

5. 함수

- (1) **익명 함수**: 이름이 없는 함수이며, 변수에 넣어 사용한다.
- (2) **선언적 함수**: 함수명을 선언한다.

<함수의 재정의 - 익명함수와 선언적 함수의 차이>

익명함수: 정의하기 전에는 호출 불가	선언적 함수: 정의하기 전에 선언 가능
<pre>var funVar = function () { alert('함수 A'); }; funVar(); // 선언 후 호출</pre>	<pre>funVar(); // 선언 전 호출 가능 function funVar () { alert('함수 A'); }; funVar(); // 선언 후 호출도 가능</pre>

- (3) **매개변수(인자, 인수)**: JavaScript는 지정한 매개변수보다 많거나 적은 매개변수 사용을 허용함
 - 선언된 매개변수보다 많은 매개변수가 들어갈 경우, 뒷부분은 무시된다.
 - 선언된 매개변수보다 적은 매개변수가 들어갈 경우, 빈 자리에 undefined가 들어간다.

(4) 리턴값

- 리턴값이 없는 경우, 매개변수가 undefined로 인식된다.

(5) 가변 인자 함수

- 매개변수의 개수가 변할 수 있는 함수 ex) Array()
- 매개변수를 선언된 형태와 다르게 사용했을 때, 매개변수를 모두 활용할 수 있는 함수

```
<script>
var array = function () {
    var result = [];
    var length = arguments.length; // 예약어 arguments:
    if (length == 0) {
        result = [];
    } else if (length == 1) {
        // 매개변수 수만큼 result.push(); 수행
        for (var i = 0; i < arguments[0]; i++) {
            result.push('');
        }
    } else if (length >= 2)
        for (var i = 0; i < arguments.length; i++) {
            result.push(arguments[i]);
        }
    }
    return result;
};
</script>
```