



HTML

HTML(Hyper Text Markup Language)

웹개발자도구 - HTML 수업

● <https://opentutorials.org/module/1892/11049>

2023.01.28

개발 도구

- atom (추천 확장기능 emmet) → Free
- sublime text (추천 확장기능 emmet)
- bracket
- 열린태그 / 닫힌태그 항상 해두기

<p>

- CSS → 행과 열을 구분 가능

p태그의 대안 →

- A forced line-break의 줄임
- 줄바꿈 한칸
- 두칸은 br두번
- 단락 구별은 p태그가 더 좋다.

2023.01.29

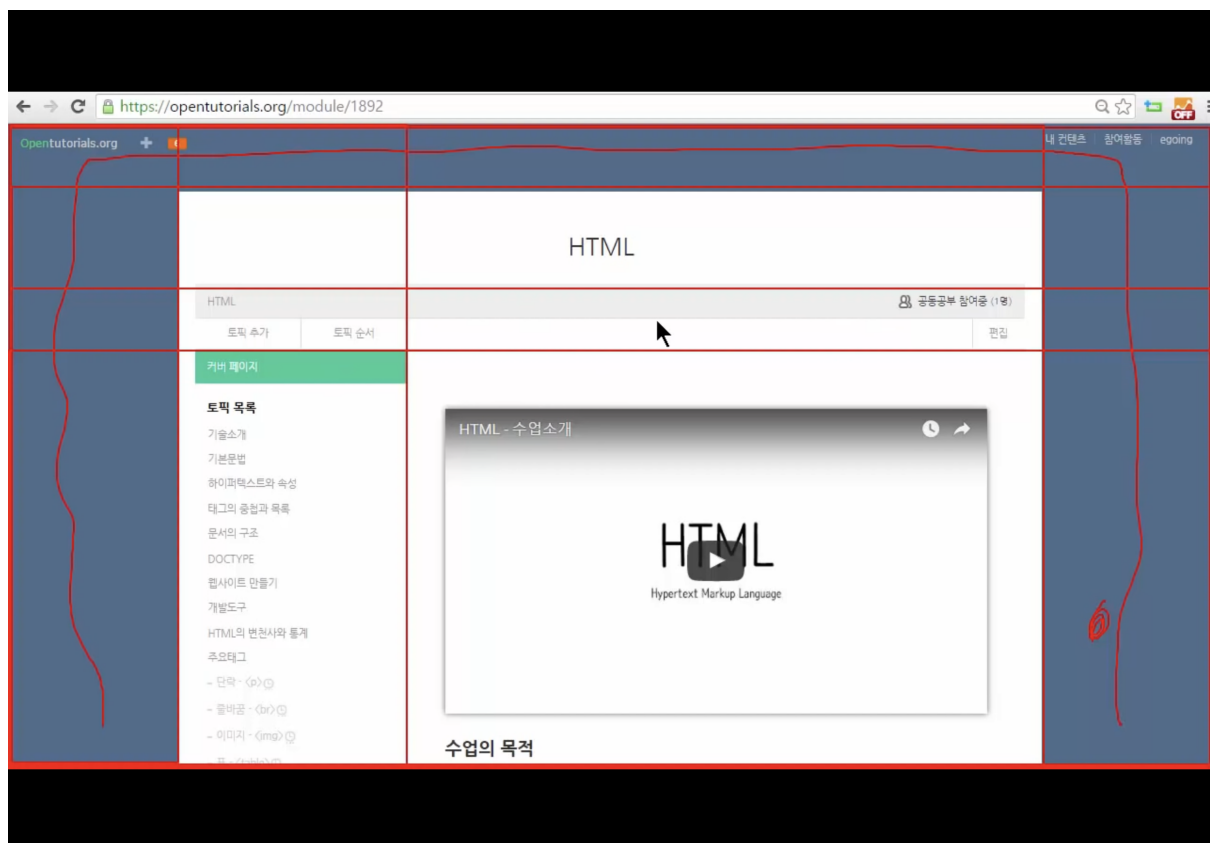

```

```

- src
- width
- height
- alt → 대체제
- title

<table> - 표

- <td> → table data
- <tr>
- <table>
- <table border → 이쁘게 꾸미고 싶으면 css로 가능
- layout 잡기
- 과거의 방법 → 현재는 잘 사용x



```
<html>
  <body>
    <table border="2"
      <table>
        <tr>
```

```

        <td>이름</td> <td>성별</td> <td>주소</td>
    </tr>
    <tr>
        <td>박준영</td> <td>남자</td> <td>구미</td>
    </tr>
    <tr>
        <td>최유빈</td> <td>여자</td> <td>구미</td>
    </tr>
</table>
</body>
</html>

