INTRODUCTION TO WEB DEVELOPMENT

Lesson 1

OVERVIEW

Web development encompasses the process of creating, building, and maintaining websites or web applications that are accessible via the internet.

It involves a combination of different disciplines such as programming, design, and infrastructure management.

FRONTEND DEVELOPMENT

- **HTML (Hypertext Markup Language)**: The standard markup language for creating web pages. It defines the structure and content of web pages.
- **CSS (Cascading Style Sheets)**: Used to style the HTML elements, defining their appearance, layout, and design.
- **JavaScript**: A dynamic scripting language used for adding interactivity and functionality to web pages. It enables features like animations, form validation, and dynamic content loading.
- **Frameworks and Libraries**: Tools like React.js, AngularJS, Vue.js provide efficient ways to build complex user interfaces and manage frontend components.

BACKEND DEVELOPMENT

- **Server-side Languages**: Such as Python (with frameworks like Django or Flask), JavaScript (Node.js), Ruby (Ruby on Rails), PHP, and others. These languages handle server-side logic and interact with databases.
- **Databases**: Systems like MySQL, PostgreSQL, MongoDB, or Firebase are used to store and manage data for web applications.
- **APIs (Application Programming Interfaces)**: Backend development involves creating APIs to allow communication between the frontend and backend systems. RESTful APIs are common for web applications.

FULL-STACK DEVELOPMENT

- Full-stack developers are proficient in both frontend and backend technologies, enabling them to handle all aspects of web development.
- They're capable of building entire web applications from scratch, including both client-side and server-side components.

WEB DESIGN

- Design plays a crucial role in user experience (**UX**) and user interface (**UI**) design. It involves creating visually appealing layouts, typography, color schemes, and navigation elements.
- Tools like Adobe XD, Sketch, Figma, or InVision are commonly used for designing web interfaces.

RESPONSIVE DESIGN

- With the proliferation of devices with different screen sizes and resolutions, responsive design ensures that websites adapt and provide optimal user experience across various devices (desktops, laptops, tablets, smartphones).

WEB PERFORMANCE OPTIMIZATION

- Techniques such as minification, compression, caching, and lazy loading are employed to improve website performance and loading times.
- Optimization is crucial for user experience and search engine ranking.

SECURITY

- Web developers need to consider security measures to protect websites from common threats like SQL injection, cross-site scripting (XSS), and cross-site request forgery (CSRF).
- Practices like input validation, secure authentication, and HTTPS encryption help enhance security.

VERSION CONTROL

- Version control systems like Git are essential for managing codebase changes, collaboration among developers, and maintaining code integrity.

CONTINUOUS INTEGRATION/CONTINUOUS DEPLOYMENT (CI/CD)

- CI/CD pipelines automate the process of testing, building, and deploying code changes, ensuring reliability and efficiency in the development workflow.

WEB HOSTING AND DEPLOYMENT

- Websites and web applications need to be hosted on servers to be accessible over the internet. Various hosting options include shared hosting, virtual private servers (VPS), cloud hosting (e.g., AWS, Google Cloud, Azure), and platform-as-a-service (PaaS) solutions.

ACCESSIBILITY

- Developers should ensure that websites are accessible to users with disabilities, adhering to web accessibility standards such as WCAG (Web Content Accessibility Guidelines).

SEARCH ENGINE OPTIMIZATION (SEO)

- Techniques like optimizing metadata, using descriptive URLs, creating high-quality content, and improving site speed contribute to better visibility and ranking in search engine results.

SEARCH ENGINE OPTIMIZATION (SEO)

- Techniques like optimizing metadata, using descriptive URLs, creating high-quality content, and improving site speed contribute to better visibility and ranking in search engine results.

INTRODUCTION TO HTML

HTML is the standard markup language for creating web pages.

WHAT IS HTML?

- HTML stands for Hyper Text Markup Language
- HTML is the standard markup language for creating Web pages
- HTML describes the structure of a Web page
- HTML consists of a series of elements
- HTML elements tell the browser how to display the content
- HTML elements label pieces of content such as "this is a heading", "this is a paragraph", "this is a link", etc.

