

JavaScript 기초과정

-2-

곽정애
jeongae716@gmail.com
ver 1.2

연산자(Operator)

http://www.w3schools.com/js/js_operators.asp

http://www.w3schools.com/js/js_comparisons.asp

숫자 계산에 사용

연산자	읽기	기능	사용법	결과
+	플러스	+ (덧셈)	$a = 5 + 5$	10
-	마이너스	- (뺄셈)	$a = 5 - 5$	0
*	애스터리스크	* (곱셈)	$a = 5 * 5$	25
/	슬래시	% (나눗셈)	$a = 5 / 5$	1
%	퍼센트	... (나머지)	$a = 5 \% 5$	0

```
console.log(6 / 4);  
console.log(6 % 4); // for_position.html
```

`a = a + 2;`
연산
대입

`a += 2`

복합 대입 연산자

연산자	기능	사용법	결과
<code>+=</code>	더해서 대입	<code>a = 5; a += 5;</code>	10
<code>-=</code>	빼서 대입	<code>a = 5; a -= 5;</code>	0
<code>*=</code>	곱해서 대입	<code>a = 5; a *= 5;</code>	25
<code>/=</code>	나눠서 대입	<code>a = 5; a /= 5;</code>	1
<code>%=</code>	나머지를 대입	<code>a = 5; a %= 5;</code>	0

증가 연산자, 감소 연산자

변수의 수치를 1씩 증가, 또는 감소시킬 때 사용

연산자	명칭	기능	사용법	결과
++	증가 연산자	변수의 값을 1 증가	a = 1; a++;	2
			a = 1; ++a;	
--	감소 연산자	변수의 값을 1 감소	a = 1; a--;	0
			a = 1; --a;	

```
var a = 1;  
var b = ++a;  
console.log(a, b);
```

```
var a = 1;  
a = a + 1;  
var b = a;
```

```
var a = 1;  
var c = a++;  
console.log(a, c);
```

```
var a = 1;  
var c = a;  
a = a + 1;
```

조건을 만족하면 참(true), 만족하지 않으면 거짓(false) 반환

연산자	기능	사용법	의미
==	= (같다)	a == b	a와 b의 값은 같다.
===	= (같다)	a === b	a와 b는 값과 데이터 타입이 같다.
<	< (적다)	a < b	a는 b보다 작다.
>	> (많다)	a > b	a는 b보다 크다.
<=	<= (이하)	a <= b	a는 b보다 작거나 같다.
>=	>= (이상)	a >= b	a는 b보다 크거나 같다.
!=	≠ (같지 않다)	a != b	a와 b는 같지 않다.
!==	≠ (같지 않다)	a !== b	a와 b는 값이 같지 않거나, 데이터 타입이 같지 않다.

조건식의 결과에 따라 값이나 처리를 선택

- (조건) ? 참일 때 실행할 구문 : 거짓일 때 실행할 구문

```
var value = 0.7;  
var direction = (value >= 0.5) ? "오른쪽" : "왼쪽";
```

// 위의 식을 풀어 쓰면 아래와 같다.

```
var value = 0.7;  
var direction;  
  
if (value >= 0.5) {  
    direction = "오른쪽";  
} else {  
    direction = "왼쪽";  
}
```

여러 개의 조건식을 조합할 때 사용

연산자	기능	사용법	의미
&&	그리고	<code>(a >= 10) && (a < 50)</code>	a는 10이상이고, 50 미만입니다.
	또는	<code>(a == 1) (a == 100)</code>	a는 1 또는 100입니다.
!	~이 아니다	<code>!(a == 100)</code>	a는 100이 아닙니다.

여기서 잠깐!

10 <= a < 50 형태로 사용할 수는 없습니다.

