

HTML



1주차

웹과 HTML / CSS / JavaScript

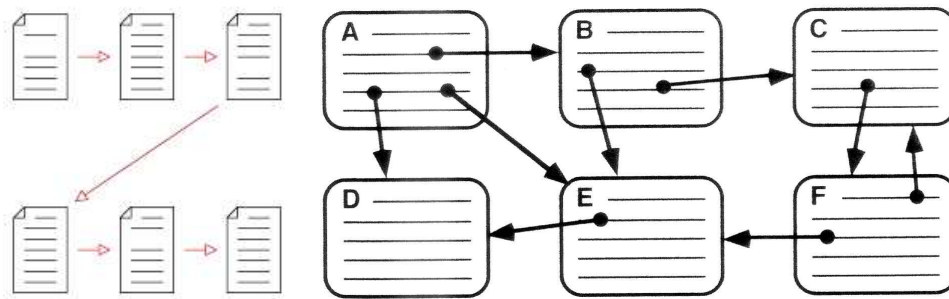
1. 웹에 대하여

- 웹

‘웹’이라는 단어는 맥락에 따라 다양하게 정의할 수 있습니다. 그러나 일반적으로 웹은 전 세계의 컴퓨터들이 연결된 네트워크인 ‘인터넷’을 통해서 ‘하이퍼텍스트’로 작성된 문서를 공유할 수 있는 서비스를 의미합니다.

- HyperText와 DOM

전통적인 방식으로 작성된 문서는 저자가 의도하고 작성한 순서대로 읽어야만 전체 정보를 습득할 수 있습니다. 또, 내가 특정한 부분만을 다시 읽고 싶다고 해도 그 부분에 도달하기 위해서는 페이지와 페이지를 넘기고 일일이 눈으로 확인하는 과정을 거쳐야만 합니다. 이렇게 선형적 구조를 가진 문서는 다루어야 하는 정보량이 비약적으로 많아지는 가운데 원하는 정보를 빠르게 검색하고 선택하는 작업이 중요해진 현대에는 어울리지 않게 되었습니다.



▲ 선형적 구조를 가진 문서와 하이퍼텍스트 문서

인터넷과 웹, 하이퍼텍스트, HTML의 창시자인 팀 버너스 리는 선형적 문서가 가진 문제점들을 ‘하이퍼링크’라는 개념을 도입해 해결하고자 했습니다. 보르헤스의 단편 『두 갈래로 갈라지는 오솔길들의 정원』에서 영감을 받았다고 전해지는 하이퍼링크는 한 문서에서 다른 문서에 즉각적으로 접근할 수 있게 해주고, 독자가 직접 클릭이라는 방식을 통해 문서를 읽는 방식을 결정하도록 해줍니다.

하이퍼링크가 가장 전형적으로, 그리고 가장 실용적으로 사용되고 있는 곳은 바로 검색엔진입니다. 대부분의 검색엔진에서 우리는 하이퍼링크들로 구성된 ‘하이퍼텍스트’를 검색결과로 받아보고, 그 비선형적인 구조의 문서 속에서 우리가 원하는 정보들만을 선택해 사용합니다. 그 때문에 우리가 접하는 대부분의 하이퍼텍스트는 컴퓨터상에서 읽고 쓰는 디지털 문서로 구현된 것들입니다. 그러나 종이로 된 매체나 음악, 영화, 게임 등에서도 하이퍼텍스트의 비선형적 구조를 도입한 시도들이 많이 있습니다. 대표적인 것이 게임북이나 인터랙티브 무비 같은 것들입니다.



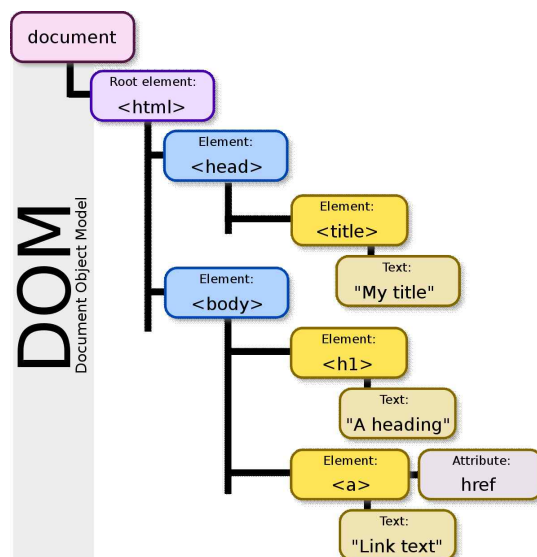
▲ 게임북도 일종의 하이퍼텍스트다.



