Android Intents

Part 2
Inter-Process Communication Using Bundles

Victor Matos
Cleveland State University

Notes are based on:

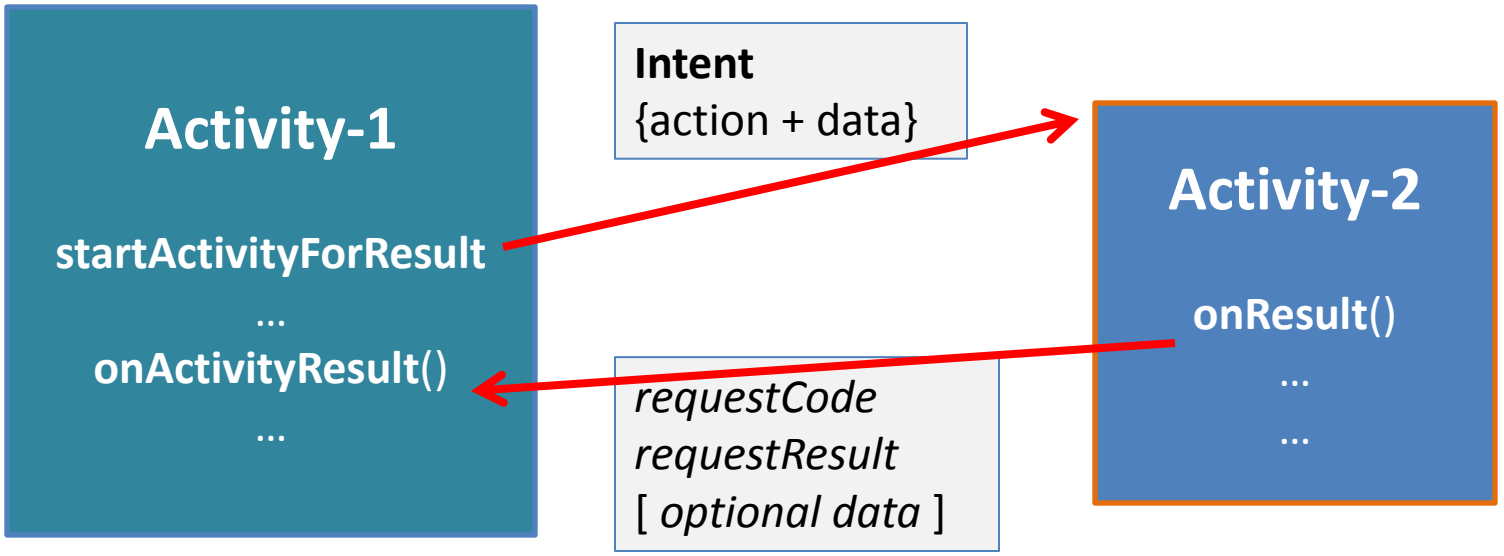
Android Developers
http://developer.android.com/index.html

Intents

Android Intents

An *activity* usually presents a single visual user interface from which a number of actions could be performed.

Moving from one activity to another is accomplished by having the current activity start the next one through so called ***intents***.



Intents

Android Bundles

Most programming languages support the notion of **IPC** *method-calling* with arguments flowing birectionally from the caller to the invoked method.

In android the calling activity issues an invocation to another activity using an **Intent** object.

Notably in Android, *the caller does not stop waiting* for the called activity to return results. Instead a listening-method [*onActivityResult(...)*] should be used.

Intents

Android Bundles

Normally the IPC expressions *actual parameter list*, and *formal parameter list* are used to designated the signature of particpating arguments, and the currently supplied data.

Instead of using the traditional *formal / actual parameter lists,* Android relies on the concept of Intents to establish Inter-process-communication.

Intents optionally carry a named actual list or **bundle** for data exchange.

Intents

Android Bundles

The Android **Bundle** container is a simple mechanism used to pass data between activities.

A **Bundle** is a type-safe collection of **<name, value>** pairs.

There is a set of ***putXXX*** and ***getXXX*** methods to store and retrieve (single and array) values of primitive data types from/to the bundles. For example

