# MongoDB

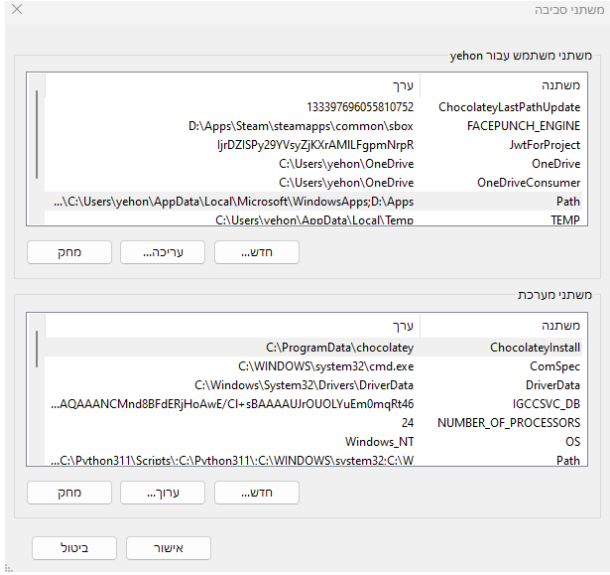
במסגרת הנושא נכיר ונעמיק ידע ב-MongoDB ונלמד על אפשרויות מתקדמות בניהול מסדי נתונים, כולל שאילתות מורכבות ועוד.

# מבוא והתקנות

# MongoDB – מבוא והתקנות

בחלק זה נכיר את MongoDB, מסד נתונים NoSQL מבוסס מסמכים, ונדון בהתקנתו. בסיום הנושא תוכלו לענות על השאלות הבאות:

- מהי **MongoDB** ומה מבדיל אותה ממסדי נתונים רלציוניים?
- למה כדאי להשתמש ב-**MongoDB**?
- כיצד מתקינים **MongoDB** על המחשב?
- מהו **MongoDB Atlas** וכיצד הוא משמש לפיתוח בענן?
- כיצד מתחברים לשרת **MongoDB**?
- מהם הכלים השונים לניהול **MongoDB**?
- איזה כלים עלינו להתקין מתוך עמוד ההורדות של **MongoDb**?
- **משתמשי Windows בלבד**: כיצד ניתן להגדיר PATH במערכת כך שניתן יהיה לגשת ל-**Mongosh** ול-**Databsae Tools** מכל מיקום במחשב שלנו?




[Download MongoDB Community Server | MongoDB]

[Download MongoDB Tools | MongoDB]

# מבני נתונים מבוססי מסמכים

# מבני נתונים מבוססי מסמכים

בחלק זה נכיר את מבנה הנתונים ב-MongoDB, ונדון כיצד המידע נשמר במסמכים ובאוספים במקום בטבלאות. בסיום הנושא תוכלו לענות על השאלות הבאות:

- מהו מסמך (**Document**) ב-**MongoDB**?
- כיצד אוספים (**Collections**) משמשים לארגון המסמכים?
- כיצד מבנה הנתונים ב- **MongoDB** שונה ממבנה טבלאי רלציוני?
- מהם שדות (**Fields**) ומערכי שדות (**Arrays**) במסמכים?
- כיצד ניתן לשמור מידע מקונן (**Nested Data**) במסמכים?

```
_id: ObjectId('6694d5528b1333cf4a2eb310')
title : "card77"
subtitle : "the card number 77"
description : "walla walla walla"
phone : "054-8339879"
email : "card77@gmail.com"
web : "https://www.youtube.com/watch?v=3vUx1E7VEhg&ab_channel=Cohen%40Mushon-…"
▾ image : Object
    alt : "user-profile"
    url : "https://picsum.photos/200/300"
    _id : ObjectId('6694d5528b1333cf4a2eb30f')
▾ address : Object
    city : "Anytown"
    state : ""
    country : "Israel"
    street : "123 Main St"
    houseNumber : 20
    zip : "12345"
    _id : ObjectId('6694d5528b1333cf4a2eb311')
createdAt : 2024-07-15T07:52:49.968+00:00
userId : "65801f1ee530f807b48d396e"
▸ likes : Array (empty)
bizNumber : 850074
__v : 0
```

