

1강. 자바스크립트 개발환경 구축



1강_ 개발환경설정 및 기초문법

1

Javascript 시작하기

2

Javascript로 할 수 있는 일

3

개발 환경 구축

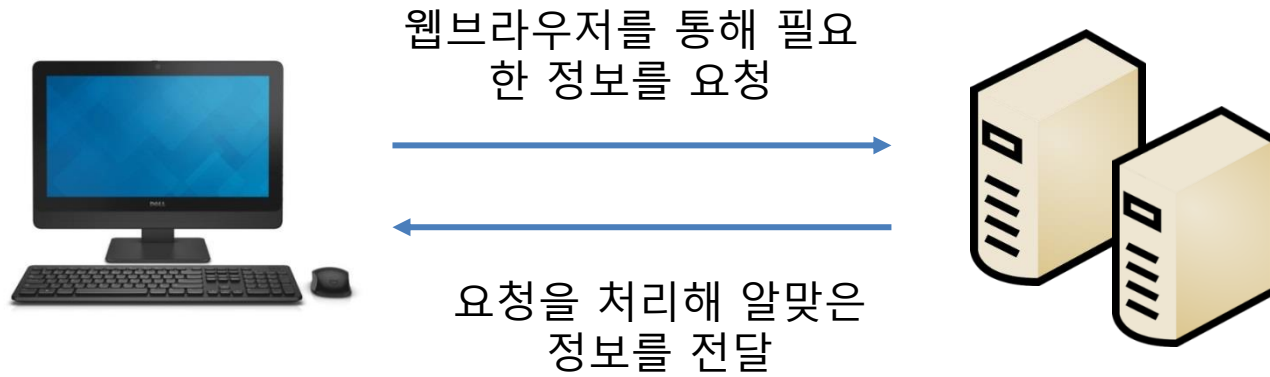
4

Javascript 기초 문법

웹 프로그래밍이란?

- 프로그래밍이란?
 - 컴퓨터 프로그램을 만드는 일
 - 프로그램이란? 컴퓨터에게 일을 시키는 명령의 집합
- 웹 프로그래밍이란?
 - 웹에 관련된 프로그램을 만드는 일
 - 웹브라우저와 관련된 프로그램을 작성하는 일
- 웹 프로그래밍의 구분
 - **프론트엔드(Front-end)** 개발
 - 서버에서 받아 온 정보를 웹 브라우저에 어떻게 보여줄 것인지를 프로그래밍
 - **백엔드(Back-end)** 개발
 - 서버에서 사용자의 요청을 처리하거나 데이터를 관리하는 프로그래밍

클라이언트와 서버



클라이언트(Client)측 언어

HTML

CSS

JavaScript

서버(Server)측 언어

Java(JSP)

파이썬

PHP

JavaScript

자바스크립트로 할 수 있는 일

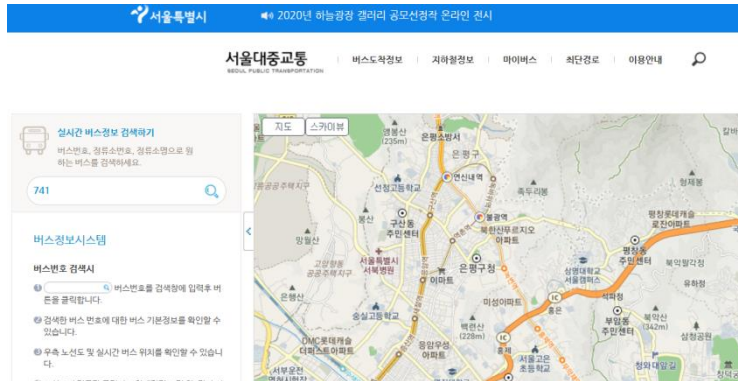
❖ JavaScript(자바스크립트)



- HTML에 소속된 프로그램으로 동적인 웹사이트 제작(웹 브라우저에서만 작동)

