

# 웹 프로그래밍 강의노트

## 웹 문서의 기본적 이해

2024년 2학기

동명대학교

# HTML에 대한 기본적 이해

- 마크업(Mark-Up)의 유래 :  
2pt 활자의 식자를 위한 수기형태의 주석
- 마크업 언어는 :  
10pt 문서의 구조와 내용에 추가적인 의미를 부여하는 마크업 규칙을 규정하는 언어

# HTML: HyperText Markup Language

- 마크업 언어(Markup Language)
  - 인쇄 교정지의 '마크-업(Mark-up)'에서 유래
  - 문서의 속성을 설정하기 위한 마크업을 태그의 형태로 표시
  - 대표적인 마크업 언어 : SGML, HTML, XML 등
    - HTML 언어는 SGML 표준에 따라 정의
    - 일반 텍스트 형식의 파일로 저장되며 확장자는 \*.html 또는 \*.htm

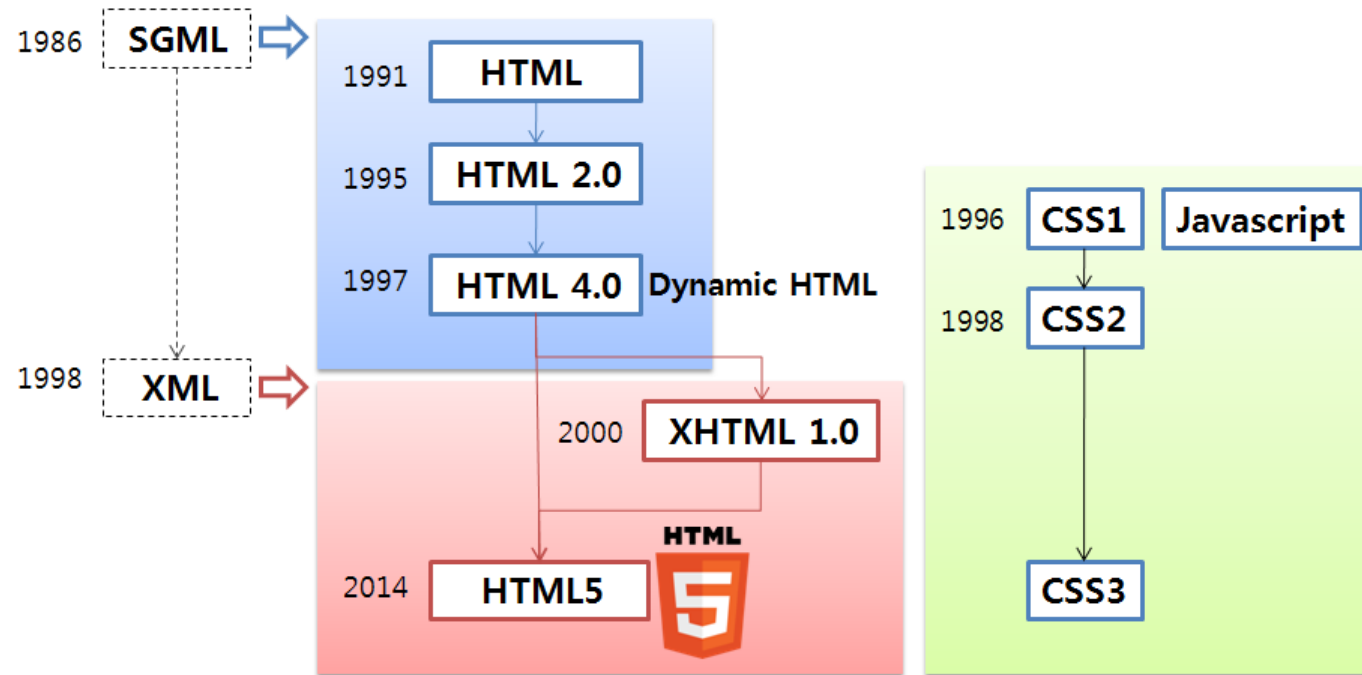
• 마크업(Mark-Up)의 유래 :  
활자의 식자를 위한 수기형태의 주석  
• 마크업 언어는 :  
문서의 구조와 내용에 추가적인 의미를 부여하는 마크업 규칙을 규정하는 언어

```
<li><font size="12pt">마크업(Mark-Up)의 유래 :</font>  
<br><font size="10pt">활자의 식자를 위한 수기 형태의 주석</font></li>  
<li><font size="12pt">마크업 언어 :</font>  
<br><font size="10pt">문서의 구조와 내용에 추가적인 의미를 부여하는 마크업 규칙을 규정하는 언어</font></li>
```