**cards**

| Storage size: | Documents: | Avg. document size: | Indexes: | Total index size: |
|---|---|---|---|---|
| 20.48 kB | 3 | 563.00 B | 2 | 40.96 kB |

**users**

| Storage size: | Documents: | Avg. document size: | Indexes: | Total index size: |
|---|---|---|---|---|
| 20.48 kB | 3 | 502.00 B | 2 | 40.96 kB |

# Mongosh

# הורדת והתקנת האוסף biz_cards_dev

Welcome/biz_cards_dev.rar

הפעילו את **MongoDB Compass** ופתרו את התרגילים לפי הסדר.

```
mongorestore --db bcards dump/biz-cards-dev
```

**cards**

| Storage size: | Documents: | Avg. document size: | Indexes: | Total index size: |
|---|---|---|---|---|
| 20.48 kB | 3 | 563.00 B | 2 | 40.96 kB |

**users**

| Storage size: | Documents: | Avg. document size: | Indexes: | Total index size: |
|---|---|---|---|---|
| 20.48 kB | 3 | 502.00 B | 2 | 40.96 kB |

| תרגיל | תיאור המשימה |
|---|---|
| Ex-2 | וודאו שפעלתם לפי הוראות ההתקנה ושאכן התקנתם את התוספים Mongosh ו-MongoDB Database Tools. |
| Ex-1 | הורידו את מסד הנתונים **biz_cards_dev** מהכתובת yehonatan604/MongoDB-Learn |
| Ex-2 | חלצו את הקבצים לתיקייה חדשה על גבי המחשב שלכם. |
| Ex-4 | אחזרו את מסד הנתונים **biz_cards_dev** על ידי שימוש ב-**mongorestore**. |
| Ex-4 | רעננו את **MongoDB Compass** ובדקו שמופיע לכם מסד הנתונים שאחזרנו. |

Mongosh/example.bash

# Mongosh - הפעלת שאילתות בסיסיות

בחלק זה נכיר את Mongosh - סביבת שורת הפקודה של MongoDB, ונריץ שאילתות בסיסיות על מנת להחזיר את כל המסמכים באוסף. בסיום הנושא תוכלו לענות על השאלות הבאות:

- מהו Mongosh וכיצד משתמשים בו?
- כיצד ניתן להוריד ולהתקין את Mongosh?
- כיצד נוכל להתחבר למסד נתונים באמצעות use?
- באמצעות איזו פקודה נוכל לצפות בכל מסדי הנתונים שיש ברשותנו?
- לשם מה משמשת המתודה find()?
- כיצד מוצאים את כל המסמכים באוסף באמצעות find()?

```
test> show dbs
```

```
test> use biz_cards_dev
switched to db biz_cards_dev
biz_cards_dev> show collections
cards
users
biz_cards_dev> db.users.find()
```

```
{
  _id: ObjectId('6694d5528b1333cf4a2eb2fb'),
  name: {
    first: 'Bruce',
    middle: '',
    last: 'Wayne',
    _id: ObjectId('6694d5528b1333cf4a2eb2fc')
  },
  address: {
    city: 'Anytown',
    state: '',
    country: 'Israel',
    street: '123 Main St',
    houseNumber: 20,
    zip: '12345',
    _id: ObjectId('6694d5528b1333cf4a2eb2fd')
  },
  image: {
    alt: 'user-profile',
    url: 'https://picsum.photos/200/300',
    _id: ObjectId('6694d5528b1333cf4a2eb2fe')
  },
  phone: '050-8123091',
  email: 'Bruce4@batcave.com',
  password: '$2b$12$GjQHzmpsLnoWEg.IyjmJz.Bt6zRx0x7Rs1NlUsUz8boH/Wt3Yq7x6',
  isAdmin: false,
  isBusiness: false,
  createdAt: ISODate('2024-07-15T07:52:48.892Z'),
  __v: 0
},
```

