

# MATLAB Cheat Sheet

## Useful Commands

clc	clear command window
clear	clear workspace
clear x	clear x from workspace
close all	close all figures
help function	help for function
doc function	documentation for function
quit, exit	terminates MATLAB
edit	opens MATLAB editor
functionName	
format	sets command window output format
short/long/compact	

## Built in Functions/Constants

pi	constant $\pi$
exp()	exponential function
eps()	floating point accuracy
le6	$1 \times 10^6$
Inf	infinity value
NaN	not-a-number value
li, lj	imaginary unit

## Vectors/Matrices Creation

x = [1 2 3]	row vector $1 \times 3$
x = [1; 2; 3]	column vector $3 \times 1$
x = [1 2; 3 4]	matrix $2 \times 2$
a:b	row vector $[a, a+1, \dots, b]$
a:x:b	row vector $[a, a+x, \dots, b]$
linspace(a, b, n)	n evenly spaced points between a and b
logspace(a, b, n)	n logarithmically spaced points between $10^a$ and $10^b$
ones(n, m)	matrix of values 1, $[n \times m]$
zeros(n, m)	matrix of values 0, $[n \times m]$
eye(n)	identity matrix, $[n \times n]$
NaN(n, m)	matrix of values NaN, $[n \times m]$
Inf(n, m)	matrix of values true, $[n \times m]$
true(n, m)	matrix of logical ones, $[n \times m]$
false(n, m)	matrix of logical zeros, $[n \times m]$
rand(n, m)	matrix of random uniformly distributed values, $[n \times m]$ (see also randi and randn)
diag()	diagonal elements of matrix, vector -> square matrix

## Vectors/Matrices Operations

a + b	element-wise addition
a - b	element-wise subtraction
a * b	matrix multiplication
a .* b	element-wise multiplication
a / b	matrix division, mrdivide
a \ b	mldivide
a ./ b	element-wise division
a ^ b	matrix power
a .^ b	element-wise power
a.'	transpose
a'	transpose + complex conjugate (Hermitian transpose)
[A B]	concatenates 2 matrices horizontally
[A; B]	concatenates 2 matrices vertically
cat(dim, A, B)	concatenates 2 matrices
length(A)	size of longest dimension of A
size(A)	size of A
ndims(A)	number of dimensions of A
numel(A)	number of elements of A
tril(A)	lower triangular part of matrix
triu(A)	upper triangular part of matrix
repmat(A, x, y)	repeats copy of matrix
reshape(A, x, y)	changes dimensions of matrix
squeeze(A)	removes dimensions of size 1
flip(A, dim)	flips array (see also fliplr and flipud)
circshift(A, K, dim)	circularly shifts array
det(A)	matrix determinant
inv(A)	matrix inversion
roots(vec)	polynomial roots
round(A)	round to nearest decimal integer (see also ceil, floor, and fix)
mod(A, n)	modulo operation
rem(A, n)	remainder after division

## Logical Operators

&	logical AND
&&	short-circuit logical AND
	logical OR
	short-circuit logical OR
~	not
all	function all
any	function any
is*	several function with logical output

## Relation Operators

>	greater then
>=	greater then or equal to
<	less then
<=	less then or equal to
==	equal to
~=	not equal to

## Vector Indexing

a(1)	the first element in vector
a([1, 2, 5])	the first, second and fifth element in vector
a(1:3)	the first three elements of an vector
a([1:3 5:6])	selected elements of a vector
a(end)	the last element of the vector
a(5:end)	from the fifth to the last element in vector
a(5:end-1)	from the fifth to the penultimate element in vector
a(1) = 10	replace the first element in vector by 10
a([1 3 5]) = [2 5 7]	replace elements in vector by values 2, 5, and 7
a([1 3 5]) = []	discard elements on position 1, 3, and 5

## Matrix Indexing

A(1, 1)	an element in the first row and first column of matrix
A([1 5], 2:4)	elements of matrix in the first and fifth rows and in columns two, three and four
A(n, :)	the nth row in array
A(:, n)	the nth column in array
A(n, :) = []	discard nth row in array
A(:, n) = []	discard nth column in array
A(1)	the first element in array (linear indexing)
A(1:3)	the first three elements of an array (linear indexing)

