**Image Registration using pystackreg library in Python :**

→ pyStackReg is used to align (register) one or more images to a common reference image, as is required usually in time-resolved fluorescence or wide-field microscopy.

→ pyStackReg provides the following five types of distortion:

* translation
* rigid body (translation + rotation)
* scaled rotation (translation + rotation + scaling)
* affine (translation + rotation + scaling + shearing)
* bilinear (non-linear transformation; does not preserve straight lines)

→ Documentation link of pystackreg : <https://pystackreg.readthedocs.io/>

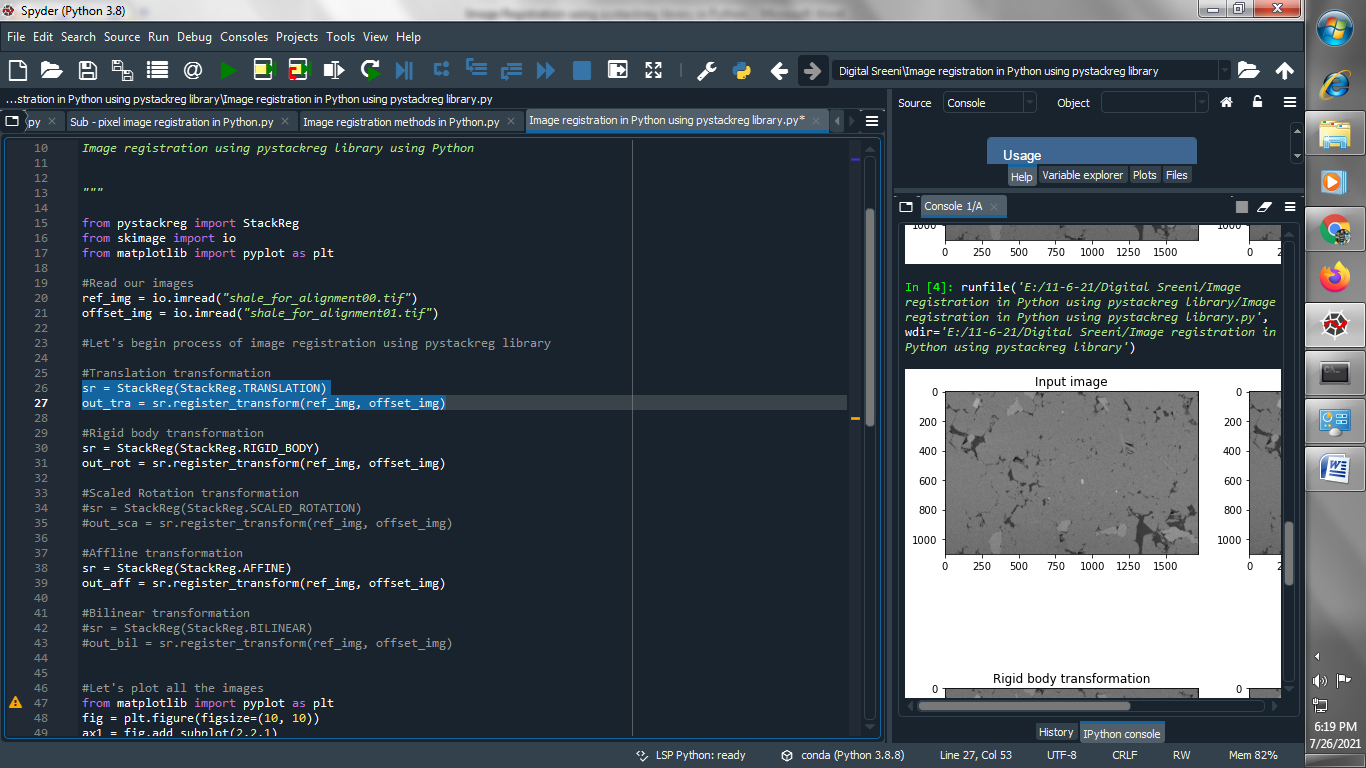
→ It uses functionality from ImageJ.

