

# LINKS WITH TIPS FOR PYTHON & MODO

Compiled by Cristóbal Vila / etereaestudios.com

Last update: July 16, 2019

---

## INTERESTING TOPICS, COMPILED FROM MODO FORUMS (not only, but mainly)

- [A bit of l-system, turtle and procedural geometry](#)
- [Access Statistics -> Polygons -> Part via Python API](#)
- [Access to curve parameters with TD SDK](#)
- [A single index.cfg that can load selectively a script from multiple versions, for a specific modo version](#)
- [Accessing all elements of a Material Group in the shader tree](#)
- [Add appropriate UNITS to a given result calculated by a script](#)
- [Add item channels to schematic macro](#)
- [Add \(and see\) frame width and height to custom form, no matter the polyRender Item is not selected](#)
- [Add ! or !! before the eval command to suppress any dialogs](#)
- [Adding Custom Channel To Schematic](#)
- [Add intermediate joints - Interesting commented script by Ivo Grigull \(TD SDK\)](#)
- [Advices to create paths to load something in Modo](#)
- [Applying Command to Each Item in Group Locator](#)
- [Applying UV mapping on poly groups \(Script\)](#)
- [Area calculating tool \(script, pyModo is needed\)](#)
- [Asking about few variables in one dialog box \(using Python API\)](#)
- [Automatically retrieve the part name associated with selected polygons \(and eventually select it\) \(was "Script to select same polygons with same "part" name"\)](#)
  
- [Batch converting files to native LXO format \(Script\)](#)
- [Button \(command\) to switch Render Region on/of but remains permanent in the UI](#)
- [Build a rotation matrix based on 3 selected verts. Calculate the euler angles \(derived from this post, describing a bug with workplane\)](#)
  
- [Calculate a world normal, rather than an object-space normal](#)
- [Calculate euler angles of rotation as \(X,Y,Z\) from 3x3 matrix and desired rot order \(Script\)](#)
- [Calculate x,y,z sizes of selected geometry\\_ \(since 'layer.bounds' will give you the bounding box of the entire layer\)](#)
- [Capture and using the Sequence of Components selection in a script](#)
- [Check if channel exists](#)
- [Checking for Polygon or Sub-D mode](#)
- [Classes and Sel • Python Programming Tutorial \(videotutorial\)](#)
- [Clear 'render browser' thumbnails at bottom of render window on launch](#)
- [Cycles through all mesh layers, select the points in each in groups and creates a curve through them](#)
- [Code for 'Jitter' effect \(look posts by Dongju\)](#)
- [Command for duplicating the selected morph map](#)
- [Command for freezing a morph to a new mesh \(for Modo 901\)](#)
- [Commands to Toggle Side Panels Modo Beta Layout – Mark B Tomlinson](#)
- [Command to prevent modo from writing to the config when it closes](#)
- [Command that returns layer index and polygon index without selecting it](#)
- [Command to re-fire the latest script](#)

- [Config and Kit issues](#) (isolate certain custom kits to certain versions of Modo)
- [Convert curve to bezier](#) (and the opposite)
- [Convert 3x3 matrix to rotX, rotY, rotZ](#) (using the Matrix3 from the TD SDK to get the euler angles)
- [Convert Radians to Degrees using "math.degrees\(radian\\_value\)"](#)
- [Converting collection of 3d meshes to native modo format LXO](#)
- [Couple of TIPS about "don't edit the scene state via script using the SDK" - "there can be many foreground layers and the easiest way to get them all is..."](#)
- [Create gradients set to Particle ID as group masks for any selected groups in the shader tree](#) (Script)
- [Creating bezier & curves based on points information stored in an arrays](#)
- [Creating random verts 100,000 times \(takes a second or two\)](#)
- [Creating Smart Function on MODO](#) (fire commands depending on: component mode / selected components / if it failed (error) then run another command)
- [Creating a 'YES' / 'NO' dialog](#)
- [Creating Mutually Exclusive or Toggle Buttons](#) (includes example with 3 buttons, to work with 901 and up)
- [Curve color from Texture](#) (script)
- [Cycle through available viewport cameras](#)
  
