

Table of Contents

1.1	2
1.2	5
1.3	5
2.1	6
2.2	8
3.1	9
3.2	13
3.3	17

1.1

Screenshot of output of JavaScript array in a table format (Matthus, 2019).

Cars in my Garage	
myArray[0]	BMW M4
myArray[1]	Nissan Versa
myArray[2]	Honda Civic
myArray[3]	Hyundai Elantra Hybrid
myArray[4]	Nissan Magnite
myArray[5]	Hyundai Alcazar

Code:

```
<!DOCTYPE html>
```

```
<html lang="en">
```

```
<head>
```

```
  <meta charset="UTF-8">
```

```
  <meta http-equiv="X-UA-Compatible" content="IE=edge">
```

```
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
```

```
  <title>Neighbor's Cars</title>
```

```
<!-- css styling within html -->
```

```
<style>
```

```
  table{
```

```
    border: 1px solid black;
```

```
  }
```

```

tbody tr td {
    border: 1px solid black;
}
</style>

</head>
<body>
    <table style="border-spacing: 0px; margin-left: auto; margin-right: auto; margin-top: 150px;
margin-bottom: auto;">
        <tr class=" ">
            <th>Cars in my Garage</th>
        </tr>

        <tbody id=" tableView ">

            </tbody>
        </table>

<script>
    let myCars = [
        {'position':'myArray[0]', 'carType':'BMW M4'},
        {'position':'myArray[1]', 'carType':'Nissan Versa'},
        {'position':'myArray[2]', 'carType':'Honda Civic'},
        {'position':'myArray[3]', 'carType':'Hyundai Elantra Hybrid'},
        {'position':'myArray[4]', 'carType':'Nissan Magnite'},
        {'position':'myArray[5]', 'carType':'Hyundai Alcazar'},
    ]

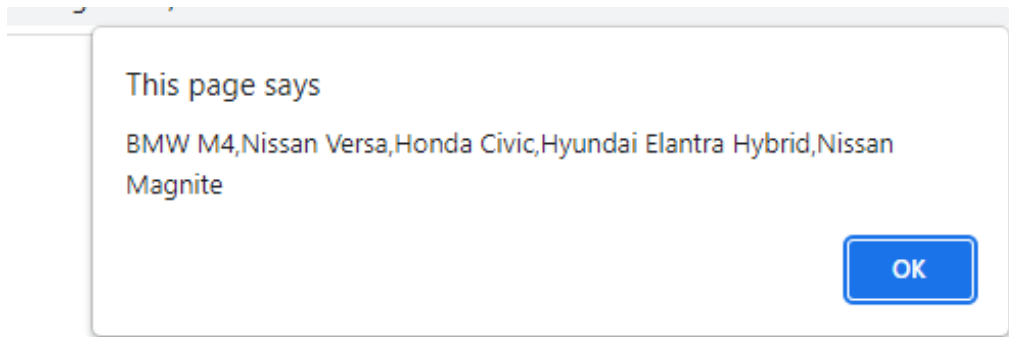
```

```
tableFunction(myCars)
```

```
function tableFunction (data){  
  var table = document.getElementById('tableView')  
  
  for (var i = 0; i < data.length; i++){  
    var row = `<tr>  
  
      <td>${data[i].position}</td>  
  
      <td>${data[i].carType}</td>  
  
    </tr>`  
    table.innerHTML += row  
  
  }  
}
```

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

1.2



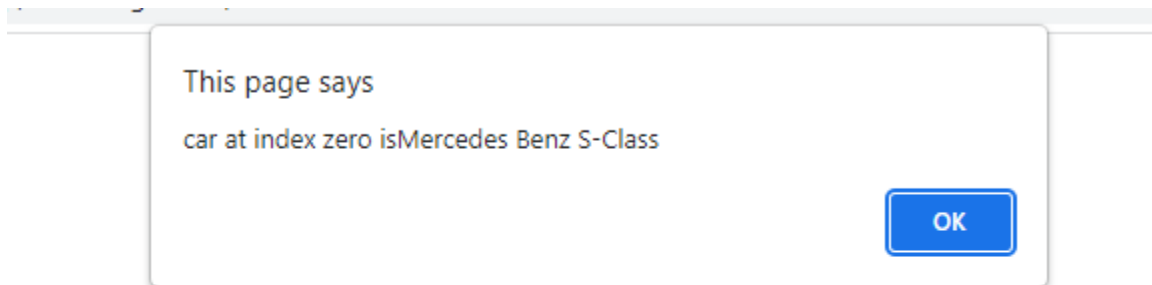
Code

```
let myCarsToDelete = ["BMW M4", "Nissan Versa", "Honda Civic", "Hyundai Elantra Hybrid",  
"Nissan Magnite", "Hyundai Alcazar"];
```

```
myCarsToDelete.pop();  
window.alert(myCarsToDelete);
```

1.3

The zeroth index element `index[0]` is the first element in an array (Pearson, 2019).



