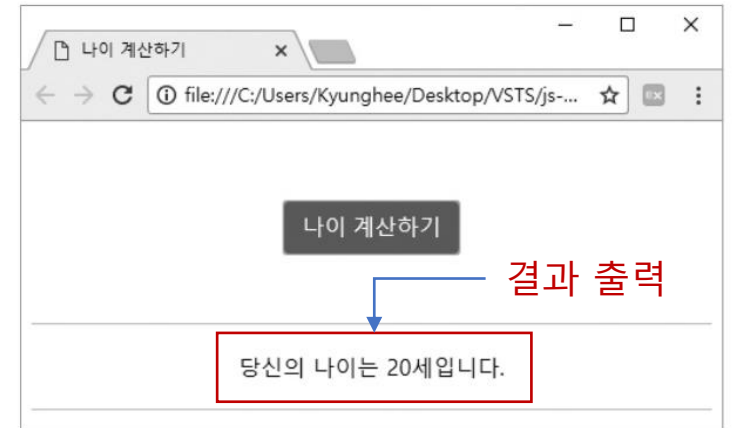
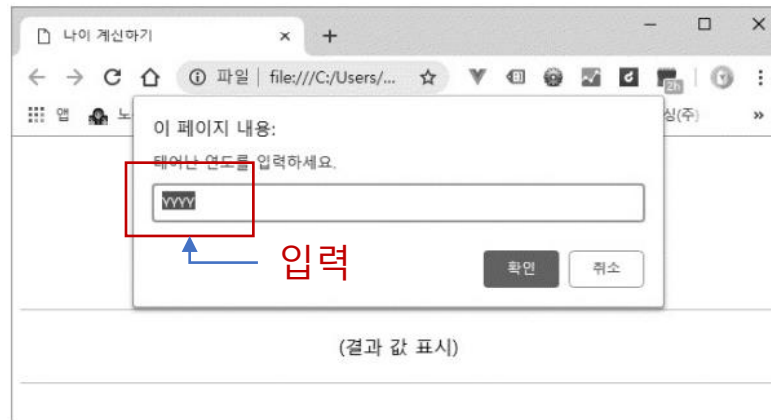
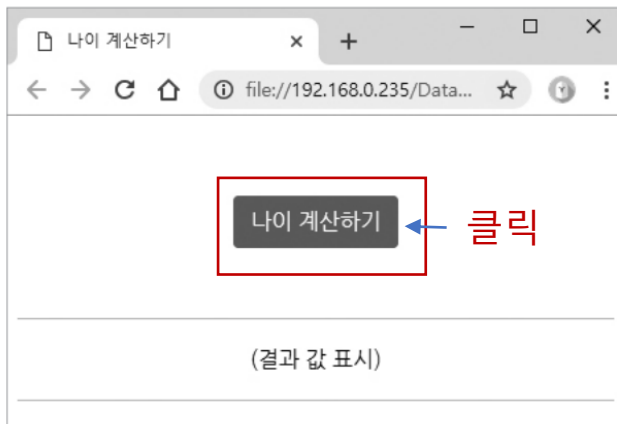


