

웹 프로그래밍 1

7주차 : DOM (문서 객체)

응용소프트웨어학과 김태호

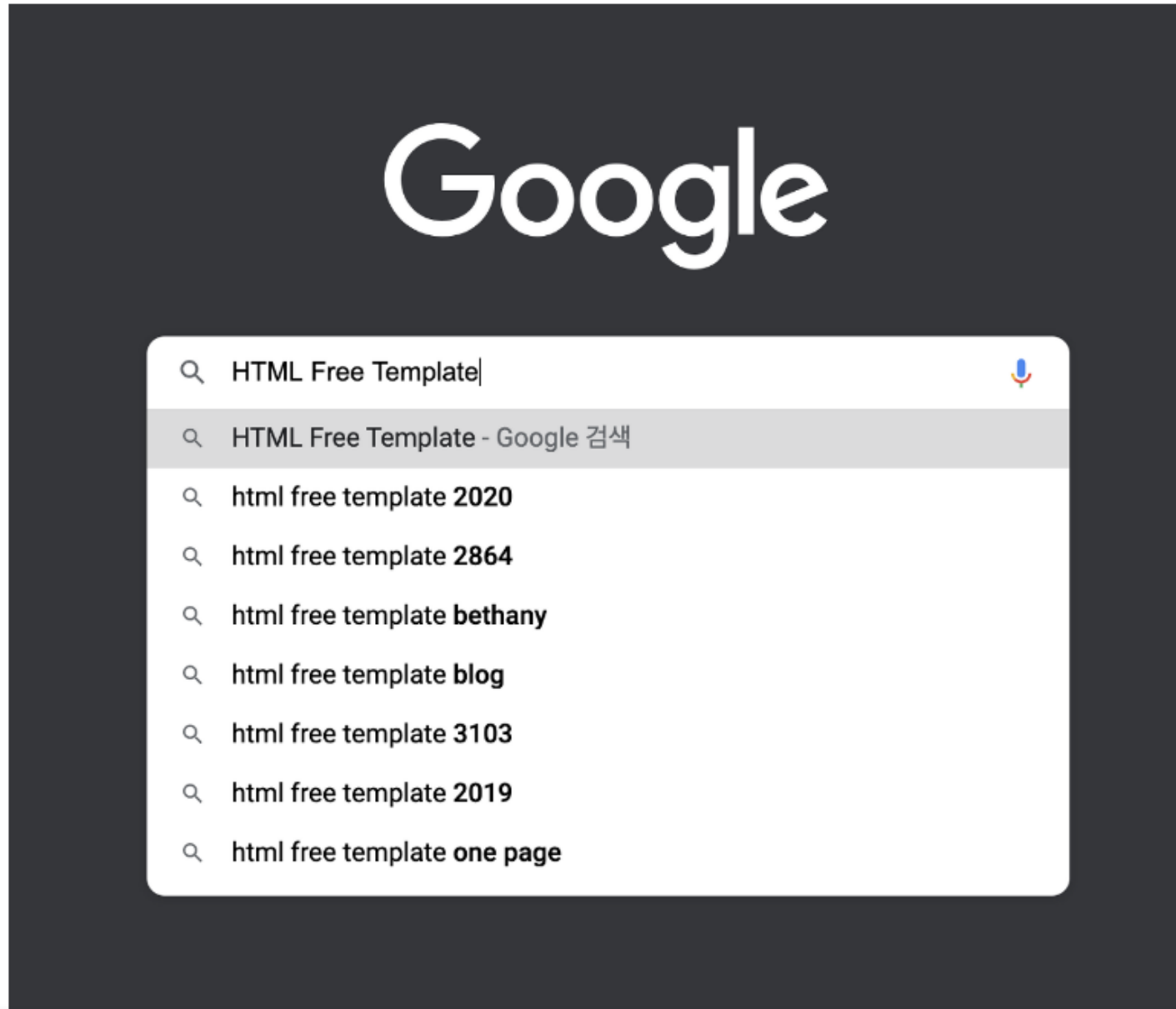


명지대학교
MYONGJI UNIVERSITY

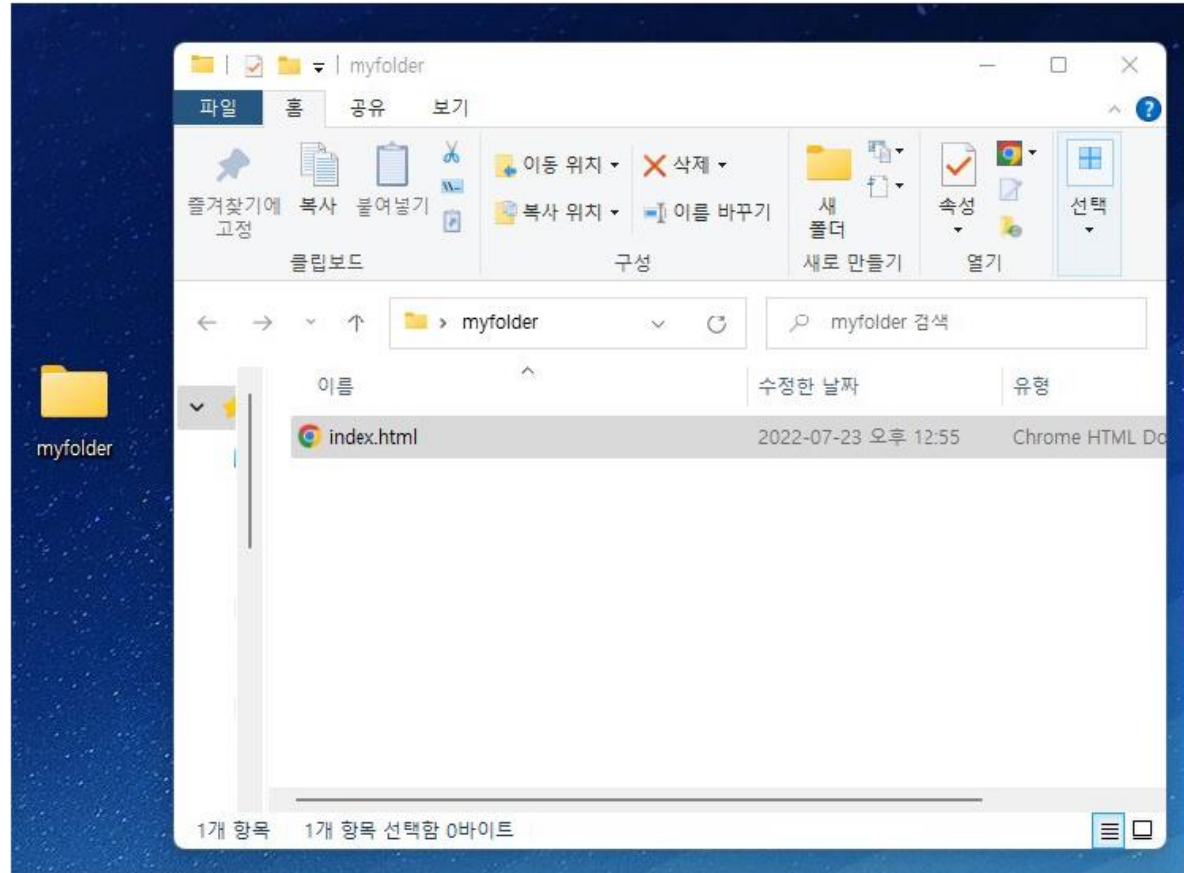
템플릿 사용



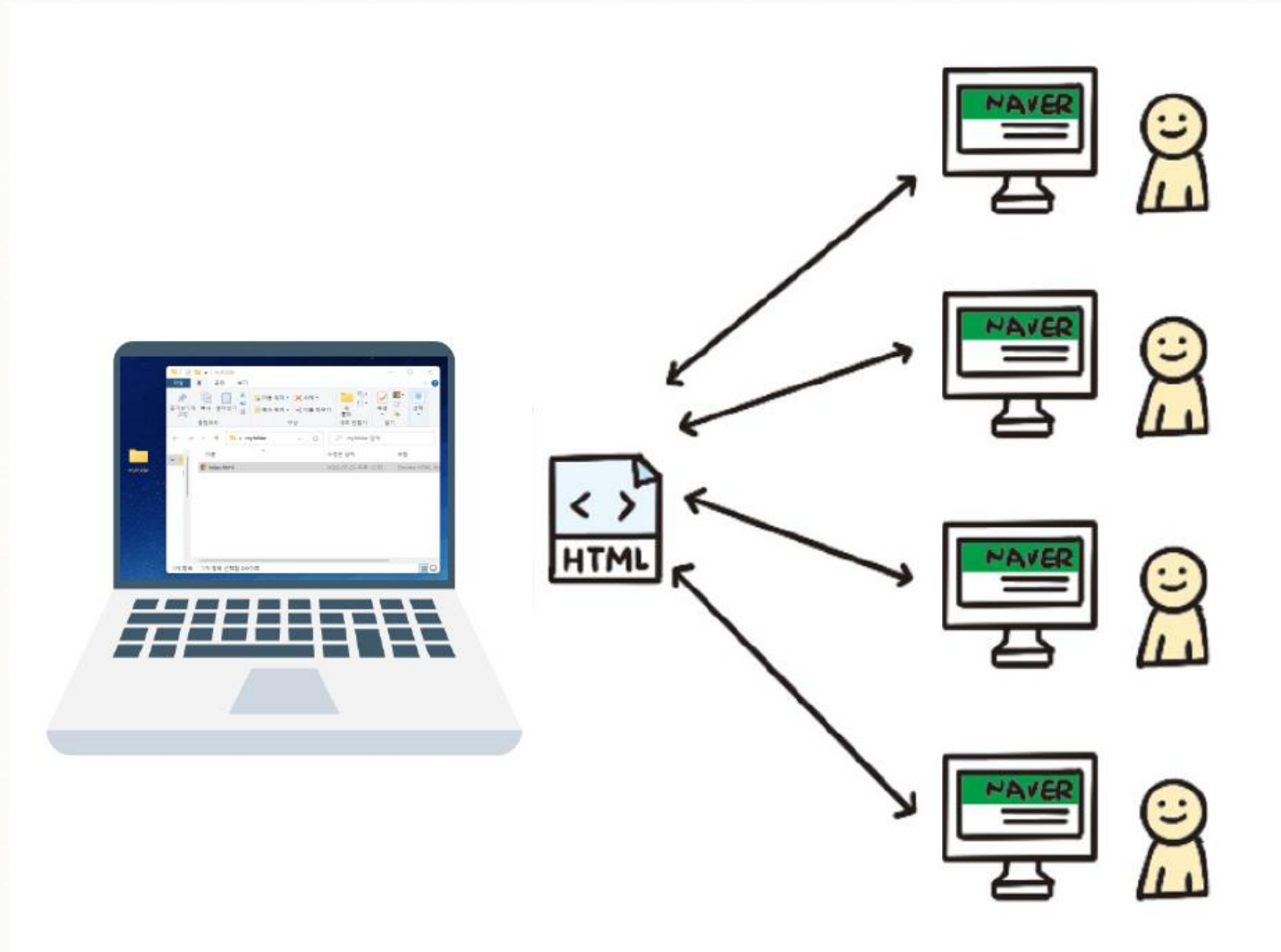
| HTML free template



Deploy



