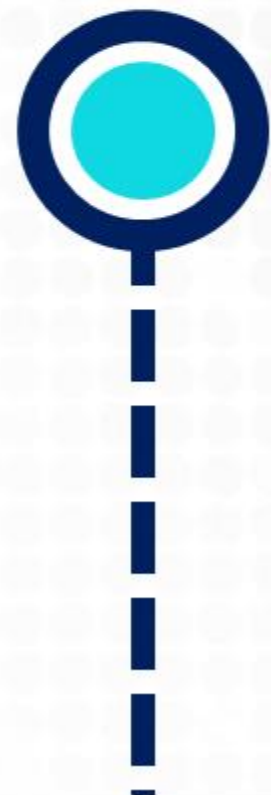


JavaScript DOM

במסגרת הנושא נלמד כיצד JavaScript יכולה לשנות את מסמכי HTML וליצור עמודים דינמיים.



מה זה DOM?

מודל אובייקט של האלמנטים בעמוד ה-html.
האובייקט נוצר על ידי הדפדפן ובנוי במבנה ההיררכי המאפשר גישה לכל רכיביו.

מה מאפשר?

השימוש במודל מאפשר למתכנת גישה ישירה לכל **התגיות**, **המאפיינים**, **העיצוב והאירועים** העמוד.

הגישה כוללת אפשרות עריכה, הוספה ומחיקה של אלמנטים מהעמוד. כמו כן ניתן לאסוף מידע מהמסך ולקבוע באיזה אירוע כל פעולה תתבצע.

מבוא



JS_DOM_Welcome.html

JS_DOM_welcome_Async_Deffer.html

בחלק זה נעמיק ונלמד בנושא תוך שימוש במספר כלים ופתרונות קוד.
בסיום הנושא נוכל לענות על השאלות הבאות:

- מהו DOM?
- מה מאפשר השימוש?
- מתי אפשר לגשת לאלמנטים?
- איפה צריך להטמיע את הסקריפט?
- מה יקרה אם ננסה לגשת לאלמנטים שלא נוצרו?
- איזה הודעת השגיאה מופיע כשמנסים לגשת לאלמנט שלא נוצר?
- מה זה null?
- כיצד ניתן לכתוב ישירות למסמך?

```
<script>
    console.log(document);
</script>
```

Method	הסבר
<code>window.onload = function(){ }</code>	הפונקציה תופעל לאחר שהדף נטען כולל משאבים.
<code>document.write("string text");</code>	כתוב ישירות למסמך HTML

צרו קובץ חדש בשם **JS_DOM_Welcome.html** לצורך תרגול המשימות הבאות.
חשוב לפתור למשימות בהתאם לסדר ולבצע בדיקה בסיום כל שלב.

תרגיל	תיאור המשימה
Ex1	פתחו תגית script באזור head בחלק העליון של המסמך. שנו את התוכן של אלמנט [demo1]. תוך שימוש ב- window.onload
Ex2	בתוך תגית script שפתחנו בחלק העליון של העמוד. נסו לשנות את התוכן של אלמנט [demo2] בתוך אזור זה. הסתכלו בקונסולה <u>בהודעה השגירה</u> והכירו אותה.
Ex3	בתוך תגית script שפתחנו בחלק העליון של העמוד. כתבו פונקציה חדשה והפונקציה תדע לשנות את התוכן של אלמנט [demo4].

צרו קובץ חדש בשם **JS_DOM_Welcome.html** לצורך תרגול המשימות הבאות.
חשוב לפתור למשימות בהתאם לסדר ולבצע בדיקה בסיום כל שלב.

תרגיל	תיאור המשימה
Ex4	פתחו תגית script בחלק התחתון של אזור ה body. שנו את התוכן של אלמנט [demo3].
Ex5	בתוך תגית script שפתחנו בחלק התחתון של העמוד הדפיסו בקונסולה את אובייקט document הדפיסו בקונסולה את אובייקט demo3 הדפיסו בקונסולה את סוג [type] האובייקט demo3
Ex6	בתוך תגית script שפתחנו בחלק התחתון של העמוד צרו משתנה חדש ושימו בתוך את אובייקט demo3 הדפיסו בקונסולה את המשנה שיצרתם.
Ex7	בתוך תגית script שפתחנו בחלק התחתון של העמוד צרו משתנה חדש ושימו בתוך את אובייקט demo100 (שימו לב, אין אלמנט כזה) הדפיסו בקונסולה את המשנה שיצרתם.
Ex8	כתבו ישירות למסמך לפחות 3 שורות שונות באמצעות הפקודה document.write

תפיסת אלמנטים



JS_DOM_getElement.html

JS_DOM_getElement.js

בחלק זה נעמיק ונלמד בנושא תוך שימוש במספר כלים ופתרונות קוד.
בסיום הנושא נוכל לענות על השאלות הבאות:

- איזה מתודות קיימות לתפיסת אלמנט באמצעות id?
- מה המתודות מקבלות כערך? ומה הן מחזירות?
- האם ניתן לשמור את האובייקט במשתנה?
- כיצד ניתן לגשת למידע בתוך האובייקט?

```
<div id="demo1" style="color: blue; font-size: 20px;">Hola Class</div>

<script>

  console.log(document);
  console.log(document.getElementById("demo1"));

  var myElement = document.getElementById("demo1");
  console.log(myElement);

</script>
```

Method	הסבר
document.getElementById("id")	תפוס אלמנט אחד לפי id
document.querySelector("#demo")	תפוס אלמנט אחד לפי סלקטור

JS_DOM_getElements.html

JS_DOM_getElements.js

בחלק זה נעמיק ונלמד בנושא תוך שימוש במספר כלים ופתרונות קוד.
בסיום הנושא נוכל לענות על השאלות הבאות:

- איזה מתודות קיימות לתפיסת אלמנטים?
- מה המתודות מקבלת כערך? ומה המתודה מחזירה?
- האם ניתן לשמור את האובייקט במשתנה?
- כיצד ניתן לגשת למידע בתוך האובייקט?
- כיצד ניתן לעבור בלולאה על האובייקט שמתקבל?

