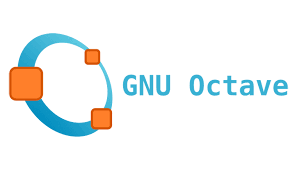
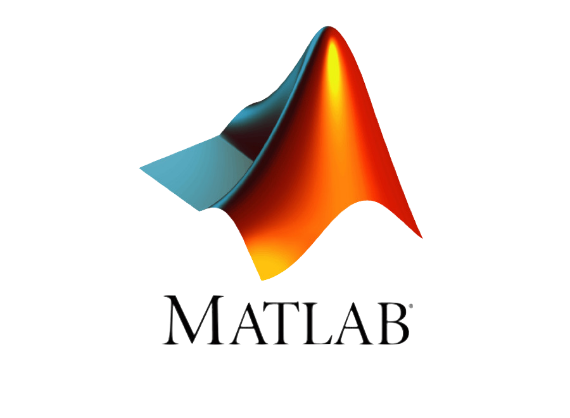
MATLAB Tutorial





This tutorial maintains a purpose to allow a user to understand how to use, understand and navigate MATLAB commands.

The tutorial that was used to acquire this information was:

[Tutorial](https://www.tutorialspoint.com/matlab/index.htm)

Table of Contents

[Downloading the Software 3](#_Toc184217147)

[Navigating the software 4](#_Toc184217148)

[Computing operations 5](#_Toc184217149)

[Using the semicolon 6](#_Toc184217150)

[Adding Comments 6](#_Toc184217151)

[Saving, Clearing & Loading 6](#_Toc184217152)

[Declaring Variables 7](#_Toc184217153)

[Formatting decimals 8](#_Toc184217154)

[Creating vectors and matrices 9](#_Toc184217155)

[Helpful Commands 10](#_Toc184217156)

[Data Types 10](#_Toc184217157)

[Operation symbols 11](#_Toc184217158)

[If-then Statements 12](#_Toc184217159)

[For-loops and While Loops 13](#_Toc184217160)

[Arrays 14](#_Toc184217161)

[Functions 14](#_Toc184217162)

[Plotting 15](#_Toc184217163)

# Downloading the Software

To start, you can navigate to either the MATLAB website or Octave (the free version).

A screenshot of a computer

Description automatically generated

To download MATLAB, you’d need to create an account and buy the software. The website can be found here: [MATLAB download](https://www.mathworks.com/help/install/ug/install-products-with-internet-connection.html)

Below is the free version: Octave.

A screenshot of a computer

Description automatically generated

You must navigate to <octave.org>. It is compatible with GNU/Linux, MacOS and Windows. You will need to choose your operating system and download the software.

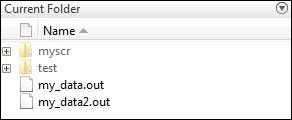
# Navigating the software

The command window is where you can get started and type numerous commands.

A close-up of a computer screen

Description automatically generated

You may see a current folder tab which shows your current files.



You will be able to navigate through all your files through the file browser too.

A white screen with a black text

Description automatically generated with medium confidence

The workspace is where all the values you create will appear. Their name, class, dimension, value and attribute will appear here.

A screenshot of a computer

Description automatically generated

# Computing operations

Say you want to discover the answer of 5 + 5. You would navigate to the command windows and enter exactly that, and you would be greeted with “ans = 10”.

A screenshot of a computer

Description automatically generated

The same goes for which would be inputted with the “^” symbol.



Inputting trigonometry functions is easy as typing the abbreviation and the angle which outputs an answer.



MATLAB has some quirks to it. INF stands for infinity, pi typed out will be used instead of the symbol (for ease) and nan stands for not a number. There are some more than can be found as you navigate through the program.

# Using the semicolon

Just like in certain programming languages (ahem, Java), MATLAB uses the semicolon to end statements.

A number and symbols on a white background

Description automatically generated

3 is assigned to x here and y takes that x value adding 5 to it. The final value for y is 8.

# Adding Comments

To add a comment, you would use the % symbol next to your commands.

A black and white text

Description automatically generated

# Saving, Clearing & Loading

To save a file, you would use the “save” command and put the file name you desire after it. For example, below the file was just named “file” but it can be anything.

A screenshot of a computer

Description automatically generated

To load a file, you would need to use the load command.

A screenshot of a computer

Description automatically generated

These files will show up in the file browser.

A computer screen shot of a computer screen

Description automatically generated

Clearing the command window is as simple as using the “clc” command.

A grey and white text

Description automatically generated

# Declaring Variables

Declaring a variable is as simple as using a variable letter and assigning it a value.

A grey and white text

Description automatically generated with medium confidence

Here 7 is assigned to x.



