



Web Engineering (SEEC-304)

Topic: Introduction to HTML5

Course Instructor: Dr. Ahmad Mustafa

Department of Software Engineering
University of Sargodha



What This Course Will

- **The BASICS**
 - **Syntax and Semantics**
 - **Accessibility**
 - **Getting Started**
 - **Right down to How to Create a File**



Agenda

- **Focus is on questions:**

What happens when you type an address into a URL?

- **What types of tools you need to code (editors and browsers)?**
- **What is HTML5?**



Final Project

- **Create a syntactically valid website**
- **Must pass W3C validation**

**Must pass WAVE accessibility
review**



Review

- **This class is for beginners**
- **You will leave with the ability to write and understand HTML5 code – not as a web developer**

You will understand the importance of accessibility in technology



The Evolution of HTML

**What it is... and why we aren't
starting at HTML 1.0**



What is HTML?

- **HTML stands for Hypertext Markup Language**
- **Markup languages use tags to annotate (or “mark-up”) documents.**
- **In HTML the tags indicate where headings `<h1>`, images ``, list items ``, links `<a>`, line breaks `
`, and other components should be displayed.**



.html Files

- **When your computer opens a .html file, it knows to open it in an Internet browser (Chrome, Firefox, Safari, etc.)**
- **The browser can read this file and know how to display it on the screen.**
- **Screen readers and other assistive devices can also utilize the HTML tags to present the information in special ways.**



HTML Files

- **HTML is similar to English, so you can understand it even if you don't know much about it.**



“Learning” HTML

- **In the beginning you worry about syntax**
 - **What tags are there?**
 - **Did I remember to “end” my tag?**
- **Later, you will worry about semantics**
 - **Is there a tag that better conveys the meaning I am trying to get across?**
 - **If someone is searching my page can they find what they need and access it easily?**



Early Years

- **HTML (I) was created in 1990 as a way to electronically connect documents via hyperlinks
(hence a “web” of connections)**





Early Years

It is required that HTML be a common language between all platforms. This implies no device-specific markup, or anything which requires control over fonts or colors, for example.



Mosaic

- **In 1993, Mosaic emerged as the first graphical browser.**
- **WWW proliferates at a 341,634% annual growth rate of service traffic**
- **Mosaic had challengers though in the form of Netscape (1994), Internet Explorer (1995) and others.**



“Images caused a lot of angst among

- the early web community because
- we just went and decided this was
- a cool thing and decided to put
- them in..... We're humans.
- That's more interesting to look at
- than a page of text. – Jon
- Mittelhauser,



The Browser Wars

- **Browsers had proprietary tags**
 - `<marquee>...</marquee>` (scrolling text)
 - `<blink>...</blink>` (blinking text).
- **Other tags that went against the spirit of the original tenets of HTML were added, e.g. ``, `<center>`, and `<bgcolor>`**
- **Origination of “Best viewed on” messages.**



Web Standards

- **No one “runs” the Internet or the Web, some groups do take proactive roles:**
 - **Internet Engineering Task Force (IETF)**
 - **World Wide Web Consortium (W3C)**
 - **The Web Accessibility Initiative (WAI)**



Internet Engineering Task Force (IETF)

- The Internet Engineering Task Force (IETF) is an international standards organization responsible for developing and promoting voluntary Internet standards.
- Founded in 1986 and Operates under the Internet Society (ISOC).
- **Primary Role**
 - Develops and maintains core Internet protocols.
 - Ensures interoperability between different systems and networks.
 - Focuses on technical standards that make the Internet function properly.
- **Key Standards Developed**
 - HTTP (Hypertext Transfer Protocol)
 - TCP/IP (Transmission Control Protocol / Internet Protocol)
 - SMTP (Simple Mail Transfer Protocol)
 - DNS (Domain Name System)
 - FTP (File Transfer Protocol)



World Wide Web Consortium (W3C)

Introduction

- The **World Wide Web Consortium (W3C)** was founded in 1994 by Tim Berners-Lee.
- It develops standards for the World Wide Web.
- Headquartered at MIT, ERCIM, Keio University, and Beihang University.

Primary Role

- Standardizes web technologies.
- Ensures long-term growth of the Web.
- Promotes compatibility across browsers and platforms.

Key Standards Developed

- HTML (HyperText Markup Language)
- CSS (Cascading Style Sheets)
- XML (Extensible Markup Language)
- SVG (Scalable Vector Graphics)
- Web APIs



Web Accessibility Initiative (WAI)

Introduction

- The **Web Accessibility Initiative (WAI)** is a part of W3C.
- Focuses on making the web accessible to people with disabilities.
- Promotes inclusive web design.