Method	הסבר
<code>document.getElementsByTagName("name")</code>	תפוס אלמנטים לפי שם התגית / אלמנט
<code>document.getElementsByClassName("name")</code>	תפוס אלמנטים לפי class
<code>document.querySelectorAll(".demos")</code>	תפוס אלמנטים לפי סלקטור

תרגול תפיסת אלמנטים

יש להיעזר בקובץ `JS_DOM_getElement.html` של המרצה ואילו לחבר מסמך `JS_DOM_getElement.js`.
מסמך ה JS יכיל פונקציות בהתאם לסדר הבא:

תרגיל	תיאור המשימה
Ex1	הדפיסו לקונסולה את אובייקט document
Ex2	הדפיסו לקונסולה את אובייקט demo1
Ex3	הדפיסו לקונסולה את סוג [type] האובייקט demo1

תרגול תפיסת אלמנטים

יש להיעזר בקובץ `JS_DOM_getElements.html` של המרצה ואילו לחבר מסמך `JS_DOM_getElements.js`.
מסמך ה JS יכיל פונקציות בהתאם לסדר הבא:

תרגיל	תיאור המשימה
demo1	צורו פונקציה חדשה. הפונקציה תדע לשמור במשתנה את כל האלמנטים מסוג li. כמו כן הריצו לולאה על משנה והדפיסו את תוכן האלמנטים.
demo2	צורו פונקציה חדשה. הפונקציה תדע לשמור במשתנה את כל האלמנטים הכוללים את המחלקה [ex1] כמו כן הריצו לולאה על משנה והדפיסו את תוכן האלמנטים. * הפונקציה צריכה להכיל שימוש ב - <code>getElementsByClassName</code>
demo3	צורו פונקציה חדשה. הפונקציה תדע לשמור במשתנה את כל האלמנטים הכוללים את המחלקה [ex1] כמו כן הריצו לולאה על משנה והדפיסו את תוכן האלמנטים. * הפונקציה צריכה להכיל שימוש ב - <code>querySelectorAll</code>
demo4	צורו פונקציה חדשה. הפונקציה תדע לשמור במשתנה את כל האלמנטים הכוללים את המחלקה [ex1] כמו כן הריצו לולאה על משנה והדפיסו את תוכן האלמנטים. • הפונקציה צריכה להכיל שימוש ב - <code>querySelector</code>

שינוי תוכן אלמנטים



שינוי תוכן האלמנט

בחלק זה נעמיק ונלמד בנושא תוך שימוש במספר כלים ופתרונות קוד. בסיום הנושא נוכל לענות על השאלות הבאות:

- כיצד ניתן לשנות את תוכן האלמנט?
- מה ההבדל בין innerHTML ל innerText?

JS_DOM_inner.html

JS_DOM_inner.js

Method	הסבר
<code>element.innerHTML = "some text demo";</code>	החלף את תוכן האלמנט (התייחסות כ-html)
<code>element.innerText = "some text demo";</code>	החלף את תוכן האלמנט (התייחסות כטקסט)

תרגול שינוי תוכן אלמנטים

יש להיעזר בקובץ `JS_DOM_inner.html` של המרצה ואליז לחבר מסמך `JS_DOM_inner.js`.
מסמך ה-JS יכיל פונקציות בהתאם לסדר הבא:

תרגיל	תיאור המשימה
demo1	צורו פונקציה חדשה. הפונקציה תדע להדפיס בקונסולה את תוכן הטקסט של אלמנט <code>[para1]</code> .
demo2	צורו פונקציה חדשה. הפונקציה תדע להדפיס בקונסולה את תוכן ה-html של אלמנט <code>[para1]</code> .
demo3	צורו פונקציה חדשה, הפונקציה תכניס לאלמנט <code>[para2]</code> באמצעות הפקודה <code>.innerHTML</code> את תוכן <code>["Hola Class from DOM"]</code> .
demo4	צורו פונקציה חדשה, הפונקציה תכניס לאלמנט <code>[para2]</code> באמצעות הפקודה <code>.innerText</code> את תוכן <code>["Hola Class from DOM"]</code> .

שינוי מאפייני אלמנטים



שינוי מאפייני אלמנטים

בחלק זה נעמיק ונלמד בנושא תוך שימוש במספר כלים ופתרונות קוד.
בסיום הנושא נוכל לענות על השאלות הבאות:

- כיצד ניתן לגשת לערכים של מאפייני האלמנט?
- כיצד ניתן לשנות את הערכים של מאפייני האלמנט?
- איזה מאפיינים ניתן לערוך?

