

CS 696 Emerging Web and Mobile Technologies
Spring Semester, 2011
Doc 18 Android GUI & Intents
Mar 22, 2011

Copyright ©, All rights reserved. 2011 SDSU & Roger Whitney, 5500 Campanile Drive, San Diego, CA 92182-7700 USA. OpenContent (<http://www.opencontent.org/opl.shtml>) license defines the copyright on this document.

References

Android Developer's Guide, <http://developer.android.com/guide/index.html>

CS 696 Mobile Phone Application Development, Fall 2009,
<http://www.eli.sdsu.edu/courses/fall09/cs696/notes/index.html>

Mercurial & Example Repository

Mercurial

<http://mercurial.selenic.com/wiki/>

Distributed Source Control System

Written in Python

Mercurial Eclipse

<http://javaforge.com/project/HGE>

Eclipse Plugin for Mercurial

Adds functionality to Team menu item

Course Examples Mercurial Repository

<http://bismarck.sdsu.edu/cgi-bin/hg/>

Web interface

- View online

- Download zipped file

- You can import the unzipped file using Mercurial Eclipse plugin

Mercurial clone

- hg clone <http://bismarck.sdsu.edu/cgi-bin/hg/android/HelloWorld>

- You can import the unzipped file using Mercurial Eclipse plugin

Load directly into Eclipse

- Import...

- Select "Clone repository using MercurialEclipse"

- Enter URL:

 - <http://bismarck.sdsu.edu/cgi-bin/hg/android/HelloWorld>

- You may have to clean the project

History

Graph	Changeset	Tag	Branch	User	Date	Summary
●	8:d975fdcb4c8f	tip		whitney	2009-09-07 2	Added two menus which add to the text field
●	7:643f24e4beb7			whitney	2009-09-07 2	Modified layout of button example
●	6:600b9fc894bc			whitney	2009-09-07 2	Added button, which activity listens to
●	5:f9b052f62914			whitney	2009-09-07 2	Made HelloAndroid a OnClickListener
●	4:9243a0dc450a			whitney	2009-09-07 2	Use EditText to display message
●	3:2a6af1fb6937			whitney	2009-09-07 1	Added Logging to the example
●	2:10d6e0414310			whitney	2009-09-07 1	Corrected import for String resource example
●	1:bdc1698677a5			whitney	2009-09-07 1	Hello World using String Resources
●	0:210e16e0a592			whitney	2009-09-06 1	Start

Warning

Example use old version of Android

Can cause lots of wasted time trying to get them to work

Have to add existing version of Android

Make sure that the android SDK is included in build path

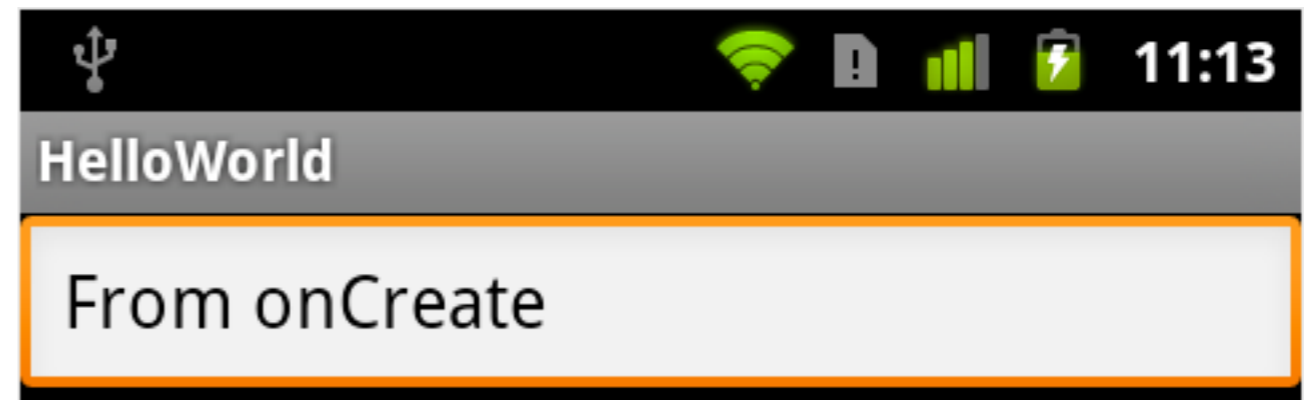
GUI Examples

EditText Example

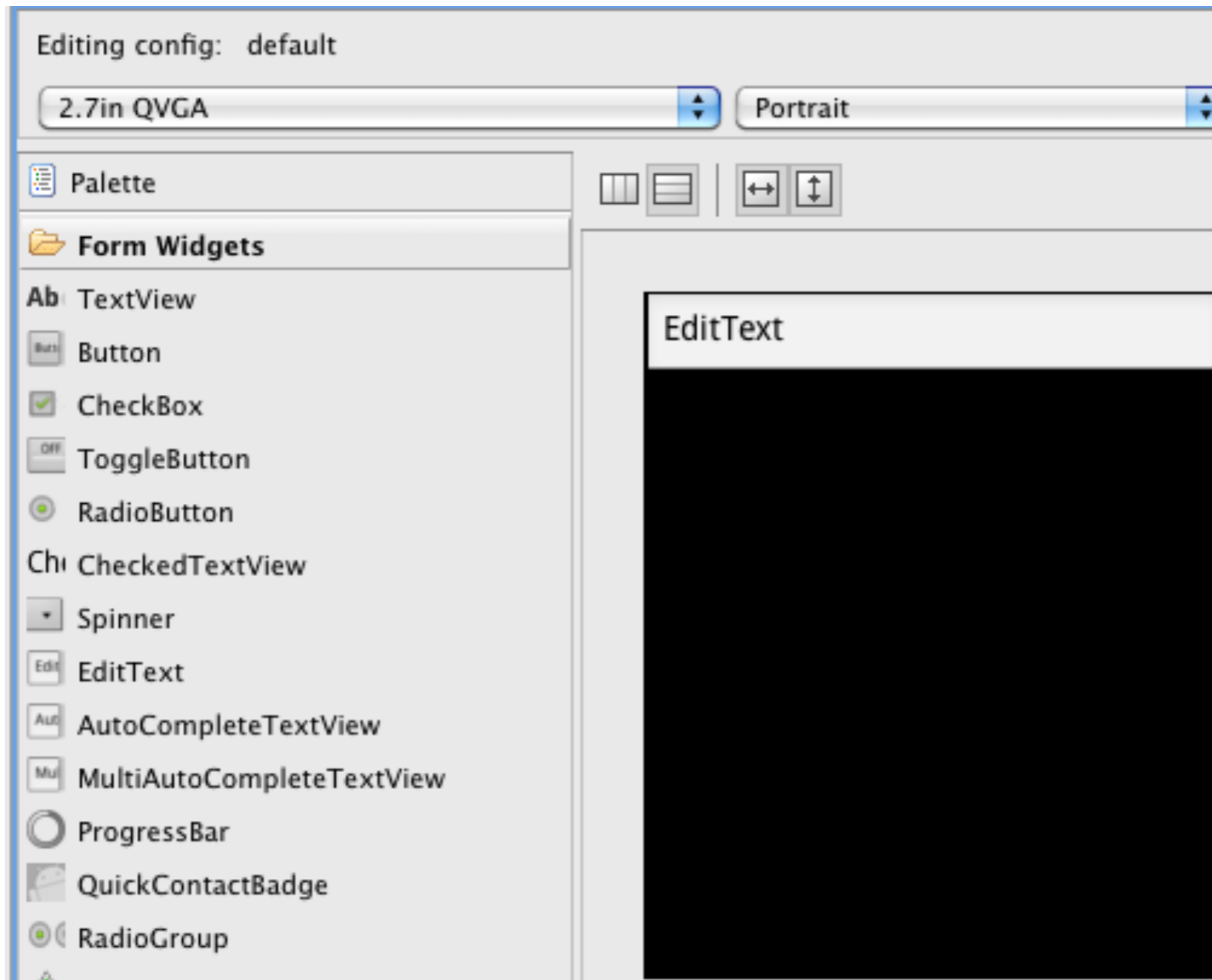
Issues

Create UI widget using GUI builder

Accessing widget in code



GUI builder



main.xml with id

```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:orientation="vertical"
    android:paddingLeft="6dip"
    android:paddingRight="6dip"
    android:paddingBottom="3dip">
<EditText android:id="@+id/message"
    android:layout_width="fill_parent"
    android:layout_height="fill_parent"
    android:text="Hello, Android from xml2"/>
</LinearLayout>
```

