

웹의 구조와 요소에 대한 이론적 고찰 및 웹 페이지 구현



이름	김은수
학과	인공지능학과
학번	2024108139
과목명	오픈소스개발방법론
담당교수	변태보 교수님
제출일자	25.05.23 (금)

< 목 차 >

I. 서론	3
-------------	---

II. 본론	3
1.Web이란?	3
1)Web의 구성요소	
(1) 웹 클라이언트(Web client)	
(2) 웹 브라우저(Web Browser)	
(3) HTTP(Hyper Text Transfer Protocol)	
(4) 웹 서버(Web Server)	
(5) 웹 어플리케이션(Web Application)	
2)Web의 구조	
3)Web의 특징	
4)동작 원리	
2.Web개발의 기술과 원리	5
1) HTML	
2) CSS	
3) JavaScript	
4) PHP	
3.반응형 웹과 핵심 원리 및 기술	8
4.웹 페이지 구현	9
III. 결론	9
1.느낀점	
참고문헌	10

I. 서론

웹이란 1989년 팀 버너스 리가 제안한 하이퍼 텍스트 기반의 정보 공유 시스템으로 시작되었다. 웹을 개발한 이후 웹은 인터넷의 확산과 모바일 기기의 폭발적 증가로 전세계에 확산되었고 이제는 웹이 없는 삶을 상상할 수 없는 상황에 이르렀다. 수많은 사람들이 이용하고 보다 많은 정보가 오가는 웹은 단순한 정보 검색을 넘어 일상생활의 소통, 업무, 쇼핑, 등 사회 전반에 걸쳐 핵심적인 플랫폼으로 자리 잡고 있다. 이러한 웹 환경은 클라이언트, 서버 등 다양한 상호작용에 의해 구성된다. 본 보고서는 웹의 기본적인 개념과 구조를 고찰하고 핵심 기술들의 역할과 원리를 정리함으로써 웹이 어떻게 작동되는지 그 흐름을 이해하는 데 목적이 있다. 또한 이해한 개념들을 토대로 웹 페이지를 구현해보고자 한다.

II. 본론

1.WEB이란?

컴퓨터 서버에 저장되는 문서 모음 시스템으로 인터넷을 기반으로 동작하는 하나의 서비스이다. 인터넷에 연결된 컴퓨터들을 통해 사람들이 정보를 공유할 수 있는 인터넷 표준 정보 공간이다. 웹 사이트의 서버는 인터넷에 연결되어 있어 컴퓨터나 휴대전화 브라우저를 통해 해당 웹 사이

트를 이용할 수 있으며 이 브라우저를 클라이언트라고 한다. 소규모 비즈니스 웹 사이트 및 블로그 등 모두 브라우저에서 액세스 가능하며 웹 개발을 통해 만들 수 있다.

1)Web의 구성요소

웹은 다양한 구성요소로 이루어져 상호작용을 통해 사용자가 원하는 정보를 쉽게 접근할 수 있다. 웹 환경은 기본적으로 클라이언트, 서버 구조이며 사용자의 요청과 서버의 응답을 기반으로 작동한다.

(1) 웹 클라이언트(Web client)

사용자가 웹 서비스를 이용하기 위해 사용한다. 필요한 데이터를 웹 서버에 요청하는 주체로서 주로 브라우저를 통해 서비스를 이용한다. 사용자는 브라우저에 URL을 입력하여 서버로 데이터를 요청하고 HTTP를 통해 전달된다.

(2) 웹 브라우저(Web Browser)

클라이언트의 요청을 받아서 Request Message를 작성하여 웹 서버에 전달하고 웹 서버로부터 응답 받은 Response Message를 해석하여 사용자에게 보여주는 소프트웨어로 대표적인 웹 브라우저는 Chrome, Edge, Firefox, Safari 등이 있으며 웹 표준 기반 다양한 콘텐츠를 표현할 수 있다.

