MATLAB Cheat Sheet

MATLAB R2026b (announced 2026-09-28) · checked 2026-10-22 · base MATLAB only · examples use the `heatflow` plate scenario · AI-assisted content: verify against the official documentation · albertoirurueta.github.io/docs

Environment

`help f` / `doc f` / `lookfor kw`; `which -all f` shows every definition (shadowing); `whos`, `who`, `clear x`, `clc`, `close all`; `format long/short` (display only)

`save f.mat x y`, `load f.mat`, `matfile` (partial access); `addpath`, `path`, `pwd`, `cd`; `startup.m` runs at launch; batch: `matlab -batch "runtests"`

R2025a web-based desktop; R2026b: no Java installed, live scripts plain-text `.m` by default.

Syntax

```
x = 3;        % ; hides output, ans = last
%{ block comment %}   ... continue line
%% section  (Ctrl+Enter runs it)
hold on       % command syntax: words
hold("on")    % function syntax: same call
a = 1; b = 2, c = 3;  % , shows, ; hides
```

Names: letter then letters/digits/_; case-sensitive; `iskeyword` lists keywords; avoid shadowing (`sum`, `max`). Values are copied lazily (copy-on-write).

Arrays

```
A = [1 2; 3 4];   v = 1:5;  w = (0:4)/4;
linspace(0,1,5)  zeros(2,3)  ones(2)
eye(3)
rand(3) randn(3) randi(9,2)  repmat(A,2,2)
[A B] horzcat  [A; B] vertcat  cat(3,A,A)
reshape(A,1,[])  permute(X,[2 1 3])  A.'
A'
size  numel  ndims  length  isempty
isvector
```

Column-major: `A(:)` reads down column 1, then column 2. 1-based. `[]` is 0-by-0; `zeros(0,3)` is not `[]`.

Indexing

```
A(i,j) A(k) A(end,:) A(:,2) A(:) A(2:end-1)
A(A>2)=0          % logical mask
A(2,:)=[]          % delete row
[r,c]=find(A>2)  i=sub2ind(size(A),r,c)
A(5,5)=1           % grows, pads with 0
```

	`()`	`{}`	`.`
array	elements	–	–
cell	sub-cell	contents	–
struct	element	–	field
table	sub-table	block as array	variable

`c{:}` and `s.f` on struct arrays make comma-separated lists: `[c{:}]`, `f(c{:})`.

Operators

```
+ - * / \ ^   matrix      .* ./ .^
element-wise
'  conj-transpose   .'  transpose
== ~= < <= > >=   & | ~  element-wise
&& ||  scalar short-circuit   xor(a,b)
```

Precedence, high to low: `()`; transpose, power; unary `+ - ~`; `* / \ .* ./`; binary `+ -`; `:`; relational; `&`; `|`; `&&`; `||`.

Implicit expansion (R2016b): sizes are compatible if each dimension is equal or 1.

```
M = [1 2 3] + [10; 20]   % 2-by-3
bsxfun not needed; [1 2 3]+[1 2] is an
error
```

Types

Type	Notes
`double` `single`	IEEE 754; default double; `eps`, `Inf`, `NaN`; 0.1+0.2~=0.3
`int8`..`uint64`	saturate (`uint8(300)` = 255), round to nearest; `intmax`
`logical`	`true` `false`; masks
`char` / `string`	`'ab'` vs `"ab"`
`cell` `struct`	`{1,"a"}`; `s.f`, `s.("f")`
`table` `timetable`	`T.Var`, `T(rows,:)`; time-stamped rows
`datetime` `duration`	`datetime("now")`, `days(2)`
`categorical`	`categorical(["a" "b"])`
`dictionary` (R2022b)	`d = dictionary("k",1); d("k")`
`missing` `NaN` `NaT`	`ismissing`, `rmmissing`, `fillmissing`
`@` handle	`f = @sin`

`class(x)`, `isa(x,"numeric")`, `cast(x,"int16")`, `double(x)`.

Text

```
s = "heat" + "flow"      % string: + joins
c = ['ab' 'cd']          % chars: [ ] joins
compose("T=%d",3)  sprintf("%6.2f",pi)
split(s,"-") join(["a" "b"],",")
contains(s,"he") replace(s,"he","HE")
strlength upper strip extractBefore
regexp(s,'(\w+)flow','tokens')
p = letterBoundary + digitsPattern;
extract(s,p)
str2double("3.5")  string(7)  num2str(7)
```

`"..."` arrays can hold any text and missing (`<missing>`); `'...'` is a numeric row of chars.

Control flow

```
if a>0, ... elseif a==0, ... else, ... end
switch s, case {"a","b"}, ... otherwise,
... end
for k = A      % iterates over COLUMNS
while cond, ... break / continue, end
return    % leave function
try, f(), catch ME, disp(ME.message), end
```

`if` on an array is true only if all elements are nonzero; use `any/all`.

