



# CODE LEVELING



PROVE OF CONCEPT

semicolon





# RULE FOR THIS GAME GIVEN BY THE SYSTEM



⇒ TEST THE CODE FOR EDGE CASES AND WRITE UNIT  
TEST FOR EACH FUNCTION

⇒ THE CODE SHOULD BE CLEANED UP AND OPTIMISED  
WITH GOOD NAMING CONVENTIONS

⇒ ONLINE OUTSOURCING OF THE CODE IS PROHIBITED, I  
SHOULD SEE JUST YOUR THOUGHT PROCESS IN THE  
CODE NOT OUR UNCLES OWN



**semicolon**



1. Create a tuple of numbers and append a new number to it **without** directly modifying the tuple.
2. Given `numbers = (1, 2, [3, 4], 5)`, change the third element's second value to 99.
3. Convert a tuple of strings ( 'apple' , 'banana' , 'cherry' ) into a list, add 'mango' , and convert back to tuple.
4. Write a program to unpack this tuple: (10, 20, 30, 40) into variables a, b, and the rest in the last variable.

**semicolon**





1. Given a 2D list, write a function that sum up all the element in each of the inner list and return a single list.

Sample input : [ [2, 3, 4], [1, 5, 7], [4, 6, 8] ]

Sample ouput: [9, 13, 18]

2. Wite a function that sum all the element in the corresponding index for each inner list. i.e all the elements in index 0 summed up, the same for index 1, down to the last index

Sample input : [ [2, 3, 4], [1, 5, 7], [4, 6, 8] ]

Sample ouput: [7, 14, 19]





# RULE FOR THIS GAME GIVEN BY THE SYSTEM

IN ADDY TO THE PREVIOUS RULE

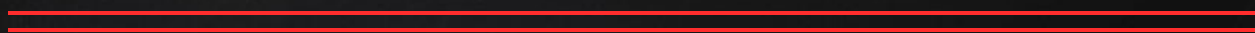
⇒ FROM THIS POINT EVERY TASK MUST BE COMPLETED

USING MAP, FILTER OR REDUCE

⇒ NO EXPLICIT LOOPS IN THE CODE



**semicolon**







1. Using `filter()`, get all even numbers from the list `range(1, 21)`.
2. Write a function that extract only words longer than 5 characters from `['cat', 'elephant', 'tiger', 'lion']`.
3. Given a list of tuples `[(1, 'A'), (4, 'B'), (2, 'C')]`, use `filter()` to keep only tuples where the first value is greater than 2.
4. Use `filter()` to get all numbers from `range(1, 51)` divisible by both 3 and 5.
5. Filter all palindromes from `['level', 'world', 'madam', 'python']`.





1. Using `map()`, convert all strings in `['python', 'java', 'c++']` to uppercase.
2. Use `map()` to square all numbers from 1 to 10.
3. Given `names = ['john', 'mary', 'steve']`, use `map()` to capitalize the first letter of each name.
4. Given `prices = [100, 200, 300]`, use `map()` to add 10% tax to each price.



**semicolon**

---

---



1. Using `reduce()`, find the sum of numbers from 1 to 50.
2. Using `reduce()`, find the maximum number in `[3, 5, 9, 2, 8]`.
3. Using `reduce()`, concatenate all strings in `['I', 'love', 'Python']` into a single string with spaces.
4. Using `reduce()` and `map()`, find the product of the squares of numbers `[1, 2, 3, 4]`.



**semicolon**

---

---





1. Given a list of tuples `[(1, 2), (3, 4), (5, 6)]`, write a function that sum the elements of each tuple, then use `filter()` to keep only sums greater than 5.
2. Given `data = ['123', '456', '789', 'abc', 'def']`, write a function that remove non-numeric strings, then convert them into integers, then find and return their total sum.





THANKS!

Hope that was fun?



semicolon