**(1) Read our images :**

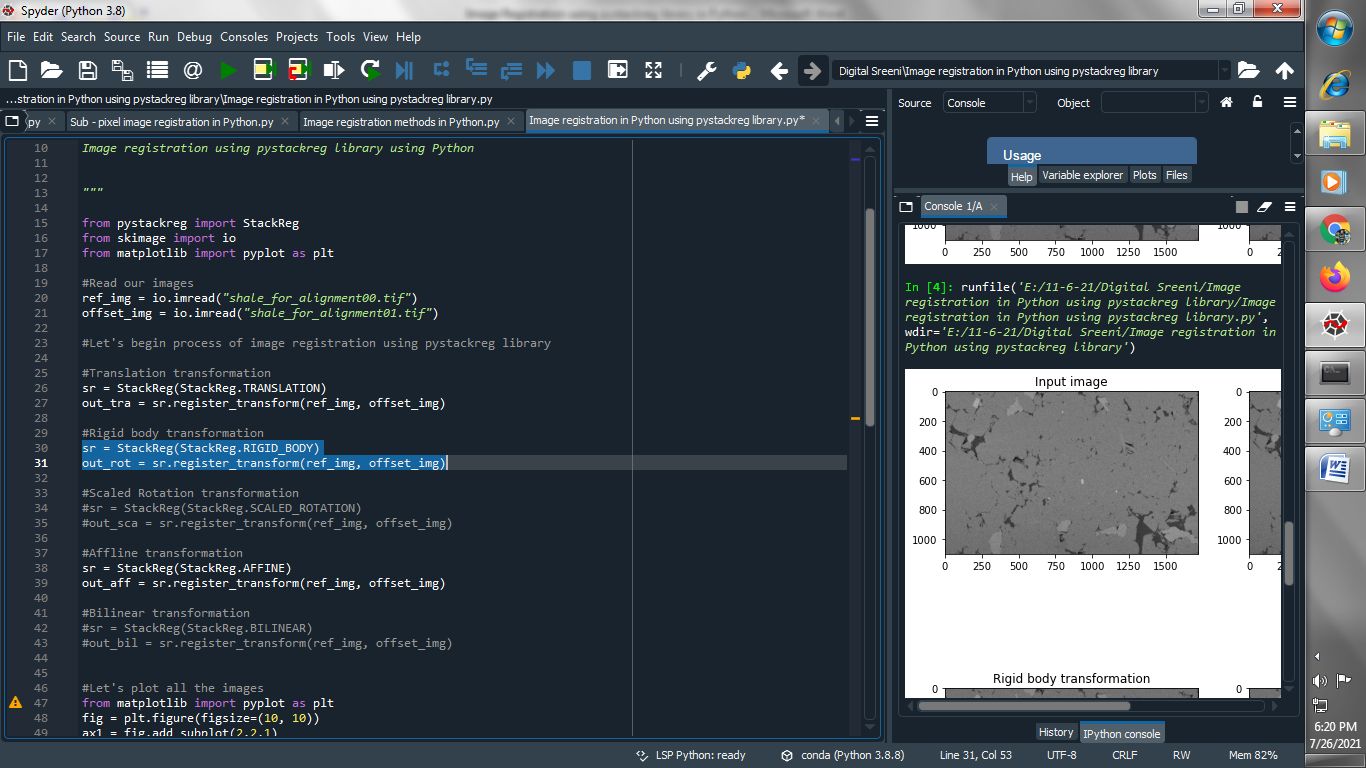
****

**→ Let's begin process of image registration using pystackreg library**

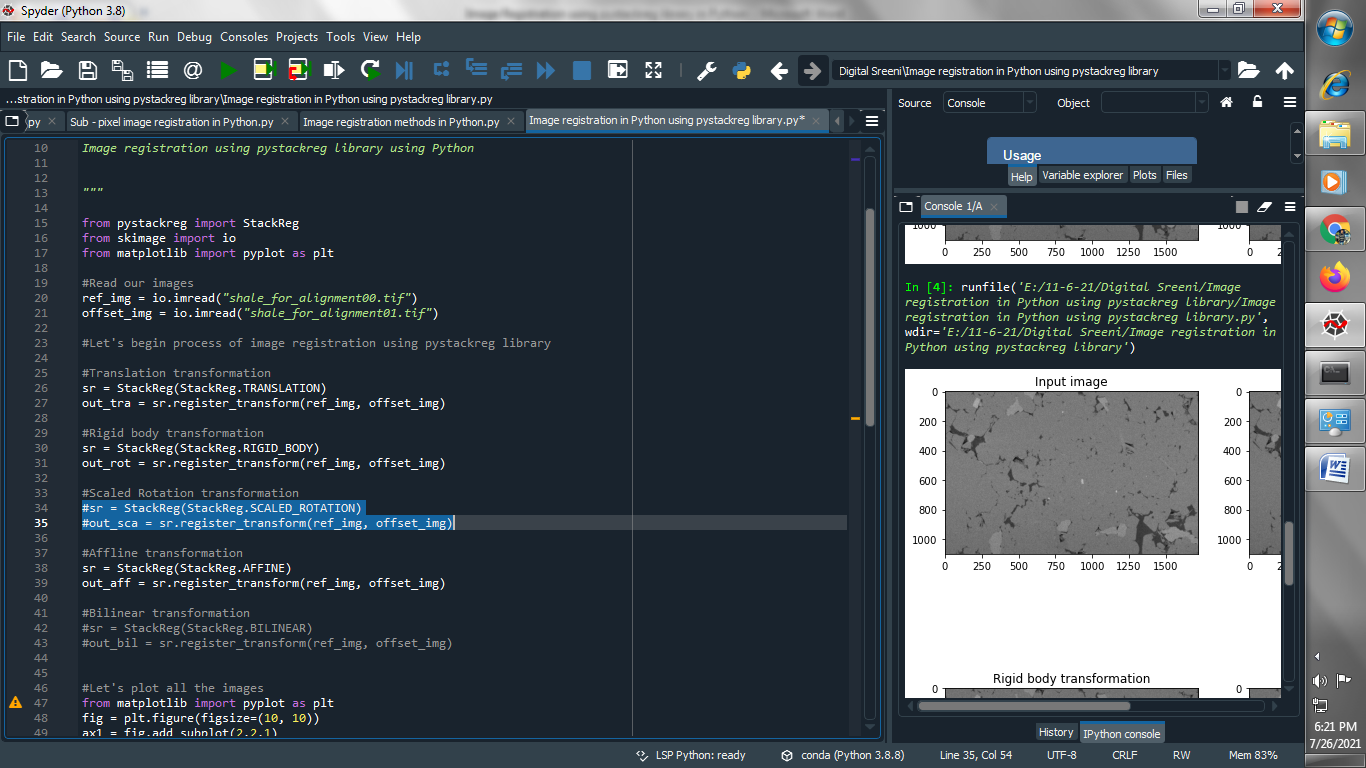
**(2) Apply Translation transformation :**

****

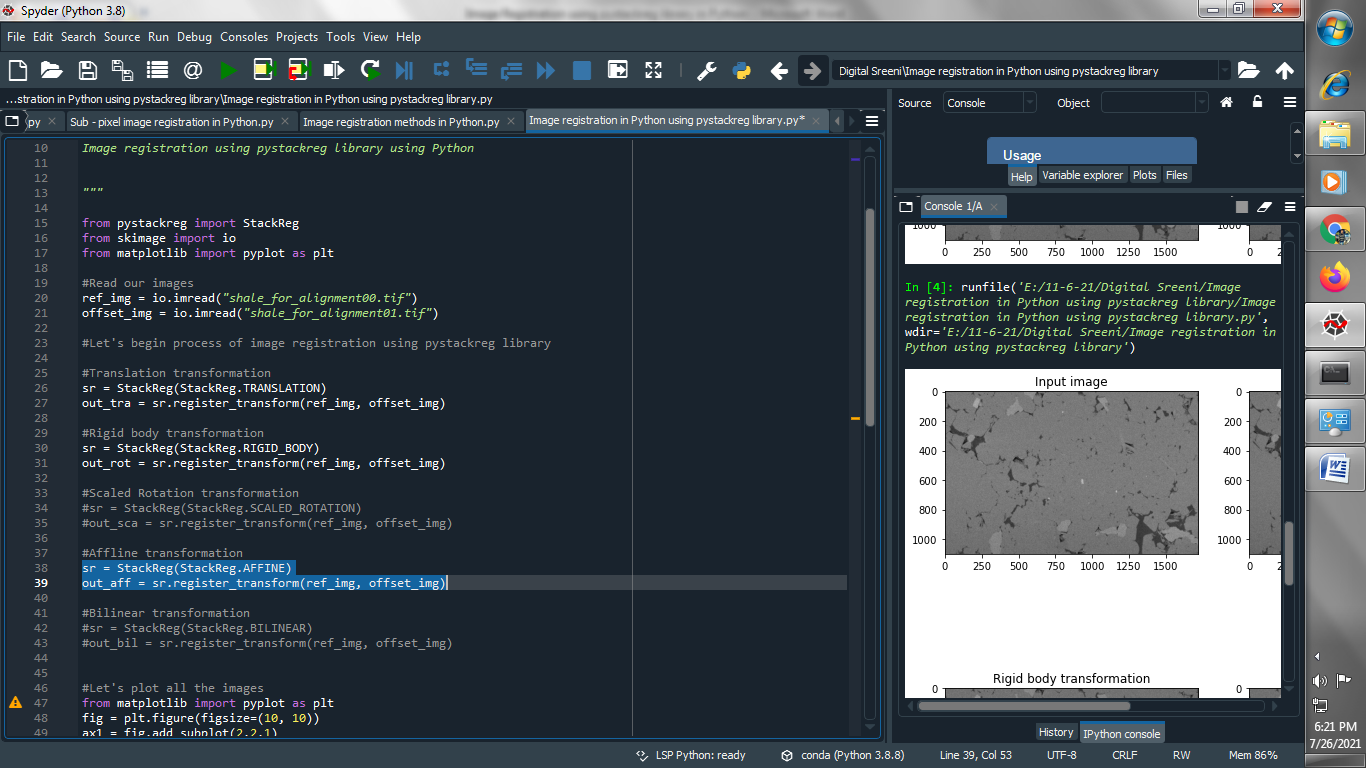
**(3) Apply Rigid body transformation :**

****

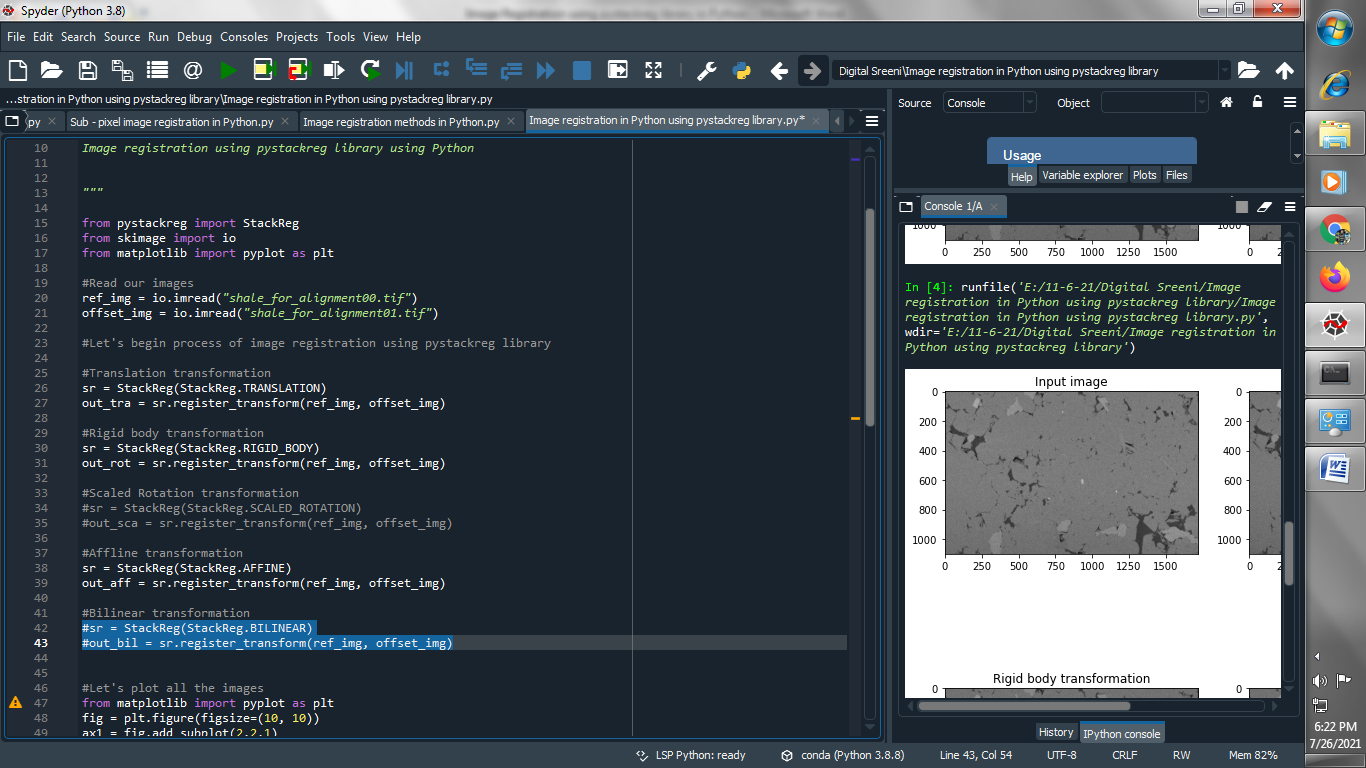
**(4) Apply Scaled Rotation transformation :**