Primary Objective

- To ensure that people with:
- Visual impairments
- Hearing disabilities
- Motor limitations
- Cognitive disabilities
can access and use web content effectively.

Key Guidelines

- WCAG (Web Content Accessibility Guidelines)
- ATAG (Authoring Tool Accessibility Guidelines)
- UAAG (User Agent Accessibility Guidelines)



Evolution of Browsers

1990 – 1994	HTML was simple, content was primarily text-based
1993	Mosaic enters the scene with images and ... BOOM!!!
1995 – 1999	Cross-browser compatibility falls apart
2000 – 2005	Browsers move toward separating content from style.
2005 – 2008	Using HTML files in coordination with CSS becomes new standard.



Evolution of HTML

1993	HTML 1.0 - Developed by Tim Berners-Lee to link document
1995	HTML 2.0 - Developed by Internet Engineering Task Force RFC to include stylized text and tables
1996	CSS 1
1997	HTML 3.2 – Developed by W3C and included browser specific features
1997	HTML 4.0 – A move back to normalizing the pages across platforms.
1998	CSS 2
1999	HTML 4.01 – Introduced different document types
2012	HTML 5 - Back to HTML plus multimedia and semantic tags



Where we are now

- **HTML5 is a cooperation between W3C and the Web Hypertext Application Technology Working Group(WHATWG)**
- **Established Guidelines**
 - **New features should be based on HTML, CSS, the DOM, and JavaScript**
 - **Reduce the need for external plugins (e.g. Flash)**
 - **More markup to replace scripting**
 - **HTML5 should be device independent**



The Request/Response Cycle

Or, what happens when you type something into the address bar



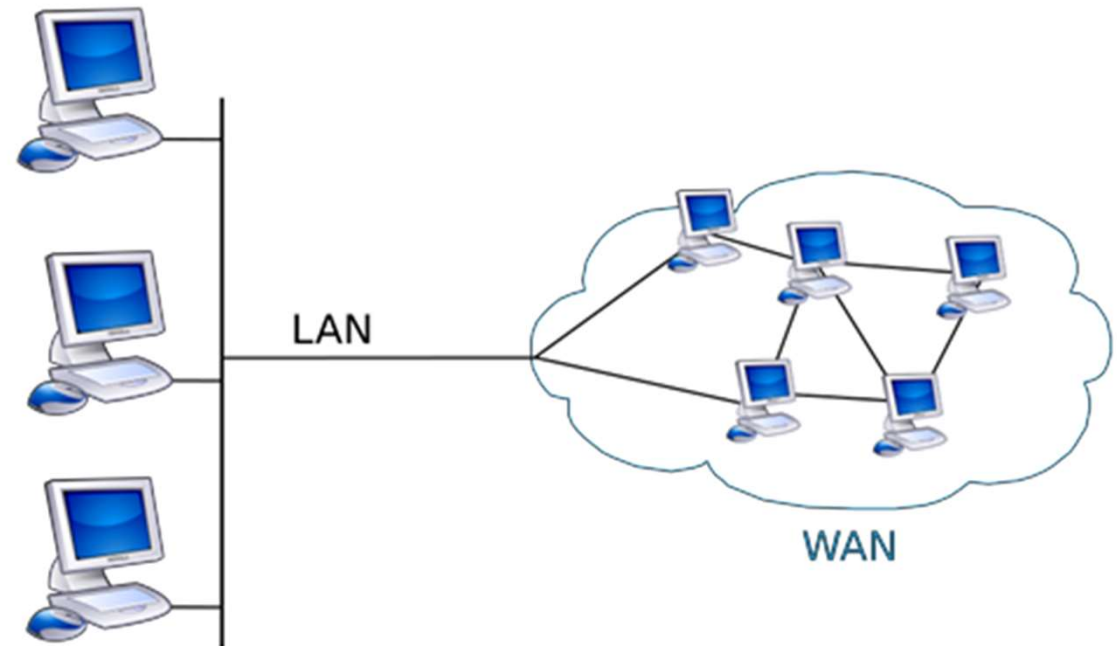
How Does This All Work?

