

## 基础

### 命令窗口

```
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
```

### 帮助与信息

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

### 运算符

```
+ - * / ^ 算术运算 (矩阵运算)
.* ./ .^ 元素级运算
== ~= < > <= >= 比较运算符
&& || ~ 逻辑 AND、OR、NOT (标量)
& | ~ 元素级逻辑 (数组)
```

### 变量与类型

#### 数值类型

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

#### 类型检查

```
class(x) 返回类型名称字符串
isa(x, 'double') 检查是否为指定类型
isnumeric(x) 是否为数值类型
ischar(x) 是否为字符数组
islogical(x) 是否为逻辑类型
```

#### 特殊常量

```
pi 3.14159...
Inf / -Inf 无穷大
NaN 非数值
eps 机器精度 (-2.2e-16)
i / j 虚数单位
```

### 数组与矩阵

#### 创建数组

```
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]
```

#### 内置构造函数

```
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]
```

#### 索引与切片

```
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
```

#### 矩阵运算

```
A' 转置 (共轭)
A.' 转置 (不共轭)
inv(A) 矩阵求逆
det(A) 行列式
eig(A) 特征值与特征向量
A \ b 求解 Ax = b
size(A) 维度 [行数 列数]
numel(A) 元素总数
```

### 控制流

#### if / elseif / else

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

#### for 与 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
```

#### 循环控制

```
break 退出最内层循环
continue 跳到下一次迭代
return 立即退出函数
```

### 函数

#### 函数文件

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

#### 多返回值

```
function [mn, mx] = minmax(v)
    mn = min(v);
    mx = max(v);
end
```

```
[lo, hi] = minmax([3 1 4 1 5]);
```

#### 匿名函数

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

#### 常用内置函数

```
sum(v) 元素求和
mean(v) 均值
max(v) / min(v) 最大值 / 最小值
sort(v) 升序排序
find(v > 3) 条件为真的索引
length(v) 向量长度
```

### 绘图

#### 二维绘图

```
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)')
```

#### 多图叠加

```
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))
```

#### 其他图表类型

```
bar(x, y) 柱状图
histogram(data) 直方图
scatter(x, y) 散点图
pie(data) 饼图
surf(X, Y, Z) 三维曲面图
imagesc(A) 将矩阵显示为图像
```

#### 保存图形

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

### 文件 I/O

#### 文本文件

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

#### MAT 文件

```
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
```

#### 低级文件 I/O

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

### 字符串操作

#### 字符串与字符数组

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

#### 字符串函数

```
strlen(s) 字符串长度
upper(s) / lower(s) 大小写转换
contains(s, pat) 是否包含模式
replace(s, old, new) 替换子字符串
split(s, delim) 按分隔符拆分为数组
join(arr, delim) 连接字符串数组
strip(s) 去除首尾空白
```

#### 格式化

```
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 数组与结构体

#### Cell 数组

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

#### 结构体

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

#### 结构体数组

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

### 常见模式

#### 向量化运算

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

#### 表操作

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

#### 错误处理

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

#### 计时代码

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