

Announcements

- Acid3 Test
 - Checks Web Browser's compliance with web standards
 - <http://acid3.acidtests.org/>
- <http://caniuse.com/>
 - Provides up-to-date support tables for support of front-end web technologies
 - http://caniuse.com/#info_about

Getters/Setters

- Getter method – gets the value of a property
- Setter method – sets the value of a property
- You can define getters and setters on objects and user-defined objects
 - Use get and set to define them
 - **Example:** GetterSetter.html

Private Variables

- Properties cannot be made private
- JavaScript can have functions defined inside of functions
- JavaScript has closures, therefore, an inner function can access variables and parameters of its outer function (even after the outer function has finished)
- By placing functions inside of the function constructor we can provide access to private variables
- **Example:** PrivateVarFunctions.html
- Reference
 - <http://javascript.crockford.com/private.html>
 - <http://philipwalton.com/articles/implementing-private-and-protected-members-in-javascript/>

Nonextensible and Sealed Obj

- In JavaScript you can add properties and methods to an object any time (extensible)
- You can restrict this behavior by using `Object.preventExtensions()`
- What if you don't want properties deleted as well?
 - Seal the object
 - By sealing an object you create the same abstraction in class definition where once a class is defined the class values and methods are set
- **Example:** ExtensibleSealed.html

Freeze

- Strictest protection
- Not extensible, sealed, and data properties can not be modified
 - Constant object
- **Example:** Freeze.html

Errors

- **Example:** Errors1.html
- Error types
 - Error → Base type for errors
 - URIError
 - TypeError
 - EvalError → Error while using eval()
 - RangeError
 - SyntaxError
 - ReferenceError
- **Example:** Errors2.html

throw

- Can be used to throw custom errors
 - `throw "NOOOO!";`
 - `throw { type1Error: "Done" }`
- **Example:** Errors3.html

HTML5 Local Storage API

- **localStorage** → stores data with no expiration date
- To store data:
 - `localStorage.setItem("name", "Mary");`
- To retrieve data:
 - `localStorage.getItem("name");`
- You can only store strings. To store objects you could use `JSON.stringify`
- **sessionStorage** → equivalent to `localStorage`, but data is deleted when the browser is closed
- **Example:** `ToDoList.html`
 - Notice the use of the **contenteditable** property in the HTML
 - Try the property in other HTML elements
- To clean `localStorage`
 - Right click on page in Chrome
 - Select "Inspect"
 - Select "Console"
 - Type `"window.localStorage.clear()"`
- Reference: http://www.w3schools.com/html/html5_webstorage.asp

HTML5 Canvas

- **Canvas**

- Container for graphics
- It is a rectangular area on the page
- You use JavaScript to draw on the fly
- Suited for game applications
- You can store the image as a jpg or png file

- Reference and Examples:

- http://www.w3schools.com/html/html5_canvas.asp

- **Example:** DrawingPointer.html

- How would you create an animation of a drawing you have done?

HTML5 SVG

- SVG – Scalable Vector Graphics
 - Language use to describe 2D graphics
 - Pure XML
 - SVG graphics do not lose any quality when resized or zoomed
 - Best suited for applications with large rendering areas
 - You can embed in your HTML
- <svg> element → container for SVG Graphics
- Reference, Examples and comparison table available at:
 - http://www.w3schools.com/html/html5_svg.asp
- Open-source vector graphics editor
 - <https://inkscape.org/en/>

FileReader API

- Using HTML5 FileReader API
- FileReading folder
 - **Example:** FileReadingText.html
 - **Example:** FileReadingImage.html

HTML5 Geolocation API

- **Example:** Geolocation.html
- Reference:
 - http://www.w3schools.com/html/html5_geolocation.asp

Miscellaneous

- **Example:** `GuessGameWithSound.html`, `ThrowDie.html`