(3) HTTP(Hyper Text Transfer Protocol)

클라이언트와 서버 사이에 이루어지는 요청-응답 프로토콜로 Hyper Text¹링크를 통해서 문서를 이동하거나 페이지 정보를 전송하는 프로토콜이다. 텍스트 뿐만 아닌 이미지, 영상, 오디오 등 다양한 형태의 데이터를 요청, 전송 시에 사용된다. 비연결성과 무상태성을 특징으로 하며 웹 애플리케이션의 유연한 확장과 설계를 가능하도록 한다. 최근에는 보안성을 강화한 HTTPS 프로토콜이 사용된다.

(4) 웹 서버(Web Server)

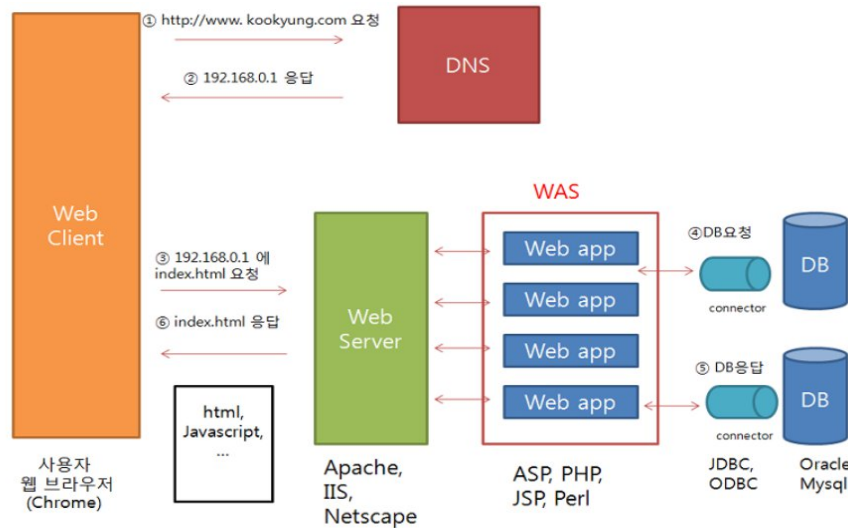
클라이언트 요청에 따라 이에 해당하는 html문서나 리소스 클라이언트에게 반환하는 주체이다. Apache, Nginx, Microsoft IIS 등이 대표적 웹 서버 소프트웨어로 사용되며 일반적으로 80(HTTP), 443(HTTPS)포트를 사용하여 클라이언트와 통신한다.

(5) 웹 어플리케이션(Web Application)

Browser를 통해 접근할 수 있는 응용프로그램으로 HTTP에서 동작되는 프로그램이다. Html, CSS, JavaScript를 활용하여 동적인 기능을 구현하며 서버 측 언어(PHP, Python, Node.js 등)와 데이터 베이스 시스템이 함께 동작하여 사용자의 입력을 처리하고 맞춤형 정보를 제공한다.

2)Web의 구조

¹ 사용자의 선택에 따라 관련된 문서로 옮겨갈 수 있도록 조직화된 정보, 링크(link) 또는 하이퍼링크(hyperlink)라고 부른다.



웹은 클라이언트-서버 구조를 기반으로 하며 유저(user)가 1번과 같이 웹 브라우저를 통해 `www.naver.com`에 접속을 요청한다고 하자. DNS(Domain Name System) 서버로 도메인 이름이 보내지면 DNS는 이를 IP주소(예: `192.0.2.44`)로 변환한다. 다음 웹 브라우저는 Web Server에 IP주소로 접속 요청을 보내고 그 요청은 WAS를 거쳐 웹 서버를 통해 웹 브라우저에게 전달된다. 이로써 우리는 웹 브라우저상에 네이버 화면을 볼 수 있다.

3)Web의 특징

전 세계 컴퓨터를 연결하며 Hyper Text Transfer Protocol을 사용하여 데이터를 송수신하며 Hyper Text Markup Language로 작성된 문서를 연결하는 구조를 가진다. 사용자는 텍스트, 그래픽, 오디오, 비디오, 프로그램 파일 등 다양한 멀티미디어 서비스에 접근할 수 있다. 특히 플랫폼 독립적으로 설계되어 운영체제에 관계없이 다양한 환경에서 동일한 웹 서비스를 이용할 수 있다.