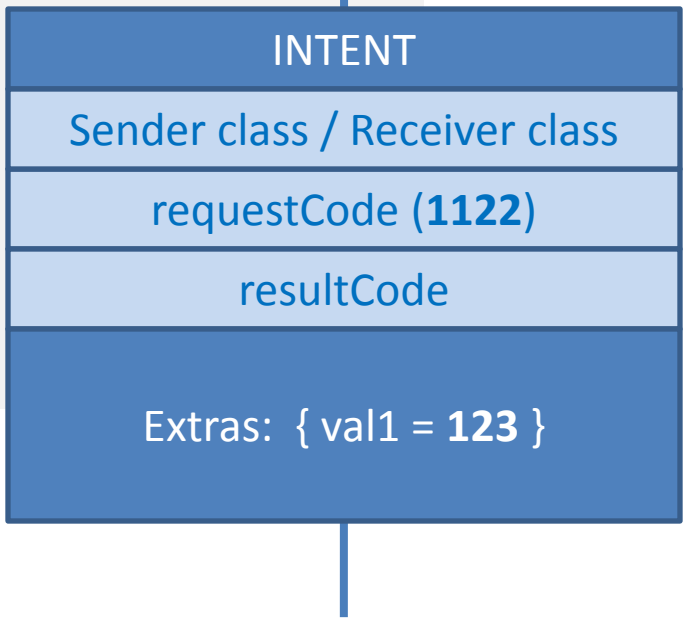
```
Bundle myBundle = new Bundle();
myBundle.putDouble ("var1", 3.1415);
...
Double v1 = myBundle.getDouble("var1");
```

Intents

Android Intents & Bundles

Activity1: Sender → **Activity2: Receiver**

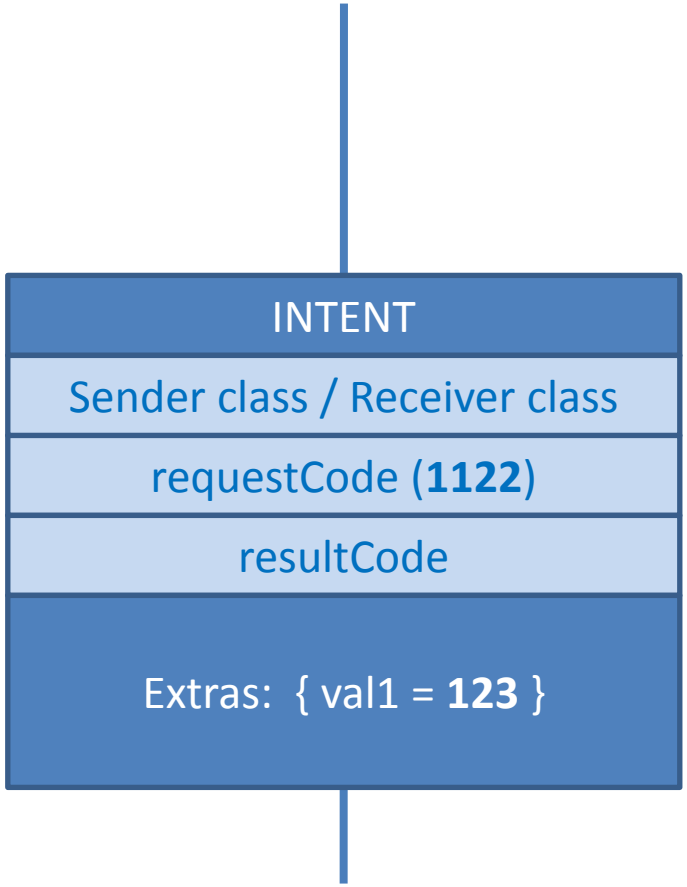
```
Intent myIntentA1A2 = new Intent  (Activity1.this, Activity2.class);

Bundle myBundle1 = new Bundle();
myBundle1.putInt (“val1”, 123);

myIntentA1A2.putExtras(myBundle1);

startActivityForResult(myIntentA1A2, 1122);
```



Intents

Android Intents & Bundles

Activity1: Sender



Activity2: Receiver

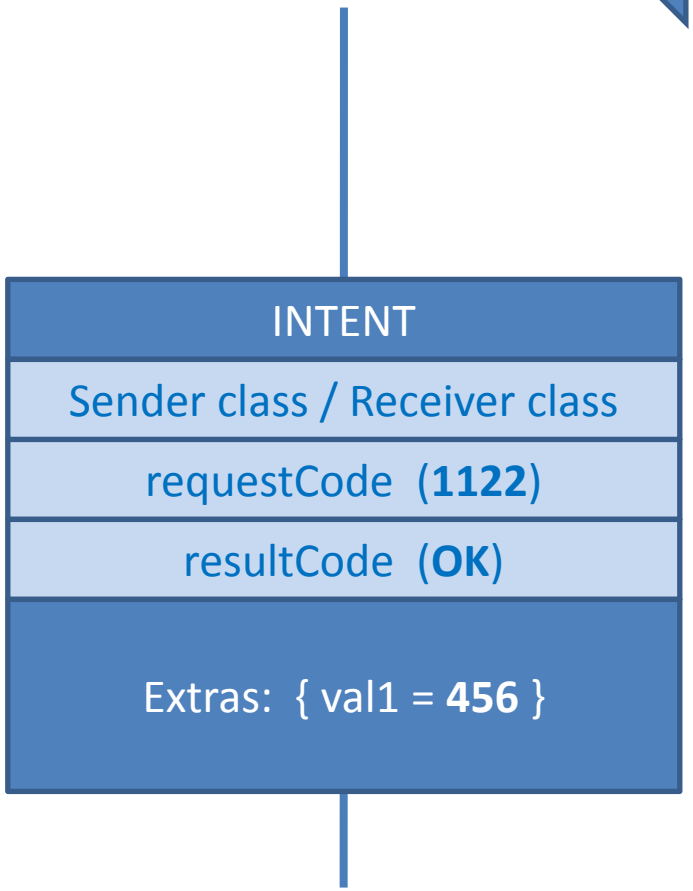
```
Intent myLocalIntent2 = getIntent();

Bundle myBundle = myLocalIntent.getExtras();

int val1 = myBundle.getInt("val1");
```