## Logical Indexing

A(A == 3)	extract elements equal to 3
A(A > 5)	extract elements greater than 5
A(A > 5 & A < 30)	extract elements greater than 5 and at the same time less than 30

## Debugging and Time Measurement

keyboard	pauses execution
dbcont	resume execution (end debug mode)
dbclear all	removes all breakpoints
tic, toc	start/stop time measurement
profile	profile code execution
on/off/clear/viewer	

## Loops and Branching

<code>if</code> expression	program branching using if-else
...	
<code>elseif</code> expression	
...	
<code>else</code> ...	
<code>end</code>	
<code>switch</code> variable	program branching using switch
<code>case</code> value1	
...	
<code>case</code> {value2, value3}	
...	
<code>otherwise</code> ...	
<code>end</code>	
<code>for</code> n = 1:10	for cycle
...	
<code>end</code>	
<code>while</code> expression	while cycle
...	
<code>end</code>	
<code>break</code>	terminates execution of loop
<code>continue</code>	pass control to next iteration
<code>return</code>	return to invoking function

## Functions

<code>function</code> [out1, out2] = foo(in1, in2)	function definition
@sin	handle function definition
@(x) x^2 + sin(x)	anonymous function definition
nargin	returns number of inputs
nargout	returns number of outputs
varargin	input variable allowing multiple inputs
varargout	output variable allowing multiple outputs

## Set operations

<code>intersect</code> (A, B)	set intersection of two arrays
<code>union</code> (A, B)	set union of two arrays
<code>setdiff</code> (A, B)	set difference of two arrays
<code>setxor</code> (A, B)	set exclusive or of two arrays
<code>unique</code> (A)	unique values in array
<code>sort</code> (X)	sort array elements
<code>sortrows</code> (X)	sorts rows of matrix
<code>issorted</code> (X)	determines if the array is sorted
<code>ismember</code> (A, B)	array elements that are members of set array

## Basic Visualization

<code>figure</code>	opens empty figure
<code>axes</code>	creates Cartesian axes
<code>plot</code> (x, y)	2D line plot
<code>hold on/off</code>	retain current plot with new line
<code>grid</code>	display/hide grid lines
<code>on/off/minor</code>	
<code>title</code> (txt)	adds title to figure
<code>xlabel</code> (txt)	adds x-axis label (same for y and z)
<code>xlim</code> ([min, max])	sets limits of x-axis (same for y and z)
<code>colormap</code>	view colormap
<code>stem</code> (x, y)	discrete stem plot
<code>pcolor</code> (A)	displays array data as colored cells
<code>surf</code> (X, Y, Z)	displays surface plot
<code>semilogx</code> (X)	semi-logarithmic plot (same for y)
<code>image</code> (X)	Show image from array
<code>doc</code> <code>LineStyleSpec</code>	plot parameters to customize curves
<code>set</code> (ref, Name, Value)	sets graphics object property
<code>get</code> (ref, Name)	query graphics object property value
<code>ref.Name = Value</code>	dot notation, sets graphics object property
<code>Value = ref.Name</code>	dot notation, query graphics object property value
<code>view</code> (az, el)	camera line of sight

## Character arrays and Strings

<code>'Hello world!'</code>	character vector is created using single quotation marks
<code>A = char</code> (B)	convert another data type to character array
<code>strfind</code> (str, pattern)	find string within other string
<code>strcmp</code> (str1, str2)	compare strings
<code>strjoin</code> (str, delimiter)	join strings in array
<code>strtok</code> (str, delimiter)	split string into parts
<code>regexp</code> (str, reg)	match regular expression
<code>"Hello world!"</code>	string is created using double quotation marks
<code>A = string</code> (B)	convert another data type to string

## File Handling

<code>dir</code> <code>folderName</code>	lists folder content
<code>pwd</code>	identifies current folder
<code>exist</code> <code>name</code>	check existence of variable, function, folder, ...
<code>cd</code> ( <code>path</code> )	changes current folder
<code>mkdir</code> ( <code>path</code> , <code>folderName</code> )	makes new folder
<code>rmdir</code> ( <code>path</code> )	remove folder