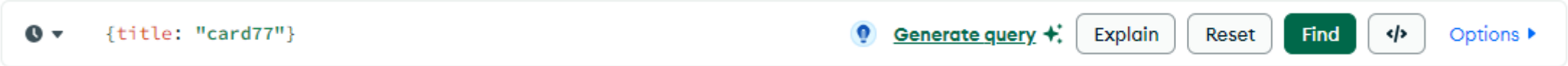
# MongoDb Compass

# Compass - הפעלת שאילתות בסיסיות

בחלק זה נכיר את MongoDB Compass - כלי ה- GUI לניהול מסדי נתונים ב-MongoDB, ונלמד כיצד לבצע שאילתות בסיסיות בצורה ויזואלית. בסיום הנושא תוכלו לענות על השאלות הבאות:

- מה זה **GUI**?
- מהו **MongoDB Compass** ומה מטרתו?
- כיצד ניתן להוריד ולהתקין את **MongoDB Compass**?
- כיצד נוכל להתחבר למסד נתונים באמצעות **MongoDB Compass**?
- כיצד נוכל לצפות במסמכים של אוסף על ידי שימוש בממשק הגרפי?
- כיצד ניתן להריץ שאילתות באמצעות שימוש ב-**MongoDB Compass**?
- כיצד נוכל לצפות במסמך ספציפי באמצעות שימוש בשאילתת-**find()**?

```
_id: ObjectId('6694d5528b1333cf4a2eb310')
title : "card77"
subtitle : "the card number 77"
description : "walla walla walla"
phone : "054-8339879"
email : "card77@gmail.com"
web : "https://www.youtube.com/watch?v=3vUx1E7VEhg&ab_channel=Cohen%40Mushon-…"
▸ image : Object
▸ address : Object
createdAt : 2024-07-15T07:52:49.968+00:00
userId : "65801f1ee530f807b48d396e"
▸ likes : Array (empty)
bizNumber : 850074
__v : 0
```



Compass/exercise.js

# Compass והפעלת שאילתות בסיסיות - תרגול

הפעילו את **MongoDB Compass**, גשו למסד הנתונים **biz_cards_dev**, ופתרו את התרגילים לפי הסדר.

**cards**

| Storage size: | Documents: | Avg. document size: | Indexes: | Total index size: |
|---|---|---|---|---|
| 20.48 kB | 3 | 563.00 B | 2 | 40.96 kB |

**users**

| Storage size: | Documents: | Avg. document size: | Indexes: | Total index size: |
|---|---|---|---|---|
| 20.48 kB | 3 | 502.00 B | 2 | 40.96 kB |

```
_id: ObjectId('6694d5528b1333cf4a2eb310')
title : "card77"
subtitle : "the card number 77"
description : "walla walla walla"
phone : "054-8339879"
email : "card77@gmail.com"
web : "https://www.youtube.com/watch?v=3vUx1E7VEhg&ab_channel=Cohen%40Mushon-…"
▸ image : Object
▸ address : Object
createdAt : 2024-07-15T07:52:49.968+00:00
userId : "65801f1ee530f807b48d396e"
▸ likes : Array (empty)
bizNumber : 850074
__v : 0
```

| תרגיל | תיאור המשימה |
|---|---|
| Ex-1 | גשו לאוסף **Cards** וצפו במסמכים. |
| Ex-2 | הציגו באמצעות שאילתת **find()** את הכרטיס שהכותרת שלו "card77". |
| Ex-3 | הציגו באמצעות שאילתת **find()** את הכרטיס שכתובת המייל שלו היא "**card88@gmail.com**". |
| Ex-4 | גשו לאוסף **Users** וצפו במסמכים. |
| Ex-5 | הציגו באמצעות שאילתת **find()** את כל המשתמשים שהכתובת שלהם נמצאת בישראל. |