A SIMPLE HTML DOCUMENT

```
<!DOCTYPE html>
<html>
<head>
<title>Page Title</title>
</head>
<body>

<h1>My First Heading</h1>
<p>My first paragraph.</p>

</body>
</html>
```

EXAMPLE EXPLAINED

- The <!DOCTYPE html> declaration defines that this document is an HTML5 document
- The <html> element is the root element of an HTML page
- The <head> element contains meta information about the HTML page

```
<!DOCTYPE html>
<html>
<head>
<title>Page Title</title>
</head>
<body>

<h1>My First Heading</h1>
<p>My first paragraph.</p>

</body>
</html>
```

EXAMPLE EXPLAINED

- The <title> element specifies a title for the HTML page (which is shown in the browser's title bar or in the page's tab)
- The <body> element defines the document's body, and is a container for all the visible contents, such as headings, paragraphs, images, hyperlinks, tables, lists, etc.

```
<!DOCTYPE html>
<html>
<head>
<title>Page Title</title>
</head>
<body>

<h1>My First Heading</h1>
<p>My first paragraph.</p>

</body>
</html>
```

EXAMPLE EXPLAINED

- The <h1> element defines a large heading
- The <p> element defines a paragraph

```
<!DOCTYPE html>
<html>
<head>
<title>Page Title</title>
</head>
<body>

<h1>My First Heading</h1>
<p>My first paragraph.</p>

</body>
</html>
```

WHAT IS AN HTML ELEMENT?

An HTML element is defined by a start tag, some content, and an end tag:

<tagname> Content goes here… </tagname>

WHAT IS AN HTML ELEMENT?

The HTML element is everything from the start tag to the end tag:

<h1>My First Heading</h1>

My first paragraph.

WHAT IS AN HTML ELEMENT?

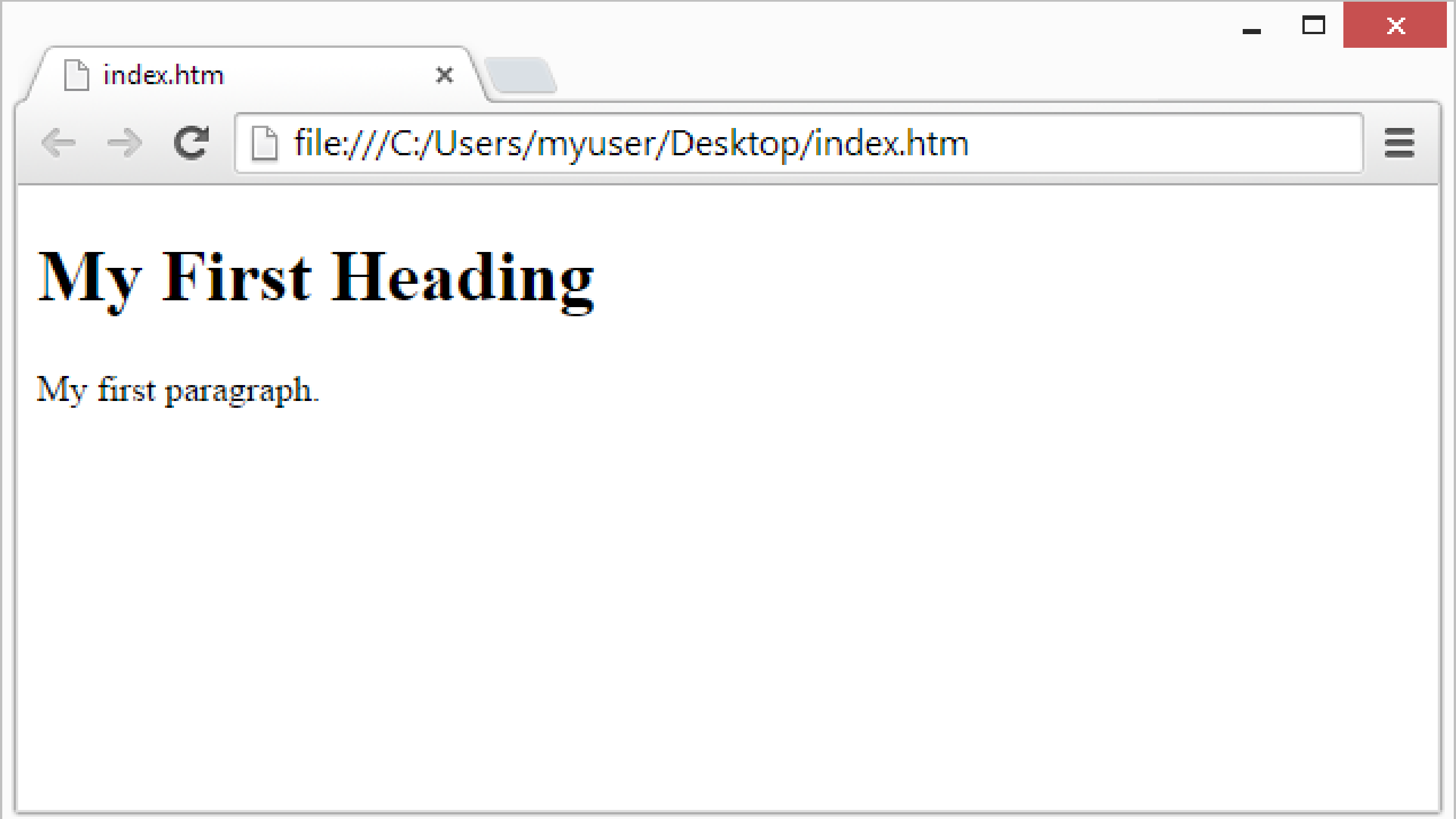
Start tag	Element content	End tag
<h1>	My First Heading	</h1>
	My first paragraph.	
 	none	*none*

