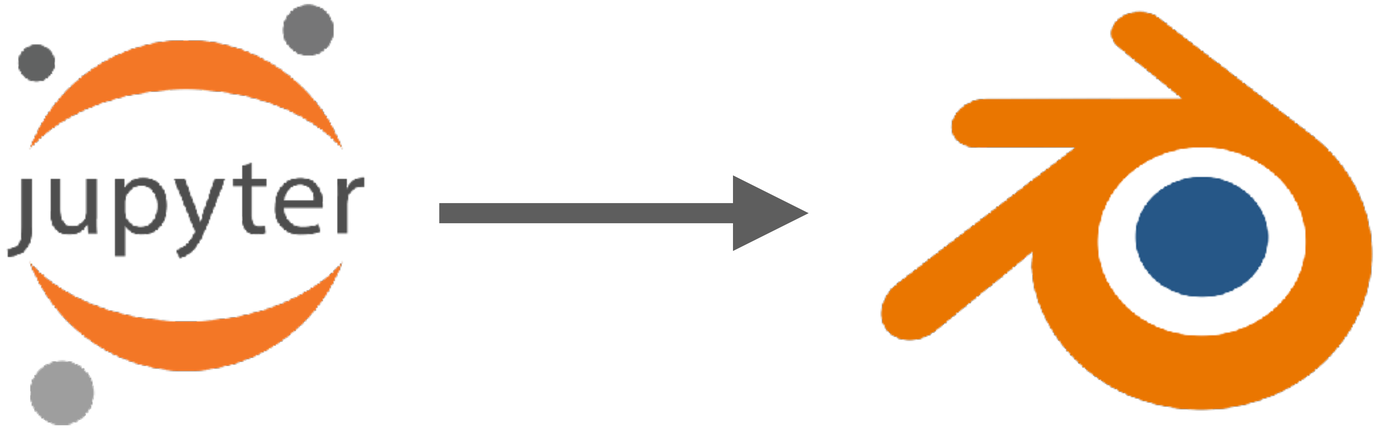
Python-Blender-Workshop

Jan-Hendrik Müller
12.11.2024



Ablaufplan

0: Intro

1: Objects in Blender

2: Using Python Packages

3: Using Meshes

4: Manipulating Data Using Pandas

5: Introduction to Geometry Nodes

6: Advanced Pandas Operations

7: Using Attributes in Shaders

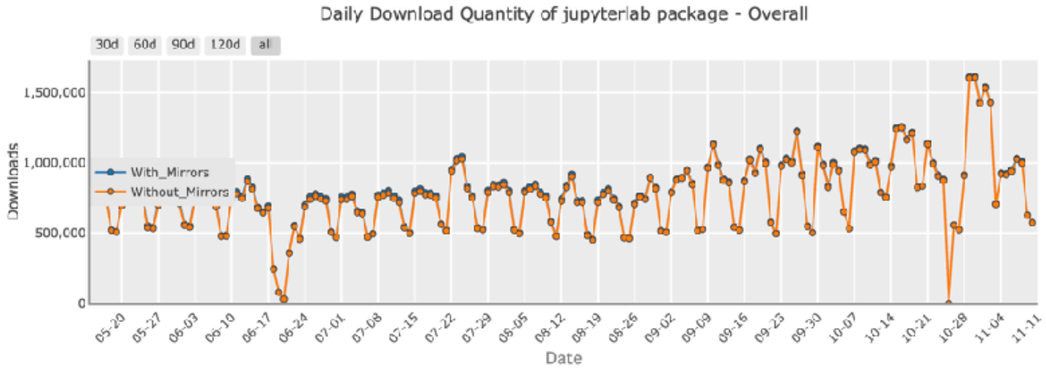
8: Exploring Climate Data

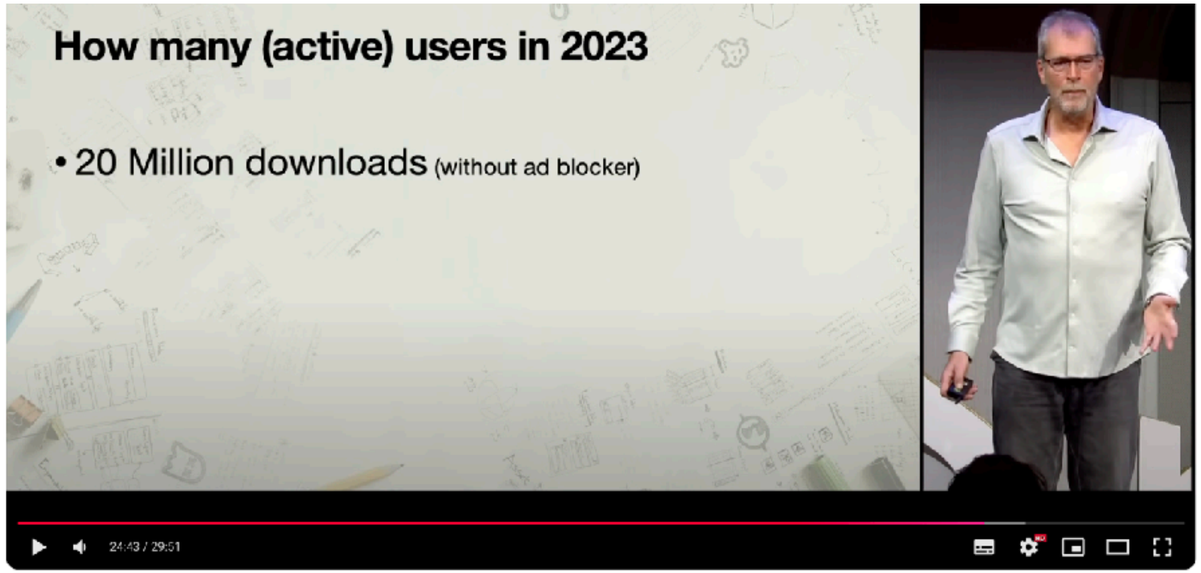
9: (Data From JavaScript)

