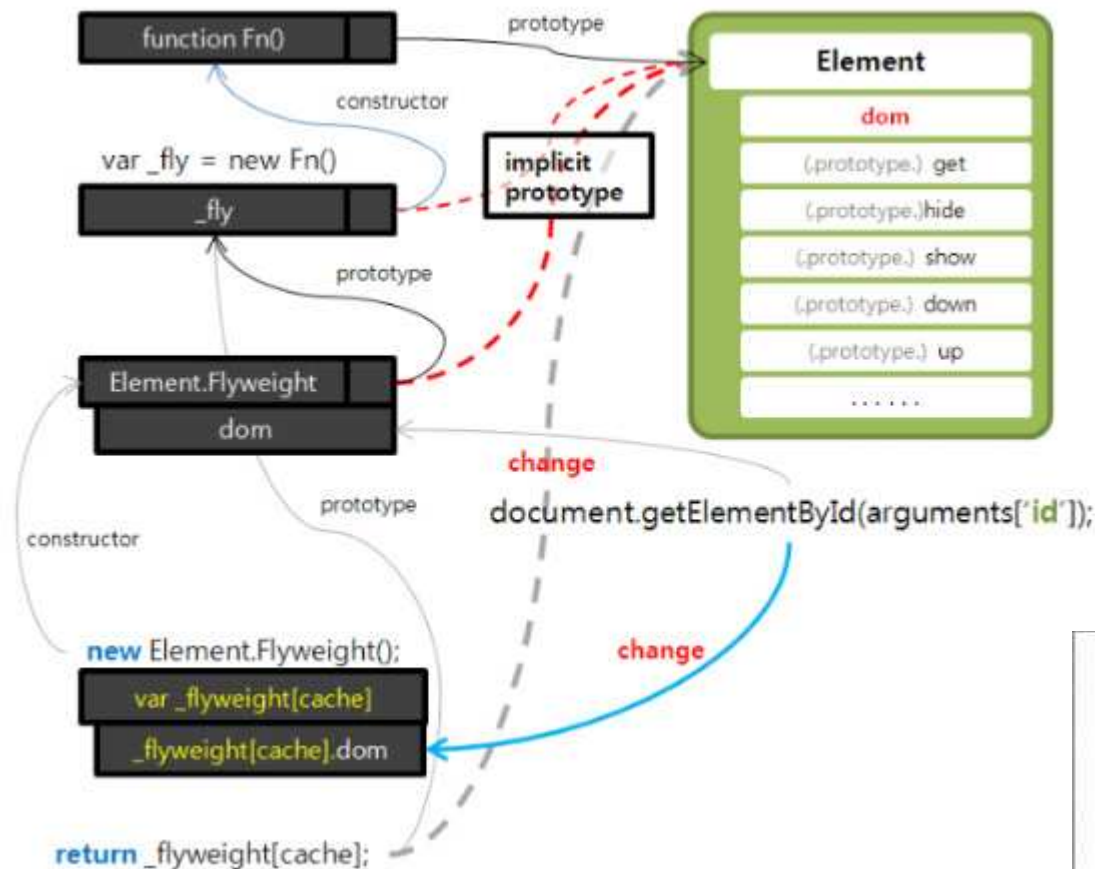


자바스크립트의 소개 및 개요



 **JAVA
SCRIPT**

자바스크립트의 시작

❖ 자바스크립트의 정의

- 클라이언트쪽에서 독립적으로 실행되는 프로그램을 작성하기 위한 스크립트 언어
- 넷스케이프 사의 브랜든 아이히(Brendan Eich)에 의해 모카라는 이름으로 만들어짐 -> 이후 LiveScript 라는 이름으로 개발
- 썬마이크로시스템사와 공동으로 라이브스크립트를 확장한 JavaScript 탄생
- ECMA 인터내셔널에서 넷스케이프사의 자바스크립트와 마이크로소프트사의 Jscript 기반 ECMAScript(ECMA-262)를 Technical Committee 39(TC-39)에서 정의
- 플랫폼에 독립적
- 모든 웹 브라우저에서 동일한 실행 결과를 얻을 수 있음(무료, 쉽고 자유로움)