Code

```
let carsToUnshift = ["BMW M4", "Nissan Versa", "Honda Civic", "Hyundai Elantra Hybrid",  
"Nissan Magnite", "Hyundai Alcazar"];
```

```
carsToUnshift.unshift("Mercedes Benz S-Class")  
window.alert("car at index zero is"+carsToUnshift[0]);
```

The unshift() method is used to add elements to the beginning of an array (Tutorialspoint, 2020).

2.1

Update Form

Update Date

When was this website updated? :

This page says
This page was last updated: 2022-10-30

Code:

```
<!DOCTYPE html>  
<html lang="en">  
<head>  
  <meta charset="UTF-8">  
  <meta http-equiv="X-UA-Compatible" content="IE=edge">  
  <meta name="viewport" content="width=device-width, initial-scale=1.0">  
  <title>Date</title>
```

</head>

<body>

<h2>Update Form</h2>

<div >

<form>

<fieldset>

<legend>Update Date</legend>

When was this website updated? : <input id="date" name="date" placeholder="YYYY-MM-DD" value="2022-10-29" min="2022-01-01" max="2022-12-12" type="year">

<div style="text-align: right ;">

<button type="submit" style="margin-left: auto;" onclick="submitDate()">Send Update Date</button>

</div>

</fieldset>

</form>

</div>

<script>

```
function submitDate(){  
    let dateInput = document.getElementById("date");  
    window.alert("This page was last updated: "+dateInput.value);  
}  
</script>  
  
</body>  
</html>
```

2.2

```
I LOVE JAVASCRIPT PROGRAMMING WITH ALL MY HEART  
I LOVE JAVASCRIPT PROGRAMMING WITH ALL MY HEART  
I LOVE JAVASCRIPT PROGRAMMING WITH ALL MY HEART  
I LOVE JAVASCRIPT PROGRAMMING WITH ALL MY HEART  
I LOVE JAVASCRIPT PROGRAMMING WITH ALL MY HEART
```

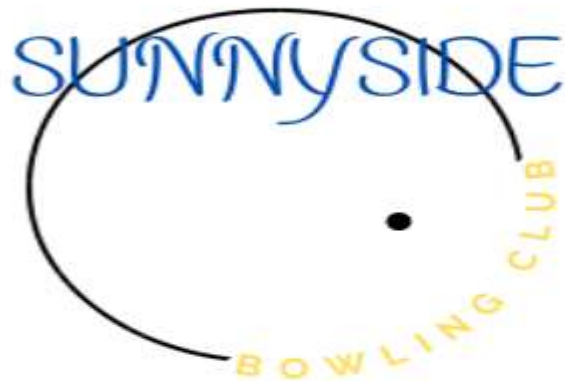
Code:

```
let iLoveJs = "i love javascript programming with all my heart";  
for (var i = 0; i < 5; i++){  
    console.log(iLoveJs.toUpperCase());  
}
```


3.1

Welcome to Pretoria Sunnyside Bowling club

Sign-Up Form



First name:

Last name:

Address:

Gender : ☐ Male ☐ Female

Email Id:

Mobile:

This page says

Name is required

OK

SUNNYSIDE

Code:

```
<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="UTF-8">

<meta http-equiv="X-UA-Compatible" content="IE=edge">

<meta name="viewport" content="width=device-width, initial-scale=1.0">

<title>Sunny Bowside Club</title>


<style>

    btns{

        display: inline; margin-right: 20px;


        margin: 0 auto;

    }

</style>

</head>

<body>


    <h2>Welcome to Pretoria Sunnyside Bowling club</h2>

    <form name="myForm" action="" onsubmit="confirmSubmittedData()">


        <h2 style="text-align: center ;">Sign-Up Form</h2>

        
```

```

<label>First name: <input name="firstName" /></label>
<label>Last name: <input name="lastName" /></label>
<label>Address: <input name="address" /></label>
Gender :
<label><input type="radio" name="gender" value="male" /> Male</label>
<label><input type="radio" name="gender" value="female" /> Female</label>
<label>Email Id:<input type="email" name="email" /></label>
<label>Mobile: <input name="mobile" /></label>
<label> Location
<br />
<select name="location">
  <option>select option</option>
  <option>Boardwalk</option>
  <option>Brooklyn</option>
  <option>Lynnwood</option>
  <option>Sunnyside</option>
</select>
</label>
</p>
  <button style="display: inline; margin-right: 20px; margin: 0 auto;"
class="btns" type="submit">Sign-Up</button>

  <button style="display: inline; margin-right: 20px; margin: 0 auto;" class="btns"
>Reset</button>
</form>

<script>
  function confirmSubmittedData (){

```

```
var fName = document.forms['myForm']['firstName'].value;
var lName = document.forms['myForm']['lastName'].value;
var address = document.forms['myForm']['address'].value;
var gender = document.forms['myForm']['gender'].value;
var email = document.forms['myForm']['email'].value;
var mobile = document.forms['myForm']['mobile'].value;
var location = document.forms['myForm']['location'].value;
if(fName == ""){
    window.alert("first Name is required");
}
if(lName == ""){
    window.alert("last Name is required");
}
if(address == ""){
    window.alert("address is required");
}
if(gender == ""){
    window.alert("gender is required");
}
if(email == ""){
    window.alert("email is required");
}
if(mobile == ""){
    window.alert("mobile is required");
}
if(location == ""){
```

```
        window.alert("location is required");
    }
}
</script>
</body>
</html>
```

3.2

Try catch expressions are used for error handling within functions.

```
try {
```

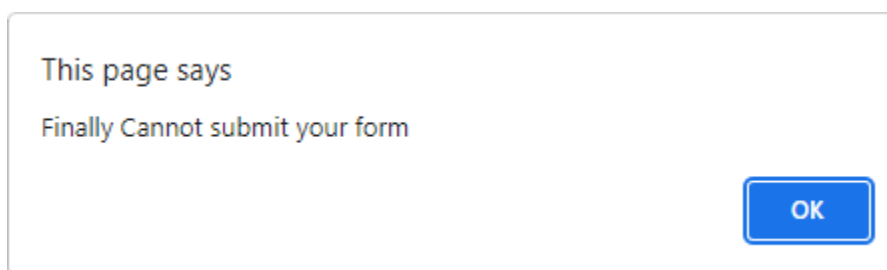
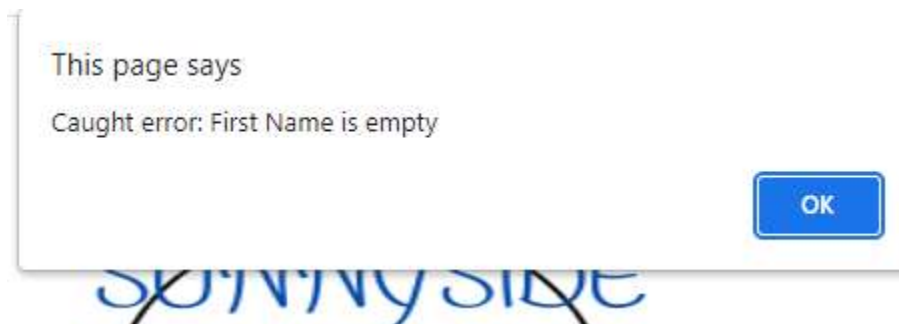
```
}
```

is used to execute a function while

```
catch(error){
```

```
}
```

allows handling of errors that might occur during execution (Tutorialspoint, 2019).



```
finally{  
}
```

