1. Java script( 내용을 기록하는 언어 ) and the web

① HTML : 브라우저에 내용을 보이고 싶을 때 어떻게 보여줄 것인가 말해주는 언어

    ⤷ viewing

② javascript : dynamic and interactive

: 문서의 contents, form , animation 등을 바꿀 수 있다.

③ scripting( 기록하는 ) language

: 웹 브라우저에 담겨있는 interpreter에 의해 실행된다.

④ Interpreter

: interpret = compile

컴퓨터가 알아들을 수 있는 말로 바꾸는 작업

.cpp → .obj로 바뀌는 작업이 컴파일

.obj → .exe(최종)로 바뀌는 작업이 빌드

interpret은 속도가 오래 걸리기에 자주 사용하지는 않는다.

java script는 속도가 빠를 필요가 없기에 interpret을 사용한다.

각각의 code를 실행할 수 있는 방식으로 변환한다.

⑤ HTML and JavaScript : client가 요청하면 html파일을 해석해서 화면에 보여준다.

       ⤷ html은 script를 안에 품고 있다.

⑥ JavaScript( 대표적인 언어 )

ex )  C, Java, PHP, etc.

2. <script> tag : can be placed either within the <head> or <body>

① external javascript : <script src=“your\_source.js( 파일 이름 )”></script>

\* script tag를 사용하여 가져오기 때문에 <script> elelment 는 중복되면 안된다.

② internal javascript : <script type=“text/javascript”> … </script>

③ <script> attributes

3. semi – colon ;

① Semi‐colon (;) at the end of a line is not mandatory in JavaScript ( 하지만 쓰자 )

② 두 개 이상의 말을 한 줄에 적고자 할 때 ;이 필요하다.

ex ) x=10; y=5; z=0;

4. comments

① // Single‐line comment

② /\* Multi‐line comment \*/

5. Debugging JavaScript -  나의 잘못을 고쳐내는 작업이 디버깅

6. pros and cons ( 찬반 )

: pros

- 웹사이트의 ui를 개선

- 사람들이 웹사이트를 좀 더 쉽게 볼 수 있도록 한다.

- 팝업 창을 만들기가 쉽다.

- 입력하지 않아도 자동적으로 바꾸어준다.

- form을 validate한다.

: cons

- 파일이나 데이터베이스에 접속할 수 없다.

- cannot do networking

7. javascript vs java(applet)

① javascript

- html안에 포함되어 있다.

- 웹페이지에 특화되어있다.

- java(applet)보다 간단하다.

- interpret 방식으로 번역을 한다.

- client에게서 실행이 된다. ( client에서 번역하고 해석 )

② java(applet)

- 다운로드 받아야 실행이 된다.

- 범용적이다.

- 컴파일러 방식으로 번역을 한다.

- client에게서 실행이 된다. ( client에서 번역하고 해석 )

※ document.write( )의 “ ” 안에는 html 코드를 적어준다.

8. JavaScript Variables

① using variables

- Declare and assign a value ( 초기값을 설정해주어야 한다 )

- Keyword varis not mandatory

② Variable names

- Use a‐z, A‐Z, 0‐9, $, \_

- Spaces, symbols are not allowed

③ Variables are case‐sensitive ( 대/소문자를 구별한다 )

9. Data types

① java script는 ‘int’, ‘float’과 같이 data type을 선언할 필요가 없다.

\* special data types

- NULL : 비어있는 상태이지만 null이라는 값이 지정된 상태

- Undefined : 변수는 만들어져있으나 값이 지정되지 않은 상태

② Boolean type : 값이 논리적으로 참인지 거짓인지 판별한다.

③ String type

\* javascript는 c언어와 다르게 하나의 문자만 저장하는 char가 없다.

④ Object

- 어떠한 기능을 가진 것을 의미한다.

- 모든 object는 property를 가진다.

ex ) 자동차가 object라면 색,차종 등이 property

⑤ Array object

- 가상의 object = "new"

- [5]가 아닌 (5)로 표시

- javjascript는 array의 length를 지정하지 않아도 된다.

⑥ typeof operator

- 지정된 변수의 type이 무엇인지 알게 해준다.

⑦ special characters

10. Expression

① Arithmetic operators

- binary

-unary

ex ) ++count , count++

② Assignment operators

③ comparison operators

ex)

var currentword = “75”;

var currentvalue = 75;

var outcome1 = ( currentword == currentvalue ) ;

var outcome2 = ( currentword === currentvalue ) ;

outcome1 = true ( 두 개의 모양이 같기 때문에 )

outcome2 = false ( 두 개의 모양은 같으나 의미가 다르기 때문에 )

\* ===이 c언어에서 ==와 동일하다.

