



# CHAPTER 8.

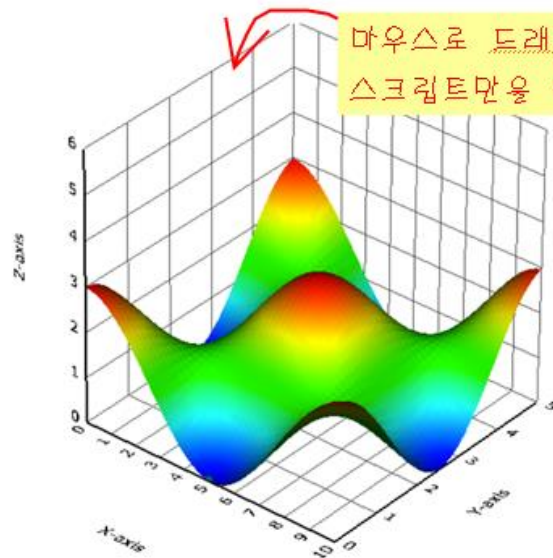
## 자바 스크립트 기초





# 자바 스크립트 소개

- 자바스크립트(javascript): 동적인 웹 페이지를 작성하기 위하여 사용되는 언어
- 웹의 표준 프로그래밍 언어
- 모든 웹브라우저들은 자바스크립트를 지원



마우스로 드래그하여 회전할 수 있는 이러한 화면도 자바스크립트만을 이용하여 구현되었다.



# HTML5 기술의 핵심





# 자바 vs 자바 스크립트

특징	자바 언어	자바스크립트
언어 종류	소스 파일을 컴파일하여 실행하는 컴파일 언어이다.	브라우저가 소스 코드를 직접 해석하여 실행하는 인터프리트 언어이다.
실행 방식	자바 가상 기계 위에서 실행한다.	브라우저 위에서 실행된다.
작성 위치	별도의 소스 파일에 작성	HTML 파일 안에 삽입 가능
변수 선언	변수의 타입을 반드시 선언해야 함	변수의 타입을 선언하지 않아도 사용 가능



# 자바 스크립트 역사

- 넷스케이프의 브렌던 아이크(Brendan Eich)가 개발
- 처음에는 라이브스크립트(LiveScript)
- 최신 버전은 자바스크립트 1.8.5
- ECMA(European Computer Manufacturer's Association)이 ECMAScript라는 이름으로 표준을 제정-> ECMA-262





# 자바 스크립트 특징

- 인터프리트 언어
- 동적 타이핑(dynamic typing)
- 구조적 프로그래밍 지원
- 객체 기반
- 함수형 프로그래밍 지원
- 프로토타입-기반(prototype-based)

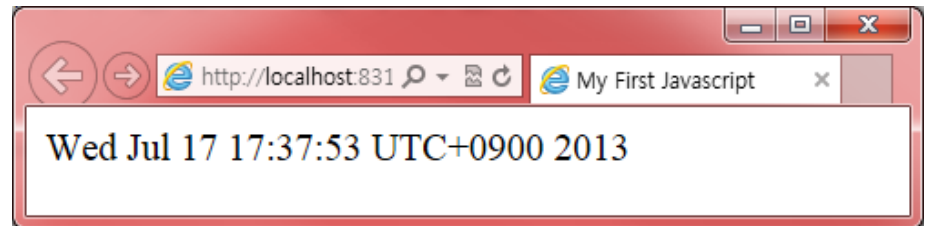


# 첫 번째 예제

```
<!DOCTYPE HTML>
<html>
<head>
  <title> My First Javascript </title>
</head>
<body>

  <script>
    var now = new Date();
    document.write(now);
  </script>

</body>
</html>
```



웹브라우저에서 실행



# 자바 스크립트의 용도

- 이벤트에 반응하는 동작을 구현
- AJAX
- HTML 요소들의 크기나 색상을 동적으로 변경
- 게임이나 애니메이션
- 사용자가 입력한 값들을 검증





# 자바 스크립트의 위치

- 내부 자바스크립트
- 외부 자바스크립트
- 인라인 자바스크립트



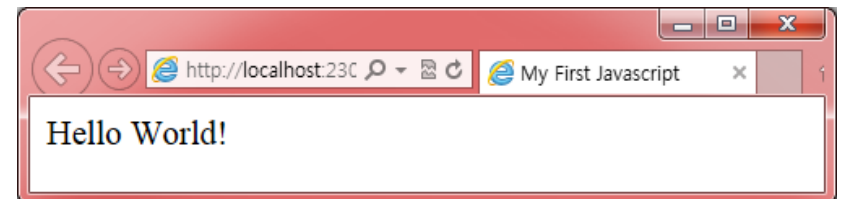
# 내부 자바 스크립트

```
<!DOCTYPE HTML>
<html>
<head>

  <title>My First Javascript </title>

  <script>
    document.write("Hello World!");
  </script>

</head>
<body></body>
</html>
```



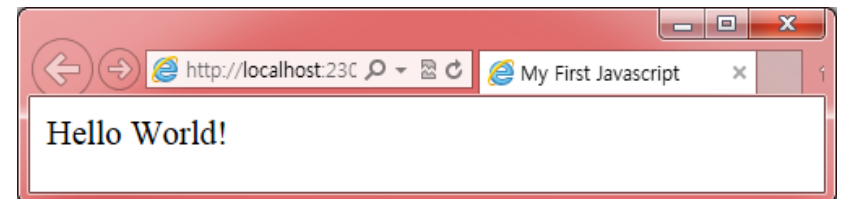


# 외부 자바 스크립트

```
<!DOCTYPE html>
<html>
<head>
  <script src="myscript.js"></script>
</head>
<body>
</body>
</html>
```

*myscript.js*

```
document.write("Hello World!");
```





# 인라인 자바 스크립트

```
<!DOCTYPE html>  
<html>  
<body>  
  <button type="button" onclick="alert('반갑습니다.')">버튼을 누르세요!</button>  
</body>  
</html>
```





# 문장

- 자바스크립트 문장(statement)들은 웹 브라우저에게 내리는 명령

1. 문서에 "Hello World!"를 추가하시오.
2. 화면에 경고창을 띄우시오.
3. 변수를 하나 만드시오.
4. ...



```
document.write("Hello World!");  
alert("warning!!");  
var count;  
...
```

순차적  
으로  
실행된  
다.



