

Creating Activities for Custom Objects

According to the DTD of the settings.xml file, when you have a custom object setting in your **settings.xml** file, you need to define an Android activity that will serve as a GUI for the user, so that he/she can edit the value of the custom object. This tutorial will help you create this activity. If you have any questions, please send an email to sw6b@lcdev.dk.

Walkthrough

1. Create a class that extends **android.app.Activity**
2. In your **AndroidManifest.xml** file, register the activity, and mark it as exported (see snippet 1).
3. In your activity, you are responsible for retrieving the object from sw6.admin's database. To get the object you can use the attributes specified in **settings.xml** for the particular setting. The attributes are bundled in the intent sent by sw6.admin, and in snippet 2 it is shown how to retrieve the attributes of the setting. The following attributes can be read:

○ varName	(Settings.OBJECT_ATTRIBUTE_VARNAME)
○ type	(Settings.OBJECT_ATTRIBUTE_DATATYPE)
○ desc	(Settings.OBJECT_ATTRIBUTE_DESCRIPTION)
○ realName	(Settings.OBJECT_ATTRIBUTE_REALNAME)
○ packageName	(Settings.OBJECT_ATTRIBUTE_PACKAGENAME)

In snippet 3, it is shown how to retrieve the custom object via sw6.lib.

4. Implement the GUI in the activity. When you want to update the value of the setting, use the **updateObject(...)** method specified in sw6.lib.Settings.

Code Snippet 1

```
<activity
    android:name="sw6.programExample.CustomActivity"
    android:exported="true">
</activity>
```

Code Snippet 2

```
public class CustomActivity extends Activity {
    @Override
    public void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.custom);
        Button button = (Button) findViewById(R.id.button1);
        Bundle extra = getIntent().getExtras();
        String varName =
            extra.getString(Settings.OBJECT_ATTRIBUTE_VARNAME);
        button.setText(varName);
    }
}
```

Code Snippet 3

```
StringBuilder sb = Settings.getObject(this, "someVarName", StringBuilder.class);
```

Note: the last variable is the class instance of the Object type to retrieve. This should match with the type specified for the object in settings.xml.