```matlab
clear;                                   %clear variables ad fuctions from memory
clf;                                     %clear curret figures
y= 0:0.1:4*pi;
z = 0:0.1:4*pi;                          % direction of propogation
n= length(y);
filename = 'swr.gif';
figure(1)
Z0=50;                                   %characteristic impeadence is assigned a value of 50ohms
prompt='Enter the value of the load impedance'; %prompt is a variable of type 'char' which acce
ZL=input(prompt)                         % ZL store's the value of load impeadnce which was enter
if (ZL==0)                               %checks if the value entered by user is '0', if it is tr
    for  t= 1:n % it reapeat's statement 'n' times
%clf
V1 = sin(y(t)- z);                       %forward propogating wave equation
V2 = sin(y(t)+ z);                       %backward propogating wave equation
s1 = subplot(2,1,1);
h1 = plot(z, V1, 'LineWidth',3);         %forward travelling wave
grid on                                  %adds major grid lines to the current axes
grid minor                               %toggles the minor grid lines of the current axes
ylim([-2 2]);                            %limit for y-axis is -2 to 2
hold on                                  %holds the current plot & all axis properties
h2 = plot(z, V2, 'LineWidth',3);         %backward travelling wave
hold on                                  %holds the current plot & all axis properties
xlim([0 4*pi])                           %limit for x axis is 0 to 12.56(4*pi)

h3 =plot(z, V1+ V2, 'LineWidth',3); %resultant is a STANDING wave
hold on                                  %holds the current plot & all axis properties
grid on                                  %holds the current plot & all axis properties
grid minor                               %toggles the minor grid lines of the current axes
ylim([-2 2]);                            %limit for y-axis is -2 to 2
xlim([0 4*pi])                           %limit for x axis is 0 to 12.56(4*pi)
frame = getframe(1);                     %returns movie frame of standing wave
    im = frame2im(frame);                %returns image data associated with movie frame
    [imind,cm] = rgb2ind(im,256); %converts rgb image to indexed image
    if t == 1
        imwrite(imind,cm,filename,'gif', 'Loopcount',inf); %write image to graphic file
    else
        imwrite(imind,cm,filename,'gif','DelayTime',0.0001, 'WriteMode','append'); %write ima
    end
cla(s1)                                  %clear current axes
    end
end
if (ZL==50) %checks if the value entered by user is '50', if it is true the statements under it
    for  t= 1:n                          % it reapeat's statement 'n' times
%clf
V1 = sin(y(t)- z);                       %forward propogating wave equation
V2 = sin(y(t)+ z);                       %backward propogating wave equation
s1 = subplot(2,1,1);
h1 = plot(z, V1, 'LineWidth',3);         %forward travelling wave
grid on                                  %adds major grid lines to the current axes
grid minor                               %toggles the minor grid lines of the current axes
ylim([-2 2]);                            %limit for y-axis is -2 to 2
hold on                                  %holds the current plot & all axis properties
```

```matlab
xlim([0 4*pi])                          %limit for x axis is 0 to 12.56(4*pi)

grid on                                 %adds major grid lines to the current axes
grid minor                              %toggles the minor grid lines of the current axes
ylim([-2 2]);                           %limit for y-axis is -2 to 2
xlim([0 4*pi])                          %limit for x axis is 0 to 12.56(4*pi)

frame = getframe(1);                    %returns movie frame of TRAVELLING wave
     im = frame2im(frame);         %returns image data associated with movie frame
     [imind,cm] = rgb2ind(im,256); %converts rgb image to indexed image
     if t == 1
         imwrite(imind,cm,filename,'gif', 'Loopcount',inf); %write image to graphic file
     else
         imwrite(imind,cm,filename,'gif','DelayTime',0.0001, 'WriteMode','append'); %write ima
     end
cla(s1)                                 %clear current axes
     end
end
if (ZL==100)                            %checks if the value entered by user is '100', if it is t
     for  t= 1:n                        % it reapeat's statement 'n' times
%clf
V1 = sin(y(t)- z);                       %forward propogating wave equation
V2 = 0.707*sin(y(t)+ z);                 %backward propogating wave with decreased amplitude
s1 = subplot(2,1,1);
h1 = plot(z, V1, 'LineWidth',3);         %forward travelling wave
grid on                                 %adds major grid lines to the current axes
grid minor                              %toggles the minor grid lines of the current axes
ylim([-2 2]);                           %limit for y-axis is -2 to 2
hold on                                 %holds the current plot & all axis properties
h2 = plot(z, V2, 'LineWidth',3);         %backward travelling wave
hold on                                 %holds the current plot & all axis properties
xlim([0 4*pi])                          %limit for x axis is 0 to 12.56(4*pi)

h3 =plot(z, V1+ V2, 'LineWidth',3);      %resultant of V1 & V2
hold on                                 %holds the current plot & all axis properties

grid on                                 %adds major grid lines to the current axes
grid minor                              %toggles the minor grid lines of the current axes
ylim([-2 2]);                           %limit for y-axis is -2 to 2
xlim([0 4*pi])                          %limit for x axis is 0 to 12.56(4*pi)
frame = getframe(1);                    %returns movie frame of HYBRID wave
     im = frame2im(frame);              %returns image data associated with movie frame
     [imind,cm] = rgb2ind(im,256);      %converts rgb image to indexed image
     if t == 1
         imwrite(imind,cm,filename,'gif', 'Loopcount',inf);   %write image to graphic file
     else
         imwrite(imind,cm,filename,'gif','DelayTime',0.0001, 'WriteMode','append');     %write
     end
cla(s1)                                 %clear current axes
     end
end
```