

**appschool**

think. learn. know.



# **E-JAVASCRIPT HOME PRACTICE BOOKLET SOLUTIONS**

appschool

think. learn. know

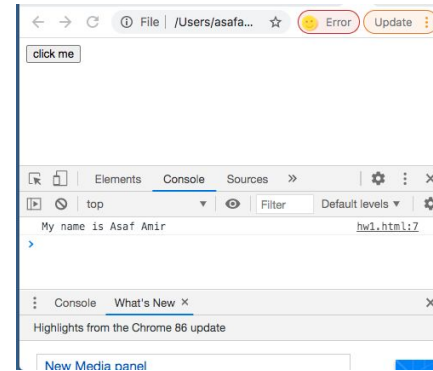


# Output

# Do it Yourself 1

Create an html page named index.html

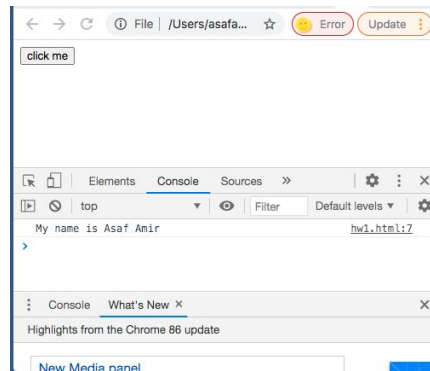
- Add a button with the caption click me
- Add a function named showMyName The function will print to the console your full name (print to console by console.log
- Clicking the button will activate the showMyName function



# Do it Yourself 1 solution

```
<!DOCTYPE html>

<html>
  <body>
    <button onclick="showMyName()">click me</button>
    <script>
      function showMyName() {
        console.log("My name is Asaf Amir")
      }
    </script>
  </body>
</html>
```

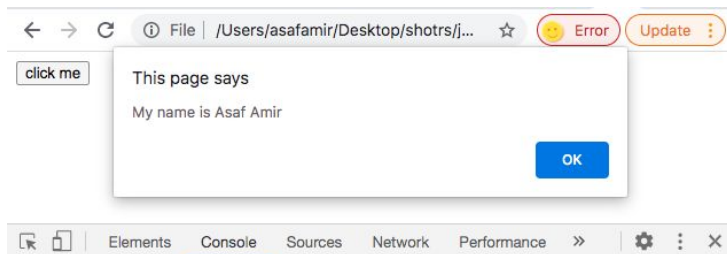


# Do it Yourself 2

For practice complete the following exercises.

Create an html page named index.html

- Add a button with the caption click me
- Add a function named showMyName The function will display your name on alert
- Clicking the button will activate the showMyName function



# Do it Yourself 2 solution

```
<!DOCTYPE html>

<html>

  <body>

    <button onclick="showMyName()">click me</button>

    <script>

      function showMyName() {

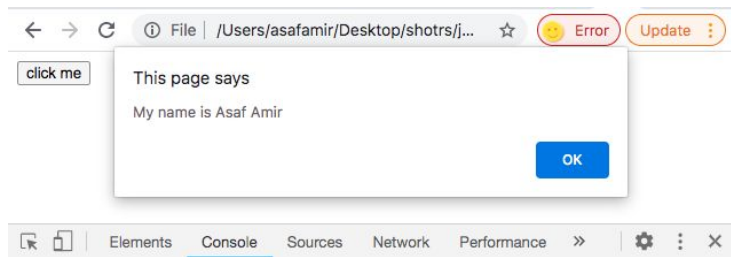
        alert("My name is Asaf Amir")

      }

    </script>

  </body>

</html>
```

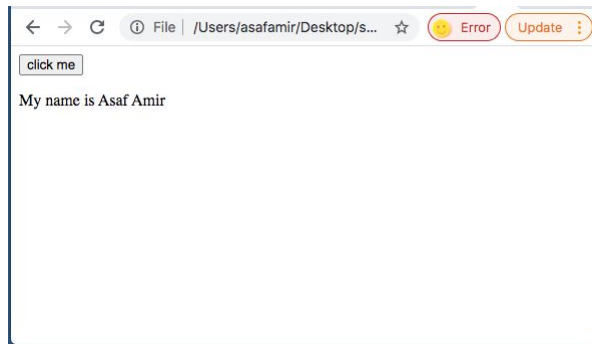


# Do it Yourself 3

For practice complete the following exercises.

5.) Create an html page named index.html

- Add a button with the caption click me
- Add a paragraph with id = "details"
- Add a function named showMyName The function will display your name on a break
- Clicking the button will activate the showMyName function



# Do it Yourself 3 solution

```
<!DOCTYPE html>

<html>

  <body>

    <button onclick="showMyName()">click me</button>

    <p id="details"></p>

    <script>

      function showMyName() {

        document.getElementById("details").innerText="My name is Asaf Amir"

      }

    </script>

  </body>

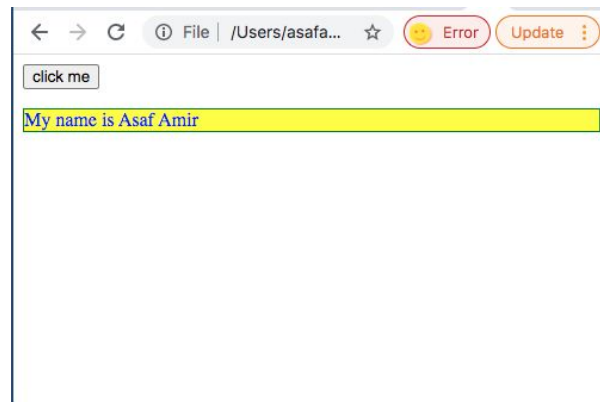
</html>
```



# Do it Yourself 4

5.) Create an html page named index.html

- Add a button with the caption click me
- Add a paragraph with id = "details"
- Add a function named showMyName The function will display your name on a blue break with a yellow background, with a 1px solid green frame
- Clicking the button will activate the showMyName function



# Do it Yourself 4 solution

```
<!DOCTYPE html>

<html>
  <body>
    <button onclick="showMyName ()">click me</button>
    <p id="details"></p>
    <script>
      function showMyName () {
        document.getElementById ("details").innerText="My name is Asaf Amir"
        document.getElementById ("details").style.color="blue"
        document.getElementById ("details").style.backgroundColor="yellow"
        document.getElementById ("details").style.border="1px solid green"
      }
    </script>
  </body>
</html>
```



appschool

think. learn. know



# If else

# Do it Yourself – 5

1. Add a button with the caption click me
  - a. Add a paragraph with id = "details"
  - b. Write a script that receives 3 numbers from the prompt.
  - c. If all numbers are equal show the sum of numbers on paragraph with id = "details" with blue color **else** show the sum of numbers on paragraph with id = "details" with red color
  - d. Call the functions by pressing a button.



