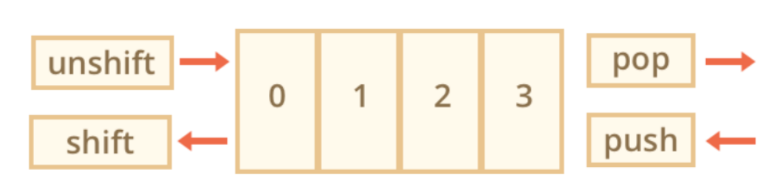
**Learning Objectives**

* To learn about common methods of array.

**Learning Contents**

* In JavaScript, in addition to Object, Array is the most common data type. An array is an order list of data. In JavaScript, each entry in an array can save any type of data. In other words, we can use the first entry of an array to store strings, the second entry to store numbers and the third entry to store objects.Besides, length of an array in JavaScript can be adjusted dynamically, or rather it grows automatically with addition of data in order to include new data.
* There are two basic forms to create an array. The first one is to use array constructor function in JavaScript. For example:
* **var fruits = new Array ();**
* If we know the number of entries to be saved in the array in advance, we can input the parameter to the constructor function. For example, the following codes will create an array with a length being 10.
* **var fruits = new Array (10);**
* The second method of creating an array is array literal representation method. As for array literal, it means we list out every data item in the array directly, including separation of different data items in square brackets with a comma. For example:
* **var fruits = ['apple', 'pear', 'peach']; *// Create an array containing three strings***
* **var names = []; *// Create an empty array***
* **var values = [1, 2, ,]; *// Do not do it! It will create an array containing 2 items or 4 items.***
* When we read an array, we can use square brackets and provide data index based on 0. For example:
* **var fruits = ['apple', 'pear', 'peach'];**
* **console.log(fruits[0]); *// Display the first entry - 'apple'***
* **fruits[1] = 'grape'; *// Modify the second entry***
* **console.log(fruits[1]); *// Display the second entry - 'grape'***
* **fruits[3] = 'banana'; *// Add the fourth entry***
* **console.log(fruits[3]); *// Display the fourth entry - 'banana'***
* Index in the square brackets represents the value to be accessed. If the index is less than the number of items in the array, the value of corresponding item will be returned just as fruits [0](https://school.thoughtworks.cn/learn/program-center/student/index.html) in the above case will return 'apple'. The same syntax is used to set value of certain item in the array, but it will substitute value of designated position. If the index setting certain value is greater than current length of the array just like the fruits[3](https://school.thoughtworks.cn/learn/program-center/student/index.html) in the above case, the array will automatically increase to the length of the index value plus 1 (in the above case, the index is 3, so the length of the array is 4.)
* Number of items in the array is saved in the length attribute of the array. This attribute will eventually return 0 or a greater value. For example:
* **var fruits = ['apple', 'pear', 'peach'];**
* **console.log(fruits.length); *// 3***
* **var colors = [];**
* **console.log(colors.length); *// 0***
* length attribute in the array is very characteristic. It is **not readonly**. Hence, we can also remove data item at the end of the array by setting this attribute. For example:
* **var fruits = ['apple', 'pear', 'peach'];**
* **fruits.length = 2;**
* **console.log(fruits[2]); *// undefined – It has the equivalent effect of deleting the third item from the array.***
* Common methods in JavaScript:
  + push() method
    - Add a new element at the end of the array. For example:
    - **var fruits = ['apple', 'pear', 'peach'];**
    - **fruits.push('banana');**
    - **console.log(fruits); *// ['apple', 'pear', 'peach', 'banana']***
    - pop() method
      * Remove an element from the end of the array. For example:
      * **var fruits = ['apple', 'pear', 'peach'];**
      * **fruits.pop();**
      * **console.log(fruits); *// ['apple', 'pear']***
    - unshift() method
      * Add a new element at the front of the array. For example:
      * **var fruits = ['apple', 'pear', 'peach'];**
      * **fruits.unshift('banana');**
      * **console.log(fruits); *// ['banana', 'apple', 'pear', 'peach']***
    - shift() method
      * Remove an element at the front of the array. For example:
      * **var fruits = ['apple', 'pear', 'peach'];**
      * **fruits.shift();**
      * **console.log(fruits); *// ['pear', 'peach']***
    - The above four methods are basic operations for us to add or delete elements in the array.



* + - indexOf() method
      * Find index value of certain data item in the array. For example:
      * **var fruits = ['apple', 'pear', 'peach'];**
      * **console.log(fruits.indexOf('pear')); *// 1***

**Recommended Resources**

* JavaScript Array - MDN（<https://developer.mozilla.org/zh-CN/docs/Web/JavaScript/Reference/Global_Objects/Array>）
* Comprehensive Summarization of js Array Methods（<http://www.jianshu.com/p/a339893df4aa>）
* javascript：Summarization of Array（<https://segmentfault.com/a/1190000002957308>）