.

**Part A. Create a Bookstore Document Database**

**A1. Solution to First Part**

* **Schema Design for the BookStore Application**

Given schema is based on the following points

1. Books description
2. Sample images for front pages
3. User Comments

There are three tables in it

1. User Tables
2. Books Table
3. Comments Tables

* User tables consists of username, first name, last name, email, and password.
* Books table is based on book title, author name, description, short abstract, price, link to cover image, and link to sample pages for quality.
* Comments table is based on information regarding user ID and comments and date of comment.
* **Example Documents**

1. **Add\_Book Collection**

{

bookid: "1x11111c",

book\_title: "Programming with R",

book\_category: "Education",

book\_author: "Mark Tedd",

book\_pub\_year: 1982,

books\_in\_stock: 1589,

book\_description: "Programming language exclusively designed for undergraduate students",

samplePagesImages: ["https://sso.gcu.ac.uk/idp/SSO.saml2",

" https://sso.gcu.ac.uk/idp/SSO.saml2"]

}

1. **Add\_User Collection**

{

"\_id": ObjectId("112233445566778899001234"),

“userid”: “12a12345-12b4-1a2b-9876-95278bc981d8”

"user\_name "GCU",

"first\_name "FirstName",

"last\_name "SecondName",

"email": "gcu.123@gmail.com",

"password": "gcustudent1122"

}

1. **Comments Collection Example Document**

{

review\_id: "1",

bookid: "1x11111c ",

userid: "12a12345-12b4-1a2b-9876-95278bc981d8”

date\_of\_review: new Date("2023-03-12"),

review\_text: "this book is a state-of-the-art contribution to learn"

}

* **Justification for Schema Design of the BookStore Application**

The specifications of the bookstore application are used to guide the selection of collections and data modelling techniques. The books collection includes details about the books i.e., book title, book category, author name of the book, publication year, stock details and few more details. Meanwhile, it also saves user and comments (review) details. In this regards it is obsivious no other details are required for the books. If we look into the some bookstore applications it can be observed that they are only based on this data domain. In this case all the required information can be easily saved.

**A2. Database creation for the bookstore application**

As per schema defined in the part A1. I will first enter 6 users and 10 books. Moreover, 6 books will have reviews (comments with them)

**Code:**

use('cw1PartA')

**Output:**

Graphical user interface, text, application

Description automatically generated

**Code:**

* Creating “add\_user” collection

db.createCollection("add\_users")

**Output:**

Graphical user interface, text, application

Description automatically generated

* **Inserting 6 Users in BookStore Application**

db.add\_users.insertMany([

{

"userid": "12a12345-12b4-1a2b-9876-95278bc981d8",

"user\_name": "GCU1",

"first\_name":"FirstName1",

"last\_name": "SecondName1",

"email": "gcu1.123@gmail.com",

"password": "1gcustudent1122"

},{

"userid": "12a12345-12b4-1a2b-9876-95278bc981d9",

"user\_name": "GCU2",

"first\_name":"FirstName2",

"last\_name": "SecondName2",

"email": "gcu2.123@gmail.com",

"password": "2gcustudent1122"

},{

"userid": "12a12345-12b4-1a2b-9876-95278bc981d1",

"user\_name": "GCU3",

"first\_name":"FirstName3",

"last\_name": "SecondName3",

"email": "gcu3.123@gmail.com",

"password": "3gcustudent1122"

},{

"userid": "12a12345-12b4-1a2b-9876-95278bc981d2",

"user\_name": "GCU4",

"first\_name":"FirstName4",

"last\_name": "SecondName4",

"email": "gcu4.123@gmail.com",

"password": "4gcustudent1122"

},{

"userid": "12a12345-12b4-1a2b-9876-95278bc981d3",

"user\_name": "GCU5",

"first\_name":"FirstName5",

"last\_name": "SecondName5",

"email": "gcu5.123@gmail.com",

"password": "5gcustudent1122"