# 03. 변수와 자료형 그리고 연산자

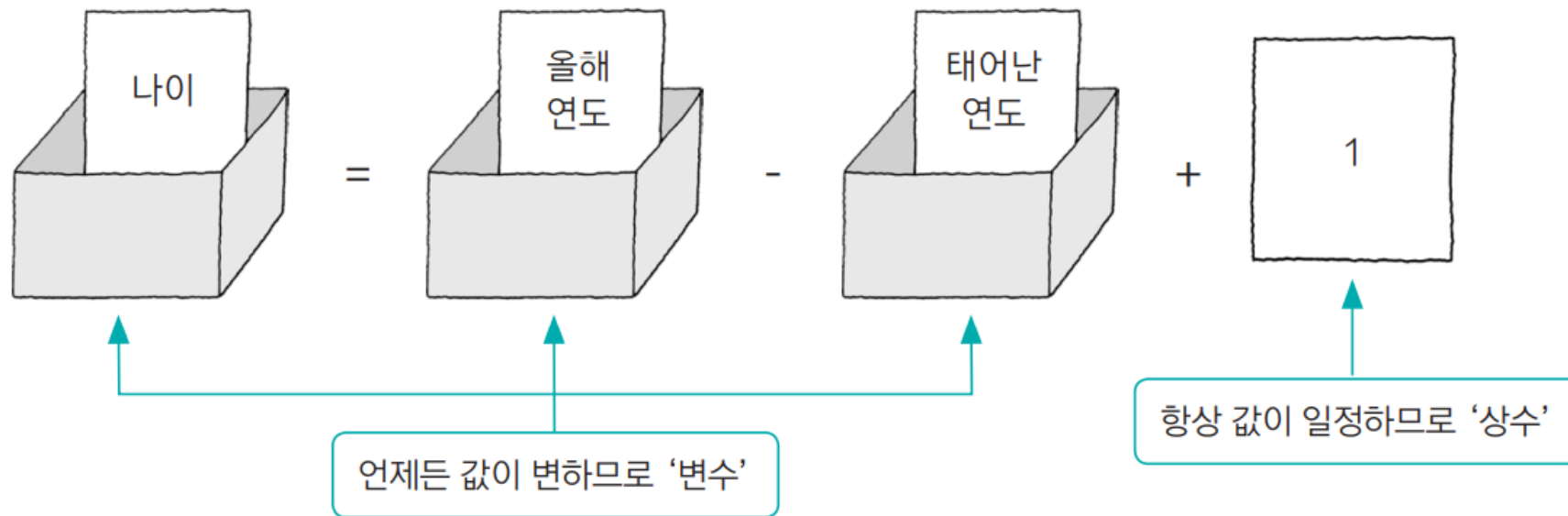
# [미리보기] 나이 계산 프로그램

- 올해 연도와 태어난 연도를 입력하면 사용자의 나이를 계산



# 변수란 무엇일까

- 변할 수 있는 값 (ex. 올해 연도, 태어난 연도, 나이)



# 변수란 무엇일까

## 변수를 선언하는 규칙 세 가지

- 이름은 의미있게 짓는다
- 여러 단어를 연결한 변수 이름을 낙타 모양으로 만들어 준다 (ex. currentYear, birthYear)
- 변수 이름의 첫 글자는 반드시 문자나 밑줄(\_), 달러 기호(\$)로 시작해야 한다.

(ex. Friday\_13, \$Friday)

## 변수 선언과 값/식 할당

- ① var 다음에 변수 이름을 적어서 변수를 선언하고,
- ② 변수 오른쪽에 = 기호를 붙이고 오른쪽에 저장할 값이나 식을 작성

※ 변수 선언과 값 할당을 동시에 할 수도 있음

# 변수란 무엇일까?

실습 파일 : 03Wage.html

나이 계산 프로그램 - 변수

```
13  <script>
14      function calc( ) {
15          var currentYear = 2019;
16          var birthYear = 1990;
17          var age;
18          age = currentYear - birthYear + 1;
19          document.querySelector("#result").innerHTML = "당신의 나이는 " + age + "세입니다.";
20      }
21  </script>
```

적절하게 수정해 보세요

# 변수란 무엇일까?

나이 계산 프로그램 - 변수

실습 파일 : 03Wage.html

```
13  <script>
14      function calc( ) {
15          var currentYear = 2019;
16          var birthYear = prompt("태어난 연도를 입력하세요.", "YYYY"); // 프롬프트 창에 태어난 연도를
                                입력하게 합니다.
17          var age;
18          age = currentYear - birthYear + 1;
19          document.querySelector("#result").innerHTML = "당신의 나이는 " + age + "세입니다.";
20      }
21  </script>
```

# 자료형 이해하기

- 자료형이란 컴퓨터가 처리하는 자료의 형태

## 자료형의 종류

자료형		설명
기본형	number(숫자)	따옴표 없이 표기한 숫자를 나타냅니다.
	string(문자열)	작은따옴표(')나 큰따옴표(")로 묶어 나타냅니다.
	boolean(논리형)	참(true)과 거짓(false)이란 두 가지 값만 가지고 있는 유형입니다.
	undefined	자료형을 지정하지 않았을 때의 유형입니다. 예를 들어 변수를 선언만 하고 값을 정의하지 않으면 undefined가 됩니다.
	null	값이 유효하지 않을 때의 유형입니다.
복합형	array(배열)	하나의 변수에 여러 값을 저장하는 유형입니다.
	object(객체)	함수와 속성이 함께 포함된 유형입니다.

