

# 01. 자바스크립트 시작하기

---

# 자바스크립트 기본 상식

---


# 자바스크립트의 특징



Web Browser 위에서 동작하는 언어

# 자바스크립트의 특징

- 모든 웹 브라우저에서 작동한다.
- 웹 브라우저에서 실행 결과를 즉시 확인할 수 있다.
- 풀스택 웹 개발 뿐 아니라 다양한 용도의 프로그램을 만들 수 있다
- 다양한 자바스크립트 공개 API를 사용할 수 있다
- 다양한 라이브러리와 프레임워크를 사용할 수 있다



기본으로 제공하는 소스를 수정하거나 추가하는 방법으로 웹 프로그램을 만들 수 있게 미리 준비되어 있는 환경

# 자바스크립트와 ECMAScript

초기 자바스크립트            JScript(마이크로소프트)



브라우저 상의 스크립트 기술 표준 규격을 제정해 달라고,  
ECMA에 '자바스크립트'를 제출



ECMA에서 자바스크립트를 기반으로 표준 채택 공식명칭: ECMAScript

- ECMAScript는 자바스크립트를 기반으로 표준화된 스크립트 언어
- 자바스크립트는 ECMAScript 표준사양을 따르는 가장 유명한 언어 (자바스크립트 외에도 여러 스크립트 언어가 있음)

# 왜 ES6를 이야기할까?

- 1997년에 ECMAScript 초판이 채택된 후 'ES2'와 'ES3'처럼 ES 뒤에 판 번호를 붙여서 구별
- 이후 'ECMAScript 2015'처럼 연도를 붙인 것을 공식적인 이름으로 결정
- 기존의 자바스크립트는 단순히 웹 브라우저의 동적인 효과를 위해 개발된 언어였고 아주 단기간에 개발된 언어였기 때문에 활용 범위가 좁았음
- HTML5가 등장할 즈음 웹 개발에 새로운 변화가 찾아왔고, 모든 브라우저에서 지원하는 자바스크립트를 확장시켜야 할 필요성이 생김 → 이 변화를 수용해서 나온 것이 ES6(ES 2015)
- ES6 이후부터 일반적인 개발 언어처럼 복잡한 개념이 추가되었는데, 기존 자바스크립트를 확장한 것이기 때문에 옛날 개념들을 하나씩 버리면서 새로운 개념들로 교체해 나가는 중.
- ECMAScript는 1년마다 업데이트되고 있음.