```

- thead
- th
- tbody
- tfoot
- tr
- td

```

<html>
  <body>
    <table border="2">
      <thead>
        <tr>
          <th>이름</th> <th>성별</th> <th>주소</th> <th>회비</th>
        </tr>
      </thead>
      <tbody>
        <tr>
          <td>박준영</td> <td>남자</td> <td>구미</td> <td>1000</td>
        </tr>

```

```

        <tr>
            <td>최유빈</td> <td>여자</td> <td>구미</td> <td>500</td>
        </tr>
    </tbody>
    <tfoot>
        <tr>
            <td>합계</td> <td></td> <td></td> <td>1500</td>
        </tr>
    </tfoot>
</body>
</html>

```

- 표 병합

- 수직병합

- td rowspan

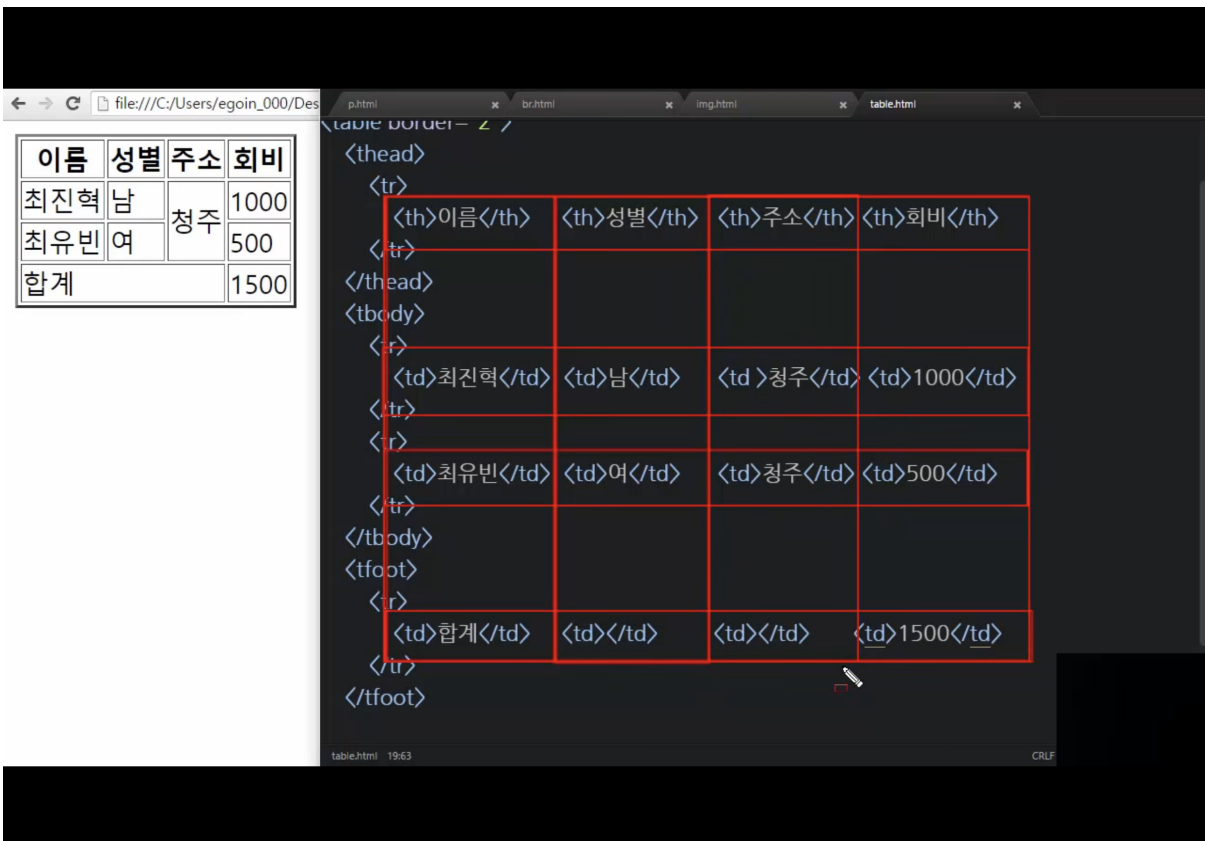
- 수평병합

- td colspan

```

<html>
  <body>
    <table border="2">
      <thead>
        <tr>
          <th>이름</th> <th>성별</th> <th>주소</th> <th>회비</th>
        </tr>
      </thead>
      <tbody>
        <tr>
          <td>박준영</td> <td>남자</td> <td rowspan="2">구미</td> <td>1000</td>
        </tr>
        <tr>
          <td>최유빈</td> <td>여자</td> <td>500</td>
        </tr>
      </tbody>
      <tfoot>
        <tr>
          <td colspan="3">합계</td> <td>1500</td>
        </tr>
      </tfoot>
    </body>
  </html>