# 자료형 이해하기

- 자료형 검사

- typeof 연산자

```
> typeof('문자열')  
"string"      ← 문자열을 의미  
typeof(273)  
"number "    ← 숫자를 의미  
> typeof(true)  
"boolean"
```

- boolean, undefined, function, object, symbol, bigint라는 8가지 중에 하나를 출력



# 자료형 이해하기

## 숫자형

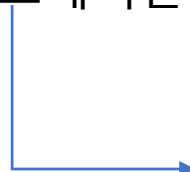
### 1) 정수

- 소수점 없는 숫자
- 표현 방법에 따라 10진수, 8진수, 16진수

```
> var birthYear = 2000;  
< undefined  
> typeof birthYear;  
< "number"
```

### 2) 실수

- 소수점이 있는 숫자
- 자바스크립트에서는 정밀한 실수 계산을 못 함



```
> 0.1 + 0.2  
< 0.30000000000000004
```

```
> var average = 17.5;  
< undefined  
> typeof average;  
< "number"
```

# 자료형 이해하기

## 문자형

- 작은 따옴표(' ') 나 큰 따옴표(" ")로 묶은 자료
- 숫자도 따옴표로 묶으면 문자형이 됨.
- 따옴표 안에 따옴표를 넣어야 할 경우 ' ' ' ' , 또는 " " " " 처럼 사용

```
> var message = "두잇공부중";  
< undefined  
> typeof message;  
< "string"
```

- 특수 문자
  - 이스케이프\ : 따옴표를 문자 그대로 사용해야 할 때
  - \n: 줄바꿈    \t: 탭    \\: 역슬래시(\) 그 자체를 의미
- 문자열 연산자
  - 숫자 자료와 마찬가지로 문자열도 기호를 사용해서 연산 처리

---

```
> '가나다' + '라마' + '바사아' + '자차카타' + '파하'  
"가나다라마바사아자차카타파하"
```
- 문자 선택 연산자
  - 문자열 내부의 문자 하나를 선택

---

```
> '안녕하세요'[0]  
"안"  
> '안녕하세요'[1]  
"녕"  
> '안녕하세요'[2]  
"하"
```

# 자료형 이해하기

## 논리형

- 참(True)과 거짓(False)이라는 값을 표현하는 자료형
- 프로그램에서 조건을 확인할 때 많이 사용

```
> var age = 21;  
< undefined  
> age > 20  
> true  
> age < 20  
< false
```

```
> typeof true;  
< "boolean"  
> typeof false;  
< "boolean"
```

# 자료형 이해하기

## undefined

- 자료형이 정의되지 않았을 때의 상태
- 변수가 undefined라는 것은 '처음부터 변수에 값이 할당되지 않았다'는 의미

## null

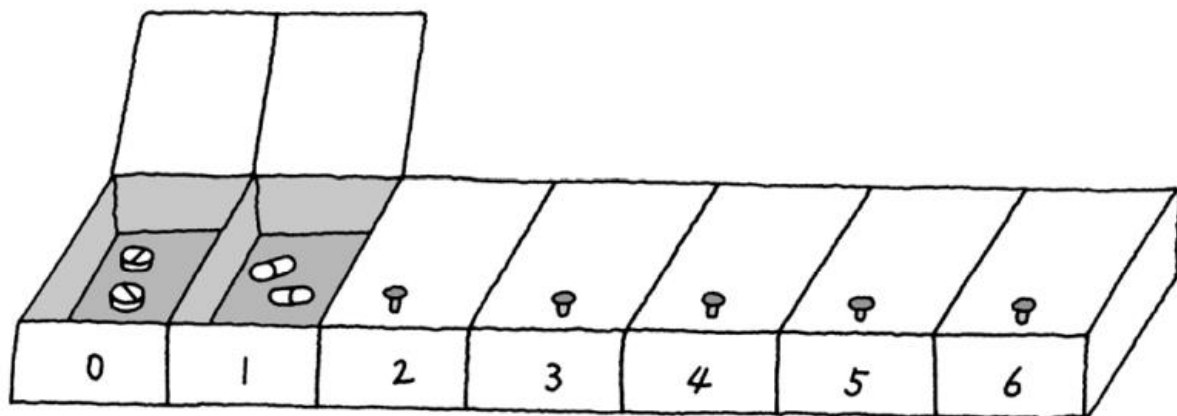
- '처음에 할당된 값이 더 이상 유효하지 않다'는 의미

```
> var first;  
< undefined  
> var second = 10;  
< undefined  
> second = null;  
< null
```

# 자료형 이해하기

## 배열

- 하나의 변수에 여러 값 저장
- 배열의 인덱스(index)는 0부터 시작
- 배열에 있는 값을 가져오려면 배열 이름과 대괄호([ ]) 안에 인덱스 사용.



