A blue cylinder with white text

Description automatically generated 

SQL + NoSQL Exam Sets

Roll No: 18

Smit Joshi | NoSQL | 20/09/2023

[](https://github.com/smit-joshi814)

# CREATING DATABASE

-- Creating Database

CREATE DATABASE SQLMid;

-- Selecting Database for Use

use SQLMid;

# SET 1

### Create the following tables (in MYSQL) and insert THREE records.

### CUST\_DETAILS (CustomerID, CustomerFirstName, CustomerCity, CustomerEmailID,Cust\_Gender)

### PRODUCT\_DETAILS (Product\_ID, Product\_Name, Product\_Description, ProductQty, ProductCost)

-- Creating Table CUST\_DETAILS

CREATE TABLE CUST\_DETAILS(

    CustomerID int primary key,

    CustomerFirstName varchar(50),

    CustomerCity varchar(20),

    CustomerEmailID varchar(50),

    CustomerGender varchar(6)

);

-- Creating Table PRODUCT\_DETAILS

CREATE TABLE PRODUCT\_DETAILS(

    ProductID int primary key,

    Product\_Name varchar(100),

    ProductDescription varchar(255),

    ProductQty DECIMAL(2),

    ProductCost DECIMAL(6,2)

);

-- Adding Three Records In Each Table

INSERT INTO CUST\_DETAILS (

    CustomerID,

    CustomerFirstName,

    CustomerCity,

    CustomerEmailID,

    CustomerGender

) VALUES

(1,'Smit Joshi','Deesa','smitjoshi814@gmail.com','Male'),

(2,'Vijay Joshi','Patan','vijayJoshi12@gmail.com','Male'),

(3,'Viahva Joshi','Ahmedabad','vishu4@gmail.com','Female');

INSERT INTO PRODUCT\_DETAILS(

    ProductID,

    Product\_Name,

    ProductDescription,

    ProductQty,

    ProductCost

) VALUES

(101,'Parle-G','Ek Bar Khao Khate hi reh jao',10,4.8),

(102,'LED',"It's Show Time",10,9999.99),

(103,'LCD','Full HD Display',10,8999.99);

-- 1. Add an additional field C\_ CustomerLastName in CUST\_DETAILS table.

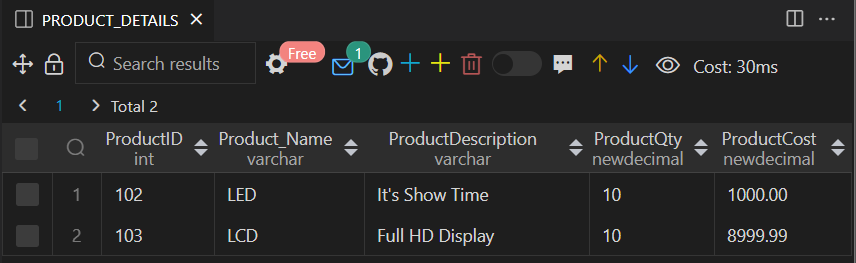
ALTER TABLE CUST\_DETAILS ADD C\_CustomerLastName varchar(50);

-- 2. Change the cost of product to 1000 where Product\_ID is 102.

UPDATE PRODUCT\_DETAILS SET ProductCost=1000 WHERE ProductID=102;

-- 3. . Display only those records from Product where the name of the product is LED or LCD. (Do not use OR)

 SELECT \* FROM PRODUCT\_DETAILS WHERE Product\_Name IN('LED','LCD');



### MongoDB

//Databse For NoSQL Queries

Enterprise NoSQLMid> Use NoSQLMid;

* Create the Freshman Collection and insert THREE documents according to the instructions.(Freshma \_Name, Address, Area\_of\_interest, Certification\_course, Age, Score\_Entrance)

Enterprise NoSQLMid> db.createCollection("Freshman");

