An array is a special variable, which can hold more than one value at a time.

If you have a list of items (a list of car names, for example), storing the cars in single variables could look like this:

```
let car1 = "Saab";
let car2 = "Volvo";
let car3 = "BMW";
```

Accessing Array Elements

You access an array element by referring to the index number:

```
const cars = ["Saab", "Volvo", "BMW"];
let x = cars[0];    // x = "Saab"
```

---------------------------\

```
<!DOCTYPE html>
<html>
<body>

<h2>JavaScript Array Methods</h2>
<h2>toString()</h2>
<p>The toString() method returns an array as a comma separated string:</p>

<p id="demo"></p>

<script>
const fruits = ["Banana", "Orange", "Apple", "Mango"];
document.getElementById("demo").innerHTML = fruits.toString();
</script>

</body>
</html>
```

-----------------------------------\

```
<!DOCTYPE html>
```

```
<html>
<body>

<h2>JavaScript Array Methods</h2>
<h2>join()</h2>
<p>The join() method joins array elements into a string.</p>
<p>It this example we have used " * " as a separator between the elements:</p>

<p id="demo"></p>

<script>
const fruits = ["Banana", "Orange", "Apple", "Mango"];
document.getElementById("demo").innerHTML = fruits.join(" * ");
</script>

</body>
</html>

------------------------------------------/
<!DOCTYPE html>
<html>
<body>

<h2>JavaScript Array Methods</h2>
<h2>pop()</h2>
<p>The pop() method removes the last element from an array.</p>

<p id="demo1"></p>
<p id="demo2"></p>

<script>
const fruits = ["Banana", "Orange", "Apple", "Mango"];
document.getElementById("demo1").innerHTML = fruits;
fruits.pop();
document.getElementById("demo2").innerHTML = fruits;
</script>

</body>
</html>
```

```
-----------------------------------------/
<!DOCTYPE html>
<html>
<body>

<h2>JavaScript Array Methods</h2>
<h2>push()</h2>
<p>The push() method appends a new element to an array.</p>

<button onclick="myFunction()">Try it</button>
<p id="demo"></p>

<script>
const fruits = ["Banana", "Orange", "Apple", "Mango"];
document.getElementById("demo").innerHTML = fruits;

function myFunction() {
  fruits.push("Kiwi");
  document.getElementById("demo").innerHTML = fruits;
}
</script>

</body>
</html>
```

```
-------------------------------------------\
<!DOCTYPE html>
<html>
<body>


<h2>JavaScript Array Methods</h2>
<h2>shift()</h2>
<p>The shift() method removes the first element of an array (and "shifts" all other elements to
the left):</p>


<p id="demo1"></p>
<p id="demo2"></p>


<script>
const fruits = ["Banana", "Orange", "Apple", "Mango"];
document.getElementById("demo1").innerHTML = fruits;
fruits.shift();
document.getElementById("demo2").innerHTML = fruits;
</script>


</body>
</html>
```

```
-----------------------------------------------\
<!DOCTYPE html>
<html>
<body>

<h2>JavaScript Array Methods</h2>
<h2>shift()</h2>
<p>The shift() method returns the element that was shifted out.</p>

<p id="demo1"></p>
<p id="demo2"></p>
<p id="demo3"></p>

<script>
const fruits = ["Banana", "Orange", "Apple", "Mango"];
document.getElementById("demo1").innerHTML = fruits;
document.getElementById("demo2").innerHTML = fruits.shift();
document.getElementById("demo3").innerHTML = fruits;
</script>

</body>
</html>
```

```
-------------------------------------------\

<!DOCTYPE html>
<html>
<body>

<h2>JavaScript Array Methods</h2>
<h2>unshift()</h2>
<p>The unshift() method adds new elements to the beginning of an array.</p>

<button onclick="myFunction()">Try it</button>
<p id="demo"></p>

<script>
const fruits = ["Banana", "Orange", "Apple", "Mango"];
document.getElementById("demo").innerHTML = fruits;

function myFunction() {
  fruits.unshift("Lemon");
  document.getElementById("demo").innerHTML = fruits;
}
</script>

</body>
</html>
```

```
-------------------------------------------------\
<!DOCTYPE html>
<html>
<body>

<h2>JavaScript Array Methods</h2>
<p>Array elements are accessed using their index number:</p>

<p id="demo1"></p>
<p id="demo2"></p>

<script>
const fruits = ["Banana", "Orange", "Apple", "Mango"];
document.getElementById("demo1").innerHTML = fruits;
fruits[0] = "Kiwi";
document.getElementById("demo2").innerHTML = fruits;
</script>

</body>
</html>
```

```
-----------------------------------------------\


<!DOCTYPE html>
<html>
<body>


<h2>JavaScript Array Methods</h2>
<p>The length property provides an easy way to append new elements to an array without using the push() method.</p>


<button onclick="myFunction()">Try it</button>
<p id="demo"></p>


<script>
const fruits = ["Banana", "Orange", "Apple", "Mango"];
document.getElementById("demo").innerHTML = fruits;


function myFunction() {
  fruits[fruits.length] = "Kiwi";
  document.getElementById("demo").innerHTML = fruits;
}
</script>


</body>
</html>
-------------------------------------------------\
```

```
<!DOCTYPE html>
<html>
<body>

<h2>JavaScript Array Methods</h2>
<p>Deleting elements leaves undefined holes in an array.</p>

<p id="demo1"></p>
<p id="demo2"></p>

<script>
const fruits = ["Banana", "Orange", "Apple", "Mango"];

document.getElementById("demo1").innerHTML =
"The first fruit is: " + fruits[0];

delete fruits[0];

document.getElementById("demo2").innerHTML =
"The first fruit is: " + fruits[0];
</script>

</body>
</html>
```

