```
function [elk] = MD_estiff_bothnode_MyMz_release (A, E, L)

    % Code developed by Mrunmayi Mungekar and Devasmit Dutta
%
% MD_estiff.m computes the element stiffness matrix for a given element with both
nodes flexurally released
%

%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%
%  Functions Called
%              none
%
%  Dictionary of Variables
%  Input information
                % A = cross-sectional area
                % Izz = moment of inertia about local z-axis
                % Iyy = moment of inertia about local y-axis
                % J = torsional constant
                % Ayy = shear area along local y-axis
                % Azz = shear area along local z-axis
                % E = Young's modulus
                % v = Poisson's ratio
                % L = element length

                % G = shear modulus
                % elk_temp = temporary element stiffness matrix (just the lower
triangular part)
                % kA = axial stiffness
                % kJ = torsional stiffness
                % etaz = shear coefficient along local z-axis
                % etay = shear coefficient along local y-axis
%
% Output information
                % elk = element stiffness matrix
%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%
kA = E * A / L;
elk = zeros(12,12);
elk(1,1) = kA;
elk(7,7) = kA;
elk(7,1) = -kA;
elk(1,7) = -kA;
```