

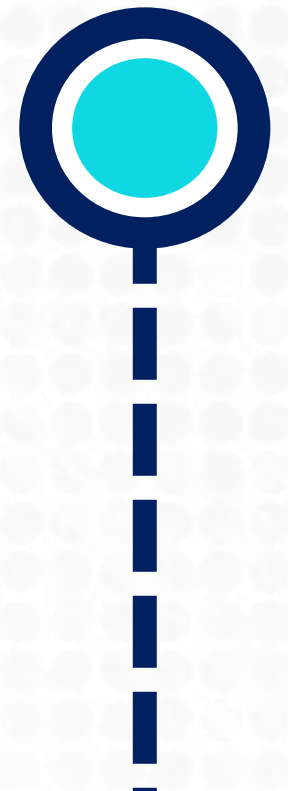
Node.js

מבוא והתוכנית הראשונה שלי

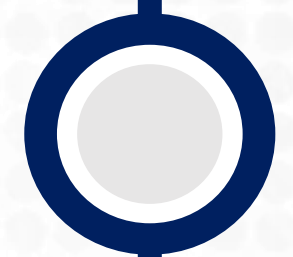
27/11/2024

Node.js with Express

במסגרת הנושא נכיר ונעמיק ידע ב- Node.js ו-Express.js, ונלמד על בניית יישומי צד שרת מודרניים, ניהול ניתוב, טיפול בנתונים, ואינטגרציה עם מסדי נתונים.



מבוא והתקנות



Node.js – מבוא והתקנות

בחלק זה נכיר את **Node.js** ונדון בהתקנתו. בסיום הנושא תוכלו לענות על השאלות הבאות:

[Node.js — Download Node.js®](#)

CLI / Commands / npm init

npm-init

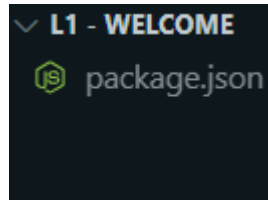
create a package.json file

```
{
  "name": "l1---welcome",
  "version": "1.0.0",
  "description": "Node.js with Express - Welcome",
  "main": "index.js",
  "scripts": {
    "test": "echo \"Error: no test specified\" && exit 1"
  },
  "author": "yehonatan Moravia",
  "license": "MIT"
}
```

- מה זה **Node.js**?
- מתי ארצה להשתמש ב-**Node.js**?
- מהם היתרונות והחסרונות מול טכנולוגיות מקבילות?
- מהם העקרונות הבסיסיים של **Node.js**?
- כיצד מורידים ומתקינים את **Node.js**?
- מה זה **npm**?
- כיצד נוכל ליצור תוכנית **Node.js** חדשה?
- מה עלינו להגדיר במהלך דיאלוג היצירה בעת הרצת הפקודה **npm init**?
- כיצד נוכל להריץ את התוכנית שיצרנו?

Node.js – התוכנית הראשונה שלי

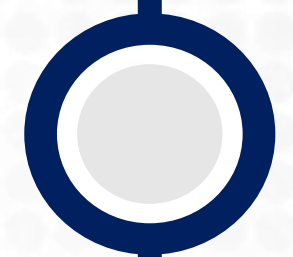
בחלק זה ניצור ונריץ את תוכנית ה-**Node.js** הראשונה שלנו. בסיום הנושא תוכלו לענות על השאלות הבאות:



```
console.log("Hello Node.JS!!!");
```

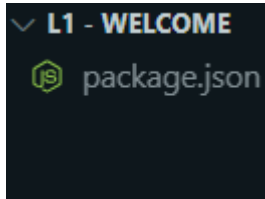
- מה נראה בתיקיית הקבצים לאחר דיאלוג היצירה?
- מדוע שנרצה ליצור קובץ בשם `index.js`?
- איזה קוד נוכל לכלול בקובץ זה?
- כיצד נוכל להריצו על ידי שימוש ב-`Node`?
- כיצד נוכל להגדיר פקודה שתריץ את הקובץ?
- באיזה קובץ עלינו לעשות זאת?

