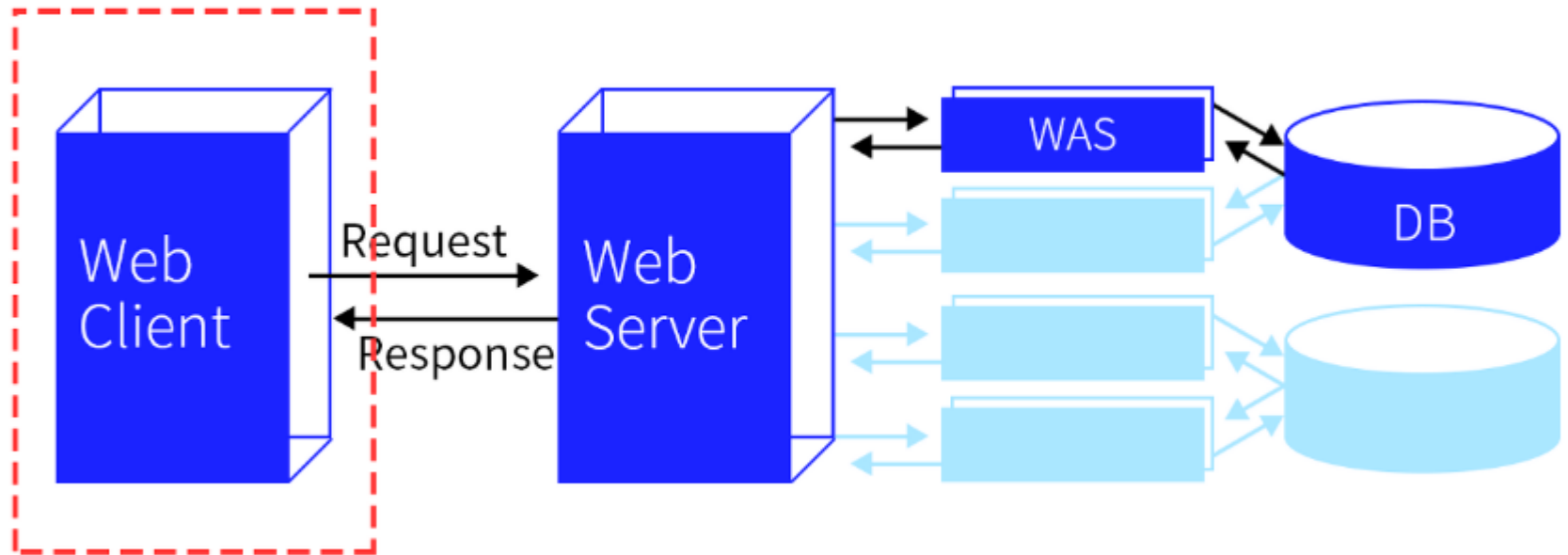


웹 브라우저 렌더링

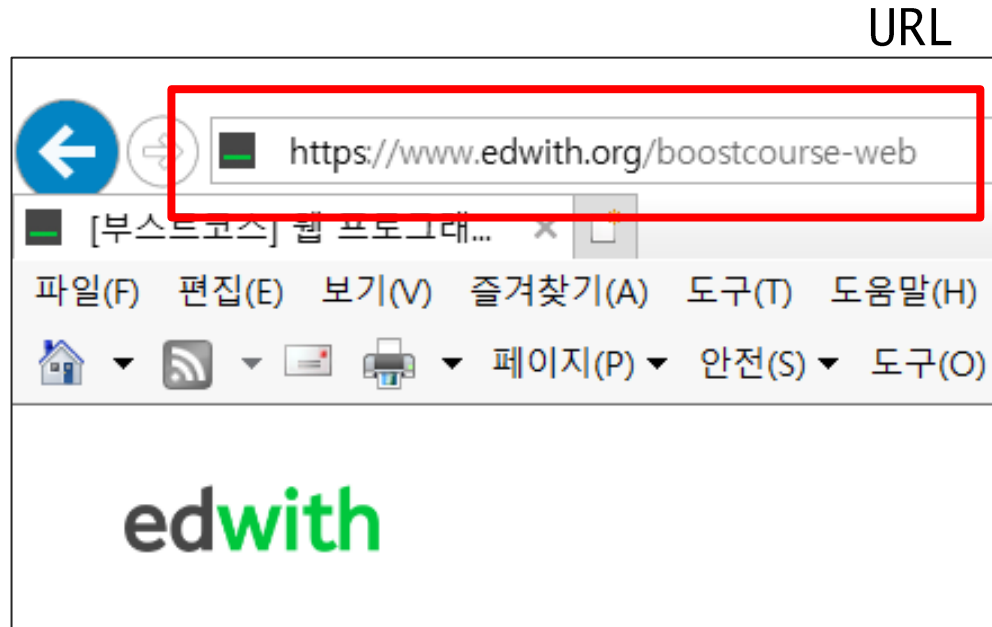
I. 웹 브라우저 렌더링

웹 클라이언트 (Web Client)