Intents

Android Intents & Bundles

Activity1: Sender ← **Activity2: Receiver**

```
myBundle.putString("val1", 456 );

myLocalIntent.putExtras(myBundle);

setResult(Activity.RESULT_OK, myLocalIntent);
```



Intents

Android Bundles

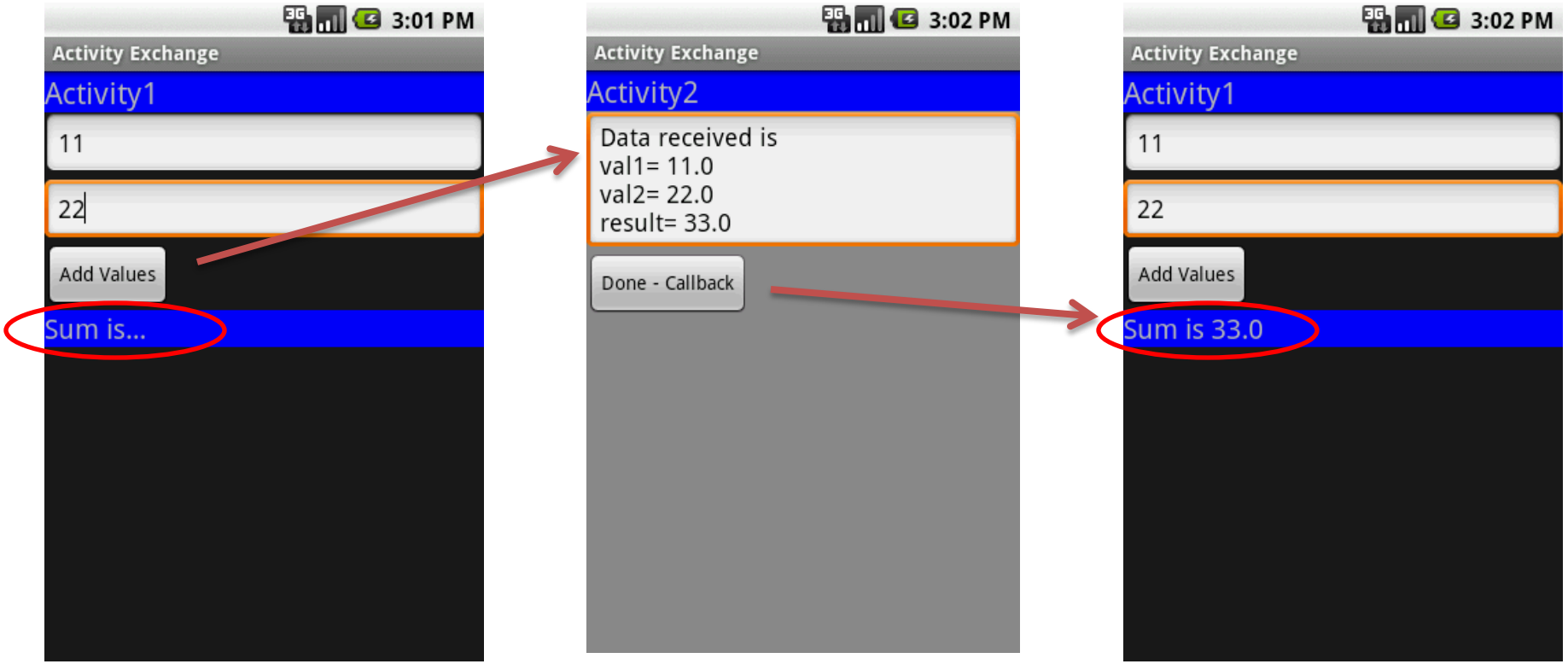
Available at: http://developer.android.com/reference/android/os/Bundle.html

Example of Public Methods	
void	clear() Removes all elements from the mapping of this Bundle.
Object	clone() Clones the current Bundle.
boolean	containsKey(String key) Returns true if the given key is contained in the mapping of this Bundle.
void	putIntArray(String key, int[] value) Inserts an int array value into the mapping of this Bundle, replacing any existing value for the given key.
void	putString(String key, String value) Inserts a String value into the mapping of this Bundle, replacing any existing value for the given key.
void	putStringArray(String key, String[] value) Inserts a String array value into the mapping of this Bundle, replacing any existing value for the given key.
void	putStringArrayList(String key, ArrayList<String> value) Inserts an ArrayList value into the mapping of this Bundle, replacing any existing value for the given key.
void	remove(String key) Removes any entry with the given key from the mapping of this Bundle.
int	size() Returns the number of mappings contained in this Bundle.