****

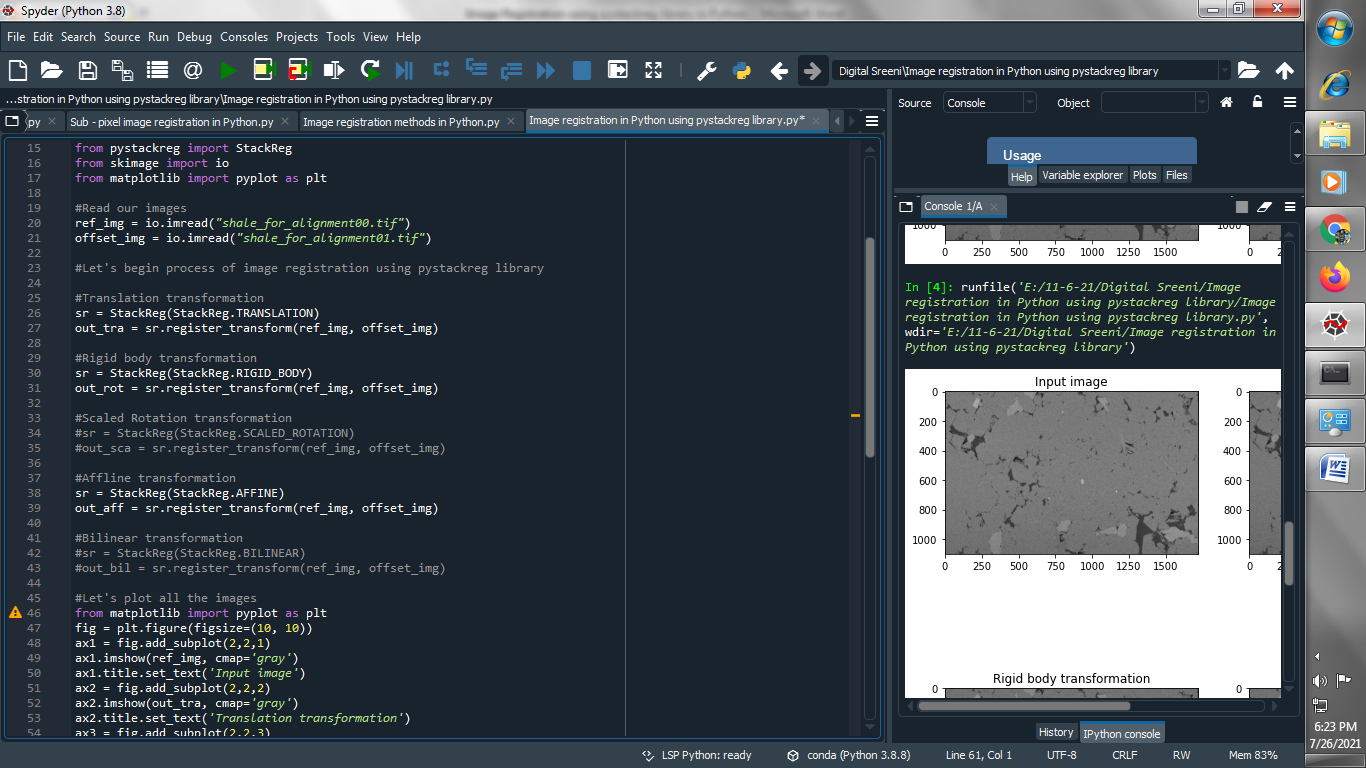
**(5) Apply Affline transformation :**

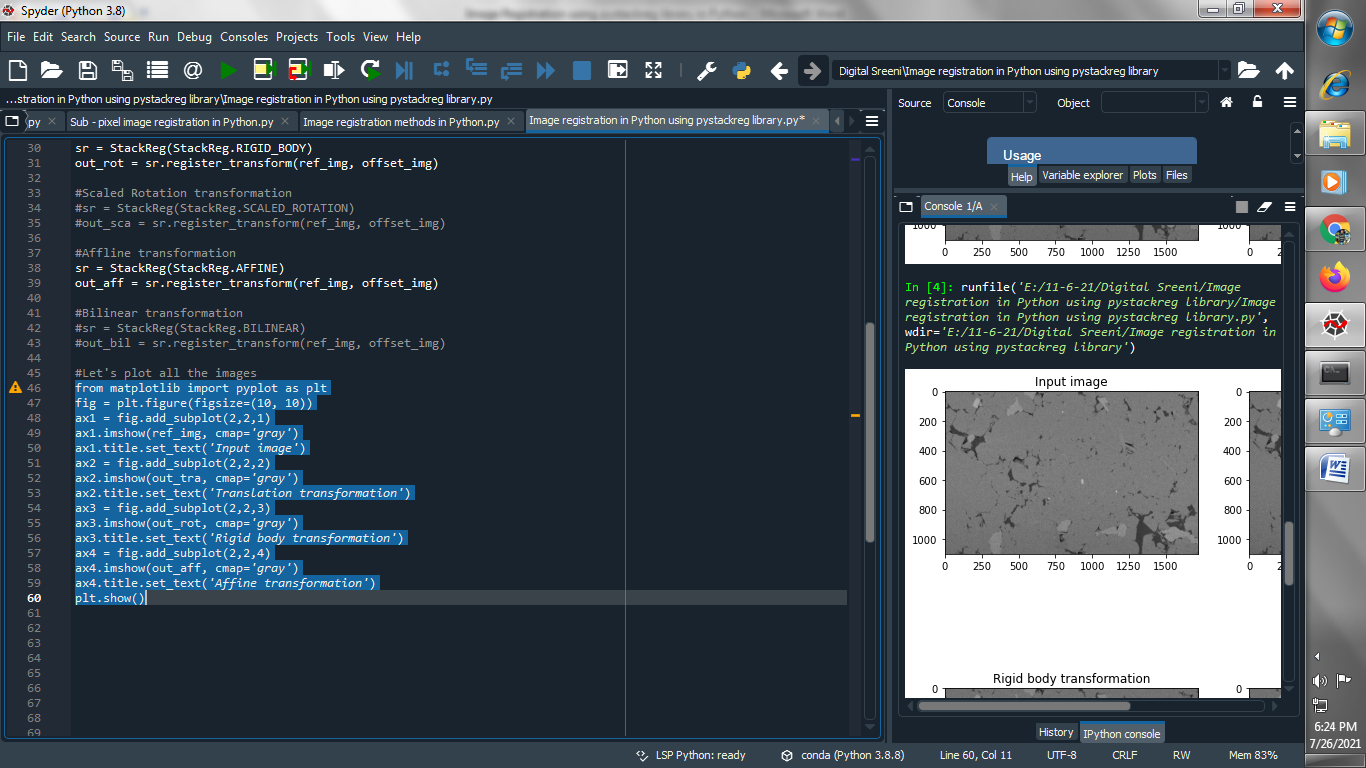
****

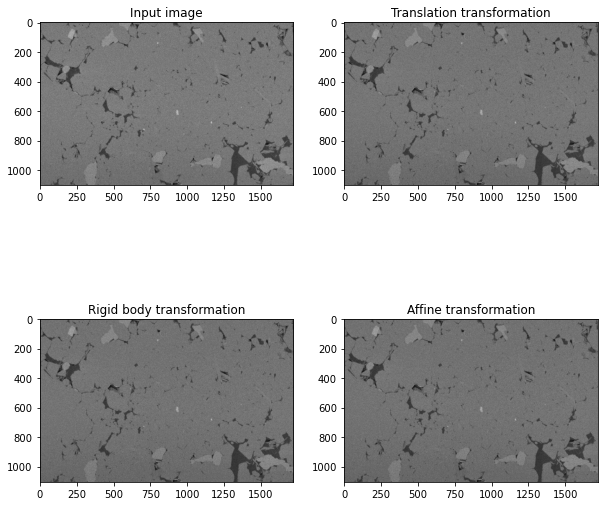
**(6) Apply Bilinear transformation :**

****

**(7) Let’s plot all the transformation of image registration :**

****

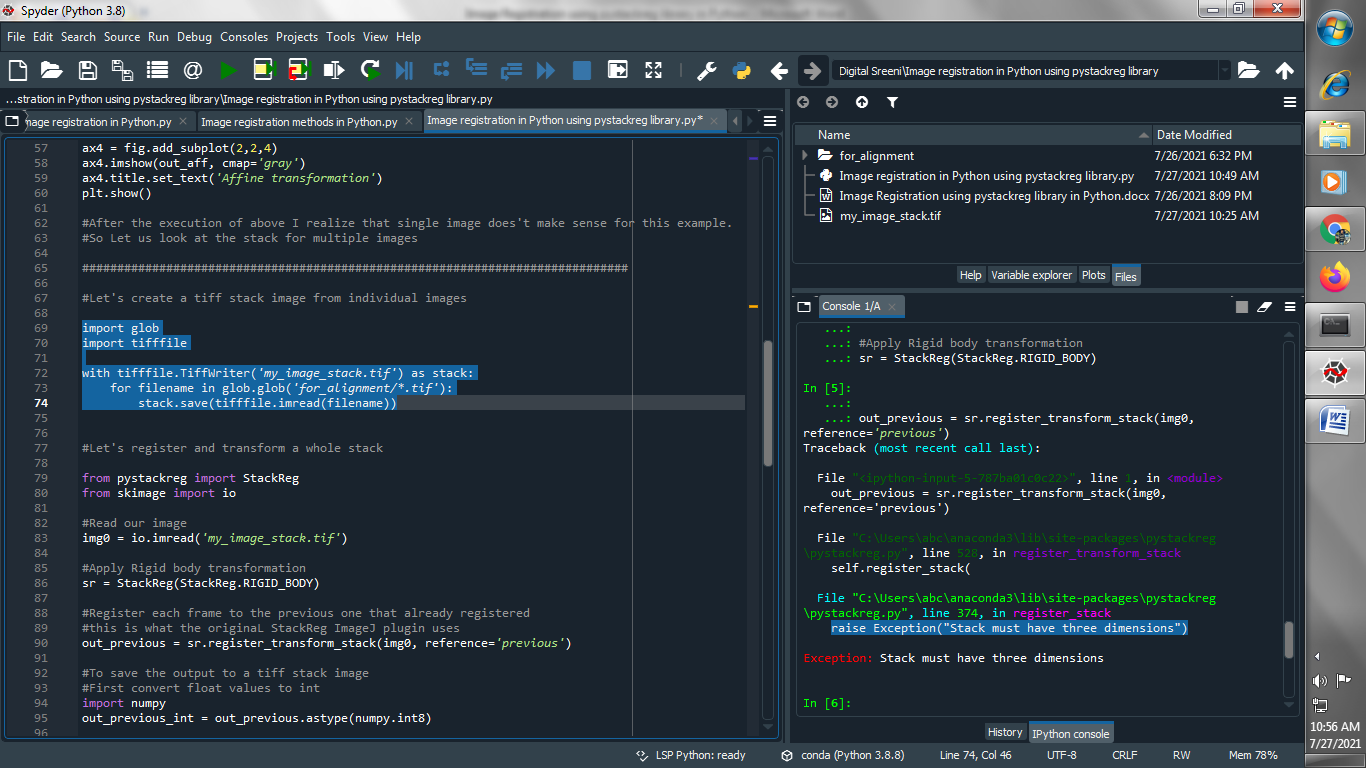
