BURSA TEKNİK ÜNİVERSİTESİ

Mühendislik ve Doğa Bilimleri Fakültesi – Bilgisayar Mühendisliği Bölümü

BLM0470 NodeJS İle Web Programlama

2023-2024 Bahar Dönemi

12.Hafta Raporu

Berkan SERBES – 22360859353

İçindekiler

Not: Bu rapor sadece kodların ekran görüntülerini içermektedir. Kodlara dair herhangi bir açıklama yazılmamıştır.

1.Görev Yönetim Uygulaması Geliştirme

```js
index.js > ...
1    require("dotenv").config();
2    const { MongoClient } = require("mongodb");
3
4    const client = new MongoClient(process.env.CONNECTION_URL);
5
6    const DB_NAME = "task_manager";
7
8    const main = async () => {
9      try {
10       const result = await client.connect();
11       console.log("Connected to the database");
12     } catch (err) {
13       console.error(err);
14     } finally {
15       await client.close();
16     }
17   };
18
19   main();
```

```
PROBLEMS    OUTPUT    DEBUG CONSOLE    PORTS    TERMINAL                                          powersh

PS C:\Users\berka\OneDrive\Masaüstü\CENG File\BTU Ders Dosyaları\2023-2024 Bahar Dönemi\BLM0470 - NodeJS ile Web Programlama\Projects\RaporKodlar
i> node .\index.js
Connected to the database
```

. . .

```js
package.json        index.js  ●      .env
index.js > ...
1    require("dotenv").config();
2    const { MongoClient } = require("mongodb");
3
4    const client = new MongoClient(process.env.CONNECTION_URL);
5
6    const DB_NAME = "task_manager";
7
8    const main = async () => {
9      const usersCollection = client.db(DB_NAME).collection("users");
10     const tasksCollection = client.db(DB_NAME).collection("tasks");
11
12     try {
13       await client.connect();
14       //console.log("Connected to the database");
15       const result = await usersCollection.insertOne({
16         name: "Bill Gates",
17         age: 68,
18       });
19
20       console.log(`A document was inserted with the _id: ${result.insertedId}`);
21     } catch (err) {
22       console.error(err);
23     } finally {
24       await client.close();
25     }
26   };
27
28   main();
29
```

```
PROBLEMS    OUTPUT    DEBUG CONSOLE    PORTS    TERMINAL                                    powershell

PS C:\Users\berka\OneDrive\Masaüstü\CENG File\BTU Ders Dosyaları\2023-2024 Bahar Dönemi\BLM0470 - NodeJS ile Web Programlama\Projects\RaporKodlari\Hafta12_RaporKodlar
i> node .\index.js
A document was inserted with the _id: 664b1f087cfac973bd728818
```

```js
require("dotenv").config();
const { MongoClient } = require("mongodb");

const client = new MongoClient(process.env.CONNECTION_URL);

const DB_NAME = "task_manager";

const main = async () => {
  const usersCollection = client.db(DB_NAME).collection("users");
  const tasksCollection = client.db(DB_NAME).collection("tasks");

  try {
    await client.connect();
    //console.log("Connected to the database");

    const users = [
      { name: "Leo", age: 19 },
      { name: "Luis", age: 20 },
      { name: "Luisa", age: 21 },
    ];

    const result = await usersCollection.insertMany(users);
    console.log(`${result.insertedCount} documents inserted successfully!`);
  } catch (err) {
    console.error(err);
  } finally {
    await client.close();
  }
};

main();
```

PROBLEMS   OUTPUT   DEBUG CONSOLE   PORTS   TERMINAL

PS C:\Users\berka\OneDrive\Masaüstü\CENG File\BTU Ders Dosyaları\2023-2024 Bahar Dönemi\BLM0470 - NodeJS ile Web Programlama\Projects\RaporKodlari\Hafta12_RaporKodlar
i> node .\index.js
3 documents inserted successfully!

