

Scripting in Icy

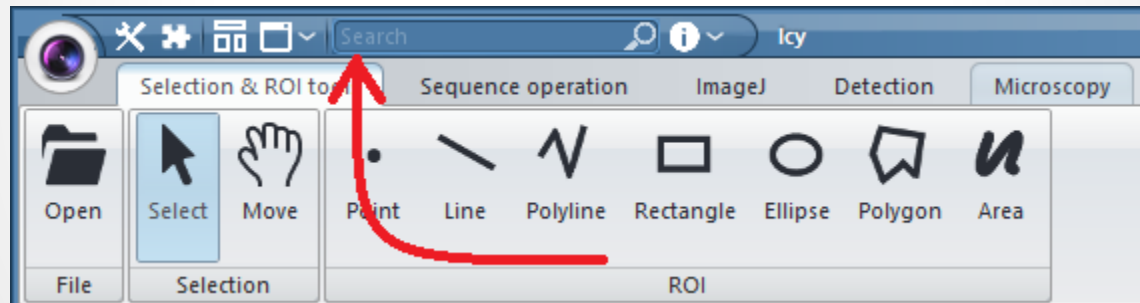
fast edition - creating batches

Scripts: the basics

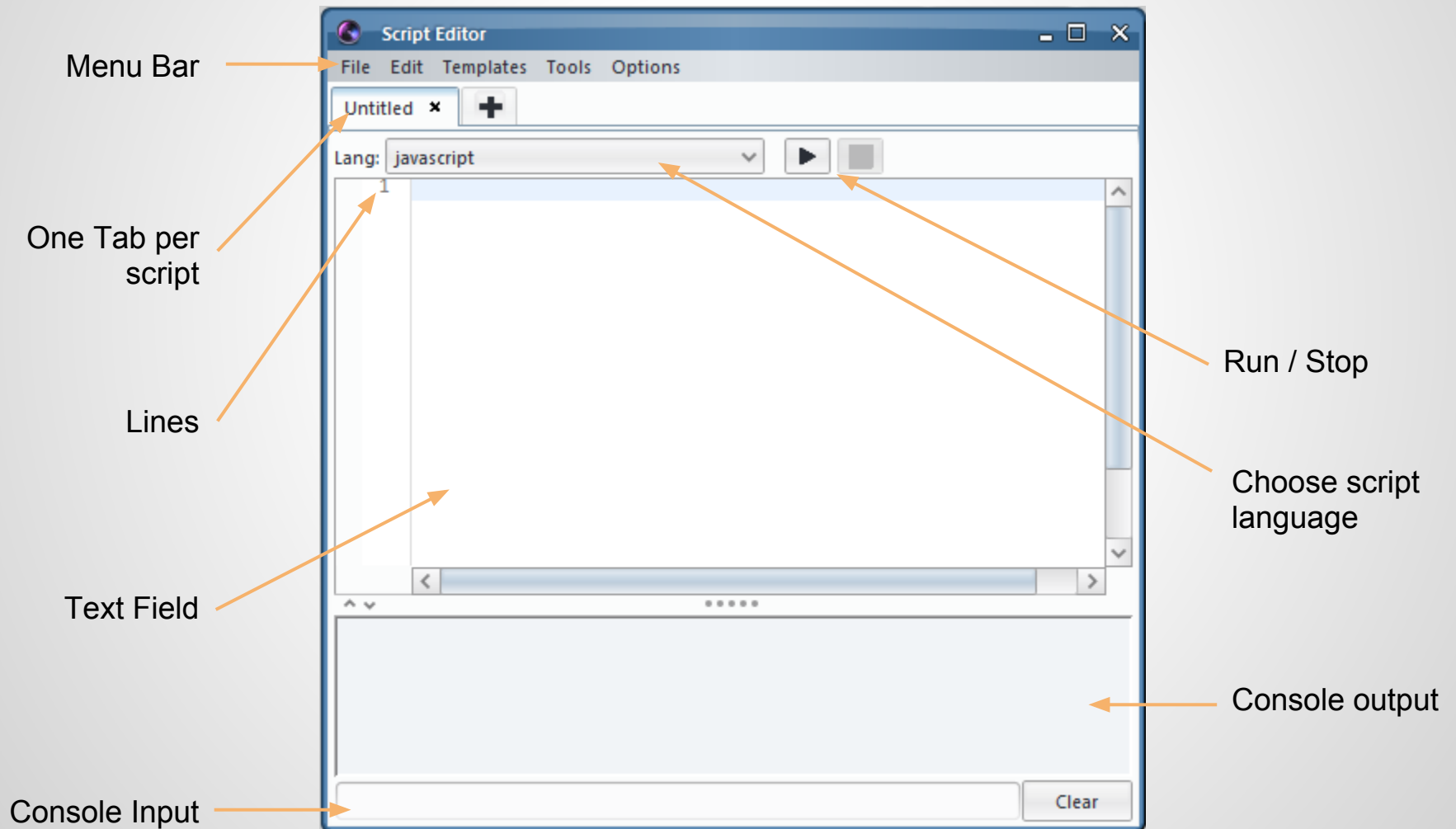
"A script is a program written for a software environment that automates the execution of tasks. The script benefits from a high level interface, much more accessible."

