RayStation Scripting

# Description

Miscellaneous notes on Python scripting in RayStation 11A. We use CPython 3.8. Our GUIs are WinForms.

# Notes

## Tools

### Version control

#### [Git](https://git-scm.com/book/en/v2)

A version control system (VCS). A VCS tracks and manages changes to code. It is useful for collaborating on code without interfering with each other's changes.

Useful resource: [Dangit, Git!?!](https://dangitgit.com)

#### [Git for Windows](https://gitforwindows.org)

Git was originally developed for and comes with Unix-based operating systems such as Mac and Linux. To use Git on Windows, you must download Git for Windows.

**Note:** If you search *git* in the Windows Search bar, you may see Git Bash, Git CMD, and/or Git GUI. You should also see the git *command*.This command is "the actual Git," and those other apps are just tools for using the command. These apps are part of the Git for Windows download, but they are not necessary for using Git.

#### [GitHub](https://docs.github.com/en)

An online platform for Git that allows teams to collaborate on code.

#### [GitHub Desktop](https://desktop.github.com)

*Not* the same as Git GUI. GitHub Desktop is a GUI alternative to using Git from the command line and is useful for beginners to Git and GitHub.

### IDE

We use [Visual Studio Code](https://code.visualstudio.com), but any IDE is better than a text editor, including the very minimal text editor in RayStation. VS Code interfaces with Git and GitHub, too.

### [Python](https://wiki.python.org/moin/BeginnersGuide)

Python is a versatile programming language. RayStation comes with Python support.

One of Python's many advantages is easy installation of packages, third-party libraries of Python code. We use [pip](https://pypi.org/project/pip) as our package manager.

#### Python.NET

Package [pythonnet](http://pythonnet.github.io) lets Python use the .NET 4.0+ Common Language Runtime (CLR).

.NET is a software framework, a type of library that provides "scaffolding" to extend to build your own app. .NET is designed for Web development.

The CLR is a runtime environment, a platform that contains everything you need to run an app.

.NET uses DLL assembly files to provide functionality. Python.NET allows you to treat .NET DLLs as Python packages.

Python.NET is for CPython, not IronPython. CPython and IronPython are implementations of Python. Python is the language while the implementation is the way the language is interpreted under the hood. For example, the CPython interpreter is written in the C language. When you think *Python*, you are thinking of CPython. I mention this distinction because RayStation comes with IronPython in addition to CPython.

We use Python.NET instead of a more popular Python GUI tool such as Qt because RaySearch's scripting examples use Python.NET. RaySearch’s reason for using Python.NET is that previous versions of RayStation supported only IronPython, not CPython, and the tool used to create GUIs in IronPython is most similar to Python.NET for CPython.

Useful resource: [.NET documentation](https://docs.microsoft.com/en-us/dotnet). Geared toward C# but easily translatable to Python.

## Misc. Tips

* Script execution logs are stored in C:\ProgramData on the RS server. RS support often looks at these.
* Potential reasons that a beam set can’t be exported:
  + No machine is set: beam\_set.MachineReference is None
  + Machine is not commissioned (often occurs w/ imported dose): machine\_db.GetTreatmentMachine(machineName=beam\_set.MachineReference.MachineName) is None
  + Dose is imported: beam\_set.HasImportedDose()
  + No beams
  + Beam(s) have no segments
  + No dose (or dose needs re-computing)
  + Dose stats are missing
  + Planning exam has no external geometry
* Geometries do not automatically after the ROI derived expression is set with roi.SetAlgebraExpression(). You must then call roi.UpdateDerivedGeometry() to set the geometry on a particular exam.
* It may be tempting to use a ListBox instead of checkboxes for multi-selection in a GUI because the former has keyboard shortcuts built in. But it actually doesn’t. For example, even if the ListBox’s SelectionMode is set to MultiExtended, Ctrl+A to select all items is not automatically supported.