Refreshing Java Script and CSS

Unit#1

Marwadi University

Department of Information Technology

Advanced Web Programming (3161611)

Tejas Chauhan

Highlights

- CSS benefits
- CSS syntax
- Responsive design
- Bootstrap introduction
- Javascript syntax
- Javascript inbuilt objects
- Error handling and event handling
- DOM
- Asynchronous Programming

CSS benefits

- CSS – Cascading Style Sheet
- Presentation purpose in web pages

- Separates web content from web presentation
- Manages large-scale complex websites
- Easier to maintain and update
- Greater consistency in design
- Faster download times
- Search engine optimization benefits
- Ease of presenting different styles to different viewers
- Greater accessibility

CSS syntax

- CSS basic building blocks are:
- The property which is an identifier, that is a human-readable name, that defines which feature is considered.
- The value which describe how the feature must be handled by the engine.

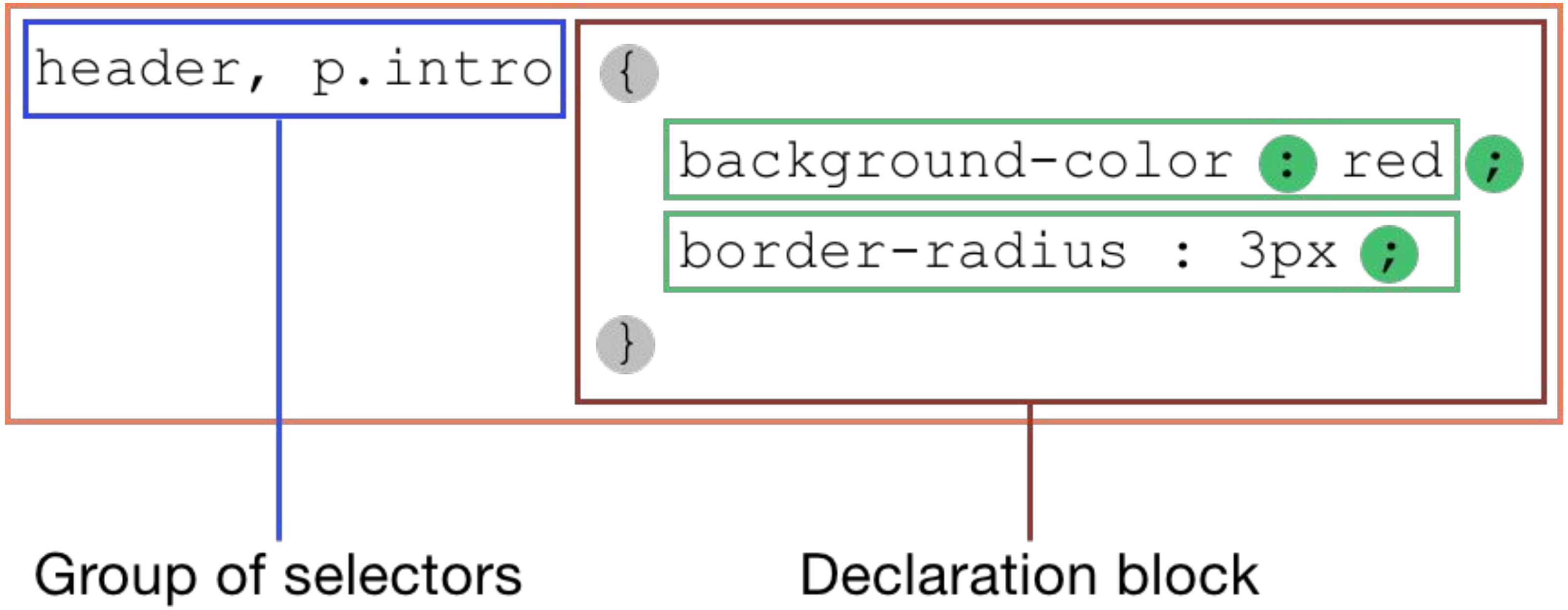
CSS syntax

- A property and value pair is called a declaration.
- Declarations are grouped in blocks (declaration blocks), that is in a structure delimited by an opening brace, '{', and a closing one, '}'.

- Selectors are conditions selecting some elements of the page.
- A selector group and an associated declarations block, together, are called a ruleset, or often simply a rule.

CSS syntax

- A CSS ruleset (or rule) is visualized in the diagram below.