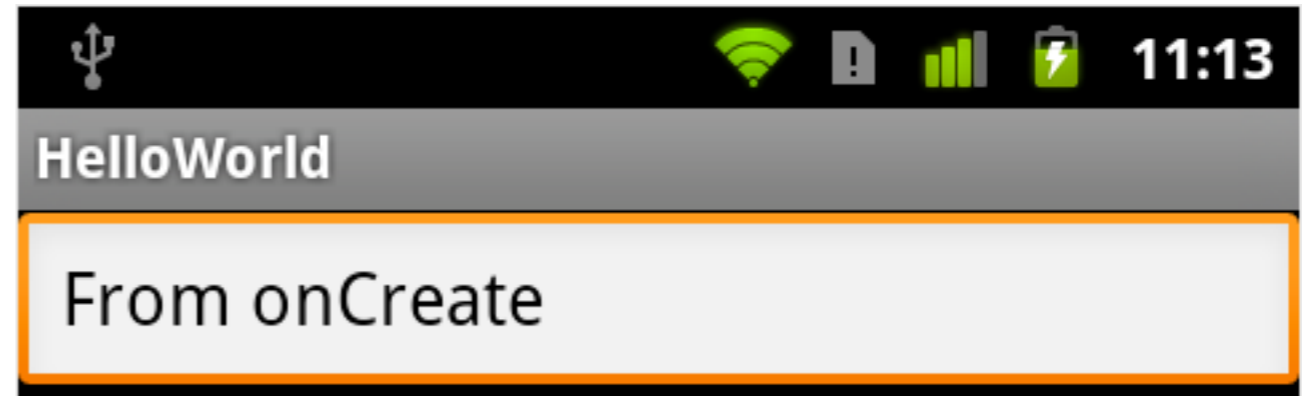
Accessing widget

```
package sdsu.cs696;

import android.app.Activity;
import android.os.Bundle;
import android.widget.EditText;

public class HelloAndroid extends Activity {
    private EditText messageText;

    public void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.main);
        messageText = (EditText) this.findViewById(R.id.message);
        messageText.setText("From onCreate");
    }
}
```



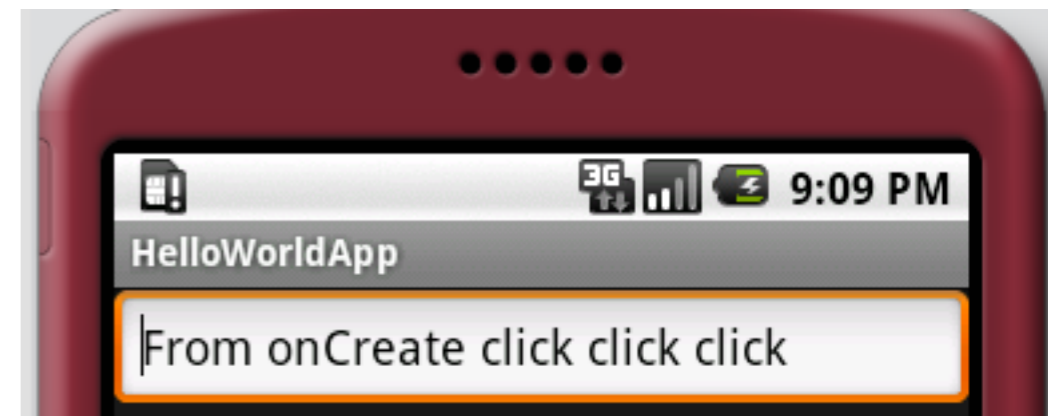
R.java

```
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 message=0x7f050000;  
    }  
    public static final class layout {  
        public static final int main=0x7f030000;  
    }  
    public static final class string {  
        public static final int Foo=0x7f040002;  
        public static final int app_name=0x7f040001;  
        public static final int hello=0x7f040000;  
    }  
}
```

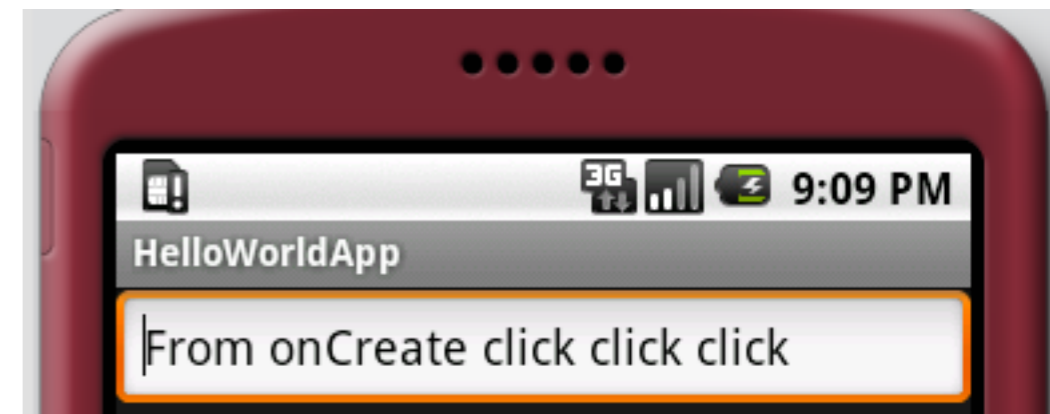
Click Listener Example

Issue

Reacting to events from UI widgets



Click Listener Example



```
public class HelloAndroid extends Activity implements View.OnClickListener {  
    private EditText messageText;  
  
    public void onCreate(Bundle savedInstanceState) {  
        super.onCreate(savedInstanceState);  
        setContentView(R.layout.main);  
        messageText = (EditText) this.findViewById(R.id.message);  
        messageText.setText("From onCreate");  
        messageText.setOnClickListener(this);  
    }  
  
    public void onClick(View v) {  
        messageText.setText(messageText.getText() + " click");  
    }  
}
```

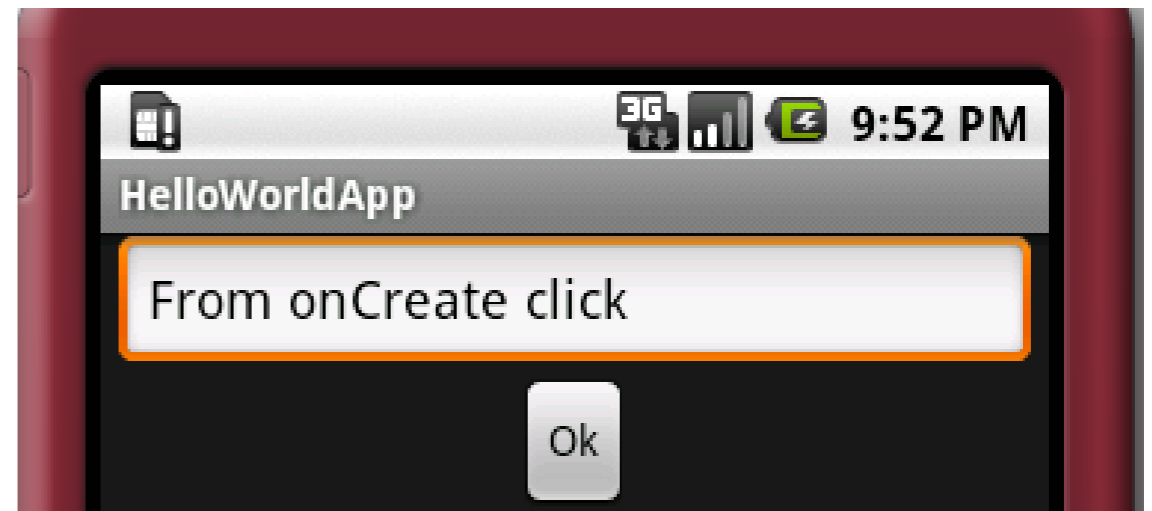

Listeners

<code>GestureDetector.OnGestureListener</code>	Notify when gestures occur
<code>MenuItem.OnMenuItemClickListener</code>	a menu item is clicked.
<code>View.OnClickListener</code>	a view is clicked.
<code>View.OnCreateContextMenuListener</code>	the context menu for this view is being built.
<code>View.OnFocusChangeListener</code>	the focus state of a view changed.
<code>View.OnKeyListener</code>	a key event is dispatched to this view.
<code>View.OnLongClickListener</code>	a view has been clicked and held.
<code>View.OnTouchListener</code>	a touch event is dispatched to this view.
<code>ViewGroup.OnHierarchyChangeListener</code>	the hierarchy within this view changed.
<code>ViewStub.OnInflateListener</code>	ViewStub has successfully inflated its layout resource.
<code>ViewTreeObserver.OnGlobalFocusChangeListener</code>	the focus state within the view tree changes.
<code>ViewTreeObserver.OnGlobalLayoutListener</code>	the global layout state or the visibility of views within the view tree changes.
<code>ViewTreeObserver.OnPreDrawListener</code>	the view tree is about to be drawn.
<code>ViewTreeObserver.OnTouchModeChangeListener</code>	the touch mode changes.