JS_DOM_attribute.html

JS_DOM_attribute.js

Method	הסבר
<code>element.attribute = "new value";</code>	החלף את ערך המאפיין של האלמנט
<code>element.setAttribute("attribute", "new value");</code>	החלף את ערך המאפיין של האלמנט

תרגול שינוי מאפייני אלמנטים

יש להיעזר בקובץ `JS_DOM_attributes.html` של המרצה ואילו לחבר מסמך `JS_DOM_attributes.js`.
מסמך ה JS יכיל פונקציות בהתאם לסדר הבא:

תרגיל	תיאור המשימה
<code>demo1get</code>	צרו פונקציה חדשה, הפונקציה תדפיס בקונסולה את כל הערכים של מאפייני האלמנט <code>[demo1]</code> כלומר את המאפיינים: <code>[href, id, target]</code> כמו כן הפונקציה תדפיס את התוכן של האלמנט.
<code>demo1set</code>	צרו פונקציה חדשה, הפונקציה תדע לשנות את הערכים של מאפייני האלמנט <code>[demo1]</code> כלומר: <code>[href, id, target]</code> בסיום הקישור לעביר את המשתמש לאתר <code>hacker.co.il</code> באותה הלשונית. כמו כן הפונקציה תשנה את התוכן של האלמנט בהתאם.
<code>demo2get</code>	צרו פונקציה חדשה, הפונקציה תדפיס בקונסולה את כל הערכים של מאפייני האלמנט <code>[demo2]</code> כלומר את המאפיינים: <code>[type, id, placeholder, value]</code>
<code>demo2set</code>	צרו פונקציה חדשה, הפונקציה תדע לשנות את הערכים של מאפייני האלמנט <code>[demo2]</code> את ערך מאפיין <code>type</code> יש לשנות ל <code>password</code> . את ערך מאפיין <code>value</code> יש לשנות ל <code>"hackeru123"</code> את ערך מאפיין <code>placeholder</code> יש לשנות ל <code>"Enter password"</code>

תרגול שינוי מאפייני אלמנטים

יש להיעזר בקובץ `JS_DOM_attr.html` של המרצה ואילו לחבר מסמך `JS_DOM_attr.js`.
מסמך ה JS יכיל פונקציות בהתאם לסדר הבא:

תרגיל	תיאור המשימה
login	צרו פונקציה חדשה, הפונקציה תבדוק את הסיסמה שהמשתמש הקיש בשדה [userPassword] במידה והמשתמש הקיש hackeru123 תופיע למשתמש הודעה "Welcome" וצבע רקע שדה הקלט ישתנה לירוק. אחרת תופיע למשתמש הודעה "NOT Welcome" וצבע רקע שדה הקלט ישתנה לאדום.
changeType	צרו פונקציה חדשה, הפונקציה תבדוק אם סוג שדה הקלט [userPassword]. במידה וסוג השדה הקלט מוגדר password הפונקציה תשנה את ל-text. אחרת תגדיר את סוג השדה ל password.

עיצוב אלמנטים



JS_DOM_style.html

JS_DOM_style.js

בחלק זה נעמיק ונלמד בנושא תוך שימוש במספר כלים ופתרונות קוד.
בסיום הנושא נוכל לענות על השאלות הבאות:

- כיצד ניתן לעצב את האלמנט?
- מה ההבדל בין `setAttribute` ל `style`?

Method	הסבר
<code>element.setAttribute("attribute", "new value");</code>	החלק את ערך המאפיין של האלמנט
<code>element.style.property = "new value";</code>	החלק ערך של פקודת עיצוב.

JS_DOM_style.html

JS_DOM_style.js

בחלק זה נעמיק ונלמד בנושא תוך שימוש במספר כלים ופתרונות קוד.
בסיום הנושא נוכל לענות על השאלות הבאות:

- כיצד ניתן לעצב את האלמנט?
- מה ההבדל בין `setAttribute` ל `style`?

```
element.style.fontSize = "18px";
```

שימו לב – בשיטה זו אנו משתמשים בערכים של JavaScript, כלומר בצורת camelCase, לדוגמה: `fontSize` | `fontWeight` | `borderLeft` | `border` | `minWidth` | `maxWidth` | `maxHeight` | `margin` | `boxShadow` | `textAlign` | `textDecoration` | `color` | `fontFamily` | `fontSize` | `right` | `left` | `top` | `zIndex`

```
element.setAttribute("style", "font-size: 18px;");
```

שימו לב – בשיטה זו אנו משתמשים בערכים של CSS:

ערכים של CSS יהיו: `font-color`

ערכים של JavaScript יהיו: `fontColor` (כלומר בשיטת ה-camelCase)

תרגול עיצוב אלמנטים

יש להיעזר בקובץ `JS_DOM_style.html` של המרצה ואליו לחבר מסמך `JS_DOM_style.js`.
מסמך ה JS יכיל פונקציות בהתאם לסדר הבא:

תרגיל	תיאור המשימה
demo1	צרו פונקציה חדשה, הפונקציה תבצע את הפעולות הבאות: תשנה את צבע הרקע לאפור תשנה את צבע הטקסט לכחול תשנה את גודל הגופן ל 20 פיקסל. תוסיף צל לטקסט האלמנט. תשנה את יישור הטקסט למרכז.
demo2	צרו פונקציה חדשה, הפונקציה תבצע את הפעולות הבאות: תשנה את צבע הרקע לשחור תשנה את צבע הטקסט לצהוב תשנה את הריווח הפנימי והחיצוני ל 20 פיקסל. תוסיף מסגרת רציפה שחורה בעובי של 1 פיקסל. תוסיף צל לכל האלמנט.
demo3	צרו פונקציה חדשה, הפונקציה תבצע את הפעולות הבאות: הפונקציה תדפיס בקונסולה את צבע הרקע, צבע הטקסט, גודל הגופן.

