

Rest Operator And Spread Operator in Javascript



@_codevalley



Zuhaib Asif



@_codevalley



Zuhaib Asif

The rest and spread operators are two features in JavaScript that were introduced with ES6. These operators can make working with arrays and objects much more convenient.





What is Spread Operator?

The spread operator, represented by three dots (...), can be used to spread the contents of an array or object into a new array or object. This can be helpful when you want to create a copy of an array or object, or when you want to merge the properties of two or more objects.

Here's an example of how you can use the spread operator to create a copy of an array.



```
const originalArray = [1, 2, 3];  
const copyOfArray = [...originalArray];  
console.log(originalArray); // [1, 2, 3]  
console.log(copyOfArray); // [1, 2, 3]
```

In this example, we create a new array `copyOfArray` by spreading the contents of `originalArray` into it. We can see that `copyOfArray` is an exact copy of `originalArray`.



Example of Spread Operator?

Here's an example of how you can use the spread operator to merge two objects:



```
const object1 = { foo: 'bar' };  
const object2 = { baz: 'qux' };  
const mergedObject = { ...object1, ...object2  
}; console.log(mergedObject);  
//Output { foo: 'bar', baz: 'qux' }
```

In this example, we create a new object `mergedObject` by spreading the properties of `object1` and `object2` into it. We can see that `mergedObject` contains all the properties of `object1` and `object2`.



What is Rest Operator?

The rest operator, also represented by three dots (...), can be used to gather the remaining parameters in a function into an array. This can be helpful when you want to create a function that can accept a variable number of arguments.

Here's an example of how you can use the rest operator to gather arguments into an array:



```
function sum(...numbers) {  
  let total = 0;  
  for (let number of numbers) {  
    total += number;  
  }  
  return total;  
}  
  
console.log(sum(1, 2, 3)); // 6  
console.log(sum(4, 5, 6, 7)); // 22
```




Explanation of Previous Example:

In the previous example, we define a function `sum` that uses the rest operator to gather any number of arguments into an array called `numbers`. We then loop through `numbers` and add up all the values to get the total sum. We can see that we can call the `sum` function with any number of arguments, and it will still work correctly.

Conclusion:

In conclusion, the spread and rest operators in JavaScript can be incredibly useful for working with arrays and objects. The spread operator allows for easy copying, merging, and passing of arrays and objects, while the rest operator enables the creation of functions that can take a variable number of arguments. These operators can simplify code and improve readability, making them valuable tools for developers.