Button Example

Issue

Button events



Button Example

```
import android.app.Activity;
import android.os.Bundle;
import android.view.View;
import android.widget.Button;
import android.widget.EditText;
```

```
public class HelloAndroid extends Activity implements View.OnClickListener {
    private EditText messageText;

    public void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.main);
        messageText = (EditText) this.findViewById(R.id.message);
        messageText.setText("From onCreate");
        Button ok = (Button) findViewById(R.id.ok);
        ok.setOnClickListener(this);
    }

    public void onClick(View v) {
        messageText.setText(messageText.getText() + " click");
    }
}
```

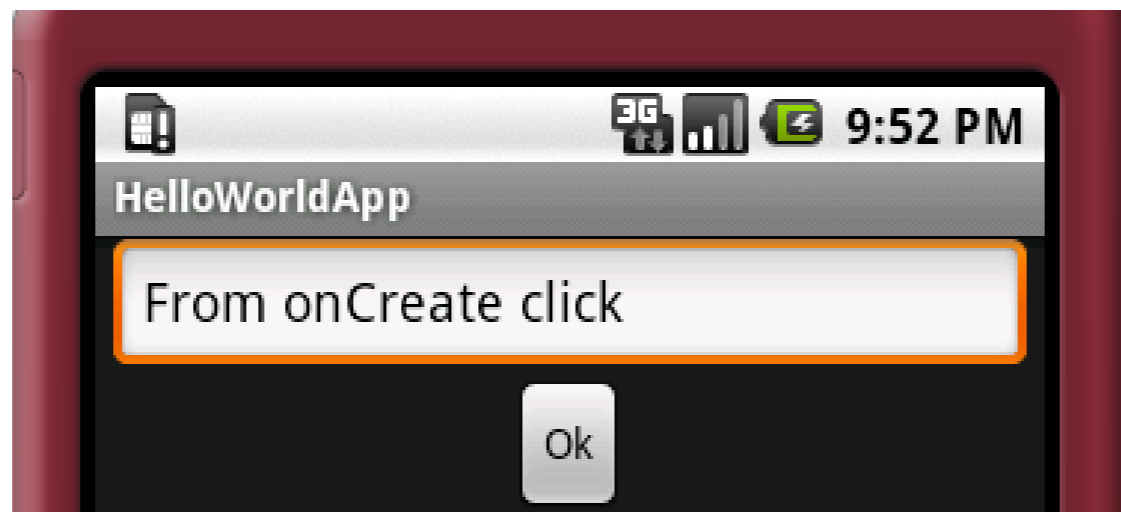
res/layout/main.xml

```
<?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"
    android:paddingLeft="6dip"
    android:paddingRight="6dip"
    android:paddingBottom="3dip">

<EditText android:text="First Edit text"
    android:id="@+id/message"
    android:autoText="true"
    android:ems="25"
    android:capitalize="sentences"
    android:layout_width="fill_parent"
    android:layout_height="wrap_content"/>
<Button
    android:id="@+id/ok"
    android:layout_gravity="center"
    android:text="@string/button_ok"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"/>
</LinearLayout>
```

res/values/strings.xml

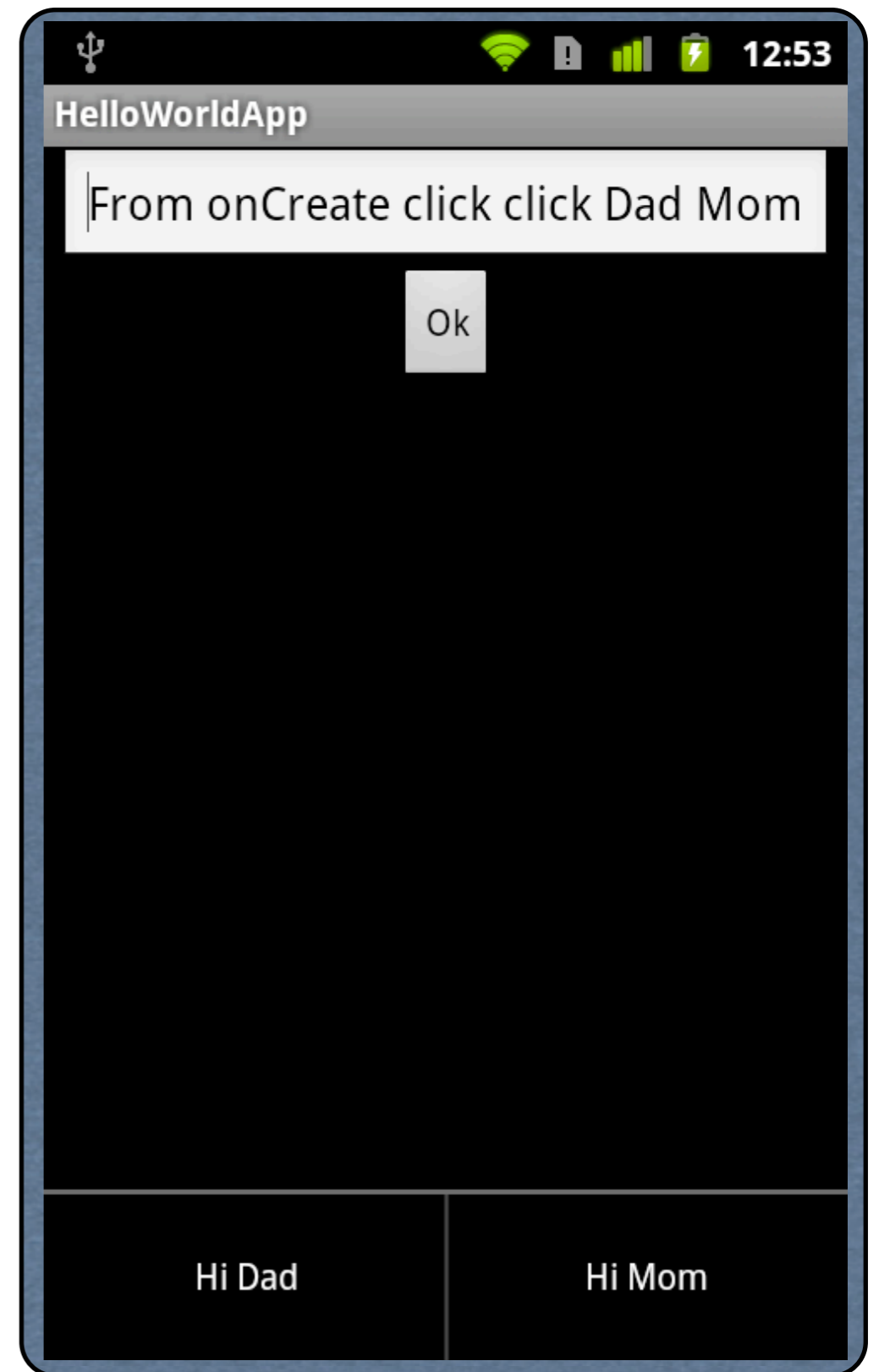
```
<?xml version="1.0" encoding="utf-8"?>  
<resources>  
  <string name="hello">Hello using String resources!</string>  
  <string name="app_name">HelloWorldApp</string>  
  <string name="button_ok">Ok</string>  
</resources>
```



Menu Example

How to create menus

How to respond to menus



Menus Example

```
import android.app.Activity;
import android.os.Bundle;
import android.view.Menu;
import android.view.MenuItem;
import android.view.View;
import android.widget.Button;
import android.widget.EditText;

public class HelloAndroid extends Activity implements View.OnClickListener {
    private EditText messageText;
    private static final int DAD_ID = Menu.FIRST;
    private static final int MOM_ID = Menu.FIRST + 1;

    public void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.main);
        messageText = (EditText) this.findViewById(R.id.message);
        messageText.setText("From onCreate");
        Button ok = (Button) findViewById(R.id.ok);
        ok.setOnClickListener(this);
    }
}
```

Menus Cont.

```
public void onClick(View v) {
    messageText.setText(messageText.getText() + " click");
}

public boolean onCreateOptionsMenu(Menu menu) {
    super.onCreateOptionsMenu(menu);

    menu.add(0, DAD_ID, 0, R.string.menu_dad).setShortcut('0', 'd');
    menu.add(0, MOM_ID, 0, R.string.menu_mom);
    return true;
}

public boolean onOptionsItemSelected(MenuItem item) {
    switch (item.getItemId()) {
    case DAD_ID:
        messageText.setText(messageText.getText() + " Dad");
        return true;
    case MOM_ID:
        messageText.setText(messageText.getText() + " Mom");
        return true;
    }
    return super.onOptionsItemSelected(item);
}
}
```


Menus Cont.

```
public boolean onPrepareOptionsMenu(Menu menu) {  
    super.onPrepareOptionsMenu(menu);  
}  
}
```

res/values/strings.xml

```
<?xml version="1.0" encoding="utf-8"?>
<resources>
  <string name="hello">From String resource</string>
  <string name="app_name">Lecture Examples</string>
  <string name="button_ok">OK</string>

  <string name="menu_dad">Hi Dad</string>
  <string name="menu_mom">Hi Mom</string>

</resources>
```

Other GUI Widgets

Details are similar

Check documentation

Intents

Intents - Calling Activities

Android application consists of multiple activities

Activity represents one screen or view

Going from one screen to another

Requires calling activity

Can't call new activity directly

Use intent to indicate activity to start

Intent Resolution - Explicit Intents

Specify the component (class) an intent is to run

Common way to call your own code

```
Intent go = new Intent();  
go.setClassName("edu.sdsu.cs696", "edu.sdsu.cs696.Hello");  
startActivity(go);
```

Intent Resolution - Implicit Intents

Provide information about activity you want to run

display web page

System determines which component to run

If more than one activity can handle request

User is asked to select

Each activity declares in manifest what it can handle

```
<intent-filter>  
  <action android:name="android.intent.action.MAIN" />  
  <category android:name="android.intent.category.LAUNCHER" />  
</intent-filter>
```

Intent

Abstract description of an operation to be performed

Methods in android.content.Context	Action
startActivity(Intent) startActivityForResult(Intent, int)	Launch an Activity
sendBroadcast(Intent)	send it to any interested BroadcastReceiver components
startService(Intent) bindService(Intent, ServiceConnection, int)	

Intent Data

Primary Attributes

action

Action to be taken

data

Data to operation on as Uri

Secondary Attributes

category

additional information about the action to execute

type

Mime type of intent data

component

Explicit class to run

extras

Data for other component

<http://code.google.com/android/reference/android/content/Intent.html>

Implicit Intents

action

If given, must be listed by the component as one it handles.