יצירה והוספת אלמנטים



JS_DOM_create.html

JS_DOM_create.js

בחלק זה נעמיק ונלמד בנושא תוך שימוש במספר כלים ופתרונות קוד.
בסיום הנושא נוכל לענות על השאלות הבאות:

- כיצד ניתן ליצור אלמנט חדש?
- כיצד ניתן להוסיף מאפיינים לאלמנט?
- כיצד להוסיף טקסט לאלמנט?
- כיצד מוסיפים אלמנט חדש למסך?

```
<div id="demo1"></div>

<script>
  var myBox = document.createElement("div");
  myBox.id = "box1";
  myBox.innerHTML = "hola from myBox.. ";
  console.log(myBox);
  document.getElementById("demo1").appendChild(myBox);
</script>
```

Method	הסבר
var myDiv = document.createElement("div");	יצירת אלמנט חדש
element.appendChild(element)	הוסף אלמנט
var txt = document.createTextNode("string text");	יצירת אזור טקסט (שימוש לטקסט)

תרגול יצירה והוספת אלמנטים

יש להיעזר בקובץ `JS_DOM_create.html` של המרצה ואליו לחבר מסמך `JS_DOM_create.js`.
מסמך ה JS יכיל פונקציות בהתאם לסדר הבא:

תרגיל	תיאור המשימה
Demo1	צרו פונקציה חדשה הפונקציה תדע ליצור אלמנט חדש מסוג <code>div</code> . האלמנט יהיה בגובה וברוחב של 100 פיקסל עם צבע רקע כחול ויכיל את הטקסט "HOLA"
Demo2	צרו פונקציה חדשה הפונקציה תדע ליצור אלמנט חדש מסוג <code>H2</code> . האלמנט יהיה עם צבע רקע אפור וגופן בצבע כחול, טקסט ממורכז ויכיל את הטקסט "new title"
Demo3	צרו פונקציה חדשה הפונקציה תדע ליצור אלמנט חדש מסוג <code>a</code> . הוספת האלמנט יצור קישור אשר יכיל את הטקסט "Go To Google". בעת לחיצה על הקישור המשתמש יועבר לאתר גוגל אשר יפתח בלשונית חדשה.
Demo4	צרו פונקציה חדשה הפונקציה תדע ליצור אלמנט חדש מסוג <code>div</code> . האלמנט יהיה בגובה וברוחב של 100 פיקסל עם מסגרת שחורה ורציפה. כל אלמנט יקבל <code>id</code> ייחודי. (זכרו שאסור שלשני אלמנטים יהיה את אותו <code>id</code>). בעת לחיצה על הקישור המשתמש יועבר לאתר גוגל אשר יפתח בלשונית חדשה.

מחיקה והחלפת אלמנטים



מחיקה והחלפת אלמנטים

בחלק זה נעמיק ונלמד בנושא תוך שימוש במספר כלים ופתרונות קוד. בסיום הנושא נוכל לענות על השאלות הבאות:

- כיצד ניתן למחוק אלמנט?
- כיצד ניתן להחליף בין אלמנטים?
- כיצד ניתן נכתוב ישירות למסמך?

JS_DOM_remove.html

JS_DOM_remove.js

```
<div id="demo1">
  Hola <span id="box1" style="color: blue; font-size: 20px;">Class</span>
</div>

<script>

  var x = document.getElementById("box1");
  document.getElementById("demo1").removeChild(x);

</script>
```

Method	הסבר
document. removeChild (element)	הסר אלמנט
element. replaceChild (newObj, oldObj)	החלף אלמנט
<script> document. write ("string text");</script>	כתוב ישירות למסמך HTML

מחיקה והחלפת אלמנטים

יש להיעזר בקובץ `JS_DOM_remove.html` של המרצה ואליו לחבר מסמך `JS_DOM_remove.js`.
מסמך ה JS יכיל פונקציות בהתאם לסדר הבא:

תרגיל	תיאור המשימה
demo1	צרו פונקציה חדשה הפונקציה תסיר את אלמנט box1 מהמסך.
demo2	צרו פונקציה חדשה הפונקציה תסיר את אלמנט container2 מהמסך.
demo3	צרו פונקציה חדשה הפונקציה תסיר את אלמנט הפריטים מהרשימה שאינם כוללים את המילים "Yes" או "No". אתגר: צבעו את כל הפריטים הכוללים את המילה Yes בצבע ירוק ואת No באדום.
demo4	צרו פונקציה חדשה הפונקציה תסיר את הקישור לקובץ ה - css.

שימוש באירועים



JS_DOM_events.html

JS_DOM_events.js

בחלק זה נעמיק ונלמד בנושא תוך שימוש במספר כלים ופתרונות קוד.
בסיום הנושא נוכל לענות על השאלות הבאות:

- כיצד ניתן להאזין אירוע של אלמנט?
- לאיזה אירועים ניתן להאזין?
- אילו ערכים המתודה `addEventListener` מקבלת?

Method	הסבר
<code>document.getElementById(id).onclick = function(){code}</code>	הוספת אירוע לאלמנט
<code>document.getElementById(id).addEventListener("click", function(){ });</code>	הוספת אירוע לאלמנט
<code>document.getElementById(id).addEventListener("click", myFuncName);</code>	הוספת אירוע לאלמנט

JS_DOM_events.html

JS_DOM_events.js

בחלק זה נעמיק ונלמד בנושא תוך שימוש במספר כלים ופתרונות קוד.
בסיום הנושא נוכל לענות על השאלות הבאות:

- כיצד ניתן להוסיף אירוע לאלמנט?
- איזה אירועים ניתן להוסיף?
- אילו ערכים המתודה `addEventListener` מקבלת?

