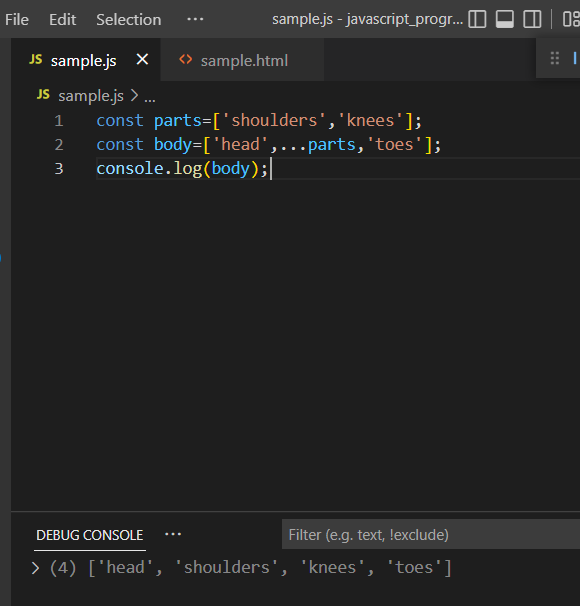
**Chapter 9: rest and spread**

**Spread:** (...args)

The spread allows an iterable such as array or string to be expanded in places where a function call or array literals are expected.

In a more simple way, the spread expands an array into its individual elements.

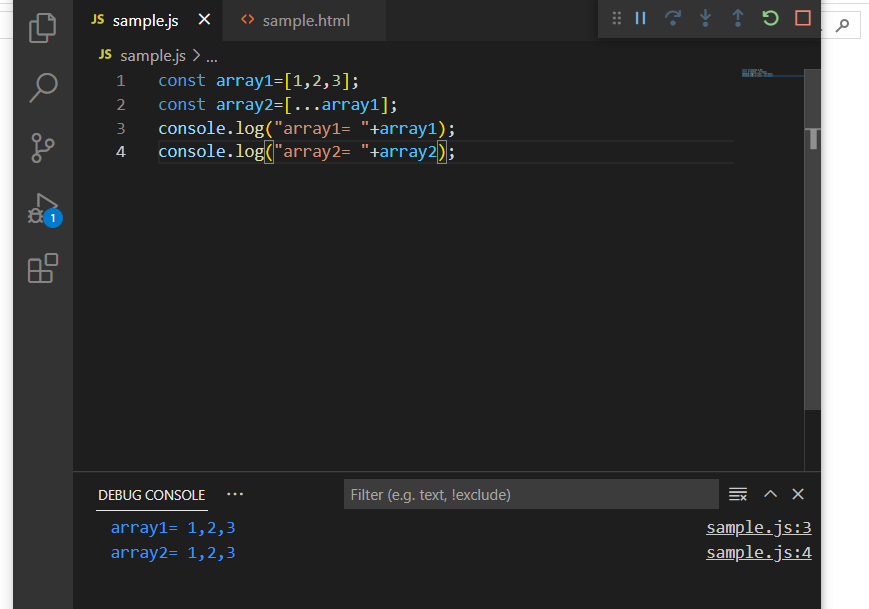
**Spread in array literal:**



In the above example, the spread operator expands array literals in the body array.

