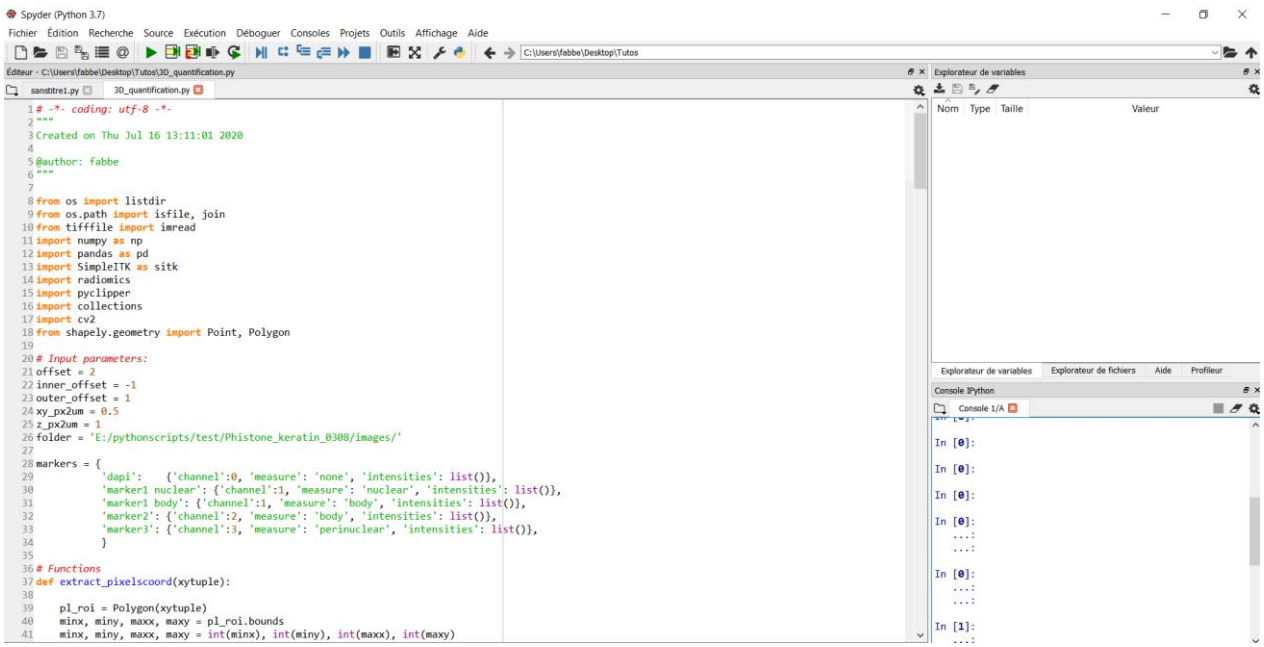
Tutorial:

How to generate quantifications from my 3D nuclei segmentation

If you haven't installed Python yet please first go to the tutorial *how to install python for StarDist*

1) Open Spyder with the environment you have created

2) Load the script 3D_quantification.py, you should see something like this:




```python
# -*- coding: utf-8 -*-
"""
Created on Thu Jul 16 13:11:01 2020

@author: fabbe
"""

from os import listdir
from os.path import isfile, join
from tifffile import imread
import numpy as np
import pandas as pd
import SimpleITK as sitk
import radiomics
import pyclipper
import collections
import cv2
from shapely.geometry import Point, Polygon

# Input parameters:
offset = 2
inner_offset = -1
outer_offset = 1
xy_px2um = 0.5
z_px2um = 1
folder = 'E:/pythonscripts/test/Phistone_keratin_0308/images/'

markers = {
          'dapi':    {'channel':0, 'measure': 'none', 'intensities': list()},
          'marker1 nuclear': {'channel':1, 'measure': 'nuclear', 'intensities': list()},
          'marker1 body': {'channel':1, 'measure': 'body', 'intensities': list()},
          'marker2': {'channel':2, 'measure': 'body', 'intensities': list()},
          'marker3': {'channel':3, 'measure': 'perinuclear', 'intensities': list()},
          }

# Functions
def extract_pixelscoord(xytuple):

    pl_roi = Polygon(xytuple)
    minx, miny, maxx, maxy = pl_roi.bounds
    minx, miny, maxx, maxy = int(minx), int(miny), int(maxx), int(maxy)
```

