|  |  |
| --- | --- |
| **Chapter 1 - Introduction to HTML** | **פרק 1 – מבוא ל-HTML** |

File name: skeleton.html

<!DOCTYPE html>

<html lang="en">

<head>

<title></title>

<meta charset="utf-8"/>

</head>

<body>

<h1></h1>

<p></p>

</body>

</html>

File name: helloworld.html

<!DOCTYPE html>

<html lang="en">

<head>

<title>My First Program</title>

<meta charset="utf-8"/>

</head>

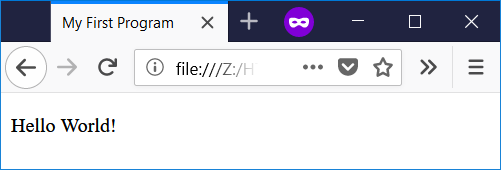
<body>

<h1>My First Program</h1>

<p>Hello World!</p>

</body>

</html>



|  |  |
| --- | --- |
| **Chapter 2 – Text and Lists** | **פרק 2 –** **טקסט ורשימות** |

File name: words.html

<!DOCTYPE html>

<html lang="en">

<head>

<title>Text Attributes</title>

<meta charset="utf-8"/>

</head>

<body>

<p>Some <b>of this text</b> is bold.</p>

<p><strong>This is strong text.</strong></p>

<p>Here <i>is text in italics.</i></p>

<p>Here is <em>emphasized text.</em></p>

<p>Here is <u>underlined text.</u></p>

<p>I can combine <b><i>bold and italics.</i></b></p>

<p><del>This is deleted text.</del></p>

<p>This is <mark>highlighted text.</mark></p>

<p>Here is some superscripted text:

a<sup>2</sup>+b<sup>2</sup>=c<sup>2</sup>

</p>

<p>Here is some subscripted text:

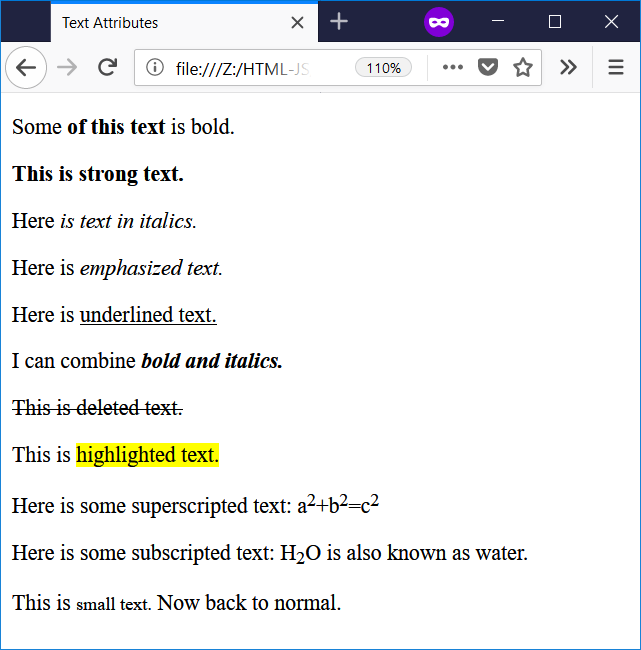
H<sub>2</sub>O is also known as water.

</p>

<p>This is <small>small text.</small> Now back to normal.</p>

</body>

</html>



File name: lists.html

<!DOCTYPE html>

<html lang="en">

<head>

<title>Lists</title>

<meta charset="utf-8"/>

</head>

<body>

<h4>An Unordered List</h4>

<ul>

<li>First item</li>

<li>Second item</li>

<li>Third item</li>

</ul>

<h4>An Ordered List</h4>

<ol>

<li>First item</li>

<li>Second item</li>

<li>Third item</li>

</ol>

<h4>An Ordered List Inside Another Ordered List</h4>

<ol>

<li>First item</li>

<li>Second item

<ol>

<li>First nested item</li>

<li>Second nested item</li>

</ol>

</li>

<li>Third item</li>

</ol>

<h4>An Ordered List Inside an Unordered List</h4>

<ul>

<li>First item</li>

<li>Second item

<ol>

<li>First nested item</li>

<li>Second nested item</li>

</ol>

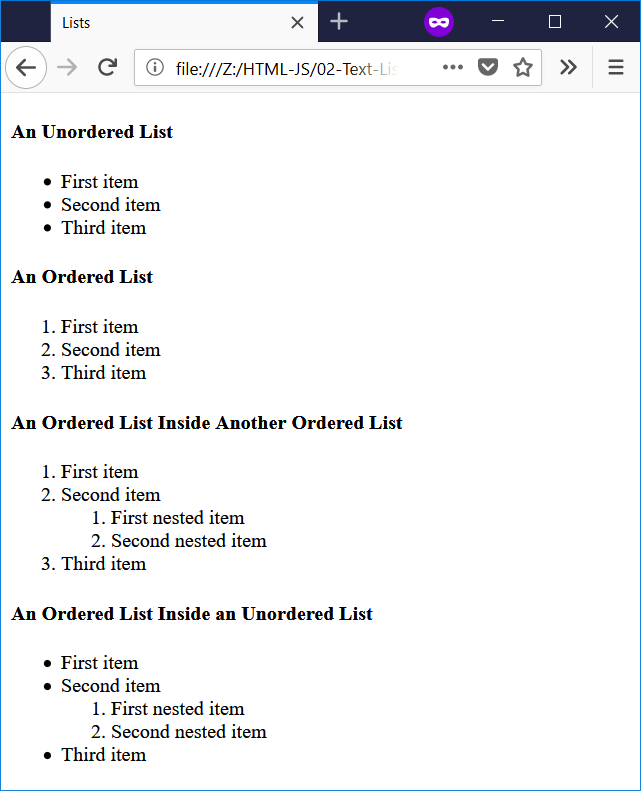
</li>

<li>Third item</li>

</ul>

</body>

</html>



|  |  |
| --- | --- |
| **Chapter 3 – Advanced HTML** | **פרק 3 – HTML מתקדם** |

File name: lists-comments.html

<!--

Filename: lists-comments.html

Author: Joshua Males

Date Created: October 12, 2018

Last Fix: October 12, 2018

Description: Document with lots of comments

-->

<!DOCTYPE html>

<html lang="en">

<head>

<title>Lists</title>

<meta charset="utf-8"/>

</head>

<body>

<h4>An Unordered List</h4>

<ul>

<li>First item</li>

<li>Second item</li>

<li>Third item</li>

</ul><!-- End of first Unordered List -->

<h4>An Ordered List</h4>

<ol>

<li>First item</li>

<li>Second item</li>

<li>Third item</li>

</ol><!-- End of first Ordered List -->

<h4>An Ordered List Inside Another Ordered List</h4>

<ol>

<li>First item</li>

<li>Second item

<ol>

<li>First nested item</li>

<li>Second nested item</li>

</ol><!-- End of nested ordered List -->

</li>

<li>Third item</li>

</ol>

<h4>An Ordered List Inside an Unordered List</h4>

<ul>

<li>First item</li>

<li>Second item

<ol>

<li>First nested item</li>

<li>Second nested item</li>

</ol><!-- End of nested ordered List -->

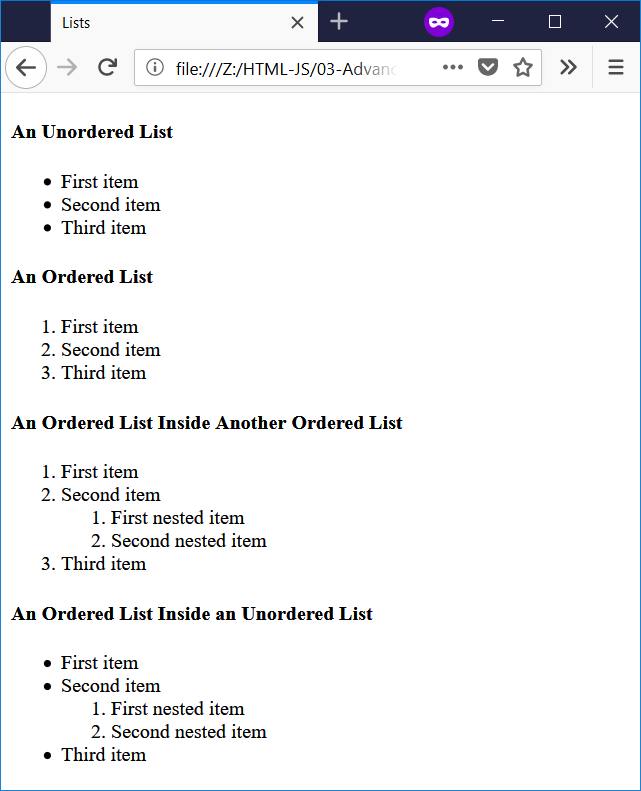
</li>

<li>Third item</li>

</ul><!-- End of Last List -->

</body>

</html>



File name: lists-comments.html

<!--

Filename: line-break-1.html

Author: Joshua Males

Date Created: October 12, 2018

Last Fix: October 12, 2018

Description: Line Break Test

-->

<!DOCTYPE html>

<html lang="en">

<head>

<title>Line Break Test</title>

<meta charset="utf-8"/>

</head>

<body>

<h1>Line Break Test</h1>

<p>

1

2

3

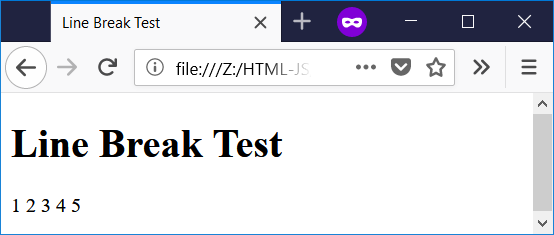
4

5

</p>

</body>

</html>



File name: line-break-2.html

<!--

Filename: line-break-2.html

Author: Joshua Males

Date Created: October 12, 2018

Last Fix: October 12, 2018

Description: Show difference between paragraph and line break

-->

<!DOCTYPE html>

<html lang="en">

<head>

<title>Line Break Test</title>

<meta charset="utf-8"/>

</head>

<body>

<h1>Line Break Test</h1>

<p>Five Paragraphs</p>

<p>1</p>

<p>2</p>

<p>3</p>

<p>4</p>

<p>5</p>

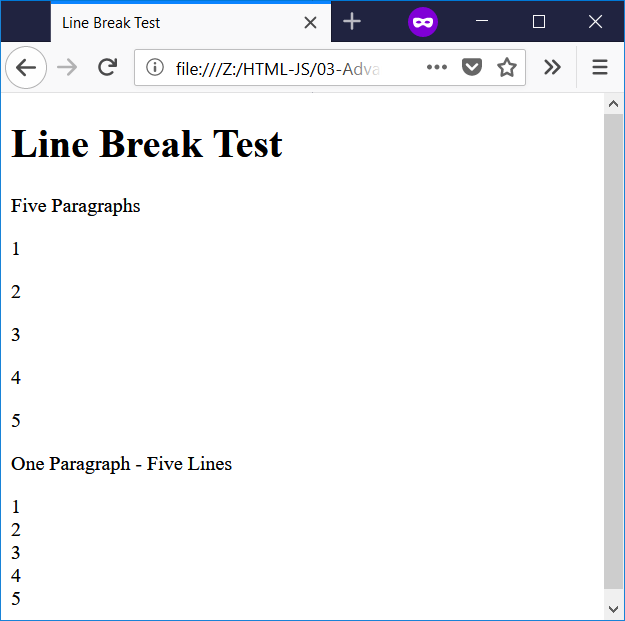
<p>One Paragraph - Five Lines</p>

<!-- A line break can be placed in the middle of a line -->

<p>1<br />2<br />3<br />4<br />5</p>

</body>

</html>



File name: sheled.html

<!--

Filename: sheled.html

Date Created: October 12, 2018

Last Fix: October 12, 2018

Description: Skeleton File for Hebrew HTML

-->

<!DOCTYPE html>

<html lang="en">

<head>

<title>

שם של העמוד

</title>

<meta charset="utf-8"/>

</head>

<body dir="rtl">

<h1>

כותרת

</h1>

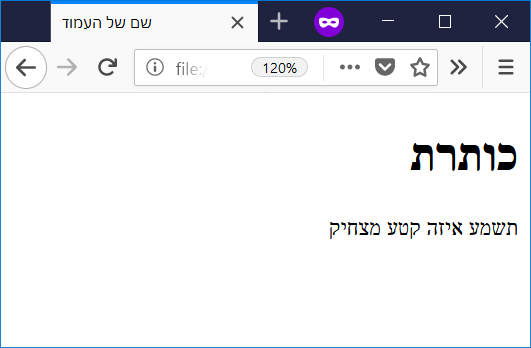
<p>

תשמע איזה קטע מצחיק

</p>

</body>

</html>



File name: fave-sites.html

<!--

Filename: fave-sites.html

Author: Joshua Males

Date Created: October 14, 2018

Last Fix: October 14, 2018

Description: Links

-->

<!DOCTYPE html>

<html lang="en">

<head>

<title>

אתרים אהובים שלי

</title>

<meta charset="utf-8"/>

</head>

<body dir="rtl">

<h1>

אתרים אהובים שלי

</h1>

<ul>

<li>

<a href="https://www.zap.co.il/">

זאפ השוואת מחירים

</a>

</li>

<li>

<a href="https://www.makorrishon.co.il/" target="\_blank">

עיתון מקור ראשון - בלשונית חדשה

</a>

</li>

<li>

<a href="http://nba.sport5.co.il/">

NBA

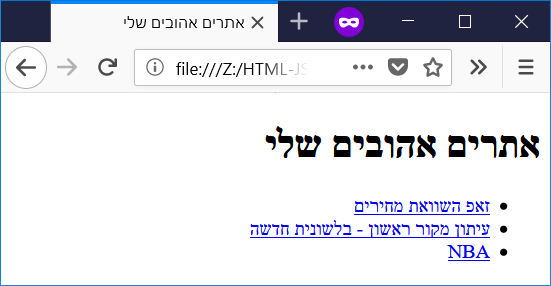
</a>

</li>

</ul>

</body>

</html>



File name: fave-sites-img.html

<!--

Filename: fave-sites-img.html

Author: Joshua Males

Date Created: October 14, 2018

Last Fix: October 14, 2018

Description: Links with Images

-->

<!DOCTYPE html>

<html lang="en">

<head>

<title>

אתרים אהובים שלי

</title>

<meta charset="utf-8"/>

</head>

<body dir="rtl">

<h1>

אתרים אהובים שלי

</h1>

<ul>

<li>

<a href="https://www.zap.co.il/">

<img src="zap-logo.PNG" >

</a>

</li>

<li>

<a href="https://www.makorrishon.co.il/" target="\_blank">

<img src="makor1-logo-1.png" >

</a>

</li>

<li>

<a href="http://nba.sport5.co.il/">

<img src="logo\_nba2.png" >

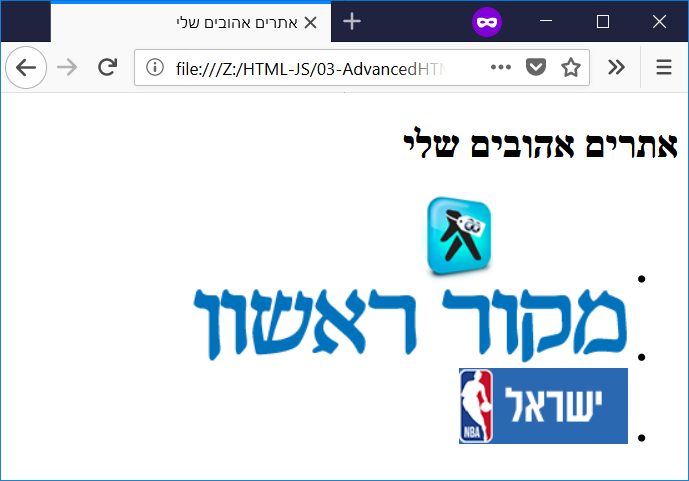
</a>

</li>

</ul>

</body>

</html>



File name: fave-comments.html

<!--

Filename: fave-comments.html

Author: Joshua Males

Date Created: October 16, 2018

Last Fix: October 16, 2018

Description: Commented Links

-->

<!DOCTYPE html>

<html lang="en">

<head>

<title>

אתרים אהובים שלי

</title>

<meta charset="utf-8"/>

</head>

<body dir="rtl">

<h1>

אתרים אהובים שלי

</h1>

<ul>

<!-- <li>

<a href="https://www.zap.co.il/">

זאפ השוואת מחירים

</a>

</li> -->

<li>

<a href="https://www.makorrishon.co.il/" target="\_blank">

עיתון מקור ראשון - בלשונית חדשה

</a>

</li>

<li>

<a href="http://nba.sport5.co.il/">

NBA

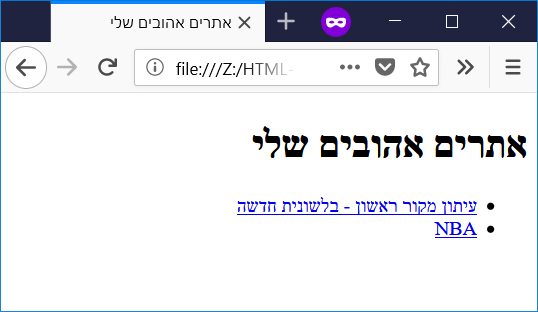
</a>

</li>

</ul>

</body>

</html>



|  |  |
| --- | --- |
| **Chapter 4 – Introduction to JavaScript** | **פרק 4 – מבוא ל-JavaScript** |

File name: js01.html

<!--

Filename: js01.html

Author: Joshua Males

Date Created: October 14, 2018

Last Fix: October 14, 2018

Description: Use javascript for math #1

-->

<!DOCTYPE html>

<html lang="en">

<head>

<title>First JavaScript Program</title>

<meta charset="utf-8"/>

</head>

<body>

<h1>Math with JavaScript</h1>

<!-- Here is the javascript for the string -->

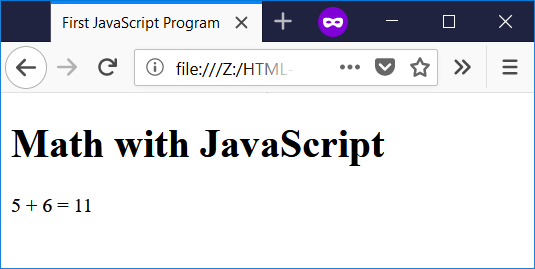
<script>document.write("5 + 6");</script>

<!-- Here is the javascript for the math -->

<script>document.write(5 + 6);</script>

</body>

</html>



File name: js02.html

<!--

Filename: js02.html

Author: Joshua Males

Date Created: October 14, 2018

Last Fix: October 14, 2018

Description: Use javascript for math #2

-->

<!DOCTYPE html>

<html lang="en">

<head>

<title>Second JavaScript Program</title>

<meta charset="utf-8"/>

</head>

<body>

<h1>More Math with JavaScript</h1>

6 + 5 = <script>document.write(6 + 5);</script><br />

6 - 5 = <script>document.write(6 - 5);</script><br />

<!-- \* is used to multiply -->

6 \* 5 = <script>document.write(6 \* 5);</script><br />

<!-- / is used to divide -->

30 / 5 = <script>document.write(30 / 5);</script><br />

6 / 5 = <script>document.write(6 / 5);</script><br />

<!-- % is used for a remainder -->

6 % 5 = <script>document.write(6 % 5);</script><br />

Whole result of 6 / 5 =

<script>document.write(Math.floor(6 / 5));</script><br />

Rounding of 1.8 =

<script>document.write(Math.round(1.8));</script><br />

Rounding of 3.49 =

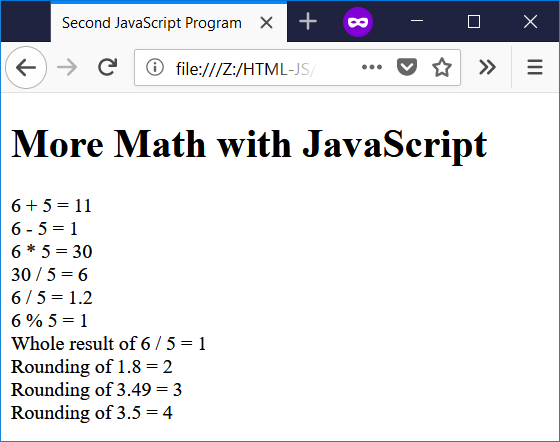
<script>document.write(Math.round(3.49));</script><br />

Rounding of 3.5 =

<script>document.write(Math.round(3.5));</script><br />

</body>

</html>



File name: js-error.html

<!--

Filename: js-error.html

Author: Joshua Males

Date Created: October 14, 2018

Last Fix: October 14, 2018

Description: JavaScript Errors

-->

<!DOCTYPE html>

<html lang="en">

<head>

<title>JavaScript Errors</title>

<meta charset="utf-8"/>

</head>

<body>

<h1>JavaScript Errors</h1>

6 + 5 =

<script>

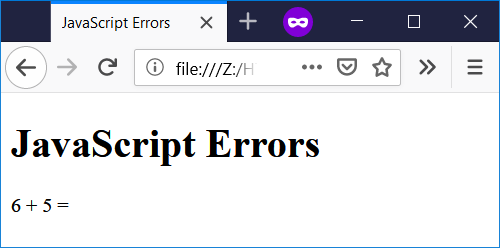
// Next line is missing a ) at the end

document.write(6 + 5;

</script><br />

</body>

</html>



File name: js-quotes.html

<!--

Filename: js-quotes.html

Author: Joshua Males

Date Created: November 1, 2018

Last Fix: November 1, 2018

Description: Printing Quote Marks

-->

<!DOCTYPE html>

<html lang="en">

<head>

<title>Printing Quote Marks</title>

<meta charset="utf-8"/>

</head>

<body>

<h1>Printing Quote Marks</h1>

<script>

// To use " inside ", you must use a backslash (\)

document.write("This has no quote marks<br />");

// This next line will not work

// document.write("This has "quote marks"<br />");

document.write("This has \"quote marks\" using a backslash<br />");

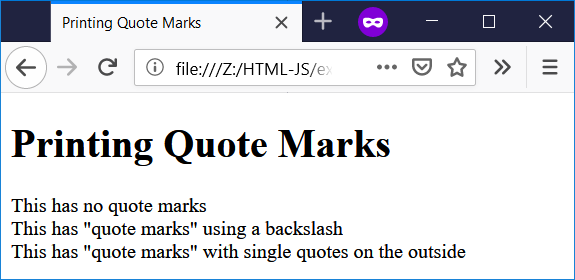
document.write(

' 'This has "quote marks" with single quotes on the outside<br />');

</script>

</body>

</html>



File name: alert.html

<!--

Filename: alert.html

Author: Joshua Males

Date Created: October 14, 2018

Last Fix: October 14, 2018

Description: Show an alert window

-->

<!DOCTYPE html>

<html lang="en">

<head>

<title>Alert Window</title>

<meta charset="utf-8"/>

</head>

<body>

<h1>Alert Window</h1>

<script>

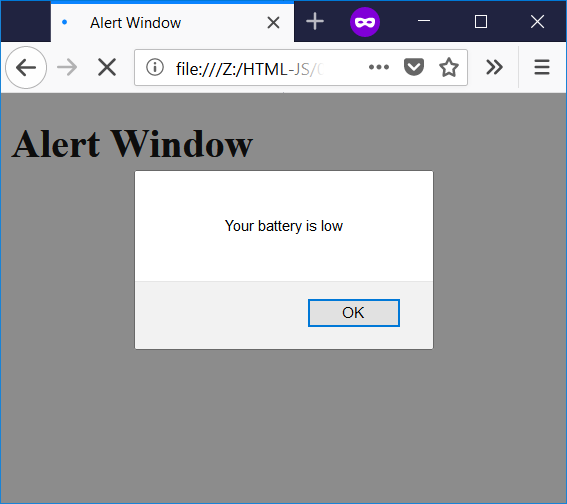
window.alert("Your battery is low");

alert("Just using alert also works");

</script>

</body>

</html>



File name: change-title.html

<!--

Filename: change-title.html

Author: Joshua Males

Date Created: October 14, 2018

Last Fix: October 14, 2018

Description: Use javascript to change the title

-->

<!DOCTYPE html>

<html lang="en">

<head>

<title>Original Title</title>

<meta charset="utf-8"/>

</head>

<body>

<h1>Change the Title</h1>

<script>

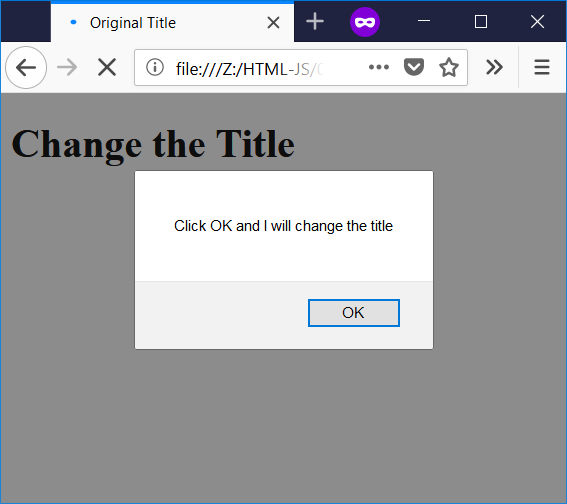
window.alert("Click OK and I will change the title");

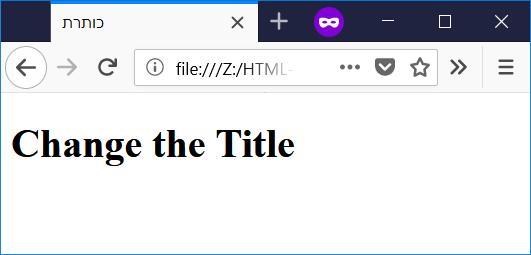
document.title = "כותרת";

