**Ecma Script (ES)6**

**קצת היסטוריה**

**בשנות ה-60 המוקדמות הבינו שיש צורך לבנות סטנדרטים עבור עבודה עם מחשבים על מנת שתוכנות שכתובות במחשב אחד יוכלו לעבוד גם במחשבים אחרים והיצרנים לא יבזבזו את זמנם בעבודה כפולה ויפתחו שפות דומות למחשבים.**

**ראשי חברות מובילות באירופה החליטו להקים ארגון שיהיה אחראי על כתיבת וארגון סט חוקים עבור יצרני המחשבים. לארגון הם קראוEuropean Computer Manufacturers Association  או בקיצור ECMA.**

**אז מה זה EcmaScript?**

EcmaScript **(**ES**)** **היא אוסף סטנדרטים לכתיבת שפת סקריפט.**

ES **נוצרה בשנת 1997 לאחר יצירת שפת** JavaScript **(שנוצרה בשנת 1996)**

**יוצרי** JS **הגישו את השפה ל**ECMA **על מנת שיבחנו אותה ויגדירו תקנות לכיצד אמורה להראות שפת סקריפט. לאחר הגשת** JS **למוסד נוצרה** EcmaScript **שבעצם הכילה מפרט קריטריונים ליצירת שפת סקריפט.**

JS **נשארה עדיין השפה הכי מוכרת המממשת את המפרט של** ES **מאז שנוצר.**

**עד היום משתמשים בעיקר ב**ES5 **שזה בגדול כל מה שאתם מכירים עד עכשיו ב**JavaScript.

**ES6 –** המהדורה השישית של **ECMAScript**

**יצאה ב2015 ועשתה שינוי גדול בדרך בה אנחנו משתמשים ב**javaScript

**ל**ES6 **יכולות מדהימות המשפרות את עיצוב הקוד, קריאות הקוד והיעילות שלו .**

**חלק מהחומר שלמדנו עד כה במקצוע גם הוא חלק מ**ES6**:**

* let and const **– סוגי טיפוסים שנוצרו על מנת לייעל את העבודה עם משתנים ולעזור בהפרדה בין טווחי הכרה שונים לעומת** var **שהיה בשימוש נרחב יותר בעבר.**
* for … of **– סוג לולאה שמאפשר לנו לעבור על כל האיברים במערך בצורה נוחה בלי להיות תלויים באינדקס, עד אז היה רק** for... in **שחייב אותנו לגשת למקום במערך לפי אינדקס.**
* find **– פונקציה שמאפשרת למצוא איבר ספציפי במערך, עד** ES6 **היינו צריכים למצוא איבר ספציפי באמצעות** filter**.**

**עכשיו שהבנו בגדול מה זה ES6 בואו נראה מה באמת הוא נותן לנו**

**את רוב הלמידה אתם תעשו בעזרת** [**סרטונים אינטראקטיביים**](https://scrimba.com/g/gintrotoes6?utm_source=freecodecamp.org&utm_medium=referral&utm_campaign=gintrotoes6_launch_article) **בפלטפורמה הנקראת Scrimba,   
להלן סדר צפייה מומלץ:**

* 1. Let + const **(שיעור 18)**
  2. Arrow Functions **(שיעור 13)**
  3. Export/Import **(שיעור 19)** + יש לקרוא בנוסף גם על [Export](https://developer.mozilla.org/en-US/docs/Web/JavaScript/Reference/Statements/export) ו[Import](https://developer.mozilla.org/en-US/docs/Web/JavaScript/Reference/Statements/import)
  4. Spread Operator **(שיעור 10)**
  5. Classes **(שיעור 23)** - פחות נשתמש בצד לקוח, תכירו בהמשך שימושים נוספים
  6. Destructuring **(שיעורים 3 ו-5)**
  7. Template Literals **(שיעור 2)**
  8. [Property Shorthand](https://alligator.io/js/object-property-shorthand-es6/) - **נמצא באתר שונה, לחצו על הקישור**
  9. [Computed property name](https://tylermcginnis.com/computed-property-names/) – **נמצא באתר שונה, לחצו על הקישור**
  10. Default Parameter מקורות מידע נוספים:
* [**ההבדל בין** JS **ל**ES](https://www.freecodecamp.org/news/whats-the-difference-between-javascript-and-ecmascript-cba48c73a2b5/)
* [כל ההבדלים בין ES5 לES6 כולל דוגמאות קוד](http://es6-features.org/#PropertyShorthand)
* [דוקומנטציה מפורטת מאוד לכל הסטנדרטים הקיימים בES6](https://github.com/metagrover/ES6-for-humans)

**Babel**

בעקבות החידושים המתמידים בES נוצרה בעיה –

לא כל הדפדפנים תמכו בגרסה זו של ECMA ונוצר הצורך בתרגום הקוד מגרסאות מתקדמות של ES לגרסאות ישנות יותר – על מנת שהאתרים יהיו מותאמים לכלל הדפדפנים ויוכלו לפעול אצל כל המשתמשים בצורה טובה.

לשם כך יצר סבסטיאן מקנזי, מהנדס תוכנה בפייסבוק, קומפיילר המתרגם קוד שנכתב על פי הסטנדרטים של ES6 ומעלה, לסטנדרטים הישנים יותר של ES5 על מנת שמתכנתים יוכלו להמיר את הקוד שלהם בקלות כך שיתאים לכל הדפדפנים. הקומפיילר הוא open-source לחלוטין והקוד שלו פתוח לציבור הרחב בgitHub, נכון להיום יש לקומפיילר קרוב ל1000 תורמים, 650 גרסאות ומעל ל13,000 קומיטים!

יכולים לנסות להשתמש בקומפיילר בעצמכם [כאן](https://babeljs.io/)

שפת JavaScript היא שפה שמתפתחת בצורה משמעותית בשנים האחרונות וכמעט כל שנה יוצאת גרסה חדשה כמו – ES7, ES8 אשר כבר נתמכות בחלק מהפלטפורמות. מי שמתעניין מוזמן לקרוא [כאן](https://www.cronj.com/blog/javascript-es7-es8-new-features/) או לחפש מידע מעבר לכך באינטרנט.