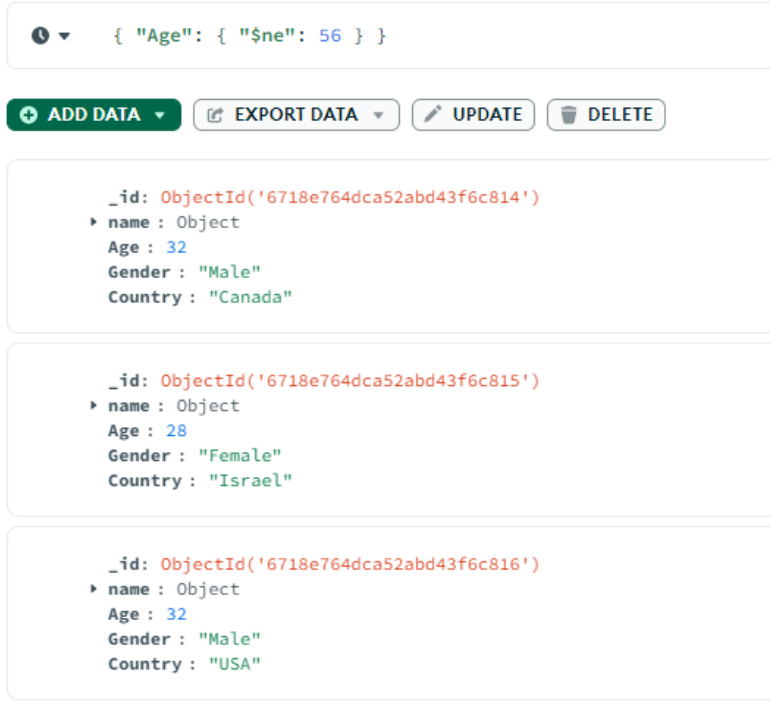
# C.R.U.D Operations

CRUD/example.js

# מבוא - C.R.U.D Operations

בחלק זה נכיר את הפעולות הבסיסיות שניתן לבצע על מסמכים – יצירה (**Create**), קריאה (**Read**), עדכון (**Update**), ומחיקה (**Delete**). בסיום הנושא תוכלו לענות על השאלות הבאות:

- מה הן פעולות ה-**C.R.U.D**?
- כיצד יוצרים מסמך חדש ב-**Compass**?
- כיצד ניתן לקרוא מידע ממסד הנתונים באמצעות שימוש ב-**Compass**?
- מה משמעות האופרטורים: (**$ne ,$eq ,$gt, $lt**)?
- כיצד נוכל להשתמש באופרטרים למטרת סינון בשאילתות **find()**?
- כיצד ניתן לעדכן מידע קיים במסד הנתונים באמצעות שימוש ב-**Compass**?
- כיצד ניתן למחוק מידע ממסד הנתונים באמצעות שימוש ב-**Compass**?



CRUD/exercise.js

# C.R.U.D Operations – תרגול חלק א'

הפעילו את **MongoDB Compass**, גשו למסד הנתונים **biz_cards_dev**, ופתרו את התרגילים לפי הסדר.

| תרגיל | תיאור המשימה |
|---|---|
| Ex-1 | צרו מסד נתונים חדש בשם **MoviesDB**. |
| Ex-2 | צרו בו אוסף בשם **Movies** ואוסף בשם **Actors**. |
| Ex-3 | הוסיפו 3 מסמכים לכל אוסף בהתאם לדוגמה. |

```
[
  {
    "name": "The Shnitz",
    "Year": "2012",
    "TimeInMinutes": "95",
    "Genere": "Action"
  },
  {
    "name": "Dodger",
    "Year": "1998",
    "TimeInMinutes": "83",
    "Genere": "Action"
  },
  {
    "name": "Entropy",
    "Year": "2024",
    "TimeInMinutes": "119",
    "Genere": "Sci-Fi"
  }
]
```

```
[
  {
    "name": {
      "first": "Bob",
      "last": "Sponk"
    },
    "Age": 32,
    "Gender": "Male",
    "Country": "Canada"
  },
  {
    "name": {
      "first": "Nully",
      "last": "Zeimesh"
    },
    "Age": 28,
    "Gender": "Female",
    "Country": "Israel"
  },
  {
    "name": {
      "first": "Noriss",
      "last": "Chluck"
    },
    "Age": 32,
    "Gender": "Male",
    "Country": "USA"
  }
]
```



מחבר: גל לביא

CRUD/exercise.js

# C.R.U.D Operations – תרגול חלק ב'

המשיכו את התרגיל הקודם ופתרו את התרגילים לפי הסדר.

