**Scenario-Based Questions**

**Solutions**

### **Scenario: Identifying Transaction Type**

**Logic:**

* Read the input number.
* If the number is greater than 0, print "Positive (Deposit)".
* Else if the number is less than 0, print "Negative (Withdrawal)".
* Else, print "Zero (No Transaction)".

### **Scenario: Summing the Digits of a Number**

**Logic:**

* Read the input number.
* Convert the number into individual digits.
* Initialize a sum variable to 0.
* For each digit in the number, add it to the sum variable.
* Print the sum of the digits.

### **Scenario: Reversing a Transaction ID**

**Logic:**

* Read the input number.
* Convert the number into a string.
* Reverse the string.
* Convert it back to a number.
* Print the reversed number.

### **Scenario: Checking if a Number is Prime**

**Logic:**

* Read the input number.
* If the number is less than 2, print "Not Prime".
* Loop from 2 to the square root of the number:
  + If the number is divisible by any of these values, print "Not Prime" and exit.
* If no divisors are found, print "Prime".

### **Scenario: Finding the Factorial Using Recursion**

**Logic:**

* Read the input number.
* If the number is 0 or 1, return 1.
* Else, return the number multiplied by the factorial of (number - 1).
* Print the result.

### **Scenario: Checking if a Number is an Armstrong Number**

**Logic:**

* Read the input number.
* Count the number of digits.
* Initialize a sum variable to 0.
* For each digit in the number:
  + Raise the digit to the power of the total number of digits.
  + Add the result to the sum variable.
* If the sum is equal to the original number, print "Armstrong Number".
* Else, print "Not an Armstrong Number".

### **Scenario: Swapping First and Last Characters of a String**

**Logic:**

* Read the input string.
* If the string length is less than 2, print the string as is.
* Swap the first and last characters while keeping the middle part unchanged.
* Print the modified string.

### **Scenario: Converting Decimal to Binary**

**Logic:**

* Read the input decimal number.
* Initialize an empty string for binary representation.
* While the number is greater than 0:
  + Divide the number by 2 and store the remainder.

**Ramishahope Artificial Intelligence Pvt Ltd**

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

**+91 6385383227 |** [**www.hopelearning.net**](http://www.hopelearning.net/) **|** [**mdaravind@hopelearning.net**](mailto:mdaravind@hopelearning.net) **| 33AAMCR3722R1ZU**

* + Add the remainder to the binary string.
  + Update the number by dividing it by 2.
* Reverse the binary string.
* Print the binary representation.

### **Scenario: Finding the Longest Word in a Sentence**

**Logic:**

* Read the input sentence.
* Split the sentence into individual words.
* Initialize a variable to store the longest word.
* Loop through each word:
* if the current word is longer than the stored longest word, update the longest word.
* Print the longest word.

### **Scenario: Checking if Two Strings are Anagrams**

**Logic:**

* Read the two input strings.
* Remove spaces and convert both strings to lowercase.
* Sort the characters of both strings.
* If the sorted versions of both strings are identical, print "Anagram".
* Else, print "Not an Anagram".