**BT\_MẢNG\_ES6**

1. Tìm tất cả phần tử trong mảng ngoại trừ phần tử đầu tiên với chiều dài mảng lớn hơn một.

VD:

*[1, 2, 3] => kq = [2, 3]*

*[1] =>kq = [1]*

1. Kiểm tra xem tất cả các phần tử trong một mảng đã cho có bằng nhau hay không

Ví dụ:

[1, 2, 3, 4, 5, 6] => false

[12, 12, 12] => true

1. Viết chương trình JavaScript để loại bỏ các phần tử được chỉ định ở bên phải của một mảng phần tử đã cho

Vd:

([1, 2, 3], -2) => kq = [2, 3]

1. Viết chương trình JavaScript để loại bỏ các phần tử được chỉ định từ bên trái của một mảng phần tử đã cho.

Vd:

([1, 2, 3], 2) => [3]

1. Viết chương trình JavaScript để tạo một chuỗi mới từ một chuỗi đã cho thay đổi vị trí của các ký tự đầu tiên và cuối cùng. Độ dài chuỗi phải lớn hơn hoặc bằng 1.
2. Viết chương trình JavaScript để tính tổng của hai số nguyên đã cho. Nếu hai giá trị giống nhau, thì trả về gấp ba tổng của chúng.
3. Viết chương trình JavaScript để tạo một chuỗi mới thêm "Py" vào trước một chuỗi  
   đã cho. Nếu chuỗi đã cho bắt đầu bằng "Py" thì trả về chuỗi ban đầu
4. Viết chương trình JavaScript để xóa một ký tự tại vị trí xác định của một chuỗi đã  
   cho và trả về chuỗi mới.
5. Tìm kiếm điểm số lớn/nhỏ hơn 3 trong mảng các điểm số.
6. Lấy phần tử đầu tiên trong mảng các số có giá trị lớn/nhỏ hơn 10.
7. Đoạn code này sẽ hiển thị gì?
8. let cars = ['BMW', 'Audi', 'Mercedes', 'Ford']
9. cars.push('Toyota')
10. console.log(cars.length);
11. SV thực hiện yêu cầu sau:
12. Tạo mảng fruits gồm "apple", "orange"
13. Thêm 'Mango' vào cuối mảng
14. Thay thế giá trị ở giữa bằng 'cherry'. Nếu mảng có độ dài lẻ, code để tìm giá trị giữa.
15. Xóa phần tử đầu tiên của mảng
16. Thêm 'banana', 'pineapple' vào đầu mảng
17. SV cho biết kết quả là gì? Tại sao?
18. let array = [1, 2];
19. array.push(function () {
20. console.log(array)
21. })
22. array[2]();
23. SV thực hiện yêu cầu sau:
24. Sử dụng ham prompt nhập các giá trị vào mảng
25. Nếu giá trị nhập vào không phải là số, hay chuỗi rỗng thì nhấn “hủy” hoặc kết thúc yêu cầu
26. Tính tổng các giá trị trong mảng
27. Viết hàm loại bỏ tất cả các dấu gạch ngang, mỗi từ sau dấu gạch ngang chuyển sang chữ hoa.

Ví dụ:

1. font-size =>fontSize
2. background-color => backgroundColor
3. Viết một hàm tìm kiếm các phần tử giữa [a, b] và trả về một mảng mới trong số đó. Mảng ban đầu không đổi.
4. Viết một hàm lấy một mảng loại bỏ tất cả các giá trị nằm trong khoảng [a, b]. Mảng ban đầu có thể thay đổi.
5. Cho một mảng chuỗi. Tạo một bản sao được sắp xếp tăng dần/giảm dần. Mảng ban đầu không đổi.
6. Cho một mảng đối tượng người dùng gồm tên, tuổi, …Viết hàm chuyển nó thành một mảng tên.
7. Tạo một mảng đối tượng khác gồm mssv, fullname (gồm fname, lname) từ một mảng các đối tượng người dùng với mỗi đối tượng có mssv , fname, lname.
8. Giả sử chúng ta có mảng các đối tượng người dùng gồm {mssv, tên, tuổi…}

Tạo một đối tượng từ nó với mssv là khóa, các phần tử trong mảng là giá trị

(Chức năng này sẽ thuận lợi khi làm việc với dữ liệu máy chủ)

Ví dụ:

let users = [

{mssv: “A123”, ten: “An Huynh”, tuoi: 20},

{mssv: “A456”, ten: “Ha Tran”, tuoi: 19}

]

* Đối tượng mới

usersByMssv = {

A123: {mssv: “A123”, ten: “An Huynh”, tuoi: 20},

A456: {mssv: “A456”, ten: “Ha Tran”, tuoi: 19}

}