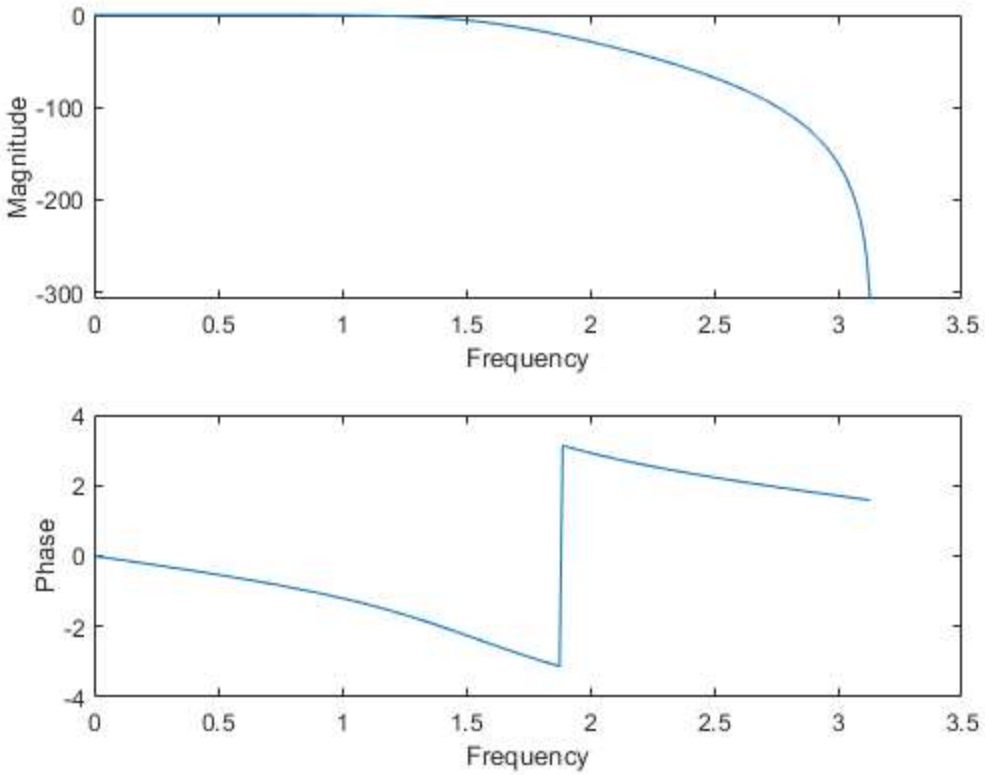
```
% Implementation on a low pass filter

clc;
clear all;
close all;

alphaP = 4;     % attenutaion in pass band
alphaS = 30;    % attenuation in stop band
fP = 400;       % pass band frequency
fS = 800;       % stop band frequency
FS = 2000;      % sampling frequency

% now declaring wP and wS

wP = ((2*3.14*(fP/FS))/3.14);   % need to balance the output for buttord
wS = ((2*3.14*(fS/FS))/3.14);   % function, by dividing the entire thing by pi

[n,Wc] = buttord(wP,wS,alphaP,alphaS); % used to get the order of the
                                       % buterworth filter
disp('The order of the filter is');
disp(n);
disp('The cutoff frequency is');
disp(Wc);

[b,a] = butter(n,Wc);
[H,W] = freqz(b,a,256);     % frequency response

Magnitude_resp = 20*log(abs(H)); % magnitude response
phase_resp = angle(H);              % phase response

subplot(2,1,1);
plot(W,Magnitude_resp);
xlabel('Frequency');
ylabel('Magnitude');

subplot(2,1,2);
plot(W,phase_resp);
xlabel('Frequency');
ylabel('Phase');
```

```
The order of the filter is
     3

The cutoff frequency is
    0.4914
```



Published with MATLAB® R2018a