

# Tutorial 1: Introduction to the VRED Python Interface

To view the video, visit: <https://www.youtube.com/watch?v=urkrfSfdug0>

**NOTE:** There are no accompanying scripts for the *Tutorial 1: Introduction to the VRED Python Interface* video.

## Video Captions

Hello and welcome to our new tutorial series for scripting in Python for VRED Pro. My name is Christopher and I'm a software engineer that specializes in 3D graphics and developing plugins and workflows for 3D software. In the next 12 videos, I'll show you an overview of the Python interfaces that VRED Pro has to offer and show you how we can use Python scripting to write tools, modules, and plug-ins for VRED. We will explore how you could use the new services in the API version 2, and how you can use the new data types to your advantage.

I have to point out though that this will not be a Python coding tutorial. So, if you are just starting out, don't be disappointed that you will not learn a lot of Python language from these videos, but there are a lot of great tutorials courses out there that will help you become a professional Python developer. Instead, we focus on high-level concepts and best practices that you can combine to develop great and valuable tools. You will get a deep understanding of how you can use Python scripting to achieve what you want.

In this first tutorial, I will give you a quick overview over the possibilities with Python scripting and how you can start. We will show you scripting examples on how we can work with them. Last, but not least, I will show you how you can work with documentation efficiently.

VRED already has a lot to offer besides its main purpose, 3D rendering, especially with variant sets, it's possible to script and automate a whole range of different actions. And with the implemented VR tools, you are basically already set up for remote VR collaboration.

So, maybe you wonder why should I ever start scripting in VRED with Python. Well, VRED offers an extensive API, which gives you access to all functions that it has to offer. You can access all data and can manipulate any data you like and Python is basically the tool that allows you to unlock all these possibilities. You can just play around in the terminal, try functions and see how they change the scene. You can install small scripts and variant sets that execute when the variant sets are triggered or you can write more complex scripts in the Script Editor. You can develop plug-ins with custom interfaces that basically can change the whole way you work with VRED and share it with your coworkers or you can build up an automatic rendering or data preparation pipeline that can prepare CAD data for you. Well, you see, the possibilities are nearly endless.

If this sounds like a lot to take in, don't worry. We will cover each of these possibilities in several tutorials. And step-by-step, you will get to know VRED's Python interfaces and learn how to work with them.

Maybe you're thinking right now, when I can do everything with Python, where do I start? When you open VRED, you find an entry for all kinds of examples and a File in the menu bar. Here you will find a lot of example Python scripts to all kinds of topics. For example, you'll find scripts on how you can create and manipulate sceneplates, how you can write script plug-ins, or how you can integrate custom menus in VRED. This example here shows how you can create transformation constraints for objects. When we open the example, we find a few geometrical primitives and a text editor as a sceneplate. When we hit C and move, for example, the red ball, we see that all objects are connected in some way.

The Python magic happens in the Script Editor. Here is all the code that is responsible for this behavior. The code is automatically executed when you open the scene. Most of the example script how their own code edit in the Script Editor, so I can only recommend to have a look at the some of these examples. Try them out and also try to understand what is happening here. While exploring the samples, you can look at unknown functions and modules in the documentation. VRED also offers these examples on their website. They have lots of resources there if you want to learn something, and we will add a link to them in the description. When looking at these example scripts, you can see lots of identifiers that you may not be familiar with, like `vrnodeservice` or `createaimconstraint`.

To interact with VRED, we have to use these identifiers, or to be more accurate, modules, classes, and functions to tell VRED what to do. With their name, they already kind of tell you what they are supposed to do, but how can you find the correct functions on your own? This is where the rather boring part software development takes place, reading and understanding documentation. The VRED Python documentation contains all modules, classes, and functions that we can use to write scripts with.

You can access the documents in the menu bar under the Help > Python Documentation. The documentation is split into two parts, the old API version 1 documentation and the new API version 2 documentation. The most important skill in developing Python scripts for VRED is to find out how you can access the functionality for the result you want to achieve. In the API version 1, we can find functions by clicking the Everything link in the left menu of the page. This will open up a list with all functions in this documentation. From here, we can do a full text search with Ctrl F in our browser. For example, if I want to create a material, I would search for "create" or "creatematerial" and then go through the search results. When you click on one of these search functions, you directly land on the documentation for it.

In the API version 2, we can go to the search page and search for keywords. The results page will show a lot of examples, but it will also show the modules and classes that contain your search words. The VRED modules and classes always begin with "vr" or "vrd" and when you open these pages, you can continue with a full text search.

VRED also has a forum where you can find answers or post your own questions. People there are really helpful and there are a lot of great answers that you can search for. Before asking a question, be sure to if your problem already has an answer. This is marked with a green check mark. Besides the community, the Autodesk staff is also really active and tries to help where they can.

Python scripting in VRED is a great way to build your own tools and extend the functionality how you need it. Each tutorial in the series will focus on one aspect of Python scripting in VRED, so be sure to check them all out. That's it for this introduction for scripting with Python. Thanks for joining me for this tutorial and I hope we see each other again in the next videos.