</script>

</body>

</html>





File name: change-paragraph.html

<!--

Filename: change-paragraph.html

Author: Joshua Males

Date Created: October 14, 2018

Last Fix: October 14, 2018

Description: Use javascript to change the contents of a paragraph

-->

<!DOCTYPE html>

<html lang="en">

<head>

<title>Change Text in Paragraph</title>

<meta charset="utf-8"/>

</head>

<body>

<h1>Change Text in Paragraph</h1>

<p id="output">This is my output area</p>

<script>

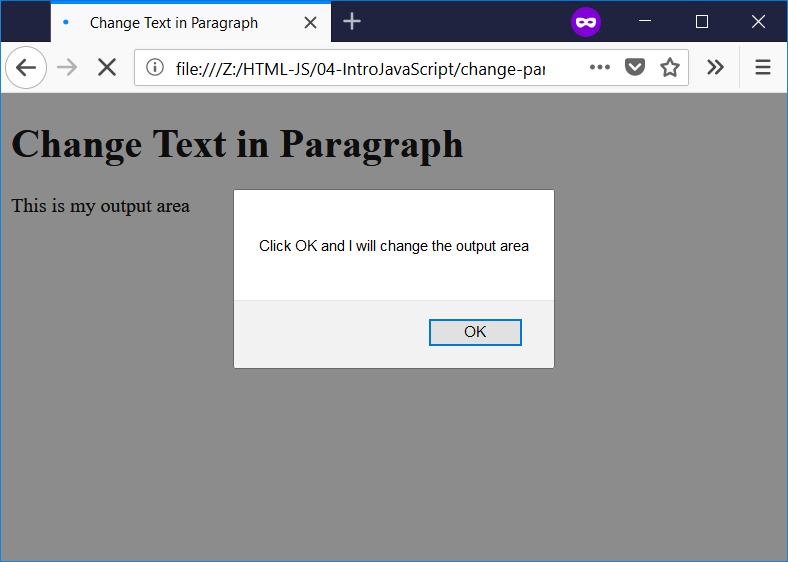
window.alert("Click OK and I will change the output area");

document.getElementById('output').innerHTML = 5 + 6;

</script>

</body>

</html>





|  |  |
| --- | --- |
| **Chapter 5 – Variables and Math** | **פרק 5 – משתנים ומתמטיקה** |

File name: var-define1.html

<!--

Filename: var-define1.html

Author: Joshua Males

Date Created: October 14, 2018

Last Fix: October 14, 2018

Description: Define JS variables

-->

<!DOCTYPE html>

<html lang="en">

<head>

<title>Define Variables #1</title>

<meta charset="utf-8"/>

</head>

<body>

<h1>Define Variables #1</h1>

<script>

let x;

let y;

x = 7;

y = 9;

document.write("x = " + x);

document.write("<br />"); // Jump to a new line

document.write("y = " + y);

</script>

</body>

</html>



File name: var-define2.html

<!--

Filename: var-define2.html

Author: Joshua Males

Date Created: October 14, 2018

Last Fix: October 21, 2018

Description: Define more JS variables

-->

<!DOCTYPE html>

<html lang="en">

<head>

<title>Define Variables #2</title>

<meta charset="utf-8"/>

</head>

<body>

<h1>Define Variables #2</h1>

<script>

let x = 7;

let y = 9;

z = x + y;

document.write("x = " + x);

document.write("<br />"); // Jump to a new line

document.write("y = " + y);

document.write("<br />"); // Jump to a new line

document.write("z = " + z);

</script>

</body>

</html>



File name: var-define3.html

<!--

Filename: var-define3.html

Author: Joshua Males

Date Created: October 14, 2018

Last Fix: October 14, 2018

Description: Use an undefined JS variable

-->

<!DOCTYPE html>

<html lang="en">

<head>

<title>Define Variables #3</title>

<meta charset="utf-8"/>

</head>

<body>

<h1>Define Variables #3</h1>

<script>

let x = 7;

let y = 9;

document.write("z = ");

document.write(z);

</script>

</body>

</html>



File name: arithmetic-operators.html

<!--

Filename: arithmetic-operators.html

Author: Joshua Males

Date Created: October 14, 2018

Last Fix: October 14, 2018

Description: Use of Arithmetic Operators

-->

<!DOCTYPE html>

<html lang="en">

<head>

<title>Arithmetic Operators</title>

<meta charset="utf-8"/>

</head>

<body>

<h1>Arithmetic Operators</h1>

<script>

let x = 7; // This is a JavaScript comment

let y = 5; /\* This is also a JavaScript comment \*/

</script>

x = <script>document.write(x);</script><br />

y = <script>document.write(y);</script><br />

x + y = <script>document.write(x + y);</script><br />

x - y = <script>document.write(x - y);</script><br />

x \* y = <script>document.write(x \* y);</script><br />

x / y = <script>document.write(x / y);</script><br />

x % y = <script>document.write(x % y);</script><br />

<!-- Do math before output -->

--x = <script>document.write(--x);</script><br />

++y = <script>document.write(++y);</script><br />

<!-- Do math after output -->

x-- = <script>document.write(x--);</script><br />

y++ = <script>document.write(y++);</script><br />

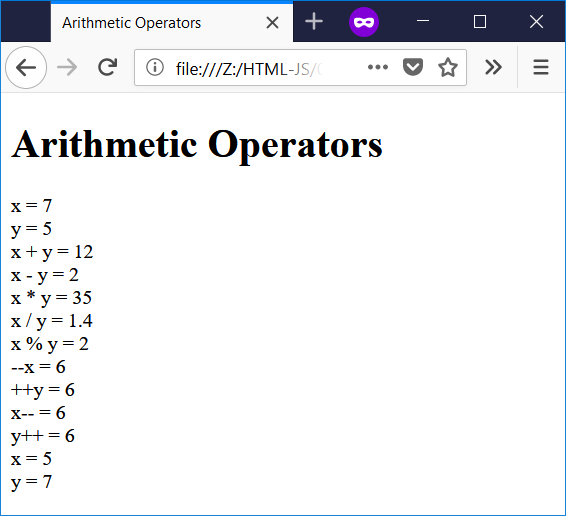
<!-- Now show the real value -->

x = <script>document.write(x);</script><br />

y = <script>document.write(y);</script><br />

</body>

</html>



File name: assignment-operators.html

<!--

Filename: assignment-operators.html

Author: Joshua Males

Date Created: October 14, 2018

Last Fix: October 14, 2018

Description: Use of Assignment Operators

-->

<!DOCTYPE html>

<html lang="en">

<head>

<title>Assignment Operators</title>

<meta charset="utf-8"/>

</head>

<body>

<h1>Assignment Operators</h1>

<script>

let x = 7; // This is a JavaScript comment

let y = 5; /\* This is also a JavaScript comment \*/

</script>

x = <script>document.write(x);</script><br />

y = <script>document.write(y);</script><br />

<!-- Do multiple commands inside script -->

<!-- &emsp; is a wide space in HTML -->

x += y &emsp; x now equals

<script>x += y; document.write(x);</script><br />

x -= y &emsp; x now equals <script>x -= y;

document.write(x);</script><br />

x \*= y &emsp; x now equals <script>x \*= y;

document.write(x);</script><br />

x /= y &emsp; x now equals <script>x /= y;

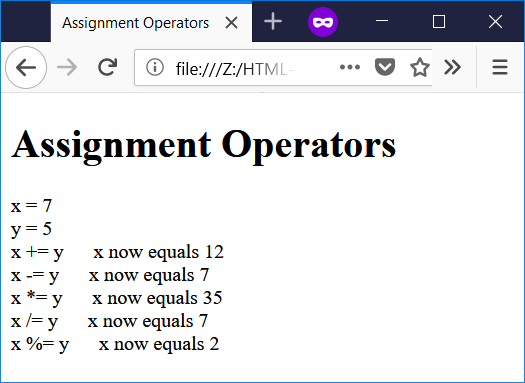
document.write(x);</script><br />

x %= y &emsp; x now equals <script>x %= y;

document.write(x);</script><br />

</body>

</html>



File name: var-define4.html

<!--

Filename: var-define4.html

Author: Joshua Males

Date Created: October 14, 2018

Last Fix: October 14, 2018

Description: Combine Variables

-->

<!DOCTYPE html>

<html lang="en">

<head>

<title>Combine Variables</title>

<meta charset="utf-8"/>

</head>

<body>

<h1>Combine Variables</h1>

<script>

let twoNumbers = 5 + 7;

let stringNumber = "5" + 7;

let numberString = 5 + "7";

document.write("twoNumbers (5+7) = " + twoNumbers + "<br />");

document.write("stringNumber (\"5\" + 7) = " + stringNumber +

"<br />");

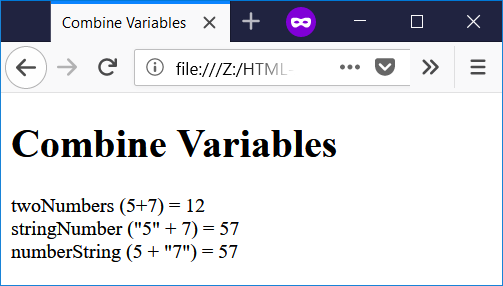
document.write("numberString (5 + \"7\") = " + numberString +

"<br />");

</script>

</body>

</html>



File name: constants.html

<!--

Filename: constants.html

Author: Joshua Males

Date Created: June 18, 2019

Last Fix: June 18, 2019

Description: Constants

-->

<!DOCTYPE html>

<html lang="en">

<head>

<title>Constants</title>

<meta charset="utf-8"/>

</head>

<body>

<h1>Constants</h1>

<script>

let pi = 3.14159; // This can change

const PI = 3.14159; // This cannot change

let radius = 2;

let area = PI \* (radius \* radius);

document.write("Radius #1 = " + radius + ". " +

"PI #1 = " + PI + ". " +

"Area = " + area + "<br />");

pi = 3.14;

area = pi \* (radius \* radius);

document.write("Radius #2 = " + radius + ". " +

"PI #2 = " + pi + ". " +

"Area = " + area + "<br />");

PI = 3.14; // This will produce an error. Look at the console.

area = PI \* (radius \* radius);

document.write("Radius #3 = " + radius + ". " +

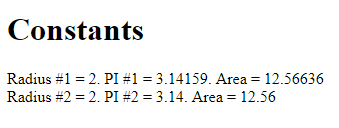
"PI #3 = " + PI + ". " +

"Area = " + area + "<br />");

</script>

</body>

</html>



File name: console.html

<!--

Filename: console.html

Author: Joshua Males

Date Created: July 3, 2019

Last Fix: July 3, 2019

Description: Console Use

-->

<!DOCTYPE html>

<html lang="en">

<head>

<title>Console Use</title>

<meta charset="utf-8"/>

</head>

<body>

<h1>Console Use - Press F12</h1>

<script>

let twoNumbers = 5 + 7;

console.log(twoNumbers);

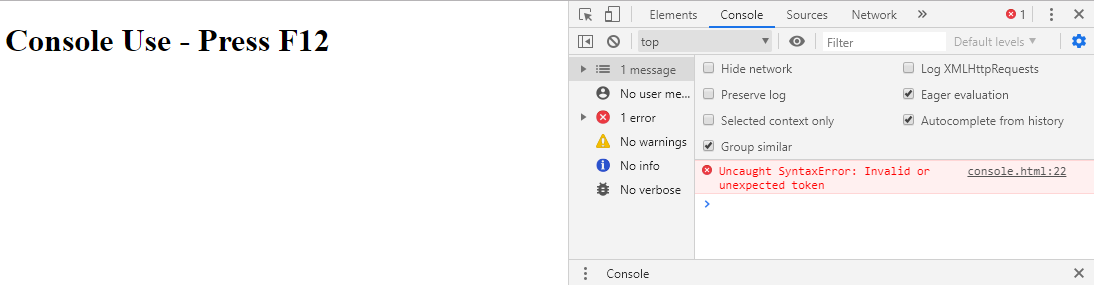
document.write("Here is a Syntax Error"); // missing " at end

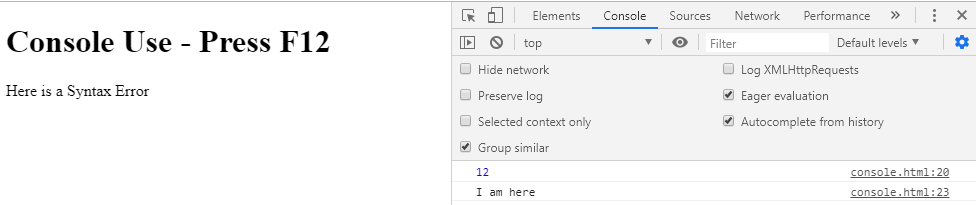
console.log("I am here");

</script>

</body>

</html>





|  |  |
| --- | --- |
| **Chapter 6 – Input and Output** | **פרק 6 – קלט ופלט** |

File name: button1.html

<!--

Filename: button1.html

Author: Joshua Males

Date Created: October 15, 2018

Last Fix: October 15, 2018

Description: Use of boring button

-->

<!DOCTYPE html>

<html lang="en">

<head>

<title>Button #1</title>

<meta charset="utf-8"/>

</head>

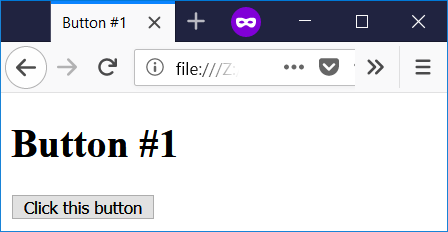
<body>

<h1>Button #1</h1>

<button>Click this button</button>

</body>

</html>



File name: button2.html

<!--

Filename: button2.html

Author: Joshua Males

Date Created: October 14, 2018

Last Fix: October 15, 2018

Description: Button changes itself

-->

<!DOCTYPE html>

<html lang="en">

<head>

<title>Button #2</title>

<meta charset="utf-8"/>

</head>

<body>

<h1>Button #2</h1>

<script>

let x = 0;

let y = 0;

</script>

<!-- Every click increments the number in the button -->

<button id="button1"

onclick="document.getElementById('button1').innerHTML = ++x">

0

</button>

<!-- Every click increments the number in the paragraph below -->

<button id="button2"

onclick="document.getElementById('keta').innerHTML += ++y +

'<br />'">

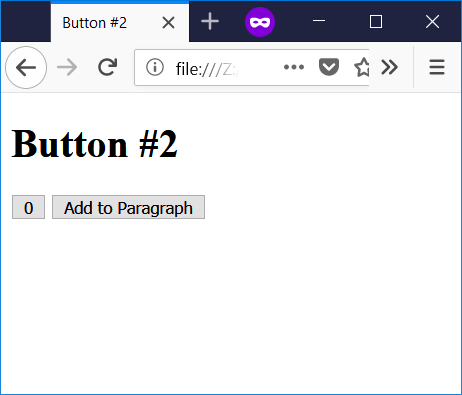
Add to Paragraph

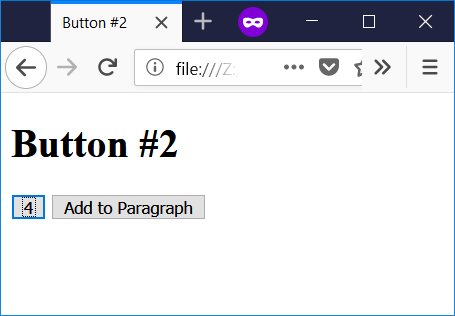
</button>

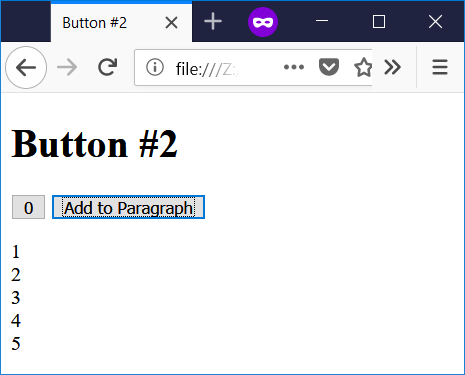
<p id="keta"></p>

</body>

</html>







File name: erase-page.html

<!--

Filename: erase-page.html

Author: Joshua Males

Date Created: October 14, 2018

Last Fix: October 14, 2018

Description: Use document.write to erase the page

-->

<!DOCTYPE html>

<html lang="en">

<head>

<title>Erase Page</title>

<meta charset="utf-8"/>

</head>

<body>

<h1>Erase Page</h1>

<p>Here is some text.</p>

<button type="button"

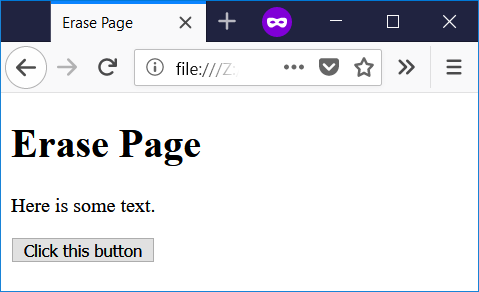
onclick='document.write("Oops. I erased the page.")'>

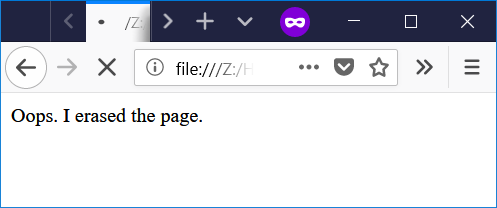
Click this button

</button>

</body>

</html>





File name: button3.html

<!--

Filename: button3.html

Author: Joshua Males

Date Created: October 14, 2018

Last Fix: October 15, 2018

Description: Button calls function

-->

<!DOCTYPE html>

<html lang="en">

<head>

<title>Button #3</title>

<meta charset="utf-8"/>

</head>

<body>

<h1>Button #3</h1>

<!-- Every click increments the number in the button -->

<!-- Every click increments the number in the paragraph below -->

<button id="button1" onclick="doThings()">0</button>

<p id="keta"></p>

<script>

let x = 0;

function doThings()

{

x++;

document.getElementById('button1').innerHTML = x;

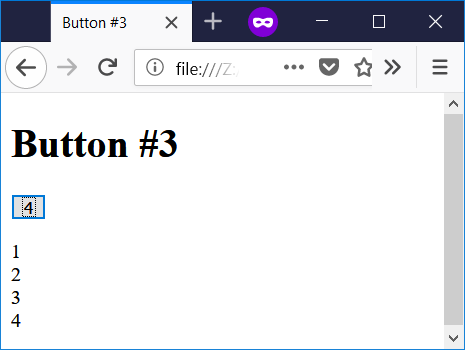
document.getElementById('keta').innerHTML += x + '<br />';

} // end of doThings()

</script>

</body>

</html>



File name: button4.html

<!--

Filename: button4.html

Author: Joshua Males

Date Created: October 14, 2018

Last Fix: October 15, 2018

Description: Button calls function #2

-->

<!DOCTYPE html>

<html lang="en">

<head>

<title>Button #4</title>

<meta charset="utf-8"/>

</head>

<body>

<h1 id="koteret">Button #4</h1>

<!-- Every click increments the number in the button -->

<!-- Every click increments the number in the paragraph below -->

<button id="button1" onclick="doThings()">0</button>

<p id="keta"></p>

<script>

let x = 0;

function doThings()

{

x++;

document.getElementById('button1').innerHTML = x;

document.getElementById('keta').innerHTML += x + '<br />';

document.title = "x=" + x;

document.getElementById('koteret').innerHTML = "Button: x=" + x;

} // end of doThings()

</script>

</body>

</html>



File name: input1.html

<!--

Filename: input1.html

Author: Joshua Males

Date Created: October 15, 2018

Last Fix: October 15, 2018

Description: First use of input boxes

-->

<!DOCTYPE html>

<html lang="en">

<head>

<title>Input Boxes #1</title>

<meta charset="utf-8"/>

</head>

<body>

<h1>Input Boxes #1</h1>

x: <input type="number" id="xinput" value="0">

<br><br>

y: <input type="number" id="yinput" value="0">

<br><br>

<button id="button1" onclick="doThings()">Add</button>

<p id="keta"></p>

<script>

function doThings()

{

// Get the value from the input box and store in x

let x = document.getElementById("xinput").value;

// Get the value from the input box and store in y

let y = document.getElementById("yinput").value;

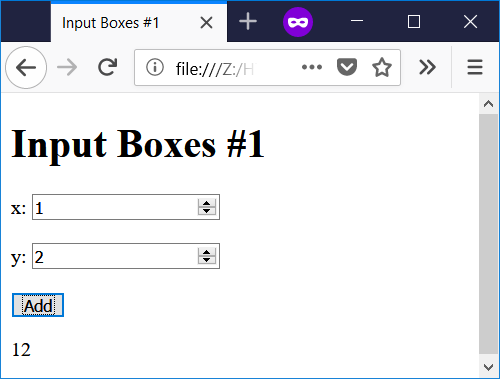
document.getElementById('keta').innerHTML = x + y;

} // end of doThings()

</script>

</body>

</html>



File name: input2.html

<!--

Filename: input2.html

Author: Joshua Males

Date Created: October 15, 2018

Last Fix: October 15, 2018

Description: Use of input boxes with parseInt

-->

<!DOCTYPE html>

<html lang="en">

<head>

<title>Input Boxes #2</title>

<meta charset="utf-8"/>

</head>

<body>

<h1>Input Boxes #2</h1>

x: <input type="number" id="xinput" value="0">

<br><br>

y: <input type="number" id="yinput" value="0">

<br><br>

<button id="button1" onclick="doThings()">Add</button>

<p id="keta"></p>

<script>

function doThings()

{

// Get the value from the input box and store in x

let x = document.getElementById("xinput").value;

x = parseInt(x);

// Get the value from the input box and store in y

let y = document.getElementById("yinput").value;

y = parseInt(y);

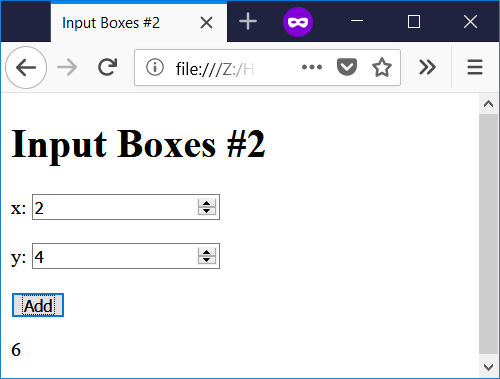
document.getElementById('keta').innerHTML = x + y;

} // end of doThings()

</script>

</body>

</html>



File name: input3.html

<!--

Filename: input3.html

Author: Joshua Males

Date Created: October 15, 2018

Last Fix: October 15, 2018

Description: Use of input boxes with text plus parseInt

-->

<!DOCTYPE html>

<html lang="en">

<head>

<title>Input Boxes #3</title>

<meta charset="utf-8"/>

</head>

<body>

<h1>Input Boxes #3</h1>

x: <input type="number" id="xinput" value="0">

<br><br>

y: <input type="number" id="yinput" value="0">

<br><br>

<button id="button1" onclick="doThings()">Add</button>

<p id="keta"></p>

<script>

function doThings()

{

// Get the value from the input box and store in x

let x = document.getElementById("xinput").value;

x = parseInt(x);

// Get the value from the input box and store in y

let y = document.getElementById("yinput").value;

y = parseInt(y);

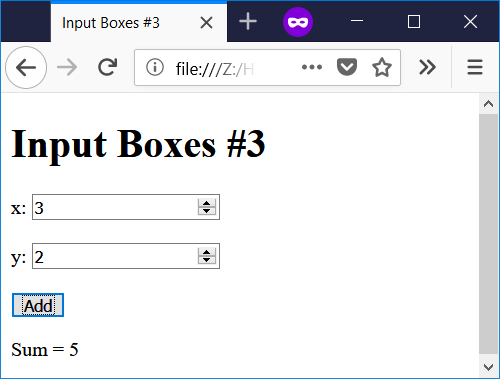
document.getElementById('keta').innerHTML = "Sum = " + (x + y) ;

} // end of doThings()

</script>

</body>

</html>



File name: input4.html

<!--

Filename: input4.html

Author: Joshua Males

Date Created: October 15, 2018

Last Fix: October 15, 2018

Description: Use of input boxes with Hebrew

-->

<!DOCTYPE html>

<html lang="en">

<head>

<title>קופסאות קלט</title>

<meta charset="utf-8"/>

</head>

<body dir="rtl">

<h1>קופסאות קלט</h1>

x: <input type="number" id="xinput" value="0">

<br><br>

y: <input type="number" id="yinput" value="0">

<br><br>

<button id="button1" onclick="doThings()">לחבר</button>

<p id="keta"></p>

<script>

function doThings()

{

// Get the value from the input box and store in x

let x = document.getElementById("xinput").value;

x = parseInt(x);

// Get the value from the input box and store in y

let y = document.getElementById("yinput").value;

y = parseInt(y);

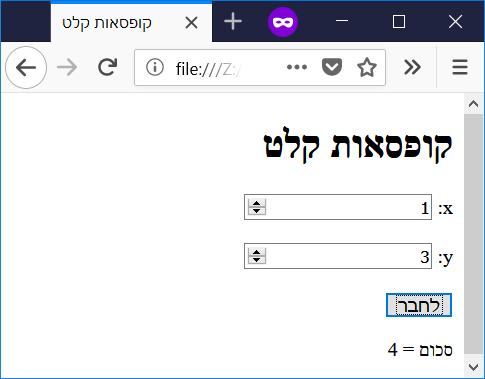
document.getElementById('keta').innerHTML = "סכום = " + (x + y) ;

} // end of doThings()

</script>

</body>

</html>



|  |  |
| --- | --- |
| **Chapter 7 - Conditions** | **פרק 7 – תנאים** |

File name: cond1.html

<!--

Filename: cond1.html

Author: Joshua Males

Date Created: October 16, 2018

Last Fix: October 16, 2018

Description: Compare Two Numbers

-->

<!DOCTYPE html>

<html lang="en">

<head>

<title>Compare Two Numbers</title>

<meta charset="utf-8"/>

</head>

<body>

<h1>Compare Two Numbers</h1>

x: <input type="number" id="xinput" value="0">

<br><br>

y: <input type="number" id="yinput" value="0">

<br><br>

<button id="button1" onclick="doThings()">Compare</button>

<p id="answer"></p>

<script>

function doThings()

{

// Get the value from the input box and store in x

let x = parseInt(document.getElementById("xinput").value);

// Get the value from the input box and store in y

let y = parseInt(document.getElementById("yinput").value);

if (x == y)

