

גירסה 1.00 – 13.10.2002

JavaScript למתחילים – חלק ראשון

מסמך זה הורד מהאתר <http://underwar.livedns.co.il>.
אין להפיץ מסמך זה במדיה כלשהי, ללא אישור מפורש מאת המחבר.
מחבר המסמך איננו אחראי לכל נזק, ישיר או עקיף, שיגרם עקב השימוש במידע המופיע במסמך, וכן לנכונות התוכן של הנושאים המופיעים במסמך. עם זאת, המחבר עשה את מירב המאמצים כדי לספק את המידע המדויק והמלא ביותר.

כל הזכויות שמורות לניר אדר

Nir Adar

Email: underwar@hotmail.com

Home Page: <http://underwar.livedns.co.il>

אנא שלחו תיקונים והערות אל המחבר.

מסמך זה מיועד לאנשים הבקיאים בשפת HTML, והרוצים להרחיב את כישוריהם בעיצוב אתרים.

אם אינך יודע HTML, תוכל להוריד באתר <http://underwar.livedns.co.il> מסמכים המלמדים את השפה.
הדוגמאות במסמך זה מתאימות לדפדפנים השונים בגרסאות 4 ומעלה.

JavaScript היא שפת תכנות קטנה, המשתלבת בין דפי ה-HTML, ומאפשרת להם להיות דינמיים יותר ולהגיב למשתמש.

זו איננה שפה שעוברת הידור (Compile) וקוד הנכתב בשפה זו נשלח אל הצופה באתר כטקסט פשוט, ביחד עם קוד ה-HTML. JavaScript מתפענחת בצד המשתמש, ולא בצד השרת, כמו למשל בשפת ASP.

דברים ש-JavaScript תאפשר לנו לבצע, נעים בין הצגת מידע הקשור למשתמש כגון תאריכים ושעה, לבין יצירת טקסט הנע מעצמו, או הוספת פקדים (controls) לדפים והגבה ללחיצות הגולש עליהם.

דוגמא ראשונה

נביט כעת בדף ה-HTML הבא.

```
<HTML>
<HEAD>
<TITLE>Our First Script</TITLE>
</HEAD>
<BODY>
<CENTER>
<H1>Our First Script</H1>
</CENTER>
<P>Today is
</BODY>
</HTML>
```

הדף נראה כך :

Our First Script

Today is

נרצה לגרום לך שאחרי המילים Today is יופיע התאריך של היום. על מנת לבצע זו, נוסיף את הקוד הבא אחרי המילים הנ"ל :

```
<SCRIPT Language="JavaScript">
var today = new Date()
document.write(today)
</SCRIPT>
```

קוד הדף שלנו יהיה כעת כמוצג :

```
<HTML>
<HEAD>
<TITLE>Our First Script</TITLE>
</HEAD>
<BODY>
<CENTER>
<H1>Our First Script</H1>
</CENTER>
<P>Today is
<SCRIPT Language="JavaScript">
var today = new Date()
document.write(today)
</SCRIPT>
</BODY>
</HTML>
```

הדף יראה משהו בסגנון :

Our First Script

Today is Sat Oct 12 06:15:41 UTC+0200 2002

בזאת סיימנו לכתוב את הדף הראשון שלנו המשתמש ב-JavaScript. דף זה הוא די פשוט, ואין לו כל כך שימוש באתרים גדולים, אולם ראינו כיצד JavaScript מבצעת מטלה שלא יכולנו לבצע בעזרת HTML בלבד.

התג <SCRIPT>

בעזרת התגים `<SCRIPT language="JavaScript">` `</SCRIPT>` אנו מזהים קוד ב-JavaScript. כל טקסט שיופיע בין התג הפותח לתג המסיים, ייחשב כקוד JavaScript. רוב הדפדפנים יזהו קוד של JavaScript ויבצעו את ההוראות. אולם, ישנם מספר דפדפנים ישנים שאינם תומכים ב-JavaScript. הדפדפנים יתעלמו מהתג `<SCRIPT>`, אולם הם ידפיסו את הטקסט שבין התגים. נרצה למנוע זאת, על מנת שהדפים שלנו יוצגו כראוי גם בדפדפנים אלו. נעשה זאת על ידי הוספת תגי הערות של HTML, בהתחלת וסיום הסקריפט. דפדפנים ישנים יתעלמו מהסקריפט ולא יציגו אותו, ואילו דפדפנים חדשים ידעו להתעלם מההערה ולבצע את הסקריפט.