# Do it Yourself – 5

```
<!DOCTYPE html>
<html>
  <body>
    <h1>Exercise1</h1>
    <button onclick="clicki()"><b>click me</b></button>
      <div id="details">1</div>
      <script>
        function clicki(){
          var num1 = parseInt( prompt("print number 1") )
          var num2 = parseInt( prompt("print number 1") )
          var num3 = parseInt( prompt("print number 1") )
          if(num1 == num2 && num2==num3) {
            document.getElementById("details").style.color = "blue";
          }
          else{
            document.getElementById("details").style.color = "red";
          }
          document.getElementById("details").innerHTML=num1;
        }
      </script>
    </body>
  </html>
```

# Do it Yourself – 6

1. Create an html page named index.html
  - a. Add a button with the caption click me
  - b. Add a paragraph with id = "details"
  - c. Write a script that receives from the user a number from the prompt.
  - d. **If** the number is even, display the word "even" with a blue color and a yellow background, 1px solid green border on paragraph with id = "details".
  - e. **Else** display the word "odd" with a yellow color and a black background, with a 1px solid red border on paragraph with id = "details"
  - f. Clicking the button will activate the function



# Do it Yourself – 6

```
<!DOCTYPE html>
<html>
  <body>
    <h1>Exercise2</h1>
    <button onclick="clicki()">click me</button>
    <div id="details">1</div>
    <script>
      function clicki() {
        var num1 = parseInt(prompt("print a number"))
        if(num1%2==0) {
          document.getElementById("details").style.color = "blue";
          document.getElementById("details").style.backgroundColor="yellow"
          document.getElementById("details").style.border="1px solid green"
          document.getElementById("details").innerHTML="Even"
        }
        else {
          document.getElementById("details").style.color = "yellow";
          document.getElementById("details").style.backgroundColor="black"
          document.getElementById("details").style.border="1px solid red"
          document.getElementById("details").innerHTML="Odd"
        }
      }
    </script>
  </body>
</html>
```

appschool

think. learn. know



# For Loops



# Do it Yourself - 1

Write a program that picks a number and shows all the small positive numbers smaller than it.

# Do it Yourself solution 1

```
<html>
  <body>
    <button onclick="solution1()">solution1</button>
    <script>
      function solution1() {
        var num = parseInt(prompt("print num"))
        for (var i=num; i>=0; i--) {
          console.log(i);
        }
      }
    </script>
  </body>
</html>
```

# Do it Yourself - 2

Write a program that picks a positive number and shows all small positive **even** numbers smaller than it.

# Do it Yourself solution 2

```
<html>
  <body>
    <button onclick="solution2()">solution1</button>
    <script>
      function solution2() {
        var num = parseInt(prompt("print num"))
        for (var i=num; i>=0; i--) {
          if (i%2==0)
            console.log(i);
        }
      }
    </script>
  </body>
</html>
```

# Do it Yourself - 3

Write a program that picks a positive number and shows all small positive **odd** numbers smaller than it.

# Do it Yourself solution 3

```
<html>
  <body>
    <button onclick="solution3()">solution3</button>
    <script>
      function solution3() {
        var num = parseInt(prompt("print num"))
        for (var i=num; i>=0; i--) {
          if (i%2==1)
            console.log(i);
        }
      }
    </script>
  </body>
</html>
```

# Do it Yourself 4

Write a program that picks 10 numbers and alerts the **maximum** number.

# Do it Yourself solution 4

```
<html>
  <body>
    <button onclick="solution4()">solution4</button>
    <script>
      function solution4() {
        var max = 0
        for (var i=0; i<10; i++) {
          var num = parseInt(prompt("print num"))
          if (num > max)
            max = num
        }
        console.log(max)
      }
    </script>
  </body>
</html>
```



# Do it Yourself 5

Write a program that picks 10 numbers and alerts the **minimum** number.

# Do it Yourself solution 5

```
<html>
  <body>
    <button onclick="solution5()">solution5</button>
    <script>
      function solution5() {
        var num = parseInt(prompt("print num"))
        min = num;
        for (var i=0; i<10; i++) {
          var num = parseInt(prompt("print num"))
          if (num <= min)
            min = num
        }
        console.log(min)
      }
    </script>
  </body>
</html>
```

# Do it Yourself 6

Write a program that picks 10 numbers and alerts the **index** of minimum number.

# Do it Yourself solution 6

```
<html>
  <body>
    <button onclick="solution6()">solution6</button>
    <script>
      function solution6(){
        var num = parseInt(prompt("print num"))
        min = num;
        index = 0;
        for(var i=0;i<10;i++){
          var num = parseInt(prompt("print num"))
          if(num <= min){
            min = num
            index =i;
          }
        }
        console.log(index);
      }
    </script>
  </body>
</html>
```

appschool

think. learn. know



# While

# Do it Yourself only with **while**

1.) Write a program that picks a number and shows all the small positive numbers smaller than it.

# Do it Yourself only with **while** solution

```
<html>
  <body>
    <button onclick="solution1()">solution1</button>
    <script>
      function solution1() {
        var num = parseInt(prompt("print num"))
        while(num >= 0) {
          console.log(num);
          num -= 1
        }
      }
    </script>
  </body>
</html>
```

# Do it Yourself only with **while**

2.) Write a program that picks a positive number and shows all small positive **even** numbers smaller than it.



# Do it Yourself only with **while** solution

```
<html>
  <body>
    <button onclick="solution2()">solution1</button>
    <script>
      function solution2(){
        var num = parseInt(prompt("print num"))
        while(num>=0){
          if(num%2==0)
            console.log(num);
          num-=1
        }
      }
    </script>
  </body>
</html>
```

# Do it Yourself only with **while**

3.) Write a program that picks a positive number and shows all small positive **odd** numbers smaller than it.

# Do it Yourself only with **while** solution

```
<html>
  <body>
    <button onclick="solution3()">solution3</button>
    <script>
      function solution3(){
        var num = parseInt(prompt("print num"))
        while(num>=0){
          if(num%2==1)
            console.log(num);
          num-=1
        }
      }
    </script>
  </body>
</html>
```

# Do it Yourself only with **while**

4.) Write a program that picks 10 numbers and alerts the **maximum** number.

# Do it Yourself only with **while** solution

```
<html>
  <body>
    <button onclick="solution4()">solution4</button>
    <script>
      function solution4() {
        var max = 0;
        var i=10;
        while(i>0) {
          var num = parseInt(prompt("print num"))
          if(num > max)
            max = num
          i-=1
        }
        console.log(max)
      }
    </script>
  </body>
</html>
```

# Do it Yourself only with **while**

5.) Write a program that picks 10 numbers and alerts the **minimum** number.

# Do it Yourself only with **while** solution

```
<html>
  <body>
    <button onclick="solution5()">solution5</button>
    <script>
      function solution5(){
        var num = parseInt(prompt("print num"))
        min = num;
        var i=9;
        while(i>0){
          num = parseInt(prompt("print num"))
          if(num <= min)
            min = num
          i-=1
        }
        console.log(min)
      }
    </script>
  </body>
</html>
```

# Do it Yourself only with **while**

6.) Write a program that picks 10 numbers and alerts the **index** of minimum number.



# Do it Yourself only with **while** solution

```
<html>
  <body>
    <button onclick="solution6()">solution6</button>
    <script>
      function solution6() {
        var num = parseInt(prompt("print num"))
        min = num;
        index = 0;
        var i=1;
        while(i<9) {
          num = parseInt(prompt("print num"))
          if(num <= min) {
            min = num
            index =i;
          }
          i+=1
        }
        console.log(index)
      }
    </script>
  </body>
</html>
```

appschool

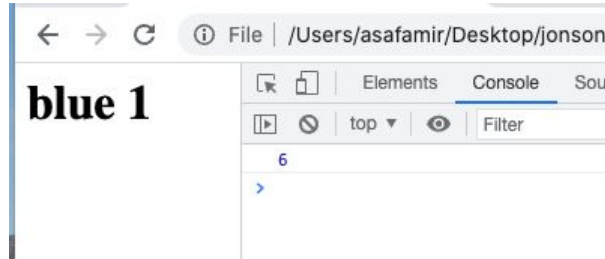
think. learn. know



# Array

# Do it Yourself 1

Write a script that initializes an array with the numbers 4,6,10,12. Print the second number to the console. (the second number is arr[1])



# Do it Yourself solution 1

```
<html>
```

```
<body>
```

```
  <h1>blue 1</h1>
```

```
  <script>
```

```
    var arr = [4,6,10,12]
```

```
    console.log(arr[1]) // print 6
```

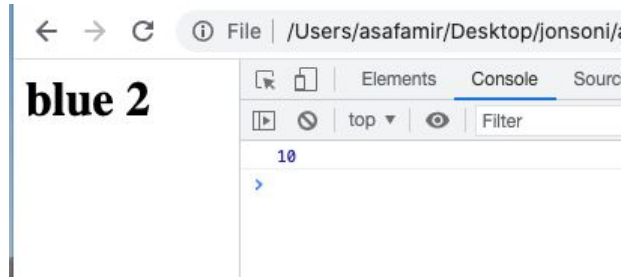
```
  </script>
```

```
</body>
```

```
</html>
```

# Do it Yourself 2

Write a script that initializes an array with the numbers 4,20,30,10. Print the last number to the console.