},{

"userid": "12a12345-12b4-1a2b-9876-95278bc981d48",

"user\_name": "GCU6",

"first\_name":"FirstName6",

"last\_name": "SecondName6",

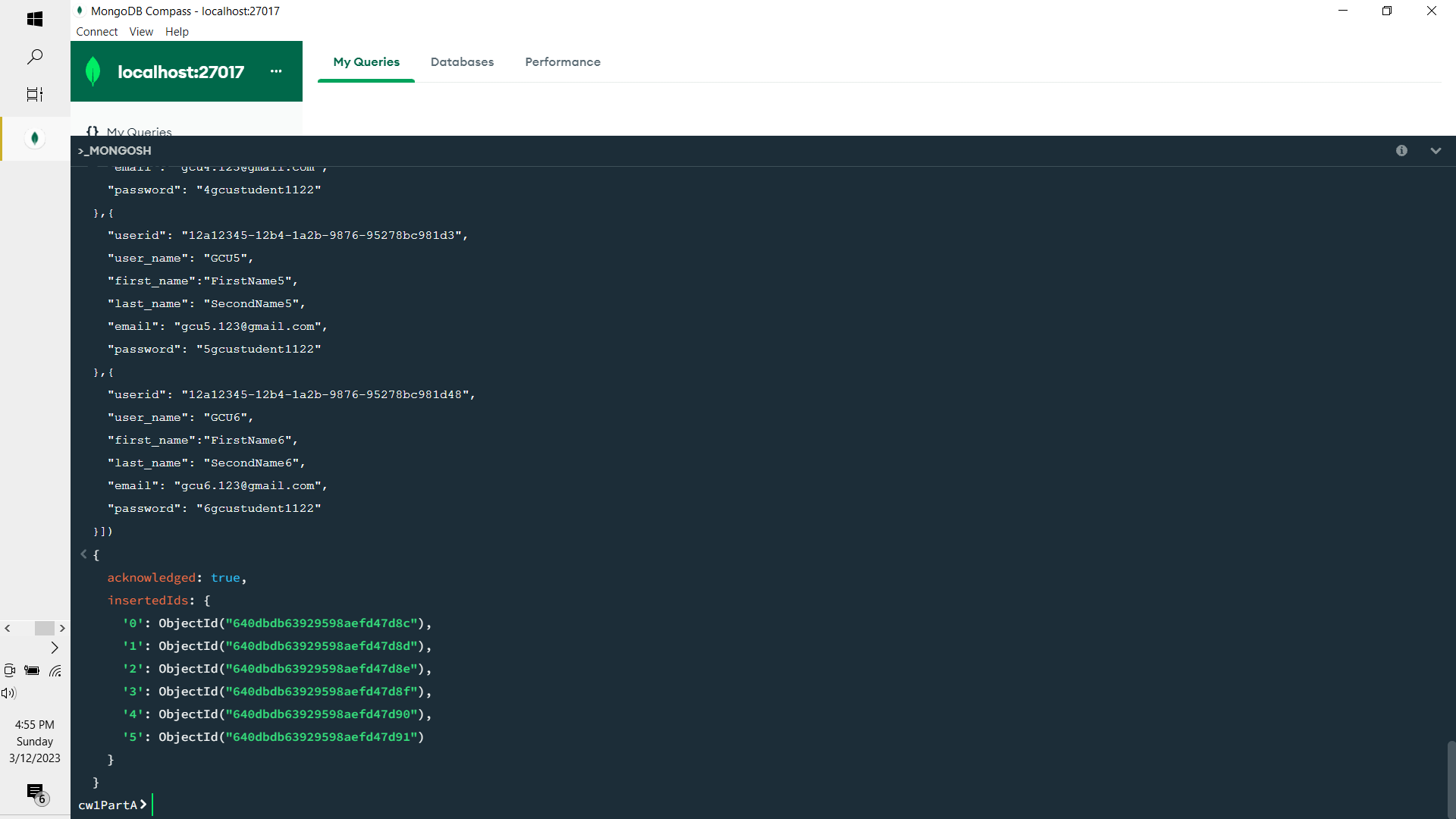
"email": "gcu6.123@gmail.com",

"password": "6gcustudent1122"

}

])

**Output**



* Creating “add\_user” collection

**Code**

db.createCollection("add\_books")

**Output**

Text

Description automatically generated

**Code:**

db.add\_books.insertMany([

{

"bookid": "2x11111c",

"book\_title": "Programming with R",

"book\_category": "Education",

"book\_author": "Mark Tedd",

"book\_pub\_year": 1982,

"books\_in\_stock": 1589,

"book\_description": "Programming language exclusively designed for undergraduate students",

