

HTML

Fundamentals

Naveen Pete

Introduction to the Web

- URL
 - User is requesting something
 - Server is responding
- Request – Response
- When user types URL and presses Enter key in the browser
 - Finding the right place
 - Your query is submitted to your ISP
 - Within your ISP, the DNS takes the domain name and turns it into an IP address (unique identifier)

Introduction to the Web

- Going to that address
 - A request is sent to the desired IP address via HTTP
 - Your request finds the fastest possible path to the server with the specified IP
 - Not a direct journey, it requires hopping from server to server until we arrive
- Server responds
 - Server figures out exactly what we are asking for
 - Builds the right content, pulling information from data source (database)
 - Responds with a combination of HTML, CSS & JavaScript
- Browser takes the code and shows it to the user as a web page

Front End Basics

- Front End vs Back End
 - Front end is the stuff that you see and interact with
 - HTML, CSS and JavaScript
 - Back end constructs the HTML and CSS that is sent back
 - Python, PHP, JavaScript, Java, C#
 - MongoDB, SQL Server, MySQL, Oracle
- HTML
 - Stands for Hypertext Markup Language
 - Defines the content and structure of a web page
 - Put an image here
 - Put a form here
 - The **nouns** of a web page

Front End Basics

- CSS

- Cascading Style Sheets
- Defines the style of HTML
 - Make text blue
 - Give the image a border
- The **adjectives** of a web page

- JavaScript

- Adds logic and interactivity to a page
 - Do some math
 - Change color when the user clicks
 - Load data from Twitter
- The actions or **verbs** of a web page

HTML

- A markup language used to create web pages and web applications
- Is used for structuring and presenting content on the WWW
- Allowed publishing and exchanging of scientific and technical documents
- Allowed electronic linking of documents via hyperlinks
- Created in 1989 / 1990 by Tim Berners Lee
- Latest version is HTML5, published in 2014
- Includes markup and APIs for complex web applications

HTML5

- Represents two concepts
 1. New version of the language HTML, with new elements, attributes and behaviors
 2. A larger set of technologies for building more diverse and powerful web sites and applications
- HTML5 technologies can be classified into several groups
 - Semantics
 - Connectivity
 - Offline & Storage
 - Multimedia
 - Graphics
 - Performance & Integration
 - Device Access
 - Styling

HTML5

- Semantics: allowing you to describe more precisely what your content is
 - Sections & outlines in HTML5
 - New outlining and sectioning elements in HTML5: `<section>`, `<article>`, `<nav>`, `<header>`, `<footer>` and `<aside>`
 - Audio and video
 - The `<audio>` and `<video>` elements embed and allow the manipulation of new multimedia content
 - Forms improvements
 - Constraint validation API, several new attributes, new values for the `<input>` attribute type and the new `<output>` element
 - New semantic elements
 - Numerous new elements, like `<mark>`, `<figure>`, `<figcaption>`, `<data>`, `<time>`, `<output>`, `<progress>`, or `<meter>` and `<main>`

HTML5

- Connectivity: allowing you to communicate with the server in new and innovative ways
 - Web Sockets
 - Makes it possible to open an interactive communication session between the user's browser and a server. You can send messages to a server and receive event-driven responses without having to poll the server for a reply
 - Server Sent Events
 - Allows a server to push events to a client
 - WebRTC
 - Allows connecting to other people and controlling videoconferencing directly in the browser, without the need for a plugin or an external application
 - RTC stands for Real-Time Communication

HTML5

- Offline and storage: allowing webpages to store data on the client-side locally and operate offline more efficiently
 - Online & offline events
 - Let applications and extensions detect whether or not there's an active Internet connection, as well as to detect when the connection goes up and down
 - Web Storage API
 - Provides mechanisms by which browsers can store key/value pairs, in a much more intuitive fashion than using cookies
 - IndexedDB
 - A web standard for the storage of significant amounts of structured data in the browser and for high performance searches on this data using indexes

HTML5

- Multimedia: making video and audio first-class citizens in the Open Web
 - Using HTML5 audio and video
 - The `<audio>` and `<video>` elements embed and allow the manipulation of new multimedia content.
 - WebRTC
 - Same as in Connectivity section
 - Using the Camera API
 - Allows using, manipulating, and storing an image from the computer's camera.