# 변수

- 변수(**variable**)는 데이터를 저장하는 상자
- var 키워드를 사용하여 선언(declare)한다.

변수 이름이다. 스크립트 안에서 유일해야 한다.

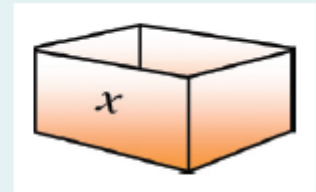
var

x

;

var 키워드는 변수가 선언된다는 것을 나타낸다.

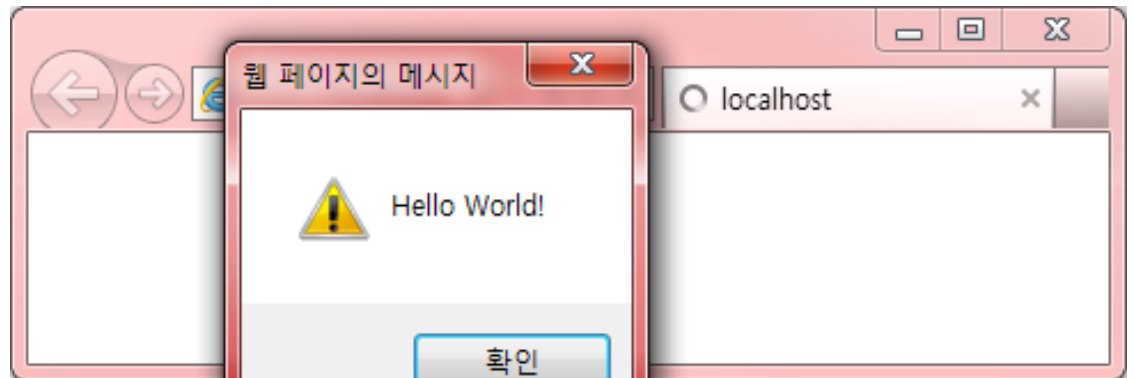
세미콜론은 하나의 문장이 끝났음을 나타낸다.





# 예제

```
<script>  
  var x;  
  x = "Hello World!";  
  alert(x);  
</script>
```





# 자료형

- 수치형(number)
- 문자열(string)
- 부울형(boolean)
- 객체형(object)
- undefined

아직 설정이안됨.



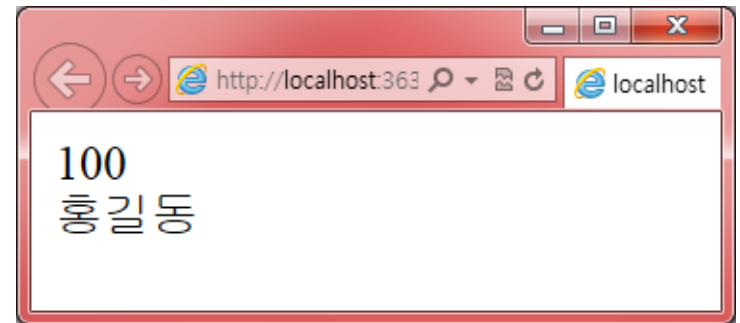


# 예제

```
<script>
  var s;

  s = 100;
  document.write(s + "<br>");

  s = "홍길동";
  document.write(s + "<br>");
</script>
```

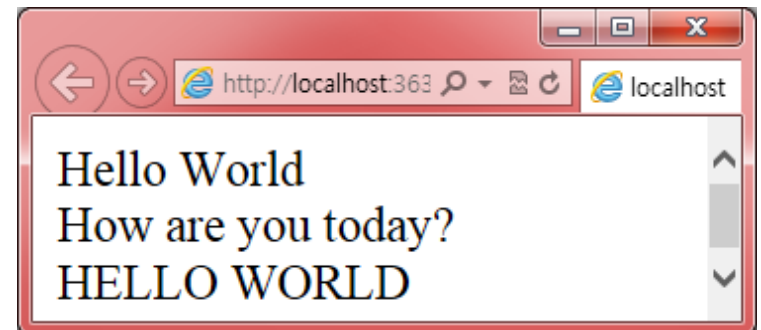




# 예제

```
<script>
  var s = "Hello World";
  var t = "How are you" + " today?";

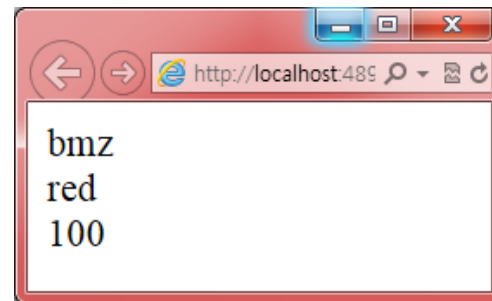
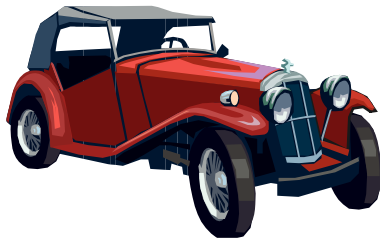
  document.write(s + "<br>");
  document.write(t + "<br>");
  document.write(s.toUpperCase() + "<br>");
</script>
```