Intents

Tutorial. Activity Excahange

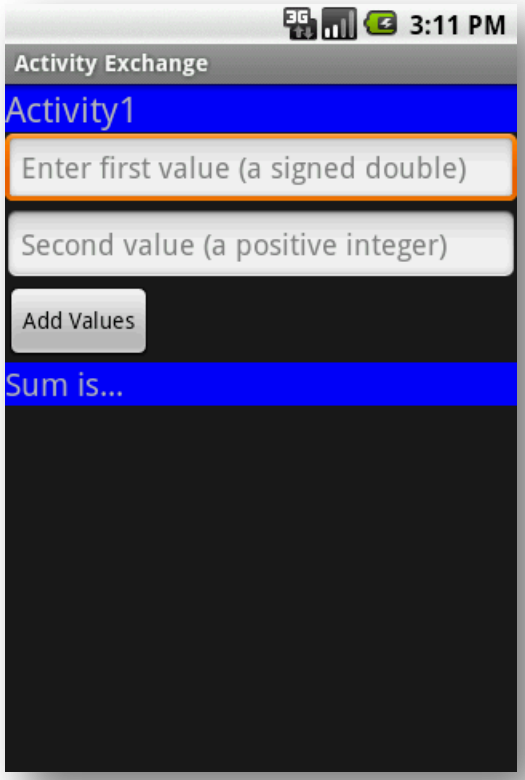
Activity1 collects two values from its UI and calls Activity2 to compute the sum of them. The result is sent back from Activity 2 to Activity1.



Intents

Tutorial. Activity Excahange

Step1. Create GUI for Activity1(main1.xml)



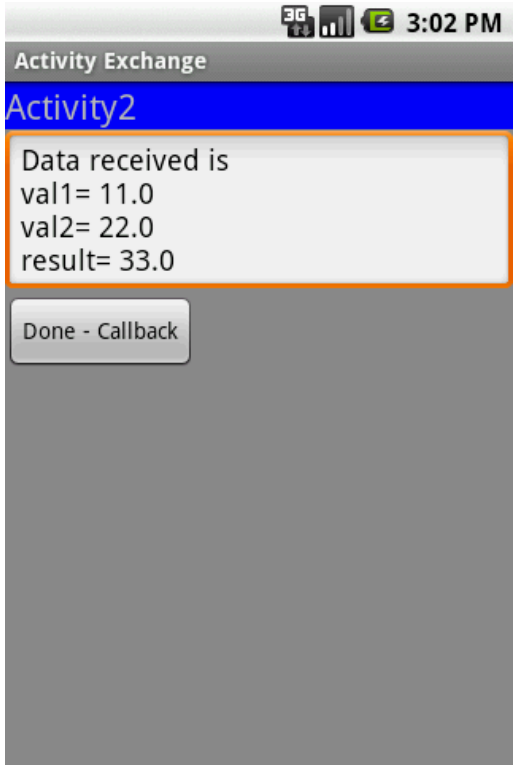
Note. The element **android:inputStyle** indicates the first value could be numeric, with optional decimals and sign.

```
<?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:text="Activity1"
        android:textSize="22sp"
        android:background="#ff0000ff"
        android:layout_width="fill_parent"
        android:layout_height="wrap_content" />
<EditText
        android:hint="Enter first value (a signed double)"
        android:id="@+id/EditText01"
        android:layout_width="fill_parent"
        android:layout_height="wrap_content"
        android:inputType="numberDecimal|numberSigned|number" />
<EditText
        android:hint="Second value (a positive integer)"
        android:id="@+id/EditText02"
        android:layout_width="fill_parent"
        android:layout_height="wrap_content"
        android:inputType="number" />
<Button
        android:text="Add Values"
        android:id="@+id/btnAdd"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content" />
<TextView
        android:background="#ff0000ff"
        android:text="Sum is..."
        android:textSize="28sp"
        android:id="@+id/TextView01"
        android:layout_width="fill_parent"
        android:layout_height="wrap_content" />
</LinearLayout>
```

Intents

Tutorial. Activity Excahange

Step2. Create GUI for Activity2(main2.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:background="#ff888888">

<TextView
        android:text="Activity2"
        android:textSize="22sp"
        android:background="#ff0000ff"
        android:layout_width="fill_parent"
        android:layout_height="wrap_content" />

<EditText
        android:text="Data reveived..."
        android:id="@+id/etDataReceived"
        android:layout_width="fill_parent"
        android:layout_height="wrap_content" />

<Button
        android:text="Done - Callback"
        android:id="@+id/btnDone"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content" />

</LinearLayout>
```

Intents

Tutorial. Activity Excahange

Step3. Activity1. After clicking the button data, from UI is put in a bundle and sent to Activity2. A listener remains alert waiting for results to come from the called activity.