// One command. No brackets needed.

document.getElementById('answer').innerHTML =

"Numbers are equal" ;

if (x != y)

document.getElementById('answer').innerHTML =

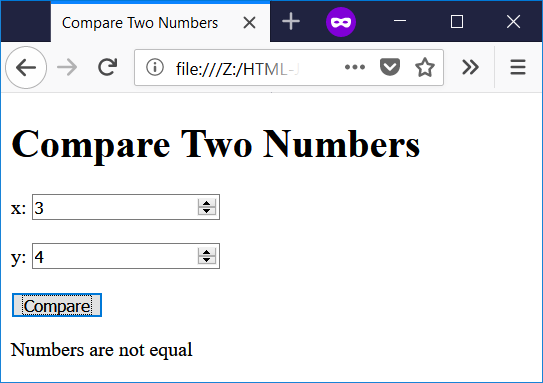
"Numbers are not equal" ;

} // end of doThings()

</script>

</body>

</html>



File name: cond-zero.html

<!--

Filename: cond-zero.html

Author: Joshua Males

Date Created: October 16, 2018

Last Fix: October 16, 2018

Description: Check for Zero Value

-->

<!DOCTYPE html>

<html lang="en">

<head>

<title>Check for Zero Value</title>

<meta charset="utf-8"/>

</head>

<body>

<h1>Check for Zero Value</h1>

x: <input type="number" id="xinput" value="0">

<br><br>

<button id="button1" onclick="doThings()">Check</button>

<p id="answer"></p>

<script>

function doThings()

{

// Get the value from the input box and store in x

let x = parseInt(document.getElementById("xinput").value);

if (x)

document.getElementById('answer').innerHTML =

"Number is Non-Zero" ;

if (!x)

document.getElementById('answer').innerHTML =

"Number is Zero" ;

if (x % 2)

document.getElementById('answer').innerHTML +=

"<br />Number is Odd" ;

if (!(x % 2))

document.getElementById('answer').innerHTML +=

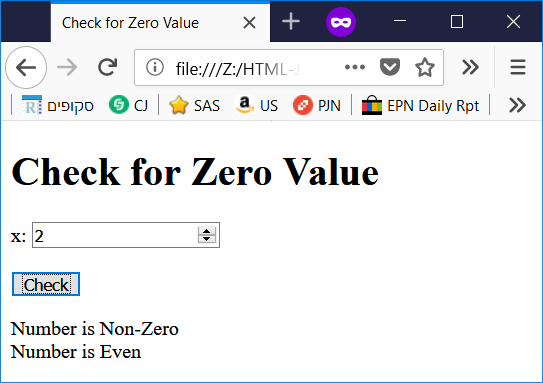
"<br />Number is Even" ;

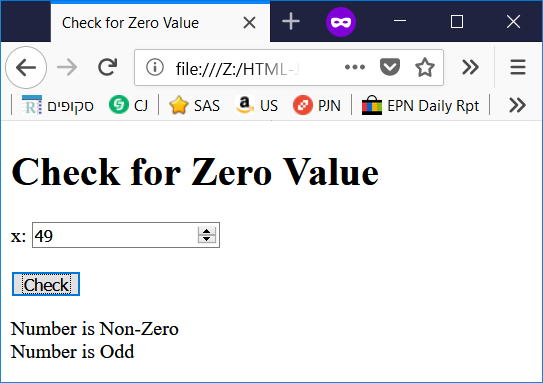
} // end of doThings()

</script>

</body>

</html>





File name: cond-else.html

<!--

Filename: cond-else.html

Author: Joshua Males

Date Created: October 16, 2018

Last Fix: October 16, 2018

Description: Compare Two Numbers with "else"

-->

<!DOCTYPE html>

<html lang="en">

<head>

<title>Compare Two Numbers with "else"</title>

<meta charset="utf-8"/>

</head>

<body>

<h1>Compare Two Numbers with "else"</h1>

x: <input type="number" id="xinput" value="0">

<br><br>

y: <input type="number" id="yinput" value="0">

<br><br>

<button id="button1" onclick="doThings()">Compare</button>

<p id="answer"></p>

<script>

function doThings()

{

// Get the value from the input box and store in x

let x = parseInt(document.getElementById("xinput").value);

// Get the value from the input box and store in y

let y = parseInt(document.getElementById("yinput").value);

if (x == y) {

document.getElementById('answer').innerHTML =

"Numbers are equal" ;

}

else {

document.getElementById('answer').innerHTML =

"Numbers are not equal" ;

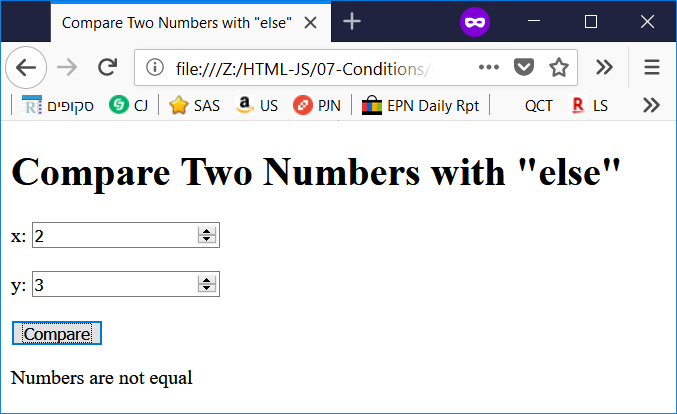
}

} // end of doThings()

</script>

</body>

</html>



File name: cond-else2.html

<!--

Filename: cond-else2.html

Author: Joshua Males

Date Created: October 16, 2018

Last Fix: October 16, 2018

Description: Compare Two Numbers with multiple "else"

-->

<!DOCTYPE html>

<html lang="en">

<head>

<title>Compare Two Numbers with multiple "else"</title>

<meta charset="utf-8"/>

</head>

<body>

<h1>Compare Two Numbers with multiple "else"</h1>

x: <input type="number" id="xinput" value="0">

<br><br>

y: <input type="number" id="yinput" value="0">

<br><br>

<button id="button1" onclick="doThings()">Compare</button>

<p id="answer"></p>

<script>

function doThings()

{

// Get the value from the input box and store in x

let x = parseInt(document.getElementById("xinput").value);

// Get the value from the input box and store in y

let y = parseInt(document.getElementById("yinput").value);

if (x == y) {

document.getElementById('answer').innerHTML =

"Numbers are equal" ;

}

else if (x < y) {

document.getElementById('answer').innerHTML =

"x is less than y" ;

}

else {

document.getElementById('answer').innerHTML =

"y is greater than x" ;

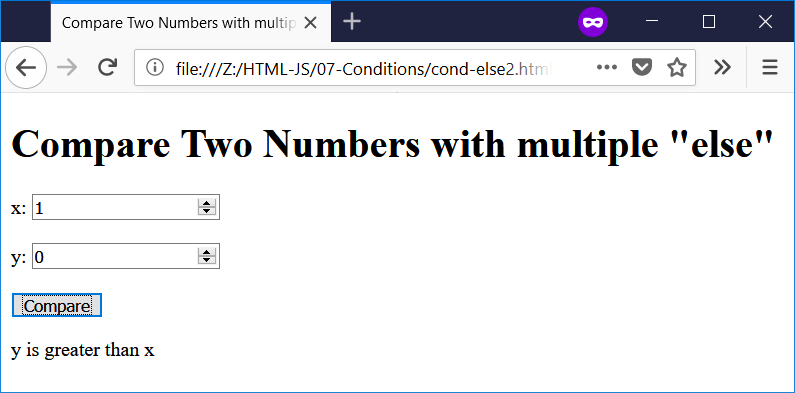
}

} // end of doThings()

</script>

</body>

</html>



File name: cond-else-nested.html

<!--

Filename: cond-else-nested.html

Author: Joshua Males

Date Created: October 16, 2018

Last Fix: October 16, 2018

Description: Compare Two Numbers with Nesting

-->

<!DOCTYPE html>

<html lang="en">

<head>

<title>Compare Two Numbers with Nesting</title>

<meta charset="utf-8"/>

</head>

<body>

<h1>Compare Two Numbers with Nesting</h1>

x: <input type="number" id="xinput" value="0">

<br><br>

y: <input type="number" id="yinput" value="0">

<br><br>

<button id="button1" onclick="doThings()">Compare</button>

<p id="answer"></p>

<script>

function doThings()

{

// Get the value from the input box and store in x

let x = parseInt(document.getElementById("xinput").value);

// Get the value from the input box and store in y

let y = parseInt(document.getElementById("yinput").value);

if (x == y) {

document.getElementById('answer').innerHTML =

"Numbers are equal";

}

else {

// We are nesting an if-else inside the else

if (x < y) {

document.getElementById('answer').innerHTML =

"x is less than y" ;

}

else {

document.getElementById('answer').innerHTML =

"y is greater than x" ;

}

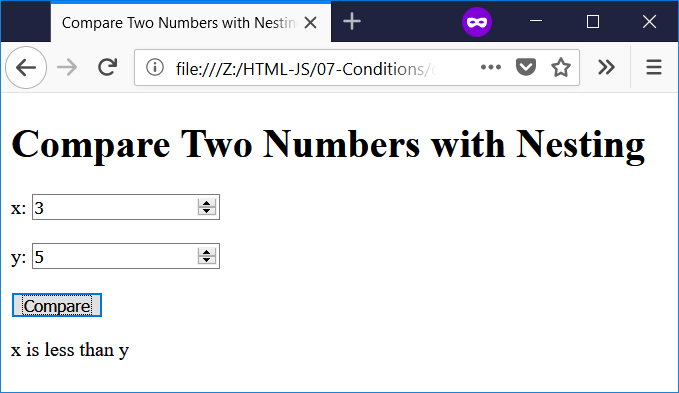
}

} // end of doThings()

</script>

</body>

</html>



File name: cond-multiple.html

<!--

Filename: cond-multiple.html

Author: Joshua Males

Date Created: October 17, 2018

Last Fix: November 6, 2018

Description: Multiple Comparisons

-->

<!DOCTYPE html>

<html lang="en">

<head>

<title>Multiple Comparisons</title>

<meta charset="utf-8"/>

</head>

<body>

<h1>Multiple Comparisons</h1>

x: <input type="number" id="xinput" value="0">

<br><br>

y: <input type="number" id="yinput" value="0">

<br><br>

<button id="button1" onclick="doThings()">Compare</button>

<p id="answer"></p>

<script>

function doThings()

{

// Get the value from the input box and store in x

let x = parseInt(document.getElementById("xinput").value);

// Get the value from the input box and store in y

let y = parseInt(document.getElementById("yinput").value);

// Clear the answer

document.getElementById('answer').innerHTML = "";

if (x && y) {

document.getElementById('answer').innerHTML +=

"<br />Both numbers are non-zero";

}

if (!x || !y) {

document.getElementById('answer').innerHTML +=

"<br />At least one number is zero";

}

if ((x % 2) && (y % 2)) {

document.getElementById('answer').innerHTML +=

"<br />Both numbers are odd";

}

if (!(x % 2) || !(y % 2)) {

document.getElementById('answer').innerHTML +=

"<br />At least one number is even";

}

if ((x >= 100) && (x < 10000)) {

document.getElementById('answer').innerHTML +=

"<br />x has 3 or 4 digits";

}

if (((x >= 100) && (x < 10000)) && (x % 2)) {

document.getElementById('answer').innerHTML +=

"<br />x has 3 or 4 digits and is odd";

}

if ((1) || (++x < 10000)) {

document.getElementById('answer').innerHTML +=

"<br />OR Test: x = " + x;

}

if ((0) && (++x < 10000)) {

document.getElementById('answer').innerHTML +=

"<br />AND Test (true): x = " + x;

} else

document.getElementById('answer').innerHTML +=

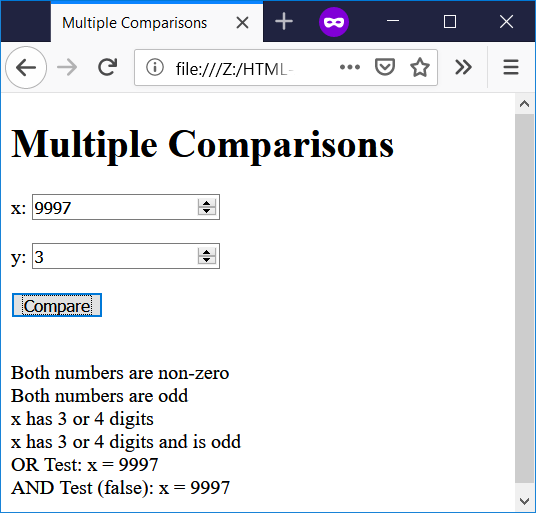
"<br />AND Test (false): x = " + x;

} // end of doThings()

</script>

</body>

</html>



File name: cond-assignment.html

<!--

Filename: cond-assignment.html

Author: Joshua Males

Date Created: October 17, 2018

Last Fix: October 22, 2018

Description: Compare with Assignment

-->

<!DOCTYPE html>

<html lang="en">

<head>

<title>Compare with Assignment</title>

<meta charset="utf-8"/>

</head>

<body>

<h1>Compare with Assignment</h1>

x: <input type="number" id="xinput" value="0">

<br><br>

y: <input type="number" id="yinput" value="0">

<br><br>

<button id="button1" onclick="doThings()">Compare</button>

<p id="answer"></p>

<script>

function doThings()

{

// Get the value from the input box and store in x

let x = parseInt(document.getElementById("xinput").value);

// Get the value from the input box and store in y

let y = parseInt(document.getElementById("yinput").value);

if (x = y)

document.getElementById('answer').innerHTML =

"Numbers are equal. " +

"x = " + x + ". y = " + y + ".";

if (x != y)

document.getElementById('answer').innerHTML =

"Numbers are not equal. " +

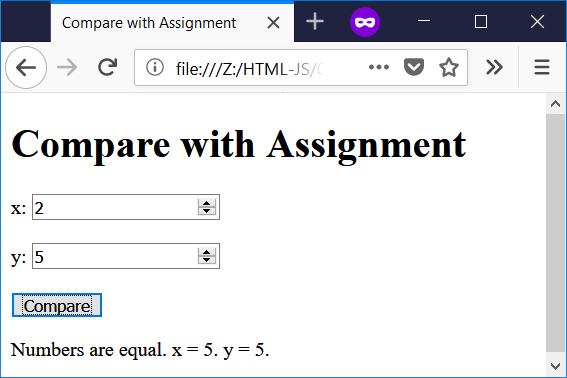
"x = " + x + ". y = " + y + ".";

} // end of doThings()

</script>

</body>

</html>



File name: cond-short.html

<!--

Filename: cond-short.html

Author: Joshua Males

Date Created: October 17, 2018

Last Fix: October 17, 2018

Description: Short Compare

-->

<!DOCTYPE html>

<html lang="en">

<head>

<title>Short Compare</title>

<meta charset="utf-8"/>

</head>

<body>

<h1>Short Compare</h1>

x: <input type="number" id="xinput" value="0">

<br><br>

y: <input type="number" id="yinput" value="0">

<br><br>

<button id="button1" onclick="doThings()">Compare</button>

<p id="answer"></p>

<script>

function doThings()

{

// Get the value from the input box and store in x

let x = parseInt(document.getElementById("xinput").value);

// Get the value from the input box and store in y

let y = parseInt(document.getElementById("yinput").value);

document.getElementById('answer').innerHTML =

"Numbers are " +

(x == y ? "" : "not ") +

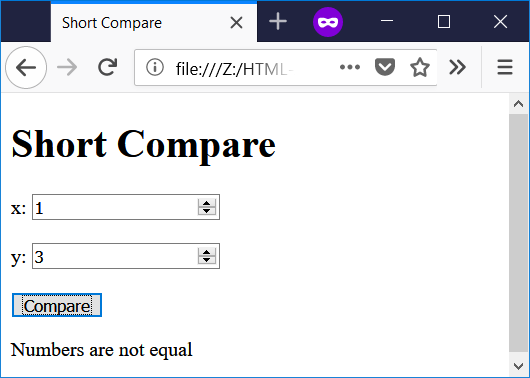
"equal" ;

} // end of doThings()

</script>

</body>

</html>



File name: cond-check-input.html

<!--

Filename: cond-check-input.html

Author: Joshua Males

Date Created: October 17, 2018

Last Fix: October 22, 2018

Description: Short Compare

-->

<!DOCTYPE html>

<html lang="en">

<head>

<title>Check Input</title>

<meta charset="utf-8"/>

</head>

<body>

<h1>Check Input</h1>

x: <input type="number" id="xinput" value="0">

<br><br>

y: <input type="number" id="yinput" value="0">

<br><br>

<button id="button1" onclick="doDivisionWithCheck()">

Get x / y with Check</button><br />

<button id="button2" onclick="doDivisionWithoutCheck()">

Get x / y without Check</button>

<p id="answer"></p>

<script>

function doDivisionWithCheck()

{

// Get the value from the input box and store in x

let x = parseInt(document.getElementById("xinput").value);

// Get the value from the input box and store in y

let y = parseInt(document.getElementById("yinput").value);

if (y == 0) {

document.getElementById('answer').innerHTML = "";

window.alert("ERROR: Division by Zero!");

}

else

document.getElementById('answer').innerHTML = x / y ;

} // end of doDivisionWithCheck()

//==============================================================

function doDivisionWithoutCheck()

{

// Get the value from the input box and store in x

let x = parseInt(document.getElementById("xinput").value);

// Get the value from the input box and store in y

let y = parseInt(document.getElementById("yinput").value);

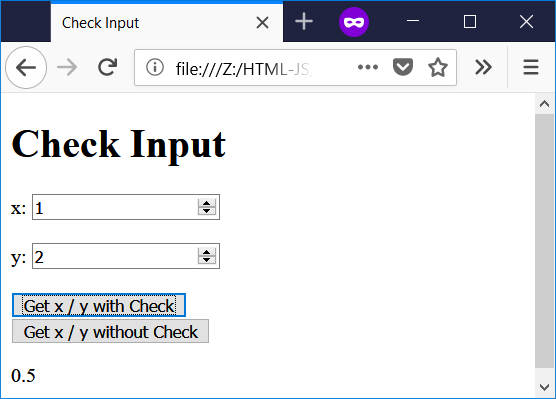
document.getElementById('answer').innerHTML = x / y ;

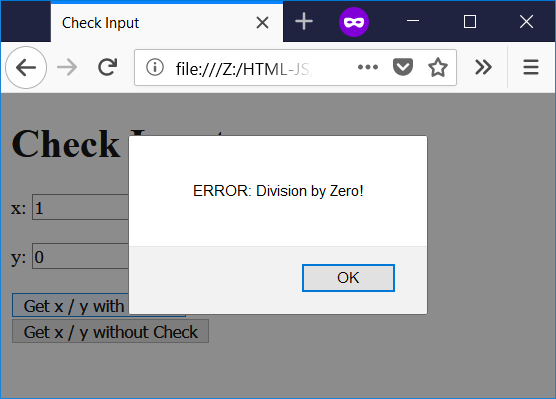
} // end of doDivisionWithoutCheck()

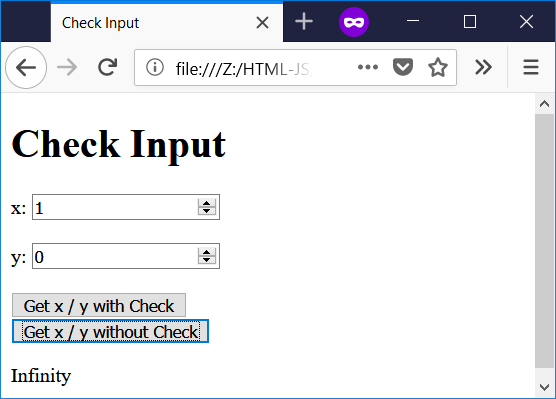
</script>

</body>

</html>







File name: cond-switch-why.html

<!--

Filename: cond-switch-why.html

Author: Joshua Males

Date Created: October 16, 2018

Last Fix: October 22, 2018

Description: Long "if" with options

-->

<!DOCTYPE html>

<html lang="en">

<head>

<title>Long "if" with options</title>

<meta charset="utf-8"/>

</head>

<body>

<h1>Long "if" with options</h1>

Day (1-7): <input type="number" id="xinput" value="1">

<br><br>

<button id="button1" onclick="doThings()">Check</button>

<p id="answer"></p>

<script>

function doThings()

{

// Get the value from the input box and store in day

let day = parseInt(document.getElementById("xinput").value);

let dayName;

if (day == 1)

dayName = "Sunday";

else if (day == 2)

dayName = "Monday";

else if (day == 3)

dayName = "Tuesday";

else if (day == 4)

dayName = "Wednesday";

else if (day == 5)

dayName = "Thursday";

else if (day == 6)

dayName = "Friday";

else if (day == 7)

dayName = "Shabbat";

else

dayName = "No such day";

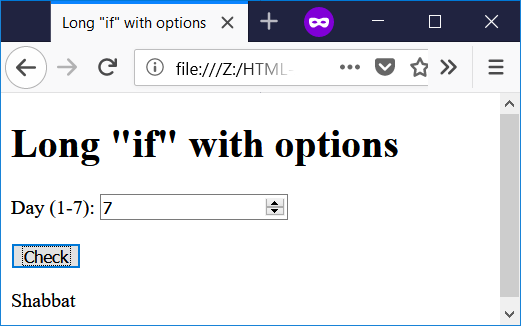
document.getElementById('answer').innerHTML = dayName;

} // end of doThings()

</script>

</body>

</html>



File name: cond-switch1.html

<!--

Filename: cond-switch1.html

Author: Joshua Males

Date Created: October 17, 2018

Last Fix: October 22, 2018

Description: Switch - Basic

-->

<!DOCTYPE html>

<html lang="en">

<head>

<title>Switch - Basic</title>

<meta charset="utf-8"/>

</head>

<body>

<h1>Switch - Basic</h1>

Day (1-7): <input type="number" id="xinput" value="1">

<br><br>

<button id="button1" onclick="doThings()">Show Day Name</button>

<p id="answer"></p>

<script>

function doThings()

{

// Get the value from the input box and store in x

let x = parseInt(document.getElementById("xinput").value);

switch (x) {

case 1:

day = "Sunday";

break;

case 2:

day = "Monday";

break;

case 3:

day = "Tuesday";

break;

case 4:

day = "Wednesday";

break;

case 5:

day = "Thursday";

break;

case 6:

day = "Friday";

break;

case 7:

day = "Shabbat";

break;

default:

day = "Unknown Day";

} // end of switch

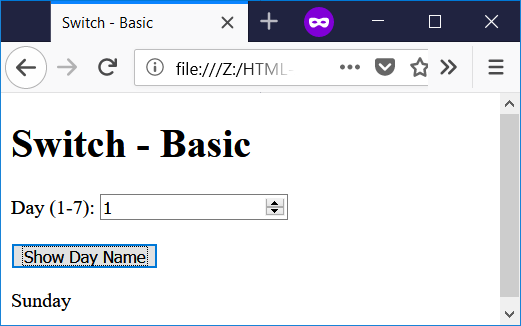
document.getElementById('answer').innerHTML = day ;

} // end of doThings()

</script>

</body>

</html>



File name: cond-switch-groups.html

<!--

Filename: cond-switch-groups.html

Author: Joshua Males

Date Created: October 17, 2018

Last Fix: October 17, 2018

Description: Switch

-->

<!DOCTYPE html>

<html lang="en">

<head>

<title>Switch - Groups</title>

<meta charset="utf-8"/>

</head>

<body>

<h1>Switch - Groups</h1>

Month (1-12): <input type="number" id="xinput" value="1">

<br><br>

<button id="button1" onclick="doThings()">Show Season Name</button>

<p id="answer"></p>

<script>

function doThings()

{

// Get the value from the input box and store in x

let month = parseInt(document.getElementById("xinput").value);

switch (month) {

// January, February, March

case 1:

case 2:

case 3:

document.getElementById('answer').innerHTML = "Winter" ;

break;

// April, May, June

case 4:

case 5:

case 6:

document.getElementById('answer').innerHTML = "Spring";

break;

// July, August, September

case 7:

case 8:

case 9:

document.getElementById('answer').innerHTML = "Summer" ;

break;

// October, November, December

case 10:

case 11:

case 12:

document.getElementById('answer').innerHTML = "Autumn" ;

break;

default:

document.getElementById('answer').innerHTML =

"Unknown Month" ;

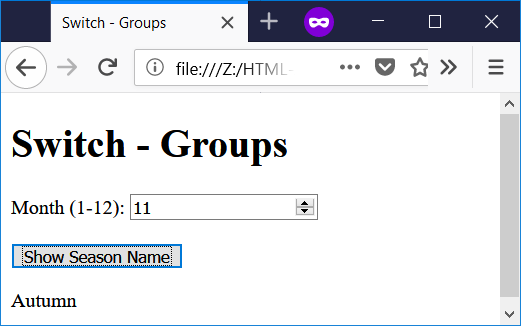
} // end of switch

} // end of doThings()

</script>

</body>

</html>



File name: cond-switch-workdays.html

<!--

Filename: cond-switch-workdays.html

Author: Joshua Males

Date Created: October 17, 2018

Last Fix: October 22, 2018

Description: Switch - Combined Cases

-->

<!DOCTYPE html>

<html lang="en">

<head>

<title>Switch - Case Fall-through</title>

<meta charset="utf-8"/>

</head>

<body>

<h1>Switch - Case Fall-through</h1>

Day (1-7): <input type="number" id="xinput" value="1">

<br><br>

<button id="button1" onclick="doThings()">Show Day Type</button>

<p id="answer"></p>

<script>

function doThings()

{

// Start with a blank

// Get the value from the input box and store in x

let x = parseInt(document.getElementById("xinput").value);

let answer = document.getElementById('answer');

answer.innerHTML = "" ;

switch (x) {

case 3: answer.innerHTML = "Short " ; // Tuesday

case 1:

case 2:

case 4:

case 5:

answer.innerHTML += "Work Day" ;

break;

case 6:

case 7:

answer.innerHTML = "No work today" ;

break;

default:

answer.innerHTML = "Unknown Day" ;

break;

