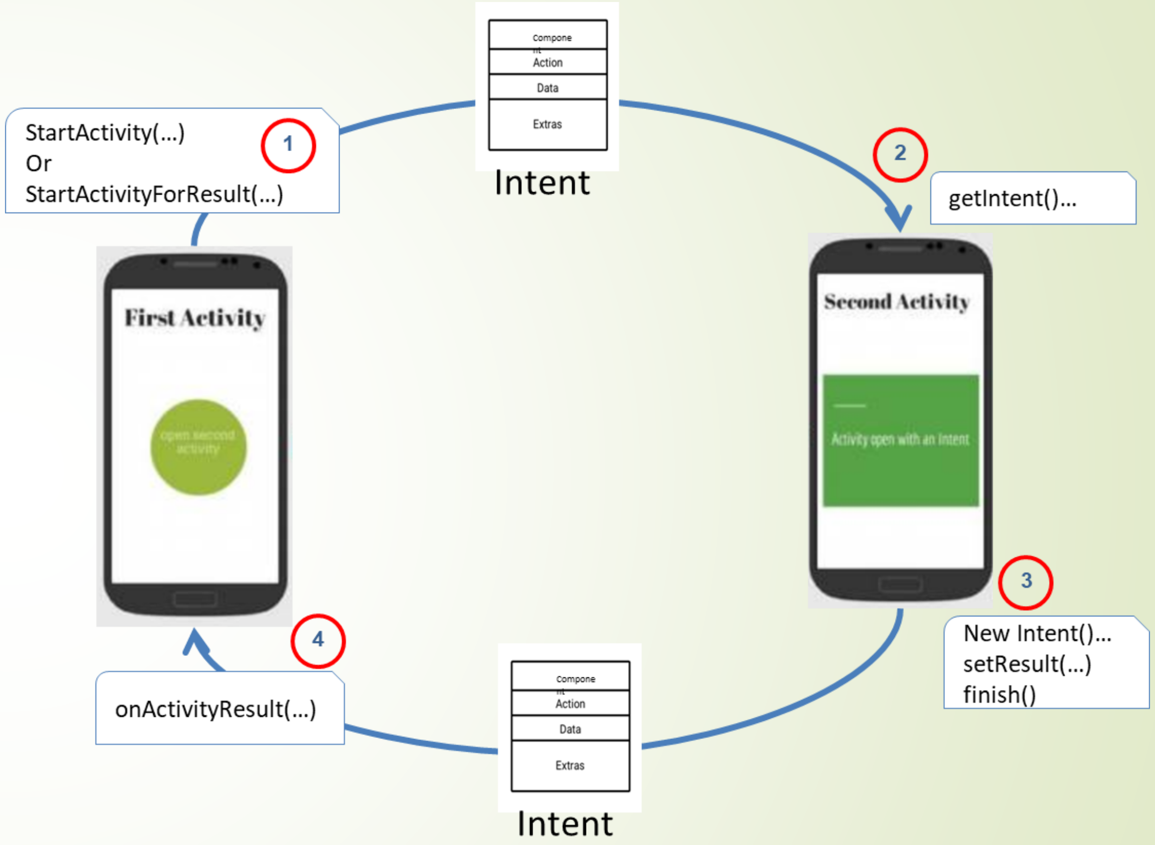
Android Development Lab

- Course no.: 61985

- Lecturer: Daniel Bouenos
- Email: dbouenos@braude.ac.il
- Topics: Communication between Activities

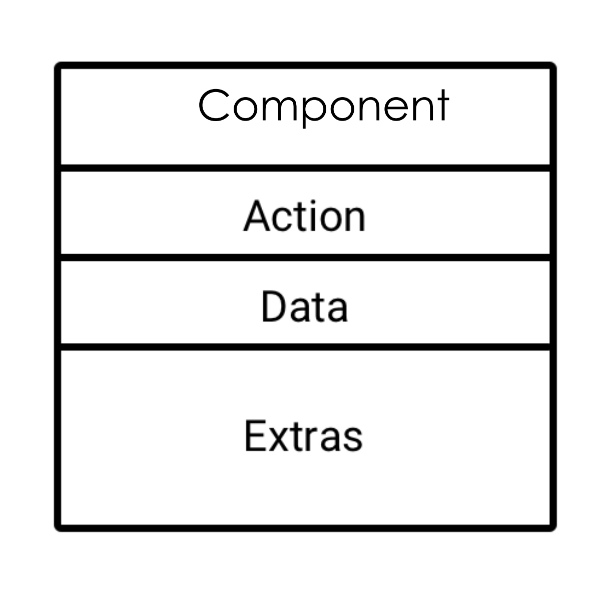
Multiple Activities & Communication

- Your app may contain multiple activities
  - Loosely couples from each other
  - All must be registered in the manifest file
- An activity may start other activity in the same app or in an other app
- Calling for other activity is done asynchronously
- Calling is done with
  - startActivity method – in the case that no response is expected, or…
  - startActivityForResult – in case that a response is expected. It will be handled by "onActivityResult" callback
- Passing data between Activities is done using the "Intent" object (see next slide…)



