

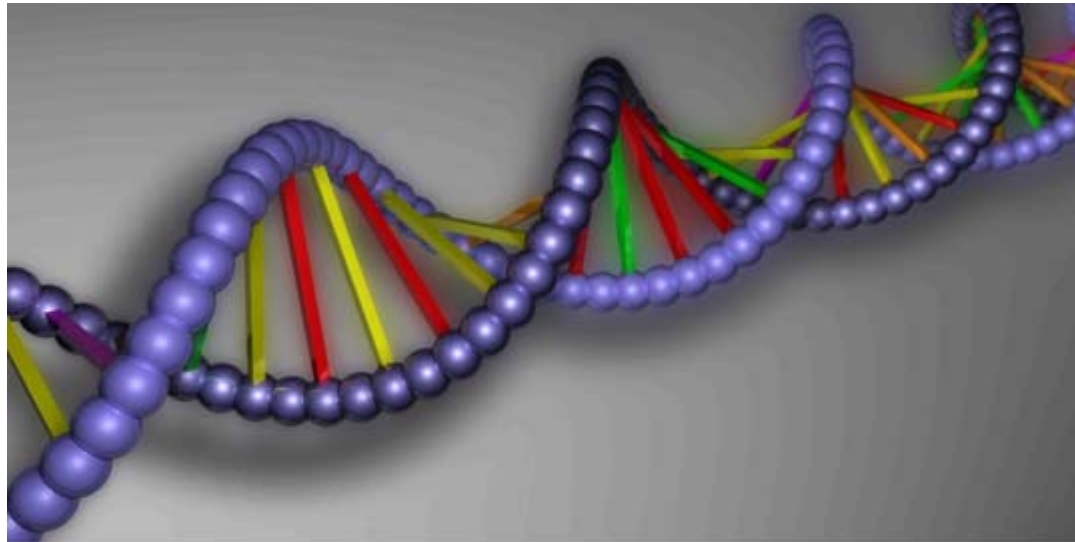


Utilisation de Mel dans Maya

G. Gesquière

- Extrait de tutoriel Maya
(http://download.autodesk.com/us/maya/maya2013_getting_started/index.html)
- Livre : Maya Python for Games and Film (A. Mechtley, R. Trowbridge)
- Langage MEL introduction, ESIEE Lilian Buzer
- Conversational MEL Part 1, 2, 3, Nimble Studio

Introduction



Procedural geometry created with MEL scripting

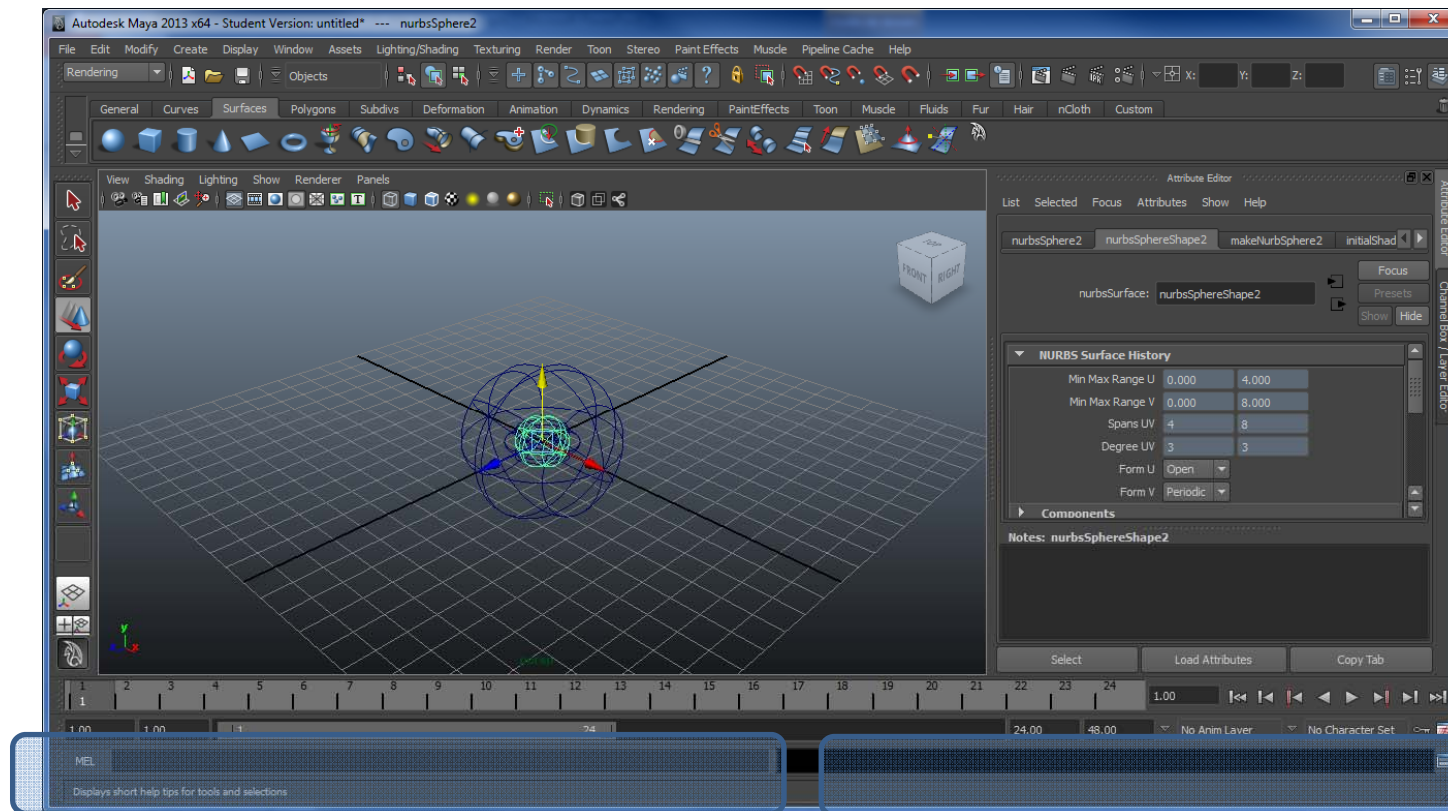
http://download.autodesk.com/us/maya/maya2013_getting_started/index.html

Qu'est ce que MEL

- MEL
 - “Maya Embedded Language”
 - “an interpreted scripting language that interfaces directly with Maya’s core libraries”.
- MEL is an easy way to take your art to the next level. Everything you do, and almost everything you see, in Maya is driven by MEL.
 - Menu items, check boxes, sliders, heads up display, outliner, attribute editor, etc...
 - All of it is created using MEL. When you check a check box, slide a slider, manipulate a manipulator, or do pretty much anything in Maya, a little bit of MEL is executed.
- MEL is a powerful tool for doing such things as:
 - Automating repetitive tasks and reducing your ‘click-count’ - with MEL you can write a script that performs far more clicks and drags than you could do in your typical 100-hour work week!
 - Building custom UI to suits **your** needs - if you’ve complained about the apparent stupidity of some part of Maya’s UI, then MEL is for you!
 - Leverage the power of modern computers to perform advanced math and invent the next great effect or tool!
 - And much more!