Enterprise NoSQLMid> db.Freshman.insertMany([

    {

        Freshma\_Name:"Smit Joshi",

        Address:"Deesa",

        Area\_of\_intrest:"Parita",

        Certification\_course:"Java Hero",

        Age:21,

        Score\_Entrance:80

    },

    {

        Freshma\_Name:"Tejasv Modi",

        Address:"Patan",

        Area\_of\_intrest:"Web Desisning",

        Certification\_course:"Web Developing",

        Age:21,

        Score\_Entrance:50

    },

    {

        Freshma\_Name:"Nisha",

        Address:"Ranuj",

        Area\_of\_intrest:"Dancing",

        Certification\_course:"Dancer's Things",

        Age:22,

        Score\_Entrance:30

    }

]);

{

    acknowledged: true,

    insertedIds: {

      '0': ObjectId("650a78069bddeec3c7d3d584"),

      '1': ObjectId("650a78069bddeec3c7d3d585"),

      '2': ObjectId("650a78069bddeec3c7d3d586")

    }

  }

Solve the following queries:

1. Display the details of the Freshman whose entrance exam score is greater than 40.

Enterprise NoSQLMid> db.Freshman.find({ Score\_Entrance: { $gt: 40 } });

[

  {

    \_id: ObjectId("650a78069bddeec3c7d3d584"),

    Freshma\_Name: 'Smit Joshi',

    Address: 'Deesa',

    Area\_of\_intrest: 'Parita',

    Certification\_course: 'Java Hero',

    Age: 21,

    Score\_Entrance: 80

  },

  {

    \_id: ObjectId("650a78069bddeec3c7d3d585"),

    Freshma\_Name: 'Tejasv Modi',

    Address: 'Patan',

    Area\_of\_intrest: 'Web Desisning',

    Certification\_course: 'Web Developing',

    Age: 21,

    Score\_Entrance: 50

  }

]

1. Change the interest area of Parita to Analysis.

Enterprise NoSQLMid> db.Freshman.updateMany({ Area\_of\_intrest: "Parita" }, { $set: { Area\_of\_intrest: "Analysis" } });

{

  acknowledged: true,

  insertedId: null,

  matchedCount: 1,

  modifiedCount: 1,

  upsertedCount: 0

}

1. Display details of Freshman who are not living in Ahmedabad and Delhi (use nor operator).

Enterprise NoSQLMid> db.Freshman.find({ $nor: [

{ Address: "Ahmedabad" }, { Address: "Delhi" }

]});

[

  {

    \_id: ObjectId("650a78069bddeec3c7d3d584"),

    Freshma\_Name: 'Smit Joshi',

    Address: 'Deesa',

    Area\_of\_intrest: 'Analysis',

    Certification\_course: 'Java Hero',

    Age: 21,

    Score\_Entrance: 80

  },

  {

    \_id: ObjectId("650a78069bddeec3c7d3d585"),

    Freshma\_Name: 'Tejasv Modi',

    Address: 'Patan',

    Area\_of\_intrest: 'Web Desisning',

    Certification\_course: 'Web Developing',

    Age: 21,

    Score\_Entrance: 50

  },

  {

    \_id: ObjectId("650a78069bddeec3c7d3d586"),

    Freshma\_Name: 'Nisha',

    Address: 'Ranuj',

    Area\_of\_intrest: 'Dancing',

    Certification\_course: "Dancer's Things",

    Age: 22,

    Score\_Entrance: 30

  }

]

# SET 2

### Create the following tables (in MYSQL) and insert THREE records.

1. Publisher\_DETAILS (Publisher\_ID, Publisher\_Name, Publisher \_City)
2. ORDER (OrderNo, Order\_Date, OrderAmount)

-- CREATING TABLE Publisher\_DETAILS

CREATE TABLE Publisher\_DETAILS(

Publisher\_ID int primary key,

Publisher\_Name varchar(50),

Publisher\_City varchar(30)

);

-- CREATING TABLE ORDERS

CREATE TABLE ORDERS(

OrderNo int Primary Key,

Order\_Date Date,

OrderAmount DECIMAL(6,2)