https://github.com/Octoframes/bpy-workshop/blob/main/task1.ipynb


End Result

Downloads last month: 29,176,117



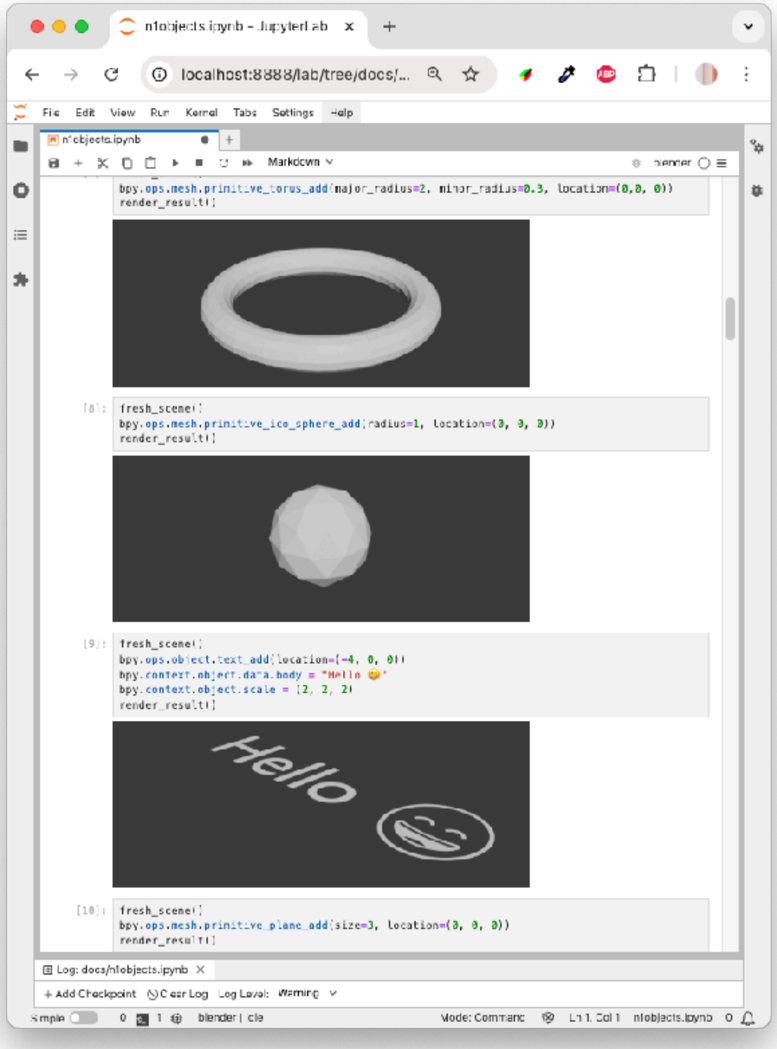


Keynote — Blender Conference 2024

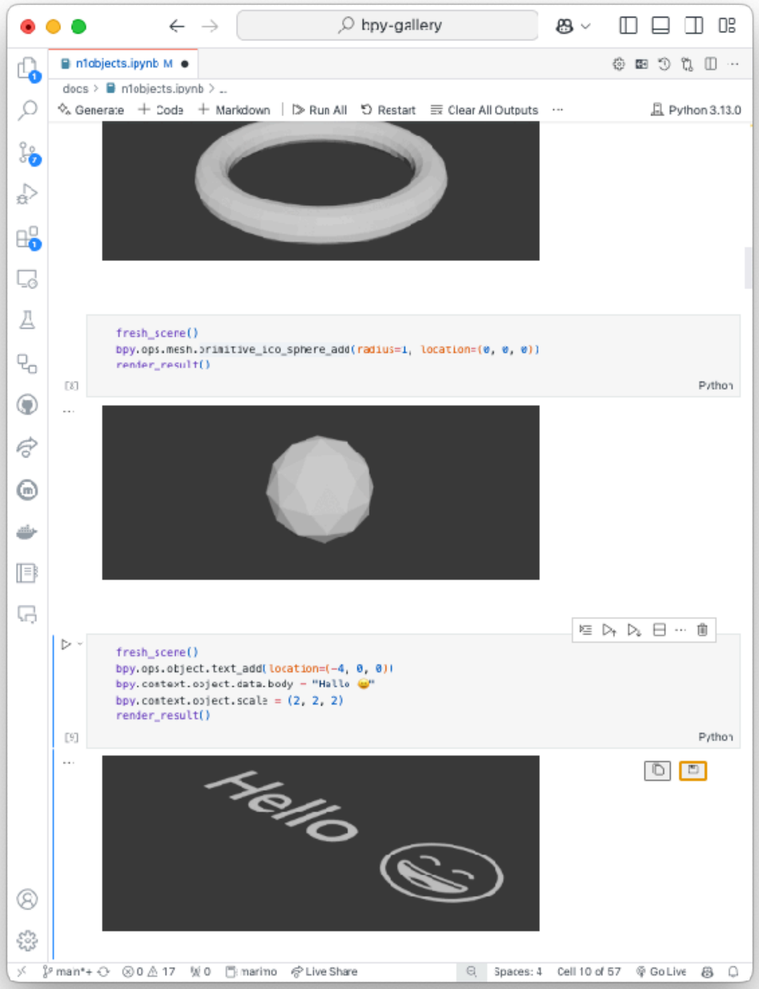
Blender
1,17 Mio. Abonnenten
Abonnieren
1405
Teilen
Herunterladen
Clip