"samplePagesImages": ["https://sso.gcu.ac.uk/idp/SSO.jpg",

" https://sso.gcu.ac.uk/idp/SSO.jpg"]

},{

"bookid": "3x11111c",

"book\_title": "Getting Started with IoTs",

"book\_category": "Education",

"book\_author": "Henry David",

"book\_pub\_year": 2005,

"books\_in\_stock": 400,

"book\_description": "IoT will revolutionise the era, get this book to know more",

"samplePagesImages": ["https://sso.gcu.ac.uk/idp/SSO.jpg",

" https://sso.gcu.ac.uk/idp/SSO.jpg"]

},{

"bookid": "4x11111c",

"book\_title": "Tales of Love and Joy",

"book\_category": "Fantacy",

"book\_author": "Julia Elexender",

"book\_pub\_year": 1989,

"books\_in\_stock": 35,

"book\_description": "A bedtime novel based on two characters from scotland",

"samplePagesImages": ["https://sso.gcu.ac.uk/idp/SSO.jpg",

" https://sso.gcu.ac.uk/idp/SSO.jpg"]

},{

"bookid": "5x11111c",

"book\_title": "The Modern Era and Big Data",

"book\_category": "Science",

"book\_author": "Elon Scarlet",

"book\_pub\_year": 2013,

"books\_in\_stock": 14,

"book\_description": "A book based on the modern trends of bigdata",

"samplePagesImages": ["https://sso.gcu.ac.uk/idp/SSO.jpg",

" https://sso.gcu.ac.uk/idp/SSO.jpg"]

},{

"bookid": "6x11111c",

"book\_title": "Rise and Fall of etrading",

"book\_category": "Science",

"book\_author": "Mark Tedd",

"book\_pub\_year": 2018,

"books\_in\_stock": 10,

"book\_description": "Etrading is dying! read the book to know more",

"samplePagesImages": ["https://sso.gcu.ac.uk/idp/SSO.jpg",

" https://sso.gcu.ac.uk/idp/SSO.jpg"]

},{

"bookid": "7x11111c",

"book\_title": "Halloween in COVID-19",

"book\_category": "Fantacy",

"book\_author": "Julia Elexender",

"book\_pub\_year": 2021,

"books\_in\_stock": 16,

"book\_description": "In Covid-19, everything changed. Read more!",

"samplePagesImages": ["https://sso.gcu.ac.uk/idp/SSO.jpg",

" https://sso.gcu.ac.uk/idp/SSO.jpg"]

},{

"bookid": "8x11111c",

"book\_title": "To whom it may concern!",

"book\_category": "Science",

"book\_author": "Elon Scarlet",

"book\_pub\_year": 2014,

"books\_in\_stock": 100,

"book\_description": "In Covid-19, everything changed. Read more!",

"samplePagesImages": ["https://sso.gcu.ac.uk/idp/SSO.jpg",

" https://sso.gcu.ac.uk/idp/SSO.jpg"]

},{

"bookid": "9x11111c",

"book\_title": "SampleBook1",

"book\_category": "Education",

"book\_author": "Mark Tedd",

"book\_pub\_year": 1942,

"books\_in\_stock": 1589,

"book\_description": "In Covid-19, everything changed. Read more!",

"samplePagesImages": ["https://sso.gcu.ac.uk/idp/SSO.jpg",

" https://sso.gcu.ac.uk/idp/SSO.jpg"]

},{

"bookid": "10x11111c",

"book\_title": "SampleBook2",

"book\_category": "Education",

"book\_author": "Mark Tedd",

"book\_pub\_year": 1952,

"books\_in\_stock": 1589,

"book\_description": "In Covid-19, everything changed. Read more!",

"samplePagesImages": ["https://sso.gcu.ac.uk/idp/SSO.jpg",

" https://sso.gcu.ac.uk/idp/SSO.jpg"]