# Do it Yourself solution 2

```
<html>
```

```
<body>
```

```
  <h1>blue 2</h1>
```

```
  <script>
```

```
    var arr = [4,20,30,10]
```

```
    console.log(arr[arr.length-1])// print 10
```

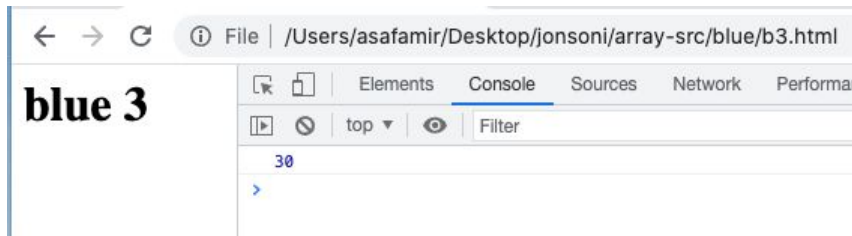
```
  </script>
```

```
</body>
```

```
</html>
```

# Do it Yourself 3

Write a script that initializes an array with the numbers 4,20,30,10,20. Print the middle number to the console.



# Do it Yourself solution 3

```
<html>
```

```
<body>
```

```
  <h1>blue 3</h1>
```

```
  <script>
```

```
    var arr = [4,20,30,10]
```

```
    console.log(arr[(arr.length)/2]) // print 30
```

```
  </script>
```

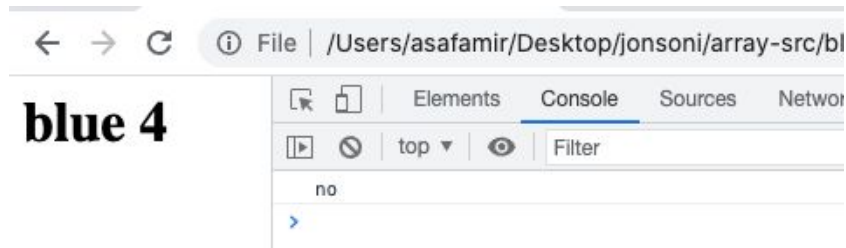
```
</body>
```

```
</html>
```



# Do it Yourself 4

Write a script that initializes an array with the numbers 23,22,30,11. Print yes to the console if the last number equals to the first number **else** print no



# Do it Yourself solution 4

```
<html>

<body>

  <h1>blue 4</h1>

  <script>

    var arr = [23,22,30,11]

    if(arr[0] == arr[arr.length-1])

      console.log("yes")

    else console.log("no")

  </script>

</body>

</html>
```

# Do it Yourself 5

Write a script that initializes an array with the numbers 23,5,30,11,12. Print the minimum number to the console

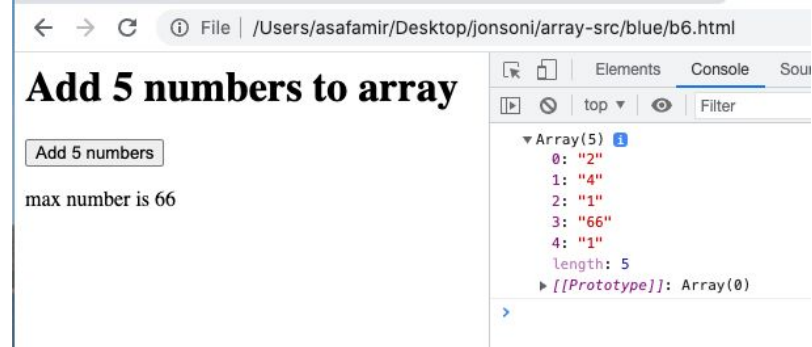


# Do it Yourself solution 5

```
<html>
<body>
  <h1>blue 5</h1>
  <script>
    var arr = [23,5,30,11,12]
    console.log(arr)
    min = arr[0]
    for(let i=0;i<arr.length;i++){
      if(arr[i]<min)
        min = arr[i]
    }
    console.log(min)
  </script>
</body>
</html>
```

# Do it Yourself 6

1. Add a button with the caption Add 5 numbers to array
  - a. Add a paragraphs with id = "max"
  - b. Declare empty array (let/var arr=[])
  - c. Write a script that receives 5 numbers from the prompt and push them to arr.
  - d. The script will display on id = "max" the max number in the array
  - e. Call the functions by pressing a button.



# Do it Yourself solution 6

```
<html>
```

```
<body>
```

```
<h1>Add 5 numbers to array</h1>
```

```
<button onclick="addNumbersToArray () ">Add 5 numbers</button>
```

```
<p id="max"></p>
```

```
<script>
```

```
function addNumbersToArray () {  
    arr = []  
    for (let i=0;i<5;i++) {  
        var n = prompt ("print num ")  
        arr.push (n);  
    }  
}
```

# Do it Yourself solution 6

```
console.log(arr)

max = arr[0];

for(let i=0;i<arr.length;i++){

    if(arr[i] > max)

        max = arr[i]

}

document.getElementById("max").innerText = "max number is " + max

}

</script>

</body>

</html>
```

appschool

think. learn. know



# Functions



# Do it Yourself 1

- A. Write an expression function that accepts 3 numbers as a parameter and displays the minimum number in alert .
- B. Call the function from A with  $n1=3$ ,  $n2=5$ ,  $n3=9$  and alert the minimum number

# Do it Yourself 1 solution

```
<html>
  <body>
    <button onclick="partB()">bigger</button>
    <p id="bigger"></p>
  </body>
  <script>
    bigger = function (n1, n2, n3) {
      if (n1>=n2&& n1>=n3)
        return n1;
      else if (n2>=n1&& n2>=n3)
        return n2;
      return n3;
    }
    function partB(){
      let answer = bigger(3,5,9)
      document.getElementById("bigger").innerText ="The bigger is  :" + answer
    }
  </script>
</html>
```

# Do it Yourself 2

- A. Write an **arrows** function that accepts 3 numbers as a parameter and displays the minimum number in alert .
- B. Call the function from A with  $n1=3$ ,  $n2=5$ ,  $n3=9$  and alert the minimum number

# Do it Yourself 2 solution

```
<html>
  <body>
    <button onclick="partB()">bigger</button>
    <p id="bigger"></p>
  </body>
  <script>
    let bigger = (n1, n2, n3) => {
      if(n1>=n2&& n1>=n3)
        return n1;
      else if(n2>=n1&& n2>=n3)
        return n2;
      return n3;
    }
    function partB(){
      let answer = bigger(3,5,9)
      document.getElementById("bigger").innerText = "The bigger is : " + answer
    }
  </script>
</html>
```

# Do it Yourself 3

- A. Write a **Declaration** function that accepts an array, and return the sum of the numbers of the array.
- B. Call the function from A using `arr=[20,60,12,9]` and alert the sum of array

