Example 1:

```
<script>
       function multiply(a, b = 1) {
  return a * b;
}

console.log(multiply(5, 2));
// expected output: 10

console.log(multiply(5));
// expected output: 5

        </script>
```

Example 2: Earlier parameters are available to later default parameters

```
<script>
function greet(name, greeting, message = greeting + ' ' + name) {
  return [name, greeting, message]
}

greet('David', 'Hi')                     // ["David", "Hi", "Hi David"]
greet('David', 'Hi', 'Happy Birthday!')  // ["David", "Hi", "Happy Birthday!"]
</script>
```

Example 3:

```
<script>
  function say(message='Hi') {
    console.log(message);
}
say(); // 'Hi'
say('Hello') // 'Hello'
 </script>
```

Example 4: multiple parameter and how to call function

```
<script>
 function createDiv(height = '100px', width = '100px', border = 'solid 1px red')
{
    let div = document.createElement('div');
    //The createElement() method creates an Element Node with the specified name
    div.style.height = height;
    div.style.width = width;
    div.style.border = border;
    document.body.appendChild(div);
    return div;
}
createDiv();
createDiv(undefined,undefined,'solid 5px blue');
 </script>
```

Example 5:

```
 function date(d = today()) {
    console.log(d);
}
function today() {
    return (new Date()).toLocaleDateString("en-US");
}
date();
 </script>
```

Explanation

The date() function takes one parameter whose default value is the returned value of the today() function. The today() function returns today's date in a specified string format.

When we declared the date() function, the today() function has not yet evaluated until we called the date() function.

Example 6: Using other parameters in default values

```
<script>
 function add(x = 1, y = x, z = x + y) {
    return x + y + z;
}

console.log(add()); // 4
 </script>
```

Explanation:

In the add() function:

The default value of the y is set to x parameter.
The default value of the z is the sum of x and y
The add() function returns the sum of x, y, and z.

Example 7:

You can use a return value of a function as a default value for a parameter. For example:

```
<script>
 let taxRate = () => 0.1;
let getPrice = function( price, tax = price * taxRate() ) {
    return price + tax;
}

let fullPrice = getPrice(100);
console.log(fullPrice); // 110
 </script>
```

Explanation

In the getPrice() function, we called the taxRate() function to get the tax rate and used this tax rate to calculate the tax amount from the price.

Example 8: The arguments object

The value of the arguments object inside the function is the number of actual arguments that you pass to the function. For example:

```
<script>  function add(x, y = 1, z = 2) {
    console.log( arguments.length );
    return x + y + z;
}

add(10); // 1
add(10, 20); // 2
add(10, 20, 30); // 3
 </script>
```

Example 9:

```
<script>
function sum(x = 3, y = 5) {
// return sum
return x + y;
}

console.log(sum(5, 15));  // 20
console.log(sum(7));        // 12
console.log(sum());          // 8
 </script>
```

Example 10: Passing Function Value as Default Value

```
<script>
// using a function in default value expression
const sum = () => 15;
const calculate = function( x, y = x * sum() ) {
    return x + y;
}
const result = calculate(10);
console.log(result);            // 160
 </script>
```

Explanation

In the above program,

10 is passed to the calculate() function.

x becomes 10, and y becomes 150 (the sum function returns 15).

The result will be 160.

Example 11:

```
<!DOCTYPE html>
<html>
<body>

<p>Setting a default value to a function parameter (y = 2).</p>
<p id="demo"></p>

<script>
function myFunction(x, y = 2) {
  return x * y;
}
document.getElementById("demo").innerHTML = myFunction(4);
</script>

</body>
</html>
```