task_manager.users

STORAGE SIZE: 36KB   LOGICAL DATA SIZE: 190B   TOTAL DOCUMENTS: 4   INDEXES TOTAL SIZE: 36KB

Find   Indexes   Schema Anti-Patterns ⓪   Aggregation   Search Indexes

Generate queries from natural language in Compass⌕

Filter⌕   Type a query: { field: 'value' }

QUERY RESULTS: **1-4 OF 4**

```
_id: ObjectId('664b1f087cfac973bd728818')
name : "Bill Gates"
age : 68
```

```
_id: ObjectId('664b20423a90dbf7b364353c')
name : "Leo"
age : 19
```

```
_id: ObjectId('664b20423a90dbf7b364353d')
name : "Luis"
age : 20
```

```
_id: ObjectId('664b20423a90dbf7b364353e')
name : "Luisa"
age : 21
```
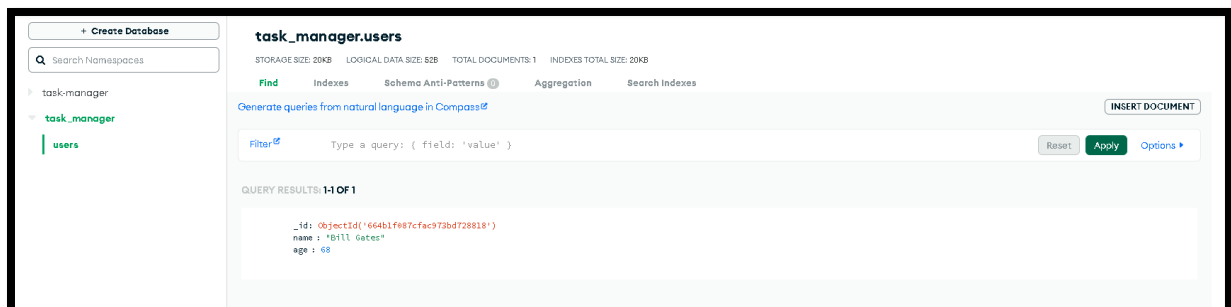
```js
index.js > ...
  1  require("dotenv").config();
  2  const { MongoClient } = require("mongodb");
  3
  4  const client = new MongoClient(process.env.CONNECTION_URL);
  5
  6  const DB_NAME = "task_manager";
  7
  8  const main = async () => {
  9    const usersCollection = client.db(DB_NAME).collection("users");
 10    const tasksCollection = client.db(DB_NAME).collection("tasks");
 11
 12    try {
 13      await client.connect();
 14      //console.log("Connected to the database");
 15
 16      const result = await usersCollection.updateOne(
 17        { name: "Leo" },
 18        { $set: { name: "Leonardo", age: 25 } }
 19      );
 20
 21      console.log(`Updated ${result.modifiedCount} document(s)`);
 22    } catch (err) {
 23      console.error(err);
 24    } finally {
 25      await client.close();
 26    }
 27  };
 28
 29  main();
```

PROBLEMS   OUTPUT   DEBUG CONSOLE   PORTS   TERMINAL   powershell +

```
PS C:\Users\berka\OneDrive\Masaüstü\CENG File\BTU Ders Dosyaları\2023-2024 Bahar Dönemi\BLM0470 - NodeJS ile Web Programlama\Projects\RaporKodlari\Haf
i> node .\index.js
Updated 1 document(s)
```

