

## Learning to Program in MATLAB

### Part I (Resources for self-learning)

At the first meeting of MATLAB Club, I explained that one of the reasons MATLAB was my chosen language for forming a club was the ready availability of tutorials on their knowledge base.

Specifically:

<https://www.mathworks.com/products/matlab/webinars.html>

<https://www.mathworks.com/videos/search.html>

<https://matlabacademy.mathworks.com/>

[http://www.mathworks.com/matlabcentral/fileexchange/?s\\_tid=gn\\_mlc\\_fx](http://www.mathworks.com/matlabcentral/fileexchange/?s_tid=gn_mlc_fx)

Now, if you have not already completed MATLAB Academy: I would suggest doing so, however this video(<https://www.mathworks.com/videos/getting-started-with-matlab-68985.html>) is a good primer regardless.

### Part II (Basic Programming Concepts)

Although Mathworks Academy covers topics surrounding Matrix/Array/Vector manipulation in MATLAB pretty well, I think there is some level of assumed familiarity with "if" "for" "while" control logic so I think that will be our largest uphill battle to fight.

[http://www.mathworks.com/help/matlab/learn\\_matlab/flow-control.html](http://www.mathworks.com/help/matlab/learn_matlab/flow-control.html)

There is no substitute for actually using these tools in order to learn how to use them; however, I think C, C++, and Java are probably the languages with the largest variety of introductory tutorials and exercises in order to offer problems to solve with them:

<https://www.arduino.cc/en/Reference/If>

<https://www.arduino.cc/en/Reference/While>

<https://www.arduino.cc/en/Reference/For>

[www.learncpp.com/cpp-tutorial/51-control-flow-introduction/](http://www.learncpp.com/cpp-tutorial/51-control-flow-introduction/)

<http://www.learntosolveit.com/cprogramming/#chapter-3-control-flow>

Hopefully once the ENGE116 curriculum becomes available: we will have a little more straight-forward approach to practicing these ideas.

Unfortunately, I'm spread pretty thin so my ability to provide substantial support on a one-on-one basis is limited; however, if we divide and conquer the problem of drafting some tutorials and exercises for MATLAB Club, I definitely think it's ideal if everyone can spearhead small pieces of the puzzle and then we combine those pieces into a "Club Resources for Beginners" collection.

### Part III (Final Remarks)

One of my mentors frequently talked about the mantra  
"1) Watch One 2) Do One 3) Teach One"

I'm not formally trained as a tutor, but this workflow is one that I've come to believe in. I'm skeptical how much benefit you can get from watching lectures without a "Workshop" format which reinforces the skills, or even-better: self-study on your own time.

IMHO:

The most important skill that can be developed in a classroom environment like this is an ability to "self-rescue" from compile errors, or when encountering code that you've never seen before and don't recognize. For that reason, I've created a "self-rescue" document with some basic "FAQs" about coding, and generic solutions to those FAQs.

That resource can be found at the following location:

<https://github.com/matlabclub/Knowledge-Base/blob/master/SelfRescueFAQs.pdf>