- [Deleting a part. Which command?](#)
- [Deselect a Percentage of Polygons](#) TD-SDK example
- [Deselect all vmaps except a specific type](#)
- [Determine Layer Order in Shader Tree](#)
- [Dictionary-file-for-Luxology-Modo-Python](#) (compiled by Kursad Karatas, 2010)
- [Differences between lx.eval / lx.eval1 / lx.evalN](#) ([more](#), [more](#), [more](#), [more](#), [more](#))
  
- [Editing a mesh without LayerScan](#)
- [Embedding User values in a scene?](#) Not possible. TIP: use scene-tags instead
- [Examples of wrapped basic commands using SDK](#) by Keith
- [Exporting layouts. How to set up configs kits \(for multiple modo versions\)](#)
- [Exporting OBJ sequence](#)
  
- [File open dialog and default paths](#) (set up a file open dialog with a script in a way that the path to open is in the same directory as the scene file is in)
- [Filter the allowed items using value hints. Simple example which is just only showing items with type "mesh" in the drop-down \(Python API\)](#)
- [Finding a mesh associated with an MDD / Identify 'orphaned' mdd deformer \(without forward or reverse connections in the deform graph\)](#)
- [Finding polygon tag on a mesh item](#) (a way to get any associated polygon tags on a mesh item to grab a list of material groups attached to a mesh item)
- [Find the euler rotation values of selected face on a mesh in Modo](#) • [Another approach \(more simple\)](#)
- [Flat in x,y,z automatically](#) (script)
- [For loop to move items](#)
- [Function that returns all items in the scene of the specified type\(s\) as a list of Python objects \(Python API\)](#)
  
- [Get a node connected through a channel](#) (Python API)
- [Get active mesh item in python](#)
- [Get all the parts on a mesh \(Python API\)](#)
- [Get all polys connected to an edge or vertex](#) (various solutions)
- [Get an item's particular position/direction inside of the scene \(Python API\)](#)
- [Getting a channel's value \(Python API\)](#)
- [Getting a curve's length](#) (script using Python API)
- [Get information about the active camera \(\\*\\*\\*Or even viewport perspective?\\*\\*\\*\)](#) [through the Python API](#)
- [Get UV maps and set UVs in Python](#) (a way to get the number of UV maps and UV map)

names of each mesh item)

- [Getting Item's Internal Name](#) (trick: if you're passing name strings you should surround them with curly brackets to escape special characters & spaces)
- [Getting the layer position of an item in the Shader Tree & Item List](#) (only use the python API and command via lx.eval() no TD SDK)
- [Getting vertex map values across layers](#)
- [Get pass groups and passes by name \(Commands that take an item argument can actually take either NAME or ID\)](#)
- [Get the path of all dead clips](#)
- [Get total length of all edges in a selection](#)
- [Get the bounding box of selection for the primary layer](#) (fast Python API Script by Farfarer)
- [Get the Render item's outPat \(output pattern\) settings?](#)
- [Get Vmap name](#)
- [Getting orientation of viewport \(Python API\)](#)
- [Getting the transform items connected to an item using sceneservice](#) (and some code using the new API to walk the transform graph)
- [Getting weight maps names in readable form](#)
- [Guide to embedding custom GUI's in modo](#) (setup and install custom PySide Viewports for Modo)
- [Gooch Tone Shading Fix for 801](#)
  
- [Hidden commands](#) (mesh.octree, apericube)
- [How to isolate certain custom kits to certain versions of Modo](#)
- [How to query shader.lock / unlock state](#)
- [How can I cycle shading mode for inactive Mesh?](#)
- [How can I make my command button face highlighted when in the specific state?](#) [Python API]
- [How many polygons are selected and show to the user](#)
- [How is this that sometimes an 'else' statement doesn't work? \(more about complex 'if'/'else' statements\)](#)
- [How to bind key depending on component mode](#)
- [How to get an item's weight and morph map names](#)
- [How to get all available values of a dropdown list via Python](#)
- [How to get the currently selected channels](#)
- [How to get the length of MeshEdge](#) (including various edges at a time)
- [How to get the list of selected channel of an item](#) "like i select position.x on the propertie tab and i get myItem.position.x or the name of the channel"
- [How to get vertex position relative to a work plane](#) (some problems/bug here... Solved in 801, I think...)
- [How to get polygon tag type from material group name or ID](#)
- [How to point modo at content/kits folder when loading asset](#)
- [How to select a material & material group by name](#)
- [How to set up configs kits for multiple modo versions](#)
- [How to start with scripting](#)
  