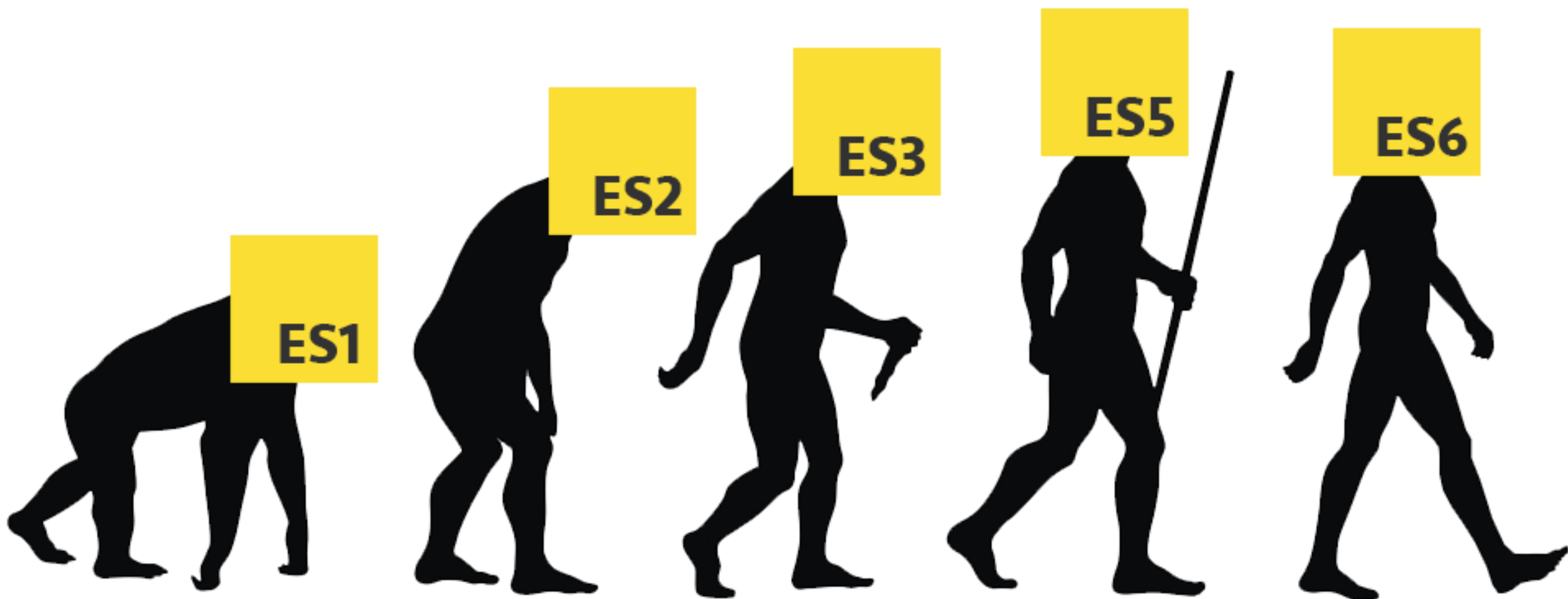
1997

1998

1999

2009

2015



---

# 웹 개발과 자바스크립트

---



# 클라이언트와 서버

사이트가 동작하는 것은, 서버 컴퓨터와 클라이언트 컴퓨터 간에 정보를 주고받으면서 그 결과를 사용자에게 보여 주는 것



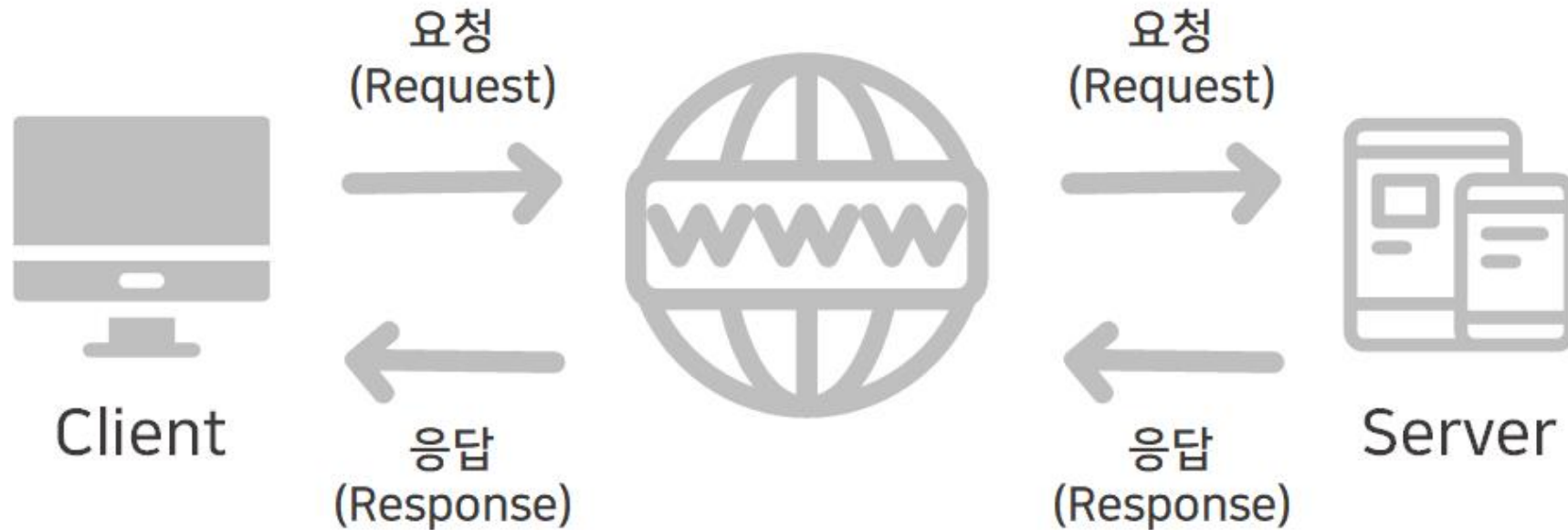
(예) 사용자가 쇼핑몰에서 상품 정보를 보려면

- 1) 클라이언트 컴퓨터에서 서버 컴퓨터로 관련 정보 요청
- 2) 서버에서는 해당 상품 정보를 찾아서 클라이언트 컴퓨터로 보내줌

# 클라이언트와 서버

클라이언트 측에서의 언어로 자바스크립트가 사용

뿐만 아니라 최근 NodeJS의 등장으로 웹 서버에서도 사용되면서,  
JavaScript가 광범위하게 사용됨



# 프론트엔드 개발과 자바스크립트

- 프론트엔드(front-end)란, 앞(front)에 있어서 사용자에게 보이는 부분을 가리킴
- 웹 사이트나 애 플리케이션에서 내용을 작성하고 화면을 디자인하는 것부터 사용자의 동작에 반응해서 결과를 만드는 것들이 모두 프론트엔드 개발의 역할
- 최근 의 웹 사이트는 사용자와 실시간으로 정보를 주고받으면서 애플리케이션처럼 동작하기 때문에 그만큼 프론트엔드 개발이 중요함
- 프론트엔드 개발에서 자바스크립트의 기능도 함께 강력해지고 있음  
→ 프레임워크와 라이브러리가 많이 등장

표 1-1 자주 사용하는 자바스크립트 프레임워크와 라이브러리

종류	기능	관련 URL
React	페이스북에서 만든 라이브러리로, 사용자 인터페이스를 만들 수 있습니다.	<a href="https://reactjs.org">reactjs.org</a>
Vue.js	사용자 인터페이스를 만들어 주는 프레임워크로, 웹 애플리케이션을 만들 수 있습니다.	<a href="https://vuejs.org">vuejs.org</a>
Angular	구글에서 개발한 프레임워크로, 수많은 기능을 지원하기 때문에 큰 규모의 웹 사이트를 개발하는 데 적합합니다.	<a href="https://angular.io">angular.io</a>

# 유명한 웹 사이트에서 사용되는 언어들

웹사이트	인기도 (1달 간 방문객 수) <sup>[1]</sup>	프론트엔드 (클라이언트 측)	백엔드 (서버 측)	데이터베이스	세부 사항
구글 <sup>[2]</sup>	1,600,000,000	JavaScript	C, C++, Go, <sup>[3]</sup> Java, Python	BigTable, <sup>[4]</sup> MariaDB <sup>[5]</sup>	세계에서 가장 많이 사용되는 검색 엔진
