```
% Bisection Method in MATLAB

% Define the function for which you want to find the root
f = @(x) x^3 - x - 2; % Example: f(x) = x^3 - x - 2

% Define the interval [a, b]
a = 1; % Left endpoint
b = 2; % Right endpoint

% Tolerance and maximum iterations
tol = 1e-6; % Desired accuracy
max_iter = 100; % Maximum number of iterations

% Check if the function has opposite signs at the endpoints
if f(a) * f(b) >= 0
    error('The function must have opposite signs at a and b.');
end

% Initialize variables
iter = 0; % Iteration counter

fprintf('Iter\t   a\t\t   b\t\t   c\t\t   f(c)\n');
fprintf('-------------------------------------------------------------\n');

while (b - a) / 2 > tol && iter < max_iter
    % Increment iteration counter
    iter = iter + 1;

    % Compute the midpoint
    c = (a + b) / 2;

    % Display current step
    fprintf('%d\t%.6f\t%.6f\t%.6f\t%.6f\n', iter, a, b, c, f(c));

    % Check if the root is found or narrow the interval
    if f(c) == 0
        break; % c is the root
    elseif f(c) * f(a) < 0
        b = c; % Root is in the left subinterval
    else
        a = c; % Root is in the right subinterval
    end
end

% Output the result
root = (a + b) / 2;
fprintf('-------------------------------------------------------------\n');
fprintf('The root is approximately: %.6f\n', root);
fprintf('Number of iterations: %d\n', iter);
```