1. **Scenario:** A user is required to enter a valid number in a form, but users sometimes input invalid data.

   Write logic to repeatedly prompt the user until they enter a valid integer.

   **Logic:**

   a) **Get the input from user.**
   b) **Check if meets requirements.**
   c) **If not, Ask user until they enter the valid number.**

2. **Scenario:** A data analysis tool processes a list of numbers and needs to identify the most frequently occurring value.

   Write logic to find the most frequently occurring number in a given list.

   **Logic:**

   a) **Read the numbers.**
   b) **Check the numbers frequently occuring number.**
   c) **If yes, print the number**.

3. **Scenario:** A text-processing application needs to compare words and check if they are anagrams (contain the same letters in a different order).

   Write logic to determine whether two given strings are anagrams.

   Logic:

   a) **Read the given words.**
   b) **Check the words have the same length.**
   c) **If not, print the words not anagrams.**
   d) **if yes, convert words into characters and compare the words are having exactly same letters.**
   e) **If true print the words are anagrams.**

4. **Scenario:** A speech analysis program needs to count the number of vowel sounds in a given input.

   Write logic to count the number of vowels in a given string.

   **Logic:**

   a) **Define the set of Vowels.**
   b) **Initialize the counter and set as zero.**
   c) **Convert the given string into characters**
   d) **Iterate through the character by character and Check for Vowels.**
   e) **If yes, add to the counter = counter+1**
   f) **Once loop finishes print counter.**

5. **Scenario:** A text-editing software includes a feature to reverse the order of words in a sentence for stylistic effects.

   Write logic to reverse the order of words in a sentence while keeping the words themselves intact.

   **Logic:**

   a) **Read the sentence.**
   b) **Splitting the sentence into words.**
   c) **Reversing the order.**
   d) **Reconstructing the sentence.**

6. **Scenario:** A missing number is detected in a sequence of values stored in a database.

   Write logic to find the missing number in a list containing n-1 numbers from 1 to n.

   **Logic:**

   a) **Determine the full sequence length.**
   b) **Calculate the expected sum.**
   c) **Calculate the actual sum.**
   d) **Subtract the actual sum from the expected sum.**
   e) **Print missing number.**

7. **Scenario:** An ATM machine processes withdrawal requests and needs to ensure that users cannot withdraw more than their account balance.

   Write logic to allow a withdrawal only if the balance is sufficient.

   **Logic:**

   a) **Read the account balance.**
   b) **Check If the withdrawal amount is less than or equal to the account balance.**
   c) **If yes, allow the withdrawal.**
   d) **Else, print no sufficient balance.**

8. **Scenario:** A system needs to verify whether a given dataset contains duplicate entries.

   Write logic to check whether a given list contains duplicate values.

   **Logic:**

   a) **Read the given list.**
   b) **Create an empty set and initialize.**
   c) **Iterate through the given list, one by one.**

d) **Check the duplicates.**
e) **If find duplicates, print duplicate exist.**
f) **Else, the element is unique add it to the set.**

9. **Scenario:** A digital calculator includes a feature to sum the digits of a number for verification purposes.

   Write logic to calculate the sum of all digits in a given integer.

   **Logic:**

   a) **Read the numbers.**
   b) **Calculate sum of digits.**
   c) **Start loop that input number is greater than 0.**
   d) **When the number is divided by 10. Remainder is always last digit.**
   e) **Add last digit to sum**

10. **Scenario:** A language-learning app wants to verify whether a given sentence is a pangram (contains every letter of the alphabet at least once).

    Write logic to check if a given sentence is a pangram.

    **Logic:**

    a) **Define the alphabet containing 26 small letters.**
    b) **Convert the sentence to lowercase if any.**
    c) **Convert the word into characters.**
    d) **Check if the 26 letters are present.**
    e) **Print pangram.**
    f) **Else, not pangram.**