페이스북	1,100,000,000	JavaScript	Hack, PHP (HHVM), Python, C++, Java, Erlang, D, <sup>[6]</sup> XHP, <sup>[7]</sup> Haskell <sup>[8]</sup>	MariaDB, MySQL, <sup>[9]</sup> HBase, Cassandra <sup>[10]</sup>	가장 많이 방문되는 소셜 네트워크 서비스
유튜브	1,100,000,000	JavaScript	C, C++, Python, Java, <sup>[11]</sup> Go <sup>[12]</sup>	Vitess, BigTable, MariaDB <sup>[13]</sup>	가장 많이 방문되는 동영상 공유 사이트
야후!	750,000,000	JavaScript	PHP	MySQL, PostgreSQL, <sup>[14]</sup> VB.NET	야후는 현재 Node.js로 언어를 변경하는 중임 <sup>[15]</sup>
아마존	500,000,000	JavaScript	Java, C++, Perl <sup>[16]</sup>	Oracle Database <sup>[17]</sup>	유명한 온라인 쇼핑몰 사이트
위키백과	475,000,000	JavaScript	PHP, Hack	MySQL, MariaDB <sup>[18]</sup>	"미디어위키"는 PHP 로 프로그래밍 되었으며,, HHVM 으로 운영됨.
트위터	290,000,000	JavaScript	C++, Java, Scala, Ruby <sup>[19]</sup>	MySQL <sup>[20]</sup>	유명 소셜 네트워크 서비스
빙	285,000,000	JavaScript	ASP.NET	Microsoft SQL Server	
이베이	285,000,000	JavaScript	Java, <sup>[21]</sup> JavaScript, <sup>[22]</sup> Scala <sup>[23]</sup>	Oracle Database	온라인 중고매장
MSN	280,000,000	JavaScript	Microsoft SQL Server	"메신저"로 알려진 단순한 이메일 클라이언트 서비스	
마이크로소프트	270,000,000	JavaScript	ASP.NET	세계 최대 소프트웨어 회사 중 한 곳	
링크드인	260,000,000	JavaScript	Java, JavaScript, <sup>[24]</sup> Scala	Voldemort <sup>[25]</sup>	세계 최대 직업 네트워크
핀터레스트	250,000,000	JavaScript	Django, <sup>[26]</sup> Erlang	MySQL, Redis <sup>[27]</sup>	
워드프레스	240,000,000	JavaScript	PHP, JavaScript <sup>[28]</sup> (Node.js)	MariaDB, MySQL	