Functions

```
function [y,n] = f(x,varargin)   % f.m
  arguments
    x (1,:) double {mustBePositive}
    opts.Tol (1,1) double = 1e-6
    opts.Mode (1,1) string
{mustBeMember(opts.Mode,["a","b"])} = "a"
  end
  n = nargin;  y = x.^2;
end
function h = local(x), end     % local fn
(same file)
y = f(1:3, Tol=1e-3)           % Name=Value
R2021a
g = @(x) x.^2 + 1;  g(2)  fn = @f
feval(fn,2)
[~,i] = max(v)   nargin nargout varargout
global G  persistent P  % scope: avoid
global
```

Nested functions share the parent's variables; `private/` folder fns visible to parent only. Resolution order: variable, nested/local, private, class/namespace, path. `mustBeBetween` arrived in R2025a.

Errors & debugging

```
error("pkg:id","bad %d",x)
warning("pkg:w","...")
assert(x>0,"pkg:neg","x=%d",x)
try, f(), catch ME, rethrow(ME), end
ME = MException("a:b","msg");  throw(ME)
c = onCleanup(@() fclose(fid));
dbstop if error   dbstep dbcont dbquit
codeIssues("src")  % Code Analyzer
```

classdef

```
classdef Plate < handle     % value if no <
handle
  properties (SetAccess=private)
    T (:,:) double
  end
  properties (Dependent), Mean, end
  methods
    function obj = Plate(T), obj.T = T; end
    function m = get.Mean(obj), m =
mean(obj.T,"all"); end
  end
  events StepCompleted, end
  enumeration, Dirichlet, Neumann, end
end
addlistener(p,"StepCompleted",@(s,e)
disp("ok")); notify(p,"StepCompleted")
```

Folders: `+heatflow/` namespace (R2024a name), `@Plate/` class folder. `?Plate` metaclass. Handle copies alias; value copies are independent.

Graphics

```
plot(x,y,"r--o","LineWidth",1.5); hold on
tiledlayout(2,1); nexttile; scatter(x,y)
title xlabel ylabel legend  histogram(v)
[X,Y]=meshgrid(x,y); surf(X,Y,Z);
contour(X,Y,Z)
ax = gca; ax.XLim = [0 1];   % dot notation
exportgraphics(gcf,"f.png",Resolution=300)
```

Data I/O

```
T = readtable("sensors.csv");
writetable(T,"o.csv")
M = readmatrix("m.csv")
datastore("*.csv")
fid=fopen("f.txt","w");
fprintf(fid,"%d\n",5); fclose(fid);
s = jsondecode(txt); jsonencode(s)
cfg = readstruct("plate.json");
webread(url)
```

Maths

```
x = A\b   (not inv(A)*b)   det  rank  norm
cond
[V,D]=eig(A)  [U,S,W]=svd(A)  lu qr chol
S = sparse(i,j,v,m,n)  spy  nnz
p = polyfit(x,y,2); polyval(p,x);
interp1(x,y,xi)
integral(@f,0,1)  [t,y]=ode45(@f,[0 1],y0)
fzero  fminsearch  fft/ifft  rng(1)
mean(x,"omitmissing")
```

Performance

Vectorise; preallocate (`zeros(1,n)`); avoid growing in loops. `tic; toc`, `timeit(@f)`, `profile on; profile viewer`. `backgroundPool`, `parfeval`, `timer`, `tall` arrays.

Testing & build

```
% function test file
function tests = fooTest, tests =
functiontests(localfunctions); end
function t1(tc),
verifyEqual(tc,f(2),4,AbsTol=1e-9); end
% class test: classdef FooTest <
matlab.unittest.TestCase
%   methods (Test), function t(tc), ...
end, end
runtests("tests")
% buildfile.m
function plan = buildfile
  plan = buildplan(localfunctions);
  plan("test") =
matlab.buildtool.tasks.TestTask("tests");
end   % run: buildtool test
```

Interop

`py.math.sqrt(2)`, `pyenv(Version="3.13")`, `pyrun("x=1")`; R2026b supports Python 3.10 to 3.14 (3.14 new, 3.9 dropped). Python side: `eng = matlab.engine.start_matlab()`, `pip install matlabengine`. C/C++: `mex f.c`, `clibgen.generateLibraryDefinition`. `javaObject` (add OpenJDK in R2026b), `NET.addAssembly`, `system("ls")`.

Tools & sharing

Live scripts: plain-text `.m` (R2025a, default R2026b) or binary `.mlx`. App Designer (GUIDE removed R2025a). Projects with `matlab.toml` (R2026b). `mpminstall/mpmcreate` (MATLAB Package Manager, R2024b). `pcode f.m` hides source (not secure). MATLAB Vault for secrets. GNU Octave runs much code but not all; Octave 11.3.0.

Legacy → modern (release) `bsxfun` → implicit expansion (R2016b) · `strmatch`/char rows → `string` (R2016b) · `datenum` → `datetime` · `inputParser`/`nargchk` → `arguments` (R2019b) · `containers.Map` → `dictionary` (R2022b) · packages → namespaces (R2024a) · GUIDE → App Designer (R2025a) · `.mlx` → plain-text live code (R2025a/R2026b) · toolbox → package (R2026b) · `hist` → `histogram` · `quad` → `integral` · `rand('seed')` → `rng` · `mustBeInRange` → `mustBeBetween` (R2025a)