가상 클래스 선택자(Pseudo Classes Selector)

요소가 특정 상태를 만족해야 함

동적 가상 클래스와 구조적 가상클래스 있음

동적 가상 클래스(Dynamic Pseudo classes)

어떤 상태나 조건이 발생할 때, 사용자의 액션에 따라 스타일이 바뀌는 선택자임

사용자와의 인터랙션이 필요한 부분에서 자주 사용

x : hover(맴돌다)

선택자 x 요소에 마우스 커서가 올라가 있는 동안 선택

x:active

선택자 x 요소에 마우스를 클릭하고 있는 동안 선택

x:focus

선택자x 요소가 포커스되면 선택(클릭하는 순간)

x:visited

사용자가 이미 방문한 링크 표시

x: disabled

비활성화된 요소 선택

x:enabled

활성화된 요소 선택

x:checked

체크된 요소 선택(ex.체크박스)

구조적 가상 클래스(structural Pseudo classes)

x : first class

선택자 x가 형제요소 중 첫 번째면 선택

x : last class

선택자 x가 형제요소 중 마지막면 선택

x : nth-child(n)

선택자 x가 형제 요소 중 (n)번째라면 선택

n 키워드 사용시 0부터 해석이 됨(zero-base)

n을 통해 다양한 조건 선택 가능

ex. 2n : 짝수 ,2n+1 : 홀수 ,n+3 : 3번째 부터 선택

가상 요소 선택자 (Pseudo-Elements classes)

내용의 일부만 선택해 스타일을 적용

가상 클래스와 구분 위해 ::로 표시

x :: before와 after

선택자 x 요소의 앞뒤에 내용을 삽입

속성 선택자

특정 속성값을 가지고 있는 요소 선택

[x]

속성x를 포함한 요소 선택

[x=”OOO”]

속성x를 포함하고 값이 OOO인 요소 선택

ex.[disabled] {color :red;}

[type="password"]{color:orange;}

JavaScript

웹브라우저에서 사용하기 위하여 만들어진 프로그래밍 언어

웹 브라우저 상 UI를 동적으로 보여주기 위해 사용

Node.js 런타임 통하여 서버 쪽에서도 사용할 수 있게 됨

주석은 Java와 같음

; 안 붙여도 됨

변수 선언

let XXX : 변수 생성

생성 후 초기화 안 해도 됨

const : 변수의 값이 바뀌지 않을 때 사용

자료형

Boolean, Null, Undefined, Number,String

Undefined

JS에선 변수 선언 안 하면 지정되는 값

Null은 의도적인 부재를 의미

Number

정수 형식을 따로 제공하지 않음

String

‘’ 또는 “”로 선언

템플릿 문자열

내장된 표현식을 허용하는 문자열 리터럴

ES6 이후 사용 가능

이전

var a = 안녕하세요.

var b = 환영힙니다

var str = a+”이젠학원입니다.” +b+”.”;

이후

let a = 안녕하세요

let b = 환영합니다

var str = `${a} 이젠학원입니다. ${b}.`

백틱(``)으로 구분

${} 안의 표현식을 평가한 다음, 필요하면 문자열로 변환되고 템플릿으로 합쳐짐

${....}표현식 안에 템플릿 리터럴을 중첩할 수 있음

객체

이름 /값 또는 프로퍼티의 집합

공개 데이터만 포함, 캡슐화(공개+비공개) 불가능, 동작 포함할 수 없음

자바스크립트 객체는 특정 클래스의 인스턴스가 아님

기존 프로퍼티 값을 바꾸거나 추가 가능

const choi={first:’choi’ , last=’hee’}

choi={first:’kimi’ , last=’hee’} <= 불가능

cho.first=’kim’ <=가능

delete 연산자로 프로퍼티 삭제함

존재하지 않는 프로퍼티에 접근하면 undefined가 반환됨

숫자는 프로퍼티의 이름으로 가질 수 없음 => 배열이용

배열

배열의 요소는 각자 다른 형식을 가질 수 있음

배열은 값이 비어있어도 됨 ex) const array=[1,,3,4,]

배열 끝에 새 요소를 추가할 수 있음

마지막 쉼표는 요소가 누락되었다는 의미가 아님

추후에 length 확장이 가능함을 의미

모든 객체에서 존재하지 않는 프로퍼티는 undefined 값을 갖음

Array.isArray() : 객체가 배열인지 확인

산술 연산

대부분 Java와 같음

/는 연산자와 피연산자가 정수라도 부동소수점의 수를 결과로 반환

\*\* 제곱

문자열\*수 =문자열

DDL

VALIDATE

DISABLE NOVALIDATE CONSTRAINT (제약조건명)

제약 조건이 없는 것처럼 작동

ENABLE NOVALIDATE CONSTRAINT (제약조건명)

ENABLE 이후의 데이터만 제약조건 적용

VALIDATE

DISABLE VALIDATE CONSTRAINT (제약조건명)

테이블 변경X(INSERT,UPDATE,DELETE)

ENABLE VALIDATE CONSTRAINT (제약조건명)

이전, 신규 모두 검사