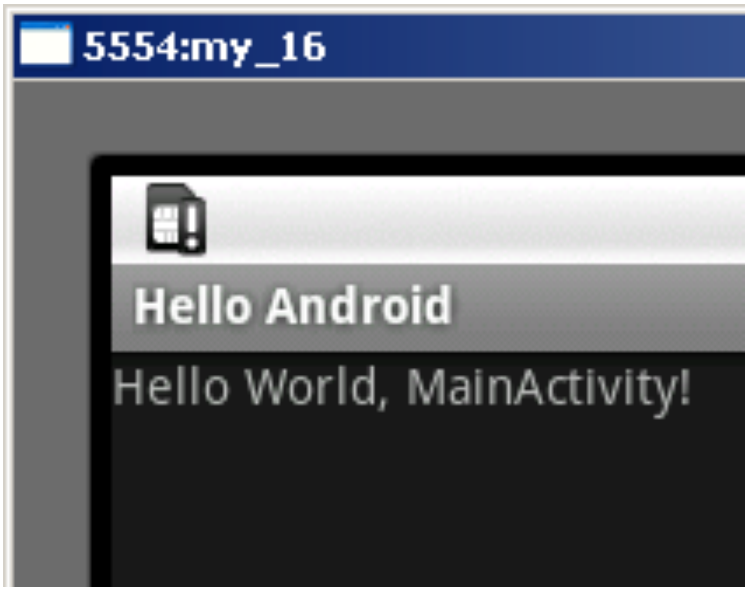
Exercise:

Install Android + Create “Hello World”