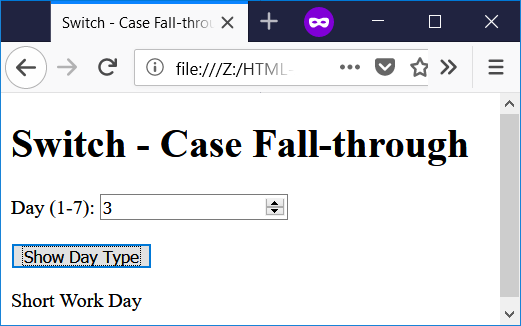
} // end of switch

} // end of doThings()

</script>

</body>

</html>



|  |  |
| --- | --- |
| **Chapter 8 - Popups** | **פרק 8 – חלונות קופצים** |

File name: popups-confirm.html

<!--

Filename: popups-confirm.html

Author: Joshua Males

Date Created: October 18, 2018

Last Fix: October 18, 2018

Description: Use of Confirm Popup

-->

<!DOCTYPE html>

<html lang="en">

<head>

<title>Use of Confirm Popup</title>

<meta charset="utf-8"/>

</head>

<body>

<h1>Use of Confirm Popup</h1>

<button id="button1" onclick="doThings()">Click Me</button>

<p id="answer"></p>

<script>

function doThings()

{

result = window.confirm("Do you want to learn more JavaScript?");

if (result == true)

document.getElementById('answer').innerHTML = "Good choice.";

else

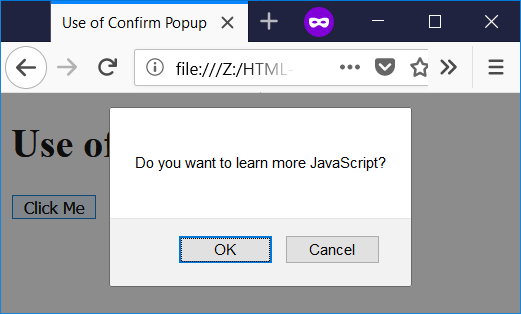
document.getElementById('answer').innerHTML = "Why not?";

} // end of doThings()

</script>

</body>

</html>







File name: popups-prompt.html

<!--

Filename: popups-prompt.html

Author: Joshua Males

Date Created: October 18, 2018

Last Fix: October 18, 2018

Description: Use of Prompt Popup

-->

<!DOCTYPE html>

<html lang="en">

<head>

<title>Use of Prompt Popup</title>

<meta charset="utf-8"/>

</head>

<body>

<h1>Use of Prompt Popup</h1>

<script>

name = window.prompt ("What is your name?", "Superman");

if ((name == null) || (name == "null") || (name == ""))

document.write("User Cancelled");

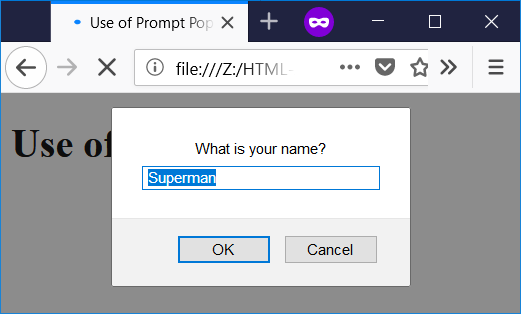
else

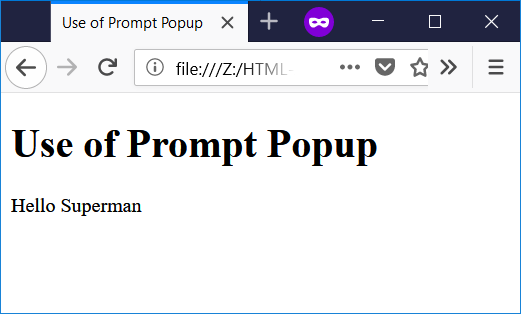
document.write("Hello " + name);

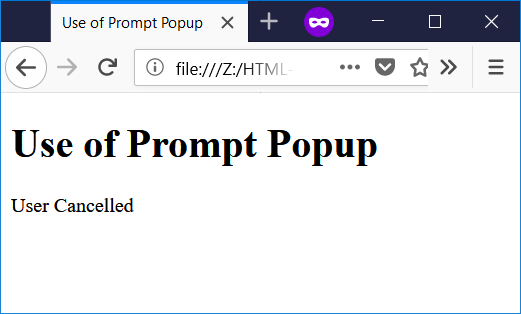
</script>

</body>

</html>







File name: popups-print.html

<!--

Filename: popups-print.html

Author: Joshua Males

Date Created: October 18, 2018

Last Fix: October 18, 2018

Description: Print

-->

<!DOCTYPE html>

<html lang="en">

<head>

<title>Print Page</title>

<meta charset="utf-8"/>

</head>

<body>

<h1>Print Page</h1>

<p id="answer">Some text.</p>

<button id="button1" onclick="doThings()">Print Page</button>

<script>

function doThings()

{

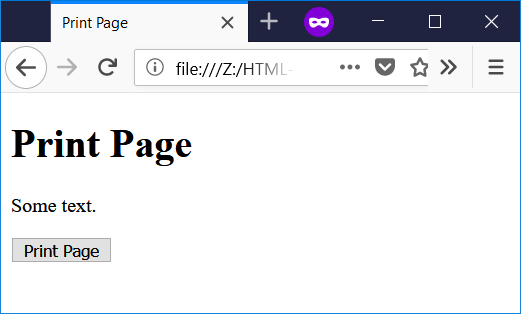
window.print();

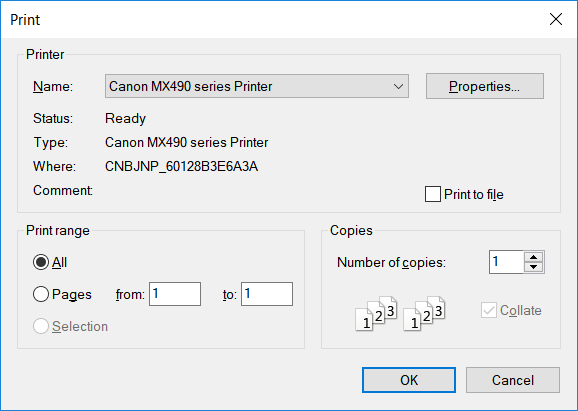
} // end of doThings()

</script>

</body>

</html>





|  |  |
| --- | --- |
| **Chapter 9 - Random Numbers** | **פרק 9 – מספרים רנדומליים** |

File name: random-1-to-max.html

<!--

Filename: random-1-to-max.html

Author: Joshua Males

Date Created: October 18, 2018

Last Fix: October 18, 2018

Description: Pick Random Number from 1 to Maximum

-->

<!DOCTYPE html>

<html lang="en">

<head>

<title>Pick Random Number from 1 to Maximum</title>

<meta charset="utf-8"/>

</head>

<body>

<h1>Pick Random Number from 1 to Maximum</h1>

Maximum: <input type="number" id="maxinput" value="10">

<br><br>

<button id="button1" onclick="doThings()">

Pick Random Number

</button>

<p id="answer"></p>

<script>

function doThings()

{

let max = parseInt(document.getElementById("maxinput").value);

let r = Math.floor(Math.random() \* max) + 1;

document.getElementById('answer').innerHTML =

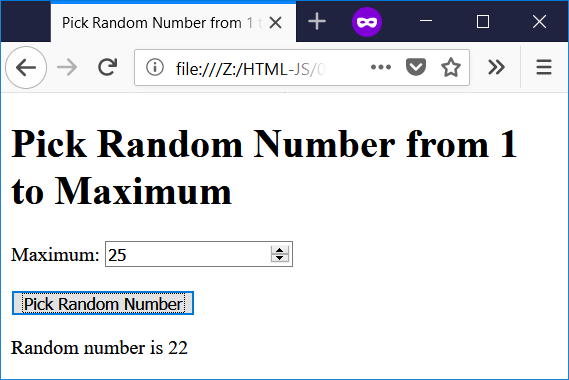
"Random number is " + r;

} // end of doThings()

</script>

</body>

</html>



File name: random-range.html

<!--

Filename: random-range.html

Author: Joshua Males

Date Created: October 18, 2018

Last Fix: October 25, 2018

Description: Pick Random Number from Range

-->

<!DOCTYPE html>

<html lang="en">

<head>

<title>Pick Random Number from Range</title>

<meta charset="utf-8"/>

</head>

<body>

<h1>Pick Random Number from Range</h1>

Minimum: <input type="number" id="mininput" value="0">

<br><br>

Maximum: <input type="number" id="maxinput" value="0">

<br><br>

<button id="button1" onclick="doThings()">

Pick Random Number

</button>

<p id="answer"></p>

<script>

function doThings()

{

let min = parseInt(document.getElementById("mininput").value);

let max = parseInt(document.getElementById("maxinput").value);

let r = Math.floor(Math.random() \* (max - min + 1)) + min;

document.getElementById('answer').innerHTML =

"Random number is " + r;

} // end of doThings()

</script>

</body>

</html>





File name: random-pick-1-5.html

<!--

Filename: random-pick-1-5.html

Author: Joshua Males

Date Created: October 23, 2018

Last Fix: October 23, 2018

Description: Pick a random number and prompt user

-->

<!DOCTYPE html>

<html lang="en">

<head>

<title>Pick a Random Number</title>

<meta charset="utf-8"/>

</head>

<body>

<h1>Pick a Random Number</h1>

<p id="answer"></p>

<script>

let r = Math.floor(Math.random() \* 5) + 1;

myNumber = window.prompt ("Pick a number from 1 to 5:", "");

if ((myNumber == null) || (myNumber == "null") || (myNumber == "")) {

document.write("User Cancelled");

}

else {

if (myNumber == r)

document.getElementById('answer').innerHTML =

"Correct (" + r + "). You can read my mind.";

else

document.getElementById('answer').innerHTML =

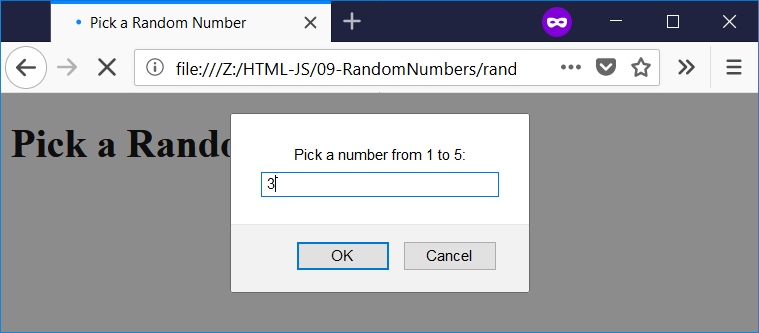
"Sorry. The random number is " + r;

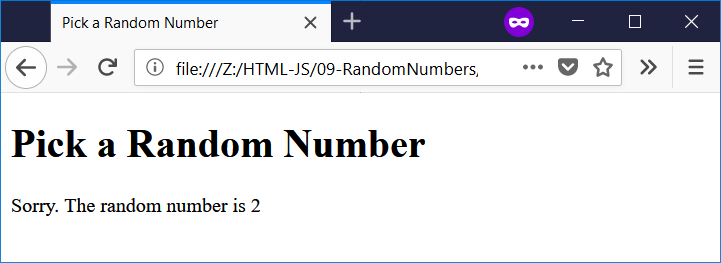
}

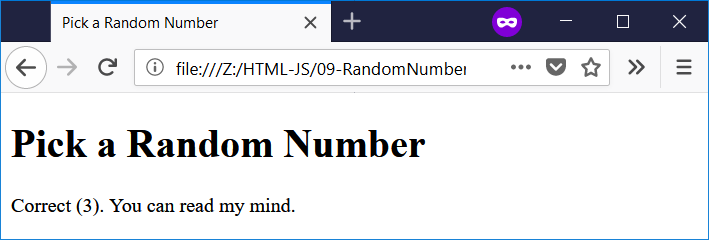
</script>

</body>

</html>







File name: random-bingo-draw-number.html

<!--

Filename: random-bingo-draw-number.html

Author: Joshua Males

Date Created: October 23, 2018

Last Fix: October 25, 2018

Description: Bingo Draw Number

-->

<!DOCTYPE html>

<html lang="en">

<head>

<title>Bingo Draw Number</title>

<meta charset="utf-8"/>

</head>

<body>

<h1>Bingo Draw Number</h1>

<button id="button1" onclick="doThings()">Draw a Number</button>

<p id="answer"></p>

<script>

function doThings()

{

// Letters: 0=B, 1=I, 2=N, 3=G, 4=O

// Pick a number from 1-75

let myNumber = Math.floor(Math.random() \* 75) + 1;

// Get the letter (column from 0-4)

let myLetter = Math.floor((myNumber - 1) / 15) ;

let ball;

// Numbers:

// B: 1-15

// I: 16-30

// N: 31-45

// G: 46-60

// O: 61-75

switch (myLetter) {

case 0: ball = "B"; break;

case 1: ball = "I"; break;

case 2: ball = "N"; break;

case 3: ball = "G"; break;

case 4: ball = "O"; break;

} // switch

ball += " " + myNumber;

document.getElementById('answer').innerHTML =

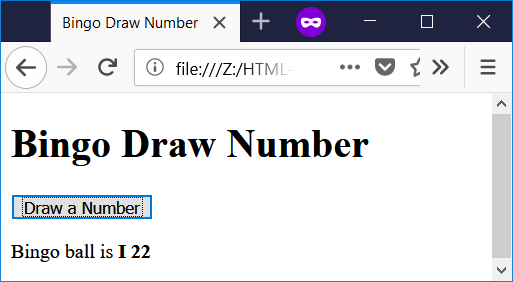
"Bingo ball is <b>" + ball + "</b>";

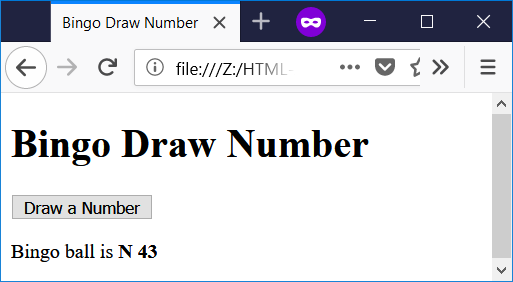
} // end of doThings()

</script>

</body>

</html>





File name: random-greetings.html

<!--

Filename: random-greetings.html

Author: Joshua Males

Date Created: October 25, 2018

Last Fix: October 25, 2018

Description: Random Greeting Generator

-->

<!DOCTYPE html>

<html lang="en">

<head>

<title>מחולל ברכות</title>

<meta charset="utf-8"/>

</head>

<body dir="rtl">

<h1>מחולל ברכות</h1>

<button id="button1" onclick="doThings()">תן ברכה</button>

<p id="answer"></p>

<script>

function doThings()

{

let day = new Date().getDay(); // Get the day of the week: 0-6

let r;

let greeting = "";

if (day == 0) {

greeting += " שבוע"; // We are on Sunday

}

else {

// Not Sunday

r = Math.floor(Math.random() \* 2) ;

switch (r) {

case 0: greeting += " יום";

break;

case 1: greeting += " בוקר";

break;

} // switch

}

r = Math.floor(Math.random() \* 5) ;

switch (r) {

case 0: greeting += " טוב ";

break;

case 1: greeting += " נעים ";

break;

case 2: greeting += " מבורך ";

break;

case 3: greeting += " מוצלח ";

break;

case 4: greeting += " מקסים ";

break;

} // switch

r = Math.floor(Math.random() \* 5) ;

switch (r) {

case 0: greeting += "לכם";

break;

case 1: greeting += "בנים ובנות";

break;

case 2: greeting += "לכולם";

break;

case 3: greeting += "תלמידים";

break;

case 4: greeting += "ילדים";

break;

} // switch

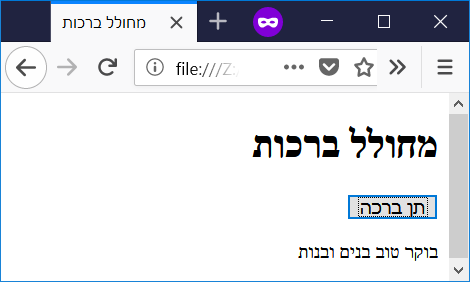
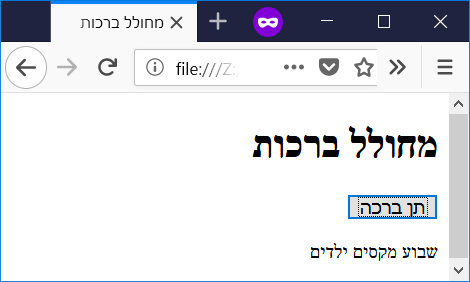
document.getElementById('answer').innerHTML = greeting;

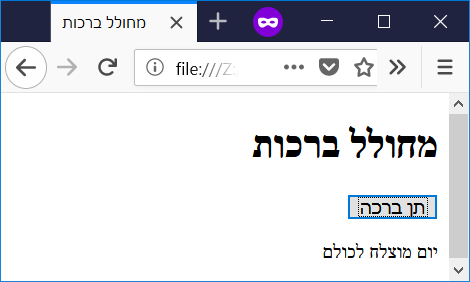
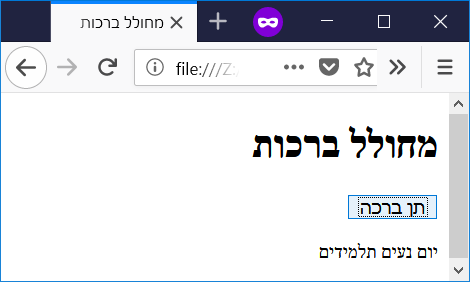
} // end of doThings()

</script>

</body>

</html>

|  |  |
| --- | --- |
| **Chapter 10 - Loops** | **פרק 10 - לולאות** |

File name: while-random.html

<!--

Filename: while-random.html

Author: Joshua Males

Date Created: October 23, 2018

Last Fix: October 23, 2018

Description: Guess Random 1-100

-->

<!DOCTYPE html>

<html lang="en">

<head>

<title>Guess Random 1-100</title>

<meta charset="utf-8"/>

</head>

<body>

<h1>Guess Random 1-100</h1>

<script>

let randomNumber = Math.floor(Math.random() \* 100) + 1;

console.log("Random number is " + randomNumber);

let numTries = 1;

let promptMessage = "Guess a number between 1-100:";

myNumber = window.prompt (promptMessage, "");

while (myNumber != randomNumber) {

console.log("Guess #" + numTries + " is " + myNumber);

promptMessage = "After " + numTries +

". Your guess of " + myNumber;

if (myNumber < randomNumber)

promptMessage += " is too low. Try again: " ;

else

promptMessage += " is too high. Try again: " ;

myNumber = window.prompt (promptMessage, "");

numTries++;

} // while

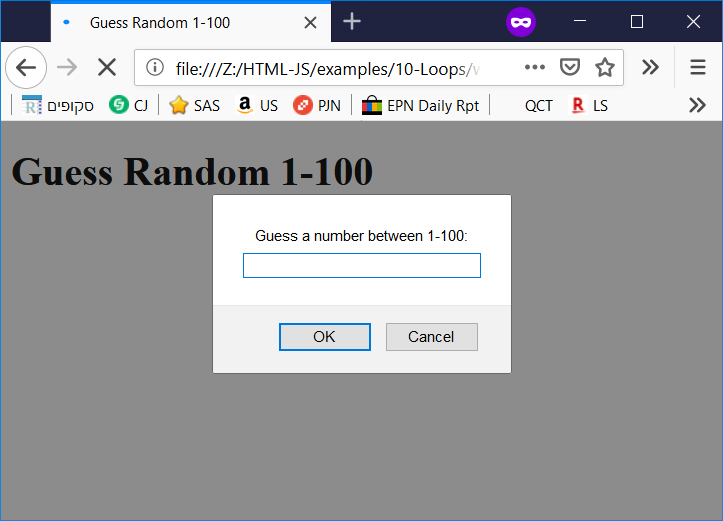
document.write("After " + numTries + " tries. Your guess of " +

myNumber + " is correct.");

</script>

</body>

</html>





File name: do-while-counter.html

<!--

Filename: do-while-counter.html

Author: Joshua Males

Date Created: October 16, 2018

Last Fix: October 16, 2018

Description: Do-While Counter

-->

<!DOCTYPE html>

<html lang="en">

<head>

<title>Do-While Counter</title>

<meta charset="utf-8"/>

</head>

<body>

<h1>Do-While Counter</h1>

Counter: <input type="number" id="xinput" value="5">

<br><br>

<button id="button1" onclick="doThings()">Start Counter</button>

<p id="answer"></p>

<script>

function doThings()

{

// Get the value from the input box and store in x

let x = parseInt(document.getElementById("xinput").value);

let message = "";

if (!x) { // if (x == 0)

document.getElementById('answer').innerHTML = "ERROR: Zero" ;

return; // Exit the function

}

do {

message += "Number = " + x + "<br />";

x--; // Subtract 1

console.log(x); // Write the number to the console

} while (x > 0);

message += "Time is up";

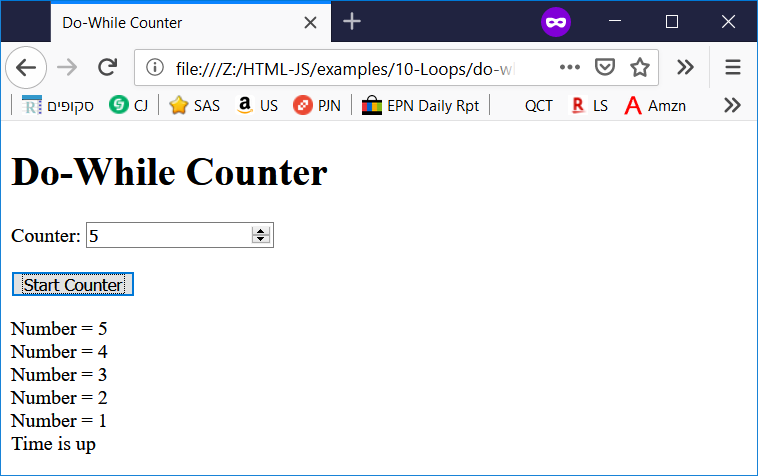
document.getElementById('answer').innerHTML = message ;

} // end of doThings()

</script>

</body>

</html>



File name: for-numbers.html

<!--

Filename: for-numbers.html

Author: Joshua Males

Date Created: October 23, 2018

Last Fix: October 28, 2018

Description: First for loop

-->

<!DOCTYPE html>

<html lang="en">

<head>

<title>First for Loop</title>

<meta charset="utf-8"/>

</head>

<body>

<h1>First for Loop</h1>

<script>

let i;

for (i = 0; i < 5; i++) {

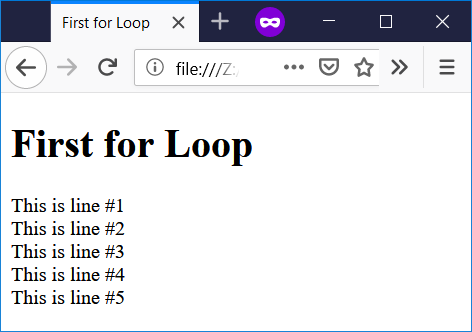
document.write("This is line #" + (i + 1) + "<br />");

}

</script>

</body>

</html>



File name: for-input-boxes.html

<!--

Filename: for-input-boxes.html

Author: Joshua Males

Date Created: October 23, 2018

Last Fix: October 23, 2018

Description: For loop for creating input boxes

-->

<!DOCTYPE html>

<html lang="en">

<head>

<title>for Loop - Inputs</title>

<meta charset="utf-8"/>

</head>

<body>

<h1>for Loop - Inputs</h1>

<script>

let i;

let idInput, thisHTML;

// Create 10 input boxes with unique ids and values

// Examples: <input id="input1" value="10" type="number">

// <input id="input2" value="9" type="number">

// <input id="input3" value="8" type="number">

for (i = 0; i < 10; i++) {

// Build an HTML input box

idInput = "input" + (i+1);

thisHTML = "Input " + (i+1) + ": " +

'<input type="number" ' +

'id="' + idInput + '" ' +

'value="' + (10-i) + '"><br/>';

document.write(thisHTML);

}

</script>

<br><br>

Box to Increment: <input type="number" id="xinput" value="5">

<br><br>

<button id="button1" onclick="doThings()">

Increment Input Box

</button>

<script>

function doThings()

{

// Get the value from the input box and store in x

let x = parseInt(document.getElementById("xinput").value);

let inputFix = "input" + x;

let fixMe = parseInt(document.getElementById(inputFix).value);

fixMe++;

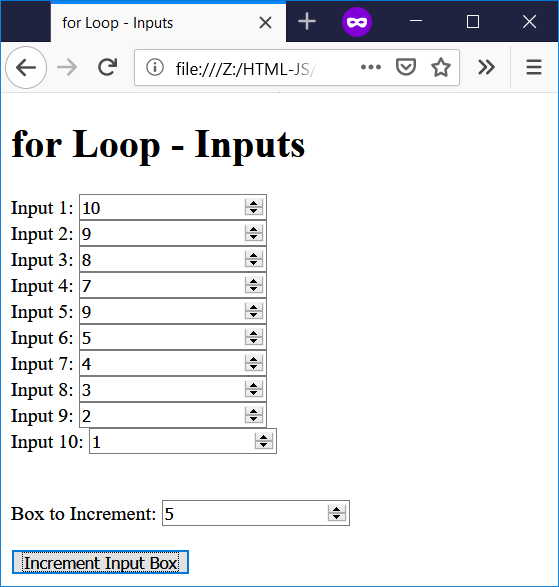
document.getElementById(inputFix).value = fixMe;

} // end of doThings()

</script>

</body>

</html>



File name: do-while-break.html

<!--

Filename: do-while-break.html

Author: Joshua Males

Date Created: October 23, 2018

Last Fix: October 23, 2018

Description: Break out of loop

-->

<!DOCTYPE html>

<html lang="en">

<head>

<title>Break out of loop</title>

<meta charset="utf-8"/>

</head>

<body>

<h1>Break out of loop</h1>

Counter: <input type="number" id="xinput" value="5">

<br><br>

<button id="button1" onclick="doThings()">Start Counter</button>

<p id="answer"></p>

<script>

function doThings()