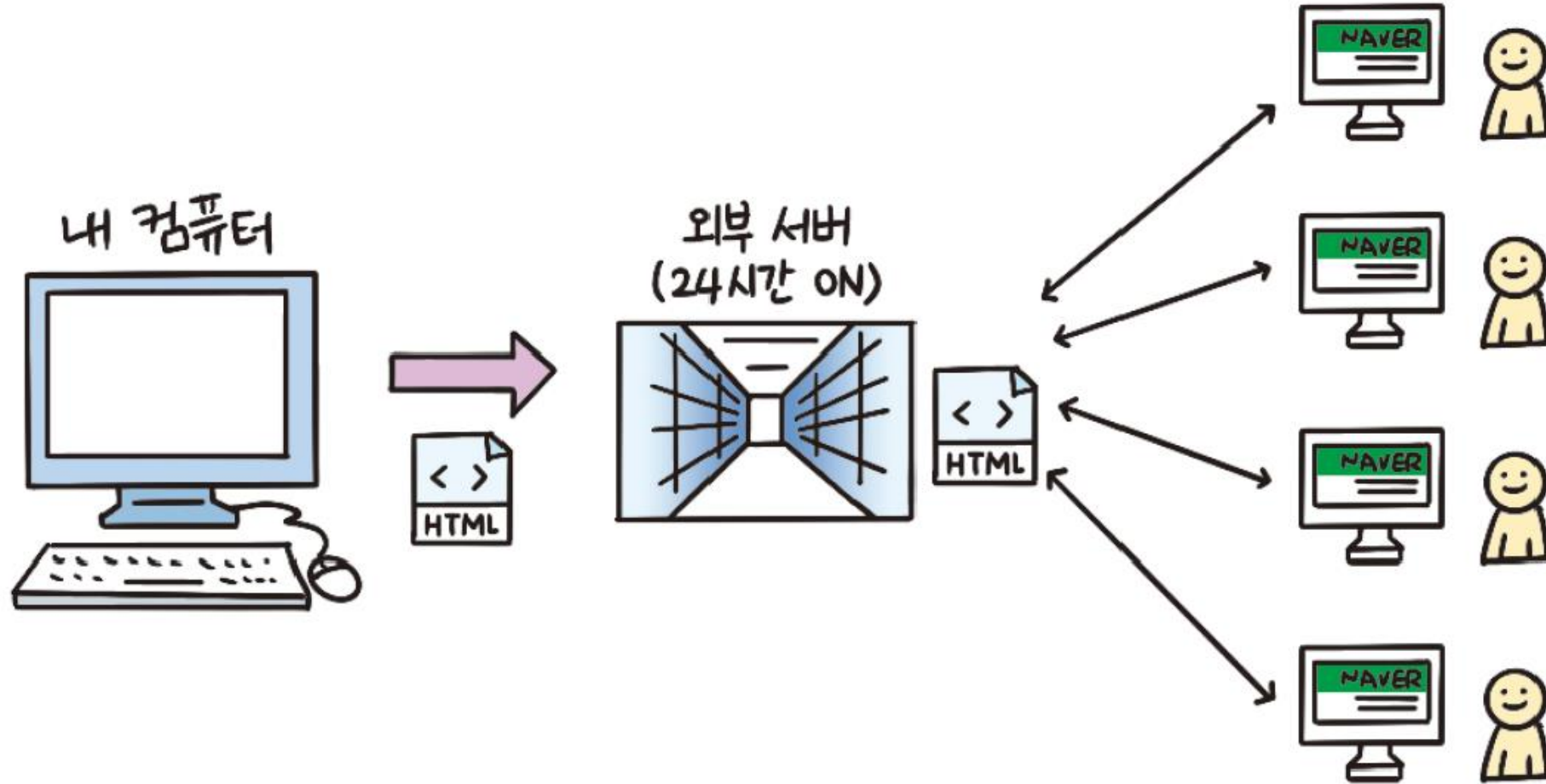
Deploy



| Deploy



| Deploy





Microsoft



Google Cloud



netlify



Vercel



HEROKU



cloudtype



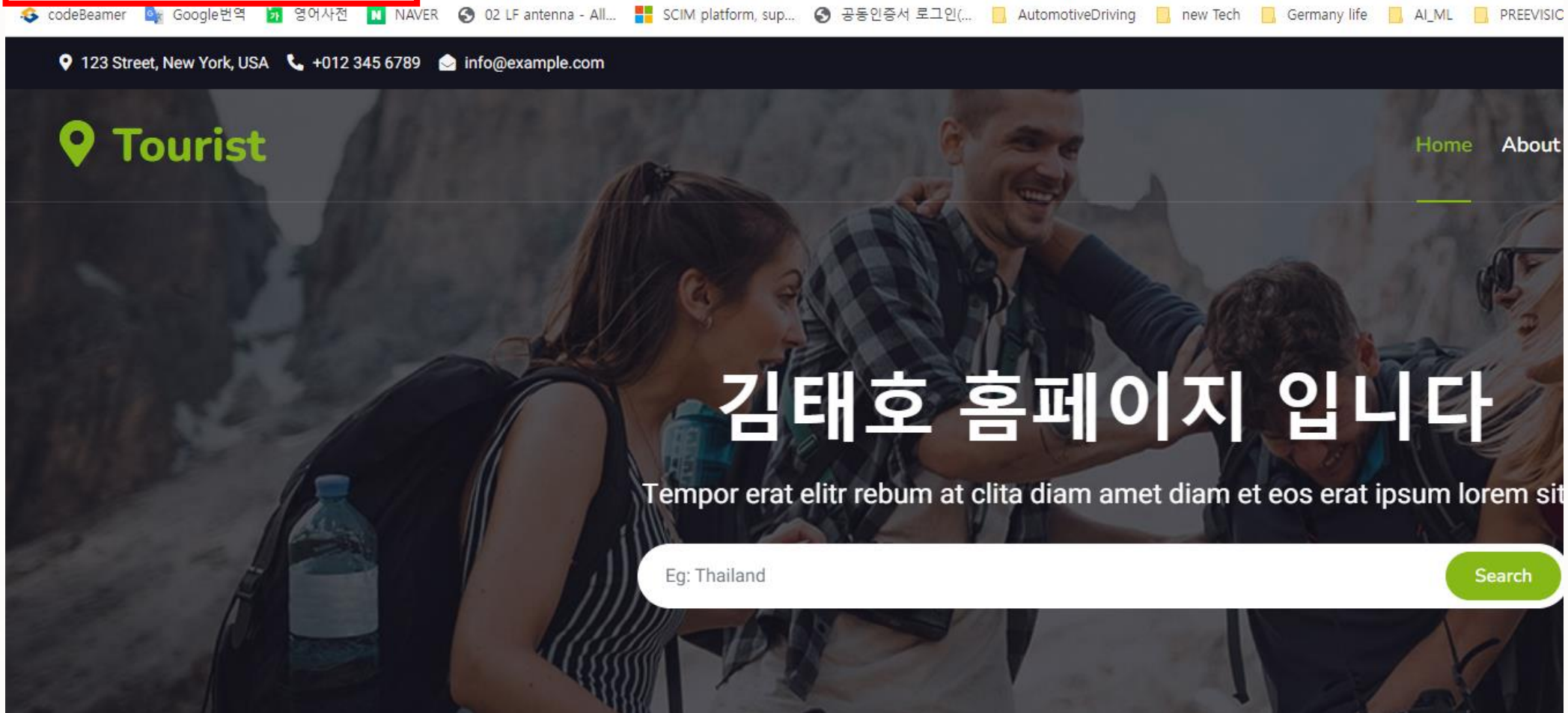
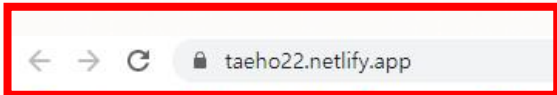
Cloudflare
Pages

