

NODE 01 - START

מה זה NODE

Node.js® is an open-source, cross-platform JavaScript runtime environment

איך נבדוק אם מותקן אצלינו במערכת?
נריץ בטרמינל

node -v

ההבדלים בין NODEJS וVANILLAJJS

1. סרבר קליינט - NODE רץ על הסרבר, בעוד VANILLA רץ על הקליינט
2. מהסיבה לעיל CONSOLE רץ על הטרמינל במקום בדפדפן
נריץ node על הטרמינל על ידי כתיבת שורת קוד

node

על מנת לסיים את הפקודה Ctrl+c

אפשר להריץ כל קובץ JS על ידי שורת הפקודה

node [FILENAME]

לדוגמא

node server

יריץ את קובץ server.js

3. קיים אובייקט global במקום אובייקט window
4. עבודה עם קבצים
5. עבודה עם commonJS במקום es6 שזה אומר require במקום ...import from

```
const os = require('os')
const path = require('path')

console.log(os.type())
console.log(os.version())
console.log(os.homedir())

console.log(__dirname)
console.log(__filename)

console.log(path.dirname(__filename))
console.log(path.basename(__filename))
```

```
console.log(path.extname(__filename))

console.log(path.parse(__filename))
```

והתוצאה

```
Windows_NT
Windows 10 Home
C:\Users\geula
D:\דומיל\node-mongo\ex01
Windows 10 Home
C:\Users\geula
D:\דומיל\node-mongo\ex01
D:\דומיל\node-mongo\ex01\server.js
D:\דומיל\node-mongo\ex01
server.js
.js
{
  root: 'D:\\',
  dir: 'D:\\דומיל\\node-mongo\\ex01',
  base: 'server.js',
  ext: '.js',
  name: 'server'
}
```

ניצור קובץ math.js

ונכניס בו את 4 הפונקציות הבאות

```
const add = (a,b)=>a+b
const subtract = (a,b)=>a-b
const multiply = (a,b)=>a*b
const divide = (a,b)=>a/b
```

על מנת להשתמש בפונקציות האלו בקובץ server צריך לבצע 2 פעולות

א. לייצא את הפונקציות בסוף הקובץ כאובייקט (הדרך המקובלת - והמועדפת)

```
const add = (a,b)=>a+b
const subtract = (a,b)=>a-b
const multiply = (a,b)=>a*b
const divide = (a,b)=>a/b

module.exports = {add,subtract, multiply, divide}
```

או לייצא ישירות את הפונקציות ישירות

```
exports.add = (a,b)=>a+b
exports.subtract = (a,b)=>a-b
exports.multiply = (a,b)=>a*b
```

```
exports.divide = (a,b)=>a/b
```

ב. לייבא את האובייקט המלא או לייבא פונקציות אינדבדואליות מתוך האובייקט

```
const math = require('./math')
console.log(math.add(2,3))
```

אינדבדואלי הדרך המועדפת

```
const {add, subtract, multiply, divide} = require('./math')
console.log(add(2,3))
```

6. חסרים מעט API שקיימים לשימוש בVANILLAJJS כגון fetch

משימה EX1

צרו שני קבצים - קובץ של עבודה בסיסית עם מערך בשם arr.js

```
const arr = [
  {id:1, name:"avi", count:4},
  {id:2, name:"yohai", count:18},
  {id:3, name:"moshe", count:5},
  {id:4, name:"meir", count:1},
]

const remove=(name) =>{} //using filter
const add=(item) =>{}
const update=(name)=>{} //add count using map
```

קובץ שני בשם server.js בו ניתן להריץ את הפונקציות והתוצאה מוצגת בטרנמינל

```
const {remove, add, update} = require('./array')
console.log(update("yohai"))
```

והתוצאה

```
[
  { id: 1, name: 'avi', count: 4 },
  { id: 2, name: 'yohai', count: 19 },
  { id: 3, name: 'moshe', count: 5 },
  { id: 4, name: 'meir', count: 1 }
]
```