

# 1월21일\_2

## [실습1]

파일명 : exercise2.html

1. 다음 기능을 처리하는 함수 sum()을 구현해 본다.
  - 매개변수를 한 개 선언한다.
  - 아규먼트가 아무값도 전달되지 않으면 리턴값 없이 리턴한다.
  - 1부터 아규먼트로 전달된 숫자값 까지 합을 구하여 리턴한다.
2. 0부터 5사이의 난수를 하나 추출하여 아규먼트로 전달하면서 sum() 함수를 호출하는데 0이 추출된 경우에는 아규먼트 없이 호출한다.

호출한 다음 리턴값이 있으면 다음 형식으로 브라우저의 도큐먼트 영역 <h2> 태그와 함께 출력하고

**호출 결과값 : XX**

리턴값이 없으면 다음과 같이 브라우저의 도큐먼트 영역에 <h3> 태그와 함께 출력한다.

**결과값이 없어요!!**

```
<!DOCTYPE html>
<html>
<head>
<meta charset="UTF-8">
<title>Insert title here</title>
</head>
<body>
<script>
function sum(n) {
    if (typeof n == "undefined")
        return;
    else {
        var hap = 0 ;
        for(var i=0; i <= n; i++)
            hap += i;}
        return hap;
    }
var n = Math.floor(Math.random()*6)
console.log(n)
var result1 = sum(n)
    if (n == 0)
        console.log(sum())
    if (result1)
        document.write("<h2>호출 결과값 : "+result1+"</h2>");
    else
        document.write("<h3>결과값이 없어요!</h3>");

</script>
```

```
<!DOCTYPE html>
<html lang="ko">
<head>
    <meta charset="UTF-8">
```

```

    <title>Title</title>
</head>
<body>
<script>
function sum(p) {
    if(p == undefined)
        return;
    else {
        var sump = 0;
        for(var n=1; n<=p; n++)
            sump += n;
        return sump;
    }
}

var num = Math.floor(Math.random()*6)
console.log(num)
var result;
if(num == 0)
    result = sum();
else
    result = sum(num);
if(result == undefined)
    document.write('<h3>결과값이 없어요!!</h3>');
else
    document.write('<h2>호출 결과값 :'+result+'</h2>');
</script>
</body>
</html>

```

```

<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <title>sum 함수 구현</title>
</head>
<body>
    <script>
        function sum(num)
        {
            var sum = 0;
            if(num==undefined)
                return num;

            for(var i=1; i<=num; i++)
                sum += i;

            return sum;
        }

        var rand = Math.floor(Math.random()*6);
        var result;
        if(rand==0)
            result = sum();
        else
            result = sum(rand);
    </script>

```

```

        if(result!==undefined)
            document.write("<h2>호출 결과값: " + result + "</h2>");
        else
            document.write("<h3>결과값이 없어요!</h3>");
    </script>
</body>
</html>

```

```

<!DOCTYPE html>
<html lang="ko">
<head>
    <meta charset="UTF-8">
    <title>Title</title>
</head>
<body>
<script>
    function sum(p) {
        if (typeof p == 'undefined')
            return;
        else {
            psum=0;
            for (var i=1; i<=p; i++)
                psum+=i;
            return psum;
        }
    }
    var num = Math.floor(Math.random()*6);
    var result;
    if (num == 0)
        result = sum();
    else
        result = sum(num);
    if(result == undefined)
        document.write('<h3>결과값이 없어요!!!</h3>');
    else
        document.write('<h2>호출 결과값 : ' + psum + '</h2>');
</script>
</body>
</html>

```

호출 결과값 : 10

## [실습2]

파일명 : exercise3.html

- 다음 기능을 처리하는 함수 calc() 를 구현해 본다.
  - 아규먼트의 개수에 제한이 없게 한다.
  - 아규먼트가 하나도 전달되지 않으면 0 을 리턴한다.
  - 전달된 아규먼트중 하나라도 숫자가 아니면 "숫자만 전달하세요" 라는 메시지를 리턴한다.
  - 전달된 값을 모두 더하여 리턴한다.
- 다음과 같이 다양하게 calc() 를 호출하여 결과를 <h3> 태그와 함께
   
 도큐먼트 영역에 출력한다.

calc(), calc(10, 20, '30'), calc(10, '가나다', 20), calc(1,2,3,4,5)

0

6000

숫자만 전달하세요

120

힌트 : isNaN() 사용하면 숫자 및 숫자문자열과 다른 문자열을 구분할 수 있음

```
<!DOCTYPE html>
<html>
<head>
<meta charset="UTF-8">
<title>Insert title here</title>
</head>
<body>
<h1>가변아규먼트 처리 함수 만들기</h1>
<hr>
<script>
function calc() {
    if (!arguments.length)
        return 0;
    else
        var multiply = 1;
        for(var i = 0; i < arguments.length; i++) {
            if (isNaN(arguments[i])){
                return "숫자만 전달하세요";
                break
            }
            else
                multiply *= arguments[i];
        }
        return multiply;
    }

    document.write("<h3>" + calc() + "</h3>");
    document.write("<h3>" + calc(10, 20, '30') + "</h3>");
    document.write("<h3>" + calc(10, '가나다', 20) + "</h3>");
    document.write("<h3>" + calc(1,2,3,4,5) + "</h3>");
</script>
</body>
</html>
```

## 가변아규먼트 처리 함수 만들기

0

6000

숫자만 전달하세요

120