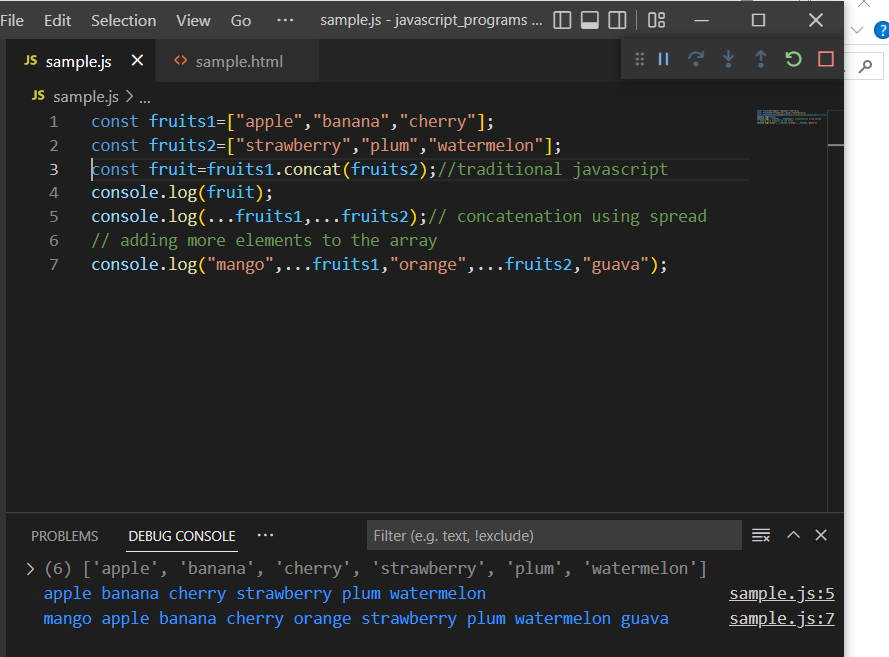
**Copy array using spread:**

Using spread we can copy one array to another.



**Concatenation of arrays using spread:**

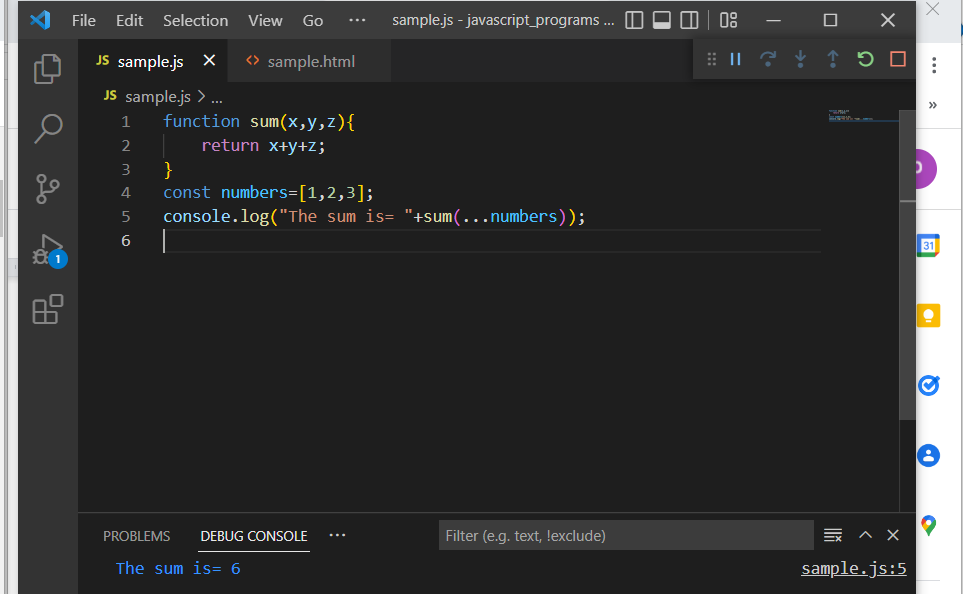
Spread allows to concatenate arrays easily rather than using the traditional javascript to concatenate.



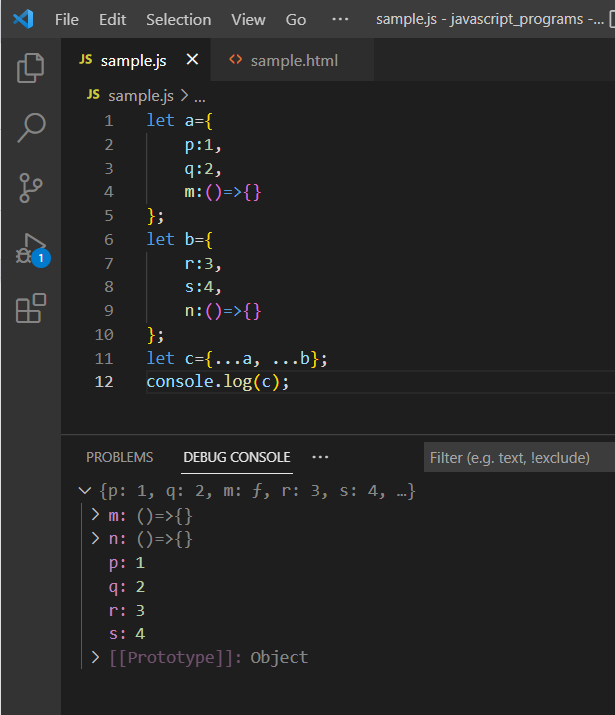
It also allows to add new elements in the array during concatenation.

