=== - משווה גם בין הערכים וגם בין ה Types .

== - משווה רק בין הערכים ולא מתייחס לTypes.

<https://www.geeksforgeeks.org/difference-between-and-operator-in-javascript/>

var – יצירת משתנה בגירסא הישנה

Let - יצירת משתנה בגירסא החדשה

למה הפסיקו להשתמש ב var ?

מה ההבדלים ביניהם?

מה הם הדרכים לייצר פונקציה:

• שאתה מייצר פונקציה בתוך אובייקט אתה לא צריך להצהיר שזה פונקציה, מספיק לרשום ככה:

const alienShip = {

invade () {

let moshe = 1;

return moshe;

}

};

שבעת סוגי Data Type שיש ב Javascript:

String –

Number –

Boolean –   
null –

Undefined –

Symbol –

Object –

1. To create object you just need to create a normal variable but with {}. Like ->

let spaceship = {}; // this is an empty object;

1. Example with more one variables inside:

let spaceship = {

'Fuel Type': 'diesel', // Here we write ' ' in the Fuel Type because it's contain space.

color: 'silver'

};

1. When you create an object with const like this: const spaceship = {type: 'shuttle'};

You cant change the inside variable by spaceship assign:

spaceship = {type: 'haim'}; // const error

but you can change the inside variables like this:

Spaceship.type = 'haim'; // That will work.

1. You can create an a function inside an Object (there is many option to do that, this is one example):

const alienShip = {

invade () {

let moshe = 1;

return moshe;

}

};

1. You should need to know, you can create a object inside an object it’s called Nested Objects
2. ניתן לשנות את האובייקט כולו על ידי השמה מחדש של האובייקט וגם ניתן משתנה של האובייקט על ידי נקודה.

באובייקטים שעוברים לפונקציות (By Reference) ובתוך הפונקציה עושים השמה מחדש של האובייקט , זה לא ניתן.

String או Number לפעמיים יקראו גם אובייקט, במקרים הספציפיים האלה:

classes

הם כמו אובייקטים רק יותר משודרגים

פשוט עושים constructor ושם מאפסים את הערכים וגם יש את האופציה לירושה

כשאנחנו בConstructor של הילד אנחנו נשתמש בsuper() ונשלח את הערכים שקיבלנו לconstructor של האבא

דוגמת קוד : <https://gist.github.com/haimhugi/3c9e79c2a2658e47f800bf4d01ab5763>