String, which we can create

type

Retrieved from the Intent's data, if not already supplied in the Intent.

If given, must be listed by the component as one it handles

data that is not a content: URI and where no explicit type,

The scheme of the intent data (such as http: or mailto:) is considered

If given, must be listed by the component as one it handles

Categories

If given, all must be listed by the component as ones it handles

Intents Handled By Google Android Apps

Scheme	Action	Description
http://web_address https://web_address	VIEW	Open a browser window to the URL specified.
"" (empty string) http://web_address https://web_address	WEB_SEARCH	Opens the file at the location on the device in the browser.
tel: phone_number	CALL	Calls the entered phone number.
tel:phone_number voicemail:	DIAL	Dials but does not actually initiate the call the number given
geo:latitude,longitude geo:latitude,longitude?z=zoom geo:0,0?q=my+street+address geo:0,0?q=business+near+city	VIEW	Opens the Maps application to the given location

Intent Examples

First Intent Example - Dial Phone

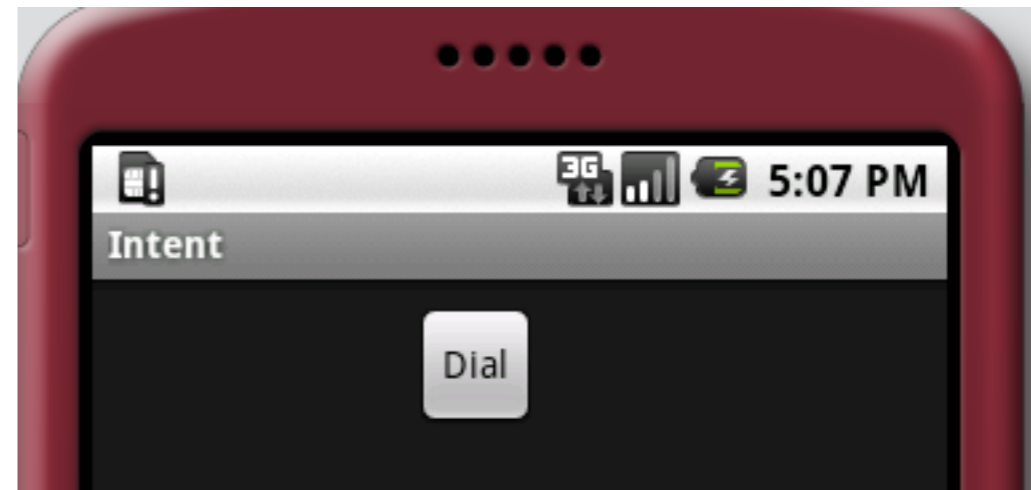
Activity with button

When button is pressed

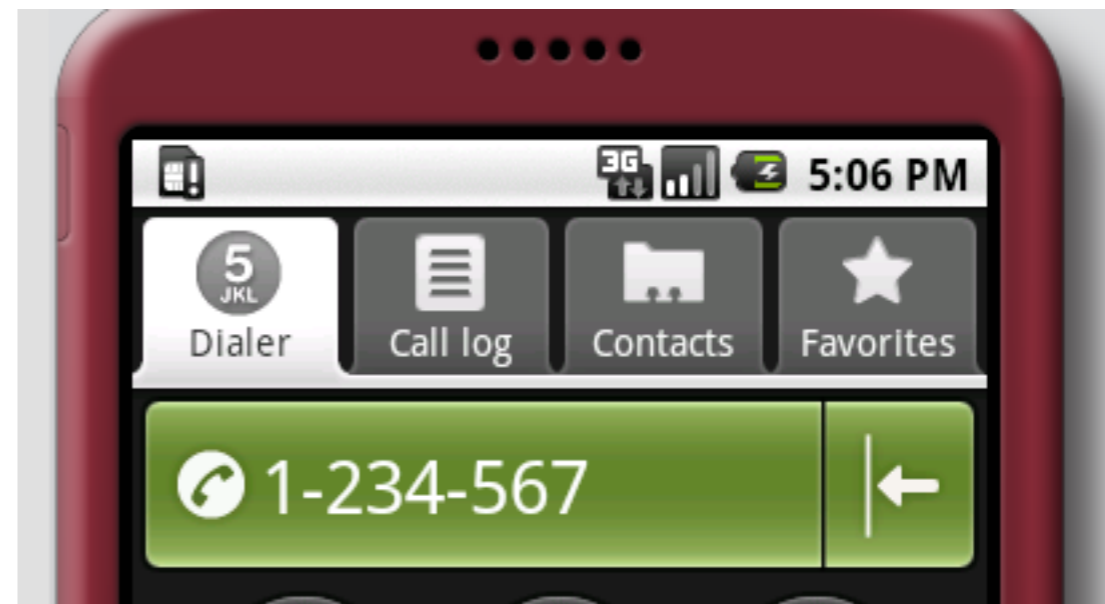
- Phone activity is run

- Phone number is entered

- Phone number is hard coded



Implicit Intent to another application



IntentExample.java

```
public class IntentExample extends Activity implements View.OnClickListener {  
  
    public void onCreate(Bundle savedInstanceState) {  
        super.onCreate(savedInstanceState);  
        setContentView(R.layout.intent);  
        Button ok = (Button) findViewById(R.id.go);  
        ok.setOnClickListener(this);  
    }  
  
    public void onClick(View v) {  
        Intent dial = new Intent();  
        dial.setAction(android.content.Intent.ACTION_DIAL);  
        dial.setData(Uri.parse("tel:1234567"));  
        startActivity(dial);  
    }  
}
```

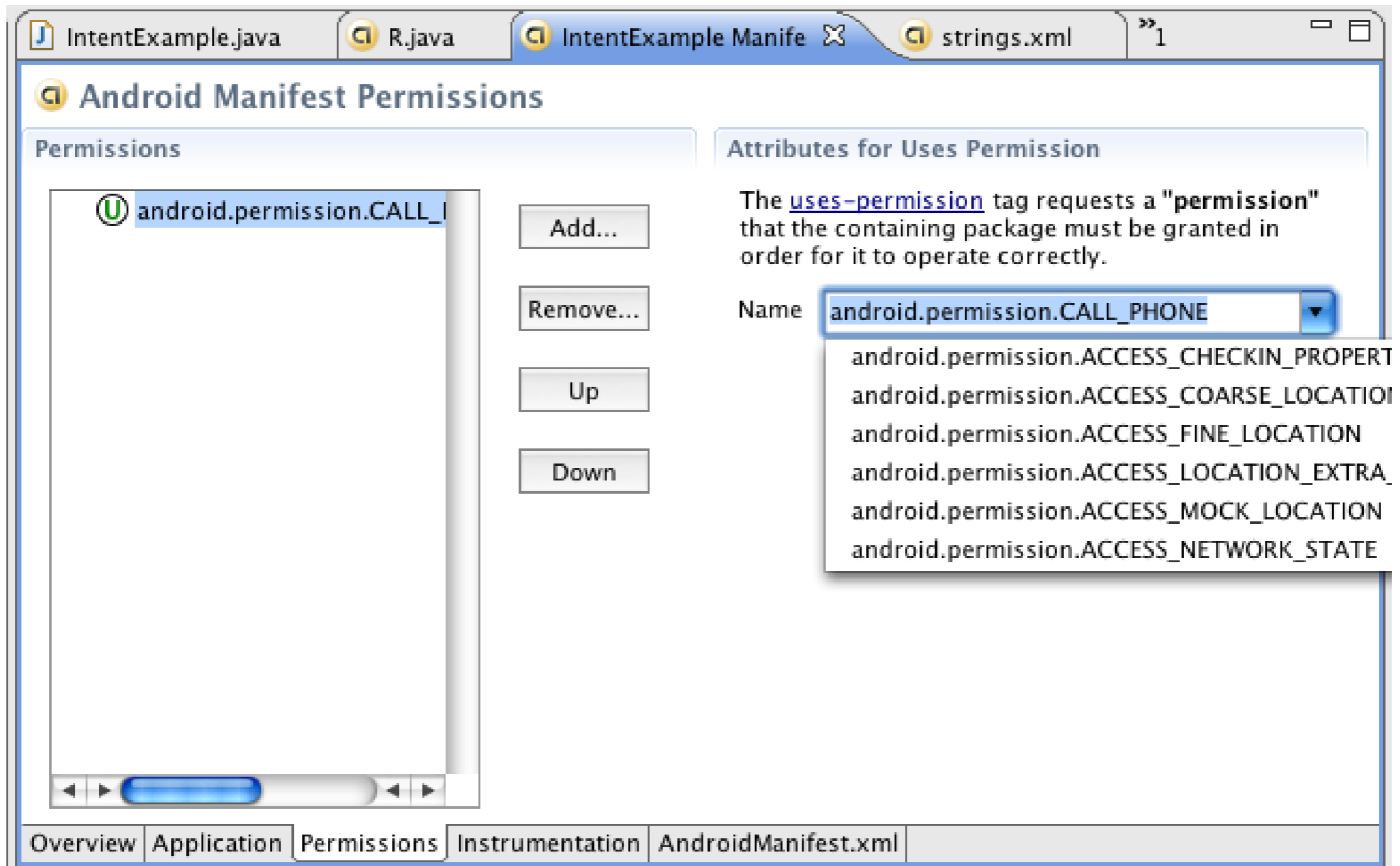
Other Ways to create the Intent

```
public void onClick(View v) {  
    Intent dial = new Intent(Intent.ACTION_DIAL, Uri.parse("tel:1234567"));  
    startActivity(dial);  
}
```