```



2023.01.30

<form>

- input - (text = box)
- input - submit
- ***** → type = password
- form action → input address

```

<html>
  <body>
    <form action="http://localhost/login.php">
      <p>아이디 : <input type="text" name="id"></p>
      <p>비밀번호 : <input type="password" name="pwd"></p>
      <p>주소 : <input type="text" name="address"></p>
      <input type="submit">
    </form>
  </body>
</html>

```

form_text - 텍스트 입력

- value
- textarea

```
<html>
  <body>
    <form action="">
      <p>id : <input id : type="text" name="id" value="input id"></p>
      <p>password : <input type="text" name="pwd" value="input pwd"></p>
      <p><textarea name="메모" id="" cols="50" rows="10">memo</textarea></p>
    </form>
  </body>
</html>
```

Dropdown List

- option - value
- multiple → 다중 선택 가능
- 다중선택콤보박스

```
<html>
  <head>
    <meta charset="utf-8">
  </head>
  <body>
    <form action="http://localhost/color.php">
      <h1>색상</h1>
      <select name="color" id="색상">
        <option value="red">붉은색</option>
        <option value="blue">파란색</option>
        <option value="black">검은색</option>
      </select>

      <h1>색상2(다중선택)</h1>
      <select name="color2" multiple>
        <option value="red">붉은색</option>
        <option value="blue">파란색</option>
        <option value="black">검은색</option>
      </select>
      <input type="submit">
    </form>
  </body>
</html>
```

선택

- input type → radio
- input name → 같은 name 값으로 설정하면 하나만 선택이 가능하고 나머지는 해제되어진다. → 다른 name값 또는 name값을 설정하지 않으면 다중 선택 가능
- 다중 선택 → checkbox
- 선택한것 → 데이터서버로 전송'

```

<html>
  <head>
    <meta charset="utf-8">
  </head>
  <body>
    <form action="http://localhost/order.php">
      <p>
        <h1>색상(단일선택)</h1>
        붉은색 : <input type="radio" name="color" value="red">
        검은색 : <input type="radio" name="color" value="black" checked>
        파란색 : <input type="radio" name="color" value="blue">
      </p>

      <p>
        <h1>사이즈(다중선택)</h1>
        95 : <input type="checkbox" name="size" value="95">
        100 : <input type="checkbox" name="size" checked value="100">
        105 : <input type="checkbox" name="size" checked value="105">
      </p>
      <input type="submit">
    </form>
  </body>
</html>

```

버튼

- input type → submit
- input type → button
 - js → onclick
- input type → reset

```

<html>
  <head>
    <meta charset="utf-8">
  </head>
  <body>
    <form action="http://localhost/form.php">
      <input type="text" name="" id="">
      <input type="submit" value="전송">
      <input type="button" value="버튼" onclick="alert('hello world')">
      <input type="reset">
    </form>
  </body>
</html>

```

hidden field

- 서버로 데이터 전송 → UI 필요없거나 몰래 서버쪽으로 데이터를 전송해야할때 사용

```

<html>
  <head>
    <meta charset="utf-8">
  </head>
  <body>
    <form action="http://localhost/hidden.php">

```

```

        <input type="text" name="id">
        <input type="hidden" name="hide" value="egoing">
        <input type="submit">
    </form>
</body>
</html>

```

label

- label로 감싸주면 무엇인가의 이름표이다 라는것을 정확하게 기술이 가능하다.
- 각각의 label들이 누구의 이름표인지를 연결시켜줘야한다
 - for 사용
- 시각적인 변화는 없다.
- label 설정방법
 1. label id를 같게 설정하기
 2. label로 감싸기

```

<html>
  <body>
    <form action="">
      <p>
        <label for="text.txt"> text : </label>
        <input id="text.txt" type="text" name="id" value="default value">
      </p>
      <p><label for="id.txt"> id : </label>
        <input id="id.txt" type="text" name="id" value="input id"></p>
      <p><label for="pwd.txt"> password : </label>
        <input id="pwd.txt" type="text" name="pwd" value="input pwd"></p>
      <p><label for="textarea.txt"> textarea</label>
      <textarea id="textarea.txt" name="메모" id="" cols="50" rows="10">memo</textarea></p>
    </form>
  </body>
</html>