자바스크립트의 시작

❖ 자바스크립트의 특징

- 웹 문서(XHTML, HTML)에 끼워서 사용하는 스크립트 언어
- 웹 브라우저에서 자바스크립트 해석기가 웹 문서를 실행할 때 프로그램 코드를 해석하는 인터프리터 언어의 형태
- 자료형(primitive , reference data type) 조사를 철저하게 하지 않음
- 클래스를 정의, 클래스를 상속할 수 없다
- 객체를 정의하여 사용할 수는 있다.
- 소스가 노출되어 보안성이 없다.



자바 스크립트 시작

❖ 자바 스크립트 특성

- 컴포넌트 기반
- 다양한 프로그래밍 언어 및 여러 데이터베이스와의 인터페이스로 생성된 서버측 프로그램과 긴밀히 통합될 수 있다
- 웹 브라우저와 여타 프로그램에 내장되어 사용되는 스크립트 언어
- 브라우저 안에서 폼 검사, 드롭다운 메뉴, 데이터 업데이트 표시, 웹 페이지 직접 수정등을 할 수 있다.
- 웹 브라우저와 같은 특정 환경 내에서 실행되기 때문에, 파일이나 메모리 관리 등이 필요 없다.
- 여러 기능을 두루 갖추고 있으며, 충분한 보안 기능을 제공하는 객체 기반 언어
- 웹 페이지 내의 엘리먼트를 생성/제거하고, 속성을 변경한다.



자바스크립트의 시작

❖ 자바스크립트의 기능

- 웹 문서에 동적으로 움직일 수 있는 프로그램적인 요소를 추가
- 어떤 이벤트에 대한 반응을 할 수 있도록 웹 문서를 구성
- 웹 문서 내의 XHTML 요소의 내용을 읽거나 변경
- 클라이언트의 정보를 서버에 보내기 전에 먼저 검증
- 클라이언트가 어떤 웹 브라우저를 사용하는지 알아내어 사용자의 웹 브라우저에 맞는 웹 문서를 호출하는 맞춤형 서비스를 제공
- 클라이언트 컴퓨터에 쿠키를 새로 만들거나 생성된 쿠키를 삭제 및 회수할 수 있다.



자바 스크립트 시작

❖ 자바스크립트로 할 수 있는 것들

- 폼 필드 검사
- 웹 쿠키 저장 및 읽기
- 페이지 엘리먼트를 동적으로 변경
- 엘리먼트 감추기/보이기
- 엘리먼트 이동
- 사용자 이벤트의 캡처와 그에 따른 페이지 변경
- 스크롤
- 현재 웹페이지에 있는 상태에서 서버측 프로그램과 통신



자바스크립트의 시작

❖ 자바스크립트의 장점과 단점

- 작업 속도가 빠르다.
 - 자바스크립트는 HTML 파일 내에서 작성할 수 있으므로 개발 속도가 빠르다.
 - 자바, C/C++ 등 프로그래밍 언어의 기본적인 틀을 그대로 갖추고 있어 사용자로부터 익숙한 작업 가능.
- 운영체제의 제한을 받지 않는다.
- 배우기 쉽다.
 - 객체의 단순화로 인한 프로그램 제작 용이
- 소스 코드가 노출된다. (컴파일하지 않는 언어이므로 복사하여 사용할 수 있다.)
- 한정된 객체와 객체 함수다. (제한된 객체와 객체 함수를 사용)