# 자료형 이해하기

## 배열

```
> var seasons = ["봄", "여름", "가을", "겨울"];  
> seasons  
< ["봄", "여름", "가을", "겨울"]  
▶ (4)
```

```
> seasons[0]  
< "봄"  
> seasons[1]  
< "여름"  
> seasons[2]  
< "가을"  
> seasons[3]  
< "겨울"
```

# 자료형 이해하기

## 객체

- 여러 자료를 중괄호({ })로 묶은 것
- 키(key)와 값(value)을 한 쌍으로 여러 자료 저장

```
> var kim = {  
    firstName: "John",  
    lastName: "Kim",  
    age: 35,  
    address: "Seoul"  
}
```

# 자료형 이해하기

## 자바스크립트 자료형의 특징

- 느슨한 자료형 체크(weak datatype check)
- 자바스크립트는 미리 변수의 자료형을 지정하지 않음
- 변수를 지정하고 원하는 값을 할당만 하면 됨



# 자료형 이해하기

- **템플릿 문자열**은 백틱(`) 기호로 감싸 만들
  - 문자열 내부에 **`\${...}`** 기호를 사용하여 표현식을 넣으면 표현식이 문자열 안에서 계산됨

---

```
> console.log(`표현식 273 + 52의 값은 ${273 + 52}입니다...!`)
```

---

표현식 273 + 52의 값은 **325**입니다...!

# 자료형 변환

- 문자열 입력
  - prompt(메시지 문자열, 기본 입력 문자열)

```
01 <script>
02 // 상수를 선언합니다.
03 const input = prompt('message', '_default')
04 // 출력합니다.
05 alert(input)
06 </script>
```

prompt() 함수는 사용자로부터 내용을 입력받아서 사용

이 페이지 내용:

message

이 페이지 내용:

message

이 페이지 내용:

안녕하세요.

이 페이지 내용:

message

이 페이지 내용:

안녕하세요.

# 자료형 변환

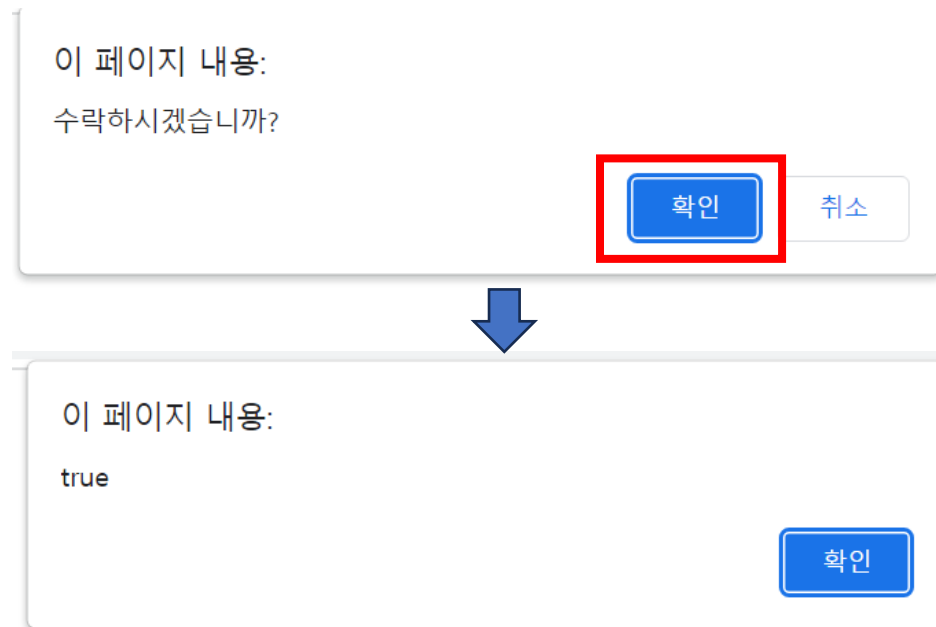
- 불 입력

- confirm() 함수는 prompt() 함수와 비슷한 형태로 사용

---

```
01 <script>
02  // 상수를 선언합니다.
03  const input = confirm('수락하시겠습니까?')
04
05  // 출력합니다.
06  alert(input)
07 </script>
```

---



- confirm() 함수를 사용하면 사용자에게 확인을 요구하는 메시지 창이 나타남
  - 사용자가 [확인] 버튼을 클릭하면 true를 리턴하고, [취소] 버튼을 클릭하면 false를 리턴