IntentExamples Manifest

```
<?xml version="1.0" encoding="utf-8"?>
<manifest xmlns:android="http://schemas.android.com/apk/res/android"
    package="edu.sdsu.cs696"
    android:versionCode="1"
    android:versionName="1.0">
    <application android:icon="@drawable/icon" android:label="@string/app_name">
        <activity android:name=".IntentExample"
            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="2" />
    <uses-permission android:name="android.permission.CALL_PHONE"></uses-
permission>
</manifest>
```


Adding the Permission



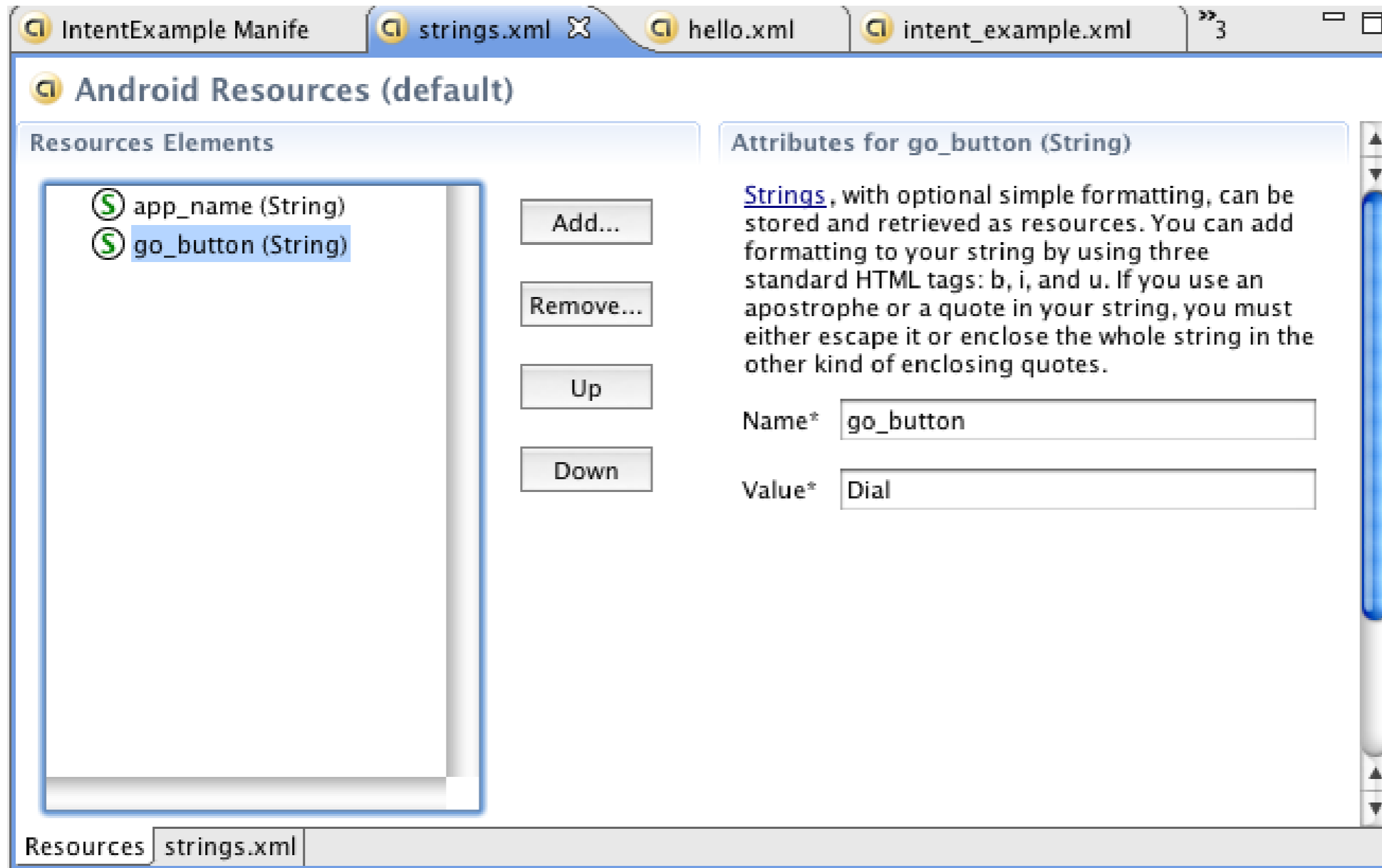
intent.xml

```
<?xml version="1.0" encoding="utf-8"?>
<AbsoluteLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:orientation="vertical"
    android:id="@+id/layout"
    android:layout_width="fill_parent"
    android:layout_height="fill_parent"
    >
<Button
    android:id="@+id/go"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="@string/go_button"
    android:gravity="center"
    android:layout_x="120px"
    android:layout_y="10px"
    >
</Button>
</AbsoluteLayout>
```

strings.xml

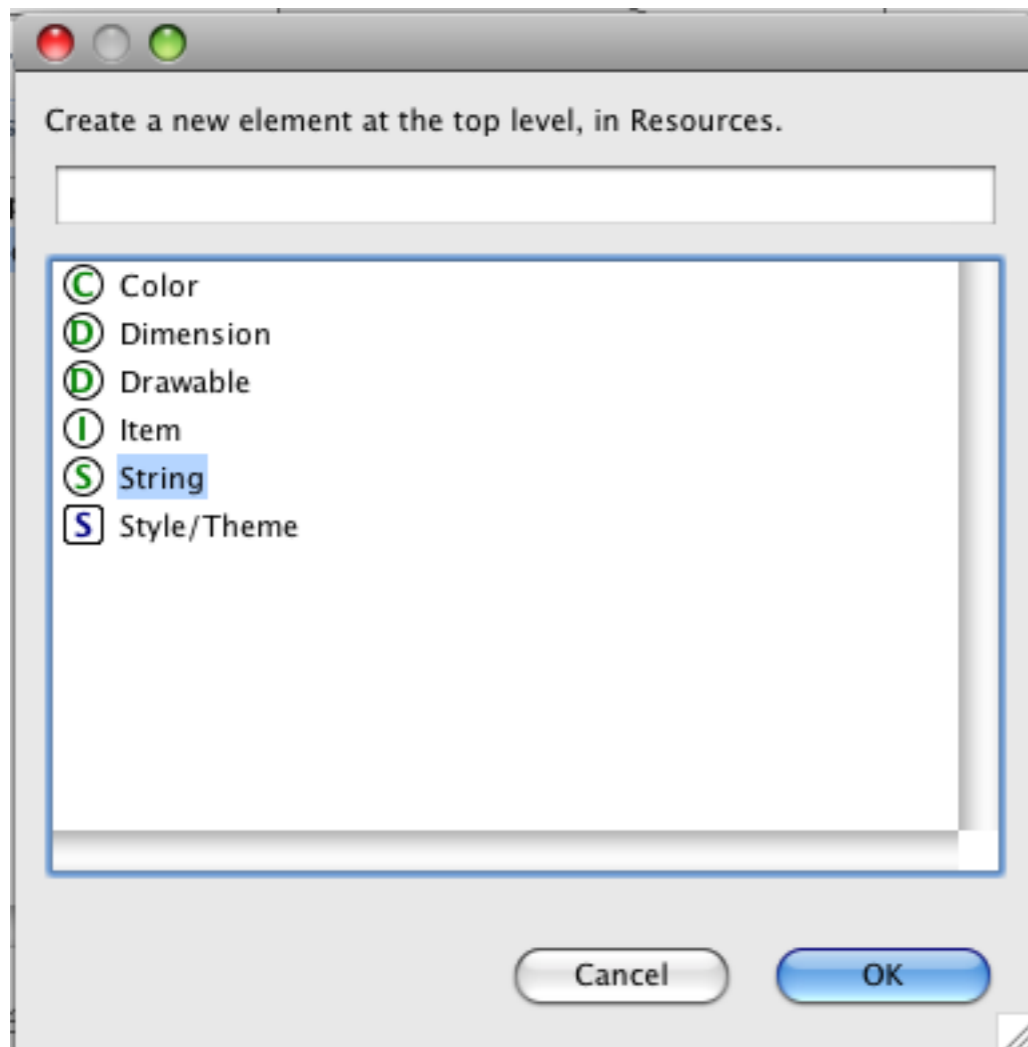
```
<?xml version="1.0" encoding="utf-8"?>  
<resources>  
  <string name="hello">Hello World, IntentExample!</string>  
  <string name="app_name">Intent</string>  
  <string name="go_button">Dial</string>  
</resources>
```