자바스크립트의 시작

❖ 자바스크립트의 종류

- 웹 브라우저나 애플리케이션에 내장된 자바스크립트의 종류

표 1-1 애플리케이션에 따른 자바스크립트의 종류

애플리케이션	자바스크립트의 종류
모질라 파이어폭스	자바스크립트
구글 크롬	자바스크립트
인터넷 익스플로러	Jscript
오페라	ECMAScript
애플 사파리	자바스크립트
마이크로소프트 닷넷 프레임워크	JScript.NET
어도비 플래시와 플렉스	액션스크립트
어도비 아크로벳	자바스크립트



자바스크립트의 시작

❖ 브라우저의 호환성 문제

- 브라우저의 호환성 문제는 언어 자체보다는 문서 객체 모델(DOM)이 서로 다른것에 기인한다.
 - 예)사파리나 네비게이터에서는 Date 또는 String 타입의 자바스크립트 객체를 모두 동일하게 취급한다.
 - 예2) document 객체의 엘리먼트를 표현하는 방식이 브라우저 마다 다르게 구현되어 있다
- 웹 페이지의 엘리먼트 중 자바스크립트가 처리하는 부분과 CSS가 처리하는 부분이 중복된다



자바스크립트의 시작

❖ 자바 스크립트 샌드박스

- 자바 스크립트는 샌드박스 내에서 동작하도록 만들어졌다.
- 샌드박스 :
 - 스크립트가 컴퓨터의 자원에 임의로 접근할 수 없도록 하는 제한된 환경
- 브라우저 마다 자체적으로 보안 기능을 구현하고, 최소한의 것만 자바 스크립트 언어가 접근할 수 있도록 허용해 놓았다.
- 각 보안 정책은 브라우저별로 다르며, 각 스크립트가 할 수 있는 것도 다르다.



자바스크립트의 시작

❖ 자바 스크립트의 취약점

- 크로스 사이트 스크립트 (XSS, cross site scripting) : 쿠키의 내용이나 클라이언트 또는 웹 사이트의 데이터가 유출되는 등의 심각한 문제가 일어날 수 있다.



자바스크립트의 시작

❖ 웹 접근성을 위한 자바 스크립트 가이드 라인

- 자바 스크립트를 사용하지 않거나 사용할 수 없는 사람들이라도 중요한 정보에는 접근할 수 있어야 한다.
- 마우스 이벤트만을 통한 피드백은 사용하지 않는 것이 좋다.
- 자바스크립트를 사용해 드롭다운 메뉴를 구성할 경우에는, 자바 스크립트를 지원하지 않는 브라우저를 고려해 별도의 페이지를 만들어 주는 것이 좋다.
- 스크린 리더와 같이 자바스크립트를 사용할 수 없거나 제한적으로 사용할 수 있는 환경에서는 noscript를 사용하여 스크립트에 의해 동적으로 생성되는 콘텐츠에 대해서 정적인 코드도 함께 제공해 주어야 한다.