```

```

<label>
textarea : <textarea name="메모" cols="50" rows="10">memo</textarea>
</label>

```

```

<html>
  <body>
    <form action="">
      <p>
        <label for="text.txt"> text : </label>
        <input id="text.txt" type="text" name="id" value="default value">
      </p>
      <p>
        <label for="id.txt"> id : </label>
        <input id="id.txt" type="text" name="id" value="input id">
      </p>
      <p>
        <label for="pwd.txt"> password : </label>
        <input id="pwd.txt" type="text" name="pwd" value="input pwd"></p>
      <p>
        <label>
          textarea : <textarea name="메모" cols="50" rows="10">memo</textarea>
        </label>
      </p>
    </form>
  </body>
</html>

```



```

        </label>
    </p>

    <p>
        <label>
            <input type="checkbox" name="color" value="red"> 붉은색
        </label>
    </p>

    <p>
        <label for="color_blue">
            <input id="color_blue" type="checkbox" name="color" value="blue"> 파란색
        </label>
    </p>
</form>
</body>
</html>

```

text : default value

id : input id

password : input pwd

memo

method

- 서버쪽에 관련된 기술을 알아야 이해하기 좋음
- method
 - get방식 → URL을 통해서 데이터를 전송하는 방식(기본 설정값)

```

<html>
  <head>
    <meta charset="utf-8">
  </head>
  <body>

```

```

        <form action="http://localhost/method.php" method="get">
            <input type="text" name="id">
            <input type="password" name="pwd">
            <input type="submit">
        </form>
    </body>
</html>

```

- post방식 → URL이 아닌 방식으로 전송하는 방법(데이터를 몰래 전송하는 방법)

```

<html>
  <head>
    <meta charset="utf-8">
  </head>
  <body>
    <form action="http://localhost/method.php" method="post">
      <input type="text" name="id">
      <input type="password" name="pwd">
      <input type="submit">
    </form>
  </body>
</html>

```

UPLOAD

- input type → file
- method → post
- enctype → multipart/form-data

```

<html>
  <head>
    <meta charset="utf-8">
  </head>
  <body>
    <form action="http://localhost/upload.php" method="post" enctype="multipart/form-data">
      <input type="file" name="profile">
      <input type="submit">
    </form>
  </body>
</html>

```

정보로서 HTML ??

- 퇴출된 태그?

```

<html>
  <head>
    <meta charset="utf-8">
  </head>
  <body>
    <font size="8" color="red">Hello</font> world
  </body>
</html>

```

- 파일 용량이 커진다
- 웹페이지 가독성이 떨어진다
- 정보로써의 가치가 떨어진다.
- CSS라는 별도의 프로그래밍언어로 대체되어졌다.
 - 별도의 언어를 배우는게 고통스러움을 감수하고 HTML을 정보의 전달만을 위하게 만들어주었다. → 정보로써 너무나 중요한 언어

meta

- 데이터가 있으면 데이터를 설명하는 데이터
- HTML 태그 → 태그가 감싸고있는 콘텐츠를 설명
- meta charset="utf-8"
- meta name → description → content → 요약 자료
- meta name → keywords → content → 분류
- meta name → author → content → 저자
- meta → http-equiv → refresh → content → 웹페이지가 30초 간격으로 리프레쉬

```
<html>
  <head>
    <meta charset="utf-8">
    <meta name="description" content="생활코딩의 소개자료">
    <meta name="keywords" content="코딩, coding, 생활코딩, 프로그래밍, css, javascript">
    <meta name="author" content="egoing">
    <meta http-equiv="refresh" content="30">
  </head>
  <body>
    생활코딩은 일반인에게 프로그래밍을 알려주는 온라인/오프라인 강의입니다.
  </body>
</html>
```

semantic

- 정보로써의 가치가 중요해지면서 사람과 검색엔진에게 활용하기 좋은 용도의 데이터로 웹을 발전시키기 위한 트렌드 중 하나!
- 추상적이고 광범위

의미론적인 태그

- 의미를 부여 → 정보
- 네비게이션 → nav
- h2 → 본문 → article
- section

article	본문
aside	광고와 같이 페이지의 내용과는 관계가 적은 내용들
details	기본적으로 표시되지 화면에 표시되지 않는 정보들을 정의
figure	삽화나 다이어그램과 같은 부가적인 요소를 정의
footer	화면의 하단에 위치하는 사이트나 문서의 전체적인 정보를 정의
header	화면의 상단에 위치하는 사이트나 문서의 전체적인 정보를 정의
main	문서에서 가장 중심이 되는 콘텐츠를 정의
mark	참조나 하이라이트 표시를 필요로 하는 문자를 정의
nav	문서의 네비게이션 항목을 정의
section	문서의 구획들을 정의 (참고)
time	시간을 정의

