

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.

Logic:

1. Get input from the user
 2. If input is greater than 0 print positive
 3. Or equal to 0 print 0
 4. Input is less 0 print negative
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.

Logic:

1. Get input from the user
 2. Already store the sum of the input
 3. Using sum to add up the input
 4. If sum is equal to stored value
 5. Then print the sum value
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.

Logic:

1. Take input from the user
2. Using slicing -1 to reverse the number

3. Print the reverse number

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.

Logic:

1. Get input
2. if input less than 2, it is not prime
3. in loop from 2 to input if divisible not prime, not divisible it's prime



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.

Logic:

1. Get input from the user
2. If input is 0, no possibility of factorial
3. Variable as 1 then loop start 1 upto input +1
4. Variable as 1 that multiply with every number in the loop
5. Then print the factorial

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.

Logic:

1. Get input from the user
2. Convert int to str to get the len
3. Using sum() to generate expression to calculate powered sum
4. Then compare with original

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.

Logic:

1. Get input from the user
2. Check atleast contain 2 charcters
3. Using operator to collect [1],[1:-1],[0]
4. Print the swapped 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.

Logic:

1. Get input from user
2. Convert into bin it has start with 0b
3. Then remove 0b using [2:]
4. Print the binary 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.

Logic:

1. Get input sentence from the user
2. Split sentence into words
3. Find the len of the word
4. Then find max value after print the longest 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.

Logic:

1. *Get input from the user as a lower case*
2. *Sorted the input remove space*
3. *If two input are equal then print anagrams*
4. *Or else print not anagrams.*

H O P E L A R N I N G

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