- SGML(Standard Generalized Markup Language)
  - 1986년 국제표준기구인 ISO에서 개발
    - 다양한 형식의 전자문서들의 구조와 내용을 기술하는 국제표준
  - 시스템 및 응용에 독립적으로 문서를 호환하기 위한 목적
    - 전자도서, 전자상거래 문서 등 다양한 문서 형식을 정의하는데 사용
    - HTML은 SGML로 정의된 문서 형식으로 주로 웹문서 작성에 사용
- HTML(HyperText Markup Language)
  - 1994년 HTML 버전 2.0, 1997년 HTML 4.0 버전
  - 배우기 쉽고 사용하기 편리하여 인터넷의 대중화에 매우 큰 기여
    - 반면에 태그가 제한적이고 정교한 페이지를 표현하기에는 부족
  - HTML 4.0에서는 동적 HTML (Dynamic HTML) 문서 표현
    - 스타일시트를 설정하는 CSS 기능과 상호작용을 코드로 표현하기 위한 자바스크립트 (Javascript) 언어가 포함.

- XML(eXtensible Markup Language) 및 XHTML
  - 문서나 자료의 교환이 필요한 경우 새로운 언어가 필요
  - SGML을 간소화한 XML이 1998년 제정
    - XML 언어로 원하는 문서 형식을 정의하여 다양한 정보를 표현/교환
  - HTML 언어도 XML에 기반한 XHTML로 발전
    - XHTML 1.0은 XML로 문서형식만 정의, 태그는 그 전의 HTML과 동일
- HTML5
  - 웹 하이퍼텍스트 워킹그룹(WHATWG)
    - 모질라, 애플, 오페라 등 웹브라우저 개발사 주도로 2006년 결성
  - W3C도 WHATWG과 협력하여 2007년 HTML5 워킹그룹 신설
  - 2014년 10월 28일 최종 표준안 확정

## • HTML의 발전 역사



# SGML 예시

```
<!DOCTYPE article [  
  <!ELEMENT article (title, body)>  
  <!ELEMENT title (#PCDATA)>  
  <!ELEMENT body (#PCDATA)>  
>]
```



이 문서는 article 타입의 문서



사용되는 요소는 article, title, body  
article은 title과 body로 구성됨  
title과 body는 “parsed character data” 가 담김

```
<article>  
  <title>Introduction to SGML</title>  
  <body>This article introduces SGML, a language used to define the structure of documents.</body>  
</article>
```

# HTML 예시

```
<!DOCTYPE html>
```

```
<html>
```

```
<head>
```

```
  <title>Note</title>
```

```
</head>
```

```
<body>
```

```
  <article>
```

```
    <h2>Note</h2>
```

```
    <p><strong>To:</strong> John</p>
```

```
    <p><strong>From:</strong> Alice</p>
```

```
    <p><strong>Message:</strong> Don't forget the meeting tomorrow at 10 AM.</p>
```

```
  </article>
```

```
</body>
```

```
</html>
```



이 문서는 html 타입의 문서

사용되는 요소는 미리 약속되어 있음



# XML 예시

`<?xml version="1.0" encoding="UTF-8"?>`  이 문서는 xml 타입의 문서

`<note>`

`<to>John</to>`

`<from>Alice</from>`

`<message>Don't forget the meeting tomorrow at 10 AM.</message>`

`</note>`

XML은 HTML에 비해 훨씬 더 잘 구조화되어 있고, 유연하다  
그러나 SGML에 비해서는 규칙이 더 많다.

유연성의 정도: SGML > XML > HTML

XML은 SGML의 부분집합에 해당하며,  
HTML보다는 더욱 메타언어에 가까워 유연하게 마크업을 정의할 수 있다.  
그러나 XML은 웹상에서 데이터를 쉽고 효과적으로 교환하는 데에 집중한 언어라 SGML보다 규칙 단순

# XHTML 예시

```
<?xml version="1.0" encoding="UTF-8"?>  ← 이 문서는 xhtml (eXtensible HyperText Markup Language)
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Strict//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-strict.dtd">
<html xmlns="http://www.w3.org/1999/xhtml" lang="en">
  <head>
    <title>Example XHTML Page</title>
  </head>
  <body>
    <h1>Welcome to XHTML</h1>
    <p>This is an example of an XHTML document.</p>
    
    <br />
    <a href="https://example.com">Visit Example.com</a>
  </body>
</html>
```

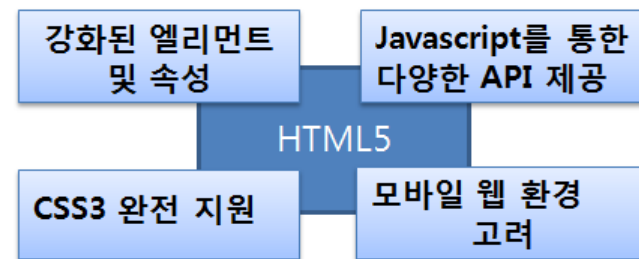
- Xhtml은 XML 구문으로 작성된 HTML
- HTML보다 엄격한 규칙
  - HTML과의 호환: 브라우저에서 볼 수 있음