Editing Resources



Add a new String

Click on the "Add" button in the Resource view of Strings.xml
(See previous slide)



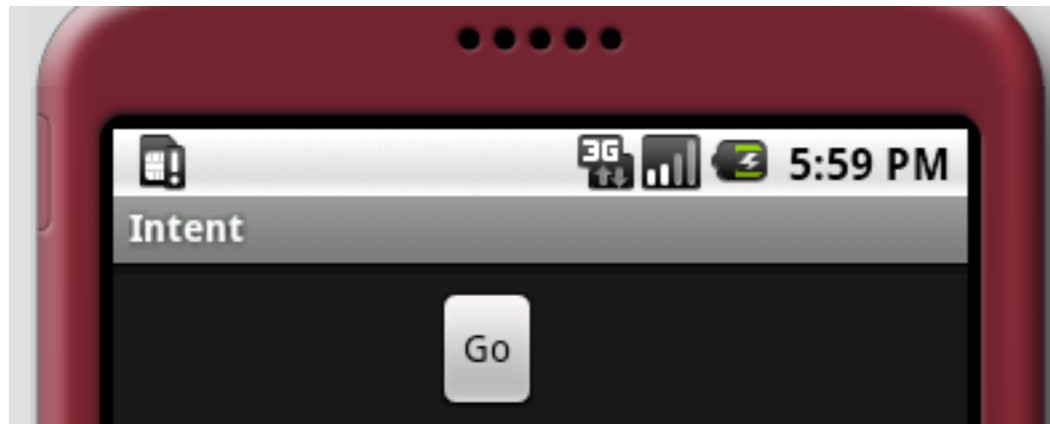
Class Name Intent Example

Button click will call another activity in same application

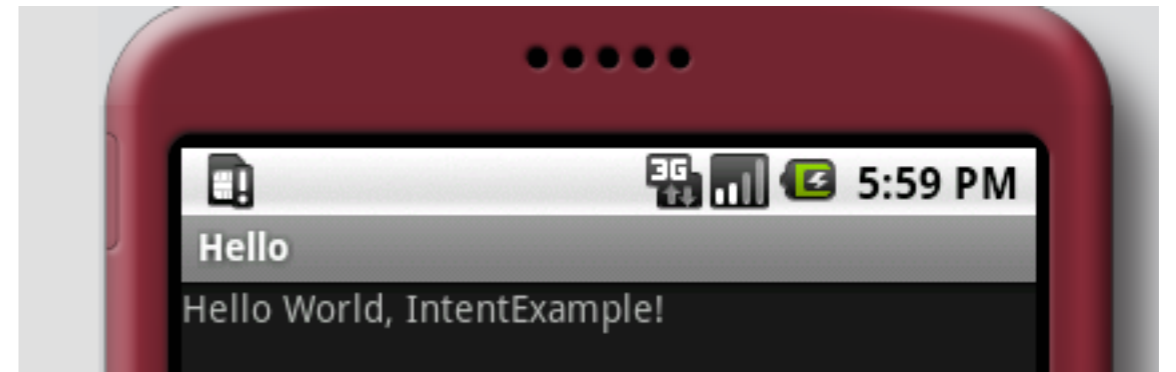
Will call the class directly

Useful when activity is private to application

IntentExample.java



Hello.java



Clicking the "Go" button will launch the Hello activity

IntentExample.java

```
package edu.sdsu.cs696;
```

```
public class IntentExample extends Activity implements View.OnClickListener {
```

```
    public void onCreate(Bundle savedInstanceState) {  
        super.onCreate(savedInstanceState);  
        setContentView(R.layout.intent);  
        Button ok = (Button) findViewById(R.id.go);  
        ok.setOnClickListener(this);  
    }
```

```
    public void onClick(View v) {  
        Intent go = new Intent();  
        go.setClassName("edu.sdsu.cs696", "edu.sdsu.cs696.Hello");  
        startActivity(go);  
    }
```

```
}
```

```
//Full class name required
```

Hello.java

```
package edu.sdsu.cs696;

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

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

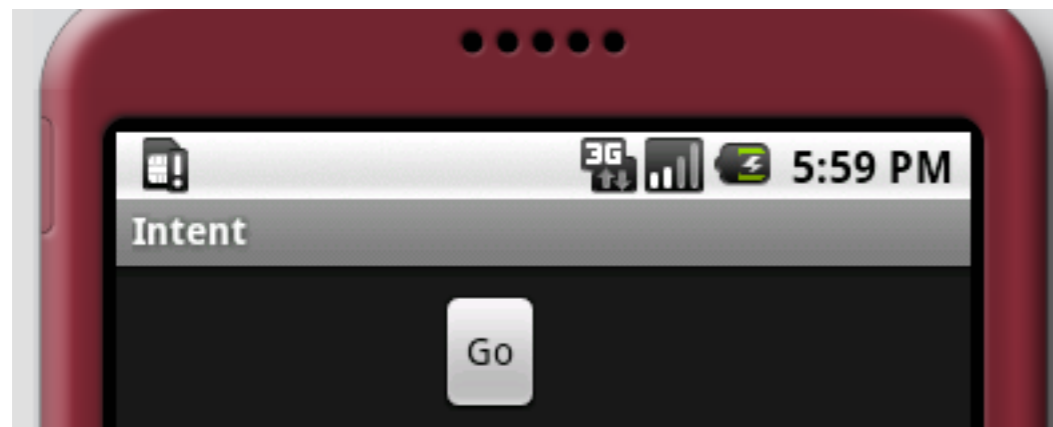

AndroidManifest.xml

```
<?xml version="1.0" encoding="utf-8"?>
<manifest xmlns:android="http://schemas.android.com/apk/res/android"
    package="edu.sdsu.cs696"
    android:versionCode="1"
    android:versionName="1.0">
    <application android:icon="@drawable/icon" android:label="@string/app_name">
        <activity android:name=".IntentExample"
            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:label="Hello" android:name="Hello">
        </activity>
    </application>
</manifest>
```

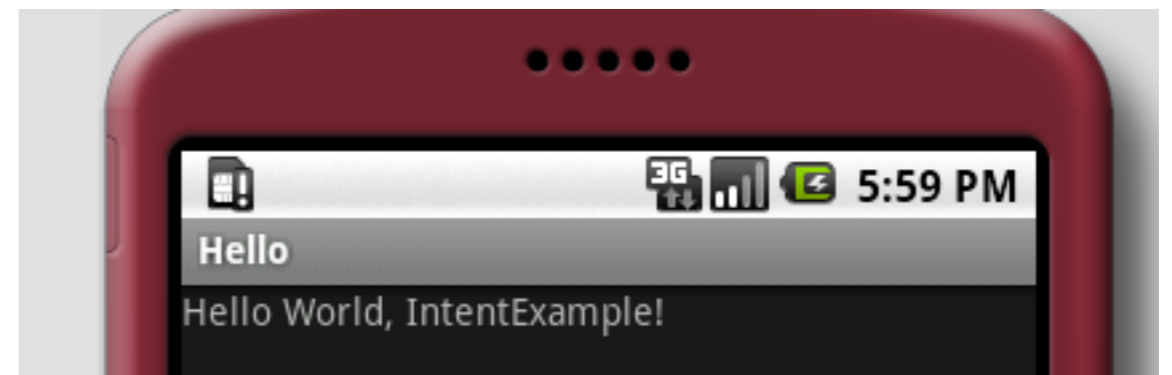
Implicit Example - Intent Filter

Button click will call another component using intent filters
Indirect access of activity

IntentExample.java



Hello.java



Clicking the "Go" button will launch the Hello activity

IntentExample

```
public class IntentExample extends Activity implements View.OnClickListener {  
  
    public void onCreate(Bundle savedInstanceState) {  
        super.onCreate(savedInstanceState);  
        setContentView(R.layout.intent_example);  
        Button ok = (Button) findViewById(R.id.go);  
        ok.setOnClickListener(this);  
    }  
  
    public void onClick(View v) {  
        Intent go = new Intent();  
        go.setAction("cs696.sender.add");  
        startActivity(go);  
    }  
}
```

Hello.java

```
package edu.sdsu.cs696;