Ligne de Commande et Editeur de Script

- To use MEL you will need to type out commands and pass them off to Maya

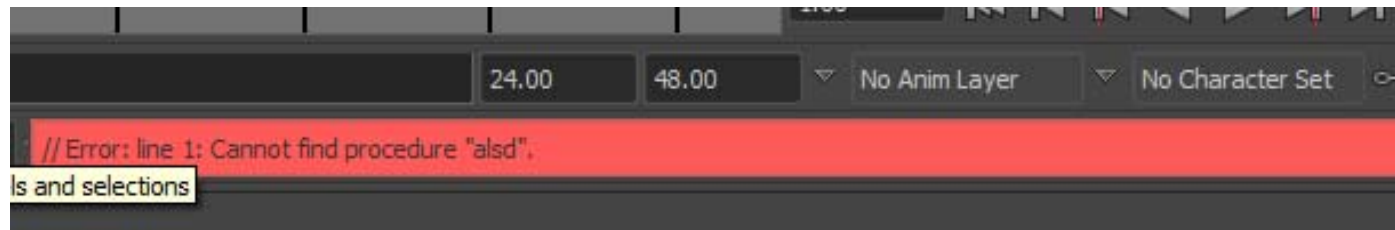
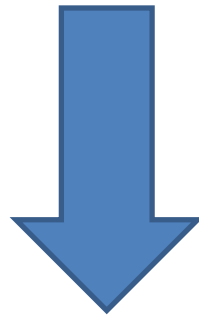
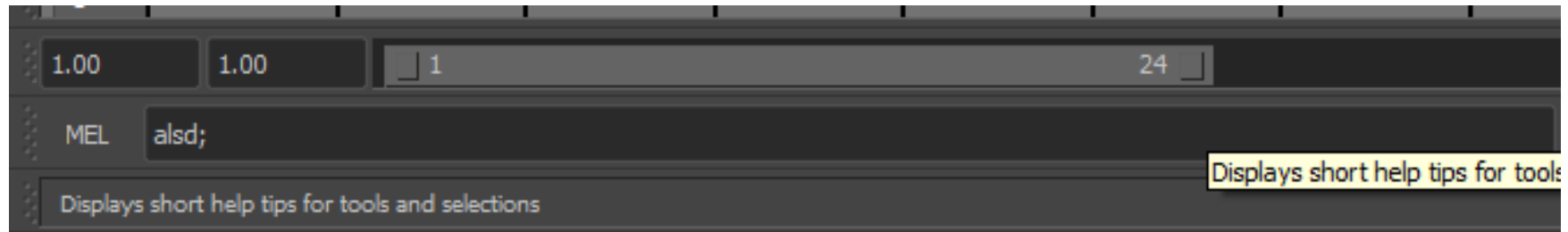


Command Line

Command Feedback

Ligne de Commande et Editeur de Script

- Saisir un premier texte



Le texte ne fait pas référence à une commande connue

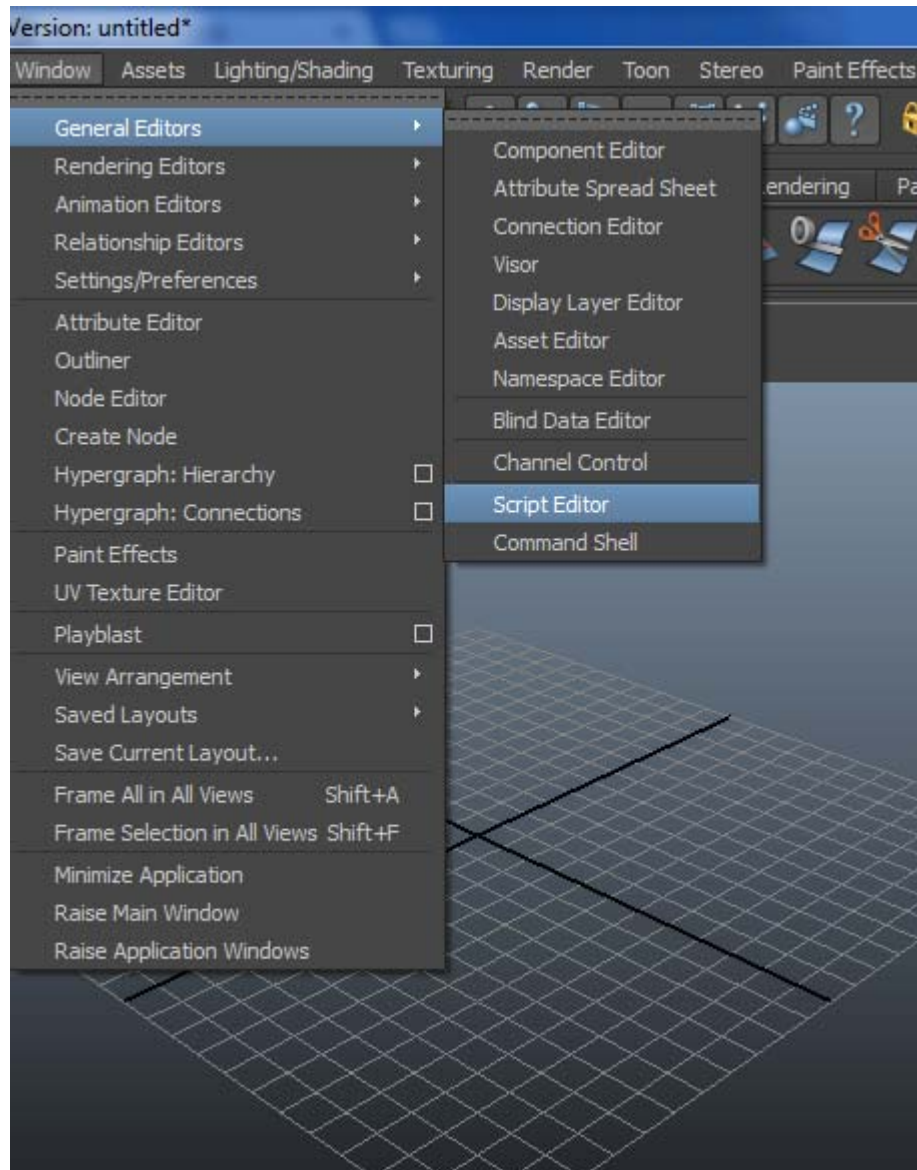
Ligne de Commande et Editeur de Script

- L'ensemble des commandes connues sont référencées dans :
 - Help > Mel Command Reference

