Mutation and String methods

[Problem-1]

i. Take input "hello world" and convert it into uppercase.

ii. Given "python is fun", convert only the word "python" into uppercase using slicing + upper().

[Problem-2]

i. Convert "HELLO PYTHON" into lowercase.

ii. Given "Python Programming", make only "Programming" lowercase.

[Problem-3]

i. Find how many times "a" appears in "banana".

ii. Count the number of times "is" occurs in "This is Python. It is easy.".

[Problem-4]

i. Remove extra spaces from " openai ".

ii. Given "***python***", strip * from both sides.

[Problem-5]

i. Replace "Java" with "Python" in the string "I love Java".

ii. In "111223344", replace all "1" with "X".

[Problem-6]

i. Join the list ["AI", "ML", "DL"] into a string separated by , .

ii. Take a list ["OpenAI", "ChatGPT", "Python"] and join with " - "

[Problem-7]

i. Split "I love Python programming" into words.

ii. Split "apple, banana, mango" into a list using , .

[Problem-8]

i. Extract "Python" from "I am learning Python Programming".

ii. Extract the last 3 characters of "OpenAI".

Real Application Tasks

1. Customer Feedback Analysis

Your company receives customer feedback in mixed cases:

["Service was GOOD, but the staff was rude. GOOD ambiance. good price."]

- Normalize the text to lowercase.
- Count how many times the word "good" appears.

(Methods: lower(), count(), split())

2. University Registration System

Students submit their names with random spaces:

[" Ali Raza ", " Sana Khan", "HASSAN "]

- Clean the names by removing spaces.
- Convert them to proper title case (first letter capitalized).

(Methods: strip(), replace(), title()).

3. Email List Processing

From a database, you get emails like:

["ahmed@gmail.com", "user@yahoo.com", "info@company.org"]

- Extract only the **username** part (before @).
- Extract only the **domain** part (after @).

(Methods: split(), substring slicing, negative indexing).

4. Error Log Report

A system log file contains entries like:

INFO: Server started\nERROR: Connection failed\nWARNING: Low memory\nERROR: Timeout"

- Count how many times "ERROR" appears.
- Extract and display only the messages after "ERROR:".

(Methods: upper(), count(), index(), slicing).

5. Library Book Titles

Book titles in the system are stored with dashes and inconsistent cases:

["harry-potter-and-the-chamber-of-secrets", "lord-OF-the-rings", "GAME-of-thrones"]

- Replace dashes with spaces.
- Convert them into proper title case.
- Join all titles into a single string separated by ;.

(Methods: replace(), title(), join()).

6. Student Roll Number Validation

Rules: Roll numbers must be like CS2023001 → **2 letters + 4 digits (year) + 3 digits (serial)**.

Given:

["CS2023001", "EE202A003", "cs2023002"]

- Validate each roll number.
- Mark them as **Valid** or **Invalid**.

(Methods: isupper(), isdigit(), substring slicing).

7. Online Store – Product Codes

An e-commerce website stores product codes like:

[" LAPTOP-123 ", "MOBILE-456", " TV-789 "]

- Clean extra spaces.
- Separate product name and product ID.
- Display in the format: "Product: Laptop | Code: 123".

(Methods: strip(), split(), title()).

8. Resume Filtering (HR Department)

HR wants to check if resumes mention "Python" (case-insensitive).

["Expert in PYTHON and Java", "C++ Developer", "python backend engineer"]

- Normalize case.
- Count how many resumes mention "python".

(Methods: lower(), count()).

9. Hospital Patient Records

Patients’ data is stored as:

"Name: Ali, Age: 24, Disease: Fever"

- Extract **Name** and **Disease** only.
- Display in format: "Patient Ali has Fever".

(Methods: split(), index(), substring slicing).

10. Flight Booking System

Passenger names are saved in reverse order (last name first):

["Khan, Ali", "Ahmed, Sana", "Malik, Hassan"]

- Split and reformat them into "Ali Khan", "Sana Ahmed", "Hassan Malik".

(Methods: split(), join()).