<code>writematrix</code> (A, <code>filename</code> )	writes matrix to file (see also <code>readmatrix</code> )
<code>imwrite</code> (A, <code>filename</code> )	writes image to graphics file (see also <code>imread</code> )
<code>fid =</code> <code>fopen</code> ( <code>filename</code> )	open file
<code>fclose</code> (fid)	close opened file
<code>fgetl</code> (fid)	read line from file
<code>fread</code> (fid)	read from binary file
<code>fwrite</code> (fid, A)	write to binary file

## Other Data Types

<code>A = cell</code> (m, n)	create cell array
<code>A = {x, y, z}</code>	create cell array
<code>A(1, 1)</code>	smooth parentheses indexing, creates subset cell array
<code>A{1, 1}</code>	curly brackets indexing, access to data in cell array
<code>B =</code> <code>struct</code> ( <code>field</code> , value)	create structure array
<code>B.field = value</code>	create structure array
<code>double</code> (A)	converts array to double precision (see also <code>single</code> )
<code>int8</code> (A)	converts array to 8-bit signed integer (see also <code>int16</code> , <code>int32</code> , <code>uint8</code> , <code>int18</code> , ...)
<code>datetime</code> ()	array representing points in time (see also <code>years</code> , <code>days</code> , <code>hours</code> , ...)
<code>sparse</code> (A)	creates sparse matrix

## Other useful functions

<code>find</code> (X)	find indices of nonzero elements
<code>fprintf</code> (fileID, format, A1)	write data to text file
<code>sprintf</code> (format, A1)	format data into character vector
<code>eval</code> (txt)	executes MATLAB expression in text
<code>feval</code> (fun, x1)	evaluate function
<code>str2num</code> (X)	convert character array to numeric array
<code>str2double</code> (X)	convert characters to double precision numbers
<code>num2str</code> (X)	convert numeric array to character array
<code>realmax</code>	largest positive floating-point number (see also <code>realmin</code> , <code>intmax</code> , and <code>realmin</code> )
<code>interp1</code>	1-D interpolation (see also <code>interp2</code> , <code>interp3</code> , and <code>interpN</code> )
<code>fminbnd</code>	find minimum of single-variable function
<code>fft</code>	Fast Fourier transform (see also <code>fft2</code> , <code>fftn</code> , <code>ifft</code> , <code>fftshift</code> , ...)
<code>timer</code>	creates object of timer to schedule execution of MATLAB commands (see documentation)

## Shortcuts Command Window

ENTER	line execution
ESC	delete whole line
DEL	delete one character (right to the cursor)
BACKSPACE	delete one character (left to the cursor)
HOME	moves cursor to the beginning of line
END	moves cursor to the end of line
CTRL + ↑	moves cursor to the beginning of next word
CTRL + ↓	moves cursor to the beginning of previous word
SHIFT + ENTER	sends cursor to the next line (without execution)
CTRL + K	deletes all to the right of cursor
CTRL + C	forces interruption of MATLAB (e.g. long/erroneous calculation)
CTRL + TAB	switching between windows of MATLAB environment
↓ and ↑	command history listing (searching in available CTRL + F)
F1	context help related to the word where the cursor is placed
TAB	function or variable name hint

## Shortcuts MATLAB Editor

CTRL +	moves between tabs
PG.UP/PG.DOWN	
CTRL + R	comment lines
CTRL + T	uncomment lines
CTRL + D	open highlighted m-file (function/script)
F5	execute current script/function
CTRL + S	save current file (done automatically after pressing F5)
CTRL + HOME	jump to the beginning of file
CTRL + END	jump to the end of file
CTRL + →/←	jump word-by-word or expression-by-expression to the right/left
CTRL + W	close current file
CTRL + O	activates open file dialog box
CTRL + F	find/replace dialog box
CTRL + G	“go to”, jumps to the indicated line number
CTRL + I	indent of block of lines corresponding to key words
F1	open context help related to the function at position of the cursor
F9	run highlighted code

## MATLAB File Suffix (not all)

.m	script/function/class
.mlx	MATLAB live script
.p	protected MATLAB code
.mat	binary data file
.fig	MATLAB figure
.mdl, .slx	Simulink model
.mdl, .slx	Simulink protected model
.mexa64,	mex libraries
.mexmaci64,	
.mexw32,	
.mexw64	
.mlappinstall	APP file – installer
.mlpkginstall	support package – installer
.mltbx	toolbox file – installer
.mn	MuPAD notebook
.mu	MuPAD code