);

-- INSERTING Three Records In Each ONE

INSERT INTO Publisher\_DETAILS(

Publisher\_ID,

Publisher\_Name,

Publisher\_City

) VALUES

(1,'Smit Joshi','Ahmedabad'),

(2,'Vijay Joshi','Mumbai'),

(3,'Jinal Patel','Surat');

INSERT INTO ORDERS(

OrderNo ,

Order\_Date ,

OrderAmount

) VALUES

(1, '2003-07-29',8999.99),

(2, '2003-08-02',899.99),

(3, '2003-09-09',4599.99);

-- 1. Add an additional field Publisher\_Email in Publisher\_DETAILS table.

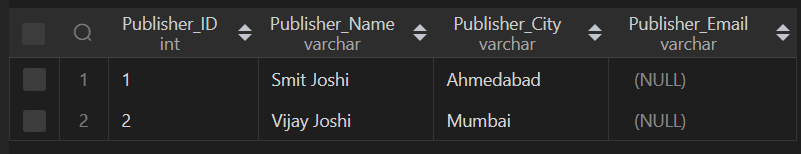
ALTER TABLE Publisher\_DETAILS ADD Publisher\_Email varchar(40);

-- 2. Change the OrderAmount to 5000 where OrderNo is 3

UPDATE ORDERS SET OrderAmount=5000 WHERE OrderNo=3;

-- 3. Display only those records from Publisher where the Publisher\_city is Ahmedabad or Mumbai. (Do not use OR )

SELECT \* FROM Publisher\_DETAILS WHERE Publisher\_city IN ('Ahmedabad','Mumbai');



### **MongoDB**

//Databse For NoSQL Queries

Enterprise NoSQLMid> Use NoSQLMid;

* Create the Employee Collection and insert THREE documents according to the instructions

(Name, Address, Experience, Department, Age, Rating)

Enterprise NoSQLMid> db.createCollection("Employee");

Enterprise NoSQLMid> db.Employee.insertMany([

{

        Name: "Laxman",

        Address: "Surat",

        Experience: "2 years",

        Department: "Purchase",

        Age: 21,

        Rating: 4

    },

    {

        Name: "Ram",

        Address: "Ayodhya",

        Experience: "1 Year",

        Department: "Sales",

        Age: 19,

        Rating: 9

    },

    {

        Name: "Hanuman",

        Address: "Ahmedabad",

        Experience: "2 years",

        Department: "Sales",

        Age: 22,

        Rating: 5

    }

]);

{

    acknowledged: true,

    insertedIds: {

      '0': ObjectId("650a834c9bddeec3c7d3d58d"),

      '1': ObjectId("650a834c9bddeec3c7d3d58e"),

      '2': ObjectId("650a834c9bddeec3c7d3d58f")

    }

  }

Solve the following queries:

1. Display the details of the Employee whose rating is greater than 5.

Enterprise NoSQLMid> db.Employee.find({ Rating: { $gt: 5 } });

[

  {

    \_id: ObjectId("650a834c9bddeec3c7d3d58e"),

    Name: 'Ram',

    Address: 'Ayodhya',

    Experience: '1 Year',

    Department: 'Sales',

    Age: 19,

    Rating: 9

  }

]

1. Change the department of Ram to Purchase.

Enterprise NoSQLMid> db.Employee.updateOne({ Name: "Ram" }, { $set: { Department: "Purchase" } });

{ acknowledged: true,

  insertedId: null,

  matchedCount: 1,

  modifiedCount: 1,

  upsertedCount: 0

}

1. Display details of employee who are not living in Ahmedabad and Surat (use nor operator).

Enterprise NoSQLMid> db.Employee.find({ $nor: [

{ Address: "Ahmedabad" }, { Address: "Surat" }

] });