- [Ideas for a script to planarizing n-gons.](#)
- [Import point cloud data](#) (some tips)
- [Interesting use for "query sceneservice item.rootIndex ? %s"](#)
- [Interesting post about Scripting, Memory, ways to iterate over a sequence and some simple functions by Keith](#) (Python API & TD-SDK)
- [Interesting Pithon API / TD utilities](#)
  - # To know everything you can do with the first selected item. drop selection, then select any item and run that code to find out what it can do
  - # If you tac `.\_\_doc\_\_` on the end of pretty much any TD python object, you'll see the internal documentation for that object.
  
- [Is there a way to define a user variable as temporary \(in the CFG\)? ---> \(...\) make](#)

[config\\_user values and mark as transient \(...\)](#) the user value will always have the same value on launch.

- [Items Random Select](#) (script)
- [Item.worldRot query](#)
- [Iterate through edges \(select them progressively\)](#).
- [Iterate through a few group locators in my scene, and delete everything but one named test](#)
  
- [Key to toggle through Transform tools](#)
- [KOMODO - Open source dev tools for MODO](#) (by Adam O'Hern)
  
- [Layered modo Config Setup](#)
- [Layered modo Configs under OS X 10.8 and above](#)
- [Learn Python The Hard Way: Symbol Review](#)
- [Link subitem manually](#)
- [List all material tags being used by meshes in the scene](#)
- [List everything that's in the shader tree, and what it's name, id, parent, and children are](#)
- [List of all the categories you can query with the sceneservice](#)
- [List up all the commands in the command history](#)
- [Load an image in a kit with a relative path](#)
- [Load an assembly preset from a script](#)
- [Local and world matrices. This is a big timesaver if you need to find local, global, or relative transforms](#)
- [Looking for work around: Local action center and Rotate](#)
- [Loop Through Materials and get all textures assigned to them](#)
  
- [Make the Render Item be the parent of our recently created ShaderTree element](#)
- [Mathematics \(and " \\_init\\_" classes\) in Python](#)
- [Mathematics behind Computer Graphics: 2D & 3D - Transforms, Homogenizing, Algebra, Planes, Parametrics, Polynoms, Bezier...](#) (Østfold University College)
- [Matrices for Tech Artists](#) (a set of posts explaining how matrices work from the perspective of a technical artist/rigger)
- [Matrices y Determinantes](#) (curso básico - Spanish)
- [Method for activating most recently-created pass from a pass group](#)
- [Model a subdivided plane and extract tiles of this giant plane for texturing purpose in my game once finished](#)
- [Modify a vertex using WORLD, WORKPLANE and ITEM coordinates, no matter there is an active Workplane](#)
- [Modo script shelf](#) (this kit will add a script shelf palette to modo. You can simply drag and drop scripts on the shelf, it will be saved and a button is generated to run this script)
- [Moving object on grid with arrow keys](#) (interesting idea about querying gridFixed value for increments)
  
- [Open a simple UV or 3D window from a script](#)
- [Operations between Lists: add SUM of values of two LISTS into new LIST](#)
- [Operations between Lists: wise multiplication of two LISTS into new LIST](#)
- [Optimising / cleaning shader tree](#)
  
- [Parsing layer names from large assemblies and restoring hierarchy](#)
- [Perfectly Centered Bones Script](#)
- [Preset browser paths arent exported in config fragments. Copy this out of your modoXXX.cfg manually](#)
- [Printing the Bounding box Dimensions values](#)
- [pyModo + pyModoT \(tools\) + pyModoS \(schematics\)](#) (little —BIG— library to make modo scripting easier, by Keith Sheppard)
- [Python API example demonstrating how to modify GL drawing of an item](#)
- [Python API Q&A: custom item draw & unprocessed exception message](#)

