>> a = 5

a =

5

>> b=2

b =

2

>> c=1.5

c =

1.5000

>> c=a+b

c =

7

>> c=b-a

c =

-3

>> c=cos(a)

c =

0.2837

>> cos(b)

ans =

-0.4161

>> d=a\*b

d =

10

>> d=a\*\*B

d=a\*\*B

↑

Invalid use of operator.

>> d=a\*\*b

d=a\*\*b

↑

Invalid use of operator.

>> d=a^b

d =

25

>> a=[2 4 6 8]

a =

2 4 6 8

>> type(a)

Error using type

Arguments must be character vectors or strings

>> b=[1 3; 2 4]

b =

1 3

2 4

>> z=zers(3,1)

Unrecognized function or variable 'zers'.

Did you mean:

>> z=zeros(3,1)

z =

0

0

0

>> a

a =

2 4 6 8

>> b

b =

1 3

2 4

>> b+3

ans =

4 6

5 7

>> tan(b)

ans =

1.5574 -0.1425

-2.1850 1.1578

>> a'

ans =

2

4

6

8

>> b'

ans =

1 2

3 4

>> b

b =

1 3

2 4

>> p=b\*b'

p =

10 14

14 20

>> p=b\*inv(b)

p =

1 0

0 1

>> format long

>> p=b\*inv(b)

p =

1 0

0 1

>> p=b\*2

p =

2 6

4 8

>> p=b.\*b

p =

1 9

4 16

>> b

b =

1 3

2 4

>> p=b\*b

p =

7 15

10 22

>> b.^4

ans =

1 81

16 256

>> B=[b,b]

B =

1 3 1 3

2 4 2 4

>> B=[b;b]

B =

1 3

2 4

1 3

2 4

>> sqrt(-1)

ans =

0.000000000000000 + 1.000000000000000i

>> format short

>> sqrt(-1)

ans =

0.0000 + 1.0000i

>> C=[1+i, 2-2i;3+3j,4-4j]

C =

1.0000 + 1.0000i 2.0000 - 2.0000i

3.0000 + 3.0000i 4.0000 - 4.0000i

>> A=[1 3 5 7 9, 0 2 4 6 8,11 13 15 17 19, 10 12 14 16 18]

A =

Columns 1 through 7

1 3 5 7 9 0 2

Columns 8 through 14

4 6 8 11 13 15 17

Columns 15 through 20

19 10 12 14 16 18

>> A=[1 3 5 7 9; 0 2 4 6 8;11 13 15 17 19; 10 12 14 16 18]

A =

1 3 5 7 9

0 2 4 6 8

11 13 15 17 19

10 12 14 16 18

>> A(3,4)

ans =

17

>> A(4,3)-14

ans =

0

>> A(1)

ans =

1

>> A(6)

ans =

2

>> A(5)

ans =

3

>> A(3)

ans =

11

>> t=A(6,6)

Index in position 1 exceeds array bounds (must

not exceed 4).

>> A(6,6)=30

A =

1 3 5 7 9 0

0 2 4 6 8 0

11 13 15 17 19 0

10 12 14 16 18 0

0 0 0 0 0 0

0 0 0 0 0 30

>> A(6,5)=30

A =

1 3 5 7 9 0

0 2 4 6 8 0

11 13 15 17 19 0

10 12 14 16 18 0

0 0 0 0 0 0

0 0 0 0 30 30

>> A(5,6)=30

A =

1 3 5 7 9 0

0 2 4 6 8 0

11 13 15 17 19 0

10 12 14 16 18 0

0 0 0 0 0 30

0 0 0 0 30 30

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

B =

1 2 3

4 5 6

7 8 9

>> B(4,4)=10

B =

1 2 3 0

4 5 6 0

7 8 9 0

0 0 0 10

>> A(6,6)=A(6,5)=0

A(6,6)=A(6,5)=0

↑

Incorrect use of '=' operator. Assign a value

to a variable using '=' and compare values for

equality using '=='.