First Script

- Open the Script Editor plugin:
 - Search it with the SearchBar
 - (Install and) Run it by clicking on it

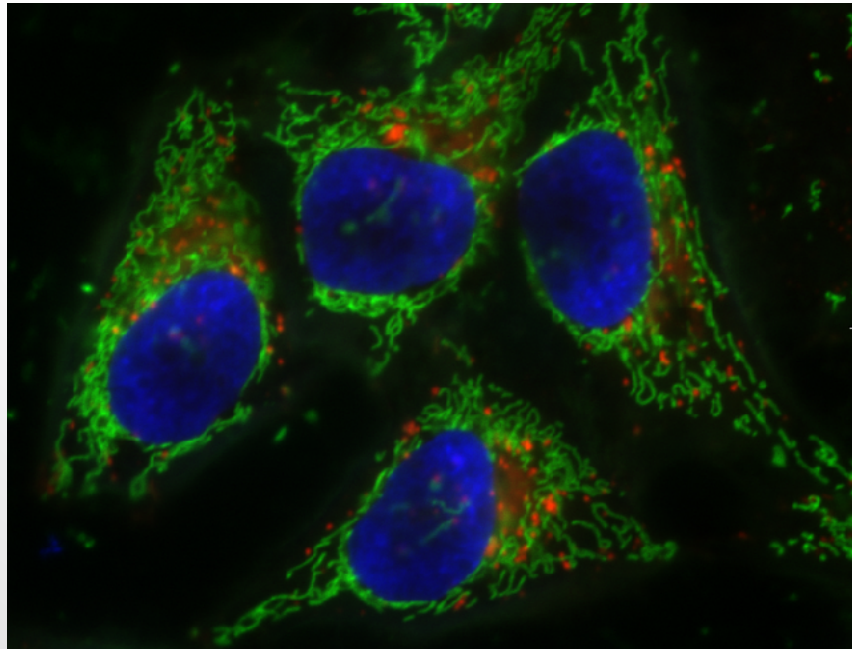


First Script



Predefined features

- Open the sequence: hela-cells.tif
- Get the current Sequence:
`seq = getSequence()`