Notebook Platformen

Jupyter Lab

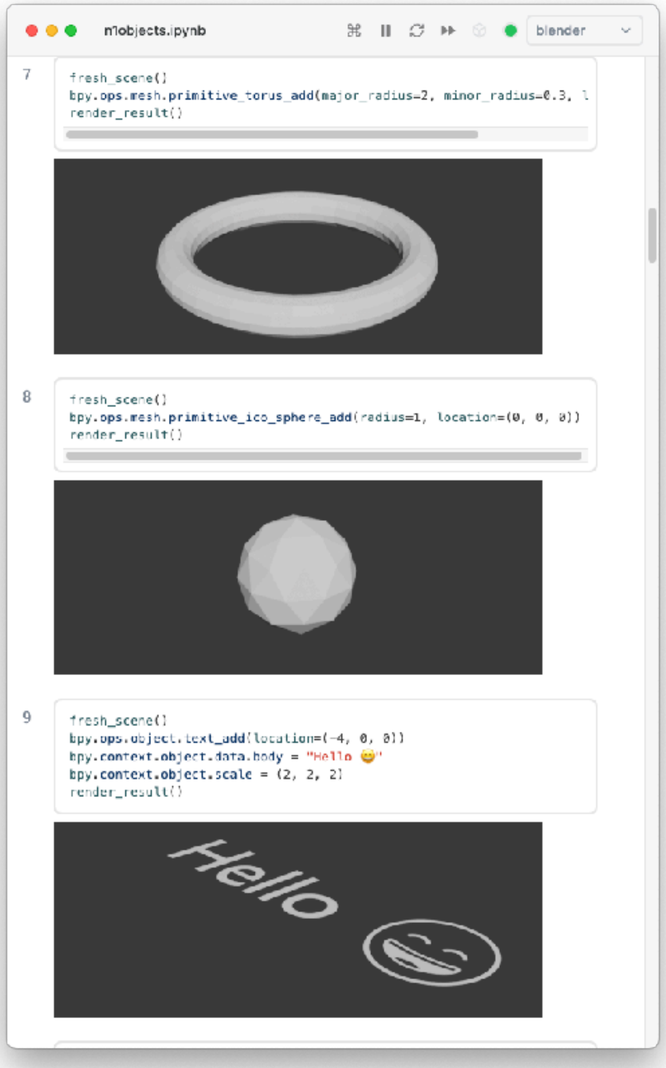


VS Code Notebook

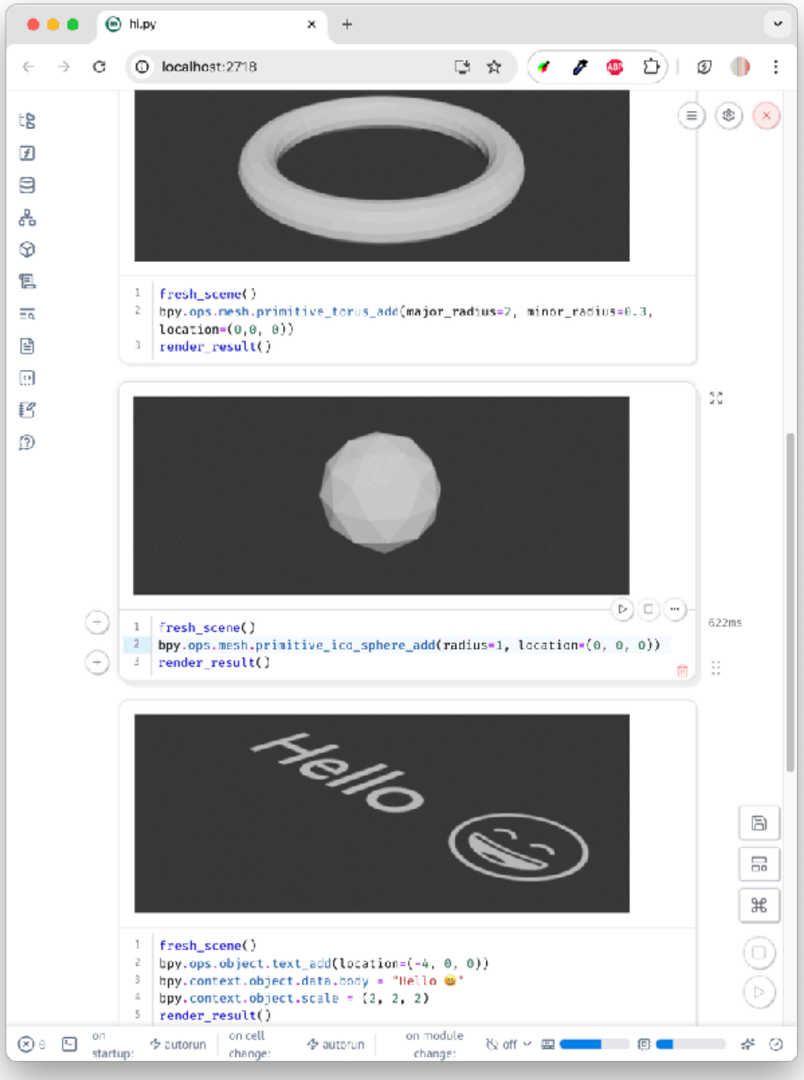


Notebook Platformen

Satyrn



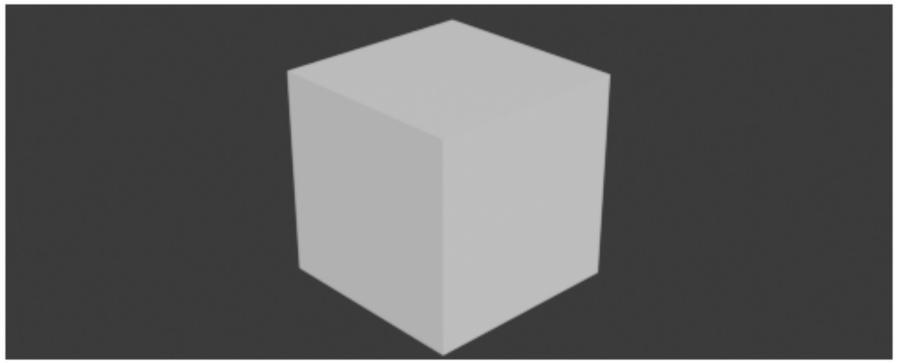
Marimo



Headless mode

Ohne Blender GUI

```
fresh_scene()
bpy.ops.mesh.primitive_cube_add(size=2, location=(0, 0, 0))
render_result()
```



GUI mode

Blender GUI an der Seite

```
fresh_scene()
bpy.ops.mesh.primitive_monkey_add(size=2.5, l
```

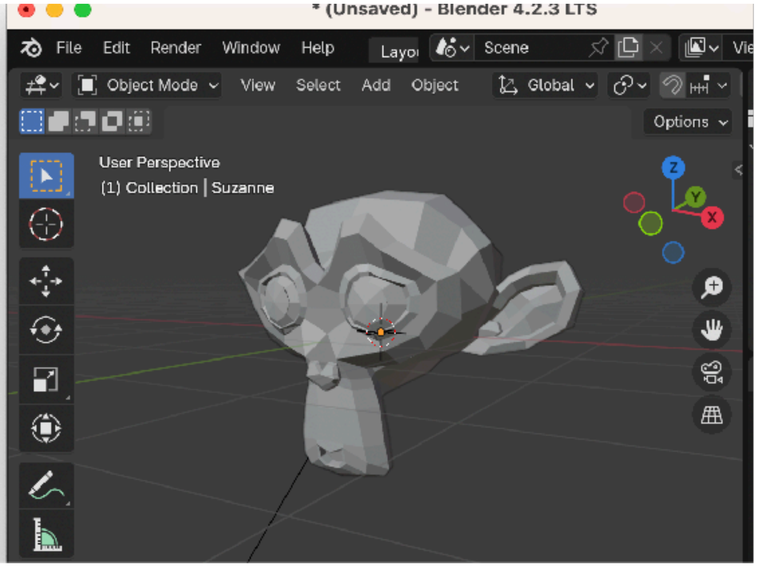
✓ 0.0s Python

```
{'FINISHED'}
```

Python

Python

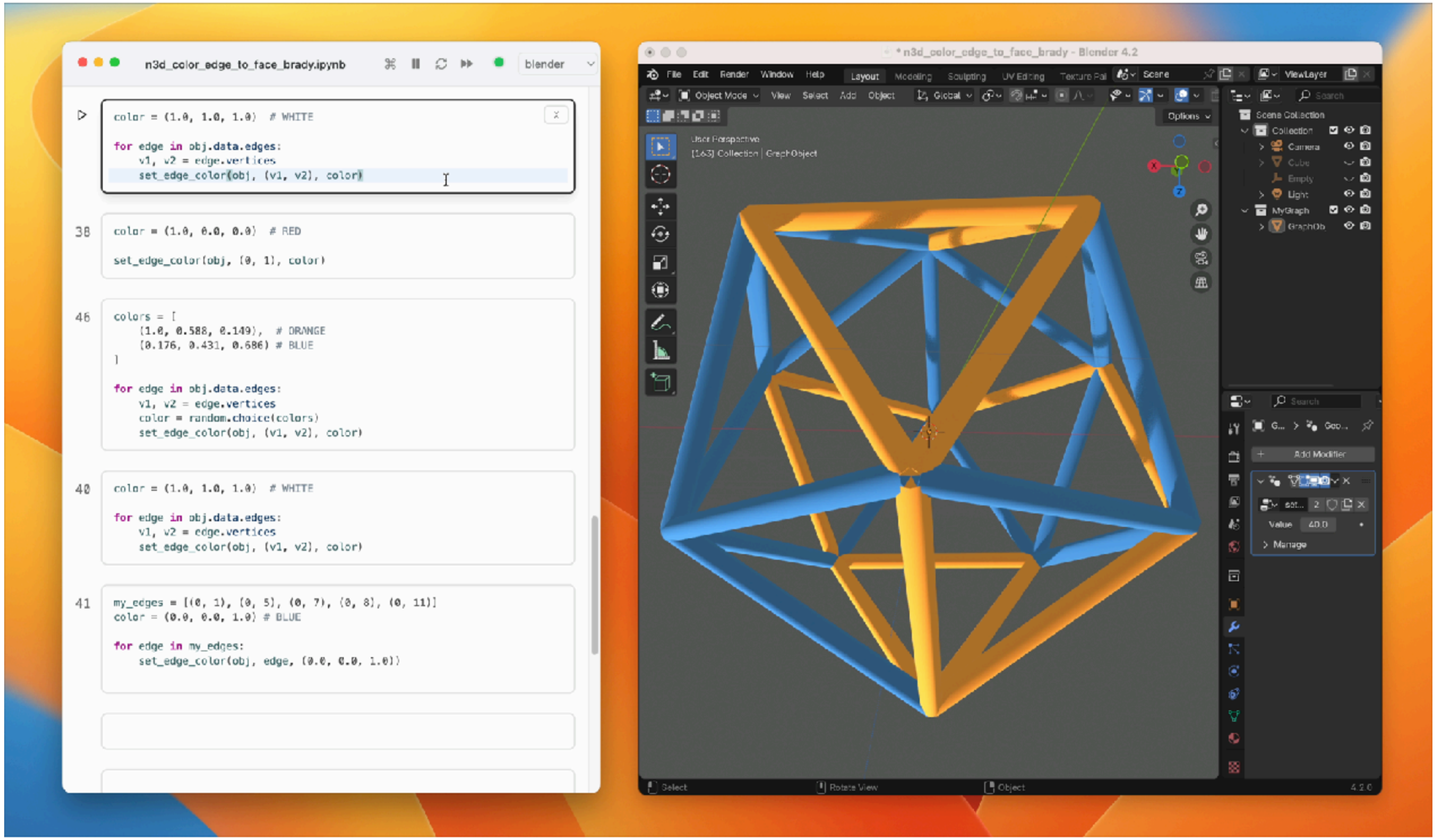
Press ⌘ I to ask GitHub Copilot to do some