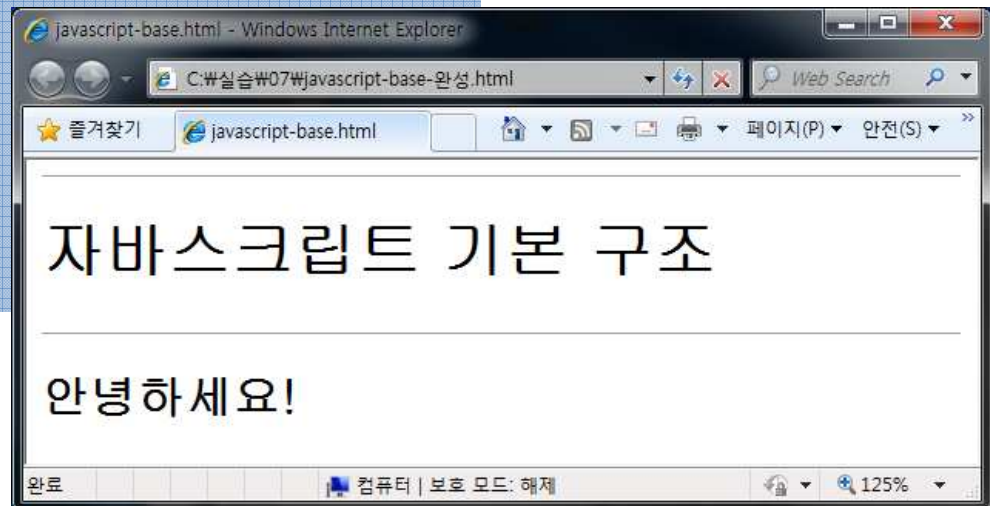


자바스크립트의 기본 문법

❖ 자바스크립트의 기본 구조

- <script> 태그를 사용
 - HTML5에서는 script 태그에 type 속성을 적지 않는 게 원칙

```
<!DOCTYPE html >
<html >
  <head>
    <meta charset= " utf-8" />
    <title> javascript-base.html </title>
  </head>
  <body>
    <hr /><h1> 자바스크립트 기본 구조 </h1><hr />
    <script type="text/javascript">
      document.write("<h2>안녕하세요!</h2>");
    </script>
  </body>
</html>
```



자바스크립트의 기본 문법

❖ 자바 스크립트 코드의 위치

- 페이지를 로드할 때 동적으로 웹 페이지의 콘텐츠를 생성하는 경우에는 body 태그 안에 위치시키는 것이 좋다.
- 함수 안에서 정의하고, page 이벤트에 사용되는 자바스크립트 코드는 head 태그 안에 위치시키는 것이 좋다
 - body 태그 안의 내용을 읽어 들이기 전에 로드되기 때문
- <script>태그의 defer 속성
 - 브라우저에게 스크립트가 문서 콘텐츠를 생성하지 않는다는 것을 명시, 즉 화면 표시와는 관련이 없다는 것을 브라우저에게 알려준다.
 - 자바스크립트 코드의 분량이 많거나 자바스크립트 라이브러리를 사용하는 경우에는 페이지를 로드할 때 상당한 속도 향상 효과를 볼 수 있다.



자바스크립트의 기본 문법

❖ 자바스크립트의 실행 위치

- 자바스크립트가 <head> 태그 영역에 배치되는 경우
 - 클라이언트 작업전 자바스크립트에서 지정한 이벤트가 웹 브라우저에 전달
 - <body> 태그 영역에서도 공통적으로 반응

```
01 <!DOCTYPE html >
02 <html>
03 <head>
04   <meta charset="utf-8" />
05   <title> javascript-head.html </title>
06   <script >
07     function message( ){
08       alert("이 메시지 상자는 onload 이벤트가 발생할 때 보인다.");
09     }
10     document.write("<h2>자바스크립트 세계에 오신 걸 환영합니다.</h2>");
11   </script>
12 </head>
13 <body onload="message( );">
14   <hr /><h1> 자바스크립트가 head 요소 영역에 위치했을 때 </h1><hr />
15 </body>
16 </html>
```



자바스크립트의 기본 문법

❖ 자바스크립트의 실행 위치

- 자바스크립트를 외부 파일로 사용되는 경우
 - <head> 태그 부분이든 <body> 태그 부분이든 기능을 구현할 곳에 삽입
 - 외부 파일 자바스크립트의 확장자는 ".js"
 - script element의 src 속성값이 호출할 자바스크립트의 파일명
 - 사용 방법

```
<script src="호출할 자바스크립트 파일명.js"></script>
```