# 객체형

- 객체(**object**)는 사물의 속성과 동작을 묶어서 표현하는 기법
- (예) 자동차는 메이커, 모델, 색상, 마력과 같은 속성도 있고 출발하기, 정지하기 등의 동작도 가지고 있다.
- `var myCar = { model: "bmz", color: "red", hp: 100 };`
- `document.write(myCar.model + "<br>");`
- `document.write(myCar.color + "<br>");`
- `document.write(myCar.hp + "<br>");`





# 연산자

연산자	설명	예	수식의 값
+	덧셈	$x=3+2$	5
-	뺄셈	$x=3-2$	1
*	곱셈	$x=3*2$	6
/	나눗셈	$x=3/2$	1.5
%	나머지	$x=3\%2$	1
++	증가	$++x$	x의 값 3 → 4
--	감소	$--x$	x의 값 3 → 2



# prompt() 함수

```
<script>  
  var age = prompt("나이를 입력하세요", "만나이로 입력합니다.");  
</script>
```





# 예제

```
<script>
  var x, y;
  var input;

  input = prompt("정수를 입력하시오", "정수로");
  x = parseInt(input);

  input = prompt("정수를 입력하시오", "정수로");
  y = parseInt(input);

  document.write(x + y + "<br>");
</script>
```



웹브라우저에서 실행



# 예제

```
<html>
<head>

  <title>Calculator</title>
  <script>
    function calc() {

      var x = document.getElementById("x").value;

      var y = document.getElementById("y").value;
      var sum;

      sum = parseInt(x) + parseInt(y);
      document.getElementById("sum").value = sum;
    }
  </script>
</head>
```

웹브라우저에서 실행



# 예제

```
<body>
  <h3>덧셈 계산기</h3>
  <form name="myform" action="..." method="POST">
    첫째 정수:
    <input id="x" /><br />
    두번째 정수:
    <input id="y" /><br />
    합계:
    <input id="sum" /><br />
    <input type="button" value="계산" onclick="calc();" />
  </form>

</body>
</html>
```

웹브라우저에서 실행

덧셈 계산기

첫번째 정수: 10

두번째 정수: 20

합계: 30

계산



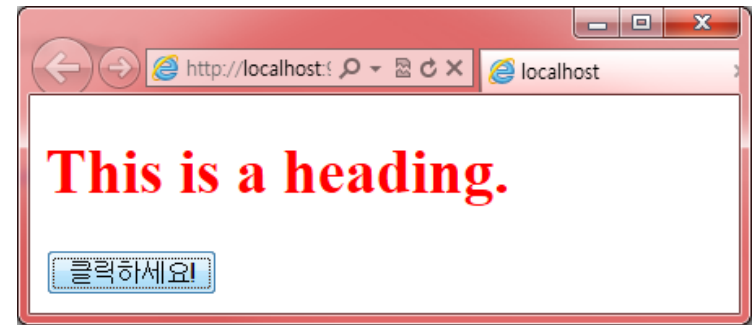


# HTML 요소에 접근하기

```
<!DOCTYPE html>
<html>
<body>

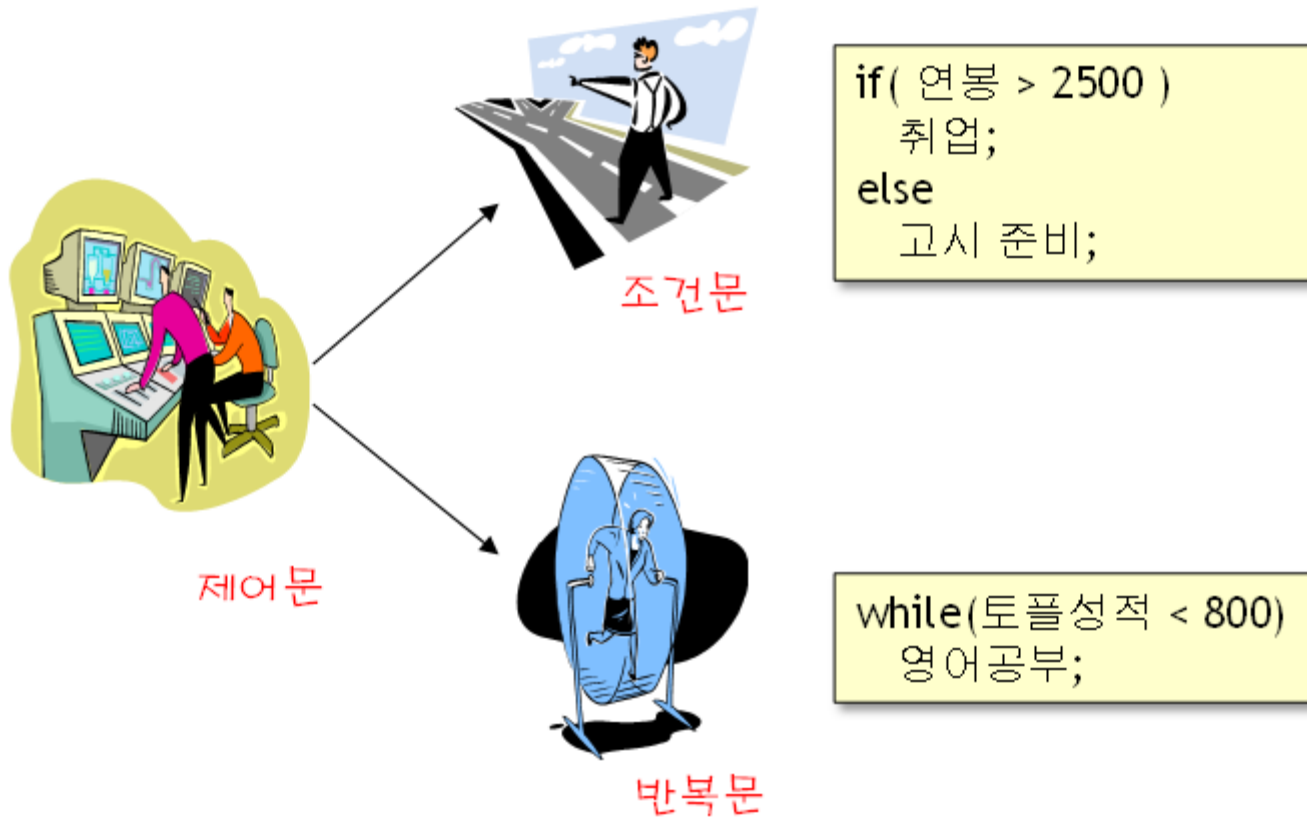
  <h1 id="test">This is a heading.</h1>
  <script>
    function func() {
      e = document.getElementById("test");
      e.style.color = "red";
    }
  </script>
  <button type="button" onclick="func()">클릭하세요!</button>
</body>
</html>
```

