

Referensi Cepat MATLAB

Array, matriks, plotting, I/O file, alur kontrol

Dasar

Command Window

```
x = 5;           % assign (semicolon suppresses output)
x = 5           % assign and display result
disp('Hello')   % print to console
clc             % clear command window
clear          % clear all variables
```

Bantuan & Info

```
help sin        % quick help for function
doc sin         % open documentation
who             % list variables in workspace
whos            % list with details (size, type)
```

Operator

```
+ - * / ^       Aritmatika (operasi matriks)
.* ./ .^       Operasi element-wise
== ~= < > <= >= Operator perbandingan
&& || ~        AND, OR, NOT logika (skalar)
& | ~         Logika element-wise (array)
```

Variabel & Tipe

Tipe Numerik

```
x = 3.14;       % double (default)
n = int32(42);  % 32-bit integer
z = 2 + 3i;     % complex number
tf = true;      % logical
```

Pemeriksaan Tipe

```
class(x)        Kembali nama tipe sebagai string
isa(x, 'double') Periksa apakah x adalah tipe tertentu
isnumeric(x)    True jika tipe numerik
ischar(x)       True jika array karakter
islogical(x)    True jika tipe logika
```

Konstanta Khusus

```
pi              3.14159...
Inf / -Inf      Tak hingga
NaN             Bukan Angka
eps             Epsilon mesin (~2.2e-16)
i / j           Satuan imajiner
```

Array & Matriks

Membuat Array

```
v = [1 2 3 4 5]; % row vector
v = [1; 2; 3];   % column vector
A = [1 2; 3 4];  % 2x2 matrix
r = 1:5;         % [1 2 3 4 5]
r = 0:0.5:2;     % [0 0.5 1 1.5 2]
```

Konstruktor Bawaan

```
zeros(3)        % 3x3 of zeros
ones(2, 4)      % 2x4 of ones
eye(3)          % 3x3 identity
rand(2, 3)      % 2x3 uniform random
linspace(0,1,5) % 5 evenly spaced [0..1]
```

Pengindeksan & Slicing

```
A(2, 3)         % row 2, col 3
A(1, :)         % entire first row
A(:, 2)         % entire second column
A(1:2, 1:2)     % submatrix
A(end, :)       % last row
```

Operasi Matriks

```
A'              Transpos (konjugat)
A.'             Transpos (tanpa konjugat)
inv(A)          Invers matriks
det(A)          Determinan
eig(A)          Nilai eigen dan vektor eigen
A \ b           Selesaikan Ax = b
size(A)         Dimensi [baris kolom]
numel(A)        Jumlah total elemen
```

Alur Kontrol

if / elseif / else

```
if x > 0
    disp('positive')
elseif x == 0
    disp('zero')
else
    disp('negative')
end
```

for & while

```
for i = 1:10
    fprintf('i = %d\n', i);
end
while x > 0
    x = x - 1;
end
```

switch

```
switch grade
case 'A'
    disp('Excellent')
case {'B', 'C'}
    disp('Good')
otherwise
    disp('Try harder')
end
```

Kontrol Perulangan

```
break          Keluar dari perulangan terdalam
continue       Lanjut ke iterasi berikutnya
return         Keluar dari fungsi segera
```

Fungsi

File Fungsi

```
% Save as myfunc.m
function result = myfunc(x, y)
    result = x.^2 + y.^2;
end
```

Beberapa Output

```
function [mn, mx] = minmax(v)
    mn = min(v);
    mx = max(v);
end
[lo, hi] = minmax([3 1 4 1 5]);
```

Fungsi Anonim

```
f = @(x) x.^2 + 1;
f(3) % returns 10
g = @(x,y) x + y;
arrayfun(f, [1 2 3]) % apply to each element
```

Built-in Berguna

```
sum(v)          Jumlah elemen
mean(v)         Nilai rata-rata
max(v) / min(v) Maksimum / minimum
sort(v)         Urutkan naik
find(v > 3)     Indeks di mana kondisi benar
length(v)       Panjang vektor
```

Plotting

Plot 2D

```
x = 0:0.1:2*pi;
plot(x, sin(x), 'r-', 'LineWidth', 2)
xlabel('x'); ylabel('sin(x)')
title('Sine Wave'); grid on
legend('sin(x)')
```

Plot Ganda

```
hold on
plot(x, sin(x), 'b-')
plot(x, cos(x), 'r--')
hold off
subplot(1,2,1); plot(x, sin(x))
subplot(1,2,2); plot(x, cos(x))
```

Tipe Plot Lain

```
bar(x, y)       Diagram batang
histogram(data) Histogram
scatter(x, y)    Plot sebar
pie(data)       Diagram lingkaran
surf(X, Y, Z)   Plot permukaan 3D
imagesc(A)      Tampilkan matriks sebagai gambar
```

Simpan Gambar

```
saveas(gcf, 'plot.png')
exportgraphics(gcf, 'plot.pdf')
```

I/O File

File Teks

```
data = readmatrix('data.csv');
writematrix(A, 'output.csv')
T = readtable('data.csv');
writetable(T, 'output.csv')
```

File MAT

```
save('workspace.mat') % save all variables
save('data.mat', 'x', 'y') % save specific vars
load('data.mat')       % load into workspace
S = load('data.mat');  % load into struct
```

I/O File Level Rendah

```
fid = fopen('log.txt', 'w');
fprintf(fid, 'Value: %f\n', 3.14);
fclose(fid);
lines = readlines('log.txt');
```

Operasi String

String vs Char Array

```
s = "Hello"; % string (double quotes)
c = 'Hello'; % char array (single quotes)
s + " World" % "Hello World" (string)
[c, ' World'] % "Hello World" (char concat)
```

Referensi Cepat MATLAB

Fungsi String

<code>strlength(s)</code>	Panjang string
<code>upper(s) / lower(s)</code>	Konversi huruf
<code>contains(s, pat)</code>	True jika pola ditemukan
<code>replace(s, old, new)</code>	Ganti substring
<code>split(s, delim)</code>	Pisahkan menjadi array
<code>join(arr, delim)</code>	Gabungkan array string
<code>strip(s)</code>	Hapus spasi di awal/akhir

Pemformatan

```
sprintf('x = %.2f', 3.14159) % "x = 3.14"
fprintf('i = %d\n', 42)    % print to console
num2str(3.14)              % number to string
str2double("3.14")        % string to number
```

Cell & Struct

Cell Array

```
C = {1, 'hello', [1 2 3]}; % mixed types
C{2} % access: 'hello'
C{end+1} = true; % append element
cellfun(@length, C) % apply func to each
```

Struct

```
s.name = 'Alice';
s.age = 30;
s.scores = [90 85 92];
fieldnames(s) % {'name', 'age', 'scores'}
rmfield(s, 'age') % remove field
```

Array Struct

```
people(1).name = 'Alice'; people(1).age = 30;
people(2).name = 'Bob';   people(2).age = 25;
{people.name} % {'Alice', 'Bob'}
[people.age] % [30, 25]
```

Pola Umum

Operasi Vektorisasi

```
% Avoid loops – use vectorization
v = 1:1000;
result = sum(v.^2); % fast
idx = v(v > 500 & v < 600); % logical indexing
```

Operasi Tabel

```
T = table([25;30], ["A";"B"], 'VariableNames', ...
          {'Age', 'Grade'});
T.Age % access column
T(T.Age > 25, :) % filter rows
```

Penanganan Error

```
try
    result = riskyFunction(x);
catch ME
    fprintf('Error: %s\n', ME.message);
end
```

Mengukur Waktu

```
tic
heavyComputation();
toc % prints elapsed time
```