Communicating using "Intents"

- A dynamic data structure that is used to:
  - Identify the target activity "component"
  - State the requested "action"
  - transmit a "data" link to work on
  - Provide key-value dictionary for "extras"
  - See details in this link1 and link2…
- The "component" piece may me missing
  - In such case, we call it "Implicit Intent"
  - Otherwise, it's called "Explicit Intent"
- The "data" piece includes URI based format link
  - See URI details here…



Component

Action

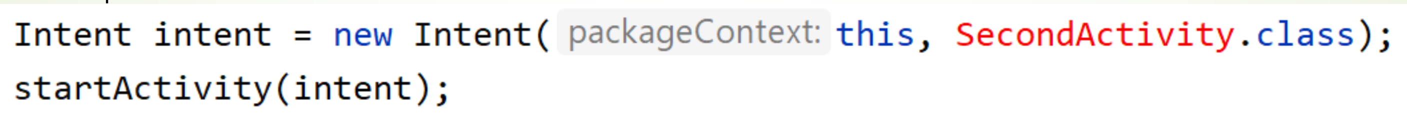
Data

Extras

Explicit Intent

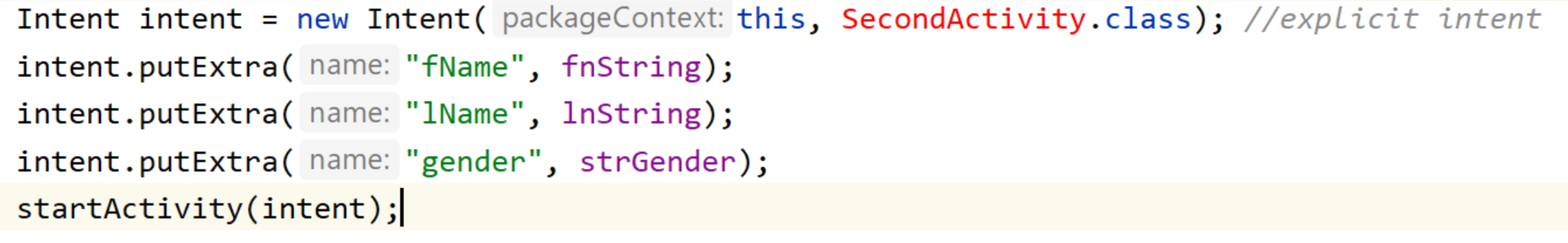
- Used mainly for communication between activities in the same app
- The source knows the exact name of the target "component"
- Let's see some examples…

Example 1:



```
Intent intent = new Intent( packageContext: this, SecondActivity.class);
startActivity(intent);
```

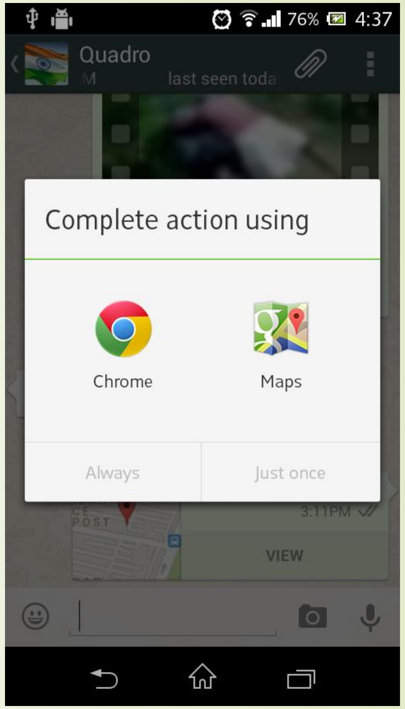
Example 2:



```
Intent intent = new Intent( packageContext: this, SecondActivity.class); //explicit intent
intent.putExtra( name: "fName", fnString);
intent.putExtra( name: "lName", lnString);
intent.putExtra( name: "gender", strGender);
startActivity(intent);
```