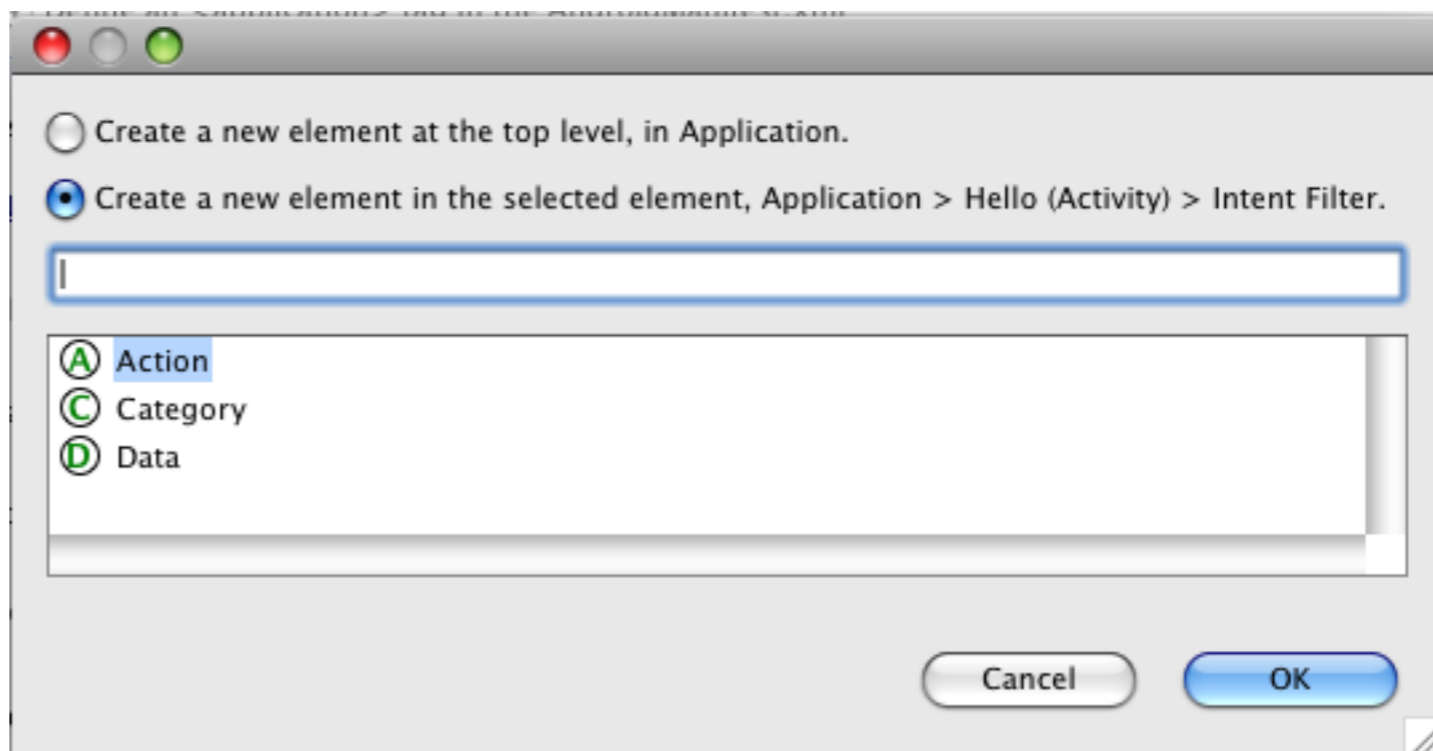
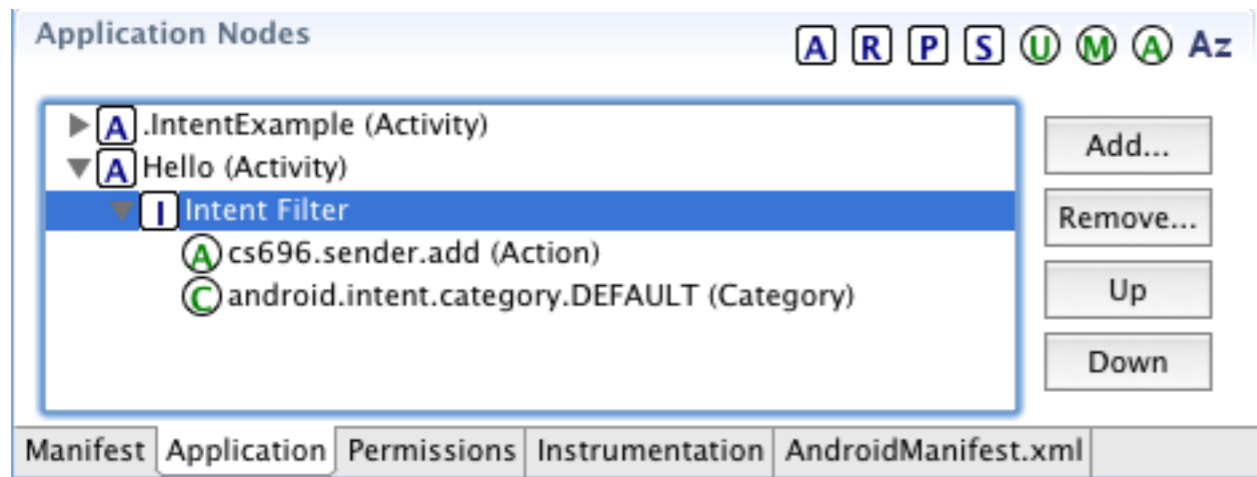
import android.app.Activity;
import android.os.Bundle;

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

AndroidManifest.xml

```
<?xml version="1.0" encoding="utf-8"?>
<manifest xmlns:android="http://schemas.android.com/apk/res/android"
    package="edu.sdsu.cs696"
    android:versionCode="1"
    android:versionName="1.0">
    <application android:icon="@drawable/icon" android:label="@string/app_name">
        <activity android:name=".IntentExample"
            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:label="Hello" android:name="Hello">
            <intent-filter>
                <action android:name="cs696.sender.add"></action>
                <category android:name="android.intent.category.DEFAULT">
                </category>
            </intent-filter>
        </activity>
    </application>
</manifest>
```

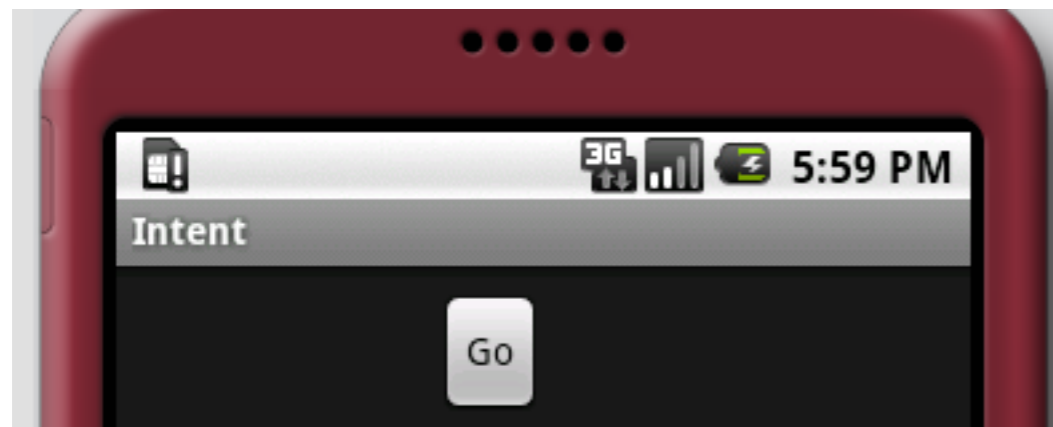
Adding Categories etc



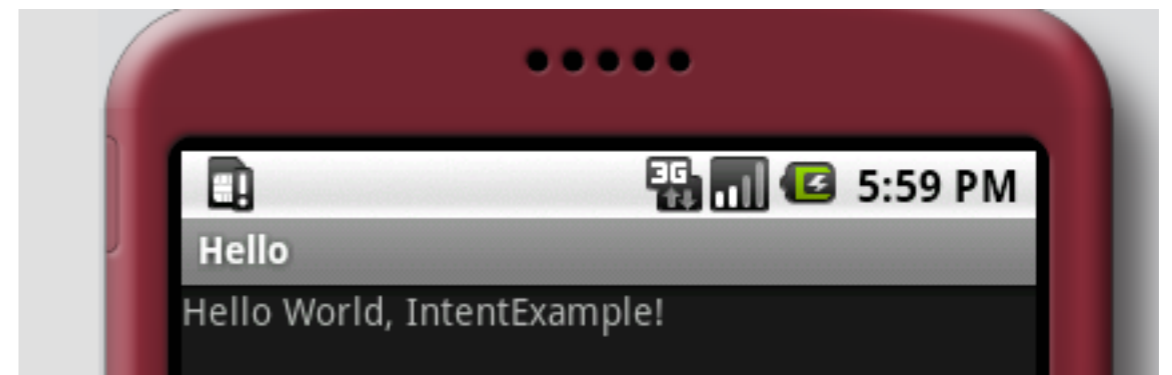
Implicit Example - Filter with Categories

Button click will call another component

IntentExample.java



Hello.java



Clicking the "Go" button will launch the Hello activity

Category Example

```
public class IntentExample extends Activity implements View.OnClickListener {

    public void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.intent);
        Button ok = (Button) findViewById(R.id.go);
        ok.setOnClickListener(this);
    }

    public void onClick(View v) {
        Intent go = new Intent();
        go.setAction("cs696.sender.add");
        go.addCategory("foo");
        startActivity(go);
    }
}
```