```
package cis493.matos.intents6;
// Activity1
// get input data from user, call Activity2, show result
import android.app.Activity;
import android.content.Intent;
import android.os.Bundle;
import android.view.View;
import android.view.View.OnClickListener;
import android.widget.Button;
import android.widget.EditText;
import android.widget.TextView;

public class Activity1 extends Activity {
    EditText txtVal1;
    EditText txtVal2;
    TextView lblResult;
    Button   btnAdd;

    @Override
    public void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.main1);
        txtVal1 = (EditText)findViewById(R.id.EditText01);
        txtVal2 = (EditText)findViewById(R.id.EditText02);
        lblResult = (TextView) findViewById(R.id.TextView01);

        btnAdd = (Button) findViewById(R.id.btnAdd);
        btnAdd.setOnClickListener(new OnClickListener() {

@Override
public void onClick(View v) {
 // get values from the UI
 Double v1 = Double.parseDouble(txtVal1.getText().toString());
 Double v2 = Double.parseDouble(txtVal2.getText().toString());

 // create intent to call Activity2
 Intent myIntentA1A2 = new Intent (Activity1.this,
                                   Activity2.class);
 // create a container to ship data
 Bundle myData = new Bundle();

 // add <key,value> data items to the container
 myData.putDouble("val1", v1);
 myData.putDouble("val2", v2);

 // attach the container to the intent
 myIntentA1A2.putExtras(myData);

 // call Activity2, tell your local listener to wait response
 startActivityForResult(myIntentA1A2, 101);

 }//onClick
 });
}//onCreate

////////////////////////////////////////////////////////////////////
/ local listener receiving callbacks from other activities
@Override
protected void onActivityResult(int requestCode,
                                int resultCode, Intent data) {
super.onActivityResult(requestCode, resultCode, data);

try{
   if ((requestCode == 101 ) && (resultCode == Activity.RESULT_OK)){
   Bundle myResults = data.getExtras();
   Double vresult = myResults.getDouble("vresult");
   lblResult.setText("Sum is " + vresult);
   }
}
catch (Exception e) {
   lblResult.setText("Oops! - " + requestCode + " " + resultCode);
}
}//onActivityResult

}//Activity1
```

Intents

Tutorial. Activity Excahange

Step4. Activity2. Called from Activity1. Extracts input data from the bundle attached to the intent. Performs local computation. Adds result to bundle. Returns OK signal.

```
package cis493.matos.intents6;

import android.app.Activity;
import android.content.Intent;
import android.os.Bundle;
import android.view.View;
import android.view.View.OnClickListener;
import android.widget.Button;
import android.widget.EditText;

public class Activity2 extends Activity
                       implements OnClickListener{
EditText dataReceived;
Button  btnDone;

@Override
protected void onsuper.onCreate(savedInstanceState);
  setContentView(R.layout.main2);
  dataReceived = (EditText) findViewById(R.id.etDataReceived);
  btnDone = (Button) findViewById(R.id.btnDone);
  btnDone.setOnClickListener(this);
  Create(Bundle savedInstanceState) {

  // pick call made to Activity2 via Intent
  Intent myLocalIntent = getIntent();

  // look into the bundle sent to Activity2 for data items
  Bundle myBundle =  myLocalIntent.getExtras();
  Double v1 = myBundle.getDouble("val1");
  Double v2 = myBundle.getDouble("val2");

  // operate on the input data
  Double vResult =  v1 + v2;

  // for illustration purposes. show data received & result
  dataReceived.setText("Data received is \n"
              + "val1= " + v1 + "\nval2= " + v2
              + "\n\nresult= " + vResult);

   // add to the bundle the computed result
   myBundle.putDouble("vresult", vResult);

   // attach updated bumble to invoking intent
   myLocalIntent.putExtras(myBundle);

   // return sending an OK signal to calling activity
   setResult(Activity.RESULT_OK, myLocalIntent);

  }//onCreate

  @Override
  public void onClick(View v) {
     // close current screen - terminate Activity2
     finish();
  }//onClick