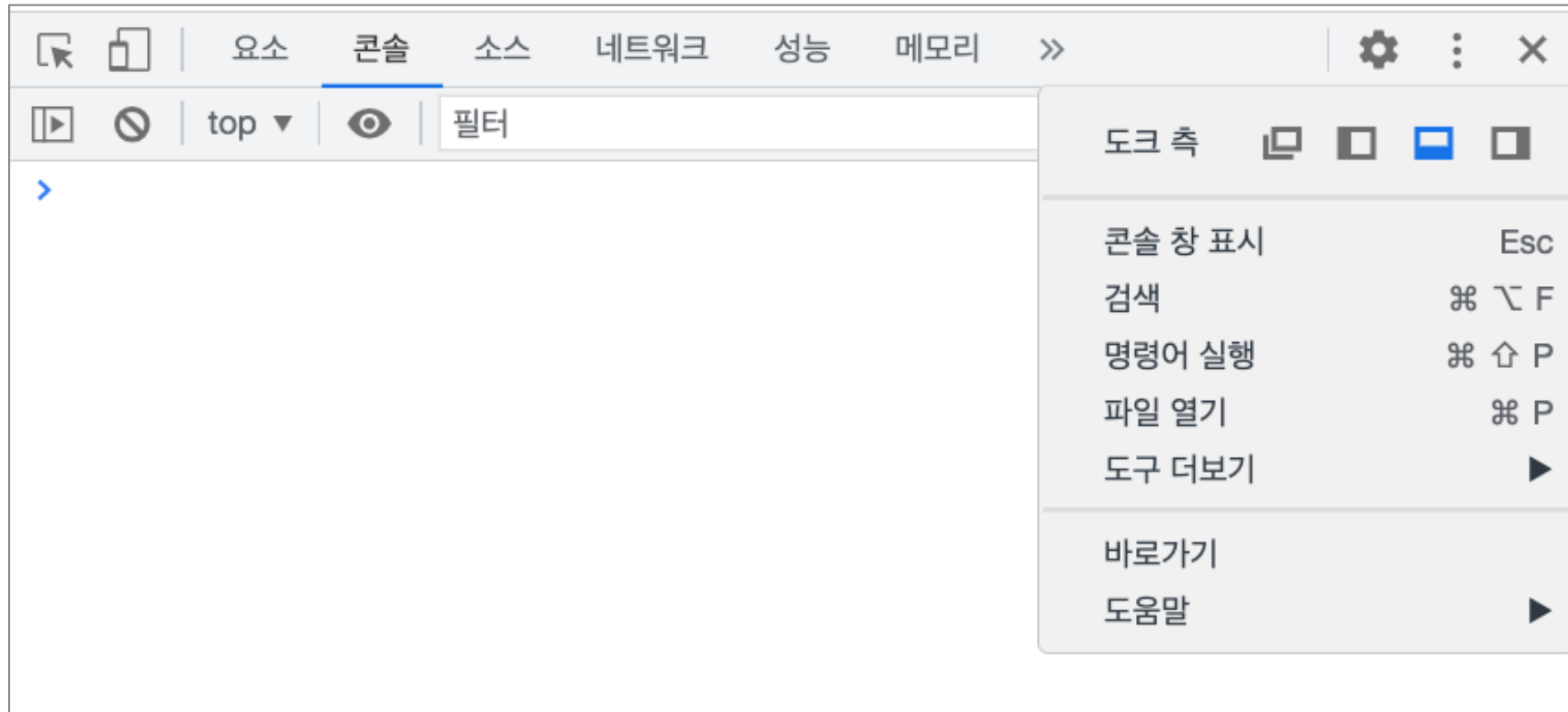
[위키피디아](#)

# 콘솔 창에서 자바스크립트 사용하기

웹 브라우저에는 자바스크립트 소스를 실행할 수 있는 자바스크립트 엔진이 포함되어 있어서 웹 브라우저의 콘솔 창을 통해 간단한 자바스크립트 소스를 실행해 볼 수 있다.