# 자료형 변환

- 숫자 자료형으로 변환하기

- 다른 자료형을 숫자 자료형으로 변환할 때는 Number() 함수를 사용

---

```
> Number("273")
```

```
273
```

```
> typeof(Number("273"))
```

```
"number"
```

—————→ 자료형은 숫자

- 다른 문자가 들어있어서 숫자로 변환할 수 없는 문자열의 경우, NaN(Not a Number)라는 값을 출력
    - NaN은 자바스크립트에서 숫자이지만, 숫자로 나타낼 수 없는 숫자를 의미
  - 숫자 연산자를 사용해 자료형 변환하기

---

```
> "52" - 0
```

```
52
```

```
> typeof("52" - 0)
```

```
"number"
```

```
> true - 0
```

```
1
```

```
> typeof(true - 0)
```

```
"number"
```

---

---

```
> 1 + true
```

```
2
```

```
> 1 + false
```

```
1
```

---

# 자료형 변환

- 문자열 자료형으로 변환하기

- 다른 자료형을 문자열 자료형으로 변환할 때는 String() 함수를 사용

---

```
> String(52.273)
"52.273"
> String(true)
"true"
> String(false)
"false"
```

---

숫자 자료형이 문자열 자료형으로 변환

불 자료형이 문자열 자료형으로 변환

- 문자열 연산자를 사용해 자료형 변환하기

- 문자열 연결 연산자(+)를 사용

---

```
> 273 + ""
"273"
> true + ""
"true"
```

---

빈 문자열을 연결해 문자열 자료형으로 변환

# 자료형 변환

- 불 자료형으로 변환하기

- 다른 자료형을 불 자료형으로 변환할 때는 Boolean() 함수를 사용

- 대부분의 자료는 불로 변환했을 때 true로 변환되나, 0, NaN, '' 혹은 ""(빈 문자열), null, undefined라는 5개의 자료형은 false로 변환됨

---

```
> Boolean(0)
```

```
false
```

```
> Boolean(NaN)
```

```
false
```

```
> Boolean("")
```

```
false
```

---

```
> Boolean(null)
```

```
false
```

```
> let 변수
```

```
undefined
```

```
> Boolean(변수)
```

```
false
```

---

- 논리 부정 연산자를 사용해 자료형 변환하기

- Boolean() 함수를 사용하지 않고 논리 부정 연산자(!)를 사용해서 다른 자료형을 불 자료형으로 변환
    - 불이 아닌 다른 자료에 논리 부정 연산자를 2번 사용하면 불 자료형으로 변환

# [혼자해보기] inch 단위를 cm 단위로 변경

- inch 단위를 cm 단위로 변환할 때, 2.54를 곱하기

```
1  <!DOCTYPE html>
2  <html>
3    <head>
4      <title></title>
5      <script>
6        // 숫자를 입력받습니다.
7        const rawInput = prompt('inch 단위의 숫자를 입력해주세요.')
8
9        // 입력받은 데이터를 숫자형으로 변경하고 cm 단위로 변경합니다.
10       const inch = Number(rawInput)
11       const cm = inch * 2.54
12
13       // 출력합니다.
14       alert(`${inch}inch는 ${cm}cm 입니다.`)
15     </script>
16   </head>
17   <body>
18   </body>
19 </html>
```

# 연산자

## 산술 연산자

분류	연산자 이름	기호	설명
사칙 연산자	더하기	+	두 값을 더합니다.
	빼기	-	앞의 값에서 뒤의 값을 뺍니다.
	곱하기	*	두 값을 곱합니다.
	나누기	/	앞의 값을 뒤의 값으로 나눕니다.
나머지 연산자	나머지	%	앞의 값을 뒤의 값으로 나눈 나머지 값을 구합니다.
증감 연산자	증가	++	변숫값을 1만큼 증가시킵니다.
	감소	--	변숫값을 1만큼 감소시킵니다.

