

Method finder

This is a tool that I use at home to enable me to look at what methods are available in most of the controls I use.

It is incomplete as it does not drill down far enough but it is much better than nothing.

Warnings

Please be aware that using undocumented functions is not recommended by the DroidScript developers and you do so at your own risk.

Please also be aware that the **“Manually entered code”** option can be very dangerous.

Acknowledgements

Andreas Rozek would recognize much of the code as the heart of it is stolen from his post at https://groups.google.com/forum/#!topic/androidscript/o_RCNR5ozAE

Thanks also to Salvatore Fusto for finding a bug in the app.

The code

[introspector.js](#)

```
// Introspection lite tool for home use by Steve Garman
// Main code stolen from AppInspector by Andreas Rozek

//Global variables
var myobj = app;
var ListView, menuView, FunctionView
var main_header_txt, edtFilter, dlg, edtDlg;
var lastMenu = "app";
var pasteable = "null";
var vers = Math.floor((app.GetVersion()+.0005) * 100);
var docfold = "/sdcard/sjgDocs/androidscript/"+vers;
var docfile = docfold + "/null.txt";
var evalsfile = docfold + "/evals.txt"

//Called when application is started.
function OnStart(){
    app.EnableBackKey( false );

    //Create a layout with objects vertically centered.
    var lay = app.CreateLayout( "linear", "Vertical,FillXY" );
```

```
add_top_bar( lay );

/**** prepare page area ****/
var PageArea = app.CreateLayout('frame','fillxy');
lay.AddChild(PageArea);

var control_list = "app,layout,Manually entered
code,BluetoothList,BluetoothSerial,Button,CameraView,CheckBox"
+",Crypt,Database,Dialog,Email,Image,List,Locator,Notification,NXT,NxtR
emote,Odroid,SeekBar"
+",Sensor,SMS,Spinner,Tabs,Text,TextEdit,ToggleButton"
+",WebView,WebServer,YesNoDialog";

menuView = app.CreateList( control_list, -1,-1, 'fillxy');
menuView.SetOnTouch(menuView_OnTouch);
PageArea.AddChild(menuView);
menuView.SetVisibility('Hide');

ListView = app.CreateList("", -1,-1, 'fillxy');
prepareFunctionList();

ListView.SetOnTouch(showAppFunction);
PageArea.AddChild(ListView);

/**** prepare detail view ****/

FunctionView = app.CreateTextEdit('',-1,-1);
//FunctionView = app.CreateTextEdit('',-1,-1,"nospell");
PageArea.AddChild(FunctionView);
FunctionView.SetVisibility('Hide');

//Add layout to app.
app.AddLayout( lay );
showMenuView();
} //OnStart

function add_top_bar (layout, opts){
//basic top bar layout
var lay_bar = app.CreateLayout( "Linear", "Horizontal,Left, FillX" );
lay_bar.SetBackGradient( "#444444", "#888888" );

//add app name if required - stolen for object/function name in this
app
main_header_txt = app.CreateText( lastMenu , -1, -1 );
main_header_txt.SetTextSize( 24 );
main_header_txt.SetPadding( 0.01, 0, 0.01, 0 );
lay_bar.AddChild( main_header_txt );

var filterlabel = app.CreateText("Func filter:");
filterlabel.SetTextSize(18);
filterlabel.SetTextColor("#ff222222");
```

```
lay_bar.AddChild( filterlabel );
edtFilter = app.CreateTextEdit("",0.3);
edtFilter.SetBackColor("#ff222222");
edtFilter.SetTextSize(18);
edtFilter.SetPadding( 0.01, 0, 0, 0.001 );
edtFilter.SetOnChange(edtFilterOnChange);
lay_bar.AddChild(edtFilter);

lyout.AddChild( lay_bar );
} //add_top_bar

function menuView_OnTouch(name){
    lastMenu = name;
    switch(name) {
        case "layout":
            myobj = app.CreateLayout( "linear" );
            break;
        case "BluetoothList":
            myobj = app.CreateBluetoothList();
            break;
        case "BluetoothSerial":
            myobj = app.CreateBluetoothSerial();
            break;
        case "Button":
            myobj = app.CreateButton("x");
            break;
        case "CameraView":
            myobj = app.CreateCameraView( 0.8,0.8 );
            break;
        case "CheckBox":
            myobj = app.CreateCheckBox( "x" );
            break;
        case "Crypt":
            myobj = app.CreateCrypt();
            break;
        case "Database":
            myobj = app.OpenDatabase("MyData");
            break;
        case "Dialog":
            myobj = app.CreateDialog("");
            break;
        case "Email":
            myobj = app.CreateEmail( "mymail@gmail.com", "MyPass" );
            break;
        case "Image":
            myobj = app.CreateImage( "/Sys/Img/Hello.png" );
            break;
        case "List":
            myobj = ListView;
            break;
        case "Locator":
```

```
    myobj = app.CreateLocator( "GPS,Network" );
    break;
case "Notification":
    myobj = app.CreateNotification() ;
    break;
case "NXT":
    myobj = app.CreateNxt();
    break;
case "NxtRemote":
    myobj = app.CreateNxtRemote();
    break;
case "Odroid":
    myobj = app.Odroid( "a", "b", "c" );
    break;
case "SeekBar":
    myobj = app.CreateSeekBar( "x" );
    break;
case "Sensor":
    myobj = app.CreateSensor( "Accelerometer" );
    break;
case "SMS":
    myobj = app.CreateSMS();
    break;
case "Spinner":
    myobj= app.CreateSpinner("1,2");
    break;
case "Tabs":
    myobj = app.CreateTabs( "tab1,tab2" );
    break;
case "Text":
    myobj = main_header_txt;
    break;
case "TextEdit":
    myobj = FunctionView;
    break;
case "ToggleButton":
    myobj = app.CreateToggle( "x" );
    break;
case "WebView":
    myobj = app.CreateWebView( 0.8,0.8 );
    break;
case "WebServer":
    myobj = app.CreateWebServer( 8080, "Upload,ListDir" );
    break;
case "YesNoDialog":
    myobj = app.CreateYesNoDialog("");
    break;
case("Manually entered code"):
    showEval();
    return;
default:
```

```

        myobj=app;
        lastMenu = "app";
    }
    prepareFunctionList();
    showOverview();
    main_header_txt.SetText( lastMenu );
} //menuView_OnTouch

function prepareFunctionList(){
    /*** prepare list view ***/

    var objFunctionList;
    objFunctionList = [];
    var filter = edtFilter.GetText().trim().toLowerCase()
    for (var Key in myobj) {
        var keyword= Key.toLowerCase()
        if ((filter == "") || (keyword.indexOf(filter) > -1)){
            if (myobj.hasOwnProperty(Key) && (typeof myobj[Key] ===
'function')) {
                objFunctionList.push(Key);
            }
        }
    }
    objFunctionList.sort();
    ListView.SetList( objFunctionList.join(',') + ",")
    if(filter != "" ) filter = "Filtered [" + filter + "]\n";
    pasteable = filter + lastMenu + ".";
    var copyDelim = "\n" + lastMenu + ".";
    pasteable += objFunctionList.join( copyDelim )
    docfile = docfold + "/" + lastMenu + ".txt";
} //prepareFunctionList
function showMenuView(){
    ListView.SetVisibility('hide');
    FunctionView.SetVisibility('hide');
    menuView.SetVisibility('show');
    main_header_txt.SetText( "Choose Control" );
} //showMenuView

function showAppFunction (Name) {
    FunctionView.SetText(String(myobj[Name]));
    main_header_txt.SetText( lastMenu + "." + Name );
    ListView.SetVisibility('hide');
    menuView.SetVisibility('hide');
    FunctionView.SetVisibility('show');
} //showAppFunction

function showOverview () {
    ListView.SetVisibility('show');
    FunctionView.SetVisibility('hide');
    menuView.SetVisibility('hide');
    main_header_txt.SetText( lastMenu );
}

