

Lesson:

Function with an unlimited number of parameters



Topics Covered

1. Function with an unlimited number of parameters.
2. The arguments keyword.
3. Implementation.
4. Code Walk Through.

From the previous lectures, we are clear with the concept of function declarations, parameters, arguments, and function calls.

In this lecture, we will be looking at a situation where an unlimited number of parameters are passed as parameters to the function. As a developer, what are the options available to handle this situation?

Assume a condition where we want to find the sum of all the numbers passed as parameters to the function. In this case, we have no idea of the number of parameters. So, we cannot make use of what we have learned in previous lectures.

To do this we need to have an idea of the “arguments” keyword.

In JavaScript, the arguments keyword refers to an object that contains all the arguments passed to a function.

It is similar to an array, but not an actual array. We can make use of the “.length” method to know how many parameters are passed.

The arguments object is available within all function bodies and can be used to access the values of the arguments passed to the function.

Now to find out the sum of unlimited numbers passed parameter into the function we write.

```
JavaScript
// Function declaration

function sumOfAllParameters() {
  let sum = 0;
  for (var i = 0; i < arguments.length; i++) sum +=
arguments[i];
  return sum;
}

// Function call

let result = sumOfAllParameters(1, 2, 3, 4, 5);
console.log(result);

// OUTPUT: 15
```

The function “**sumOfAllParameters**” accepts an unlimited number of parameters. The function uses the arguments object to access all the parameters passed to it, regardless of the number of parameters. Inside the function, a for loop iterates through the arguments object, adding each value to a variable sum. The final value of the sum variable is returned as the result of the function.

When the function is called, multiple arguments are passed to the function, in this case, it will be 1, 2, 3, 4, 5, The

function will add all these numbers and return the total, which is 15, and this value gets stored in the variable `result` and then printed to the console using `console.log(result)`.

It is not recommended to use the `arguments` object, as it is not as efficient as declaring the function parameters.