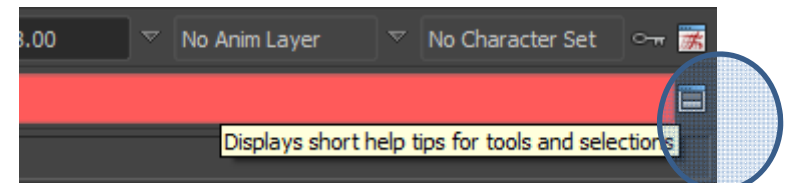
The screenshot shows a web browser window with the following details:

- Browser tabs: "A Maya Getting Started: Ent...", "A Maya commands x"
- Address bar: "download.autodesk.com/global/docs/maya2013/en_us/Commands/index.html"
- Navigation bar: "M Détails du télécharg...", "about:blank", "Admin-Cours", "Perso", "Nouvel onglet", "Recherche", "Paramètres"
- Section: "All commands" with a navigation bar "A B C D E F G H I J K L M N O P Q R S T U V W X"
- Section: "By substring(s)" with an input field.
- Section: "By category" with sub-sections:
 - General**: [Attributes](#), [Display](#), [Selection](#), [Contexts](#)
 - Language**: [Math](#), [Strings](#), [Array](#), [Scripting](#)
 - Modeling**: [Polygons](#), [NURBS](#), [Curves](#), [SubDs](#)
 - Animation**: [Deformation](#), [Skinning](#), [Constraints](#), [IK](#), [MoCap](#)
 - Rendering**: [Camera](#), [Layers](#), [Lights](#)
- Section: "A" listing commands:
 - [aaf2fcp](#)
 - [about](#)
 - [abs](#)
 - [addAttr](#)
 - [addAttributeEditorNodeHelp](#) M
 - [addCustomPrefsTab](#) M
 - [addDynamic](#)
 - [addExtension](#)
 - [addMetadata](#)
 - [addNewShelfTab](#) M
 - [alignCtx](#)
 - [alignCurve](#)
 - [alignSurface](#)
 - [allNodeTypes](#)
 - [allViewFit](#) M
 - [ambientLight](#)
 - [angle](#)
 - [angleBetween](#)
 - [animCurveEditor](#)
 - [animDisplay](#)
 - [arcLengthDimension](#)
 - [arrayMapper](#)
 - [art3dPaintCtx](#)
 - [artAttrCtx](#)
 - [artAttrPaintVertexCtx](#)
 - [artAttrSkinPaintCtx](#)
 - [artAttrTool](#)
 - [artBuildPaintMenu](#)
 - [artFluidAttrCtx](#)
 - [artPuttyCtx](#)