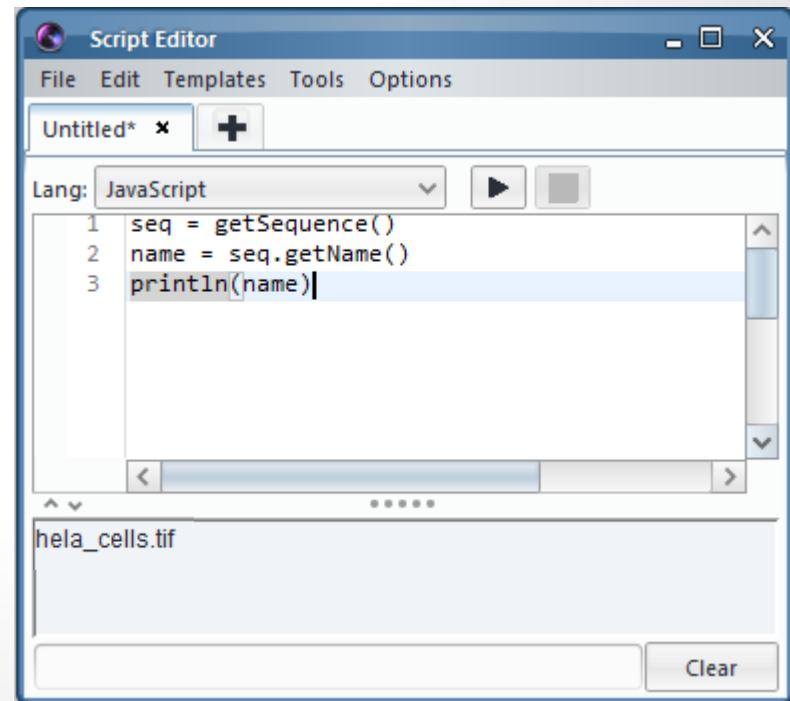


Display sequence information

- ```
seq = getSequence()
name = seq.getName()
println(name)
```
- Displays the name of the sequence.

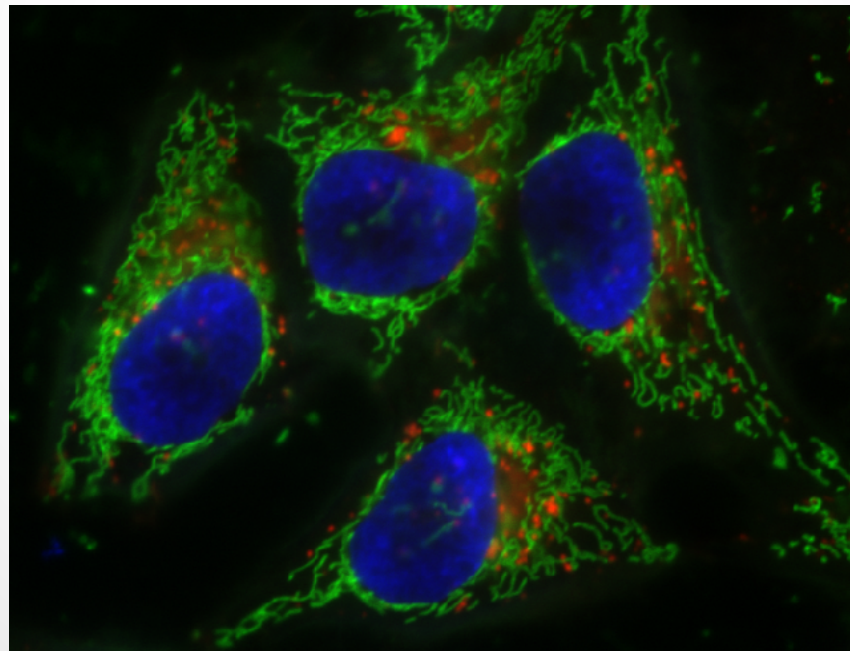
# Display sequence information

- ```
seq = getSequence()  
name = seq.getName()  
println(name)
```
- Displays the name of the sequence.



Using Icy tools (extractChannel.js)

- Extract the channel containing the spots (channel 0)



Using Icy tools (extractChannel.js)

- All interesting methods for Sequence Operations are stored in the SequenceUtil.

```
seq = getSequence()
```