# Do it Yourself 3 solution

```
<html>
  <body>
    <button onclick="partB()">sum</button>
    <p id="sum"></p>
  </body>
</html>
<script>
  function sumOfArray(arr) {
    let sum = 0;
    for(let i=0;i<arr.length;i++){
      sum+=arr[i]
    }
    return sum;
  }
  function partB(){
    let arr = [1,6,2,9]
    let s = sumOfArray(arr)
    document.getElementById("sum").innerText ="sum is : " + s
  }
</script>
</html>
```

# Do it Yourself 4

- A. Write an **expression** function that accepts an array, and return the sum of the numbers of the array.
- B. Call the function from A using `arr=[20,60,12,9]` and alert the sum of array

# Do it Yourself 4 solution

```
<html>
  <body>
    <button onclick="partB()">sum</button>
    <p id="sum"></p>
  </body>
  <script>
    let sumOfArray = function(arr) {
      let sum = 0;
      for(let i=0;i<arr.length;i++){
        sum+=arr[i]
      }
      return sum;
    }
    function partB(){
      let arr = [1,6,2,9]
      let s = sumOfArray(arr)
      document.getElementById("sum").innerText ="sum is :" + s
    }

  </script>
</html>
```



# Do it Yourself 5

- A. Write an **Arrow** function that accepts an array, and return the sum of the numbers of the array.
- B. Call the function from A using `arr=[20,60,12,9]` and alert the sum of array

# Do it Yourself 5 solution

```
<html>
  <body>
    <button onclick="partB()">sum</button>
    <p id="sum"></p>
  </body>
  <script>
    let sumOfArray = (arr)=> {
      let sum = 0;
      for(let i=0;i<arr.length;i++){
        sum+=arr[i]
      }
      return sum;
    }
    function partB(){
      let arr = [1,6,2,9]
      let s = sumOfArray(arr)
      document.getElementById("sum").innerText ="sum is :" + s
    }
  </script>
</html>
```

appschool





think. learn. know



DOM

# Do it Yourself 1

Given the following HTML page. Add a function called `colorGreenOrBlue` to the page. If the value of the item is less than 100 then the item will be red otherwise the item will be blue

```
<html>
<body>
  <h1>Color  or  </p>
  <button onclick="colorGreenOrBlue ()">Color  or
 </button>
  <ul id="items">
    <li>1</li>
    <li>208</li>
    <li>99</li>
  </ul>
  <script>
    function colorGreenOrBlue () {
  </script>
</body>
</html>
```



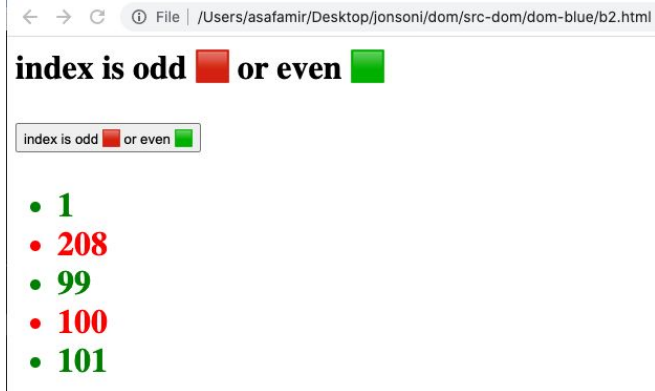
# Do it Yourself 1 solution

```
<html>
  <body>
    <h1>Color  or  </h1>
    <button onclick="colorGreenOrBlue ()">Color  or
 </button>
    <ul id="items">
      <li>1</li>
      <li>208</li>
      <li>99</li>
    </ul>
    <script>
      function colorGreenOrBlue () {
        var list = document.querySelectorAll ('#items li');
        for (var i=0; li=list[i]; i++) {
          if (parseInt (li.innerText) > 100)
            li.style.color = "blue"
          else
            li.style.color = "red"
        }
      }
    </script>
  </body>
</html>
```

# Do it Yourself 2

Given the following HTML page. Add a function called `evenOrOdd` to the page. If the index of an item is odd then item will be red otherwise the item will be green. (remember index start from 0)

```
<html>
<body>
  <h1>index is odd ■ or even ■</h1>
  <button onclick="evenOrOdd()">index is odd ■ or even ■</button>
  <ul id="items">
    <li>1</li>
    <li>208</li>
    <li>99</li>
    <li>100</li>
    <li>101</li>
  </ul>
  <script>
    function evenOrOdd(){}
  </script>
</body>
</html>
```



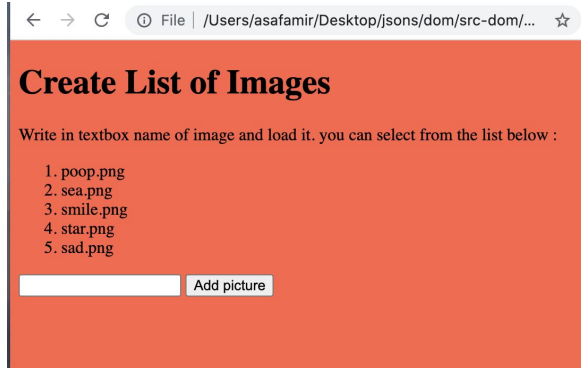
# Do it Yourself 2 solution

```
<html>
<body>
  <h1>index is odd ■ or even ■</h1>
  <button onclick="evenOrOdd()">index is odd ■ or even ■</button>
  <ul id="items">
    <li>1</li>
    <li>208</li>
    <li>99</li>
    <li>100</li>
    <li>101</li>
  </ul>
  <script>
    function evenOrOdd() {
      var list = document.querySelectorAll('#items li');
      for(var i=0; li=list[i]; i++) {
        if(i%2==1)
          li.style.color = "red"
        else
          li.style.color = "green"
      }
    }
  </script>
</body>
</html>
```

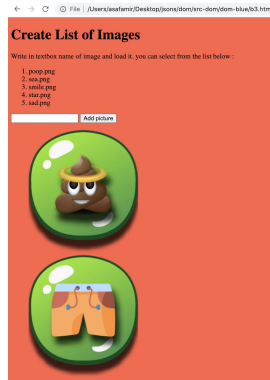


# Do it Yourself 3

Create function named add. When the user inputs the name of a picture from the list and presses add picture, the picture will added to ul element with class id=list.



After user added  
poof.png and  
sea.png



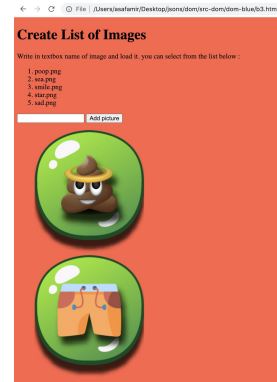
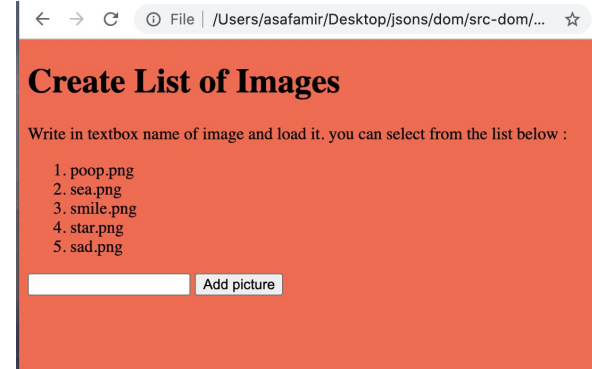
```
<!DOCTYPE html>
<html>
  <head>
    <style>
      body{
        background: tomato;
      }
      ul {
        list-style-type: none;
      }
    </style>
  </head>
  <body>
    <h1>Create List of Images</h1>
    <p>Write in textbox name of image and load it.
    you can select from the list below :</p>
    <ol class="txtList">
      <li>poop.png</li>
      <li>sea.png</li>
      <li>smile.png</li>
      <li>star.png</li>
      <li>sad.png</li>
    </ol>
    <input type="text" id="name">
    <button class="add" onclick="add()" >
    Add picture</button>
    <ul id="list">
    </ul>
    <script>
    </script>
  </body>
</html>
```