{

// Get the value from the input box and store in x

let x = parseInt(document.getElementById("xinput").value);

let firstX = x;

let message = "";

if (!x) { // if (x == 0)

document.getElementById('answer').innerHTML = "ERROR: Zero" ;

return;

}

do {

message += "Number = " + x + "<br />";

// If x == 3 and the original number was 5, then break out of loop

if ((x == 3) && (firstX == 5))

break;

x--; // Subtract 1

} while (x > 0);

message += "Time is up";

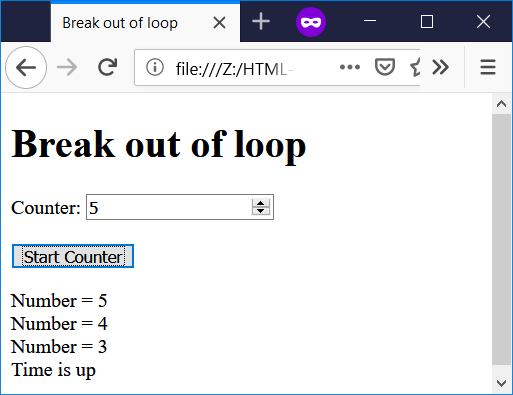
document.getElementById('answer').innerHTML = message ;

} // end of doThings()

</script>

</body>

</html>



File name: nested1.html

<!--

Filename: nested1.html

Author: Joshua Males

Date Created: October 23, 2018

Last Fix: October 23, 2018

Description: Nested Loop #1

-->

<!DOCTYPE html>

<html lang="en">

<head>

<title>Nested Loop #1</title>

<meta charset="utf-8"/>

</head>

<body>

<h1>Nested Loop #1</h1>

<script>

let row = 1, column;

while (row <= 5) {

column = 1;

while (column <= row) {

document.write(column + " ");

column++;

} // inner while

document.write("<br />"); // Jump down to the next row

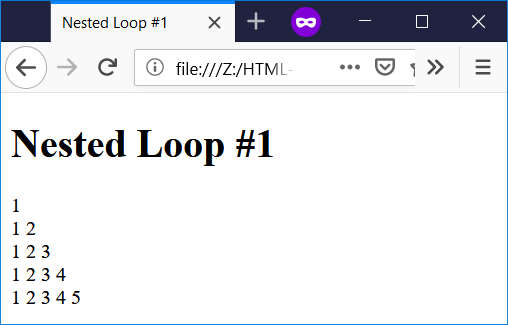
row++;

} // outer while

</script>

</body>

</html>



File name: nested-primes.html

<!--

Filename: nested-primes.html

Author: Joshua Males

Date Created: October 25, 2018

Last Fix: October 25, 2018

Description: Prime Numbers < 50

-->

<!DOCTYPE html>

<html lang="en">

<head>

<title>Prime Numbers < 50</title>

<meta charset="utf-8"/>

</head>

<body>

<h1>Prime Numbers < 50</h1>

<script>

let testMe, j, isPrime;

for (testMe = 2; testMe < 50; testMe++) { // outer loop

// See if "testMe" is prime

isPrime = 1; // Assume it is prime

for (j = 2; j < testMe; j++) { // inner loop

if ((testMe % j) == 0) {

isPrime = 0; // if factor found, testMe is not prime

break; // No need to check more numbers

}

} // end of inner loop (j)

if (isPrime)

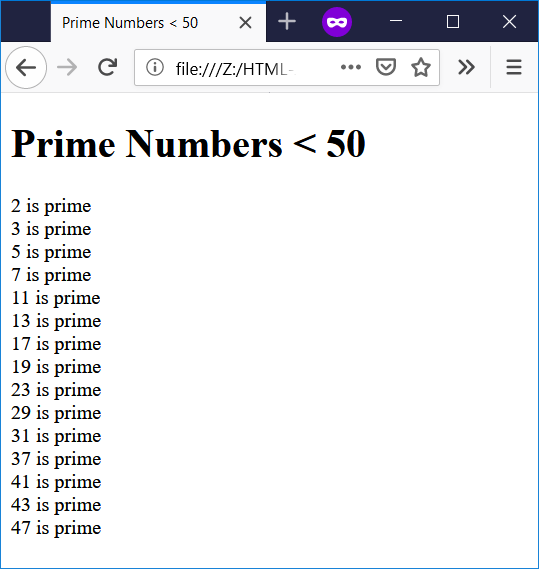
document.write(testMe + " is prime" + "<br />");

} // end of outer loop

</script>

</body>

</html>



File name: for-continue.html

<!--

Filename: for-continue.html

Author: Joshua Males

Date Created: October 23, 2018

Last Fix: October 23, 2018

Description: For with continue

-->

<!DOCTYPE html>

<html lang="en">

<head>

<title>for loop with continue</title>

<meta charset="utf-8"/>

</head>

<body>

<h1>for loop with continue</h1>

<script>

let i;

let idInput, thisHTML;

// Create 10 input boxes with unique ids and values

// Examples: <input id="input1" value="10" type="number">

// <input id="input2" value="9" type="number">

// <input id="input3" value="8" type="number">

for (i = 0; i < 10; i++) {

// Build an HTML input box

if ((i>2) && (i<7))

continue;

idInput = "input" + (i+1);

thisHTML = "Input " + (i+1) + ": " +

"<input type=\"number\" " +

"id=\"" + idInput + "\" " +

"value=\"" + (10-i) + "\"><br/>";

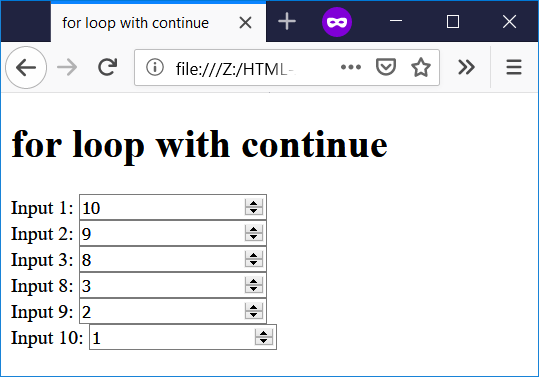
document.write(thisHTML);

}

</script>

</body>

</html>



|  |  |
| --- | --- |
| **Chapter 11 - Tables** | **פרק 11 - טבלאות** |

File name: table-2x2.html

<!--

Filename: table-2x2.html

Author: Joshua Males

Date Created: October 31, 2018

Last Fix: October 31, 2018

Description: Simple 2x2 Table

-->

<!DOCTYPE html>

<html lang="en">

<head>

<title>Simple 2x2 Table</title>

<meta charset="utf-8"/>

</head>

<body>

<h1>Simple 2x2 Table</h1>

<table>

<tr>

<td>111</td>

<td>222</td>

</tr>

<tr>

<td>333</td>

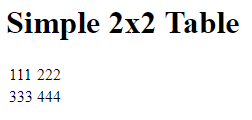
<td>444</td>

</tr>

</table>

</body>

</html>



File name: table-2x2-header.html

<!--

Filename: table-2x2-header.html

Author: Joshua Males

Date Created: October 31, 2018

Last Fix: October 31, 2018

Description: Simple 2x2 Table with Header

-->

<!DOCTYPE html>

<html lang="en">

<head>

<title>Simple 2x2 Table with Header</title>

<meta charset="utf-8"/>

</head>

<body>

<h1>Simple 2x2 Table with Header</h1>

<table>

<tr>

<th>Height</th>

<th>Weight</th>

</tr>

<tr>

<td>111</td>

<td>222</td>

</tr>

<tr>

<td>333</td>

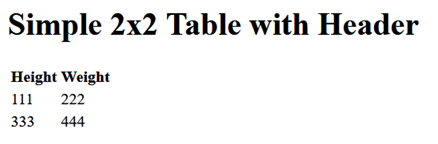
<td>444</td>

</tr>

</table>

</body>

</html>



File name: table-border.html

<!--

Filename: table-border.html

Author: Joshua Males

Date Created: October 31, 2018

Last Fix: October 31, 2018

Description: Table with Border

-->

<!DOCTYPE html>

<html lang="en">

<head>

<title>Table with Border</title>

<meta charset="utf-8"/>

</head>

<body>

<h1>Table with Border</h1>

<table border="1"

bordercolor="black"

cellspacing="0"

cellpadding="10"

width="50%"><!-- Use 50% of the screen -->

<tr align="center">

<td width="20%" bgcolor="black">

<font color="white">black</font>

</td>

<td width="15%" bgcolor="silver">silver</td>

<td width="25%" bgcolor="gray">gray</td>

<td width="40%" bgcolor="white">white</td>

</tr>

<tr align="right">

<td bgcolor="Maroon"><font color="white">maroon</font></td>

<td align="left" bgcolor="Red">

<font color="white">red</font>

</td>

<td bgcolor="Purple"><font color="white">purple</font></td>

<td bgcolor="Fuchsia">fuchsia</td>

</tr>

<tr align="left">

<td bgcolor="Green">green</td>

<td align="right" bgcolor="Lime">lime</td>

<td bgcolor="Olive">olive</td>

<td bgcolor="Yellow">yellow</td>

</tr>

<tr>

<td bgcolor="Navy"><font color="white">navy</font></td>

<td bgcolor="Blue"><font color="white">blue</font></td>

<td bgcolor="Teal">teal</td>

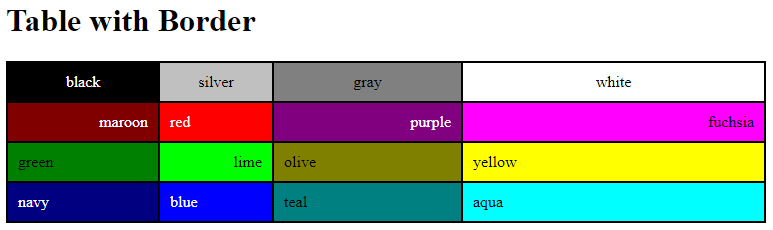
<td bgcolor="Aqua">aqua</td>

</tr>

</table>

</body>

</html>



File name: table-colspan.html

<!--

Filename: table-colspan.html

Author: Joshua Males

Date Created: October 31, 2018

Last Fix: October 31, 2018

Description: Merge with Colspan

-->

<!DOCTYPE html>

<html lang="en">

<head>

<title>Merge with Colspan</title>

<meta charset="utf-8"/>

</head>

<body>

<h1>Merge with Colspan</h1>

<table border="1"

bordercolor="black"

cellspacing="0"

cellpadding="10">

<tr>

<td>1</td>

<td>2</td>

<td>3</td>

<td>4</td>

</tr>

<tr>

<td colspan="2">5</td>

<td>6</td>

<td>7</td>

</tr>

<tr>

<td>8</td>

<td>9</td>

<td colspan="2">10</td>

</tr>

<tr>

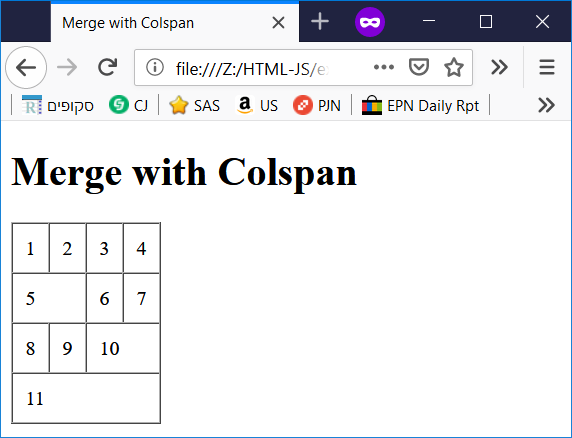
<td colspan="4">11</td>

</tr>

</table>

</body>

</html>



File name: table-mini-multiply.html

<!--

Filename: table-mini-multiply.html

Author: Joshua Males

Date Created: October 31, 2018

Last Fix: October 31, 2018

Description: Mini Multiplication Table

-->

<!DOCTYPE html>

<html lang="en">

<head>

<title>Mini Multiplication Table</title>

<meta charset="utf-8"/>

</head>

<body>

<h1>Mini Multiplication Table</h1>

<table border="1"

bordercolor="black"

cellspacing="0"

cellpadding="10">

<tr bgcolor="silver"><!-- Use bgcolor for the whole row -->

<td></td><td>1</td><td>2</td><td>3</td><td>4</td>

</tr>

<tr>

<td bgcolor="silver">1</td>

<td>1</td>

<td>2</td>

<td>3</td>

<td>4</td>

</tr>

<tr>

<td bgcolor="silver">2</td>

<td>2</td>

<td>4</td>

<td>6</td>

<td>8</td>

</tr>

<tr>

<td bgcolor="silver">3</td>

<td>3</td>

<td>6</td>

<td>9</td>

<td>12</td>

</tr>

<tr>

<td bgcolor="silver">4</td>

<td>4</td>

<td>8</td>

<td>12</td>

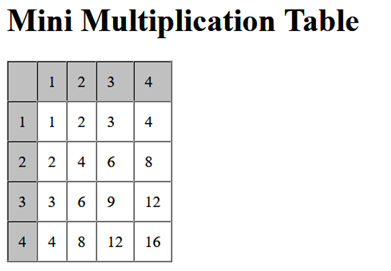
<td>16</td>

</tr>

</table>

</body>

</html>



|  |  |
| --- | --- |
| **Chapter 12 - Functions** | **פרק 12 - פונקציות** |

File name: func1.html

<!--

Filename: func1.html

Date Created: November 3, 2018

Last Fix: November 3, 2018

Description: Simple Function

-->

<!DOCTYPE html>

<html lang="en">

<head>

<title>

פונקציה פשוטה

</title>

<meta charset="utf-8"/>

</head>

<body dir="rtl">

<h1>

פונקציה פשוטה

</h1>

<p>

דוגמא זאת מפעילה פונקציה לחשב את המכפלה של שני מספרים.

<br />

הפונקציה מקבלת שני פרמטרים, ומחזירה ערך.

</p>

<p id="result"></p>

<script>

function myFunction(p1, p2) {

return p1 \* p2;

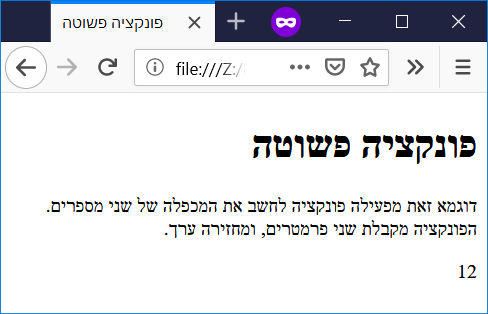
} // myFunction

document.getElementById("result").innerHTML = myFunction(4, 3);

</script>

</body>

</html>



File name: func-no-return.html

<!--

Filename: func-no-return.html

Date Created: November 3, 2018

Last Fix: November 3, 2018

Description: Simple Function without return

-->

<!DOCTYPE html>

<html lang="en">

<head>

<title>

פונקציה פשוטה ללא return

</title>

<meta charset="utf-8" />

</head>

<body dir="rtl">

<h1>

פונקציה פשוטה ללא return

</h1>

<p>

דוגמא זאת מפעילה פונקציה לחשב את המכפלה של שני מספרים.

<br />

הפונקציה מקבלת שני פרמטרים, ולא מחזירה ערך.

</p>

<p id="result"></p>

<script>

function myFunction(p1, p2) {

document.getElementById("result").innerHTML = (p1 \* p2);

} // myFunction

myFunction(5, 6);

</script>

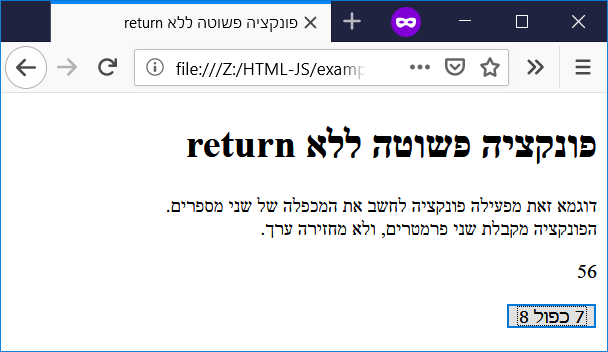
<button id="button1" onclick="myFunction(7, 8)">

7 כפול 8

</button>

</body>

</html>



File name: func-celsius.html

<!--

Filename: func-celsius.html

Author: Joshua Males

Date Created: November 3, 2018

Last Fix: November 3, 2018

Description: Use function to convert Celsius to Fahrenheit

-->

<!DOCTYPE html>

<html lang="en">

<head>

<title>Function: Celsius to Fahrenheit</title>

<meta charset="utf-8"/>

</head>

<body>

<h1>Convert Celsius to Fahrenheit</h1>

<script>

// This must be defined before we call the function below.

function toFahrenheit(celsius) {

return (celsius \* 9/5 + 32);

} // toFahrenheit

</script>

38&deg;C = <script>

document.write( toFahrenheit(38) );

</script>&deg;F<br />

100&deg;C = <script>

document.write( toFahrenheit(100) );

</script>&deg;F<br />

0&deg;C = <script>

document.write( toFahrenheit(0) );

</script>&deg;F<br />

</body>

</html>



File name: func-vars.html

<!--

Filename: func-vars.html

Author: Joshua Males

Date Created: November 4, 2018

Last Fix: November 4, 2018

Description: Show Use of Local Variables

-->

<!DOCTYPE html>

<html lang="en">

<head>

<title>Show Use of Local Variables</title>

<meta charset="utf-8"/>

</head>

<body>

<h1>Show Use of Local Variables</h1>

<script>

function toFahrenheit(celsius) {

let f;

f = celsius \* 9/5 + 32;

return (f);

} // toFahrenheit

function myFunction() {

f = "Some text";

return (f);

} // myFunction

</script>

38&deg;C = <script>

document.write( toFahrenheit(38) );

</script>&deg;F<br />

<script>

document.write("f = " + f + "<br />"); // Look at Console

let f = 1; // global

document.write("f = " + f + "<br />");

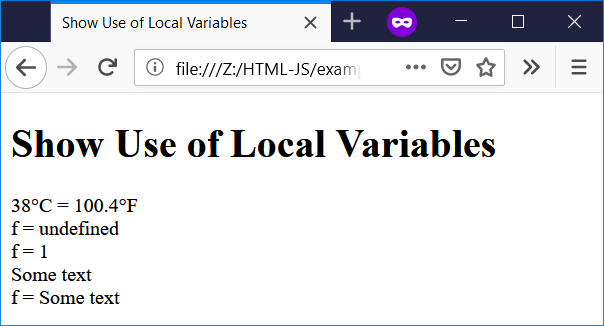
document.write( myFunction() + "<br />" );

document.write("f = " + f + "<br />");

</script>

</body>

</html>



File name: func-area-circle.html

<!--

Filename: func-area-circle.html

Author: Joshua Males

Date Created: November 3, 2018

Last Fix: November 3, 2018

Description: Function for Area of Circle

-->

<!DOCTYPE html>

<html lang="en">

<head>

<title>Area of Circle</title>

<meta charset="utf-8"/>

</head>

<body>

<h1>Area of Circle</h1>

Enter Radius: <input type="number" id="rinput" value="0">

<br><br>

<button id="button1" onclick="doThings();">Calculate Area</button>

<p id="answer"></p>

<script>

function areaOfCircle(radius)

{

let area = Math.PI \* radius \* radius;

return area.toFixed(2);

} // areaOfCircle

function doThings()

{

// Get the value from the input box and store in radius

let radius = parseFloat(document.getElementById("rinput").value);

let area = areaOfCircle(radius);

document.getElementById('answer').innerHTML = "Area = " + area ;

} // end of doThings()

</script>

</body>

</html>



File name: func-digits.html

<!--

Filename: func-digits.html

Author: Joshua Males

Date Created: November 4, 2018

Last Fix: November 4, 2018

Description: Use of Digit Functions

-->

<!DOCTYPE html>

<html lang="en">

<head>

<title>Use of Digit Functions</title>

<meta charset="utf-8"/>

</head>

<body>

<h1>Use of Digit Functions</h1>

Number: <input type="number" id="xinput" value="123">

<br><br>

<button id="button1" onclick="doThings()">

Get Sum and Count

</button>

<p id="answer"></p>

<script src="digits.js"></script>

<script>

function doThings()

{

// Get the value from the input box and store in x

let x = parseInt(document.getElementById("xinput").value);

document.getElementById('answer').innerHTML =

"Sum: " + sumOfDigits(x) + "<br />" +

"Number of Digits: " + numberOfDigits(x);

} // doThings()

</script>

</body>

</html>

File name: digits.js

//==========================================================

// Filename: digits.js

// Author: Joshua Males

// Date Created: November 4, 2018

// Last Fix: November 4, 2018

// Description: Digit Functions

//==========================================================

function getOnesDigit(inNumber)

{

return inNumber % 10;

} // getOnesDigit

//==========================================================

function sumOfDigits(inNumber)

{

let sum = 0;

while (inNumber) {

sum += getOnesDigit(inNumber); // Add the ones digit

inNumber = Math.floor(inNumber / 10); // Cut off the ones digit

} // while

return sum;

} // sumOfDigits

//==========================================================

function numberOfDigits(inNumber)

{

let digits = 0;

while (inNumber) {

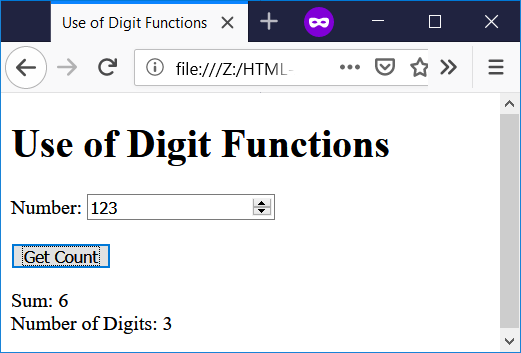
digits++;

inNumber = Math.floor(inNumber / 10); // Cut off the ones digit

} // while

return digits;

} // numberOfDigits



File name: func-default.html

<!--

Filename: func-default.html

Author: Joshua Males

Date Created: November 3, 2018

Last Fix: November 3, 2018

Description: Function with Default Parameters

-->

<!DOCTYPE html>

<html lang="en">

<head>

<title>Sum with Default Parameters</title>

<meta charset="utf-8"/>

</head>

<body>

<h1>Sum with Default Parameters</h1>

x: <input type="number" id="xinput" value="0">

<br><br>

y: <input type="number" id="yinput" value="0">

<br><br>

<button id="button1" onclick="doBoth();">Sum with Both</button>

<button id="button1" onclick="doDefault();">

Sum with Default y (11)

</button>

<p id="answer"></p>

<script>

function sumFunction(x, y = 11)

{

return x + y;

} // sumFunction

function doBoth()

{

let x = parseInt(document.getElementById("xinput").value);

let y = parseInt(document.getElementById("yinput").value);

document.getElementById('answer').innerHTML =

"Sum = " + sumFunction(x, y) ;

} // end of doBoth()

function doDefault()

{

let x = parseInt(document.getElementById("xinput").value);

document.getElementById('answer').innerHTML =

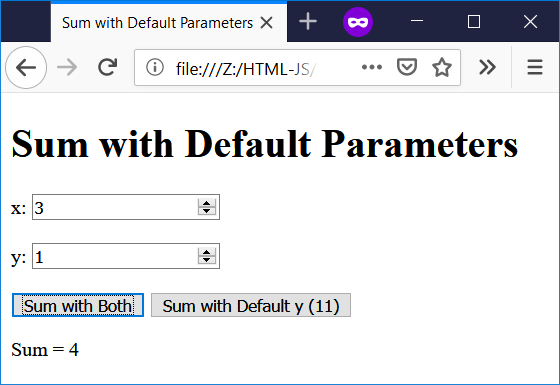
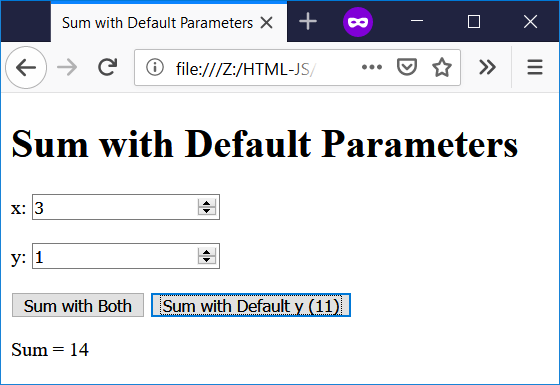
"Sum = " + sumFunction(x) ;

} // end of doDefault()

</script>

</body>

</html>

File name: func-default-old.html

<!--

Filename: func-default-old.html

Author: Joshua Males

Date Created: November 3, 2018

Last Fix: November 3, 2018

Description: Function with Old Default Parameters

-->

<!DOCTYPE html>

<html lang="en">

<head>

<title>Sum with Old Default Parameters</title>

<meta charset="utf-8"/>

</head>

<body>

<h1>Sum with Old Default Parameters</h1>

x: <input type="number" id="xinput" value="0">

<br><br>

y: <input type="number" id="yinput" value="0">

<br><br>

<button id="button1" onclick="doBoth();">Sum with Both</button>

<button id="button1" onclick="doDefault();">

Sum with Default y (11)

</button>

<p id="answer"></p>

<script>

function sumFunction(x, y)

{

if (typeof y === 'undefined')

y = 11;

return x + y;

} // sumFunction

function doBoth()

{

let x = parseInt(document.getElementById("xinput").value);

let y = parseInt(document.getElementById("yinput").value);

document.getElementById('answer').innerHTML =

"Sum = " + sumFunction(x ,y) ;

} // end of doBoth()

function doDefault()