4)동작 원리

사용자가 웹 브라우저를 통해 찾고 싶은 웹 페이지의 URL주소를 입력하고 입력 받은 URL 주소 중에서 도메인 네임 부분을 DNS(Domain Name System) 서버에서 검색한다. 해당 도메인 네임에 해당하는 IP주소를 찾아 사용자가 입력한 URL 정보와 함께 전달한다. 웹 페이지 URL 정보와 전달받은 IP주소는 HTTP 프로토콜을 사용하여 HTTP 요청 메시지를 생성한다. 이렇게 생성된 HTTP 요청 메시지는 TCP 프로토콜을 사용하여 인터넷을 거쳐 해당 IP 주소의 컴퓨터로 전송한다. 이 과정을 거쳐 도착한 HTTP 요청 메시지는 HTTP 프로토콜을 사용하여 웹 페이지 URL 정보로 변환된다. 웹 서버는 도착한 웹 페이지 URL 정보에 해당하는 데이터를 검색하고 검색된 웹 페이지 데이터는 다시 HTTP 프로토콜을 사용하여 HTTP 응답을 생성한다. 이렇게 생성된 HTTP 응답은 TCP 프로토콜을 사용하여 인터넷을 거쳐 원래 컴퓨터로 전송된다. 도착한 HTTP 응답은 HTTP 프로토콜을 사용하여 웹 페이지 데이터로 변환되고 이 웹 페이지 데이터는 웹 브라우저에 의해 출력되어 사용자가 볼 수 있게 된다. 이러한 과정을 거쳐 웹이 동작한다.

2.WEB개발의 기술과 원리

1)HTML

web브라우저 창에 웹 문서의 내용을 출력하기 위한 표준 마크업 언어로 HTML에서 정해진 표기법을 사용하여 문서를 작성해야 한다. 이 정해진 구조를 기반으로 웹 페이지가 브라우저에 표시된다. HTML문서는 `<html>`태그로 시작되며 실제문서 정보와 내용이 시작되고 끝나는 것을 표시한다. `lang`속성을 사용하여 문서의 주 언어를 지정한다. 브라우저에게 정보를 주는 `<head>`태그는 문서관련 정보를 입력하는 태그이다. 웹 브라우저 화면에 직접적으로 노출되지

않지만 중요한 문서 정보를 포함한다. 이 영역에는 문자 세트 등 문서의 정보를 지정하는 <meta>태그, 문서 탭의 제목을 나타내는 <title>태그 등이 포함한다. 실제로 웹 브라우저에 콘텐츠를 출력하는 <body>태그는 실제 브라우저에 표시될 내용을 입력하며 대부분의 HTML요소들이 이 영역 안에 사용된다.

HTML5부터는 웹 문서의 의미를 명확하게 표현하고 구조를 구분하기 위한 시맨틱(semantic) 태그가 도입되었다. 시맨틱 태그는 문서의 구조를 보다 명확하게 표현할 수 있어 사용자뿐만이 아닌 화면 낭독기와 같은 보조 기기에서도 사이트의 구조를 이해할 수 있다. 또한 반응형 웹 구현 시에 문서 구조가 정확히 나뉘어져 pc나 모바일 등 다양한 화면에서 웹 문서를 표현하기 쉬우며 웹 사이트를 검색할 때 필요한 내용을 정확히 찾을 수 있기 때문에 사용된다. 먼저 <header>태그는 헤더 영역을 나타내며 사이트 전체의 헤더, 특정 영역의 헤더의 영역으로 주로 검색 창, 로고, 사이트 메뉴 삽입에 사용된다. <nav>태그는 네비게이션 영역을 나타내며 웹 문서 위치에 영향을 받지 않고 문서 안을 id로 구분하여 여러 개의 태그를 만들 수 있다. <main>태그는 웹 문서의 핵심 콘텐츠가 되는 내용을 담는 영역이다. 웹 문서마다 다르게 보여주는 내용으로 구성되며 웹 문서에서 한 번만 사용된다. <section>태그는 콘텐츠 영역을 나타낸다. 여러 개의 콘텐츠를 묶는 용도로 사용되며 CSS 적용을 위한 즉, <div>태그의 용도로 사용하면 안 된다. <aside>태그는 사이드 바 영역을 나타낸다. 본문 내용 외에 왼쪽, 오른쪽, 혹은 아래쪽에 사이드 바로 표시된다. 필수요소가 아니므로 필요한 경우에만 사용된다. <footer>태그는 푸터 영역을 나타낸다. 사이트 제작 정보나 저작권 정보, 연락처 등 다른 시맨틱 태그를 사용하여 다양한 정보를 포함한다. <div>태그는 여러 소스를 묶는 용도이다. 영역을 구별하거나 스타일 적용을 위해 사용된다.