- 1) 웹 브라우저 주소 표시줄에 about:blank 입력해서 빈 화면 표시
- 2) 웹 개발자 도구 창에서 [콘솔] 선택  
(윈도우 [Ctrl]+[Shift]+[J] , 맥 [Option]+[Comman]+[J] 단축키 주로 사용)
- 3) 10 + 20 입력
- 4) console.log("안녕하세요?") 입력

콘솔 창을 브라우저 창의 왼쪽이나 오른쪽으로 옮길 수도 있고, 따로 열어놓을 수도 있습니다.  
편한 방법을 선택하세요.



---

# 간단한 자바스크립트부터 시작하기

---

# 웹 문서에서 스크립트 작성하기

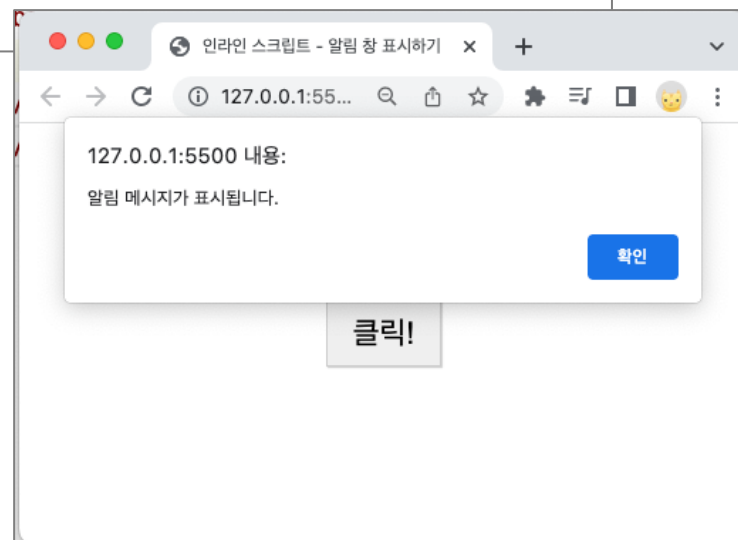
- 자바스크립트 처리기는 웹 브라우저 안에 포함되어 있기 때문에 자바스크립트 소스는 웹 문서에 작성하면 웹 브라우저가 해석함.
- 자바스크립트 소스는 웹 문서 안에 직접 작성할 수도 있고,
- 자바스크립트 소스만 따로 파일로 저장해서 서로 연결해서 사용할 수도 있다.



# 인라인 스크립트

- HTML 태그 안에 직접 작성하는 자바스크립트.
- 팝업 창을 열고 닫거나, 알림 메시지를 표시할 때처럼 간단한 명령을 처리할 경우 인라인 스크립트를 자주 사용

```
<body>  
  <button onclick = "alert('알림 메시지가 표시됩니다.')">클릭!</button>  
</body>
```



# 내부 스크립트

- 웹 문서에서 `<script>` 태그와 `</script>` 태그 사이에 실행할 자바스크립트 소스 작성
- `<script>` 태그는 웹 문서에서 모든 곳에 위치할 수 있고 삽입된 위치에서 바로 스크립트가 실행됨
- 한 문서 안에 여러 개의 `<script>` 태그를 사용할 수 있음
- 내부 스크립트는 주로 `</body>` 태그 앞에 사용함

```
<body>
```

```
<p>새로 고침해 보세요</p>
```

```
<script>
```