****

**Output :  
**

**################################################################################**

**→ After the execution of above I realize that single image doesn't make sense for this example. So Let us look at the stack for multiple images :**

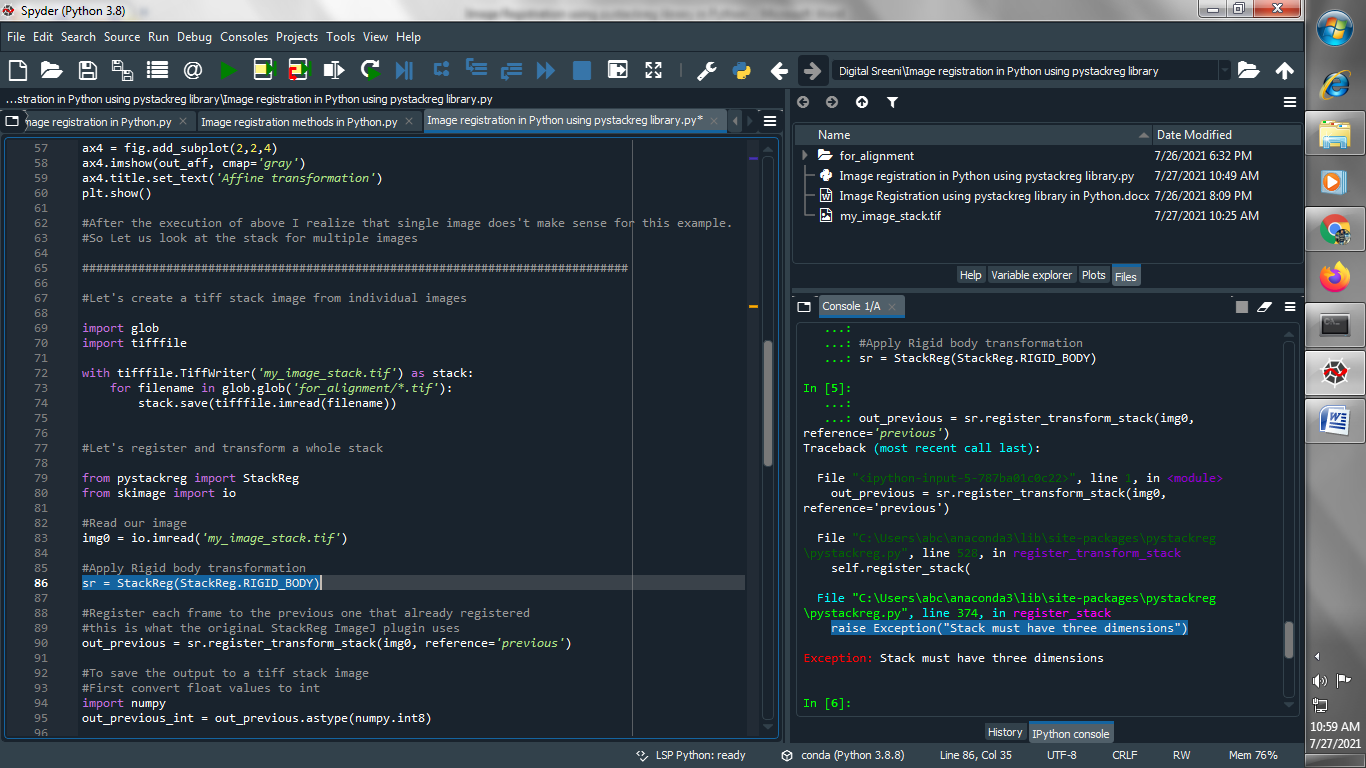
**(1) Let's create a tiff stack image from individual images :**