Aspect	XHTML	HTML
Syntax	Stricter, XML-compliant	More flexible
Tag Closing	All tags must be closed ( <code>&lt;br /&gt;</code> )	Some tags can remain unclosed ( <code>&lt;br&gt;</code> )
Nesting	Tags must be properly nested	HTML allows some flexibility with nesting
Case Sensitivity	Tags and attributes must be lowercase	Case-insensitive
Parsing	Can be parsed as XML	Parsed as HTML

# HTML5 언어의 특징 – HTML4 + XHTML

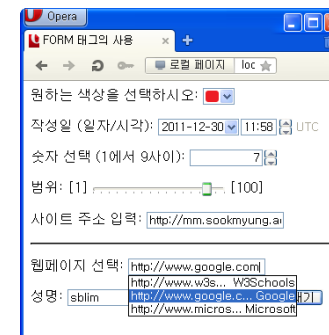
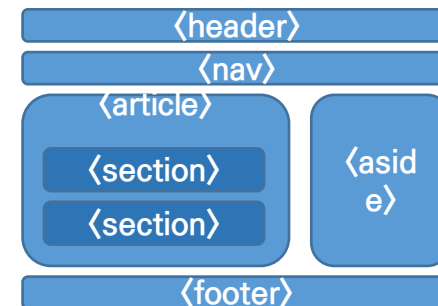
- HTML 4.0 이후 요구사항
  - 다양한 인터페이스, 다양한 형식의 미디어 파일, 비동기 처리
  - 웹 표준 기술을 사용하자는 시도
    - 웹 애플리케이션 개발에 XML, CSS, 자바스크립트 등 이용
- HTML5의 방향
  - 마크업에 보다 의미를 부여, 스타일은 분리하도록 CSS3 활용
  - 플러그인 대신 웹 표준을 적용하도록 SVG, MathML 지원
  - 인터랙션 개발을 위해 자바스크립트를 지원
  - 웹 애플리케이션의 개발을 위하여 다양한 API를 제공
    - 특히 위치관련 및 오프라인 등 모바일 환경까지 고려한 API 제공

가장 최신식 HTML



# 강화된 마크업 요소

- 의미를 부여할 수 있는 구조적 마크업 요소 [2장에서 설명]
  - 페이지 단위의 문서 구조를 표현
- 다양하고 편리한 웹 폼(WebForm) 입력 기능 [6장 참조]
  - 편리한 사용자 인터페이스 개발을 위해 Form 기능 대폭 개선
  - <input> 요소에 date, number, color, file 등 각종 type 속성 추가
- 웹 미디어 기능의 강화 [3장, 11장 참조]
  - 멀티미디어 및 그래픽스 관련 기능의 추가
  - <video> 요소와 <audio> 요소
  - MathML



# CSS3의 완전 지원

- 스타일시트(Stylesheet)
  - 웹 문서의 외형 스타일을 지정하는 언어
  - HTML 문서에는 일반적으로 CSS(Cascading Style Sheet)가 사용
  - 1996 CSS1, 1998 CSS2, 2005년 이후 CSS3는 모듈별로 개발 중
- HTML5에서는 CSS1, CSS2, CSS3까지 완전히 지원
  - 기존 CSS는 주로 텍스트, 배경 및 색상, 목록, 박스모델 등 속성
  - CSS3에는 더욱 다양한 스타일 지정 기능을 포함
  - 예제 화면
    - 박스의 모서리 처리, 그림자 효과, 다단(multi-column) 지정, 텍스트의 그림자, 콘텐츠 위치 지정, 등 다양한 효과
    - 이 외에도 장면 변환(transition), 애니메이션, 사용자 인터페이스에 관련된 속성



# 다양한 API 지원

- 웹 애플리케이션 개발에 많은 도움을 줄 수 있는 다양한 API 제공
  - 자바스크립트로 구현 가능
  - <video>와 <audio> 요소를 제어하는 API
  - <canvas> 요소에 그림을 그리는 API 등
- 별도의 사양으로 분리된 API
  - 드래그앤 드롭(Drag & Drop), 위치정보(Geo-Location), 웹 스토리지(Web Storage), 웹 소켓(Web Sockets), 파일(File), 애플리케이션 캐시 (Application Cache), 인덱스드 데이터베이스(Indexed Database) 등
  - 이들 모두가 HTML5의 기능
  - 자바스크립트로 매우 다양하고 강력한 기능의 웹애플리케이션 구현 가능해짐 (자바스크립트는 8~10장에서 설명)

# 모바일 웹 환경 고려

- 모바일 환경
  - 일찍부터 HTML5를 지원하는 웹 브라우저를 탑재
- 모바일 환경을 위한 HTML5의 특별한 기능
  - 위치정보 API
    - 위치정보는 모바일 애플리케이션 개발에 최적 활용
  - 오프라인 어플리케이션 API
    - 모바일 환경에서 접속이 끊기거나 트래픽 최적화에 긴요하게 활용
    - 오프라인이 될 경우에 로컬 스토리지, 웹 데이터베이스, 어플리케이션 캐시 등이 유용하게 활용
  - 새로운 유형의 다양한 입력 폼 지원
    - 모바일 애플리케이션의 사용자 인터페이스 개발 및 사용이 편리

