Some MATLAB Built-in Functions

Function	Description
`sqrt(x)`	Square root of `x`
`nthroot(x,n)`	`n`th root of `x`
`abs(x)`	Absolute value of `x`
`exp(x)`	Exponential (e^x)
`log(x)`	Natural logarithm (Base e logarithm) of `x`
`log10(x)`	Base 10 logarithm of `x`
`factorial(x)`	The factorial of `x`
`rem(x,y)`	The remainder after `x` is divided by `y`
`max(A)`	If A is a vector, returns the largest value in A. If A is a matrix, returns a vector in which each element is the largest number in the corresponding column of A.
`min(A)`	If A is a vector, returns the smallest value in A. If A is a matrix, returns a vector in which each element is the smallest number in the corresponding column of A.
`sum(A)`	If A is a vector, returns the sum of the elements in A. If A is a matrix, returns a vector in which each element is the sum of the values in the corresponding column of A.
`mean(A)`	If A is a vector, returns the mean value of the elements in A. If A is a matrix, returns a vector in which each element is the average of the values in the corresponding column of A.
`median(A)`	If A is a vector, returns the median value of elements in A**.** If A is a matrix, returns a vector in which each element is the median value of the corresponding column of A.
`corrcoef(x,y)`	Returns a 2×2 matrix where the values at positions (1,2) and (2,1) are the Pearson's correlation coefficient between two vectors x and y.
`Corrcoef(X)`	where X is a matrix containing k columns. It returns a k×k matrix where values at positions (i, j) and (j, i) are the Pearson's correlation coefficient between ith and jth columns of X.
`sort(A)`	If A is a vector, output the elements in A in value ascending order. If A is a matrix, sort each column of A in value ascending order.
`sort(A,'descend')`	If A is a vector, output the elements in A in value descending order. If A is a matrix, sort each column of A in value descending order.
`sortrows(A,col)`	Sort the whole rows in matrix A according to the values in the column indicated by col in value-ascending order (or in value-descending order if `–col` is used).
`length(A)`	If A is a vector, returns the number of elements in A. If A is a matrix, returns the larger of its number of rows and columns.
`size(A)`	Returns a row vector [m,n], where m is the number of rows in A and n is the number of columns in A.
`sin(x)`	Sine of angle x (x in radians)

sind(x)	Sine of angle x (x in degrees)
cos(x)	Cosine of an angle x (x in radians)
cosd(x)	Cosine of an angle x (x in degrees)
tan(x)	Tangent of angle x (x in radians)
tand(x)	Tangent of angle x (x in degrees)
round(x)	Round x to the nearest integer
ceil(x)	Round x towards infinity. (Round x to the nearest integer greater than or equal to x)
floor(x)	Round x towards negative infinity. (Round x to the nearest integer less than or equal to x)
zeros(m,n)	Create a m×n matrix of zeros
ones(m,n)	Create a m×n matrix of ones
eye(m)	Create a m×m square matrix in which the main diagonal elements are equal to 1 and the rest are 0.
rand	Generates a single uniformly distributed random number between 0 and 1
rand(1,n)	Generates a row vector of n uniformly distributed random numbers between 0 and 1.
rand(m,n)	Generates an m×n matrix of uniformly distributed random numbers between 0 and 1
rand(n)	Generates a n×n matrix of random numbers between 0 and 1
randn(1,n)	Generates a row vector of n normally distributed random numbers with mean 0 and standard deviation of 1.
randn(m,n)	Generates a m×n matrix of normally distributed random numbers with mean 0 and standard deviation of 1
randn(n)	Generates a n×n matrix of normally distributed random numbers with mean 0 and standard deviation of 1
randperm(n)	Generates a row vector with n elements that are random permutation of integers 1 through n
plot(x,y,‘line specifiers’)	Plot y against x. x is a vector of horizontal coordinates of the data points y is a vector of vertical coordinates of the data points. Line specifiers are listed in the two tables on the next page.
fplot(‘function’, limits,‘line specifiers’)	Plot the curve of the function specified by ‘function’. function is the function to be plotted. limits is the range of x. line specifiers specify the type and color of the line and markers, listed in the two tables on the next page.
polyfit(x,y,n)	Fit a polynomial function into data points. x is a vector of horizontal coordinates of the data points

	`y` is a vector of vertical coordinates of the data points. `n` is the order (or degree) of polynomial The function returns a vector of the coefficients of the polynomial that fits the data. Its size is n+1
`polyval(p,x)`	Calculates the value of a polynomial function at one or more points given the coefficients of the polynomial. `p` is a vector containing the coefficients of the polynomial. `x` is a scalar, a vector or a matrix of values The function returns the value(s) of the polynomial function with coefficients in `p` for the corresponding values in `x`.

Line color specifiers and marker type specifiers used in the plot command

Line Color	**Specifier**	**Marker Type**	**Specifier**
Red	**r**	plus sign	+
Green	**g**	circle	**o**
Blue	**b**	asterisk	*
Cyan	**c**	point	**.**
Magenta	**m**	cross	**x**
Yellow	**y**	square	**s**
Black	**k**	diamond	**d**
White	**w**	five-pointed star	**p**

Line style specifiers

Line Style	**Specifier**
solid (default)	**-**
dashed	**--**
dotted	**:**
dash-dot	**-.**