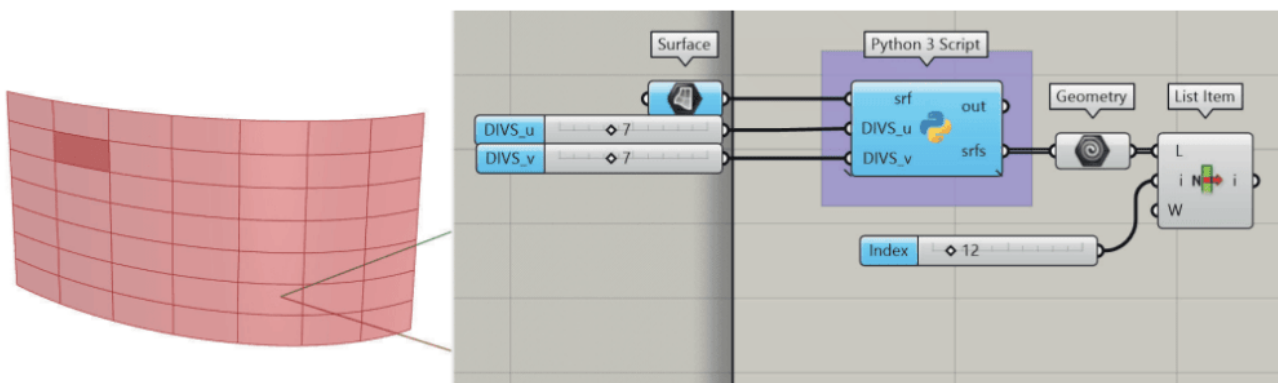


“Learn Rhino Python without knowing Python”

Learn Rhino Python without knowing Python by Francisco Pérez Arribas is a practical guide that positions [Python](#) not as a replacement for Grasshopper, but as a complementary tool to extend its capabilities where visual logic reaches its limits.

The book is designed for users with a basic understanding of [Rhino](#), whether they come from a purely visual [Grasshopper](#) workflow or have no programming experience at all. Rather than introducing coding as an abstract discipline, it is framed as a means of gaining precision and control over operations that are difficult to manage visually, such as loops, conditionals, and data filtering.



```
1 import rhinoscriptsyntax as rs
2
3 ##### List definition
4 srfs = []
5
6 ##### Geometric operations
7 dom_u = rs.SurfaceDomain(srf, 0)[1]
8 delta_u = dom_u / DIVS_u
9
10 dom_v = rs.SurfaceDomain(srf, 1)[1]
11 delta_v = dom_v / DIVS_v
12
13 # Isotrim technique with rs.TrimSurface()
14 for I in range(DIVS_u):
15     for J in range(DIVS_v):
16         s = rs.TrimSurface(srf, 2, ... # Breaking lines for clarity
17         ... [[I*delta_u, (I+1)*delta_u],
18         ... [J*delta_v, (J+1)*delta_v]])
19         ...
20         srfs.append(s)
```

Using the GH Python component to divide a surface into patches

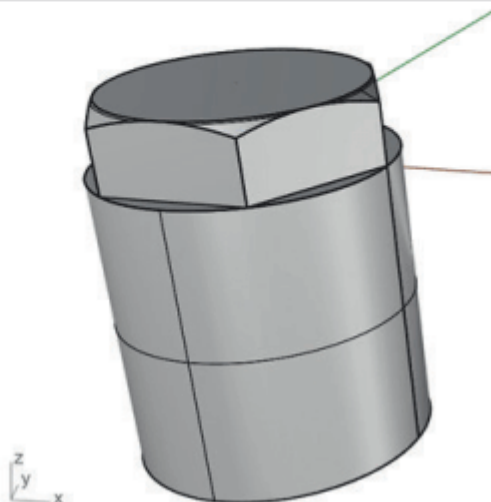
and extract the patches (srfs) into GH. Then, using the List Item component to explore them.

A key premise is familiar to many Grasshopper users: definitions that start clean and readable can quickly become complex systems of tangled connections. Rather than abandoning Grasshopper, the book proposes a hybrid workflow in which Python is used to encapsulate complexity within custom components, while the overall structure remains visual and intuitive.