I. 웹 브라우저 렌더링

클라이언트가 URL Request 요청 → 브라우저는 URL을 해독해 메시지 생성



http://

액세스 대상 프로토콜

www.edwith.org

웹 서버 호스트명

:80

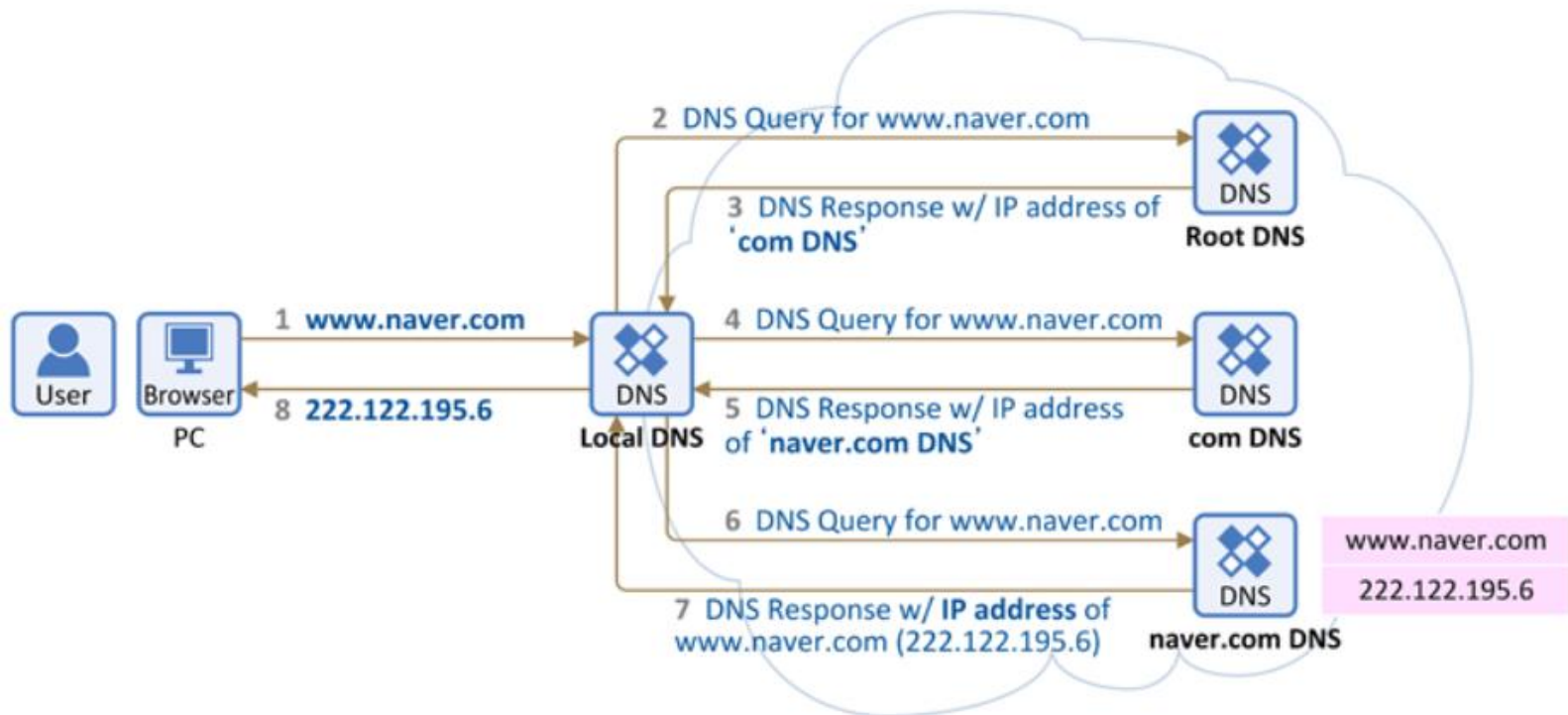