| תרגיל | תיאור המשימה |
|---|---|
| Ex-4 | הציגו באמצעות שאילתת find את כל סרטי האקשן. |
| Ex-5 | הציגו באמצעות שאילתת find את כל השחקנים שמעל לגיל 30. |
| Ex-6 | הציגו באמצעות שאילתת find את כל השחקנים שמתחת לגיל 40 וגרים בישראל. |
| Ex-7 | מיחקו את אחד הסרטים. |
| Ex-8 | שנו את השם הפרטי של אחד השחקנים. |

```
[
  {
    "name": "The Shnitz",
    "Year": "2012",
    "TimeInMinutes": "95",
    "Genere": "Action"
  },
  {
    "name": "Dodger",
    "Year": "1998",
    "TimeInMinutes": "83",
    "Genere": "Action"
  },
  {
    "name": "Entropy",
    "Year": "2024",
    "TimeInMinutes": "119",
    "Genere": "Sci-Fi"
  }
]
```

```
[
  {
    "name": {
      "first": "Bob",
      "last": "Sponk"
    },
    "Age": 32,
    "Gender": "Male",
    "Country": "Canada"
  },
  {
    "name": {
      "first": "Nully",
      "last": "Zeimesh"
    },
    "Age": 28,
    "Gender": "Female",
    "Country": "Israel"
  },
  {
    "name": {
      "first": "Noriss",
      "last": "Chluck"
    },
    "Age": 32,
    "Gender": "Male",
    "Country": "USA"
  }
]
```

# התממשקות בסיסית עם תוכנית Node.js



מחבר: גל לביא

# תוכנית ה-Node הראשונה שלי

JS/example.bash
JS/package.json
JS/server.js

בחלק זה נבנה את תוכנית ה-Node.js הראשונה שלנו שתתחבר למסד נתונים MongoDB נלמד כיצד להתקין את הספרייה הנדרשת להריץ קוד בסיסי. בסיום הנושא תוכלו לענות על השאלות הבאות:

- מדוע עלינו להקים פרויקט **Node.js** (לא ניתן פשוט להתממשק עם **MongoDB** דרך תוכנית **JS** סטנדרטית אשר רצה על גבי הדפדפן)?
- כיצד יוצרים פרויקט **Node.js** חדש?
- מה עלינו להגדיר ב-**package.json**?
- כיצד מתקינים את **MongoDB Driver** עבור **Node.js**?
- כיצד יוצרים חיבור למסד הנתונים **MongoDB** דרך הקוד?
- מדוע הפעולות שמתבצעות מול מסד הנתונים הן אסינכרוניות?
- כיצד נוכל להשתמש בשאילתות **find()**?
- כיצד סוגרים את החיבור לאחר סיום הפעולה?

```json
{
  "name": "mongo-with-node",
  "version": "1.0.0",
  "type": "module",
  "main": "server.js",
  ▷ Debug
  "scripts": {
    "dev": "node ."
  },
  "keywords": [],
  "author": "",
  "license": "ISC",
  "dependencies": {
    "mongodb": "^6.10.0"
  }
}
```

```javascript
import { MongoClient } from 'mongodb';

async function main() {
    const uri = "mongodb://localhost:27017";
    const client = new MongoClient(uri);

    try {
        await client.connect();
        const database = client.db('MoviesDB');
        const collection = database.collection('Movies');

        // Find all documents
        const movies = await collection.find({}).toArray();
        console.log(movies);
    } finally {
        await client.close();
    }
}

main().catch(console.error);
```

# תוכנית ה-Node הראשונה שלי - תרגול

JS/exercise.js

```
*** Exercise 1 ***
[
  {
    _id: new ObjectId('6718e547dca52abd43f6c80f'),
    name: 'The Shnitz',
    Year: '2012',
    TimeInMinutes: '95',
    Genere: 'Action'
  },
  {
    _id: new ObjectId('6718e547dca52abd43f6c810'),
    name: 'Dodger',
    Year: '1998',
    TimeInMinutes: '83',
    Genere: 'Action'
  }
]

*** Exercise 2 ***
[
  {
    _id: new ObjectId('6718e764dca52abd43f6c814'),
    name: { first: 'Bob', last: 'Sponk' },
    Age: 32,
    Gender: 'Male',
    Country: 'Canada'
  },
  {
    _id: new ObjectId('6718e764dca52abd43f6c816'),
    name: { first: 'Noriss', last: 'Chluck' },
    Age: 32,
    Gender: 'Male',
    Country: 'USA'
  }
]

*** Exercise 3 ***
[
  {
    _id: new ObjectId('6718e764dca52abd43f6c815'),
    name: { first: 'Nully', last: 'Zeimesh' },
    Age: 28,
    Gender: 'Female',
    Country: 'Israel'
  }
]
```

