always begins with the word function

file name matches the name of the function

```
function y = linspace(d1, d2, n)
%LINSPACE Linearly spaced vector.
%   LINSPACE(X1, X2) generates a row vector of 100 linearly
%   equally spaced points between X1 and X2.
%
%   LINSPACE(X1, X2, N) generates N points between X1 and X2.
%   For N = 1, LINSPACE returns X2.
%
%   Class support for inputs X1,X2:
%      float: double, single
%
%   See also LOGSPACE, COLON.

%   Copyright 1984-2018 The MathWorks, Inc.

if nargin == 2
   n = 100;
else
   n = floor(double(n));
end
if ~isscalar(d1) || ~isscalar(d2) || ~isscalar(n)
   error(message('MATLAB:linspace:scalarInputs'));
end
n1 = n-1;
if d1 == -d2 && n > 2 && isfloat(d1) && isfloat(d2)
   % For non-float inputs, fall back on standard case.
   if isa(d1, 'single')
      % Mixed single and double case always returns single.
      d2 = -d1;
   end
   y = d2*((-n1:2:n1)./n1);
   if rem(n1, 2) == 0 % odd case
      y(n1/2+1) = 0;
   end
else
   c = (d2 - d1).*(n1-1); %check intermediate value for appropriate treatment
   if isinf(c)
      if isinf(d2 - d1) %opposite signs overflow
         y = d1 + (d2./n1).*(0:n1) - (d1./n1).*(0:n1);
      else
         y = d1 + (0:n1).*((d2 - d1)./n1);
      end
   else
```

comments documenting how the function is called

allows variable number of input arguments - advanced technique; see Chapter 8

checking for valid inputs - see Chapter 9

These values (d1, d2, and n) were passed to the function through its input arguments.

The output argument (y) is being assigned values here; that's how the function knows what values to return.

```
        y = d1 + (0:n1).*(d2 - d1)./n1;
    end
    if ~isempty(y)
        if d1 == d2
            y(:) = d1;
        else
            y(1) = d1;
            y(end) = d2;
        end
    end
end
```

end of the function; no return statement needed in MATLAB