--\

```
<!DOCTYPE html>
<html>
<body>

<h2>JavaScript Array Methods</h2>
<h2>splice()</h2>
<p>The splice() method adds new elements to an array.</p>

<button onclick="myFunction()">Try it</button>

<p id="demo1"></p>
<p id="demo2"></p>

<script>
const fruits = ["Banana", "Orange", "Apple", "Mango"];
document.getElementById("demo1").innerHTML = "Original Array:<br>" + fruits;

function myFunction() {
  fruits.splice(2, 0, "Lemon", "Kiwi");
  document.getElementById("demo2").innerHTML = "New Array:<br>" + fruits;
}
</script>

</body>
</html>
```

```
-----------------------------------------------\
<!DOCTYPE html>
<html>
<body>

<h2>JavaScript Array Methods</h2>
<h2>splice()</h2>
<p>The splice() methods can be used to remove array elements.</p>

<button onclick="myFunction()">Try it</button>

<p id="demo"></p>

<script>
const fruits = ["Banana", "Orange", "Apple", "Mango"];
document.getElementById("demo").innerHTML = fruits;
function myFunction() {
  fruits.splice(0, 1);
  document.getElementById("demo").innerHTML = fruits;
}
</script>

</body>
</html>
```

```
-----------------------------------------\
<!DOCTYPE html>
<html>
<body>

<h2>JavaScript Array Methods</h2>
<h2>concat()</h2>
<p>The concat() method is used to merge (concatenate) arrays:</p>

<p id="demo"></p>

<script>
const myGirls = ["Cecilie", "Lone"];
const myBoys = ["Emil", "Tobias", "Linus"];
const myChildren = myGirls.concat(myBoys);

document.getElementById("demo").innerHTML = myChildren;
</script>

</body>
</html>
```

```
-----------------------------------------------\
<!DOCTYPE html>
<html>
<body>

<h2>JavaScript Array Methods</h2>
<h2>concat()</h2>
<p>The concat() method is used to merge (concatenate) arrays:</p>

<p id="demo"></p>

<script>
const array1 = ["Cecilie", "Lone"];
const array2 = ["Emil", "Tobias", "Linus"];
const array3 = ["Robin", "Morgan"];

const myChildren = array1.concat(array2, array3);

document.getElementById("demo").innerHTML = myChildren;
</script>

</body>
</html>
```

```
----------------------------------------\
<!DOCTYPE html>
<html>
<body>

<h2>JavaScript Array Methods</h2>
<h2>concat()</h2>
<p>The concat() method can also merge string values to arrays:</p>

<p id="demo"></p>

<script>
const myArray = ["Emil", "Tobias", "Linus"];
const myChildren = myArray.concat("Peter");
document.getElementById("demo").innerHTML = myChildren;
</script>

</body>
</html>
```

```
------------------------------\

<!DOCTYPE html>
<html>
<body>

<h2>JavaScript Array Methods</h2>
<h2>slice()</h2>
<p>This example slices out a part of an array starting from array element 1 ("Orange"):</p>

<p id="demo"></p>

<script>
const fruits = ["Banana", "Orange", "Lemon", "Apple", "Mango"];
const citrus = fruits.slice(1);
document.getElementById("demo").innerHTML = fruits + "<br><br>" + citrus;
</script>

</body>
</html>
-----------------------------------\