ניתן לתפוס מגוון אירועים:

אירועי עכבר – תנועה | קליק | דאבל קליק ועוד..

אירועי מקלדת – הקשה במקלדת ועוד..

אירועי טופס – פוקוס | שינוי | בחירה | שליחה ועוד..

אירועי מדיה – טעינה | הפעלה | בזמן פעולה ועוד..

אירועים נוספים - אנימציה | מסכי מגע | ועוד..

[onclick](#) | [ondblclick](#) | [onmousedown](#) |
[onkeydown](#) | [onkeypress](#) | [onkeyup](#) |
[onload](#) | [onscroll](#) | [onchange](#) | [onfocus](#) |
[onsubmit](#) | [onplay](#) | [onplaying](#)



יש שאלות

או שאפשר לסכם?

-
-
-
-

○

○



משימות תרגול

בונה מסכים

משימה כוללת יצירת ממשק ליצירת עמודי אינטרנט.
המשימה מתחילה עם הממשק הבא:

JS_DOM_Mission_PageBuilder_1.html

JS_DOM_Mission_PageBuilder_2.html

בונה מסך

לזרם איפסום או בקיצור ליפסום, הוא מלל מקובל וחסר משמעות המשמש

צבע רקע

הוסף

בונה מסכים

משימה כוללת יצירת ממשק ליצירת עמודי אינטרנט.
המשימה מתפתחת וככה נראה הממשק בהמשך:

JS_DOM_Mission_PageBuilder_3.html

JS_DOM_Mission_PageBuilder_4.html

בונה מסך

לזרם איפסום או בקיצור ליפסום, הוא מלל מקובל וחסר משמעות המשמש

סוג האלמנט

רוחב

גובה

תוכן

צבע הגופן

גודל הגופן

סוג הגופן

שמור הוסף

צרו קובץ לטובת הפרויקט, התחילו מיצירת ממשק המשתמש.
עבור כל משימה צרו מסמך חדש.

JS_DOM_Mission_PageBuilder_1.html

JS_DOM_Mission_PageBuilder_2.html

JS_DOM_Mission_PageBuilder_3.html

JS_DOM_Mission_PageBuilder_4.html

תרגיל	תיאור המשימה
Demo1	<p>הוספת אלמנט למסך במסמך ה-JS צרו פונקציה חדשה. הפונקציה תיצור אלמנט חדש ותוסיף אותו למסך. האלמנט מסוג div ברוחב וגובה של 100 פיקסל ומסגרת רציפה שחורה.</p>
demo2	<p>הוספת אפשרות להגדיר צבע רקע לאלמנט: הוסיפו למסמך ה-HTML שדה קלט אשר יאפשר לקבל [צבע רקע] במסמך ה-JS צרו פונקציה חדשה. הפונקציה תיצור אלמנט חדש ותוסיף אותו למסך. האלמנט מסוג div ברוחב וגובה של 100 פיקסל ומסגרת רציפה שחורה. צבע הרקע של האלמנט יהיה בהתאם [צבע רקע]</p>
demo3	<p>הוספת אפשרות להגדיר רוחב וגובה לאלמנט: הוסיפו למסמך ה-HTML 2 שדות קלט אשר יאפשר לקבל [רוחב אלמנט] [גובה אלמנט] במסמך ה-JS צרו פונקציה חדשה. הפונקציה תיצור אלמנט חדש ותוסיף אותו למסך. האלמנט מסוג div ומסגרת רציפה שחורה. ברוחב וגובה של האלמנט יהיו בהתאם [רוחב אלמנט] [גובה אלמנט] האלמנט יהיה בהתאם [צבע רקע] מהמשימה הקודמת</p>

צרו קובץ לטובת הפרויקט, התחילו מיצירת ממשק המשתמש.
עבור כל משימה צרו מסמך חדש.

תרגיל	תיאור המשימה
Demo4	הוספת אפשרות להוסיף תוכן לאלמנט: הוסיפו למסמך ה-HTML שדה קלט [textarea] אשר יאפשר לקבל [תוכן האלמנט] במסמך ה-JS צרו פונקציה חדשה. הפונקציה תיצור אלמנט חדש ותוסיף אותו למסך. האלמנט מסוג div ומסגרת רציפה שחורה. האלמנט יהיה בהתאם [רוחב אלמנט] [גובה אלמנט] [צבע רקע] מהמשימה הקודמת. האלמנט יכיל את התוכן בהתאם ל [תוכן האלמנט] שהתקבל
Demo5	הוספת אפשרות להוסיף צבע וגודל הגופן האלמנט: הוסיפו למסמך ה-HTML 2 שדות קלט אשר יאפשרו לקבל [צבע גופן] [גודל גופן] במסמך ה-JS צרו פונקציה חדשה. הפונקציה תיצור אלמנט חדש ותוסיף אותו למסך. האלמנט מסוג div ומסגרת רציפה שחורה. האלמנט יהיה בהתאם [רוחב אלמנט] [גובה אלמנט] [צבע רקע] [תוכן האלמנט] מהמשימה הקודמת. האלמנט יכיל את התוכן בהתאם ל [צבע גופן] [גודל גופן] שהתקבל

בונה מסכים

צרו קובץ לטובת הפרויקט, התחילו מיצירת ממשק המשתמש.
עבור כל משימה צרו מסמך חדש.