# Do it Yourself 3 hint

## 1. Hint

- a. Create element `img` and give it `img.src` dynamically
- b. Create `li` element dynamically
- c. Get reference to `li` with `id = list`
- d. Append the picture to `li`
- e. Append `li` to the `ul`



## Do it Yourself 3 solution

```
<!DOCTYPE html>
<html>
  <head>
    <style>
      body{
        background: tomato;
      }
      ul {
        list-style-type : none;
      }
    </style>
  </head>
  <body>
    <h1>Create List of Images </h1>
    <p>Write in textbox name of image and load it.
    you can select from the list below : </p>
    <ol class="txtList">
      <li>poop.png</li>
      <li>sea.png</li>
      <li>smile.png</li>
      <li>star.png</li>
      <li>sad.png</li>
    </ol>
```

## Do it Yourself 3 solution

```
<input type="text" id="name">
<button class="add" onclick="add()">Add picture</button>
<ul id="list">
</ul>
<script>
  function add(){
    let val = document.getElementById("name").value;
    let img = document.createElement("img")
    img.src = val;
    let ul = document.getElementById("list")
    let li = document.createElement("li")
    li.appendChild(img)
    ul.appendChild(li)
    document.getElementById("name").value = ""
  }
</script>

</body>
</html>
```

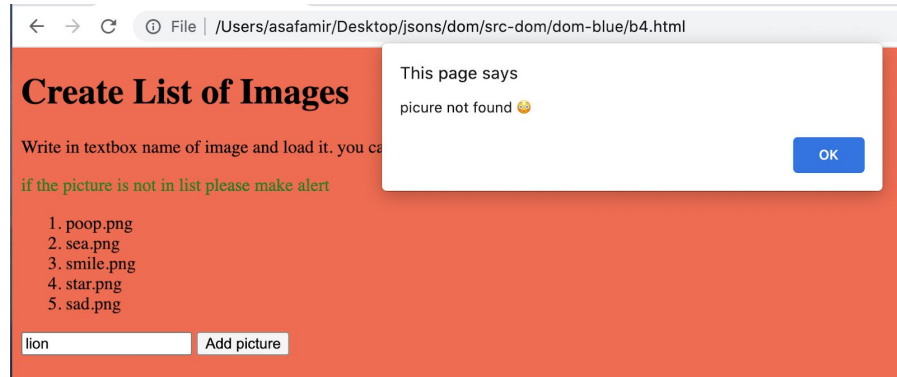
## Do it Yourself 4

To the previous Add function write code to check if the picture is in the list, If the picture is not in the list make an alert.

Hint :

Add array to the function with pictures names and check if the value that the user prints is in the array.

```
let arr =  
['poop.png', 'sea.png', 'smile.png', 'star.png', 'sad.png']
```



## Do it Yourself 4 Solution

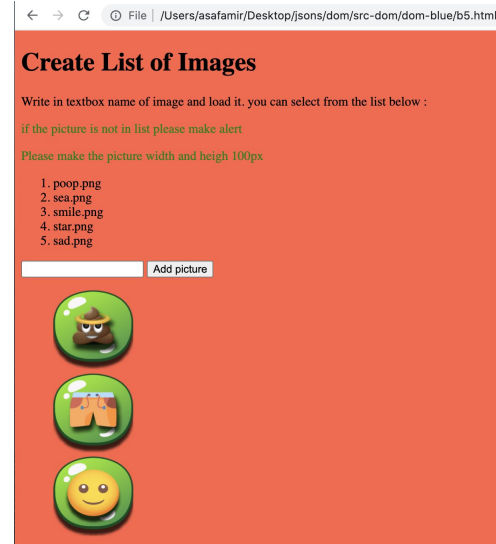
```
<!DOCTYPE html>
<html>
  <head>
    <style>
      body{
        background: tomato;
      }
      ul {
        list-style-type: none;
      }
    </style>
  </head>
  <body>
    <h1>Create List of Images</h1>
    <p>Write in textbox name of image and load it.
    you can select from the list below </p>
    <p style="color:green">if the picture is not in list please make alert</p>
    <ol class="txtList">
      <li>poop.png</li>
      <li>sea.png</li>
      <li>smile.png</li>
      <li>star.png</li>
      <li>sad.png</li>
    </ol>
```

## Do it Yourself 4 solution

```
<input type="text" id="name">
<button class="add" onclick="add()">Add picture</button>
<ul id="list">
</ul>
<script>
  function add() {
    let val = document.getElementById("name").value;
    let arr = ['poop.png', 'sea.png', 'smile.png', 'star.png', 'sad.png' ]
    if(arr.indexOf(val) >= 0) {
      let img = document.createElement("img")
      img.src = val;
      let ul = document.getElementById("list")
      let li = document.createElement("li")
      li.appendChild(img)
      ul.appendChild(li)
      document.getElementById("name").value = ""
    }
    else {
      alert("picure not found 😬")
    }
  }
</script>
</body>
</html>
```

## Do it Yourself 5

Add on to the previous function code that dynamically creates the image with a width and height of 100px



## Do it Yourself 5 solution

```
<!DOCTYPE html>
<html>
  <head>
    <style>
      body{
        background: tomato;
      }
      ul {
        list-style-type: none;
      }
    </style>
  </head>
  <body>
    <h1>Create List of Images</h1>
    <p>Write in textbox name of image and load it.
    you can select from the list below :</p>
    <p style="color:green">if the picture is not in list please make alert</p>
    <p style="color:green">Please make the picture width and height 100px</p>
```



## Do it Yourself 5 solution

```
<ol class="txtList">
  <li>poop.png</li>
  <li>sea.png</li>
  <li>smile.png</li>
  <li>star.png</li>
  <li>sad.png</li>
</ol>
<input type="text" id="name">
<button class="add" onclick="add()">Add picture</button>
<ul id="list">
</ul>
```

## Do it Yourself 5 solution

```
<script>
  function add() {
    let arr = ['poop.png', 'sea.png', 'smile.png', 'star.png', 'sad.png' ]
    let val = document.getElementById("name").value;
    if(arr.indexOf(val)>=0) {
      let img = document.createElement("img")
      img.src = val;
      img.style.width='100px'
      img.style.height='100px'
      let ul = document.getElementById("list")
      let li = document.createElement("li")
      li.appendChild(img)
      ul.appendChild(li)
      document.getElementById("name").value = ""
    }
    else {
      alert("Image Not Found, try again")
    }
  }
</script>
</body>
</html>
```

## Do it Yourself 6

Create a new function named `remove` that accepts a string parameter called `name` that removes from list picture names that were previously added.

```
function remove(name){}
```

Call the function from the function `add`.