자바스크립트의 기본 문법

❖ 실습

- ext-1.js 는 <body> 태그 영역 배치, ext-2.js 는 <head> 태그 영역 배치,

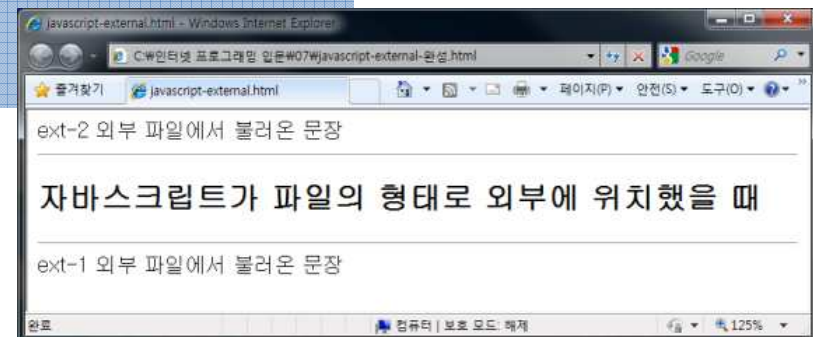
ext-1.js

```
document.write("ext-1 외부 파일에서 불러온 문장");
```

ext-2.js

```
document.write("ext-12외부 파일에서 불러온 문장");
```

```
01 <!DOCTYPE html >
02 <html >
03 <head>
04 <meta charset= " utf-8" />
05 <title> javascript-external.html </title>
06 <script src="ext-2.js"></script>
07 </head>
08 <body>
09 <hr /><h2> 자바스크립트가 파일의 형태로 외부에 위치했을 때 </h2><hr />
10 <script src="ext-1.js"></script>
11 </body>
12 </html>
```



자바스크립트의 기본 문법

❖ 자바스크립트의 표현 방식

■ 자바스크립트 구문

- 웹 브라우저에 문자 표현

```
document.write("자바스크립트 출력 구문");
```

- 문장 끝에 세미콜론. 구문사이를 구분하기 위해 사용. 생략해도 무관하나 오류방지를 위해 권장

■ 출력물

- document 객체 사용. write()와 writeln() 메소드 제공.
 - write() 메소드 : 개행 문자를 문자열 뒤에 출력하여 줄바꿈 없음
 - writeln() 메소드 : 개행 문자를 문자열 뒤에 출력하여 줄바꿈 있음

