

CMSC335

Web Application Development with JavaScript



JavaScript5

Department of Computer Science
University of MD, College Park

Slides material developed by Ilchul Yoon, Nelson Padua-Perez

Sets

- Collection of keys
- Keys can be primitive or references
- The Set constructor has zero or more arguments. With no arguments an empty set will be created
- If an argument is specified, it needs to be iterable (e.g., array)
- When iterating over sets, elements will be processed in the order they were inserted
- **Example:** Set.html

Maps

- Collection of key - value pairs
- Keys and values can be primitive or references
- Can define a map via iterable over key-value pairs
- `keys()` method returns iterable for keys in the map
- `values()` method returns iterable for values in the map
- `entries()` method returns iterable over (key, value) pairs
- **Example:** `Map.html`

WeakMaps

- A collection of key-value pairs **in which primitive data types are not allowed as keys)**
- The keys must be objects
- The values can be arbitrary values
- Key is weakly held; if the object representing a key is the only reference to the object, the object will be garbage-collected
- You cannot iterate over keys, values, or entries
- You need a key to get some content out of the map

WeakMaps

- **Example:** WeakMap.html
 - As defined the example relies on a Map
 - Run the example in Chrome and under the **Memory** option execute “Take Heap Snapshot”
 - Search for “TerpObject” in the **Constructor** column
 - Repeat the above process, but using a WeakMap rather than a map
- Uses
 - To define private data in a “class”

Immediately Invoked Function Expression (IIFE)

- A JS function that runs as soon as it is defined
- A design pattern known as a Self-Executing Anonymous Function
- Two parts
 - Anonymous function with lexical scope enclosed within the Grouping Operator ().
 - Prevents accessing variables within the IIFE idiom as well as polluting the global scope
- Emulating block-scoped variables
- Not needed, if “let” is used instead of “var”
- **Example:** life.html

Form Validation

- Form data validation
 - We can validate the data associated with a form by recognizing the submit event
 - **window.onSubmit = validateData;**
 - » validateData is a function that will check for data validity
 - » It will return true or false
- Keep in mind that JavaScript can be disabled therefore the server application must also validate the data
- Notice the organization code (HTML, CSS, JS separate) in the example
- **Example:** FormValidation