



AhnLab SEM

2022. 3. 16 (chapter. 02)

- 자바스크립트 기본 -

인선미

2. 자료와 변수

- 1) 기본 자료형
- 2) 상수와 변수
- 3) 자료형 변환



기본 자료형



기본 자료형

자료(data)

- 프로그래밍에서 프로그램이 처리할 수 있는 모든 것

자료형(data type)

- 자료 형태에 따라 나눠 놓은 것
- 숫자(number), 문자열(string), 불(Boolean)

기본 자료형



문자열 자료형

- 자바스크립트에서는 문자가 하나든 여러 개든 모두 문자열 자료형

```
<script>  
  console.log("안녕하세요");  
  console.log('안녕하세요');  
</script>
```



기본 자료형

문자열 자료형

- 이스케이프 문자
- \n: 줄바꿈
- \t: 탭
- \\: 역슬래시(\) 그 자체를 의미

기본 자료형

문자열 연산자

- 문자열 + 문자열
- 문자열[숫자]

> '가나다'+'라마바'

< '가나다라마바'

> '안녕하세요'[2]

< '하'

기본 자료형



문자열 길이 구하기

- 문자열.length

```
> '안녕하세요'.length
```

```
< 5
```

```
> '자바스크립트 반가워요'.length
```

```
< 11
```

```
> ''.length
```

```
< 0
```



기본 자료형

숫자 자료형

- 소수점이 있는 숫자와 없는 숫자를 모두 같은 자료형으로 인식
- 숫자 연산자
 - + : 더하기 연산자
 - - : 빼기 연산자
 - * : 곱하기 연산자
 - / : 나누기 연산자
 - % : 나머지 연산자

기본 자료형

불 자료형

- 참과 거짓 값을 표현할 때 불 자료형을 사용
- true, false
- 비교 연산자

=== 양쪽이 같다

!== 양쪽이 다르다

> 왼쪽이 더 크다

< 오른쪽이 더 크다

>= 왼쪽이 더 크거나 같다

<= 오른쪽이 더 크거나 같다

기본 자료형

불 논리합/논리곱 연산자

- && 논리곱 연산자
- || 논리합 연산자

기본 자료형

자료형 검사

- typeof 연산자

```
> typeof('123')
```

```
<> 'string'
```

```
> typeof(123)
```

```
<> 'number'
```

```
> typeof(true)
```

```
<> 'boolean'
```

기본 자료형

템플릿 문자열

- 백틱(`) 기호로 감싸 만들기
- 문자열 내부에 ``${...}`` 기호를 사용하여 표현식을 넣으면 표현식이 문자열 안에서 계산됨

```
> console.log(`표현식 123 + 45 의 값은 ${123+45}입니다.`)  
표현식 123 + 45 의 값은 168입니다. VM876:1
```



상수와 변수



상수와 변수



상수



- 항상 같은 수(값에 이름을 한 번 붙이면 값을 수정할 수 없음)
- `const` 키워드 사용
- **const** 이름 = 값

```
> const pi = 3.141592
```

```
< undefined
```

```
> pi
```

```
< 3.141592
```

```
> const r = 10
```

```
< undefined
```

```
> 2*pi*r
```

```
< 62.83184
```

EM

상수와 변수



상수



- Identifier has already declared(구문 오류)
- 특정한 이름의 상수는 한 파일에서 한 번만 선언
- 만약 같은 이름으로 상수를 한 번 더 선언하면 다음과 같은 오류를 발생

Uncaught SyntaxError: Identifier 'name' has already been declared

```
<script>  
  const name = "name이라는 이름의 상수를 선언해볼게요."  
  const name = "한 번 더 선언해볼게요."  
</script>
```

상수와 변수

상수

- Missing initializer in const declaration(구문 오류)
- 상수는 한 번만 선언할 수 있으므로 선언할 때 반드시 값을 함께 지정해줘야 함
- 만약 상수를 선언할 때 값을 지정해주지 않는다면 다음과 같은 오류를 발생

