**DOCS – JavaScript 2022-12-05**

**Hoisting**

פעולה ש Javascript, מבצעת עם משתנים שהוגדרו ע"י var ופונקציות. פעולה זו לוקחת את הגדרת המשתנה או הפונקציה ומעבירה אותם הכי גבוה שניתן ב scope שלהם.

**Var**

מילה היוצרת משתנה, מתבצע עבורו hoisting (מונף לראש ה scope - פונקציה או script) , ניתן להגדיר מספר משתנים בעלי אותו שם. השני דורס את הראשון.

**let (block scop )**

מילה היוצרת משתנה, שניתן להשתמש בה מאותה השורה בה המשתנה הוגדר ועד לסוף הבלוק בו הוגדר המשתנה. אם הוא גלובלי (ישירות בתוך ה script), הוא ניתן לשימוש מהשורה בה הוגדר עד סוף ה script

לא ניתן לפתוח באותו ה scope שני משתנים בעלי אותו שם.

ניתן לפתוח משתנה ולא לאתחל אותו, אלא להכניס ערך בהמשך.

**const**

מילה היוצרת משתנה בעל scope זהה לlet, לא מתבצע עבורו hoisting

חובה לאתחל אותו.

אין אפשרות לשנות את הערך לאחר מכן.

מתי להשתמש עם var/let/const

var – לא להשתמש לעולם.

let - כאשר אנו מגדירים משתנה ורוצים לשנות את ערכו לאחר מכן או לאתחל אותו לאחר מכן.

const – כאשר אנו מגדירים משתנה שום צורך לשנות אותו לאחר מכן.

**DOM**

Document Object Model

זה הוא שם המתאר את כלל האובייקטים שהדפדפן יותר מכלל תגיות ה HTML.

כאשר הדפדפן טוען קובץ HTML, הוא יוצר בזכרון הפנימי שלו אובייקט מכל תגית. כל אובייקט מתאר תגית ומכיל את כל מה שיש בתגית.

יצרתם לחצן, הדפדפן יוצר בזכרון אובייקט המייצג את הלחצן הזה.

יצרתם פיסקה, הדפדפן יוצר בזכרון אובייקט המייצג את הפסקה.

אובייקטיי ה DOM, נוצרים באותה ההיררכיה של תגיות ה HTML (אובייקט בתוך אובייקט)

קוד JAVASCRIPT יכול לגשת לאובייקטים הללו ולשנות אותם. ברגע שנשנה אובייקט , הדפדפן ירענן את התצוגה ויציג את השינוי שבצענו. כך javascript יכולה לבצע שינויים דינמיים בתוך הדף.

נוהל קריאת ערך מתוך ה HTML ל Javascript:

1. מתן Id לתגית ה HTML
2. יצירת משתנה const בעל שם ה id ב javascript שמקבל לתוכו את האובייקט ה DOM שמייצג את התגית ע"י document.getElementById(id)
3. ע"י האובייקט שכעת יש לנו גישה אליו, אנו יכולים לגשת למאפיינים (שדות) ; אנחנו יכולים לקרוא ממנו או לכתוב אליו.

HTML

<span id="myCoolSpan"> Some Text </span>

JS

const myCoolSpan = document.getElementById("myCoolSpan ");

myCoolSpan.innerText = "Something else…"

ולידציה – Validation

אימות קלט מהמשתמש.

כאשר אנו קולטים מידע מהמשתמש, יכול להיות שיש לנו חוקיות עבור המידע הזה. וולידציה זו בדיקת המידע – האם הוא חוקי.

אם לא חוקי, הצגת הודעת שגיאה מתאימה.

יש שתי שיטות לביצוע וולידציה :

1. ע"י ה html attributes, בשילוב עם form (הודעות שגיאה מוצגות אוטומטית)
2. ע"י קוד Javascript – הודעות השגיאה מוצגות – מתי שאנחנו מחליטים להציג אותן.

Regular Expression (בקיצור RegEx)

זו שפה המבצעת בדיקה או חיפוש מידע במחרוזת (נלמד אותה בהמשך).