Ouverture de l'éditeur de script

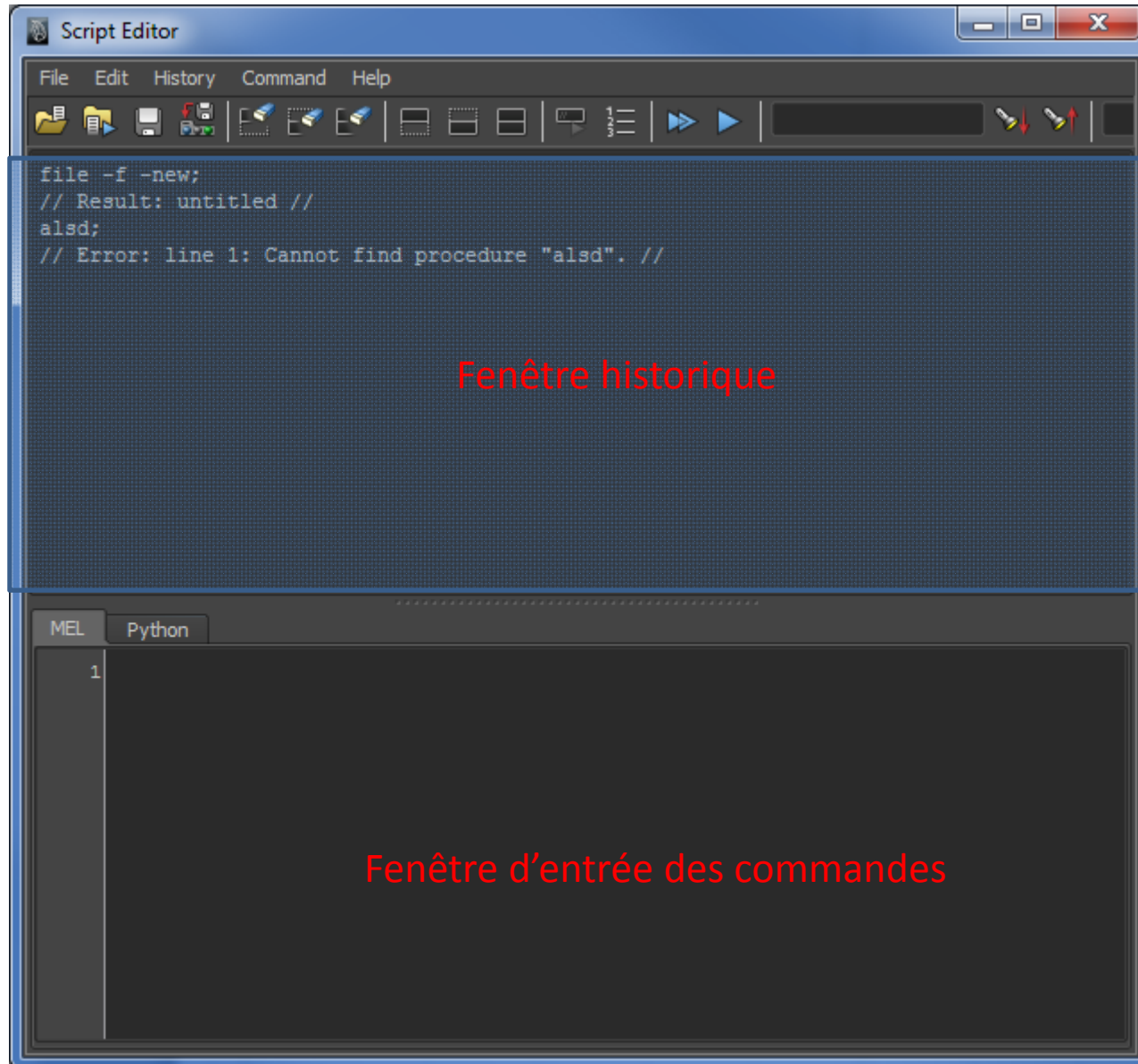


Menu



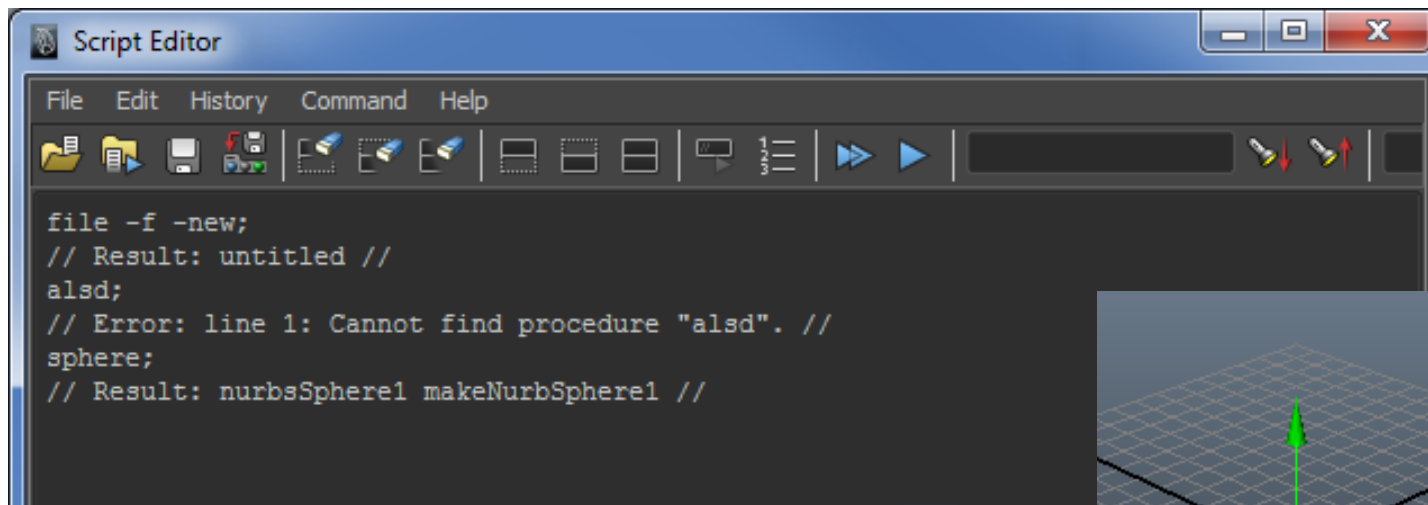
Icones

Editeur de script



Mes premières commandes

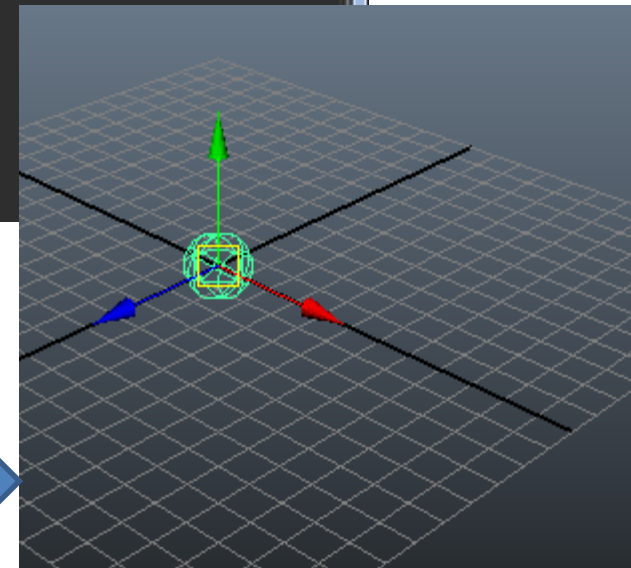
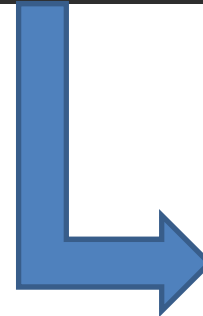
- Taper « sphere; » dans la ligne de commande ou dans l'éditeur de script



```
Script Editor
File Edit History Command Help
file -f -new;
// Result: untitled //
alsd;
// Error: line 1: Cannot find procedure "alsd". //
sphere;
// Result: nurbsSphere1 makeNurbSphere1 //
```

Attention :

- Les entrées MEL et Python doivent respecter la « casse »
- Pour exécuter une commande, taper « Entrée » sur la partie numérique du clavier ou Ctrl+ entrée

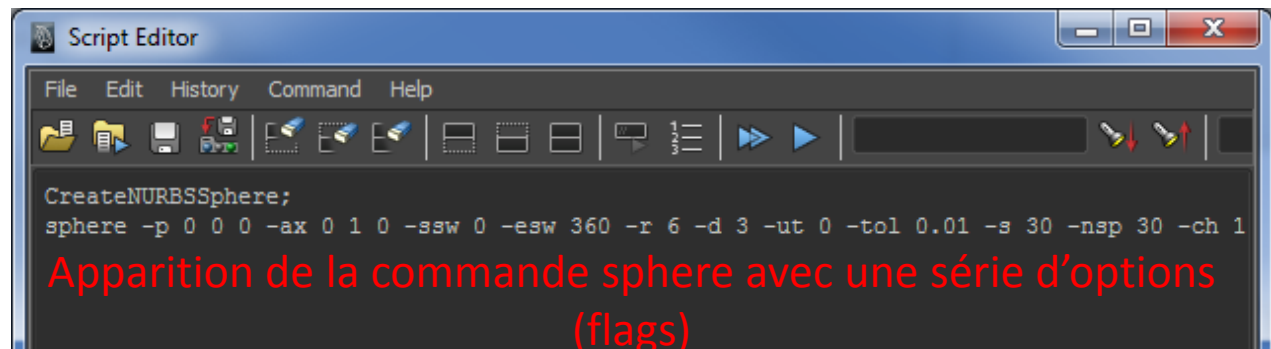


Observer l'historique des commandes

- Dans l'éditeur de script, appuyer sur le bouton « Clear History »



- Créer une sphère directement sous Maya



Maya Getting Started: Obs x sphere command x

download.autodesk.com/global/docs/maya2013/en_us/Commands/

Détails du télécharg... about:blank Admin-Cours Perso Nouvel onglet Recherche Paramètres

command (MEL) [Python version](#)

sphere

In categories: [Modeling](#), [NURBS](#)

[Show frames](#)

Go to: [Synopsis](#). [Return value](#). [Related](#). [Flags](#). [MEL examples](#).