[

  {

    \_id: ObjectId("650a834c9bddeec3c7d3d58e"),

    Name: 'Ram',

    Address: 'Ayodhya',

    Experience: '1 Year',

    Department: 'Purchase',

    Age: 19,

    Rating: 9

  }

]

# SET 3

### Create the following tables (in MYSQL) and insert THREE records.

1. PATIENT (Patient\_ID, Patient\_Name, Patient\_Age, Charges, Patient\_Gender)
2. DOCTOR\_DETAILS(Doct\_ID, DoctFName, salary, D\_Age,Experience)

-- Creating table PATIENT

CREATE TABLE PATIENT(

Patient\_ID int primary Key, Patient\_Name varchar(50),

Patient\_Age DECIMAL(2), Charges DECIMAL(6,2),

Patient\_Gender VARCHAR(6) );

-- Creating table DOCTOR\_DETAILS

CREATE TABLE DOCTOR\_DETAILS(

Doct\_ID int primary key, DoctFName VARCHAR(50), salary DECIMAL(7,2),

D\_Age DECIMAL(2), Experience DECIMAL(2) );

-- Inserting Three Records In Each ONE

INSERT INTO PATIENT(

Patient\_ID,

Patient\_Name,

Patient\_Age,

Charges,

Patient\_Gender

)VALUES

(1, 'Vikas',28,999.99,'Male'),

(2, 'Sakshi',30,499.99,'Female'),

(3, 'Sarthak',17,899.99,'Male');

INSERT INTO DOCTOR\_DETAILS(

Doct\_ID,

DoctFName,

salary,

D\_Age,

Experience

)VALUES

(1,'Drashti',20000,29,2),

(2,'Bhavik',10000,39,10),

(3,'Vijay',20000,48,18);

-- 1. Add an additional field DoctLName in DOCTOR\_DETAILS table.

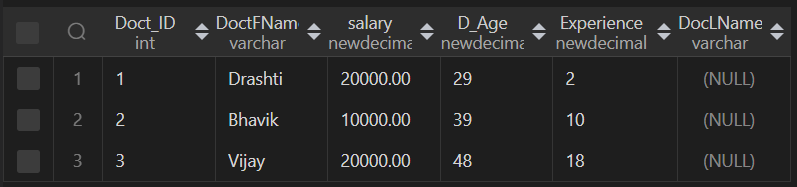
ALTER TABLE DOCTOR\_DETAILS ADD DocLName varchar(20);

-- 2. Change the Charges to 2000 where Patient\_ID is 3.

UPDATE PATIENT SET Charges=2000 WHERE Patient\_ID=3;

-- 3. Display only those records from DOCTOR\_DETAILS where the salary is 10000 or 20000. (Do not use OR)

SELECT \* FROM DOCTOR\_DETAILS WHERE salary IN(10000,20000);



### **MongoDB**

//Databse For NoSQL Queries

Enterprise NoSQLMid> Use NoSQLMid;

* Create the Seller Collection and insert THREE documents according to the instructions. (Name, Address, Products\_Supplied, T\_C, Age, Type, Total\_Order, Pending\_order)

Enterprise NoSQLMid> db.createCollection("Selller");

Enterprise NoSQLMid> db.Seller.insertMany([{

        Name: "Smit",

        Address: "Ahmedabad",

        Products\_Supplied: ["Brush", "Colgate", "Buiscuits"],

        T\_C: true,

        Age: 28,

        Type: "x type",

        Total\_Order: 1200,

        Pending\_order: 100

    },

    {

        Name: "Switi",

        Address: "Bhavnagar",

        Products\_Supplied: ["Chocolets", "Milk", "Buiscuits"],

        T\_C: truer,

        Age: 21,

        Type: "y type",

        Total\_Order: 100,

        Pending\_order: 50

    },

    {

        Name: "Tejasv",

        Address: "Surat",

        Products\_Supplied: ["Cosmatics", "Hurbal", "Shempoo"],

        T\_C: true,

        Age: 25,

        Type: "z type",

        Total\_Order: 200,

        Pending\_order: 10

    }

]);