Package.json



Node.js – הגדרות בקובץ package.json

בחלק זה נכיר ונלמד לעבוד מול קובץ ההגדרות **package.json**. בסיום הנושא תוכלו לענות על השאלות הבאות:



```
console.log("Hello Node.JS!!!");
```

- לשם מה משמש הקובץ **package.json**?
- מה עלינו לשים בשדה **"main"**?
- מה נוכל לשים בשדה **"scripts"**?
- לשם מה נשתמש בפקודה **"echo"**?
- לשם מה נשתמש בפקודה **"exit 1"**?
- לשם מה נשתמש באופרטור **"&&"**?
- כיצד נוכל להריץ **script** ספציפי מתוך השדה **"scripts"**?
- כיצד נוכל להריץ את הקובץ שיצרנו (**index.js**) באמצעות הפקודה **"npm start"**?
- מה המשמעות של נקודה (.) ואיך נוכל להשתמש בה על מנת לקצר פקודת הרצה?

package.json - תרגול

צרו תוכנית Node.js חדשה עבור הנושא ופתרו את התרגילים לפי הסדר.

```
> l1---welcome@1.0.0 test
> echo "This is a test" && node .

"This is a test"
Random number: 5344 is even
```

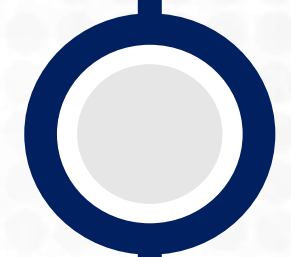
```
> l1---welcome@1.0.0 start
> node .

Random number: 6807 is odd
```

תרגיל	תיאור המשימה
Ex-1	צרו קובץ חדש בשם index.js.
Ex-2	הגרילו מספר אקראי בן 4 ספרות, הדפיסו לקונסול את המספר והאם הוא זוגי או לא.
Ex-3	הריצו את הקובץ באמצעות Node.
Ex-4	עירכו את שורת ה-test כך שתודפס ההודעה "This is a test." ולאחר מכן ירוץ הקובץ index.js והריצו באמצעות "npm run test".
Ex-5	טפלו ב-"package.json" כך שהפקודה "npm run start/npm start" תריץ את הקובץ שיצרתם, השתמשו בנקודה (.).

Javascript בצד השרת

ES Modules-1



JavaScript בצד השרת ו-ECMAScript Modules

בחלק זה נכיר ונלמד לעבוד עם שפת **JavaScript** בצד השרת ונלמד לעבוד עם מודולים. בסיום הנושא תוכלו לענות על השאלות הבאות:

```
const { sizes } = require('./sizes');
```

```
import { sizes } from './sizes.js';
```

```
exports.sizes = sizes;
```

```
export { sizes };
```

- מה זה מודול?
- כיצד ניתן לייצא משתנה מתוך מודול באמצעות "exports"?
- כיצד ניתן לייבא משתנה מודול באמצעות "require"?
- מדוע אין ביכולתנו להשתמש ב-"import/export" כמו שעשינו ב-React?
- מה זה "ECMAScript Modules"?
- במה זה שונה מההגדרה הדיפולטיבית?
- כיצד נוכל להגדיר שימוש ב-"ECMAScript Modules" בתוכנית ב-Node.js?
- מה עלינו להגדיר בשדה "type" בקובץ "package.json"?
- כיצד נוכל לייצא\לייבא משתנה ממודול לאחר מכן?
- מדוע חשובה לנו הסיומת של הקובץ שאנחנו מייבאים מתוכו?

JavaScript בצד השרת ו-ECMAScript Modules - תרגול

צרו תוכנית **Node.js** חדשה עבור הנושא ופתרו את התרגילים לפי הסדר.

```
const sizes = {
  "0-6": "newborn",
  "6-12": "infant",
  "12-18": "small",
  "18-24": "medium",
  "24-36": "large",
  "36-48": "x-large"
};
```

size: 48 - x-large

size: 22 - medium

תרגיל	תיאור המשימה
Ex-1	צרו קובץ חדש בשם sizes.js אשר מכיל מידות לבגדים כמפורט בנספח וייצאו אותה (ללא שימוש ב-ES Modules).
Ex-2	צרו את הקובץ "index.js" וייבאו אליו את רשימת המידות.
Ex-3	צרו משתנה בשם randomSize אשר יכיל מספר רנדומלי בין 0-48 .
Ex-4	הדפיסו הודעה לקונסול המתארת את המידה לפי הדוגמה בנספח.
Ex-5	חיזרו על התרגיל באמצעות שימוש ב-ES Modules.