****

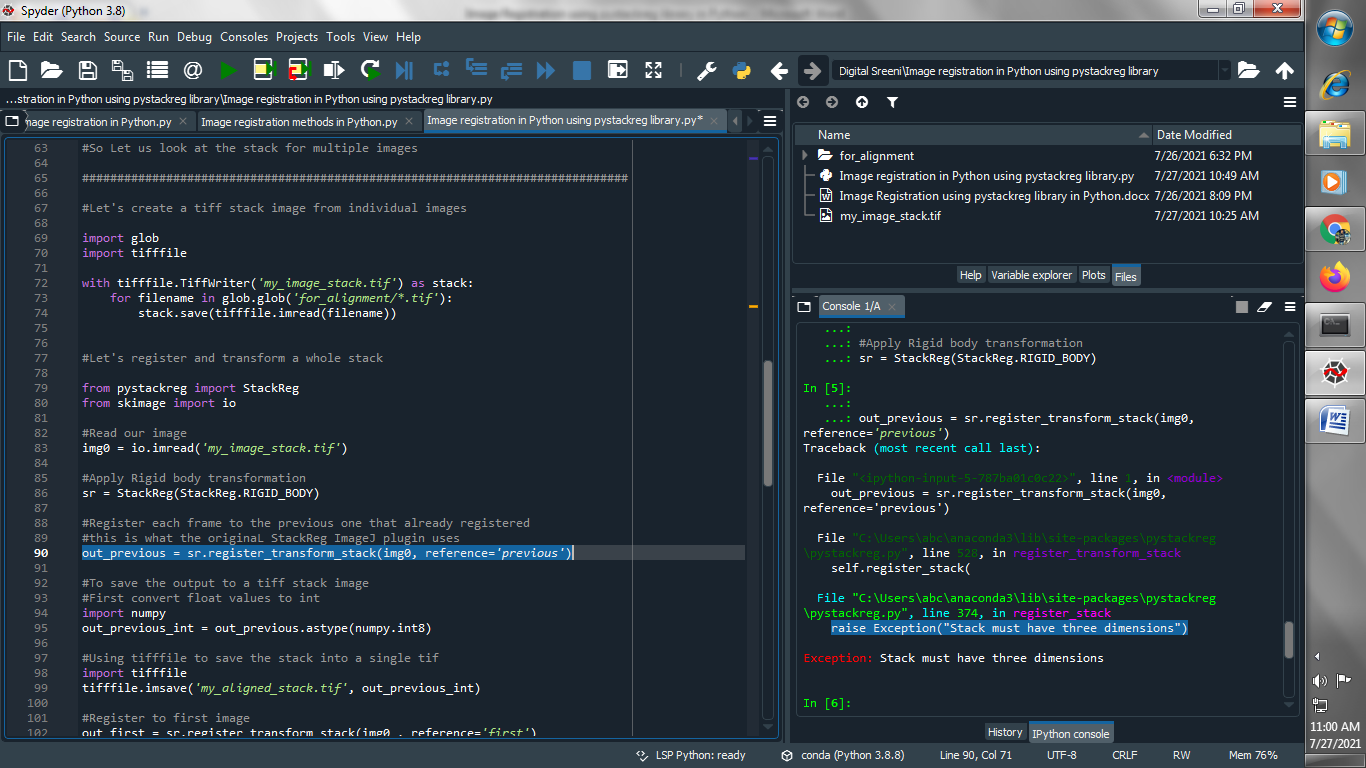
**(2) Let's register and transform a whole stack so first of all read our stack image :**

****

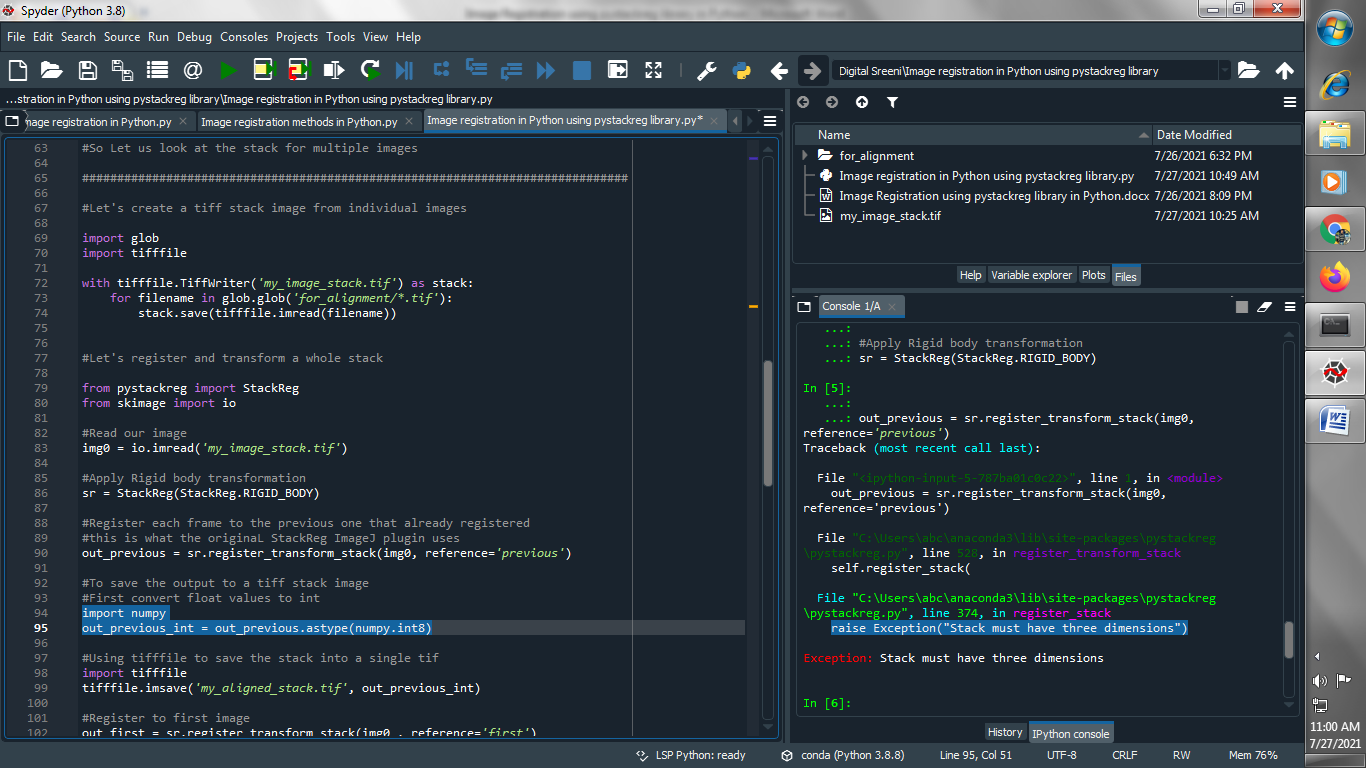
**(3) Apply Rigid body transformation :**

****

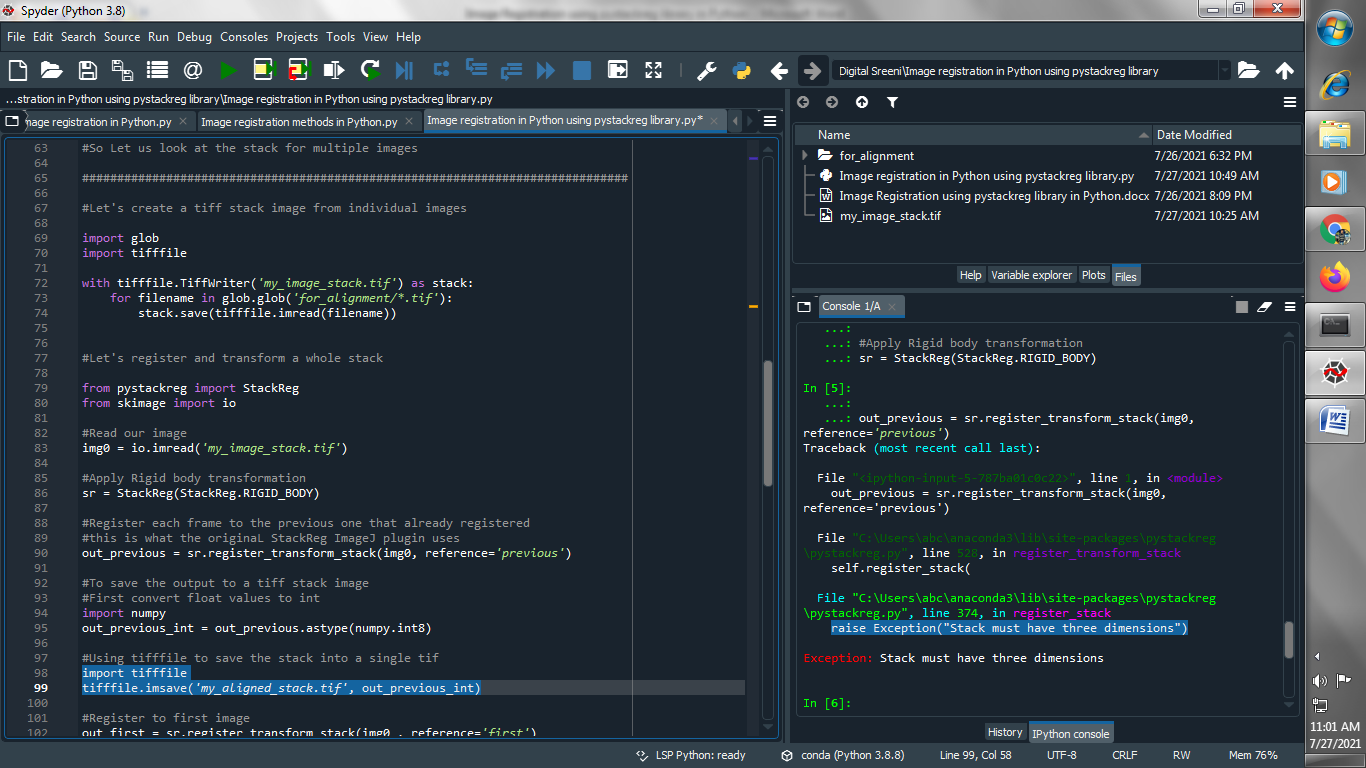
**(4) Register each frame to the previous one that already registered :**

