

기본 문법

▶ 주석 처리

주석 종류		설명
자바스크립트 주석	한 줄 주석	// 로 시작
	여러 줄 주석	/*로 시작하고 */로 끝남
HTML주석		<!--로 시작하고 -->로 끝남 자바스크립트나 스타일시트 태그 이외의 HTML태그에서는 이 주석 태그를 사용해서 주석 처리를 해주어야 함

▶ 변수

✓ 변수 선언

- 변수 종류 : 멤버 변수와 지역 변수
- 멤버 변수 : 전역 변수, 기본적으로 window객체의 멤버 변수
변수에 대한 자료형은 있으나 선언하지는 않음

형식	설명
변수명 = 값;	window.변수 명 또는 this.변수명과 같은 의미 선언 시 변수 명에 var를 붙이지 않으면 전역변수로 간주
var 변수명 = 값;	변수 선언 시 변수명 앞에 var를 붙이면 지역변수

▶ 변수

✓ 명명 규칙

- 영어 대/소문자, 숫자, _, \$ 사용 가능
- 첫 글자 숫자 사용 불가
- 예약어 사용 불가
- 한글 사용 가능
- 생성자 함수는 항상 대문자로 시작
- 변수, 인스턴스, 함수, 메소드는 항상 소문자로 시작
- 이름에 의미 있는 단어의 조합 권장
- 두 단어 결합 시 낙타봉 표기법 권장

▶ 변수

✓ typeof()

값의 자료형을 확인하는 연산자

* 선언 시 자료형을 지정하지 않아 변수 명을 보고 데이터 확인 불가
자료형 확인 시 자주 사용

typeof("문자열값") 또는 typeof('문자열값') → string

typeof(숫자) → number

typeof(참/거짓) → boolean

typeof(객체) → object

typeof(초기값이 없는 변수) → undefined

typeof(function) → function

▶ 자료형

✓ 문자열(String)

"" , ""로 묶여있는 리터럴

내장 객체로 String객체, 기본적인 메소드 존재

메소드	내용
toUpperCase()	모든 문자 대문자로 변환
toLowerCase()	모든 문자 소문자로 변환
length	글자 개수 조회용 멤버변수
indexOf()	찾는 문자의 순번(위치) 리턴
lastIndexOf()	뒤에서부터 찾는 문자의 순번 리턴
charAt()	찾는 위치의 문자 리턴
substring()	값을 일부분만 리턴
split()	토큰 문자로 분리한 문자열 배열 리턴

▶ 자료형

✓ 숫자(number)

정수형 숫자와 부동소수점 숫자로 구분

내장 객체로 Math객체 제공, 기본 메소드 존재

메소드	내용
Math.abs()	절대값 리턴
Math.random()	임의의 난수발생 리턴(소수점)
Math.round()	반올림처리 후 리턴
Math.floor()	부동소수점 숫자를 정수로 리턴(소수점 자리버림)
Math.ceil()	소수점 자리에서 무조건 올림

* NaN(Not a Number) : 숫자가 아닌 데이터를 숫자처럼 사용할 때 출력

▶ 자료형

✓ 기타 자료형

- 논리값(Boolean)

true, false 두 가지 값을 가짐

- 객체(Object)

new로 선언된 사용자 객체와 자바스크립트 내장 객체

- undefined

변수 명이나 함수 명으로 선언되지 않은 식별자일 때 지정

- 함수(function)

함수(메소드)를 가지는 자료형

▶ 데이터 형변환

✓ 숫자 → 문자열

숫자와 문자를 +연산하게 되면 문자가 우선되어 숫자를 문자로 변환
강제 형변환 : String()함수 이용

✓ 문자열 → 숫자

숫자, 문자 + 이외의 사칙 연산 시 숫자가 우선돼 문자를 숫자로 변환
강제 형변환 : Number(), parseInt(), parseFloat()함수 이용

* parseInt()함수는 인자가 2개로 문자열, radix(해당진수)를 선택할 수 있음

▶ 연산자

연산자 종류	연산자
최우선 연산자	() , [] , .
단항 연산자	++, --, + sign, - sign
산술 연산자	+, -, *, /, %
관계 연산자	>, <, >=, <=, ==, !=, ===, !==
논리 연산자	&&,
대입 연산자	=
복합대입연산자	+=, -=, *=, /=, %=
삼항 연산자	? : ;

* 연산자 우선순위 : 최우선 > 단항 > 산술 > 관계 > 논리 > 삼항 > 대입

▶ 제어문

✓ 조건문

if, if~else, if~else if~else, switch문, 짧은 조건문(&&, ||)

✓ 반복문

for, while, do~while, for in문

✓ 분기문

continue, break문