ATANAS AVKOV
FATCATTALKS.COM

Mastering JavaScript

Array Methods with Cat Breeds

A Paw-sible Approach

ATANAS AVKOV
FATCATTALKS.COM

Data Setup

```
1

const catBreeds = [Siamese', 'Persian', 'Maine Coon', 'Sphynx', 'Bengal'];
```

ATANAS AVKOV
FATCATTALKS.COM

The every() method checks if all elements in an array pass a test (provided as a function) and returns a Boolean value.

every()

The All-Checker

```
const isAllBreedsShort = catBreeds.every(breed => breed.length < 10);
console.log(isAllBreedsShort);

// Output: false, because "Maine Coon" has 10 characters



const catBreeds = [Siamese', 'Persian', 'Maine Coon', 'Sphynx', 'Bengal'];
```

ATANAS AVKOV
FATCATTALKS.COM

The some() method tests whether at least one element in the array passes the test implemented by the provided function.

some()

The Any-Checker

```
1

const hasShortBreed = catBreeds.some(breed => breed.length < 7);
console.log(hasShortBreed);

// Output: true, because "Sphynx" has only 6 characters



const catBreeds = [Siamese', 'Persian', 'Maine Coon', 'Sphynx', 'Bengal'];
```

ATANAS AVKOV
FATCATTALKS.COM

The copyWithin() method shallow-copies part of an array to another location in the same array and returns it without modifying its length.

copyWithin()

The Copier

```
1

const copiedBreeds = catBreeds.copyWithin(0, 3, 4);
console.log(copiedBreeds);

// Output: ["Sphynx", "Persian", "Maine Coon", "Sphynx", "Bengal"]



const catBreeds = [Siamese', 'Persian', 'Maine Coon', 'Sphynx', 'Bengal'];
```

ATANAS AVKOV
FATCATTALKS.COM

The valueOf() method returns the array itself. It's used more internally and often doesn't need to be called explicitly.

valueOf()

Self-Returner

```
1

console.log(catBreeds.valueOf() === catBreeds);
// Output: true




const catBreeds = [Siamese', 'Persian', 'Maine Coon', 'Sphynx', 'Bengal'];
```

ATANAS AVKOV
FATCATTALKS.COM

The forEach() method executes a function once for each array element.

forEach()

The Looper

```
1

catBreeds.forEach(breed => console.log(`I love ${breed} cats!`));

//Output I love Siamese cats!, I love Persian cats!, I love Maine Coon.....



const catBreeds = [Siamese', 'Persian', 'Maine Coon', 'Sphynx', 'Bengal'];
```

ATANAS AVKOV
FATCATTALKS.COM

The filter() method returns a new array containing elements that pass a test provided as a function.

filter()

The Selector

```
1

const longNamedBreeds = catBreeds.filter(breed => breed.length > 6);
console.log(longNamedBreeds);

// Output: ["Persian", "Maine Coon", "Bengal"]


const catBreeds = [Siamese', 'Persian', 'Maine Coon', 'Sphynx', 'Bengal'];
```

ATANAS AVKOV
FATCATTALKS.COM

The reduce() method applies a function against an accumulator and each element in the array to reduce it to a single value.

reduce()

The Accumulator

```
1

const concatenatedBreeds = catBreeds.reduce((acc, breed) => acc +
breed + ', ', '');
console.log(concatenatedBreeds);

// Output: "Siamese, Persian, Maine Coon, Sphynx, Bengal, "


const catBreeds = [Siamese', 'Persian', 'Maine Coon', 'Sphynx', 'Bengal'];
```

ATANAS AVKOV
FATCATTALKS.COM

The reduce() method applies a function against an accumulator and each element in the array to reduce it to a single value.

reduce()

The Accumulator

```
const concatenatedBreeds = catBreeds.reduce((acc, breed) => acc + breed + ', ', '');
console.log(concatenatedBreeds);

// Output: "Siamese, Persian, Maine Coon, Sphynx, Bengal, "


const catBreeds = [Siamese', 'Persian', 'Maine Coon', 'Sphynx', 'Bengal'];
```

ATANAS AVKOV
FATCATTALKS.COM

Similar to reduce(), but works from right to left.

reduceRight() The Accumulator

```
1

const reverseConcatenatedBreeds = catBreeds.reduceRight((acc, breed) => acc + breed + ', ', '');
console.log(reverseConcatenatedBreeds);

// Output: "Bengal, Sphynx, Maine Coon, Persian, Siamese, "

const catBreeds = [Siamese', 'Persian', 'Maine Coon', 'Sphynx', 'Bengal'];
```

ATANAS AVKOV
FATCATTALKS.COM

Similar to reduce(), but works from right to left.

reduceRight() The Accumulator

```
1

const reverseConcatenatedBreeds = catBreeds.reduceRight((acc, breed) => acc + breed + ', ', '');
console.log(reverseConcatenatedBreeds);

// Output: "Bengal, Sphynx, Maine Coon, Persian, Siamese, "

const catBreeds = [Siamese', 'Persian', 'Maine Coon', 'Sphynx', 'Bengal'];
```