의미론적 태그 - HTML 수업

문서의 정보를 보다 잘 표현하기 위해서는 의미에 맞는 태그를 잘 사용해야 합니다. 특히 HTML5에서는 웹페이지에서 통상 많이 사용하는 구조에 의미를 분명히 부여하기 위해서 의미론적 태그(semantic element)를 새롭게 정의해서 제공하고 있습니다. 그 목록은 아래와 같습니다.

● <https://opentutorials.org/module/1892/10954>



검색엔진 최적화(SEO)Search Engine Optimizer

- 검색엔진들이 콘텐츠를 잘 해석할 수 있게
- 검색엔진들이 콘텐츠를 상위에 잘 나타낼 수 있게
- HTML → 의미론에 적합하게, 의미론적으로 타당하게 잘 설명
 - 검색엔진의 기본
 - 검색엔진에게 잘 보일려고하지말고 검색엔진을 의미론적으로 잘 쓰면 가장좋은 최적화 엔진

2023.02.11

title태그를 위한 권장 사항

- 페이지의 콘텐츠를 정확하게 설명하는 제목
 - 페이지 내용의 주제를 효과적으로 설명하는 제목을 선택합니다.
 - 피해야 할 사항:
 - 페이지 내용과 관련이 없는 제목 선택
 - “무제목서” 또는 “새 페이지 1” 과 같은 기본 제목이나 모호한 제목 사용

- 페이지마다 고유한 title 태그 작성
 - 페이지마다 고유한 title 태그가 있는 것이 효과적입니다. 사이트 내의 서로 다른 페이지를 구별하는데 도움이 됩니다.
 - 피해야 할 사항:
 - 사이트의 모든 페이지 또는 대다수의 페이지에 동일한 title 태그 사용
- 간결하면서 내용을 포함한 제목 작성
 - title 태그는 간단명료하면서 페이지 내용에 대한 정보를 제공할 수 있도록 작성합니다. 제목이 너무 길면 검색결과에 일부분만 표시됩니다.
 - 피해야 할 사항:
 - 사용자에게 도움이 되지 않는 지나치게 긴 제목 사용
 - 지나치게 불필요한 키워드를 많이 사용하기

리다이렉션

```
<html>
  <head>
    <meta charset="utf-8">
  </head>
  <body>
    
  </body>
</html>
```

검색엔진 최적화

- 스크린 리더와 같은 다른 기기를 사용하는 경우 alt 속성의 내용이 그림에 대한 정보 제공
- alt로 검색 결과 제공 가능

```
<html>
  <head>
    <meta charset="utf-8">
  </head>
  <body>
    
  </body>
</html>
```

→ 이미지만 보관하는 폴더의 경로 사용

→ 의미를 부여하는 이름 사용

```
<html>
  <head>
    <meta charset="utf-8">
```

```

</head>
<body>
  <a href="test1.html"></a>
</body>
</html>

```

- 이미지에 링크 가능

2023.02.21

HTML - 검색엔진최적화 6 : robotstxt & sitemap

- `URL/robots.txt` → 파일 다운로드가 된다
 - `naver.com/robots.txt`

```

User-agent: * -> 검색엔진처럼 여러분의 사이트에 접속하는 일반적인 사용자의 웹 브라우저를 제외한 로봇들
-> 네트워크를 통해 접속되는 소프트웨어들이다.
-> 모든 로봇들의 접속을 Disallow 한다.
Disallow: /
/ -> 모든 웹페이지에 대해서 어떠한 로봇의 접속도 허용하지 않는다. (https://www.naver.com/robots.txt)
-> 네이버에서 만들어진 컨텐츠들은 다른 검색 엔진들에서는 노출되어지지 않는다.
Allow : /$

```

- `google.com/robots.txt`
source link : <https://www.google.com/robots.txt>

```

User-agent: *
Disallow: /search => 검색필요x, 보호, 타사가 가져가지 말아야 할 정보
-> 검색 부분
Allow: /search/about
-> 구글과 관련된 도움말 같은 부분