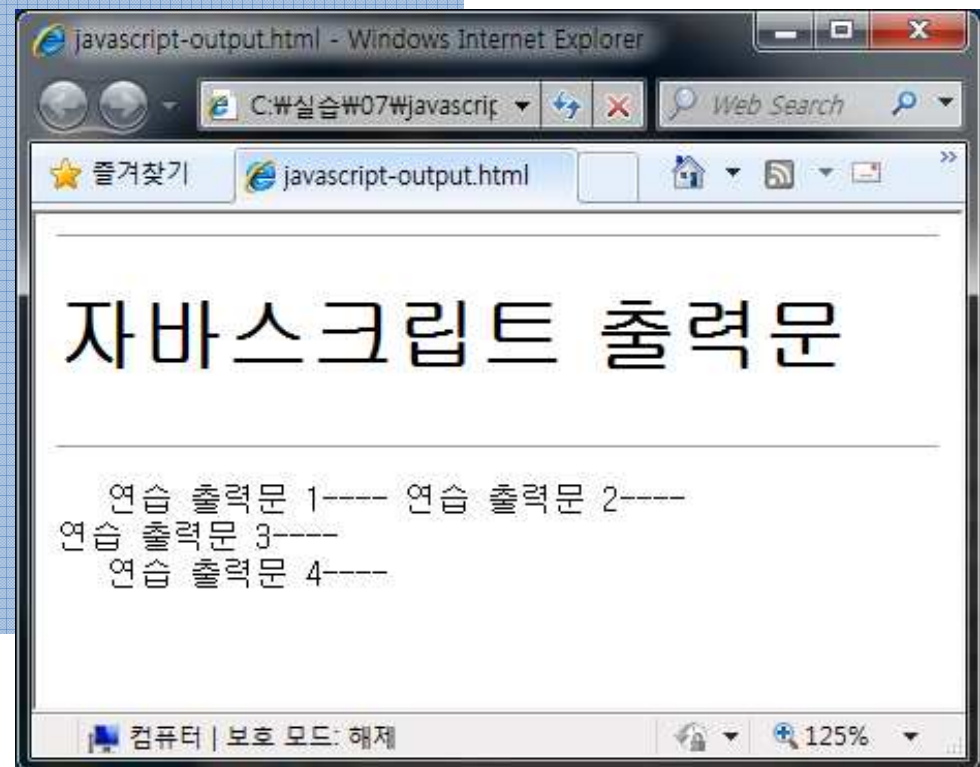


자바스크립트의 기본 문법

❖ 실습

- 출력문 1과 3인 write() 함수, 출력문 2는 writeln() 함수

```
01 <!DOCTYPE html >
03 <html >
04 <head>
05 <meta charset= " utf-8" />
06 <title> javascript-output.html </title>
07 </head>
08 <body>
09 <hr /><h1> 자바스크립트 출력문 </h1><hr />
10 <pre>
11 <script >
12   document.write("연습 출력문 1----");
13   document.writeln("연습 출력문 2----");
14   document.write("연습 출력문 3----");
15 </script>
16   연습 출력문 4----
17 </pre>
18 </body>
19 </html>
```



자바스크립트의 활용

❖ 웹에서 웹 애플리케이션으로

■ 초기의 웹

- 변화 없는 정적 글자들의 나열
- 웹은 하이퍼링크라는 매개체를 사용해 웹 문서가 연결된 거대한 책에 불과

■ 자바스크립트의 등장

- 웹 문서의 내용을 동적으로 바꾸거나 마우스 클릭 같은 이벤트 처리

■ 웹은 애플리케이션의 모습에 점점 가까워짐

- 웹 문서 작성 도구 : 구글 , 마이크로소프트에서는 웹 브라우저만으로 워드, 엑셀, 파워포인트 같은 애플리케이션 사용 가능
- 웹 애플리케이션은 웹 브라우저만 있으면 언제 어디서나 사용 가능

❖ 구글에서 제공하는 크롬 웹 스토어

- 웹 스토어에서 웹 브라우저상에서 실행되는 웹 애플리케이션 거래



자바스크립트의 활용

❖ 인터넷 연결 없이 웹 브라우저에서 실행 가능한 웹 애플리케이션

- 스마트 폰이나 스마트패드 내의 애플리케이션을 만들 때에도 자바스크립트 사용
 - 웹 워크를 사용하면 블랙 베리 스마트 폰과 플레이 북 스마트패드에 들어가는 애플리케이션 제작 가능

❖ 웹 애플리케이션의 장점

- 웹 브라우저가 돌아가는 데스크톱, 노트북, 스마트폰 등 모든 장치에서 사용 가능
- HTML5가 도입됨에 따라 웹 상에서 플래시나 실버라이트의 도움 없이도 게임과 같은 복잡한 프로그램 제작 가능
- 현재 크롬과 사파리 웹 브라우저에서는 3D 개발도 가능



표현식과 문장

❖ 표현식이란?

- 값을 만들어내는 간단한 코드

- Ex)

- 273
- $10 + 20 + 30 * 2$
- 'RintIanTta'

❖ 문장이란?

- 하나 이상의 표현식이 모인 것
- 문장이 모여 프로그램 구성
- 문장의 끝에는 세미콜론을 찍어 문장의 종결을 알려줌

- Ex)

- $10 + 20 + 30 * 2;$
- `var rintiantta = 'Rint' + 'Ian' + 'Tta';`
- `alert('Hello JavaScript..!');`
- 273;



키워드

❖ 키워드란?