Some HTML elements have no content (like the
 element). These elements are called empty elements. Empty elements do not have an end tag!

WEB BROWSER

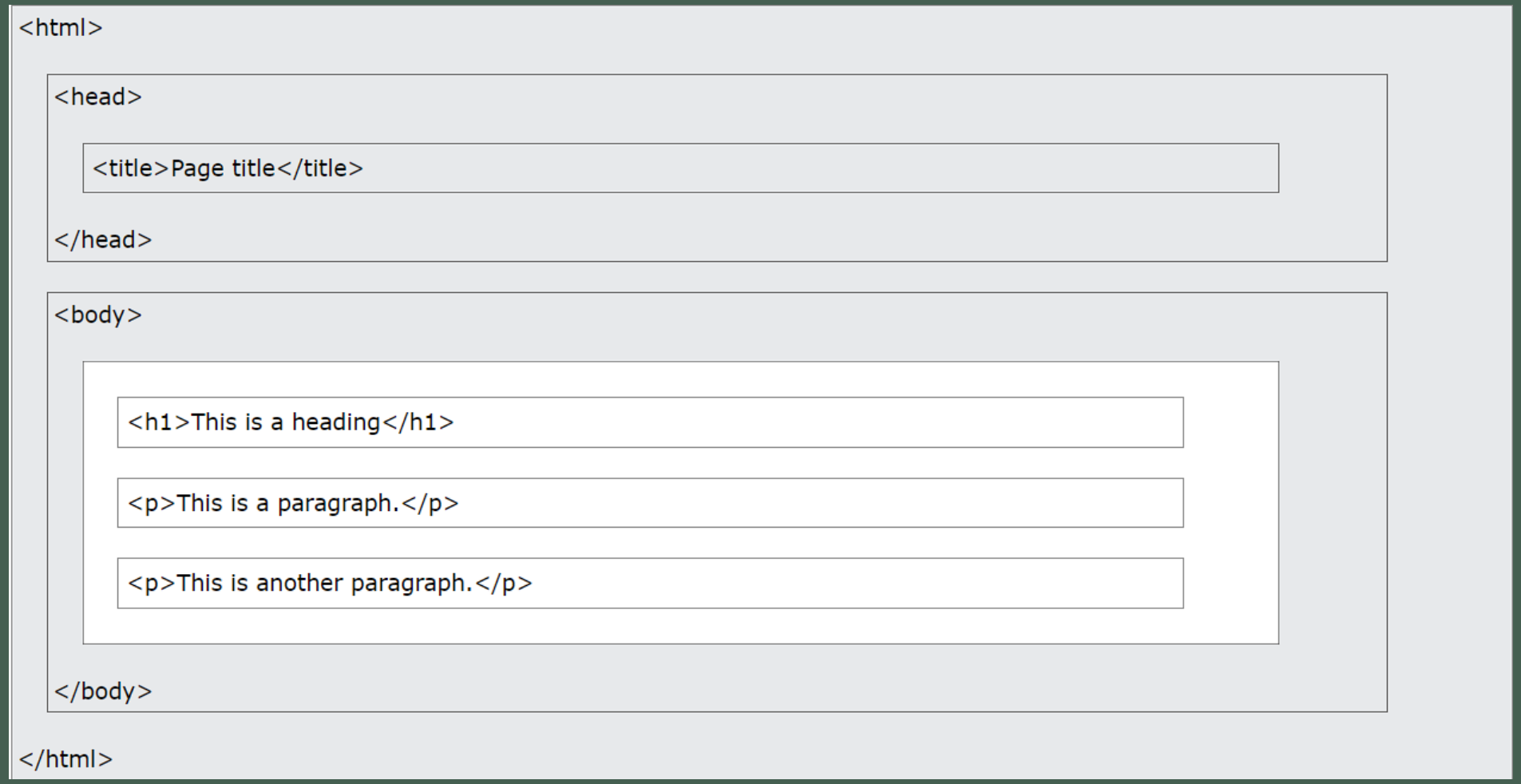
The purpose of a web browser (Chrome, Edge, Firefox, Safari) is to read HTML documents and display them correctly.

