

# 연산자

## - 기본 연산자 -

# 기본 연산자

---

## ❖ 복합 대입 연산자

- 대입 연산자와 다른 연산자를 함께 사용하는 연산자

연산자	설명
<code>+=</code>	기존 변수의 값에 값을 더합니다.
<code>-=</code>	기존 변수의 값에 값을 뺍니다.
<code>*=</code>	기존 변수의 값에 값을 곱합니다.
<code>/=</code>	기존 변수의 값에 값을 나눕니다.
<code>%=</code>	기존 변수의 값에 나머지를 구합니다.

# 기본 연산자

## ❖ 증감 연산자

- 복합 대입 연산자를 간략하게 사용한 형태

연산자	설명
변수++	기존의 변수 값에 1을 더합니다(후위).
++변수	기존의 변수 값에 1을 더합니다(전위).
변수--	기존의 변수 값에 1을 뺍니다(후위).
--변수	기존의 변수 값에 1을 뺍니다(전위).

# 기본 연산자

---

## ❖ 단항 연산자

- 양수를 음수로 바꾸거나 음수를 양수로 바꾸는 -기호
  - -52
  - -(-52)
- typeof 연산자
  - typeof 52
  - typeof(typeof 52)
- typeof 연산자의 괄호 사용
  - typeof 1 + '문자열'
  - typeof(1 + '문자열')
  - typeof(1) + '문자열'

## ❖ 이항 연산자

- 양쪽에 항을 써서 사용 : + 연산자/\*연산자

# 기본 연산자

---

## ❖ 자료형을 출력하는 예제 코드

- undefined
  - 정의하지 않은 자료형 의미
  - 선언하지 않은 식별자 alpha 사용

```
> typeof 5
'number'
> typeof 'String'
'string'
> typeof true
'boolean'
> typeof function(){}
'function'
> typeof {}
'object'
> typeof variable
'undefined'
```

# 기본 연산자

---

## ❖ undefined 자료형

- '존재하지 않는 것'은 undefined 자료형으로 표현
- 변수로 선언하지 않은 식별자가 갖는 자료형

```
> typeof variable  
'undefined'
```

- 변수를 선언했지만 초기화하지 않았을 때 undefined 자료형을 가짐

```
> var variable  
> typeof variable  
'undefined'
```

# 기본 연산자

---

## ❖ 문자열과 숫자를 더하는 자료형

- 숫자와 문자열을 덧셈 연산하면 문자열 우선

```
> '52 + 273'  
'52 + 273'
```

```
> '52'+'273'  
'52273'
```

```
> '52' + 273  
'52273'
```

```
> 52+273  
325
```

```
> 52 + '273'  
'52273'
```

# 기본 연산자

---

## ❖ 문자열과 숫자를 곱하는 자료형

- 더하기 연산자를 제외한 사칙 연산자는 숫자가 우선
  - 첫 번째를 제외하면 14196을 출력

```
> '52*273'  
'52*273'
```

```
> 52*273  
14196
```

```
> '52'*273  
14196
```

```
> 52*'273'  
14196
```

```
> '52'*'273'  
14196
```



# 기본 연산자

---

## ❖ 03\_1\_01\_arithmetic.html

```
<body>
<script>
  var a = 5;
  var b = 3;

  document.write("a + b = " + (a + b) + "<br>");
  document.write("a - b = " + (a - b) + "<br>");
  document.write("a * b = " + (a * b) + "<br>");
  document.write("a / b = " + (a / b) + "<br>");
</script>
</body>
```

# 기본 연산자

---

## ❖ 03\_1\_02\_increase.html

```
<body>
  <script>
    var a = 2;
    var b = a++;
    document.write("a = " + a + ", b = " + b + "<br>");

    var a = 2;
    var b = ++a;
    document.write("a = " + a + ", b = " + b + "<br>");
  </script>
</body>
```

# 기본 연산자

---

## ❖ 03\_1\_03\_stringincrease.html

```
<body>
  <script>
    var a = "3";
    a++;
    document.write("a = " + a + ", type = " + typeof(a) + "<br>");

    var a= "3";
    a += 1;
    document.write("a = " + a + ", type = " + typeof(a) + "<br>");

    var a= "3";
    a = a + 1;
    document.write("a = " + a + ", type = " + typeof(a) + "<br>");
  </script>
</body>
```

