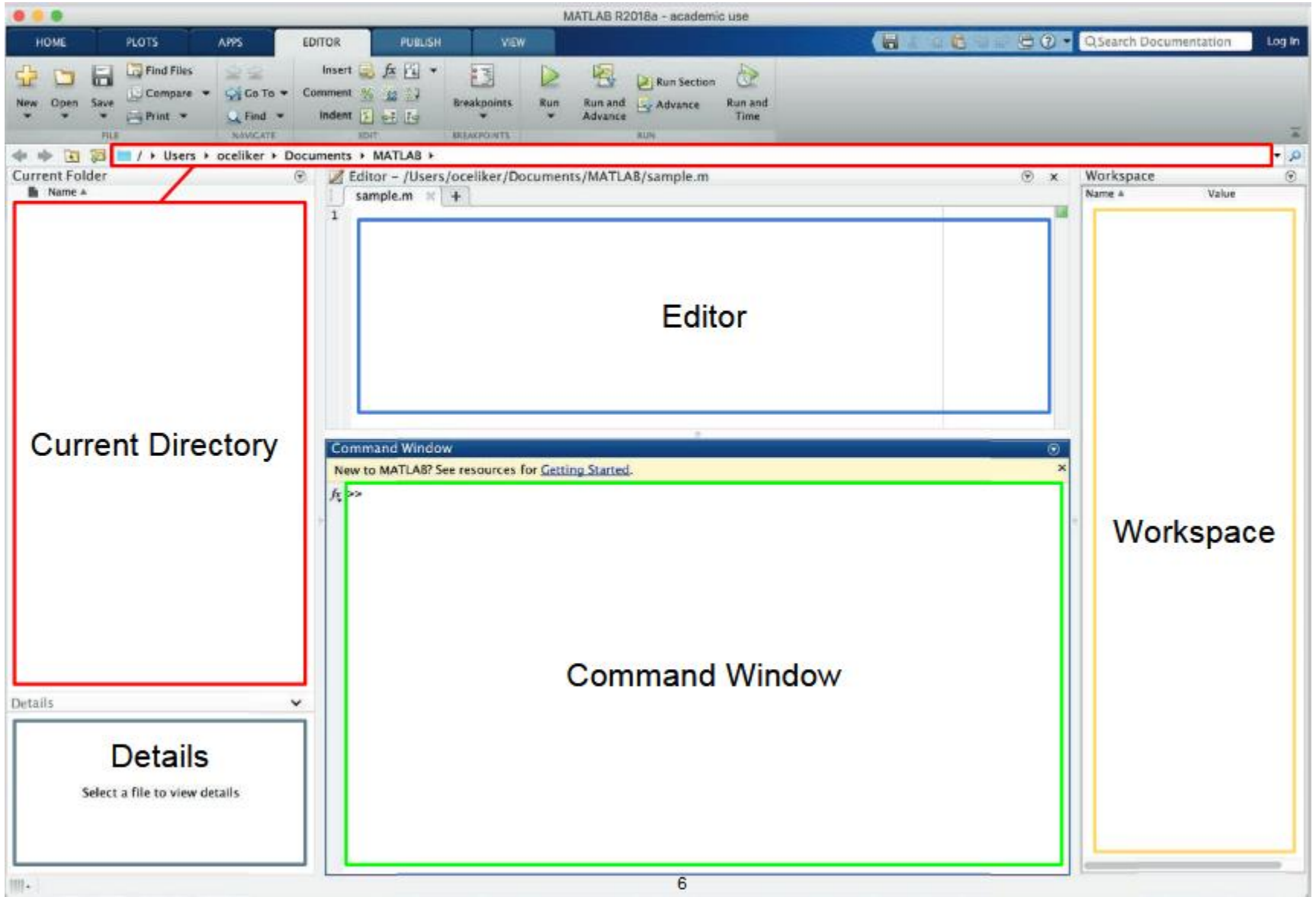
آموزش مقدماتی متلب

Kalakan.ir

سال ۱۴۰۲

آموزش متلب مقدماتی

- Matrix Laboratory
- یک زبان برنامه نویسی سطح بالا است که عناصر کاری آن ماتریس‌ها هستند.
- متغیرها در متلب به صورت پویا تخصیص داده می‌شوند.
- توسط Celve Moler اختراع شد
- نسخه اولیه (قبل از نسخه ۱,۰) متلب یک زبان برنامه نویسی نبود، یک ماشین حساب ماتریسی تعاملی ساده بود. هیچ برنامه، جعبه ابزاری نداشت.
- در دهه ۱۹۸۰، مولر متلب را بر اساس زبان C بازنویسی و زبان متلب ایجاد پایه ریزی شد.