A browser does not display the HTML tags, but uses them to determine how to display the document:


index.htm
file:///C:/Users/myuser/Desktop/index.htm
My First Heading
My first paragraph.

HTML PAGE STRUCTURE

BELOW IS A VISUALIZATION OF AN HTML PAGE STRUCTURE:



HTML HISTORY

SINCE THE EARLY DAYS OF THE WORLD WIDE WEB, THERE HAVE BEEN MANY VERSIONS OF HTML:

Year	Version
1989	Tim Berners-Lee invented www
1991	Tim Berners-Lee invented HTML
1993	Dave Raggett drafted HTML+
1995	HTML Working Group defined HTML 2.0
1997	W3C Recommendation: HTML 3.2

SINCE THE EARLY DAYS OF THE WORLD WIDE WEB, THERE HAVE BEEN MANY VERSIONS OF HTML:

1999	W3C Recommendation: HTML 4.01
2000	W3C Recommendation: XHTML 1.0
2008	WHATWG HTML5 First Public Draft
2012	WHATWG HTML5 Living Standard
2014	W3C Recommendation: HTML5
1999	W3C Recommendation: HTML 4.01

SINCE THE EARLY DAYS OF THE WORLD WIDE WEB, THERE HAVE BEEN MANY VERSIONS OF HTML:

2016	W3C Candidate Recommendation: HTML 5.1
2017	W3C Recommendation: HTML5.1 2nd Edition
2017	W3C Recommendation: HTML5.2

HTML EDITORS

A simple text editor is all you need to learn HTML.

LEARN HTML USING NOTEPAD

Web pages can be created and modified by using professional HTML editors.

However, for learning HTML we recommend a simple text editor like Notepad (PC) or TextEdit (Mac).

We believe that using a simple text editor is a good way to learn HTML.

STEP 1: OPEN NOTEPAD (PC)

Windows 8 or later:

Open the Start Screen (the window symbol at the bottom left on your screen). Type Notepad.