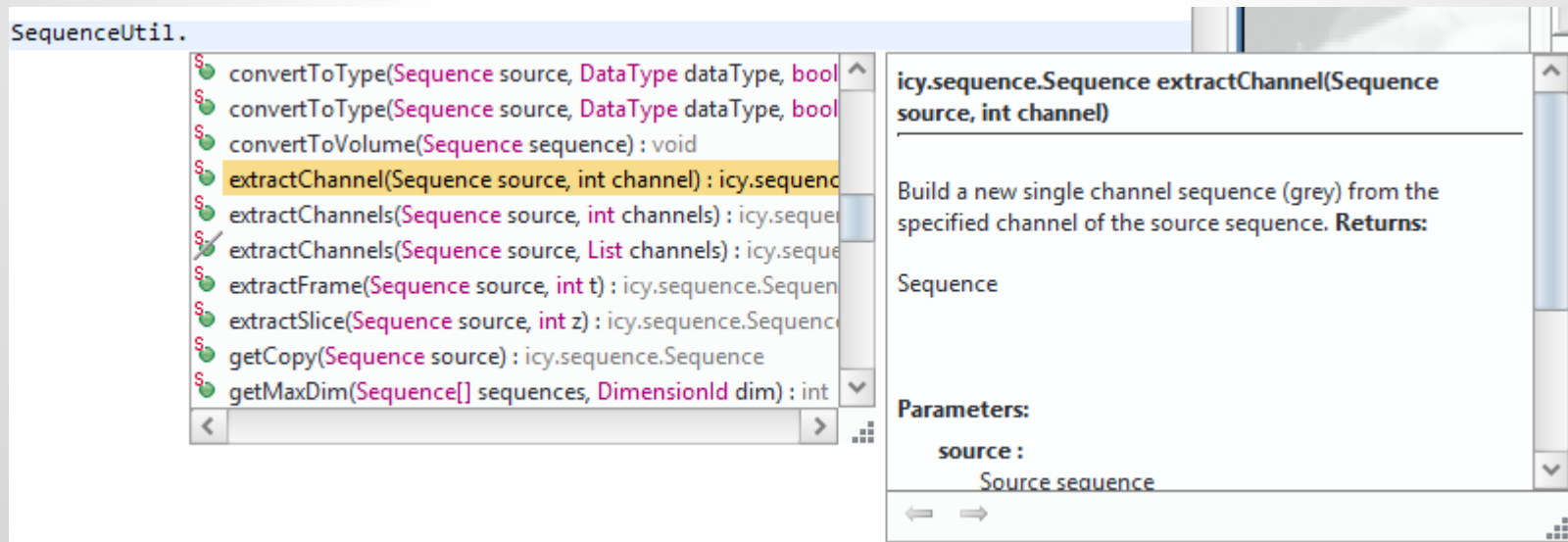
SequenceUtil.

Using Icy tools (extractChannel.js)

- All interesting methods for Sequence Operations are stored in the SequenceUtil.

```
seq = getSequence()
```

SequenceUtil.



Using Icy tools (extractChannel.js)

- All interesting methods for Sequence Operations are stored in the SequenceUtil.

