

Introduction to Web Programming

Week01. 웹프로그래밍 이란?

강사 소개

- ✧ 담당강사 : 박재석
 - ✧ NAVER (지도지역 Cell)
 - ✧ withkali@gmail.com
 - ✧ <https://www.facebook.com/jaeseok.park.3>

강의계획

- ✧ 강의계획서 (<https://github.com/withkali/webprogramming/wiki/웹프로그래밍-개론-2015-1학기>)
- ✧ **Module 1.** HTML & CSS
- ✧ **Module 2.** Ruby Programming
- ✧ **Module 3.** Ruby on Rails
- ✧ **Module 4.** Javascript
- ✧ **Module 5.** Team Project

✧ **Module 1. HTML & CSS**

- ✧ 웹프로그래밍에 대한 이해
- ✧ Static WebPage 만들기
- ✧ CSS를 이용한 스타일링과 레이아웃
- ✧ More...
 - ✧ bootstrap (<http://getbootstrap.com>)
 - ✧ material css (<http://callemall.github.io/material-ui/#/>)
 - ✧ HTML5

✧ **Module 2. Ruby Programming**

- ✧ Ruby Language
- ✧ OOP (Object Oriented Programming)
- ✧ Ruby를 이용해서 간단한 어플리케이션 만들기

✧ **Module 3. Ruby on Rails**

- ✧ Rails 구축 및 웹 서버 프로그래밍
- ✧ MVC
- ✧ Database (mysql)
- ✧ Rails를 이용하여 동적인 WebPage 제작 및 Simple 웹서비스 제작

✧ **Module 4. Javascript**

- ✧ Javascript Language
- ✧ Javascript를 이용한 Client-Side Scripting
- ✧ More...
 - ✧ jquery (<http://jquery.com>)
 - ✧ angularJS (<https://www.angularjs.org>)

✧ **Module 5. Team Project**

- ✧ 만들어 보고자 하는 웹서비스
- ✧ 주제는 상관없다!!
- ✧ 3번의 Presentation

평가

- ✦ 출석(10) + 퀴즈(10) + 개인과제(40) + 팀 프로젝트(40)

웹프로그래밍 이란?



인터넷

- ✧ 1969년 미 국방성이 ARPANET을 개발 (연구용)
- ✧ MILNET으로 발전 (군사용)
 - ✧ 인류 재앙에서도 살아 남을 수 있는 네트워크가 목적

인터넷

- ✧ TCP/IP 프로토콜 개발
- ✧ Ethernet과 TCP/IP 기반의 LAN환경의 구축
 - ✧ 1980년 중반, NSF의 NSFNET 구축 (연구정보 공유 목적의 네트워크)
 - ✧ 많은 연구소/기관/대학등이 참여함으로써, 네트워크의 범위가 확대

인터넷

- ✦ 크고 작은 네트워크의 집합체
- ✦ 전체 네트워크의 주인은 없지만, IP Address의 관리 및 네트워크 망의 연결 서비스 제공을 위한 ISP(Internet Service Provider)가 필요

인터넷

- ✧ 파일전송 : FTP
- ✧ 전자우편 : Email
- ✧ Remote Access : Telnet

...

인터넷

- ✧ Web의 등장
- ✧ www는 인터넷 서비스의 일종
- ✧ www가 인터넷의 또다른 대명사로 올라서게 됨.