Executes code irregardless of an error

Code:

```
<!DOCTYPE html>  
<html lang="en">  
<head>  
<meta charset="UTF-8">  
<meta http-equiv="X-UA-Compatible" content="IE=edge">  
<meta name="viewport" content="width=device-width, initial-scale=1.0">  
<title>Sunny Bowside Club</title>  
  
<style>  
  btns{  
    display: inline; margin-right: 20px;  
  
    margin: 0 auto;  
  
  }  
  
</style>  
</head>  
<body>  
  
  <h2>Welcome to Pretoria Sunnyside Bowling club</h2>  
<form name="myForm" action="" onsubmit=" confirmSubmittedData ()">
```

```

<h2 style="text-align: center ;">Sign-Up Form</h2>



<label>First name: <input name="firstName" /></label>
<label>Last name: <input name="lastName" /></label>
  <label>Address: <input name="address" /></label>
Gender :
<label><input type="radio" name="gender" value="male" /> Male</label>
<label><input type="radio" name="gender" value="female" /> Female</label>
  <label>Email Id:<input type="email" name="email" /></label>
<label>Mobile: <input name="mobile" /></label>
<label> Location
<br />
<select name="location">
  <option>select option</option>
  <option>Boardwalk</option>
  <option>Brooklyn</option>
  <option>Lynnwood</option>
  <option>Sunnyside</option>

</select>
</label>

  <button style="display: inline; margin-right: 20px; margin: 0 auto;"
class="btns" type="submit">Sign-Up</button>

```

```
<button style="display: inline; margin-right: 20px; margin: 0 auto;" class="btns"
>Reset</button>
```

```
</form>
```

```
<script>
```

```
function confirmSubmittedData (){
```

```
    var fName = document.forms['myForm']['firstName'].value;
```

```
    var lName = document.forms['myForm']['lastName'].value;
```

```
    var address = document.forms['myForm']['address'].value;
```

```
    var gender = document.forms['myForm']['gender'].value;
```

```
    var email = document.forms['myForm']['email'].value;
```

```
    var mobile = document.forms['myForm']['mobile'].value;
```

```
    var location = document.forms['myForm']['location'].value;
```

```
    try{
```

```
        if(fName == "") throw "First Name is empty";
```

```
        if(lName == "") throw "Last Name is empty";
```

```
        if(address == "") throw "Address is empty";
```

```
        if(gender == "") throw "Gender is empty";
```

```
        if(email == "") throw "Email is empty";
```

```
        if(mobile == "") throw "Mobile number is empty";
```

```
        if(location == "") throw "Location is empty";
```

```
    }
```

```
    catch(error){
```



```

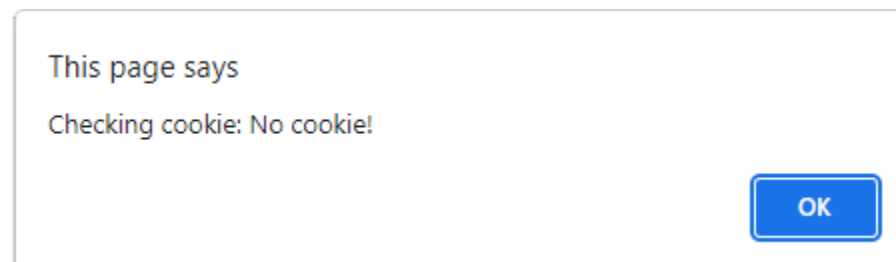
        window.alert("Caught error: "+ error);
    }
    finally{
        window.alert("Finally Cannot submit your form");
    }

}
</script>
</body>
</html>

```

3.3

Cookies in JavaScript are set using document.cookie (Programming, 2020).



Code:

```

function cookieFunction(){
    var fName = document.forms['myForm']['firstName'].value;
    var email = document.forms['myForm']['email'].value;
    var mobile = document.forms['myForm']['mobile'].value;

    document.cookie=RegCookie;
}

```

```
RegCookie = "username"+"="+fName+ ";" +  
"email"+"="+email+ ";" + "mobile"+"="+mobile+ ";" + "expires"+"="+ "30*24*60*60*1000";
```

```
if (RegCookie == ""){  
    window.alert('Cookie not set');  
}else{  
    window.alert('Cookie set');  
}  
  
}
```

```
function checkCookie(){  
    if(document.cookie.length != 0){  
        alert("There are cookies");  
    }else{  
        window.alert("Checking cookie: No cookie!");  
    }  
}
```

References

Matthus, F. (2019). *Insert Array Into Table Javascript*.

Pearson, E. (2019). *JavaScript : Arrays Arrays ¶ Data structures consisting of related data items JavaScript arrays ¶ “ dynamic ” entities that can change size after they*. 1–39.

Programming, I. (2020). *JavaScript : Using Cookies Why cookies ?* 1–8.

Tutorialspoint. (2019). *JavaScript - Errors & Exceptions Handling - Tutorialspoint*.
https://www.tutorialspoint.com/javascript/javascript_error_handling.htm

Tutorialspoint. (2020). *JAVASCRIPT - ARRAY UNSHIFT METHOD*. 5.