Synopsis

```
sphere [-axis linear linear linear] [-caching boolean] [-constructionHistory boolean] [-degree int] [-endSweep angle] [-heightRatio float] [-name string] [-nodeState int] [-object boolean] [-pivot linear linear linear] [-polygon int] [-radius linear] [-sections int] [-spans int] [-startSweep angle] [-tolerance linear] [-useTolerance boolean]
```

sphere is undoable, queryable, and editable.

The sphere command creates a new sphere. The number of spans in the in each direction of the sphere is determined by the useTolerance attribute. If -ut is true then the -tolerance attribute will be used. If -ut is false then the -sections attribute will be used.

Return value

string[] Object name and node name

In query mode, return type is based on queried flag.

Related

[cone](#), [cylinder](#), [nurbsCube](#), [nurbsPlane](#), [torus](#)

Flags

[axis](#), [caching](#), [constructionHistory](#), [degree](#), [endSweep](#), [heightRatio](#), [name](#), [nodeState](#), [object](#), [pivot](#), [polygon](#), [radius](#), [sections](#), [spans](#), [startSweep](#), [tolerance](#), [useTolerance](#)

Flags

[axis](#), [caching](#), [constructionHistory](#), [degree](#), [endSweep](#), [heightRatio](#), [name](#), [nodeState](#), [object](#), [pivot](#), [polygon](#), [radius](#), [sections](#), [spans](#), [startSweep](#), [tolerance](#), [useTolerance](#)

Long name (short name)	Argument types	Properties
-radius (-r) The radius of the object <i>Default: 1.0</i>	linear	C Q E
-startSweep (-ssw) The angle at which to start the surface of revolution <i>Default: 0</i>	angle	C Q E
-endSweep (-esw) The angle at which to end the surface of revolution. Default is 2Pi radians, or 360 degrees. <i>Default: 6.2831853</i>	angle	C Q E
-useTolerance (-ut) Use the specified tolerance to determine resolution. Otherwise number of sections will be used. <i>Default: false</i>	boolean	C Q E
-degree (-d) The degree of the resulting surface: 1 - linear, 3 - cubic <i>Default: 3</i>	int	C Q E
-sections (-s) The number of sections determines the resolution of the surface in the sweep direction. Used only if useTolerance is false. <i>Default: 8</i>	int	C Q E
-spans (-nsp) The number of spans determines the resolution of the surface in the opposite direction. <i>Default: 1</i>	int	C Q E
-tolerance (-tol) The tolerance with which to build the surface. Used only if useTolerance is true <i>Default: 0.01</i>	linear	C Q E
-heightRatio (-hr) Ratio of "height" to "width" <i>Default: 2.0</i>	float	C Q E

C Flag can appear in Create mode of command E Flag can appear in Edit mode of command
Q Flag can appear in Query mode of command M Flag can be used more than once in a command.

Modification des attributs d'un objet

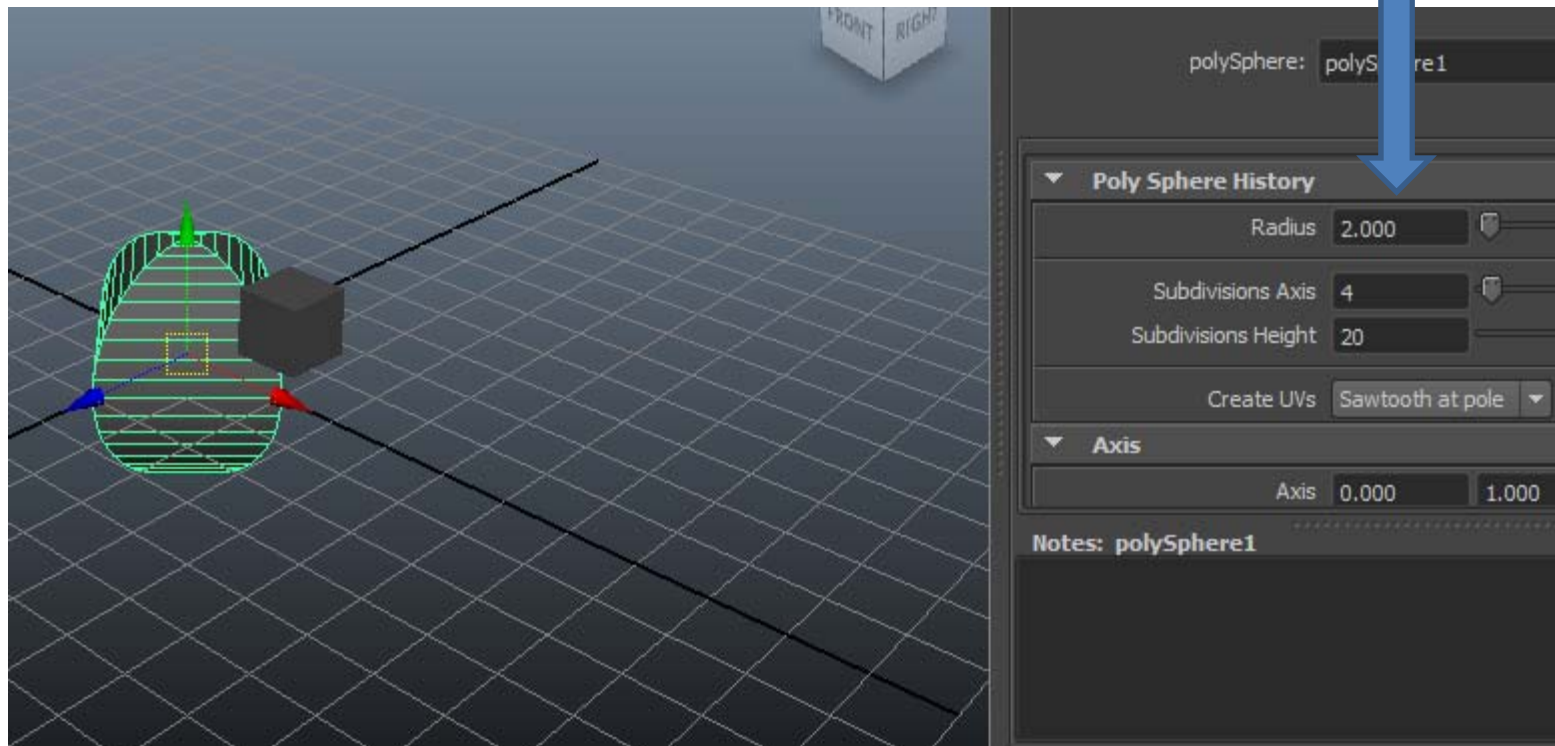
- Création d'un cube
 - polyCube;
- Modification de la position du cube
 - move 3 2 1;
- Le cube (objet sélectionné) est translaté vers (3,2,1)

```
polyCube;  
// Result: pCube1 polyCube1 //  
move 3 2 1;
```

Spécifier des attributs d'une géométrie

- Utilisation des paramètres (« Flags »).