- [Python API - Basic questions from a newbie](#)
  - [Python API - Accessing Modeling Falloffs](#)
  - [Python API - Render progress listener that can fire a command on render begin and another one on render end](#)
  - [Python API - Set undo point in script](#)
  - [Python API - Vector and Matrix operations](#)
  - [Python API - Visitors, Mark Modes and Selections](#)
  - [Python script to open external file \(SDK\)](#)
  - [Poly, Edge and Vert selection Count over multiple items](#)
- 
- [Quaternions: rotating coordinate system via a quaternion](#)
  - [Quaternions: using Quaternions to represent rotation](#)
  - [Querying items in a material group \(Script that adds a render output to each selected shader tree group mask\)](#)
  - [Query all item names in "if" statement • Check if the name contains the search string](#)
  - [Query Backdrop image?](#) (command to find out what image the selected backdrop item is using)
  - [Query the action center currently set](#)
  - [Query bounding box size of layer with and without morph](#)
  - [Query channel values directly, no need for the sceneservice](#)
  - [Query deformed vert position](#)
  - [Query for a list of channels on an item](#) (to determine if an item has a transform (Rotation), and if it does, select the X Rotation channel)
  - [Query edge selection sets in modo](#)
  - [Query if a given vmap exists for the MAIN layer \(to select or create one\)](#) (using Python API but usable inside a fire&forget script)
  - [Query if a selected vertex belongs to a Polyline, Curve or Bèzier](#)
  - [Query if Center Item is Selected](#)
  - [Query edge length + ask user for new length + scale mesh to make the edge desired length](#)
  - [Query list of installed Kits](#)
  - [Query the FIRST and LAST verts on a bunch of consecutive selected edges](#)
  - [Query to get material name](#) (different ways)
  - [Query to get the number of selected islands](#) (script example by Farfarer using pyAPI)
  - [Query to detect component mode](#)
  - [Query operator for ShaderTree](#)
  - [Query Render Outputs in a Scene using Python API](#) (faster way)
  - [Query vert position in different layers](#)
  - [Question about tuple list](#)
  - [Query the angle between the two polys connected to an edge](#)
  - [Query sceneservice item.rotOrder](#) (and other interesting things about Items, Rot, etc)
  - [Quick way to build a string list for a dropdown \(TD SDK, I think\)](#)
- 
- [Random wire-frame colors](#)
  - [Randomly change the diffuse colours of selected materials](#) (Script)
  - [Read float values from an external .TXT file for a script](#)
  - [Reading UV Map Values using Python API](#)
  - [Recognize each mesh island in the same layer - Another version](#)
  - [Remove a link between an imageMap and its linked videoSequence](#)
  - [Resolving problems with a path to file included on a script command](#)
  - [Reset main config file](#) (Script + Mac-Automator tip)
  - [Run commands without selection](#) (a way to run this command by giving it an item name rather than relying on an item selection)
  - [Running pen tool via fire and forget script](#) (best approach)
- 
- [SaveAs without opening saved scene](#)
  - [Save the indices of the selected polys in an array instead of creating a selection set • \(solution\)](#)