Windows 7 or earlier:

Open Start > Programs > Accessories > Notepad

STEP 2: WRITE SOME HTML

```
<!DOCTYPE html>
<html>
<body>

<h1>My First Heading</h1>

<p>My first paragraph.</p>

</body>
</html>
```

STEP 3: SAVE THE HTML PAGE

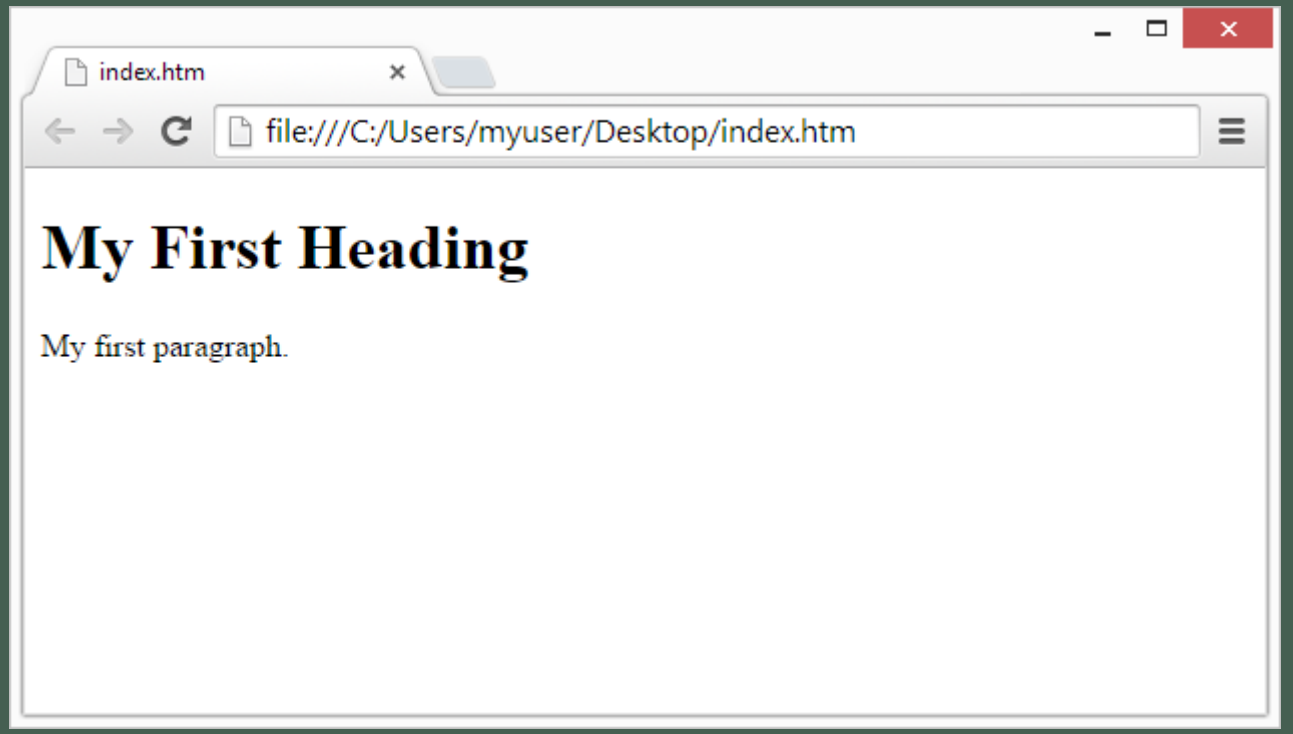
Save the file on your computer. Select File > Save as in the Notepad menu.

Name the file "index.htm" and set the encoding to UTF-8 (which is the preferred encoding for HTML files).

Tip: *You can use either .htm or .html as file extension. There is no difference; it is up to you.*

STEP 4: VIEW THE HTML PAGE IN YOUR BROWSER

Open the saved HTML file in your favorite browser (double click on the file, or right-click - and choose "Open with").



ACTIVITY

ACTIVITY 1

Create a page that contains the following (**5 points each**):

1. Page title "Activity 1 – You complete name".
2. Your complete name (use second heading)
3. Age
4. Complete Address
5. Motto in Life
6. Explain what you expect to learn in the subject.