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

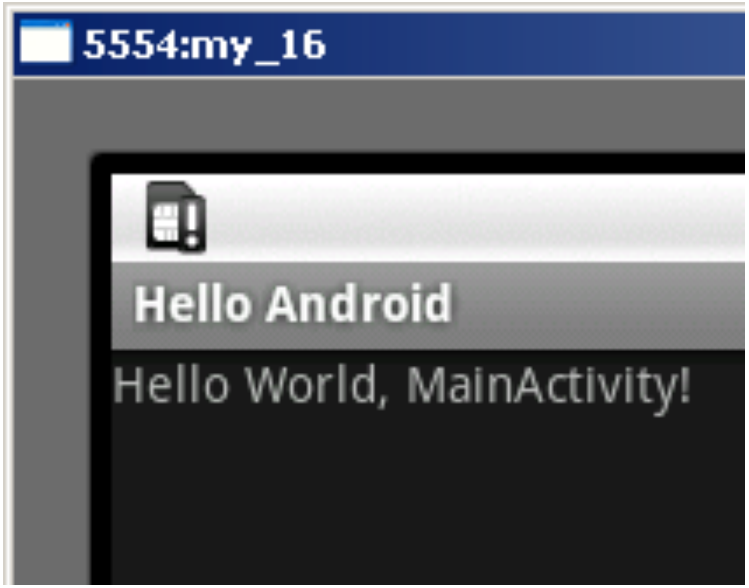
```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:orientation="vertical"
    android:layout_width="fill_parent"
    android:layout_height="fill_parent"
    >
<TextView
    android:layout_width="fill_parent"
    android:layout_height="wrap_content"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

