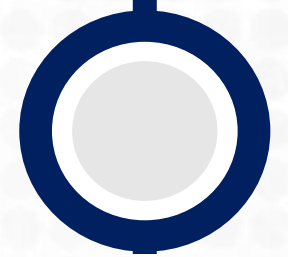


Node.Js

חיבור ל-MongoDb

11/12/2024

חיבור למסד נתונים MongoDB



חיבור ל-MongoDB באמצעות mongoose

בחלק זה נכיר ונלמד כיצד ניתן לבצע פעולות במסד הנתונים באמצעות שימוש בקריאות API ו-mongoose. בסיום הנושא תוכלו לענות על השאלות הבאות:



elegant mongodb object modeling for node.js

- מה היא mongoose?
- כיצד נוכל לחבר את התוכנית שלנו למסד הנתונים באמצעות mongoose?
- מה היא סכמת נתונים?
- כיצד נוכל ליצור סכמה?
- כיצד נוכל להוסיף לקרוא ולמחוק לעדכן נתונים באמצעות שימוש בנתיב בשרת?
- מדוע על הפניות למסד הנתונים להיות אסינכרוניות?

```
router.get("/:id", async (req, res) => {
  try {
    const { id } = req.params;
    const user = await User.findById(id);

    if (!user) {
      return res.status(404).json({ message: "User not found" });
    }

    return res.json(user);
  } catch (error) {
    return res.status(500).json({ message: "Internal server error" });
  }
});
```

חיבור ל-MongoDb באמצעות mongoose

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

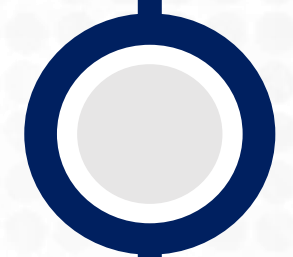
```
import { Schema, model } from "mongoose";

const UserSchema = new Schema({
  username: {
    type: String,
    required: true,
  },
  password: {
    type: String,
    required: true,
  },
  email: {
    type: String,
    required: true,
    unique: true,
  },
  authLevel: {
    type: Number,
    default: 1,
  },
});

export default model("users", UserSchema);
```

תרגיל	תיאור המשימה
Ex-1	צרו תיקייה בשם users, צרו בה תיקיות עבור מודלים ונתיבים.
Ex-2	צרו סכמה עבור משתמשים בהתאם לדוגמה בספח.
Ex-3	חברו את mongoDB לתוכנית באמצעות שימוש ב-connect.
Ex-4	צרו נתיבים עבור הוספת וקריאת משתמשים.
Ex-5	טפלו בנתיבי ההוספה והקריאה כך שיבצעו את הפעולה האמורה במסד הנתונים.

Data Seed



Data Seed

בחלק זה נכיר ונלמד מה היא "זריעת נתונים" ומתי נרצה לעשות זאת.
בסיום הנושא תוכלו לענות על השאלות הבאות:

```
Server is running on port 8080
Connected to the database
User created: john@email
User created: jane@email
User created: jim@email
```

- מה זה "זריעת נתונים" (Data Seed)?
- כיצד נוכל ליצור מסמכים מראש בעת הרצת התוכנית במידה והם לא קיימים עדיין?
- מה זה נותן לנו?
- מדוע כדאי לנו להחזיק נתונים אלו בקובץ JSON?

```
[
  {
    "username": "John Doe",
    "email": "john@email",
    "password": "password"
  },
  {
    "username": "Jane Melony",
    "email": "jane@email",
    "password": "password"
  },
  {
    "username": "Jim Berry",
    "email": "jim@email",
    "password": "password",
    "authLevel": 2
  }
]
```

```
Server is running on port 8080
Connected to the database
User created: john@email
User created: jane@email
User created: jim@email
```

```
[
  {
    "_id": "6740c7ce4fbec3a9d63b4026",
    "username": "John Doe",
    "password": "password",
    "email": "john@email",
    "authLevel": 1,
    "__v": 0
  },
  {
    "_id": "6740c7ce4fbec3a9d63b4027",
    "username": "Jane Melony",
    "password": "password",
    "email": "jane@email",
    "authLevel": 1,
    "__v": 0
  },
  {
    "_id": "6740c7ce4fbec3a9d63b4028",
    "username": "Jim Berry",
    "password": "password",
    "email": "jim@email",
    "authLevel": 2,
    "__v": 0
  }
]
```

המשיכו את התרגיל הקודם ופתרו את התרגילים לפי הסדר.

תרגיל	תיאור המשימה
Ex-1	צרו קובץ JSON ובו מערך עם נתונים של 3 משתמשים.
Ex-2	צרו פונקציה שבמידה ומשתמשים אלו עדיין לא קיימים –יוצרת אותם באמצעות שימוש בקובץ ה-JSON.
Ex-3	קראו לפונקציה זו בעת הפעלת השרת ובדקו שהנתונים נוספו.
Ex-4	מחקו משתמשים אלו, הפעילו את השרת מחדש ובדקו שהמשתמשים נוצרו מחדש לאחר ההפעלה.

משתני סביבה ו-dotenv



בחלק זה נכיר ונלמד כיצד להשתמש בספריה **dotenv** על מנת לשמור משתני סביבה לשימוש התוכנית. בסיום הנושא תוכלו לענות על השאלות הבאות:

```
PORT=8080

## Database
MONGO_LOCAL=mongodb://localhost:27017/your_db_name
```

```
import env from 'dotenv';
env.config();

export const PORT = process.env.PORT || 8181;
export const MONGO_LOCAL = process.env.MONGO_LOCAL;
```

- מה היא **dotenv**?
- במה היא מסייעת לנו?
- כיצד נוכל להתקין אותה על הפרויקט?
- איזה סוג של מידע נרצה לאחסן בקובץ **.env**?
- מדוע שלא נרצה שקובץ זה יעלה ל-**github** וכיצד נוכל למנוע זאת?
- כיצד נוכל לשלוף מידע אשר מאוחסן בקובץ **.env**?
- מדוע שנרצה **service** שמגיש לנו משתנים מה-**.env**?
- כיצד נוכל לאחסן ולשלוף את ה-**Port** וה-**Connection string** למסד הנתונים שלנו מתוך קובץ ה-**.env**. על מנת להשתמש בהם בתוכנית?

המשיכו את התרגיל הקודם ופתרו את התרגילים לפי הסדר.

```
PORT=8080

## Database
MONGO_LOCAL=mongodb://localhost:27017/your_db_name
```

```
import env from 'dotenv';
env.config();

export const PORT = process.env.PORT || 8181;
export const MONGO_LOCAL = process.env.MONGO_LOCAL;
```

תרגיל	תיאור המשימה
Ex-1	התקינו את dotenv וצרו קובץ env. בתיקייה הראשית של השרת שלכם.
Ex-2	צרו בו משתנים עבור ה-PORT וה-Connection String.
Ex-3	צרו service אשר ייבא משתנים אלו מתוך קובץ ה-env. וייצא אותם.
Ex-4	ייבאו את המשתנים אל קובץ ההרצה של השרת והשתמשו בהם.
Ex-5	בדקו שהשרת רץ בצורה תקינה והפעולות מול מסד הנתונים פועלות כצפוי.

תודה על ההקשבה

אני וצוות המכללה כאן עבורכם