Express.js



Express.js – מבוא והתקנות

בחלק זה נכיר את **Express.js** ונדון בהתקנתה. בסיום הנושא תוכלו לענות על השאלות הבאות:

```
import express from 'express';

const app = express();

const PORT = 8080;

app.listen(PORT, () => {
  console.log(`Server is running on port ${PORT}`);
});
```

- מה זה **Express.js**?
- במה היא מקלה עלינו?
- כיצד נוכל להתקין אותה על גבי התוכנית שלנו?
- כיצד נוכל להשתמש ב-**express** בקובץ ההרצה הראשי של התוכנית?
- מה זה **PORT**?
- לשם מה נשתמש במתודה **"listen()"**?

nodemon

בחלק זה נכיר את **nodemon** ונדון בהתקנתה. בסיום הנושא תוכלו לענות על השאלות הבאות:

To install nodemon globally:

```
npm install -g nodemon
```

To install nodemon as a dev dependency:

```
npm install --save-dev nodemon
```

```
{
  "name": "express-welcome",
  "version": "1.0.0",
  "description": "",
  "main": "server.js",
  "type": "module",
  > Debug
  "scripts": {
    "test": "echo \"Error: no test specified\" && exit 1",
    "start": "node .",
    "dev": "nodemon ."
  },
  "author": "Yehonatan Moravia",
  "license": "MIT",
  "dependencies": {
    "express": "^4.21.1"
  }
}
```

- מה היא **nodemon**?
- לשם מה נשתמש בה?
- כיצד נוכל להתקין אותה על גבי הפרויקט?
- מה ההבדל בין התקנה לוקלית לגלובלית ומה הן הפקודות הרלוונטיות?
- מה עלינו להגדיר בפקודת ההרצה על מנת להשתמש בה?
- מדוע זה רלוונטי לסביבת הפיתוח בלבד?
- אם כן, מדוע כדאי לנו ליצור פקודת הרצה ספציפית עבור סביבת הפיתוח?
- כיצד נוכל להריץ את התוכנית באמצעות שימוש בפקודת ההרצה החדשה שיצרנו?

המשיכו את התרגיל הקודם ופתרו את התרגילים לפי הסדר.

```
> express-welcome@1.0.0 dev
> nodemon .

[nodemon] 3.1.7
[nodemon] to restart at any time, enter `rs`
[nodemon] watching path(s): *.*
[nodemon] watching extensions: js,mjs,cjs,json
[nodemon] starting `node .`
Server is running on port 8080
[nodemon] restarting due to changes...
[nodemon] restarting due to changes...
[nodemon] starting `node .`
Hello World! Server is running on port 8080
█
```

תרגיל	תיאור המשימה
Ex-1	התקינו את nodemon על גבי הפרויקט שלכם.
Ex-2	צרו פקודת הרצה חדשה בשם "dev" שתריץ את התוכנית באמצעות שימוש ב-nodemon.
Ex-3	הריצו את התוכנית באמצעות הפקודה שיצרתם.
Ex-4	שנו את ההודעה המוצגת ושימרו את המסמך. בידקו שההודעה החדשה אכן מוצגת ושהשרת הופעל מחדש לאחר השמירה.

צרו תוכנית Node.js חדשה עבור הנושא ופתרו את התרגילים לפי הסדר.

תרגיל	תיאור המשימה
Ex-1	התקינו את nodemon על גבי הפרויקט שלכם.
Ex-2	צרו פקודת הרצה חדשה בשם "dev" שתריץ את התוכנית באמצעות שימוש ב-nodemon.
Ex-3	הריצו את התוכנית באמצעות הפקודה שיצרתם.
Ex-4	הדפיסו לקונסול רצף של מספרים רנדומליים זוגיים בלבד בין 100 ל-1000.

תודה על ההקשבה

אני וצוות המכללה כאן עבורכם