Uncaught SyntaxError: Missing initializer in const declaration

```
> const pi
```

✖ Uncaught SyntaxError: Missing initializer in const VM1189:1
declaration

상수와 변수



상수

- Assignment to constant variable(예외 처리)
- 한 번 선언된 상수의 자료는 변경할 수 없음

TypeError: Assignment to constant variable

```
> const name = "name이라는 이름의 상수를 선언합니다"
< undefined
```

```
> name = "그 값을 변경합니다"
```

✖ ▶ Uncaught TypeError: Assignment to constant variable.
at <anonymous>:1:6

VM1448:1

상수와 변수



변수



- 데이터를 담을 수 있는 공간
- 변수에 값을 대입할 때 데이터 타입이 결정됨
- let 키워드를 사용해 변수 선언
- **let** 이름 = 값

```
> let pi = 3.141592
```

```
< undefined
```

```
> let r = 10
```

```
< undefined
```

```
> pi * r * r
```

```
< 314.1592
```


상수와 변수



변수



- Identifier has already been declared(구문 오류)
- 상수와 마찬가지로 특정한 이름의 변수는 한 파일에서 한 번만 선언
- 만약 같은 이름으로 변수를 한 번 더 선언하면 다음과 같은 오류를 발생

Uncaught SyntaxError: Identifier 'name' has already been declared

```
<script>  
  let name = "name이라는 이름의 변수를 선언합니다";  
  let name = "한번 더 선언합니다";  
</script>
```

✖ Uncaught SyntaxError: Identifier 'name' has already been declared first.html:11

변수에 적용할 수 있는 연산자



복합 대입 연산자

- 대입 연산자와 다른 연산자를 함께 사용하는 연산자

복합 대입 연산자	설명	사용 예	의미
+=	기존 변수의 값에 값을 더하기	a += 1	a = a+1
-=	기존 변수의 값에 값을 빼기	a -= 1	a = a-1
*=	기존 변수의 값에 값을 곱하기	a *= 1	a = a*1
/=	기존 변수의 값에 값을 나누기	a /= 1	a = a/1
%=	기존 변수의 값에 나머지를 구하기	a %= 1	a = a%1



변수에 적용할 수 있는 연산자

복합 대입 연산자

```
> let value = 10
```

```
< undefined
```

```
> value += 10
```

```
< 20
```

```
> value -= 5
```

```
< 15
```


변수에 적용할 수 있는 연산자



증감 연산자

- 복합 대입 연산자를 약간 간략하게 사용한 형태

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)



변수에 적용할 수 있는 연산자

복합 대입 연산자

```
<script>
    //변수 선언
    let number = 10;
    //변수 출력
    alert(number++);
    alert(number);
    number += 1;
    alert(number);
    alert(++number);
    number++;
    alert(number);
</script>
```

간단한 입출력 정리



알림창 출력

- alert(메시지)
- alert("안녕하세요?");

확인창 출력

- confirm(메시지)
- let reply = confirm("정말 배경 이미지를 바꾸겠습니까?");

콘솔창 출력

- console.log(메시지)

간단한 입출력 정리



프롬프트 창 입력

- prompt(메시지)
- prompt(메시지, 기본값)
- let name = prompt("이름을 입력하세요", "홍길동")