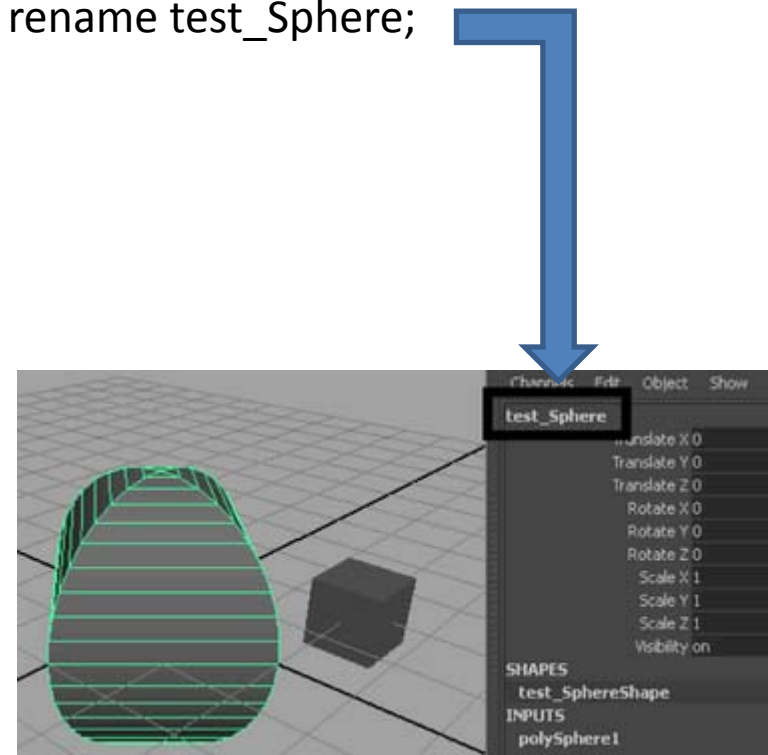
```
polySphere -radius 2 -subdivisionsX 4;  
// Result: pSphere1 polySphere1 //
```



Spécifier des attributs d'une géométrie

- Renommer une géométrie

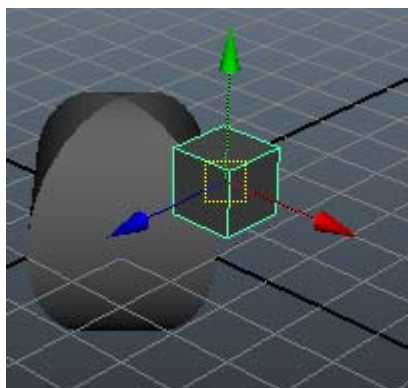
```
rename test_Sphere;
```



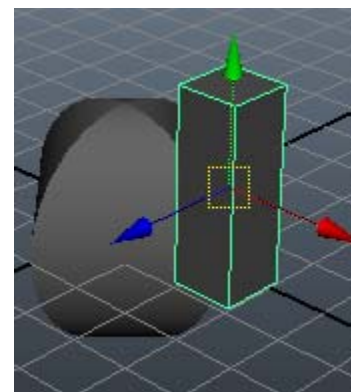
Editer des objets avec des commandes Maya

- Modifier la hauteur du cube

Sélectionner le cube



```
polyCube -edit -height 3;  
// Result: Values edited. //
```



Ajouter un bouton au shelf (étagère Maya)

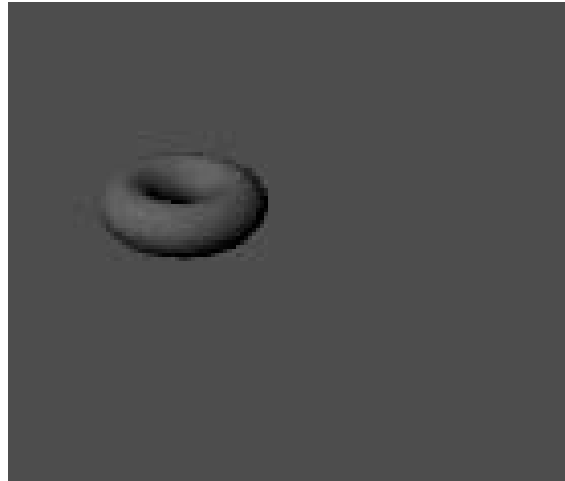
- Possibilité d'enchaîner un ensemble de commandes MEL.
- On peut ensuite sauvegarder cet ensemble de commandes afin de les réutiliser par la suite.
- Exemple d'un script d'éclairage

Ajouter un bouton au shelf (étagère Maya)

- Création de la scène

```
polyPlane -height 40 -width 40;  
torus -axis 0 1 0 -heightRatio 0.5;  
move 0 0.5 0;
```

- Rendu initial de la scène

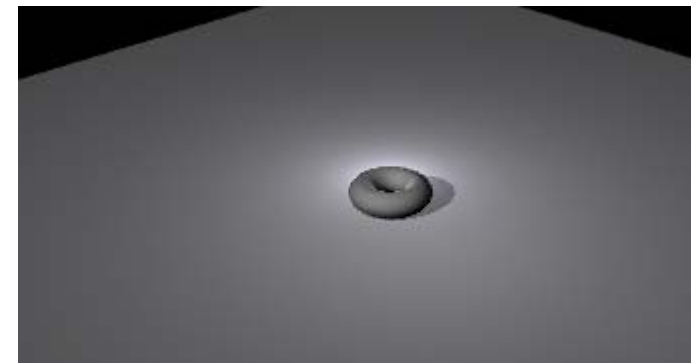


Ajouter un bouton au shelf (étagère Maya)