### Create List of Images

Write in textbox name of image and load it. you can select from the list below :

if the picture is not in list please make alert

Please make the picture width and height 100px

add function named `remove` that remove from list picture name that added

1. star.png
2. sad.png

## Do it Yourself 6 solution

```
<!DOCTYPE html>
<html>
  <head>
    <style>
      body{
        background: tomato;
      }
      ul {
        list-style-type: none;
      }
    </style>
  </head>
  <body>
    <h1>Create List of Images </h1>
    <p>Write in textbox name of image and load it.
    you can select from the list below : </p>
    <p style="color:green">if the picture is not in list please make alert </p>
    <p style="color:green">Please make the picture width and height 100px </p>
```

## Do it Yourself 6 solution

`<p style="color:green">add function named remove that removes from the list picture names that were added</p>`

```
<ol class="txtList">
  <li>poop.png</li>
  <li>sea.png</li>
  <li>smile.png</li>
  <li>star.png</li>
  <li>sad.png</li>
</ol>
<input type="text" id="name">
<button class="add" onclick="add()">Add picture</button>
<ul id="list">
</ul>
```

## Do it Yourself 6 solution

```
<script>
function add(){
  let arr = ['poop.png', 'sea.png', 'smile.png', 'star.png', 'sad.png' ]
  let val = document.getElementById("name").value;
  if(arr.indexOf(val)>=0){
    let img = document.createElement("img")
    img.src = val;
    img.style.width='100px'
    img.style.height='100px'
    let ul = document.getElementById("list")
    let li = document.createElement("li")
    li.appendChild(img)
    ul.appendChild(li)
    document.getElementById("name").value = ""
    remove(val)
  }
  else {
    alert("Image Not Found, try again" )
  }
}
```

## Do it Yourself 6 solution

```
function remove(name){
  var list = document.querySelectorAll('.txtList li');
  console.log(list)
  for(var i=0; li=list[i]; i++) {
    console.log(li.innerText + " | " + name)
    if(li.innerText == name)
      li.parentNode.removeChild(li);
  }
}
```

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

appschool

think. learn. know



# Event Handler



# Do it yourself 1

Given the following HTML page. Add a function called `blurFunction` to the page. When the user leaves the text box, the background color will change to red.

```
<html>
  <body>
    <h3>Change Background Color of Textbox</h3>
    Enter your name:
    <input type="text" id="myInput"
      onblur="blurFunction(this)">

    <script>

      function blurFunction(obj) {
      }

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

← → ↻ ⓘ File | /Users/asafamir/Desktop/jsons

## Change Background Color of Textbox

Enter your name:

← → ↻ ⓘ File | /Users/asafamir/Desktop/jsons/dom/src-dor

## Change Background Color of Textbox

Enter your name:

# Do it yourself 1 solution

```
<html>
  <body>
    <h3>Change Background Color of Textbox</h3>
    Enter your name:
    <input type="text" id="myInput"
      onblur="blurFunction(this)">
    <script>
      function blurFunction(obj) {
        obj.style.background="red";
      }
    </script>
  </body>
</html>
```

← → ↻ ⓘ File | /Users/asafamir/Desktop/jsons

## Change Background Color of Textbox

Enter your name:

← → ↻ ⓘ File | /Users/asafamir/Desktop/jsons/dom/src-dor

## Change Background Color of Textbox

Enter your name:

# Do it yourself 2

Given the following HTML page. Add a function called `focusFunction` to the page. When the user is focused on the text box, the background color will change to yellow

```
<html>
  <body>
    <h3>Change Background Color of Textbox</h3>
    Enter your name:
    <input type="text" id="myInput"
onblur="blurFunction(this) "
onfocus="focusFunction(this) ">
    <script>
      function blurFunction(obj) {
      }
      function focusFunction(obj) {
      }
    </script>
  </body>
</html>
```

← → ↻ ⓘ File | /Users/asafamir/Desktop/jsons/

## Change Background Color of Textbox

Enter your name:

← → ↻ ⓘ File | /Users/asafamir/Desktop/jsons/

## Change Background Color of Textbox

Enter your name:

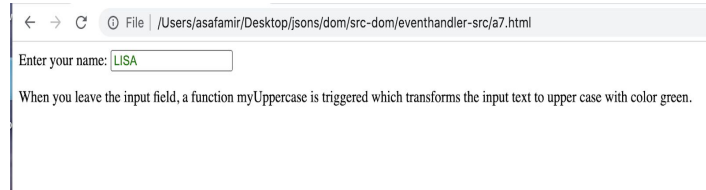
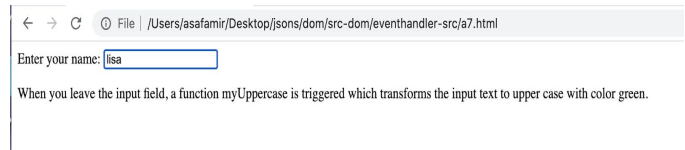
# Do it yourself 2 solution

```
<html>
  <body>
    <h3>Change Background Color of Textbox</h3>
    Enter your name:
    <input type="text" id="myInput"
onblur="blurFunction(this)" onfocus="focusFunction(this)">
    <script>
      function blurFunction(obj) {
      }
      function focusFunction(obj) {
        obj.style.background = "yellow"
      }
    </script>
  </body>
</html>
```

# Do it yourself 3

Given the following HTML page. Add a function called `myUppercase` to the page. When the user leaves the input field, the function `myUppercase` is triggered which transforms the input text to uppercase with color green.

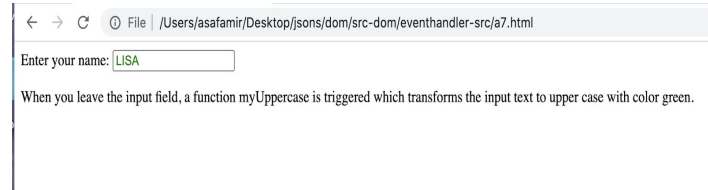
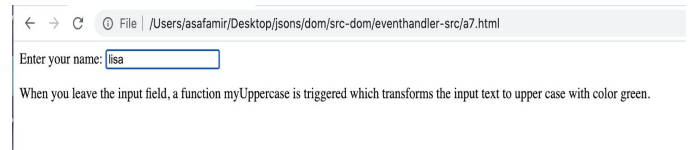
```
<!DOCTYPE html>
<html>
  <body>
    Enter your name:
    <input type="text" id="firstname" onblur="myUppercase(this)">
    <p>When you leave the input field,
    a function myUppercase is triggered which transforms the
    input text to upper case with color green</p>
    <script>
      function myUppercase(obj) {
      }
    </script>
  </body>
</html>
```



# Do it yourself 3 solution

Given the following HTML page. Add a function called `myUppercase` to the page. When the user leaves the input field, the function `myUppercase` is triggered which transforms the input text to uppercase with color green.

```
<!DOCTYPE html>
<html>
  <body>
    Enter your name:
    <input type="text" id="firstname" onblur="myUppercase(this)" >
    <p>When you leave the input field,
    a function myUppercase is triggered which transforms the
    input text to upper case with color green</p>
    <script>
      function myUppercase(obj) {
        let txt = obj.value;
        obj.style.color="green"
        obj.value=txt.toUpperCase()
      }
    </script>
  </body>
</html>
```



appschool

think. learn. know



# Data Structure and Algorithms

# Data structure for exercise

```
let restaurant= {
  timeShift1: '8:00-16:00',
  timeShift2: '16:00-00:00',
  shiftsWorkers: [
    [
      'Mike',
      'James',
      'John',
      'Michael',
      'William',
      'David',
    ],
    [
      'Joseph',
      'Thomas',
      'Charles',
      'Christopher',
      'Daniel',
      'Mark',
    ],
  ],
  totalTipShift1:4000,
  totalTipShift2:5000,
  tipped: ['Mike', 'James', 'Michael', 'David', 'Thomas'],
  date: '12/12/2021'
};
```



# Do it yourself 1

Create one worker array for each shift

# Do it yourself 2

The first worker in any shiftsWorkers array is the shiftManager and

the others are simple workers.

For Mike (shift 1) create one variable ('SM')

with the worker's name, and one array

('simpleWorkers') with all the remaining field

workers

# Do it yourself 3

Create an array 'allWorkers' containing all workers  
of both shifts (12 workers)

# Do it yourself 4

4. Write a function ('tipped') that receives an arbitrary number of worker names (NOT an array) and prints each of them to the console,

**Hint** - use spread operator

# Solution 1-4

