

MEL Scripting Class 1 Adrian Herbez - purplestatic@yahoo.com

Welcome to the wide wonderful world of MEL scripting! Congratulations on deciding to expand your skills into the realm of programming.

MEL (Maya Embedded Language), with a little bit of effort, will allow you to create your own custom tools and utilities, expanding Maya's range and allowing almost anything you could possibly want to have happen to happen. But it is programming, and therefore requires a slightly different approach than you may be used to. Stick with it, though, and you'll find yourself doing things you may never have thought possible.

What is programming?

At the most fundamental level, computers deal with one thing and one thing only- ones and zeros. In order to give a computer meaningful instructions, those instructions must end up as combinations of 1s and 0s. Operating at that level is **very** difficult and almost never done.

Through binary notation, strings of 1s and 0s can be used to represent numbers. Numbers can be used to represent commands. Being able to issue commands instead of writing 111000 is better, but still a bit obtuse and hard. Operating at that level is what happens when one engages in Assembly Language programming. Assembly languages are one step up from binary numbers, but are still a bit hard to use. The benefit of programming in assembly is that you're operating very "close to the metal", and the resulting program can therefore be quite fast.

But assembly language is still pretty tricky, so there are so-called "high-level languages" such as Java and C++. They offer a syntax much closer to natural language, and are therefore easier to use while still being quite flexible. Code written in them must be converted to machine code via a process called "compiling".

On top of high-level languages are so-called "scripting languages". The main difference between a scripting language and a high-level language is that a scripting language is intended for a specific environment. MEL code, for example, will only ever run from within Maya. It is possible, however, to create stand-alone applications using C++ that interface with Maya by using Maya's API (Application Programming Interface)

What is MEL?

As mentioned above, MEL is a scripting language, which means it's usage is confined to within Maya. What makes it so exciting is that in many ways, Alias created MEL, and then used MEL to make everything that we regard as Maya. Whenever you take action within Maya, you are actually executing lines of MEL code to achieve your task. That ends up meaning that any tool you've ever used in Maya is accessible to you for decomposition and tinkering.

Your primary tool in learning MEL is the script editor, available by going to Window->General Editors->Script Editor. This will bring up a window split into two vertical parts. The script editor can be thought of as a two-way communication device between you and Maya. In the lower half, you can give Maya MEL commands to execute. The top half, on the other hand, is one of the most vital tools for learning about MEL, as every time you take action in Maya, Maya will output the MEL command or commands corresponding to your activity there, as well as any additional commentary it has as to the output of those commands.