**Spread in function calls:**

In this example, the array is passed as a parameter using spread.

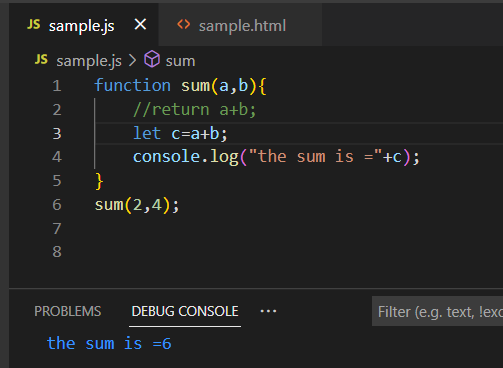


**Merging objects using …spread:**

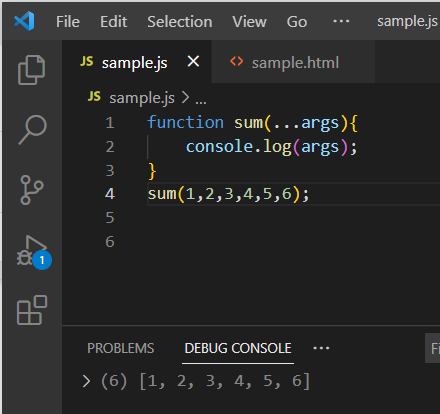
****

**Rest: (...args):**

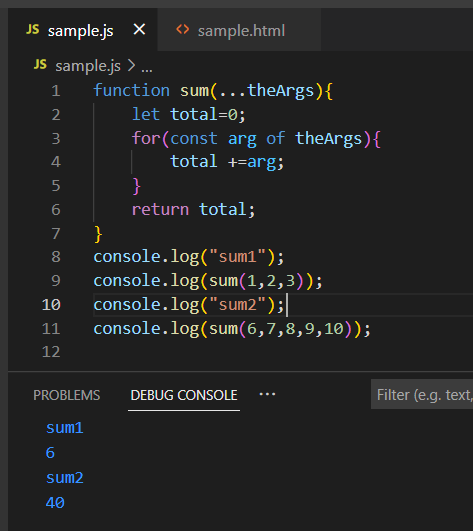
The rest parameter collects all the remaining parameter into an array.



In the above example the sum function is restricted to two values. In order to perform the sum function for n number of values there …rest is used.

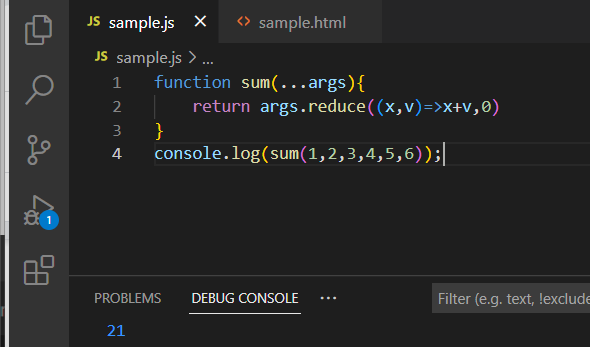


Here the values passed are grouped into an array.

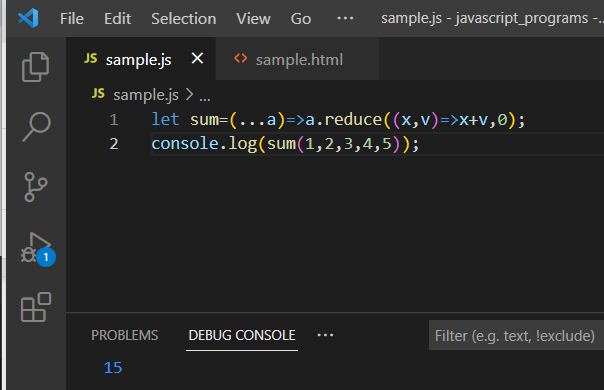


Here the sum of n different numbers is performed using the …rest and the addition is performed using the for..of loop to access the values.

In the above example …args produces an array, which is iterable so we can use a reducer to perform the sum operator instead of for..of loop.



This also works for arrow function as follows. This allows to reduce the steps in the program.