{

let x = parseInt(document.getElementById("xinput").value);

document.getElementById('answer').innerHTML =

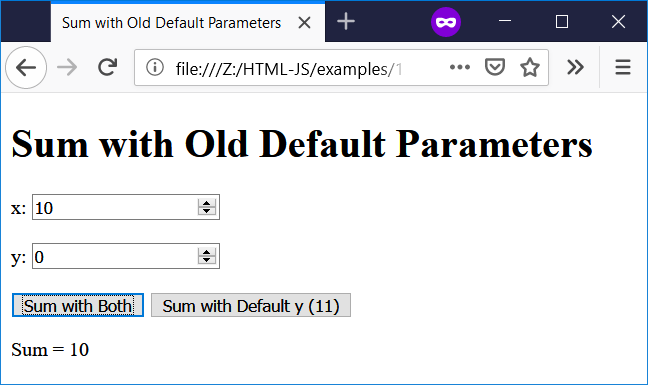
"Sum = " + sumFunction(x) ;

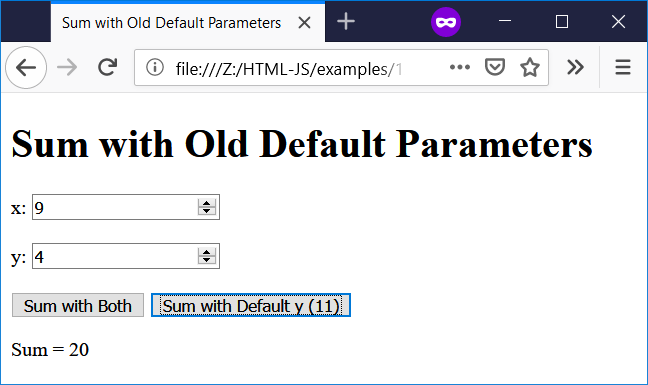
} // end of doDefault()

</script>

</body>

</html>





File name: func-tz.html

<!--

Filename: func-tz.html

Author: Joshua Males

Date Created: November 4, 2018

Last Fix: November 4, 2018

Description: Calculate Check Digit for Teudat Zehut

-->

<!DOCTYPE html>

<html lang="en">

<head>

<title>

ספרת ביקורת לתעודת זהות

</title>

<meta charset="utf-8"/>

</head>

<body dir="rtl">

<h1>

ספרת ביקורת לתעודת זהות

</h1>

מספר תעודת זהות (בלי הספרה האחרונה)

<input type="number" id="tzinput" value="">

<br><br>

<button id="button1" onclick="doThings()">

חשב ספרת ביקורת

</button>

<p id="answer"></p>

<script src="digits.js"></script>

<script>

//==========================================================

// We're going backwards:

// If position is even (0,2,4,6) return 2; if odd return 1

function getWeight(position)

{

if ((position % 2) == 0) // 0,2,4,6

return 2;

else // 1,3,5,7

return 1;

} // getWeight

//==========================================================

function calcCheckDigit(tz)

{

let i, product, sum = 0, weight, digit;

// We are moving from right to left

for (i = 0; i < 8; i++) {

digit = getOnesDigit(tz);

weight = getWeight(i); // 1 or 2

product = digit \* weight;

sum += sumOfDigits(product);

// Remove the ones digit

tz = Math.floor(tz / 10);

} // for i

onesDigit = getOnesDigit(sum);

return 10 - onesDigit;

} // calcCheckDigit

//==========================================================

function doThings()

{

// Get the value from the input box and store in tz

let tz = parseInt(document.getElementById("tzinput").value);

let checkDigit;

checkDigit = calcCheckDigit(tz);

document.getElementById('answer').innerHTML =

"ספרת ביקורת: " +

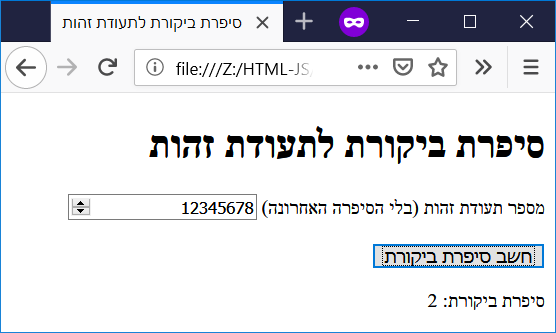
checkDigit;

} // doThings()

</script>

</body>

</html>



|  |  |
| --- | --- |
| **Chapter 13 - Arrays** | **פרק 13 - מערכים** |

File name: array-mothers.html

<!--

Filename: array-mothers.html

Author: Joshua Males

Date Created: November 24, 2018

Last Fix: November 24, 2018

Description: Display Matriarchs

-->

<!DOCTYPE html>

<html lang="en">

<head>

<title>

Display Matriarchs

</title>

<meta charset="utf-8"/>

</head>

<body>

<h1>

Display Matriarchs

</h1>

<script>

let mothers, text, i;

mothers = ["Sarah", "Rivka",

"Rachel", "Leah"];

text = "<ul>";

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

text += "<li>" + mothers[i] + "</li>";

}

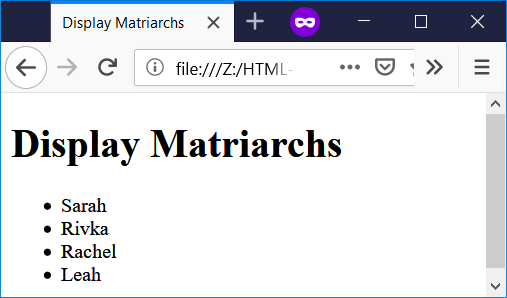
text += "</ul>";

document.write(text);

</script>

</body>

</html>



File name: array1.html

<!--

Filename: array1.html

Author: Joshua Males

Date Created: November 5, 2018

Last Fix: November 5, 2018

Description: Pick 10 random numbers. Find average, biggest and smallest.

-->

<!DOCTYPE html>

<html lang="en">

<head>

<title>

Array Scan

</title>

<meta charset="utf-8"/>

</head>

<body>

<h1>

Array Scan

</h1>

<script>

let aNumbers = [];

let i;

for (i=0; i < 10; i++)

aNumbers[i] = Math.floor(Math.random() \* 100) + 1;

document.write("Numbers: ");

for (i=0; i < 10; i++)

document.write(aNumbers[i] + " ");

document.write("<br />");

// Find the average, biggest and smallest

let total = 0;

let biggest = aNumbers[0], smallest = aNumbers[0];

for (i=1; i<10; i++) {

total += aNumbers[i];

if (aNumbers[i] > biggest)

biggest = aNumbers[i];

if (aNumbers[i] < smallest)

smallest = aNumbers[i];

} // for i

document.write("Average: " + (total / 10).toFixed(2) + "<br />");

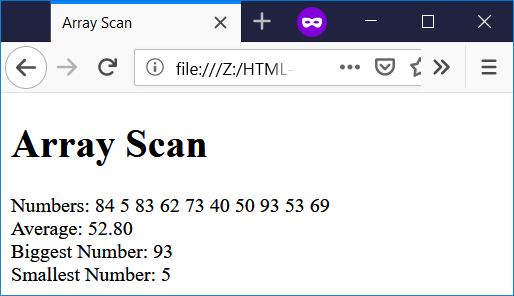
document.write("Biggest Number: " + biggest + "<br />");

document.write("Smallest Number: " + smallest + "<br />");

</script>

</body>

</html>



File name: array-big.html

<!--

Filename: array-big.html

Author: Joshua Males

Date Created: November 22, 2018

Last Fix: November 22, 2018

Description: Pick 10 random numbers. Find the biggest number.

-->

<!DOCTYPE html>

<html lang="en">

<head>

<title>

Downwards Array Scan

</title>

<meta charset="utf-8"/>

</head>

<body>

<h1>

Downwards Array Scan

</h1>

<script>

function findMax(arr)

{

let len = arr.length

let max = arr[--len];

while (len--) {

if (arr[len] > max) {

max = arr[len];

}

}

return max;

} // findMax

let aNumbers = [];

let biggest, i;

for (i=0; i < 10; i++)

aNumbers[i] = Math.floor(Math.random() \* 100) + 1;

document.write("Numbers: ");

for (i=0; i < aNumbers.length; i++)

document.write(aNumbers[i] + " ");

document.write("<br />");

biggest = findMax(aNumbers);

document.write("Biggest Number: " + biggest + "<br />");

</script>

</body>

</html>



File name: array-methods.html

<!--

Filename: array-methods.html

Author: Joshua Males

Date Created: November 22, 2018

Last Fix: June 18, 2019

Description: Array Methods

-->

<!DOCTYPE html>

<html lang="en">

<head>

<title>

Array Methods

</title>

<meta charset="utf-8"/>

</head>

<body>

<h1>

Array Methods

</h1>

<script>

let aNumbers = [];

let biggest, i;

// Create array with random numbers (1-100)

for (i=0; i < 10; i++)

aNumbers[i] = Math.floor(Math.random() \* 100) + 1;

document.write("Numbers: " + aNumbers.toString() + "<br />");

document.write("join: " + aNumbers.join(" and ") + "<br />");

aNumbers.reverse();

document.write("reverse: Reversed Numbers: " +

aNumbers.toString() + "<br />");

let x = aNumbers.pop();

document.write("pop: Removed " + x + ", Numbers: " +

aNumbers.toString() + "<br />");

x = aNumbers.push(222);

document.write("push: " + x +

" Numbers: " + aNumbers.toString() + "<br />");

x = aNumbers.shift();

document.write("shift: Removed " + x + ", Numbers: " +

aNumbers.toString() + "<br />");

x = aNumbers.unshift(333);

document.write("unshift: " + x +

" Numbers: " + aNumbers.toString() + "<br />");

let y = aNumbers.indexOf(222);

document.write("indexOf: 222 found at index: " + y + "<br />");

x = aNumbers.push(333);

document.write("push: " + x + " Numbers:

" + aNumbers.toString() + "<br />");

y = aNumbers.lastIndexOf(333);

document.write("lastIndexOf: 333 last found at index: " + y +

"<br />");

aNumbers = [];

document.write("Cleared Numbers: " + aNumbers.toString() +

"<br />");

let names = "Reuven + Shimon + Levi";

let aNames = names.split(" + ");

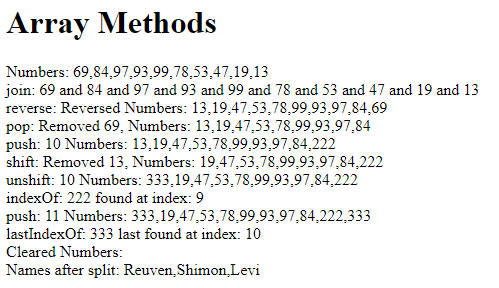
document.write("Names after split: " + aNames.toString() +

"<br />");

</script>

</body>

</html>



File name: array-insert.html

<!--

Filename: array-insert.html

Author: Joshua Males

Date Created: November 5, 2018

Last Fix: November 5, 2018

Description: Insert number into sorted array

-->

<!DOCTYPE html>

<html lang="en">

<head>

<title>

Array Insert

</title>

<meta charset="utf-8"/>

</head>

<body>

<h1>

Array Insert

</h1>

<script>

let aNumbers = [3,6,22,35,47,56,68,74,87,99];

let i;

document.write("Numbers: ");

for (i=0; i < aNumbers.length; i++)

document.write(aNumbers[i] + " ");

document.write("<br />");

</script>

Enter number to insert:

<input type="number" id="numinput" value="">

<br><br>

<button id="button1" onclick="doThings()">

Insert Number

</button>

<p id="answer"></p>

<script>

function doThings()

{

// Get the value from the input box and store in insertMe

let insertMe = parseInt(document.getElementById("numinput").value);

let i = aNumbers.length-1; // Start at the end

while ((i >= 0) && (aNumbers[i] > insertMe) ) {

aNumbers[i + 1] = aNumbers[i]; // Move item up

i--;

}

// If we left the loop, then i+1 = the place to insert the new number

aNumbers[i + 1] = insertMe; // Insert into array

// Build output string

output = "";

for (i=0; i < aNumbers.length; i++)

output += aNumbers[i] + " ";

output += "<br />";

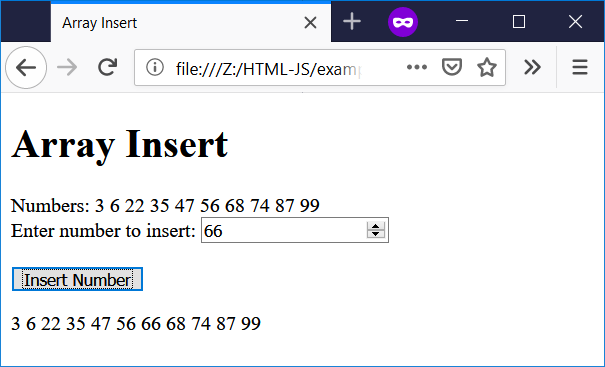
document.getElementById('answer').innerHTML = output;

} // doThings

</script>

</body>

</html>



File name: array-copy.html

<!--

Filename: array-copy.html

Author: Joshua Males

Date Created: December 19, 2018

Last Fix: December 23, 2018

Description: Copy Array

-->

<!DOCTYPE html>

<html lang="en">

<head>

<title>

Copy Array

</title>

<meta charset="utf-8"/>

</head>

<body>

<h1>

Copy Array

</h1>

<script>

//==========================================================

// Global variables

//==========================================================

let aFirst = [], aSecond = [];

let text, i;

//==========================================================

function buildTable(title, arrSource, arrDest)

{

let text;

text = "<h4>" + title + "</h4>";

text += '<table border="1" bordercolor="black" cellpadding=5>';

text += '<tr align="center"><td>Source</td>';

for (i=0; i<arrSource.length; i++)

text += '<td>' + arrSource[i] + '</td>';

text += '</tr>';

text += '<tr align="center"><td>Destination</td>';

for (i=0; i<arrDest.length; i++)

text += '<td>' + arrDest[i] + '</td>';

text += '</tr>';

text += '</table>';

return text;

} // buildTable

</script>

<table border="1" bordercolor="black" cellpadding=10>

<tr>

<td colspan=2>

<script>

for (i=0; i < 4; i++)

aFirst[i] = Math.floor(Math.random() \* 100) + 1;

// Try to make an a=b copy

aSecond = aFirst;

aSecond[0] = 0;

document.write(buildTable("b = a Copy", aFirst, aSecond));

</script>

</td>

</tr>

<tr>

<td>

<script>

for (i=0; i < 4; i++)

aFirst[i] = Math.floor(Math.random() \* 100) + 1;

aSecond = []; // Stop pointing at the first array

// Copy with b[i] = a[i]

for (i=0; i < 4; i++)

aSecond[i] = aFirst[i];

aSecond[0] = 0;

document.write(buildTable("b[i] = a[i] Copy",

aFirst, aSecond));

</script>

</td>

<td>

<script>

for (i=0; i < 4; i++)

aFirst[i] = Math.floor(Math.random() \* 100) + 1;

// Copy with [...]

aSecond = [...aFirst];

aSecond[0] = 0;

document.write(buildTable("b = [...a] Copy",

aFirst, aSecond));

</script>

</td>

</tr>

<tr>

<td>

<script>

for (i=0; i < 4; i++)

aFirst[i] = Math.floor(Math.random() \* 100) + 1;

// Copy with concat

aSecond = [].concat(aFirst);

aSecond[0] = 0;

document.write(buildTable("b = [].concat(a) Copy",

aFirst, aSecond));

</script>

</td>

<td>

<script>

for (i=0; i < 4; i++)

aFirst[i] = Math.floor(Math.random() \* 100) + 1;

// Copy with from

aSecond = Array.from(aFirst);

aSecond[0] = 0;

document.write(buildTable("b = Array.from(a) Copy",

aFirst, aSecond));

</script>

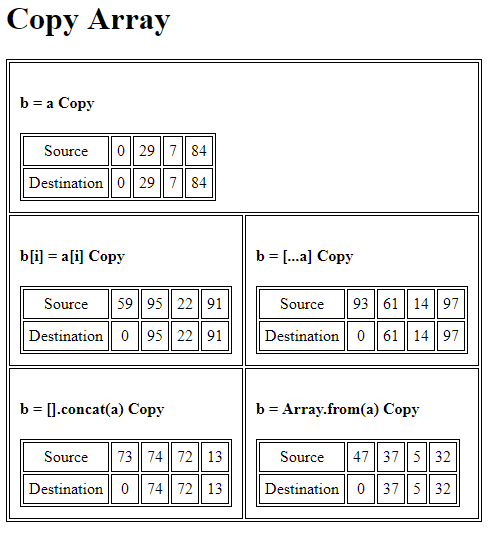
</td>

</tr>

</table>

</body>

</html>



|  |  |
| --- | --- |
| **Chapter 14 – Advanced Arrays** | **פרק 14 – מערכים מתקדמים** |

File name: array-2D-1.html

<!--

Filename: array-2D-1.html

Author: Joshua Males

Date Created: November 25, 2018

Last Fix: November 25, 2018

Description: 2D Array - Simple Define

-->

<!DOCTYPE html>

<html lang="en">

<head>

<title>

2D Array - Simple Define

</title>

<meta charset="utf-8"/>

</head>

<body>

<h1>2D Array - Simple Define</h1>

<script>

let aNumbers = [

[0,1,2,3,4,5,6,7,8,9],

[0,1,2,3,4,5,6,7,8,9],

[0,1,2,3,4,5,6,7,8,9],

[0,1,2,3,4,5,6,7,8,9],

[0,1,2,3,4,5,6,7,8,9],

[0,1,2,3,4,5,6,7,8,9],

[0,1,2,3,4,5,6,7,8,9],

[0,1,2,3,4,5,6,7,8,9],

[0,1,2,3,4,5,6,7,8,9],

[0,1,2,3,4,5,6,7,8,9],

];

let row, column;

for (row=0; row < aNumbers.length; row++) {

for (column=0; column < aNumbers[0].length; column++)

document.write(aNumbers[row][column] + " ");

// Jump to new line

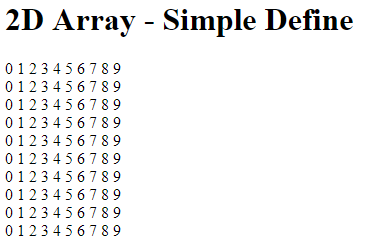
document.write("<br />");

}

</script>

</body>

</html>



File name: array-2D-2.html

<!--

Filename: array-2D-2.html

Author: Joshua Males

Date Created: November 25, 2018

Last Fix: November 25, 2018

Description: 2D Array - Define #2

-->

<!DOCTYPE html>

<html lang="en">

<head>

<title>

2D Array - Define #2

</title>

<meta charset="utf-8"/>

</head>

<body>

<h1>2D Array - Define #2</h1>

<script>

let aNumbers = [];

let row, column;

// Define 2D array - an array of arrays

for (row=0; row < 10; row++)

aNumbers[row] = new Array(10);

// Fill 2D array with values

for (row=0; row < 10; row++)

for (column=0; column < 10; column++)

aNumbers[row][column] = (row + column) % 10;

// Print 2D array

for (row=0; row < aNumbers.length; row++) {

for (column=0; column < aNumbers[0].length; column++)

document.write(aNumbers[row][column] + " ");

// Jump to new line

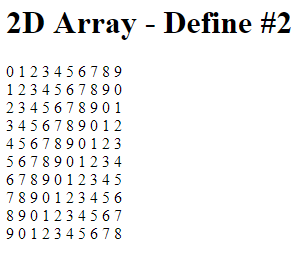
document.write("<br />");

} // for row

</script>

</body>

</html>



File name: array-2D-chess.html

<!--

Filename: array-2D-chess.html

Author: Joshua Males

Date Created: November 25, 2018

Last Fix: November 25, 2018

Description: Chessboard

-->

<!DOCTYPE html>

<html lang="en">

<head>

<title>

Chessboard

</title>

<meta charset="utf-8"/>

</head>

<body>

<h1>Chessboard</h1>

<script>

let board = [

['R','N','B','Q','K','B','N','R'],

['P','P','P','P','P','P','P','P'],

[' ',' ',' ',' ',' ',' ',' ',' '],

[' ',' ',' ',' ',' ',' ',' ',' '],

[' ',' ',' ',' ',' ',' ',' ',' '],

[' ',' ',' ',' ',' ',' ',' ',' '],

['p','p','p','p','p','p','p','p'],

['r','n','b','q','k','b','n','r'] ];

function print2D(inArray, inRows, inColumns)

{

let row, column;

let text = '<table border="1" \

bordercolor="black" \

cellspacing="0" \

cellpadding="10">';

// Print 2D array

for (row=0; row < inRows; row++) {

text += "<tr>";

for (column=0; column < inColumns; column++)

text += "<td>" + inArray[row][column] + "</td>";

text += "</tr>";

} // for row

text += "</table>";

document.write( text );

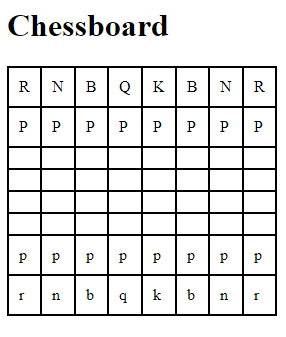
} // print2D

print2D(board, 8, 8);

</script>

</body>

</html>



|  |  |
| --- | --- |
| **Chapter 15 – Searching, Sorting, Merging** | **פרק 15 – חיפוש, מיון, מיזוג** |

File name: linear-search.html

<!--

Filename: linear-search.html

Author: Joshua Males

Date Created: December 17, 2018

Last Fix: June 26, 2019

Description: Linear Search through Array

-->

<!DOCTYPE html>

<html lang="en">

<head>

<title>

Linear Search

</title>

<meta charset="utf-8"/>

</head>

<body>

<h1>

Linear Search

</h1>

<script>

let aNumbers = [5,10,15,20,25,30,35,40,45,50,

55,60,65,70,75,80,85,90,95,100];

let text, i, compares;

// Build and display the table

text = '<table border="1" bordercolor="black" cellpadding=5>';

text += '<tr><td>Index</td>';

for (i=0; i<aNumbers.length; i++)

text += '<td>' + i + '</td>';

text += '</tr>';

text += '<tr><td>Numbers</td>';

for (i=0; i<aNumbers.length; i++)

text += '<td>' + aNumbers[i] + '</td>';

text += '</tr></table><br /><br />';

document.write(text);

</script>

Number to Find:

<input type="number" id="ninput" value="">

<br><br>

<button id="button1" onclick="doThings()">

Search

</button>

<p id="answer"></p>

<script>

//==========================================================

function linearSearch(arr, findMe)

{

let index = 0

compares = 1;

index = 0;

while ((index < arr.length) && (arr[index] < findMe)) {

index++;

compares++;

}

if (arr[index] == findMe)

return index;

return -1; // Not found

} // linearSearch

//==========================================================

function doThings()

{

// Get the value from the input box and store

let findMe = parseInt(document.getElementById("ninput").value);

let position = linearSearch(aNumbers, findMe);

if (position != -1)

document.getElementById('answer').innerHTML =

"Found at index " + position +

". Compares = " + compares;

else

document.getElementById('answer').innerHTML =

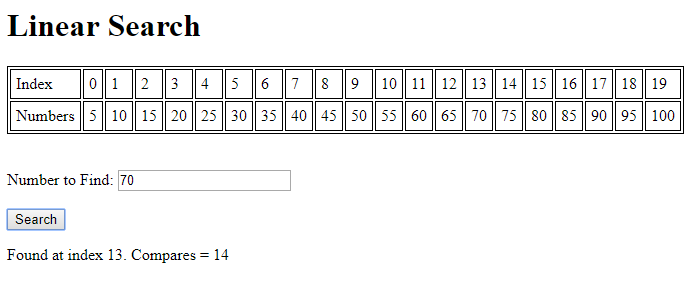
"Not found. Compares = " + compares;

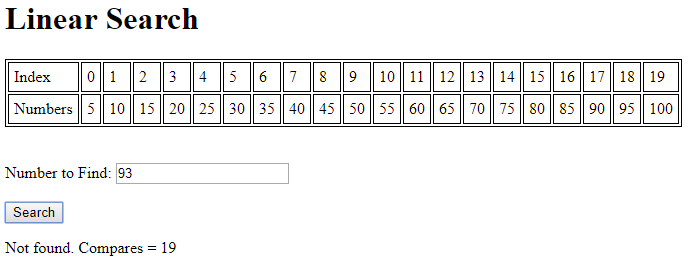
} // doThings()

</script>

</body>

</html>





File name: binary-search.html

<!--

Filename: binary-search.html

Author: Joshua Males

Date Created: November 28, 2018

Last Fix: June 26, 2019

Description: Binary Search through Array.