תרגיל	תיאור המשימה
Demo6	הוספת אפשרות להגדיר את עיצוב המסגרת האלמנט [עובי מסגרת][סגנון מסגרת][צבע מסגרת]
Demo7	הוספת אפשרות להגדיר את הריווח הפנימי והריווח החיצוני לאלמנט [ריווח פנימי][ריווח חיצוני]
Demo8	הוספת אפשרות להגדיר פינות מעוגלות לאלמנט [עיגול פינות לאלמנט]
Demo9	הוספת אפשרות להגדיר את סוג האלמנט [סוג אלמנט] בחירה מבין האפשרויות מוגדרות: h1,h2,h3,p,div,span (פקד מסוג select option)
Demo10	הוספת אפשרות להגדיר צל לאלמנט [ציר X] [ציר Y] [צבע צל]

בונה מסכים

צרו קובץ לטובת הפרויקט, התחילו מיצירת ממשק המשתמש.
עבור כל משימה צרו מסמך חדש.

תרגיל	תיאור המשימה
Demo11	הוספת מאפיין title הכולל מידע על האלמנט. אתגר: הוספת מועד מתי נוצר האלמנט.
Demo12	הוספת מאפיין id לאלמנט. חשוב לזכור: ערכו של מאפיין id חייב להיות ייחודי.
Demo13	הוספת אפשרות למחוק את כל הלוח [עיגול פינות לאלמנט]
Demo14	הוספת כפתור שמור (אתגר) המסמך יישמר ב- LocalStorage
Demo15	הוספת שמירה אוטומטית (אתגר)

רשימת קניות

משימה כוללת יצירת ממשק לניהול רשימת קניות.
המשימה מתחילה עם הממשק הבא

רשימת קניות


לרום איפסום או בקיצור ליפסום, הוא מלל מקובל
וחסר משמעות המשמש "מחלא מקום" בעת עריכ

שם הפריט

הוספה

מערכת התחברות

משימה כוללת יצירת ממשק התחברות למערכת.
המשימה מתחילה עם הממשק הבא



מערכת התחברות

לזרם אינטרנט או בקיצור ליפסום, הוא מלל מקובל
וחסר משמעות המשמש "ממלא מקום" בעת עריכת


שם משתמש

סיסמה

התחברות

מערכת הרשמה

משימה כוללת יצירת ממשק הרשמה לאתר.
המשימה מתחילה עם הממשק הבא



מערכת הרשמה

לורם איפסום או בקיצור ליפסום, הוא מלל מקובל
לחסר משמעות המשמש "ממלא מקום" בעת שריד