# 인터넷의 기본 개념

클라이언트-서버 모델

인터넷의 전송방식

주소체계



# 클라이언트-서버 모델

- 클라이언트-서버 모델(Client-Server Model)을 기반

- 서버 vs 클라이언트

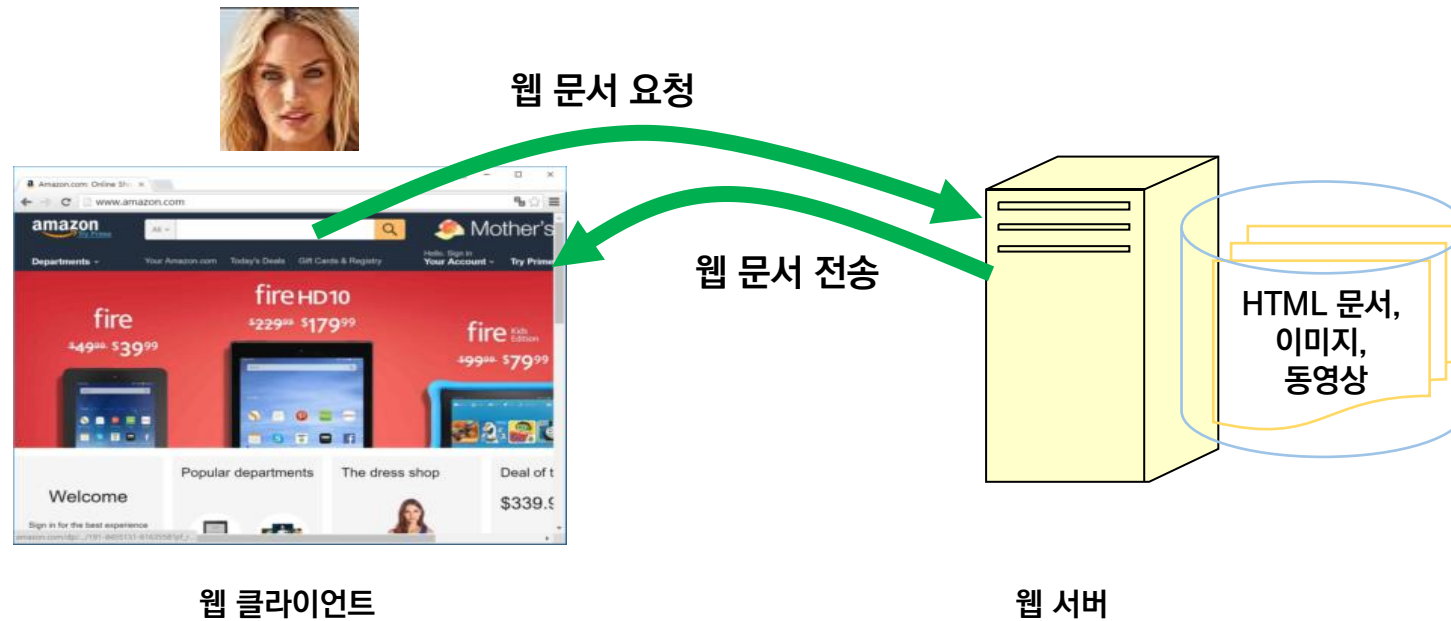
- 서버 : 제공하는 서비스에 적합한 정보들을 보관하고 이를 외부에 공개해주는 컴퓨터
    - 클라이언트 : 서버에서 제공하는 정보를 받기 위해 사용자가 작동시키는 컴퓨터

- 클라이언트-서버에 필요한 프로그램

- 서버 역할을 수행하는 서버 프로그램이 필요
      - 웹을 이용하려면 아파치(Apache) 또는 IIS라는 웹 서버 프로그램
    - 서비스를 이용하기 위한 클라이언트 프로그램
      - 크롬(chrome)이나 익스플로러(Explorer)와 같은 웹브라우저



# 웹 서버와 웹 클라이언트의 작동



# 인터넷과 웹은 다르다

- 인터넷
  - 웹의 개념이 나오기 전부터 만들어진 컴퓨터 연결 네트워크
    - 1969년 미 국방성 고등 연구 계획국(ARPA)
    - 여러 대학들과 계약 업체 사이의 컴퓨터 연결
  - 컴퓨터마다 고유한 주소(IP 주소, 113.198.80.208)를 부여하여 컴퓨터 구분
  - 인터넷을 활용하는 응용 서비스 사례
    - 전자우편(e-mail)
    - 뉴스(news)
    - 파일 전송(ftp)
    - 채팅(Internet Relay Chat)
    - 메신저(Messenger)
    - P2P(Peer to Peer)
    - 스트리밍 서비스(Streaming Service)
    - 인터넷 전화기(Internet Phone)
    - 월드 와이드 웹(World Wide Web)
- 월드 와이드 웹, 웹(WWW)
  - 인터넷을 활용하는 응용 서비스 중의 하나
  - 웹 서버와 웹 브라우저로 구성되는 정보 전달 및 공유 서비스
- 인터넷이 고속도로라면 웹은 고속도로 망을 이용한 물류 산업