웹브라우저에서 실행





# 제어문



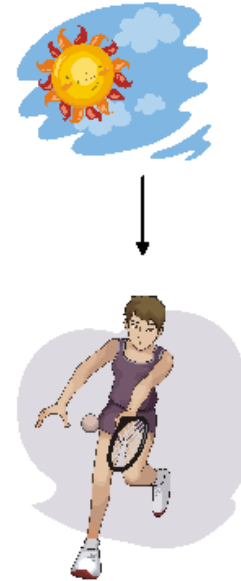
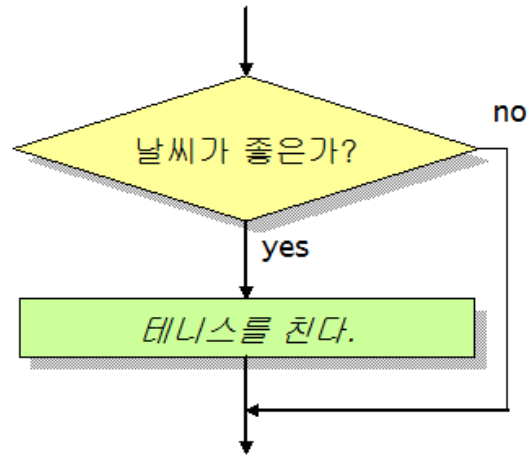


# 조건문의 종류

- if 문
- if...else 문
- switch 문



# if 문



```
if (time<12)
{
    greeting="Good Morning!";
}
```



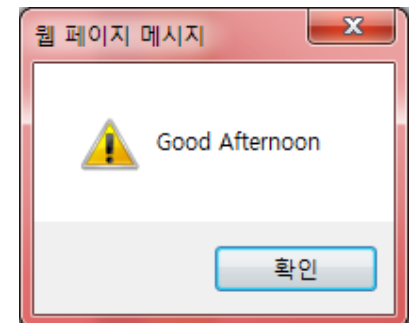
# if-else 문

```
if (time<12)
{
    msg="Good Morning!";
}
else{
    msg="Good Afternoon!";
}
```



# 연속적인 if 문

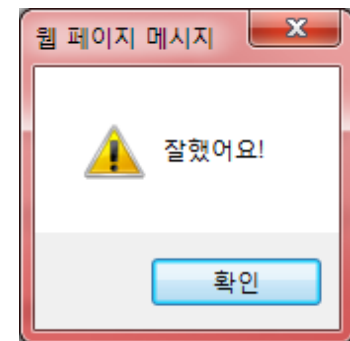
```
<script>
  var msg = "";
  var time = new Date().getHours();
  if (time < 12) {                                // 12시 이전이면
    msg = "Good Morning";
  }
  else if (time < 18) {                           // 오후 6시 이전이면
    msg = "Good Afternoon";
  }
  else {                                          // 그렇지 않으면(오후 6시
    이후이면)
    msg = "Good evening";
  }
  alert(msg);
</script>
```





# switch 문

```
<script>
  var grade = prompt("성적을 입력하시오:", "A-F사이의 문자로");
  switch (grade) {
    case 'A': alert("잘했어요!");
              break;
    case 'B': alert("좋은 점수군요");
              break;
    case 'C': alert("괜찮은 점수군요");
              break;
    case 'D': alert("좀더 노력하세요");
              break;
    case 'F': alert("다음학기 수강하세요");
              break;
    default: alert("알수없는 학점입니다.")
  }
</script>
```





# 숫자 게임 예제

```
<html>
<head>
  <title></title>
  <script>
    var computerNumber = 53;      // 정답
    var nGuesses = 0;             // 추측 횟수

    function guess() {
      var result = "";            // 결과 메시지

      // 사용자가 입력한 값을 받아서 변수 number에 대입한다.
      var number = parseInt(document.getElementById("user").value);
      nGuesses++;                 // 추측 횟수를 증가시킨다.

      if (number == computerNumber) result = "성공입니다.";
      else if (number < computerNumber) result = "낮습니다.";
      else result = "높습니다.";

      document.getElementById("result").value = result;
      document.getElementById("guesses").value = nGuesses;
      return true;
    }
  </script>
</head>
```





# 숫자 게임 예제

```
<body>
  <h2>숫자 맞추기 게임</h2>

  이 게임은 컴퓨터가 생성한 숫자를 맞추는 게임입니다. 숫자는 1부터 100 사이에
  있습니다.
  <form>
    숫자:
    <input type="text" id="user" size="5">
    <input type="button" value="확인" onclick="guess();">
    추측횟수:
    <input type="text" id="guesses" size="5">
    힌트:
    <input type="text" id="result" size="16">
  </form>
</body>
</html>
```

The screenshot shows a web browser window with the URL `http://localhost:7424/Projects/HTMLPz`. The page title is "숫자 맞추기 게임". The content includes a description of the game, a text input field for the number (containing "23"), a "확인" button, a text input field for the number of guesses (containing "1"), and a hint text field (containing "낮습니다").