Notebook Pros & Cons

+ Interaktives Coding

+ Einfaches Debugging

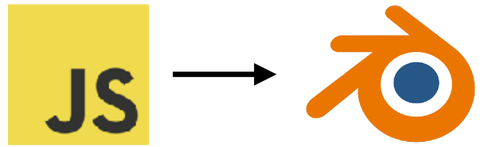


Notebook Pros & Cons

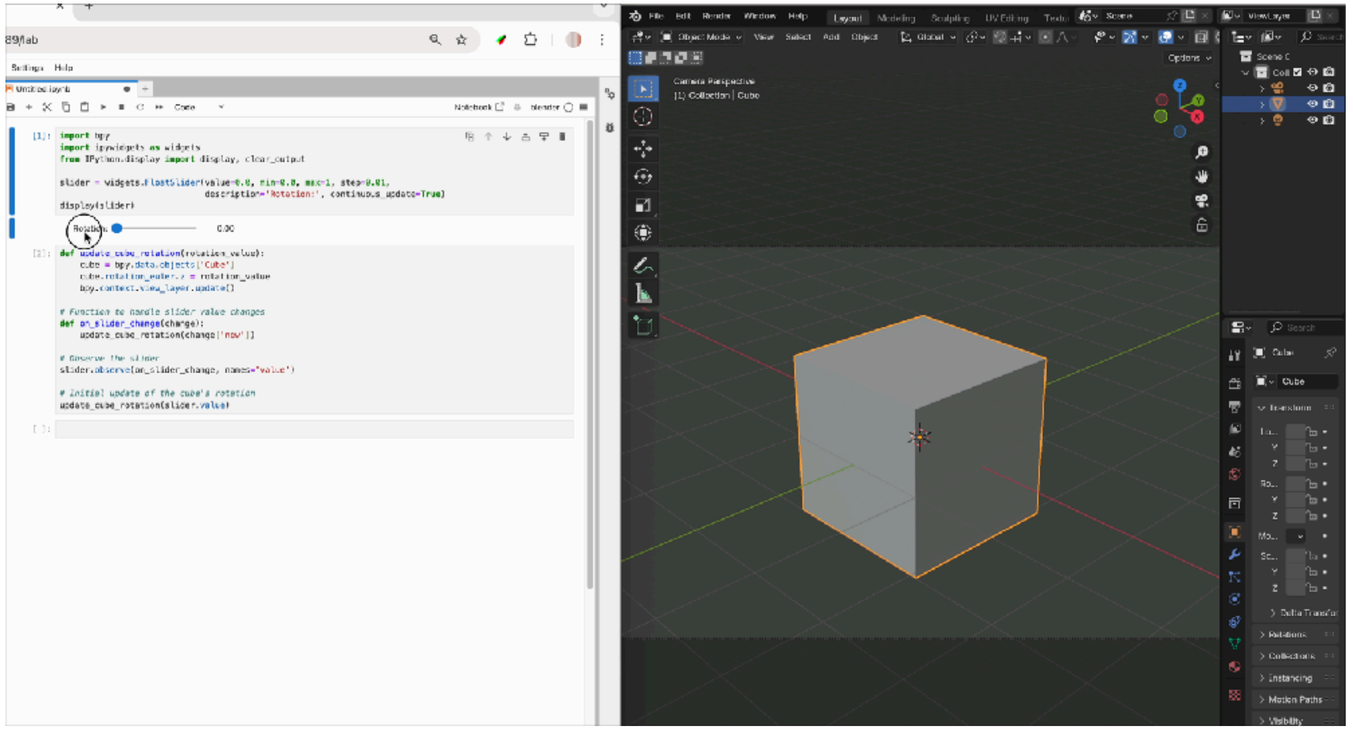
+ Dokumentation und Code in einem

https://kolibril13.github.io/bpy-gallery/n1objects/

Notebook Pros & Cons

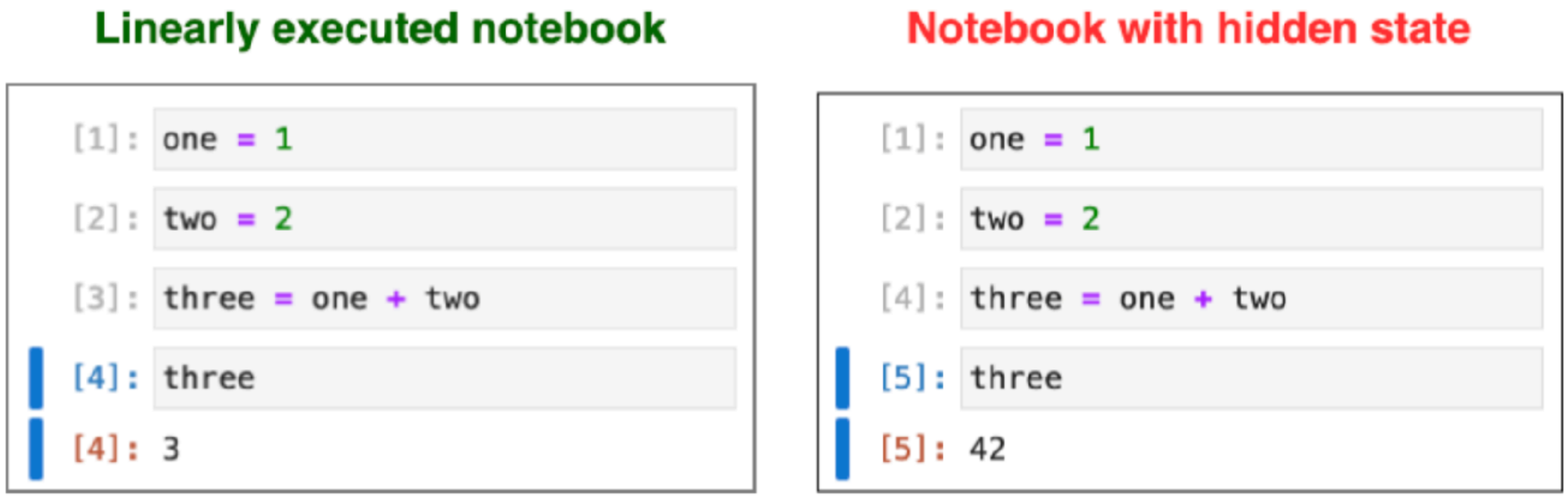


+ Integration von Widgets