### ※ 연산자와 피연산자

아래와 같은 식이 있을 때

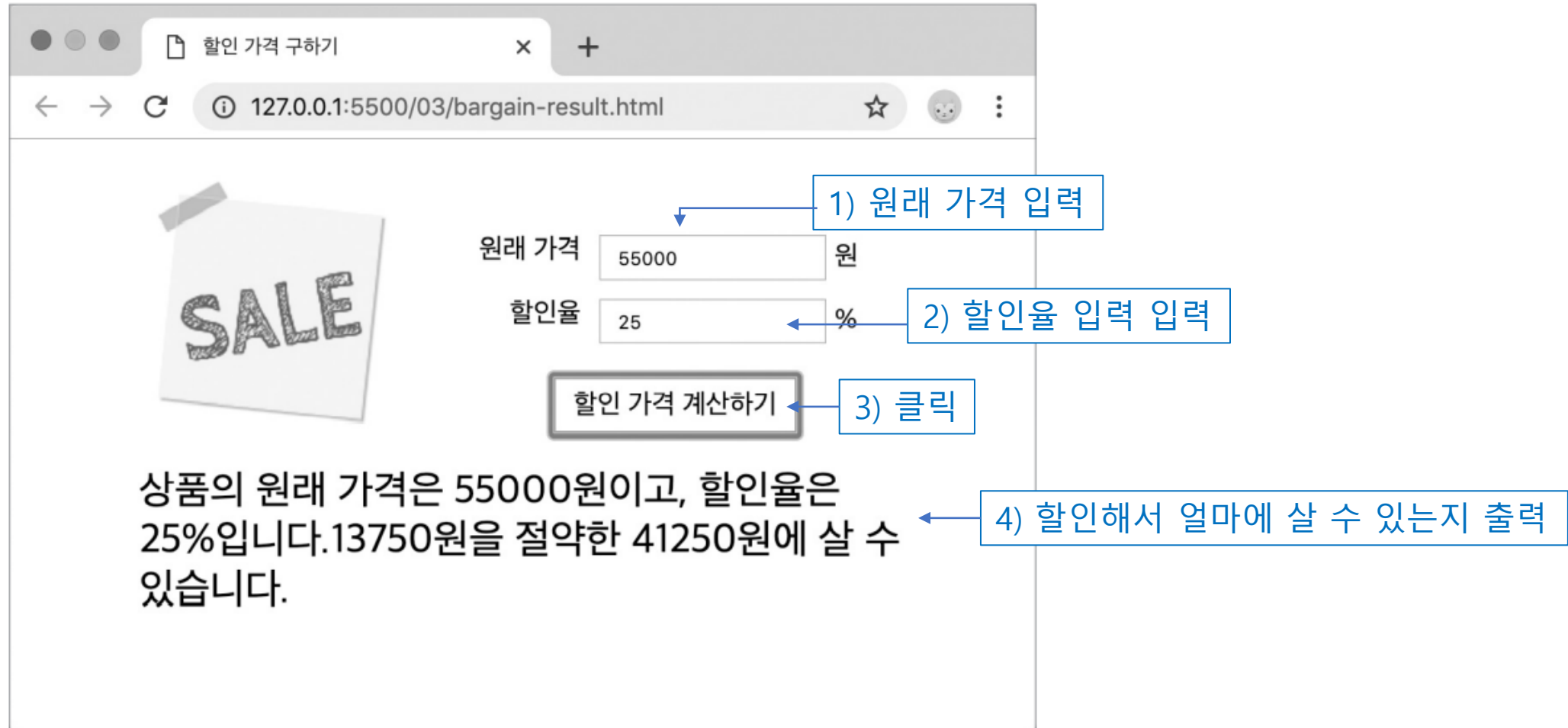
`currentYear - birthYear + 1`

currentYear, birthYear, 1은 연산 대상이 되기 때문에 '피연산자'라고 부름

피연산자를 제외한 더하기, 빼기 같은 것을 '연산자'라고 부름



# 할인 가격 계산하기



The screenshot shows a web browser window with the title '할인 가격 구하기' and the URL '127.0.0.1:5500/03/bargain-result.html'. The page content includes a 'SALE' sign icon, a form with two input fields, a calculation button, and a text output area. Four numbered annotations are present:

- 1) 원래 가격 입력: Points to the '원래 가격' input field containing '55000'.
- 2) 할인율 입력 입력: Points to the '할인율' input field containing '25'.
- 3) 클릭: Points to the '할인 가격 계산하기' button.
- 4) 할인해서 얼마에 살 수 있는지 출력: Points to the output text area.

원래 가격 55000 원

할인율 25 %

할인 가격 계산하기

상품의 원래 가격은 55000원이고, 할인율은 25%입니다. 13750원을 절약한 41250원에 살 수 있습니다.