```

- `robots.txt` 는 보안에 사용해서는 안된다
 - 보안은 확실한 보안을 사용하자

@ `/sitemap` → 사이트내의 지도(이동하기 쉽게 만들어줌)

- 링크는 사람들이 보기 쉽게 되어있고 클릭해서 이동하기 쉽게해주지만 기계가 이해하기 쉬운 형태는 아니다 → `sitemap` 은 기계 즉 검색엔진들이 이해하기 쉽게 되어있다.
- `sitemap.xml`

```

<?xml version="1.0"encoding="UTF-8">
<urlset xmlns="http://www.sitemaps.org/schemas/sitemap/0.9">
  <url>
    <loc>http://opentutorials.org/1.html</loc>
  </url>
  <url>
    <loc>http://opentutorials.org/2.html</loc>
  </url>
  <url>
    <loc>http://opentutorials.org/3.html</loc>
  </url>
</urlset>

```

- URL들을 URLSET태그 안에 담아서 쪽 나열을 해준다.
- 기계가 해석하기 수월하다.
- `robots.xml`

```
User-agent: *
Sitemap: /sitemap.xml
```

⇒ 검색엔진이 사이트에 들어와서 robots.txt파일을 읽고 sitemap.xml의 이름을 보고 들어가서 파일을 다운로드 받는다. → xml을 통해서 로봇들이 어떠한 홈페이지들이 있는지, 어떠한 파일들이 있는지 본다. ⇒ 검색엔진의 활용

→ 검색엔진 로봇들의 동작 방식을 제어 가능하다!!

2023.02.22

검색엔진 최적화

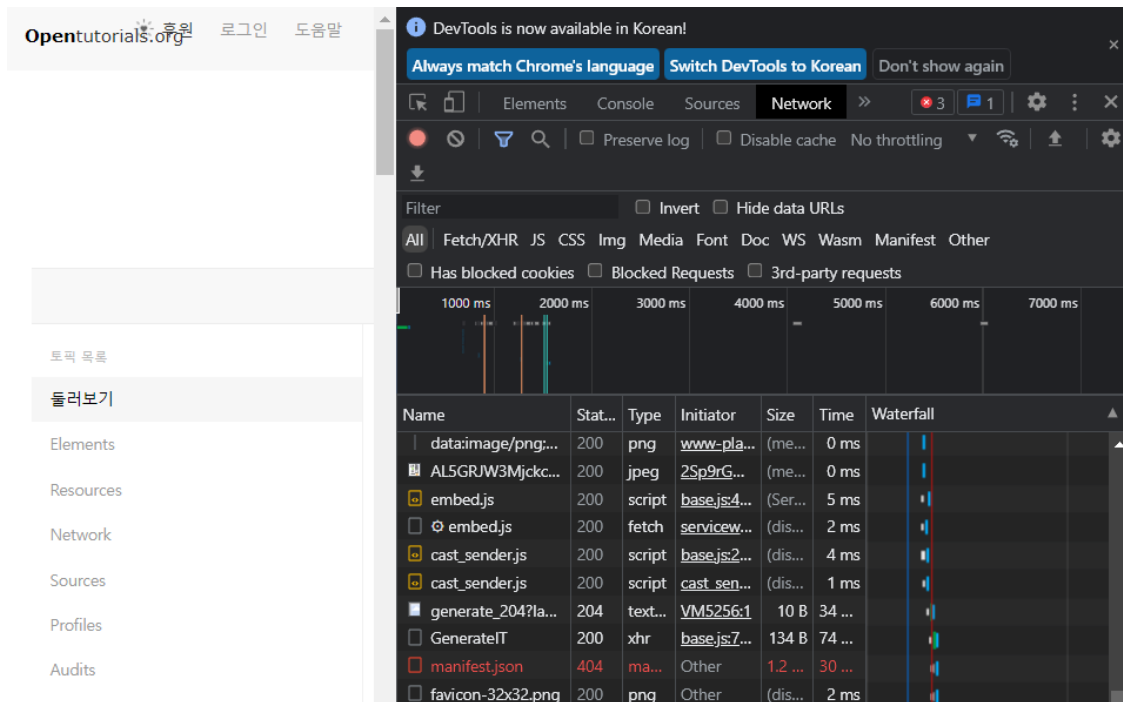
- 페이지의 랭크(순위)
 - 똑같은 단어를 두개의 사이트가 동시에 가지고 있을 때 어떤 사이트를 먼저 검색결과에 노출해주는 것

