

# 자바스크립트

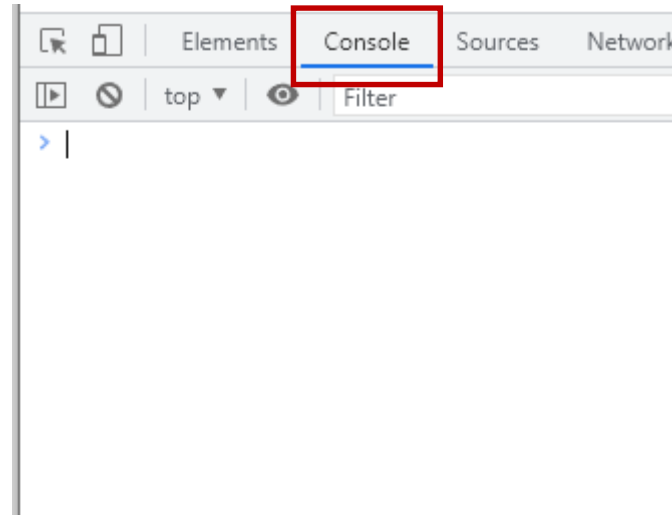
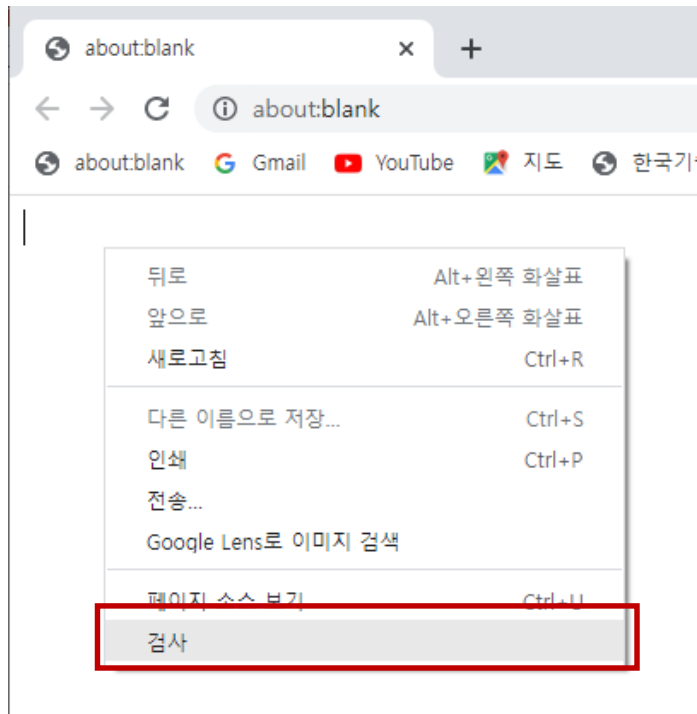
자바스크립트 기초

# 목차

- Hello World
- 사전지식
- Datatype
- 데이터의 입력
- 자료형의 변환

# Hello World

- Hello world를 브라우저에서 직접 실행
  - about:blank를 즐겨찾기로 추가
  - 개발자 도구 실행 → Console 탭 이동 → `console.log("Hello world")`



# Hello World

- Hello world를 html을 통해 실행
  - VSCODE 실행
  - 기본 HTML5 문서 작성
  - <head>안에 <script>구현
- 코드오류시 console탭에서 확인