대상포트

/index.html

액세스 대상 경로

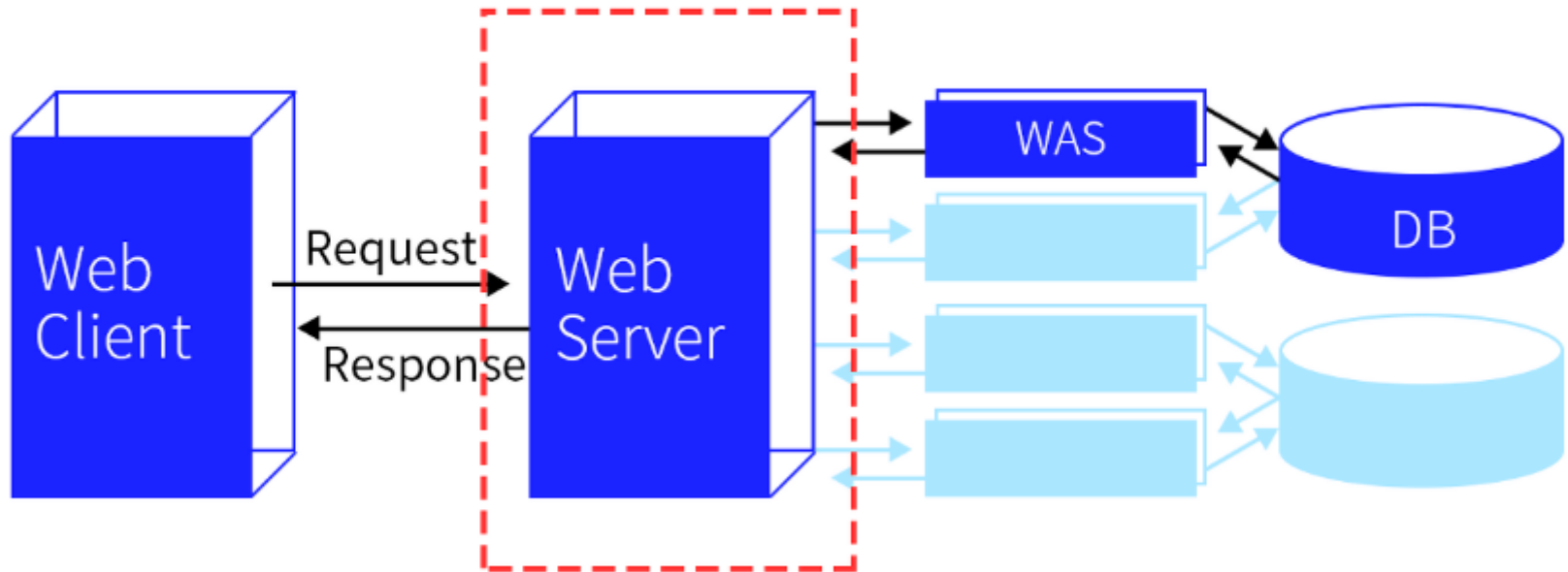
I. 웹 브라우저 렌더링

메시지를 보내기 위해 URL 주소 정보 해석 (DNS)