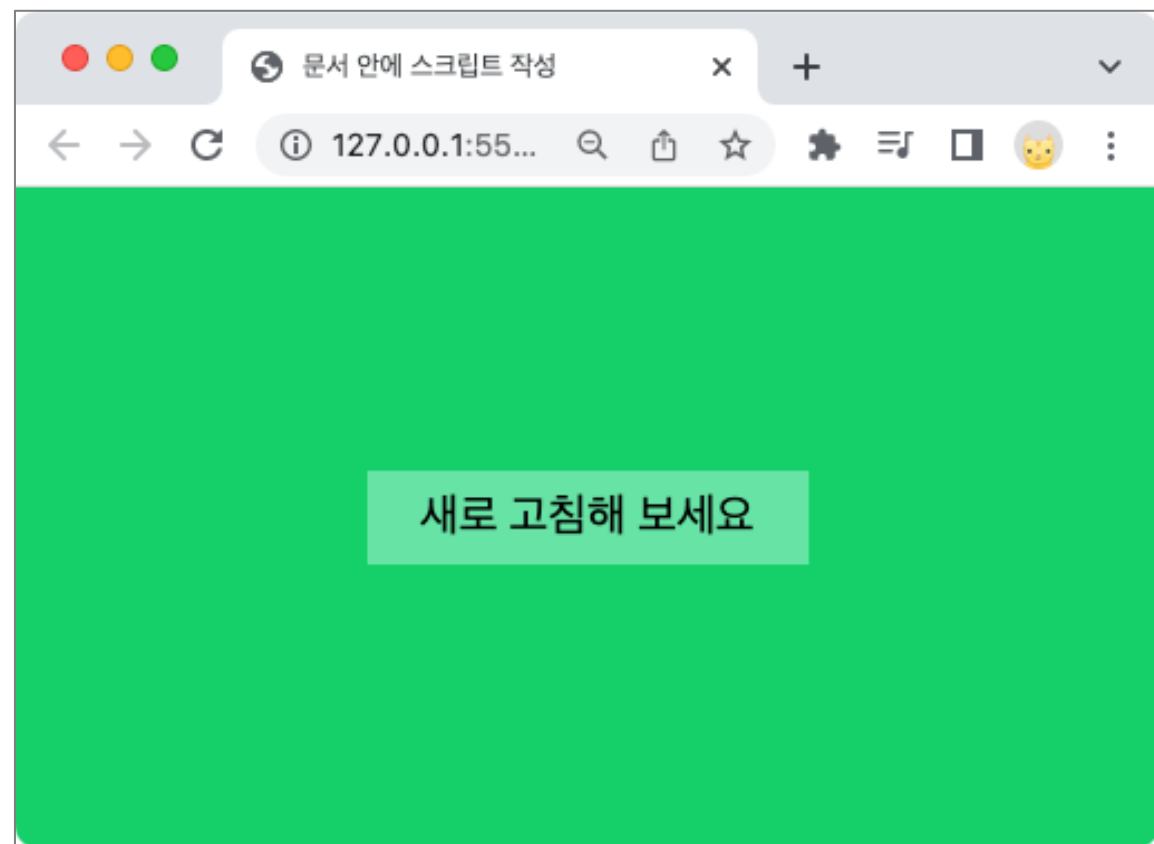
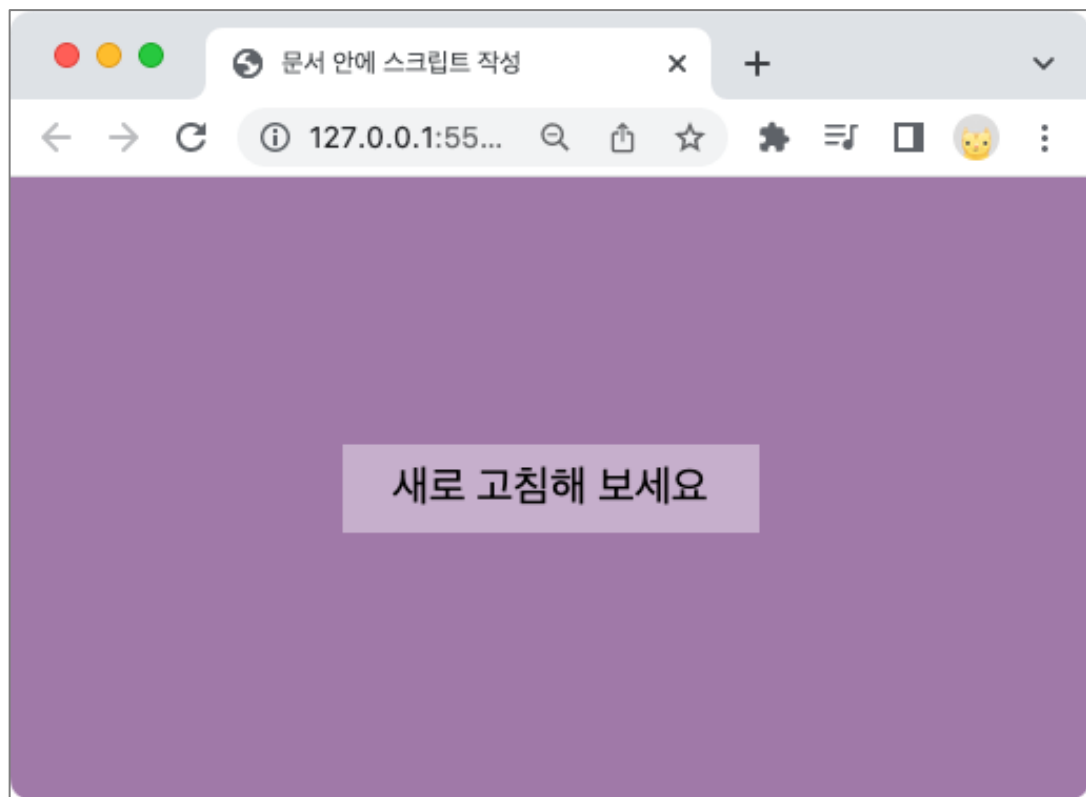
```
function random(number) {  
    return Math.floor(Math.random() * number); }
```

```
function bgChange() {  
    const rndCol = 'rgb(' + random(255) + ',' + random(255) + ',' + random(255) + ') ' ;  
    document.body.style.backgroundColor = rndCol;  
}
```

```
bgChange();
```

```
</script>
```

```
</body>
```