שם פרטי

שם משפחה

שם משתמש

סיסמה

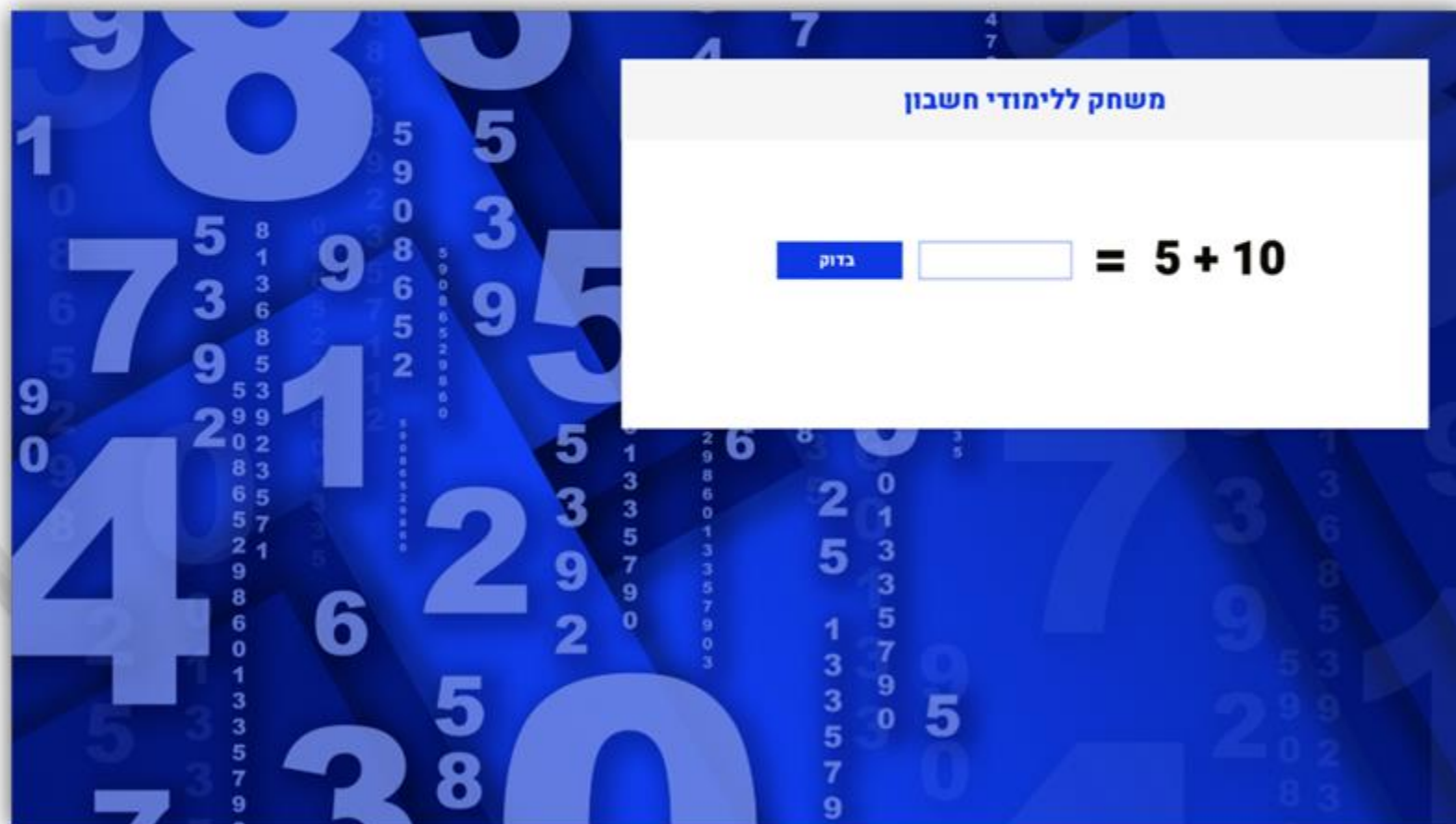
הרשמה

לומדת חשבון

משימה כוללת יצירת משחק ללימוד חשבון

JS_DOM_Mission_Math_Game.html

JS_DOM_Mission_Math_Game.js



לומדת חשבון

משימה כוללת יצירת משחק ללימוד חשבון

JS_DOM_Mission_Math_Game.html

JS_DOM_Mission_Math_Game.js

משחק ללימודי חשבון

= 5 + 10

תוצר	תשובה נכונה	תשובה שגויה	נקודות
5 + 7	12	12	10
9 + 8	17	15	0
3 + 2	5	5	10

לומדת חשבון

משימה כוללת יצירת משחק ללימוד חשבון

JS_DOM_Mission_Math_Game.html

JS_DOM_Mission_Math_Game.js

משחק ללימודי חשבון

אפשרויות משחק

בחר טווח מספרים

1 - 100

1 - 10

1 - 1000

1 - 10000

בדוק = 35 + 5

נקודות	תשובה שגויה	תשובה נכונה	תרגיל
10	12	12	5 + 7
0	15	17	9 + 8
10	5	5	3 + 2

לומדת חשבון

משימה כוללת יצירת משחק ללימוד חשבון

JS_DOM_Mission_Math_Game.html

JS_DOM_Mission_Math_Game.js

משחק ללימודי חשבון

אפשרויות משחק

בחר אופרטור

▼ פלוס +

פלוס +

מינוס -

כפל *

חלק /

בחר טווח מספרים

▼ 1 - 100

1 - 10

1 - 100

1 - 1000

בדוק

= 2 * 105

תוצר	תשובה נכונה	תשובה שגויה	נקודות