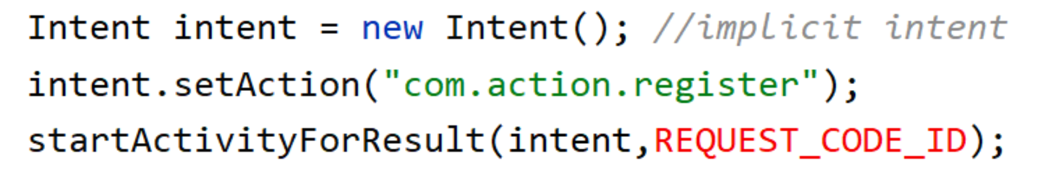
Implicit Intent

- The goal in this case is not to load a specific activity but instead look for any activity in the phone that may handles my request
- In such case, the intent you prepare doesn't include the "component" piece but the "action" that you need to fulfil
- In case that Android finds more that one app that can fulfil the action it will pop out a "chooser dialog"
- How an activity publish what "actions" it may fulfil?
  - Add the action to the <Intent-Filter> tag in the Manifest file



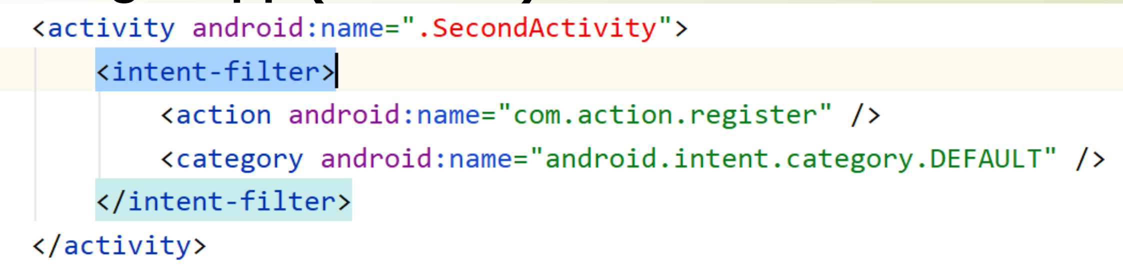
Complete action using

Source app



```
Intent intent = new Intent(); //implicit intent
intent.setAction("com.action.register");
startActivityForResult(intent,REQUEST_CODE_ID);
```

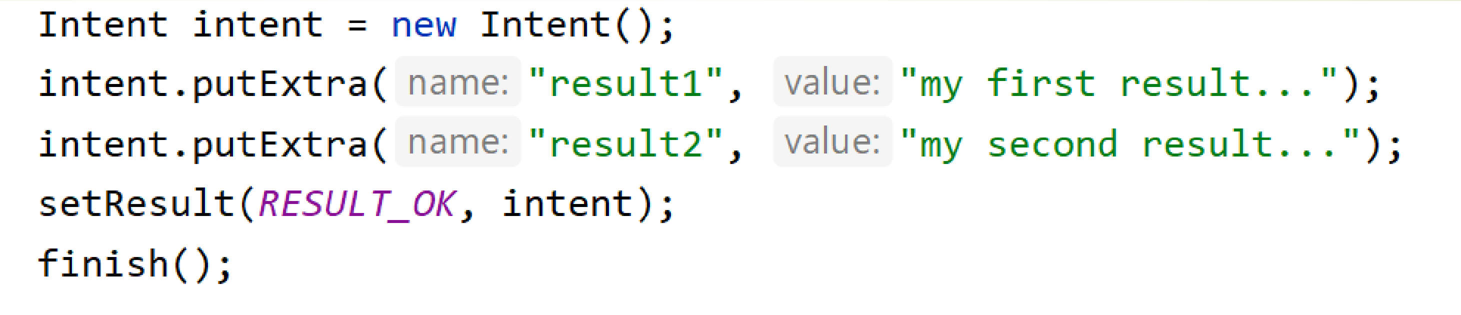
Target app (Manifest)



```
<activity android:name=".SecondActivity">
    <intent-filter>
        <action android:name="com.action.register" />
        <category android:name="android.intent.category.DEFAULT" />
    </intent-filter>
</activity>
```

How results are sent back from Target activity?