연산자 우선 순위

우선 순위	연산자	같은 순위일 경우 계산방향	우선 순위	연산자	같은 순위일 경우 계산방향
1	., []	→	9	==, !=, ===, !==	→
	new	←	10	&	→
2	()	→	11	^	→
3	++, --	—	12		→
4	!, ~, +, -(단항연산자), typeof void delete	←	13	&&	→
5	*, /, %	→	14		→
6	+, -(덧셈, 뺄셈)	→	15	?:	←
7	<<, >>, >>>	→	16	=, +=, -=, *=, /=, % =, < <=, >>=, >>>=, &=, ^ =, =	←
8	<, <=, >, >=, instanceof	→	17	,	→

//우선 순위가 다를 때

```
a + b * c
c = a == b
```

//우선 순위가 같을 때

```
a + b - c
a = b = c = 1
```

조건문(Conditional Statement)

if (조건) {

// 조건을 만족할 경우 실행

}

```
// num의 값이 짝수일 경우, "~는 짝수입니다."를 출력하기
```

```
var num = 4;
```

```
if (num % 2 == 0) {  
    console.log( num + "는 짝수입니다.");  
}
```

문제 1. study의 값이 "javascript"일 경우, "자바스크립트 모임!!"을 출력해주세요.

문제 2. number의 값이 0보다 작을 경우, "음수입니다."를 출력해주세요.

문제 3. myName의 값이 본인 이름이 아닐 경우, "제 이름이 아닙니다."를 출력해주세요.

```
if (조건) {  
    // 조건을 만족할 경우 실행  
} else {  
    // 조건을 만족하지 않을 경우 실행  
}
```

http://www.w3schools.com/js/js_if_else.asp

```
/*
    num의 값이 짝수일 경우, "~는 짝수입니다."를 출력하기
    짝수가 아닐 경우에는 "~는 홀수입니다."를 출력하기
*/
var num = 4;

if (num % 2 == 0) {
    console.log( num + "는 짝수입니다.");
} else {
    console.log( num + "는 홀수입니다.");
}
```

```
// 위의 코드를 삼항연산자로 표현하기
var num = 4;
var str = "";

str = (num % 2 == 0) ? num + "는 짝수입니다." : num + "는 홀수입니다.";
console.log(str);
```

문제 1. id의 값이 "javascript", pw의 값이 "0123"일 경우 "로그인 되었습니다."를 출력하고,
그렇지 않을 경우에는 "아이디나 비밀번호가 틀렸습니다."를 출력해주세요.

문제 2. grade의 값이 75 이상일 경우 "합격입니다."를 출력하고,
그렇지 않을 경우에는 "불합격입니다."를 출력해주세요.

문제 3. str의 문자열 길이가 10 이하일 경우 str의 값을 출력하고,
그보다 클 경우에는 "입력할 수 있는 글자는 최대 10글자입니다."라고 출력해주세요.

문제 4. subject의 문자열에 "[javascript스터디]"가 포함되어 있을 경우 "OK"를 출력하고,
포함되어 있지 않을 경우에는 "유효하지 않은 메일 제목입니다."를 출력해주세요.

if...else if...else문

```
if (조건1) {  
    // 조건1을 만족할 경우 실행  
} else if (조건2) {  
    // 조건2를 만족할 경우 실행  
} else {  
    // 위의 조건들을 만족하지 않을 경우 실행  
}
```

if...else if...else문

문제 1. callNumber의 값에 따라 각각 아래와 같이 출력해 주세요.

1588-1111일 경우 "카드 사용 등록 안내입니다.",

1588-2222일 경우 "결제 금액 조회 안내입니다.",

1588-3333일 경우 "카드 분실 신고 안내입니다.",

1588-0000일 경우 "상담원을 연결합니다.",

그 외의 번호는 "잘못 누르셨습니다. 다시 입력해주세요."

문제 2. grade의 값에 따라 아래와 같이 출력해 주세요.

40미만은 F학점,

40이상은 D학점,

55이상은 C학점,

70이상은 B학점,

85이상은 A학점


```
switch (변수) {  
  case 값1:  
    // 변수가 값1이면 실행  
    break;  
  case 값2:  
    // 변수가 값2이면 실행  
    break;  
  default:  
    // 위의 값들에 해당하지 않을 경우 실행  
}
```

http://www.w3schools.com/js/js_switch.asp

```
// 요일을 숫자로 반환
// (0은 일요일, 1은 월요일...)
var day = new Date().getDay();

switch (day) {
case 1:
    console.log("월요일");
    break;
case 2:
    console.log("화요일");
    break;
case 3:
    console.log("수요일");
    break;
case 4:
    console.log("목요일");
    break;
```

```
case 5:
    console.log("금요일");
    break;
case 0:
case 6:
    console.log("주말");
    break;
default:
    console.log("요일이 아닙니다.");
}
```

문제 1. callNumber의 값에 따라 각각 아래와 같이 출력해 주세요.

