```
% ### EXcreate3D2.m ###      11.21.16

% code to demonstrate/fiddle w/ 3-D plots in Matlab and function
handles

% ---------
% Notes
% o once you make the plot, in the figure window, look up top for the
icon
% that is a cube w/ a circular arrow around it ("Rotate 3D"). Click on
that
% and then have some fun....

% ---------
% Different functions to try for f= @(X,Y):
% o (1./sqrt(Y)).*exp(-(X.^2)./Y)               [sol. to Diffusion Eq.
re point source]
% o Y.*cos(2*pi*X)                              [slanted sinusoid]
% o (1-Y).*tanh(8*X-3)                          [slanted asympotote]
% o exp((-(X-0.5).^2 + -(Y-0.5).^2)/0.05)       [3-D Gaussian]
% o others??

clear
% ===================
% function handle to express function (see options above)
f= @(X,Y) exp((-(X-0.5).^2 + -(Y-0.5).^2)/0.05);

% other plotting params.
N= 40;              % grid resolution
xB= [0 1];         % min/max values along x
yB= [0 1];         % min/max values along y
vP= [-62 36];       % view angle (vals. between 0 and 90)
usePeaks= 0;        % eschew function above and use Matlab's built-in
peaks.m) {0}
% ===================

% ---------
% create x and y axes
x= linspace(xB(1),xB(2),N);
y= linspace(yB(1),yB(2),N);
% from those, create a base "grid" for f
[X,Y]= meshgrid(x,y);  % multi-dimensionlize as required
% Note: this is a trick I learned from Hanselman & Littlefield ch.27.2

% ---------
% now determine F
if usePeaks==0
    F= f(X,Y);
else
    [X,Y,F] = peaks(N);     % use built-in Matlab function to create
```

```
"z"
end

% ---------
figure(1); clf;
surf(X,Y,F); view(vP);
colormap jet;    hCB= colorbar; ylabel(hCB,'f');
xlabel('x'); ylabel('y'); zlabel('f');
if (1==0);  shading interp;     end     % turn on to "smooth" coloring
{0}
if (1==0);  shading flat;      end     % turn on to eschew grid lines
{0}
```