※ 자바스크립트의 탄생 – 1995년 네스케이프(netscape)의 브랜던 아이크가 창시

자바스크립트로 할 수 있는 일

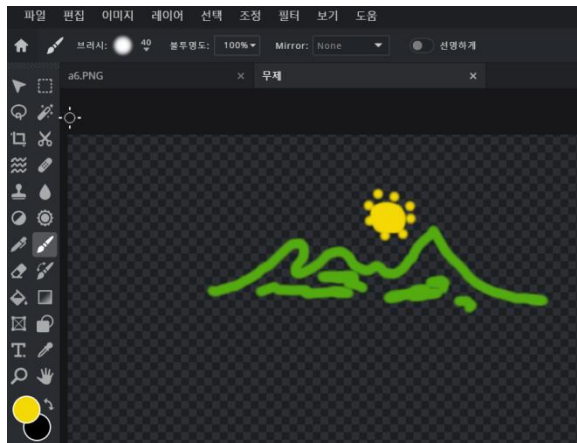


서울 대중교통 지도웹

- 웹 애플리케이션 제작
 - 지도관련, 웹 오피스 등
- 게임제작
- 서버용 프로그램 제작

node.js (자바스크립트 실행환경)로 가능

구글 크롬의 해석엔진(V8)을 확장한 프로그램