},{

"bookid": "11x11111c",

"book\_title": "SampleBook4",

"book\_category": "Education",

"book\_author": "Mark Tedd",

"book\_pub\_year": 1962,

"books\_in\_stock": 1589,

"book\_description": "In Covid-19, everything changed. Read more!",

"samplePagesImages": ["https://sso.gcu.ac.uk/idp/SSO.jpg",

" https://sso.gcu.ac.uk/idp/SSO.jpg"]

},{

"bookid": "112x11111c",

"book\_title": "SampleBook3",

"book\_category": "Education",

"book\_author": "Mark Tedd",

"book\_pub\_year": 1992,

"books\_in\_stock": 1589,

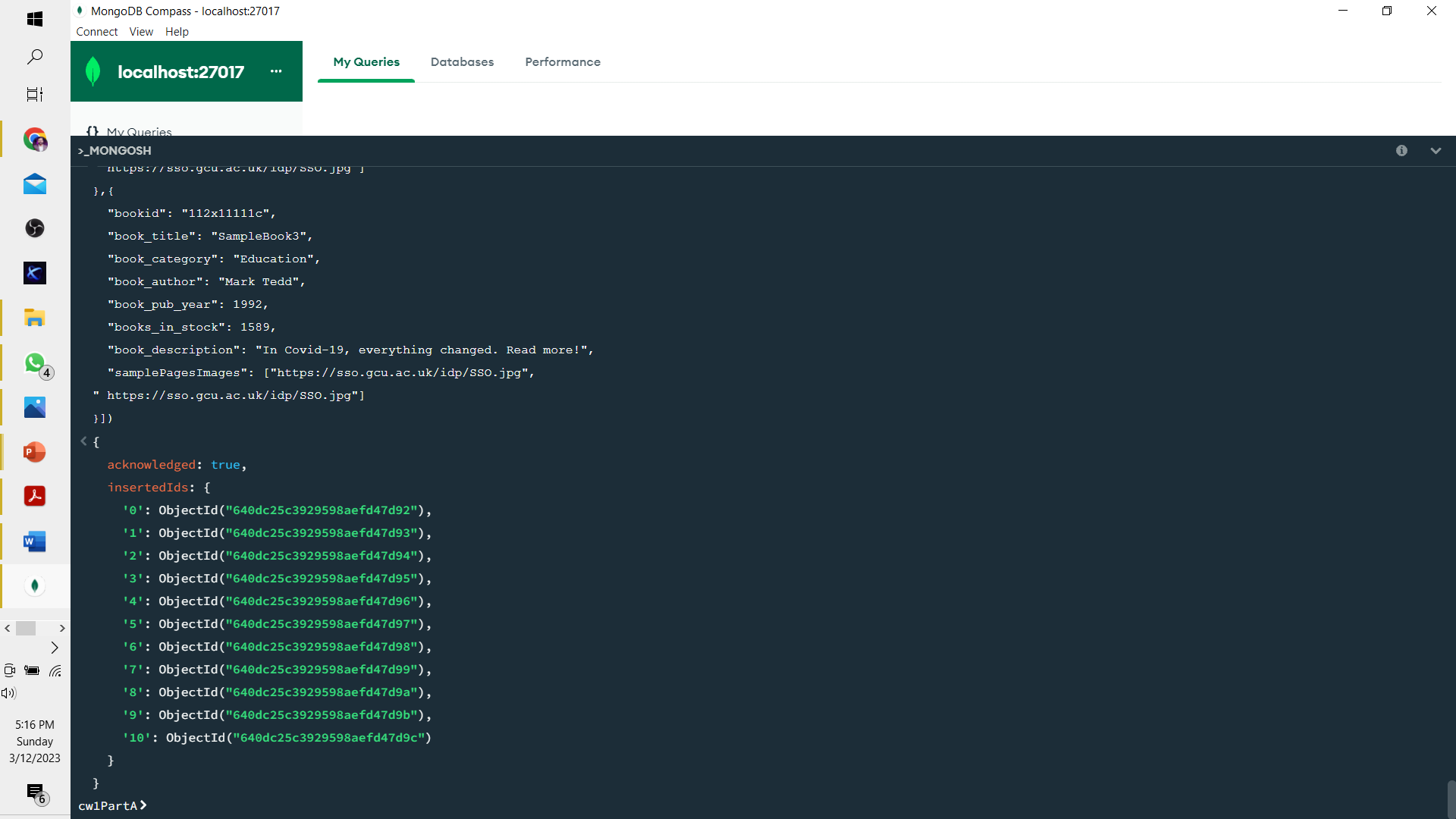
"book\_description": "In Covid-19, everything changed. Read more!",