④ Logical operators

ex ) && , || , !

⑤ string concatenation operator : +

“one” + “two” + “three” → “one two three”

⑥ string assignment operater : (+=)

var onestring = “one”;

onestring += “two three”;

we get onestring = “one two three”

11. string object

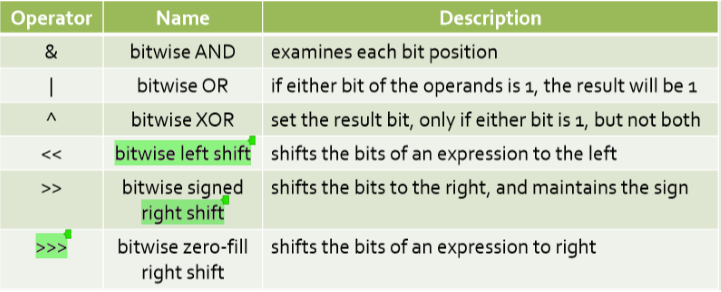
① “ ”안에 존재하면 더 이상 태그가 아니다. ex ) “<p>”는 하나의 문자가 된다.

② property와 method가 존재한다.

③ length : string의 길이 / substr() : extracts a substring

     ⤷ property              ⤷ method

12. bitwise operators



13. operator precedence

14. Built in Functions ( java에서 기본적으로 제공하는 함수 )

① alert() method

② confirm() method : 사용자가 확인과 취소를 선택하도록 알림을 띄워

③ prompt() method :  사용자가 답을 적을 수 있도록 해, 초기값을 지정가능

④ console.log() method : 브라우저가 아닌 디버깅 용으로 콘솔 창에 띄워

15. “ if ” statement

① if(조건){문장;}의 형식

② if는 소문자로만 적는다.

③ 조건은 논리적으로 참/거짓 값이어야 한다.

④ 조건이 참이면 문장이 실행이 된다.

\*

16. “switch/case” statement

ex ) switch(expression){

        case label1 :

           statements-1;break;

        ....

        defalult:

           statements;

}

17. “for” statement

ex ) for(i=0;i<5;i++){

       statements;

     }

18. “for/in” statement

ex ) for(counter in object) {

       statements;

     }

\* 개수를 다 할 때까지 반복문을 실행하는 데 보통은 object의 길이까지 진행한다.

\* array를 이용하여 반복의 횟수를 지정하게 된다.

19. “while” statement

ex ) while(condition){

        statement;

        increment/decrement\_statement;

     }

20. “do/while” statement

ex ) do{

       statements;

       increment/decrement\_statement;  → do는 무조건 실행

     } while(condition);

21. Functions

\* 함수는 name / parameters / return value가 필요하다

① User-defined function( 함수를 정의 ) \* 정의는 head안에 위치

② Function call \* function call은 body안에 위치

22. Built-in global Functions(이미 만들어진 함수)

① isNAN(value) : 숫자가 아니다

숫자면 false, 숫자가 아니면 true!

② parseInt(string,radix) : 문자를 숫자(정수)로 바꿔준다.

\* radix = 진수 (ex)16진수 - 첨두어 ‘ox’를 두기도 한다, 2진수, 10진수(기본값)

\* 10.33의 경우 10으로 반환

\* 맨 처음 수만 정수로 변환해준다.

\* 맨 처음에 수가 존재하지 않는 경우 NAN이 출력된다.

③ parseFloat(object)

④ Number(object) : object의 value들을 숫자로 바꿔준다.

⑤ String(object)

23. string의 여러 property

① string.length : 문자열의 길이 출력

② string.charAt(n) : 문자열을 배열로 저장 후 n번째에 해당하는 문자 출력

③ string.concat(string1,string2) : 두 개의 문자열을 합쳐서 출력

④ string.indexOf(substring) : 내가 입력한 문자가 가장 처음 나오는 위치를 출력

\* 해당하는 문자가 없을 경우 –1 출력

⑤ string.substr(start,length) : 내가 입력한 시작 위치부터 해당하는 길이만큼 출력

⑥ string.toUppercase() : 입력한 문자열을 대문자로 출력

⑦ string.toLowercase() : 입력한 문자열을 소문자로 출력

24. Object overview

① OOP(object oriented programming) : object를 가지고 프로그래밍을 짜는 것

\* c언어에서는 예약어를 가지고 프로그래밍을 한다. object는 그것의 상위개념

( 즉, 무엇인가를 만들기 위해 도구들을 가지고 있는 집합체 같은 개념 )

- property(변수로 지정된 어떤 값)와 method(기능)를 가지고 있다.

