```
close all; clear all;   % clear all plots and variables
%
% create the test signal
%
fs=4000;               % sample frequency
xfa=[80 0.5; 100 -1i];  % signal frequencies and phasor amplitudes
vfa=[210 1; 280 0.5];   % tonal noise frequencies and phasor amplitudes
snr=-8;                % white noise SNR in dB
nt=round(4*fs);        % 4 seconds worth of samples
[y,t,x,v]=mb1_testsig(xfa,vfa,snr,nt,fs); % y=test signal, t=time axis, x=clean signal, v=noise signal
snr0=mb1_snrtone(y,xfa,fs); % Find the SNR of the noisy signal
%
% plot the signal and its power spectrum
%
fplot=300;             % max frequency to plot
fax=linspace(0,fplot,100); % frequncy axis for magnitude responses
iplot=0.1*fs:0.2*fs;    % samples to plot
figure(1);
subplot(2,1,2);
mb1_plotpsd(y,fplot,fs);  % plot PSD of noisy signal
ylabel('Noisy PSD (dB)');
subplot(2,1,1);
plot(t(iplot),y(iplot),'-r',t(iplot),x(iplot),'-b');     % plot time waveforms
axisenlarge([-1 -1.05]);               % make axes fit the plot
title(sprintf('Clean and Noisy signals, SNR = %.1f dB, fs = %.2gkHz',snr0,fs/1000));
xlabel('Time (s)');
%
% now design a butterworth IIR filter
%
rp=0.1;                % target passband ripple (dB)
rs=35;                 % target stopband attenuation (dB)
ftr=[100 200];         % transition frequency range: 100 to 200 Hz
[n1,wn1]=buttord(2*ftr(1)/fs,2*ftr(2)/fs,rp,rs); % determine order and f0
[b1,a1]=butter(n1,wn1); % design a Butterworth filter
z1=filter(b1,a1,y);     % filter the noisy signal, y
[snr1,ax1,e1,v1]=mb1_snrtone(z1,xfa,fs);  % find the filtered SNR, gain errors and residual noise
figure(2);
subplot(2,1,2);
mb1_plotpsd(v1,fplot,fs);          % plot PSD of residual noise
texthvc(0.02,0.1,['Gains:' sprintf('\n%.0fHz: %+.1fdB \\angle%+.0f^\\circ',[xfa(:,1) e1(:,1) e1(:,2)*180/pi]')],'LBk');
ylabel('Noise PSD (dB)');
subplot(2,1,1);
plot(fax,20*log10(abs(freqz(b1,a1,fax*2*pi/fs)))); % plot the magnitude response
axis([fax(1) fax(end) -60 4]);      % limit the gain range to -60 dB
xlabel('Frequency (Hz)');
ylabel('Gain (dB)');
title(sprintf('Butterworth Filter Order %d, SNR = %.1f dB',n1,snr1));
%
tilefigs([0 0.5 0.8 0.5]);   % display all the figures in the top half of the screen
```