

## Making of "Layers Cleaner"

This would be a very short MAXScript tutorial but I hope is helpful.

-- their is the finished code:

```
Macroscript LayersCleaner category:"Tools" internalCategory:"Tools"
tooltip:"Clean up Empty Layers" buttonText:"Layers Cleaner"
(
    (LayerManager.getLayer 0).current = true
    for i = (LayerManager.count-1) to 1 by -1 do
        LayerManager.deleteLayerByName (LayerManager.getlayer i).name
)
```

Simple task, right? Well, what we need to know to write such a script? Simply, 2 thinks:

1. Use backward loop then deleting a items from arrays.
2. Search for "LayerManager" into MAXScript Reference.

Yes, that's all.

So, what means the first one?

If you're familiar with this, just skip this part.

Let's to say you have an array with 3 items and you intended to delete them with a standard (forward) loop like this:

```
exampleArray = #("a", "b", "c")
for itm = 1 to exampleArray.count do deleteItem exampleArray itm
```

Well, you'll see error message:

```
-- Error occurred in itm loop;
-- Frame:
-- itm: 3
-- Runtime error: array index out of bounds: 3
```

That means the error appear in a third iteration. Also if you tape the name of the array (exampleArray) to the Listener, you'll see that it return #("b"). So, lets see what's happen...

In the first iteration the script delete item 1, i.e. "a", and now "b" is no more item 2 but become item 1, and "c" become item 2, because **the size of the array decrease**. Next, in the second iteration the script delete item 2, i.e. delete "c" ('cause it's a item 2). And what's happen in the third iteration is that the script looking to delete item 3, but the array now has only one item. So, "array index out of bounds" is deserved runtime error :-)

Let's make now correct (for deleting items) backward loop:

```
exampleArray = #("a", "b", "c")
for itm = exampleArray.count to 1 by -1 do
    deleteItem exampleArray itm
```

Well, nothing fancy, right? It's just routine.

Now, maybe you're wonder what I mean with: Search for "LayerManager" in the help? Well, this may sound trivial, everyone will check how to access the layers and how to delete them, of course, but ... seems to me not everyone make efforts to understand the notes inside the MAXScript Reference.

Ok, let's see the description for **LayerManager.deleteLayerByName**:

Deletes the layer with the given name. Returns true if layer deleted, false otherwise. **A layer can only be deleted if no nodes are in the layer and the layer is not the currently active layer.**

What's this means?

1. Not empty layers are **protected** from deletion.
  2. The currently active layer is **protected** as well.
- Also the default layer **cannot** be deleted.

And the first step in the script will be to set the default layer as active layer and **not** to protect it, but to **unlock the rest layers for deletion**.

Also the layer index is 0-based, and the default layer is always 0. So...

```
LayerManager.getLayer 0 -- get default layer
(LayerManager.getLayer 0).current = true -- set it as current
```

Because the layer index is 0-based, our loop will start with (LayerManager.count-1), so there is the whole script:

```
(
  (LayerManager.getLayer 0).current = true
  for i = (LayerManager.count-1) to 1 by -1 do
    LayerManager.deleteLayerByName (LayerManager.getlayer i).name
)
```

Did you wonder why I not check for empty layers and just call `deleteLayerByName` on all? I think so, If you was read **carefully** the description of `LayerManager.deleteLayerByName`, you'll never ask such a question.

Let's repeat then: Not empty layers are **protected** from deletion.

That means **it's safe to try deleteLayerByName on everything in the list!**

Or in other words, perform a check for empty layers is vain overweight for the code.

Due to the `deleteLayerByName` return true on success we can easy implicate counter:

```
(
  clean = 0 -- variable to count deleted layers
  (LayerManager.getLayer 0).current = true
  for i = (LayerManager.count-1) to 1 by -1 do
    if (LayerManager.deleteLayerByName (LayerManager.getlayer i).name) do
      clean += 1
      format "% layers removed\n" clean -- format the result to the Listener
)
```

...but this is just a cosmetics for me.

*-- so, their is the finished code:*

**Macroscript** LayersCleaner category:"Tools" internalCategory:"Tools"  
tooltip:"Clean up Empty Layers" buttontext:"Layers Cleaner"

```
(
  (LayerManager.getLayer 0).current = true
  for i = (LayerManager.count-1) to 1 by -1 do
    LayerManager.deleteLayerByName (LayerManager.getlayer i).name
)
```

**About the script compatibility:**

`deleteLayerByName` has been added to the LayerManager Core Interface in 3ds Max 5.1

Cheers,

P. Karabakalov (Anubis)  
[anubiss@abv.bg](mailto:anubiss@abv.bg)