CS472 WAP

Rest and Spread

JavaScript Functional Programming

Maharishi International University Fairfield, Iowa © 2020

Main Point Preview

- Both Rest and Spread allow the programmer to treat multiple items as one
- *This lets us get things done more easily – do less and accomplish more.*

Good things to know

- No function overloading in JavaScript
  - Extra arguments ignored and missing ones ignored
- Arguments object and Rest (ES6) parameters
- Spread operator
- Destructuring assignments

Function Signature

- If a function is called with missing arguments(less than declared), the missing values are set to : `undefined`
- Extra arguments are ignored

```
function f(x) {
 console.log("x: " + x);
}
f(); //undefined
f(1); //1
f(2, 3); //2
```

No overloading!

```
function log() {
 console.log("No Arguments");
}
function log(x) {
 console.log("1 Argument: " + x);
}
function log(x, y) {
 console.log("2 Arguments: " + x + ", " + y);
}
log();
log(5);
log(6, 7);
```

- Why? JavaScript ignores extra arguments and uses undefined for missing arguments. Last declaration overwrites earlier ones.

`arguments` Object

The **arguments** object is an Array-like object corresponding to the arguments passed to a function.

```
function findMax() {
 let max = -Infinity;
 for (let i = 0; i < arguments.length; i++) {
   if (arguments[i] > max) {
   max = arguments[i];
   }
 }
 return max;
}
const max1 = findMax(1, 123, 500, 115, 66, 88);
const max2 = findMax(3, 6, 8);
```

Exercise: write a function that can be called with any number of arguments and returns the sum of the arguments.

Rest parameters (ES6)

- rest parameters are only the ones that haven't been given a separate name, while the arguments object contains all arguments passed to the function
  - ES6 compatible code, then rest parameters should be preferred.
  - Array vs array-like

```
function sum(x, y, ...more) {
 //"more" is array of all extra passed params
 let total = x + y;
 if (more.length > 0) {
  for (let i = 0; i < more.length; i++) {
  total += more[i];
  }
 }
 console.log("Total: " + total);
 return total;
}
sum(5, 5, 5);
sum(6, 6, 6, 6, 6);
```

Spread operator (ES6)

- The same … notation can be used to unpack iterable elements (array, string, object) where zero or more
  - arguments (for function calls) or
  - elements (for array literals) are expected

```
let a, b, c, d, e;
a = [1,2,3];
b = "dog";
c = [42, "cat"];

// Using the concat method.
d = a.concat(b, c); // [1, 2, 3, “dog”, 42, “cat”]

// Using the spread operator.
e = [...a, b, ...c]; // [1, 2, 3, “dog”, 42, “cat”]
```

Spread operator (ES6) (cont)

- The Rest/Spread Properties for ECMAScript proposal (ES2018) added spread properties to object literals. It copies own enumerable properties from a provided object onto a new object.

```
const obj1 = { foo: 'bar', x: 42 };
const obj2 = { foo: 'baz', y: 13 };

const clonedObj = { ...obj1 };
// Object { foo: "bar", x: 42 }

const mergedObj = { ...obj1, ...obj2 };
// Object { foo: "baz", x: 42, y: 13 }
```

Destructuring assignment (ES6)

copy elements/properties from array/object and assign to variables

```
const address = [221, 'Baker Street','London’];
const [ houseNo, , city ] = address;
console.log(houseNo, city)// 221 'London’

const details ={firstName:'Code', lastName:'Burst', age: 22};
const { firstName, age } = details;
console.log(firstName, age);// Code 22
```

JavaScript “strict” mode

```
(function() {
 "use strict";
 your code...
})();
```

- writing "use strict"; at the very top of your JS file turns on strict syntax checking:
  - helps identify common issues (or “bad” parts)
  - shows an error if you try to assign to an undeclared variable
  - eliminates some JavaScript silent errors by changing them to throw errors.
  - …
- You should always turn on strict mode for your code in this class
  - Important for ES6
- IIFE syntax will be discussed further in next lesson
  - Immediately Invoked function expression

Main Point

- Both Rest and Spread allow the programmer to treat multiple items as one
- *This lets us get things done more easily – do less and accomplish more.*


MAHARISHI
HIGHER EDUCATION
FOR HIGHER CONSCIOUSNESS
INTERNATIONAL UNIVERSITY