ATANAS AVKOV
FATCATTALKS.COM

Converts the array into a string.

toString()

The String Converter

```
1

const breedsAsString = catBreeds.toString();
console.log(breedsAsString);

// Output: "Siamese,Persian,Maine Coon,Sphynx,Bengal"



const catBreeds = [Siamese', 'Persian', 'Maine Coon', 'Sphynx', 'Bengal'];
```

ATANAS AVKOV
FATCATTALKS.COM

Joins the elements into a string separated by a custom string.

join()

The Custom String Connector

```
1

const joinedBreeds = catBreeds.join(" - ");
console.log(joinedBreeds);

// Output: "Siamese - Persian - Maine Coon - Sphynx - Bengal"



const catBreeds = [Siamese', 'Persian', 'Maine Coon', 'Sphynx', 'Bengal'];
```

ATANAS AVKOV
FATCATTALKS.COM

Removes the last element of the array.

pop()

The Last Element Remover

```
const poppedBreed = catBreeds.pop();
console.log(poppedBreed);
// Output: "Bengal"
console.log(catBreeds);
// Output: ["Siamese", "Persian", "Maine Coon", "Sphynx"]


const catBreeds = [Siamese', 'Persian', 'Maine Coon', 'Sphynx', 'Bengal'];
```

ATANAS AVKOV
FATCATTALKS.COM

Adds an element to the end of the array.

push()

The Last Element Adder

```
1

catBreeds.push("Ragdoll");
console.log(catBreeds);

// Output: ["Siamese", "Persian", "Maine Coon", "Sphynx", "Ragdoll"]



const catBreeds = [Siamese', 'Persian', 'Maine Coon', 'Sphynx', 'Bengal'];
```

ATANAS AVKOV
FATCATTALKS.COM

Removes the first element of the array.

shift()

The First Element Remover

```
1


const shiftedBreed = catBreeds.shift();
console.log(shiftedBreed);
// Output: "Siamese"
console.log(catBreeds);
// Output: ["Persian", "Maine Coon", "Sphynx", "Ragdoll"]


const catBreeds = [Siamese', 'Persian', 'Maine Coon', 'Sphynx', 'Bengal'];
```

ATANAS AVKOV
FATCATTALKS.COM

Adds an element to the beginning of the array.

unshift()

The First Element Adder

```
1

catBreeds.unshift("Siamese");
console.log(catBreeds);

// Output: ["Siamese", "Persian", "Maine Coon", "Sphynx", "Ragdoll"]


const catBreeds = [Siamese', 'Persian', 'Maine Coon', 'Sphynx', 'Bengal'];
```

ATANAS AVKOV
FATCATTALKS.COM

Adds or removes elements from the array.

splice() The Element Injector/Remover

```
1
```

```
catBreeds.splice(2, 1, "Scottish Fold");
console.log(catBreeds);

// Output: ["Siamese", "Persian", "Scottish Fold", "Sphynx", "Ragdoll"]


const catBreeds = [Siamese', 'Persian', 'Maine Coon', 'Sphynx', 'Bengal'];
```

ATANAS AVKOV
FATCATTALKS.COM

Joins two or more arrays.

concat()

The Array Joiner

```
1

const moreBreeds = ["Abyssinian", "Manx"];
const allBreeds = catBreeds.concat(moreBreeds);
console.log(allBreeds);

// Output: ["Siamese", "Persian", "Scottish Fold", "Sphynx", "Ragdoll",
"Abyssinian", "Manx"]

const catBreeds = [Siamese', 'Persian', 'Maine Coon', 'Sphynx', 'Bengal'];
```

ATANAS AVKOV
FATCATTALKS.COM

Returns a shallow copy of a portion of an array.

slice()

The Sub-array Extractor

```
1

const someBreeds = catBreeds.slice(1, 4);
console.log(someBreeds);

// Output: ["Persian", "Scottish Fold", "Sphynx"]


const catBreeds = [Siamese', 'Persian', 'Maine Coon', 'Sphynx', 'Bengal'];
```

ATANAS AVKOV
FATCATTALKS.COM

Sorts the elements of an array.

sort()

The Sorter

```
1

const sortedBreeds = catBreeds.sort();
console.log(sortedBreeds);

 // Output: ["Persian", "Ragdoll", "Scottish Fold", "Siamese", "Sphynx"]


const catBreeds = [Siamese', 'Persian', 'Maine Coon', 'Sphynx', 'Bengal'];
```

ATANAS AVKOV
FATCATTALKS.COM

Reverses the elements of an array.

reverse()

The Reverser

```
1

const reversedBreeds = catBreeds.reverse();
console.log(reversedBreeds);

// Output: ["Sphynx", "Siamese", "Scottish Fold", "Ragdoll", "Persian"]



const catBreeds = [Siamese', 'Persian', 'Maine Coon', 'Sphynx', 'Bengal'];
```