HTML5

- 2D/3D graphics and effects: allowing a much more diverse range of presentation options
 - `<canvas>`
 - An HTML element which can be used to draw graphics using scripting
 - WebGL (Web Graphics Library)
 - A JavaScript API for rendering interactive 3D and 2D graphics within any compatible web browser without the use of plug-ins
 - SVG
 - An XML-based format of vectorial images that can directly be embedded in the HTML

HTML5

- Performance and integration: providing greater speed optimization and better usage of computer hardware
 - Web Workers
 - A simple means for web content to run scripts in background threads, without interfering with the user interface.
 - XMLHttpRequest
 - Allows fetching asynchronously some parts of the page, allowing it to display dynamic content, varying according to the time and user actions
 - Drag and drop
 - The HTML5 drag and drop API allows support for dragging and dropping items within and between web sites
 - Fullscreen API
 - Controls the usage of the whole screen for a Web page or application, without the browser UI displayed

HTML5

- Device access: allowing for the usage of various input and output devices
 - Using the Camera API
 - Allows using, manipulating, and storing an image from the computer's camera
 - Touch events
 - Handlers to react to events created by a user pressing touch screens
 - Using geolocation
 - Let browsers locate the position of the user using geolocation
 - Detecting device orientation
 - Get the information when the device on which the browser runs changes orientation

HTML

- HTML elements

- The building blocks of HTML pages
- Are represented by tags
- HTML tags label pieces of content such as "heading", "paragraph", "table", etc.
- Browsers do not display the HTML tags, but use them to render the content of the page
- Element names are surrounded by angle brackets
- Tags normally come in pairs like `<div>` and `</div>`

- The General Syntax

```
<tagName> Some Content </tagName>
```

HTML Boilerplate

```
<!DOCTYPE html>
<html>

<head>
  <!-- Metadata goes in head -->
  <title>Page Title</title>
</head>

<body>
  <!-- Content goes in the body -->
</body>

</html>
```


HTML Boilerplate

- DOCTYPE
 - Indicates that your HTML content uses HTML5
- <html>
 - Represents the root (top-level element) of an HTML document
 - Also referred to as the root element
 - All other elements must be descendants of this element
- <head>
 - Provides general information (metadata) about the document
 - For e.g., it provides title, links to scripts and style sheets
- <body>
 - Represents the content of an HTML document
 - Only one <body> element in a document

HTML Boilerplate

- `<title>`
 - Defines the title of the document, shown in a browser's title bar or on the page's tab
 - Can only contain text
- HTML comments
 - Comments
 - Are ignored by the browser and invisible to the user
 - Can help document your source

```
<!-- Comment here -->
```

Basic Tags

- MDN - HTML elements reference
 - <https://developer.mozilla.org/en-US/docs/Web/HTML/Element>
- `<h1>` - `<h6>` - Heading elements
 - Represent six levels of section headings
 - `<h1>` is the highest section level and `<h6>` is the lowest
 - Are block-level elements – each element will get its own line
- `<p>` - Paragraph
 - Represents a paragraph of text
 - Is a block-level element
- ``
 - Indicates that its contents have strong importance, seriousness, or urgency. Browsers typically render the contents in bold type
 - Is an inline element
- ``
 - Marks text that has stress emphasis
 - Is an inline element

HTML Lists

- `` - Ordered list
 - Represents an ordered list of items, typically rendered as a numbered list
 - Can contain zero or more `` elements, which in turn often contain nested `` or `` elements
 - Attributes
 - `reversed` - boolean attribute specifies that the items of the list are specified in reversed order
 - `start` - integer attribute specifies the start value for numbering the individual list items
 - `type` - indicates the numbering type – 'a', 'A', 'i', 'I', '1'
- `` - Unordered list
 - Represents an unordered list of items, typically rendered as a bulleted list
 - Can contain zero or more `` elements, which in turn often contain nested `` or `` elements
- `` - List item
 - Is used to represent an item in a list
 - It must be contained in a parent element
 - an ordered list (``), an unordered list (``)

