

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:**

- Read the amount from the input.
- Select the transaction type from the given list
- Then, based on the transaction type if it's Deposit, the amount will be add to the account and it will be the positive transaction
- If it's withdrawal, then it's a negative transaction
- Else, zero denotes no 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.

**Logic:**

- Read the number from the input.
- Write a logic, Separate the numbers to the individual digits.
- Then verify the digits are greater than 0, If it's greater than 0 then sum the digits and store them to the variable.
- Compare the final value to the existing value, If it's true print as correct.
- Else, Print as incorrect.

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:**

- Enter the transaction ID as an input.
- Read the ID and convert it to the string
- Reverse the string and convert it back to the integer
- Print the reversed 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:**

- Enter the number as an input
- Verify the number is prime number or not
- If the given number is prime, then apply the given features for the users
- Else, the user will have the basic feature by default

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:**

- Enter the number as an input
- Write the logic to find the factorial of the given number
- If the number is 0 or 1, then result will be 1.
- Else, Multiply the given number by the factorial of (factorial number -1)
- Then, print the output

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:**

- Enter the number as an input
- Split the given numbers into the individual digits and check the number of digits
- Use the number of digits value as the power of all the digits and sum the same assign it to the variable
- Then, If the given input and the Sum of the digits were equal
- Print "Armstrong number"
- Else, Print "Not a Armstrong 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.

**Logic:**

- Read the input from the user
- Check the length of the string
- If the String length is less than 2, print the same as output
- Else, Swap the first and last digit
- Then print the output

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:**

- Get the input from the user
- Check whether the given number is greater than 0
- If the given number is greater than 0, then start dividing the numbers and hold the remainder to the binary string
- Then, Reverse the remainder
- Print the output

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:**

- Get the input from the user as a string
- Split the words using split() function
- Then compare the length of the word with each other
- Store it in the variable and loop through all the words
- If any longest word is found in the sentence, update the variable
- Then, Print the result

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:**

- Get the 2 inputs from the user as a string
- Read the string, Remove the spaces and convert it to the lower case to avoid the case sensitive issues
- Then use the sort to arrange the letters, If the sorted both the strings are same
- It's an Anagram.
- Else, Not an Anagram.