"samplePagesImages": ["https://sso.gcu.ac.uk/idp/SSO.jpg",

" https://sso.gcu.ac.uk/idp/SSO.jpg"]

}])

**Output**

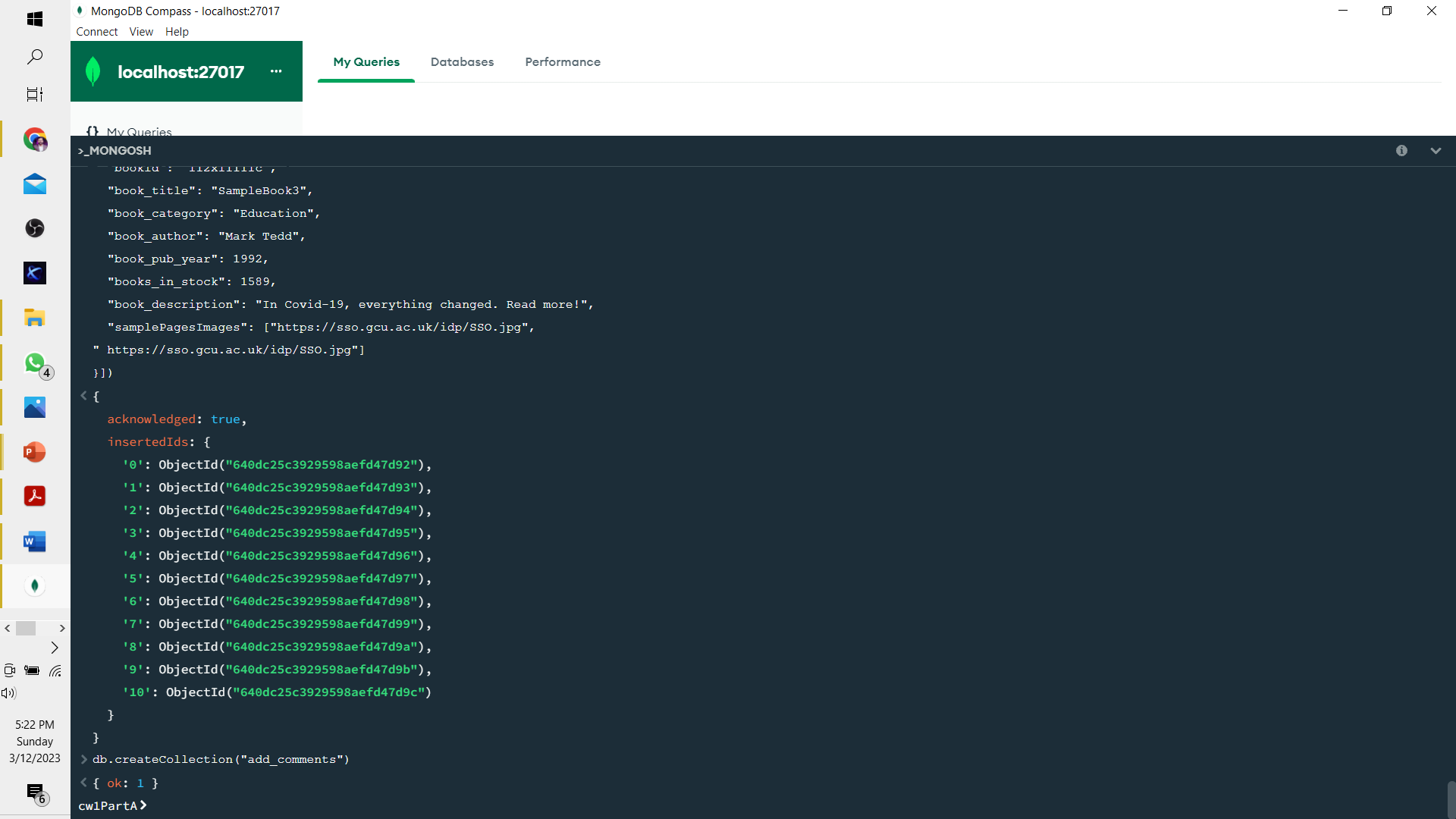


* Creating Comment Collections

**Code:**

db.createCollection("add\_comments")