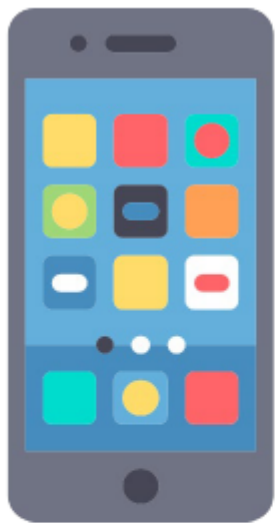
도메인 등록



|도메인 등록

https://taeho22.netlify.app/

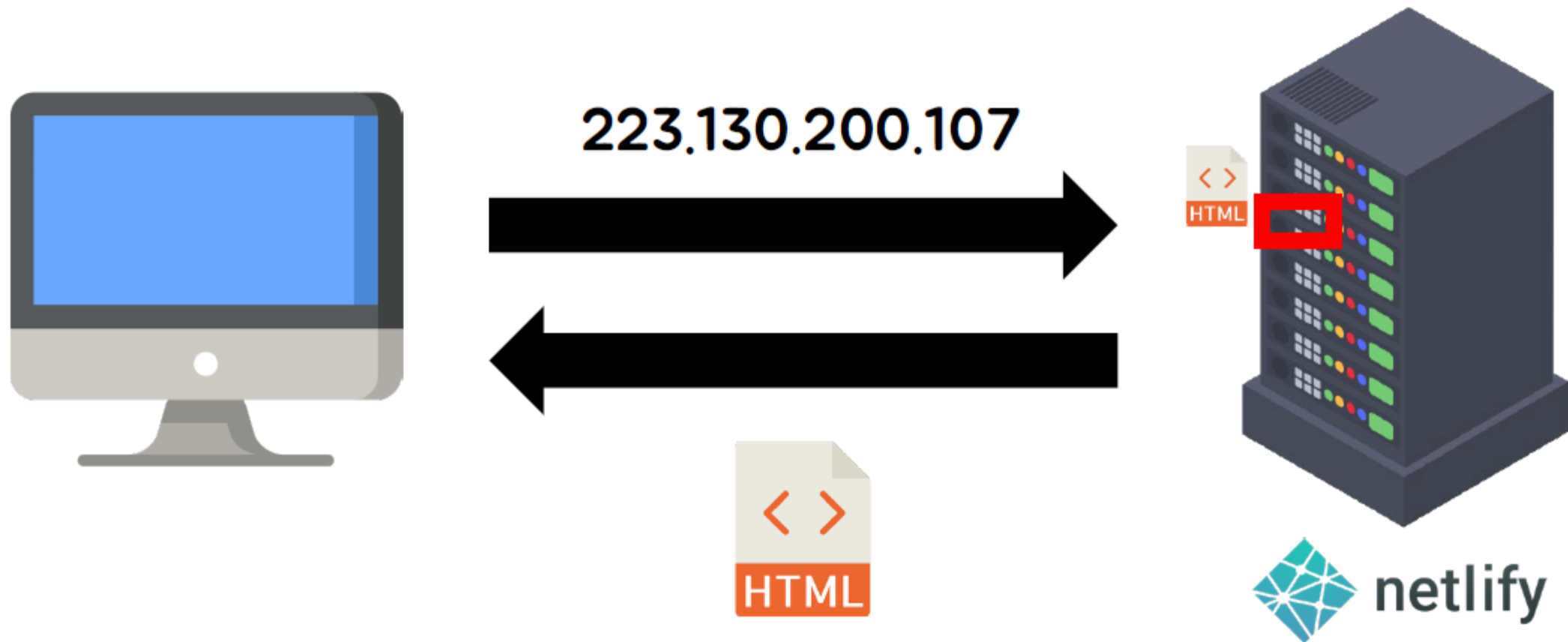




010-xxxx-xxxx



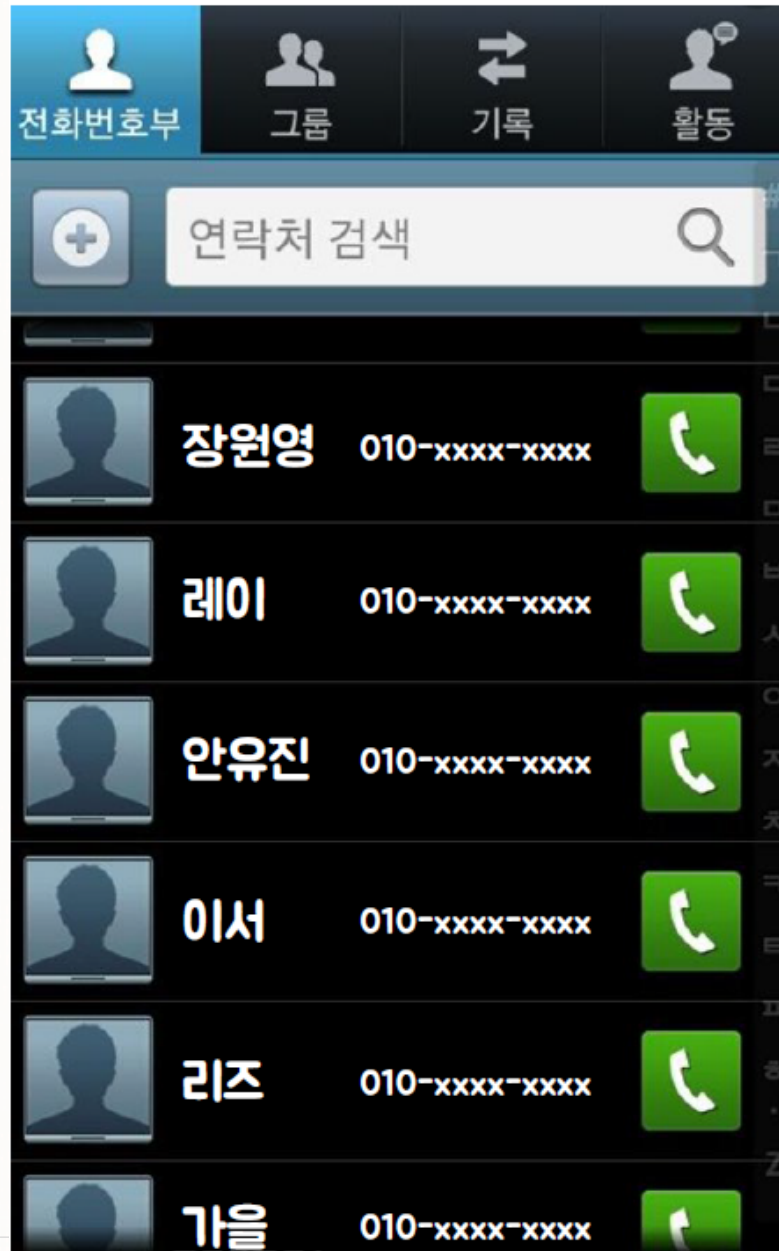
223.130.200.107

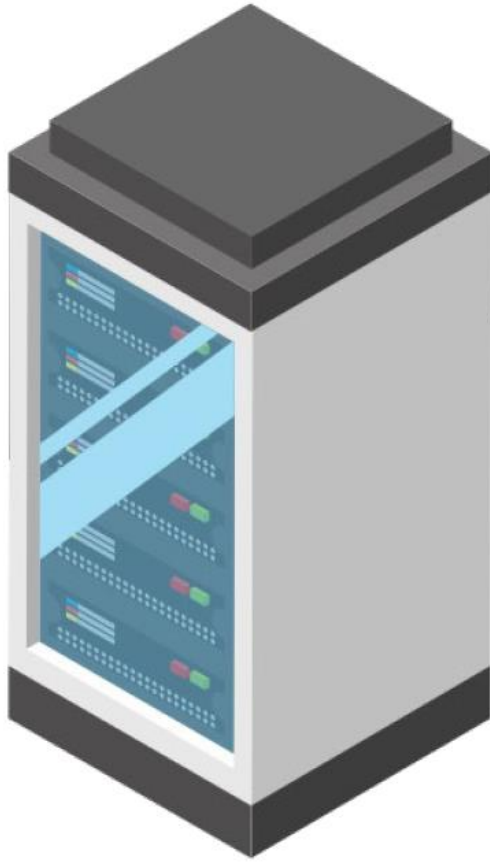


