

# Image Grid Plugin

*(Information and examples taken from the Plugin documentation)*

## Note:

**This documentation is just for your information. You can just work with the samples after you have downloaded and payed for the Plugin.**

## Description

The ImageGrid plugin provides a high performance, multi-threaded thumbnail viewing grid, with in-built memory and disk caching.

This control is suitable for providing 'gallery style' image browsing experiences to the user. It can display images from both local and remote sources.

To use the ImageGrid, you must first load the plugin at the top of your script using the **LoadPlugin** method like this:

```
app.LoadPlugin( "ImageGrid" );
```

Then you can create an instance of the image grid when you need it using the **CreateImageGrid** method:

```
grid = app.CreateImageGrid( list, width, height, cols, rows, cacheSize, options );
```

The **list** parameter should be an array or comma separated list of image file names. The **width** and **height** parameters set the size of the grid as a fraction of the screen width and height.

The **cols** parameter sets the number of grid columns required and the **rows** parameter sets the number of grid rows shown per screen.

The optional **cacheSize** parameter sets the number of megabytes reserved on disk to cache images, which vastly improves performance when re-displaying the same images (it defaults to 50).

The **options** parameter may contain the following values that determine the way that the images are scaled within each grid cell: **crop**, **fit**, **fill**, **shrink**.

---

You may want use the **SetSpacing** method to set the size of the border between each image and the **SetBackColor** method to change the background color (for example to black rather than the default white).

The following methods are provided by the ImageGrid object:

- AddItem( name )
- GetLength()

- InsertItem( index, name )
- RemoveAll()
- RemoveItemByIndex( index )
- SetList( list, delim )
- SetOnLongTouch( callback )
- SetOnTouch( callback )
- SetSpacing( width, height )

The following general methods are also available:

- Focus()
- GetAbsWidth()
- GetAbsHeight()
- GetWidth()
- GetHeight()
- GetPosition()
- GetType()
- GetVisibility()
- SetVisibility( visibility )
- SetPadding( left, top, right, bottom )
- SetMargins( left, top, right, bottom )
- SetBackground( imageFile, options )
- SetBackColor( colorCode )
- SetBackGradient( color1, color2, color3 )
- SetBackGradientRadial( x, y, r, color1, color2, color3 )
- SetPosition( left, top, width, height )
- SetSize( width, height )
- SetScale( width, height )

---

## Example - Display and select

```
app.LoadPlugin( "ImageGrid" );

function OnStart()
{
    lay = app.CreateLayout( "Linear", "VCenter,FillXY" );

    var files = app.ListFolder( "/sdcard/DCIM/Camera", ".jpg", 1000,
"FullPath" );

    grid = app.CreateImageGrid( files, 1,1, 3,5 );
    grid.SetOnTouch( grid_OnTouch );
    grid.SetOnLongTouch( grid_OnLongTouch );
    lay.AddChild( grid );

    app.AddLayout( lay );
}

function grid_OnTouch( fileName )
```

```
{
  alert( fileName );
}

function grid_OnLongTouch( fileName )
{
  alert( fileName );
}
```

## Example - Remove image

```
app.LoadPlugin( "ImageGrid" );

function onStart()
{
  lay = app.CreateLayout( "Linear", "VCenter,FillXY" );

  var files = app.ListFolder( "/sdcard/DCIM/Camera", ".jpg", 1000,
  "FullPath" );

  grid = app.CreateImageGrid( files, 1,0.7, 3,5 );
  lay.AddChild( grid );

  btn = app.CreateButton( "Remove", 0.3 );
  btn.SetOnTouch( btnRemove_OnTouch );
  lay.AddChild( btn );
  app.AddLayout( lay );
}

function btnRemove_OnTouch()
{
  grid.RemoveItemByIndex( 0 );
}
```

## Example - Album art

```
app.LoadPlugin( "ImageGrid" );

function onStart()
{
  lay = app.CreateLayout( "Linear", "VCenter,FillXY" );
  grid = app.CreateImageGrid( "", 0.9,0.9, 1,2 );
  lay.AddChild( grid );
  app.AddLayout( lay );

  media = app.CreateMediaStore();
  media.SetOnAlbumsResult( media_OnAlbumsResult );
  media.QueryAlbums( "", "", "external" );
}
```

```
}  
  
function media_OnAlbumsResult( result )  
{  
    s = "";  
    for( var i=0; i<result.length; i++ )  
    {  
        if( result[i].albumArt!="null" )  
            s += (i>0?", ":"") + result[i].albumArt;  
    }  
    grid.SetList( s, ", " );  
}
```

## Example - Assets

```
app.LoadPlugin( "ImageGrid" );  
  
function OnStart()  
{  
    lay = app.CreateLayout( "Linear", "VCenter,FillXY" );  
  
    var files = app.ListFolder( "/Assets/Img", ".png", 0, "FullPath" );  
  
    grid = app.CreateImageGrid( files, 1,1, 5,7 );  
    lay.AddChild( grid );  
  
    app.AddLayout( lay );  
}
```

## Example - Remote

```
app.LoadPlugin( "ImageGrid" );  
  
function OnStart()  
{  
    lay = app.CreateLayout( "Linear", "VCenter,FillXY" );  
  
    var files =  
    [  
        "http://androidscript.org/Plugins/img/BarcodeReader.png",  
        "http://androidscript.org/Plugins/img/GoProController.png",  
        "http://androidscript.org/Plugins/img/Moga.png",  
        "http://androidscript.org/Plugins/img/NXT.png",  
        "http://androidscript.org/Plugins/img/PluginApk.png",  
        "http://androidscript.org/Plugins/img/SpheroBall.png",  
    ];  
  
    grid = app.CreateImageGrid( files, 1,1, 3,6, -1, "fit" );  
}
```

```
lay.AddChild( grid );  
app.AddLayout( lay );  
}
```

From:

<http://sgarman.net/wiki/> - **DroidScript** wiki

Permanent link:

[http://sgarman.net/wiki/doku.php?id=plugins:image\\_grid](http://sgarman.net/wiki/doku.php?id=plugins:image_grid)

Last update: **2015/11/02 10:19**