```

```
<!DOCTYPE html>
<html>
<body>

<h2>JavaScript Array Methods</h2>
<h2>slice()</h2>
<p>When the slice() method is given two arguments, it selects array elements from the start argument, and up to (but not included) the end argument:</p>

<p id="demo"></p>

<script>
const fruits = ["Banana", "Orange", "Lemon", "Apple", "Mango"];
const citrus = fruits.slice(1,3);
document.getElementById("demo").innerHTML = fruits + "<br><br>" + citrus;
</script>

</body>
</html>
```

```
------------------------------\

aARRAY SORTING

----------------------------------\
<!DOCTYPE html>
<html>
<body>

<h2>JavaScript Array Sort</h2>
<p>The sort() method sorts an array alphabetically.</p>

<button onclick="myFunction()">Try it</button>

<p id="demo"></p>

<script>
const fruits = ["Banana", "Orange", "Apple", "Mango"];
document.getElementById("demo").innerHTML = fruits;

function myFunction() {
  fruits.sort();
  document.getElementById("demo").innerHTML = fruits;
}
</script>

</body>
</html>
```

```
-----------------------------------------\
<!DOCTYPE html>
<html>
<body>

<h2>JavaScript Array Sort Reverse</h2>

<p>The reverse() method reverses the elements in an array.</p>
<p>By combining sort() and reverse() you can sort an array in descending order.</p>
<button onclick="myFunction()">Try it</button>

<p id="demo"></p>

<script>
// Create and display an array:
const fruits = ["Banana", "Orange", "Apple", "Mango"];
document.getElementById("demo").innerHTML = fruits;

function myFunction() {
  // First sort the array
  fruits.sort();
  // Then reverse it:
  fruits.reverse();
  document.getElementById("demo").innerHTML = fruits;
}
</script>

</body>
</html>
```

```
----------------------------------------\
<!DOCTYPE html>
<html>
<body>

<h2>JavaScript Array Sort</h2>
<p>Click the button to sort the array in ascending order.</p>

<button onclick="myFunction()">Try it</button>
<p id="demo"></p>

<script>
const points = [40, 100, 1, 5, 25, 10];
document.getElementById("demo").innerHTML = points;

function myFunction() {
  points.sort(function(a, b){return a - b});
  document.getElementById("demo").innerHTML = points;
}
</script>

</body>
</html>
```

```
---------------------------------------\
<!DOCTYPE html>
<html>
<body>

<h2>JavaScript Array Sort</h2>

<p>Click the buttons to sort the array alphabetically or numerically.</p>

<button onclick="myFunction1()">Sort Alphabetically</button>
<button onclick="myFunction2()">Sort Numerically</button>

<p id="demo"></p>

<script>
const points = [40, 100, 1, 5, 25, 10];
document.getElementById("demo").innerHTML = points;

function myFunction1() {
  points.sort();
  document.getElementById("demo").innerHTML = points;
}
function myFunction2() {
  points.sort(function(a, b){return a - b});
  document.getElementById("demo").innerHTML = points;
}
</script>

</body>
</html>
```