We only need to edit the input parameter code block:

```python
# Input parameters:
offset = 2
inner_offset = -1
outer_offset = 1
xy_px2um = 0.5
z_px2um = 1
folder = 'E:/pythonscripts/test/Phistone_keratin_0308/images/'

markers = {
          'dapi':    {'channel':0, 'measure': 'none', 'intensities': list()},
          'marker1 nuclear': {'channel':1, 'measure': 'nuclear', 'intensities': list()},
          'marker1 body': {'channel':1, 'measure': 'body', 'intensities': list()},
          'marker2': {'channel':2, 'measure': 'body', 'intensities': list()},
          'marker3': {'channel':3, 'measure': 'perinuclear', 'intensities': list()},
          }
```

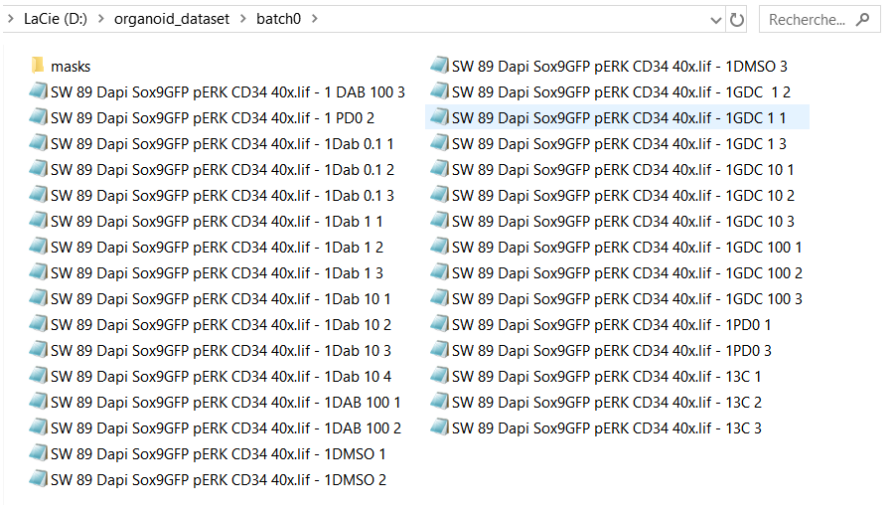
3) Edit the offset parameter. The offset parameter control how much pixels outside the nuclei you want to account to perform intensities measurements. This parameter is in pixels.

If you don't need this parameter for your quantification (no measurement in the cell body) just leave things as they are, it will have no impact.

4) Edit the inner and outer offset parameter. Those parameters control both how much pixels outside and inside the nuclei you want to account to perform intensities measurements. This parameter is useful to quantify intensities of perinuclear proteins. Again if you don't need those parameters (no measurement in the perinuclear area) just leave things as they are, it will have no impact.

5) Edit the xy_px2um and the z_px2um parameters. Those 2 parameters are the conversion factors of pixel to micron respectively for xy and z. If you don't know them, you can often find them in your images metadata (Image/Show info in Image J). If your dataset have images with different resolutions, you should consider running the script independently for each batch of images.

6) Edit the folder path containing your images and masks, you should have a similar structure to this:



Always use the '/' separator when specifying a folder Path and ends your path with a separator as shown in the example.

7) Edit the markers block code:

```
markers = {
            'dapi':    {'channel':0, 'measure': 'none', 'intensities': list()},
            'marker1 nuclear': {'channel':1, 'measure': 'nuclear', 'intensities': list()},
            'marker1 body': {'channel':1, 'measure': 'body', 'intensities': list()},
            'marker2': {'channel':2, 'measure': 'body', 'intensities': list()},
            'marker3': {'channel':3, 'measure': 'perinuclear', 'intensities': list()},
            }
```

Add or remove lines to specify in which cell compartment to perform intensity measurements for each marker.

How to proceed?

- On each line, the left part is the marker name, this is what will appear on the output excel file as your column names. You can set any name you want for your markers except for the 'dapi'. The DAPI should always be named 'dapi'. If you need to measure the intensity for several cell compartment for one marker, please add multiple lines for the same marker but don't repeat the name multiple time, as shown in the example below:

```
'marker1 nuclear': {'channel':1, 'measure': 'nuclear', 'intensities': list()},
'marker1 body': {'channel':1, 'measure': 'body', 'intensities': list()},
```

- Then set the 'channel' option. It is the position of your marker in your image. In Python the first index starts at 0. If your DAPI channel is the first channel in imageJ, set 0 in Python.

- Finally, set the 'measure' option to set which cell compartment you want to measure for this marker. There are 4 options:

    - 'none': don't measure anything, it is only useful for the DAPI marker if you don't want to measure dapi intensity. For other markers just delete or comment the line.
    - 'nuclear': measure intensity inside the nucleus
    - 'body': measure intensity inside the cell body
    - 'perinuclear': measure intensity around the nucleus

8) Run the code and once finished you should see an excel file generated in your input folder.