<http://uos.edu.pk>

- **When you type an address into the URL bar, what happens?**
- **Warning: This lecture is heavy on the acronyms.**

Networks

- **The Internet**
 - **LAN**
 - **WAN**





Client/Server Relationship

- **Servers**
 - **Machines that hold shared resources**
 - **Always connected to the network**
- **Clients**
 - **Machines for personal use (laptops, phones, etc.)**



Request/Response Cycle

- **This is what happens when your computer (the client) requests a page and a server responds with the appropriate files.**



Uniform Resource Locator

- **URL – three parts:**
 - **protocol – how to connect**
 - **domain – the server***
 - **(optional) document – the specific file needed**
 - **Most pages are made up of multiple files**



Protocols

- **HTTP – Hypertext Transfer Protocol**
- **HTTPS – Secure Hypertext Transfer Protocol**
- **FTP – File Transfer Protocol**



Domain Names

- **Identifies the entity you want to connect to**
 - **umich.edu, google.com, wikipedia.org**
- **Each has different top-level domain**
 - **Determined by Internet Corporation for Assigned Names and Numbers (ICANN)**
 - **<https://www.icann.org/resources/pages/tlds-2012-02-25-en>**



IP Addresses and the Domain Name Server (DNS)

- Internet Protocol Version 6 (IPv6) is the communication protocol that identifies computers on networks.
- Every computer has a unique IP address
xxxx:xxxx:xxxx:xxxx:xxxx:xxxx:xxxx:xxxx
- where x can have 16 different values.
- Can represent over 300 trillion unique combinations (2^{128})!
- DNS looks up the domain and returns the IP address



The Domain Name Server (DNS)

- **Luckily, you don't need to remember the IP address of a domain.**
- **The DNS will lookup the IP address based on the URL you type in.**



Document

- **URLs can specify a specific document**
 - <http://www.intro-webdesign.com/contact.html>
 - <http://www.intro-webdesign.com/Ashtabula/harbor.html>
- **If no document is specified, the default document is returned.**
 - **Convention is index.html**



The Request

- **Once the IP address is determined, the browser creates an HTTP request.**
- **Lots of hidden information in this request**
 - **header, cookies, form data, etc**



The Response

- **The server returns files, not “web pages”**
 - **It is up to the browser to decide what to do with those files**
- **If the server can’t fulfill the request it will send back files with error codes: 404, 500, etc.**

How it Works

What happens when you type “http://su.edu.pk/” into the address bar.

1. The browser look up the domain in the DNS

2. The DNS returns the IP address: 54.88.175.189

The Request/ Response
Cycle is initiated

3. The browser sends an HTTP request to the server located at that address.

4. The server finds the requested file and sends it back as a response.

5. The browser takes the response and renders the HTML code as a nice graphical presentation, often repeating steps 3 – 4 as needed to request images and other supporting files.





Differing Browsers

- **Different browsers have their pros and cons**
- **Most people have a preferred browser**
- **You need to test your site on multiple browsers.**



Internet Explorer

- **For a long time, one of the most popular browsers because it was preinstalled on Microsoft Windows.**
- **Platform-dependent - doesn't automatically work on a Mac.**



Microsoft Edge

- **In 2015 the new Windows 10 operating system included Microsoft Edge.**
- **Edge is meant to replace IE.**

Safari

- **The default browser on Mac devices is Safari.**
- **It also work on Windows.**
- **Free to download**





Google Chrome

- **Freeware browser developed by Google**
- **First released in 2008, for Microsoft Windows, it was later ported to Linux, macOS, iOS and Android**
- **Greater security**



Firefox

- **Free and open-source browser developed by Mozilla.**
- **Available for Windows, macOS, Linux, and BSD operating systems.**

Overview & Usage Statistics (2026)

Global Browser Market Share (All Platforms)

- **Google Chrome** — ~71.3% of global browser usage (leader worldwide)
- **Safari** — ~14.8% (especially strong on Apple devices)
- **Microsoft Edge** — ~4.6% (Windows integration helps adoption)
- **Firefox** — ~2.2% (niche & privacy-oriented)
- **Internet Explorer** — very low usage (<1%, largely retired)

These figures reflect broad trends in 2025–2026, showing Chrome's clear dominance and smaller shares for other browsers.



Issues of Accessibility

- <http://www.html5accessibility.com/> keeps a review of the accessibility of browsers.
- **Browsers should have keyboard functionality, support HTML5 tags, and support features for assistive technology.**

Accessibility Issues & Cross-Browser Challenges



Why Accessibility Matters

- Web accessibility ensures users with sensory, motor, or cognitive disabilities can use web content.
- It relies on standards like **WCAG** and assistive tech (screen readers, magnification).