Divs and Spans

- `<div>` - Content division element
 - Is a generic container for content
 - Has no effect on the content or layout until styled using CSS
 - Used to group content
 - Should be used only when no other semantic element (such as `<article>` or `<nav>`) is appropriate
 - Is a block-level element
- ``
 - Is a generic inline container for phrasing content
 - Can be used to group elements for styling purposes
 - Should be used only when no other semantic element is appropriate
 - Is an inline element

HTML Attributes

- MDN – HTML attribute reference
 - <https://developer.mozilla.org/en-US/docs/Web/HTML/Attributes>
- Attributes
 - Provide additional values to elements
 - Can be used to
 - Configure the elements
 - Adjust the behavior in various ways to meet the criteria the users want
 - Take the form of key="value" pair for an element

```
<element name="value"></element>
```

Images

- ``
 - Embeds an image into the document
 - Does not have end tag
 - Attributes
 - `alt`
 - defines the alternative text describing the image
 - users will see this text displayed if the image URL is wrong, or if the image is not yet downloaded
 - `src`
 - specifies image URL. This attribute is mandatory for the `` element
 - `height`
 - the intrinsic height of the image in pixels
 - `width`
 - the intrinsic width of the image in pixels

```

```

Links

- `<a>` - Anchor element
 - Creates a hyperlink to other web pages, files, locations within the same page, email addresses, or any other URL
- Attributes
 - href - Contains a URL or a URL fragment that the hyperlink points to
 - URL can be relative or absolute
 - URLs are not restricted to Web (HTTP)-based documents, but can use any protocol supported by the browser
 - For example, file:, ftp:, and mailto: work in most browsers
 - target - Specifies where to display the linked URL
 - It is a name of, or keyword for, a browsing context: a tab, window, or `<iframe>`
 - Keywords: `_self` (default), `_blank`, `_parent`, `_top`

```
<a href="https://www.mozilla.com/">  
  Go to Mozilla  
</a>
```


Tables

- `<table>`
 - Represents tabular data, information presented in a two-dimensional table comprised of rows and columns of cells containing data
- `<tr>`
 - Defines a row of cells in a table
 - Within each row (`<tr>`), the `<th>` and `<td>` elements can be used to create header and data cells, respectively
 - Each cell is placed into its own column
- `<td>`
 - Defines a cell of a table that contains data
- `<th>`
 - Defines a cell as header of a group of table cells

Tables

- `<thead>`
 - Defines a set of rows defining the head of the columns of the table
- `<tbody>`
 - Encapsulates a set of table row (`<tr>`) elements, indicating that they comprise the body of the table (`<table>`)
 - You may use more than one `<tbody>` per table as long as they are all consecutive. This lets you divide the rows in large tables into sections
- `<tfoot>`
 - Defines a set of rows summarizing the columns of the table

Tables

- `<caption>` - Table Caption element
 - Specifies the caption (or title) of a table
 - If used, is always the first child of a `<table>`
- `<colgroup>`
 - Defines a group of columns within a table
- `<col>`
 - Defines a column within a table
 - Used for defining common semantics on all common cells
 - Generally found within a `<colgroup>` element
 - Allows styling columns using CSS

Forms

- `<form>`
 - Represents a document section that contains interactive controls for submitting information to a web server
- Attributes
 - action - The URI of a program that processes the form information
 - method - The HTTP method that the browser uses to submit the form
 - Possible values are
 - post - Corresponds to the HTTP POST method; form data are included in the body of the form and sent to the server
 - get - Corresponds to the HTTP GET method; form data are appended to the action attribute URI with a '?' as separator, and the resulting URI is sent to the server