```

```
//showOverview
function edtFilterOnChange(){
    if( ListView.GetVisibility() == "Show" ){
        prepareFunctionList();
    }
}
//edtFilterOnChange

function OnBack(){
    if (menuView.GetVisibility() == "Show") {
        var pasteOnExit = app.CreateYesNoDialog( "Copy to clipboard and
create text file?" );
        pasteOnExit.SetOnTouch( pasteOnExit_OnTouch );
    }
    else if( ListView.GetVisibility() == "Show" ) showMenuView();
    else showOverview();
}
//OnBack

function pasteOnExit_OnTouch( result ){
    if( result=="Yes" ){
        app.SetClipboardText( pasteable );
        if( ! app.FolderExists( docfold ) )
            app.MakeFolder ( docfold );
        app.WriteFile( docfile, pasteable );
    }
    app.Exit();
}
//pasteOnExit
function showEval(){
    dlg = app.CreateDialog("eval - use with care");
    var layDlg = app.CreateLayout( "linear", "vertical,fillxy,left" );
    layDlg.SetPadding( 0.02, 0, 0.02, 0.02 );
    edtDlg = app.CreateTextEdit('app.CreateText( "Hello
)", -1, -1, 'NoSpell');
    edtDlg.SetCursorPos(edtDlg.GetText().length);
    layDlg.AddChild(edtDlg);
    dlg.AddLayout( layDlg );
    var horiz = app.CreateLayout("linear", "horizontal,fillxy");
    horiz.SetPadding( 0.02, 0.02, 0.02, 0 );
    var btn = app.CreateButton("Ok", 0.2);
    btn.SetOnTouch(dlg_ok);
    horiz.AddChild(btn);
    btn = app.CreateButton("Cancel", 0.2);
    btn.SetOnTouch(dlg_canc);
    horiz.AddChild(btn);
    layDlg.AddChild(horiz);
    dlg.Show();
}
//showEval
function dlg_ok(){
    app.HideKeyboard();
    dlg.Dismiss();
    myobj = eval(edtDlg.GetText());
    if ((myobj != null) && (myobj.hasOwnProperty("GetType"))){
```

```
    lastMenu = myobj.GetType();
    app.WriteFile(evalsfile, edtDlg.GetText()+"\n", "append");
}
else{
    lastMenu = "Eval";
}
main_header_txt.SetText( lastMenu );
prepareFunctionList();
showOverview();
} //dlg_ok
function dlg_canc(){
    app.HideKeyboard();
    dlg.Dismiss();
} //dlg_canc
```

I have placed an spk of this app at <http://hudl.sgarman.net/public/spk/SJGinspect.spk> which should create a project in the DroidScript IDE for you.

And for the sake of completeness, there is an apk at <http://droidscript.sgarman.net/apk/SJGinspect.apk> which you could download and install if your device is set up to allow apks that don't come from Google Play.

From:

<http://wiki.droidscript.me.uk/> - **DroidScript wiki**

Permanent link:

http://wiki.droidscript.me.uk/doku.php?id=sample_code:introspector

Last update: **2015/03/28 21:06**