숫자 맞추기 게임

이 게임은 컴퓨터가 생성한 숫자를 맞추는 게임입니다. 숫자는 1부터 100 사이에 있습니다.

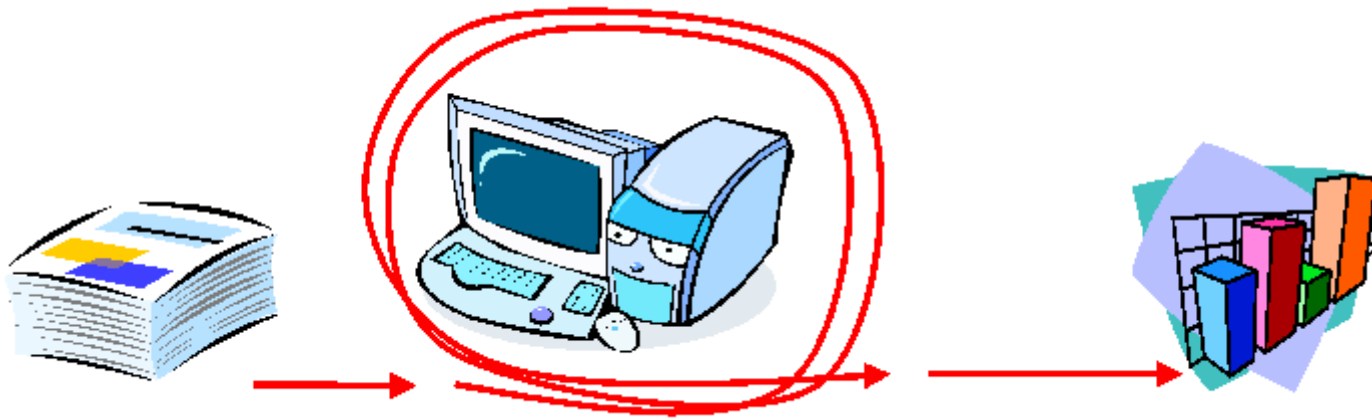
숫자:

추측횟수:  힌트:



# 반복문

- 같은 처리 과정을 여러 번 되풀이하는 것





# 반복문의 종류

- `while` – 지정된 조건이 참이면 반복 실행한다.
- `for` – 정해진 횟수 동안 코드를 반복 실행한다.



# while 문

```
var i = 0;
```

```
while( i < 10 ) ←반복 조건, 변수 i가 10보다 작으면 반복 계속
```

```
{
```

```
    document.write(i+" <br>"); ←반복되는 문장, i를 화면에 출력한다.
```

```
    i++; ←한 번의 반복마다 i를 하나 증가시킨다.
```

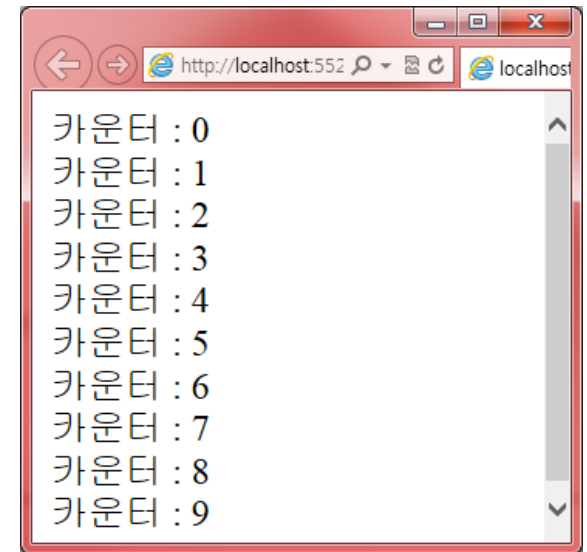
```
}
```

이 문장이 없으면 무한히 반복한다.



# while 문

```
<script>
  var i = 0;
  while (i < 10) {
    document.write("카운터 : " + i + "<br />");
    i++;
  }
</script>
```