② object의 개념 : array를 이용하여 불러온다

ex ) var A = [“apple”,“banana”,“mango”,“orange”];

        ⤷ A[0] = “apple”

③ index대신에 property를 이용하기도 한다(data에 접근하기 위해서)

\* println : document.write 와 <br> 두 기능을 모두 가지고 있다.

25. javascript objects

① String objects(S가 대문자여야 object이다)

- new를 이용하여 새로운 것을 만들어

\* javascript는 html안에 둘러싸인 string을 반환할 수 있다(하지만 표준은 아니다)

② Boolean objects

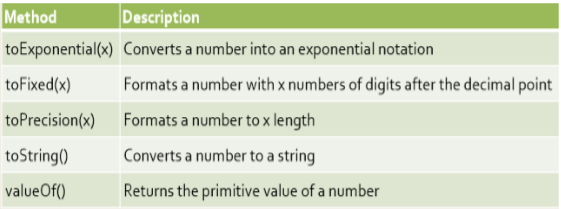
- var myBoolean= new Boolean(true);

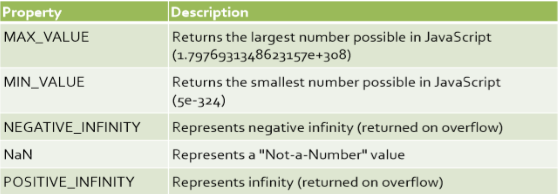
- method : toString(), valueOf() → 스트링을 값(0,1)으로 바꾸어줘

              ⤷ 값을 스트링(“true”)으로 바꾸어줘

③ Number objects(하나의 숫자 object)

- var num= new Number(value);

- method : 

- property : 

④ Array objects

표현 방법이 3가지가 존재

- var myCars = new Array();

my Cars[0] = "Hyundai";

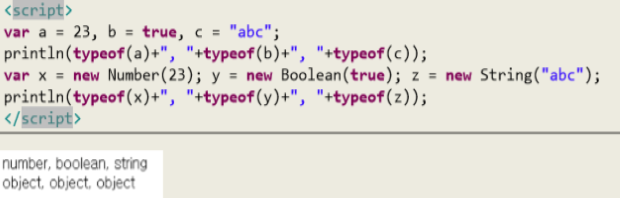
my Cars[1] = "Nissan";

my Cars[2] = "BMW";

- var myCars = new Array ("Hyundai", "Nissan", "BMW");

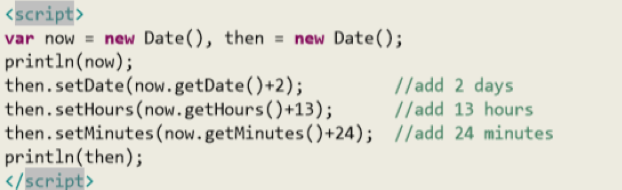
- var myCars = ["Hyundai", "Nissan", "BMW"];

26. javascript variables VS objects



\* 대문자로 적어주면 object이다

27. Date objects(현재의 시간을 가져온다)



+

\* set은 날짜를 변경할 때

\* get은 날짜를 가져올 때

28. Math object(c에서 math.h와 유사)

ex ) Math.min(0, 150, 30, 20, -8, -120) = -200

Math.round(4.7) = 5 (4.7과 가장 가까운 정수를 보여준다)

Math.ceil(4.4) = 5

Math.floor(4.7) = 4

29. Global properties and functions(object없이 쓸 수 있는 것을 global이라 한다)

- Global properties : Infinity , NaN  , undefined

- Global functions : eval() , isFinite() , isNaN() , Number() , parseFloat() , ParseInt()

\* 보통 object를 사용하면 object.method 이런 형태인데 위의 것들은 그게 아니다

30. Browser object : Allows JavaScript to “talk to” the browser

① Window object

- 브라우저 창을 열면 window object 하나가 자동으로 만들어진다.

- 무조건 생기기 때문에 생략해도 상관이 없다.

\* window.open은 새로 만들어진 window 안에 open이라는 method이다.

\* 함수이기에 **return value**가 있고 그것은 새로 만들어진 **window의 고유한 값**이다.

\* mywindow가 새로 만들어진 **window의 고유한 값**을 가지고 있어서 **!NULL**이다.

\* mywindow.close(mywindow) = 새로 만들어진 mywindow(고유한 값)를 닫아라.

② Screen object : window안의 화면을 표시한다.

- height : screen의 total height

- availHeight : window taskbar(내가 그리거나 쓰지 않는 부분)를 제외한 height

③ Location object : 어디를 보고 있는지 알려준다

println(location.href) = 현재 내 주소를 출력한다.

④ History object

기록을 알려준다

⑤ Navigator object : browser에 대한 정보를 가지고 있다.