LAB 2

---

1. Write a program to input n numbers and store them in a list. Then perform the following operations: i) Using built in functions ii) without using built-in functions:
    a. Find the maximum and minimum number
    b. Sort the list in ascending order
    c. Remove duplicate elements
2. Given two lists of integers, write a program to merge them into a single list and then remove the elements that are common in both.
3. Create a program that reads a sentence from the user and stores each word as an element of a list. Then count the frequency of each word using only lists.
4. Write a program to simulate a basic stack and queue using a list. Provide options to push, pop (stack), enqueue, and dequeue (queue).
5. Write a Python function that accepts a list and returns a new list containing only the elements at even indexes and those that are prime numbers.
6. Write a program to create a tuple of n numbers, then find:
    a. The average of the numbers
    b. The median
    c. The mode (without using libraries)
7. Write a program that receives a list of tuples representing (x, y) coordinates. Determine whether the points form a straight line.
8. Write a program to input two sets of student roll numbers: one who play cricket and another who play football. Find:
    a. Students who play both sports
    b. Students who play only one sport
    c. Students who play neither (given a master list of all students)
9. Create a set of random numbers. Add more numbers until the set has 10 unique elements. Also, remove the smallest and largest element.
10. Write a Python function that accepts a sentence and returns a set of all unique vowels used.
11. Given a list of numbers with duplicates, use a set to remove the duplicates. Then, convert it back to a sorted list and display the result.
12. Create a dictionary to store student names as keys and their scores in three subjects as values (in a list). Write functions to:
    a. Display the average marks of each student
    b. Find the topper
    c. Update the marks of a student
13. Write a program that reads a text and counts the frequency of each character (excluding spaces and special characters) using a dictionary.
14. Build a dictionary where the keys are product names and the values are their prices. Implement options to:
    a. Add a new product
    b. Update price of an existing product

c. Find products within a given price range

MINI PROJECT: Student Report Card Management System

Problem Statement:

Design and implement a Student Report Card Management System using Python that allows a teacher to:

- Add new student records (name, roll number, subject-wise marks).
- View the report of all students.
- Display the topper(s) of the class based on average marks.
- Search for a student by roll number.
- Display all students who have failed in one or more subjects.
- Optionally update marks of any student.