# 인라인 스크립트와 내부 스크립트의 단점

같은 자바스크립트 소스를 여러 웹 문서에서 사용해야 할 경우, 필요한 문서마다 똑같은 소스를 반복해서 삽입해야 함.

만일 여러 페이지에 삽입한 자바스크립트 소스를 수정해야 한다면 소스가 포함된 모든 문서를 다 찾아다니면서 하나씩 수정해야 함.

→ 자바스크립트 소스를 작성할 때 외부 스크립트 파일로 저장해서 링크하는 방법을 많이 사용함.

→ 마크업과 구별되기 때문에 스크립트 소스를 관리하기 쉬움

# 외부 스크립트 연결해서 사용하기

- 외부 스크립트 파일 확장자 : .js
- HTML 문서에서 <script> 태그를 사용해서 외부 스크립트 연결

기본형:

**<script src="스크립트 파일 경로"></script>**

- 외부 스크립트 파일 안에는 <script> 태그 없이 자바스크립트 소스만 작성함
- 따로 js 폴더를 만들어 저장하는 것이 좋음  
(외부 css 파일을 css 폴더에 저장하는 것처럼~)

## [실습] 외부 스크립트 연결해서 사용하기

16external.html

```
<body>
  <p>새로 고침해 보세요</p>
</body>
```

연결해 보자

16jschangeBg.js

```
function random(number) {
  return Math.floor(Math.random() * number);
}

function bgChange() {
  const rndCol = 'rgb(' + random(255) + ',' + random(255) + ',' + random(255) + ')';
  document.body.style.backgroundColor = rndCol;
}

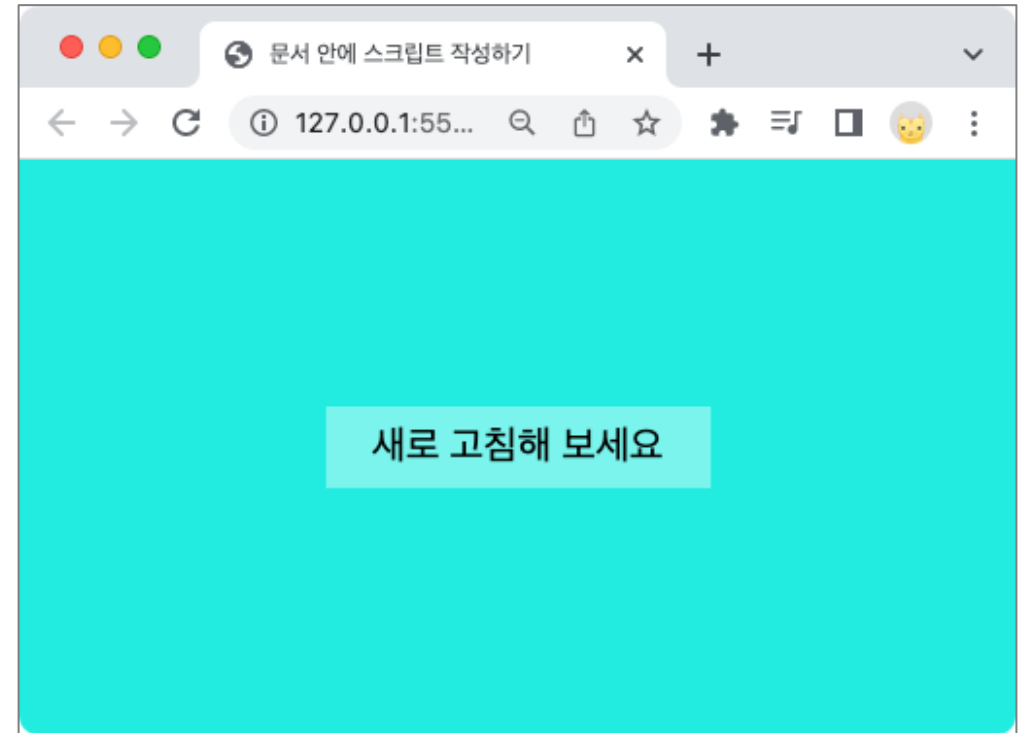
bgChange();
```

