

PROGRAMMING HANDHELD SYSTEMS

ADAM PORTER

THE INTENT CLASS

TODAY'S TOPICS

THE INTENT CLASS

STARTING ACTIVITIES WITH INTENTS

EXPLICIT ACTIVATION

IMPLICIT ACTIVATION VIA INTENT
RESOLUTION

THE INTENT CLASS

A DATA STRUCTURE THAT REPRESENTS
AN OPERATION TO BE PERFORMED, OR
AN EVENT THAT HAS OCCURRED

TODAY'S FOCUS

USING INTENTS TO SPECIFY
OPERATIONS TO BE PERFORMED, NOT
FOR EVENT NOTIFICATION

I.E., INTENTS USED TO START A SINGLE
ACTIVITY

WE'LL COVER USING INTENTS FOR EVENT
NOTIFICATION WHEN WE TALK ABOUT
BROADCASTRECEIVERS

INTENTS AS DESIRED OPERATIONS

INTENTS PROVIDE A FLEXIBLE LANGUAGE
FOR SPECIFYING OPERATIONS TO BE
PERFORMED

E.G., PICK A CONTACT, TAKE A PHOTO, DIAL
A PHONE NUMBER

INTENTS AS DESIRED OPERATIONS

INTENT IS CONSTRUCTED BY ONE
COMPONENT THAT WANTS SOME WORK
DONE

RECEIVED BY ONE ACTIVITY THAT CAN
PERFORM THAT WORK

INTENT FIELDS

ACTION

DATA

CATEGORY

TYPE

COMPONENT

EXTRAS

FLAGS

ACTION

STRING REPRESENTING DESIRED OPERATION

EXAMPLES

ACTION_DIAL – DIAL A NUMBER

ACTION_EDIT – DISPLAY DATA TO EDIT

ACTION_SYNC – SYNCHRONIZE DEVICE
DATA WITH SERVER

ACTION_MAIN – START AS INITIAL
ACTIVITY OF APP

SETTING THE INTENT ACTION

```
Intent newInt = new  
    Intent(Intent.ACTION_DIAL);
```

OR

```
Intent newInt = new Intent();  
newInt.setAction(Intent.ACTION_DIAL);
```

DATA

DATA ASSOCIATED WITH THE INTENT
FORMATTED AS A UNIFORM RESOURCE
IDENTIFIER (URI)

EXAMPLES

DATA TO VIEW ON A MAP

```
Uri.parse("geo:0,0?  
q=1600+Pennsylvania+  
Ave+Washington+DC")
```

NUMBER TO DIAL IN THE PHONE DIALER

```
Uri.parse("tel:+15555555555")
```

SETTING INTENT DATA

```
Intent newInt = new Intent (  
    Intent.ACTION_DIAL,  
    Uri.parse("tel:+15555555555"));
```

OR

```
Intent newInt =  
    new Intent(Intent.ACTION_DIAL);  
newInt.setData(  
    Uri.parse("tel:+15555555555"));
```

CATEGORY

ADDITIONAL INFORMATION ABOUT THE
COMPONENTS THAT CAN HANDLE THE INTENT

EXAMPLES

CATEGORY_BROWSABLE – CAN BE INVOKED BY A BROWSER TO DISPLAY DATA REF'S BY A URI

CATEGORY_LAUNCHER – CAN BE THE INITIAL ACTIVITY OF A TASK & IS LISTED IN TOP-LEVEL APP LAUNCHER

TYPE

SPECIFIES THE MIME TYPE OF THE INTENT
DATA

EXAMPLES

image/*, image/png, image/jpeg

text/html, text/plain

IF UNSPECIFIED, ANDROID WILL INFER THE
TYPE

SETTING THE TYPE

```
Intent.setType(String type)
```

OR

```
Intent.setDataAndType(Uri data,  
                        String type)
```

COMPONENT

THE COMPONENT THAT SHOULD RECEIVE
THIS INTENT

USE THIS WHEN THERE'S EXACTLY ONE
COMPONENT THAT SHOULD RECEIVE THE
INTENT

SETTING THE COMPONENT

```
Intent newInt = Intent(  
    Context packageContext, Class<?> cls);
```

SETTING THE COMPONENT

OR

```
Intent newInt = new Intent ();
```

and one of:

```
setComponent(), setClass(), or setClassName()
```

EXTRAS

ADD'L INFORMATION ASSOCIATED WITH INTENT
TREATED AS A MAP (KEY-VALUE PAIRS)

EXAMPLES

Intent.EXTRA_EMAIL: EMAIL RECIPIENTS

```
Intent newInt = new Intent(Intent.ACTION_SEND);
newInt.putExtra(android.content.Intent.EXTRA_EMAIL,
    new String[]{
        "aporter@cs.umd.edu", "ceo@microsoft.com",
        "potus@whitehouse.gov", "mozart@musician.org"
    }
);
```

SETTING THE EXTRA ATTRIBUTE

SEVERAL FORMS DEPENDING ON DATA TYPE

```
putExtra(String name, String value);
```

```
putExtra(String name, float[] value);
```

...