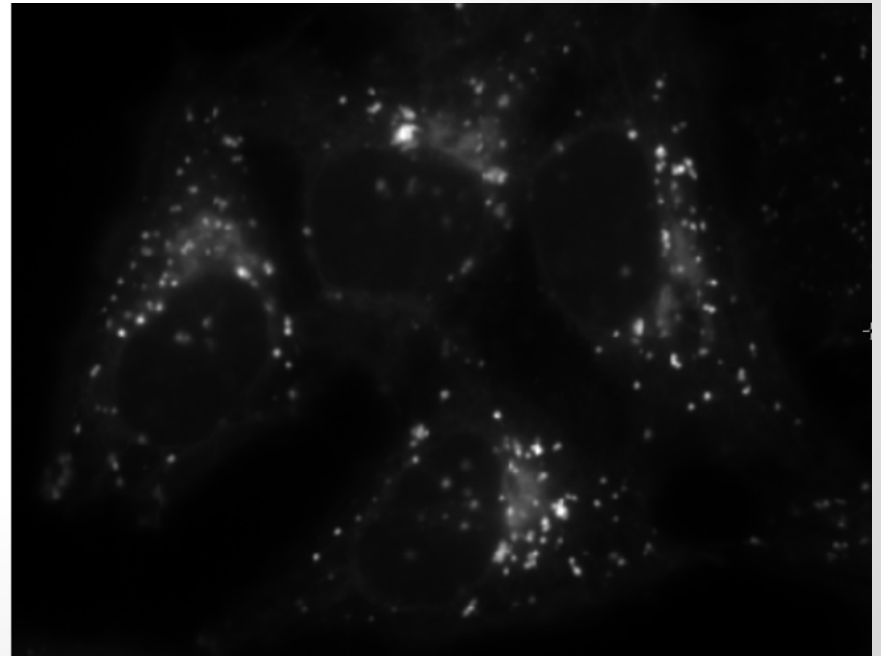
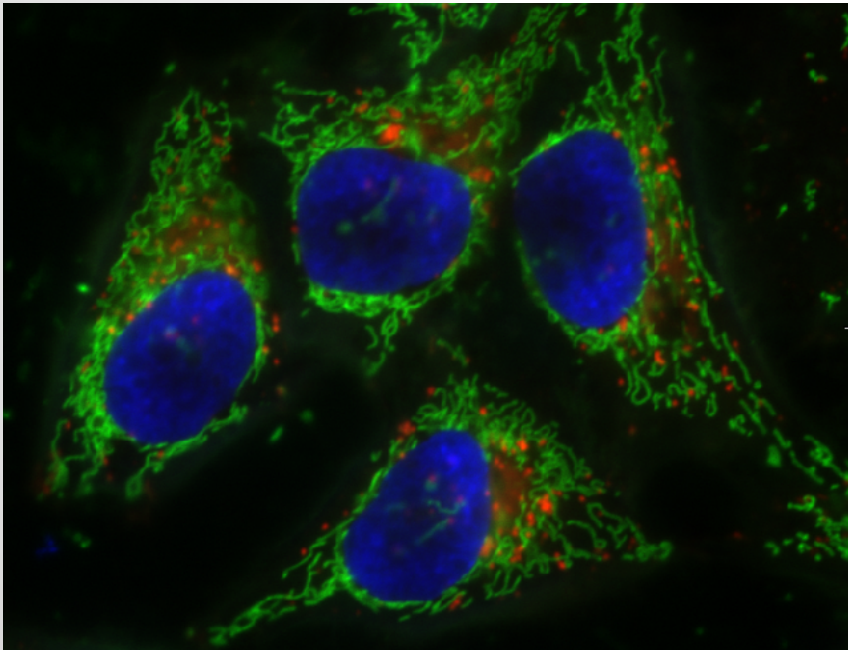
```
seq = getSequence()
```

```
SequenceUtil.extractChannel(seq, 0)
```

- The index always starts at "zero" and not "one".

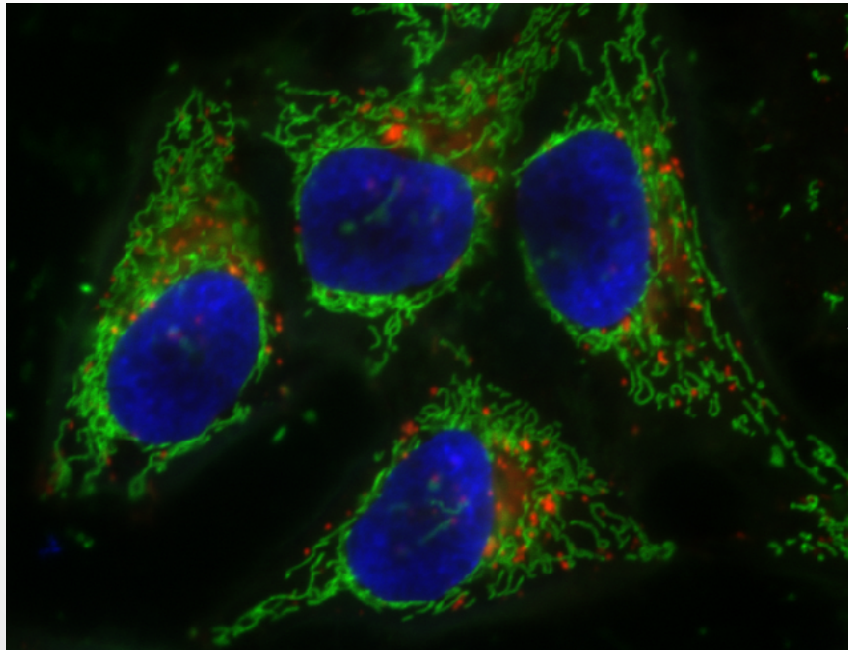
Using Icy tools (extractChannel.js)