MATLAB R2018a - academic use
HOME
PLOTS
APPS
EDITOR
PUBLISH
VIEW
Search Documentation
Log In
New
Open
Save
Find Files
Compare
Print
Go To
Find
Insert
Comment
Indent
Breakpoints
Run
Run and Advance
Run Section
Advance
Run and Time
/ ▸ Users ▸ oceliker ▸ Documents ▸ MATLAB ▸
Current Folder
Name
Current Directory
Editor – /Users/oceliker/Documents/MATLAB/sample.m
sample.m
1
Editor
Workspace
Name
Value
Workspace
Command Window
New to MATLAB? See resources for Getting Started.
>>
Command Window
Details
Details
Select a file to view details
6

متغیرها

Defining variables

```
int a;
a=1;
double b;
b=2+4;
```

C/C++

```
>>a=1;
>>b=2+4;
```

Matlab

```
>> a=1
a =
     1
>> b=2+4
b =
     6
```

متغیرها

```
a = 1;
```

a <1x1 double>

	1	2
1	1	
2		

```
>> whos a
  Name        Size                Bytes  Class     Attributes

  a           1x1                     8  double
```

```
b = false;
```

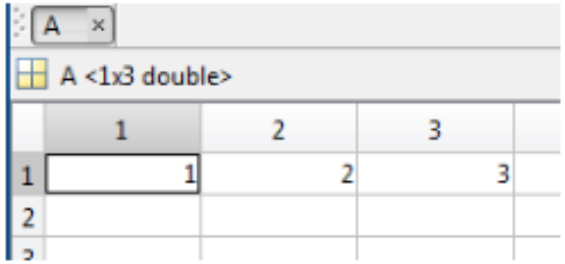
b <1x1 logical>

	1	2
1	0	
2		

متغیرها

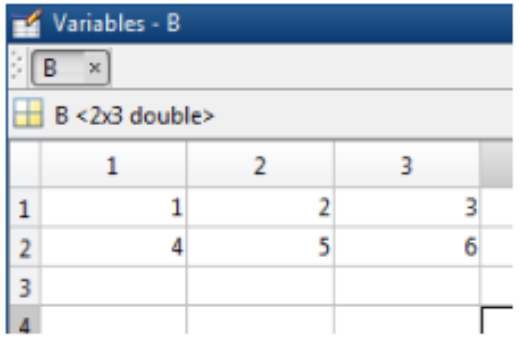
A = [1, 2, 3]

```
>> A=[1 2 3]
A =
     1     2     3
```