| 도메인 등록



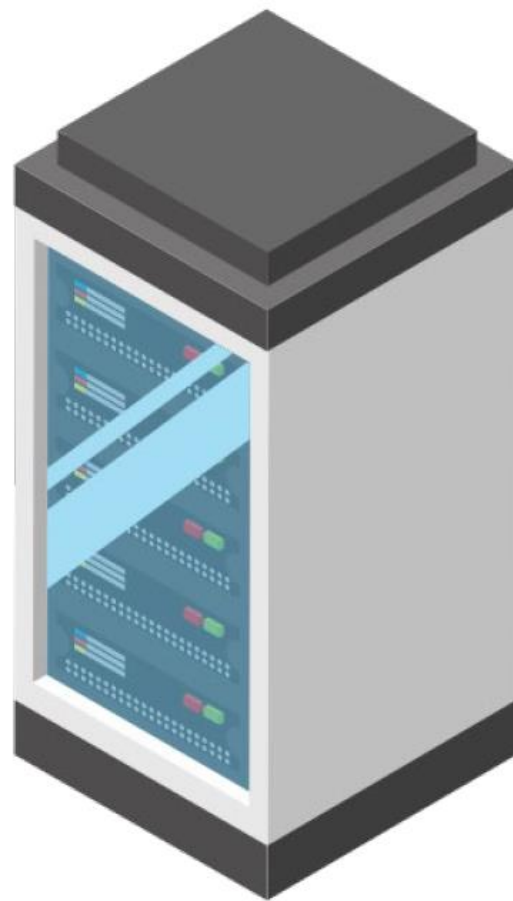
| 도메인 등록





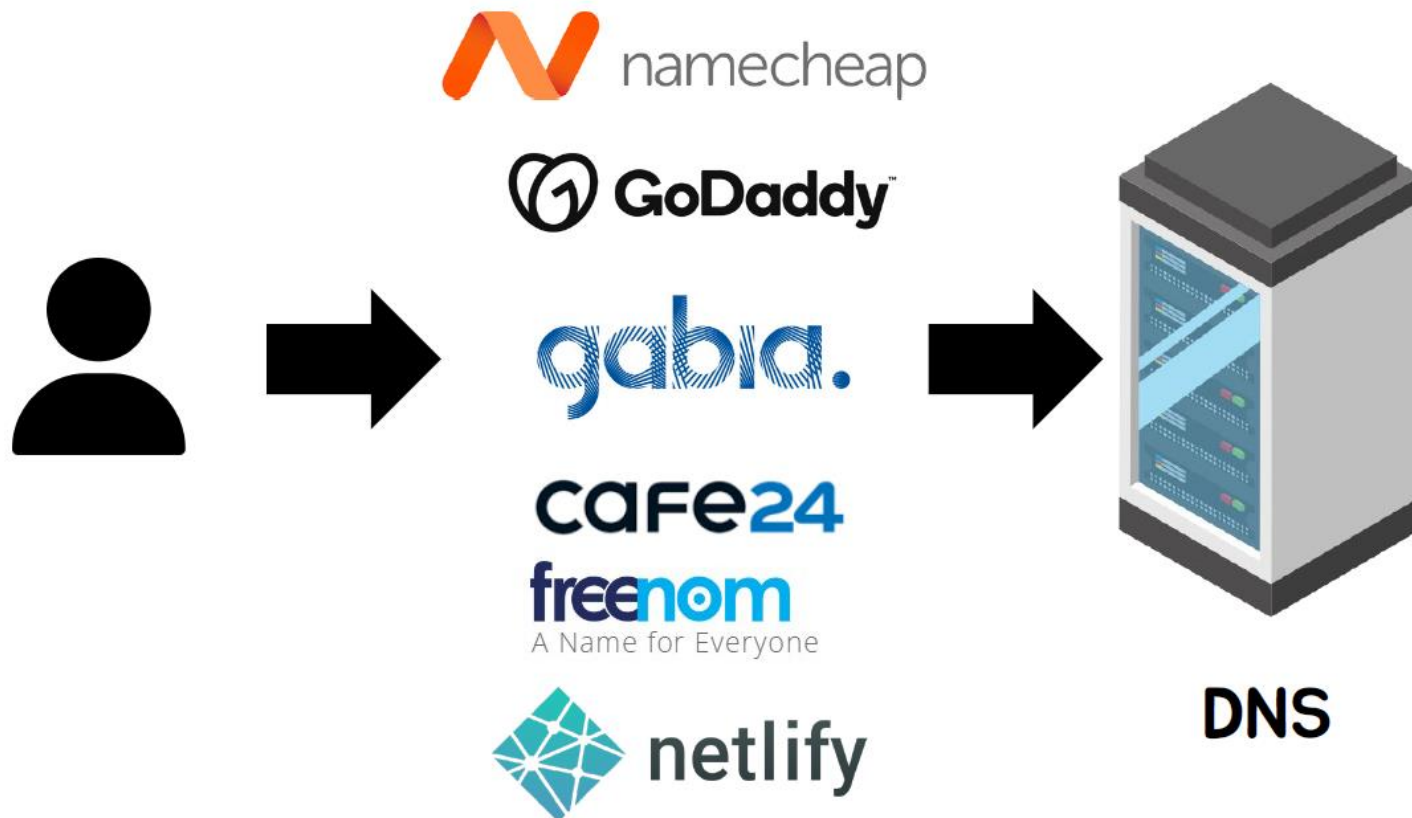
DNS
(Domain Name System)

naver.com = 223.130.200.107
google.com = 142.250.199.110
....



DNS
(Domain Name System)

