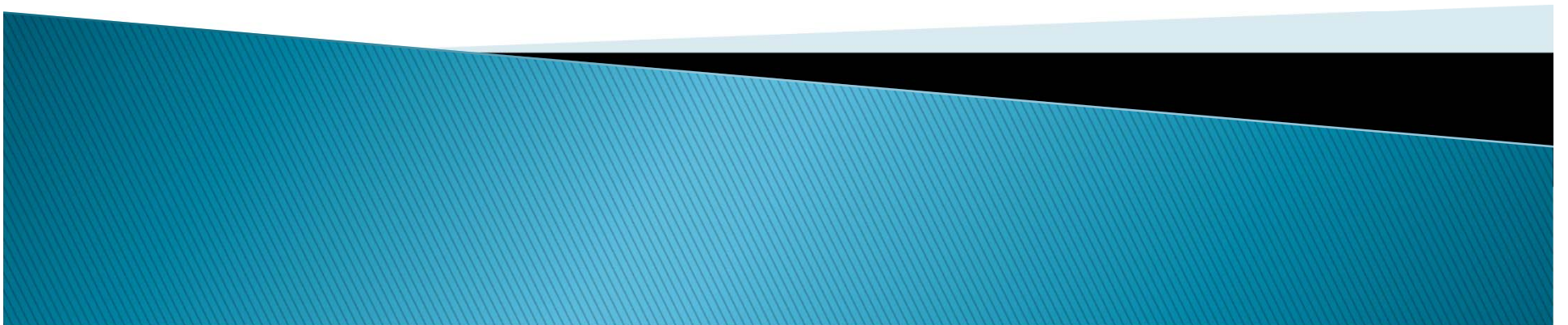
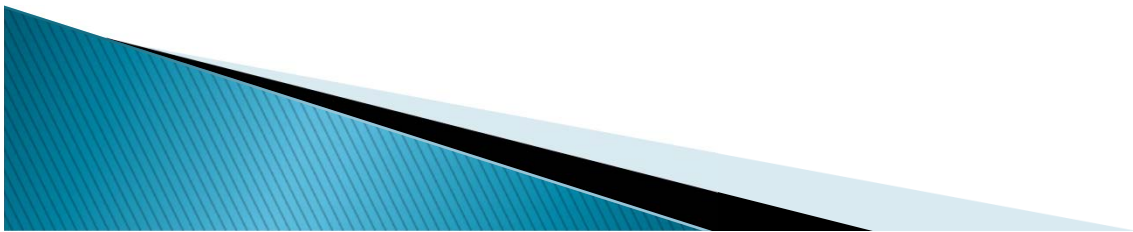


CST8285 – Week 8

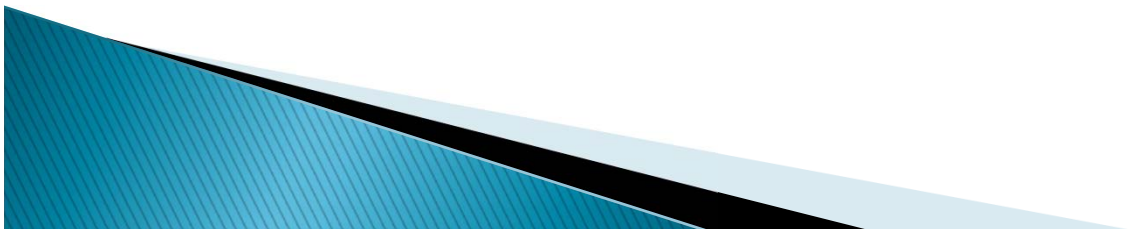
JavaScript DOM and Arrays





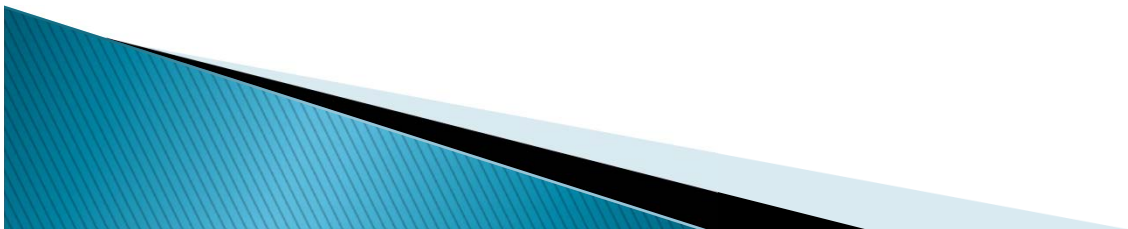
DOM – Document Object Model

- ▶ Used for HTML and XML documents
- ▶ In web programming, it provides the link between web pages and programming languages
- ▶ Organizes HTML elements into objects
- ▶ The HTML DOM is **big!** We don't have time to cover everything in class.



The Document Object

- ▶ The document object contains all of the elements of a webpage that's loaded into a browser.
- ▶ The elements, attributes, and comments of an HTML page are **nodes** of the document object.
- ▶ The document object is the **root node** for all other nodes.
- ▶ The document object provides access to these nodes through its properties and methods.