2)CSS

CSS(Cascading Style Sheet)는 스타일 시트 언어로 웹 문서의 디자인과 레이아웃을 정의한다. 'Cascading'이라는 단어에서 알 수 있듯이 우선순위 규칙에 따라 스타일이 적용된다. html요소의 시각적인 요소를 적절하게 배치하는 방법으로 다양한 디바이스에 맞는 반응형 웹 디자인 구현에 용이하다. 또한 html문서와 스타일 시트를 분리하여 각각의 수정, 관리가 편리하다. CSS는 선택자를 통해 html 요소를 선택하고 중괄호{} 안에 속성과 값을 입력하여 스타일을 정의한다. 코드의 가독성과 유지보수를 위해 한 줄 또는 여러 줄로 작성하며 실제 협업 시에는 CSS경량화를 통해 주석, 공백 등의 부분을 줄이는 작업을 통해 효율성을 높이기도 한다.

CSS에서 스타일을 적용할 html요소를 선택하는 것을 선택자라 하며 주요 선택자는 다음과 같다. 먼저 전체 선택자(*)는 웹 문서 내 모든 요소에 스타일을 적용한다. 브라우저의 기본 스타일을 초기화할 때 사용되는 선택자이다. 타입 선택자(태그 선택자)는 html 문서 내 스타일을 적용하고자 하는 태그에 스타일을 적용하는 방식이다. 클래스 선택자는 html문서를 작성할 때 특정 요소만 스타일 적용을 시키기 위해 태그 내 class 속성으로 class명을 지정하여 스타일을 적용하는 방식이다. html문서에 작성된 클래스를 .class명으로 스타일을 적용한다. 하나의 클래스명을 여러 요소에 사용할 수 있어 반복적인 스타일 적용이 용이하다. 아이디 선택자는 한 문서 내에서 같은 id를 중복 사용할 수 없으며 특정요소만 스타일을 적용하기 위해 html 문서 내 id속성으로 아이디 명을 지정하고 해당 아이디명으로 스타일을 적용하는 방식이다. Html 문서에 작성된 아이디명을 #아이디명으로 선택한다. 그룹 선택자는 여러 선택자에 대해 같은 스타일을 적용할 때 사용된다. 침표로 각 선택자들을 구분하여 나열하는 방식으로 사용한다.

CSS를 웹 문서에 적용하는 방법으로는 인라인 스타일 작성법, 내부 스타일 시트, 외부 스타일 시트 세 가지 방식이 있다. 먼저 인라인 스타일 작성법은 스타일 시트를 따로 작성하지 않고 간단한 스타일을 적용하고자 할 때 사용한다. Html 요소 내부에 style 속성을 이용해 직접 적용할 수 있어 간단하지만 유지보수가 어렵다. 내부 스타일 시트 작성법은 웹 문서 내부 head태그 내에 <style> 시트를 작성하는 방법이다. 해당 웹 문서 전체에 대하여 스타일 시트를 적용하는 방식이다. 외부 스타일 시트 작성법은 웹 문서와 별도로 .CSS 확장자를 사용하는 스타일 시트 파일을 따로 작성하는 방식이다. 스타일 시트 파일을 따로 작성하여 웹 문서 <head> 내에 <link>태그로 웹 문서와 스타일 시트 파일을 연결할 수 있다. CSS파일을 별도로 작성 후 해당 CSS를 적용시킬 웹 문서들을 연결시켜주는 방식으로 사용한다.