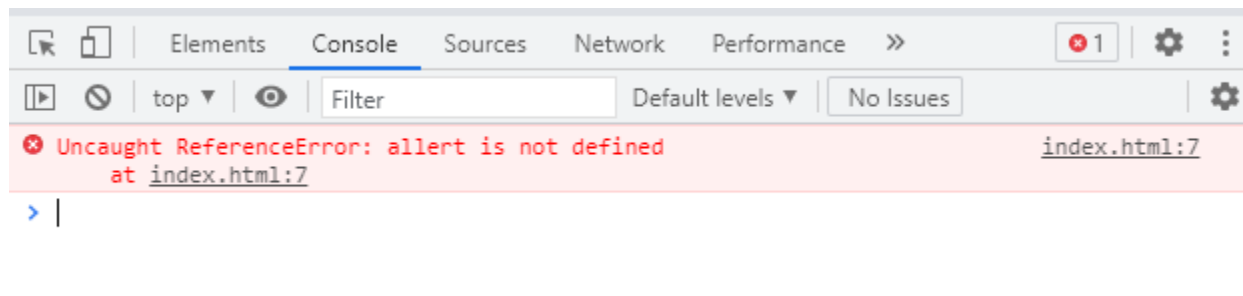
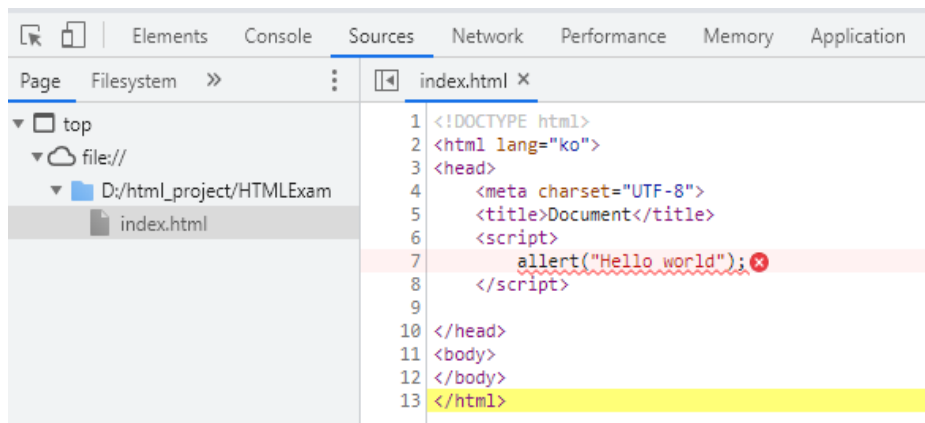
```
<!DOCTYPE html>
<html lang="ko">
<head>
  <meta charset="UTF-8">
  <title>Document</title>

  <script>
    alert("Hello world");
    console.log("Hello world")
  </script>
</head>
<body>

</body>
</html>
```

# Hello World

- 오류의 확인
  - 임의로 오류 발생
  - alert → allert
- 크롬 개발자 도구에서 확인



# Hello world

- 연습문제

- 아래 코드를 삽입해보고 그 결과를 살펴봅시다.

```
<!DOCTYPE html>
<html lang="ko">
<head>
  <meta charset="UTF-8">
  <title>Document</title>
</head>
<body>
  <script>
    document.body.innerHTML = "<h1>Hello world</h1>"
  </script>
</body>
</html>
```

- google에서 REPL이라는 용어를 찾아봅시다.

# 사전지식

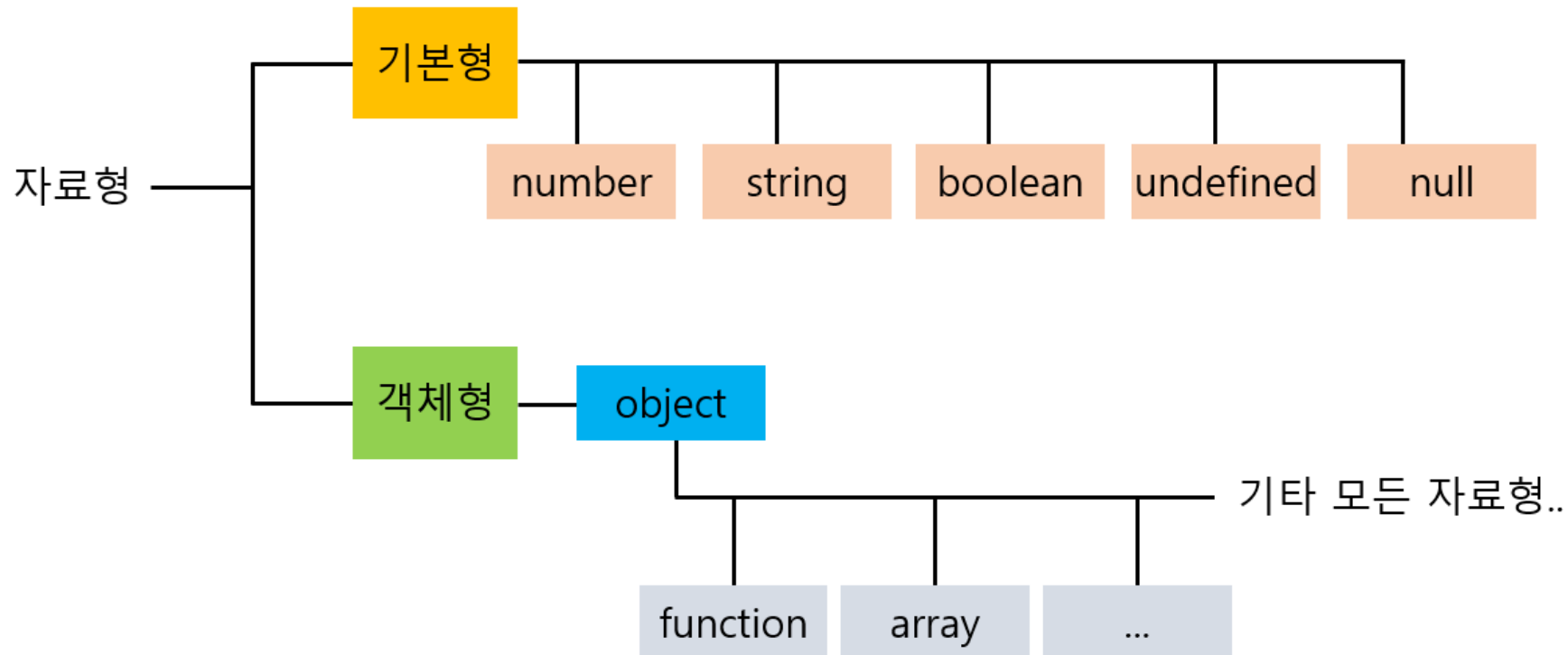
- Expression과 Statement
  - Expression : 값을 만들어 내는 코드
  - Statement : 표현식이 모인 문장, 또는 값을 만들지 않는 문장
  - 세미콜론은 ASI에 의해 자동으로 삽입되거나 명시적으로 붙여준다.
- 주석
  - // : <script> 영역에서 사용
  - /\*\*/ : <script> 영역에서 사용
  - <!-- --> : 비 <script> 영역에서 사용