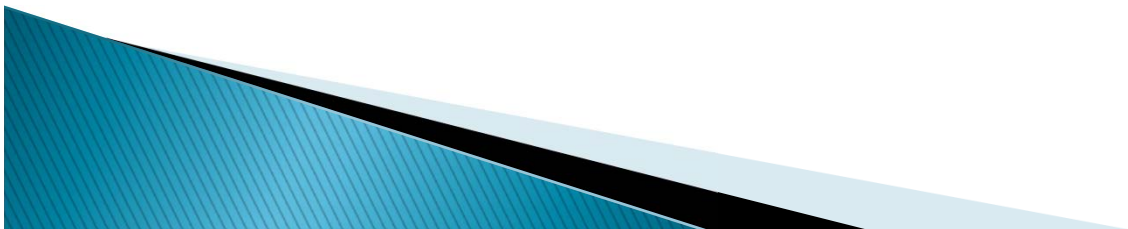
Key Document Object Properties

- ▶ `document.cookie` – returns name/value pairs for all cookies in the document
- ▶ `document.forms` – returns an array of all forms in the document object
- ▶ `document.images` – returns an array of all images in the document object
- ▶ `document.title` – sets or returns the title of the document
- ▶ `document.URL` – returns the full URL of the document



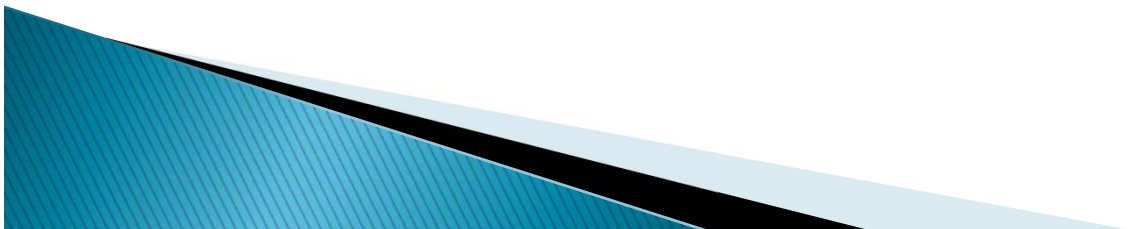
Key Document Object Methods

- ▶ `document.getElementById()` – returns an element based on its id attribute
- ▶ `document.write()` – writes text or HTML code to the document.
- ▶ `document.getElementsByTagName()` – returns a `NodeList` of all elements of a specified tag name
- ▶ `document.createElement()` – creates a new element



The Element Object

- ▶ The Element object represents an HTML element from a web page.
- ▶ Nodes contained within an element object are called **child nodes**.
 - Can be comments, other elements, or attributes.
- ▶ All types of elements have shared properties and methods (ex: `element.childNodes` will return a `NodeList` containing all child nodes of an element)



The NodeList

- ▶ Contains a list of nodes.
- ▶ Can loop through the items of a NodeList using a for loop.
- ▶ Although similar, a NodeList is not an array.
- ▶ Nodes are accessed using an index.
 - Ex. `document.childNodes[1]`

* List is not live if retrieved using the `querySelectorAll` method



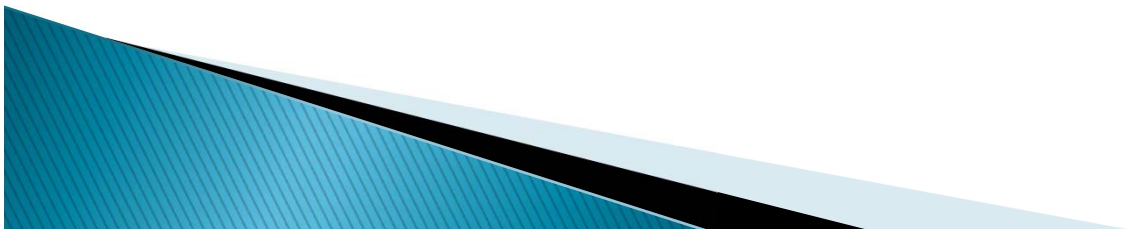
Key Element Object Properties

- ▶ `element.attributes` – returns a `NamedNodeList` of all the element's attributes
- ▶ `element.id` – sets or returns the id of the element
- ▶ `element.innerHTML` – sets or returns the content of an element
- ▶ `element.className` – sets or returns the class of an element (useful for applying CSS styles dynamically)