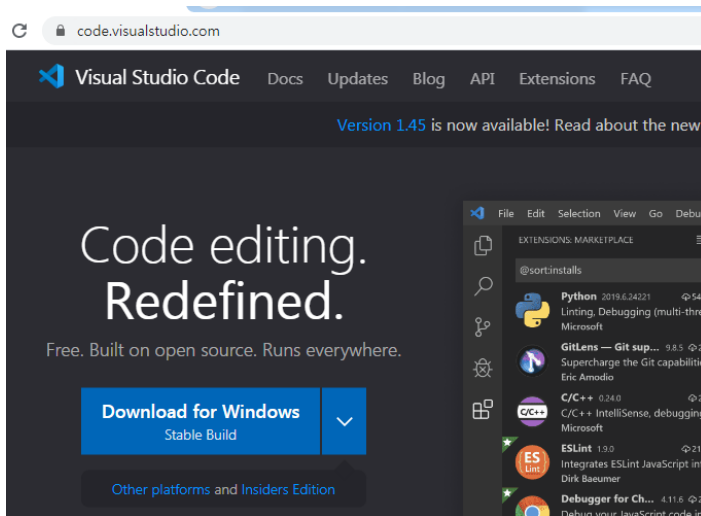


픽사(Pixlr) 그림판

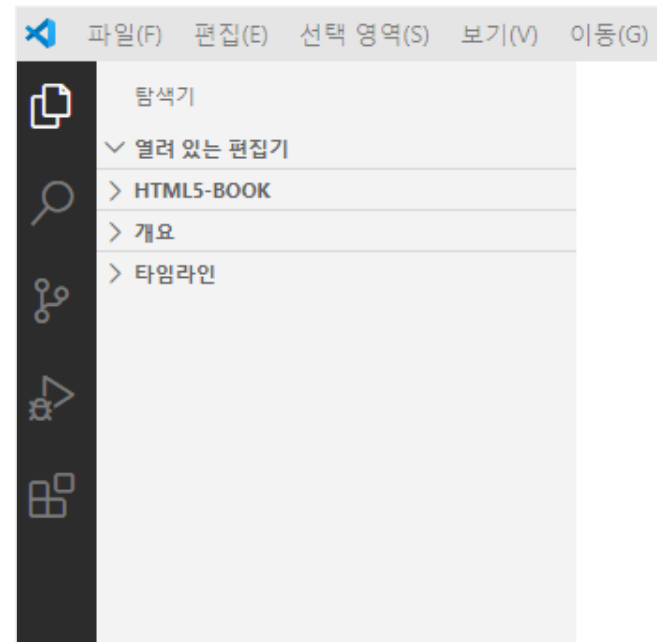
편집기 – 비주얼 스튜디오 코드

비주얼 스튜디오 코드

- 비주얼 스튜디오 코드 다운로드 -> 설치



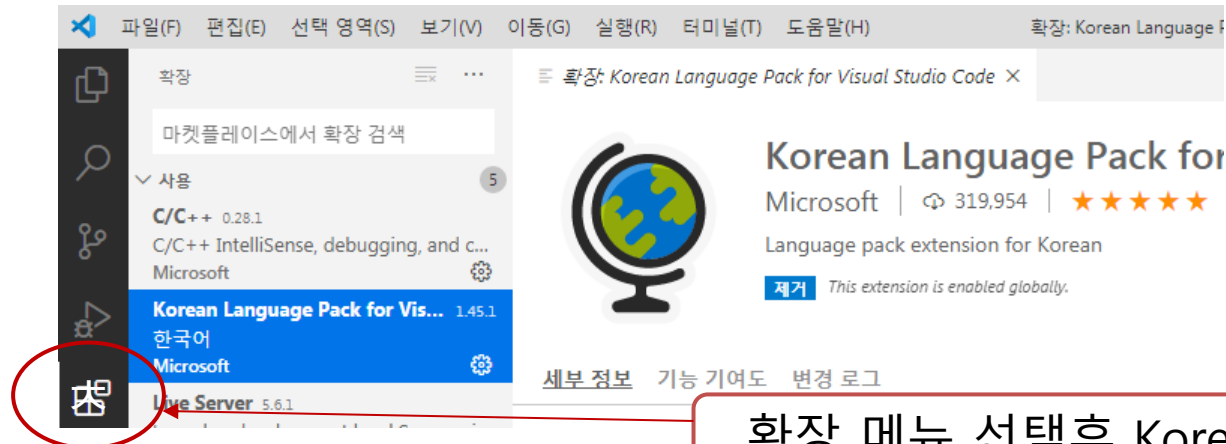
다운로드



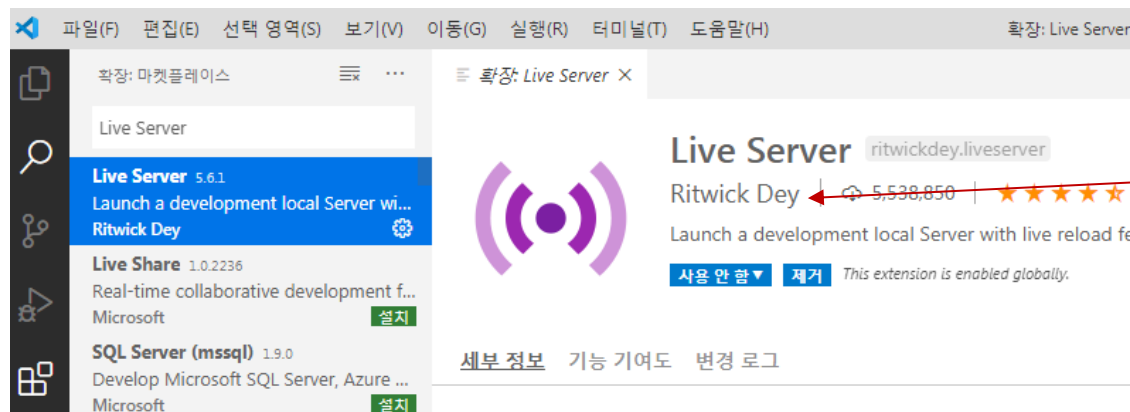
