

ZomDroid

Using Intents to Transfer Data Between Activities

One way to have your service receive game data that has been updated by other activities is to use an `INTENT` to activate your service and have the `INTENT` carry the information as an `EXTRA`.

The following code shows how a java object can be delivered to an activity via an Intent. The basic idea is to

- serialize the java object with all its data values
- add the object as an Intent Extra giving it some lookup key for the receiving activity
- fire the intent

Here's a Java object we want to serialize. All we need to do is declare `implements java.io.Serializable`

Note that the data fields are all public and do not use accessor methods. This may be poor OO design but it is recommended Android design to reduce the overhead of method calls.

```
package com.hwk.intent;

public class Player implements java.io.Serializable {

    public int id;
    public char zhType;
    public float gpx;
    public float gpy;
    public int score;

    public Player(){
        id= 1234;
        zhType = 'Z';
        gpx = (float) 37.12123;
        gpy = (float) -96.431231;
        score = 5000;
    }

    public String getStats()
    {
        return("id= " + id +
            " zhType= " + zhType + " gpx= " +
            gpx + " gpy= " + gpy + "score= " + score);
    }
}
```

SENDER

This activity creates an instance of Player. The constructor sets values for our convenience.

```
package com.hwk.intent;

import java.util.ArrayList;
import edu.smu.zomdroid.Player;
import android.app.Activity;
import android.content.Intent;
import android.os.Bundle;

public class Send extends Activity {
    private Player plyr;

    /** Called when the activity is first created. */
    @Override
    public void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.main);

        plyr = new Player(); // object has values

        // create an intent and specify the class
        // to handle the Intent
        Intent i = new Intent(this, Receive.class);

        // attach some extra data to the Intent
        // the first parameter is the key for the receiver
        // to access the object which is passed as second param
        // this only works because plyr is declared serializable
        i.putExtra("plyr", plyr);

        // transfer control
        startActivity(i);
    }
}
```

RECEIVER CLASS

```
package com.hwk.intent;

import edu.smu.zomdroid.Player;
import android.app.Activity;
import android.os.Bundle;

public class Receive extends Activity {

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

        // ask the Intent for the Bundle that holds the extras
        Bundle extras = getIntent().getExtras();

        // use the key "plyr" to the serializable object
        // note that we must cast it to what we know the class to be
        Player plyr = (Player) extras.getSerializable("plyr");

        // access the data we were passed
        String data = plyr.getStats();

        Toast.makeText(getBaseContext(), data,
                       Toast.LENGTH_SHORT).show();
    }
}
```

Manifest.xml

```
<?xml version="1.0" encoding="utf-8"?>
<manifest xmlns:android="http://schemas.android.com/apk/res/android"
    package="com.hwk.intent"
    android:versionCode="1"
    android:versionName="1.0">
    <application android:icon="@drawable/icon"
        android:label="@string/app_name">

        <activity android:name=".Send"
            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=".Receive"
            android:label="@string/app_name"
            android:debuggable="true">
        </activity>
    </application>

    <uses-sdk android:minSdkVersion="3" />
</manifest>
```

Strings.xml

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