לכן, הצורה המלאה לפתוח ולסיים קטע סקריפט היא כלהלן:

```
<SCRIPT language="JavaScript">
<!-- hide from old browsers

(some text that is JavaScript code)

//-->
</SCRIPT>
```

נוסיף את שורות אלו לדוגמא שלנו.
התוצאה הסופית תראה כך:

```
<HTML>
<HEAD>
<TITLE>Our First Script</TITLE>
</HEAD>
<BODY>
<CENTER>
<H1>Our First Script</H1>
</CENTER>
<P>Today is
<SCRIPT Language="JavaScript">
<!-- hide from old browsers
var today = new Date()
document.write(today)
//-->
</SCRIPT>
</BODY>
</HTML>
```

מיקום JavaScript

ניתן למקם קוד של JavaScript בחלקים שונים של המסמך.
ניתן למקם JavaScript בין התג <BODY> לתג הסוגר שלו כפי שעשינו בדוגמא זו.
ניתן למקם סקריפטים בין התג <HEAD> לתג הסוגר שלו,
וכן ניתן למקם בחלק מקרים סקריפטים גם בתור פרמטרים של תגים אחרים.
בהמשך נראה דוגמאות לכך.

דפדפנים שונים

כמו קוד HTML, סקריפטים שאנו כותבים עלולים להראות שונה תחת דפדפנים שונים. רצוי לבדוק את הסקריפט תחת מספר דפדפנים.

דוגמא

נרצה כעת להדפיס טקסט על המסך, תוך שימוש בפקודה document.write. מומלץ שהקורא ינסה לכתוב קוד זה בעצמו, על מנת להתרגל לכתיבת JavaScript.

על מנת להדפיס את הטקסט, נכתוב קוד הדומה לקוד של הדוגמא הקודמת. כאשר נרצה להדפיס טקסט בעזרת document.write, נכתוב את הטקסט הרצוי במרכאות.

פתרון:

```
<HTML>
<HEAD>
<TITLE>Example 1</TITLE>
</HEAD>
<BODY bgcolor="white">

<SCRIPT language="JavaScript">
<!--
document.write("Hello World")
//-->
</SCRIPT>

</BODY>
</HTML>
```

לאחר שתכתוב קוד זה, שמור אותו בתור קובץ HTML והבט בו בדפדפן. אם הופיעה הודעת שגיאה, במקום שהסקריפט יפעל, שים לב לעובדה הבאה: JavaScript היא שפה שהיא Case Sensitive, כלומר, הביטוי document.write והביטוי DOCUMENT.WRITE הינם שני ביטויים שונים. עלינו להקפיד לכתוב את הפקודה באותיות קטנות, על מנת לא לקבל שגיאות.

ערכים, משתנים ואופרטוריםערכים (Value)

ערך ב-JavaScript הוא פיסת מידע כלשהי, שיכולה להיות מספר, מחרוזת, ערך בוליאני, Null, אובייקט או פונקציה. הטבלה הבאה מגדירה את הערכים השונים בשפת JavaScript:

ערכים ב-JavaScript		
דוגמא	תיאור	סוג
"JavaScript Tutorial"	רצף של אותיות המצויות בין מרכאות.	מחרוזת (String)
12.54	ערך מספרי כלשהו (לא מופיע בין מרכאות).	מספר (Number)
true	אמת (true) או שקר (false).	ערך בוליאני (Boolean)
null	ריקנות, חוסר כל ערך משמעותי.	Null
	משתנים ושיטת של אובייקט.	אובייקט
	הערך המוחזר מהפונקציות.	פונקציות

במהלך מסמך זה ומסמכי ההמשך בנושא JavaScript, נשתמש כל הזמן בערכים אלו.

משתנים

ניתן לשמור נתונים בזיכרון המחשב, על מנת להשתמש בנתונים מאוחר יותר. משתנה הוא מעין מקום בזיכרון, המזוהה על ידי שם, בו אנו יכולים לשמור נתונים, ואחר כך לשלוף אותם מהם.

על מנת ליצור משתנה, נשתמש במילה השמורה var של JavaScript כלהלן:

```
var myVariable
```

לאחר שהגדרנו משתנה, נוכל לשים בו את כל אחד מהערכים שפירטנו בטבלה לעיל. למשל:

```
myVariable = 5
```

בהמשך התוכנית שנכתוב, נוכל לשנות את ערכו של המשתנה לערך אחר:

```
myVariable = 45
```

המשתנה שלנו מכיל כרגע מספר. נוכל גם לשים בו מחרוזת, למשל:

```
myVariable = "Hello, World!!!"
```

נשים לב כעת להבדל בין שפת JavaScript לשפות אחרות, דוגמת C או C++, בהן לכל משתנה יש את הסוג שלו (למשל, מספר או מחרוזת), ולא ניתן להפוך משתנה מסוג אחד למשתנה מסוג אחר.