1588-1111일 경우 "카드 사용 등록 안내입니다.",

1588-2222일 경우 "결제 금액 조회 안내입니다.",

1588-3333일 경우 "카드 분실 신고 안내입니다.",

1588-0000일 경우 "상담원을 연결합니다.",

그 외의 번호는 "잘못 누르셨습니다. 다시 입력해주세요."

문제 2. 변수 값에 따라 아래와 같이 출력해 주세요.

"red" 또는 "FF0000"일 경우 "set red color.",

"green" 또는 "00FF00"일 경우 "set green color.",

"blue" 또는 "0000FF"일 경우 "set blue color."

그 외에는 "red, green, blue만 컬러 설정이 가능합니다."

문제 1. 각 달에 따라 아래와 같이 출력해주세요.

입력한 달이 1일 경우 "1월은 31일이 있습니다."

입력한 달이 2일 경우 "2월은 28일이 있습니다."

... (중략)

입력한 달이 12일 경우 "12월은 31일이 있습니다."

입력한 달이 1~12가 아닐 경우, "1부터 12까지 올바른 달을 입력해 주세요."

반복문 (Repetitive Statement)

```
// li 태그를 10번 출력하기
```

```
console.log("<li>내용</li>");
```

```
console.log("<li>내용</li>");
```

```
... (중략)
```

```
console.log("<li>내용</li>");
```

```
// li 태그의 내용 뒤에 숫자를 붙여서 20번 출력하기
```

```
console.log("<li>내용 1</li>");
```

```
console.log("<li>내용 2</li>");
```

```
... (중략)
```

```
console.log("<li>내용 20</li>");
```

```
// li 태그의 내용 뒤에 숫자를 붙여서 1000번 출력해야 한다면?!
```

```
// li 태그를 10번 출력하기
console.log("<li>내용</li>");
console.log("<li>내용</li>");
... (중략)
console.log("<li>내용</li>");

// li 태그의 내용 뒤에 숫자를 붙여서
// 20번 출력하기
console.log("<li>내용 1</li>");
console.log("<li>내용 2</li>");
... (중략)
console.log("<li>내용 20</li>");

// li 태그의 내용 뒤에 숫자를 붙여서
// 1000번 출력해야 한다면?!
```



```
// li 태그를 10번 출력하기
for (var i = 0; i < 10; i++) {
    console.log("<li>내용</li>");
}

// li 태그의 내용 뒤에 숫자를 붙여서
// 20번 출력하기
for (var i = 1; i <= 20; i++) {
    console.log("<li>내용 " + i + "</li>");
}

// li 태그의 내용 뒤에 숫자를 붙여서
// 1000번 출력해야 한다면?!
for (var i = 1; i <= 1000; i++) {
    console.log("<li>내용 " + i + "</li>");
}
```

```
for (초기화; 조건문; 증감문) {  
    // 조건을 만족할 경우 실행  
}
```

```
// 0부터 9까지 출력  
for (var i = 0, total = 10; i < total; i++) {  
    console.log(i);  
}  
  
// 10부터 1까지 출력  
for (var i = 10; i > 0; i--) {  
    console.log(i);  
}
```

http://www.w3schools.com/js/js_loop_for.asp


```
// for.html
// 모든 li에 "on" 클래스 적용하기
// 태그 이름으로 엘리먼트를 찾을 때: document.getElementsByTagName(태그명)
var objLi = document.getElementsByTagName("li");
for (var i = 0, length = objLi.length; i < length; i++ ) {
    objLi[i].className = "on";
}
```

문제 1. "자바스크립트 스터디 모임!!"을 한 글자씩 출력해 주세요.

문제 2. 10부터 30에서 짝수만 출력해주세요.

문제 3. ul 안에 "data 1"부터 "data 12"까지의 글자를 가진 총 12개의 li를 생성해주세요.

각각의 li는 style1, style2, style3의 클래스를 순서대로 적용해주세요.

```
for (초기화; 조건문; 증감문) {  
    for (초기화; 조건문; 증감문) {  
        // 조건을 만족할 경우 실행  
    }  
}
```

```
for (var i = 0; i < 3; i++) {  
    for (var j = 0; j < 5; j++) {  
        console.log("i: " + i, "j: " + j);  
    }  
    console.log("end inner loop..");  
}  
console.log("end outer loop..");
```

문제 1. 구구단을 아래와 같이 출력해 주세요.