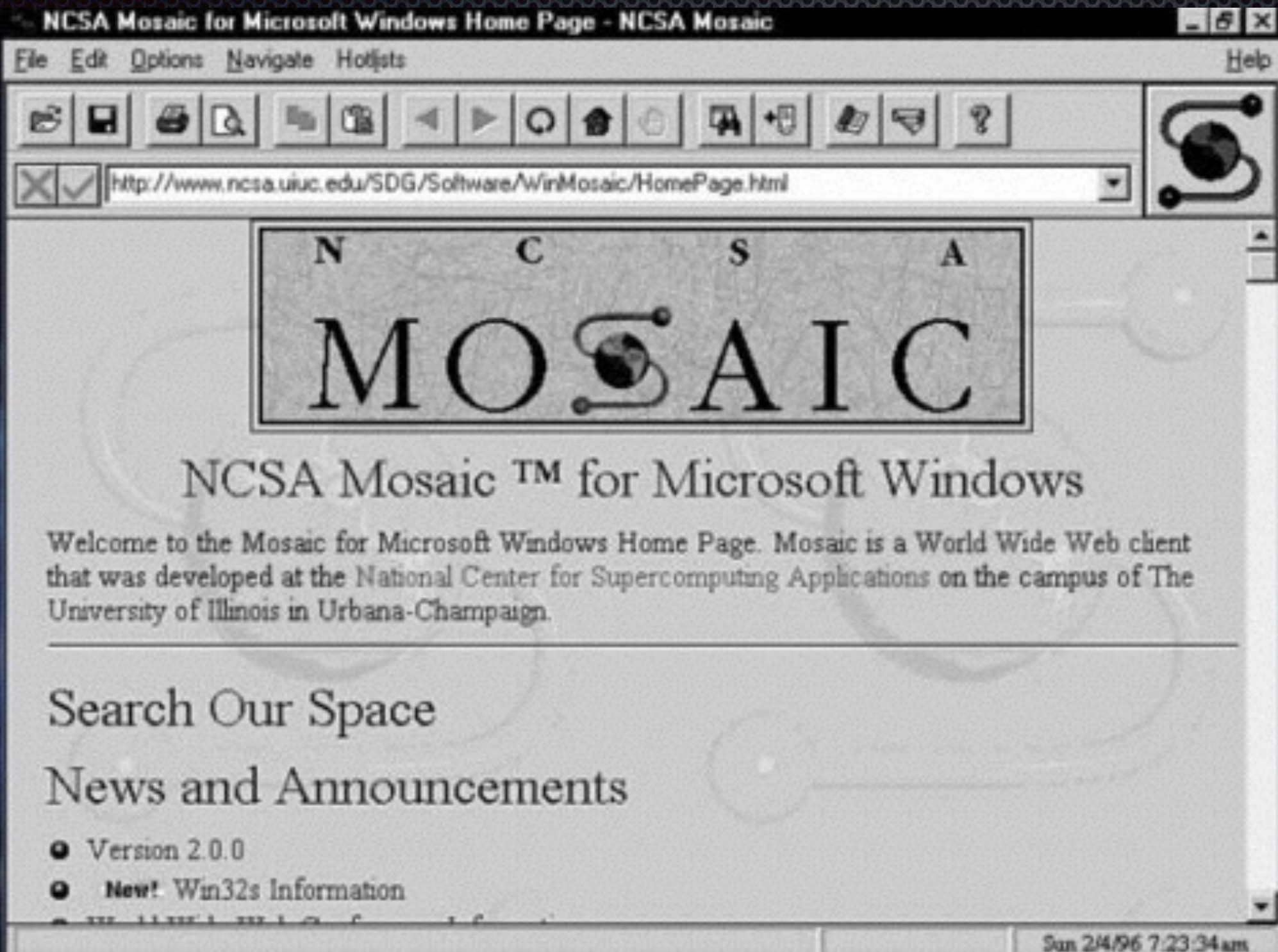


WEB

- ✧ www (World Wide Web)
- ✧ www의 등장으로 인터넷은 초고속 확산
- ✧ 1989년 스위스 CERN 연구소에서 개발
 - ✧ 인터넷을 이용한 최초의 Hyper Media System

WEB

- ✦ 1993년 Mosaic의 등장
- ✦ GUI방식의 웹 브라우저



WEB

- ✧ 1994년 웹 컨소시엄에서 웹 표준 기술 개발 (<http://www.w3c.org>)
- ✧ Netscape 와 Explorer의 등장
- ✧ 표준 경쟁

WEB

- 웹브라우저 전쟁

- Internet Explorer의 승리
- 표준을 지키지 않거나, 포괄적 적용으로 인한 문제
- 최근 Safari, Chrome등의 대안 브라우저들의 성장

WEB

✧ Hypertext

- ✧ 텍스트들이 링크를 통해 서로 연결된다.
- ✧ 정보의 조각들이 비순차적이다.

✧ Hypermedia

- ✧ 텍스트 + 이미지, 그래픽, 사운드, 멀티미디어등이 링크를 통해 서로 연결된다.

WEB

- ✦ 1965년 Ted Nelson이 제안한 Xanadu 시스템에에서 처음으로 Hypertext 언급
 - ✦ 세상의 모든 문서를 Hypertext로 연결하고자 함
- ✦ 1977년 MIT의 Andrew Lippman이 Aspen Movie Map이라는 Hypermedia를 공개
- ✦ 이후 Apple의 HeyperCard 및 Hypermedia를 이용한 레퍼런스들이 등장

WEB

- ✧ Hypermedia의 개념을 기반으로 인터넷상에서 정보를 교환
 - ✧ www(World Wide Web)
- ✧ HTTP(Hyper Text Transfer Protocol)를 프로토콜로 사용
- ✧ HTML(Hyper Text Markup Language)기반의 문서를 이용
- ✧ URL(Unirom Resource Locator)라는 표준 주소 표기 방식 사용

WEB

✧ HTTP

- ✧ www상에서 정보를 주고받는 프로토콜
- ✧ Stateless한 방식으로 요청/응답구조를 이루고 있다.
- ✧ 80포트(Default)를 사용
- ✧ 서비스 제공 시, 방화벽으로 인한 문제를 해결