FLAGS

SPECIFY HOW INTENT SHOULD BE HANDLED

EXAMPLES

FLAG_ACTIVITY_NO_HISTORY

DON'T PUT THIS ACTIVITY IN THE HISTORY STACK

FLAG_DEBUG_LOG_RESOLUTION

PRINT EXTRA LOGGING INFORMATION WHEN THIS
INTENT IS PROCESSED

SETTING FLAGS

```
Intent newInt =  
    new Intent(Intent.ACTION_SEND);  
newInt.setFlags(  
    Intent.FLAG_ACTIVITY_NO_HISTORY);
```

STARTING ACTIVITIES WITH INTENTS

`startActivity(Intent intent,...)`

`startActivityForResult(Intent intent, ...)`

THE TARGET ACTIVITY

CAN BE NAMED EXPLICITLY BY SETTING
THE INTENT'S COMPONENT

CAN BE DETERMINED IMPLICITLY

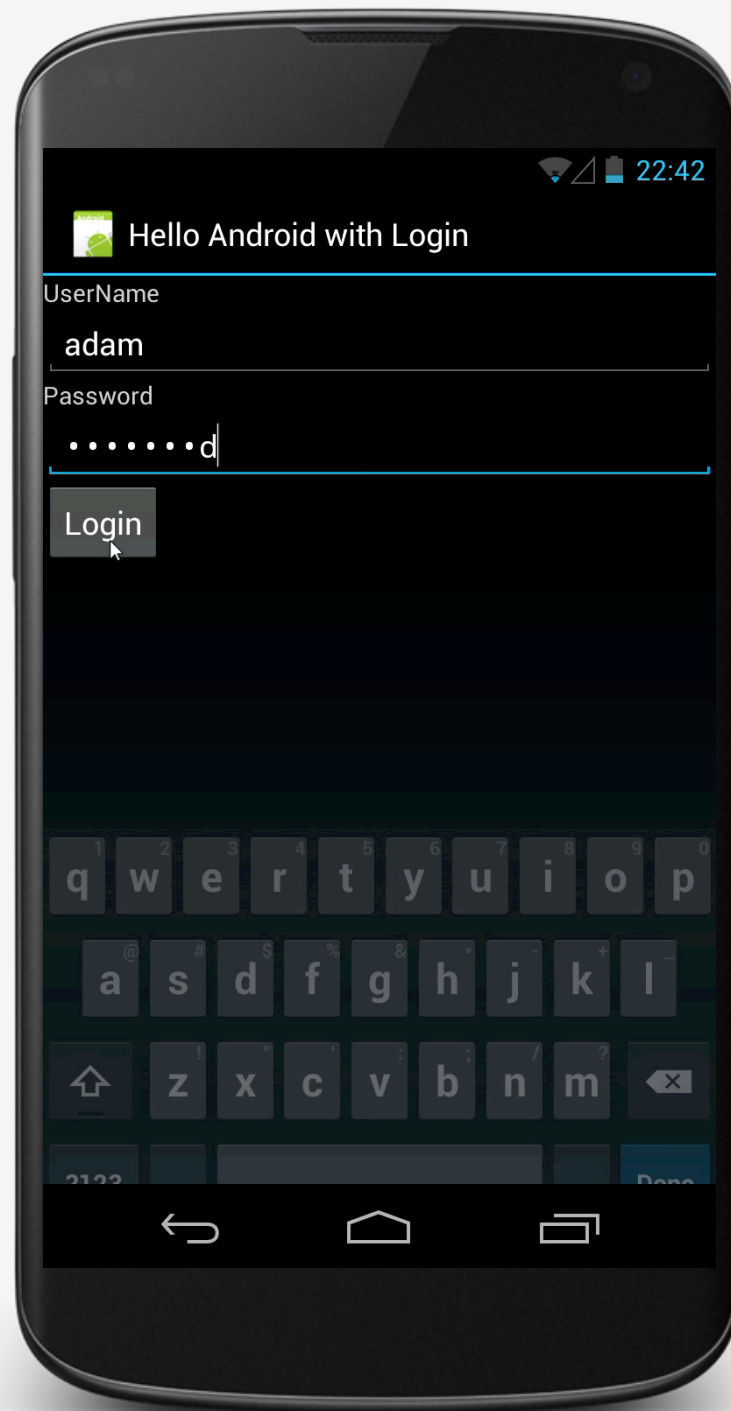
EXPLICIT ACTIVATION

HELLOWORLDWITHLOGIN

TWO ACTIVITIES

LOGINACTIVITY CHECKS USERNAME &
PASSWORD AND THEN STARTS
HELLOANDROIDACTIVITY

HELLOANDROIDACTIVITY SHOWS "HELLO
ANDROID" MESSAGE



HELLOWORLDWITHLOGIN

```
@Override
public void onCreate(Bundle savedInstanceState) {
    super.onCreate(savedInstanceState);
    setContentView(R.layout.loginscreen);

    final EditText uname = (EditText) findViewById(R.id.username_edittext);
    final EditText passwd = (EditText) findViewById(R.id.password_edittext);

    final Button loginButton = (Button) findViewById(R.id.login_button);
    loginButton.setOnClickListener(new OnClickListener() {

        public void onClick(View v) {

            if (checkPassword(uname.getText(), passwd.getText())) {

                // Create an explicit Intent for starting the HelloAndroid Activity
                Intent helloAndroidIntent = new Intent(LoginScreen.this,
                    HelloAndroid.class);

                // Use the Intent to start the HelloAndroid Activity
                startActivity(helloAndroidIntent);

            } else {
                uname.setText("");
                passwd.setText("");
            }
        }
    });
}
```

IMPLICIT ACTIVATION

WHEN THE ACTIVITY TO BE ACTIVATED
IS NOT EXPLICITLY NAMED, ANDROID
TRIES TO FIND ACTIVITIES THAT MATCH
THE INTENT

THIS PROCESS IS CALLED INTENT
RESOLUTION

INTENT RESOLUTION PROCESS

AN INTENT DESCRIBING A DESIRED
OPERATION

INTENTFILTERS WHICH DESCRIBE WHICH