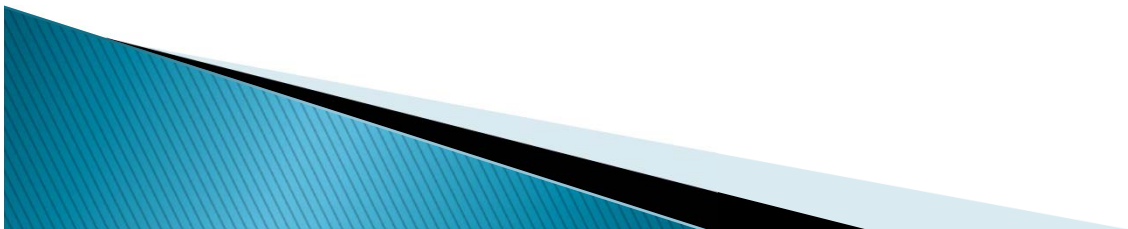
Key Element Object Methods

- ▶ `element.setAttribute()` – sets the named attribute's value
- ▶ `element.cloneNode()` – clones an element
- ▶ `element.appendChild()` – appends a new node as the element's last child node
- ▶ `element.insertBefore()` – inserts a new node before specified child node
- ▶ `element.removeChild()` – removes a specified child node.



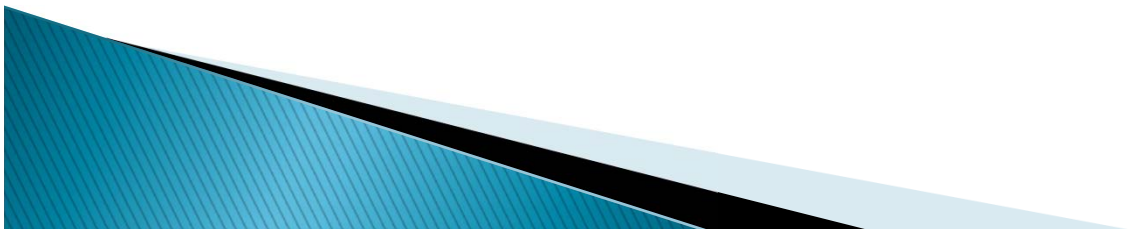
The Attribute Object

- ▶ Represents an HTML attribute
- ▶ Attribute objects are always a child node of an element object.
- ▶ Attributes accessed via the element object's attributes property are returned in a NamedNodeList.
 - The NamedNodeList is like the NodeList, but you can access an attribute by its index or by its name.



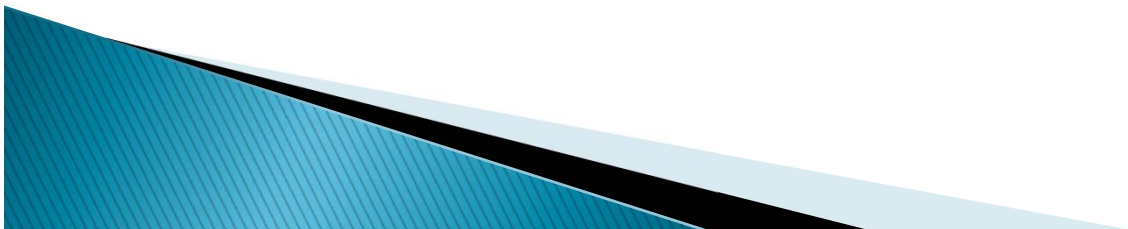
Key Attribute Object Properties

- ▶ `attribute.name` – returns the attribute's name
- ▶ `attribute.value` – sets or returns the attribute's value
- ▶ `attribute.specified` – returns true if the attribute is specified, otherwise returns false



Events

- ▶ Used to register event handlers
- ▶ Examples of different types of events:
 - Mouse events (onclick, onmouseup, onmousedown)
 - Frame/Object events (onload, onerror, onabort)
 - Form events (onchange, onfocus, onsubmit, onreset)
- ▶ Usually used in conjunction with a function
 - ex. `document.forms["myForm"].onsubmit = validateForm();`
will call the `validateForm()` method when the form is submitted.



Element-Specific Objects

- ▶ Certain HTML elements have their own objects
 - Examples include form inputs (checkboxes, textboxes, etc), images, links, buttons, etc
- ▶ These objects have properties/methods specific to the element
 - Example: the image object has `height` and `src` properties.



The Input Text Object

▶ Properties:

- value – returns or sets the text inside the text box
- maxlength – returns or sets the maxlength of the textbox
- size – returns or sets the size of the textbox, in characters

▶ Methods

- select() – selects the contents of the text box.



The Input Radio/Checkbox Objects

- ▶ Although they are different types of objects, their key properties are the same.
- ▶ Properties
 - checked – returns or sets the checked value of the button
 - value – returns or sets the value attribute's value



JavaScript Arrays

- ▶ Stores multiple values in one object
- ▶ Can store different object types in one array
- ▶ Arrays in JavaScript are objects, and have predefined methods and properties, such as:
 - length – returns or sets the number of elements in an array
 - sort() – sorts an array's elements
 - push() – adds a new element onto the end of an array
 - concat() – joins two or more arrays, returns the joined array



Defining JavaScript Arrays

- ▶ Constructor

- `var newArray = new Array();`

- ▶ Literal

- `var newArray = ["Item1", "Item2", "Item3"];`

- ▶ Condensed

- `var newArray = new Array("Item1", "Item2", "Item3");`