AndroidManifest.xml

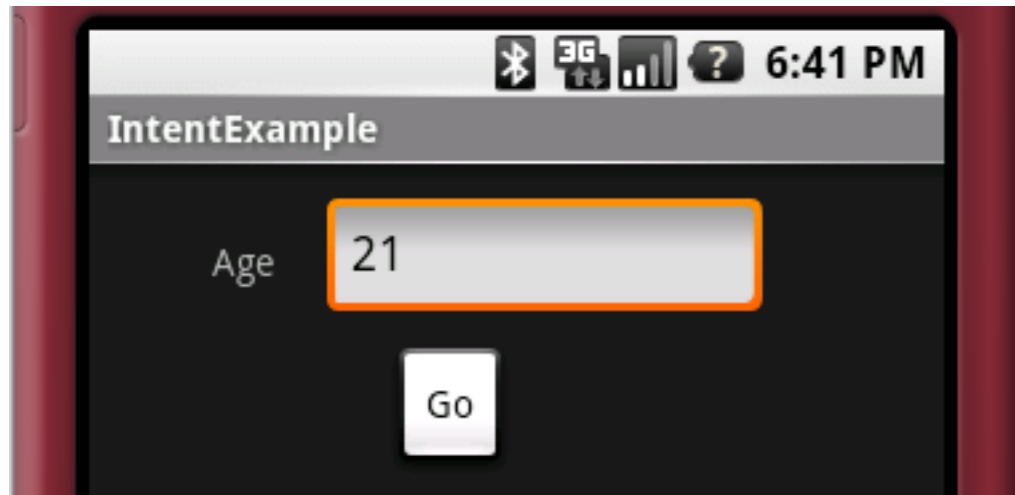
```
<?xml version="1.0" encoding="utf-8"?>
<manifest xmlns:android="http://schemas.android.com/apk/res/android"
    package="edu.sdsu.cs696"
    android:versionCode="1"
    android:versionName="1.0">
    <application android:icon="@drawable/icon" android:label="@string/app_name">
        <activity android:name=".IntentExample"
            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:label="Hello" android:name="Hello">
            <intent-filter>
                <action android:name="cs696.sender.add"></action>
                <category android:name="android.intent.category.DEFAULT">
                </category>
                <category android:name="foo"></category>
            </intent-filter>
        </activity>
    </application>
</manifest>
```

Some Application Settings

<u>Name*</u>	<input type="text" value=".IntentExample"/>	<input type="button" value="Browse..."/>
Theme	<input type="text"/>	<input type="button" value="Browse..."/>
Label	<input type="text" value="@string/app_name"/>	<input type="button" value="Browse..."/>
Icon	<input type="text"/>	<input type="button" value="Browse..."/>
Launch mode	<input type="text"/>	<input type="button" value="▼"/>
Screen orientation	<input type="text"/>	<input type="button" value="▼"/>
Config changes	<input type="text"/>	<input type="button" value="Select..."/>
Permission	<input type="text"/>	<input type="button" value="▼"/>
Multiprocess	<input type="text"/>	<input type="button" value="▼"/>
Process	<input type="text"/>	<input type="button" value="Browse..."/>
Task affinity	<input type="text"/>	<input type="button" value="Browse..."/>
Allow task reparenting	<input type="text"/>	<input type="button" value="▼"/>
Finish on task launch	<input type="text"/>	<input type="button" value="▼"/>
Clear task on launch	<input type="text"/>	<input type="button" value="▼"/>
Always retain task state	<input type="text"/>	<input type="button" value="▼"/>
State not needed	<input type="text"/>	<input type="button" value="▼"/>
Exclude from recents	<input type="text"/>	<input type="button" value="▼"/>
Enabled	<input type="text"/>	<input type="button" value="▼"/>

Intent - Passing Data

IntentExample



Displays/Edits age

Go button

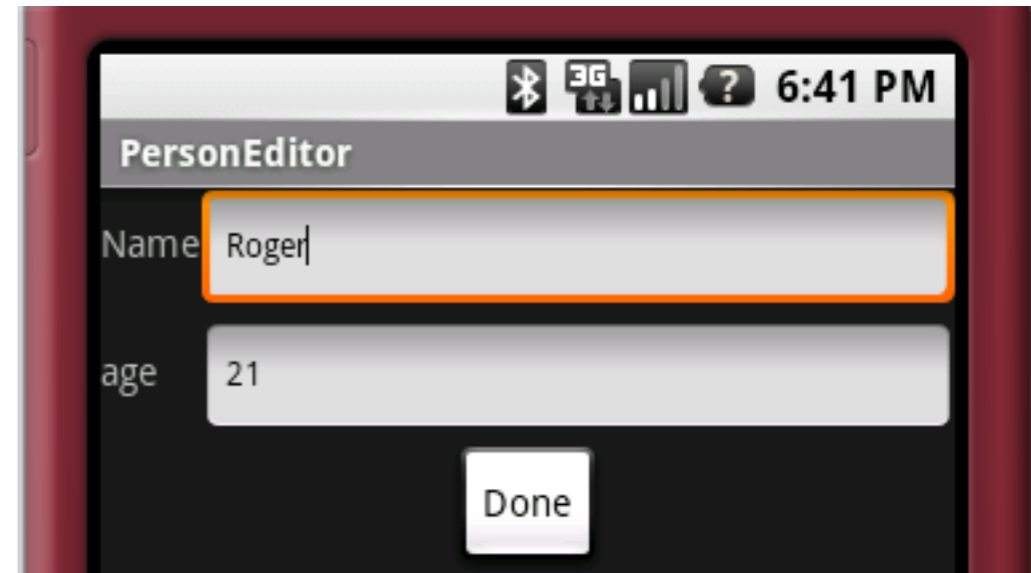
Calls PersonEditor

Passes data

Name

Age

PersonEditor



Displays/Edits Name and age

Done button

Returns edited data back

Age = 0 cancels edit

IntentExample.java

```
public class IntentExample extends Activity implements View.OnClickListener {  
    private EditText numberText;  
    private static final int INTENT_EXAMPLE_REQUEST = 123;  
  
    public void onCreate(Bundle savedInstanceState) {  
        super.onCreate(savedInstanceState);  
        setContentView(R.layout.intent);  
        Button ok = (Button) findViewById(R.id.go);  
        ok.setOnClickListener(this);  
        numberText = (EditText) this.findViewById(R.id.number);  
        numberText.setText("21");  
    }  
}
```

IntentExample.Java continued

Sending the data to PersonEditor

```
public void onClick(View v) {  
    Intent go;  
    go = new Intent();  
    go.setAction("android.intent.action.EDIT");  
    go.addCategory("person_editor");  
    String newAge = numberText.getText().toString();  
    go.putExtra("age", newAge);  
    go.putExtra("name", "Roger");  
    startActivityForResult(go, INTENT_EXAMPLE_REQUEST);  
}
```

IntentExample.Java continued

Getting the Results back

```
protected void onActivityResult(int requestCode, int resultCode, Intent data) {  
    if (requestCode != INTENT_EXAMPLE_REQUEST) {  
        numberText.setText("Not from me");  
        return;  
    }  
    switch (resultCode) {  
    case RESULT_OK:  
        String editedAge = data.getStringExtra("age");  
        numberText.setText(editedAge);  
        break;  
    case RESULT_CANCELED:  
        numberText.setText("Cancelled");  
        break;  
    }  
    }  
}
```

PersonEditor.java

```
public class PersonEditor extends Activity implements View.OnClickListener {
    private EditText ageText;
    private EditText nameText;

    public void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.person_editor);
        Button done = (Button) findViewById(R.id.edit_done);
        done.setOnClickListener(this);
        ageText = (EditText) this.findViewById(R.id.edit_age);
        nameText = (EditText) this.findViewById(R.id.edit_name);
        Bundle personData = getIntent().getExtras();
        String age = personData.getString("age");
        String name = personData.getString("name");
        if ((age != null) && (name != null)) {
            ageText.setText(age);
            nameText.setText(name);
        }
    }
}
```

PersonEditor.java

Returning the data

```
public void onClick(View v) {  
    String newAge = ageText.getText().toString();  
    Intent result = getIntent();  
    result.putExtra("age", newAge);  
    if (newAge.equals("0"))  
        setResult(RESULT_CANCELED, result);  
    else  
        setResult(RESULT_OK, result);  
    finish();  
}
```


AndroidManifest.xml

```
<?xml version="1.0" encoding="utf-8"?>
<manifest xmlns:android="http://schemas.android.com/apk/res/android"
    package="edu.sdsu.cs683.example" android:versionCode="1"
    android:versionName="1.0.0">
    <application android:icon="@drawable/icon" android:label="@string/app_name">
        <activity android:name=".IntentExample" 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:label="PersonEditor" android:name="PersonEditor">
            <intent-filter>
                <action android:name="android.intent.action.EDIT"></action>
                <category android:name="person_editor"></category>
                <category android:name="android.intent.category.DEFAULT">
                    </category>
            </intent-filter>
        </activity>
    </application>
</manifest>
```