This approach is reflected in the examples, where Python scripts are embedded within Grasshopper components to perform geometric operations such as surface subdivision, curve evaluation, and vector manipulation. Tasks like dividing surfaces into panels, extracting frames along curves, or performing Boolean operations on solids are handled through concise scripts, reducing visual clutter while maintaining parametric control.

Rhinoscriptsyntax plays a central role throughout the book, allowing users to interact directly with Rhino geometry through code while staying within a familiar environment. This reinforces the idea that learning Python in this context is not about becoming a software developer, but about enhancing design workflows with more structured logic.

```
27
28 # Solid union
29 pts = rs.CurvePoints(pol_0)
30 rad = rs.Distance(Cen, pts[0])
31 cyl_1 = rs.AddCylinder(Cen, -4*H, rad)
32 sol = rs.BooleanUnion([nut, cyl_1], True)
```

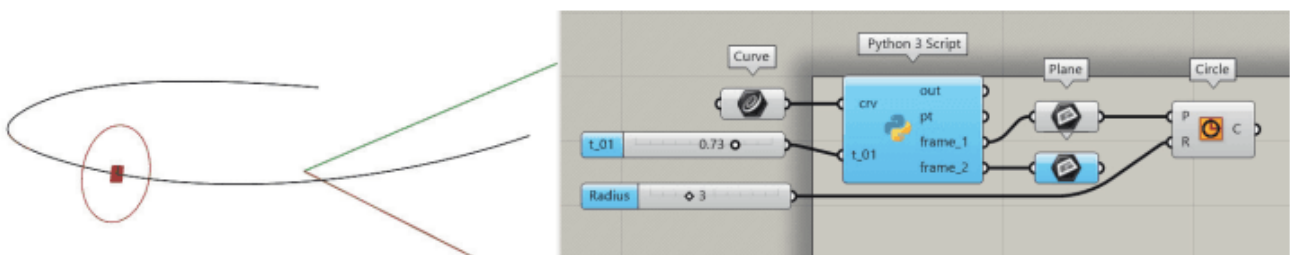


Working with solids in the Rhinoceros Python Editor without

using GH.

For users who prefer scripting over visual programming, the content can also be followed entirely within Rhino's Python editor. However, the real strength of the book lies in demonstrating the combined potential of Rhino, Grasshopper, and Python as a unified ecosystem.

Francisco Pérez Arribas, a professor at the Polytechnic University of Madrid with a background in naval architecture, brings a teaching approach grounded in years of experience with CAD and computational tools. The book reflects this pedagogical clarity, translating complex concepts into accessible, step-by-step processes aimed at designers rather than programmers.



```
1 import rhinoscriptsyntax as rs
2
3 ##### Geometric operations
4 t = rs.CurveParameter(crv, t_01) # Converts [0,1] to curve's domain
5 pt = rs.EvaluateCurve(crv, t) # Evaluates curve at t
6
7 # Associated vectors
8 frame_1 = rs.CurvePerpFrame(crv, t)
9 frame_2 = rs.CurveFrame(crv, t)
10
11 print (frame_1)
12
13 tan = frame_1.ZAxis # tan = frame_1[3]
14 nor = frame_1.XAxis # tan = frame_1[1]
15 bi = frame_1.YAxis # tan = frame_1[2]
```

Terminal

```
Origin=3.7424683999999955, -19.969481950000006, 8.134329265403613
XAxis=-0.10833894219966726, 0.9211406020617106, -0.3738484516972705,
YAxis=-0.22920466915056553, 0.34278473688988426, 0.911023514402846,
ZAxis=0.9673302917011525, 0.18438713455315175, 0.1739927911398658
```

Exploring the properties of curves inside the GH Python component and collaborate with other GH components.

Ultimately, this publication offers a clear pathway for designers to move from purely visual workflows into a more controlled and scalable

computational practice, without losing the intuitive nature that makes Grasshopper so powerful.

You can find *Learn Rhino Python without knowing Python* by Francisco Pérez Arribas and even download chapter examples in [English](#) and in [Spanish](#).