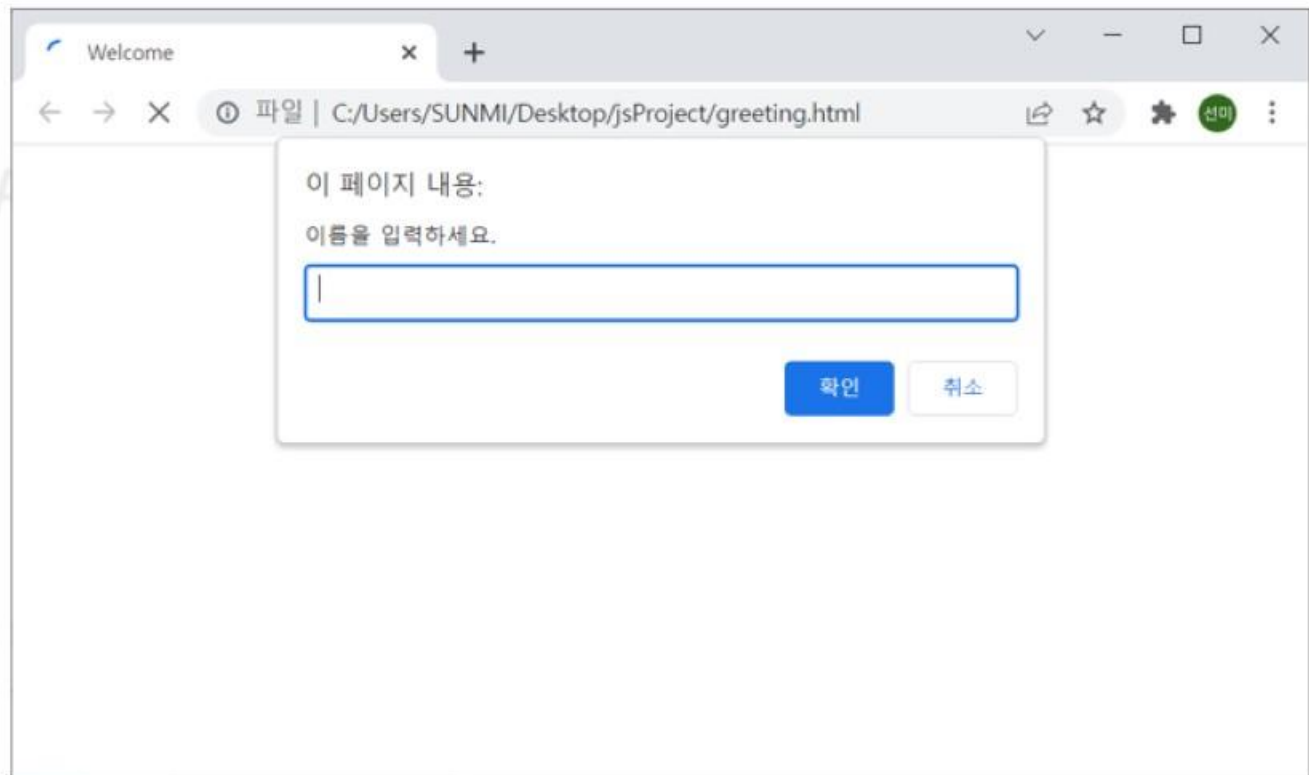
웹 브라우저 화면에 출력

- document.write(메시지)
- document.write("<h1>어서오세요</h1>")



미션> 이름 받아서 화면에 출력하기

- 사용자의 이름을 입력 받는다.
- 이름을 웹 브라우저 화면에 표시한다.
- 콘솔창에도 표시한다.



미션> 이름 받아서 화면에 출력하기



Welcome

← → × ① 파일 | C:/Users/SUNMI/Desktop/jsProject/greeti

이 페이지 내용:
이름을 입력하세요.

Welcome

← → ↻ ① 파일 | C:/Users/SUNMI/Desktop/jsProject/greeting.html

어서오세요

인선미님, 환영합니다.