Notebook Pros & Cons

- Hidden state



https://ploomber.io/images/blog/nbs-myths/hidden-state.png

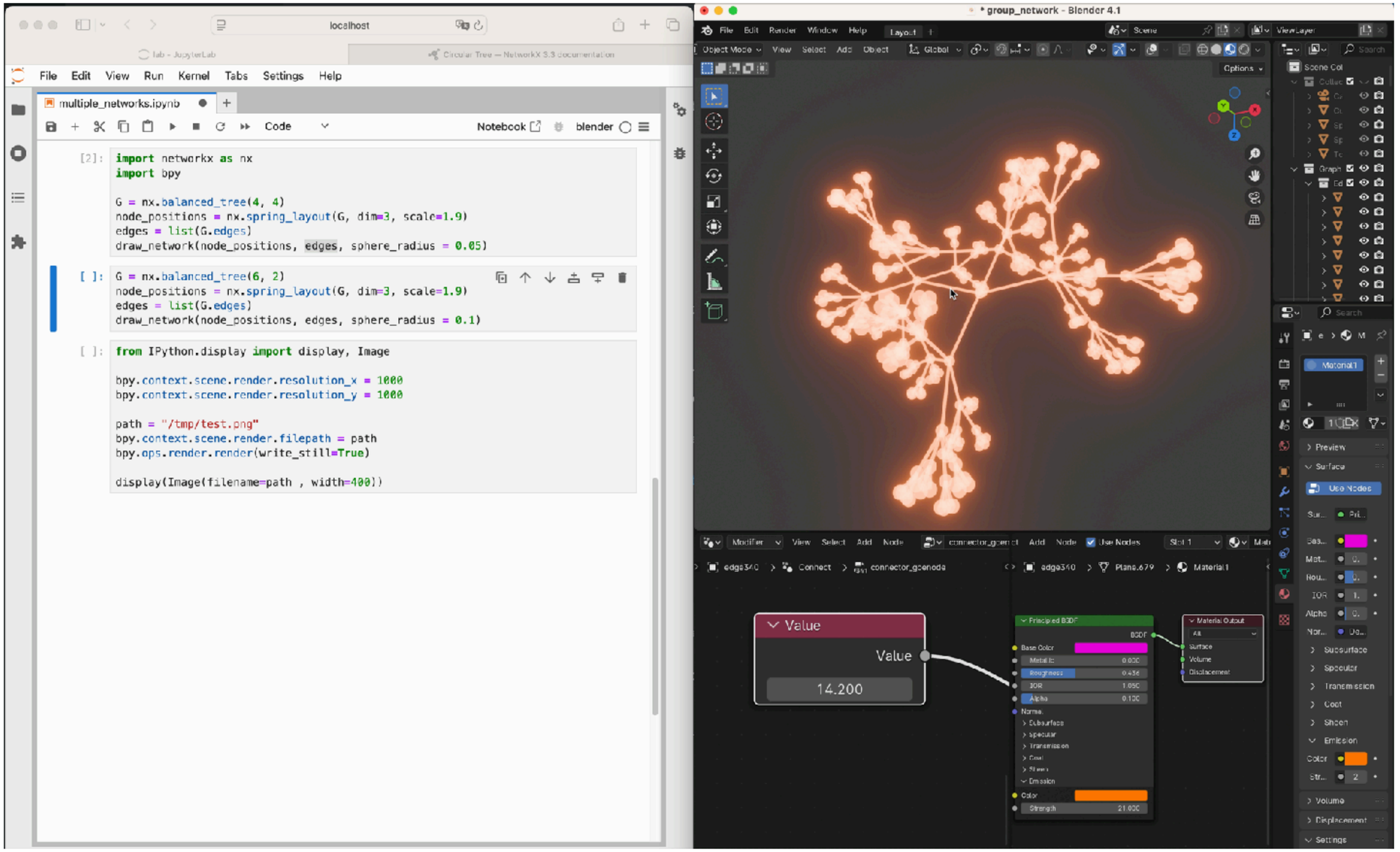
Notebook Pros & Cons

- Code in Notebooks schnell unübersichtlich bei großer Codbasis

Notebook Pros & Cons

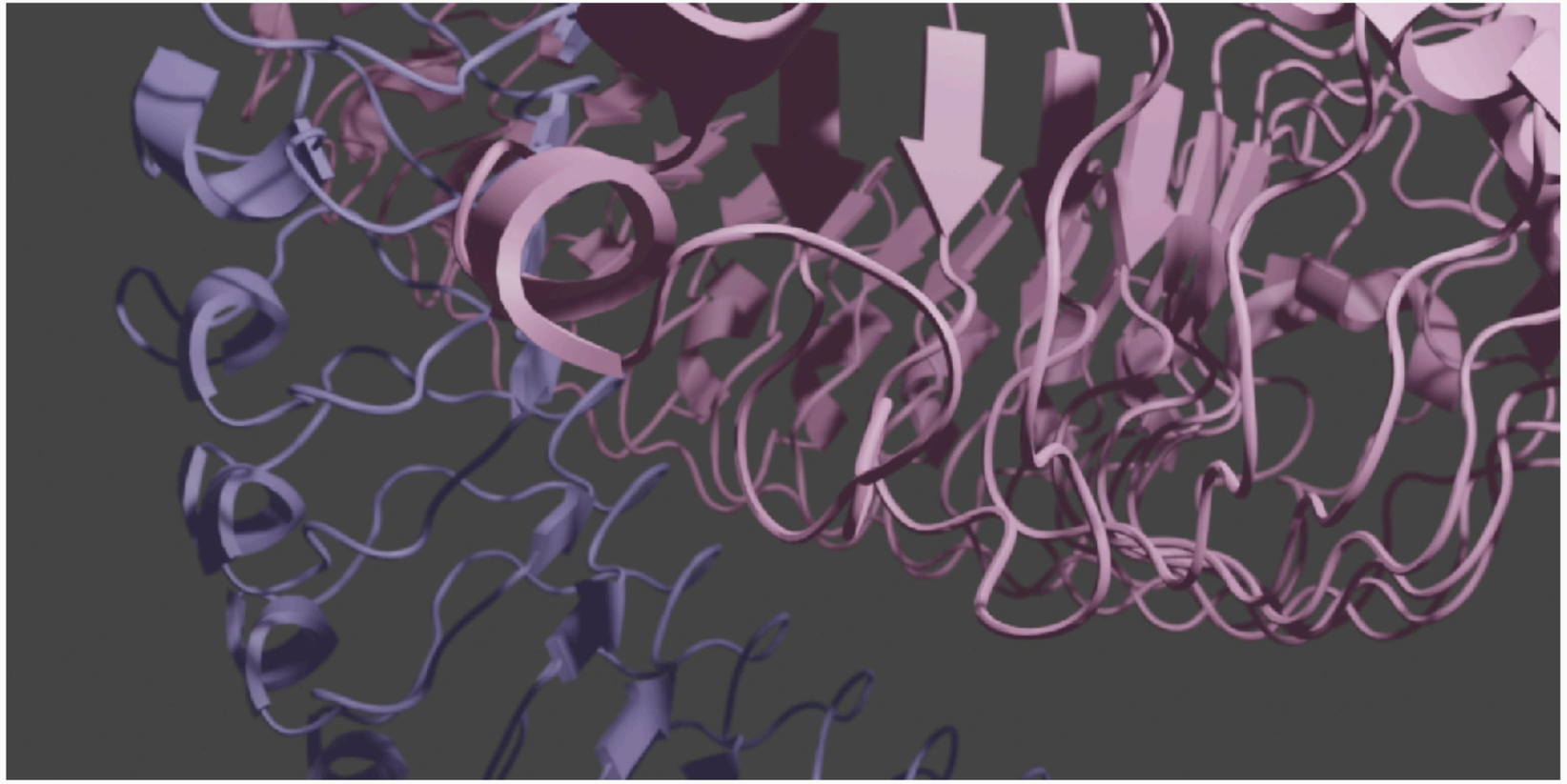
-> Gut geeignet, um Ideen auszuprobieren!