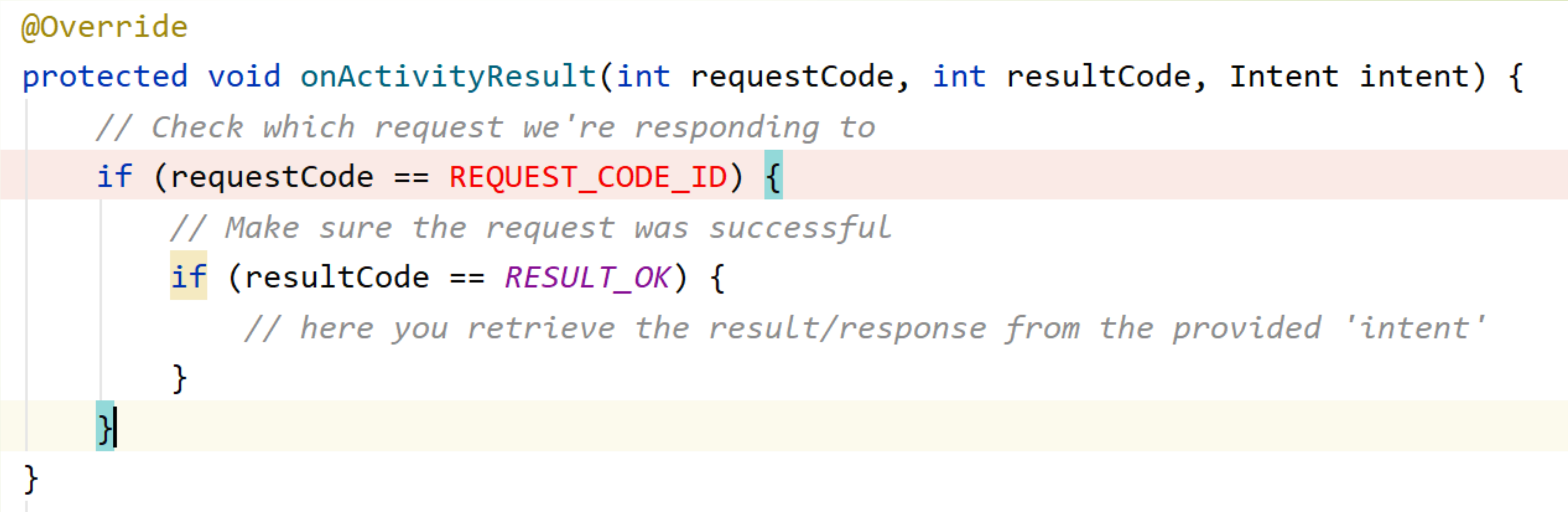
- Open a new intent
- Add your result to the ‘extras’ slot
- Call ‘setResult’ with the intent
- Call ‘finish()’ to close the current activity



```
Intent intent = new Intent();
intent.putExtra( name: "result1",  value: "my first result...");
intent.putExtra( name: "result2",  value: "my second result...");
setResult(RESULT_OK, intent);
finish();
```

Getting the result/response back

- In the case where a response/result is expected to return from the target activity:
  1. The launching of the target activity is different:
     - startActivityForResult(intent, requestCode);
     - The request code is an arbitrary number that will be used later when the result is received
  2. You also need to override the method
     - onActivityResult(int requestCode, int resultCode, Intent data)



```java
@Override
protected void onActivityResult(int requestCode, int resultCode, Intent intent) {
    // Check which request we're responding to
    if (requestCode == REQUEST_CODE_ID) {
        // Make sure the request was successful
        if (resultCode == RESULT_OK) {
            // here you retrieve the result/response from the provided 'intent'
        }
    }
}
```

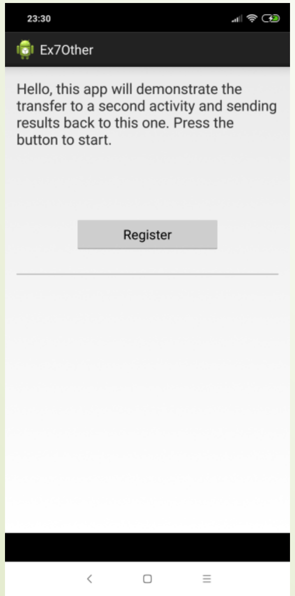
This week assignment – EX3

- Three steps:
  1. Create and test **ex3a** with 2 activities, see next slide…
  2. Clone it to become **ex3b,** see next^2 slide…
  3. Create and test **ex3Client** with one activity that communicate with ex3a/ex3b. See next^3 slide…

