JavaScript Class – June 17, 2024

Give assignment that has them look up String, Number methods, other methods for data types

Object.assign()

for (let entry of journal) {

}

1. Arrays - Mutating vs non-mutating methods
   1. pop, push, shift and unshift are mutating arrays
      1. They alter the original array
   2. concat, slice, map return a new array without altering the original array
2. Array.splice(start, delete, element1, element2…)
   1. Removes and adds elements from anywhere within array
      1. start – the index position at which you want to start the operation
      2. delete – the number of elements you want to delete
         1. if you don’t want to delete any, put 0 here
      3. then add the elements you want to add to the array
   2. const hatsArray = ['baseball', 'bucket', 'boonie', 'straw', 'boater', 'beret']; console.log(hatsArray); // ['baseball', 'bucket', 'boonie', 'straw', 'boater', 'beret']  
      hatsArray.splice(2, 1);   
      console.log(hatsArray); // ['baseball', 'bucket', 'straw', 'boater', 'beret']  
      hatsArray.splice(3, 0, 'fedora');   
      console.log(hatsArray); // ['baseball', 'bucket', 'straw', ‘fedora’, 'boater', 'beret']  
      hatsArray.splice(1, 2, 'beanie', 'cloche');   
      console.log(hatsArray); // ['baseball', ‘beanie’, 'cloche', ‘fedora’, 'boater', 'beret']
3. Array.reverse() – reverses original array and modifies it  
   const prime = [2, 3, 5, 7];  
   prime.reverse();  
   console.log(prime); // [7, 5, 3, 2];
4. Array.fill() – have them lookup and put in homework - <https://codedamn.com/news/javascript/javascript-array-methods>
5. Array.indexof()
   1. find the index of the specified element which is present in array
   2. if element not found, the method returns -1

const strings = [‘code’, ‘dang’, ‘web3’];  
console.log(strings.indexOf(‘dang’)); // 1  
console.log(strings.indexOf(‘hello’)) // -1

* 1. multiple items with same name or same number – have students look up and then put in homework - <https://codedamn.com/news/javascript/javascript-array-methods>
  2. Array.lastindexof() - have students look up and then put in homework - <https://codedamn.com/news/javascript/javascript-array-methods>

1. Array.find()
   1. Returns value of the first element n the array that satisfies provided condition
   2. If none satisfy provided condition, undefined is returned
      1. const numbers = [4, 9, 16, 25, 32];  
         const first = numbers.find((value, index, array) => { return value > 18; });  
         console.log(first); // 25
      2. function isPrime(element, index, array) {  
          let start = 2;   
          while (start <= Math.sqrt(element)) {  
          console.log('start = ' + start);   
          console.log('element = ' + element);   
          console.log('Math.sqrt(element) = ' + Math.sqrt(element));  
          console.log('element % start = ' + element % start);   
          if (element % start++ < 1) {  
          console.log('element % start++ \< 1');   
          return false;   
          }  
          }  
          console.log('element > 1 = ' + element > 1);   
          return element > 1;  
         }  
         console.log([4, 6, 8, 12].find(isPrime)); // undefined, not found  
         console.log([4, 5, 8, 11, 12].find(isPrime)); // 5
2. Array.findIndex() – look up and homework
3. Array.some() – look up and homework
4. Array.filter()
   1. Creates new array with all elements that pass the condition implemented by the provided function

const strings = [‘functional’, ‘Hello’, ‘Startup’, ‘Web’];

const longwords = strings.filter(word => word.length > 6);

console.log(longwords); // [“functional”, “Startup”]

1. Array.map()
   1. Creates new array populated with the results of calling a provided function on every element in the calling array  
      const num = [2, 3, 4, 5];  
      const square = num.map(x => x \* x);  
      console.log(square); // [4, 9, 16, 25]
2. Array.slice()
   1. takes start and end indices and returns an array that only has the elements between them
      1. start index is inclusive and end index is exclusive
         1. console.log([“cat”, “dog”, “bird”, “fish”, “rabbit”]).slice(2,4));  
            // [“bird”, “fish”]
      2. if end index is not given, slice takes all of the elements after the start index
         1. console.log([“cat”, “dog”, “bird”, “fish”, “rabbit”]).slice(2));  
            // [“bird”, “fish”, “rabbit”]
      3. if start index is also omitted the entire array is copied
   2. creates a new array, doesn’t alter existing array  
      const petArray = [“cat”, “dog”, “bird”, “fish”, “rabbit”];  
      petArray.slice(2, 4);  
      console.log(petArray); // [“cat”, “dog”, “bird”, “fish”, “rabbit”]
3. Array.includes() – look up and homework
4. Array.reduce()
   1. Applies a function to each element in an array, resulting in a single output value
   2. Executes function for each value from left to right
   3. Return value of function is used in next call to function  
      const numbers = [1, 2, 3, 4];  
      const sum = numbers.reduce(function(accumulator, currentValue) {  
       return accumulator + currentValue;  
      }, 0);  
      console.log(sum); // 10  
      // change 0 to 5 // sum now 15  
      // change accumulator to bob and currentValue to dylan, to show that names of parameters don’t matter
   4. accumulator is total value returned by the function so far
   5. currentValue is the current element being processed in the array
   6. reduce() also takes an optional second argument as the initial value of the accumulator
5. Destructuring Arrays and Objects – maybe cover later
   1. const inventory = [  
       { name: “apples”, quantity: 2 },  
       { name: “bananas”, quantity: 0 },  
       { name: “cherries”, quantity: 5 }  
      ];  
      const result = inventory.find(({ name} ) => name === “cherries”);  
      console.log(result); // { name: ‘cherries’, quantity: 5 }
6. String methods
   1. slice() – homework
   2. indexOf()
      1. difference between Array.indexOf() and String.indexOf – String.indexOf() can look for a string containing more than one character, but array only looks for a single element  
         console.log(“one two three”.indexOf(“ee”)); // 11
   3. trim()
      1. removes whitespace (spaces, newlines, tabs and similar characters) from start and end of string  
         console.log(“ okay \n “.trim()); // okay
   4. repeat() – homework
7. Rest parameters – cover later