-->

<!DOCTYPE html>

<html lang="en">

<head>

<title>

Binary Search

</title>

<meta charset="utf-8"/>

</head>

<body>

<h1>Binary Search</h1>

<script>

let aNumbers =

[5,10,15,20,25,30,35,40,45,50,55,60,65,70,75,80,85,90,95,100];

let text, i, compares;

// Build and display the table

text = '<table border="1" bordercolor="black" cellpadding=5>';

text += '<tr><td>Index</td>';

for (i=0; i<aNumbers.length; i++)

text += '<td>' + i + '</td>';

text += '</tr>';

text += '<tr><td>Numbers</td>';

for (i=0; i<aNumbers.length; i++)

text += '<td>' + aNumbers[i] + '</td>';

text += '</tr></table><br /><br />';

document.write(text);

</script>

Number to Find:

<input type="number" id="ninput" value="">

<br><br>

<button id="button1" onclick="doThings()">Search </button>

<p id="answer"></p>

<script>

//==========================================================

function binarySearch(arr, findMe)

{

let lowIndex = 0, highIndex = arr.length-1;

let middle;

compares = 0;

while (lowIndex <= highIndex) {

middle = Math.floor((lowIndex + highIndex) / 2);

compares++;

if (arr[middle] == findMe)

return middle;

// Number was not found. Move boundaries.

compares++;

if (arr[middle] < findMe)

lowIndex = middle + 1;

else

highIndex = middle - 1;

} // while

return -1; // Not found

} // binarySearch

//==========================================================

function doThings()

{

// Get the value from the input box and store in x

let findMe = parseInt(document.getElementById("ninput").value);

let position = binarySearch(aNumbers, findMe);

if (position != -1)

document.getElementById('answer').innerHTML =

"Found at index " + position +

". Compares = " + compares;

else

document.getElementById('answer').innerHTML =

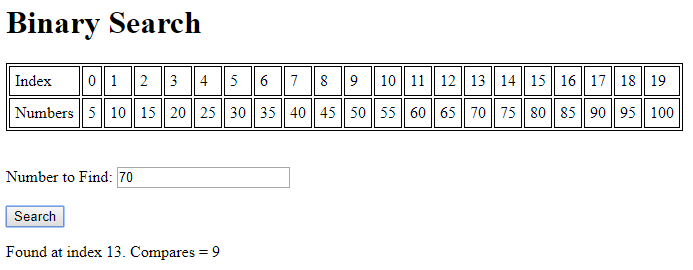
"Not found. Compares = " + compares;

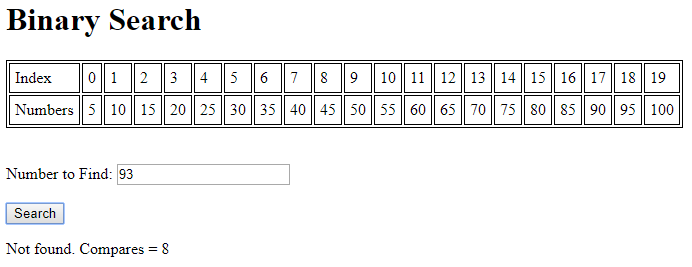
} // doThings()

</script>

</body>

</html>





File name: bubble-sort.html

<!--

Filename: bubble-sort.html

Author: Joshua Males

Date Created: November 29, 2018

Last Fix: June 26, 2019

Description: Bubble Sort

-->

<!DOCTYPE html>

<html lang="en">

<head>

<title>

Bubble Sort

</title>

<meta charset="utf-8"/>

</head>

<body>

<h1>

Bubble Sort

</h1>

<script>

//==========================================================

let aNumbers = [];

let text, i, compares;

for (i=0; i < 10; i++)

aNumbers[i] = Math.floor(Math.random() \* 100) + 1;

// Build and display the table as we sort

text = '<table border="1" bordercolor="black" cellpadding=5>';

document.write(text);

document.write(buildTable("Original Order", aNumbers));

bubbleSort(aNumbers);

text = '</table><br />';

document.write(text);

//==========================================================

function buildTable(title, arr)

{

let text;

text = '<tr><td width="35%">' + title + '</td>';

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

text += '<td>' + arr[i] + '</td>';

text += '</tr>';

return text;

} // buildTable

//==========================================================

function bubbleSort(arr)

{

let i, isSorted, temp, tripCount=0;

do {

isSorted = true;

for (i=0; i<arr.length-1; i++)

if (arr[i] > arr[i+1]) {

// Numbers are out of order

isSorted = false;

// Swap the numbers

temp = arr[i];

arr[i] = arr[i+1];

arr[i+1] = temp;

}

// These next three lines are for nice output

tripCount++;

if (!isSorted)

document.write(buildTable("After Trip " +

tripCount, aNumbers));

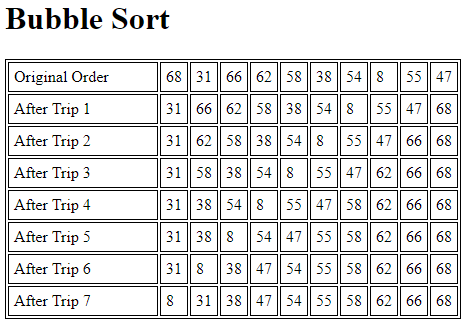
} while (!isSorted);

} // bubbleSort

</script>

</body>

</html>



File name: merge.html

<!--

Filename: merge.html

Author: Joshua Males

Date Created: December 17, 2018

Last Fix: June 26, 2019

Description: Merge Two Sorted Arrays

-->

<!DOCTYPE html>

<html lang="en">

<head>

<title>

Merge Two Sorted Arrays

</title>

<meta charset="utf-8"/>

</head>

<body>

<h1>

Merge Two Sorted Arrays

</h1>

<script>

//==========================================================

// Global variables

//==========================================================

let aFirst = [], aSecond = [];

let aMerged = []; // Destination for merged array

let aIndex1 = [], aIndex2 = [];

let text, i;

//==========================================================

for (i=0; i < 10; i++)

aFirst[i] = Math.floor(Math.random() \* 100) + 1;

bubbleSort(aFirst);

document.write(buildTable("First Array", aFirst));

for (i=0; i < 7; i++)

aSecond[i] = Math.floor(Math.random() \* 100) + 1;

bubbleSort(aSecond);

document.write(buildTable("Second Array", aSecond));

// Merge the two arrays and display.

merge(aFirst, aSecond);

document.write(buildTable("Merged Array", aMerged, true));

//==========================================================

function bubbleSort(arr)

{

let i, isSorted, temp;

do {

isSorted = true;

for (i=0; i<arr.length-1; i++)

if (arr[i] > arr[i+1]) {

isSorted = false;

// Swap the numbers

temp = arr[i];

arr[i] = arr[i+1];

arr[i+1] = temp;

}

} while (!isSorted);

} // bubbleSort

//==========================================================

function buildTable(title, arr, withIndices=false)

{

let text;

text = '<table border="1" bordercolor="black" cellpadding=5>';

text += '<tr align="center"><td>Index</td>';

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

text += '<td>' + i + '</td>';

text += '</tr>';

// For the last table we also show the index arrays

if (withIndices) {

// First Index

text += '<tr align="center"><td>Index1</td>';

for (i=0; i<aIndex1.length; i++)

text += '<td>' + aIndex1[i] + '</td>';

text += '</tr>';

// Second Index

text += '<tr align="center"><td>Index2</td>';

for (i=0; i<aIndex2.length; i++)

text += '<td>' + aIndex2[i] + '</td>';

text += '</tr>';

}

text += '<tr><td>' + title + '</td>';

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

text += '<td>' + arr[i] + '</td>';

text += '</tr>';

text += '</table><br />';

return text;

} // buildTable

//==========================================================

function merge(arr1, arr2)

{

let index1=0, index2=0;

while ((index1 < arr1.length) && (index2 < arr2.length)) {

if (arr1[index1] < arr2[index2]) {

aMerged.push(arr1[index1]); // Add item from arr1

aIndex1.push(index1++);

aIndex2.push(index2);

}

else {

aMerged.push(arr2[index2]); // Add item from arr2

aIndex1.push(index1);

aIndex2.push(index2++);

}

}

// If we made it here, at least one array has been copied.

// If there is more to copy from arr1, then copy

while (index1 < arr1.length) {

aMerged.push(arr1[index1]); // Add item from arr1

aIndex1.push(index1++);

aIndex2.push(" ");

}

// If there is more to copy from arr2, then copy

while (index2 < arr2.length) {

aMerged.push(arr2[index2]); // Add item from arr2

aIndex1.push(" ");

aIndex2.push(index2++);

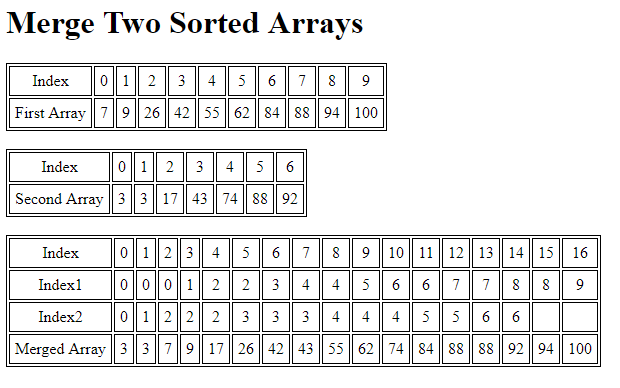
}

} // merge

</script>

</body>

</html>



|  |  |
| --- | --- |
| **Chapter 16 – Strings** | **פרק 16 – מחרוזות** |

File name: strings-functions1.html

<!--

Filename: strings-functions1.html

Author: Joshua Males

Date Created: June 12, 2019

Last Fix: June 12, 2019

Description: Strings - Basic Functions

-->

<!DOCTYPE html>

<html lang="en">

<head>

<title>Strings - Basic Functions</title>

<meta charset="utf-8"/>

</head>

<body>

<h1>Strings - Basic Functions</h1>

Our String: <input id="xinput" value="I Like Dogs">

<br><br>

<button onclick="showSize()">Show String Size</button><br /><br />

<button onclick="showLowerCase()">Show Lower Case</button><br /><br />

<button onclick="showUpperCase()">Show Upper Case</button><br /><br />

<p id="answer"></p>

<script>

function showSize()

{

// Get the value from the input box and store in sText

let sText = document.getElementById("xinput").value;

let sResult = "String size is " + sText.length;

document.getElementById('answer').innerHTML = sResult ;

} // end of showSize()

//=============================================

function showLowerCase()

{

// Get the value from the input box and store in sText

let sText = document.getElementById("xinput").value;

let sResult = "Lower case string is: " + sText.toLowerCase();

document.getElementById('answer').innerHTML = sResult ;

} // end of showLowerCase()

//=============================================

function showUpperCase()

{

// Get the value from the input box and store in sText

let sText = document.getElementById("xinput").value;

let sResult = "Upper case string is: " + sText.toUpperCase();

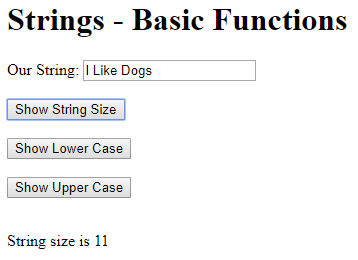
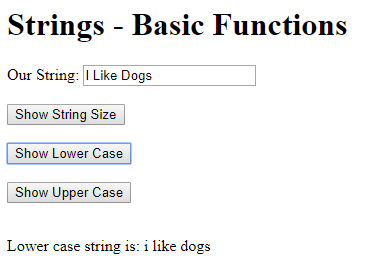
document.getElementById('answer').innerHTML = sResult ;

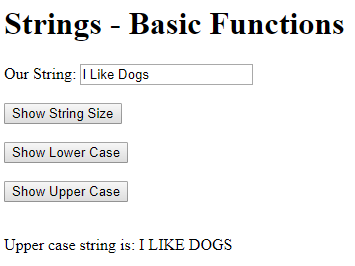
} // end of showUpperCase()

</script>

</body>

</html>



File name: strings-append.html

<!--

Filename: strings-append.html

Author: Joshua Males

Date Created: June 12, 2019

Last Fix: June 12, 2019

Description: Strings - Append Functions

-->

<!DOCTYPE html>

<html lang="en">

<head>

<title>Strings - Append Functions</title>

<meta charset="utf-8"/>

</head>

<body>

<h1>Strings - Append Functions</h1>

String #1: <input id="input1" value="I Like Dogs.">

<br><br>

String #2: <input id="input2" value="I Like Cats.">

<br><br>

String #3: <input id="input3" value="I Like Horses.">

<br><br>

<button onclick="add1And2()">Add 1 and 2</button><br /><br />

<button onclick="addAll()">Add All</button><br /><br />

<p id="answer"></p>

<script>

function add1And2()

{

// Get the value from the input box and store in x

let str1 = document.getElementById("input1").value;

let str2 = document.getElementById("input2").value;

let sResult = "String #1 + String #2 = " +

str1 + " " + str2;

document.getElementById('answer').innerHTML = sResult;

sResult = "Using str1.concat(\" \") and str1.concat(str2): ";

str1 = str1.concat(" ");

str1 = str1.concat(str2);

document.getElementById('answer').innerHTML += "<br />" +

sResult + str1;

} // end of add1And2()

//=============================================

function addAll()

{

// Get the value from the input box and store in x

let str1 = document.getElementById("input1").value;

let str2 = document.getElementById("input2").value;

let str3 = document.getElementById("input3").value;

let sResult = "String #1 + String #2 + String #3 = " +

str1 + " " + str2 + " " + str3;

document.getElementById('answer').innerHTML = sResult;

sResult = "Using str1.concat(\" \", str2, \" \", str3): ";

str1 = str1.concat(" ", str2, " ", str3);

document.getElementById('answer').innerHTML += "<br />" +

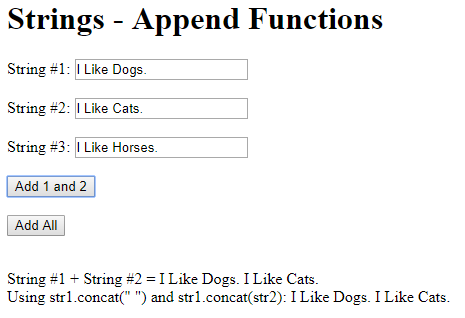
sResult + str1;

} // end of addAll()

</script>

</body>

</html>





File name: strings-part.html

<!--

Filename: strings-part.html

Author: Joshua Males

Date Created: June 13, 2019

Last Fix: July 2, 2019

Description: Extract part of a string

-->

<!DOCTYPE html>

<html lang="en">

<head>

<title>Extract Part of a String</title>

<meta charset="utf-8"/>

</head>

<body>

<h1>Extract Part of a String</h1>

String: <input id="input1" value="Good Morning!">&nbsp;&nbsp;

Index: <input type="number" id="index1" value="0">&nbsp;&nbsp;

<button onclick="index1()">Get Character</button>&nbsp;&nbsp;

<button onclick="getCharCode()">Get CharacterCode</button>

<br /><br />

String: <input id="input2" value="I have a cold">&nbsp;&nbsp;

<button onclick="getLast()">Get Last Character</button>

<br /><br />

String: <input id="input3" value="Felafel in a Pita">&nbsp;&nbsp;

Index: <input type="number" id="index2" value="11">&nbsp;&nbsp;

Length: <input type="number" id="length2" value="6">&nbsp;&nbsp;

<button onclick="getSubStr1()">

Get Substring using substr(index, length)

</button><br /><br />

String: <input id="input4" value="Summer Camp">&nbsp;&nbsp;

Start Index: <input type="number" id="index3" value="0">&nbsp;&nbsp;

End Index: <input type="number" id="index4" value="5">&nbsp;&nbsp;

<button onclick="getSubStr2()">

Get Substring using substring(start, end)

</button><br /><br />

String: <input id="input5" value="Mobile Phone">&nbsp;&nbsp;

Start Index: <input type="number" id="index5" value="4">&nbsp;&nbsp;

End Index: <input type="number" id="index6" value="8">&nbsp;&nbsp;

<button onclick="getSubStr3()">

Get Substring using slice(start, end)

</button><br /><br />

String: <input id="input6"

value="I like the Cleveland Browns">&nbsp;&nbsp;

Index: <input type="number" id="index7" value="11">&nbsp;&nbsp;

<button onclick="getSubStr4()">

Get Substring using substr(index)

</button><br /><br />

<p id="answer"></p>

<script>

function index1()

{

let str1 = document.getElementById("input1").value;

let index = parseInt(document.getElementById("index1").value);

let character = str1[index];

let sResult = "Character at index " + index +

" (using str1[index]) = " + character;

document.getElementById('answer').innerHTML = sResult;

character = str1.charAt(index);

sResult = "Character at index " + index +

" (using str1.CharAt(index)) = " + character;

document.getElementById('answer').innerHTML += "<br />" + sResult;

} // end of index1()

//=====================================

function getCharCode()

{

let str1 = document.getElementById("input1").value;

let index = parseInt(document.getElementById("index1").value);

let characterCode = str1.charCodeAt(index);

let sResult = "Character Code at index " + index +

" (using str1.charCodeAt(index)) = " +

characterCode;

document.getElementById('answer').innerHTML = sResult;

} // end of getCharCode()

//=====================================

function getLast()

{

let str1 = document.getElementById("input2").value;

let character = str1[str1.length-1];

let sResult = "Last Character (using str1[str1.length-1]) = " +

character;

document.getElementById('answer').innerHTML = sResult;

character = str1.charAt(str1.length-1);

sResult = "Last Character (using str1.CharAt(str1.length-1) = " +

character;

document.getElementById('answer').innerHTML += "<br />" + sResult;

} // end of getLast()

//=====================================

// The second parameter specifies the length of the extracted part.

function getSubStr1()

{

let str1 = document.getElementById("input3").value;

let index = parseInt(document.getElementById("index2").value);

let length = parseInt(document.getElementById("length2").value);

let sSub = str1.substr(index, length);

let sResult = "Substring (using str1.substr(index, length)) = " +

sSub;

document.getElementById('answer').innerHTML = sResult;

} // end of getSubStr1()

//=====================================

// The substring() function cannot accept negative indexes.

// The method takes 2 parameters: the start position,

// and the end position (end not included).

function getSubStr2()

{

let str1 = document.getElementById("input4").value;

let indexStart = parseInt(document.getElementById("index3").value);

let indexEnd = parseInt(document.getElementById("index4").value);

let sSub = str1.substring(indexStart, indexEnd);

let sResult = "Substring (using str1.substring(start, end)) = " +

sSub;

document.getElementById('answer').innerHTML = sResult;

} // end of getSubStr2()

//=====================================

// slice - If a parameter is negative,

// the position is counted from the end of the string.

// The method takes 2 parameters: the start position,

// and the end position (end not included).

function getSubStr3()

{

let str1 = document.getElementById("input5").value;

let indexStart = parseInt(document.getElementById("index5").value);

let indexEnd = parseInt(document.getElementById("index6").value);

let sSub = str1.slice(indexStart, indexEnd);

let sResult = "Substring (using str1.slice(start, end)) = " +sSub;

document.getElementById('answer').innerHTML = sResult;

} // end of getSubStr3()

//==========================================================

// If you omit the second parameter, the method will take

// the rest of the string

function getSubStr4()

{

let str1 = document.getElementById("input6").value;

let indexStart = parseInt(document.getElementById("index7").value);

let sSub = str1.substr(indexStart);

let sResult = "Substring (using str1.substr(start)) = " +

sSub;

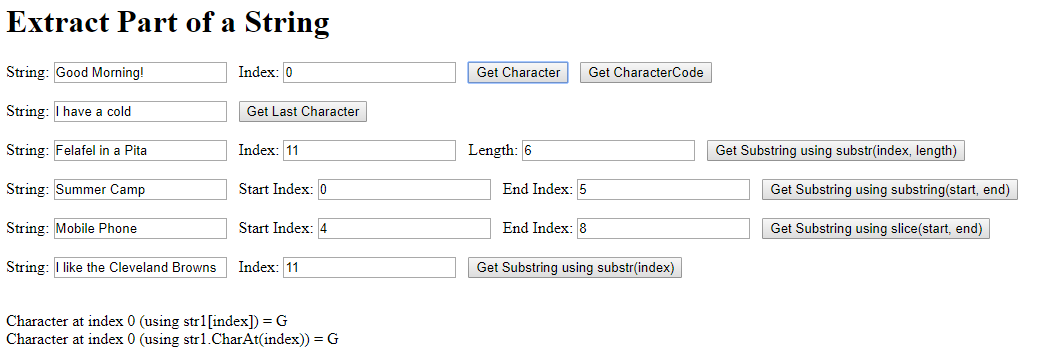
document.getElementById('answer').innerHTML = sResult;

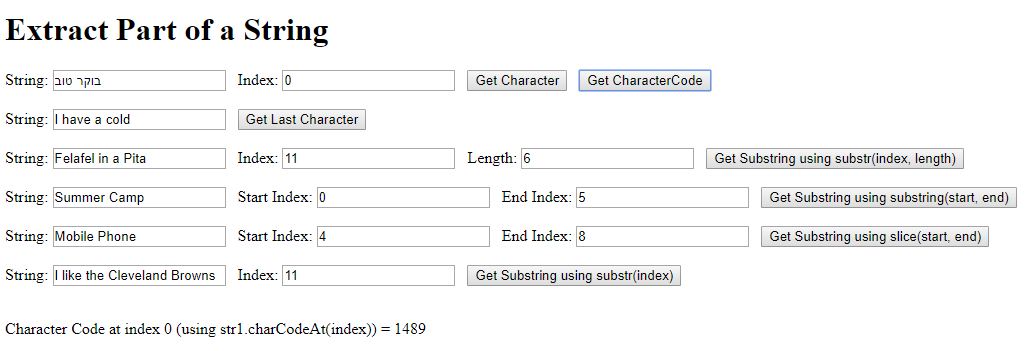
} // end of getSubStr4()

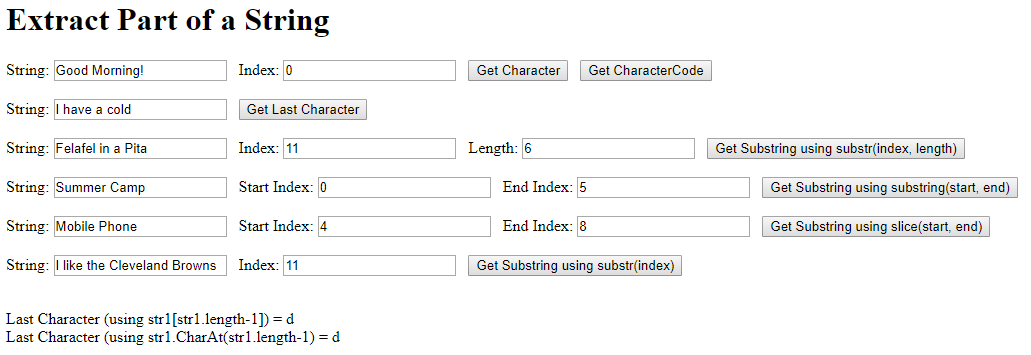
</script>

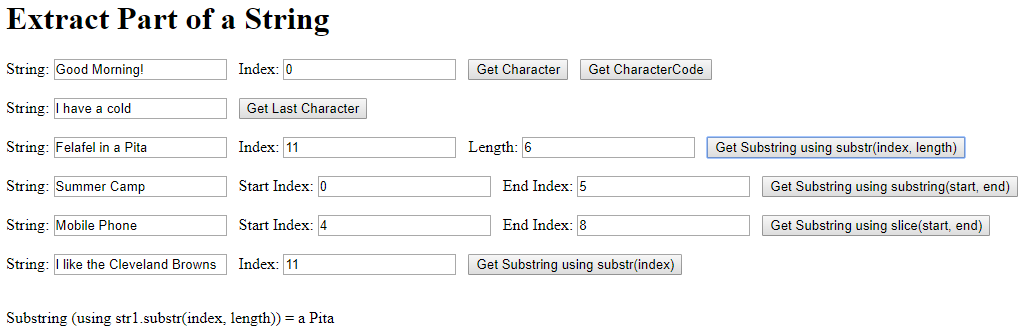
</body>

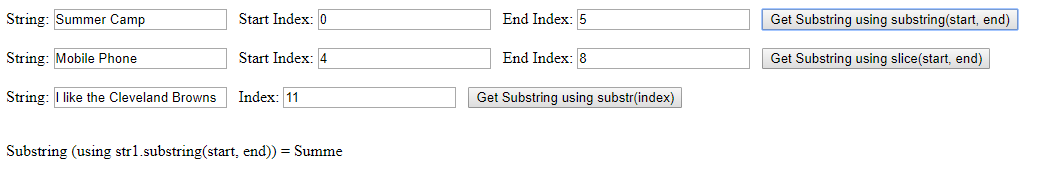
</html>

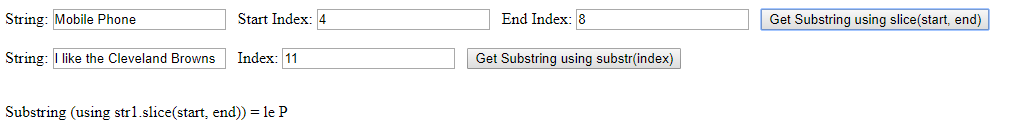


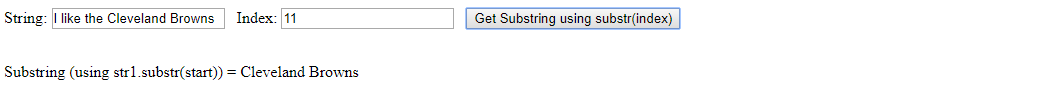












File name: strings-search.html

<!--

Filename: strings-search.html

Author: Joshua Males

Date Created: June 16, 2019

Last Fix: July 2, 2019

Description: Strings - Search Functions: indexOf, lastIndexOf

-->

<!DOCTYPE html>

<html lang="en">

<head>

<title>Strings - Search Functions</title>

<meta charset="utf-8"/>

</head>

<body>

<h1>Strings - Search Functions</h1>

Haystack: <input id="input1" size=30

value="My older brother is 18 years old.">

<br><br>

Needle: <input id="input2" value="old">

<br><br>

Start Index: <input type="number" id="index" value="0">&nbsp;&nbsp;

<button onclick="test\_indexOf()">indexOf</button>&nbsp;

<button onclick="test\_lastIndexOf()">lastIndexOf</button>

<br /><br />

<button onclick="test\_search()">Search Case-Sensitive</button>&nbsp;

<button onclick="test\_searchNoCase()">Search Case-Insensitive</button>

<br />

<p id="answer"></p>

<script>

function test\_search()

{

let sHaystack = document.getElementById("input1").value;

let sNeedle = document.getElementById("input2").value;

let location = sHaystack.search(sNeedle);

if (location == -1)

sResult = "\"" + sNeedle + "\" not found." ;

else

sResult = "\"" + sNeedle + "\" found at index " + location;

document.getElementById('answer').innerHTML = sResult;

} // end of test\_search()

//================================================

function test\_searchNoCase()

{

let sHaystack = document.getElementById("input1").value;

let sNeedle = document.getElementById("input2").value;

let sFindMe = RegExp(sNeedle, "i"); // i=case-insensitive

let location = sHaystack.search(sFindMe);

if (location == -1)

sResult = "\"" + sNeedle + "\" not found." ;

else

sResult = "\"" + sNeedle + "\" found at index " +

location;

document.getElementById('answer').innerHTML = sResult;

} // end of test\_searchNoCase()

//================================================

function test\_indexOf()

{

let sHaystack = document.getElementById("input1").value;

let sNeedle = document.getElementById("input2").value;

let index = parseInt(document.getElementById("index").value);

let location = sHaystack.indexOf(sNeedle, index);

if (location == -1)

sResult = "\"" + sNeedle + "\" not found." ;

else

sResult = "\"" + sNeedle + "\" found at index " + location;

document.getElementById('answer').innerHTML = sResult;

} // end of test\_indexOf()

//================================================

function test\_lastIndexOf()