▲ 넷플릭스의 <블랙미러: 밴더스내치>

하이퍼텍스트가 도입되면서, 작성자가 문서를 작성할 때 고려해야 할 것들도 상당히 달라졌습니다. 기존의 글이나 책을 저술할 때는 문서의 내용과 전개방식이 중요했다면 하이퍼텍스트는 문서 내 요소들 각각이 얼마나 체계적으로 객체화되어있고 효율적으로 검색될 수 있는지 까지를 신경써야합니다. 하이퍼텍스트를 작성하는 언어는 여러 가지가 있지만 현재 가장 널리 사용되는 언어는 HTML (Hyper Text Markup Language)이고, HTML로 작성된 문서를 전송하기 위한 통신 규약이 바로 HTTP (Hyper Text Transfer Protocol)입니다. 기존에는 이 HTML을 사용하는 방식도 여러 갈래가 존재했지만, 월드와이드웹 컨소시엄이 웹 표준과 가이드라인을 확립하면서 점차 통합되어왔습니다.

이때 특히 문서 객체 모델, DOM (Document Object Model)이라는 것을 사용하여 문서를 구조화합니다. DOM은 우리가 작성하는 프로그램이 HTML로 작성된 문서에 접근하고, 내용이나 서식을 수정할 수 있도록 도와주는 인터페이스입니다. 요즘에는 HTML 문법이 곧 DOM 문법과 마찬가지로이기 때문에 여기에 대해서는 이후에 자세히 다루도록 하겠습니다.



▲ DOM 구조도의 예시

- 홈페이지, 웹사이트, 웹페이지

HTML과 DOM을 사용하여 작성한 문서 하나하나를 ‘웹페이지’라고 부릅니다. 그런데 하이퍼텍스트의 특성상, 그 장점을 극대화하기 위해서는 웹페이지 여러 개가 구조적으로 결합되어 있어야 할 것입니다. 이렇게 여러 웹페이지들이 결합되어 있는 것을 ‘웹사이트’라고 부릅니다. 그리고 그 중에서도 사용자가 가장 먼저 접하게 되고 주된 기능을 한다고 볼 수 있는 웹페이지를 ‘홈페이지’ 또는 ‘메인페이지’라고 부릅니다.

- 웹 접근성

하이퍼텍스트 문서가 매우 편리하고 혁신적인 아이디어인 것은 사실이지만, 장점만 있는 것은 아니었습니다. 시각각 장애나 신체움직임에 장애를 가진 사람들처럼 컴퓨터를 활용하기 어려운 환경에 놓인 사람들에게는 하이퍼텍스트가 오히려 기존의 종이 문서보다도 훨씬 더 사용하기 어려웠기 때문입니다. 하이퍼링크라는 방식 자체가 시각과 손의 움직임에 워낙 많이 의존하고 있다 보니 더 심해지는 문제이기도 합니다. 그 때문에 최근에는 ‘웹 접근성’이라는 개념이 주목받고 있습니다. 화면을 그대로 소리로 바꾸어 읽어주는 스크린 리더나 키보드 오버레이, 화면 확대 도구 같은 것들이 충돌없이 작동할 수 있도록 웹사이트를 구성하여 장애인과 비장애인이 모두 동등하게 웹사이트에 접근하고 이용할 수 있도록 하자는 것입니다.

W3C가 웹 접근성을 웹 표준에 포함시키면서, 단순히 장애인을 존중하는 차원을 넘어서 웹접근성을 고려한 웹사이트가 그렇지 않은 웹사이트에 비해 훨씬 더 높은 가치를 지닌 정보로 평가받고, 검색결과에서도 더 우선 노출되고 있습니다. 최근에는 공공기관이나 포털사이트 등 일상 생활에 필수적인 웹사이트일수록 웹 접근성을 고려하여 서비스하도록 권고됩니다. 웹 접근성을 확보하는 방법에는 별다른 것이 없습니다. 똑같은 기능을 하는 버튼을 만들더라도 이미지 파일로 된 버튼 보다는 텍스트로 되어있는 버튼을 만들고, 텍스트가 있는 이미지 파일을 업로드 할 때에는 이미지에 적힌 문자를 그대로 대체 텍스트로 입력하여 스크린 리더가 읽을 수 있는 방식으로 웹페이지를 구성하는 등 간단한 방법부터 실천에 옮기면 됩니다.