B = [1,2,3;4,5,6]

```
>> B=[1,2,3;4,5,6]
B =
     1     2     3
     4     5     6
```



C=[1 2 3;4 5 6;7 8 9]

```
>> C= [1 2 3 ; 4 5 6; 7 8 9]
C =
     1     2     3
     4     5     6
     7     8     9
```

متغیرها

D=[1 ; 2 ; 3]

```
>> D= [1 ;2 ;3]
D =
     1
     2
     3
```

D <3x1 double>

	1	2
1	1	
2	2	
3	3	
4		

E=[1 2 3]'

```
>> E=[1 2 3]'
E =
     1
     2
     3
```

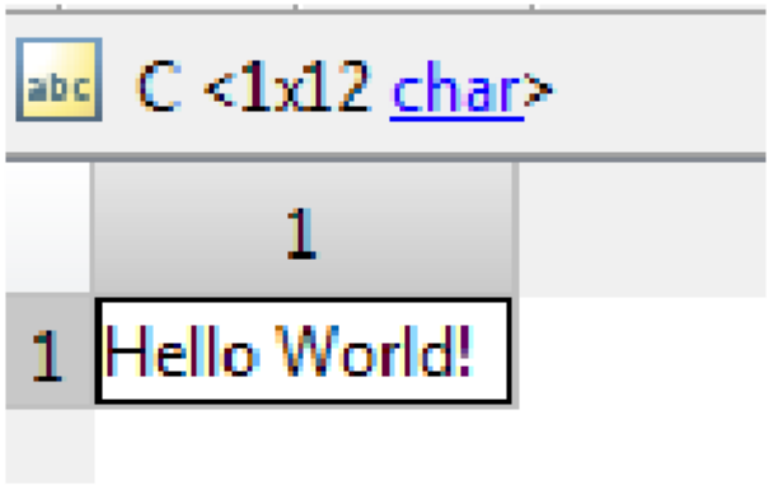
متغیرها

```
>> A=1:10
A =
     1     2     3     4     5     6     7     8     9    10


>> B= 0:2:10
B =
     0     2     4     6     8    10


>> 1:0.5:5
ans =
    1.0000    1.5000    2.0000    2.5000    3.0000    3.5000    4.0000    4.5000    5.0000
```

متغیرها

```
C =    'Hello World!';
```

```
>> C='Hello World!'
C =
Hello World!
```



متغیرها

```
A = zeros(3);
B = ones(5);
C = rand(100,2);
D = eye(20);
E = sprintf('%d\n',9);
```

```
>> A = zeros(5)

A =

     0     0     0     0     0
     0     0     0     0     0
     0     0     0     0     0
     0     0     0     0     0
     0     0     0     0     0
```

```
>> A=eye(5)

A =

     1     0     0     0     0
     0     1     0     0     0
     0     0     1     0     0
     0     0     0     1     0
     0     0     0     0     1
```

```
>> sprintf('%d',A)

ans =

    '1000001000001000001000001'
```

- ماتریس از یک شروع می شود (نه از صفر)
- اندیس‌های ماتریس مثبت هستند.

```
A =

     1     0     0     0     0
     0     1     0     0     0
     0     0     1     0     0
     0     0     0     1     0
     0     0     0     0     1

>> sprintf('%d',A)

ans =

    '1000001000001000001000001'

>> A(1,1)

ans =

     1

>> A(1,2)

ans =

     0

>> A(1,6)
Index in position 2 exceeds array bounds (must not exceed 5).
```

```
A =
     1     2     3
     4     5     6
     7     8     9
```

```
>> A(2,2:3)

ans =

     5     6
```

```
>> A(2,1:end)

ans =

     4     5     6
```

```
>> A(2,:)

ans =

     4     5     6
```

```
>> A(2,1:2:3)

ans =

     4     6
```

```
>> A(2,[1 3])

ans =

     4     6
```

```
>> A(:)

ans =

     1
     4
     7
     2
     5
     8
     3
     6
     9
```

```
>> A=rand(4)

A =

    0.5822    0.3181    0.4795    0.5439
    0.5407    0.1192    0.6393    0.7210
    0.8699    0.9398    0.5447    0.5225
    0.2648    0.6456    0.6473    0.9937
```

```
>> A(2,1)

ans =

    0.5407
```

```
>> A(:,3)

ans =

    0.4795
    0.6393
    0.5447
    0.6473
```

```
>> A(end,:)

ans =

    0.2648    0.6456    0.6473    0.9937
```

```
>> A(1:2, 1:end)

ans =

    0.5822    0.3181    0.4795    0.5439
    0.5407    0.1192    0.6393    0.7210
```

عملیات حسابی روی ماتریس‌ها

Given A and B:

```
>> A = [1 2 3;4 5 6;7 8 9]

A =

     1     2     3
     4     5     6
     7     8     9
```

```
>> B = [3 5 2; 5 2 8; 3 6 9]

B =

     3     5     2
     5     2     8
     3     6     9
```

Addition

```
>> X = A + B

X =

     4     7     5
     9     7    14
    10    14    18
```

Subtraction

```
>> Y = A - B

Y =

    -2    -3     1
    -1     3    -2
     4     2     0
```

Product

```
>> Z = A * B

Z =

    22    27    45
    55    66   102
    88   105   159
```

Transpose

```
>> T = A'

T =

     1     4     7
     2     5     8
     3     6     9
```