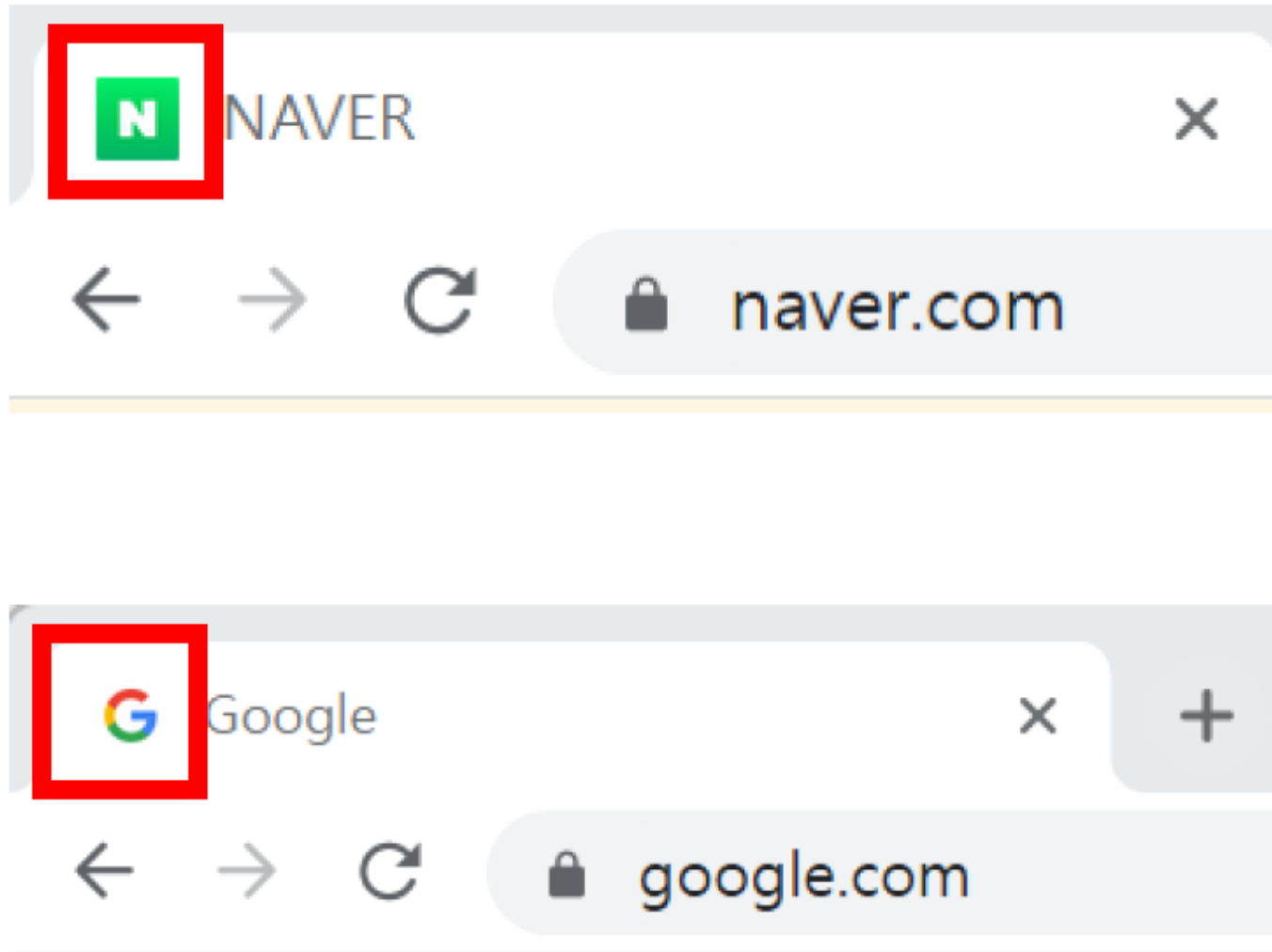
도메인 판매 네임서버 관리



Favicon(파비콘 적용)



| Favicon





<https://convertico.com/>

```
<link rel="shortcut icon" href="/favicon.ico" type="image/x-icon">  
<link rel="icon" href="/favicon.ico" type="image/x-icon">
```


검색엔진에 등록



| 검색엔진 원리

예티! 정보를 가져와.

NAVER

Google

구글 봇! 정보를 가져다 줘!

오케이! 정보 수집~!



<https://searchadvisor.naver.com/>

NAVER Search Advisor

웹마스터 가이드

웹마스터 도구

로그인

홈 시작하기 진단하기 성장하기 지원 및 커뮤니케이션

Search Advisor 와 함께 당신의 웹사이트를 성장시켜 보세요

네이버 검색 기초부터 문제 해결까지
궁금한 점이 있으신가요?

웹마스터 가이드 검색





robots.txt

sitemap.xml

robots.txt 모두 허용

```
User-agent: *  
Allow: /
```

Sitemap 생성기

<https://www.xml-sitemaps.com/>

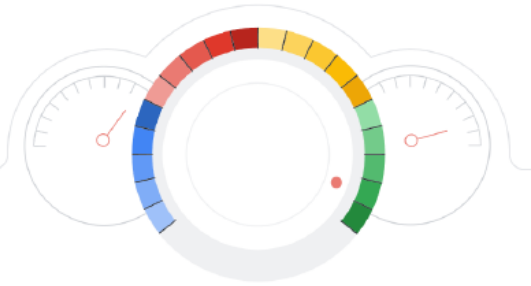
<https://search.google.com/search-console/about>

Google Search Console

Google 검색 실적 개선하기

Search Console 도구와 보고서를 사용하면 사이트의 검색 트래픽 및 실적을 측정하고, 문제를 해결하며, Google 검색결과에서 사이트가 돋보이게 할 수 있습니다.

시작하기



검색 애널리틱스로 콘텐츠 최적 화하기

사용자가 어떤 검색어를 통해 사이트를 방문하는지 알아보세요. 사이트의 노출수, 클릭수 및 Google 검색에서의 게재순위를 분석할 수도 있습니다.

문서객체 모델

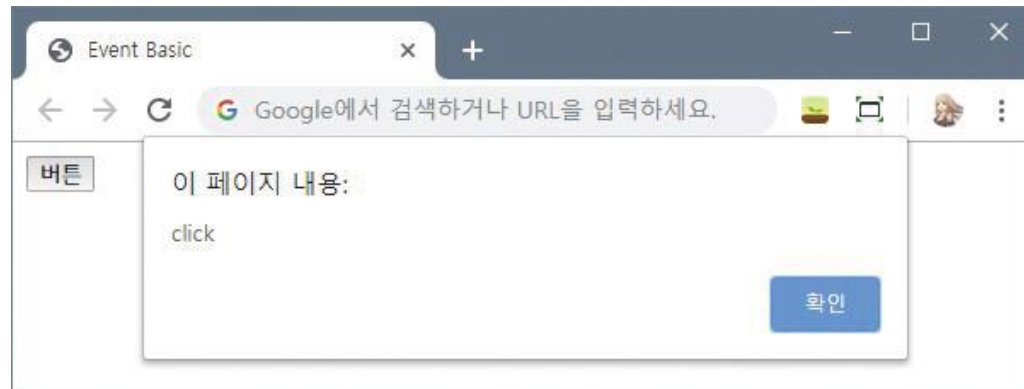


■ 학습목표

- 문서 객체와 문서 객체 모델의 개념을 이해합니다.
- 문서 객체를 선택하고 조작할 수 있습니다.
- 이벤트 종류를 알아보고 문서 객체에 이벤트를 연결합니다.

■ 내용

- 문서 객체 모델의 기본 용어와 개념
- 문서 객체 선택
- 문서 객체 조작
- 이벤트



■ 문서 객체 모델 기본 용어

- 문서 객체
 - 태그를 자바스크립트에서 사용할 수 있는 객체로 만든 것
- 문서 객체를 조작한다는 말은 결국 태그를 조작한다는 의미

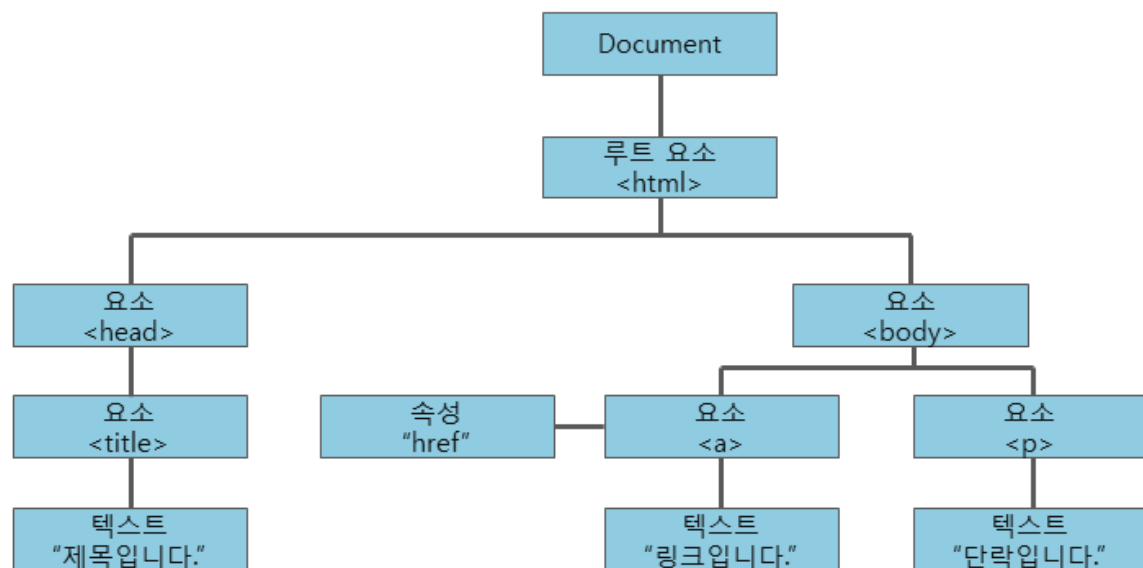
코드 10-1 dom_basic.html

```
<!DOCTYPE html>
<html>
<head>
  <title>Document Object Model</title>
  <script>

  </script>
</head>
<body>
  <h1>Header</h1>
  <p>Lorem ipsum dolor amet</p>
</body>
</html>
```

■ 문서 객체 모델(DOM) 기본 용어

- DOM, Document Object Model : HTML 문서에 접근하기 위한 일종의 인터페이스
- DOM은 W3C의 표준 객체 모델이며, 다음과 같이 계층 구조로 표현됩니다.