5 + 7	12	12	10
9 + 8	17	15	0
3 + 2	5	5	10

משחק זיכרון

משימה כוללת יצירת משחק קלפים לשיפור הזכרון

משחק זיכרון

המכו 2 קלפים ובידקו האם הם זהים

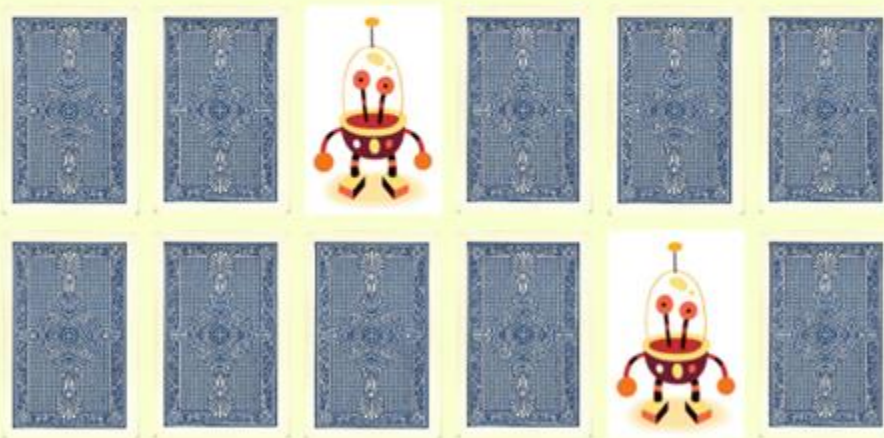


משחק זיכרון

משימה כוללת יצירת משחק קלפים לשיפור הזכרון

משחק זיכרון

המכו 2 קלפים ובידקו האם הם זהים



משחק זיכרון

משימה כוללת יצירת משחק קלפים לשיפור הזיכרון

משחק זיכרון

הפכו 2 קלפים ובידקו האם הם זהים



משחק זיכרון

משימה כוללת יצירת משחק קלפים לשיפור הזיכרון

משחק זיכרון

הפכו 2 קלפים ובידקו האם הם זהים



משחק זיכרון

משימה כוללת יצירת משחק קלפים לשיפור הזיכרון

משחק זיכרון

הסכו 2 קלפים ובידקו האם הם זהים

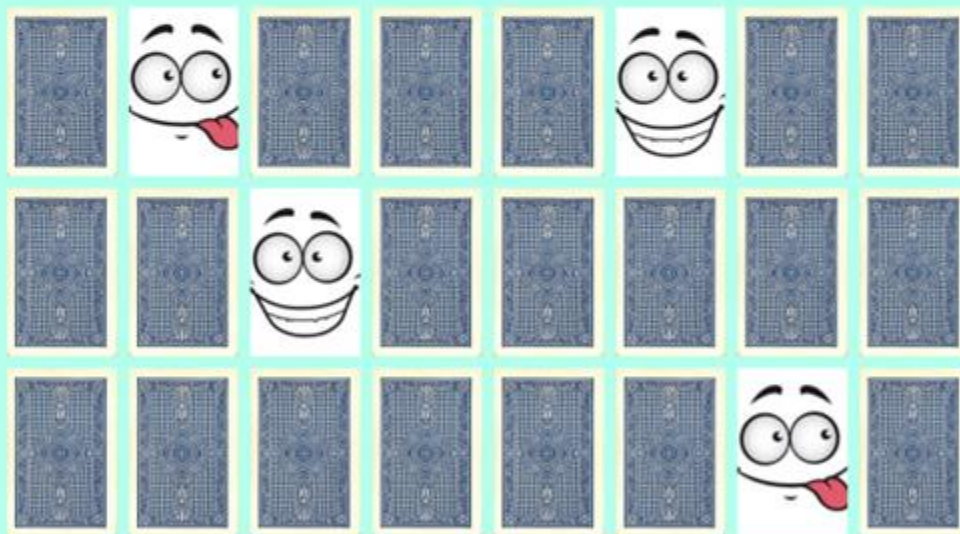


משחק זיכרון

משימה כוללת יצירת משחק קלפים לשיפור הזיכרון

משחק זיכרון

הפכו 2 קלפים ובידקו האם הם זהים



משחק זיכרון

משימה כוללת יצירת משחק קלפים לשיפור הזיכרון

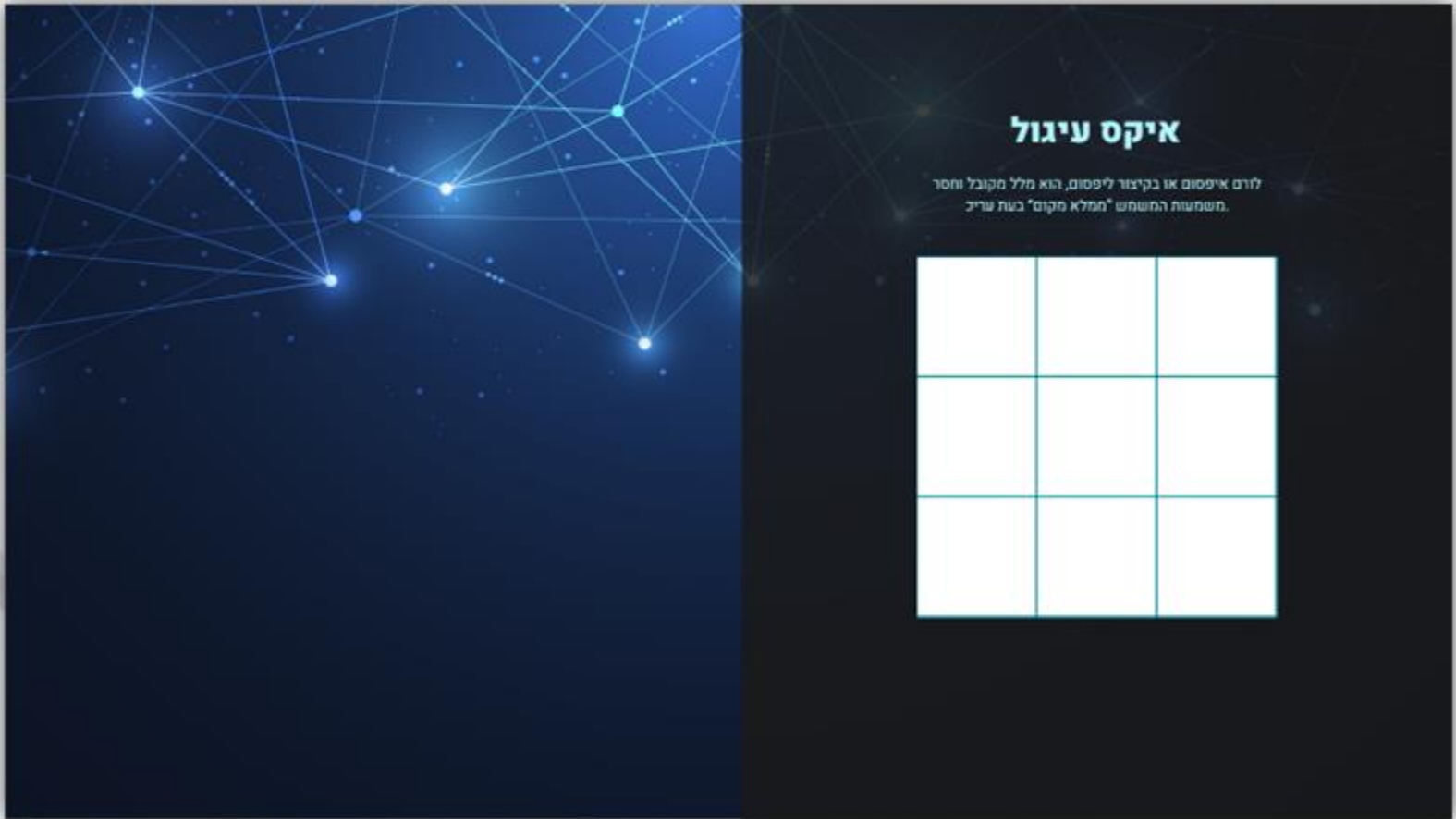
משחק זיכרון

הפכו 2 קלפים ובידקו האם הם זהים



איקס עיגול

משימה כוללת יצירת ממשק למשחק איקס עיגול המוכר.
המשימה מתחילה עם הממשק הבא





זמן שאלות



בדיקת רמת ידע



סוף

