Challenge 1: String Frequency Analyzer

Problem

- Given a string, count frequency of each character.
- Output sorted by character.

Example

Input: "banana"

Output: a:3, b:1, n:2

Constraints

- Case-sensitive
- No collections.Counter

Concepts Tested

- Dictionaries
- Looping
- Sorting

Challenge 2: Reverse Words in Sentence

Problem

- Reverse each word, not the sentence order.

Input

"Python is easy"

Output

"nohtyP si ysae"

Concepts Tested

- String slicing
- Split & join

Challenge 3: FizzBuzz++

Rules

- Multiple of 3 → Fizz
- Multiple of 5 → Buzz
- Multiple of both → FizzBuzz
- Multiple of 7 → Boom

Constraints

- Use only one loop
- No nested if-else

Concepts Tested

- Clean logic
- Condition optimization

Challenge 4: Find Missing Number

Problem

- Array of size n contains numbers from 1 to n+1
- One number missing

Constraints

- O(n) time
- O(1) space

Concepts Tested

- Mathematical thinking
- XOR / Sum formulas

Challenge 5: Second Largest Element

Problem

- Find second largest without sorting

Constraints

- One pass
- Handle duplicates

Concepts Tested

- Edge cases
- Loop optimization

Challenge 6: Valid Parentheses

Problem

- Check if string "({[]})" is valid

Constraints

- Use stack
- No regex

Concepts Tested

- Stack
- Dictionary mapping

Challenge 7: Rotate Array

Problem

- Rotate array right by k steps

Constraints

- In-place solution preferred

Concepts Tested

- Slicing
- Modulo logic

Challenge 8: Custom Map Function

Problem

- Implement your own map() function

```
def my_map(func, iterable):
    ...
```

Constraints

- Use yield

Concepts Tested

- Generators
- Higher-order functions

Challenge 9: Employee Management System (OOP)

Requirements

- Class Employee
- Attributes: name, salary
- Method: annual_salary()
- Inheritance: Manager with bonus

Concepts Tested

- OOP
- Inheritance
- Method overriding

Challenge 10: Singleton Class

Problem

- Implement Singleton pattern in Python

Constraints

- Must prevent multiple instances

Concepts Tested

- __new__
- Class behavior

Challenge 11: Word Frequency from File

Problem

- Read a file
- Count words
- Ignore punctuation & case
- Print top 5 frequent words

Concepts Tested

- File handling
- Text processing
- Sorting by value

Challenge 12: Decorator for Execution Time

Problem

- Write a decorator that prints function execution time

Constraints

- Use time module
- Must work with arguments

Concepts Tested

- Decorators
- *args, **kwargs

Challenge 13: LRU Cache

Problem

- Implement LRU cache from scratch

Constraints

- O(1) get & put
- No functools.lru_cache

Concepts Tested

- Dictionary + Doubly Linked List
- Design thinking

Challenge 14: Multithreading Task

Problem

- Print numbers using two threads:
    - Thread A prints even
    - Thread B prints odd

Concepts Tested

- threading
- Locks
- Race conditions

Challenge 15: Mini URL Shortener

Requirements

- Convert long URL to short
- Retrieve original URL

Constraints

- Use dictionary
- Generate unique keys

Concepts Tested

- Hashing
- System design basics

Challenge 16: Log File Analyzer

Problem

- Read log file
- Find:
    - Most frequent IP
    - Error count
    - Requests per minute

Concepts Tested

- File parsing
- Real-world data handling