1. **Automate the process:**
   i) Plugins->Macro->Record
   ii) Execute all the steps
   iii) Recorder->Create (Saves .ijm file of macro)
2. **To run on a new image:**
   iv) Drag and drop the image and macro into fiji and click run in the macro window
3. **Debugging:**
   i) If you made a mistake during the process, revisit the process and try to ID the step where you messed up
   ii) Change the parameters, instead of redoing the whole pipeline
4. **Window Name Errors:**
   i) When you create the macro using "FileA.jpg", it will give errors if you use it on another image
   ii) It can't generalise
   iii) Solution: Instead of string for filename, use a variable like
   Channel1 = File.name (it autocompletes)
   And e.g. if it creates another image titled c1-…..jpg, then use
   Channel2 = "C1-" + File.name
   (ImageJ Macro language is quite annoying like this)
   iv) Solution: You can choose Language->Python (but it's very limited, so you can use Java instead)
   v) *We can have for-loops, if statements etc.*
5. **To process a folder of images:**
   i) Templates-> ImageJ 1.x -> Process Folder (IJ1 macro)
   ii) This opens a template, where you can insert your own code in the processFile() function
   iii) The other function, processFolder() is a recursive function that ends with processFile() as the base case
   iv) Make sure Input folder has all ".tif" images and Output folder is empty
   v) Be sure to include, this in processFile()

```
open(input + File.separator + file);

//Actual steps

save("Tiff", output + File.separator + "FrangiOUTPUT_" + file);
run("Close All");
```

6. **Plugins:**
   i) Most of the advanced plugins are macro-recordable (Power of Fiji)
   ii) E.g. Machine learning based pixel segmentation
   iii) DO the whole process, with recording on
   iv) Now save this as a macro, so you can use it later

Resources:

- ImageJ website: Built-in functions
- Forum.image.sc (community)
- Neubias academy (talks on macro language, machine learning with fiji etc.)