\*\* ASI : Automatic semicolon insertion

# Datatype

- 자료형의 분류

- 기본형 (Primitive type)과 객체형 (Object type)으로 분류
- ES6에서 Symbol type이 기본형에 추가





# Datatype

- String type

- "string"와 'string'를 둘다 사용 (전통적으로 'string'을 사용)
- ES6에서 template string 기능이 추가되면서 `string` (backtick)도 사용
- 문자열 합연산 : 문자열 + 문자열
- 인덱스 연산 : [n]
- 길이 : .length

```
<body>
  <script>
    var str1 = "apple";
    var str2 = "banana";

    console.log(str1.length);
    console.log(str1[0]);
    console.log(str1[1]);
    console.log(str1 + str2);
    console.log(`i love ${str1} and ${str2}.`);
  </script>
</body>
```

# Datatype

- number type
  - 숫자형
  - 20, 20.5 등으로 표현, 정수형과 실수형의 구별을 하지 않음.
  - +, -, \*, /, % 연산 사용
- boolean 자료형 (bool자료형)
  - true, false
  - 논리연산자 적용
  - !, ==, !=, >, >=, <, <=
  - &&, ||
  - ==, != (피연산자의 자료형이 다를 경우 형변환이 발생)

# Datatype

- typeof()
  - 자료형의 검사
  - typeof(1) → 'number'
  - typeof("a") → 'string'

# Datatype

- 변수의 선언

- var : ES5의 선언 방식, 모든 변수를 var로 선언, 바꿀수 있음
- let : ES6의 선언 방식, 일반 변수, 값을 바꿀수 있음
- const : ES6의 선언 방식, 상수화 변수, 선언과 동시에 초기화

```
var str1 = 'apple';    // ES5
```

```
const str2 = 'banana' // ES5 상수화 변수
```

```
let str3 = 'pear'     // ES6 일반변수
```

\*\* var는 중복선언, 스코프 문제등으로 점차 사라지는 추세이며 구형코드에서 많이 사용

# 데이터의 입력

- 데이터의 입력
  - prompt(), 문자열 입력

```
<script>  
    const input = prompt('please input your message.', 'message');  
    alert(input);  
</script>
```

- confirm(), boolean의 입력

```
<script>  
    const input = confirm('Are you ok?');  
    alert(input);  
</script>
```

# 자료형 변환

- number형으로 변환

- Number(data)
- Number('123') → 123
- Number('abc') → NaN (Not a Number)
- "52" - 0 → 52
- true - 0 → 1

- String 형으로 변환

- String(data)
- String(54.22) → "54.22"
- String(true) → "true"

# 자료형 변환

- boolean형으로 변환
  - Boolean(data)
  - Boolean(0) → false
  - Boolean(NaN) → false
  - Boolean(null) → false

# 연습해봅시다.

- cm를 입력받아 inch를 출력하는 코드를 작성하시오
- 원의 반지름을 입력받아 원의 넓이를 구하는 코드를 작성하시오
- 사과, 복숭아, 딸기 를 unordered list로 출력하는 element를 자바스크립트로 만드는 코드를 작성하시오