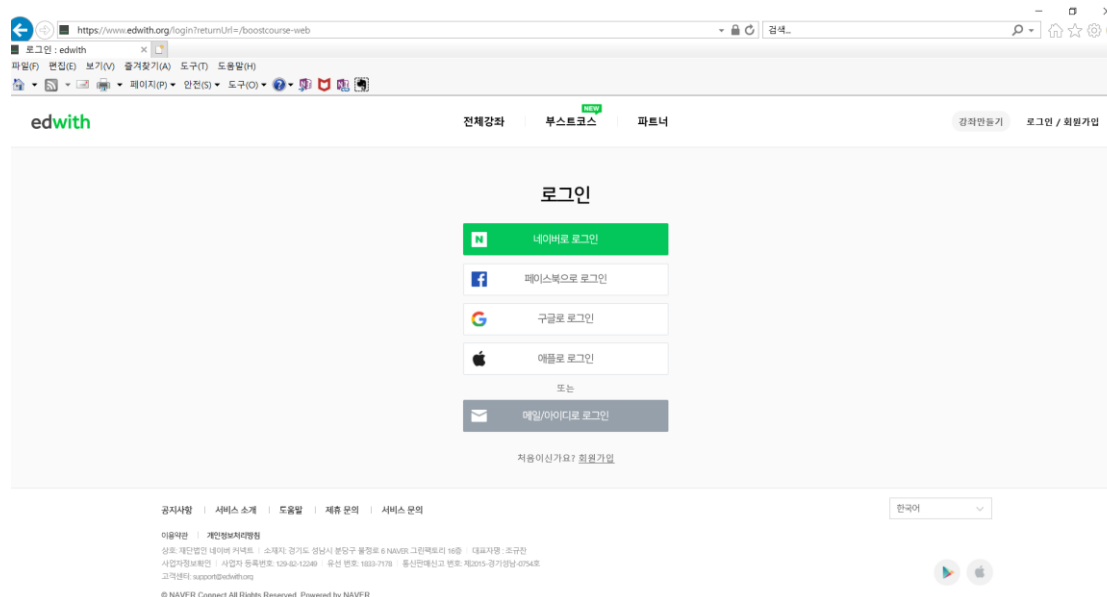
I. 웹 브라우저 렌더링

요청을 확인하고 HTML, CSS, JS, Image 등을 가져와 클라이언트에게 보낸다.



I. 웹 브라우저 렌더링

브라우저에서는 이 데이터들을 어떻게 표시해줄까?

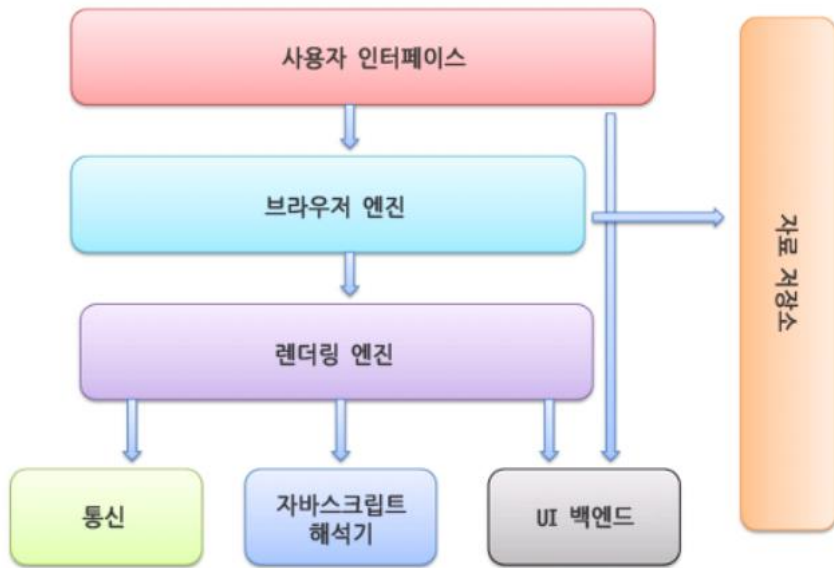


브라우저의 주요 기능 !

→ 사용자가 선택한 자원을 서버에 요청하고 브라우저에 표시
이 자원은 웹 리소스(html, pdf, 이미지 등)을 말함.