}//Activity2
```

Intents

Tutorial. Activity Excahange

Step5. Update the application‘s manifest. Add new <activity> tag for “Activity2“

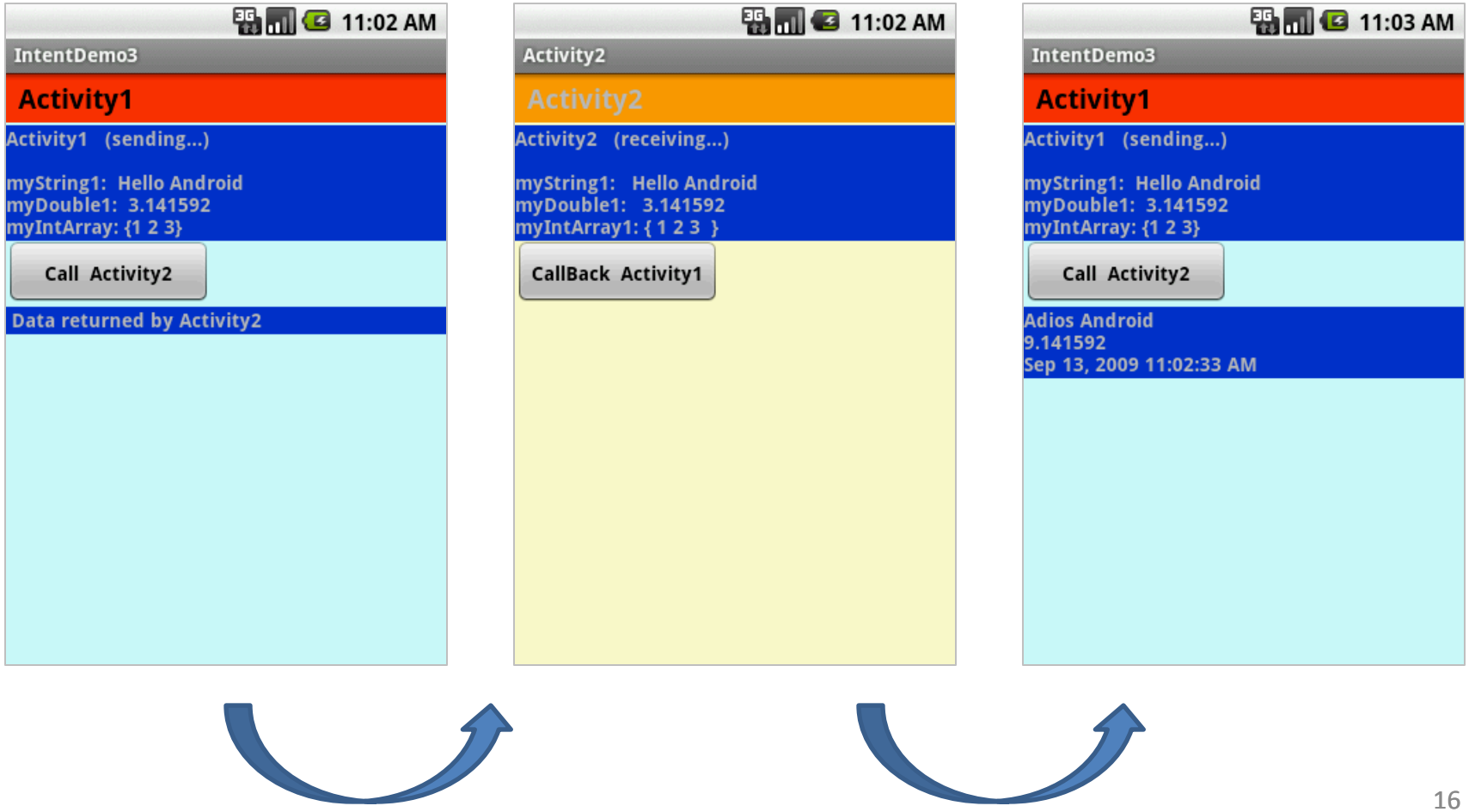
```
<?xml version="1.0" encoding="utf-8"?>
<manifest xmlns:android="http://schemas.android.com/apk/res/android"
      package="cis493.matos.intents6"
      android:versionCode="1"
      android:versionName="1.0">
    <application android:icon="@drawable/icon" android:label="@string/app_name">
        <activity android:name=".Activity1"
                  android:label="@string/app_name">
            <intent-filter>
                <action android:name="android.intent.action.MAIN" />
                <category android:name="android.intent.category.LAUNCHER" />
            </intent-filter>
        </activity>

        <activity
            android:name=".Activity2">
        </activity>

    </application>
    <uses-sdk android:minSdkVersion="4" />

</manifest>
```

add

This example is similar to previous. You may want to skip it.

Intents

Example: Activity1 invokes Activity2 using an Intent. A bundle containg a set of values is sent back-and-forth between both activities.



Intents

Example: Activity1 invokes Activity2 using an Intent. A bundle conating a set of values is sent back-and-forth between both activities (see 12IntentDemo3.zip).

```
//Activity1: Invoking a user-defined sub-activity
//sending and receiving results from the sub-activity
package cis493.intents;

import android.app.Activity;
import android.content.Intent;
import android.os.Bundle;
import android.view.View;
import android.view.View.OnClickListener;
import android.widget.*;

public class Activity1 extends Activity {
    TextView label1;
    TextView label1Returned;
    Button btnCallActivity2;
    private final int IPC_ID = 1122;
```

Intents

Example: Activity1 invokes Activity2 using an Intent. A bundle conating a set of values is sent back-and-forth between both activities.

