```
%  Matlab_intro:   A Matlab tutorial for Differential Equations        (Matlab for DiffEQ-1/2)
%  ------------------------------------------------------------
%           The lines that start with the percent sign are comments.
%           You don't have to type them.  Matlab ignores anything after a %.

% Windows
% When you start up Matlab you will see a window with many panes.  I will call each a window.
% You can move them around and resize them in very flexible ways.  First let's look at

%%%%%%%%%%%%%%%%%%% The Command Window
% This is where you type the command you want Matlab to perform.  Try
2+2
x=[1 1]           % space or comma between numbers creates a row vector.
x=[1; 1]          % semi-colon creates a column vector
A=[1 2; 3 4]      % combine spaces and semi-colons to make matrices
B=eye(2)          % This is a cute way to spell the identity matrix
y=1:9             % colon creates a list of numbers (row vector)
z=0:.25:4         % You can step the list by non-integers
z(3:8)            % you can use lists to access part of a vector
z(8:-1:3)         % you can reverse the order by stepping by negative amounts.
size(z)           % Is there one more point than you thought there would be?  Why?
zz=linspace(0,4,16)     % linspace gives evenly spaced points including endpoints
                  % Are you surprised that the spacing is not 0.25?  Why isn't it?

% Formating of numbers
format long                % how many digits to print
[1 1.25 pi]
format short
[1 1.25 pi]
format loose              % how much space between lines
[1 1.25 pi]
format compact
[1 1.25 pi]
%Set these permanently for you on this machine using the menu:  MATLAB->Preferences->Command Window
%I suggest format compact and format short for now.  Sometimes you might need format long.

% Matrix operations
A+B
2*A
A+1    % add a scalar and a matrix
B*A    % Matrix multiplication
B.*A   % The .* operator does elementwise multiplication instead of matrix multiplication.

%%%%%%%%%%%%%%%%%%%   Plotting
% To plot a function you have to specify the domain of that function.  That gives the x-axis range.
x=linspace(0,2*pi,100);  % 100 evenly spaced points up to 2*pi
plot(x, sin(x))         % A new window opens to show the plot.  Click the arrow at the top to
                        % contain it within the main matlab window.  Click the new arrow to get it
                        % separated again.  Click the x to make it disappear.
y=sin(x.^2);        % Notice I have to use .^ for elementwise square.
plot(x,y,'ro')     % The format string can include color and shape.  See the help for plot for info.
title('Great Function')
xlabel('x')
ylabel('sin(x^2)')

%%%%%%%%%%%%%%%%%%%  The Current Folder Window
% This window lets you see and manipulate the files you are using.
% The command window has a "Current Folder" associated with it.  Any files in this Folder
% ending in .m are available as commands (scripts) in Matlab.

% Switch Folders to your Desktop.  You can use the mouse, or in the command window type
cd ~/Desktop      % the ~ symbol is a shortcut to /Users/yourname/
% Start editing a file   (you can use the mouse to open a "New Script File" or in the command
% window type:
edit myscript.m
```

```
%%%%%%%%%%%%%%%%%%%  Solve an ODE                                  (Matlab for DiffEQ-2/2)
% In the file myscript.m type the following commands
trange=[0 4];                           % Timerange:  [start end]
init=[0.1];                              % initial conditions (vars at the start time)
%%% The ODE is specified by a function that returns the right hand side of the eqns.
%%% The function (called RHS here) must take two inputs: time and a column of variables.
%%% The "vars" will be the current value of each dependent variable.  So in our case
%%% there is only one dependent variable so vars(1) is t hat variable.
%%%      We are solving y' = 5-2*y
RHS= @(t,vars) 5-2*vars(1);             % RHS of ODE
[t soln] = ode45(RHS, trange, init);      % Numerically solve the ODE
plot(t,soln)
title('Solution of the ODE dy/dt= 5-2y')
xlabel('Time')
ylabel('Solution')
% Save the file as myscript.m
% Now in the command window:
myscript        % you can see all the output.  And the variable "soln" appears in the workspace.

% Now change the ODE you are solving to y'=1/(1+y*y)   y(0)=-1

%%%%%%%%%%%%%%%%%%%  The Workspace Window
% This window pane shows the variables that have been defined in this Matlab Session.
% You can also examine and edit these variable with a spreadsheet interface.
% Try double clicking on y and change its last entry to 2.1.    Does the plot redaw?
y(end)        % this should show up as 2.1 now.    You can also change it with y(end)=2.1
% This window turns out to be not very useful except for entering/checking your data.
% You can choose to remove it from the visible panes using the "X" in the upper right.
% You can also separate any pane into its own window using the arrows at the top of the pane.
% Separate and then put back the command window.


%%% Direction Field
% Create another script file called dirfield.m    Put the following lines in:
pts=0:.25:4;     % The semi-colon keeps the line from printing
t=pts;
y=pts;            % Points to put the arrows
[tt,yy]=meshgrid(t,y);    % create t and y values for all points in the grid
dt=0.1*ones(size(yy));  % find coordinates of the arrows.  Only the ratio dy to dt matters.
dy= (2*yy-5).*dt;
disp('Right click (or Contrl-click) to stop plotting trajectories')
quiver(tt,yy,dt,dy)  % A quiver holds arrows..... Get it?  These Matlab guys are toooo funny!
hold on            % Hold the picture even if another is drawn
axis([0,4,0,4])        % don't allow changing the axes

[tpoint ypoint button]= ginput(1);     % Get the position of the mouse click and which button.
while button ~= 3            % ~= means NOT EQUAL...  here a right click (button 3) stops drawing.
    dydt = @(t,vars) 2*vars(1)-5;    % This defines a function of two variables called dydt.
                                % We will use it to solve the differential equation.
    [t soln] = ode45( dydt, [tpoint 4], [ypoint] );
    plot(t,soln)
    [tpoint ypoint button]= ginput(1);
end

% So we have seen plot() and quiver().  3D versions are plot3 and quiver3.  You can find more
% in the help (Help menu, then 'Product Help')

%%%%%% PPlane and DField
% Download dfield8.m and pplane8.m from our webpage.
% Try dfield8.m first.  Run it.  Change the equation.  Repeat.
% Now try pplane8.m   What is different?   Try x'=y;  y'=-sin(x)+0.1cos(y)
```