****

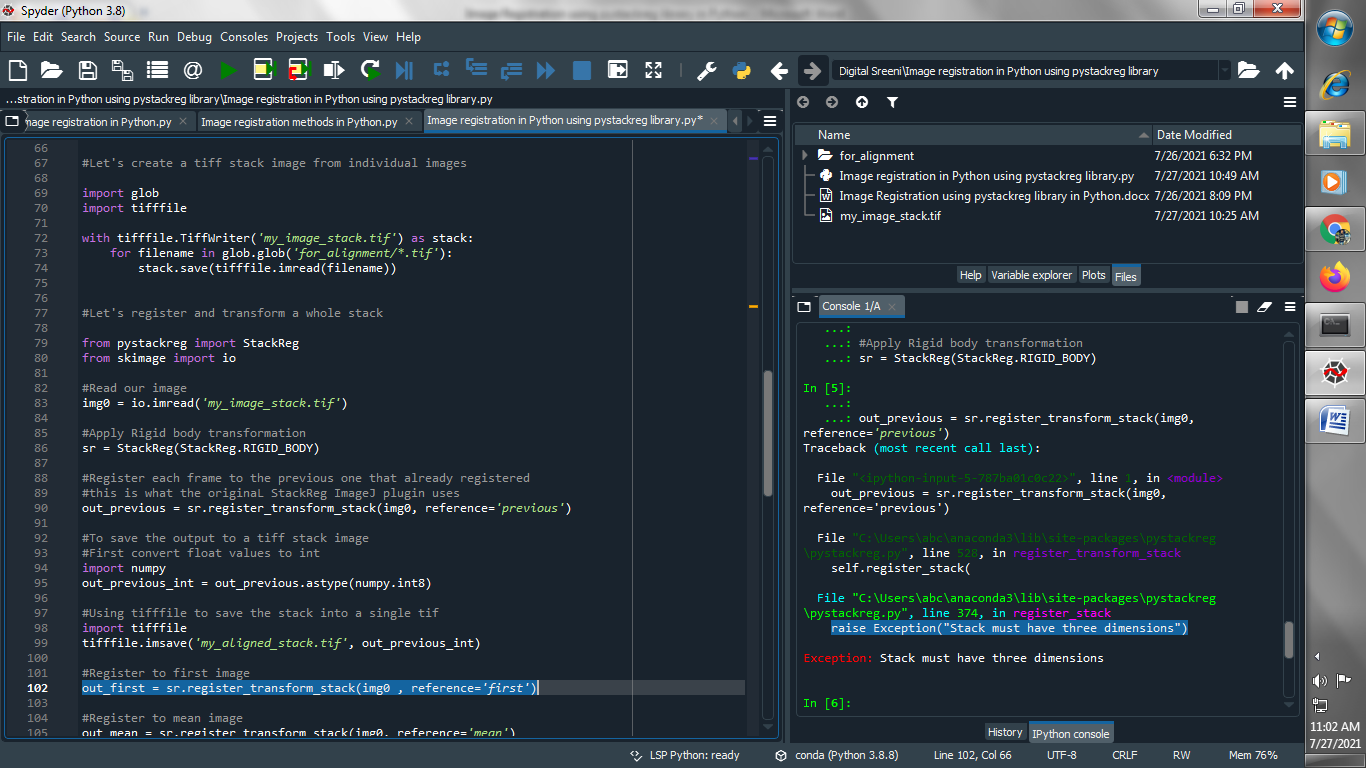
**(5) To save the output to a tiff stack image First convert float values to int :**

****

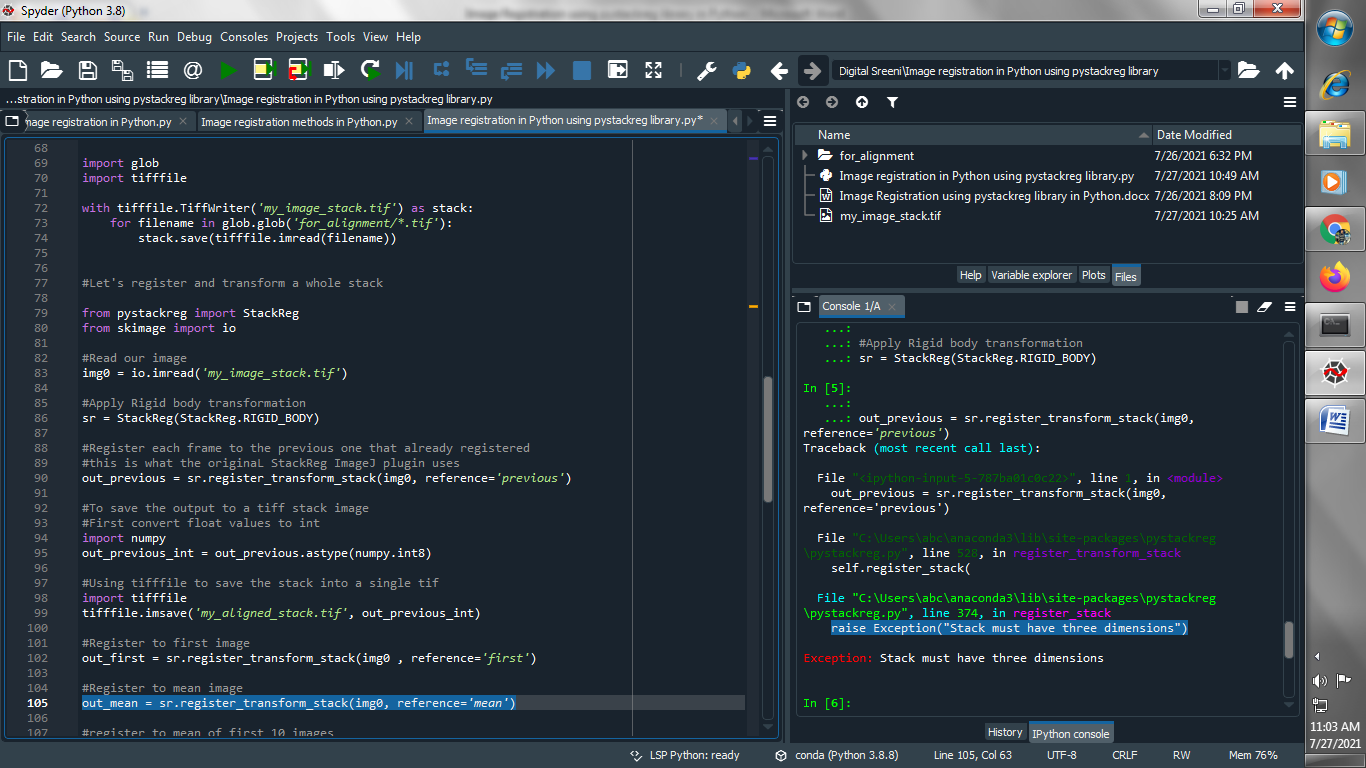
**(6) Using tifffile to save the stack into a single tif :**

