JavaScript Functions

- description : JavaScript Functions
- author : 오션
- email : shlim@repia.com
- lastupdate : 2021-05-20

The Source of this article

JavaScript Functions

JavaScript Function Syntax

JavaScript 함수는 특정 작업을 수행하도록 설계된 코드 블록(a block of code)입니다.

JavaScript 함수는 " 무언가 " 가 불러올(호출할) 때 실행됩니다.

```
function myFunction(p1, p2) {
  return p1 * p2;   // The function returns the product of p1 and p2
}
```

JavaScript Function Syntax

JavaScript 함수는 `function` 키워드, 이름, 괄호 () 순서로 정의합니다.

함수 이름에는 문자, 숫자, 밑줄 및 달러 기호 (변수와 동일한 규칙)가 포함될 수 있습니다.

괄호에는 쉼표로 구분된 매개변수(parameter) 이름이 포함될 수 있습니다.
(매개 변수 1, 매개 변수 2, ⋯)

함수에 의해 실행될 코드는 중괄호 안에 배치됩니다:{}

```
function name(parameter1, parameter2, parameter3) {
  // code to be executed
}
```

함수 **매개변수(parameter)**는 함수 정의에서 괄호 () 안에 나열됩니다.

함수 **인수(arguments)**는 함수가 호출될 때 함수가 받는 **값(value)**입니다.

함수 내에서, 인수(매개 변수)는 지역 변수(local variables)로 작동합니다.

함수는 다른 프로그래밍 언어에서 프로시저(procedure) 또는 서브루틴(Subroutine)과 거의 동일합니다.

Funcion Invocation

함수 내부의 코드는 " 어떤 것 " 이 함수를 불러올(호출할) 때 실행됩니다.

- 이벤트 발생 시 (사용자가 버튼 클릭 시)
- JavaScript 코드에서 호출 (호출)되는 경우
- 자동 (자체 호출)

이 튜토리얼의 뒷부분에서 함수 호출에 대해 더 많이 배울 것입니다.

+Function Return

JavaScript가 `return` 문(statement)에 도달하면, 함수 실행이 중지됩니다.

함수가 명령문(statement)에서 호출된 경우, JavaScript는 호출 명령문 다음에 코드를 실행하기 위해 " 반환 " 합니다.

함수는 종종 **반환 값(return value)**을 계산합니다. 반환 값은 " 호출자(caller)"에게 다시 " 반환 " 됩니다.

Example

두 숫자의 곱(product, 곱셈)을 계산하고 결과를 반환합니다:

```
var x = myFunction(4, 3);  // Function is called, return value will end up
in x

function myFunction(a, b) {
  return a * b;           // Function returns the product of a and b
}
```

Why Functions?

코드를 재사용 할 수 있습니다. 코드를 한 번 정의하고 여러 번 사용할 수 있습니다.

다른 인수로 동일한 코드를 여러 번 사용하여 다른 결과를 생성 할 수 있습니다.

Example

화씨(Fahrenheit)를 섭씨(Celsius)로 변환 :

```
function toCelsius(fahrenheit) {
  return (5/9) * (fahrenheit-32);
}
document.getElementById("demo").innerHTML = toCelsius(77);  // 25
```

The () Operator Invokes the Function

위의 예제에서 `toCelsius`는 함수 객체를, `toCelsius()`는 함수 결과를 나타냅니다.

()없이 함수에 액세스하면 함수 결과 대신 함수 객체가 반환됩니다.

Example

```
function toCelsius(f) {
  return (5 / 9) * (f - 32);
}
  document.getElementById("demo").innerHTML = toCelsius;
  /* function toCelsius(f) { return (5 / 9) * (f - 32); } */
```

+Functions Used as Variable Values

함수는 모든 유형의 공식(formulas), 할당(assignments) 및 계산(calculations)에서 변수를 사용하는 것과 동일한 방식으로 사용할 수 있습니다.

Example

다음과 같이 함수의 반환 값을 저장하기 위해 변수를 사용하는 대신 :

```
let x = toCelsius(77);
let text = "The temperature is " + x + " Celsius";
```

아래와 같이 함수를 변수 값으로 직접 사용할 수 있습니다.

```
document.getElementById("demo").innerHTML = "The temperature is " +
toCelsius(77) + " Celsius";
// The temperature is 25 Celsius

function toCelsius(fahrenheit) {
  return (5 / 9) * (fahrenheit - 32);
}
```

Local Variables

JavaScript 함수 내에서 선언된 변수는 함수에 대해 **LOCAL**이 됩니다.

지역 변수(Loval variables)는 함수 내에서만 액세스 할 수 있습니다.

Example

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

  <h2>JavaScript Functions</h2>

  <p>Outside myFunction() carName is undefined.</p>

  <p id="demo1"></p> <!-- string Volvo -->

  <p id="demo2"></p> <!-- undefined 정의되지 않은 값 -->

  <script>
    myFunction();

    function myFunction() {
      let carName = "Volvo";    // local variable
      document.getElementById("demo1").innerHTML = typeof carName + " " +
carName;
    }

    document.getElementById("demo2").innerHTML = typeof carName;
  </script>

</body>
</html>
```

지역 변수(local variable)는 함수 내에서만 인식되기 때문에, 같은 이름의 변수를 다른 함수에서 사용할 수 있습니다.

지역 변수는 함수가 시작될 때 생성되고 함수가 완료되면 삭제됩니다.

오션, Javascript, Functions

From:
http://125.132.25.164/dokuwiki/ - **문제를 잘 정의하는 것은 문제를 절반 해결한 것이다. - 2023.12**

Permanent link:
http://125.132.25.164/dokuwiki/doku.php?id=wiki:javascript:javascript_note:js_functions&rev=1621472939

Last update: **2022/03/10 19:52**