1. **Scenario:** You are developing a banking application that categorizes transactions based on the amount entered.

   Write logic to determine whether the amount is positive, negative, or zero.

   1. Read the amount entered.
   2. Check conditions with the entered amount.
      - If entered amount > 0 -> the transaction is positive (Deposit).
      - Else if the entered amount < 0 -> the transaction is negative (Withdrawal).
      - Else – zero transaction

2. **Scenario:** A digital locker requires users to enter a numerical passcode. As part of a security feature, the system checks the sum of the digits of the passcode.

   Write logic to compute the sum of the digits of a given number.

   1. Get input numerical passcode from the user.
   2. sum all the digits of the passcode using python "sum(digit) for digit in passcode".

3. **Scenario:** A mobile payment app uses a simple checksum validation where reversing a transaction ID helps detect fraud.

   Write logic to take a number and return its reverse.

   1. Read the transaction ID.
   2. Sum all the digits and take modulus of 10 == 0 -> valid

3. Reverse the ID and sum all the digits and take modulus of 10 == 0 -> valid

4. **Scenario:** In a secure login system, certain features are enabled only for users with prime-numbered user IDs.

   Write logic to check if a given number is prime.

   1. Read the User ID.
   2. check whether the given numbered user ID is a prime user, if true, enable features of prime users.

5. **Scenario:** A scientist is working on permutations and needs to calculate the factorial of numbers frequently.

   Write logic to find the factorial of a given number using recursion.

   1. Read the number
   2. crate a function to perform factorial using recursion.

6. **Scenario:** A unique lottery system assigns ticket numbers where only Armstrong numbers win the jackpot.

   Write logic to check whether a given number is an Armstrong number.

   1. Get the lotter Number
   2. Find sum of powers of length of the digits to check the number is Armstrong number.
   3. compare whether the calculated number and the given number are same. Then assign that number as jackpot winning number.

7. **Scenario:** A password manager needs to strengthen weak passwords by swapping the first and last characters of user-generated passwords.

   Write logic to perform this operation on a given string.

   1. Get the password and assign it into a variable.

2. Use swap(first letter , last letter) function to swap the first and last digit.

First letter -> text[0]

Last letter -> text [-1]

3. After, generate the new password.

8. **Scenario:** A low-level networking application requires decimal numbers to be converted into binary format before transmission.

Write logic to convert a given decimal number into its binary equivalent.

1. Get the number and convert it into binary format using python inbuilt function bin(number)

9. **Scenario:** A text-processing tool helps summarize articles by identifying the most significant words.

Write logic to find the longest word in a sentence.

1. use split() → Splits the text into a list of words.
2. max(words, key=len) → Finds the word with the largest length.
3. Return the Word

10. **Scenario:** A plagiarism detection tool compares words from different documents and checks if they are anagrams (same characters but different order).

Write logic to check whether two given strings are anagrams.

1. Convert both the strings into lower case and Remove spaces

2. Sort each characters of the two string.

3. compare the sorted list. If both are identical – strings are anagrams. Else -> not anagrams.

Ramishahope Artificial Intelligence Pvt Ltd

36, Old Anandas, SG Arcade, Marudhamalai Main Road, Vadavalli, Coimbatore -641041.

+91 6385383227 | www.hopelearning.net | mdaravind@hopelearning.net | 33AAMCR3722R1ZU