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

| תרגיל | תיאור המשימה |
|---|---|
| Ex-1 | הדפיסו לקונסול באמצעות שאילתת **find** את כל סרטי האקשן. |
| Ex-2 | הדפיסו לקונסול באמצעות שאילתת **find** את כל השחקנים שמעל לגיל 30 אבל מתחת ל-40. |
| Ex-3 | הדפיסו לקונסול באמצעות שאילתת **find** את כל השחקנים שמתחת לגיל 40 וגרים בישראל. |

# פעולות ה-C.R.U.D באמצעות שימוש ב-MongoDB Driver

# פעולות C.R.U.D באמצעות שימוש ב- MongoDB Driver

בחלק זה נלמד כיצד ניתן לבצע את כל פעולות ה-CRUD (יצירה, קריאה, עדכון ומחיקה) בתוכנית Node.js באמצעות שימוש ב-MongoDB Driver. בסיום הנושא תוכלו לענות על השאלות הבאות:

- כיצד מוסיפים מסמך חדש באמצעות שימוש ב-**insertOne**()?
- כיצד מוסיפים מספר מסמכים חדשים באמצעות שימוש ב-**insertMany**()?
- כיצד מעדכנים מסמך באמצעות שימוש ב-**updateOne**()?
- כיצד מוחקים מסמך באמצעות שימוש ב-**deleteOne**()?

JS-CRUD/exercise.js

# פעולות C.R.U.D באמצעות שימוש ב- MongoDB Driver - תרגול

```
*** Exercise 1 ***
{
  acknowledged: true,
  insertedCount: 2,
  insertedIds: {
    '0': new ObjectId('671904cd85365aebc693bbfe'),
    '1': new ObjectId('671904cd85365aebc693bbff')
  }
}

*** Exercise 2 ***
[
  {
    _id: new ObjectId('6718e764dca52abd43f6c814'),
    name: { first: 'Bob', last: 'Sponk' },
    Age: 34,
    Gender: 'Male',
    Country: 'Canada'
  },
  {
    _id: new ObjectId('6718e764dca52abd43f6c815'),
    name: { first: 'Nully', last: 'Zeimesh' },
    Age: 28,
    Gender: 'Female',
    Country: 'Israel'
  },
  {
    _id: new ObjectId('6718e764dca52abd43f6c816'),
    name: { first: 'Noriss', last: 'Chluck' },
    Age: 34,
    Gender: 'Male',
    Country: 'USA'
  }
]

*** Exercise 3 ***
{ acknowledged: true, deletedCount: 1 }
```

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

| תרגיל | תיאור המשימה |
|---|---|
| Ex-1 | הוסיפו 2 מסמכים חדשים לאוסף **Movies**. |
| Ex-2 | שנו את גילם של כל השחקנים שמעל ל-30 למספר רנדומלי בין 30 ל-50, והדפיסו את כל השחקנים לקונסול. |
| Ex-3 | מחקו את כל הסרטים שהשם שלהם מתחיל באות A (מומלץ להיעזר ב-**regex**). |

# Mongoose – עבודה עם MongoDb באמצעות ODM

# Mongoose – מבוא והתקנות

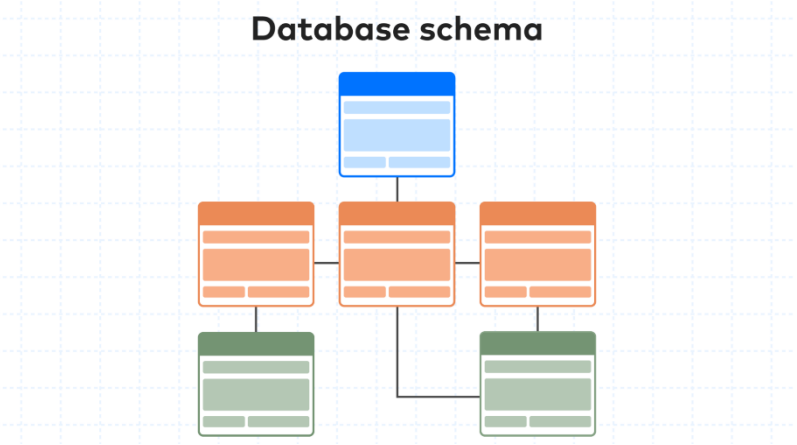
Mongoose/example.bash
Mongoose/models/User.js
Mongoose/example.js
Mongoose/server.js