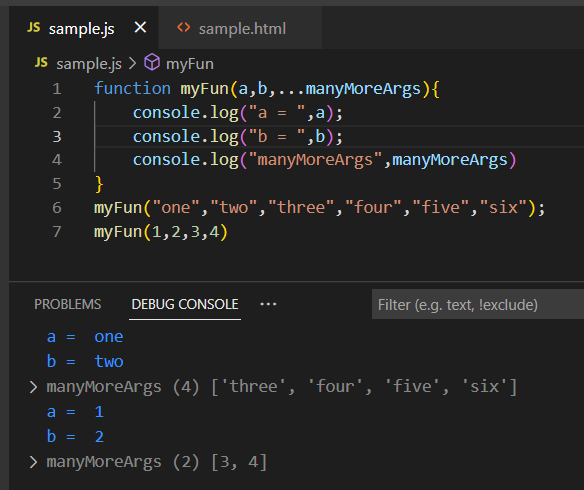


**Note:** the …rest should be only argument or it should be last on the list.

Function sum(...args,b,c){} // error.

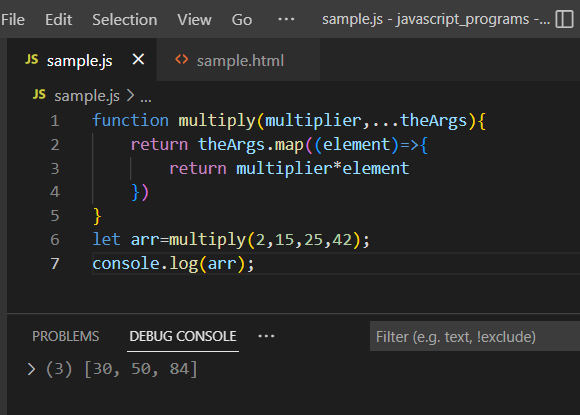
Function sum(a,…args,c){} //error

Function sum(a,b,...args){} //correct form



In the above example when a group of values are passed to the function as a parameter, the first two values will be taken as ‘a’ and ‘b’ and the rest will be grouped into an array.

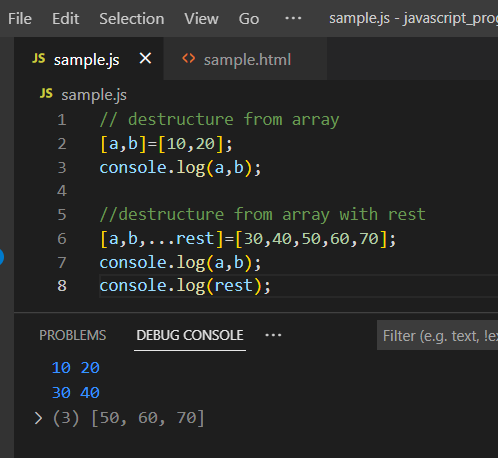
This …rest can be used for other manipulation purpose also which will intern reduce the number of steps. Like..



**Destructuring Assignment:**

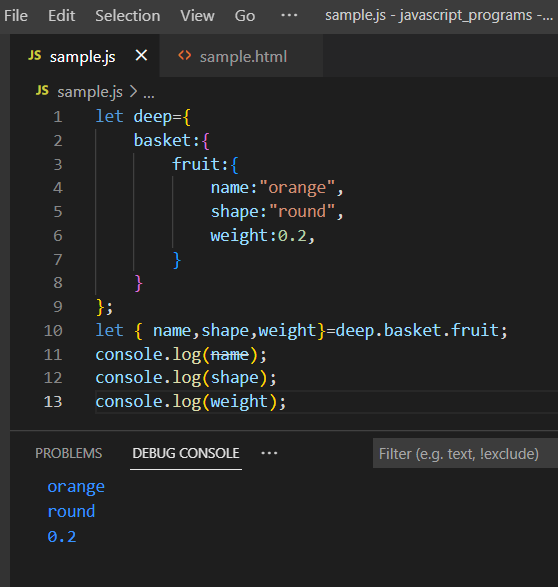
Destructuring assignment can be used to extract multiple items from arrays and objects and assign them to variables.

When var, let or const are not specified, var is assumed.



In this example, the first two lines of code destructure the array and assign the values to the variables a and b.

In the next set of lines, the first two values will be assigned to a and b and the remaining will be grouped as array itself.



The inner properties of the object can also be extracted.