Python Modul laden

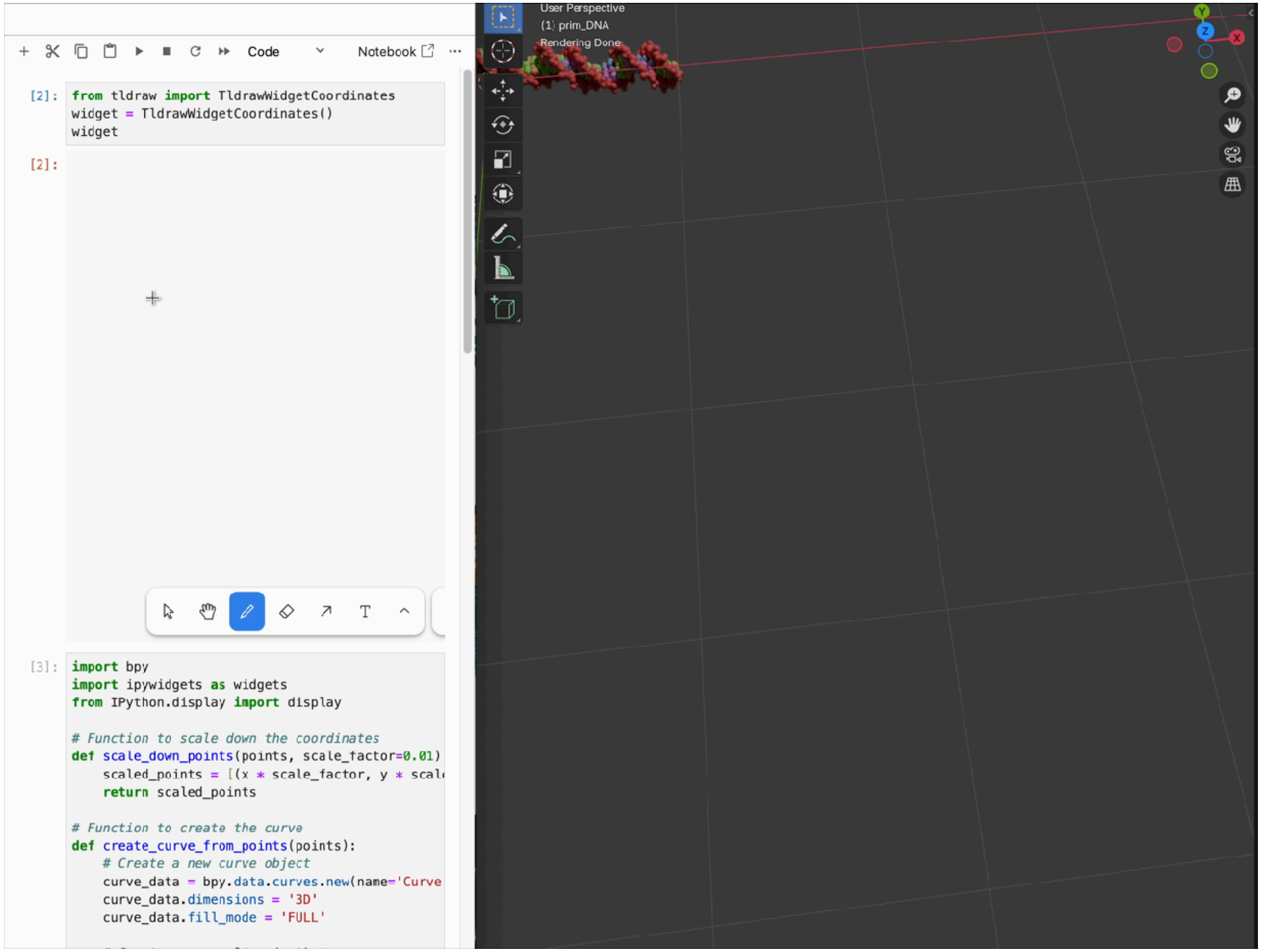


Python Modul laden

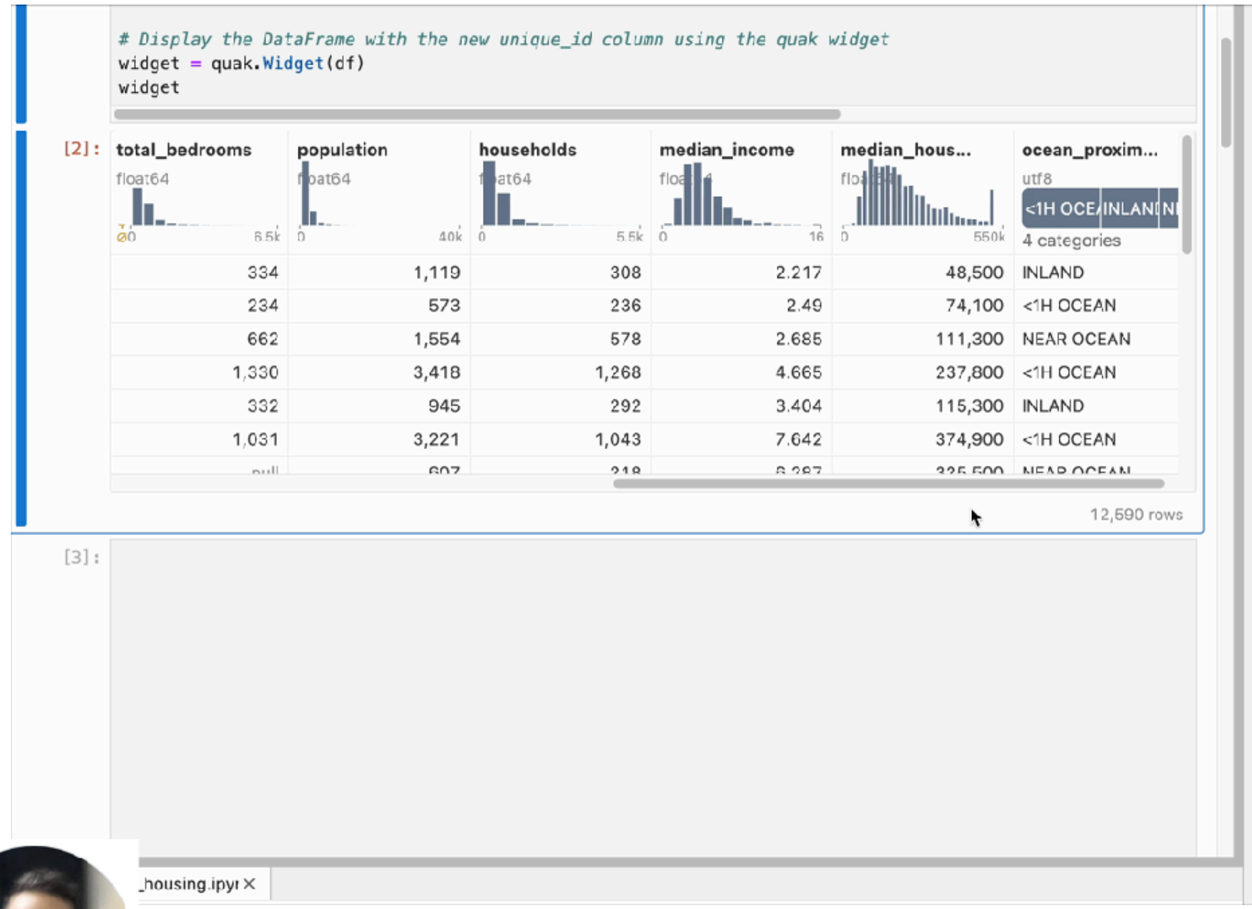
[2]:
```
import molecularnodes as mn
molecule = mn.load.molecule_rcsb("7TYG", starting_style="cartoon", center_molecule=True)
molecule.location = (0, 2, 1)
scale = 8
molecule.scale = (scale,scale,scale)

render_image()
```