OPERATIONS AN ACTIVITY CAN HANDLE

SPECIFIED EITHER IN ANDROIDMANIFEST.XML
OR PROGRAMMATICALLY

INTENT RESOLUTION DATA

ACTION

DATA (BOTH URI & TYPE)

CATEGORY

SPECIFYING INTENTFILTERS

```
<activity ...>
  <intent-filter ...>
    ...
    <action android:name="actionName" />
    ...
  </intent-filter>
  ...
</activity>
```

HANDLING INTENT.ACTION_DIAL

```
<activity ...>
  <intent-filter ...>
    ...
    <action android:name=
      "android.intent.action.DIAL" />
    ...
  </intent-filter>
  ...
</activity>
```

ADDING DATA TO INTENTFILTER

```
<intent-filter ...>
  ...
  <data
    android:mimeType="string"
    android:scheme="string"
    android:host="string"
    android:port="string"
    android:path="string"
    android:pathPattern="string"
    android:pathPrefix="string"
  />
  ...
</intent-filter>
```

See: <http://developer.android.com/guide/components/intents-filters.html>

HANDLING GEO: SCHEME INTENTS

```
<intent-filter ...>
```

```
...
```

```
<data android:scheme="geo" />
```

```
...
```

```
</intent-filter>
```

ADDING A CATEGORY TO INTENTFILTER

```
<intent-filter ...>
```

```
...
```

```
<category android:name="string" />
```

```
...
```

```
</intent-filter>
```

EXAMPLE: MAPS APPLICATION

EXAMPLE: MAPS APPLICATION

```
<intent-filter ...>  
  <action android:name =  
    "android.intent.action.VIEW" />  
  <category android:name =  
    "android.intent.category.DEFAULT" />  
  <category android:name=  
    "android.intent.category.BROWSABLE"/>  
  <data android:scheme = "geo"/>  
</intent-filter>
```

RECEIVING IMPLICIT INTENTS

NOTE: TO RECEIVE IMPLICIT INTENTS AN
ACTIVITY SHOULD SPECIFY AN
INTENTFILTER WITH THE CATEGORY
"ANDROID.INTENT.CATEGORY.DEFAULT"

PRIORITY

ANDROID:PRIORITY – PRIORITY GIVEN TO THE PARENT COMPONENT WHEN HANDLING MATCHING INTENTS

CAUSES ANDROID TO PREFER ONE ACTIVITY OVER ANOTHER

VALUE SHOULD BE GREATER THAN -1000 & LESS THAN 1000

HIGHER VALUES REPRESENT HIGHER PRIORITIES

INVESTIGATE INTENT FILTERS

```
% adb shell dumpsys package
```

```
% adb shell dumpsys package
```


NEXT TIME

PERMISSIONS