בחלק זה נלמד כיצד לעבוד עם Mongoose - ספריית ODM (Object Data Modeling) ל-Node.js שמאפשרת עבודה עם מסמכים בצורה מאורגנת ומבוססת מודלים. בסיום הנושא תוכלו לענות על השאלות הבאות:

- מה זה "ODM"?
- מהו Mongoose וכיצד הוא עוזר בניהול נתונים ב-MongoDB?
- כיצד מתקינים את Mongoose בפרויקט Node.js?
- כיצד מגדירים חיבור ל-MongoDB באמצעות Mongoose?
- מה היא "סכמה" ומה הוא "מודל"?
- כיצד יוצרים מודל וסכמה על ידי שימוש ב-Mongoose?

mongoose

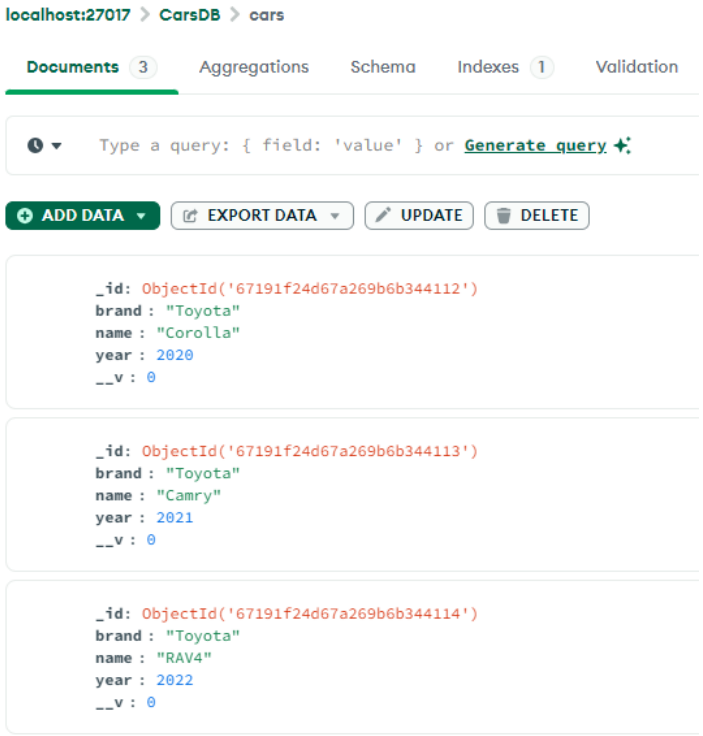
elegant mongodb object modeling for node.js



# Mongoose – תרגול

Mongoose/models/Car.js
Mongoose/exercise.js
*npm run exercise

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




| תרגיל | תיאור המשימה |
|---|---|
| Ex-1 | צרו מסד נתונים חדש בשם **CarsDB**. |
| Ex-2 | צרו מודל וסכמה ל-**Car** לפי הדוגמא. |
| Ex-3 | כתבו פונקציה שתוסיף למסד הנתונים 3 מסמכים חדשים עבור מכוניות על ידי שימוש במודל שיצרתם. |
| Ex-4 | בדקו ב-**Compass** שמסד הנתונים, האוסף והמסמכים נוצרו כהלכה. |