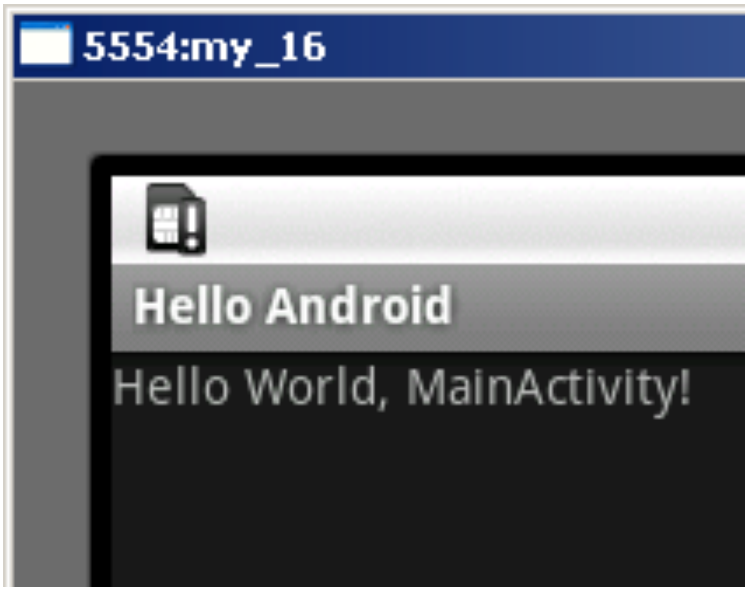
- Default language in res/values/strings.xml
- Localized languages in res/values-xx ← language qualifier
  - French in res/values-fr/strings.xml
  - Hindi in res/values-hi/strings.xml
  - etc.

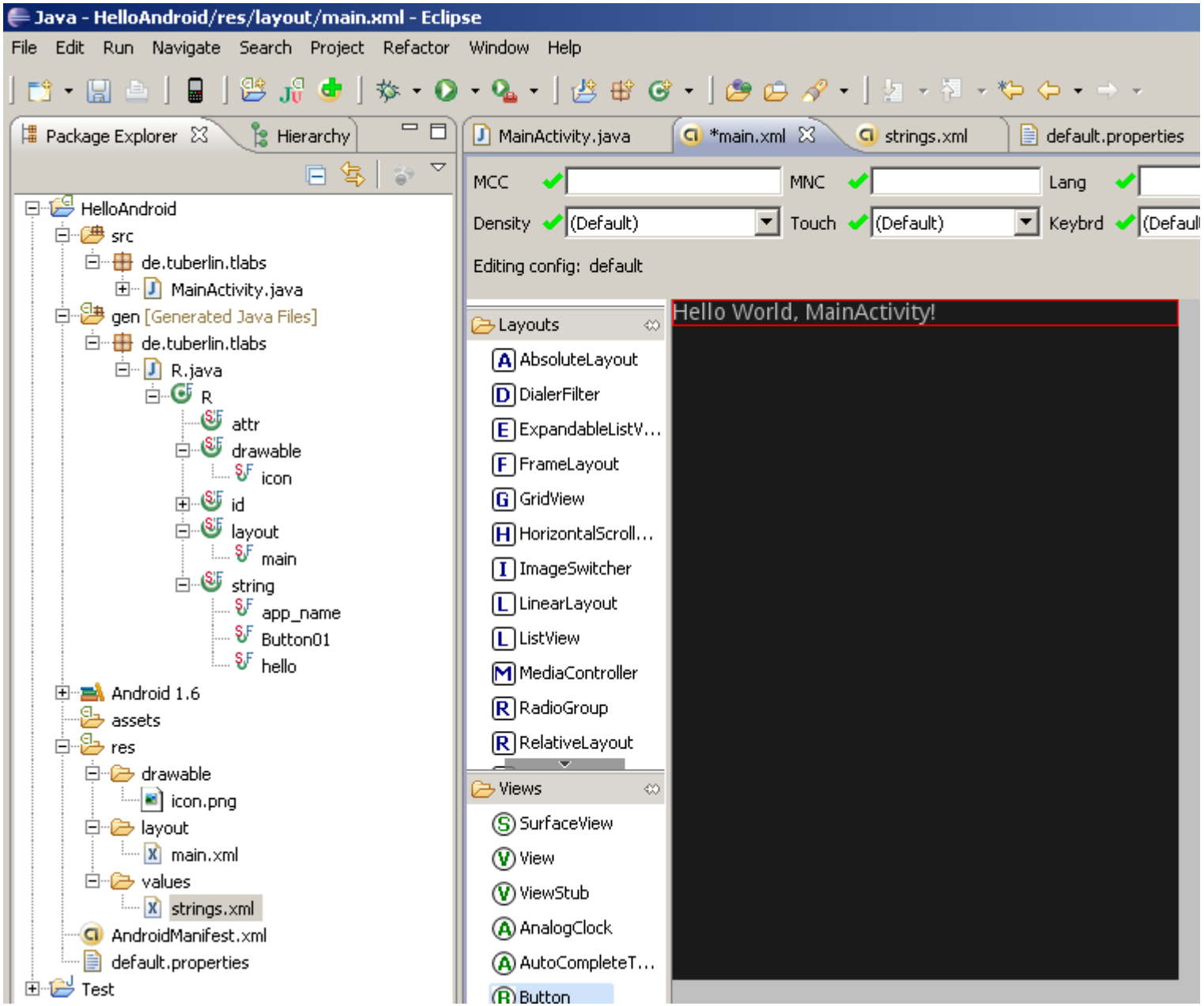


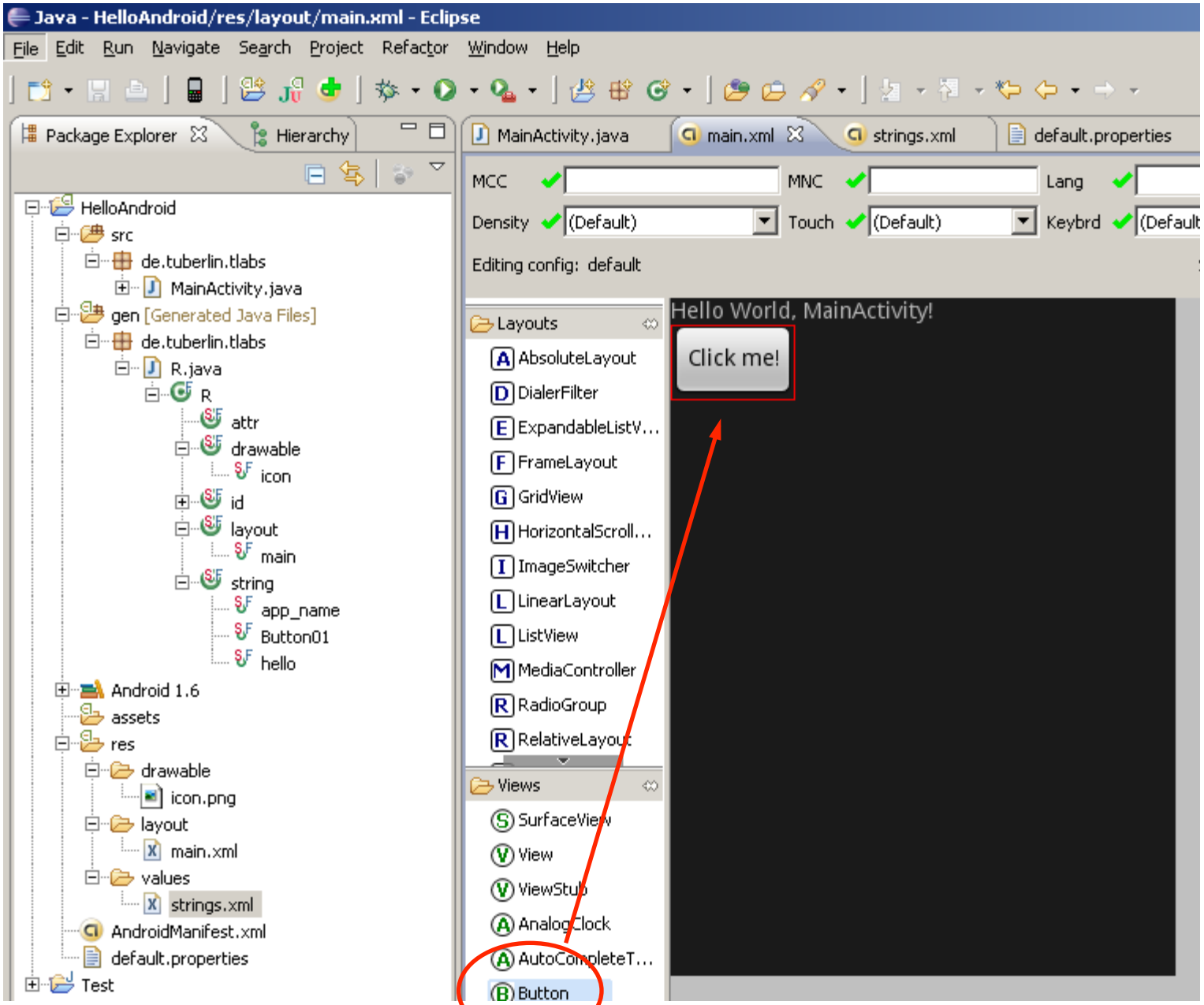
R.java

