### 

Python set2 Logic

### **1. Identifying Transaction Type**

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

### **2. Summing the Digits of a Number**

* Get 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.

**3. Reversing a Transaction ID**

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

### **4. Checking if a Number is Prime**

* Get 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".
* If no divisors are found, print "Prime".

**5. Finding the Factorial Using Recursion**

* Get 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.

### **6. Checking if a Number is an Armstrong Number**

* Get 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".

### **7. Swapping First and Last Characters of a String**

* Get 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.

**8. Converting Decimal to Binary**

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

### **9. Finding the Longest Word in a Sentence**

* Get 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.

### **10. Checking if Two Strings are Anagrams**

* Get 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".