중앙대학교 인문콘텐츠연구소 HK+사업단 2021 AIH 심포지엄 진행 김형주 HK교수 매회 14:00 - 17:00 Zoom 비대면 회의 혹은 중앙대 310관 901호에서 개최 (<https://cau.zoom.us/j/85980616202>) 1월 22일 <저술성과 발표회> "인공지능시대의 사진이미지 읽기" - 박평중, 박주석, 천경우 2월 26일 박소영 - "이시구로의 나를 보내지 마 (Never Let Me Go): 인공지능을 통해 살펴보는 감정, 예술 그리고 기술의 상호 관계" 이윤미 - "AI 리터러시 개념 설정에 관한 소고" 3월 26일 이수진 - "세운상가(X)=미디어아트" 박평중 - "이미지 생성 알고리즘의 최근 동향: GAN을 중심으로" 4월 23일 박지영 - "Attitudes towards artificial intelligence application at a work" 김희영 - "인공지능의 가치판단 알고리즘개발을 위한 도덕 가치 온톨로지 구축" 5월 27, 28일 <저술성과 발표회> "Kant and AI" - 김형주, Dieter Schonecker 6월 25일 조희련 - "딥러닝 기반 언어 모델을 이용한 유학생 한국어 쓰기 담당자 자동분류 연구" 남영자 - "딥러닝 분석을 활용한 방송의 감정 연구: 공중파, 종합편성 그리고 유튜브 방송을 중심으로" 7월 30일 황서이 - "인공지능 알고리즘을 활용한 포털 뉴스 편집" 양재혁 - "인공지능 시대 미셸데세르토(Michel de Certeau)의 하이브리드 역사 서술 다시읽기 또는 다시쓰기" 8월 20일 이재진 - "CNN 물다보기" 심지원 - "4차산업혁명에서 융합윤리학의 과제" 주최 한국연구재단 교육부

▲ 웹 접근성을 위한 이미지 대체 텍스트 예시



II. HTML이란 무엇일까요?

- HTML은 프로그래밍 언어가 아닙니다!

컴퓨터를 다루는 것이 익숙하지 않거나 이제 막 배우기 시작한 분들일수록 HTML이라는 언어에 대한 궁금증도, 기대도 크실 것이라고 생각합니다. 앞서 잠시 설명한 HTML과 하이퍼텍스트의 혁신적인 장점 때문에 ‘HTML은 정말 훌륭한 프로그래밍 언어겠군!’이라고 여기실지도 모르겠습니다.

결론부터 말씀드리자면, HTML은 프로그래밍 언어가 아닙니다. 더 정확히 말하면 HTML은 하이퍼텍스트를 구조적으로 작성할 때 사용하는 마크업 언어 중 하나로 Hyper Text Mark-up Language의 줄임말입니다. 하이퍼텍스트에 대해서는 이미 충분히 설명했기 때문에 여기에서 우리가 주목할 부분은 ‘Mark-up Language’입니다. 문서를 ‘마크 업’ 한다는 것은 문서의 내용과 화면에 표시될 형태를 표시하는 동시에 데이터의 논리적 구조를 명시하는 작업을 의미합니다. 따라서 HTML은 프로그래밍 언어들이 일반적으로 가지는 절차성, 조건 분기, 반복문 등의 특성을 지니지 않고, 그렇기 때문에 튜링완전하지도 않습니다. 다시 말해서 ‘프로그래밍 가능한 (programmable)’ 언어가 아닌 것입니다.



▲ HTML은 프로그래밍 언어가 아닙니다!

III. CSS와 JavaScript

- HTML이 차려놓은 밥상에..

그렇다면 우리가 매일 사용하는 웹사이트들은 어떻게 만든 것일까요? 단지 문서를 조금 더 체계적으로 작성하는 것뿐이라면 하이퍼링크 기능을 제외하고는 텍스트 파일과 별 차이가 없어 보이는 것 같은데도, HTML을 프로그래밍 언어로 착각할 만큼 웹에서 사용할 수 있는 프로그램이 수두룩하니 말입니다. 사실, HTML은 그 자체만으로 사용될 때 보다는 다른 두 언어의 도움을 받을 때 그 진가를 발휘합니다.