Elements Console Sources >> | ⚙️ ⋮ ×

▶ ⏏ top 👁 Filter | 1 hidden ⚙️

Custom levels ▾ No Issues

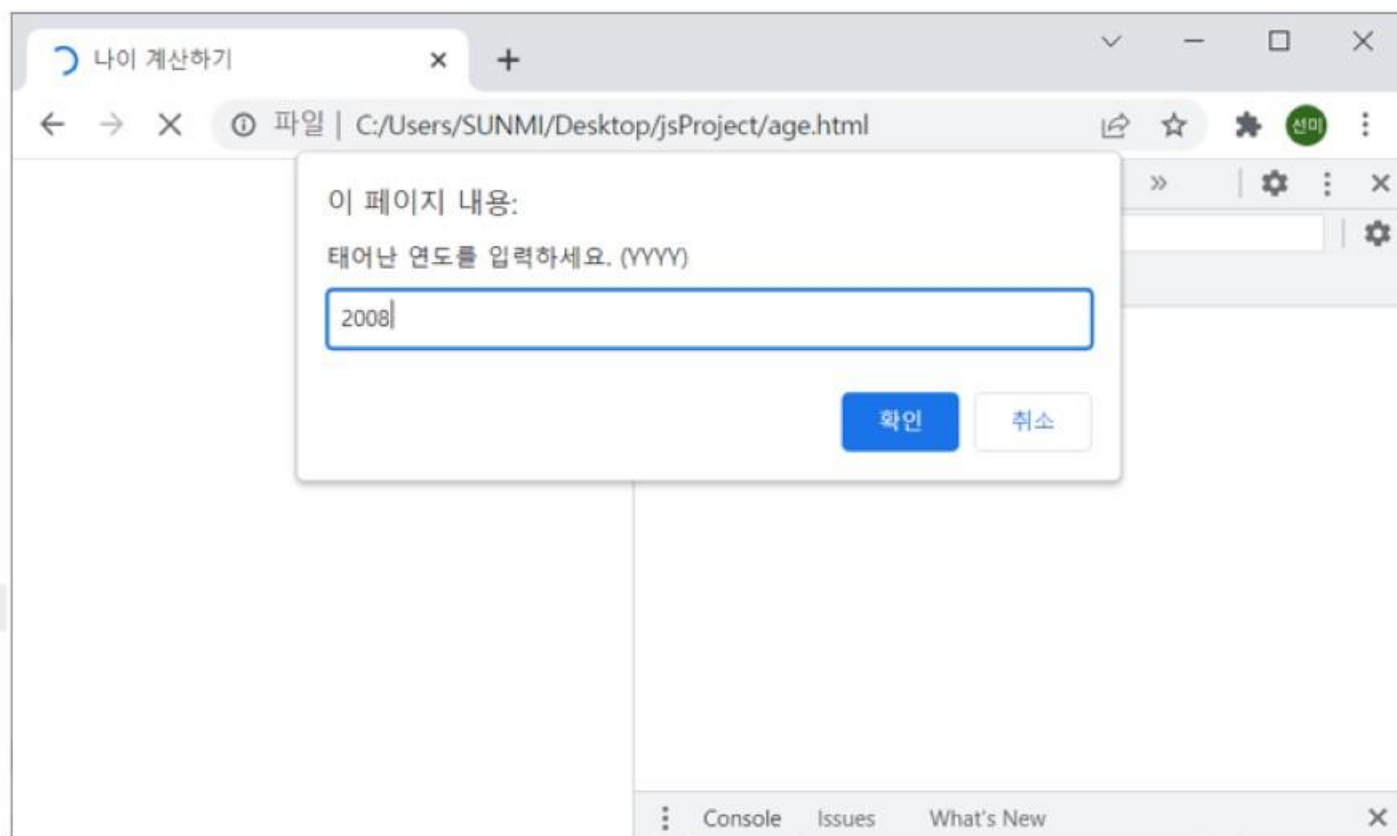
인선미님, 환영합니다. greeting.html:13

>