웹개발자 도구

- Chrome 웹 브라우저에 탑재되어 있는 개발자 도구
 - 웹페이지에서 검사버튼을 눌러 소스코드 확인 → 각 기기마다 어떻게 웹페이지가 보이는지 확인 가능

```
<html>
<body>
  
</html>
```

- img를 통해서 이미지를 삽입 : img.jpg파일을 받아서 포함된 html코드를 해석해서 웹페이지를 만들어준다.
- img의 태그를 만날 때 : src안에 들어있는 img.jpg파일을 웹페이지에 접속해서 파일을 다운로드 받고, img.jpg파일 안에 이미지를 표시해준다.
 - 무엇 때문에 느려지는지 이러한 것들을 파악하기 어려움 → Network를 통해서 확인 가능



모바일

```
<!DOCTYPE html>
<!-- Document type declaration -->
<html>
<head>
<title>HTML - 실습하기</title>
<meta charset="utf-8">
<meta name="viewport" content="width=device-width, initial-scale=1.0">
</head>

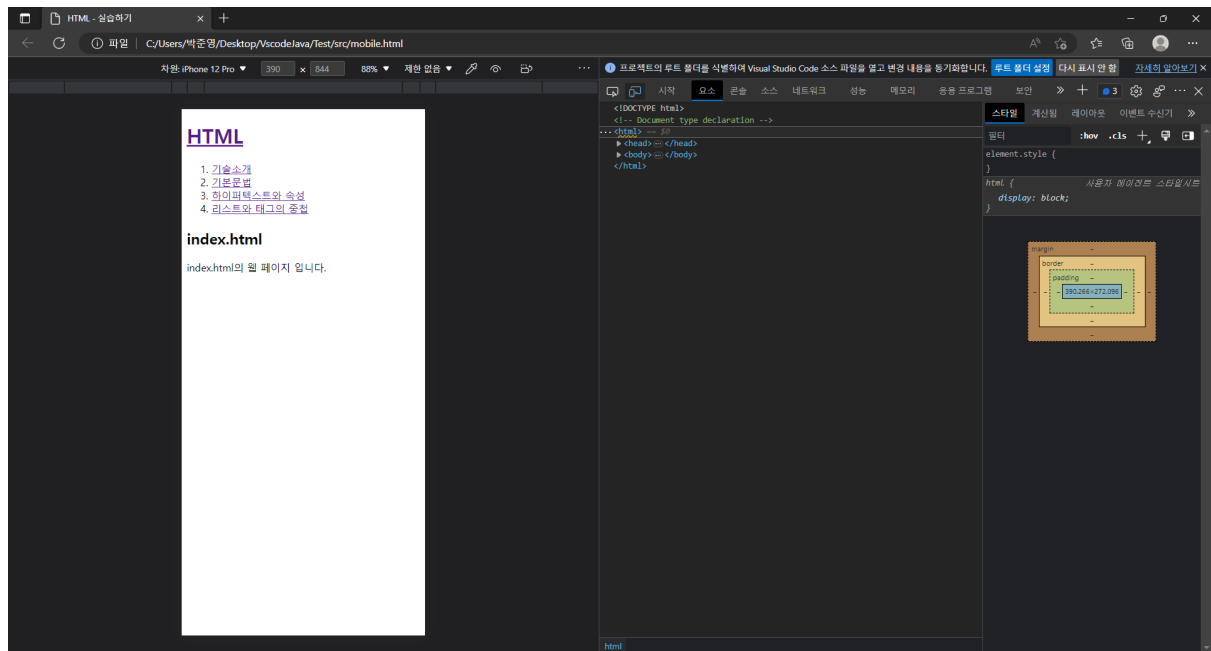
<body>
<h1><a href="index.html">HTML</a></h1>

<ol>
<li><a href="test1.html">기술소개</a></li>
<li><a href="test2.html">기본문법</a></li>
<li><a href="test3.html">하이퍼텍스트와 속성</a></li>
<li><a href="test4.html">리스트와 태그의 중첩</a></li>
</ol>

<h2>index.html</h2>
index.html의 웹 페이지 입니다.

</body>
</html>
```

- `<meta name="viewport" content="width=device-width, initial-scale=1.0">`
 - 모바일 기기의 화면에 맞춰서 자동으로 조정해주는 코드
 - PC에서는 원래의 HTML에 맞춰서 나옴



