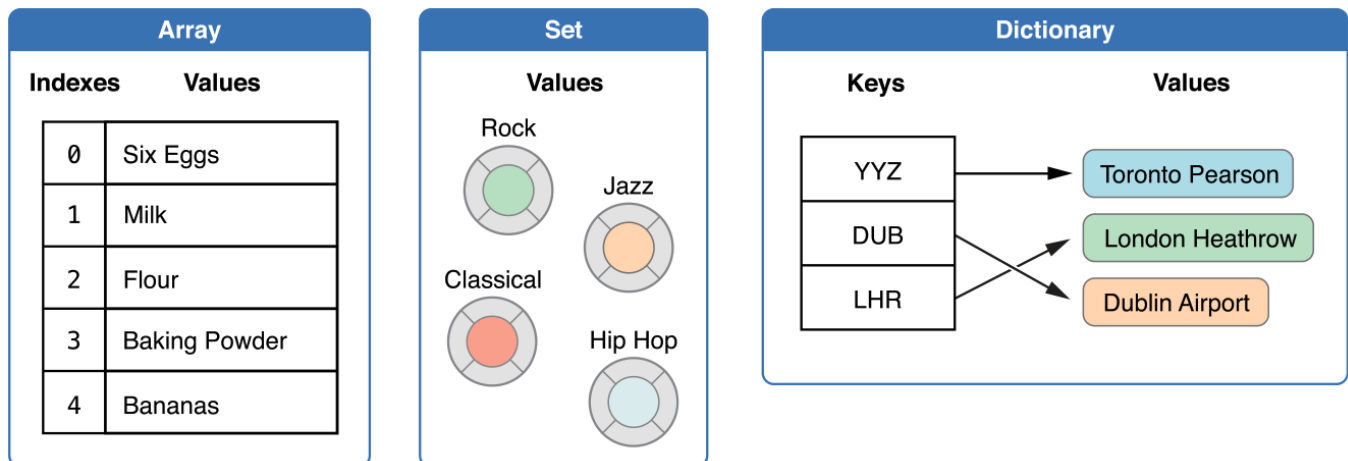


# JS built-in object: Set



Source: <https://medium.com/@mdibrahimhassan/linked-list-in-swift-3338bede5626>

## AGENDA

1. Tổng quan về Set
2. Một số phương thức của Set
3. Ví dụ demo

## 1. Tổng quan về Set

- The Set object lets you store unique values of any type, whether primitive values or object references.
- You can iterate through the elements of a set in insertion order.
- A value in the Set may only occur once
- Some examples:
  - Remove duplicated numbers from [1, 2, 3, 2, 3] --> [1, 2, 3]
  - Remove duplicated letters from string 'aabbcc' --> 'abc'
  - ...

## 2. Một số phương thức của Set

#	Name	Desc
1	<code>new Set()</code>	Khởi tạo Set
2	<code>size</code> (thuộc tính, ko phải hàm)	Lấy size của set
3	<code>add(value)</code>	Add value mới vào set
4	<code>has(value)</code>	Kiểm tra set có key không?
5	<code>delete(value)</code>	Xoá một value
6	<code>clear()</code>	Xoá tất cả values
7	<code>keys()</code>	Alias của <b><code>values()</code></b>
8	<code>values()</code>	LOOP qua tất cả values
9	<code>entries()</code>	LOOP qua tất cả cặp <code>[value, value]</code>
10	<code>forEach((value) =&gt; {})</code>	LOOP qua tất cả values

Source: [https://developer.mozilla.org/en-US/docs/Web/JavaScript/Reference/Global\\_Objects/Set](https://developer.mozilla.org/en-US/docs/Web/JavaScript/Reference/Global_Objects/Set)

### Khởi tạo Set

```
const set = new Set();  
// Set(0) {}  
  
const set = new Set([1, 1, 2, 2, 3, 3]);  
// Set(3) { 1, 2, 3}  
  
const set = new Set('aabbcc');  
// Set(3) { 'a', 'b', 'c' }  
  
const set = new Set(new Set([1, 2, 3]))  
// Set(3) { 1, 2, 3 }
```

```
> new Set()  
< ▶ Set(0) {}  
  
> new Set([1, 1, 2, 2, 3, 3])  
< ▶ Set(3) {1, 2, 3}  
  
> new Set('aabbcc')  
< ▶ Set(3) {"a", "b", "c"}  
  
> new Set(new Set([1, 2, 3]))  
< ▶ Set(3) {1, 2, 3}
```