{

    acknowledged: true,

    insertedIds: {

      '0': ObjectId("650b09b2a6ac83287629014d"),

      '1': ObjectId("650b09b2a6ac83287629014e"),

      '2': ObjectId("650b09b2a6ac83287629014f")

    }

}

Solve the following queries:

1. Display the details of the Seller whose age is greater than 25.

Enterprise NoSQLMid> db.Seller.find({ Age: { $gt: 25 } });

[

  {

    \_id: ObjectId("650b09b2a6ac83287629014d"),

    Name: 'Smit',

    Address: 'Ahmedabad',

    Products\_Supplied: [ 'Brush', 'Colgate', 'Buiscuits' ],

    T\_C: true,

    Age: 28,

    Type: 'x type',

    Total\_Order: 1200,

    Pending\_order: 100

  }

]

1. Change the T\_C of Sumit to “Allow\_Credit”.

Enterprise NoSQLMid> db.Seller.updateMany({ T\_C: true },

{ $rename: { "T\_C": "Allow\_Credit" } });

{

  acknowledged: true,

  insertedId: null,

  matchedCount: 3,

  modifiedCount: 3,

  upsertedCount: 0

}

1. Display details of Seller who are not living in Ahmedabad and Bhavnagar (use nor operator).

Enterprise NoSQLMid> db.Seller.find({ $nor: [ { Address: "Ahmedabad" },

{ Address: "Bhavnagar" }] });

[

  {

    \_id: ObjectId("650b10d0a6ac83287629015f"),

    Name: 'Tejasv',

    Address: 'Surat',

    Products\_Supplied: [ 'Cosmatics', 'Hurbal', 'Shempoo' ],

    Age: 25,

    Type: 'z type',

    Total\_Order: 200,

    Pending\_order: 10,

    Allow\_Credit: true

  }

]

# SET 4

### Create the following tables (in MYSQL) and insert THREE records.

### ARTIST (A\_ID,A\_NAME,A\_AGE,A\_EXPERIENCE, A\_Salary,A\_gender,email)

### Painting (Painting\_ID, PaintingName, Category, Price)

-- CREATING ARTIST TABLE

CREATE TABLE ARTIST(

    A\_ID INT PRIMARY KEY,

    A\_NAME VARCHAR(50),

    A\_AGE VARCHAR(50),

    A\_EXPERIENCE DECIMAL(2),

    A\_Salary DECIMAL(7,2),

    A\_gender VARCHAR(6),

    email VARCHAR(50)

);

-- CREATING PAINTING TABLE

CREATE TABLE Painting(

    Painting\_ID int PRIMARY key,

    PaintingName VARCHAR(50),

    Category VARCHAR(30),

    Price DECIMAL(7,2)

);

-- INSERTING THREE RECORDS IN ECH ONE

INSERT INTO ARTIST(

A\_ID,

A\_NAME,

A\_AGE,

A\_EXPERIENCE,

A\_Salary,

A\_gender,

email

)VALUES

(1,'Smit Joshi',21,2,70000,'Male','smitjoshi@gmail.com'),

(2,'Vishva Joshi',20,1,30000,'Female','vaishujoshi@gmail.com'),

(3,'Vijay Joshi',27,4,90000,'Male','joshivijay34@gmail.com');

INSERT INTO Painting(

Painting\_ID, PaintingName,

Category, Price ) VALUES

(101,'Radha Shyam','Abstract art',90000),

(102,'Wild Life','Oil painting',50000),

(103,'Peace','Spray Painting',40000);

-- 1. Add an additional field City in ARTIST table.

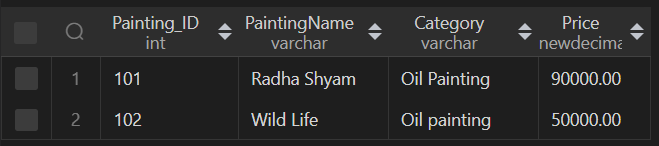
ALTER TABLE ARTIST ADD City varchar(20);

-- 2. Change the Category to Oil painting where Painting\_ID is 101.

