TypeScript Assignment:

1. Create a TypeScript file that defines a "User" interface and a function called "printUserInfo." The function should accept a parameter of type "User" and print out the user's name, age, and email.

2.Write a generic function called "reverseArray" that takes an array of any type and returns a new array with its elements reversed.

3. Define two interfaces: "Animal" and "Fish." Then, create a function called "printPet" that accepts a parameter of type "Animal | Fish" and prints the common properties and the unique properties of the input.

4.Create a TypeScript file and define a function called "combine" that takes two parameters and returns their concatenated values. The parameters can be of any type, but the return type should be automatically inferred based on the input.

5.Create a TypeScript enum called "DaysOfWeek" to represent the days of the week. Then, write a function called "getDayName" that takes a parameter of type "DaysOfWeek" and returns the corresponding day's name as a string.

6.Create an interface called "MathOperation" with a single method called "calculate." The calculate method should take two numbers as parameters and return the result of a mathematical operation between them. Then, implement the "MathOperation" interface in different classes to perform various mathematical operations.

1.Define the interface "MathOperation" with the method calculate.

2.Create a class called "Addition" that implements the "MathOperation" interface. The calculate method in the "Addition" class should return the sum of the two input numbers.

3.Create a class called "Subtraction" that implements the "MathOperation" interface. The calculate method in the "Subtraction" class should return the result of subtracting the second input number from the first one.

4.Create a class called "Multiplication" that implements the "MathOperation" interface. The calculate method in the "Multiplication" class should return the product of the two input numbers.

5.Test the functionality of each class by creating instances and calling the calculate method.