## [실습] 외부 스크립트 연결해서 사용하기

16\result\external-1.html

16\external.html

```
<body>  
  <p>새로 고침해 보세요</p>  
  
  <script src="js/changeBg.js"> </script>  
</body>
```





# 웹 브라우저에서 자바스크립트를 해석하는 과정

```
<!DOCTYPE html>
<html lang="ko">
<head>
  <meta charset="UTF-8">
  ...
  <link rel="stylesheet" href="css/main.css">
  <style>...</style>
</head>
<body>
  <p>새로 고침해 보세요</p>

  <script src="js/changeBg.js"> </script>
</body>
</html>
```

external.html

```
function random(number) {
  ...
}

function bgChange() {
  ...
}

bgChange();
```

js\changeB

external.html

1

→ `<!DOCTYPE html>`

```
<html>
<head>
  <meta charset="UTF-8">
  ...
  <link rel="stylesheet" href="css/main.css">
  <style>...</style>
</head>
<body>
  <p>새로 고침해 보세요</p>

  <script src="js/changeBg.js"> </script>
</body>
</html>
```

현재 문서가 웹 문서라는 사실을 알게 됨.  
HTML표준에 맞춰서 문서를 해석하기 시작

jsWchangeB

```
function random(number) {
  ...
}

function bgChange() {
  ...
}

bgChange();
```

external.html

```
<!DOCTYPE html>
<html lang="ko">
<head>
  <meta charset="UTF-8">
  ...
  <link rel="stylesheet" href="css/main.css">
  <style>...</style>
</head>
<body>
  <p>새로 고침해 보세요</p>

  <script src="js/changeBg.js"> </script>
</body>
</html>
```

js\changeB

```
function random(number) {
  ...
}
```

```
function bgChange() {
```

2

## HTML 태그의 순서와 포함관계 분석

- <head>~</head> 태그 사이에는 어떤 태그가 있는지,
- <body>~</body> 태그 사이에는 어떤 태그가 있는지,
- 태그와 태그 간의 포함 여부 등

external.html

```
<!DOCTYPE html>
<html lang="ko">
<head>
  <meta charset="UTF-8">
  ...
  <link rel="stylesheet" href="css/main.css">
  <style>...</style>
</head>
<body>
  <p>새로 고침해 보세요</p>

  <script src="js/changeBg.js"> </script>
</body>
</html>
```

js\changeB

```
function random(number) {
  ...
}
```

```
function bgChange() {
```

3

태그 분석하면서 따로 분류해 두었던  
외부 스타일 시트나 문서 안의 스타일 정보를  
분석하면서 화면에 표시함

external.html

```
<!DOCTYPE html>
<html lang="ko">
<head>
  <meta charset="UTF-8">
  ...
  <link rel="stylesheet" href="css/main.css">
  <style>...</style>
</head>
<body>
  <p>새로 고침해 보세요</p>

  <script src="js/changeBg.js"> </script>
</body>
</html>
```

4

js\changeB

```
function random(number) {
  ...
}

function bgChange() {
  ...
}

bgChange();
```

<script> 태그를 만나면 자바스크립트 해석기에게 스크립트 소스를 넘김.

## external.html

```
<!DOCTYPE html>
<html lang="ko">
<head>
  <meta charset="UTF-8">
  ...
  <link rel="stylesheet" href="css/main.css">
  <style>...</style>
</head>
<body>
  <p>새로 고침해 보세요</p>

  <script src="js/changeBg.js"> </script>
</body>
</html>
```

## js\changeB

```
function random(number) {
  ...
}

function bgChange() {
  ...
}

bgChange();
```

5

자바스크립트 실행