{

let sHaystack = document.getElementById("input1").value;

let sNeedle = document.getElementById("input2").value;

let index = parseInt(document.getElementById("index").value);

let location = sHaystack.lastIndexOf(sNeedle, index);

if (location == -1)

sResult = "\"" + sNeedle + "\" not found." ;

else

sResult = "Last instance of \"" + sNeedle +

"\" found at index " + location;

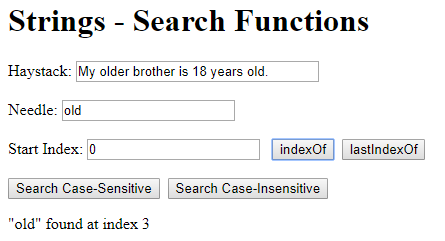
document.getElementById('answer').innerHTML = sResult;

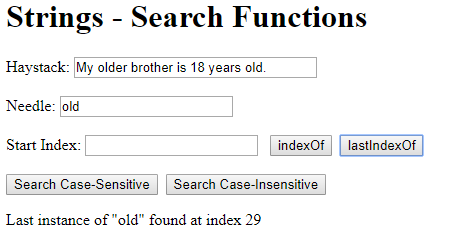
} // end of test\_lastIndexOf()

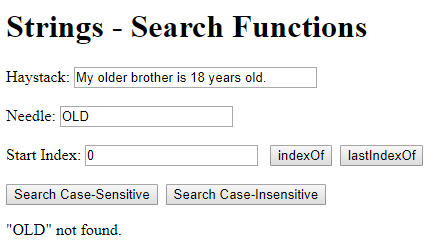
</script>

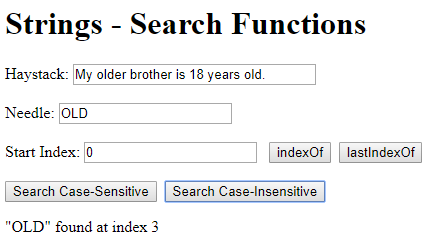
</body>

</html>









File name: strings-replace.html

<!--

Filename: strings-replace.html

Author: Joshua Males

Date Created: June 13, 2019

Last Fix: June 13, 2019

Description: Replace part of a string

-->

<!DOCTYPE html>

<html lang="en">

<head>

<title>Replace Part of a String</title>

<meta charset="utf-8"/>

</head>

<body>

<h1>Replace Part of a String</h1>

String: <input id="input1" size="40"

value="Visit Maale Shomron and Ginot Shomron">

<br /><br />

Case-sensitive Search for: <input id="input2" value="Shomron">

<br /><br />

Replace with: <input id="input3" value="Binyamin">

<br /><br />

Case-insensitive Search for: <input id="input4" value="shomroN">

<br /><br />

Pre-trim String: <input id="input5" value=" Javascript is cool ">

<br /><br />

Replace one character: <input id="input6" value="Bank">&nbsp;&nbsp;

Index: <input type="number" id="index7" value="2"

min="0" max="99">&nbsp;&nbsp;

Replace with: <input id="input8" size="3" value="c">

<br /><br />

<button onclick="replace1()">

Replace First Instance of Text (case-sensitive)

</button><br /><br />

<button onclick="replace2()">

Replace First Instance of Text (case-insensitive)

</button><br /><br />

<button onclick="replace3()">

Replace All Instances of Text (case-sensitive)

</button><br /><br />

<button onclick="replace4()">

Replace All Instances of Text (case-insensitive)

</button><br /><br />

<button onclick="trimExample()">Trim String

</button><br /><br />

<button onclick="replaceCharacter()">

Replace One Character at Location

</button><br /><br />

<p id="answer"></p>

<script>

//=========================================================

function replace1()

{

let str1 = document.getElementById("input1").value;

let sSearchFor = document.getElementById("input2").value;

let sReplaceWith = document.getElementById("input3").value;

let sResult = "Result #1 is: " +

str1.replace(sSearchFor, sReplaceWith);

document.getElementById('answer').innerHTML = sResult;

} // end of replace1()

//==========================================================

function replace2()

{

let str1 = document.getElementById("input1").value;

let sSearchFor = document.getElementById("input4").value;

let sReplaceWith = document.getElementById("input3").value;

sSearchFor = RegExp(sSearchFor, "i");

let sResult = "Result #2 is: " +

str1.replace(sSearchFor, sReplaceWith);

document.getElementById('answer').innerHTML = sResult;

} // end of replace2()

//=======================================================

function replace3()

{

let str1 = document.getElementById("input1").value;

let sSearchFor = document.getElementById("input2").value;

let sReplaceWith = document.getElementById("input3").value;

sSearchFor = RegExp(sSearchFor, "g"); // g=global

let sResult = "Result #3 is: " +

str1.replace(sSearchFor, sReplaceWith);

document.getElementById('answer').innerHTML = sResult;

} // end of replace3()

//===========================================================

function replace4()

{

let str1 = document.getElementById("input1").value;

let sSearchFor = document.getElementById("input4").value;

let sReplaceWith = document.getElementById("input3").value;

sSearchFor = RegExp(sSearchFor, "gi"); // i=case-insensitive

let sResult = "Result #4 is: " +

str1.replace(sSearchFor, sReplaceWith);

document.getElementById('answer').innerHTML = sResult;

} // end of replace4()

//==========================================================

function trimExample()

{

let str1 = document.getElementById("input5").value;

let sResult = "Trim result of \"" + str1 + "\" is: \"" +

str1.trim() + "\"";

document.getElementById('answer').innerHTML = sResult;

} // end of trimExample()

//=========================================================

function replaceCharacter()

{

let str1 = document.getElementById("input6").value;

let index = parseInt(document.getElementById("index7").value);

let sReplaceWith = document.getElementById("input8").value;

let character = sReplaceWith[0]; // make sure only one character

let sResult = "Result is: " +

str1.substr(0, index) +

character +

str1.substr(index + 1);

document.getElementById('answer').innerHTML = sResult;

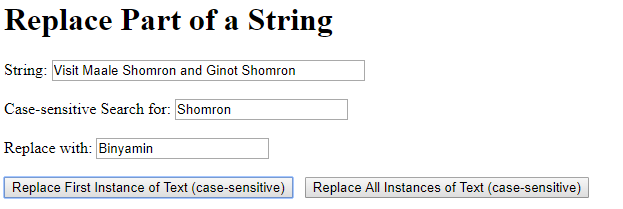
} // end of replaceCharacter()

//=============================================================

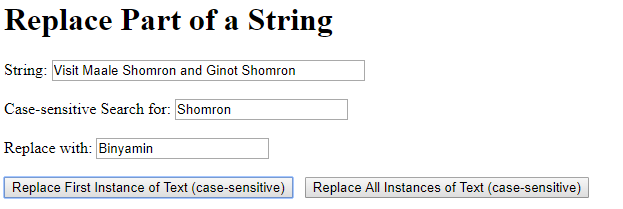
</script>

</body>

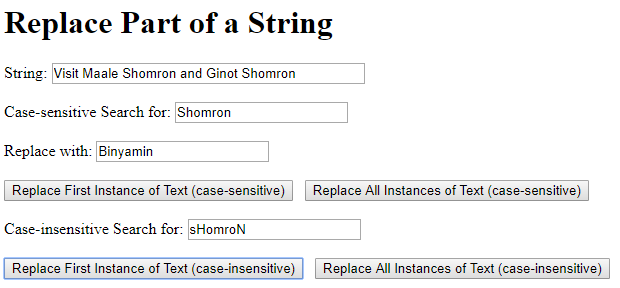
</html>



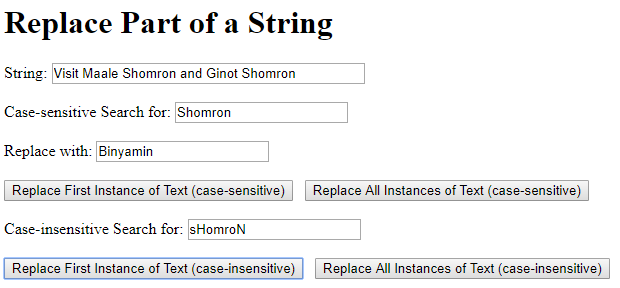




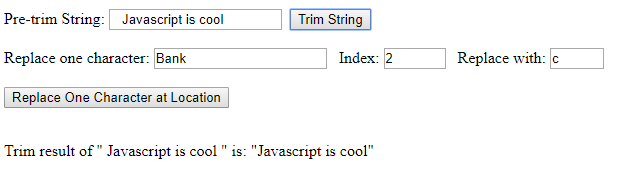


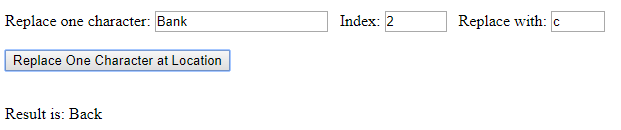












File name: strings-hospital.html

<!--

Filename: strings-hospital.html

Author: Joshua Males

Date Created: June 16, 2019

Last Fix: June 19, 2019

Description: Rotating Hospital Signs

-->

<!DOCTYPE html>

<html lang="en">

<head>

<title>

שלט בית חולים

</title>

<meta charset="utf-8"/>

</head>

<body dir="rtl">

<table border="1" width=40% bordercolor="black" cellpadding=10>

<tr align=center>

<td>

<h1>

בית חולים שערי הישועה

</h1>

</td>

</tr>

<tr align=center>

<td>

<h2 id="sign-text"></h2>

</td>

</tr>

<tr align=center>

<td>

<button onclick="toggleSign()">

שינוי שלט

</button>

</td>

</tr>

</table>

<script>

let sSignText = " רפואה שלמה רפואה שלמה";

let bWarnCohanim = false;

let cell = document.getElementById('sign-text');

cell.style.backgroundColor = "black"; // set the reverse color

cell.innerHTML = sSignText;

cell.style.color = "lime";

setInterval(rotateText, 1000); // Perform rotateText every second

//==========================================

function toggleSign()

{

bWarnCohanim = !bWarnCohanim;

if (bWarnCohanim) {

sSignText = " אזהרת כהנים אזהרת כהנים";

cell.style.color = "red";

} else {

sSignText = " רפואה שלמה רפואה שלמה";

cell.style.color = "lime";

}

cell.innerHTML = sSignText;

} // toggleSign()

//==========================================

function rotateText()

{

if (bWarnCohanim) {

// Rotate right (Hebrew)

firstChar = sSignText[0]; // Take the first character

sSignText = sSignText.slice(1) + firstChar; // Add to the end

} else {

// Rotate left (Hebrew)

// Take the last character

lastChar = sSignText[sSignText.length-1];

// Add the last character to the beginning.

sSignText = lastChar +

sSignText.substr(0, sSignText.length-1);

}

cell.innerHTML = sSignText;

} // rotateText()

</script>

</body>

</html>





File name: strings-compare.html

<!--

Filename: strings-compare.html

Author: Joshua Males

Date Created: July 1, 2019

Last Fix: July 1, 2019

Description: Strings - Compare Functions

-->

<!DOCTYPE html>

<html lang="en">

<head>

<title>Strings - Compare Functions</title>

<meta charset="utf-8"/>

</head>

<body>

<h1>Strings - Compare Functions</h1>

String #1: <input id="input1" value="a">

<br><br>

String #2: <input id="input2" value="B">

<br><br>

<button onclick="compare()">Compare</button>&nbsp;

<button onclick="compareNoCase()">Compare Case-insensitive

</button><br /><br />

<button onclick="compare()">Compare using localeCompare()

</button><br />

<p id="answer"></p>

<script>

function compare()

{

// Get the values from the input boxes

let str1 = document.getElementById("input1").value;

let str2 = document.getElementById("input2").value;

let sResult;

if (str1 < str2)

sResult = "String #1 is less than String #2";

else

if (str1 > str2)

sResult = "String #1 is greater than String #2";

else

sResult = "String #1 equals String #2";

document.getElementById('answer').innerHTML = sResult;

} // end of compare()

//=============================================

function compareNoCase()

{

// Get the values from the input boxes

let str1 = document.getElementById("input1").value;

let str2 = document.getElementById("input2").value;

str1 = str1.toLowerCase();

str2 = str2.toLowerCase();

let sResult;

if (str1 < str2)

sResult = "String #1 is less than String #2";

else

if (str1 > str2)

sResult = "String #1 is greater than String #2";

else

sResult = "String #1 equals String #2";

document.getElementById('answer').innerHTML = sResult;

} // end of compareNoCase()

//=============================================

function uselocalCompare()

{

// Get the values from the input boxes

let str1 = document.getElementById("input1").value;

let str2 = document.getElementById("input2").value;

let sResult;

let compareValue = str1.uselocalCompare(str2);

// compareValue was set to -1, 0, or 1

if (compareValue < 0)

sResult = "String #1 is less than String #2";

else

if (compareValue > 0)

sResult = "String #1 is greater than String #2";

else // compareValue == 0

sResult = "String #1 equals String #2";

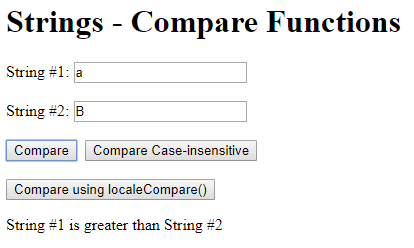
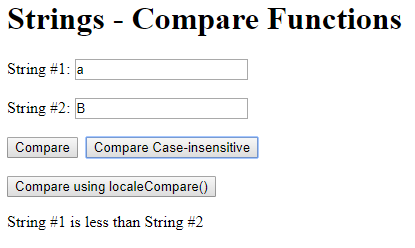
document.getElementById('answer').innerHTML = sResult;

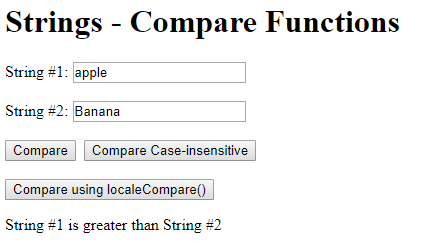
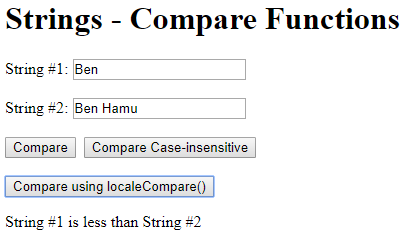
} // end of uselocalCompare()

</script>

</body>

</html>

|  |  |
| --- | --- |
| **Chapter 17 – Final Bingo** | **פרק 17 – סיום: בינגו** |

File name: final-bingo.html

<!--

Filename: final-bingo.html

Author: Joshua Males

Date Created: December 23, 2018

Last Fix: July 8, 2019

Description: Final Bingo Game

-->

<!DOCTYPE html>

<html lang="en">

<head>

<title>Final Bingo Game</title>

<meta charset="utf-8"/>

</head>

<body>

<h1>Final Bingo Game</h1>

<script src="bingo.js"></script>

<script>

drawFirstBoard();

</script>

<h2 id="lastNumber">Last Number: </h2>

<button id="btnPick" onclick="pickNumber()">Pick Number</button>&nbsp;

<button id="btnRestart" onclick="restartGame()">Restart</button>&nbsp;

<button id="btnCard" onclick="createCard()">Create Card</button>

</body>

</html>

File name: bingo.js

/\*

Filename: bingo.js

Author: Joshua Males

Date Created: December 23, 2018

Last Fix: July 4, 2019

Description: Bingo Variables and Functions

\*/

// Global Variables

const NUM\_ROWS = 5;

const NUM\_COLS = 5;

const NUMS\_PER\_COL = 15;

const TABLE\_SIZE = NUMS\_PER\_COL \* NUM\_ROWS;

let aPicked = [];

let aFirstLetter = ['B', 'I', 'N', 'G', 'O'];

//============================================

function drawFirstBoard()

{

let tdNumber = 1;

document.write('<table border="1" bordercolor="black" cellpadding=10>' +

"\n");

for (row = 0; row < NUM\_ROWS; row++) {

// Set color for each row. Write letter (B,I,N,G,O).

switch (row) {

case 0: document.write('<tr bgcolor="aqua"><td><b>B</b></td>');

break;

case 1: document.write('<tr bgcolor="yellow"> <td><b>I</b></td>');

break;

case 2: document.write('<tr bgcolor="white"> <td><b>N</b></td>');

break;

case 3: document.write('<tr bgcolor="fuchsia"><td><b>G</b></td>');

break;

case 4: document.write('<tr bgcolor="lime"> <td><b>O</b></td>');

break;

}

// Now add the numbers for each row on the board

for (column=0; column<NUMS\_PER\_COL; column++) {

document.write('<td id="td' + tdNumber + '">' + tdNumber +

'</td>');

tdNumber++;

}

document.writeln('</tr>' + "\n");

} // for row

document.writeln('</table><br />' + "\n"); // Add space at bottom

ClearBoard(); // Marked all numbers as not picked

} // drawFirstBoard()

//============================================

function redrawNumber(number) // between 1-TABLE\_SIZE

{

// Calculate the row for the color

row = Math.floor((number-1) / NUMS\_PER\_COL);

switch (row) {

case 0: color = "aqua"; break;

case 1: color = "yellow"; break;

case 2: color = "white"; break;

case 3: color = "fuchsia"; break;

case 4: color = "lime"; break;

}

let cellId = "td" + number; // Get the id for the table cell

let cell = document.getElementById(cellId);

if (aPicked[number]) {

cell.style.backgroundColor = "black"; // set the reverse color

cell.style.color = color; // set the reverse color

}

else {

cell.style.backgroundColor = color; // set the regular color

cell.style.color = "black"; // set the reverse color

}

} // redrawNumber

//============================================

function pickNumber()

{

let i, allPicked = true;

// If all the numbers were picked, give a warning.

for (i=1; i <= TABLE\_SIZE; i++) {

if (!aPicked[i]) {

allPicked = false;

break; // Exit the loop. Not all picked.

}

}

if (allPicked) {

alert("All numbers picked. Start a new game.");

return;

}

do {

// Loop until we find a number that wasn't picked

number = Math.floor(Math.random() \* TABLE\_SIZE) + 1;

} while (aPicked[number]);

aPicked[number] = true;

redrawNumber(number); // Set the colors on the board

let lastNumberId = document.getElementById("lastNumber");

let firstLettter = aFirstLetter[Math.floor((number-1) /

NUMS\_PER\_COL)];

lastNumberId.innerHTML = "Last Number: " + firstLettter + number;

} // pickNumber()

//============================================

function BoardIsEmpty()

{

for (i=0; i<=TABLE\_SIZE; i++) {

if (aPicked[i])

return false;

}

return true;

} // BoardIsEmpty

//============================================

function ClearBoard()

{

for (i=0; i<=TABLE\_SIZE; i++) {

aPicked[i] = false;

}

} // ClearBoard()

//============================================

function redrawBoard()

{

for (number=1; number<=TABLE\_SIZE; number++)

redrawNumber(number);

let lastNumberId = document.getElementById("lastNumber");

lastNumberId.innerHTML = "Last Number: ";

} // redrawBoard()

//============================================

function restartGame()

{

if (!BoardIsEmpty()) {

if (confirm("Are you sure you want to restart?")) {

ClearBoard();

redrawBoard();

alert("Game Restarted");

}

}

} // restartGame()

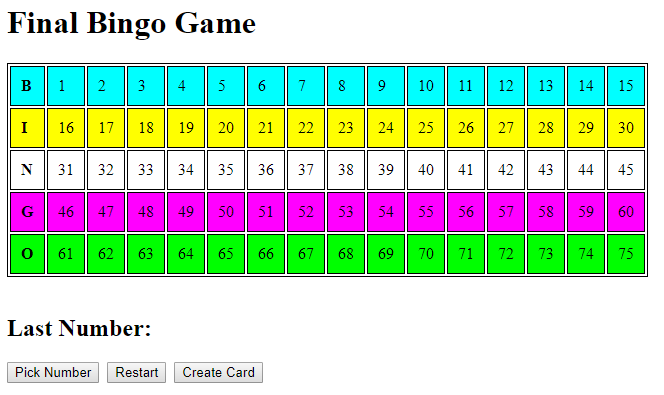
//============================================

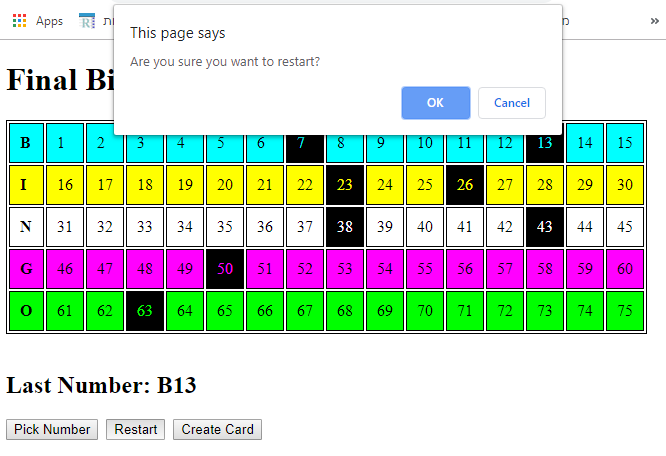
function createCard()

{

// Use the homework assignment from week 14 (2D Arrays)

} // createCard()





File name: css-bingo.html

<!--

Filename: css-bingo.html

Author: Joshua Males

Date Created: December 23, 2018

Last Fix: July 8, 2019

Description: Final Bingo Game with CSS

-->

<!DOCTYPE html>

<html lang="en">

<head>

<title>Final Bingo Game</title>

<meta charset="utf-8"/>

<style type="text/css">

body {

font-family: Arial;

font-size: 20px;

margin: 20px 20px 20px 50px;

}

</style>

</head>

<body>

<h1>Final Bingo Game with CSS</h1>

<script src="bingo.js"></script>

<script>

drawFirstBoard();

</script>

<h2 id="lastNumber">Last Number: </h2>

<button id="btnPick" onclick="pickNumber()">Pick Number</button>&nbsp;

<button id="btnRestart" onclick="restartGame()">Restart</button>&nbsp;

<button id="btnCard" onclick="createCard()">Create Card</button>

</body>

</html>



File name: bootstrap-net-bingo.html

<!--

Filename: bootstrap-net-bingo.html

Author: Joshua Males

Date Created: December 23, 2018

Last Fix: July 8, 2019

Description: Final Bingo Game with net-based Bootstrap

-->

<!DOCTYPE html>

<html lang="en">

<head>

<title>Final Bingo Game</title>

<meta charset="utf-8"/>

<!-- Start Bootstrap CSS -->

<meta name="viewport"

content="width=device-width, initial-scale=1, shrink-to-fit=no">

<link rel="stylesheet"

href="https://stackpath.bootstrapcdn.com/bootstrap/4.3.1/css/bootstrap.min.css">

<!-- End Bootstrap CSS -->

</head>

<body>

<div class="container">

<h1>Final Bingo Game with net-based Bootstrap</h1>

<script src="bingo.js"></script>

<script>

drawFirstBoard();

</script>

<h2 id="lastNumber">Last Number: </h2>

<button id="btnPick" onclick="pickNumber()">Pick Number</button>&nbsp;

<button id="btnRestart" onclick="restartGame()">Restart</button>&nbsp;

<button id="btnCard" onclick="createCard()">Create Card</button>

</div><!-- end of container -->

<!-- Start Bootstrap JS -->

<script src=<https://code.jquery.com/jquery-3.3.1.slim.min.js>></script>

<script src="https://cdnjs.cloudflare.com/ajax/libs/popper.js/1.14.7/umd/popper.min.js">

</script>

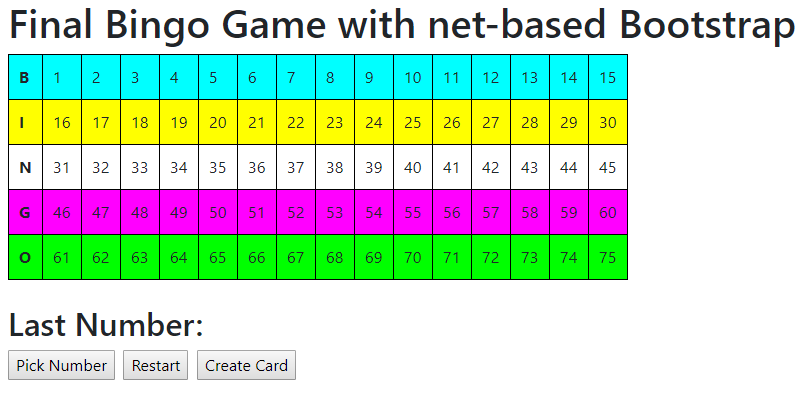
<script src="https://stackpath.bootstrapcdn.com/bootstrap/4.3.1/js/bootstrap.min.js">

</script>

<!-- End Bootstrap JS -->

</body>

</html>



File name: bootstrap-local-bingo.html

<!--

Filename: bootstrap-local-bingo.html

Author: Joshua Males

Date Created: December 23, 2018

Last Fix: July 8, 2019

Description: Final Bingo Game with Local Bootstrap

-->

<!DOCTYPE html>

<html lang="en">

<head>

<title>Final Bingo Game</title>

<meta charset="utf-8"/>

<!-- Start Bootstrap CSS -->

<meta name="viewport"

content="width=device-width, initial-scale=1, shrink-to-fit=no">

<link rel="stylesheet" href="css/bootstrap.min.css">

<!-- End Bootstrap CSS -->

</head>

<body>

<div class="container">

<h1>Final Bingo Game with Local Bootstrap</h1>

<script src="bingo.js"></script>

<script>

drawFirstBoard();

</script>

<h2 id="lastNumber">Last Number: </h2>

<button id="btnPick" onclick="pickNumber()">Pick Number</button>&nbsp;

<button id="btnRestart" onclick="restartGame()">Restart</button>&nbsp;

<button id="btnCard" onclick="createCard()">Create Card</button>

</div><!-- end of container -->

<!-- Start Bootstrap JS -->

<script src="js/jquery-3.3.1.slim.min.js"></script>

<script src="js/bootstrap.bundle.min.js"></script>

<!-- End Bootstrap JS -->

</body>

</html>

