if expression
    statements
else if expression
    statements
    ...
else
    statements
end

switch (value)
    case {v1, v2, ..., vn}
        statements
        ...
    case v3
        statements
    otherwise
        statements
end

for var = array
    statements
end

while expression
    statements
end
break terminates the execution of FOR and WHILE loops.
In nested loops, break exits from the innermost loop only.

break is not defined outside of a FOR or WHILE loop.
Use RETURN in this context instead.

continue passes control to the next iteration of FOR or WHILE loop in which it appears, skipping any remaining statements in the body of the FOR or WHILE loop.

In nested loops, continue passes control to the next iteration of FOR or WHILE loop enclosing it.

return causes a return to the invoking function or to the keyboard. It also terminates the KEYBOARD mode.

Normally functions return when the end of the function is reached. A return statement can be used to force an early return.
Matlab : פונקציות שימועיות ב-
(ב.createUser Thương)

**all**

For vectors, all(V) returns logical 1 (TRUE) if none of the elements of the vector are zero. Otherwise it returns logical 0 (FALSE). For matrices, ALL(X) operates on the columns of X, returning a row vector of logical 1's and 0's.

**any**

If A is a vector, any(A) returns logical 1 (true) if any of the elements of A is a nonzero number or is logical 1 (true), and returns logical 0 (false) if all the elements are zero. If A is a matrix, any(A) treats the columns of A as vectors, returning a row vector of logical 1's and 0's.

**ceil**

B = ceil(A) rounds the elements of A to the nearest integers greater than or equal to A.

**char**

S = char(A) converts the array A that contains nonnegative integers representing character codes into a MATLAB character array (the first codes are ASCII). The actual characters displayed depends on the 127 character encoding scheme for a given font. The result for any elements of A outside the range from 0 to 65535 is not defined (and may vary from platform to platform). Use DOUBLE to convert a character array into its numeric codes.

**find**

ind = find(X) locates all nonzero elements of array X, and returns the linear indices of those elements in vector ind. If X is a row vector, then ind is a row vector; otherwise, ind is a column vector. If X contains no nonzero elements or is an empty array, then ind is an empty array.

[row,col] = find(X) returns the row and column indices of the nonzero entries in the matrix X.

**floor**

B = floor(A) rounds the elements of A to the nearest integers less than or equal to A.
isempty

isempty(X) returns 1 if X is an empty array and 0 otherwise. An empty array has no elements.

length

n = length(X) returns the size of the longest dimension of X. If X is a vector, this is the same as its length.

max

If A is a vector, max(A) returns the largest element in A.
If A is a matrix, max(A) treats the columns of A as vectors, returning a row vector containing the maximum element from each column.

min

If A is a vector, min(A) returns the smallest element in A.
If A is a matrix, min(A) treats the columns of A as vectors, returning a row vector containing the minimum element from each column.

mod

M = mod(X,Y) if Y ~= 0, returns X - n.*Y where n = floor(X./Y). If Y is not an integer and the quotient X./Y is within roundoff error of an integer, then n is that integer.
The inputs X and Y must be real arrays of the same size, or real scalars.
Example: X = [23 24 25 26]
M = mod(X, 3)
Then M gets the array [2 0 1 2]

ones

Y = ones(n) returns an n-by-n matrix of 1s. An error message appears if n is not a scalar.
Y = ones(m,n) or Y = ones([m n]) returns an m-by-n matrix of ones.

repmat

B = repmat(A,m,n) creates a large matrix B consisting of an m-by-n tiling of copies of A. The size of B is [size(A,1)*m, (size(A,2)*n].
The statement repmat(A,n) creates an n-by-n tiling.

reshape

B = reshape(X,M,N) returns the M-by-N matrix whose elements are taken columnwise from X. An error results if X does not have M*N elements.
size

d = size(X) returns the sizes of each dimension of matrix X in a vector d with 2 elements. If X is a scalar, which MATLAB regards as a 1-by-1 array, size(X) returns the vector [1 1].

[m,n] = size(X) returns the size of matrix X in separate variables m and n.
m = size(X,dim) returns the size of the dimension of X specified by scalar dim.
size(X,1) returns the number of rows in X; size(X,2) returns the number of columns in X.

sum

If A is a vector, sum(A) returns the sum of the elements.
If A is a matrix, sum(A) treats the columns of A as vectors, returning a row vector of the sums of each column.

zeros

B = zeros(n) returns an n-by-n matrix of zeros. An error message appears if n is not a scalar.
B = zeros(m,n) or B = zeros([m n]) returns an m-by-n matrix of zeros.