CSS Types

- Inline style sheets
- Document level style sheets
- External level style sheets

Inline style sheets

- Styles can be applied to HTML tags using this rule

  <TAG style=“Property:value”>

- Example:

  <h1 style="color:#FF0000; font-style:italic">

- Advantages
  - We can apply unique style on tags for whole document
- Disadvantages
  - Actual content is mixed with presentation

Document level style sheets

- It appears in <head> section
- We need to use <style type=“text/css”>

```
<style type="text/css">
    h1 {
        color:#FF0000;
        font-style:italic;
    }
</style>
```

- Advantages
  - Helps to decide the layout of the web page
  - Useful when we want to apply unique style sheet for the web page
- Disadvantage
  - Applicable to single page only

External Style Sheets

- Useful when particular style is needed to apply more than one web document
- Desired style is stored in .css file
- To link external style file we use,
  <link rel="stylesheet" href="style.css" />

Responsive design

- It is the approach that suggests that design and development should respond to the user's behavior and environment based on screen size, platform and orientation.
- Responsive web design is about creating web pages that look good on all devices!
- A responsive web design will automatically adjust for different screen sizes and viewports.

- To create a responsive website, add the following <meta> tag to all your web pages:
  <meta name="viewport" content="width=device-width, initial-scale=1.0">

Responsive design

- CSS3 Media Queries:
- Media queries can be used to check many things, such as:
  - width and height of the viewport
  - width and height of the device
  - orientation (is the tablet/phone in landscape or portrait mode?)
  - Resolution

```
@media screen and (max-width: 699px) and (min-width: 520px)
{
    #leftsidebar {width: 200px; float: left;}
    #main {margin-left:216px;}
}
```

Bootstrap introduction

- Bootstrap is a free and open-source CSS framework for building responsive and mobile-first sites.
- It contains CSS- and JavaScript-based design templates for typography, forms, buttons, navigation, and other interface components.
- https://getbootstrap.com/

Javascript syntax

- Where to put your script?

- JavaScript programs can be used in two ways:
  - Incorporated directly into an HTML file
    - Using <script> tag
  - Placed in an external (source) file
    - Has file extension .js
    - Contains only JavaScript statements

Javascript syntax

- Creating a JS File / JavaScript source files

- Use src attribute of <script> tag to denote source of JavaScript statements
- Browser will ignore any JavaScript statements inside <script> and </script> if src attribute is used
- Cannot include HTML tags in source file
- Example:

```
<script type="text/javascript">alert(“Hello.”);</script>
<script src=“script.js”></script>
```

Javascript inbuilt objects

- Fundamental objects
- These are the fundamental, basic objects upon which all other objects are based.
  - Object
  - Function
  - Boolean
  - Symbol

Javascript inbuilt objects

- Error objects
- Error objects are a special type of fundamental object.
  - Error
  - EvalError
  - InternalError
  - RangeError
  - ReferenceError
  - SyntaxError
  - URIError, etc…

Javascript inbuilt objects

- Numbers and dates
- These are the base objects representing numbers, dates, and mathematical calculations.
  - Number
  - BigInt
  - Math
  - Date

Javascript inbuilt objects

- Text processing
- These objects represent strings and support manipulating them.
  - String
  - RegExp

Javascript inbuilt objects

- Indexed collections
- These objects represent collections of data which are ordered by an index value. This includes (typed) arrays and array-like constructs.
    - Array
    - Int8Array
    - Int16Array
    - Int32Array
    - Float32Array
    - Float64Array
    - etc…

Javascript inbuilt objects

- Structured data
- These objects represent and interact with structured data buffers and data coded using JavaScript Object Notation (JSON).
  - ArrayBuffer
  - SharedArrayBuffer
  - DataView
  - JSON

Error handling

- The try statement lets you test a block of code for errors.
- The catch statement lets you handle the error.
- The throw statement lets you create custom errors.
- The finally statement lets you execute code, after try and catch, regardless of the result.

Event handling

- Here is a list of some common HTML events:

Event	Description
onchange	An HTML element has been changed
onclick	The user clicks an HTML element
onmouseover	The user moves the mouse over an HTML element
onmouseout	The user moves the mouse away from an HTML element
onkeydown	The user pushes a keyboard key
onload	The browser has finished loading the page
etc...	

Inline event handlers

- HTML allows event handler attributes, with JavaScript code, to be added to HTML elements.
- Syntax:

  <element event='some JavaScript'>
- Example:

  <button onclick="bgChange()">Press me</button>

Event handler properties

- These are the properties that exist to contain event handler code.
- Example:

```
const btn = document.querySelector('button');
btn.onclick = function() {
    const rndCol = 'rgb(' + random(255) + ',' + random(255) + ','
    + random(255) + ')';
    document.body.style.backgroundColor = rndCol;
}
```

Event handler properties

- You could also set the handler property to be equal to a named function name.

```
const btn = document.querySelector('button');
function bgChange() {
    const rndCol = 'rgb(' + random(255) + ',' + random(255) + ','
    + random(255) + ')';
    document.body.style.backgroundColor = rndCol;
}
btn.onclick = bgChange;
```

addEventListener() & removeEventListener()

- The newest type of event mechanism is defined in the Document Object Model (DOM) Level 2 Events Specification.
- It provides browsers with a new function — addEventListener().
- This functions in a similar way to the event handler properties, but the syntax is obviously different.