task_manager.users

STORAGE SIZE: 36KB   LOGICAL DATA SIZE: 195B   TOTAL DOCUMENTS: 4   INDEXES TOTAL SIZE: 3

Find   Indexes   Schema Anti-Patterns 0   Aggregation   Se

Generate queries from natural language in Compass⬚

Filter⬚   Type a query: { field: 'value' }

_id: ObjectId('664b1f087cfac973bd728818')
name : "Bill Gates"
age : 68

_id: ObjectId('664b20423a90dbf7b364353c')
name : "Leonardo"
age : 25

_id: ObjectId('664b20423a90dbf7b364353d')
name : "Luis"
age : 20

_id: ObjectId('664b20423a90dbf7b364353e')
name : "Luisa"
age : 21

```js
require("dotenv").config();
const { MongoClient } = require("mongodb");

const client = new MongoClient(process.env.CONNECTION_URL);

const DB_NAME = "task_manager";

const main = async () => {
  const usersCollection = client.db(DB_NAME).collection("users");
  const tasksCollection = client.db(DB_NAME).collection("tasks");

  try {
    await client.connect();
    //console.log("Connected to the database");

    const cursor = usersCollection.find();
    while (await cursor.hasNext()) {
      console.log(await cursor.next());
    }
  } catch (err) {
    console.error(err);
  } finally {
    await client.close();
  }
};

main();
```

```
PS C:\Users\berka\OneDrive\Masaüstü\CENG File\BTU Ders Dosyaları\2023-2024 Bahar Dönemi\BLM0470 - NodeJS ile Web Programlama\Projects\RaporKodlari\Hafta12_RaporKodlar
i> node .\index.js
{
  _id: new ObjectId('664b1f087cfac973bd728818'),
  name: 'Bill Gates',
  age: 68
}
{
  _id: new ObjectId('664b20423a90dbf7b364353c'),
  name: 'Leonardo',
  age: 25
}
{
  _id: new ObjectId('664b20423a90dbf7b364353d'),
  name: 'Luis',
  age: 20
}
{
  _id: new ObjectId('664b20423a90dbf7b364353e'),
  name: 'Luisa',
  age: 21
}
```

```js
require("dotenv").config();
const { MongoClient } = require("mongodb");

const client = new MongoClient(process.env.CONNECTION_URL);

const DB_NAME = "task_manager";

const main = async () => {
  const usersCollection = client.db(DB_NAME).collection("users");
  const tasksCollection = client.db(DB_NAME).collection("tasks");

  try {
    await client.connect();
    //console.log("Connected to the database");
    const result = await usersCollection.findOne({ name: "Mark" });
    if (!result) {
      console.log("User not found");
    } else {
      console.log(`Found user: ${JSON.stringify(result)}`);
    }
  } catch (err) {
    console.error(err);
  } finally {
    await client.close();
  }
};

main();
```

```
PS C:\Users\berka\OneDrive\Masaüstü\CENG File\BTU Ders Dosyaları\2023-2024 Bahar Dönemi\BLM0470 - NodeJS ile Web Programlama\Projects\RaporKodlari\Hafta12_
i> node .\index.js
User not found
```

```
index.js > [∅] main > [∅] result > ♪ name
 1    require("dotenv").config();
 2    const { MongoClient } = require("mongodb");
 3
 4    const client = new MongoClient(process.env.CONNECTION_URL);
 5
 6    const DB_NAME = "task_manager";
 7
 8    const main = async () => {
 9      const usersCollection = client.db(DB_NAME).collection("users");
10      const tasksCollection = client.db(DB_NAME).collection("tasks");
11
12      try {
13        await client.connect();
14        //console.log("Connected to the database");
15    💡  const result = await usersCollection.findOne({ name: "Luis" });
16        if (!result) {
17          console.log("User not found");
18        } else {
19          console.log(`Found user: ${JSON.stringify(result)}`);
20        }
21      } catch (err) {
22        console.error(err);
23      } finally {
24        await client.close();
25      }
26    };
27
28    main();
29
```

```
PROBLEMS   OUTPUT   DEBUG CONSOLE   PORTS   TERMINAL                                                    powershell + ∨  ⊟  🗑  ⋯  ∧
PS C:\Users\berka\OneDrive\Masaüstü\CENG File\BTU Ders Dosyaları\2023-2024 Bahar Dönemi\BLM0470 - NodeJS ile Web Programlama\Projects\RaporKodlari\Hafta12_RaporKodla
i> node .\index.js
Found user: {"_id":"664b20423a90dbf7b364353d","name":"Luis","age":20}
```

```
index.js > [∅] main > [∅] tasks
 1    require("dotenv").config();
 2    const { MongoClient } = require("mongodb");
 3
 4    const client = new MongoClient(process.env.CONNECTION_URL);
 5
 6    const DB_NAME = "task_manager";
 7
 8    const main = async () => {
 9      const usersCollection = client.db(DB_NAME).collection("users");
10      const tasksCollection = client.db(DB_NAME).collection("tasks");
11
12      try {
13        await client.connect();
14        //console.log("Connected to the database");
15        const tasks = [
16          { desctiption: "Clean the house", completed: true },
17          { desctiption: "Water plants", completed: true },
18    💡    { desctiption: "See the doctor", completed: false },
19        ];
20
21        const result = await tasksCollection.insertMany(tasks);
22        console.log(`${result.insertedCount} documents inserted successfully!`);
23      } catch (err) {
24        console.error(err);
25      } finally {
26        await client.close();
27      }
28    };
29
30    main();
31
```

```
PROBLEMS   OUTPUT   DEBUG CONSOLE   PORTS   TERMINAL                                                    powershell + ∨  ⊟  🗑  ⋯
PS C:\Users\berka\OneDrive\Masaüstü\CENG File\BTU Ders Dosyaları\2023-2024 Bahar Dönemi\BLM0470 - NodeJS ile Web Programlama\Projects\RaporKodlari\Hafta12_RaporK
i> node .\index.js
3 documents inserted successfully!
```

