

1. Write a function that takes a list of booleans as input and prints a dictionary with two keys: "True" and "False". The value associated with the keys is the number of True values found in the input list, and the number of False values, respectively.
 - a. Assume that the input is correct, meaning that it is always a list of booleans.
2. Write a function that takes a list of integers in input and returns a list containing all the positive integers of the input list that are divisible by 13. You must do this in one line of code.
 - a. Assume that the input is always correct, i.e. a list of integers.
3. Write an object Product that has two attributes, "name" (a string) and "quantity" (an integer), and a method to increase or decrease the quantity by a set amount. After this, write an object Blender that inherits from Product and has an additional attribute "capacity" (an integer). Two objects of type Blender should result equal (when tested with "==") if they have the same name, and the same is true for two objects of type Product. An object of type Product and one of type Blender should never be equal, even if they have the same name.
4. Write a function that takes a sorted list of integers as input and returns a list containing all the unique integers in the input list (no repetitions), sorted in the opposite way from the input list. This means that if the input list was sorted in an ascending order, then the output list will be sorted in a descending order.
 - a. Assume that the input is always correct, i.e. it is always a sorted list of integers.
5. Write a function that takes a list of dictionaries as input and returns a list containing all the values from the dictionary (or dictionaries) in the input list that contain the highest number of keys. If the input is not of the correct type (a list of dictionaries), then print the string "WRONG INPUT TYPE" and exit the function.
6. Write a function that takes three integers in input, a, b and c. The function will return a Numpy array of shape aXb filled by random integers between 0 and c.
 - a. Ex: if a = 2, b = 3, and c= 12, a possible output can be: array([[9, 10, 10], [8, 7, 8]])
 - b. Ps. since we are dealing with random integers, your result with the same parameters might vary.
7. Write a function that takes two lists of integers and returns a numpy array that is the result of the multiplication of the two input lists. The output needs to be a vector of exactly 5 integers, so if the array resulted from the multiplication has less than 5 integers, then you need to insert 0s for padding until it reaches 5 integers of length (you can choose in which direction you want to insert the padding). If the array resulted from of the multiplication contains more than 5 integers, then it needs to be truncated to only 5. Do not check the input for correctness, but if an exception arises print "EXCEPTION" and exit the function.

- a. Ex: if the two input lists are [1,2,3,4] and [5,6,7,8], the output should be array([5, 12, 21, 32, 0]) or array([0, 5, 12, 21, 32]), depending on which padding direction was chosen.
8. Write a function that takes a Numpy array of integers as input and returns an integer representing the number of negative odd integers in the input array. Do not use for loops, while loops, if statements and list comprehensions.
9. Write a function that takes a list of Numpy arrays (containing only integers) as input and returns the array that has the highest mean, and the mean itself.
10. Write a lambda function that takes an integer "n" in input and returns a Numpy array made of "n" zeros. Write an example usage of this function.
11. Without changing the names of the columns and using Pandas, merge the three csv into a unified DataFrame. Use an outer join.
12. Perform the usual steps of EDA, annotating every step with comments that details what kind of information you are extracting.
13. Remove the redundant columns (choose wisely which ones to remove) and fill the NaN in the numeric columns with the mean value of the specific column.
14. Using pyplot, create two subplots in the same canvas. In the first you need to show a bar chart that shows the mean of the salaries for people with a parking spot, and people without (this should only have two bars). In the second subplot you need to show a bar chart of the mean of the salaries per position (this will have many more bars).
15. Using pandas, answer the question: On average, do the people that have a parking spot make more money than the people that do not have a parking spot?