- Extract the channel containing the spots



Using Plugins (Auto Threshold.js)

- Perform an Auto Threshold on the nuclei channel (channel 2)



Perform an auto threshold (autothreshold.js)

- KMeans is a probabilistic method that will compute thresholding values

```
kmeans = KMeans.computeKMeansThresholds(seq, 2, 2, 255)
```

Perform an auto threshold (autothreshold.js)

- KMeans is a probabilistic method that will compute thresholding values

```
kmeans = KMeans.computeKMeansThresholds(seq, 2, 2, 255)
```

- Why two?

Perform an auto threshold (autothreshold.js)

- KMeans is a probabilistic method that will compute thresholding values

```
kmeans = KMeans.computeKMeansThresholds(seq, 2, 2, 255)
```

- We want to separate the histogram in two parts.

Perform an auto threshold (autothreshold.js)

- KMeans is a probabilistic method that will compute thresholding values

```
kmeans = KMeans.computeKMeansThresholds(seq, 2, 2, 255)
```

```
result = Thresholder.threshold(seq, 2, kmeans, false)
```

```
result.setName(seq.getName() + " - thresholded")
```

Perform an auto threshold (autothreshold.js)

- KMeans is a probabilistic method that will compute thresholding values

```
kmeans = KMeans.computeKMeansThresholds(seq, 2, 2, 255)
```

```
result = Thresholder.threshold(seq, 2, kmeans, false)
```

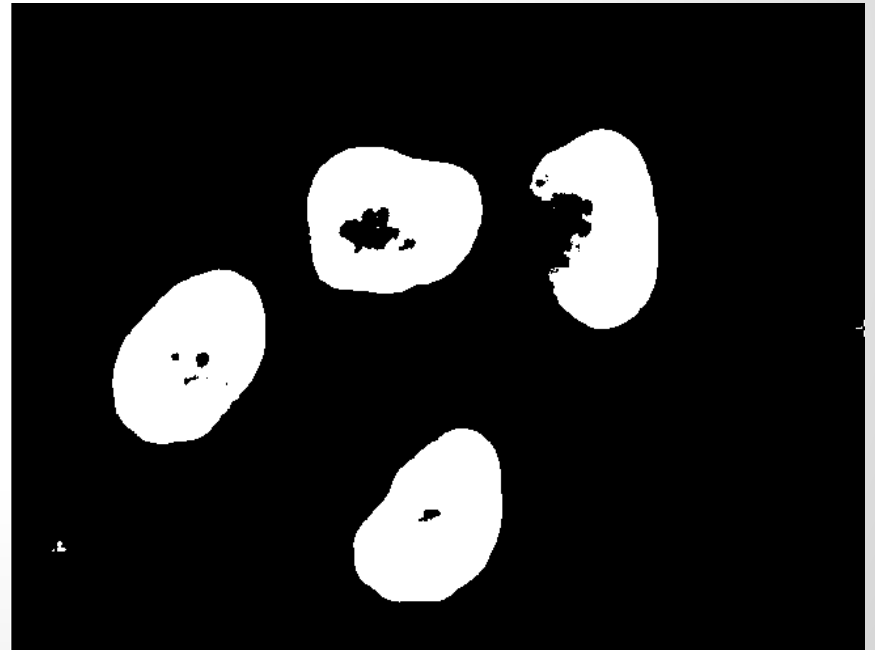
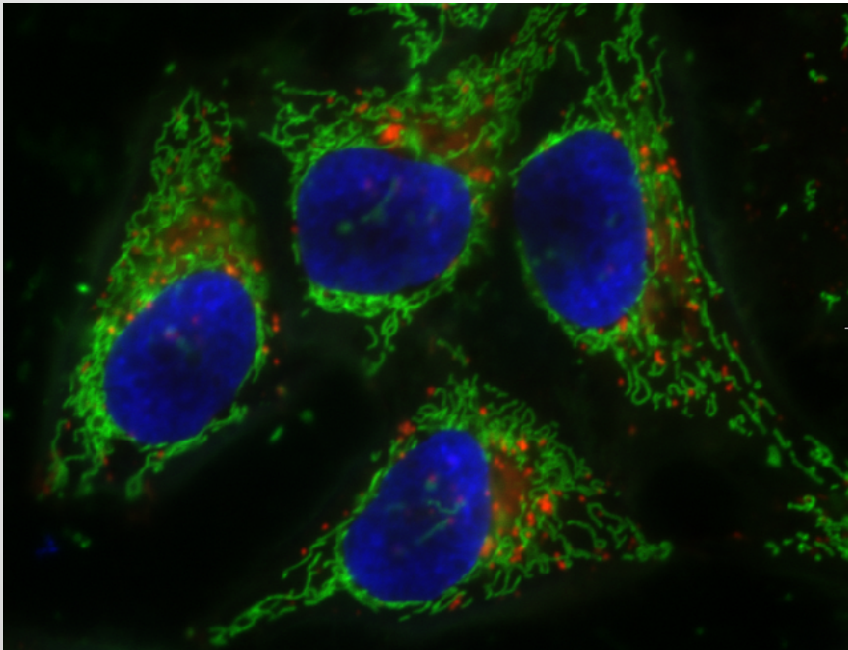
```
result.setName(seq.getName() + " - thresholded")
```