VS code 화면

비주얼 스튜디오 코드 환경 설정

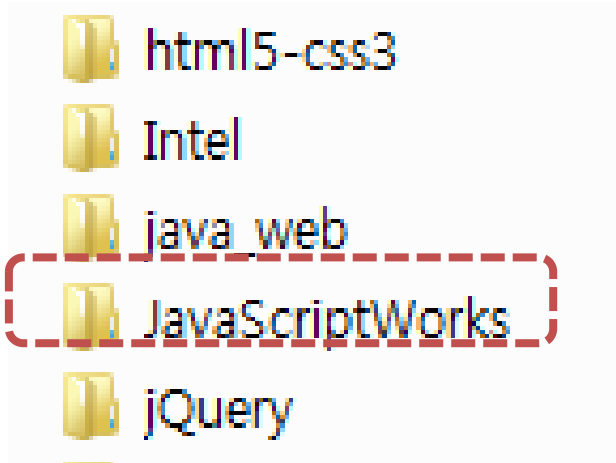
- 한국어 팩 설치



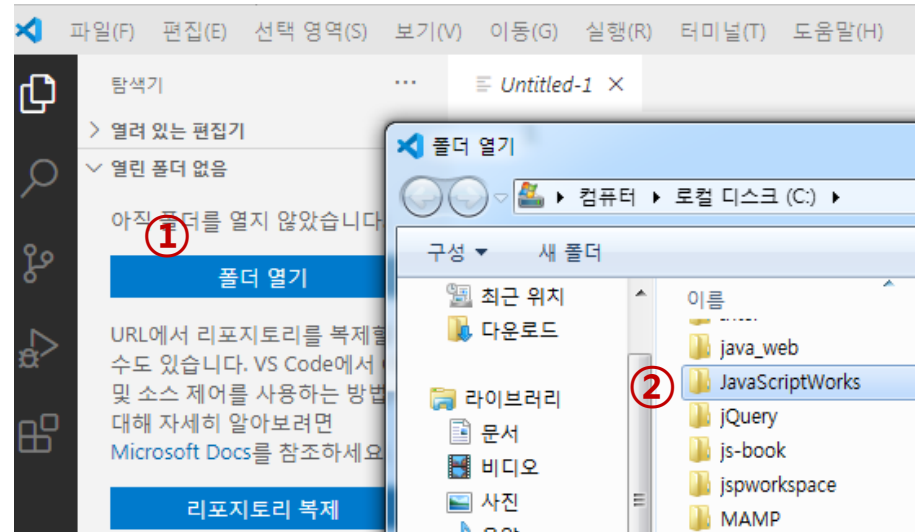
- 라이브 서버(Live Server) 설치



비주얼 스튜디오 코드 환경 설정

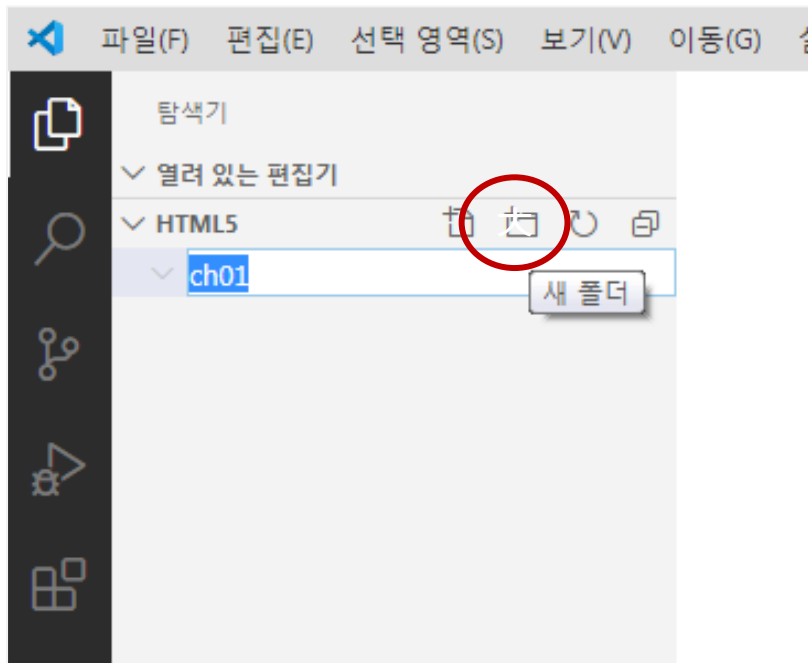


작업영역 폴더 만들기(탐색기)