# 기본 연산자

---

## ❖ 03\_1\_04\_plusequal.html

```
<body>
  <script>
    var order = "";
    order += "date : 2014-06-29";
    order += "<br>"
    order += "item : notebook";
    order += "<br>"
    order += "price : 134";
    document.write(order);
  </script>
</body>
```

# 연산자

## - 비교 연산자 -

## 비교 연산자

### ❖ 비교 연산자

연산자	설명
>	좌변이 크다.
<	좌변이 작다.
>=	좌변이 크거나 같다.
<=	좌변이 작거나 같다.
=	좌변과 우변의 값이 같다.
!=	좌변과 우변의 값이 다르다.
==	좌변과 우변의 값과 타입이 같다.
!=	좌변과 우변의 값이나 타입이 다르다.

## 비교 연산자

---

### ❖ 일치 연산자

- 자료형이 다른 것을 확실하게 구분 짓고 싶을 때 사용
  - 예제 코드의 결과는 모두 `false`

연산자	설명
<code>===</code>	양 변의 자료형과 값이 일치합니다.
<code>!==</code>	양 변의 자료형과 값이 다릅니다.

## 비교 연산자

---

### ❖ 일치 연산자의 용도

- 자료형이 다른 것을 확실하게 구분 짓고 싶을 때 사용

```
> '' == false  
true  
> '' === false  
false
```

```
> '' == 0  
true  
> '' === 0  
false
```

```
> '273' == 273  
true  
> '273' === 273  
false
```



## 비교 연산자

---

### ❖ 03\_2\_01\_compare.html

```
<body>
  <script>
    var age = prompt("당신의 나이를 입력하세요", 18);
    if (age < 19) {
      document.write("이 동영상을 볼 자격이 없습니다.");
    } else {
      document.write("즐겁게 보세요. ");
    }
  </script>
</body>
```

## 비교 연산자

---

### ❖ 문자열 비교

"korea" > "america" : true

"Korea" > "america" : false

"한글" > "english" : true

"15" > "12" : true

"015" > "12" : false

"9" > "12" : true

Number("9") > Number("12") : false

"015"

"12"



대응되는 자리끼리 비교한다.

## 비교 연산자

---

### ❖ 다른 데이터 타입의 비교

- 동일 데이터 타입으로 변환한 후 비교

```
"015" > 12;           // true
9 > "12";              // false
"9" > 12;              // false
```

## 비교 연산자

### ❖ 03\_2\_02\_stringcompare.html

```
<body>
<script>
    document.write('"korea" > "america" : ' +
                    ("korea" > "america") + "<br>");
    document.write('"Korea" > "america" : ' +
                    ("Korea" > "america") + "<br>");
    document.write('"한글" > "english" : ' +
                    ("한글" > "english") + "<br>");
    document.write('"15" > "12" : ' + ("15" > "12") + "<br>");
    document.write('"015" > "12" : ' + ("015" > "12") + "<br>");
    document.write('"9" > "12" : ' + ("9" > "12") + "<br>");
    document.write('Number("9") > Number("12") : ' +
                    (Number("9") > Number("12")) + "<br>");
</script>
</body>
```

## 비교 연산자

---

### ❖ 03\_2\_03\_equal.html

```
<body>
  <script>
    a = 2;
    b = "2";
    if (a == b) {
      document.write("== 비교 : 같음<br>");
    } else {
      document.write("== 비교 : 다름<br>");
    }
    if (a === b) {
      document.write("=== 비교 : 같음<br>");
    } else {
      document.write("=== 비교 : 다름<br>");
    }
  </script>
</body>
```

## 비교 연산자

---

### ❖ 삼항 연산자

조건 ? 참값:거짓값

## 비교 연산자

---

### ❖ 03\_2\_04\_question.html

```
<body>
  <script>
    var a = 3;
    var b = (a % 2 == 0) ? "짝":"홀"
    document.write("a는 " + b + "수이다.<br>");
  </script>
</body>
```

## 비교 연산자

---

### ❖ 03\_2\_04\_question2.html

```
<body>
  <script>
    a = 2;
    b = "2";
    document.write("== 비교 : " + (a == b ? "같음":"다름") +
                    "<br>");
  </script>
</body>
```