**Output:**



**Code:**

db.add\_comments.insertMany([{

"review\_id": "1", "bookid": "1x11111c",

"userid": "12a12345-12b4-1a2b-9876-95278bc981d8”",

date: Date("2023-03-10"),

"review\_text": "Very good book"

},{

"review\_id": "2", "bookid": "1x11111c",

"userid": "12a12345-12b4-1a2b-9876-95278bc981d9”",

date: Date("2023-03-10"),

"review\_text": "Very good book dear"

},{

"review\_id": "3",

"bookid": "2x11111c",

"userid": "12a12345-12b4-1a2b-9876-95278bc981d1”",

date: Date("2023-03-10"),

"review\_text": "Very good book"

},{

"review\_id": "4",

"bookid": "2x11111c",

"userid": "12a12345-12b4-1a2b-9876-95278bc981d2”",

date: Date("2023-03-10"),

"review\_text": "Very good book"

},{

"review\_id": "5", "bookid": "3x11111c",

"userid": "12a12345-12b4-1a2b-9876-95278bc981d3”",

date: Date("2023-03-10"),

"review\_text": "Very good book"

},{

"review\_id": "6",

"bookid": "3x11111c",

"userid": "12a12345-12b4-1a2b-9876-95278bc981d48”",

date: Date("2023-03-10"),

"review\_text": "Very good book"

},{

"review\_id": "7",

"bookid": "4x11111c",

"userid": "12a12345-12b4-1a2b-9876-95278bc981d3”",

date: Date("2023-03-10"),

"review\_text": "Very good book"

},{

"review\_id": "8",

"bookid": "5x11111c",

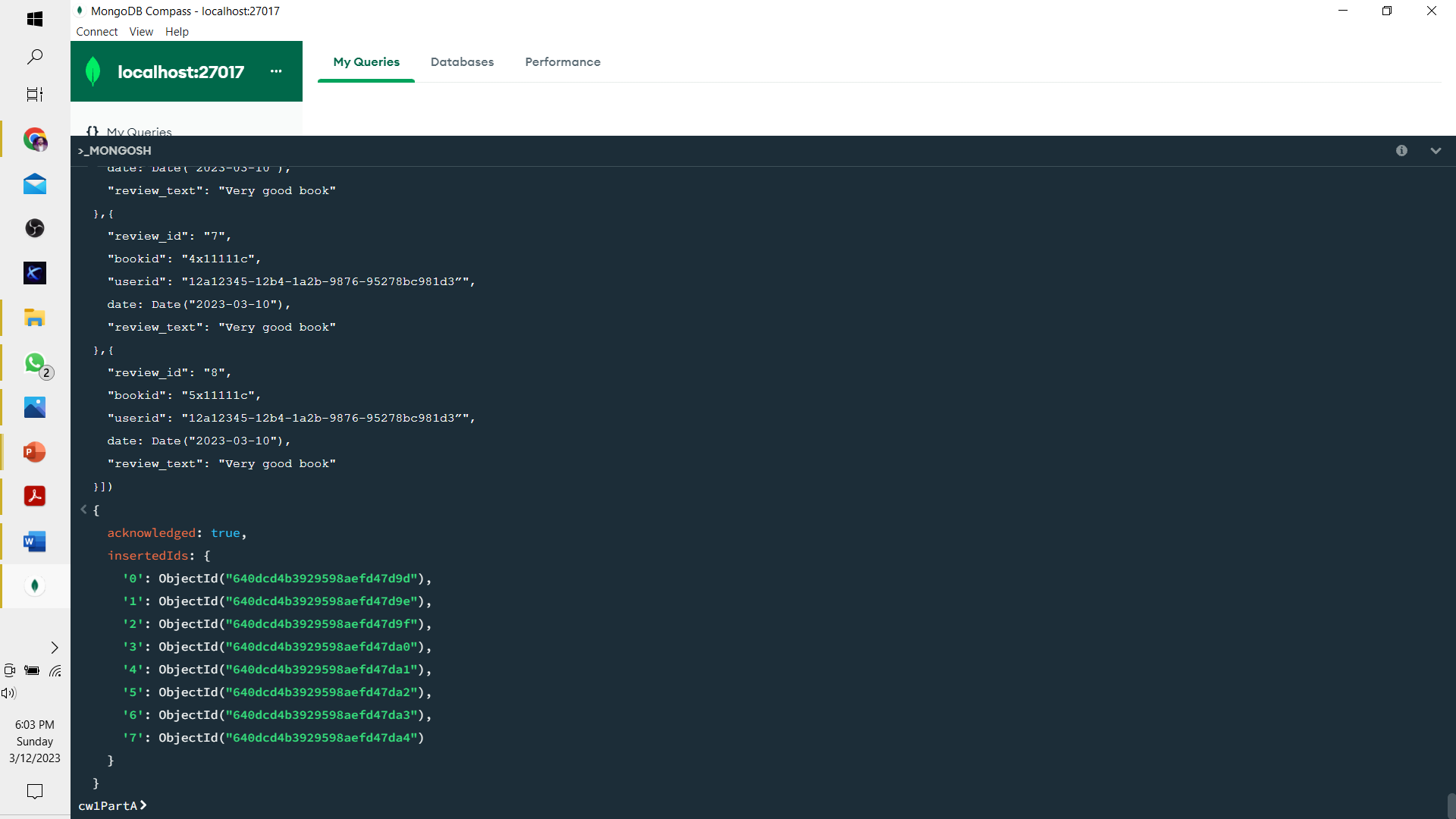
"userid": "12a12345-12b4-1a2b-9876-95278bc981d3”",

