Prerana Annapantula

**Assignment**

Write workable code snippet for all the array functions given below.7

|  |  |
| --- | --- |
| **S.No.** | **Method & Description** |
| 1. | [concat()](https://www.tutorialspoint.com/typescript/typescript_array_concat.htm)  Returns a new array comprised of this array joined with other array(s) and/or value(s). |
| 2. | [every()](https://www.tutorialspoint.com/typescript/typescript_array_every.htm)  Returns true if every element in this array satisfies the provided testing function. |
| 3. | [filter()](https://www.tutorialspoint.com/typescript/typescript_array_filter.htm)  Creates a new array with all of the elements of this array for which the provided filtering function returns true. |
| 4. | [forEach()](https://www.tutorialspoint.com/typescript/typescript_array_foreach.htm)  Calls a function for each element in the array. |
| 5. | [indexOf()](https://www.tutorialspoint.com/typescript/typescript_array_indexof.htm)  Returns the first (least) index of an element within the array equal to the specified value, or -1 if none is found. |
| 6. | [join()](https://www.tutorialspoint.com/typescript/typescript_array_join.htm)  Joins all elements of an array into a string. |
| 7. | [lastIndexOf()](https://www.tutorialspoint.com/typescript/typescript_array_lastindexof.htm)  Returns the last (greatest) index of an element within the array equal to the specified value, or -1 if none is found. |
| 8. | [map()](https://www.tutorialspoint.com/typescript/typescript_array_map.htm)  Creates a new array with the results of calling a provided function on every element in this array. |
| 9. | [pop()](https://www.tutorialspoint.com/typescript/typescript_array_pop.htm)  Removes the last element from an array and returns that element. |
| 10. | [push()](https://www.tutorialspoint.com/typescript/typescript_array_push.htm)  Adds one or more elements to the end of an array and returns the new length of the array. |
| 11. | [reduce()](https://www.tutorialspoint.com/typescript/typescript_array_reduce.htm)  Apply a function simultaneously against two values of the array (from left-to-right) as to reduce it to a single value. |
| 12. | [reduceRight()](https://www.tutorialspoint.com/typescript/typescript_array_reduceright.htm)  Apply a function simultaneously against two values of the array (from right-to-left) as to reduce it to a single value. |
| 13. | [reverse()](https://www.tutorialspoint.com/typescript/typescript_array_reverse.htm)  Reverses the order of the elements of an array -- the first becomes the last, and the last becomes the first. |
| 14. | [shift()](https://www.tutorialspoint.com/typescript/typescript_array_shift.htm)  Removes the first element from an array and returns that element. |
| 15. | [slice()](https://www.tutorialspoint.com/typescript/typescript_array_slice.htm)  Extracts a section of an array and returns a new array. |
| 16. | [some()](https://www.tutorialspoint.com/typescript/typescript_array_some.htm)  Returns true if at least one element in this array satisfies the provided testing function. |
| 17. | [sort()](https://www.tutorialspoint.com/typescript/typescript_array_sort.htm)  Sorts the elements of an array. |
| 18. | [splice()](https://www.tutorialspoint.com/typescript/typescript_array_splice.htm)  Adds and/or removes elements from an array. |
| 19. | [toString()](https://www.tutorialspoint.com/typescript/typescript_array_tostring.htm)  Returns a string representing the array and its elements. |
| 20. | [unshift()](https://www.tutorialspoint.com/typescript/typescript_array_unshift.htm)  Adds one or more elements to the front of an array and returns the new length of the array. |

var arr = [12, 13, 2, 10, 2];

var array = [14, 1, 15];

var charArr = ['d', 'e', 'a', 'z', 'm'];

console.log("---arr = " + arr + "---");

console.log("array = " + array);

function isGreaterThan(x: number): boolean {

  if (x >= 5){

      return true;

  }

      return false;

}

//1. concat()

//Returns a new array comprised of this array joined with other array(s) and/or value(s).

console.log("concat() of arr + array = " + arr.concat(array));

//2. every()

//Returns true if every element in this array satisfies the provided testing function.

console.log("every() = " + arr.every(isGreaterThan));

//3. filter()

//Creates a new array with all of the elements of this array for which the provided filtering function returns true.

console.log("filter() = " + arr.filter(isGreaterThan));

// 4. forEach()

// Calls a function for each element in the array.

console.log("forEach() = ")

arr.forEach(function (value) {

    console.log(value);

});

// 5. indexOf()

// Returns the first (least) index of an element within the array equal to the specified value, or -1 if none is found.

console.log("indexOf(2) = " + arr.indexOf(2)); //should return 2

console.log("indexOf(1) = " + arr.indexOf(1)); //should return -1

// 6.   join()

// Joins all elements of an array into a string.

console.log("join() = " + arr.join());

// 7. lastIndexOf()

// Returns the last (greatest) index of an element within the array equal to the specified value, or -1 if none is found.

console.log("lastIndexOf(2) = " + arr.lastIndexOf(2)); //should return 4

// 8. map()

// Creates a new array with the results of calling a provided function on every element in this array.

console.log("map() = " + arr.map(isGreaterThan));

// 9. pop()

// Removes the last element from an array and returns that element.

console.log("pop() = " + arr.pop());

// 10. push()

// Adds one or more elements to the end of an array and returns the new length of the array.

console.log("New length of arr after push(3) = " + arr.push(3));

console.log("---arr = " + arr + "---");

// 11. reduce()

// Apply a function simultaneously against two values of the array (from left-to-right) as to reduce it to a single value.

console.log("reduce() = " + arr.reduce(function(a, b){return a + b}));

// 12. reduceRight()

// Apply a function simultaneously against two values of the array (from right-to-left) as to reduce it to a single value.

console.log("reduceRight() = " + arr.reduceRight(function(a, b){return a + b}));

// 13. reverse()

// Reverses the order of the elements of an array -- the first becomes the last, and the last becomes the first.

console.log("reverse() = " + arr.reverse());

// 14. shift()

// Removes the first element from an array and returns that element.

console.log("shift() = " + arr.shift()); //similar to dequeue

console.log("---arr = " + arr + "---");

// 15. slice()

// Extracts a section of an array and returns a new array.

console.log("slice() = " + arr.slice(0, 2)); //exclusive

// 16. some()

// Returns true if at least one element in this array satisfies the provided testing function.

console.log("some() = " + arr.some(isGreaterThan));

// 17. sort()

// Sorts the elements of an array.

console.log("---charArr = " + charArr + "---");

console.log("sort() = " + charArr.sort()); //not efficient on numbers > 9

// 18. splice()

// Adds and/or removes elements from an array.

console.log("---charArr = " + charArr + "---");

charArr.splice(0, 0, 'j');

console.log("splice(0, 0, 'j') = " + charArr);

charArr.splice(4, 1);

console.log("splice(4, 1) = " + charArr);

// 19. toString()

// Returns a string representing the array and its elements.

console.log("toString() = " + arr.toString());

// 20. unshift()

// Adds one or more elements to the front of an array and returns the new length of the array.

console.log("---arr = " + arr + "---");

console.log("unshift(1) = " + arr.unshift(1)); //similar to enqueue

console.log("after unshift(1), arr = " + arr);