# 인터넷의 전송방식

- TCP/IP 통신 프로토콜

- 인터넷은 운영체제나 데이터 형식이 다양한 종류의 컴퓨터 연결
  - 모든 컴퓨터에 IP 주소를 할당하고, 데이터를 주고받기 위하여 필요
- 통신 프로토콜 : 컴퓨터간의 통신 규약
  - 컴퓨터 사이에 정보를 전달하기 위하여 필요한 규칙 및 약속의 집합
- TCP(Transmission Control Protocol)
  - 데이터의 흐름을 제어하고 데이터가 정확한지 확인하는 역할
  - 데이터를 여러 개의 작은 조각으로 나누어 패킷(Packet)이란 정보단위를 생성하고 패킷이 제대로 전송되는지 확인
- IP(Internet Protocol)
  - 데이터를 이동시킬 목적지를 지정하는 역할

- IP(Internet Protocol) 주소
  - 패킷을 어떻게 목적지까지 보낼 것인가에 대한 전송 프로토콜
    - 우체국에서 편지를 보내기 위한 우편규칙과도 유사
  - IP 주소 : 정보 전송을 위한 주소체계로 숫자로 표현
    - 인터넷에 접속되어 있는 모든 호스트 컴퓨터들은 고유한 IP 주소
    - 222.122.84.200와 같이 '.'으로 구분하여 표현
  - IPv4 : 4 바이트 주소체계
    - $256 \times 256 \times 256 \times 256$ , 약 42억개의 주소 사용 가능
  - IPv6 : 16바이트 주소체계
    - 사물인터넷 시대에 주변의 모든 기기에 인터넷 IP 주소 할당할 필요
    - $2^{8 \times 16} = 2^{128}$  개의 충분한 개수의 주소를 사용

# OSI 7계층 모델과 TCP/IP

OSI Layer	Corresponding TCP/IP Layer	Key Protocols/Services
7. Application	Application	HTTP, FTP, SMTP, DNS, Telnet, SNMP
6. Presentation	Application	Data translation, encryption, SSL/TLS
5. Session	Application	Session management, RPC, NetBIOS
4. Transport	Transport	TCP (reliable), UDP (unreliable), port numbers
3. Network	Internet	IP, ICMP, ARP, routing protocols (BGP, OSPF)
2. Data Link	Network Access	Ethernet, Wi-Fi, MAC addresses
1. Physical	Network Access	Physical transmission (cables, signals)

# 주소체계

- 도메인 이름(Domain Name)

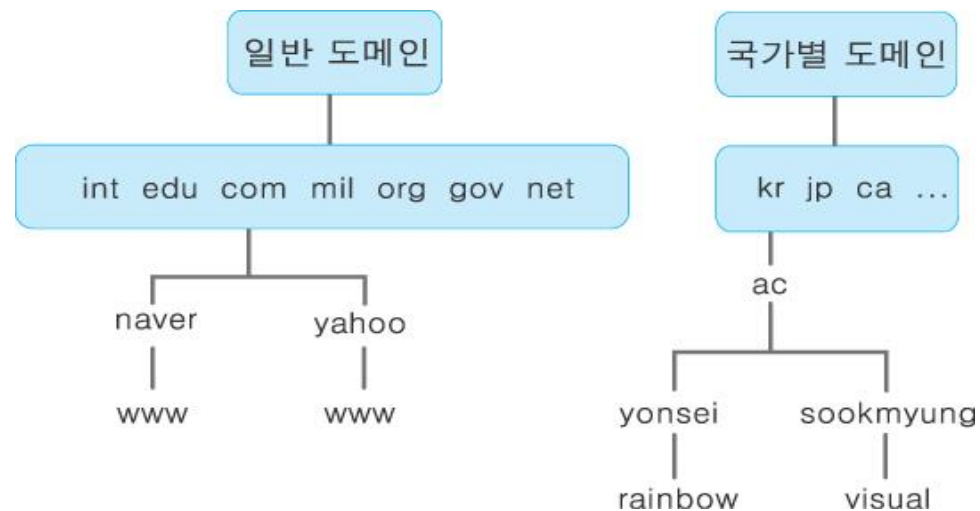
- IP 주소는 숫자로 구성되어 있어서 이용/기억하기 불편
- 도메인 이름 : 쉽게 기억할 수 있도록 IP주소를 문자로 대체
  - 각 호스트 컴퓨터는 하나씩의 IP 주소와 도메인 이름 보유
  - 도메인 이름 서버(Domain Name Server) : 도메인 이름을 IP주소로 변환