## Add/Delete

```
const set = new Set();
set.size; // 0

set.add(1);
set.add(2);
set.size; // 2

set.has(1); // true
set.has(3); // false

set.delete(1); // true
set.delete(3); // false as 3 is not existed

set.clear();
set.size; // 0
```

## Loop through set

```
const numberSet = new Set([1, 2, 3, 3]);

for (const number of numberSet) {
  console.log(number); // 1, 2, 3
}

numberSet.forEach(number => console.log(number)); // 1, 2, 3
```

## 3. Ví dụ demo

### Remove duplicated numbers

```
function uniqueNumbers(numberList) {
  if (!Array.isArray(numberList) || numberList.length === 0) return [];

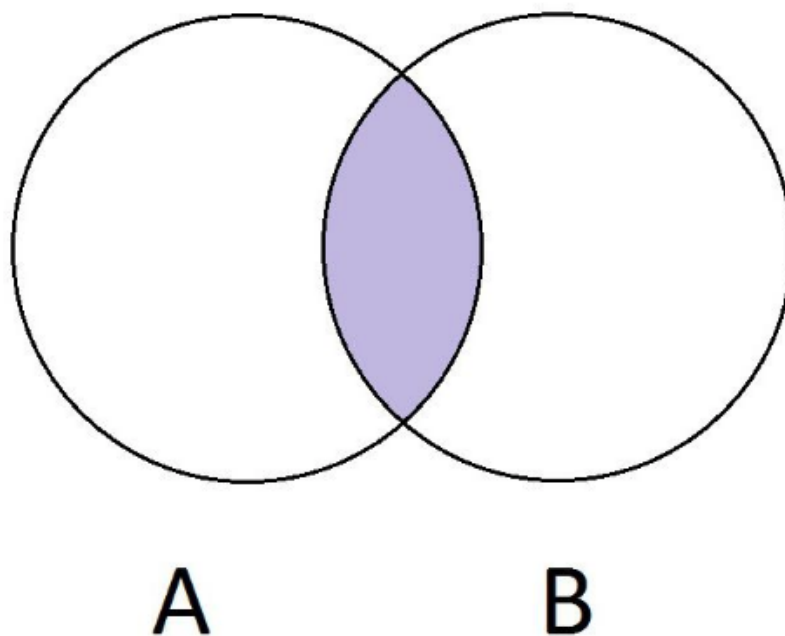
  const uniqueNumberSet = new Set(numberList);
  // return Array.from(uniqueNumberSet);
  return [...uniqueNumberSet];
}

console.log(uniqueNumbers([1, 1, 3, 3, 2, 2, 2]));
```

## Remove duplicated letters

```
function uniqueLetters(str) {  
  if (str.length === 0) return '';  
  
  const uniqueLetterSet = new Set(str);  
  return [...uniqueLetterSet].join('');  
}  
  
console.log(uniqueLetters('abcabcabc'));
```

## Intersection



Source: <https://www.thoughtco.com/intersection-in-set-theory-3126587>

```
function getIntersectionSet(set1, set2) {  
  const intersectionSet = new Set();  
  for (const item of set1) {  
    if (set2.has(item)) intersectionSet.add(item);  
  }  
  
  return intersectionSet;  
}  
  
const set1 = new Set([1, 2, 3]);  
const set2 = new Set([2, 3, 4, 5]);  
console.log(getIntersectionSet(set1, set2));
```

---

## Khoá học Javascript cho người mới bắt đầu 2021 🎉

- Tác giả: **Hậu Nguyễn** - Founder Easy Frontend
- Khoá học chỉ được published trên Udemy, không thông qua trung gian.
- Khoá học không bán dạng videos upload trên Google Drive hay bất cứ hình thức nào tương tự.
- Khoá học có nhóm discord để hỗ trợ trong quá trình học tập.

☎️ Liên hệ tác giả để được hỗ trợ:

- ✅ Facebook: <https://www.facebook.com/nvhauesmn/>
- ✅ Fanpage: <https://www.facebook.com/learn.easyfrontend>
- ✅ Youtube Channel: <https://www.youtube.com/easyfrontend>