Cheatography

stathw-js-cheatsheet Cheat Sheet

by Boge (username1) via cheatography.com/195153/cs/40822/

Array

Declaration and initialization

```
let array = ['cy ber', 'insec urity', 'sapie nza'];
let also = new Array(3);
```

Adding an element

```
array[3] = 'stati stics';
```

Setting a value

```
array[1] = 'secur ity';
```

Deleting an element

```
delete array[3];
```

Looping

```
array.f or Eac h(f unc tio n(item) {
    consol e.l og( item);
});
```

Checking the existence of a value #1

```
array.i nc lud es( 'cy ber');
```

Checking the existence of a value #2

```
array.i nd exO f(' sta tis tics') !== -1;
```

Checking the existence of a value #3

```
for (let i = 0; i < array.l ength; i++) {
    if (array[i] === search Value) {
        found = true;
        break;
    }
}
```

Checking the existence of a value #4

```
array.f in d(item => item === search Value) !== undefin ed;
```

Arrays are also used to implement List, Queue, Stack. For these data structures I'll create a JS class that exposes useful methods.

LinkedList

List

Creation

```
const list = new List();
```

Adding an element

```
list.a dd(1);
list.a ddA tHe ad(0);
```

Removing an element

```
list.r emo ve(2);
```

Setting a value

```
list.s et(0, 1337);
```

Checking existence of a value

```
list.c ont ain s(1);
```

Looping

```
for(let i = 0; i < list.s ize(); i++)
{
    consol e.l og( lis t.g et(i));
});
```

The type SortedList has the same methods, but ensures that the values are sorted.

Dictionary

Creation

```
let dict = new Dictio nary();
```

Adding an key-value pair

```
dict.a dd( 'fr equ ency', 20);
```

Setting a value for a key

```
dict.s et( 'pe rce nta ge', 40.10 );
```

Removing a key

```
dict.r emo ve( 'pe rce nta ge');
```

Checking existence of a key

```
dict.h asK ey( 'fr equ ency');
```

Checking existence of a value

```
dict.h asV alu e(20);
```

HashSet

Queue

Creation

```
let queue = new Queue();
```

Adding an element

```
queue.e nq ueu e(1);
```

Fetch of the first element

```
queue.f ro nt();
```

Removing an element

```
queue.d eq ueue();
```

Checking the existence of an element

```
queue.c on tai ns(7);
```

Stack

Creation

```
let stack = new Stack();
```

Adding an element

```
stack.p us h(1);
```

Fetching top of stack

```
stack.p eek();
```

Removing an element

```
stack.p op();
```

SortedSet

Creation

```
let sortedSet = new Sorted Set();
```

Adding an element

```
sorted Set.ad d(3);
```

Removing an element

```
sorted Set.de let e(3);
```

Checking the existence of an element

```
sorted Set.ha s(1);
```

Looping

```
for(el ement in sorted Set.to Arra ) {
    consol e.l og( ele ment);
});
```

Creation

```
let linkedList = new Linked List();
```

Adding an element

```
linked Lis t.a ppe nd(3);
linked Lis t.p rep end(1);
linked Lis t.a ddAt(1, 3);
```

Fetch of the head value

```
linked Lis t.f etc hHe ad();
```

Get the value of a node

```
linked Lis t.g etA t(2);
```

Removing an element

```
linked Lis t.r emo veA t(1);
linked Lis t.d ele te(7);
```

Check existence of a value

```
linked Lis t.c ont ain s(7);
```

Looping

```
for (const element of linked List)
{
    consol e.l og( ele ment);
}
```

Creation

```
let hashset = new HashSet();
```

Adding an element

```
hashse t.a dd(1);
```

Removing an element

```
hashse t.d ele te(1);
```

Checking the existence of an element

```
hashse t.h as(3);
```

Looping

```
let keys = hashse t.v alu es();
for(let i = 0; i < keys.l ength; i++)
{
    consol e.l og( key s[i]);
});
```

The HashSet is pretty similar to the Dictionary, but it doesn't have values; it just uses the keys.

By **Boge** (username1)

cheatography.com/username1/

Published 16th October, 2023.
Last updated 16th October, 2023.