- 국가별 도메인 방식

- “호스트이름.소속기관.단체성격.소속국가”의 4단계
- 예, www.daum.co.kr

- 일반 도메인 방식:

- “호스트이름.소속기관.단체성격”의 3단계
- 최상위 도메인이 소속국가가 아닌 단체성격
- 예, www.yahoo.com



- URL(Uniform Resource Locator)
  - 웹에서 각종 파일 등 자원의 위치를 표시하는 표준
  - 서버에서 지원되는 다양한 서비스를 표현하도록
    - HTTP, FTP, Telnet, mail-to 등
  - 구성형식: “서비스프로토콜://도메인주소/폴더경로/파일이름”
    - 예, <http://cafe.naver.com/swimmingtraining/10294>
    - 예, <ftp://it.sookmyung.ac.kr/~sblim/intro.html>



# 1.4 웹브라우저의 종류

1.4.1 웹브라우저의 소개

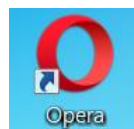
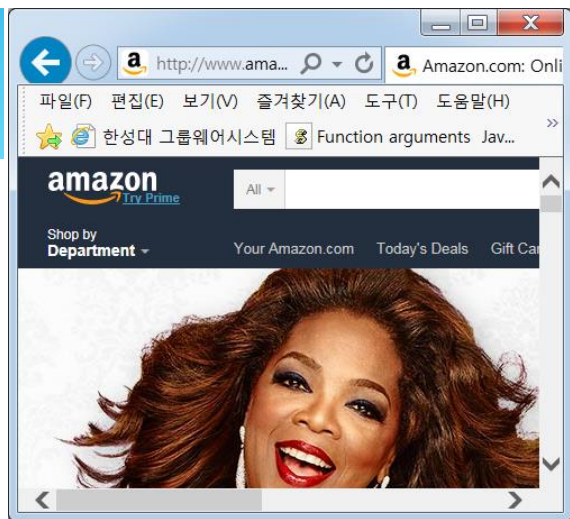
1.4.2 대표적인 웹브라우저들

# 웹브라우저의 소개

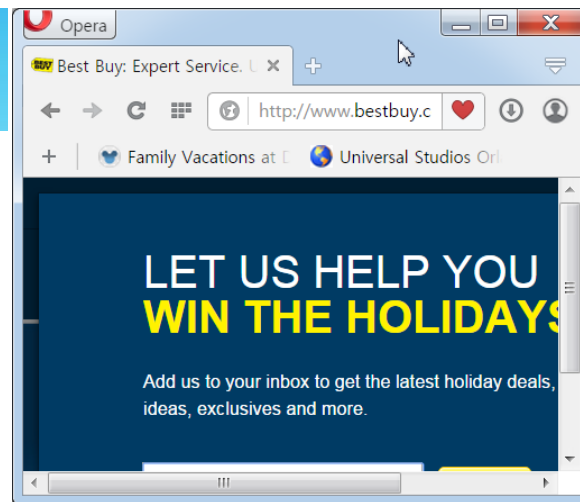
- 웹브라우저
  - 사용자 쪽의 클라이언트 소프트웨어
  - 웹서버에 있는 하이퍼텍스트정보를 사용자 화면에 보여주는 역할
  - 하이퍼링크의 내비게이션(navigation)을 도와주는 기능도 포함
  - 텍스트와 이미지 뿐만 아니라 멀티미디어 처리 기능까지 포함
- 웹은 기술의 발전 속도가 매우 빠른 분야
  - dynamic HTML, HTML5, CSS, Javascript, DOM, XML, mobile Web 등
  - 표준안이 정착되기도 전에 새로운 기술이 개발되어 신속히 적용
    - 웹브라우저는 버전에 따라서 제공되는 정보의 종류가 다르다