CSS는 동일한 요소에 여러 style이 적용되는 경우 우선순위와 상속 규칙에 따라 스타일 충돌을 방지한다. 첫번째 우선순위는 중요도이며 사용자 스타일>제작자 스타일>브라우저 스타일 순으로 적용된다. 두번째 우선순위는 적용범위이며 작성 시 적용범위가 좁을수록 우선순위가 높아

진다. ! important >인라인 스타일>아이디 선택자>클래스 선택자>타입 선택자 순으로 적용된다. 세번째 우선순위는 적용순서이다. 가장 마지막에 적용되는 스타일이 가장 우선 적용되는 것이다. 동일한 요소에 style이 여러 곳에 적용되었을 경우 가장 마지막 스타일을 기준으로 적용된다. 또한 CSS는 상속 규칙에 따라 html문서에서 포함관계에 따라 부모요소, 자식요소로 구분할 수 있다. CSS에서는 부모요소에 스타일을 적용하고 자식요소에 따로 스타일을 적용하지 않을 경우 기본적으로 부모요소의 적용된 스타일을 자식요소가 상속받는다. 모든 속성이 상속되지는 않으며 상속 여부는 종류에 따라 다르다.

3)JavaScript

객체 기반의 스크립트 언어로 html로 웹의 내용을 작성하고 CSS로 스타일을 지정한 웹 페이지에 동적인 효과를 구현하여 사용자와의 상호작용이 가능하도록 만들어진 언어이다. 객체 지향형 프로그래밍과 함수형 프로그래밍을 모두 지원하는 언어로 클라이언트 측 웹 개발에서 핵심적인 역할을 수행한다. 웹 요소를 선택하고 필요에 따라 스타일을 변경하거나 애니메이션을 적용할 수 있으며 웹 사이트 UI 부분에 많이 활용한다. 또한 웹 사이트에 사용자와 실시간으로 정보를 주고받는 애플리케이션과 비슷한 동작도 JavaScript를 활용한 것이다. 이 외에도 다양한 라이브러리와 프레임 워크를 사용하여 서버 구성 및 서버용 프로그램 개발이 가능하다. <script>태그와 </script> 태그 사이에 자바스크립트 코드를 작성하며 웹 문서 안 어디든 위치할 수 있지만 주로 </body>태그 앞에 작성한다. 또는 자바스크립트 소스를 별도의 파일(*.js)로 저장한 후 <script src="파일경로.js"></script> 형태로 웹 문서에 연결하여 사용할 수 있다. JavaScript에서 값을 생성할 수 있는 모든 것은 표현식(expression)이 될 수 있으며 이러한 식은 변수에 저장된다. 문(statement)은 표현식이나 값을 포함하는 코드의 단위로 일반적으로 세미콜론(;)을 붙여서 구분하는 것이 좋다.

JavaScript는 Document Object Model(DOM)을 통해 웹 페이지의 구조, 스타일, 내용을 동적으로 변경할 수 있으며 이를 통해 사용자와의 상호작용에 따라 페이지 내용을 실시간으로 업데이트 할 수 있다. 웹 페이지를 트리 구조 형태로 표현한 것으로 각 html 요소는 하나의 노드로 간주된다. JavaScript는 이러한 노드들에 접근하고 수정할 수 있도록 다양한 API를 제공한다. 이러한 기능은 사용자 입력이나 시스템 상태 변화에 반응하여 동적인 사용자 경험을 제공하는 데 핵심적인 역할을 한다.

또한 이벤트 기반 프로그래밍을 지원하며 사용자의 상호작용에 따라 다양한 동작 수행이 가능하다 이벤트란 키보드 입력, 폼 제출, 마우스 클릭, 화면 터치 등 사용자와 브라우저에 의해 발생하는 특정 동작을 의미한다. 이벤트 처리를 위해 JavaScript는 addEventListener() 메서드를 제공하며 이를 통해 특정 요소에 이벤트를 지정할 수 있다. 버튼 클릭 시 팝업 생성 등이 이에 해당한다. 이벤트 버블링(bubbling), 캡처링(capturing) 등 이벤트의 전파 방식도 제어할 수 있어 복잡한 상호작용도 정교한 구현이 가능하다. 마우스 이벤트는 click, dblclick, mouseover, mouseout, mousedown, mouseup, mousemove, contextmenu가 있으며 키 이벤트는 keydown, keyup, keypress 폼 이벤트는 focus, blur, change, submit, reset, select 로드 및 기타 이벤트는 load, abort, unload, resize, scroll 드래그&드롭 이벤트는 drag, dragstart, dragend, dragover, dragenter, dragleave, drop의 종류가 있다.