새로운 제출 양식들

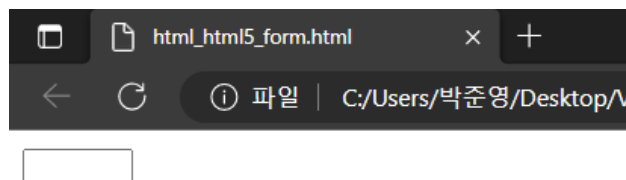
- HTML5에 들어오게 되면서 사용자들의 편의성이 대폭 향상

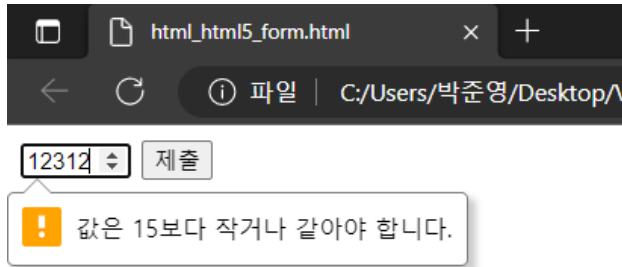
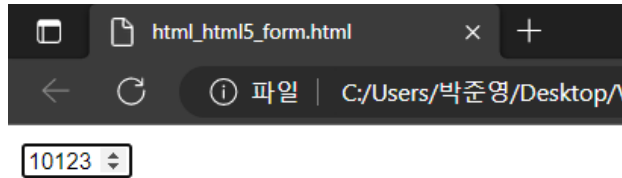
- input 타입이 다양해짐

1. `input type="number"` : 숫자만 입력 가능

- a. `min="10" max="15"` 등으로 디폴트값 설정 가능

```
<!DOCTYPE html>
<html>
  <head>
    <meta charset="utf-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
  </head>
  <body>
    <form action="form.php">
      <input type="number" min="10" max="15">
      <input type="submit">
    </form>
  </body>
</html>
```





숫자 외에는 입력 불가!!!

2. `input type="color"`
3. `input type="date"`
4. `input type="datetime"`
5. `input type="email"`
6. `input type="month"`
7. `input type="number"`
8. `input type="range"`
9. `input type="search"`
10. `input type="tel"`
11. `input type="time"`
12. `input type="url"`
13. `input type="week"`

```
<input type="number" min="10" max="15"><br>
  <input type="date" name="datev"><br>
  <input type="month" name="monthv"><br>
  <input type="week" name="weekv"><br>
  <input type="time" name="timev" id=""><br>
  <input type="email" name="emailv" id=""><br>
  <input type="search" name="searchv" id=""><br>
  <input type="tel" name="telv" id=""><br>
  <input type="url" name="urlv" id=""><br>
  <input type="range" name="rangev" id="" min="0" max="10"><br>
  <input type="submit">
```

HTML5 입력폼의 새로운 속성들

- `autocomplete="on"` : 아래의 텍스트 및 패스워드의 양식들이 활성화되어진다.
- `autocomplete="off"` : 아래의 텍스트 및 패스워드의 양식들이 비활성화되어진다.
- `placeholder="text"` : android studio의 hint
- `autofocus` : 입력란의 포커스를 맞출 수 있다.

```
<!DOCTYPE html>
<html>
  <head>
    <meta charset="utf-8">
  </head>
  <body>
    <form action="login.php" autocomplete="on">
      <input type="text" name="id" id="" placeholder="id를 입력해주세요." autofocus><br>
      <input type="password" name="password" id="" autocomplete="off" placeholder="비밀번호를 입력해주세요.">
      <input type="submit" name="submit" id="">
    </form>
  </body>
</html>
```

HTML5 입력 값 체크

- 일종의 유효성 체크라고 할 수 있다.

- `required` : 반드시 입력을 해야한다. → 유효성 검사와 관련

- `pattern="[a-zA-Z]..."` : 요청사항을 정할 수 있다.
 - 어떠한 문자든 넣을 수 있다.
 - .+ → 어떤 문자이던 하나 더 와야한다.
 - [0-9] : 숫자가 와야한다.
 - ⇒ 정규 표현식

```
<!DOCTYPE html>
<html>
  <head>
    <meta charset="utf-8">
  </head>
  <body>
    <form action="register.php">
      <input type="text" name="id" id="" placeholder="아이디를 입력해주세요." required pattern="[a-zA-Z].+[0-9]"><br>
      <input type="email" name="email" id="" placeholder="이메일 입력">
      <input type="submit" name="" id="">
    </form>
  </body>
</html>
```