

## WEEK 2: EXERCISE 2

Database System and Cloud (BERR2243)



*Provided by:*

Group Member	Id No.
Ahmad Muhaimin Azeem Bin Adri Wafik	B122310496, 2 BERR S2
Syafiq Iman Bin Suhaimi	B122310218, 2 BERR S2

### JSON QUESTIONS

1. Explain what is CRUD operations and how it is relates to the mongo functions in the exercise.

**CRUD operations is the basic operations performed on data stored in databases. CRUD stands for Create, Read, Update and Delete.**

2. Identify all the mongo operators used in the exercise, then explain the usage for each.

i. **\$gte (greater than or equal operator)**

To match value greater than or equal to a specified value

```
50 //READ: Find all drivers with rating >= 4.5 and isAvailable = true
51 const availableDrivers = await db.collection("carDrivers").find({
52   isAvailable: true,
53   rating: {$gte: 4.5}
54 }).toArray();
55 console.log("available drivers:", availableDrivers);
```

ii. **\$inc (increment or decrement operator)**

To increment or decrement the value of numeric field by a specified amount. If positive value is provided, the operator performs increment. If negative value is provided, the operator performs decrement.

```
57 //UPDATE: Update the rating of a driver by name
58 const updateResult = await db.collection('carDrivers').updateOne(
59   { name: "John Doe"},
60   { $inc: { rating: 0.1}}
61 );
62
63 console.log(`Driver updated with result: ${updateResult}`)
```

3. Replace the mongo functions in Task 5 to updateMany instead of updateOne, compare the difference based on the result in console and the mongo compass.

### updateOne() method

*Code:*

```
64 //UPDATE: Update the rating of a driver by incrementing his/ her rating by 0.1 if that driver is available
65 const updateResult = await db.collection('carDrivers').updateOne(
66   { isAvailable: true},
67   { $inc: { rating: 0.1}})
```

*Output:*

### VS Code

```
new driver inserted with result: [object Object]
Driver updated with result: [object Object]
```

### MongoDB

```
_id: ObjectId('67efeec66cfe5dedb4ab2c7a')
name: "John Doe"
vehicleType: "Sedan"
isAvailable: true
rating: 5
```

Updated the rating when first  
isAvailable: true only was found

```
_id: ObjectId('67efeec66cfe5dedb4ab2c7b')
name: "Alice Smith"
vehicleType: "SUV"
isAvailable: false
rating: 4.4
```

```
_id: ObjectId('67efeec66cfe5dedb4ab2c7c')
name: "Syafiq Iman"
vehicleType: "Convertible"
isAvailable: true
rating: 4.3
```

## updateMany() method

*Code:*

```
64 //UPDATE: Update the rating of all drivers by incrementing them by 0.1 if they are available
65 const updateResult = await db.collection('carDrivers').updateMany(
66   { isAvailable: true },
67   { $inc: { rating: 0.1 } }
68 );
69
70 console.log(`Driver updated with result: ${updateResult}`)
```

*Output:*

VS Code:

```
new driver inserted with result: [object Object]
Driver updated with result: [object Object]
```

```
_id: ObjectId('67eff3a82e54389b0c12c822')
name : "John Doe"
vehicleType : "Sedan"
isAvailable : true
rating : 5
```

```
_id: ObjectId('67eff3a82e54389b0c12c823')
name : "Alice Smith"
vehicleType : "SUV"
isAvailable : false
rating : 4.4
```

```
_id: ObjectId('67eff3a92e54389b0c12c824')
name : "Syafiq Iman"
vehicleType : "Convertible"
isAvailable : true
rating : 4.3999999999999995
```

Updated the rating for **all**  
isAvailable: true were found

4. Replace the mongo functions in Task 6 to deleteMany instead of deleteOne, compare the difference based on the result in console and the mongo compass.

### deleteOne() method

#### *Code:*


```
72 //DELETE: Delete a driver by name
73
74 let deleteResult = await db.collection('carDrivers').deleteOne({isAvailable: false});
75 console.log(`Driver deleted with result: ${deleteResult}`);
76 deleteResult = await db.collection('carDrivers').deleteOne(
77   { name: "Alice Smith"}
78 );
79 console.log(`Driver deleted with result: ${deleteResult}`)
```

#### *Output:*

#### VS Code

```
Driver deleted with result: [object Object]
Driver deleted with result: [object Object]
```

#### MongoDB



```
_id: ObjectId('67eff3a82e54389b0c12c822')
name : "John Doe"
vehicleType : "Sedan"
isAvailable : true
rating : 5.1
```

Object name: "Alice Smith" is deleted

```
_id: ObjectId('67eff3a92e54389b0c12c824')
name : "Syafiq Iman"
vehicleType : "Convertible"
isAvailable : true
rating : 4.499999999999999
```

## deleteMany() method

*Code:*

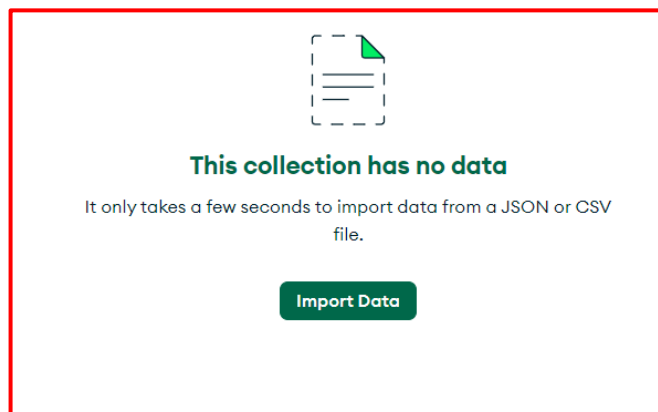
```
72 //DELETE: Delete all drivers by referring rating greater than or equal to 4.0
73 let deleteResult = await db.collection('carDrivers').deleteMany({isAvailable: false});
74 console.log(`Driver deleted with result: ${deleteResult}`);
75 deleteResult = await db.collection('carDrivers').deleteMany(
76   { rating: {$gte: 4.0}}
77 );
78 console.log(`Driver deleted with result: ${deleteResult}`)
```

*Output:*

### VS Code

```
Driver deleted with result: [object Object]
Driver deleted with result: [object Object]
```

## MongoDB



No data at all. All data  
deleted.

GitHub repository:

<https://github.com/Ahmad-Azeem/min-iman/tree/week2>