task_manager.tasks

STORAGE SIZE: 20KB   LOGICAL DATA SIZE: 197B   TOTAL DOCUMENTS: 3   INDEXES TOTAL SIZE: 20KB

Find   Indexes   Schema Anti-Patterns ⓪   Aggregation   Search Indexes

Generate queries from natural language in Compass⬀

| Filter⬀ | Type a query: { field: 'value' } | Reset Apply |

QUERY RESULTS: **1-3 OF 3**

```
_id: ObjectId('664b26134f1956e48f023a30')
desctiption : "Clean the house"
completed : true


_id: ObjectId('664b26134f1956e48f023a31')
desctiption : "Water plants"
completed : true


_id: ObjectId('664b26134f1956e48f023a32')
desctiption : "See the doctor"
completed : false
```

```js
index.js > [●] main
1    require("dotenv").config();
2    const { MongoClient } = require("mongodb");
3
4    const client = new MongoClient(process.env.CONNECTION_URL);
5
6    const DB_NAME = "task_manager";
7
8    const main = async () => {
9      const usersCollection = client.db(DB_NAME).collection("users");
10     const tasksCollection = client.db(DB_NAME).collection("tasks");
11
12     try {
13       await client.connect();
14       //console.log("Connected to the database");
15       const cursor = await tasksCollection.find({ completed: true });
16       while (await cursor.hasNext()) {
17         console.log(await cursor.next());
18       }
19     } catch (err) {
20       console.error(err);
21     } finally {
22       await client.close();
23     }
24   };
25
26   main();
```

PROBLEMS   OUTPUT   DEBUG CONSOLE   PORTS   TERMINAL   powershell + ∨ ☐ 🗑 ⋯ ∧

```
PS C:\Users\berka\OneDrive\Masaüstü\CENG File\BTU Ders Dosyaları\2023-2024 Bahar Dönemi\BLM0470 - NodeJS ile Web Programlama\Projects\RaporKodlari\Hafta12_RaporKod
i> node .\index.js
{
  _id: new ObjectId('664b26134f1956e48f023a30'),
  desctiption: 'Clean the house',
  completed: true
}
{
  _id: new ObjectId('664b26134f1956e48f023a31'),
  desctiption: 'Water plants',
  completed: true
}
```