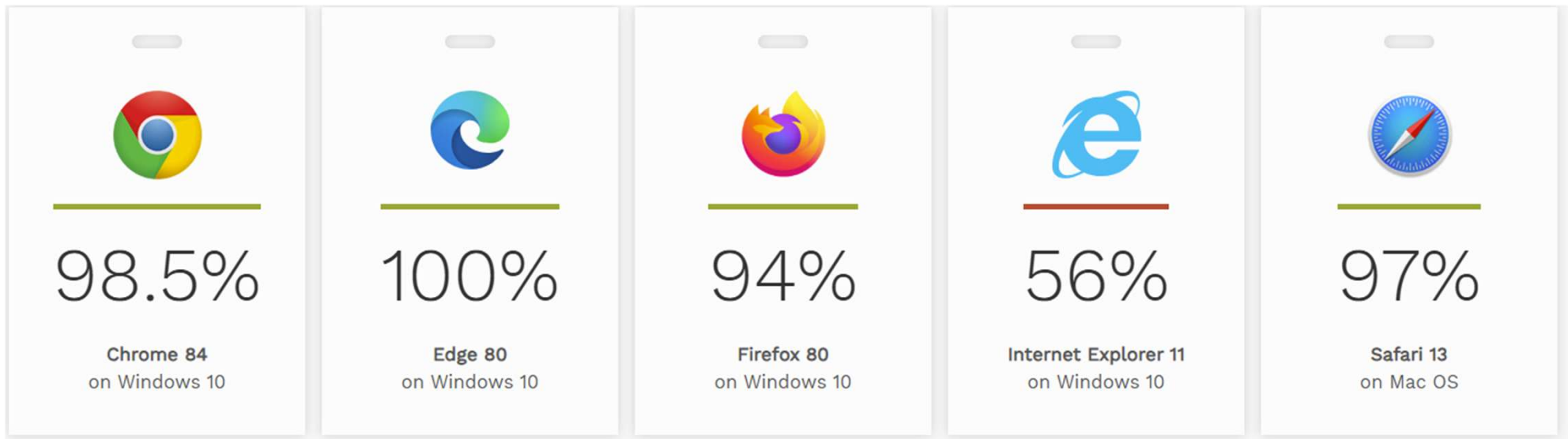
Common Accessibility Issues Across Browsers

- **Inconsistent keyboard focus indicators:**
 - Some browsers handle focus styles differently — e.g., Safari/Firefox may not show visible focus feedback without explicit CSS, while Chrome tends to.
- **ARIA role interpretation differences:**
 - ARIA attributes may work in one browser but behave inconsistently in others unless used carefully.
- **Screen reader behavior variations:**
 - VoiceOver, NVDA, and JAWS interact differently across browsers due to browser rendering and API support.
- **Low contrast & missing labels:**
 - Accessibility errors like poor contrast or missing form labels create barriers regardless of the browser used

**** **ARIA (Accessible Rich Internet Applications) attributes are HTML attributes that improve web accessibility for users with disabilities, particularly those using screen readers.**



August 2020





Editors: How to Use an Editor to Create an HTML File

Time to write some code



Creating and Editing Your Files

- 1. Decide how you will organize your files**
- 2. Decide on a naming convention**
 - **dash-names, CamelCase**
 - **No spaces, Consistent capitalization**
- 3. Decide on an editor**
 - **Windows (Notepad, Notepad++, Sublime, VS Code*)**
 - **Mac (TextEdit, TextWrangler, Sublime, VS Code*)**



Getting Started

- 1. Open your editor**
- 2. Select Save or Save As and name your file. You may need to create a new folder first**
- 3. Add Doctype, head, and body tags**
- 4. Save File (Ctrl-S or Command-S)**
- 5. Open in browser**



Troubleshooting

- **My file opens in an editor instead of a browser.**
 - Right click and select “Open With..”
- **My browser shows my tags**
 - Check that file extension is .html



Troubleshooting

- **I changed my code, but my page looks the same.**
 - **Refresh your browser**
 - **Verify file name**
- **I get “weird” characters.**
 - **Try typing code in by hand, not copy-and-paste**

How to use an Editor to Create an HTML File

Examples

- **TextEdit**
- **Sublime**
- **VS Code**
- **Replit**





Editors: How to Use Replit

**Writing code that you can share
online**



Online editors

1. **Google Docs and Microsoft 365 are common ways share documents.**
 2. **Sharing code is a little different - especially when you**
 3. **want to share your webpage with someone.**
- Today we will use Replit**

to



IDEs

- 1. IDEs are Integrated Development Environments, fancy term for software that lets you write your code, run your code, share your code, .etc**
- 2. Replit is an IDE for beginner programmers.**
- 3. In this class I will switch between using Replit and Sublime and Visual Studio Code**

How to use Replit

Using Replit





Which editor should you use?

- If you want to work “locally” (no internet connection) then your built-in editor, Sublime, VSCode are a good choice. or
- If you have internet access and want to share your code, Replit may be a better choice.



Thank you