- [Script shelf](#) Will add a script shelf palette to modo. Drag and drop python scripts on the shelf, it will be saved and a button is generated to run script
- [Scripting for dummies? Very simple code examples.](#)
- [Scripting tools for MODO by Adam O'Hern](#) (Better Command History, Project Scripts, Working Scripts, Snippets, Test Scene, etc)
- [Script that will set all render output filename to a folder in the open scene](#)
- [Script to add locators inside a folder](#) (two methods: old style fire & forget and TD-SDK)
- [Script to check if a mesh is subdivided](#)
- [Script to rename all the dynamic parent subItems in a scene to include the parent\(s\) name](#)
- [Script to count the number of polygons islands in the item](#)
- [See if an object has a certain UV that is inside of the scene. If it doesn't, change it to something else, or create a new one](#)
- [selMode = lx.eval\('query layerservice selmode ?'\) Alternative](#) (write API scripts without using lx.eval at all)
- [Selecting and deselecting multiple layers \(mesh items\) and polygons in a script](#)
- [Select a bunch of items and a bunch of material masks, modo will assign the selected masks randomly to the item selection](#) (Script)
- [Select desired rot channels for all selected items in scene](#) (Script)
- [Select/DeSelect based on vert index](#)
- [Select image in clip browser by name](#)
- [Select Polygons from a List of Polys, to apply further operations \(fast procedure for LOTS of polys\) - Using wrapped SDK](#)
- [Select mesh item and copy-paste polygons](#)
- [Select \(only\) UVs / Polygons in a specific UV space](#)
- [Select \(query\) all Polygon Selection sets](#)
- [Select vert based on position](#)
- [Selects all polygons which face straight up and are at or below 0 in the Y axis](#) Fast Python-API script by Farfare
- [Selected polygons set the normals of their vertices to be the geometric normal of the polygon](#) (Drop in .../lxserv. Then select your polygons and run the command **ffr.fullNormals**)
- [Selected Vertex Maps counter](#)
- [Set a hotkey with python scripting](#)
- [Set visibility of base material with script](#)
- [SoftSelection Falloff to Weight Map in one click](#)
- [Some misc class in Python API to speed up code](#)
- [Solve operations looping through the polys backwards](#) (and using Random)
- [Split joints \(or "bone-breaker"\). Different approaches](#)
- [Start modo with a command line switch that will stop it from writing a new config file when it closes](#)
- [Store all that is Selected in ShaderTree, Deselect and finally Reselect](#)
- [Stroke Drawing · Draw shapes in the SDK, \(it's not like drawing a curve\)](#)
- [Surface area calculator](#) (script to get the surface area of a mesh, based on Heron's formula)
- [Surface Particles from Weight Map](#)
- [Summary pages for all Modo Interfaces](#)
- [Suppress dialog popups by added an exclamation before your command](#)
- [Switching from Python Scripting to Python API - \(query all the selection sets of a mesh item\)](#)
- [Three videos by Keith Sheppard to help out those wanting to dive into scripting with python](#)
- [Tips on manipulating texture items through python](#)
- [To delete edge selection sets you better use reversed list of vmap indices.](#)
- [Toggle between the hide unselected and unhide commands](#) (It will run these commands on the item level but still retain your component selections)
- [Transfer all weight maps of a mesh to another](#)

- [Trigger an animation with a channel modifier in Python \(scene and source code included\)](#) + Tutorial
  - [Trying to add some paths to help menu on OSX Mavericks](#) (get modo to see a pdf located on an external volume)
  - [Turn a list into variables](#) (example code about how to go through the list and connect the mesh to the math nodes)
  - [User values have an "Action" attribute that allows you to specify a command or script that will be run when the user value's value changes](#)
  - [user.value command allows existing user values to be queried and modified](#)
  - [Using curve to smooth and modify edges](#) (to be enhanced as a proper Python script, and fix possible issues)
  - [UV PixSnap](#) Script for snap all vertexes from UV to closest pixel corner
  - [Vectors, matrices and quaternions — Python Computer Graphics Kit 2.0.0 documentation](#)
  - [Vector calculation \(basic operations: addition, subtraction, multiply, scaling, dot product, cross product\)](#)
  - [Walkaround for "query layerservice selection" bug](#) (since "query layerservice selection ? \*\*\*" is [or at least was] broken if you reorder the layers)
  - [Whats the logic for the undo crash so the script runs, and hitting ctrl-z bombs modo... \(sometimes\)](#)
  - [Way to control the PRECISION for a given distance "user.value" result](#)
  - [Way to get the path of all dead clips](#)
  - [Way to get the opposite edge of quad polygon](#)
  - [Way to list all \(selected\) schematic nodes](#)
  - [Way to select the nameless uv map](#)
  - [Way to check if a mesh has ngons](#)
  - [Way to check if a mesh is subdivided](#)
  - [Why are 3D transformation matrices 4x4 instead 3x3?](#)
  - [Writing Queryable Commands for Modo — Matt Cox](#)
- 

## QUERIES AND COMMAND SERVICES

- [Query LAYER.SERVICES for Modo 701](#)
- [Query SCENE.SERVICES for Modo 701](#)
- [Command Service for Modo 701](#)