- Création d'un script d'éclairage

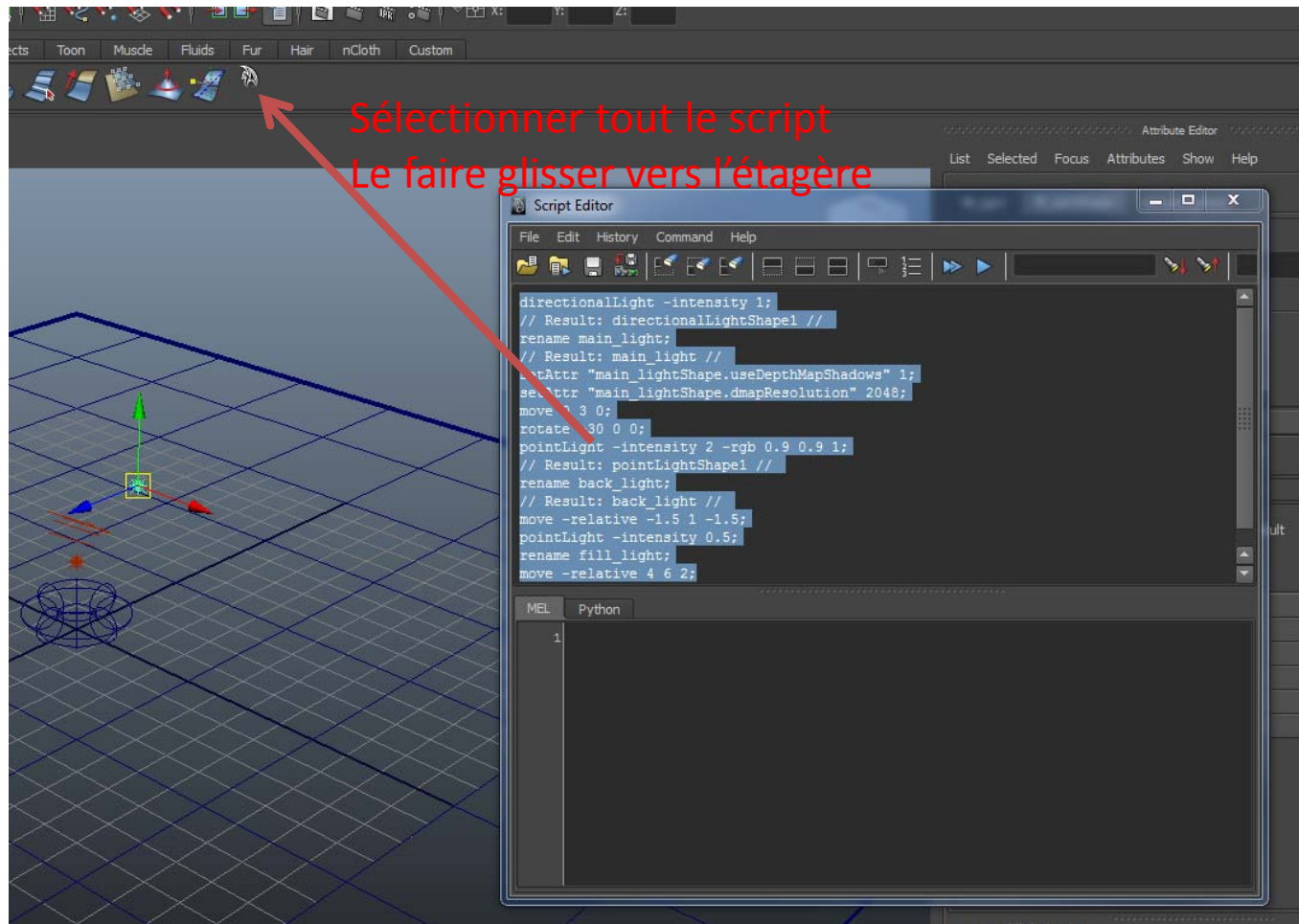


```
directionalLight -intensity 1;  
rename main_light;  
setAttr "main_lightShape.useDepthMapShadows" 1;  
setAttr "main_lightShape.dmapResolution" 2048;  
move 0 3 0;  
rotate -30 0 0;  
pointLight -intensity 2 -rgb 0.9 0.9 1;  
rename back_light;  
move -relative -1.5 1 -1.5;  
pointLight -intensity 0.5;  
rename fill_light;  
move -relative 4 6 2;
```



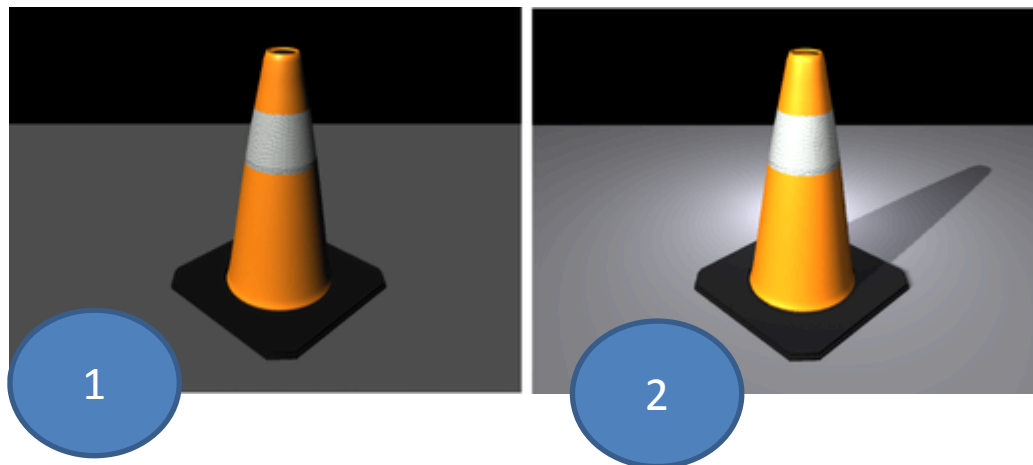
Ajouter un bouton au shelf (étagère Maya)

- Ajout du script à l'étagère



Ajouter un bouton au shelf (étagère Maya)

- Tester ce nouveau bouton sur un nouvel objet
 - Ouvrir un nouvel objet (exemple Mel_cone.mb)
 - Faire un rendu (1)
 - Lancer le script en appuyant sur le bouton que l'on vient de créer
 - Refaire un rendu (2)

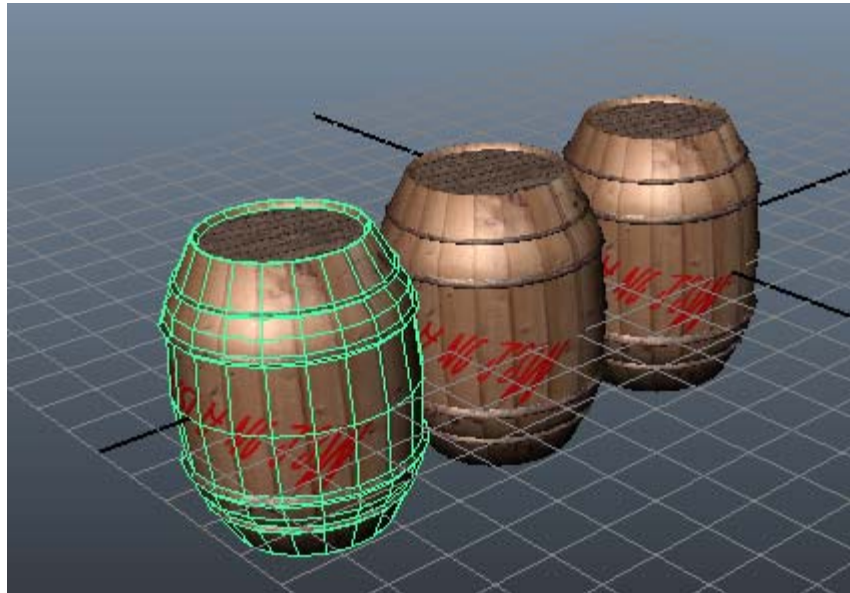


Utiliser des variables dans MEL

- Création d'une variable
 - `float $diameter_barrel;`
- Assignment d'une valeur
 - `$diameter_barrel = 4.2;`
- Création d'une variable String
 - `string $testString = "this is a test";`
- Affichage de cette variable
 - `print $testString;`