4)PHP (Hypertext Preprocessor)

서버 사이드 스크립트 언어로 웹 서버에서 실행된다. Html과 결합하여 동적인 웹 페이지를 생성하며 서버 측에서 처리되며 다양한 데이터베이스와의 연동이 용이하여 다양한 웹 애플리케이션 개발에 적합하다. 동작원리는 클라이언트가 웹 브라우저를 통해 웹 서버에 원하는 웹 페이지를 요청하면 웹 서버는 클라이언트가 요청한 웹 페이지의 로직 처리를 위해 PHP parser에 이에 대한 처리를 요청한다. PHP parser는 로직 처리 후 처리 결과를 웹 서버로 전달한다. 웹 서버는 전달받은 데이터로 웹 페이지를 완성하여 웹 브라우저로 응답을 전송한다. PHP코드는 웹 페이지 어느 부분에서나 위치하며 PHP parser가 인식하기 위해 시작과 끝을 표시해야 한다. <?php ?> 형태로 작성되어야 한다.

3.반응형 웹과 핵심 원리 및 기술

반응형 웹 디자인(Responsive Web Design)은 하나의 웹사이트로 데스크톱, 태블릿, 스마트폰 등 다양한 디바이스 환경에 맞춰 자동으로 화면 레이아웃을 변경하여 보여주는 기술이다. 사용자가 디바이스 종류에 상관없이 일관적이고 편리하게 이용할 수 있도록 한다.

1)유동형 그리드 레이아웃 (Fluid Grid Layout)

고정된 픽셀 값이 아닌 백분율(%)과 같은 상대적인 단위를 사용하여 레이아웃을 구성한다. 이를 통해 화면 크기가 변할 때 요소들이 유연하게 재배치된다. CSS Grid나 Flexbox와 같은 레이아웃 기술을 이용하여 유동형 레이아웃을 쉽게 구현할 수 있다.

2)미디어 쿼리 (Media Queries)

CSS3의 핵심 기술 중 하나로 디바이스의 특성(화면 너비, 높이, 해상도 등)에 따라 다른 CSS 스타일을 적용할 수 있게 해준다. 특정 조건 때 특정 스타일을 적용하도록 설정하여 다양한 화면 크기에 최적화된 디자인을 제공한다.

3.웹 페이지 구현



III.결론

1.느낀점

웹은 일상에서 가장 많이 접하는 디지털 플랫폼 중 하나이며 본 보고서를 통해 그 원리와 개념을 직접 학습하고 구성 요소들을 체계적으로 고찰해볼 수 있었다. 이를 통해 평소 사용하던 웹 페이지들의 요소들도 인식하게 되었고 각 요소들이 어떤 방식으로 상호 작용하여 서비스를 구현하는 지에 대해 이해할 수 있었다. 특히 화면에 보이는 프론트엔드의 영역뿐만 아니라 그 이면의 백엔드 영역까지 고려하며 웹 전반 시스템에 대한 이해와 관심의 폭이 넓어진 것 같다. 나아가 웹을 구현하기 위해 백엔드 기술에 대한 학습의 필요성도 느끼게 되었다. 본 보고서는 웹 개발 기술의 기초와 개념을 다지는 동시에 향후 심화된 웹 프로그래밍 학습을 위한 유의미한 출발점이 되었음을 느낀다.

참고문헌

- 1)『Do it! HTML+CSS+자바스크립트 웹 표준의 정석』, 고경희, 이지스퍼블리싱, 2024.
- 2)『Do it! 웹 사이트 따라 만들기: HTML, CSS, 자바스크립트, JQuery, Ajax로 웹 퍼블리싱』, 김윤미, 이지퍼블리싱, 2019
- 2)김윤, 정민서,『웹 해킹 보안의 종류와 대응 방안』, 한경국립대학교 학사학위 논문, 2021, p.2~3
- 3)https://developer.mozilla.org/ko/docs/Learn_web_development/Howto/Web_mechanics