## GISTS & GITHUB

- [Adam O'Hern](#)
  - [Good Kitty: a MODO kit for making MODO kits](#) (by Adam O'Hern)
- [Mechanical Colors / 9bstudios - FULL suite of Modo plug-ins at GitHub](#) (originally developed by Adam O'Hern, released as open source under the MIT license)
- [Bjoern Siegert aka Nicelife](#)
- [GregMalick's](#)
- [Ivo Grigull](#)
- [James O'Hare aka Farfarer](#)
- [Lukasz Pazera](#)
- [Matt Cox](#)
- [Tim Crowson](#)
- [Rockjail's aka Mirischgli](#)

- [Cristóbal Vila aka Eterea](#)

### ONLINE COURSES AND DOCS

- [Python v2.7.6 documentation](#) (official on Python.org)
- [Learn Python The Hard Way](#)
- [Python | Codecademy](#)
- [Google's Python Class | Python Education | Google Developers](#)
- [Python Programming Tutorials](#) (list of videotutorials by The New Boston)
- [The Hitchhiker's Guide to Python](#)
- [PEP 8 -- Style Guide for Python Code](#)
- [PEP 257 -- Docstring Conventions](#)
- [Code Like a Pythonista: Idiomatic Python](#)
- [Python para principiantes](#) (Spanish)
- [The Nature of Code](#) (well, this is for Processing, but it contains useful information that could be of help for Python also, I think...)
- [The Nature of Code · GitHub](#)
- [The Nature of Code · Examples for Python · GitHub](#) <--- **Stay tuned to this!**
- [The Nature of Code · Tutorials for Python in Grasshopper - YouTube](#)
- [processing.py · GitHub](#) Write Processing sketches in Python

### TUTORIALS (Both for Scripting and Modo generic)

- **[MODO: Automation 101 — pixelfondue](#) (VERY GOOD as introduction to learning to code for MODO)**
- [A look at Coding in Modo from an artist perspective](#)
- [The ever growing list of modo tutorial videos](#) • [Eglomot Database](#)
- [Matt Cox's Videos on Vimeo](#)
- [Sergio Mucino's Videos on Vimeo](#) (lot of interesting stuff on rigging there)
- [Getting Setup with Python and Modo on Vimeo](#) by Nicholas Stevenson (series of videotutorials)
- Youtube playlist about scripting in Python in modo 902 by Dominiek De Ridder - [All Videos](#) - [Python scripting in modo](#)
- [Interesting Python examples for Grasshopper \(Mesh Generation, L-Systems, Tubes, Patches...\)](#)
- [Formal Request Thread: PLEASE give us a "Learn Scripting/Python in MODO for Beginners" course!](#) (links to various courses)
- Video-tutorials on Scripting for Modo by Marc Albrecht: [Recording, editing and implementing a macro](#) / [Best ways to handle macros, script-like macros and examples](#) / [Changing Item Names](#)
- Video-tutorials on Scripting for Modo by Nicholas Stevenson: [Introduction](#) / [System Path](#) / [Installing Python and PySide](#) / [Python IDEs](#) / [Extracting Code Completion](#)
- [Video-tutorials on Scripting for Modo by Keith Sheppard](#) (YouTube List)

### CINEMA 4D (and other packages)

- [Scripting In Cinema 4D For People Who Know Zero Python on Vimeo](#) (video)
- [CINEMA 4D Python SDK Documentation](#) (some interesting stuff and links there...)
- [C4D Tutorial: Making Python Scripting As Easy As Possible For Artists on Vimeo](#)
- [Python Iterators for your Cinema 4D Scene - CG Rebel](#)
- [Hans Willem/c4dquicktools](#) (Python scripts for Cinema4D)
- [Code Vonc](#) - Lots of Python examples for Cinema4D
- [Python script for Rhino 5 and up that exports the NURBS form of selected curves as a](#)



[text \(JSON\) format](#) - Threat in Rhino forums: [Scrip needed: export 3D curves to a text file?](#)

