

# Exercise 6 - napari

Please open and follow instructions for Exercise 6:

- file: **E06\_napari\_env.md**
- [link](#)

# napari Viewer

We've been using `matplotlib` to view images — ideal for creating static, publication-ready plots. `Stackview` adds basic level of interactivity for exploring multidimensional images but remains limited.

**napari:** is an open-source, Python-based image viewer designed for scientists. It excels at being fast, interactive, and extensible.

## The Core Concept: **Layers**

- You can have an image layer, a labels layer (for segmentation masks), a points layer, a shapes layer, etc.
- You can add, remove, reorder, and change the properties (color, opacity, contrast) of each layer independently.

# The napari GUI: A Quick Tour

Let's launch Napari (by typing `napari` in terminal/prompt) and explore the interface.

1. **Canvas:** The main window where your image is displayed.
2. **Layer list (Bottom Left):** You can add, select, reorder, and toggle the visibility of layers here.
3. **Layer controls (Top Left):** This panel is **context-aware**. Its contents change depending on which layer you have active (selected). For an image layer, you'll see controls for colormap, contrast, and opacity. For a points layer, you'll see controls for point size and color.
4. **Dimension sliders (Bottom):** If you load a 3D or more dimensional image, sliders will automatically appear here to let you scroll through the dimensions.

# Exploring Plugins

**napari** has a vast plugin ecosystem. The community has built hundreds of tools that you can install directly through the Napari GUI.

## How to Install:

- *From terminal:* You can install plugins just as any other package with conda/mamba/pip.
- *From Napari GUI:* Go to **Plugins > Install/Uninstall Plugins...**

### Install example plugin: **napari-skimage-regionprops**

- This is a GUI-based way to perform the same measurements as we did in code with **measure.regionprops\_table**.

After plugin installation, you need to **restart napari**.

# Exercise Session

1. **Open Napari GUI**
2. **Load the image:** Drag and drop `nuclei_3d.tif` into napari. Then rename the image layer to **nuclei**.
3. **Adjust visualization:** Change **contrast** and **colormap** as preferred.
4. **Calibrate pixel size**
  - Switch to **3D view** and rotate the image - looks nice, but pixel size is not yet calibrated.
  - Open the **console** and enter the calibration commands:
    - `layer = viewer.layers['nuclei']`
    - `layer.scale = (2, 1, 1)`
  - Return to 2D view

**6. Create a labels layer** - Draw around a nucleus using the annotation tools.

Save the layer: **File > Save selected layers** > save it as **labels.tif**

**7. Convert image to labels layer**

- Open the image **nuclei\_3d-L.tif** and rename it to **labels**.
- Right-click on the new layer and select **Convert to labels**.
- Calibrate it with the same parameters:
  - `layer2 = viewer.layers['labels']`
  - `layer2.scale = (2, 1, 1)`

**8. Explore different views** - try **3D view** and **grid view**

switch on/off visibility of layers

**9. Measure object properties with installed plugin**

- **Tools > Measurement Tables > Object Feature / Properties**
- Measure **size** and **intensity** of objects.