



Assignment - API Overview

Topics Covered

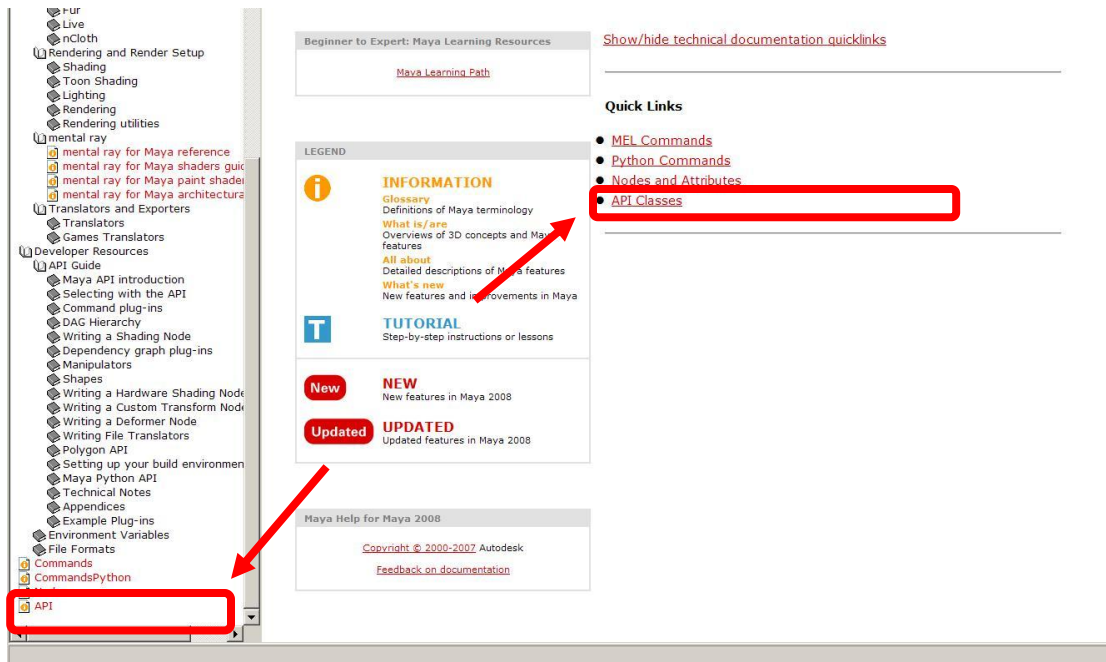
- How to find API Docs
- Build a very basic plug-in: HelloWorldCmd
- Working with the structure of plug-ins: SineNode

Assignments

- For every Plug-in, we will provide an “Exercise” folder and a “Solution” folder. In addition we will provide a C++ and Python version of the Exercise and Solutions of the assignments. Solution folder includes the complete code for you to finish the Plug-in, it is for you to reference when you are stuck at problems when adding code into the “Exercise” folder. In the “Exercise” folder, all the code you need to finish is specified with comments “##- TODO:” you need to search for all the “TODO” comments and add your code there.

Background

- **How to find API Docs**
 1. Open Maya, in the File Menu, click on Help > Maya Help, or use the shortcut F1 to open the help documentation.
 2. At the bottom of the left pane, select ‘API’. Or click on the “API Classes” links on the right upper corner under the “Quick Links” section.



- **Configuring Your Maya Plug-in Path**

1. Browse to the folder:

(Windows®)

- drive:\Documents and Settings\username\My Documents\maya\version
- drive:\Documents and Settings\username\My Documents\maya

(Mac OS X)

- /Users/username/Library/Preferences/Autodesk/maya/version
- /Users/username/Library/Preferences/Autodesk/maya

(Linux®)

- ~/maya/version
- ~/maya

2. Create a text file named: Maya.env

3. In Maya.env, insert the line, for example:

MAYA_PLUG_IN_PATH= \$SHARED_MAYA_DIR/plugin-ins

- **Tips: Safely reloading a plug-in without restarting Maya**

1. Clear the scene:

- File > New Scene
- file -f -new

2. Clear the undo queue:

- flushUndo
- 3. Unload the plug-in:
 - unloadPlugin helloWorld.py
- 4. Reload the plug-in:
 - loadPlugin helloWorld.py

helloWorldCmd Plug-in

- **Topics Covered**
Implement a simple custom python command “spHelloWorld”. It demonstrates the skeleton code implementation for a python command plug-in with MPxCommand.
- **Overview**
In this Plug-in, we will create a custom command “spHelloWorld” in python, which prints out “Hello World!” in the script editor window.
- **Exercises**
 1. Go to “helloWorld python Plug-in\Exercise” folder, open helloWorldCmd.py, the skeleton of the command is already there.
 2. Add corresponding code into the skeleton to make it work.

Relevant Classes and commands:

```
import sys
import maya.OpenMaya as OpenMaya
import maya.OpenMayaMPx as OpenMayaMPx
def __init__(self)
def doIt(self,argList)
OpenMayaMPx.asMPxPtr()
```
- **Result**
In maya script editor, python tab, execute the following line:

```
import maya
maya.cmds.loadPlugin("helloWorldCmd.py")
maya.cmds.spHelloWorld()
```

You will see “Hello World!” printed out in the script editor

sineNode Plug-in

- **Topics Covered**
Implement a custom python node “spSineNode”. It demonstrates the skeleton code implementation for a python node plug-in with MPxNode.
- **Overview**

In this Plug-in, we will create a custom node “spSineNode” in python, this node has an input attribute “input” and an output attribute ”output”, the “output” is a sine calculation of “input”.

- **Exercises**

1. Go to “sineNode python Plug-in\Exercise” folder, open sineNode.py, the skeleton of the node is already there.
2. Add corresponding code into the skeleton to make it work.

Relevant Classes and commands:

```
import sys,math
import maya.OpenMaya as OpenMaya
import maya.OpenMayaMPx as OpenMayaMPx
OpenMaya.MObject()
OpenMayaMPx.asMPxPtr()
def compute(self,plug,dataBlock)
```

- **Result**

In Maya, load the plug-in “sineNode.py”, execute the following script:

```
polySphere;
createNode spSineNode -n sine1;
connectAttr time1.outTime sine1.input;
connectAttr time1.outTime pSphere1.translateX;
connectAttr sine1.output pSphere1.translateY;
```

You will find a polySphere created and move along X axis as time goes on.