----------\

```
<!DOCTYPE html>
<html>
<body>

<h2>JavaScript Array Sort</h2>
<p>Click the button (again and again) to sort the array in random order.</p>

<button onclick="myFunction()">Try it</button>
<p id="demo"></p>

<script>
const points = [40, 100, 1, 5, 25, 10];
document.getElementById("demo").innerHTML = points;

function myFunction() {
  points.sort(function(a, b){return 0.5 - Math.random()});
  document.getElementById("demo").innerHTML = points;
}
</script>

</body>
</html>
```

```
-----------------------------------------------\
<!DOCTYPE html>
<html>
<body>

<h2>JavaScript Array Sort</h2>
<h3>The Fisher Yates Method</h3>
<p>Click the button (again and again) to sort the array in random order.</p>

<button onclick="myFunction()">Try it</button>
<p id="demo"></p>

<script>
const points = [40, 100, 1, 5, 25, 10];
document.getElementById("demo").innerHTML = points;

function myFunction() {
  for (let i = points.length -1; i > 0; i--) {
    let j = Math.floor(Math.random() * i)
    let k = points[i]
    points[i] = points[j]
    points[j] = k
  }
  document.getElementById("demo").innerHTML = points;
}
</script>

</body>
</html>
```

```
-----------------------------------\
<!DOCTYPE html>
<html>
<body>

<h2>JavaScript Array Sort</h2>
<p>The lowest number is <span id="demo"></span>.</p>

<script>
const points = [40, 100, 1, 5, 25, 10];
points.sort(function(a, b){return a-b});
document.getElementById("demo").innerHTML = points[0];
</script>

</body>
</html>

-----------------------------------\
<!DOCTYPE html>
<html>
<body>

<h2>JavaScript Array Sort</h2>
<p>The highest number is <span id="demo"></span>.</p>

<script>
const points = [40, 100, 1, 5, 25, 10];
points.sort(function(a, b){return b-a});
document.getElementById("demo").innerHTML = points[0];
</script>

</body>
</html>
```

```
-----------------------------------\
<!DOCTYPE html>
<html>
<body>

<h2>JavaScript Array Sort</h2>
<p>The highest number is <span id="demo"></span>.</p>

<script>
const points = [40, 100, 1, 5, 25, 10];
document.getElementById("demo").innerHTML = myArrayMax(points);

function myArrayMax(arr) {
  return Math.max.apply(null, arr);
}
</script>

</body>
</html>
```

```
-------------------------------------\

<!DOCTYPE html>
<html>
<body>

<h2>JavaScript Array Sort</h2>

<p>The lowest number is <span id="demo"></span>.</p>

<script>
const points = [40, 100, 1, 5, 25, 10];
document.getElementById("demo").innerHTML = myArrayMin(points);

function myArrayMin(arr) {
  return Math.min.apply(null, arr);
}
</script>

</body>
</html>
```

```
-----------------------------------------------\
<!DOCTYPE html>
<html>
<body>

<h2>JavaScript Array Sort</h2>
<p>The highest number is <span id="demo"></span>.</p>

<script>
const points = [40, 100, 1, 5, 25, 10];
document.getElementById("demo").innerHTML = myArrayMax(points);

function myArrayMax(arr) {
  let len = arr.length;
  let max = -Infinity;
  while (len--) {
    if (arr[len] > max) {
      max = arr[len];
    }
  }
  return max;
}
</script>

</body>
</html>
```

```
-----------------------------------------\
<!DOCTYPE html>
<html>
<body>

<h2>JavaScript Array Sort</h2>
<p>The lowest number is <span id="demo"></span>.</p>

<script>
const points = [40, 100, 1, 5, 25, 10];
document.getElementById("demo").innerHTML = myArrayMin(points);

function myArrayMin(arr) {
  let len = arr.length;
  let min = Infinity;
  while (len--) {
    if (arr[len] < min) {
      min = arr[len];
    }
  }
  return min;
}
</script>

</body>
</html>
```

```
-----------------------------------------------\
<!DOCTYPE html>
<html>
<body>

<h2>JavaScript Array Sort</h2>
<p>Click the buttons to sort car objects on age.</p>

<button onclick="myFunction()">Sort</button>
<p id="demo"></p>

<script>
const cars = [
  {type:"Volvo", year:2016},
  {type:"Saab", year:2001},
  {type:"BMW", year:2010}
];

displayCars();

function myFunction() {
  cars.sort(function(a, b){return a.year - b.year});
  displayCars();
}

function displayCars() {
  document.getElementById("demo").innerHTML =
  cars[0].type + " " + cars[0].year + "<br>" +
  cars[1].type + " " + cars[1].year + "<br>" +
  cars[2].type + " " + cars[2].year;
}
</script>

</body>
</html>
```

```
-----------------------------------------------\
<!DOCTYPE html>
<html>
<body>

<h2>JavaScript Array Sort</h2>
<p>Click the buttons to sort car objects on type.</p>

<button onclick="myFunction()">Sort</button>
<p id="demo"></p>

<script>
const cars = [
  {type:"Volvo", year:2016},
  {type:"Saab", year:2001},
  {type:"BMW", year:2010}
];

displayCars();

function myFunction() {
  cars.sort(function(a, b){
    let x = a.type.toLowerCase();
    let y = b.type.toLowerCase();
    if (x < y) {return -1;}
    if (x > y) {return 1;}
    return 0;
  });
  displayCars();
}

function displayCars() {
  document.getElementById("demo").innerHTML =
  cars[0].type + " " + cars[0].year + "<br>" +
  cars[1].type + " " + cars[1].year + "<br>" +
  cars[2].type + " " + cars[2].year;
}
</script>
```

```
</body>
</html>
```