1) Consider the problem which we did in the class
   a. length=2; %%% length
   b. width=5; %%% width
   c. height=7; %%% height
   d. volume=length*width*height;

   We learnt in the class that you can use disp([volume, width]) to display both the values of volume and width.

   Now I want you to learn to display the output in a sentence.

   How can you display this sentence as an output to the user:
   'The volume of the cube is 70 and its width is 5'

   Hint: Use sprintf: https://www.mathworks.com/help/matlab/ref/sprintf.html

2) There will be instances when you want to delete specific variables from the MATLAB workspace. Once you finish the above problem, there will be four variables in the workspace corresponding to length, width, height, and volume. Tell MATLAB to just delete/remove the variable length from the workspace.
   Hint: Use clearvars: https://www.mathworks.com/help/matlab/ref/clearvars.html

3) Create a matrix and name it as given_matrix
   given_matrix=[4:3:27; -30:2:-16; 7:4:38]

$$\text{given\_matrix} = \begin{bmatrix} 4 & 7 & 10 & 13 & 16 & 19 & 22 & 25 \\ -30 & -28 & -26 & -24 & -22 & -20 & -18 & -16 \\ 7 & 11 & 15 & 19 & 23 & 27 & 31 & 35 \end{bmatrix}$$

   a) Print the size of the given_matrix
   b) Replace the element -20 by 3; 15 by 6; and 22 by 8.
   c) Create a new matrix with all the elements with ones with size 3x1. Name this matrix as ones_matrix.
   d) Horizontally concatenate given_matrix and ones_matrix such that ones_matrix forms the last column. Name this new concatenated matrix as final_matrix.
   e) Calculate the mean of the first row of the final_matrix.
   f) Calculate the standard deviation of the third column of the final_matrix.
   g) What is the size of the matrices in 3e and 3f?
   h) Can you sort the elements of the third column of the final_matrix in ascending order? Now sort them in descending order.

4) In the class we learnt that we can use the syntax

   randi(max_integer,num_row, num_col)

   to generate a matrix of defined size with elements between 1 and max_integer (which is a positive integer). Look at the description of randi function and figure out how you can use this function to generate a 3X5 matrix with numbers between -3 and 3.

5) Let us once again go back to the given_matrix which we defined in the beginning of question 3. Our goal is to use the max and min function of MATLAB. Read the description of max (https://www.mathworks.com/help/matlab/ref/max.html) and min (https://www.mathworks.com/help/matlab/ref/min.html) function.

   a. Find the maximum element of each row.
   b. Find the minimum element of each row.
   c. Find the maximum element of each column.
   d. Find the minimum element of each column.

6) In this problem we will learn about creating a diagonal matrix, and matrix multiplication. In a diagonal matrix, the diagonal elements are non-zero. All other non-diagonal elements are zero. For e.g.

$$\begin{bmatrix} 1 & 0 & 0 \\ 0 & -3 & 0 \\ 0 & 0 & 7 \end{bmatrix}$$

   Read the description of MATLAB's diag function (https://www.mathworks.com/help/matlab/ref/diag.html).

   Create a diagonal matrix with diagonal elements 1,-3,7 (as shown above). Name this matrix as diag_matrix. Create another matrix with size 2x3 such that all its elements are 4 (Hint: Use ones and then do scalar multiplication by 4). Name this matrix as all_fours.

   a) Try multiplying diag_matrix with all_fours matrix? Did you get an error? Why?
   b) Take a transpose of all_fours and call it as all_fours_transpose. Now multiply diag_matrix and all_fours_transpose.

7) In the first week, we learnt 1) NaN (not a number) , and 2) Mean of an array. In this problem let us combine both these informations together.

   a. Create an array with elements [1,3,5,7,9]. Name this array as array_7.
   b. Calculate the mean of array_7.
   c. Now create another array and name it nan_array. The elements of nan_array are [1,3,5,7,9,nan]. Hint: You can generate this by using horzcat or by manually entering the elements.
   d. Calculate the mean of nan_array. What is it? NaN? Whenever MATLAB sees atleast one NaN element in an array it outputs NaN for mean and standard deviation. To overcome this, you can use nanmean (https://www.mathworks.com/help/stats/nanmean.html). Try nanmean (nan_array). What is the mean?
   e. Another way to calculate mean of arrays consisting of NaNs is to use mean function but give 'omitnan' as optional input.
      Try mean(nan_array, 'omitmean'). It should give mean to you.
   f. Calculate standard deviation of nan_array using the information provided in 7e.