# 할인 가격 계산하기

실습 파일 : bargain.css

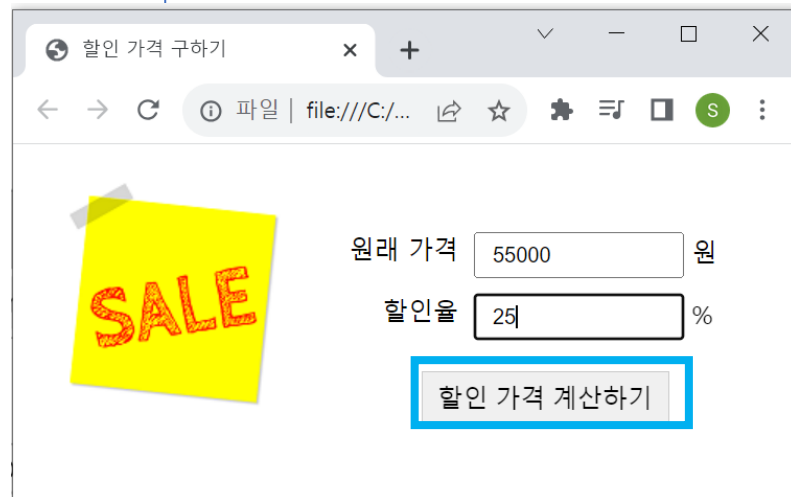
```
1  #contents {
2      width: 450px;
3      margin: 0 auto;
4  }
5  #contents > img {
6      float: left;
7      margin-right: 25px;
8  }
9  ul {
10     list-style: none;
11     padding-top: 40px;
12 }
13 li {
14     margin-bottom: 10px;
15 }
16 li > label {
17     width: 80px;
18     float: left;
19     text-align: right;
20 }
```

```
21  input[type="text"] {
22     width: 110px;
23     padding: 5px 10px;
24     margin: 0 5px 0 10px;
25 }
26  li:nth-child(3) {
27     margin-top: 20px;
28     text-align: center;
29 }
30  button {
31     padding: 5px 10px;
32     border: 1px solid #ccc;
33     font-size: 1em;
34 }
35  #showResult {
36     margin-top: 20px;
37     font-size: 1.5em;
38 }
```

# 할인 가격 계산 프로그램

실습 파일 : bargain.html

```
1  <!DOCTYPE html>
2  <html lang="ko">
3  <head>
4      <meta charset="UTF-8">
5      <meta name="viewport" content="width=device-width, initial-scale=1.0">
6      <meta http-equiv="X-UA-Compatible" content="ie=edge">
7      <title>할인 가격 구하기</title>
8      <link rel="stylesheet" href="css/bargain.css">
9  </head>
10 <body>
11     <div id="contents">
12         
13         <ul>
14             <li>
15                 <label for="oPrice">원래 가격</label>
16                 <input type="text" id="oPrice">원
17             </li>
18             <li>
19                 <label for="rate">할인율</label>
20                 <input type="text" id="rate">%
21             </li>
22             <li>
23                 <button onclick="showPrice()">할인 가격 계산하기</button>
24             </li>
25         </ul>
26         <div id="showResult"></div>
27     </div>
28 </body>
29 </html>
```



# 할인 가격 계산 프로그램

- </body> 종료태그 위에 코드 삽입

실습 파일 : [bargain.html](#)

```
29  <script>
30      function showPrice( ) {
31          var originPrice = document.querySelector("#oPrice").value;
32          var rate = document.querySelector("#rate").value;
33          var savedPrice = originPrice * (rate / 100);
34          var resultPrice = originPrice - savedPrice;
35          document.querySelector("#showResult").innerHTML = "상품의 원래 가격은 " + origin-
Price + "원이고, 할인율은 " + rate + "%입니다. " + savedPrice + "원을 절약한 " + resultPrice
+ "원에 살 수 있습니다.";
36      }
37  </script>
```

# 연산자

## 할당 연산자

- 변수에 값을 할당하는 연산자
- 사칙 연산자와 조합해서 사용할 수 있음

할당 연산자 응용	예	의미
<code>+=</code>	<code>y += x</code>	<code>y = y + x</code>
<code>-=</code>	<code>y -= x</code>	<code>y = y - x</code>
<code>*=</code>	<code>y *= x</code>	<code>y = y * x</code>
<code>/=</code>	<code>y /= x</code>	<code>y = y / x</code>
<code>%=</code>	<code>y %= x</code>	<code>y = y % x</code>