****

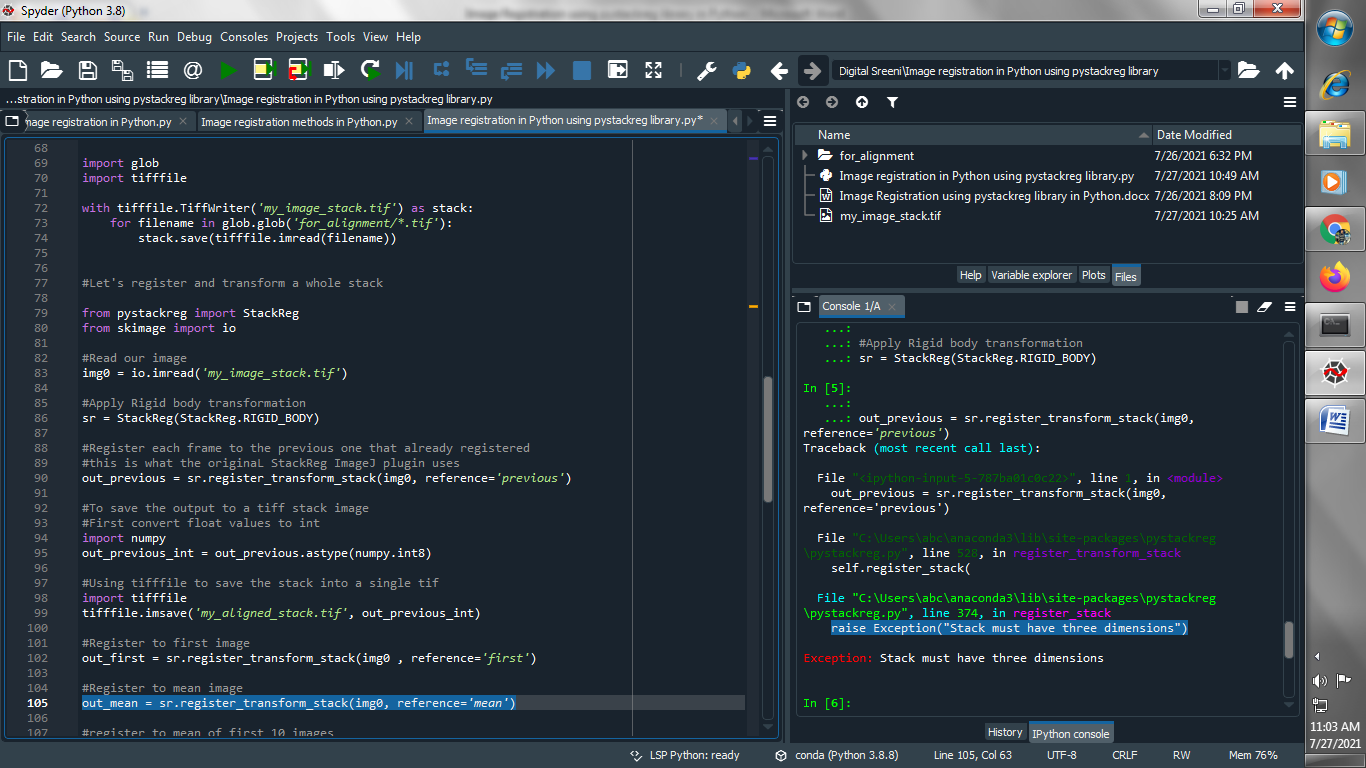
**(7) Register to first image :**

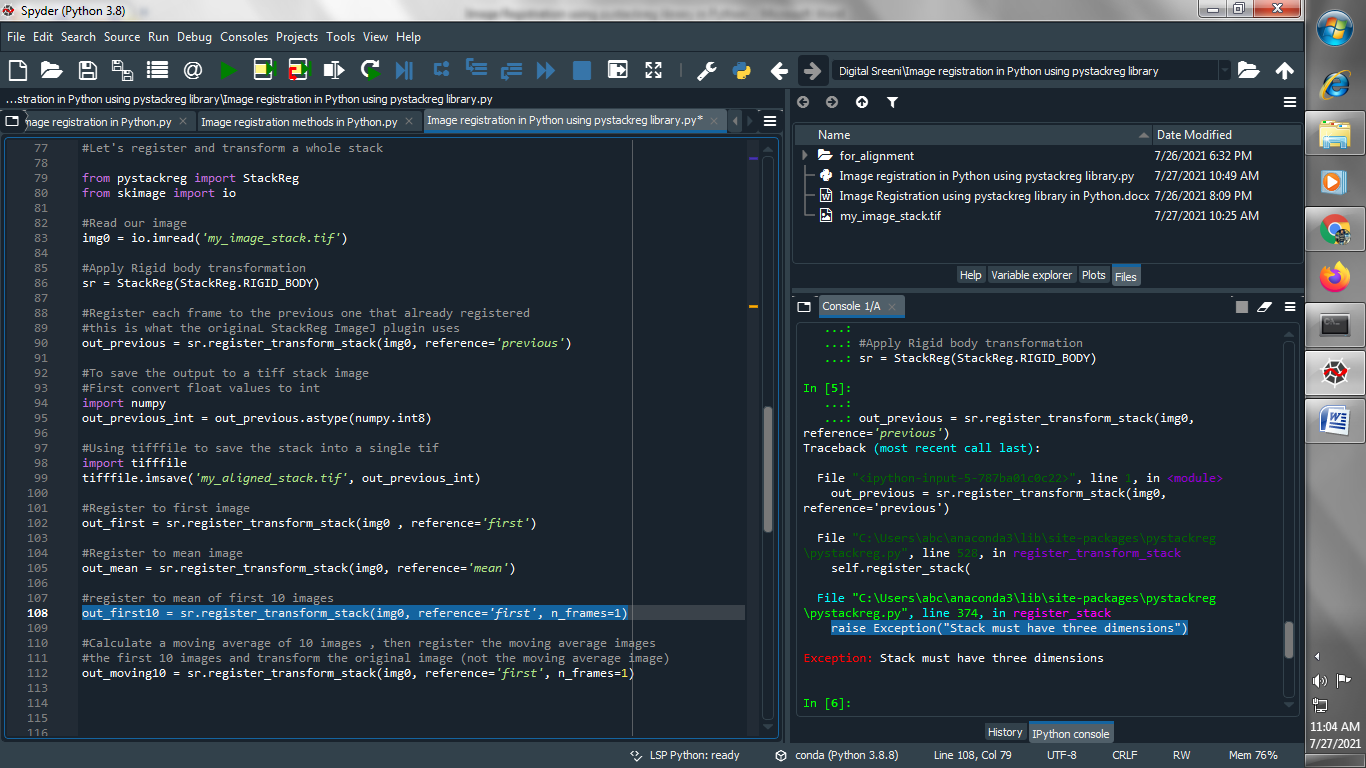
****

**(8) Register to mean image :**

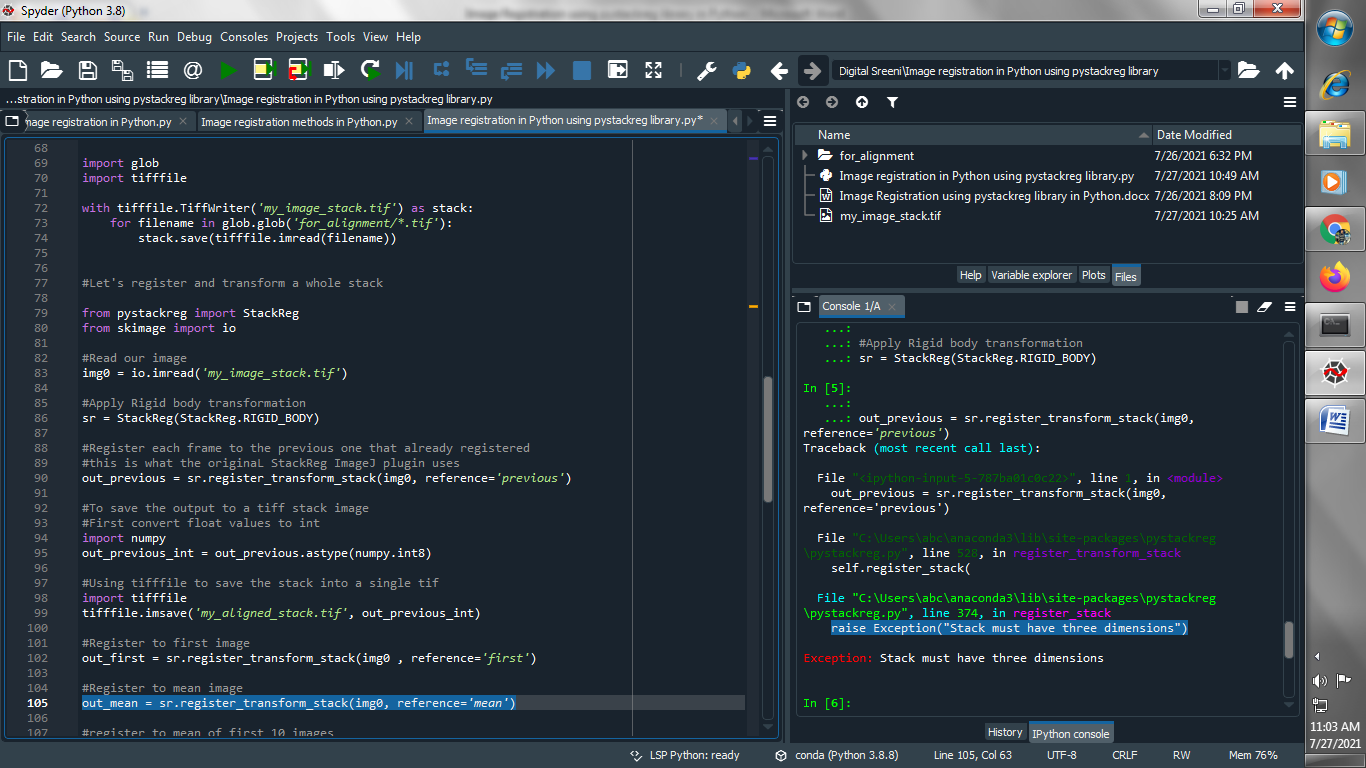
****

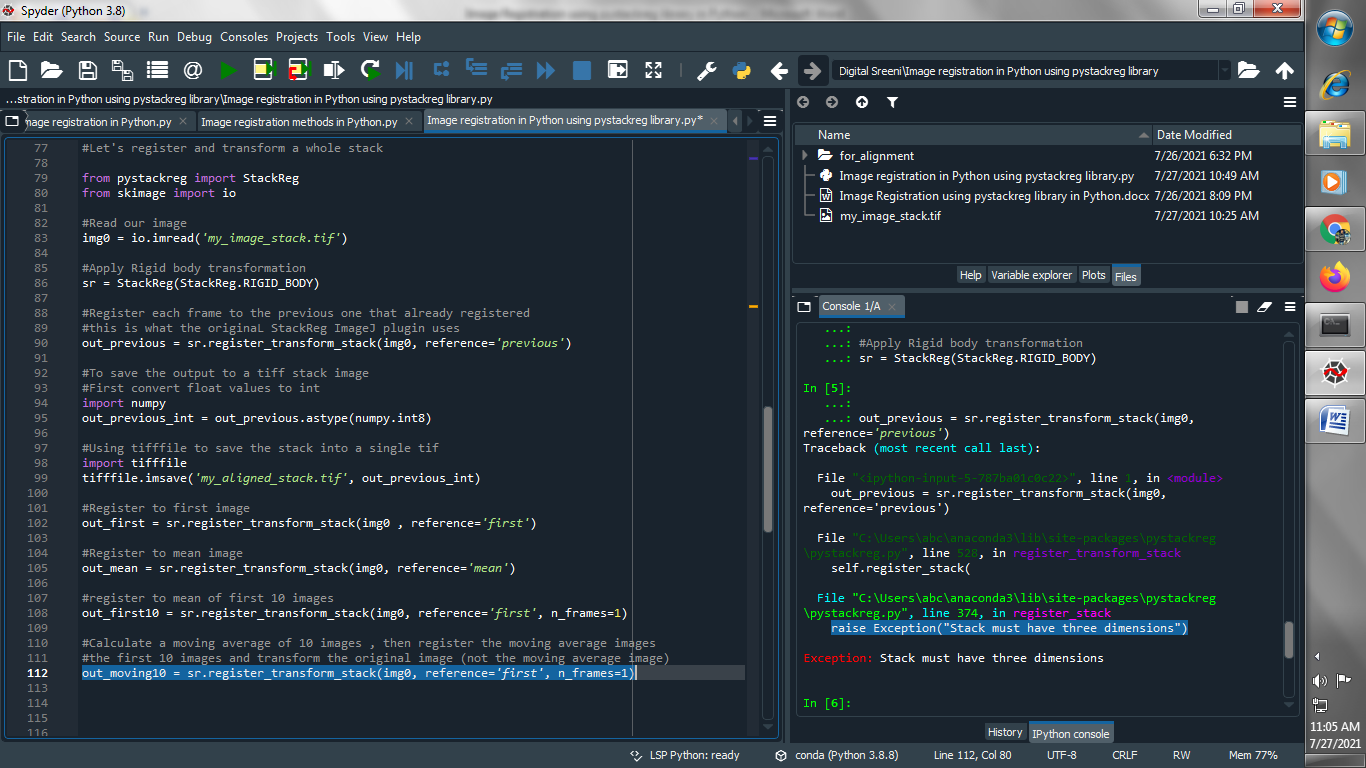