자바스크립트는 이러한 객체 모델을 이용하여 다음과 같은 작업을 할 수 있습니다.

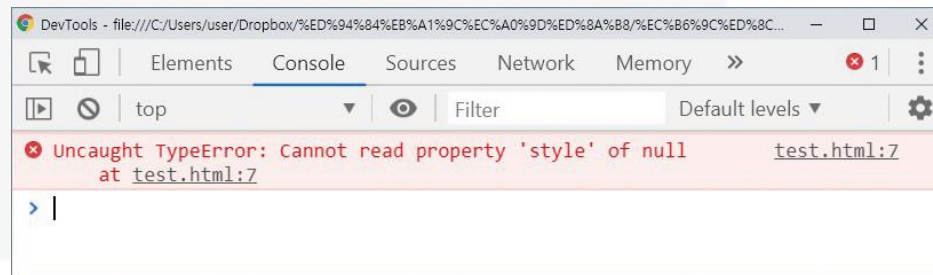
- 자바스크립트는 새로운 HTML 요소나 속성을 추가할 수 있습니다.
- 자바스크립트는 존재하는 HTML 요소나 속성을 제거할 수 있습니다.
- 자바스크립트는 HTML 문서의 모든 HTML 요소를 변경할 수 있습니다.
- 자바스크립트는 HTML 문서의 모든 HTML 속성을 변경할 수 있습니다.
- 자바스크립트는 HTML 문서의 모든 CSS 스타일을 변경할 수 있습니다.
- 자바스크립트는 HTML 문서에 새로운 HTML 이벤트를 추가할 수 있습니다.
- 자바스크립트는 HTML 문서의 모든 HTML 이벤트에 반응할 수 있습니다.

01 문서 객체 모델의 기본 용어와 개념

- 웹 페이지 실행 순서
- 실행 순서에 따른 문서 객체 사용 오류
 - 1. 문서 객체 사용 오류

코드 10-2 dom_fault.html

```
<!DOCTYPE html>
<html>
<head>
  <title>Document Object Model</title>
  <script>
    // h1 태그의 배경 색상을 변경합니다.
    document.querySelector('h1').style.backgroundColor = 'red';
    // h2 태그의 글자 색상을 변경합니다.
    document.querySelector('h2').style.color = 'red';
  </script>
</head>
<body>
  <h1>Process - 1</h1>
  <h2>Process - 2</h2>
</body>
</html>
```



01 문서 객체 모델의 기본 용어와 개념

- 웹 페이지 실행 순서
- 실행 순서에 따른 문서 객체 사용 오류
 - 해결 방법 1: script 태그 위치 옮기기

코드 10-3 dom_correct.html

```
<!DOCTYPE html>
<html>
<head>
  <title>Document Object Model</title>
</head>
<body>
  <h1>Process - 1</h1>
  <h2>Process - 2</h2>
  <script>
    // h1 태그의 배경 색상을 변경합니다.
    document.querySelector('h1').style.backgroundColor = 'red';
    // h2 태그의 글자 색상을 변경합니다.
    document.querySelector('h2').style.color = 'red';
  </script>
</body>
</html>
```



01 문서 객체 모델의 기본 용어와 개념

- 웹 페이지 실행 순서
- 실행 순서에 따른 문서 객체 사용 오류
 - 해결 방법 2: 이벤트 활용하기

코드 10-4 dom_event.html

```
<!DOCTYPE html>
<html>
<head>
  <title>Document Object Model</title>
  <script>
    window.onload = function () {
      // h1 태그의 배경 색상을 변경합니다.
      document.querySelector('h1').style.backgroundColor = 'red';
      // h2 태그의 글자 색상을 변경합니다.
      document.querySelector('h2').style.color = 'red';
    };
  </script>
</head>
<body>
  <h1>Process - 1</h1>
  <h2>Process - 2</h2>
</body>
</html>
```

- ‘문서 객체를 선택한다’.
 - 이미 존재하는 HTML 태그를 자바스크립트에서 문서 객체로 변환하는 것

표 10-1 문서 객체를 선택하는 메서드

구분	메서드	설명
1개 선택	document.getElementById(아이디)	아이디로 1개 선택
	document.querySelector(선택자)	선택자로 1개 선택
여러 개 선택	document.getElementsByName(이름)	name 속성으로 여러 개 선택
	document.getElementsByClassName(클래스)	class 속성으로 여러 개 선택
	document.querySelectorAll(선택자)	선택자로 여러 개 선택

02 문서 객체 선택

■ 문서 객체 선택

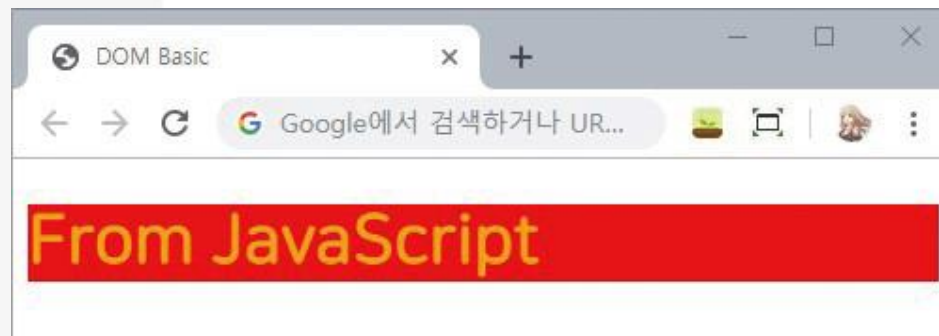
■ 1. getElementById() 메서드를 사용해 문서 객체 1개 선택하기

코드 10-6 select_id.html

```
<!DOCTYPE html>
<html>
<head>
  <title>DOM Basic</title>
  <script>
    // 이벤트를 연결합니다.
    window.onload = function () {
      // 문서 객체를 선택합니다.
      var header = document.getElementById('header');

      // 문서 객체를 조작합니다.
      header.style.color = 'orange';
      header.style.background = 'red';
      header.innerHTML = 'From JavaScript';
    };
  </script>
</head>
<body>
  <h1 id="header">Header</h1>
</body>
</html>
```

태그 내부를 의미하는 속성입니다.



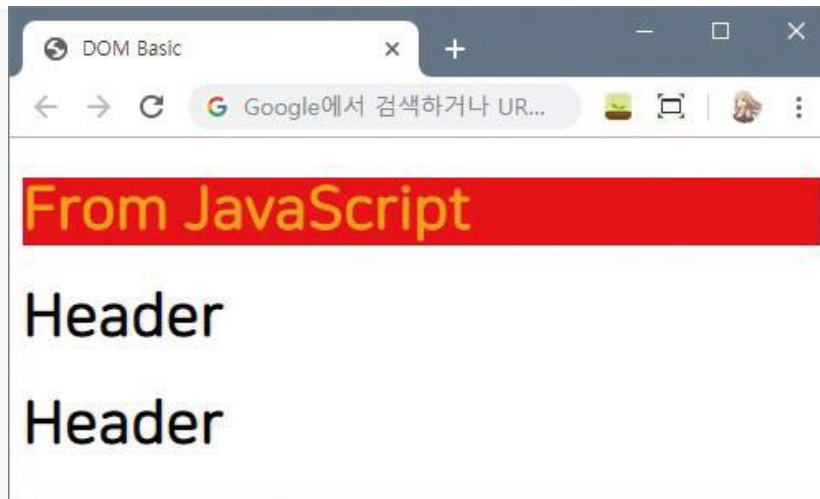
■ 문서 객체 선택

- 2. `querySelector()` 메서드를 사용해 문서 객체 1개 선택하기

코드 10-7 select_query.html

```
<!DOCTYPE html>
<html>
<head>
  <title>DOM Basic</title>
  <script>
    // 이벤트를 연결합니다.
    window.onload = function () {
      // 문서 객체를 선택합니다.
      var header = document.querySelector('h1');

      // 문서 객체를 조작합니다.
      header.style.color = 'orange';
      header.style.background = 'red';
      header.innerHTML = 'From JavaScript';
    };
  </script>
</head>
<body>
  <h1>Header</h1>
  <h1>Header</h1>
  <h1>Header</h1>
</body>
</html>
```