Here the square root of 64 (8) is assigned to y.

A yellow tape with black text

Description automatically generated with medium confidence

Here the value of y multiplied the value of x is assigned to z which is 56.

The “who” command displays all the variables.

A black text on a white background

Description automatically generated

# Formatting decimals

You can use “format long” to get a long decimal value like below.

A black text with black numbers

Description automatically generated with medium confidence

You can use “format short” to get a short decimal value like below.

A black text with black letters

Description automatically generated with medium confidence

The “format bank” command cuts the decimal to only two values.

A black text with black letters

Description automatically generated

The “Short e” command turns the value in a short exponential form.

A black and white text

Description automatically generated with medium confidence

The “Long e” command turns the value into a long exponential form.

A black and white text

Description automatically generated with medium confidence

# Creating vectors and matrices

Below shows the easy method of creating a vector. It can be created by naming it (c is the name here) and then providing the numbers with colons in between them.

A number and a mathematical equation

Description automatically generated with medium confidence

The Matrix is similar except each column is written before followed by the colon, turning into multiple rows after every colon.

A screenshot of a computer

Description automatically generated

# Helpful Commands

|  |  |
| --- | --- |
| Command | Purpose |
| clc | Clears command window. |
| clear | Removes variables from memory. |
| exist | Checks for existence of file or variable. |
| global | Declares variables to be global. |
| help | Searches for a help topic. |
| lookfor | Searches help entries for a keyword. |
| quit | Stops MATLAB. |
| who | Lists current variables. |
| whos | Lists current variables (long display). |

(Check other sections for more commands or visit <https://www.tutorialspoint.com/matlab/matlab_commands.htm> )

# Data Types

The main types of data values in MATLAB are:

Int, single, double, logical, and char.

**Int** comes in many types:

Int8, int16, int32 and int64. There’s also “u” or unsigned versions of each of these too.

**Double** refers to decimal numbers.

**Logical** is 0 or 1 values.

**Char** is character values.

# Operation symbols

Operation symbols

|  |  |
| --- | --- |
| 1 | **+**  Addition. Adds numbers or variables together. |
| 2 | **-**  Subtraction. Subtracts numbers or variables together. |
| 3 | **\***  Matrix multiplication. C = A\*B is the linear algebraic product of the matrices A and B. |
| 4 | **.\***  Array multiplication. A.\*B. |
| 5 | **/**  Slash or matrix right division. |
| 6 | **./**  Array right division. A./B. |
| 7 | **\**  Backslash or matrix left division. |
| 8 | **.\**  Array left division. A.\B. |
| 9 | **^**  Matrix power. X^p is X to the power. |
| 10 | **.^**  Array power. A.^B. |
| 11 | ‘  Matrix transpose |
| 12 | .’  Array transpose |

Comparison symbols

|  |  |
| --- | --- |
| 1 | **<**  Less than |
| 2 | **<=**  Less than or equal to |
| 3 | **>**  Greater than |
| 4 | **>=**  Greater than or equal to |
| 5 | **==**  Equal to |
| 6 | **~=**  Not equal to |
| 1 | **<**  Less than |

# If-then Statements

The syntax of an if-statement is:

if <expression>

% statement(s) will execute if the boolean expression is true

<statements>

end

Here shows an example where a is 15. Then, an if-then statement is written. If a is less than 20, then “hi” will be printed. However, since a isn’t less than 20, the if statement ends.

A white background with black text

Description automatically generated

# For-loops and While Loops

The syntax of a for loop:

for index = values

<program statements>

...

end

Here’s an example that shows the loop iterating over the list and displaying it.

A white background with black text

Description automatically generated

The syntax of a while loop:

while <expression>

<statements>

end

# Arrays

To create an array of all ones, the command

ones(4,3) can be used. This will show 4 columns and 3 rows.

To create an array of all zeros, the command

zeros(5) can be used. This will create an array of 5 rows and 5 columns.

To create an identity matrix, the command

eye(4) can be used. This’ll display the diagonal of 4 ones.

To create a random filled array, the command

rand(3, 5) can be used. It produces 3 columns and 5 rows.

# Functions

Nested function syntax:

function outerFunction()

% Outer function code function

innerFunction()

% Nested function code

end

% Outer function code can call innerFunction

End

Nested functions allow you to define functions within other functions.

The return statement helps stop a function’s execution.

Void functions don’t have any sort of meaningful return type.

There are also local and global functions.

# Plotting

Plotting is as simple as defining x and y across a range and plotting it.

For example,

X = [0:5:100]

Y = x;

Plot(x,y)

This can be done across multiple functions.

So yeah, that’s MatLab.