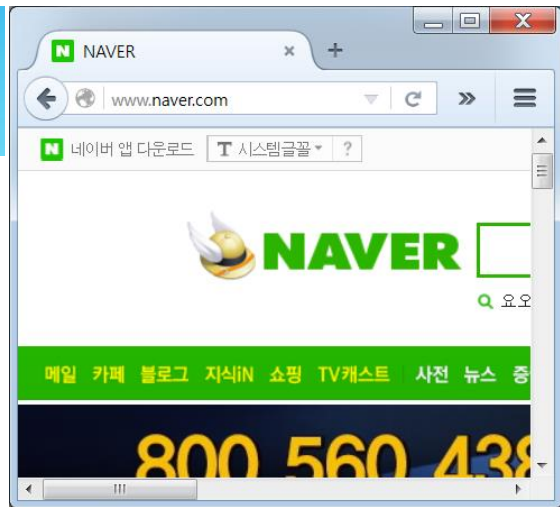
인터넷  
익스플로러



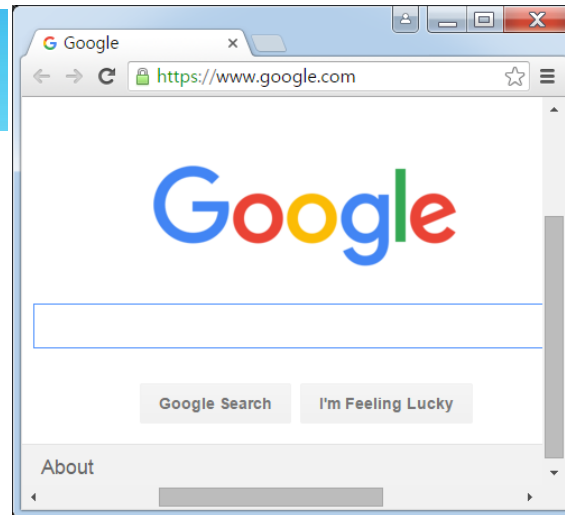
오페라



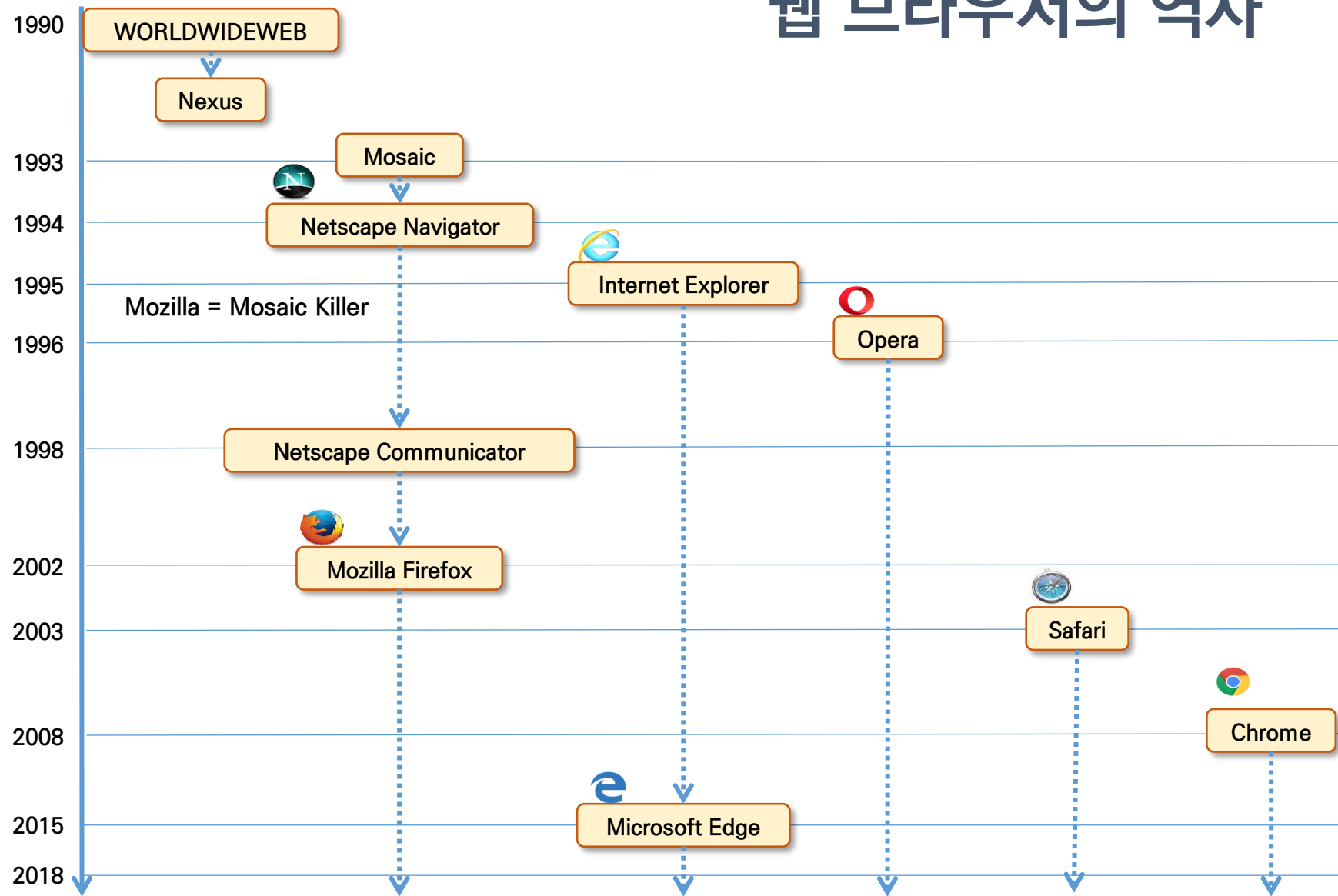
파이어폭스



크롬

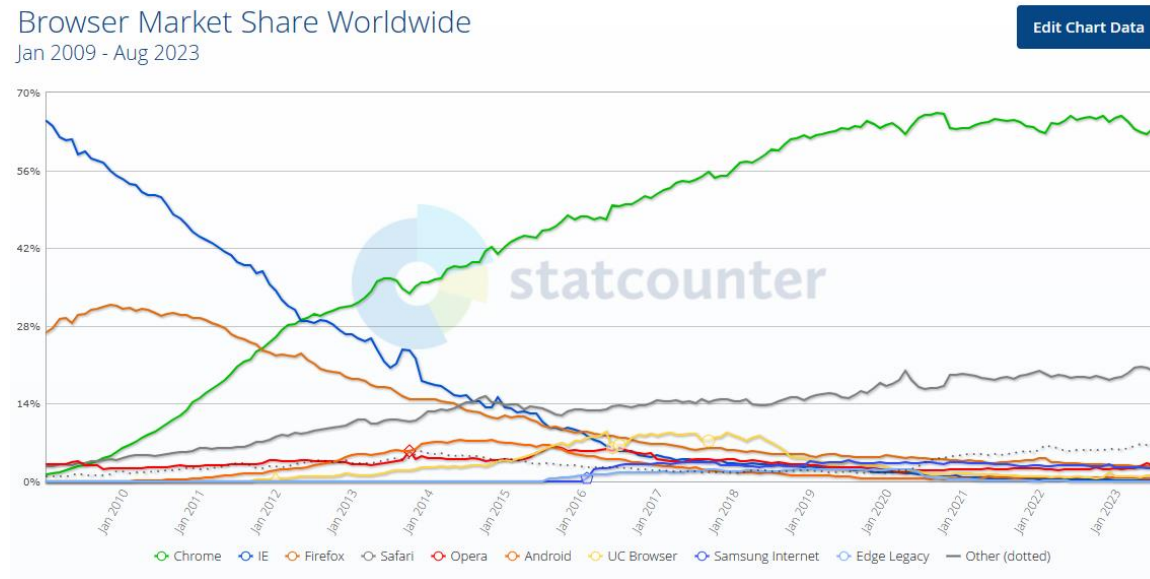


# 웹 브라우저의 역사



# 대표적인 웹브라우저들

- 1994년 넷스케이프 내비게이터 출시, 웹의 대중화에 큰 기여
- 1998년부터 당분간 인터넷 익스플로러의 독점체제
- 2000년대 중반 이후 여러 경쟁 제품이 등장, 치열한 점유율 경쟁
  - 크롬, 인터넷 익스플로러, 파이어폭스, 사파리, 오페라



출처 : <http://gs.statcounter.com>)

# 웹 프로그래밍 작업환경 준비하기

코드 편집하기

웹브라우저에서 실행하기

웹브라우저 개발자 도구 사용하기

# 코드 편집하기

- 코드 편집기 준비하기
  - 시스템에서 제공하는 ‘메모장’
  - ‘비주얼 스튜디오 코드(Visual Studio Code)’ <https://code.visualstudio.com/>,
  - ‘서브라임 텍스트(Sublime Text)’ <https://www.sublimetext.com/3> 등
- 코드 작성하기
  - 메모장 실행 – 코드 작성하기 – [파일] – [다른 이름으로 저장]
  - 저장 옵션
    - [파일 형식] – 모든 파일, 반드시 .html, .htm 확장자 명시
    - 코드 내 한글 사용 할 경우, [인코딩] – UTF-8 설정

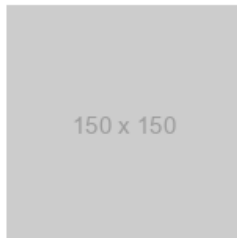
# 메모장으로는 못 만드나

- 당연히 만들 수 있다
  - HTML 문서는 단순히 텍스트 문서

## Welcome to My Page!

This is a simple HTML example.

Here's a link to [Google](https://www.google.com).



```
0101_htmlTest.html
파일 편집 보기

<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>My First Web Page</title>
  <style>
    body { font-family: Arial, sans-serif; text-align: center; }
    h1 { color: #4CAF50; }
    p { color: #555; }
  </style>
</head>
<body>
  <h1>Welcome to My Page!</h1>
  <p>This is a simple HTML example.</p>
  <p>Here's a link to <a href="https://www.google.com" target="_blank">Google</a>.</p>
  
</body>
</html>
```



# 웹브라우저에서 실행하기

- 직접 실행 : 드래깅 또는 더블클릭
  - HTML 파일을 드래깅하여 바탕화면 웹브라우저 아이콘 또는 실행중인 웹브라우저 화면으로 드래깅
  - HTML 파일 직접 더블클릭
- 웹브라우저 기능을 사용해 실행
  - 주소창에 파일 경로와 이름 직접 입력
  - 웹브라우저의 [파일열기] 메뉴 이용(웹브라우저마다 지원 상황 확인)