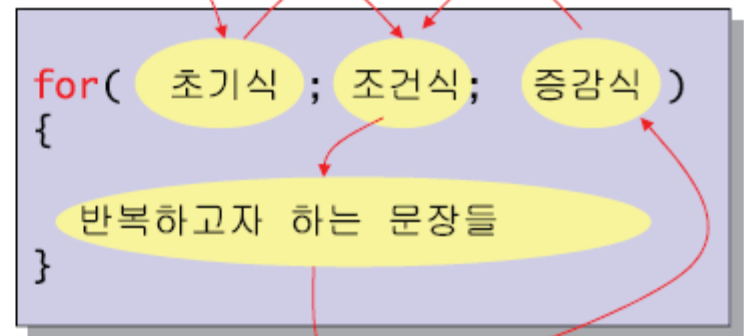


# for 문

```
for( 초기식 ; 조건식 ; 증감식 )
{
    document.write(i+"<br />");
}
```

초기식      조건식      증감식

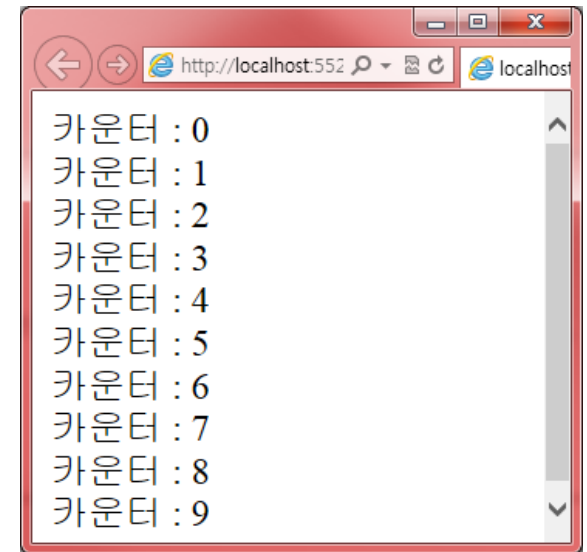
반복되는 문장





# for 문

```
<script>
  var i = 0;
  for (i = 0; i < 10; i++) {
    document.write("카운터 : " + i + "<br />");
  }
</script>
```





# 예제

```
<html>
<head>
  <title>온도 변환기</title>
</head>

<body>
  <table border="3">
    <tr>
      <td>섭씨 온도</td>
      <td>화씨 온도</td>
    </tr>
    <script>
      for (celsius = 0; celsius <= 10; celsius = celsius + 1) {
        document.write("<tr><td>" + celsius + "</td><td>"
          + ((celsius * 9.0 / 5) + 32) + "</td></tr>");
      }
    </script>
  </table>
</body>
</html>
```

섭씨 온도	화씨 온도
0	32
1	33.8
2	35.6
3	37.4
4	39.2
5	41
6	42.8
7	44.6
8	46.4
9	48.2
10	50





# 중첩 반복문 예제

```
<script>
```

```
document.write("<h1>구구단표</h1>");  
document.write("<table border=2 width=50%");
```

```
for (var i = 1; i <= 9; i++) {  
    document.write("<tr>");  
    document.write("<td>" + i + "</td>");  
  
    for (var j = 2; j <= 9; j++) {  
        document.write("<td>" + i * j + "</td>");  
    }  
}
```

```
    document.write("</tr>");  
}
```

```
document.write("</table>");
```

```
</script>
```

1	2	3	4	5	6	7	8	9
2	4	6	8	10	12	14	16	18
3	6	9	12	15	18	21	24	27
4	8	12	16	20	24	28	32	36
5	10	15	20	25	30	35	40	45
6	12	18	24	30	36	42	48	54
7	14	21	28	35	42	49	56	63
8	16	24	32	40	48	56	64	72
9	18	27	36	45	54	63	72	81



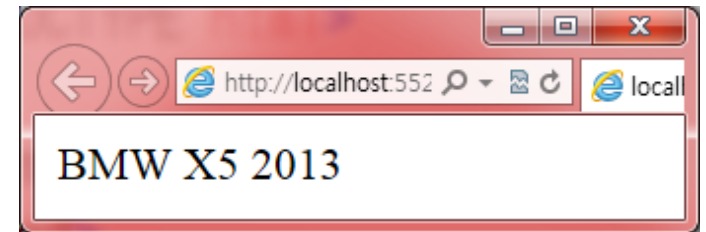
# for/in 반복문

```
<script>
  var myCar = { make: "BMW", model: "X5", year: 2013 };
  var txt="";

  for (var x in myCar) {

    txt += myCar[x] + " ";

  }
  document.write(txt);
</script>
```