date: Date("2023-03-10"),

"review\_text": "Very good book"

}])

**Output**



**A3. Queries**

1. **Query 1:**

db.add\_books.aggregate([{

$match:{book\_category: "Education",

$expr: {

$and: [{ $gte: [ { $toInt: "$book\_pub\_year " }, 1940 ] },

{ $lte: [ { $toInt: "$book\_pub\_year " }, 2023 ] }]

}}}])

**Output:**

{

\_id: ObjectId('640dc25c3929598aefd47d92'),

bookid: '2x11111c',

book\_title: 'Programming with R',

book\_category: 'Education',

book\_author: 'Mark Tedd',

book\_pub\_year: 1982,

books\_in\_stock: 1589,

book\_description: 'Programming language exclusively designed for undergraduate students',

samplePagesImages: ['https://sso.gcu.ac.uk/idp/SSO.jpg',

'https://sso.gcu.ac.uk/idp/SSO.jpg']

},{

\_id: ObjectId('640dc25c3929598aefd47d93'),

bookid: '3x11111c',

book\_title: 'Getting Started with IoTs',

book\_category: 'Education',

book\_author: 'Henry David',

book\_pub\_year: 2005,

books\_in\_stock: 400,

book\_description: 'IoT will revolutionise the era, get this book to know more',

samplePagesImages: ['https://sso.gcu.ac.uk/idp/SSO.jpg',

'https://sso.gcu.ac.uk/idp/SSO.jpg']}

},{

\_id: ObjectId('640dc25c3929598aefd47d99'),

bookid: '9x11111c',

book\_title: 'SampleBook1'',

book\_category: 'Education',

book\_author: 'Mark Tedd',

book\_pub\_year: 1942,

books\_in\_stock: 1589,

book\_description: 'In Covid-19, everything changed. Read more',

samplePagesImages: ['https://sso.gcu.ac.uk/idp/SSO.jpg',

'https://sso.gcu.ac.uk/idp/SSO.jpg']}

},{

\_id: ObjectId('640dc25c3929598aefd47d9a'),

bookid: '10x11111c',

book\_title: 'SampleBook2', book\_category: 'Education',

book\_author: 'Mark Tedd', book\_pub\_year: 1952,

books\_in\_stock: 1589,

book\_description: 'In Covid-19, everything changed. Read more',

samplePagesImages: ['https://sso.gcu.ac.uk/idp/SSO.jpg',

'https://sso.gcu.ac.uk/idp/SSO.jpg']}

},{

\_id: ObjectId('640dc25c3929598aefd47d9b'),

bookid: '11x11111c',

book\_title: 'SampleBook4', book\_category: 'Education',

book\_author: 'Mark Tedd', book\_pub\_year: 1962,

books\_in\_stock: 1589,

book\_description: 'In Covid-19, everything changed. Read more',

samplePagesImages: ['https://sso.gcu.ac.uk/idp/SSO.jpg',

'https://sso.gcu.ac.uk/idp/SSO.jpg']}

},{

\_id: ObjectId('640dc25c3929598aefd47d9c'),

bookid: '12x11111c',

book\_title: 'SampleBook3', book\_category: 'Education',

book\_author: 'Mark Tedd', book\_pub\_year: 1992,

books\_in\_stock: 1589,

book\_description: 'In Covid-19, everything changed. Read more',

samplePagesImages: ['https://sso.gcu.ac.uk/idp/SSO.jpg',

'https://sso.gcu.ac.uk/idp/SSO.jpg']}

}])

1. **Query 2:**

db.add\_book.find([book\_author: "Henry David"])