Blender & Whiteboard



Blender & interaktive Tabellen



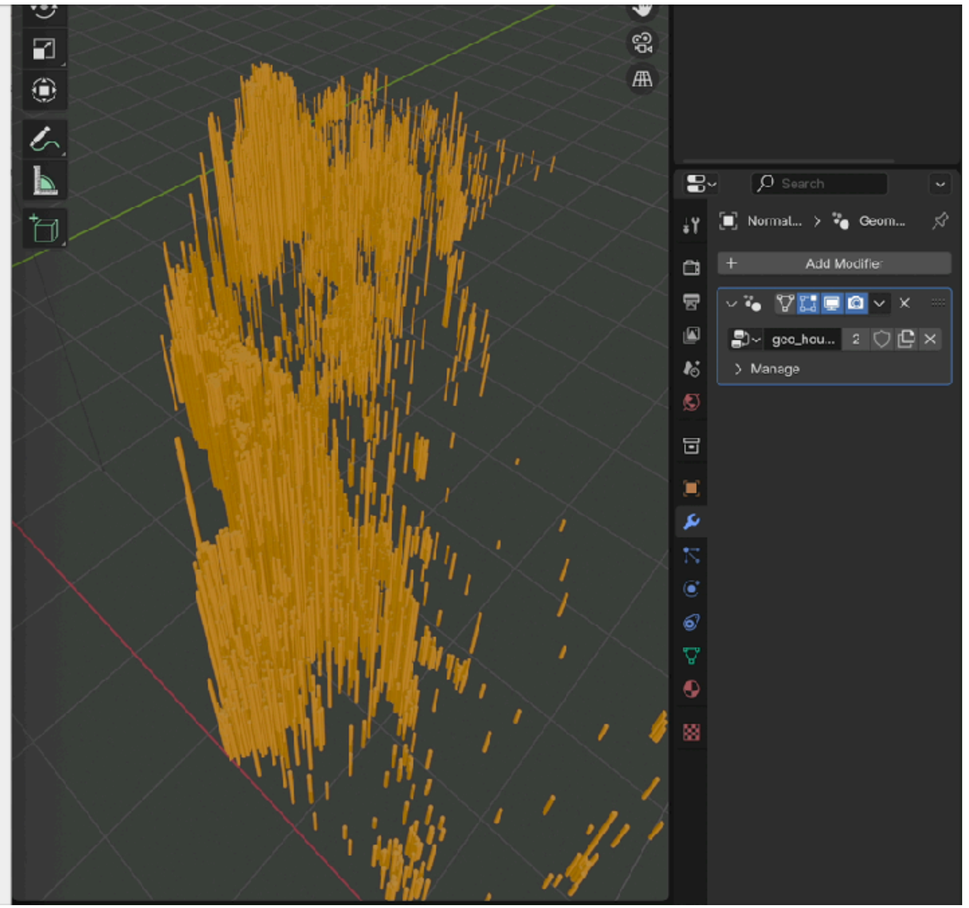
```
# Display the DataFrame with the new unique_id column using the quak widget
widget = quak.Widget(df)
widget
```

[2]:

total_bedrooms	population	households	median_income	median_hous...	ocean_proxim...
float64	float64	float64	float64	float64	utf8
					4 categories
334	1,119	308	2.217	48,500	INLAND
234	573	236	2.49	74,100	<1H OCEAN
662	1,554	578	2.685	111,300	NEAR OCEAN
1,330	3,418	1,268	4.665	237,800	<1H OCEAN
332	945	292	3.404	115,300	INLAND
1,031	3,221	1,043	7.642	374,900	<1H OCEAN

12,590 rows

[3]:



Trevor Manz
manzt · he,him

https://github.com/manzt/quak

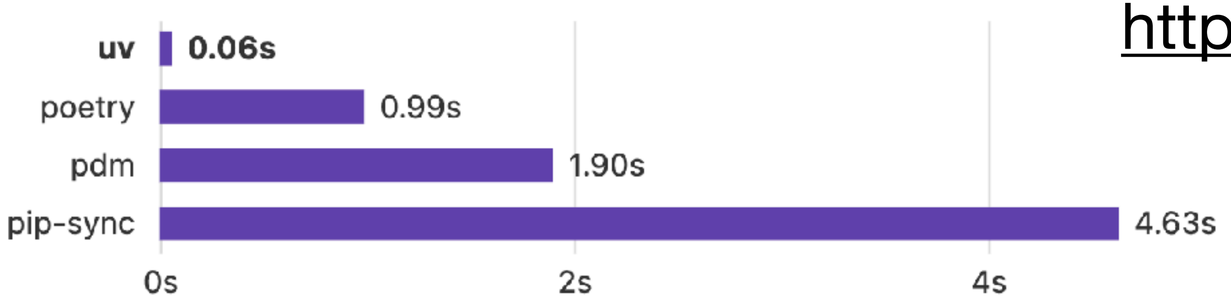
Installation

Full instructions at

https://kolibril13.github.io/bpy-gallery/n0getting_started/

Package manager UV

An extremely fast Python package and project manager, written in Rust.



https://docs.astral.sh/uv/

Install with

```
curl -LsSf https://astral.sh/uv/install.sh | sh
```

Installation mit uv

blender notebook setup

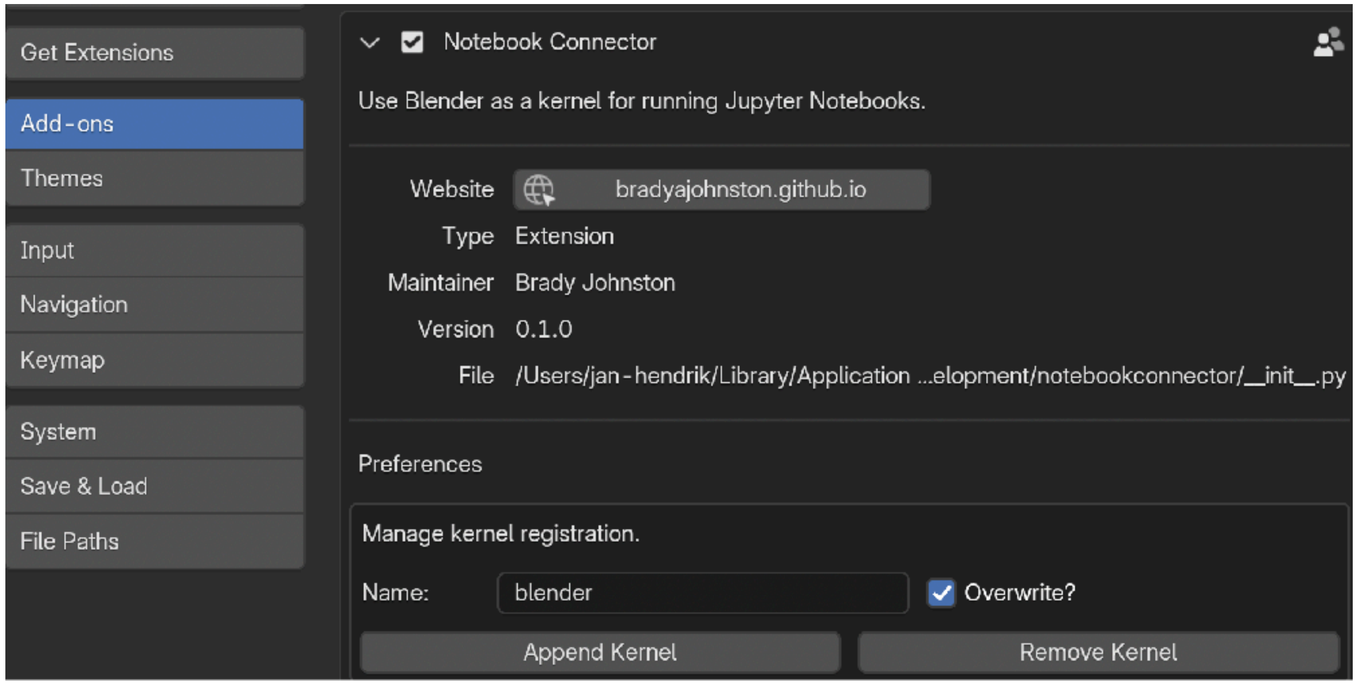
blender_notebook.txt

```
1  # Install Blender from https://www.blender.org/download/
2  # Install uv
3  curl -LsSf https://astral.sh/uv/install.sh | sh
4  mkdir blender_python_workshop && cd blender_python_workshop
5  uv venv --python 3.11
6  uv pip install jupyterlab blender_notebook
7  uv run blender_notebook install --blender-exec="/Applications/Blender.app/Contents/MacOS/Blender"
8  uv run jupyter lab
```