عملیات حسابی روی ماتریس‌ها

```
A = [1 2 3; 5 1 4; 3 2 1]
    A =
          1    2    3
          5    1    4
          3    2   -1
```

.+ .- .* ./ .^

```
x = A(1,:)

x=
     1   2   3
```

```
y = A(3 ,:)

y=
     3   4  -1
```

```
b = x .* y

b=
     3   8  -3
```

```
c = x . / y

c=
  0.33   0.5   -3
```

```
d = x .^y

d=
     1    16   0.33
```

عملیات حسابی روی ماتریس‌ها

```
>> A= rand(1,5)

A =

    0.4484    0.3658    0.7635    0.6279    0.7720

>> B= sum(A)

B =

    2.9776

 >> C= prod(A(1:3))

 C =

      0.1252
```

الحاق ماتریس‌ها (Matrixes Concatenation)

X=[1 2], Y=[3 4]

```
>> A=[X Y]
A =
     1     2     3     4
```

```
>> A=[X ; Y]
A =
     1     2
     3     4
```

تغییر شکل ماتریس

```
>> A = zeros(4,5)

A =

     0     0     0     0     0
     0     0     0     0     0
     0     0     0     0     0
     0     0     0     0     0

>> reshape(A,1,20)

ans =

  Columns 1 through 19

     0     0     0     0     0     0     0     0     0     0     0     0     0     0     0     0     0     0     0

  Column 20

     0
```

عملیات ماتریس

```
>> A = eye(5)

A =

     1     0     0     0     0
     0     1     0     0     0
     0     0     1     0     0
     0     0     0     1     0
     0     0     0     0     1
```

```
>> diag(A)

ans =

     1
     1
     1
     1
     1
```

```
>> diag([1 3 5])

ans =

     1     0     0
     0     3     0
     0     0     5
```

دوره آموزش متلب مقدماتی

تولید ماتریس تصادفی /تکرار ماتریس

```
>> A = randi([1 10],5,5)

A =

    10     5     8    10     7
    10     9    10     7     2
     2     2     7     8     8
    10     5     1     8     1
    10    10     9     4     3
```

```
>> repmat(A,1,2)

ans =

    10     5     8    10     7    10     5     8    10     7
    10     9    10     7     2    10     9    10     7     2
     2     2     7     8     8     2     2     7     8     8
    10     5     1     8     1    10     5     1     8     1
    10    10     9     4     3    10    10     9     4     3
```

```
>> det(A)

ans =

   7.1580e+03
```

نمایش اعضای غیر ماتریس

```
>> find(A)'

ans =

  Columns 1 through 19

     1     2     3     4     5     6     7     8     9    10    11    12    13    14    15    16    17    18    19

  Columns 20 through 25

    20    21    22    23    24    25
```

```
>> A = randi([1 5], 1,10)

A =

     1     1     5     4     2     5     1     3     2     4

>> sort(A)

ans =

     1     1     1     2     2     3     4     4     5     5
```

تبدیل ماتریس حقیقی به منطقی

```
>> logical(A)

ans =

  1×10 logical array

   1     1     1     1     1     1     1     1     1     1
```

```
>> mod(A,2)

ans =

     1     1     1     0     0     1     1     1     0     0
```

رشته‌ها

```
A =   ‘vision and geometry’
strfind(A,  ‘geometry')
strcmp(A,'computer vision')
B = strcat(A,' 12345')
c = [A,' 12345']
D = sprintf('I am %02d years old.\n',9)
int2str, str2num, str2double
```

سلول‌ها و ساختارها

- Cells
  - a = {}; a = cell(1)
  - b = {1,2,3}
  - c = {{1,2},2,{3}}
  - D = {'cat','dog','sheep','cow'}
  - E = {'cat',4}

```
>> D{2}
ans =
dog
```

```
>> E{1}
ans =
cat
>> E{2}
ans =
     4
```

- Structures
  - A = struct('name','1.jpg','height',640,'width',480);
  - b.name = '1.jpg‘