Ex3a app

- An app with two activities (not three!)
- Clicking on ‘Register’ will open the second activity
- Type in the first and last name, then choose the gender and click send back button
- The result will be sent back to the first activity, that will change its textView and button content accordingly (see third image below)

Activity 1

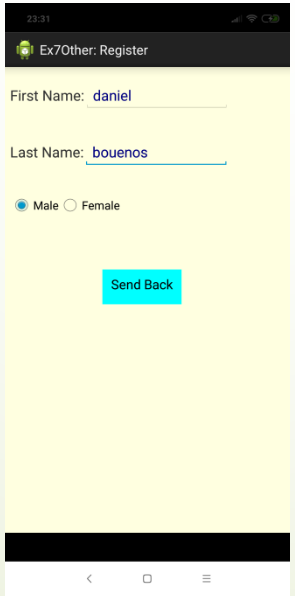

23:30

Ex7Other

Hello, this app will demonstrate the transfer to a second activity and sending results back to this one. Press the button to start.

Register

Activity 2


23:31

Ex7Other: Register

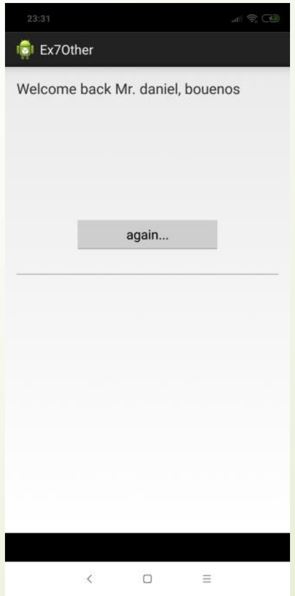
First Name: daniel

Last Name: bouenos

● Male ○ Female

Send Back

Activity 1


23:31

Ex7Other

Welcome back Mr. daniel, bouenos

again...

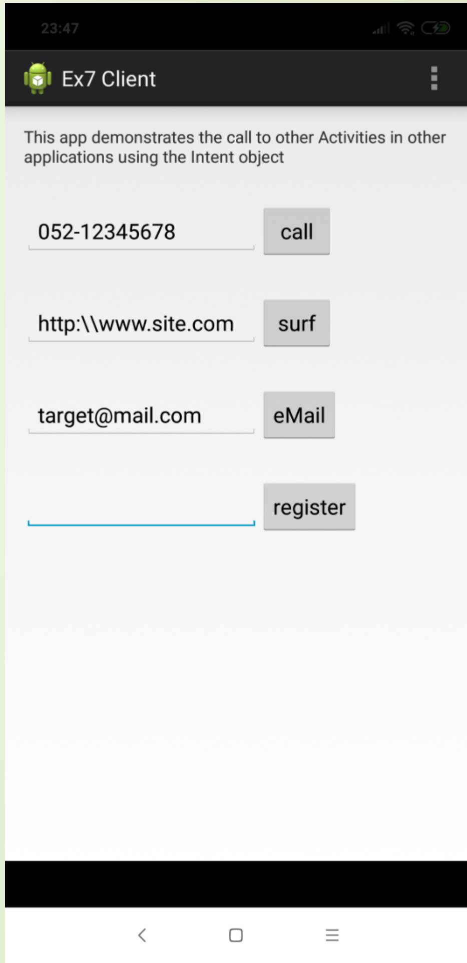
Clone ex3a to become ex3b app

1. Select the app folder and press File->’show in explorer’
2. Copy-past the ex3a folder and rename the new one ‘ex3b’
3. Open the new project in Android Studio
4. Open ‘string.xml and rename the ‘app_name’ to ‘ex3b’
5. Open ‘build.gradle’ and rename the ‘applicationId’ to "com.example.ex3b"
6. Run the new project to install it on your device

Create ex3Client app

- A new app with 1 activity as described in the image
- Exercise the implicit calls

1. Typing a number and pressing ‘call’ will open the phone dialer
2. Typing a url and pressing ‘surf’ will open the page in web explorerer
3. Typing an email address and pressing ‘email’ will open your device email app
4. Pressing the ‘register’ button will
   1. Ask for implicit activity with intent filter “com.action.ex3.register”
   2. Android will pop up a ‘app chooser’ to let you select between ex3a and ex3b
   3. Choosing one of them will open the <u>second activity </u>in the selected app
   4. Fill the first/last/gender and pressing ‘send back’ will close it and return to ex3Client and fill the textView with ‘Mr/Ms first last name’



This app demonstrates the call to other Activities in other applications using the Intent object

052-12345678

http:\\www.site.com

target@mail.com