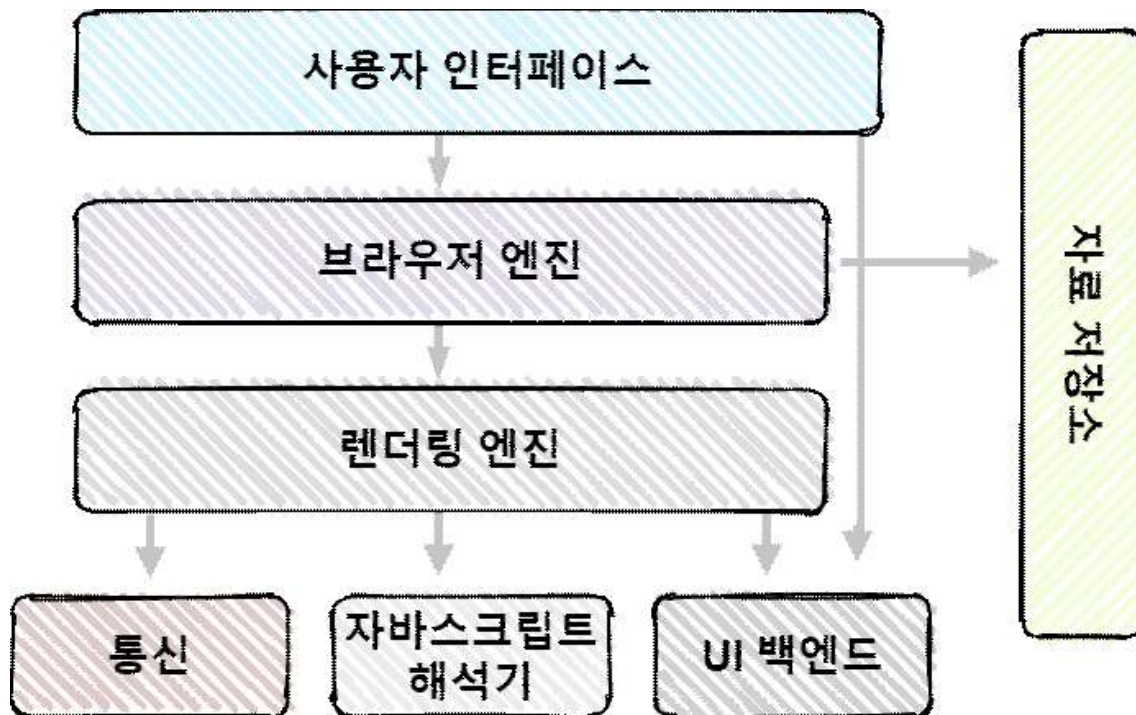


▲ HTML / CSS / JavaScript의 차이

우리가 사용하는 거의 모든 웹사이트를 구성하고 있는 것은 HTML뿐만으로 되어있는 것이 아니라 CSS, JavaScript, 그리고 그 외의 프레임워크나 라이브러리들의 조합입니다. HTML은 웹 페이지의 전체적인 구조를 객체화하는 일을 담당하고, CSS는 HTML이 만들어놓은 구조 위에 꾸미기 요소들을 덧씌워줍니다. 그리고 JavaScript가 프로그래밍 언어로서 '웹 어플리케이션'을 구성하고 개발자가 의도한 기능들을 적절히 수행합니다. 그러니까 사실은 '웹 프론트엔드 개발자'가 되려면 HTML, CSS, JavaScript를 기본으로 배우고 거기에 더해 웹 프레임워크나 라이브러리를 따로 공부해야 하는 것입니다. 이 수업에서는 각각을 아주 간단히만 훑어보도록 하겠습니다.

- 웹 브라우저와 작동 원리

웹에 접속할 때 하나하나 일일이 수동으로 서버에 요청을 보내고 다운로드 받은 내용을 코드 그대로 읽는 사람은 거의 없습니다. 대부분의 사용자는 서버로부터 받은 HTML, CSS, JavaScript 파일을 웹 브라우저가 해독하고 띄워주는 화면을 봅니다. 웹 브라우저에는 구글의 Chrome, 모질라 재단의 Firefox, 마이크로소프트의 Internet Explorer와 Edge, 애플의 Safari 등 다양한 종류가 있지만 2021년 말 기준 Chrome이 약 53%를 차지하면서 압도적인 점유율을 가지고 있습니다. 브라우저의 종류가 달라도 비슷비슷한 구조를 가지고 있는데, 크게 유저 인터페이스, 브라우저엔진, 렌더링엔진, 통신부, 자바스크립트 해석기, UI 백엔드, 데이터 저장소 등으로 이루어져 있습니다.



▲ 웹 브라우저의 구조

Q. WEB 2.0? WEB 3.0?

IV. 실습환경 구축하기

- Visual Studio Code 설치하기
- 유용한 플러그인 소개 및 설치하기
- 최초의 웹사이트를 방문해보자!
- 웹 브라우저에서 HTML 소스코드를 보는 방법
- 모바일 환경에서 HTML 소스코드를 보는 방법