< 자바스크립트의 함수 >

※ 자바스크립트는 기본적으로 심볼이 같을 경우 오류가 나는게 아니라 덮어 씌워 지기 때문에 해당 부분에 유의할 필요가 있을 듯 하다.

자바스크립트의 함수는 크게 ‘익명 함수’와 ‘선언적 함수’로 나뉜다. 익명 함수와 선언적 함수를 구분하는 기준은 ‘함수 명’ 의 유무이다.

function () {} // 익명 함수 (일반적으로 변수에 넣어서 사용)

function <함수 이름> () {} // 선언적 함수

함수가 하나의 자료 형이라는 의미는?

var a = function () {};

console.log(typeof a); // function

이런 의미이다.

< 실행 순서 >

웹 브라우저는script태그 내부에서 선언적 함수부터 읽는다. 모든 선언적 함수를 인식하고 난 뒤 태그 내부의 내용을 한 줄씩 실행한다. 따라서 선언적 함수의 호출은 함수의 정의의 순서에 의존하지 않는다.

< 리턴 값 >

함수 내부에서 리턴이 가능. 다른 언어처럼 리턴 타입이 명시되어 있지 않음.

< 매개 변수 >

자바스크립트는 매개변수에 대해 관대하다. 정해진 매개변수보다 많이 입력된 경우 추가된 매개변수는 무시하고, 정해진 매개변수보다 적게 입력된 경우 지정하지 않은 매개변수는 undefined로 처리한다. 사실 중요한 것은 가변 인자 함수이다.

< 가변 인자 함수 >

대표적인 가변 인자 함수는 Array() 이다. 모든 함수는 기본적으로 함수 내부에 arguments라는 변수를 가지고 있는데, 이 변수는 매개변수의 배열을 의미한다.

< 내부 함수 >

함수 내부에 함수를 정의할 수 있다. 해당 함수의 범위는 자신이 속해있는 함수의 내부이다.

< 자기 호출 함수 >

(Function() {})() // 함수를 생성 하자마자 호출

< 콜백 함수 >

별도의 문서에서 정리

< 클로저 >

함수 안의 지역 변수를 함수 외부에서 사용할 수 있는 것

< 내장 함수 >

setTimeout(function, millisecond) // 일정 시간 후 함수를 한 번 실행

setInterval(function, millisecond) // 일정 시간마다 함수를 반복해서 실행

clearTimeout(id) // setTimeout() 을 중지

clearInterval(id) // setInterval() 을 중지

< 코드 실행 함수 >

eval(string) : 문자열을 코드로 실행할 수 있는 특별한 함수