02 문서 객체 선택

■ 문서 객체 선택

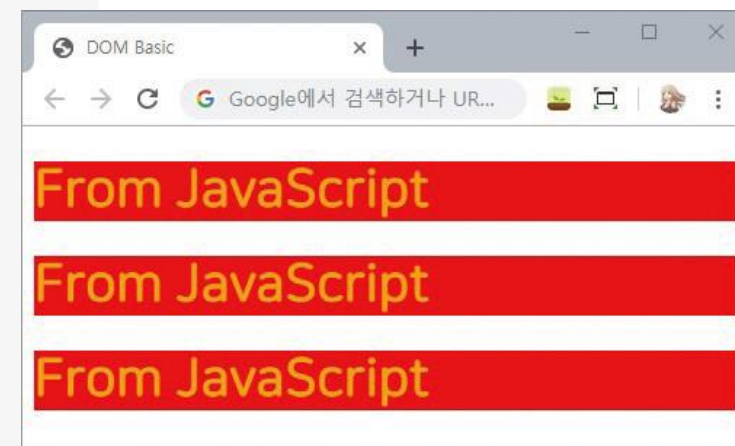
■ 3. querySelectorAll() 메서드를 사용해 문서 객체 여러 개 선택하기

코드 10-8 select_all.html

```
<!DOCTYPE html>
<html>
<head>
  <title>DOM Basic</title>
  <script>
    // 이벤트를 연결합니다.
    window.onload = function () {
      // 문서 객체를 선택합니다.
      var headers = document.querySelectorAll('h1');

      for (var i = 0; i < headers.length; i++) {
        // 변수를 선언합니다.
        var header = headers[i];
        // 문서 객체를 조작합니다.
        header.style.color = 'orange';
        header.style.background = 'red';
        header.innerHTML = 'From JavaScript';
      }
    };
  </script>
</head>
```

```
<body>
  <h1>Header</h1>
  <h1>Header</h1>
  <h1>Header</h1>
</body>
</html>
```



■ 글자 조작

■ 글자 속성

- 문서 객체 내부에 있는 글자를 조작할 때 사용

표 10-2 글자 속성

속성	설명
textContent	문서 객체 내부 글자를 순수 텍스트 형식으로 가져오도록 변경
innerHTML	문서 객체 내부 글자의 HTML 태그를 반영해 가져오도록 변경

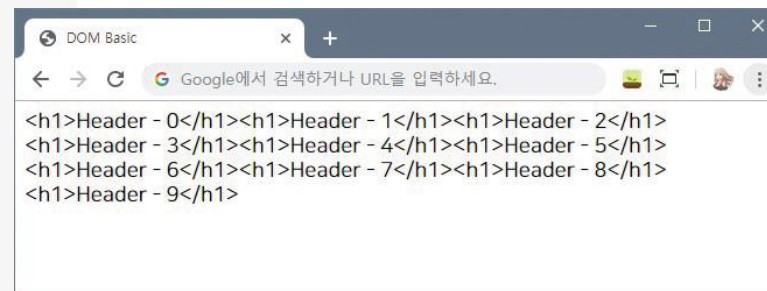
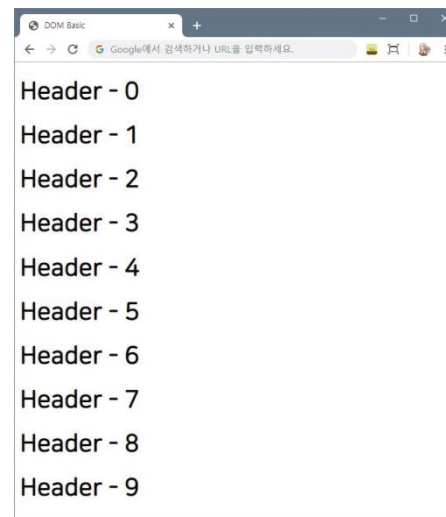
■ 내부 글자 변경

코드 10-9 control_text.html

```
<!DOCTYPE html>
<html>
<head>
  <title>DOM Basic</title>
  <script>
    // 이벤트를 연결합니다.
    window.onload = function () {
      // 변수를 선언합니다.
      var output = '';
      for (var i = 0; i < 10; i++) {
        output += '<h1>Header - ' + i + '</h1>';
      }

      ❶ // 문서 객체 내부의 글자를 변경합니다.
      document.body.textContent = output;
      ❷ // document.body.innerHTML = output;
    };
  </script>
</head>
<body>

</body>
</html>
```



■ 스타일 조작

- 자바스크립트로 CSS 속성 값을 추가·제거·변경 가능
- 1. 자바스크립트는 특수 문자 '-'을 식별자에 사용할 수 없으므로 오류 출력

```
❶ var header = document.getElementById('header');  
    header.style.background-color = 'red';
```

- 2. - 로 연결된 단어의 첫 글자를 대문자로 변경

```
❷ var header = document.getElementById('header');  
    header.style.backgroundColor = 'red';
```

표 10-3 스타일 식별자 변환

스타일시트의 스타일 속성	자바스크립트의 스타일 식별자
background-image	backgroundImage
background-color	backgroundColor
box-sizing	boxSizing
list-style	listStyle

■ 속성 조작

표 10-4 문서 객체의 속성 조작 메서드

메서드	설명
setAttribute(속성 이름, 속성 값)	속성 지정
getAttribute(속성 이름)	속성 추출

- setAttribute(), getAttribute() 메서드
 - 웹 표준에서 지정하지 않은 속성에 접근할 때 사용

```
<body>
  <div data-role="page">
    <div data-role="header"></div>
    <div data-role="content">

      </div>
    <div data-role="footer"></div>
  </div>
</body>
```

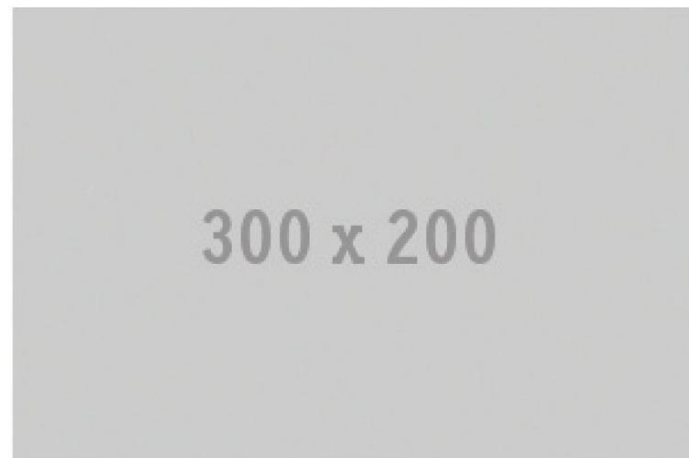
그림 10-2 jQuery Mobile 코드

- 문서 객체 속성 조작
 - 1. img 태그 속성 조작하기

코드 10-11 control_attribute.html

```
<!DOCTYPE html>
<html>
<head>
  <title>DOM Basic</title>
  <script>
    // 이벤트를 연결합니다.
    window.onload = function () {
      // 변수를 선언합니다.
      var image = document.getElementById('image');

      // 속성을 변경합니다.
      image.src = 'http://placeholder.it/300x200';
      image.width = 300;
      image.height = 200;
    };
  </script>
</head>
<body>
  <img id="image">
</body>
</html>
```



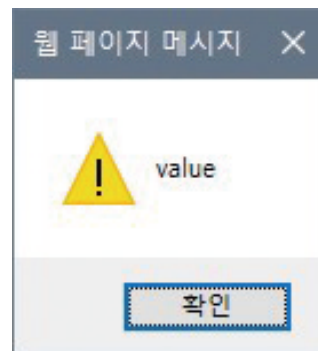
- 문서 객체 속성 조작
 - 2. body 태그 속성 조작하기

코드 10-12 control_body.html

```
<!DOCTYPE html>
<html>
<head>
  <title>DOM Basic</title>
  <script>
    // 이벤트를 연결합니다.
    window.onload = function () {
      // 속성을 지정합니다.
      document.body.setAttribute('data-custom', 'value');

      // 속성을 추출합니다.
      var dataCustom = document.body.getAttribute('data-custom');
      alert(dataCustom);
    };
  </script>
</head>
<body>

</body>
</html>
```



```
<!DOCTYPE html>
<html>
  <head>...</head>
  <body data-custom="value"></body>
</html>
```

그림 10-3 검사로 속성 조작 확인

- 문서 객체를 사용한 시간 표시
 - 2. body 태그 속성 조작하기

코드 10-13 dom_clock.html

```
<!DOCTYPE html>
<html>
<head>
  <title>Clock</title>
  <script>
    // 이벤트를 연결합니다.
    window.onload = function () {
      // 문서 객체를 선택합니다.
      var clock = document.getElementById('clock');
      // 1초마다 함수를 실행합니다.
      setInterval(function () {
        var now = new Date();
        clock.innerHTML = now.toString();
      }, 1000);
    };
  </script>
</head>
<body>
  <h1 id="clock"></h1>
</body>
</html>
```



■ 이벤트 용어 정리

- 이벤트 속성 (event property) - onload
- 이벤트 이름(event name) 또는 이벤트 타입(event type) - on을 제외한 load
- 이벤트 리스너(event listener) 또는 이벤트 핸들러(event handler)
 - 이벤트 속성에 넣는 함수