WEB

✧ URL

- ✧ 네트워크 상에서 자원이 어디 있는지를 알려주는 규약
- ✧ 웹 뿐만이 아니라 기타 네트워크에서도 사용되며, 최근 프로그래밍 기술들에서는 리소스의 접근을 URL기반으로 하고 있다.
- ✧ protocol + address + path로 구성
 - ✧ ex) <http://www.naver.com/>
 - ✧ ex) <http://news.naver.com/main/main.nhn>

WEB

- ✧ Domain Name
- ✧ DNS (Domain Name System)
 - ✧ 읽고, 기억하기 편한 domain name과 매칭되는 ip address를 제공

WEB Programming

- ✧ 웹 브라우저

- ✧ 텍스트, 마커업, 이미지, 사운드, 멀티미디어 등을 GUI환경에서 보여준다.

WEB Programming

✧ HTML

- ✧ HyperText Markup Language commonly referred to as HTML is the standard markup language used to create web pages. It is written in the form of HTML elements consisting of tags enclosed in angle brackets (like <html>) (<http://en.wikipedia.org/wiki/HTML>)
- ✧ 웹 브라우저는 HTML을 읽고, Hypermedia들과 함께 구성되어, Visible, audible한 웹 페이지를 display한다.
- ✧ 웹 브라우저는 HTML을 interpret한다.

WEB Programming

- ✧ CSS (Cacading Style Sheets)
 - ✧ HTML로 구성된 페이지에 스타일을 입히고, layout을 구성한다.
 - ✧ CSS만을 수정해도, 같은 HTML을 다르게 표현할 수 있다.

WEB Programming

✧ Javascript

- ✧ Static한 HTML 페이지에 이벤트 기반의 동적인 Feature를 추가할 수 있다.
- ✧ 최근 javascript는 서버사이드 스크립트로도 활용된다.
- ✧ javascript를 좀 더 쉽게 사용할 수 있도록 JQuery, AngularJS등의 편리한 Javascript framework library등이 개발되어 있다.

WEB Programming

- ✧ 내 웹페이지를 모두가 볼 수 있게 하려면?
 - ✧ 특정 Domain Name에 매핑된 ip address를 가진 서버가 필요
- ✧ Web Server
 - ✧ Apache, IIS, nginx, GWS
 - ✧ 웹서버는 클라이언트(브라우저 등)로 부터 오는 HTTP 요청에 따라, 적절한 페이지를 응답으로 리턴한다.

WEB Programming

- 상황에 따라 내 웹사이트의 Contents를 동적으로 변경하고 싶으면?
 - Web Application
- Web Application
 - 웹 서버는 웹페이지를 응답하기 위해서, Server-Side Script 또는 Web Application으로 HTTP 요청을 전달한다. (어디로?)
 - Ruby on Rails, NodeJS, ASP.NET, JAVA Beans, Spring etc.
 - 각각의 웹 어플리케이션 프레임워크는 동적인 응답을 구성하고, 필요에 따라 서버에 저장된 정보를 관리하기 위한 방법을 제공한다.

WEB Programming

- ✧ 서버는 정보를 어디에 저장하지?
 - ✧ 파일, DB(Database), 메모리
- ✧ Database
 - ✧ 서버가 관리하는 정보를 체계적이고, 효율적으로 관리하기 위한 도구
 - ✧ R-DB, Column DB, Graph DB etc

WEB Programming을 한다

- ✦ Markup Language(ex. HTML)를 이용해, 브라우저에서 보여줄 페이지를 작성한다.
- ✦ 브라우저에서 보여지는 페이지의 Styling과 Layout을 위해 Stylesheet언어(ex. CSS)로 페이지를 재가공한다.
- ✦ 웹 페이지를 공개하기 위해 웹서버를 세팅한다.

WEB Programming을 한다

- 동적인 웹페이지를 구성하기 위해, Server Framework(ex. Ruby on Rails)을 이용하여, 컨텐츠와 정보를 관리하고, 적절한 웹 페이지를 동적으로 생성한다.
- 서버의 정보와 컨텐츠를 관리하기 위해, Database(ex. MySql)를 이용한다.

WEB Programming을 한다

- 우리가 이 강의에서 배워야 할 것은?

- HTML + CSS
- Javascript, JQuery
- Ruby
- Ruby on Rails
- MySql
- AWS
- Github

WEB Programming을 한다

- ✧ 목표는 하나!!
- ✧ 기본적인 지식만으로도 자신의 웹페이지 또는 심플한 웹서비스를 구축할 수 있게 하는 것.

첫번째 과제

- ✦ codecademy 수강 (HTML + CSS)
 - ✦ <http://www.codecademy.com/en/tracks/web>
 - ✦ 강좌 수강 증명 제출 Due to 3/25