```
/* 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
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!

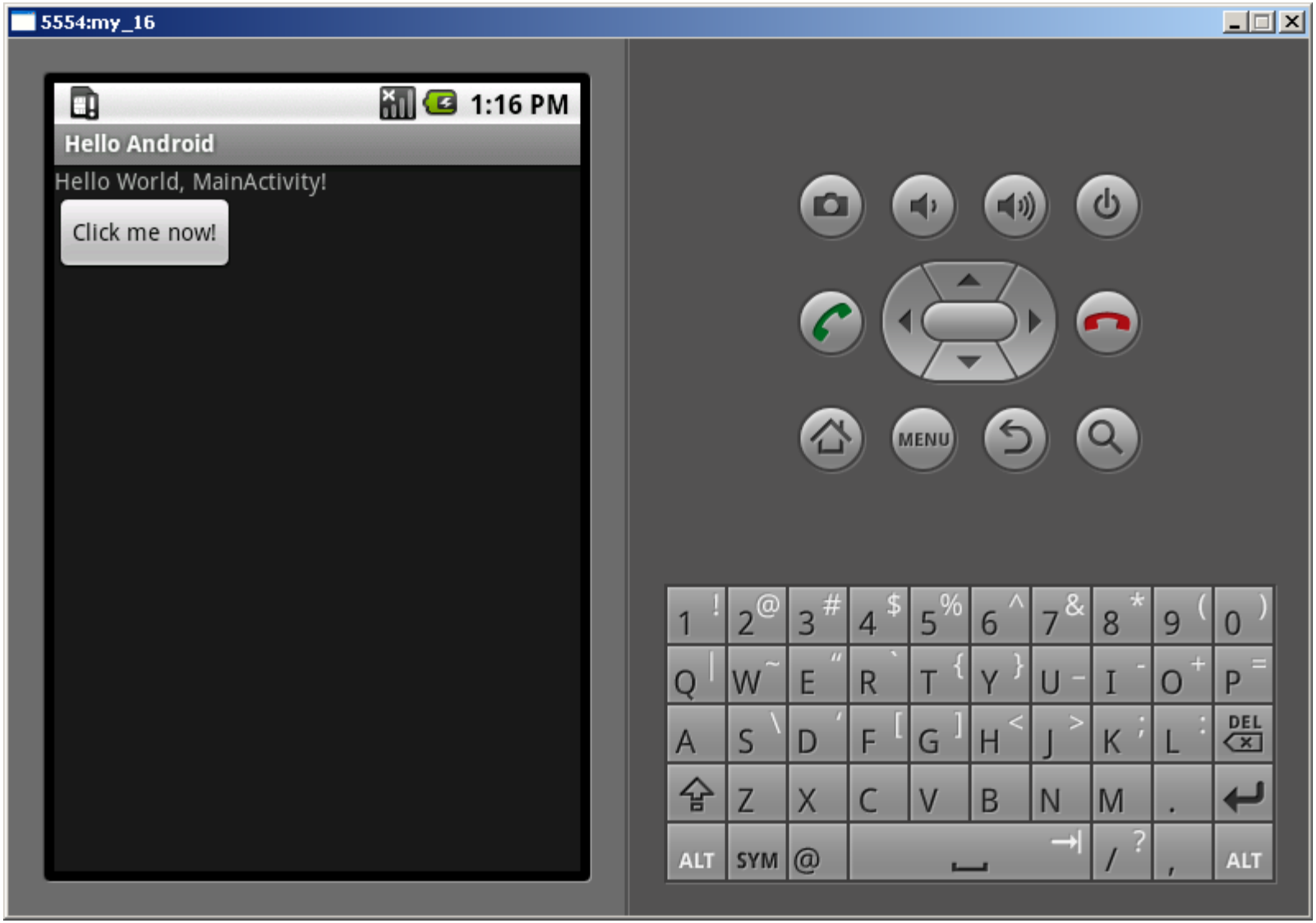

Java - HelloAndroid/res/layout/main.xml - Eclipse
File Edit Run Navigate Search Project Refactor Window Help
Package Explorer
Hierarchy
MainActivity.java
main.xml
strings.xml
default.properties
HelloAndroid
src
de.tuberlin.tlabs
MainActivity.java
gen [Generated Java Files]
de.tuberlin.tlabs
R.java
R
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/Button01"
    android:id="@+id/Button01"
    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="Button01">Click me now!</string>
</resources>
```


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

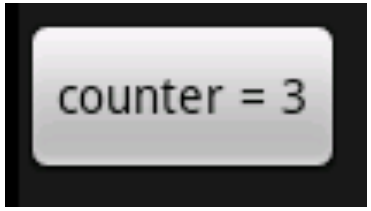
Handling Button Click Events

- XML

```
<Button android:id="@+id/button1" android:text="Basic Button"
    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:

