**Web2 – JavaScript**

**JavaScript 사용 이유**

동적

**html 태그**

1. <script> </script>

<script>

···

</script>

···의 내용은 JavaScript로 해석하도록 하는 태그

이 뒤로 나오는 JavaScript 문법은 전부 ···에 들어가는 내용

1. Event

onclick, onchange, onkeydown 등

예를 들어 아래와 같이 사용

<input type="button" value="hi" onclick="alert('hi')">

<input type="text" onchange="alert('changed')">

<input type="text" onkeydown="alert('key down')">

1. Console

웹페이지 마우스 우클릭 > 검사 > console

Element와 console을 같이 띄우고 싶다면 console 누르는 대신 element 누르고 esc키(다시 누르면 원래대로)

파일을 만들지 않고도 JavaScript를 즉석으로 실행

예시로 긴 문장의 글자수를 세 알림창에 출력하려면

alert(‘~’.length)

shift + enter : 줄 바꿈

모든 a 태그 가져와서 배열에 넣고

console.log를 이용해서 원하는 원소 출력

var alist = document.querySelectorAll(‘a’);

console.log(alist[0]);

**JavaScript 문법 – 18강부터**

1. 중복의 제거

this

target

예시)

if(document.querySelector('#night\_day').value === 'night'){

document.querySelector('body').style.backgroundColor='black';

document.querySelector('body').style.color='white';

document.querySelector('#night\_day').value = 'day';

}else{

document.querySelector('body').style.backgroundColor='white';

document.querySelector('body').style.color='black';

document.querySelector('#night\_day').value = 'night';

}

에서

var target = document.querySelector('body');

if(this.value === 'night'){

target.style.backgroundColor='black';

target.style.color='white';

this.value = 'day';

}else{

target.style.backgroundColor='white';

target.style.color='black';

this.value = 'night';

}

1. 배열 기본 문법

배열 생성 :

var coworkers = ["egoing", "leezche"];

인덱스 0인 배열 출력 :

document.write(coworkers[0]);

배열 요소 추가 :

coworkers.push("duru");

배열 요소 개수 출력 :

document.write(coworkers.length);

1. 반복문 기본 문법

while

1. 배열 & 반복문

ex8.html 파일 참조

1. document.querySelectorAll
2. 함수 function

예를 들어

function nightDayHandler(self){}

이런 식으로 사용

ex9.html 파일 참조

1. 표현식

1 + 1 은 숫자 2에 대한 표현식

1. - 1 은 숫자 1에 대한 표현식

1 === 1 은 true에 대한 표현식

this, parameter(self)

함수를 쓰면 this가 의미하는 것이 달라질 수 있음.

따라서 함수에서는 parameter를 self로 두고 안에 들어있는 코드의 this를 전부 self로 바꾼 다음

함수 부분으로 들어간 코드가 있던 곳에 함수이름(this);

이런 방식으로 정리를 한다.

1. 객체

함수 이름이 겹치는 것을 피하기 위해서 함수 이름을 길게 설정하는 것처럼 번거로운 일을 피하고 쉽게 정리정돈 할 수 있도록 하는 것이 객체

예를 들어

Body.setBackgroundColor('white');

Body.setColor('black');

Links.setColor('blue');

method

객체에 속해 있는 함수

예를 들어

Body.setBackgroundColor('white');

에서 setBackgroundColor는 Body 객체의 메소드

document.querySelector('body').style.color = color;

에서 querySelector는 document 객체의 메소드

배열이 정보를 순서대로 저장하는 그릇이라면

객체는 정보를 순서에 상관없이 저장하는 그릇

배열은 대괄호, 객체는 중괄호

객체 생성 방법

var coworkers = {

"programmer" : "egoing",

"designer" : "leezche"

};

객체 추가

coworkers.bookkeeper = "duru";

만약 띄어쓰기가 필요하다면

coworkers["data scientist"] = "taeho";

출력

document.write("programmer : " + coworkers.programmer + "<br>");

document.write("data scientist : " + coworkers["data scientist"] + "<br>");

객체에 있는 모든 데이터를 가져오기

for(var key in coworkers){

document.write(key+' : ' + coworkers[key] +'<br>');

이렇게 하면 key는 programmer, designer, bookkeeper, data scientist이고 coworkers[key]를 하면 egoing, leezche, duru, taeho가져올 수 있다.

객체에는 문자, 숫자, 배열, 함수도 담을 수 있음

객체에 함수 담기

예를 들어 coworkers라는 객체에 showAll이라는 함수 넣기

coworkers.showAll = function(){

for(var key in coworkers){

document.write(key+' : ' + coworkers[key] +'<br>');

}

}

사용은

coworkers.showAll();

이렇게

다만 이렇게 하면 coworkers라는 객체 이름이 바뀌면 수정을 해야 하므로 this 사용

coworkers.showAll = function(){

for(var key in this){

document.write(key+' : ' + this[key] +'<br>');

}

}

이렇게 할 경우 showAll도 coworkers에 포함되어있어 같이 출력됨

객체에 소속된 함수는 method

객체에 소속된 변수는 property

객체의 property와 property를 구분할 때는 ,가 필요

var Body = {

setColor : function(color){

document.querySelector('body').style.color = color;

},

SetBackgroundColor : function(color){

document.querySelector('body').style.backgroundColor = color;

}

}

1. 파일로 묶어서 그룹핑
2. 라이브러리와 프레임워크

Library : 내가 만들고 있는 프로그램에 사용할 부품을 가져오는 느낌

ex) jQuery :

<script

src="https://ajax.googleapis.com/ajax/libs/jquery/3.6.0/jquery.min.js">

</script>

라고 써서 jQuery를 사용할 수 있도록 한 뒤

예를 들어

var alist = document.querySelectorAll('a');

var i = 0;

while(i < alist.length){

alist[i].style.color = color;

i = i + 1;

}

이런 반복문을 $()함수를 사용해서 표현 가능

$('a').css('color', color);

Framework : 만들고자 하는 것이 무엇인가에 따라 공통인 부분을 framework에 작업하고 각자 다른 부분만 개별 작업

1. UI & API

어플리케이션을 만들기 위해서 프로그래밍을 할 때 사용하는 조작 장치들을 API라고 말한다.

모든 어플리케이션은 API를 프로그래밍적으로, 즉 순서대로 실행하는 것을 통해 만들어진다.

ex)alert

1. 추천 검색어

어떤 태그 삭제, 자식태그 추가 -> document 객체로, 이에 없다면 더 넓은 범위인 DOM 객체 검색 (document 객체는 DOM 객체의 일부)

웹페이지가 아니라 웹브라우저 자체를 제어 -> window 객체 조사

웹페이지를 리로드하지 않고 정보 변경 -> ajax 공부

웹페이지가 리로드되어도 현재 상태 유지 -> cookie 공부

(사용자를 위한 개인화된 서비스 제공)

인터넷이 끊겨도 동작하는 웹 페이지 -> offline web application 공부

화상 통신 웹 앱 -> webRTC

사용자의 음성을 인식하고 음성으로 정보 전달 -> speech

3차원 그래픽으로 게임과 같은 것 만들기 -> webGL

가상현실 -> webVR