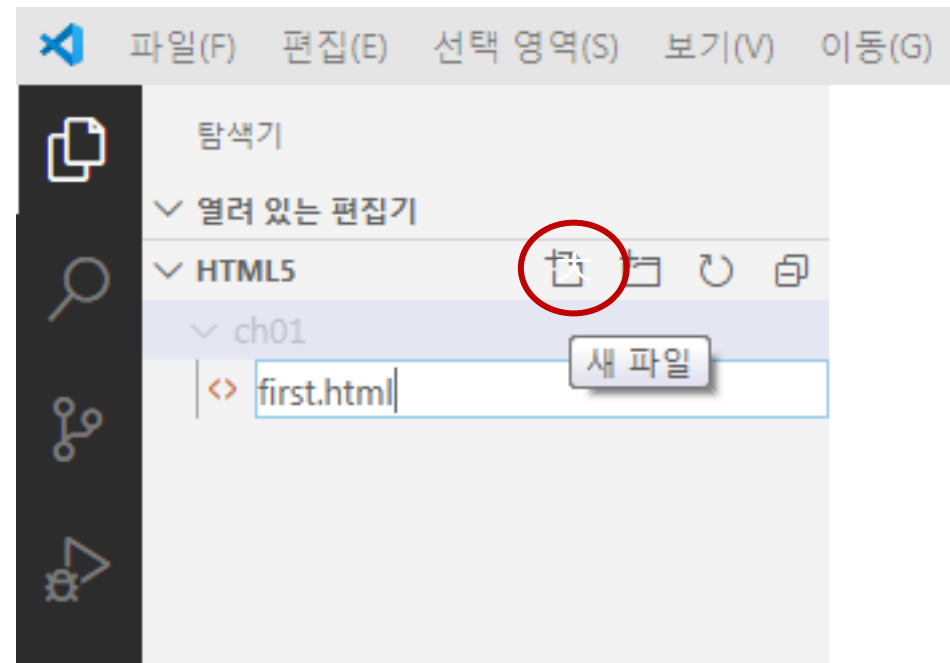


작업영역 폴더로 설정

비주얼 스튜디오 코드



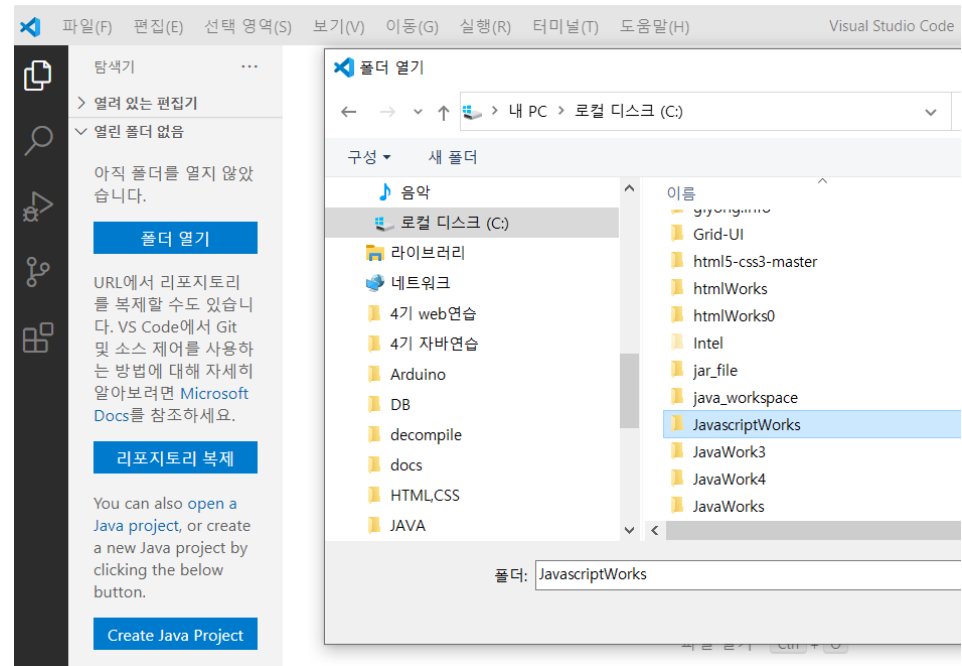
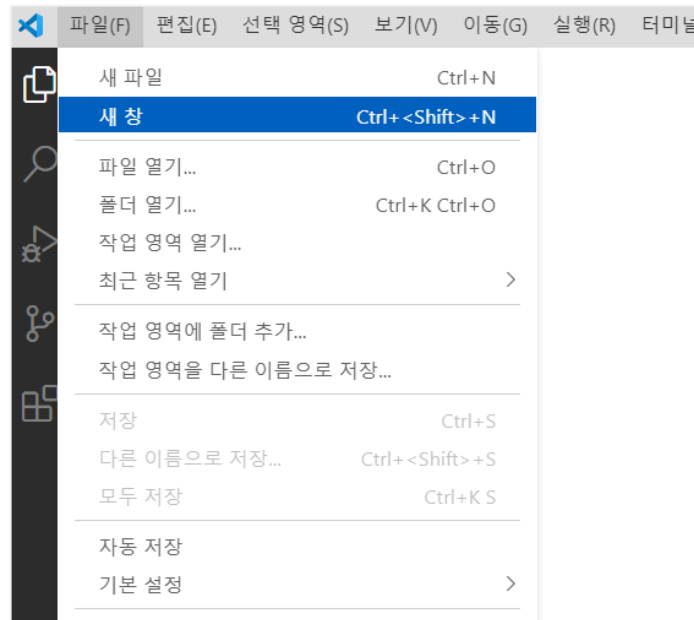
1. ch01 폴더 만들기



2. first.html 파일 만들기

비주얼 스튜디오 코드

기존에 사용중인 경우 : 새창 열기 > 폴더 열기



JavaScript 작성하기

자바스크립트 출력 - 내부 스크립트

- **<script>~</script>** 태그 안에 작성
- document(내장 객체)의 write() 메서드로 출력

자바 스크립트 예제

Hello~ JavaScript!

안녕~ 자바스크립트!

```
<h1>자바 스크립트 예제</h1>
<script>
    document.write("Hello~ JavaScript!<br>");
    document.write("<h3>안녕~ 자바스크립트!</h3>")
</script>
```

자바스크립트 Reference

자바스크립트 Reference

- <http://w3schools.com>
- Reference > JavaScript(HTML DOM) > DOM Document > write()

HTML DOM Reference

The references describe the properties and methods of each DOM object, along with examples.

Attributes	Document	Element
Event Objects	HTMLCollection	Location
Screen	Style	Window

HTML DOM write()

< Document Object

Example

Write some text directly to the HTML document

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

Try it Yourself »

- normalizeDocument()
- open()
- querySelector()
- querySelectorAll()
- readyState
- referrer
- removeEventListener()
- renameNode()
- scripts
- strictErrorChecking
- title
- URL
- write()**
- writeln()
- DOM Element