```
<html>
  <body>
    <script>
      let restaurant= {
        timeShift1: '8:00-16:00',
        timeShift2: '16:00-00:00',
        shiftsWorkers: [
          [
            'Mike',
            'James',
            'John',
            'Michael',
            'William',
            'David',
          ],
          [
            'Joseph',
            'Thomas',
            'Charles',
            'Christopher',
            'Daniel',
            'Mark',
          ],
        ],
      };
    </script>
  </body>
</html>
```

# Solution 1-4

```
totalTipShift1:4000,  
totalTipShift2:5000,  
tipped: ['Mike', 'James', 'Michael', 'David', 'Thomas'],  
date: '12/12/2021'  
};  
/*
```

1. Create one worker array for each shift
2. The first worker in any shiftsWorkers array is the shiftManager and the others are simple workers.  
For Mike (shift 1) create one variable ('SM')  
with the worker's name, and one array  
( 'simpleWorkers') with all the remaining field workers
3. Create an array 'allWorkers' containing all workers of both shifts (12 workers)
4. Write a function ('tipped') that receives an arbitrary number of worker names (NOT an array) and prints each of them to the console,  
\*/

# Solution 1-3

```
// 1.
const [workers1, workers2] = restaurant.shiftsWorkers;
console.log(workers1, workers2);
// 2.
let [sm, ...fieldWorkers] = workers1;
console.log(sm, fieldWorkers);
// 3.
const allWorkers = [...workers1, ...workers2];
console.log(allWorkers);
// 4.
const printTip = function (...workers) {
  console.log(workers);
};
printTip('Mike', 'James', 'Michael', 'David', 'Thomas');

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

# Do it yourself 5

Loop over the `restaurant.tipped` array and print each worker name to the console)



# Do it yourself 6

Use a loop to print all address properties

# Do it yourself 5-6 solution

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

/*
1. Loop over the restaurant.tipped array and print each worker name to the console)
2. Use a loop to print all address properties
*/

const days = ['monday', 'tuesday', 'wednesday', 'thuesday', 'friday', 'saturday', 'sunday'];
let restaurant= {
  timeShift1: '8:00-16:00',
  timeShift2: '16:00-00:00',
```

# Do it yourself 5-6 solution

```
shiftsWorkers: [  
  [  
    'Mike' ,  
    'James' ,  
    'John' ,  
    'Michael' ,  
    'William' ,  
    'David' ,  
  ],  
  [  
    'Joseph' ,  
    'Thomas' ,  
    'Charles' ,  
    'Christopher' ,  
    'Daniel' ,  
    'Mark' ,  
  ],  
],
```

# Do it yourself 5-6 solution

```
totalTipShift1: 4000,
totalTipShift2: 5000,
tipped: [ 'Mike', 'James', 'Michael', 'David', 'Thomas' ],
date: '12/12/2021',
address: { city: "NY", street: "Arguban 6" },
opendays : [ days[0], days[3], days[4] ]
};

// 1.

console.log("=====1=====" )

const tipped = restaurant.tipped;
for(const [i,el] of restaurant.tipped.entries())
    console.log(`${i+1}. ${el}`)
    console.log("=====2=====" )

// 2.

const properties = Object.keys(restaurant.address)
for(const [i,el] of properties.entries())
    console.log(`${i+1}. ${el}`)

</script>

</body>

</html>
```

appschool

think. learn. know



# More about functions

# Do it yourself 1

1. Create a persons object with property username, password
2. Add function addHobby to persons that get 2 parameters,hobby and description.
  - a. The function create an objHobby {hobby:hobby, description:description} and add objHobby to hobbies array .
  - b. The function will print to the console the new hobbies added to the this.user
3. Create instance of persons object mike with username mike and password 123
4. Get reference with mikeAddHobby variable to addHobby function
5. Call mikeAddHobby function with basketball hobby and I love to play description (**do it with call**)

# Do it yourself 1 solution

```
<html>
  <body>
    <script>
      function greeting(name) {
        alert('Hello ' + name);
      }
      function processUserInput(callback) {
        var name = prompt('Please enter your name. ');
        callback(name);
      }
      processUserInput(greeting);
    </script>
  </body>
</html>
```

# Do it yourself 2

1. Create a persons object with property username, password
2. Add function addHobby to persons that get 2 parameters,hobby and description.
  - a. The function create an objHobby {hobby:hobby, description:description} and add objHobby to hobbies array .
  - b. The function will print to the console the new hobbies added to the this.user
3. Create instance of persons object mike with username mike and password 123
4. Get reference with mikeAddHobby variable to addHobby function
5. Call mikeAddHobby function with basketball hobby and I love to play description (**do it with apply**)



# Do it yourself 2 Solution

```
<html>
  <body>
    <script>
      function add(n1, n2) {
        console.log(n1+n2);
      }
      function sub(n1, n2) {
        console.log(n1-n2);
      }
      function mult(n1, n2) {
        console.log(n1*n2);
      }
      function f(n1, n2, callback, callback2) {
        callback(n1,n2);
        callback2(n1,3);
      }
      f(5,6,mult,add);
    </script>
  </body>
</html>
```

# Do it yourself 3

1. Create a persons object with property username, password
2. Add function addHobby to persons that get 2 parameters,hobby and description.
  - a. The function create an objHobby {hobby:hobby, description:description} and add objHobby to hobbies array .
  - b. The function will print to the console the new hobbies added to the this.user
3. Create instance of persons object mike with username mike and password 123
4. Get reference with mikeAddHobby variable to addHobby function
5. Call mikeAddHobby function with basketball hobby and I love to play description (**do it with bind**)

# Do it yourself 3 solution

```
<html>
  <body>
    <script>
      function magic() {
        return function calc(x) { return x * 42; };
      }
      var answer = magic();
      answer(1337); // 56154
    </script>
  </body>
</html>
```

appschool

think. learn. know



# Array and map forEach

# Do it yourself 1

Given the following data structure

```
let cars = [  
  {color:"red",price:2000,benefits:["electric windows","glossy paint" ]},  
  {color:"blue",price:4500,benefits:["electric windows","modern tires",  
  ]},  
  {color:"pink",price:5000,benefits:["electric windows","modern  
tires","glossy paint" ]},  
  {color:"white",price:7800,benefits:["electric windows", ]},  
]
```

Add using foreach loop 20% for all car prices

# Do it yourself 1 solution

```
<html>
  <body>
    <script>
      let cars = [
        {color:"red",price:2000,benefits:["electric windows","glossy paint" ]},
        {color:"blue",price:4500,benefits:["electric windows","modern tires", ]},
        {color:"pink",price:5000,benefits:["electric windows","modern
tires","glossy paint" ]},
        {color:"white",price:7800,benefits:["electric windows", ]},
      ]
      //add 20 percent to all cars
      cars.forEach(car=>car.price= car.price * 1.2)
      console.log(cars)
    </script>
  </body>
</html>
```

# Do it yourself 2

Given the following data structure

