Array Methods

1. forEach( ) :

   -It is an array method that calls a function once for each element of the array.
   -No return value for the function
   -syntax: arrayName.forEach((element, index, array)=>{ //do something });
   -The callback function of forEach() has three arguments:
   ->element: The current array item.
   ->index (optional): The index of the current item.
   -> array(optional): The array forEach is being applied to.

2. map( ) :

   -it returns a new array by performing a function on each element of the array.
   -syntax: const newArray = arrayName.map((element, index, array)=>{ //do something });
   -The callback function of map() has three arguments:
   ->element: The current array item.
   ->index (optional): The index of the current item.
   -> array(optional): The array to which map is applied.

Note: Difference between map() and forEach() array methods –

1) Map is used when a new array with modified values is needed based on the original array.
2) forEach is used when performing actions on each array item without returning a new array.

3. filter( ) :

   -returns an array with values satisfying the specified condition.
   -syntax: const newArray =arrayName.filter(()=>{ // condition });
   -The callback function of the filter has three arguments:
   ->element: The current array item.
   ->index: The index of the current item.
   -> array: The array to which the filter is applied.

4. find():
    - returns a value that satisfies a condition applied to the existing array.
    - The callback function of find takes 3 arguments: item value , item index, array itself.
    - Syntax: const result = arrayName.find(()=>{//condition});

5. reduce():
    - method runs a function on every element of array to produce a single value.
    - work from left to right.
    - Used for finding highest, lowest, sum etc…
    - Syntax: result = arrayName.reduce((a,b)=>a>b?a:b); - highest element in array.
    - Will not modify original array.
    - Four args for reduce function callback: total(initial value/previously returned value), value, index , array
    - Minimum two args required.

6. reduceRight():
    - runs a function on each array element to produce a single value.
    - Works from right to left.
    - Similar to reduce syntax.

7. sort() : (Numeric sort)
    - syntax: arrayName.sort((a,b)=>a-b) – ascending order.
    - syntax: arrayName.sort((a,b)=>b-a) – decending order.
    - Modifies the original array.
    - Function inside sort known as compare function.

8. some() :
    - method checks if some elements of the array satisfy a specified condition.
    - Returns a Boolean value.
    - Syntax : let result = arrayName.some((args)=>{condition});
    - Callback function has 3 args: value, index, array

9. every() :
    - method checks if every element of array satisfy the specified the condition.
    - Returns a Boolean value.
    - Syntax : let result = arrayName.every((args)=>{condition});
    - Callback function has 3 args: value, index, array

10. pop() :
    - Array method to remove the last element of the array.
    - Syntax: arrayName.pop();

11. push() :
    - Array method that adds an element at the end of the array.
    - Syntax: arrayName.push(value);

12. shift():
    - array method that removes an element from the beginning of the array.
    - Syntax: arrayName.shift();

13. unshift() :
    - array method that adds an element to the beginning of the array.
    - Syntax: arrayName.unshift(value);