```
%  _    _                      _                          _____ _______ _____
% | |  | |                    | |                        / ____|__   __|  __ \
% | |__| | ___  _ __ ___   ___| |__  _ __ _____      __ | |       | |  | |  | |
% |  __  |/ _ \| '_ ` _ \ / _ \ '_ \| '__/ _ \ \ /\ / / | |       | |  | |  | |
% | |  | | (_) | | | | | |  __/ |_) | | |  __/\ V  V /  | |____   | |  | |__| |
% |_|  |_|\___/|_| |_| |_|\___|_.__/|_|  \___| \_/\_/    \_____|  |_|  |_____/
% (ASCII font Big from https://www.coolgenerator.com/ascii-text-generator

% Homebrew CTD Data Grapher / Visualizer
% Author: Sophie Scopazzi
% Author Website: sophiescopazzi.com
% Date: OCT 2022

% always
clear all
close all

%% INSTRUCTIONS - HOW TO USE THIS CODE
% There are comments throughout code as well
% If you want to contact me about this code, use my website

% 1. Put this script in same folder as datafile.txt
    % Check name of datafile is the same as in the script
% 2. Ensure sensors are in the correct order in the data from the Arduino
    % script, or none of it will make sense (EC, TDS, sal, etc) in order
% 3. Fine correct time values in this plot, record for next step
    % Want data from downward travel, upward water column is disturbed
% 4. Recomment step 3, put the values recorded into DEPLOY DATA PULLING
% 5. Plots or subplots as desired. Set xlims and ylims as necessary

%% 2. DATA INGEST AND CLEANING
% datafile from CTD, ensure name is correct, script in same folder as data
data = readtable('DATAFILE.TXT');

% ensure the data is labeled correctly, will depend on Arduino code
% get data from data table, BUT it has nans
    EC_nans       =   data(:,1);  %EC data
    TDS_nans      =   data(:,2);  %TDS
    sal_nans      =   data(:,3);  %salinity
    mbar_nans     =   data(:,4);  %mbar
    temp_nans     =   data(:,5);  %temp in C
    dmeter_nans   =   data(:,6);  %depth in meters

        % take nans out of above data
        EC_table      =   rmmissing(EC_nans);
        TDS_table     =   rmmissing(TDS_nans);
        sal_table     =   rmmissing(sal_nans);
        mbar_table    =   rmmissing(mbar_nans);
        temp_table    =   rmmissing(temp_nans);
        dmeter_table  =   rmmissing(dmeter_nans);

    % convert data from table to array for plotting / manipulation
    EC            =   table2array(EC_table);
    TDS           =   table2array(TDS_table);
```

```
    sal           =   table2array(sal_table);
    mbar          =   table2array(mbar_table);
    temp          =   table2array(temp_table);
    dmeter        =   table2array(dmeter_table);

% convert mbar to feet, incase you want to do that
% depth_feet = mbar.*0.033455256555148;

% convert meters to feet, in case you want to use the Mapping Toolbox
% deploy_feet = distdim(dmeter,'meters','feet'); %(needs Mapping Toolbox)
dfeet = dmeter.*3.28084; % I didn't want to use Mapping Toolbox

%% 3. TO FIND CORRECT START / STOP VALUES IN TIME
    % FIND TIME VALUES IN THIS PLOT
    % look in this plot to find where depth values start and stop
    % this is what you will put in below, the (93:135), etc
    % uncomment below --- 56-59
    % n = length(mbar);
    % time = 1:n;
    % figure;
    % plot(time,mbar);

%% 4. DEPLOY DATA PULLING
    % take out only parts of the data for deployments
    % this is the values found 'FIND DEPTH VALUE PLOT'
    % if you have more or less than four, change it

    % ONE
    deploy1_EC       =   EC      (93:135,1);
    deploy1_TDS      =   TDS     (93:135,1);
    deploy1_feet     =   dfeet   (93:135,1);
    deploy1_temp     =   temp    (93:135,1);
    deploy1_sal      =   sal     (93:135,1);

    % TWO
    deploy2_EC       =   EC      (609:660,1);
    deploy2_TDS      =   TDS     (609:660,1);
    deploy2_feet     =   dfeet   (609:660,1);
    deploy2_temp     =   temp    (609:660,1);
    deploy2_sal      =   sal     (609:660,1);

    % THREE
    deploy3_EC       =   EC      (774:799,1);
    deploy3_TDS      =   TDS     (774:799,1);
    deploy3_feet     =   dfeet   (774:799,1);
    deploy3_temp     =   temp    (774:799,1);
    deploy3_sal      =   sal     (774:799,1);

    % FOUR
    deploy4_EC       =   EC      (832:979,1);
    deploy4_TDS      =   TDS     (832:979,1);
    deploy4_feet     =   dfeet   (832:979,1);
    deploy4_temp     =   temp    (832:979,1);
    deploy4_sal      =   sal     (832:979,1);
```

```
%% 5. GRAPHS
% however many you did, make that many subplot graphs
% or you can put them on their own plots (not subplot them)
% these are the same with the numbers changed, 1-4
    figure;
    subplot(2,2,1); % subplot since I want four graphs on one figure
    plot(deploy1_temp,deploy1_feet,'red');
    title('One');
    hold
    plot(deploy1_sal,deploy1_feet,'blue');
    set(gca,'Ydir','reverse'); % to make zero the top of Y axis
    ylim([0 40]);   % use the depth you deployed
    xlim([14.5 19]) % use the range of your data
    legend('temp','salinity')
    ylabel('depth (ft)');
    xlabel('°C | PSU (ppt)');

        subplot(2,2,2);
        plot(deploy2_temp,deploy2_feet,'red');
        title('Two');
        hold
        plot(deploy2_sal,deploy2_feet,'blue');
        set(gca,'Ydir','reverse');
        ylim([0 40]);   % use the depth you deployed
        xlim([14.5 19]) % use the range of your data
        legend('temp','salinity')
        ylabel('depth (ft)');
        xlabel('°C | PSU (ppt)');

    subplot(2,2,3);
    plot(deploy3_temp,deploy3_feet,'red');
    title('Three');
    hold
    plot(deploy3_sal,deploy3_feet,'blue');
    set(gca,'Ydir','reverse');
    ylim([0 40]);   % use the depth you deployed
    xlim([14.5 19]) % use the range of your data
    legend('temp','salinity')
    ylabel('depth (ft)');
    xlabel('°C | PSU (ppt)');

        subplot(2,2,4);
        plot(deploy4_temp,deploy4_feet,'red');
        title('Four');
        hold
        plot(deploy4_sal,deploy4_feet,'blue');
        set(gca,'Ydir','reverse');
        ylim([0 40]);   % use the depth you deployed
        xlim([14.5 19]) % use the range of your data
        legend('temp','salinity')
        ylabel('depth (ft)');
        xlabel('°C | PSU (ppt)');
```