```
let cars = [  
  {color:"red",price:2000,benefits:["electric windows","glossy paint" ]},  
  {color:"blue",price:4500,benefits:["electric windows","modern tires",  
  ]},  
  {color:"pink",price:5000,benefits:["electric windows","modern  
tires","glossy paint" ]},  
  {color:"white",price:7800,benefits:["electric windows", ]},  
]
```

Add using foreach loop and remove all cars with price above 5000.

# Do it yourself 2 solution

```
<html>
  <body>
    <script>
      let cars = [
        { color: "red", price: 2000, benefits: ["electric windows", "glossy paint" ]},
        { color: "blue", price: 4500, benefits: ["electric windows", "modern tires", ]},
        { color: "pink", price: 5000, benefits: ["electric windows", "modern tires", "glossy
paint" ]},
        { color: "white", price: 7800, benefits: ["electric windows", ]},
      ]
      //remove cars with price above 5000
      cars.forEach(function(car) {
        if (car.price > 5000) {
          cars.shift()
        }
      })
      console.log(cars)
    </script>
  </body>
</html>
```



appschool

think. learn. know



# Map, filter, reduce and find

# Do it yourself 1

Given the following data structure

```
let cars = [
  {color:"red",price:2000,benefits:["electric windows","glossy paint" ]},
  {color:"blue",price:4500,benefits:["electric windows","modern tires",
  ]},
  {color:"pink",price:5000,benefits:["electric windows","modern
  tires","glossy paint" ]},
  {color:"white",price:7800,benefits:["electric windows", ]},
]
```

Used `cars.some` and print if there is a car price that below 2000

# Do it yourself 1 solution

```
<html>
  <body>
    <script>
      let cars = [
        {color:"red",price:2000,benefits:["electric windows","glossy paint" ]},
        {color:"blue",price:4500,benefits:["electric windows","modern tires", ]},
        {color:"pink",price:5000,benefits:["electric windows","modern tires","glossy paint" ]},
        {color:"white",price:7800,benefits:["electric windows", ]},
      ]
      function isBelowThan2000(element, index, array) {
        return element.price < 2000;
      }
      console.log(cars.some(isBelowThan2000)); // false
    </script>
  </body>
</html>
```

# Do it yourself 2

Given the following data structure

```
let cars = [
  {color:"red",price:2000,benefits:["electric windows","glossy paint" ]},
  {color:"blue",price:4500,benefits:["electric windows","modern tires",
  ]},
  {color:"pink",price:5000,benefits:["electric windows","modern
  tires","glossy paint" ]},
  {color:"white",price:7800,benefits:["electric windows", ]},
]
```

Used `cars.every` and `print` if every car price below 2000

# Do it yourself 2 solution

```
<html>
  <body>
    <script>
      let cars = [
        {color:"red",price:2000,benefits:["electric windows","glossy paint" ]},
        {color:"blue",price:4500,benefits:["electric windows","modern tires",
]],
          {color:"pink",price:5000,benefits:["electric windows","modern
tires","glossy paint" ]},
          {color:"white",price:7800,benefits:["electric windows", ]},
        ]
      function isBelowThan2000(element, index, array) {
        return element.price < 2000;
      }
      console.log(cars.every(isBelowThan2000)); // false
    </script>
  </body>
</html>
```

appschool

think. learn. know



# Numbers, dates, and timers

# Doit your self 1

**Make an html page that displays the time on the screen. hour, minutes and seconds**

```
<html>

  <body>

    <h1>Clock </h1>

    <p id="clock">00:00:00</p>

    <script>

      function clocki () {

        }

      setInterval (clocki,1000)

    </script>

  </body>

</html>
```

# Doit your self 1 solution

```
<!DOCTYPE html>
<html>
  <body>
    <h1>Clock</h1>
    <div id="txtClock"></div>
    <script>
      function startTime() {
        const today = new Date();
        let hours = today.getHours();
        let minuets = today.getMinutes();
        let seconds = today.getSeconds();
        minuets = checkTime(minuets);
        seconds = checkTime(seconds);
        document.getElementById('txtClock').innerHTML = hours + ":" + minuets + ":" + seconds;
        setTimeout(startTime, 1000);
      }
      function checkTime(num) {
        if (num < 10) {num = "0" + num};
        return num;
      }
      startTime()
    </script>
  </body>
</html>
```



appschool

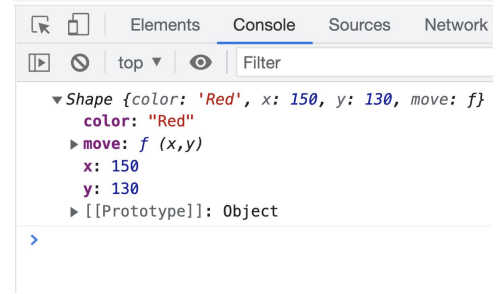
think. learn. know



oop

# Do it yourself 1

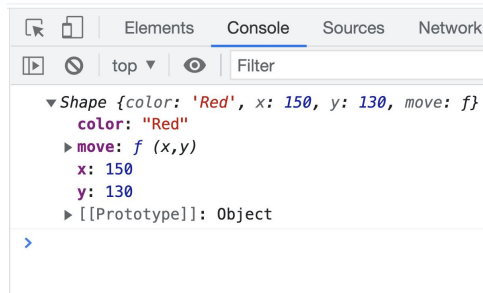
1. Create a class named Shape that contains 3 parameters. The color of the shape, the x of the shape and the y of the shape.
2. Create a constructor that receives all the parameters and initializes them
3. Add to the class a function called move that accepts as parameter x and y and changes x and y accordingly. (this.x+=x and this.y+=y)
4. Create an instance of the class with color red, x = 50 and y = 30
5. Run the move function with x = 50 and y = 60 and print the class to the console



# Do it yourself 1 Solution

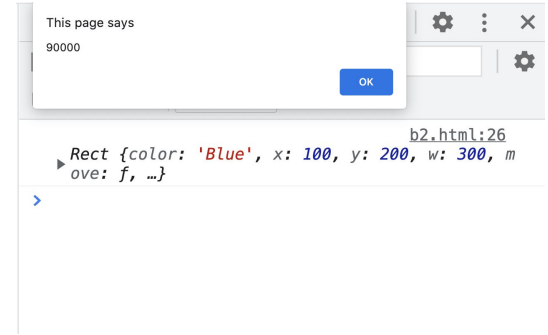
```
<html>
  <body>
    <script>
      function Shape(color,x,y) {
        this.color = color;
        this.x = x;
        this.y = y;
        this.move = function(x,y) {
          this.x+=x;
          this.y+=y;
          alert( "(" +this.x + ',' + this.y + ")" );
        };
      }
      let s1 =new Shape('Red', 50, 30);
      s1.move(100,100);
      console.log(s1)

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



# Do it yourself 2

1. Create class Rectangle that inherits from a Shape. Add two attributes to class. The width of the rectangle w, and the height of the rectangle h.
2. Add to class a function called area that calculator the rectangle area ( $w * h$ )
3. Create a class instance with red color, x 100, y 200, width 300 and height 300
4. Run the area function and print the class to the console



# Do it yourself 2 Solution

```
<html>
  <body>
    <script>
      function Shape(color,x,y) {
        this.color = color;
        this.x = x;
        this.y = y;
        this.move = function(x,y) {
          this.x+=x;
          this.y+=y;
          alert( "(" +this.x + ', ' + this.y + ")" );
        };
      }
      function Rect(color,x,y, w,h) {
        Shape.call(this, color,x,y);
        this.w = w;
        this.h =h;
        this.area = function(x,y) {
          let s= this.h*this.w;
          alert(s)
        };
      }
      let r= new Rect("Blue", 100,200, 300,300);
      console.log(r)
      r.area()
    </script>
  </body>
</html>
```