عملگرهای مقایسه‌ای

== Equal to

~= Not equal to

< Strictly smaller

> Strictly greater

<= Smaller than or equal to

>= Greater than equal to

& And operator

| Or operator

ساختارهای کنترلی

- if, for, while ....

```
if (a<3)
    Some Matlab Commands;
elseif (b~=5)
    Some Matlab Commands;
end
```

```
while  ((a>3) & (b==5))
    Some Matlab Commands;
end
```

```
for ii=1:100
    Some Matlab Commands;
end
```

```
for j=1:3:200
    Some Matlab Commands;
end
```

```
for k=[0.1 0.3 -13 12 7 -9.3]
    Some Matlab Commands;
end
```

عملیات برداری

Optimize your code for Matrix operations

Examples

In other languages:

```
tic; i = 0;
for t = 0:.001:1000
    i = i + 1;
    y(i) = sin(t);
end; toc;
```

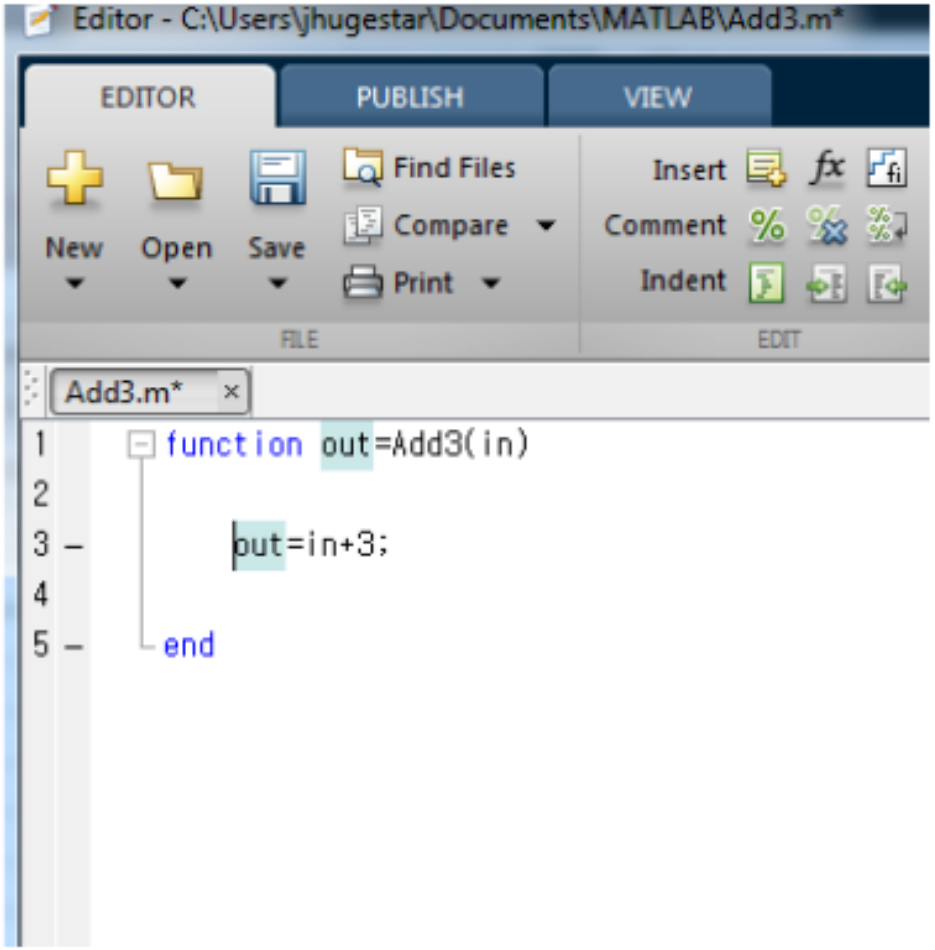
Elapsed time is 0.509381 seconds.

In MATLAB:

```
tic; t = 0:.001:1000;
y = sin(t); toc;
```

Elapsed time is 0.011212 seconds.

توابع




```
>> a =magic(3)
a =
     8     1     6
     3     5     7
     4     9     2
>> b =Add3(a)
b =
    11     4     9
     6     8    10
     7    12     5
```

توابع ترسیم

Plotting functions

plot, plot3d, bar, area, hist, contour, mesh

```
x = -pi:.1:pi;
y = sin(x);
plot(x,y)
```