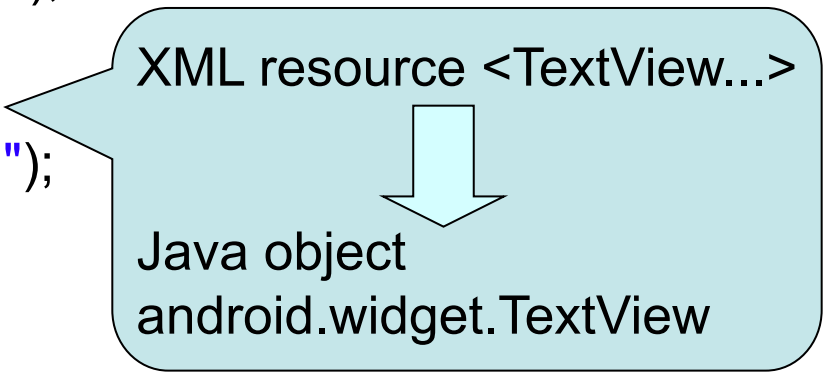
- Add a button to “Hello World”

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);
    }

}
```

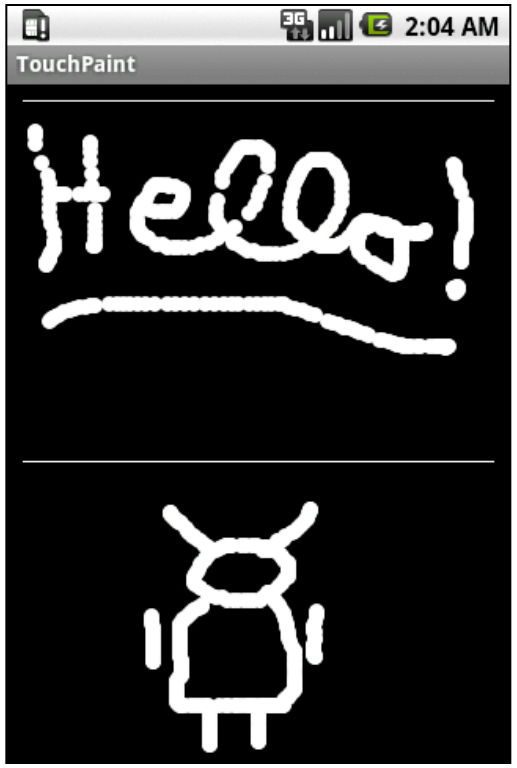


Touch Input: MotionEvent

- Method View.onTouchEvent(MotionEvent e)
- Motion event data
  - x, y, time, action, source, pressure, size
- Sources depend on hardware
  - Mouse, pen, finger, trackball
- Actions
  - ACTION_DOWN
  - ACTION_MOVE
  - ACTION_UP
  - ACTION_CANCEL
- Motion history
  - Sequence of coordinates between events

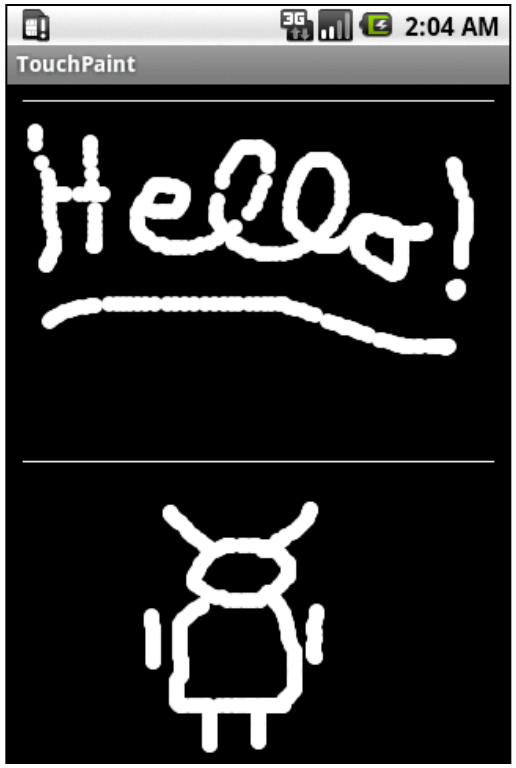
Touch Input Painting

```
public class TouchPaint extends Activity {

    private MyView myView;

    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        myView = new MyView(this);
        setContentView(myView);
    }
}
```



Touch Input Painting

```
public class MyView extends View {
    private final Paint paint = new Paint();
    private int x = 0, y = 0;

    public MyView(Context c) {
        super(c);
        paint.setARGB(255, 255, 255, 255);
    }

    protected void onDraw(Canvas c) {
        c.drawCircle(x, y, 3, paint);
    }

    public boolean onTouchEvent(MotionEvent e) {
        x = (int)e.getX(); y = (int)e.getY();
        invalidate();
        return true;
    }
}
```



Concepts so far

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