## MISC

- [Scripts and kits database, by Zero3D and Mayaterror](#)
  - [Pauls Online Math Notes](#) a FANTASTIC set of free online (and downloadable) notes and/or tutorials about MATHS (cheat sheets & tables, class notes and tutorials)
  - [CG Math | Learning Resources — pixelfondue](#)
  - [NO BULLSHIT TEXTBOOKS](#) "The best way to learn math - 100% bullshit-free textbooks on high school math, vectors, calculus, mechanics, and linear algebra. Perfect for busy adults"
  - [A Primer on Bézier Curves](#) A free, online book for when you really need to know how to do Bézier things.
  - ["EULA Generator" creates an end user agreement prototype](#)
  - [Visually stunning math concepts which are easy to explain - Mathematics Stack Exchange](#)
  - [Plug-Ins | Scripts | Content](#) (great compilation/index of scripts in Modopedia site)
  - [The Algorithmic Beauty of Plants](#) (by Dr. P. Prusinkiewicz, the "bible" of L-Systems, full PDF)
  - [Algorithmic Botany: Publications](#) (more interesting papers and docs by Dr. P. Prusinkiewicz)
  - [The Book of Shaders](#) (step-by-step guide through the abstract and complex universe of Fragment Shaders)
  - [The Essence Of Animation Nodes: Transformation Matrix](#) (Blender oriented, but could be informative)
  - [Red Blob Games](#) What if you could learn math and computer science topics the same way you learn while playing a game?
  - [A Primer on Bézier Curves](#) All about mathematics behind bézier curves
  
  - [Beautiful Soup](#) - Python library designed for quick turnaround projects like screen-scraping:
    - A toolkit for dissecting a document and extracting what you need.
    - Automatically converts incoming documents to Unicode and outgoing documents to UTF-8.
    - Sits on top of popular Python parsers like lxml and html5lib, allowing you to try out different parsing strategies or trade speed for flexibility.
- 
- 

## MODO's Python API and TD-SDK documentation:

- [Python API - The Foundry MODO SDK wiki](#) (this is hard for non professional coders)

In words of Simon Lundberg:

*"There is no actual difference between lxtsd (the TD Python API) and the normal Python API.*

*lxtsd is just a module written using the Python API to wrap up a lot of the boilerplate and make it more Pythonic.*

*So if you're using lxtsd, you can actually use all the normal Python API stuff as well"*

- [MODO's TD SDK documentation — MODO TD SDK 0.0.1 documentation](#) (much more asquible for non professional coders)

*The TD SDK is a wrapper around Modo's core Python API focusing on plugin development.*

*It aims to be*

- *more pythonic and object oriented*
- *Make common tasks easier & more intuitive.*
- *Provide comprehensive documentation.*
- *Hide the complexity of the core Python API.*

*The TD SDK is in no way intended to be a replacement to Modo's command based scripting, but is rather complementary to it.*

---

---

## **MODO OFFICIAL SDK Wiki**

- [External Resources - The Foundry MODO SDK wiki](#) (lots of links to useful things created by the community)
- [Luxology SDK wiki HOME](#)
- [Scripting and Commands.pdf](#) (local-disk document, available in your Modo installation. Outdated but useful...)
- [Python](#) (basics and foundations. Communication with Modo)
- [Python API Overview](#) (someday I will start with this, on my next life... This is not for candid souls...)
- [Macros](#) (extremely simple things, but still useful. Hey, and you can add arguments to macros, also ;-)
- [Kits](#)
- [Plug-ins vs. Scripts](#)
- [Scripting](#)
- [Command System](#)
- [ScriptQuery Overview](#)
- [Category:Top level](#) (INDEX)
- [Category:ScriptQuery Reference](#) (SUBINDEX)
  - [CommandService Reference](#)
  - [HostService Reference](#)
  - [Layerservice](#)
  - [MessageService Reference](#)
  - [PlatformService Reference](#)
  - [ScriptQuery Interface 701 Updates](#)
- [Category:Configs](#) (SUBINDEX)
  - [Argument Types](#)
  - [Categories](#)
  - [Command Help](#)
  - [Defining Item Categories](#)
  - [Form Categories and Groups](#)
  - [Help URLs](#)
  - [Icon Resources](#)
  - [Macros](#)
  - [Message Tables](#)
  - [Preference Forms](#)
  - [Preset Browser Paths](#)
- [Select.typeFrom](#) (because scripts often needs to know which selection mode Modo is

in)

- [User Values](#)
- [Dialog Commands](#)
  
- [Form System - Luxology SDK wiki](#)
- [Form Categories and Groups](#)
  
- [Path Aliases - Luxology SDK wiki](#)
- [Pattern Matching - Luxology SDK wiki](#)
- [Modo Version History](#)