

Layout Children

DroidScript's [Layout](#) comes complete with methods `ChildToFront` and `GetChildOrder` but some people seem to think it is missing a couple of methods, so I offer these functions in the hope that they may be useful.

ChildToBack

[childToBack.js](#)

```
function childToBack(layout, obj){
    if( layout.GetChildOrder( obj ) > 0 ){
        layout.RemoveChild(obj);
        layout.AddChild(obj,0);
    }
}
```

setChildOrder

[setChildOrder.js](#)

```
function setChildOrder(layout, obj, order){
    var tst = layout.GetChildOrder( obj );
    if ((tst >= 0) && (tst != order)){
        layout.RemoveChild(obj);
        layout.AddChild(obj,order);
    }
}
```

getChildren

[getChildren.js](#)

```
function getChildren(layout){
    //takes a layout as an argument
    var children = [];
    var obj, order;
    //loop through all controls
    for (var id in _map) {
        obj = _map[id]
```

```
//check if control is on our layout
order = layt.GetChildOrder( obj );
if( order > -1 ) {
    //got one!
    //zorder property wll persist until getChildren called again
    //or manually overwritten by coder
    obj.zorder = order;
    children.push( obj );
}
}
children.sort(function (a, b) { return a.zorder - b.zorder});
return children;
}
```

clearLayout

[clearlayout.js](#)

```
function clearLayout(layt)
{
    //takes a layout as an argument
    var allobjs = app.GetObjects();
    var obj, order;
    //loop through all controls
    for (var id in allobjs)
    {
        obj = allobjs[id]
        //check if control is on our layout
        order = layt.GetChildOrder( obj );
        if( order > -1 )
        {
            //got one!
            layt.DestroyChild( obj );
        }
    }
}
```

Sample app

This sample shows how to use the childToBack function

[toback.js](#)

```
//Called when application is started.
```

```
function OnStart()
{
    app.SetOrientation("Portrait");
    //Create a layout with objects vertically centered.
    lay = app.CreateLayout( "Linear", "VCenter,FillXY" );

    //Create an absolute layout so we can position objects.
    layAbs = app.CreateLayout( "Absolute" );
    lay.AddChild( layAbs );

    //Create first image.
    img1 = app.CreateImage( null, 0.5, 0.3 );
    img1.SetColor("#66000088");
    img1.SetOnTouchDown(img_OnTouch);
    layAbs.AddChild( img1 );

    //Create second image.
    img2 = app.CreateImage( null, 0.5, 0.3 );
    img2.SetColor("#66880000");
    img2.SetPosition( 0.1,0.1 );
    img2.SetOnTouchDown(img_OnTouch);
    layAbs.AddChild( img2 );

    //Create third image in middle
    img3 = app.CreateImage( null, 0.5, 0.3 );
    img3.SetColor("#66008800");
    img3.SetPosition( 0.2,0.2 );
    img3.SetOnTouchDown(img_OnTouch);
    layAbs.AddChild( img3,1 );

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

//Called when user touches image
function img_OnTouch(){
    childToBack(layAbs, this)
}

function childToBack(layt, obj){
    if( layt.GetChildOrder( obj ) > 0 ){
        layt.RemoveChild(obj);
        layt.AddChild(obj,0);
    }
}
}
```

From:

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

Permanent link:

http://www.wiki.droidsript.me.uk/doku.php?id=sample_code:layout_children

Last update: **2014/12/28 18:20**