- 자바스크립트가 처음 만들어질 때 정해진 특별한 의미가 있는 단어

표 1-2 자바스크립트의 키워드 (1)

break	else	instanceof	true
case	false	new	try
catch	finally	null	typeof
continue	for	return	var
default	function	switch	void
delete	if	this	while
do	in	throw	with

- W3C에서 자바스크립트 프로그램 작성시 추가 키워드

- const나 debugger 같은 키워드는 이미 일부 브라우저에서 사용

표 1-3 자바스크립트의 키워드 (2)

abstract	enum	int	short
boolean	export	interface	static
byte	extends	long	super
char	final	native	synchronized
class	float	package	throws
const	goto	private	transient
debugger	implements	protected	volatile
double	import	public	



식별자

❖ 식별자

- 자바스크립트에서 이름을 붙일 때 사용
- 식별자의 예
 - 변수명과 함수명
- 식별자 생성시 규칙
 - 키워드 사용 불가
 - 시작 문자 숫자 불가
 - 특수 문자는 _과 \$만 허용
 - 공백 문자 포함 불가

- Ex) `alpha`
`alpha10`
`_alpha`
`$alpha`
`AlPha`
`ALPHA`



식별자

❖ 식별자 생성 규칙

- 모든언어 사용 가능하나 알파벳 사용이 개발자들 사이 관례
- input, output 같은 의미 있는 단어 사용
- 생성자 함수의 이름은 항상 대문자로 시작
- 변수, 인스턴스, 함수, 메서드의 이름은 항상 소문자로 시작
- 여러 단어로 이뤄진 식별자는 각 단어의 첫 글자를 대문자로.
 - 식별자는 공백을 가질 수 없지만 이 규칙을 사용시 여러 단어로 이루어진 글자를 쉽게 읽을 수 있음
 - Ex)
 - will out ⇒ willOut
 - will return ⇒ willReturn
 - i am a boy ⇒ iAmABoy



식별자

❖ 자바스크립트의 식별자의 종류

표 1-4 자바스크립트 식별자의 종류

구분	단독으로 사용	다른 식별자와 함께 사용
식별자 뒤에 괄호 無	변수	속성
식별자 뒤에 괄호 有	함수	메서드

- `alert('Hello World')` ⇨ 함수
- `Array.length` ⇨ 속성
- `input` ⇨ 변수
- `prompt('Message', 'Defstr')` ⇨ 함수
- `Math.PI` ⇨ 속성
- `Math.abs(-273)` ⇨ 메서드



주석

❖ 주석이란?

- 프로그램 진행에 전혀 영향을 끼치지 않아야 함
- 코드의 특정 부분을 설명
- 1) HTML 태그 주석
 - <!-- -->로 문자열을 감싸 생성

```
<!DOCTYPE html>
<html>
<head>
  <!-- 주석입니다. -->
  <script>

    </script>
</head>
<body>
  <!-- <h1>주석입니다.</h1> -->
</body>
</html>
```



주석

❖ 주석이란?

■ 2) 자바스크립트 주석

- //를 사용해 한 줄 주석 표현
 - // 뒤의 문장은 실행되지 않음
 - Ex) // 주석문
- /*와 */을 사용해 여러 줄 주석 표현
 - Ex) /*

주석문
주석문
***/**

```
<script>  
    // 주석은 코드의 실행에 아무 영향을 미치지 않습니다.  
    /*  
        alert('Hello JavaScript');  
        alert('Hello JavaScript');  
        alert('Hello JavaScript');  
    */  
</script>
```



자바스크립트의 기본 문법

❖ 실습

■ 주석 처리 확인

```
01 <!DOCTYPE html >
03 <html >
04 <head>
05 <meta charset="utf-8" />
06 <title> javascript-comment.html </title>
07 </head>
08 <body>
09 <hr /><h1> 자바스크립트 주석 </h1><hr />
10 <script >
11 document.write("<h3>자바스크립트로 쓴 문장</h3>");
12 // h3 요소를 사용하여 제목으로 표시
13 /* document.write에는 xhtml 요소들을
14 사용할 수 있다. */
15 document.write("<p> p 요소를 이용한 문장 </p>");
16 document.write("<del> 가운데 한 줄이 생김 </del>");
17 </script>
18 </body>
19 </html>
```