Forms

- `<input>`
 - Is used to create interactive controls for web-based forms in order to accept data from the user
 - How an `<input>` works varies considerably depending on the value of its type attribute
 - The default type adopted is text
- Commonly used input types include
 - text
 - A single-line text field. Line-breaks are automatically removed from the input value
 - password
 - A single-line text field whose value is obscured
 - file
 - A control that lets the user select a file. Use the accept attribute to define the types of files that the control can select
 - hidden
 - A control that is not displayed but whose value is submitted to the server

Forms

- Button, checkbox, radio
 - submit: A button that submits the form
 - reset: A button that resets the contents of the form to default values
- button: A push button with no default behavior
- image: A graphical submit button. You must use the src attribute to define the source of the image and the alt attribute to define alternative text
- checkbox: A check box allowing single values to be selected/deselected
- radio: A radio button, allowing a single value to be selected out of multiple choices

Forms

- HTML5
 - date: A control for entering a date (year, month, and day, with no time)
 - month: A control for entering a month and year, with no time zone
 - week: A control for entering a date consisting of a week-year number and a week number with no time zone
 - time: A control for entering a time value with no time zone
- number: A control for entering a number
- range: A control for entering a number whose exact value is not important
- color: A control for specifying a color. A color picker's UI has no required features other than accepting simple colors as text
- email: A field for editing an e-mail address
- search: A single-line text field for entering search strings. Line-breaks are automatically removed from the input value
- tel: A control for entering a telephone number
- url: HTML5 A field for entering a URL

Forms

- `<label>`

- Represents a caption for an item in a user interface
- A `<label>` can be associated with a control
 - Either by placing the control element inside the `<label>` element,

```
<label>Click me <input type="text"></label>
```

- Or by using the 'for' attribute

```
<label for="username">Click me</label>  
<input type="text" id="username">
```

- Attributes

- **for** - The id of a labelable form-related element in the same document as the label element

Forms

- Form data validation
 - Form validation helps us to ensure that users fill out forms in the correct format, making sure that submitted data will work successfully with our applications
- required attribute
 - makes an input mandatory
 - the form won't submit (and will display an error message) when the input is empty
- pattern attribute
 - expects a Regular Expression as its value
 - a pattern that can be used to match character combinations in text strings
 - The `<textarea>` element does not support the pattern attribute

Forms

- Constraining the length of entries
 - Text fields can be constrained in size using 'minlength' and 'maxlength' attributes
 - A field is invalid if its value is
 - shorter than the minlength value or
 - longer than the maxlength value
- For number fields (i.e. `<input type="number">`), the 'min' and 'max' attributes also provide a validation constraint

Forms

- `<input type="checkbox">`
 - Rendered by default as square boxes that are checked (ticked) when activated
 - 'value' attribute can be used to mention the value of the checkbox
 - Has 'checked' attribute that is readable and settable
- `<input type="radio">`
 - Radio buttons let a user select one of a limited number of choices
 - 'value' attribute can be used to mention the value of the radio button
 - Has 'checked' attribute that is readable and settable
 - A radio group is defined by giving each of radio buttons in the group the same name

Forms

- `<select>`
 - Represents a control that provides a menu of options
 - The `<option>` element defines an option that can be selected
 - By default, the first item in the drop-down list is selected
 - Attributes
 - `size` - represents the number of rows in the list that should be visible at one time
 - `multiple` - boolean attribute indicates that multiple options can be selected in the list
- `<option>`
 - Is used to define an item contained in a `<select>` element
 - Attributes
 - `value` - specifies the value to be sent to a server
 - `selected` - specifies that an option should be pre-selected when the page loads

Forms

- `<textarea>`
 - Represents a multi-line plain-text editing control
 - Can hold an unlimited number of characters, and the text renders in a fixed-width font
- Attributes
 - `cols` - the visible width of the text control, in average character widths
 - `rows` - the number of visible text lines for the control
 - `maxlength` - maximum number of characters (Unicode code points) that the user can enter
 - `minlength` - minimum number of characters (Unicode code points) required that the user should enter
 - `wrap` – (hard | soft) specifies how the text in a text area is to be wrapped when submitted in a form

Q & A

- Thank you!