- the result is a sequence, that we can add into Icy with the “gui” variable:

```
gui.addSequence(result)
```

Using Plugins (Auto Threshold.js)

- Perform an Auto Threshold on the nuclei channel



Input / Output (openFile.js)

- Load a sequence

```
f = FileDialog.open()
```

```
if (f == null) throw "User cancelled!"
```

```
loadedSequence = Loader.loadSequence(f)
```

Input / Output (saveFile.js)

- Save a sequence

```
seq = getSequence()
```

```
f = FileDialog.save()
```

```
Saver.save(seq, f, true)
```

Auto threshold on a file

- Load a file, threshold it, save the result

```
f = FileDialog.open()  
seq = Loader.loadSequence(f)
```

```
kmeans = KMeans.computeKMeansThresholds(seq, 0, 2, 255)  
result = Thresholder.threshold(seq, 0, kmeans, false)  
result.setName(seq.getName() + " - thresholded")
```

```
f = FileDialog.save()  
Saver.save(result, f, true)
```

Batch auto threshold (batchAutoThreshold.js)

- Loop on a folder

```
folderInput = FileDialog.openFolder()
```

```
files = folderInput.listFiles()
```

```
for (i = 0; i < files.length; ++i) {  
    Loader.loadSequence(files[i])  
}
```

Batch auto threshold (batchAutoThreshold.js)