■ 기본적으로 지원하는 이벤트

- 마우스 이벤트
- 키보드 이벤트
- HTML 프레임 이벤트
- HTML 입력 양식 이벤트
- 사용자 인터페이스 이벤트
- 구조 변화 이벤트
- 터치 이벤트

```
onblur onfocus onfocusin onfocusout  
onload onresize onscroll onunload  
onclick ondblclick onmousedown  
onmouseup onmousemove onmouseover  
onmouseout onmouseenter  
onmouseleave onchange onselect  
onsubmit onkeydown onkeypress  
onkeyup onerror
```

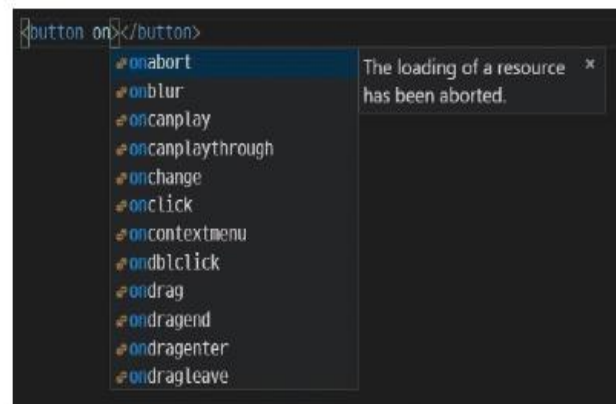


그림 10-4 이벤트 속성 종류와 통합 개발 환경의 속성 자동 완성

■ 이벤트 연결

- 이벤트 모델(event model) - 문서 객체에 이벤트를 연결하는 방식

표 10-5 이벤트 모델 종류(이벤트 연결 방식)

구분	종류
DOM 레벨 0	<ul style="list-style-type: none">• 인라인 이벤트 모델• 고전 이벤트 모델
DOM 레벨 2	<ul style="list-style-type: none">• 마이크로소프트 인터넷 익스플로러 이벤트 모델• 표준 이벤트 모델

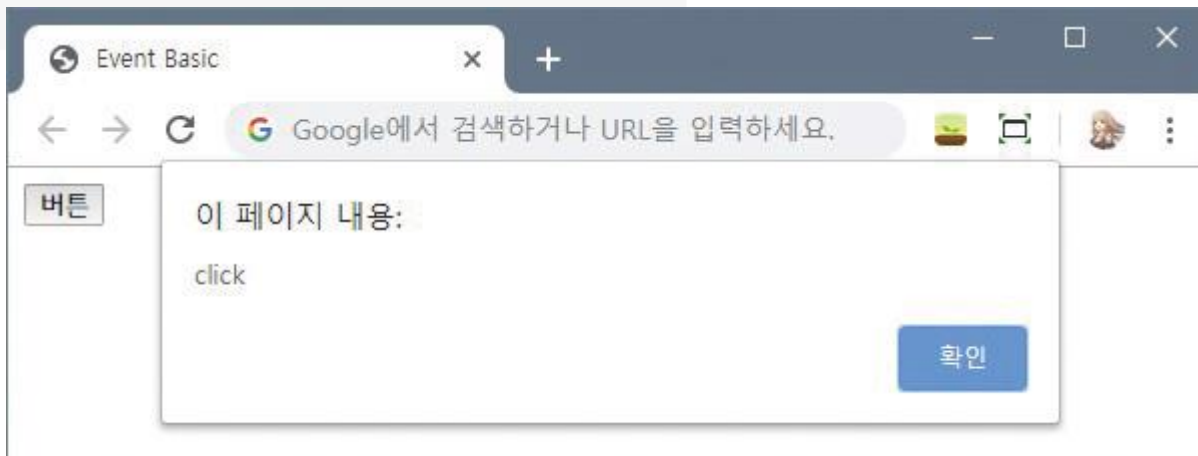
- 인라인 이벤트 모델
 - HTML 태그 내부에 자바스크립트 코드를 넣어 이벤트를 연결하는 방식

■ 이벤트 연결

■ 1. 인라인 이벤트 모델 사용하기

코드 10-14 event_inline.html

```
<!DOCTYPE html>
<html>
<head>
  <title>Event Basic</title>
</head>
<body>
  <button onclick="alert('click')">버튼</button>
</body>
</html>
```



- 이벤트 연결
 - 1. script 태그에 인라인 이벤트 모델 사용하기

코드 10-15 event_inlineWithScript.html

```
<!DOCTYPE html>
<html>
<head>
  <title>Inline Event</title>
  <script>
    function buttonClick() {
      alert('click');
    }
  </script>
</head>
<body>
  <button onclick="buttonClick()">버튼</button>
</body>
</html>
```

■ 이벤트 연결

- 고전 이벤트 모델 - 과거에 표준으로 정의되어 많이 사용하던 이벤트 모델

■ 고전 이벤트 연결

코드 10-16 event_tradition.html

```
<!DOCTYPE html>
<html>
<head>
  <title>Traditional Event</title>
  <script>
    // 이벤트를 연결합니다.
    window.onload = function () {
      // 문서 객체를 선택합니다.
      var button = document.getElementById('button');

      // 이벤트를 연결합니다.
      button.onclick = function () {
        alert('click');
      };
    };
  </script>
</head>
<body>
  <button id="button">버튼</button>
</body>
</html>
```

이벤트 리스너

■ 이벤트 사용


- 이벤트 객체를 사용하면 이벤트와 관련한 정보를 알아낼 수 있음

■ 이벤트 정보

코드 10-18 event_object.html

```
<!DOCTYPE html>
<html>
<head>
  <title>Event Object</title>
  <script>
    window.onload = function (event) {
      alert(event);
    };
  </script>
</head>
<body>

</body>
</html>
```



■ 기본 이벤트 제거

코드 10-19 event_default.html

```
<!DOCTYPE html>
<html>
<head>
  <title>Traditional Event - Default Event</title>
  <script>
    // 이벤트를 연결합니다.
    window.onload = function () {
      // 문서 객체를 선택합니다.
      var button = document.getElementById('button');

      // 이벤트를 연결합니다.
      button.onclick = function () {
        // 기본 이벤트를 제거합니다.
        return false;
      };
    };
  </script>
</head>
<body>
  <a id="button" href="http://hanbit.co.kr">버튼</a>
</body>
</html>
```



실습1 : 무엇이 문제 일까요?

- 다음 코드의 문제와 해결방법은?

```
<!DOCTYPE html>
<html lang="en">
  <head>
    <title>Documen Object Model</title>
    <script>
      var element = document.getElementById("element");
      element.style.color = "red";
      element.innerHTML = "From JavaScript";
    </script>
  </head>
  <body>
    <h1 id="element">This is a Test</h1>
  </body>
</html>
```

실습 2: 다음코드의 실행결과는?

```
<!DOCTYPE html>
<html lang="en">
  <head>
    <title>Documen Object Model</title>
    <h1>Header -1</h1>
    <h1>Header -2</h1>
    <h1>Header -3</h1>
  </head>
  <body>
    <script>
      var header = document.querySelector("h1");
      header.style.color = "white";
      header.style.backgroundColor = "black";
    </script>
  </body>
</html>
```

과제(Assignment)

7주차 : 자바스크립트

제출방법 : 다음 문제에 대해서 각각 폴더를 만들어서 HTML 코드를 작성합니다.
최종적으로 학번_이름.zip 으로 압축해서 e-class에 업로드 합니다.
기한 : 4/20 (목) 자정



| # 개인 소개 홈페이지 작성 (Update)

- 1) 요소 선택하기: HTML 문서 내의 특정 요소를 선택.
document.getElementById, document.querySelector, document.querySelectorAll 등의 메서드를 사용
- 2) 요소 스타일 변경하기: 선택한 요소의 스타일을 변경하는 기능을 구현. element.style 객체를 사용하여 요소의 스타일을 변경
- 3) 요소 속성 변경하기: 선택한 요소의 속성을 변경하는 기능을 구현. element.setAttribute와 element.getAttribute 메서드를 사용하여 요소의 속성을 변경
- 4) 요소 생성하기: HTML 문서 내에 새로운 요소를 생성하는 기능을 구현.
document.createElement 메서드를 사용하여 새로운 요소를 생성하고, element.appendChild 메서드를 사용하여 생성한 요소를 다른 요소에 추가
- 5) 이벤트 처리하기: 선택한 요소에 이벤트를 등록하여 이벤트가 발생했을 때 처리할 함수를 작성하는 기능을 구현. element.addEventListener 메서드를 사용하여 이벤트를 등록하고, 이벤트가 발생했을 때 처리할 함수를 작성