```
@Override
public void onCreate(Bundle savedInstanceState) {
    super.onCreate(savedInstanceState);
    try {
        setContentView(R.layout.main);
        label1 = (TextView) findViewById(R.id.label1);
        label1Returned = (TextView) findViewById(R.id.label1Returned);
        btnCallActivity2 = (Button) findViewById(R.id.btnCallActivity2);
        btnCallActivity2.setOnClickListener(new Clicker1());
        // for demonstration purposes- show in top label
        label1.setText("Activity1   (sending...) \n\n"
                + "myString1:  Hello Android" + "\n"
                + "myDouble1:  3.141592     " + "\n"
                + "myIntArray: {1 2 3} ");
    } catch (Exception e) {
        Toast.makeText(getBaseContext(),
            e.getMessage(), Toast.LENGTH_LONG).show();
    }
}// onCreate
```

Intents

Example: Activity1 invokes Activity2 using an Intent. A bundle conating a set of values is sent back-and-forth between both activities.

```
private class Clicker1 implements OnClickListener {
    @Override
    public void onClick(View v) {
        try {
            // create an Intent to talk to Activity2
            Intent myIntentA1A2 = new Intent(Activity1.this,  Activity2.class);
            // prepare a Bundle and add the data pieces to be sent
            Bundle myData = new Bundle();
                myData.putString("myString1", "Hello Android");
                myData.putDouble("myDouble1", 3.141592);
                int[] myLittleArray = { 1, 2, 3 };
                myData.putIntArray("myIntArray1", myLittleArray);
                // bind the Bundle and the Intent that talks to Activity2
                myIntentA1A2.putExtras(myData);
            // call Activity2 and wait for results
            startActivityForResult(myIntentA1A2, IPC_ID);
        } catch (Exception e) {
            Toast.makeText(getBaseContext(), e.getMessage(),Toast.LENGTH_LONG).show();
        }
    }// onClick
}// Clicker1
```

Intents

Example: Activity1 invokes Activity2 using an Intent. A bundle conating a set of values is sent back-and-forth between both activities.

```
@Override
protected void onActivityResult(int requestCode, int resultCode, Intent data) {
    super.onActivityResult(requestCode, resultCode, data);
    try {
        switch (requestCode) {
        case IPC_ID: {
                //OK. This is the place to process the results sent back from the subactivity
                //see next slide
            } else {
                // user pressed the BACK button
                label1.setText("Selection CANCELLED!");
            }// if
            break;
        }// case
        }// switch
    } catch (Exception e) {
        Toast.makeText(getBaseContext(), e.getMessage(), Toast.LENGTH_LONG).show();
    }// try
}// onActivityResult

}// AndroIntent1
```

Intents

Example: Activity1 invokes Activity2 using an Intent. A bundle conating a set of values is sent back-and-forth between both activities.

```
@Override
protected void onActivityResult(int requestCode, int resultCode, Intent data) {
    super.onActivityResult(requestCode, resultCode, data);
    try {
        switch (requestCode) {
        case IPC_ID: {
            //OK. This is the place to process the results sent back from the sub-activity
            //see next slide
            } else {
                // user pressed the BACK button
                label1.setText("Selection CANCELLED!");
            }// if
            break;
        }// case
        }// switch
    } catch (Exception e) {
        Toast.makeText(getBaseContext(), e.getMessage(), Toast.LENGTH_LONG).show();
    }// try
}// onActivityResult

}// AndroIntent1
```

Intents

Example: Activity1 invokes Activity2 using an Intent. A bundle conating a set of values is sent back-and-forth between both activities.

```
// Activity2 is over - see what happened
if (resultCode == Activity.RESULT_OK) {

    // good! - we have some data sent back from Activity2
    Bundle myReturnedData = data.getExtras();

    String myReturnedString1 = myReturnedData.getString("myReturnedString1");
    Double myReturnedDouble1 = myReturnedData.getDouble("myReturnedDouble1");
    String myReturnedString2 = myReturnedData.getString("myCurrentTime");

    // display in the bottom label
    label1Returned.setText(myReturnedString1 + "\n"
                           + Double.toString(myReturnedDouble1) + "\n"
                           + myReturnedString2);
}
```

Intents

Example: Activity1 invokes Activity2 using an Intent. A bundle conating a set of values is sent back-and-forth between both activities.

```
// Activity2. This subactivity receives a bundle of data, performs some work on the data and,
// returns results to Activity1.
package cis493.intents;
import java.util.Date;
import android.app.Activity;
import android.content.Intent;
import android.os.Bundle;
import android.view.View;
import android.view.View.OnClickListener;
import android.widget.*;

public class Activity2 extends Activity {
   TextView label2;
   Button   btnCallActivity1;
```

Intents

Example: Activity1 invokes Activity2 using an Intent. A bundle conating a set of values is sent back-and-forth between both activities.