**(9) register to mean of first 10 images :**

****

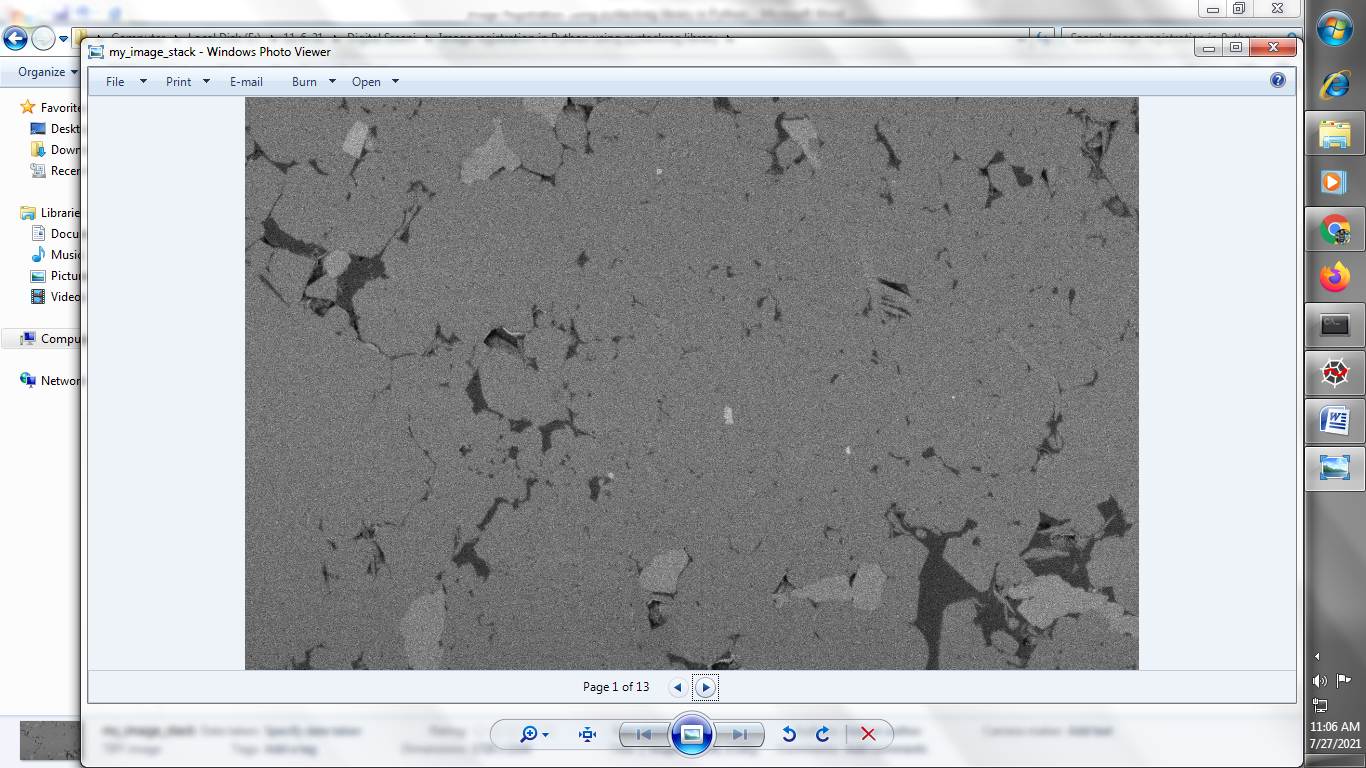
****

**(10) Calculate a moving average of 10 images , then register the moving average images the first 10 images and transform the original image (not the moving average image :**

****

****

**Output :**

****