# 비교 연산자

---

## ❖ 논리 연산자

연산자	설명
!	논리 부정 연산자
	논리합 연산자
&&	논리곱 연산자

## 비교 연산자

---

### ❖ 03\_2\_05\_shortcircuit.html

```
<body>
  <script>
    var age = 23;
    if (age > 19 && confirm("술을 잘 마십니까?") == true) {
      document.write("술집에 입장하십시오.");
    } else {
      document.write("집에 가서 쉬세요.");
    }
  </script>
</body>
```

## 비교 연산자

---

### ❖ 03\_2\_05\_shortcircuit2.html

```
<body>
  <script>
    var age = 15;
    if (confirm("술을 잘 마십니까?") == true && age > 19) {
      document.write("술집에 입장하십시오.");
    } else {
      document.write("집에 가서 쉬세요.");
    }
  </script>
</body>
```

# 연산자

## - 고급 연산자 -

## 고급 연산자

### ❖ 비트 연산자

연산자	설명
&	둘 다 1일 때 1이다.
	둘 중 하나라도 1이면 1이다.
^	두 값이 달라야 1이다.
~	모든 비트를 반전시킨다.

- <<, >>
  - 비트 이동 연산자

### ❖ 03\_3\_01\_shift.html

```
<body>
  <script>
    var color=0x123456;
    var green = (color & 0x00ff00) >> 8;
    document.write("green = " + green.toString(16));
  </script>
</body>
```

## 고급 연산자

---

### ❖ 03\_3\_02\_xor.html

```
<body>
  <script>
    var a = 101092;
    a = a ^ 0xffffffff;
    document.write("한번 반전 : ", a, "<br>");
    a = a ^ 0xffffffff;
    document.write("다시 반전 : ", a);
  </script>
</body>
```

### ❖ 03\_3\_03\_shiftmulti.html

```
<body>
  <script>
    var a = 12;
    document.write(a << 1, "<br>");
    document.write(a >> 1, "<br>");
    var b = - 12;
    document.write(b >> 1, "<br>");
    document.write(b >>> 1, "<br>");
  </script>
</body>
```



## 고급 연산자

---

### ❖ typeof()

- 데이터 타입을 문자열로 출력
  - string
  - number
  - boolean
  - function
  - object
  - undefined

```
if (typeof(value) == "string") {  
}
```

```
var a = 12;
```

```
typeof(a)    // number
```

```
a = "대한민국";
```

```
typeof(a)    // string
```

## 고급 연산자

---

### ❖ 03\_3\_04\_typeof.html

```
<body>
  <script>
    var num = 1234;
    var str = "대한민국";
    var lunar = true;
    var func = function() {};
    var obj = { name:"김상형", age:29 };
    var notinit;

    document.write("num = " + typeof(num) + "<br>");
    document.write("str = " + typeof(str) + "<br>");
    document.write("lunar = " + typeof(lunar) + "<br>");
    document.write("func = " + typeof(func) + "<br>");
    document.write("obj = " + typeof(obj) + "<br>");
    document.write("notinit = " + typeof(notinit) + "<br>");
  </script>
</body>
```

# 고급 연산자

---

## ❖ 기타 연산자

- void
  - 피연산자를 무시하고 항상 undefined를 리턴

### ❖ 03\_3\_05\_void.html

```
<body>
  <script>
    var a = prompt("값을 입력하세요", "2");
    document.write("a = " + a + "<br>");
    var a = void prompt("값을 입력하세요", "2");
    document.write("a = " + a + "<br>");
  </script>
</body>
```

## 고급 연산자

---

### ❖ , 연산자

- 왼쪽에서 오른쪽으로 순서대로 평가한 후 마지막(가장 오른쪽) 값을 리턴

## 고급 연산자

---

### ❖ 03\_3\_06\_ commahtml

```
<body>
  <script>
    var a, b, c;
    c = (a = 5, b = 6);
    document.write("a = " + a + ", b = " + b + ", c = " + c);
  </script>
</body>
```

## 고급 연산자

---

### ❖ 03\_3\_07\_priority.html

```
<body>
  <script>
    var a = 2 + "3";
    var b = 1 + 2 + "3";
    document.write('2 + "3" = ' + a + "<br>");
    document.write('1 + 2 + "3" = ' + b + "<br>");
  </script>
</body>
```