I. 웹 브라우저 렌더링

브라우저 기본 구조



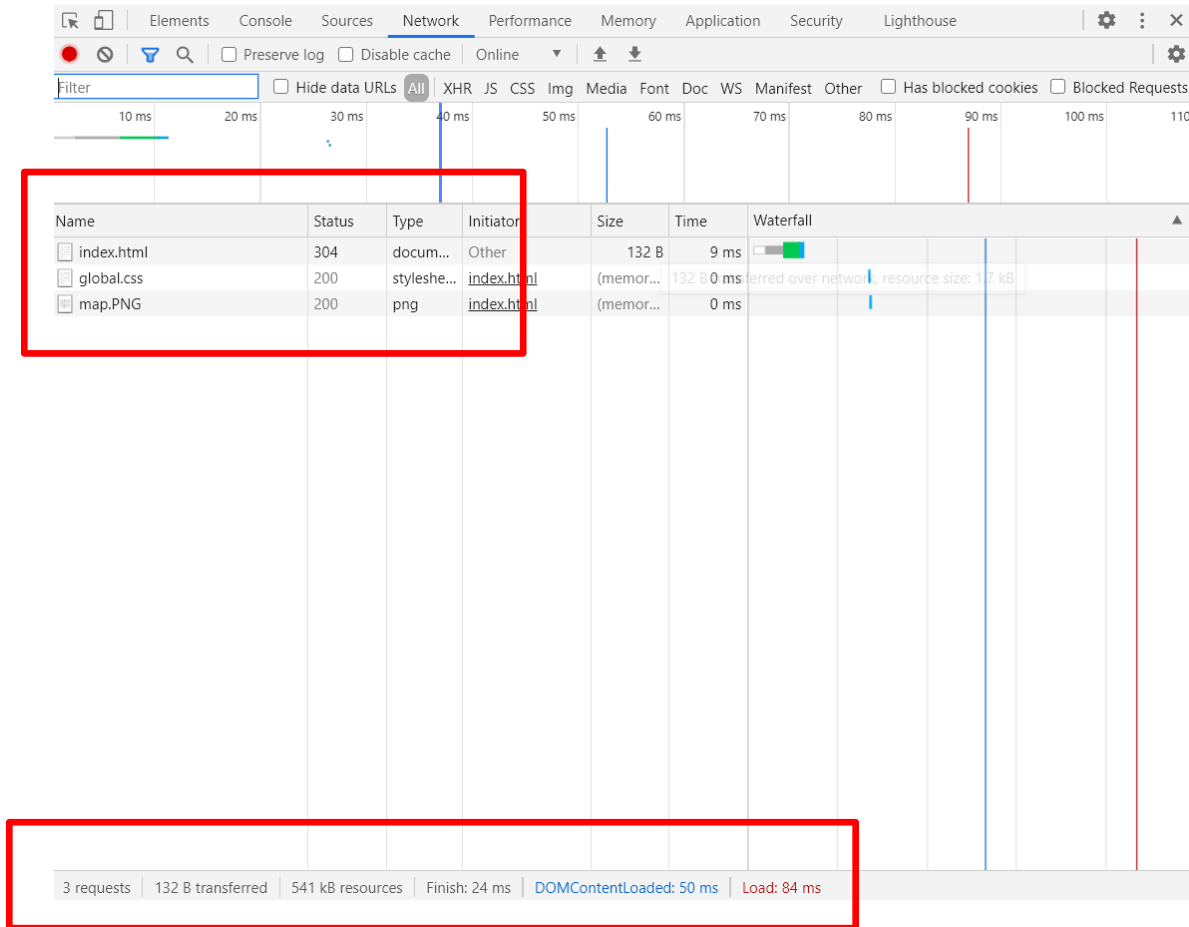
1. 사용자 인터페이스 – 주소 표시줄, 이전/다음 버튼, 북마크 메뉴 등 모든 브라우저의 공통
2. 브라우저 엔진 – 렌더링 엔진에 작업을 요청하고 다루는 인터페이스 부분
3. 렌더링 엔진 – 요청한 콘텐츠를 표시. 요청된 페이지가 HTML 이라면, HTML과 CSS를 파싱하여 화면에 표시하는 역할
4. 통신 – HTTP(HTTP Request) 요청과 같은 네트워크 호출에 사용된다.
5. UI 백엔드 – 콤보 박스와 창 같은 기본적인 장치를 그린다.
6. 자바스크립트 해석기 – 자바스크립트 코드를 해석하고 실행한다.
7. 자료 저장소(퍼시스턴스 계층) – 이 부분은 자료를 저장하는 계층이다. 쿠키를 저장하는 것과 같이 모든 종류의 자원을 하드디스크에 저장할 수 있다.