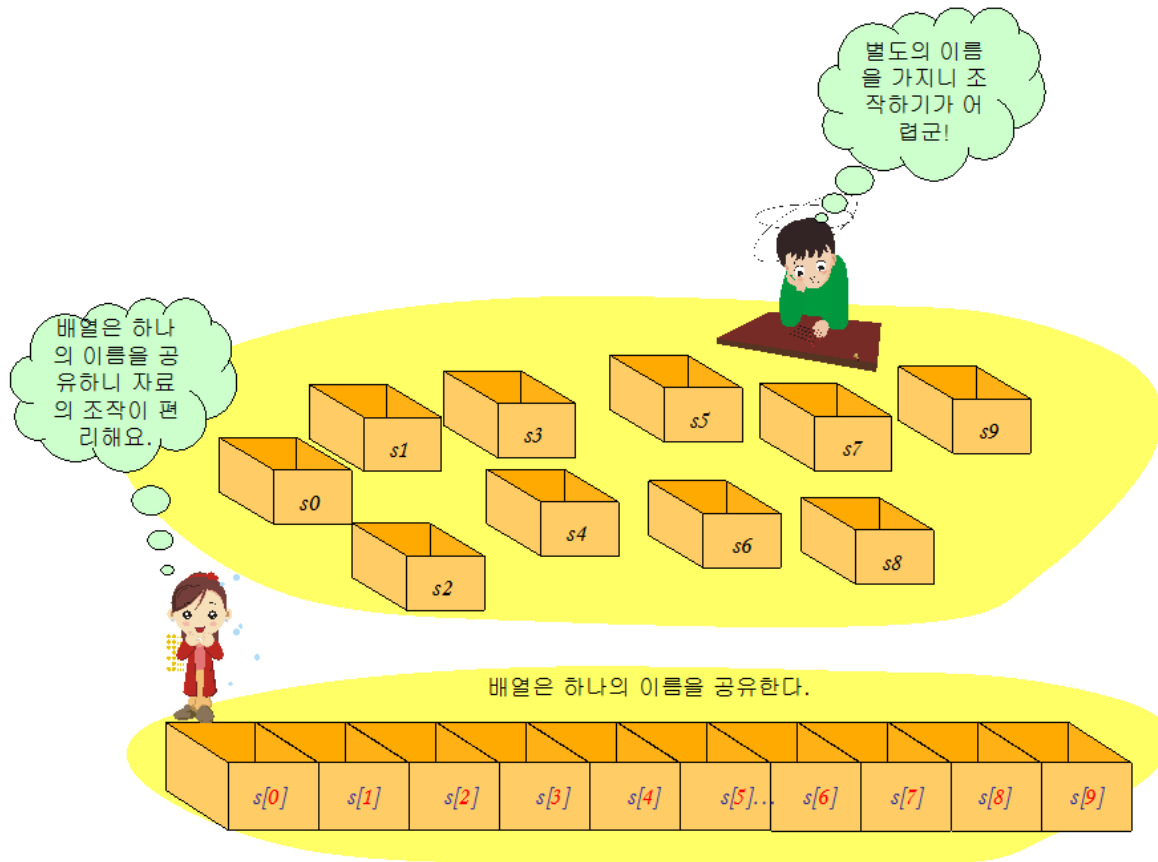




# 배열

java-List

- 많은 값을 저장할 수 있는 공간이 필요할 때 배열을 사용한다.
- 서로 관련된 데이터를 차례로 접근하여서 처리할 수 있다.





# 배열을 생성하는 2가지 방법

- 리터럴로 배열 생성
  - `var fruits = ["apple", "banana", "peach"];`
- Array 객체로 배열 생성
  - `var fruits=new Array("apple","banana","orange");`



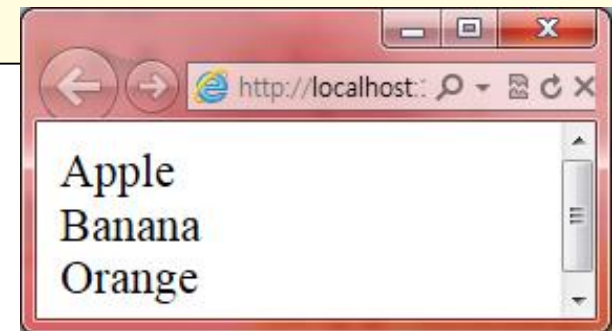
# 예제

```
<!DOCTYPE html>
<html>
<body>

<script>
  var i;
  var fruits = new Array();
  fruits[0] = "Apple";
  fruits[1] = "Banana";
  fruits[2] = "Orange";

  for (i = 0; i < fruits.length; i++) {
    document.write(fruits[i] + "<br>");
  }
</script>

</body>
</html>
```





# 연관 배열

```
<!DOCTYPE html>
<html>
<body>
  <form name="myForm">
    필드1<input type="text" name="a0"><br />
    필드2<input type="text" name="a1"><br />
    필드3<input type="text" name="a2"><br />
    <input type="button" value="초기화" onclick="init();">
  </form>

  <script>
    function init() {
      for (var i = 0; i < 3; i++) {
        document.myForm["a" + i].value = i;
      }
    }
  </script>
</body>
</html>
```

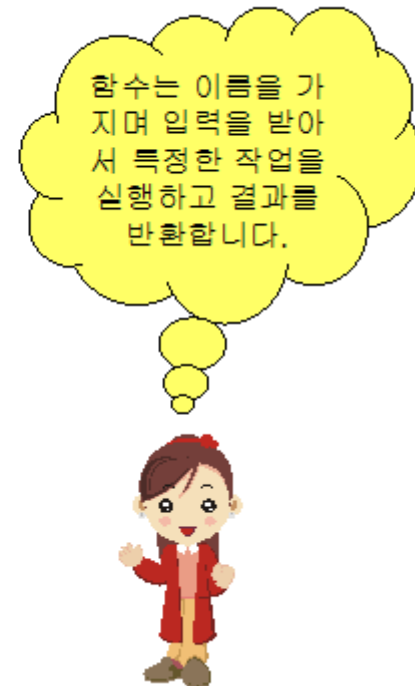
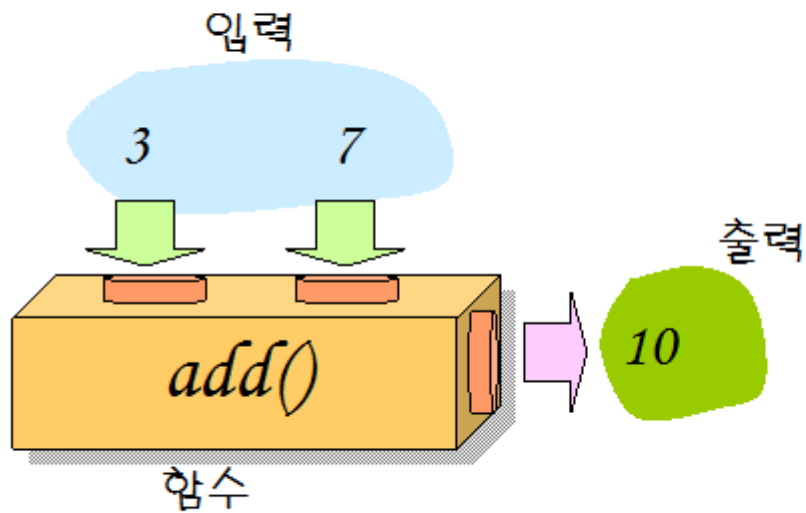
필드1	0
필드2	1
필드3	2

초기화



# 함수

- 함수는 입력을 받아서 특정한 작업을 수행하여서 결과를 반환하는 블랙 박스

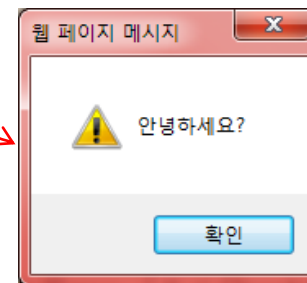
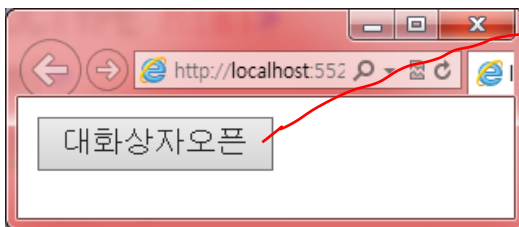