UPDATE Painting SET Category="Oil Painting" WHERE Painting\_ID=101;

-- 3. Display only those records from Painting where the Category is Oil painting or Abstract art.(Do not use OR)

SELECT \* FROM Painting WHERE Category IN("Oil Painting","Abstract art");



### **MongoDB**

//Databse For NoSQL Queries

Enterprise NoSQLMid> Use NoSQLMid;

* Create the Dealer Collection and insert THREE documents according to the instructions. (Name, Address, Product\_name, T\_C, Age, Amount)

Enterprise NoSQLMid>  db.createCollection("Dealer");

Enterprise NoSQLMid>  db.Dealer.insertMany([{

        Name: "Shyam",

        Address: "Deesa",

        Product\_name: "Dairy Milk",

        T\_C: true,

        Age: 21,

        Amount: 5000

    },

    {

        Name: "Smit Joshi",

        Address: "Rajkot",

        Product\_name: "Parle-G",

        T\_C: true,

        Age: 21,

        Amount: 15500

    },

    {

        Name: "Drashti Joshi",

        Address: "Ahmedabad",

        Product\_name: "Bread",

        T\_C: true,

        Age: 20,

        Amount: 2000

    }

]);

{

    acknowledged: true,

    insertedIds: {

      '0': ObjectId("650b1a926bd320393a123ff0"),

      '1': ObjectId("650b1a926bd320393a123ff1"),

      '2': ObjectId("650b1a926bd320393a123ff2")

    }

  }

Solve the following queries:

1. Display the details of the Dealer whose amount is greater than 15000.

Enterprise NoSQLMid> db.Dealer.find({ Amount: { $gt: 15000 } });

[

  {

    \_id: ObjectId("650b1a926bd320393a123ff1"),

    Name: 'Smit Joshi',

    Address: 'Rajkot',

    Product\_name: 'Parle-G',

    T\_C: true,

    Age: 21,

    Amount: 15500

  }

1. Change the Adress of Dealer name Shyam to Pune.

Enterprise NoSQLMid> db.Dealer.updateOne({ Name: "Shyam" }, { $set: { Address: "Pune" } });

{

  acknowledged: true,

  insertedId: null,

  matchedCount: 1,

  modifiedCount: 1,

  upsertedCount: 0

}

1. Display details of Dealer who are not living in Rajkot and Ahmedabad (use nor operator).

Enterprise NoSQLMid> db.Dealer.find({ $nor: [ { Address: "Ahmedabad" }, { Address: "Rajkot" }] });

[

  {

    \_id: ObjectId("650b1a926bd320393a123ff0"),

    Name: 'Shyam',

    Address: 'Pune',

    Product\_name: 'Dairy Milk',

    T\_C: true,

    Age: 21,

    Amount: 5000

  }

]

# SET 5

### Create the following tables (in MYSQL) and insert THREE records.

### TRANSACTION\_DETAILS (TransactionID, AccountNo,TransactionType, TransactionAmount)

### ACCOUNT\_MASTER (AccountNo, AccountHolderName, A\_City, AccountType, AccountBalance)

-- CREATING TABLE ACCOUNT\_MASTER

CREATE TABLE ACCOUNT\_MASTER(

    AccountNo int primary KEY,

    AccountHoldername VARCHAR(50),

    A\_City VARCHAR(30),

    AccountType VARCHAR(10),

    AccountBalance DECIMAL(7,2)

);

-- CREATING TABLE TRANSACTION\_DETAILS

CREATE TABLE TRANSACTION\_DETAILS(

    TransactionID int PRIMARY KEY,

    AccountNo int,

    TransactionType VARCHAR(10),

    TransactionAmount DECIMAL(6,4),

    Foreign Key (AccountNo) REFERENCES ACCOUNT\_MASTER(AccountNo)

);