I. 웹 브라우저 렌더링



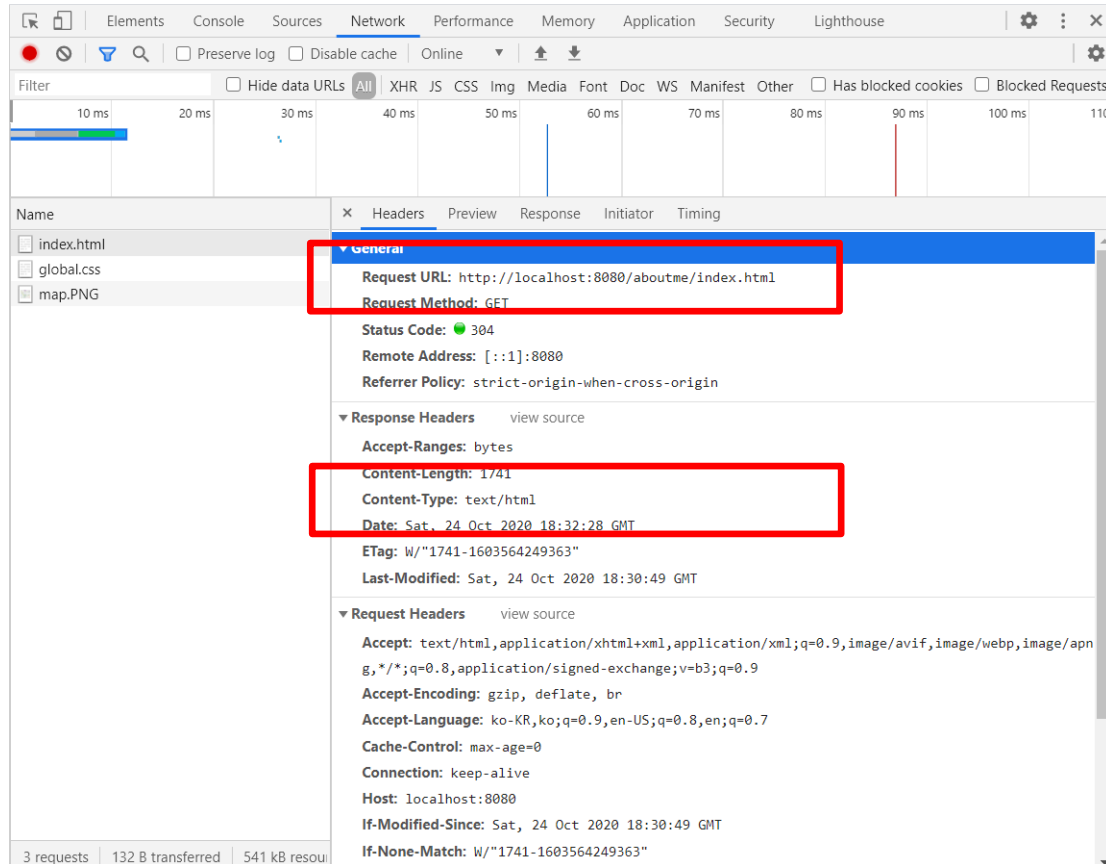
I. 웹 브라우저 렌더링

다운로드 받은 리소스 목록, 렌더링 시간 등...



I. 웹 브라우저 렌더링

웹 브라우저는 해당 URL로 웹 페이지를 요청, 웹 서버는 URL 응답을 반환



I. 웹 브라우저 렌더링

웹 브라우저는 필요한 리소스들을 다운로드하고 렌더링

Network tab showing the response for index.html. The response is HTML code. A red box highlights the body content, which includes a header with a navigation menu and a section with a heading and paragraph. A blue arrow points from the DOM tree to the highlighted body content.

DOM Tree structure:

```
graph TD
    html[html] --> head[head]
    html --> body[body]
    head --> title[title]
    body --> h1[h1]
    body --> p[p]
    p --> a1[a]
    p --> a2[a]
    p --> a3[a]
```

HTML Response (index.html):

```
1 <!--
2 project : web-programming introduce
3 -->
4
5 <!DOCTYPE html>
6 <html>
7   <head>
8     <meta charset="utf-8">
9     <title>introduce</title>
10    <link rel="stylesheet" type="text/css" href="./CSS/{
11  </head>
12
13  <body>
14    <header>
15      <nav class="topMenu"><ul>
16        <!-- flex 로 가로 메뉴바 변경 -->
17        <li><a class="menuLink" href="/aboutme/today">몇 시
18        <li><a class="menuLink" href="photo.html">내 사진</
19        <li><a class="menuLink" href="aboutme.html">자기소
20        <li><a class="menuLink" href="index.html">홈</a></
21      </ul></nav>
22    </header>
23    <section class="subject">
24      <h2> 웹 프론트엔드 개발자 ~! 도아 </h2>
25      <p>
26        함께 개발하고 같이 성장하고 싶어요.<br />
27        경기도 김포시 어딘가 살고 있어요!
28      </p>
29      <div class="btn">
```

HTML컨텐츠, 브라우저는 HTML 태그를 파싱해 DOM 트리 구성

I. 웹 브라우저 렌더링

렌더링 과정 7가지

1. 서버에서 응답으로 받은 HTML 데이터를 파싱
2. HTML을 파싱한 결과로 DOM Tree를 만든다.
3. 파싱을 하면서 CSS 링크 파일을 만나면 CSS 파일을 요청하여 받아온다.
4. CSS 파일을 읽어 CSSOM을 만든다.
5. DOM 트리와 CSSOM이 만들어지면 이 둘을 이용해 Render Tree를 만든다.
6. Render Tree에 있는 각 노드들이 화면의 어디에 위치할지 계산하는 레이아웃을 거친다.
7. 화면에 실제 픽셀을 Paint 한다.