# 예제

```
<!DOCTYPE html>
<html>
<head>
  <script>
    function showDialog() {
      alert("안녕하세요?");
    }
  </script>
</head>

<body>
  <button onclick="showDialog()">대화상자오픈</button>
</body>
</html>
```

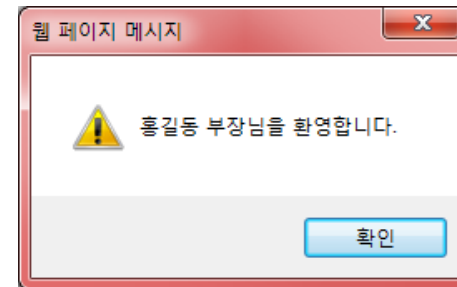
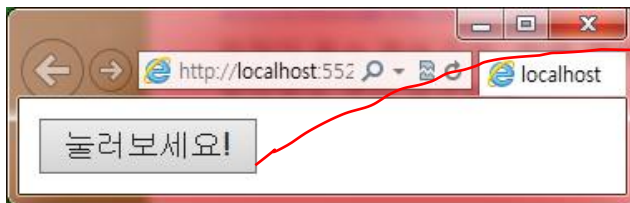






# 인수와 매개 변수

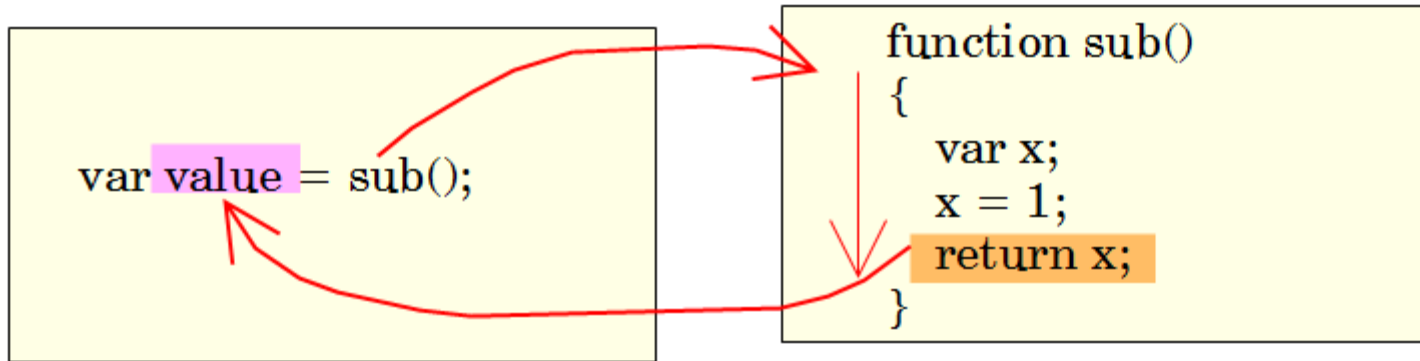
```
<!DOCTYPE html>
<html>
<head>
  <script>
    function greeting(name, position) {
      alert(name + " " + position + "님을 환영합니다.");
    }
  </script>
</head>
<body>
  <button onclick="greeting('홍길동', '부장')">눌러보세요!</button>
</body>
</html>
```





# 함수의 반환값

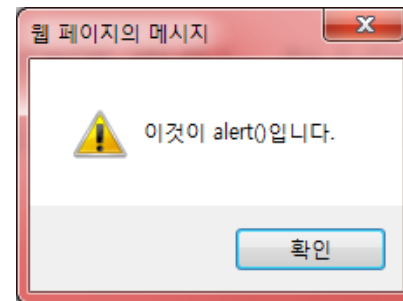
- return 문장을 사용하여 외부로 값을 반환





# alert() 함수

```
<script>  
  alert("이것이 alert()입니다.");  
</script>
```

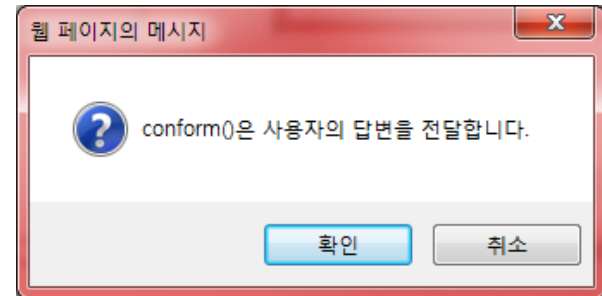




# confirm() 함수

확인용

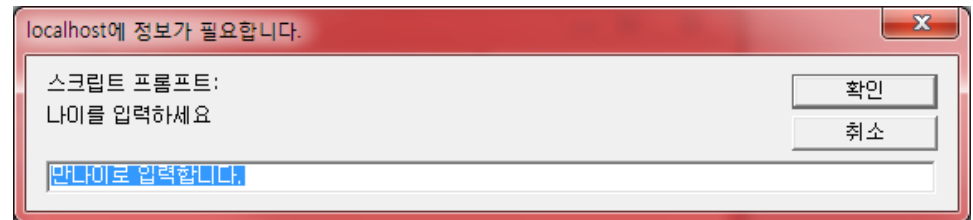
```
<script>  
  var user = confirm("confirm()은 사용자의 답변을 전달합니다.");  
</script>
```





# prompt() 함수

```
<script>  
  var age = prompt("나이를 입력하세요", "만나이로 입력합니다.");  
</script>
```





# Q & A