-- INSERTING THREE RECORDS IN EACH TABLE

INSERT INTO ACCOUNT\_MASTER(

    AccountNo,

    AccountHoldername,

    A\_City,

    AccountType,

    AccountBalance

)VALUES

(101,"Rakesh","Deesa","saving",99999.99),

(102,"Suresh","Ahmedabad","current",7999.99),

(103,"Smit Joshi","Deesa","saving",28999.99);

INSERT INTO TRANSACTION\_DETAILS(

    TransactionID,

    AccountNo,

    TransactionType,

    TransactionAmount

) VALUES (1,101,"UPI",3000),(2,102,"Cash",10000),(3,103,"Credit Card",6000);

-- 1. Add an additional field TransactionDate in TRANSACTION\_DETAILS table.

ALTER TABLE TRANSACTION\_DETAILS ADD TransactionDate DATE;

-- 2. Change the status of AccountType to “Current” where AccountNo is 101.

UPDATE ACCOUNT\_MASTER SET AccountType="current" WHERE AccountNo=101;

-- 3. Display only those records from ACCOUNT\_MASTER where the name is “Rakesh” or “Suresh”. (Do not use OR)

SELECT \* FROM ACCOUNT\_MASTER WHERE AccountHolderName IN ("Rakesh","Suresh");



### **MongoDB**

//Databse For NoSQL Queries

Enterprise NoSQLMid> Use NoSQLMid;

* Create the Associate Collection and insert THREE documents according to the instructions.

(Name, Address, Area\_of\_Expertise, Certification\_course, Age, Score\_Entrance)

Enterprise NoSQLMid> db.createCollection("Associate");

Enterprise NoSQLMid> db.Associate.insertMany([{

        Name: "Smita",

        Address: "Surat",

        Area\_of\_Expertise: "Wed Development",

        Certification\_course: "Web Dev+",

        Age: 21,

        Score\_Entrance: 80

    },

    {

        Name: "Switi",

        Address: "Surat",

        Area\_of\_Expertise: "Java Developer",

        Certification\_course: "Java Zero To Hero",

        Age: 32,

        Score\_Entrance: 89

    },

    {

        Name: "Vijay",

        Address: "Rajkot",

        Area\_of\_Expertise: "Accounting",

        Certification\_course: "Accounting Masters",

        Age: 30,

        Score\_Entrance: 60

    }

]);

{

    acknowledged: true,

    insertedIds: {

      '0': ObjectId("650b246b6bd320393a123ff8"),

      '1': ObjectId("650b246b6bd320393a123ff9"),

      '2': ObjectId("650b246b6bd320393a123ffa")

    }

}

1. Display the details of Associate whose age is less than 30.

Enterprise NoSQLMid> db.Associate.find({ Age: { $lt: 30 } });

[

  {

    \_id: ObjectId("650b246b6bd320393a123ff8"),

    Name: 'Smita',

    Address: 'Surat',

    Area\_of\_Expertise: 'Wed Development',

    Certification\_course: 'Web Dev+',

    Age: 21,

    Score\_Entrance: 80

  }

]

1. Change the address of Smita to Ahmedabad.

Enterprise NoSQLMid> db.Associate.updateOne({ Name: "Smita" }, { $set: { Address: "Ahmedabad" } });

{

  acknowledged: true,

  insertedId: null,

  matchedCount: 1,

  modifiedCount: 1,

  upsertedCount: 0

}

1. Display details of Associate who are not living in Rajkot and Surat (use nor operator)

Enterprise NoSQLMid> db.Associate.find({ $nor: [ { Address: "Rajkot" }, { Address: "Surat" }] });

[

  {

    \_id: ObjectId("650b246b6bd320393a123ff8"),

    Name: 'Smita',

    Address: 'Ahmedabad',

    Area\_of\_Expertise: 'Wed Development',

    Certification\_course: 'Web Dev+',

    Age: 21,

    Score\_Entrance: 80

  }

]

Finished