JavaScript 작성하기

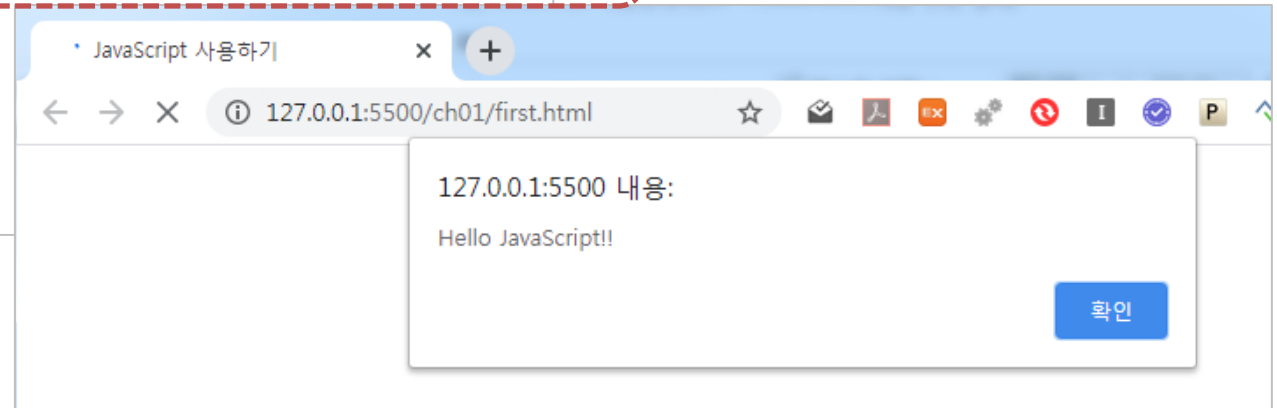
alert()으로 출력하기

- 브라우저에서 알림(경고)창으로 출력

```
<!DOCTYPE html>
<html>
  <head>
    <meta charset="utf-8">
    <title>JavaScript 사용하기</title>
    <script>
      alert("Hello JavaScript!!"); //알림
    </script>
  </head>
  <body>

</body>
</html>
```

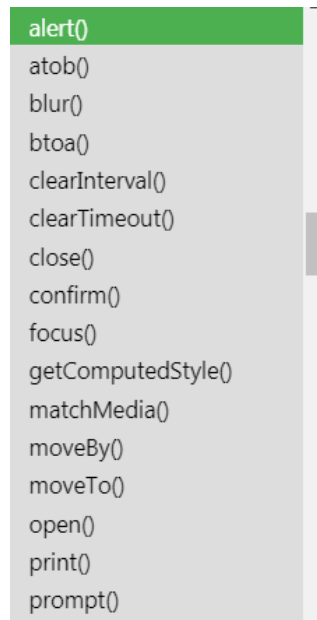
window.alert()에서
window 객체 생략



자바스크립트 Reference

자바스크립트 Reference

- <http://w3schools.com>
- Reference > JavaScript(HTML DOM) > DOM Window > Window Object Method() > **alert()** - 출력 , **prompt()** - 입력



alert()
atob()
blur()
btoa()
clearInterval()
clearTimeout()
close()
confirm()
focus()
getComputedStyle()
matchMedia()
moveBy()
moveTo()
open()
print()
prompt()

Window alert() Method

< Window Object

Example

Display an alert box:

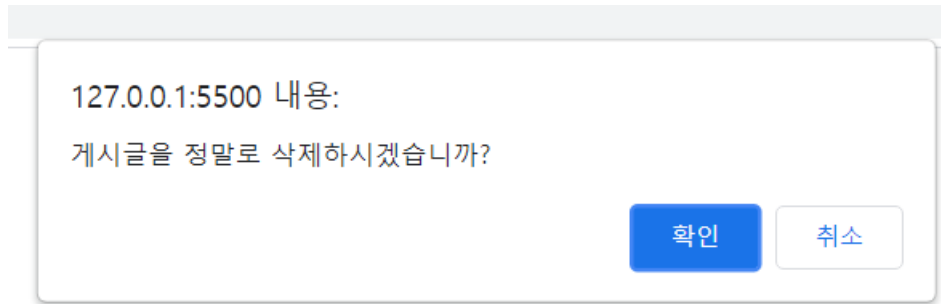
```
alert("Hello! I am an alert box!!");
```