(출력해야 할 값들을 문자열에 담아 한 번만 출력해주세요.)

$$2 * 1 = 2$$

$$2 * 2 = 4$$

... (중략)

$$9 * 8 = 72$$

$$9 * 9 = 81$$

문제 2. 구구단을 아래와 같이 출력해 주세요.

(출력해야 할 값들을 문자열에 담아 한 번만 출력해주세요.)

$$2 * 1 = 2 \quad 2 * 2 = 4 \quad \dots (\text{중략}) \quad 2 * 9 = 18$$

$$3 * 1 = 3 \quad 3 * 2 = 6 \quad \dots (\text{중략}) \quad 3 * 9 = 27$$

... (중략)

$$9 * 1 = 9 \quad 9 * 2 = 18 \quad \dots (\text{중략}) \quad 9 * 9 = 81$$

while (조건) {

// 조건이 참일 동안 계속 실행

}

// 1부터 10까지 출력

var i = 1;

while (i <= 10) {

 console.log("i: " + i);

 i++;

}

// 1부터 10까지 출력

for (var i = 1; i <= 10; i++) {

 console.log ("i: " + i);

}

http://www.w3schools.com/js/js_loop_while.asp

여기서 잠깐!

무한루프 주의!!

문제 1. 빈 배열을 하나 생성하고,

그 안에 0~10 사이의 임의의 수를 생성해서 넣습니다.

배열의 길이가 5개가 될 때까지 수를 생성해서 넣은 후, 배열을 출력해 주세요.

```
do {
```

```
    // 우선 한 번 실행.
```

```
    // while의 조건을 확인하고 참이면 계속 실행
```

```
} while (조건);
```

```
// "end"가 입력될 때까지 계속 입력창 띄우기
```

```
do {
```

```
    value = window.prompt("end가 입력될 때까지 계속 입력 받습니다.");
```

```
} while (value != "end")
```

```
// 숫자를 입력 받고 -1씩 빼서 0이 될 때까지 숫자를 출력하기
```

```
var inputNum = window.prompt("숫자를 입력해주세요.");
```

```
var num = parseInt(inputNum);
```

```
do {
```

```
    console.log(num);
```

```
    num--;
```

```
} while (num >= 0)
```

문제 1. 부르마블 게임의 무인도에 빠졌을 때 탈출 조건을 만들어보자.

"무인도에 도착했습니다."를 먼저 출력한다.

그리고 주사위처럼 1~6까지 두 개의 랜덤한 숫자를 생성하고,

그 숫자가 같지 않을 경우

"주사위를 돌렸습니다!! 주사위 1: 값, 주사위 2: 값"라고 출력하고

다시 두 개의 숫자를 생성한다.

두 개의 숫자가 같은 값이 나오거나, 같지 않을 경우가 3번째가 되면

"무인도 탈출!!"을 출력해 보자.

무인도에 도착했습니다.

주사위를 돌렸습니다!! 주사위 1: 5 주사위 2: 1

주사위를 돌렸습니다!! 주사위 1: 5 주사위 2: 5

무인도 탈출!!

주사위 두 개가 같은 수가 나왔으므로 탈출

무인도에 도착했습니다.

주사위를 돌렸습니다!! 주사위 1: 1 주사위 2: 4

주사위를 돌렸습니다!! 주사위 1: 3 주사위 2: 2

주사위를 돌렸습니다!! 주사위 1: 3 주사위 2: 1

무인도 탈출!!

주사위 두 개가 같은 수가 나오지 않았지만
3번 돌렸으므로 탈출

기타 제어문


```
// 배열 안에 5가 몇 번 인덱스에 있는지 구하기
var arr = [3, 4, 5, 1, 2, 6];
for (var i = 0, total = arr.length; i < total; i++) {
  console.log(i);
  if (arr[i] == 5) {
    console.log("5의 인덱스는 " + i + "입니다.");
    break;
  }
}
```

```
// 배열 안의 값을 출력하되, 그 값이 5일 경우에는 출력하지 않기
var arr = [3, 4, 5, 1, 2, 6];
for (var i = 0, total = arr.length; i < total; i++) {
    if (arr[i] == 5) {
        continue;
    }
    console.log("인덱스" + i + "의 값은 " + arr[i] + "입니다.");
}
```