Array methods

-------------

1. Mutating Methods (change original array)

-------------------------------------------

let arr=[1,2,3]

arr.push(4)

console.log(arr) //[ 1, 2, 3, 4 ]

arr.pop()

console.log(arr) //[ 1, 2, 3 ]

arr.unshift(0)

console.log(arr) //[ 0, 1, 2, 3 ]

arr.shift()

console.log(arr) //[ 1, 2, 3 ]

arr.splice(0,2,100,200,300)

console.log(arr) //[ 100, 200, 300, 3 ]

arr.reverse()

console.log(arr) //[ 3, 300, 200, 100 ]

arr.sort((a,b)=>a-b);

console.log(arr)

arr.fill(10,0,2)

console.log(arr) //[ 10, 10, 200, 300 ]

arr.fill(10)

console.log(arr) //[ 10, 10, 10, 10 ]

2. 📤 Non-Mutating Methods (return new array/result)

----------------------------------------------------

let arr=[1,2,3]

console.log(arr.concat(4,5)); //[ 1, 2, 3, 4, 5 ]

console.log(arr.slice(1,2)) //[2]

console.log(arr.map(item=>item\*2)) //[ 2, 4, 6 ]

console.log(arr.filter(item=>item>=2)) //[ 2, 3 ]

console.log(arr.reduce((acc,item)=>acc+=item,0)) //6

console.log([1,[2,[3,4]]].flat(2)) //[ 1, 2, 3, 4 ]

console.log(arr.flatMap(item=>[item,item\*2])) //[ 1, 2, 2, 4, 3, 6 ]

🔍 Searching & Testing Methods

------------------------------

let arr=[3,2,5,4,1,5]

console.log(arr.includes(2)) //true

console.log(arr.indexOf(5)) //2

console.log(arr.lastIndexOf(5)) //5

console.log(arr.find(x=>x>4)) //5

console.log(arr.findIndex(x=>x>2)) //0

console.log(arr.some(x=>x>4)) //true

🧪 Utility and Conversion Methods

---------------------------------

let arr=[2,1,3,5,4,7]

console.log(arr.join("-")) //2-1-3-5-4-7

console.log(arr.toString()) //2,1,3,5,4,7

console.log(Array.isArray(arr)) //true

🔧 Creation Methods

-------------------

let arr=[2,1,3,5,4,7]

console.log(Array.from("abc")) //[ 'a', 'b', 'c' ]

console.log(Array.of(1,2,3)) //[ 1, 2, 3 ]

NOTES:-

=======

=======

content = [

("Mutating Methods (change original array)", """

1. push() - Adds elements to the end: [1,2].push(3) => [1,2,3]

2. pop() - Removes the last element: [1,2,3].pop() => [1,2]

3. unshift() - Adds elements at the start: [2,3].unshift(1) => [1,2,3]

4. shift() - Removes first element: [1,2,3].shift() => [2,3]

5. splice(start, deleteCount, ...items) - Add/remove elements: [1,2,3].splice(1,1,99) => [1,99,3]

6. reverse() - Reverses array in-place: [1,2,3].reverse() => [3,2,1]

7. sort() - Sorts elements: [2,1,10].sort((a,b)=>a-b) => [1,2,10]

8. fill(value, start, end) - Fills with value: [1,2,3].fill(0,1,2) => [1,0,3]

"""),

("Non-Mutating Methods (return new array/result)", """

9. concat() - Merges arrays: [1,2].concat([3,4]) => [1,2,3,4]

10. slice(start, end) - Extracts part of array: [1,2,3].slice(1,2) => [2]

11. map(callback) - Transforms elements: [1,2].map(x=>x\*2) => [2,4]

12. filter(callback) - Filters elements: [1,2,3].filter(x=>x>1) => [2,3]

13. reduce(callback, init) - Reduces to value: [1,2,3].reduce((a,b)=>a+b,0) => 6

14. flat(depth) - Flattens array: [1,[2,[3]]].flat(2) => [1,2,3]

15. flatMap(callback) - Maps + flattens: [1,2].flatMap(x=>[x,x\*2]) => [1,2,2,4]

"""),

("Searching & Testing Methods", """

16. includes(value) - Checks for value: [1,2,3].includes(2) => true

17. indexOf(value) - Index of first match: [1,2,3].indexOf(2) => 1

18. lastIndexOf(value) - Last index: [1,2,1].lastIndexOf(1) => 2

19. find(callback) - Finds value: [1,2,3].find(x=>x>1) => 2

20. findIndex(callback) - Index of first match: [1,2,3].findIndex(x=>x>1) => 1

21. some(callback) - Any match: [1,2,3].some(x=>x>2) => true

22. every(callback) - All match: [1,2,3].every(x=>x>0) => true

"""),

("Utility & Conversion Methods", """

23. join(separator) - Joins into string: [1,2,3].join('-') => "1-2-3"

24. toString() - Converts to string: [1,2,3].toString() => "1,2,3"

25. Array.isArray() - Checks array: Array.isArray([1,2]) => true

"""),

("Creation Methods", """

26. Array.from(iterable) - Creates from iterable: Array.from("abc") => ['a','b','c']

27. Array.of(...args) - Creates from args: Array.of(1,2,3) => [1,2,3]

""")

]