Write Preview

Leave a comment

In Zukunft: Notebook von Blender aus starten



Developer:

Brady Johnston
BradyAJohnston · he/him

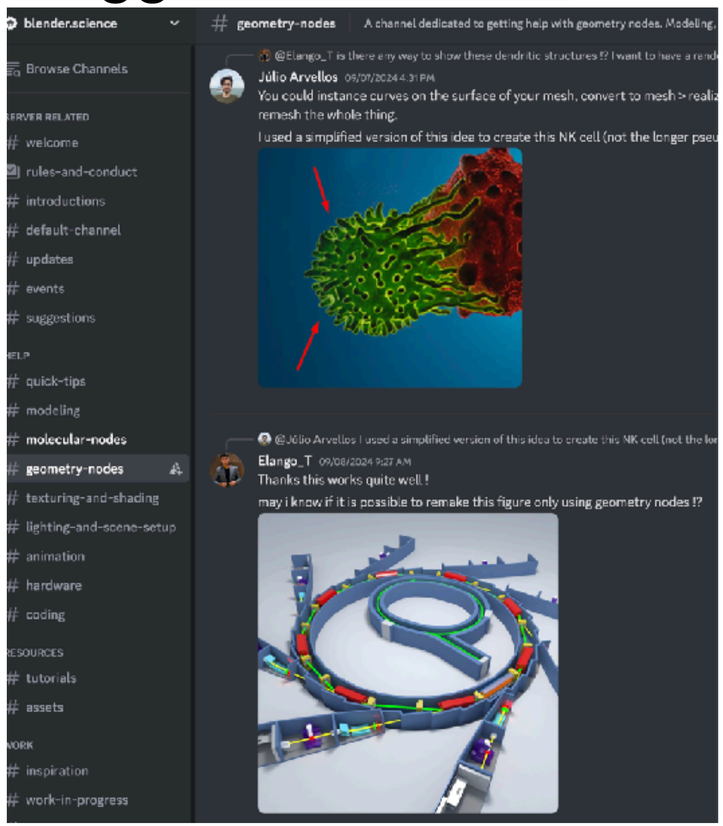
Jan-Hendrik Müller

Sofus Albert Høgsbro Rose

Austausch Blender+Python

Science Blender Discord:

https://discord.gg/k2Gd2cb4Kk



Kontakt:

https://twitter.com/kolibril13

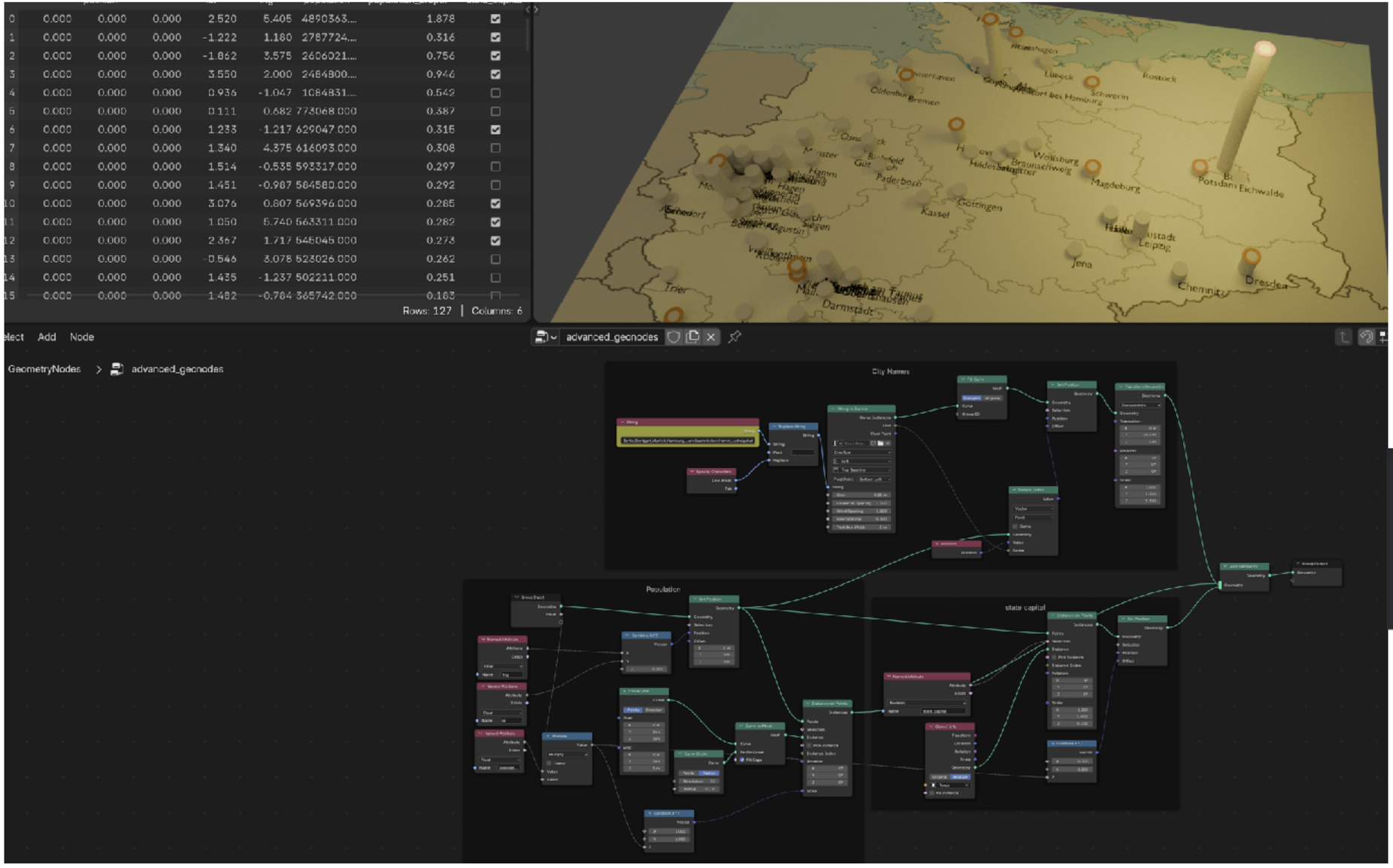
https://www.linkedin.com/in/jan-hendrik-müller-765014209/

https://bsky.app/profile/kolibril13.bsky.social

Als nächstes: Hands-On Session

https://github.com/Octoframes/bpy-workshop/blob/main/task1.ipynb

String Data




```python
obj = bpy.context.active_object
node_group = obj.modifiers['GeometryNodes'].node_group.nodes
nodes = node_group

string_node = nodes.new('FunctionNodeInputString')

string_node.use_custom_color = True
string_node.color = (0.6, 0.6, 0.2)  # RGB for a much darker yellow
string_node.width = 400

city_names = df_adjusted2.city.to_list()
new_string = ",".join(city_names)
string_node.string = new_string
```