אנו יכולים ב-JavaScript לתת למשתנה ערך מיד כשאנו יוצרים אותו. לדוגמא:

```
var myVariable = "JavaScript is cool"
```

כמו כן, ניתן ליצור משתנה על ידי השמת ערך לתוכו, ללא צורך במילה var:

```
myVariable2 = "You can create this variable, but it isn't recommended"
```

ניתן ליצור משתנים פשוט על ידי השמת ערך לתוכם. עם זאת, כאשר מתכנתים בצורה זו, למתבונן בקוד יהיה יותר קשה להבין כיצד הוא פועל, וכן ייתכנו באגים סמויים בקוד, כתוצאה משמות משתנים דומים אך לא זהים. לכן, לא מומלץ להשתמש במשתנים מבלי להגדיר אותם ראשית עם המילה השמורה var.

נביט בדוגמא לשימוש במשתנה.

נגדיר כעת משתנה, ונדפיס את ערכו בעזרת document.write:

```
<SCRIPT Language="JavaScript">
<!-- hide from old browsers

myVariable = "JavaScript is Cool"
document.write(myVariable)

//-->
</SCRIPT>
```

הקורא מוזמן להקליד קוד זה ולבדוק אותו בדפדפן, וכן לשנות את ערכו של myVariable לערכים שונים ולראות איך התוצאה משתנה.

JavaScript כשפה המאפשרת מעבר חופשי בין סוגי משתנים

נביט כעת בדוגמא הבאה.
נשים לב כי המשתנה myVariable מוגדר בהתחלה כמחרוזת, לאחר מכן אנו שמים בו מספר, ולאחר מכן סוגי משתנים רבים אחרים. הדוגמא מראה לנו בבירור את העובדה ש-JavaScript מתאימה את סוג המשתנה לתוכן שנמצא בו כעת, ואנו יכולים לשנות תוכן זה בכל עת.
כמו כן, נשים לב לשימוש בסימן //.
סימן זה מציין הערה. כאשר הדפדפן יפענח את הסקריפט שאנו כותבים, הוא יתעלם מכל טקסט שיופיע אחרי הסימן //, עד לסוף אותה שורה.
המשתנה האחרון שאנו מגדירים הוא אובייקט.
נדון באובייקטים באחד מהמסמכים הקרובים בסידרה זו.

```
<SCRIPT Language="JavaScript">
<!-- hide from old browsers

var myVariable = "This is a String" // string
document.write(myVariable)
document.write("<BR>")

myVariable = 45.55 // number
document.write(myVariable)
document.write("<BR>")

myVariable = true // Boolean
document.write(myVariable)
document.write("<BR>")

var myVariable = null // Null
document.write(myVariable)
document.write("<BR>")

myVariable = new Date // object
document.write(myVariable)
document.write("<BR>")

//-->
</SCRIPT>
```

אופרטורים

אופרטורים הם סימנים בהם אנו משתנים בשילוב עם המשתנים, על מנת לבצע פעולות שונות על המשתנים, כגון חיבור, חיסור ועוד.
 הטבלה הבאה מסכמת את האופרטורים הזמינים בשפת JavaScript :

אופרטורים	
operator	what it does
$x + y$ (numeric)	Adds x and y
$x + y$ (string)	Concatenates x and y
$x - y$	Subtracts x from y
$x * y$	Multiplies x b y
x / y	Divides x by y
$x \% y$	Remainder of x / y , modulus
$x++$, $++x$	$x = x + 1$
$x--$, $--x$	$x = x - 1$

תרגיל לדוגמא

צור שני משתנים, בשם x ו- y , ושים בהם את הערכים 34 ו-54 בהתאמה.
 צור משתנה שלישי z שיכיל את הסכום של המשתנים אלו, וצור משתנה רביעי w שיכיל את המכפלה של x ב-2. לאחר מכן, הדפס את החישובים על המסך.

פתרון

```
<HTML>
<HEAD>
<TITLE>Some Math</TITLE>
</HEAD>
<BODY>

<SCRIPT Language="JavaScript">
<!-- hide from old browsers
    var x = 34, y = 53
    var z = x + y, w = x * 2
    document.write("The sum of " + x + " + " + y + " =
" + z + "<BR>")
    document.write("Also, " + x + " * 2 = " + w)
//-->
</SCRIPT>
</BODY>
</HTML>
```

בזאת הגיע לסימו החלק הראשון בסידרת המסמכים אודות JavaScript.

EOF