>> A(6,6)=

A(6,6)=

↑

Error: Invalid expression. Check for missing or

extra characters.

>> A(6,6)=0

A =

1 3 5 7 9 0

0 2 4 6 8 0

11 13 15 17 19 0

10 12 14 16 18 0

0 0 0 0 0 30

0 0 0 0 30 0

>> A(6,5)=0

A =

1 3 5 7 9 0

0 2 4 6 8 0

11 13 15 17 19 0

10 12 14 16 18 0

0 0 0 0 0 30

0 0 0 0 0 0

>> A

A =

1 3 5 7 9 0

0 2 4 6 8 0

11 13 15 17 19 0

10 12 14 16 18 0

0 0 0 0 0 30

0 0 0 0 0 0

>> B(1:3,3)

ans =

3

6

9

>> B(2,:)

ans =

4 5 6 0

>> C=0:3:30

C =

Columns 1 through 7

0 3 6 9 12 15 18

Columns 8 through 11

21 24 27 30

>> C=0:3:30

C =

Columns 1 through 10

0 3 6 9 12 15 18 21 24 27

Column 11

30

>> A=magic(3)

A =

8 1 6

3 5 7

4 9 2

>> B=rand(1,2,3)

B(:,:,1) =

0.6753 0.0067

B(:,:,2) =

0.6022 0.3868

B(:,:,3) =

0.9160 0.0012

>> B

B(:,:,1) =

0.6753 0.0067

B(:,:,2) =

0.6022 0.3868

B(:,:,3) =

0.9160 0.0012

>> save myfile.mat

>> load myfile.mat

>> A

A =

8 1 6

3 5 7

4 9 2

>> t="Hello world!"

t =

"Hello world!"

>> t=""Hello "world!"""

t=""Hello "world!"""

↑

Invalid expression. Check for missing multiplication operator,

missing or unbalanced delimiters, or other syntax error. To

construct matrices, use brackets instead of parentheses.

>> t="Something and ""another"" something"

t =

"Something and "another" something"

>> whos t

Name Size Bytes Class Attributes

t 1x1 214 string

>> C=27

C =

27

>> K=C+273

K =

300

>> temptext= "Temperature is "+K+"K

temptext= "Temperature is "+K+"K

↑

Error: String is not terminated properly.

>> temptext= "Temperature is " +K+ "K"

temptext =

"Temperature is 300K"

>> temptext= "Temperature is "+K+"K"

temptext =

"Temperature is 300K"

>> A=["a","bbb","ccccc"]

A =

1×3 string array

"a" "bbb" "ccccc"

>> strlength(A)

ans =

1 3 5

>> seq='ABCDEFGHIJK':

seq='ABCDEFGHIJK':

↑

Error: Invalid expression. Check for missing or extra

characters.

>> seq='ABCDEFGHIJK';

>> whos seq

Name Size Bytes Class Attributes

seq 1x11 22 char

>> seq(5)

ans =

'E'

>> seq2=[seq 'LMNOPQRSTUVWXYZ']

seq2 =

'ABCDEFGHIJKLMNOPQRSTUVWXYZ'>> A=[2 4 6 8]

A =

2 4 6 8

>> max(A)

ans =

8

>> B=[1 3 5 7];

>> max(A,B)

ans =

2 4 6 8

>> B=[7 5 3 1]

B =

7 5 3 1

>> max(A,B)

ans =

7 5 6 8

>> maxA=Max(A)

Unrecognized function or variable 'Max'.

Did you mean:

>> maxA=max(A)

maxA =

8

>> [maxA,location]=max(A)

maxA =

8

location =

4

>> disp('Hello world!')

Hello world!

>>clc

>> x=0:pi/50:2\*pi;

>> y=cos(x);

>> plot(x,y)

>> xlabel('x')

>> ylabel('cos(x)')

>> title('PLot of the Cosine Function')

>> plot(x,y,r-)

plot(x,y,r-)

↑

Invalid expression. When calling a function or indexing a

variable, use parentheses. Otherwise, check for mismatched

delimiters.