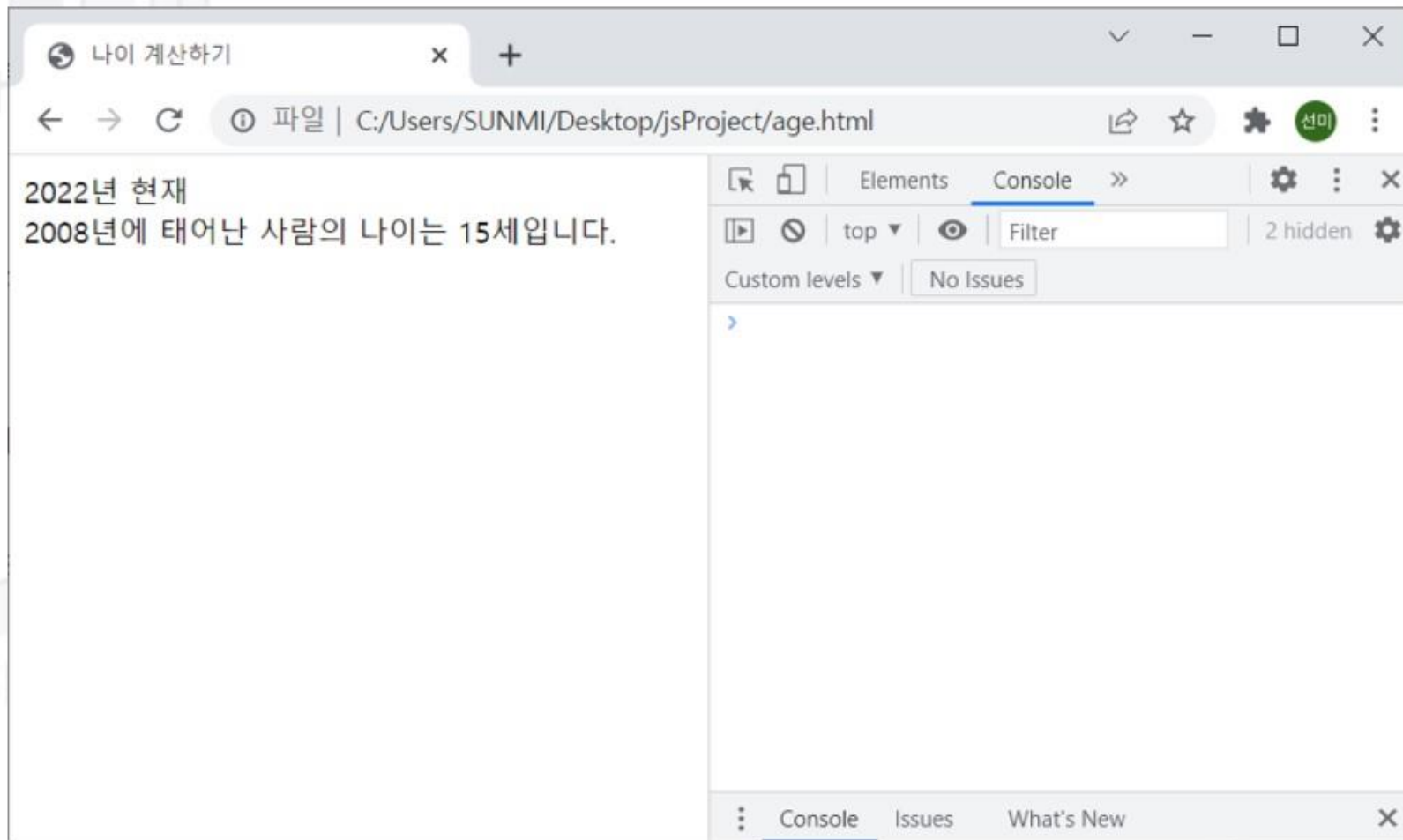
⋮ Console Issues What's New ×

미션> 나이 계산 프로그램 만들기

사용자로부터 태어난 연도를 입력받아 나이를 계산하는 코드를 작성하세요



미션> 나이 계산 프로그램 만들기



감사합니다.



AhnLab SEM