```
// Activity2 – cont…
 @Override
   public void onCreate(Bundle savedInstanceState) {
      super.onCreate(savedInstanceState);
      setContentView(R.layout.main2);
      //bind UI variables to Java code
      label2 = (TextView)findViewById(R.id.label2);
      btnCallActivity1 = (Button)findViewById(R.id.btnCallActivity1);
      btnCallActivity1.setOnClickListener(new Clicker1());

      //create a local Intent handler – we have been called!
      Intent myLocalIntent = getIntent();

      //grab the data package with all the pieces sent to us
      Bundle myBundle = myLocalIntent.getExtras();

      //extract the individual data parts of the bundle
      String str1 = myBundle.getString("myString1");
      double dob1 = myBundle.getDouble("myDouble1");
      int[]  arr1 = myBundle.getIntArray("myIntArray1");
```

Intents

Example: Activity1 invokes Activity2 using an Intent. A bundle conating a set of values is sent back-and-forth between both activities.

```
 //Activity2 – cont…
//do something with the data here (for example...)
String strArr  = "{ ";
int sumIntValues = 0;
for (int i=0; i<arr1.length; i++) {
        sumIntValues += arr1[i];
        strArr += Integer.toString( arr1[i] ) + " ";
}
strArr += " }";

//show arriving data in GUI label
label2.setText("Activity2   (receiving...) \n\n" +   "myString1:   " + str1 + "\n" +
     "myDouble1:   " + Double.toString(dob1) + "\n" +    "myIntArray1: " + strArr);

//now go back to myActivity1 with some data made here
double someNumber = sumIntValues + dob1;
myBundle.putString("myReturnedString1", "Adios Android");
myBundle.putDouble("myReturnedDouble1", someNumber);
myBundle.putString("myCurrentTime", new Date().toLocaleString() );
myLocalIntent.putExtras(myBundle);

setResult(Activity.RESULT_OK, myLocalIntent);
}//onCreate());
```

Intents

Example: Activity1 invokes Activity2 using an Intent. A bundle conating a set of values is sent back-and-forth between both activities.

Layout main.xml

```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout
android:id="@+id/linLayout"
android:layout_width="fill_parent"
android:layout_height="fill_parent"
android:background="#ffccffff"
android:orientation="vertical"
xmlns:android="http://schemas.android.com/apk/res/android"
>
<TextView
android:id="@+id/caption1"
android:layout_width="fill_parent"
android:layout_height="wrap_content"
android:background="#ffff3300"
android:padding="4sp"
android:text=" Activity1 "
android:textSize="20px"
android:textStyle="bold"
android:textColor="#ff000000"
>
</TextView>
<TextView
android:id="@+id/widget107"
android:layout_width="fill_parent"
android:layout_height="2sp"
>
</TextView>
<TextView
android:id="@+id/label1"
android:layout_width="fill_parent"
android:layout_height="wrap_content"
android:background="#ff0033cc"
android:text="Data to be sent to SubActivity:"
android:textStyle="bold"
>
</TextView>
<Button
android:id="@+id/btnCallActivity2"
android:layout_width="149px"
android:layout_height="wrap_content"
android:padding="6sp"
android:text="Call  Activity2"
android:textStyle="bold"
>
</Button>
<TextView
android:id="@+id/label1Returned"
android:layout_width="fill_parent"
android:layout_height="wrap_content"
android:background="#ff0033cc"
android:text=" Data returned by Activity2"
android:textStyle="bold"
>
</TextView>
</LinearLayout>
```

Intents

Example: Activity1 invokes Activity2 using an Intent. A bundle conating a set of values is sent back-and-forth between both activities.

Layout main2.xml

```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout
android:id="@+id/linearLayout"
android:layout_width="fill_parent"
android:layout_height="fill_parent"
android:background="#ffffffcc"
android:orientation="vertical"
xmlns:android="http://schemas.android.com/apk/res/android"
>
<TextView
android:layout_width="fill_parent"
android:layout_height="wrap_content"
android:background="#ffff9900"
android:padding="4sp"
android:text=" Activity2"
android:textSize="20px"
android:textStyle="bold"
>
</TextView>
<TextView
android:id="@+id/widget107"
android:layout_width="fill_parent"
android:layout_height="2sp"
>
</TextView>
<TextView
android:id="@+id/label2"
android:layout_width="fill_parent"
android:layout_height="wrap_content"
android:background="#ff0033cc"
android:text="Data Received from Activity1 ..."
android:textStyle="bold"
>
</TextView>
<Button
android:id="@+id/btnCallActivity1"
android:layout_width="149px"
android:layout_height="wrap_content"
android:padding="6sp"
android:text="CallBack  Activity1"
android:textStyle="bold"
>
</Button>
</LinearLayout>
```

Intents

Questions ?