## Handle Graphics

```
fig = figure
set(ref, Name, Value)
get(ref, Name)
ref.Name = Value
Value = ref.Name
```

```
groot, get(0)
axes
axis(lims)
uimenu
uicontextmenu
uitoolbar
uipanel
uitabgroup
uitab
uitable
uibuttongroup
```

```
uicontrol
```

```
uicontrol('Style', style)
uicontrol('Style', 'Text')
uicontrol('Style', 'Edit')
uicontrol('Style', 'Pushbutton')
uicontrol('Style', 'Togglebutton')
uicontrol('Style', 'Radiobutton')
uicontrol('Style', 'Checkbox')
uicontrol('Style', 'Listbox')
uicontrol('Style', 'Popupmenu')
uicontrol('Style', 'Slider')
```

```
set(ref, 'Callback', @foo)
set(ref, 'Callback', @(src,
event) foo(in1))
set(ref, 'Callback', {@foo, in1})
```

```
gcf
```

```
gca
```

```
gco
```

```
findobj(property, value)
findall
allchild(ref)
copyobj(ref)
delete(ref)
```

```
setappdata(ref, field, data)
getappdata(ref, field)
```

```
msgbox(mess)
```

```
get figure reference
sets graphics object property
query graphics object property value
dot notation, sets graphics object property
dot notation, query graphics object prop-
erty value
```

```
graphics root object (screen)
creates Cartesian axes
scales axes
creates menu or menu item
creates context menu
creates toolbar with icons
creates panel; container for other objects
creates tabgroup; container for other tabs
creates a tab; container for other objects
creates 2D table
creates superior object for group of buttons
(i.e. radiobuttons)
creates fundamental element of GUI, be-
havior depends on style of the object
changes style of uicontrol
creates text field
creates editable text field
creates one-state button
creates two-state button
creates radio button
creates checkbox
creates list of items
creates popup menu with multiple choices
creates slider
```

```
set callback as function handle
set callback as anonymous function
```

```
set callback as function handle in cell array
```

```
returns reference of current figure (last
active)
```

```
returns reference of current axes (last ac-
tive)
```

```
returns reference of last "mouse-clicked"
object
```

```
finds object(s) with required property
finds all graphics objects
finds all children objects of selected object
copies an object
deletes object
```

```
defines stored data inside the application
retrieves previously stored data
```

```
creates message dialog window
```

## App Building (new graphics)

```
uifigure
uiaxes
uipanel
uitab
uigridlayout
```

```
uibuttongroup
```

```
uimenu
uitable
uilabel
uieditfield
uitextarea
uibutton
uiradiobutton
uicheckbox
uiliblistbox
uidropdown
```

```
uislider
uispinner
uiswitch(style)
```

```
uiknob(style)
```

```
uigauge(style)
```

```
uilamp
uitree
uidatepicker
uiimage
uihtml
```

```
ref.ButtonPushedFcn = @foo
ref.ButtonPushedFcn = @(src,
event) foo(in1)
ref.ButtonPushedFcn = {@foo, in1}
```

```
setappdata(ref, field, data)
getappdata(ref, field)
```

```
uialert
uiconfirm
```

```
creates graphics window
creates axes in uifigure
creates panel; container for other objects
creates tab; container for other objects
layout manager; positioning of graphics
components
creates superior object for group of buttons
(i.e. radiobuttons)
creates menu or menu item
creates 2D table
creates text field
creates editable text field
creates multiple line editable text
creates one-state or two-state button
creates radio button component
creates checkbox component
creates list of items
creates drop-down component (popup
menu)
creates slider component
creates spinner component
creates switch object (possible styles:
'slider', 'rocker', and 'toggle')
creates knob component (possible styles:
'continuous' and 'discrete')
creates gauge component (possible styles:
'circular', 'linear', 'ninetydegree', and
'semicircular')
creates lamp component
creates tree component
creates calendar component
creates image component
creates HTML component
```

```
set callback as function handle
set callback as anonymous function
set callback as function handle in cell array
```

```
defines stored data inside the application
retrieves previously stored data
```

```
predefined alert window
predefined confirmation window
```