>> plot(x,y,r--)

plot(x,y,r--)

↑

Invalid expression. When calling a function or indexing a

variable, use parentheses. Otherwise, check for mismatched

delimiters.

>> plot(x,y,'r--')

>> plot(x,y,'r--')

>> title('Plot of the Cosine Function')

>> xlabel('x')

>> ylabel('cos(x)')

Chart, line chart

Description automatically generated

>> x=0:pi/50:2\*pi;

>> y=sin(x);

>> plot (x,y)

>> hold on

>> y2=cos(x);

>> plot(x,y2,':')

>> legend('sin','cos')

>> hold off

Chart, line chart

Description automatically generated

>> [X,Y]=meshgrid(-2:.2:2);

>> Z=X.\*exp(-X.^2-Y.^2);

>> surf(X,Y,Z)

Chart, surface chart

Description automatically generated

>> t=0:pi/5:2\*pi;

>> [X,Y,Z]=cylinder(3\*sin(t));

>> subplot(2,2,1); mesh(X); title('X');

>> subplot(2,2,2); mesh(Y); title('Y');

>> subplot(2,2,3); mesh(Z); title('Z');

>> subplot(2,2,4); mesh(X,Y,Z); title('X,Y,Z');

Chart, radar chart

Description automatically generated

>> edit mysphere

[x,y,z]=sphere;

r=2;

surf(x\*r,y\*r,z\*r)

axis equal

A=4\*pi\*r^2;

V=(4/3)\*pi\*r^3;

>> mysphere

Chart

Description automatically generated

>> edit newfile.mlx

N=100;

f(1)=1;

f(2)=1;

for n=3:N

f(n)=f(n-1)+f(n-2);

end

f(1:12)

>> newfile

ans =

Columns 1 through 10

1 1 2 3 5 8 13 21 34 55

Columns 11 through 12

89 144

num=randi(100)

if num<34

sz='low'

elseif num<67

sz='medium'

else

sz='high'

end

>> newfile

num =

47

sz =

'medium'

>> newfile

num =

43

sz =

'medium'

>> newfile

num =

47

sz =

'medium'

>> newfile

num =

78

sz =

'high'

>> doc mean

>> mean(

mean(

↑

Invalid expression. When calling a function or indexing a

variable, use parentheses. Otherwise, check for mismatched

delimiters.

Did you mean:

>> help mean

mean Average or mean value.

S = mean(X) is the mean value of the elements in X if X is a vector.

For matrices, S is a row vector containing the mean value of each

column.

For N-D arrays, S is the mean value of the elements along the first

array dimension whose size does not equal 1.

mean(X,'all') is the mean of all elements in X.

mean(X,DIM) takes the mean along the dimension DIM of X.

mean(X,VECDIM) operates on the dimensions specified in the vector

VECDIM. For example, mean(X,[1 2]) operates on the elements contained

in the first and second dimensions of X.

S = mean(...,OUTTYPE) specifies the type in which the mean is performed,

and the type of S. Available options are:

'double' - S has class double for any input X

'native' - S has the same class as X

'default' - If X is floating point, that is double or single,

S has the same class as X. If X is not floating point,

S has class double.

S = mean(...,NANFLAG) specifies how NaN (Not-A-Number) values are

treated. The default is 'includenan':

'includenan' - the mean of a vector containing NaN values is also NaN.

'omitnan' - the mean of a vector containing NaN values is the mean

of all its non-NaN elements. If all elements are NaN,

the result is NaN.

Example:

X = [1 2 3; 3 3 6; 4 6 8; 4 7 7]

mean(X,1)

mean(X,2)

Class support for input X:

float: double, single

integer: uint8, int8, uint16, int16, uint32,

int32, uint64, int64

See also median, std, min, max, var, cov, mode.

Documentation for mean

Other functions named mean