# Mongoose – פעולות ה-CRUD

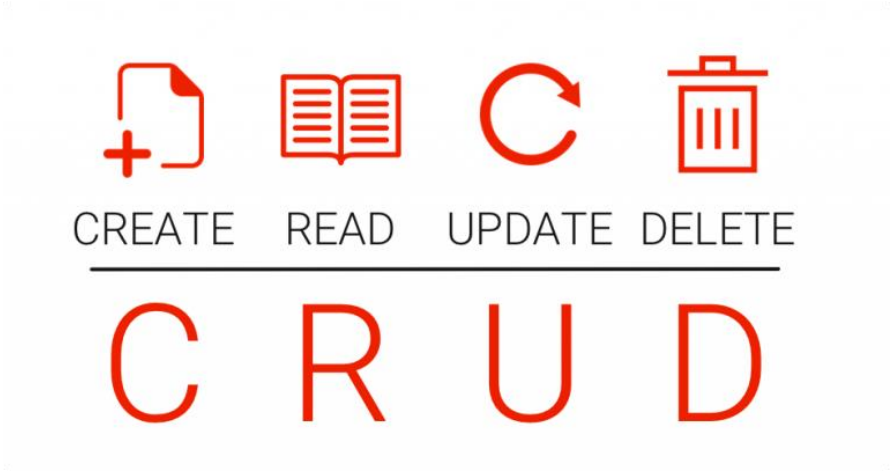
Mongoose/example.bash
Mongoose/example.js
Mongoose/models/User.js
Mongoose/server.js

בחלק זה נלמד כיצד לבצע את פעולות ה-CRUD (יצירה, קריאה, עדכון ומחיקה) באמצעות Mongoose.
בסיום הנושא תוכלו לענות על השאלות הבאות:

- איזה "טייפים" ניתן להגדיר לשדות?
- כיצד ניתן להגדיר שדה כשדה חובה?
- כיצד ניתן להגדיר שדה כשדה ייחודי?
- כיצד ניתן ליצור מסמך חדש באמצעות שימוש במתודה **save**?
- כיצד ניתן לשלוף את כל המסמכים של אוסף?
- לשם מה נשתמש באופרטור "**$exists**"?
- כיצד ניתן לשלוף מסמך בודד מתוך אוסף?
- כיצד ניתן לשלוף את כל המסמכים של אוסף לפי תנאי?
- כיצד ניתן ליצור רשימה של מסמכים בבת אחת?
- כיצד ניתן לשלוף ולעדכן מסמך בפקודה אחת?
- כיצד ניתן לשלוף ולמחוק מסמך בפקודה אחת?

mongoose

elegant mongodb object modeling for node.js



# Mongoose – פעולות ה-CRUD - תרגול

Mongoose/Models/Article.js
Mongoose/exercise.js
*npm run exercise

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

```
const articleSchema = new Schema({
    title: {
        type: String,
        required: true
    },
    creator: {
        first: {
            type: String,
            required: true
        },
        last: {
            type: String,
            required: true
        },
        middle: {
            type: String,
            required: false
        },
        email: {
            type: String,
            required: true,
            unique: true
        }
    },
    content: {
        type: String,
        required: true
    }
});
```

| תרגיל | תיאור המשימה |
|---|---|
| Ex-1 | צרו מסד נתונים חדש בשם **ArticlesDB**. |
| Ex-2 | צרו מודל וסכמה ל-**Article** לפי הדוגמא. |
| Ex-3 | כתבו פונקציה שתוסיף למסד הנתונים 5 מסמכים חדשים עבור מאמרים על ידי שימוש במודל שיצרתם. |
| Ex-4 | שלפו את כל המאמרים שהאורך של הכותרת שלהם ארוך מ-5 אותיות והדפיסו לקונסול. |
| Ex-5 | עדכנו את האות הראשונה של השם הפרטי ושם המשפחה של היוצר לאות גדולה בכל המאמרים והדפיסו את כל המאמרים לקונסול **לאחר השינוי** ללא שימוש ב-**update** (השתמשו בלולאת **for**). |
| Ex-6 | עדכנו את הכותרת ל-"**Updated Title**" בכל המאמרים והדפיסו את כל המאמרים לקונסול **לאחר השינוי**, יש להשתמש ב-**updateMany**. |
| Ex-7 | מחקו את כל המאמרים שליוצר שלהם אין שם אמצעי והדפיסו אותם לקונסול **לאחר המחיקה**. |

# זמן שאלות

# בדיקת רמת ידע

# סוף