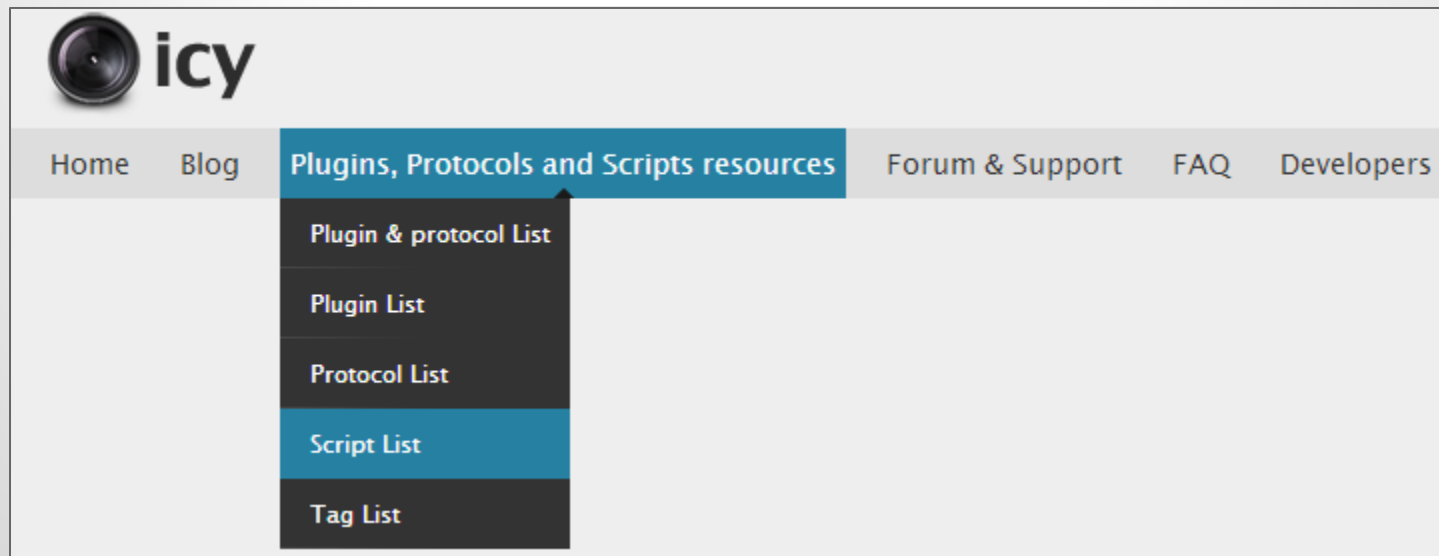
```
7
8 sep = File.separator // will be either \ or / depending on your OS
9
10 // Folder containing the input to batch on
11 folderInput = FileDialog.openFolder()
12
13 if (folderInput == null)
14     throw "User Cancelled."
15
16 // All the files contained inside the inputFolder
17 files = folderInput.listFiles()
18
19 // folder containing the result
20 folderOutput = FileDialog.openFolder()
21
22 if (folderOutput == null)
23     throw "User Cancelled."
24
25 // fileName (a string) of the output, as a base for the files
26 fileName = folderOutput.getPath() + sep
27
28 // Loop on every file.
29 for (i = 0; i < files.length; i++) {
30     seq = Loader.loadSequence(files[i])
31
32     // computes the kmeans thresholds for automatic thresholding
33     kmeans = KMeans.computeKMeansThresholds(seq, 0, 2, 255)
34
35     // performs the threshold and put the result (a sequence) inside a variable
36     result = Thresholder.threshold(seq, 0, kmeans, false)
37
38     // create a file based on the original sequence name, to which
39     // is added _kmeans.
40     outputFile = new File(fileName + seq.getName() + "_kmeans.tif")
41
42     // Save the resulting sequence to the file
43     Saver.save(result, outputFile)
44 }
45
```


Find scripts

- Scripts are available on the website:

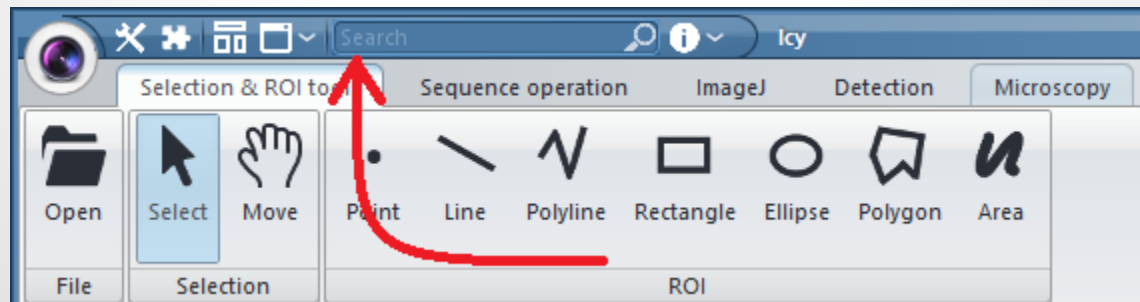
<http://icy.bioimageanalysis.org/script/list>

- You can download them:
 - Directly from the website



Find scripts

- Scripts are available on the website:
<http://icy.bioimageanalysis.org/script/list>
- You can download them:
 - Directly from the website
 - By using the Search Bar



Find scripts

- Scripts are available on the website:
<http://icy.bioimageanalysis.org/script/list>
- You can download them:
 - Directly from the website
 - By using the Search Bar
- Add your future scripts on the website, and share them!