Test Cases for Addition of Two Numbers:

- [ ] **Check the addition of two positive integers gives the correct result or not**
  Example: Input: 5 + 5, Expected output: 10
- [ ] **Check if the addition of a positive integer and a 0 results the positive integer itself**
  Example: Input: 3 + 0, Expected output: 3
- [ ] **Check if the addition of positive integer and negative integer gives the correct result or not**
  Example: Input: 10 + (-3), Expected output: 7
- [ ] **Check the addition of a negative integer and zero returns the negative number itself**
  Example: Input: (-5) + 0, Expected output: -5
- [ ] **Check the addition of two negative integers gives the correct result**
  Example: Input: (-3) + (-3), Expected output: -6
- [ ] **Check the addition of two floating point numbers gives the correct result**
  Example: Input: 10.5 + 10.5, Expected output: 21
- [ ] **Check the addition of a positive integer and a floating point number gives correct result**
  Example: Input: 20 + 5.6, Expected output: 25.6
- [ ] **Check the addition of a floating point number and zero returns the float number itself**
  Example: Input: 2.5 + 0.0, Expected output: 2.5
- [ ] **Check the maximum range for addition of two numbers of any data type**
  Example: Input: 9999999999 + 8888888888, Expected output: 18888888887 (Integer)
- [ ] **Check the minimum range for addition of two integer numbers of any data type**
  Example: Input: (-999) + (-111), Expected output: -1110 (Integer)
- [ ] **Check if addition between two 0’s returns an answer 0**
  Example: Input: 0 + 0, Expected output: 0
- [ ] **Check the addition of two numbers with large decimal places**
  Example: Input: 0.25252589635 + 0.658724789123, Expected output: 0.9112506855
- [ ] **Check the the ability to handle non-numeric input or special characters**
  Example: Input: 2@ + $, Expected output: Error or Not a Number