Try it Yourself »

자바스크립트 Reference

확인창 출력하기

confirm(메시지)



```
<body>
  <h2>게시판 목록입니다.</h2>
  <script>
    | var reply = confirm("게시글을 정말로 삭제하시겠습니까?");
  </script>
</body>
```


JavaScript 작성하기

자바스크립트 출력 - 외부 스크립트

<script src="js파일경로"></script>

```
<!DOCTYPE html>
<html lang="ko">
<head>
  <meta charset="UTF-8">
  <title>Javascript</title>
  <script src="js/hello.js"></script>
</head>
<body>
</body>
</html>
```

```
// 자바스크립트 코드 작성
document.write("Hello~ Javascript!!<br>");

document.write("<h2>행복하세요~!!<h2>");
```

세미콜론 / 주석

문장 마침

- 문장 끝에 세미콜론(;) 사용

주석

- 한 줄 주석 : `//~~`
- 여러줄 주석 : `/* ~~ */`

```
/*  
    자바스크립트 코드 작성  
    document.write()는 출력함수이다.  
*/  
document.write("Hello~ JavaScript!<br>");  
document.writ("<h3>안녕~ 자바스크립트!</h3>")
```

데이터 출력

콘솔창을 이용한 출력

- 크롬 브라우저 > F12(개발자도구) > Console
- **console.log() 메서드**

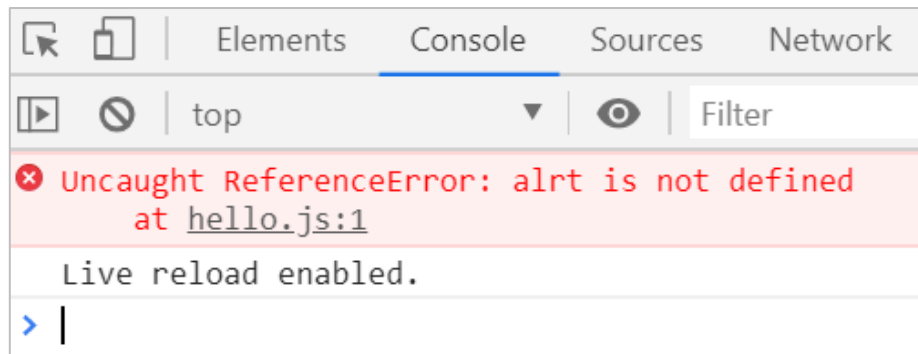
```
<h2>콘솔(Console)로 보기</h2>
<script>
  //출력 -> console() 사용
  console.log("Hello~ Javascript!!");
  console.log(2000);
  console.log(2.5415);
  console.log(3 + 5);
  console.log("face");
</script>
```

Hello~ Javascript!!
2000
2.5415
8
face

디버깅(Debugging)

디버깅(Debugging) - 오류(에러) 찾기

- 크롬 웹 브라우저의 검사 메뉴
- console 창에서 확인



```
<body>
  <script>
    document.write("Hello~ Javascript");
    document.writ("<h1>Hello~ Javascript<h1>");
  </script>
```

자바스크립트 해석 엔진

웹 브라우저에는 HTML 분석기(Parser), CSS 분석기,
자바스크립트 해석기(Interpreter)가 포함되어 있음



```
<!DOCTYPE html>
<html lang="ko">
<head>
  <meta charset="UTF-8">
  <title>Javascript</title>
  <style>
    body{
      text-align: center;
    }
  </style>
</head>
<body>
  <script>
    document.write("Hello~ Javascript");
  </script>
</body>
</html>
```

- ① HTML분석기는 HTML의 순서와 포함관계를 확인한다.
- ② CSS 분석기는 HTML 분석기가 태그 분석을 끝낸 다음 <style> 태그 사이의 스타일 정보를 분석한다.
- ③ 마지막으로 자바스트립트 해석기가 <script>태그 사이의 소스 코드를 해석한다.