## 연결 연산자

- 문자열과 문자열을 연결
- `+` 기호 사용

## 형변환

- 숫자형과 문자형을 더하면 숫자를 문자열로 인식함
- 곱하기나 나누기, 나머지 연산에서는 문자형 자료를 모두 숫자로 자동 인식함

# 연산자

## 증감 연산자

증감 연산자	설명
변수++	기존의 변수 값에 1을 더하기(후위)
++변수	기존의 변수 값에 1을 더하기(전위)
변수--	기존의 변수 값에 1을 빼기(후위)
--변수	기존의 변수 값에 1을 빼기(전위)

```
1  <!DOCTYPE html>
2  <html>
3    <head>
4      <title></title>
5      <script>
6        // 변수를 선언합니다.
7        let number = 10
8
9        // 연산자를 사용합니다.
10       number++
11
12       // 출력합니다.
13       alert(number)
14     </script>
15   </head>
16   <body>
17   </body>
18 </html>
19
```

# 연산자

## 비교 연산자

두 값을 비교하여 true나 false로 결과값을 내놓는 연산자

== , !=	두 개의 값이 같은지, 같지 않은지 확인
< , <=	왼쪽 값이 오른쪽 값보다 작은지 혹은 작거나 같은지 확인
> , >=	왼쪽 값이 오른쪽 값보다 큰지 혹은 크거나 같은지 확인
=== , !==	두 개의 값이 자료형까지 완벽하게 같은지, 같지 않은지 확인

# 연산자

## 논리 연산자

true와 false를 피연산자로 연산자

부울 연산자, 또는 불리언 연산자라고도 함

<b>OR 연산자</b>		두 개의 값 중 하나만 true이면 결과값이 true
<b>AND 연산자</b>	&&	두 개의 값 모두 true여야 결과값이 true
<b>NOT 연산자</b>	!	true와 false 값을 반대로




# 연산자

## 삼항 조건 연산자

조건식 ? 자바스크립트 코드 1 : 자바스크립트 코드 2;

-조건식이 true 이면 자바스크립트 코드 1 실행, 조건식이 false이면 자바스크립트 코드 2 실행

```
<script>  
  var a = 10;  
  var b = 3;  
  var result = a > b ? "javascript" : "hello";  
  document.write(result); //javascript  
</script>
```



# 연산자 우선 순위

단항 연산자 → 산술 연산자 → 비교 연산자 → 논리 연산자 → 할당 연산자

	1st	2nd	3rd	4th	5th	6th	7th
단항 연산자	!	++	--				
산술 연산자	*	/	%	+	-		
비교 연산자	<	<=	>	>=	==	!=	===
논리 연산자	&&						
할당 연산자	=	+=	-=	*=	/=	%=	

# 정리

1. 다음 중 상수를 선언할 때 사용하는 키워드는 어떤 것인가?

① const    ② let    ③ var    ④ comment

2. 다음 중 값을 할당할 때 사용하는 연산자는 어떤 것인가?

① :=    ② =    ③ <=    ④ =>

3. 다음 연산자들의 피연산자가 어떤 자료형인지 적어 보기

연산자	피연산자 자료형
+(문자열 연결 연산자)	문자열
+(덧셈 연산자)	숫자
&&	
-	
*	

# 정리

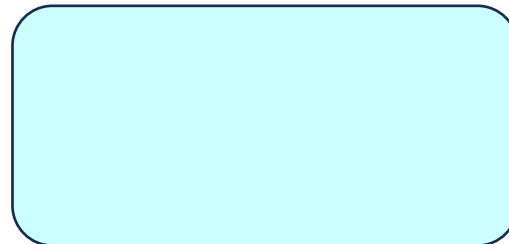
## 2. 다음 프로그램의 실행 결과를 예측해 보기

---

```
<script>  
  console.log("# 연습문제")  
  console.log("\\\\\\\\\\\\")  
</script>
```

---

실행결과



## 3. 다음 프로그램의 실행 결과를 예측

---

```
<script>  
  console.log("안녕하세요"[1])  
  console.log("안녕하세요"[2])  
  console.log("안녕하세요"[3])  
  console.log("안녕하세요"[4])  
</script>
```

---

실행결과



# 정리

4. 다음 프로그램의 실행 결과를 작성하시오.

---

```
<script>
  console.log(2 + 2 - 2 * 2 / 2 * 2)
  console.log(2 - 2 + 2 / 2 * 2 + 2)
</script>
```

---

실행결과