Utiliser des variables dans MEL

- `duplicate;` // on duplique l'objet (on crée une nouvelle instance)
- `Move -r 0 0 $diameter_barrel;` // on déplace l'objet



Utiliser des variables dans MEL

- La commande `select` avec le paramètre `-allDagObjects` permet de sélectionner tous les objets du DAG (Directed Acyclic Graph) (Ensemble des objets qui existent dans la scène comme
 - géométrie,
 - Chaines cinématiques,
 - Informations de textures et d'animation
- `select -allDagObjects;`

Utiliser des commandes plus évoluées de MEL

- Utilisation de la cinématique

- `select -allDagObjects;`
- `move -r 0 (0.5*$diameter_barrel) 0;`
- `performDynamics 1 Gravity 0;`
- `polyPlane -height 100 -width 100;`
- `rigidBody -passive;`

Déplacer tous les objets

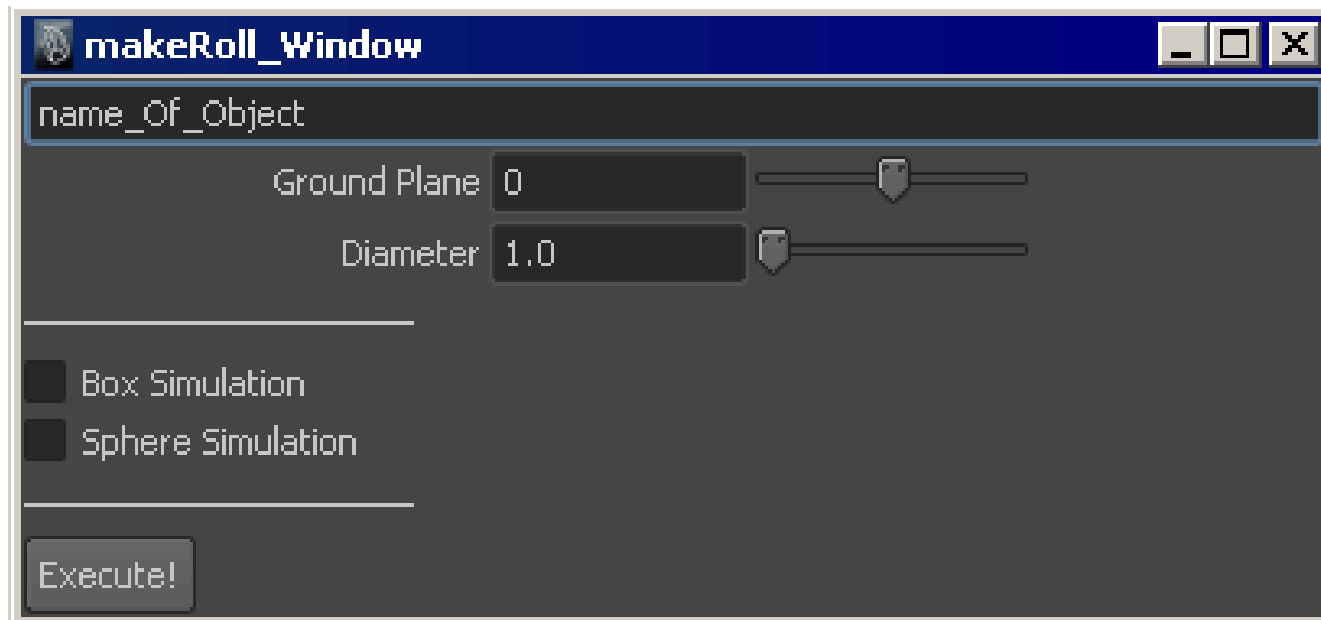
Ajouter de la gravité

Ajouter un plan non soumis à la gravité



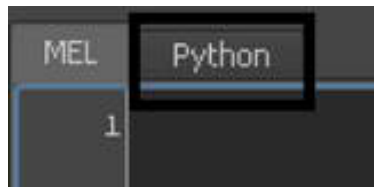
Création des fenêtres sous MEL

- Permet de créer ses propres interfaces lors du lancement d'un script



Utiliser Python dans MAYA

- Comme avec MEL, il est possible d'utiliser des commandes Python en ligne de commande ou dans l'éditeur de script sous MAYA
- Sélectionner « python » dans l'éditeur de script



- `import maya.cmds as cmds`
 - Il est maintenant possible d'accéder aux commandes Maya directement en python

Utiliser Python dans MAYA

- `cmds.polyCube()`



Commandes MEL pré-fixées

- `cmds.move(1,2,3)`
- `cmds.delete()`
- Utilisation de paramètres
 - `cmds.polySphere(radius=1, subdivisionsX=4, name="testSphere")`

Utiliser Python dans MAYA

- Exemple

- `testVarName=cmds.torus(r=1, axis=(0,1,0)) // Donne un nom de variable à l'objet`
- `cmds.select(deselect=True) // désélectionne l'objet courant`
- `cmds.torus(testVarName,edit=True, hr=0.4) // édite un paramètre de l'objet « testVarName »`

- Possibilité d'appeler du MEL directement dans python

- `import maya.mel // import du module MEL`
- `maya.mel.eval("sphere -radius 3;") // appel d'une commande MEL dans python`

Travail à faire ...

- Si vous ne connaissez pas Maya
 - Suivre le lien :
 - http://download.autodesk.com/us/maya/maya2013_getting_started/index.html
 - Lire (et faire) les tutoriaux
 - Overview
 - Maya Basics
 - Polygonal Modeling
 - NURBS Modeling
 - Subdivision Surfaces
 - Animation
 - Character Setup
 - HumanIK
 - Polygon Texturing
 - Rendering

Programme des prochaines séances

- Tutoriaux maya
 - Scripting in Maya (Lesson 1, 2 et 4)
 - Tutorial Lesson 3
- Python
 - Découverte de python
 - Utilisation de python pour reprendre les fondements de l'algorithmique
 - Projet : mise en application