addEventListener() & removeEventListener()

- We could rewrite our random color example to look like this:

```
const btn = document.querySelector('button');
function bgChange() {
    const rndCol = 'rgb(' + random(255) + ',' + random(255) + ','
    + random(255) + ')';
    document.body.style.backgroundColor = rndCol;
}
btn.addEventListener('click', bgChange);
```

addEventListener() & removeEventListener()

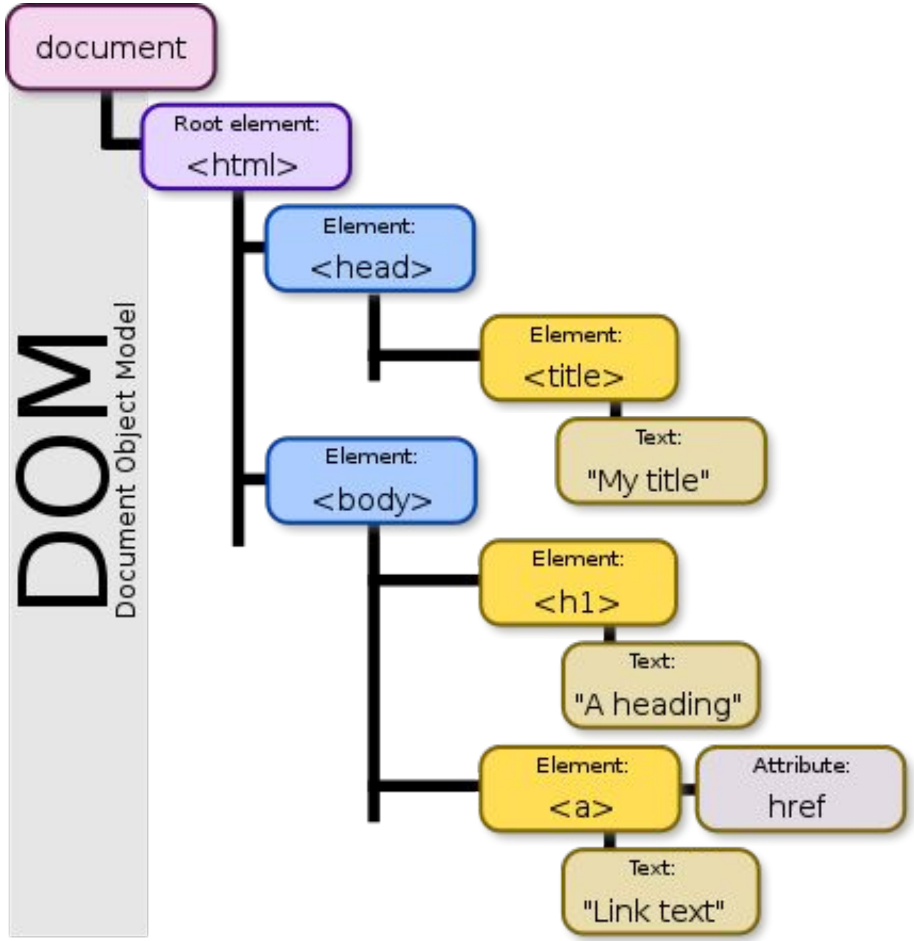
- Inside the addEventListener() function, we specify two parameters:
  - the name of the event we want to register this handler for, and
  - the code that comprises the handler function we want to run in response to it.
- This mechanism has some advantages also.
  - there is a counterpart function, removeEventListener(), which removes a previously added listener.
  - For example, this would remove the listener set in the first code block in this section:
  - btn.removeEventListener('click', bgChange);

What is the DOM?

- The DOM is a W3C (World Wide Web Consortium) standard.
- The Document Object Model (DOM) is a programming interface for HTML and XML documents.
- It represents the page so that programs can change the document structure, style, and content.
- The DOM represents the document as nodes and objects. That way, programming languages can connect to the page.
- When a web page is loaded, the browser creates a Document Object Model of the page.

What is the DOM?

- The HTML DOM model is constructed as a tree of Objects:



Asynchronous Programming

- Asynchronous programming is a means of parallel programming in which a unit of work runs separately from the main application thread and notifies the calling thread of its completion, failure or progress.
- In a synchronous programming model, things happen one at a time.
- When you call a function that performs a long-running action, it returns only when the action has finished and it can return the result.
- This stops your program for the time the action takes.

Asynchronous Programming

- An asynchronous model allows multiple things to happen at the same time.
- When you start an action, your program continues to run.
- When the action finishes, the program is informed and gets access to the result (for example, the data read from disk).
- We can compare synchronous and asynchronous programming using a small example: restaurant and waiter.

Review

- CSS benefits
- CSS syntax
- Responsive design
- Bootstrap introduction
- Javascript syntax
- Javascript inbuilt objects
- Error handling and event handling
- DOM
- Asynchronous Programming

Thank You.