Graphical user interface, text, application

Description automatically generated

1. **Query 3:**

db.add\_comments.insert({

review\_id: "9",

bookid: "5x11111c",

userid: "12a12345-12b4-1a2b-9876-95278bc981d48",",

date: Date(),

text: "I am adding new comment"

})

Graphical user interface, text, application

Description automatically generated

1. **Query 4:**

db.add\_books.aggregate([

{ $match: { book\_description: /IoT will revolutionise the era, get this book to know more /i } }, {

$lookup: {

from: "add\_comments", localField: "\_id",

foreignField: "review\_id", as: "comments"

} }, {

$unwind: { path: "$add\_comments", preserveNullAndEmptyArrays: true } },

{

$lookup: {

from: "add\_users", localField: "userid",

foreignField: "userid", as: "add\_comments.user"

}},{ $group: {

\_id: "$\_id", book\_title: { $first: "$book\_title" },

book\_author: { $first: "$book\_author" },

book\_description: { $first: "$book\_description" },

book\_category: { $first: "$book\_category" },

samplePagesImages: { $first: "$samplePagesImages " },

comments: { $push: "$add\_comments" } } }])

**Output:**

{

"bookid": "3x11111c",

"book\_title": "Getting Started with IoTs",

"book\_category": "Education",

"book\_author": "Henry David",

"book\_pub\_year": 2005,

"books\_in\_stock": 400,

"book\_description": "IoT will revolutionise the era, get this book to know more",

"samplePagesImages": ["https://sso.gcu.ac.uk/idp/SSO.jpg",

" https://sso.gcu.ac.uk/idp/SSO.jpg"]

}

**Part B. Graph Database Creation**

**B1. Cypher Commands for Database**

1. **Cypher:**

CREATE DATABASE cw1PartB

1. **Cypher:**

Use cw1PartB

1. **Airport Nodes:**
2. **Cypher:**

CREATE (:Def\_airport {name: "Qatar"})

CREATE (:Def\_airport {name: "Saudi Air"})

CREATE (:Def\_airport {name: "SereneAir"})

CREATE (:Def\_airport {name: "AirSial"})

CREATE (:Def\_airport {name: "FlyToday"})

**Output:**

Added 5 labels, created 5 nodes, set 5 properties, completed after 20 ms.

1. **Flight nodes:**
2. **Cypher:**

CREATE (:Def\_Flight {def\_airline: "A1", date: "2022-03-1", distance: 150, duration: 500})

CREATE (:Def\_Flight { def\_airline: " A2", date: "2022-03-2", distance: 106, duration: 505})

CREATE (:Def\_Flight { def\_airline: " A3", date: "2022-03-3", distance: 107, duration: 600})

CREATE (:Def\_Flight { def\_airline: " A4", date: "2022-03-4", distance: 108, duration: 605})

CREATE (:Def\_Flight { def\_airline: " A5", date: "2022-03-5", distance: 100, duration: 205})

CREATE (:Def\_Flight { def\_airline: " A6", date: "2022-03-6", distance: 105, duration: 305})

CREATE (:Def\_Flight { def\_airline: " A7", date: "2022-03-7", distance: 108, duration: 308})

CREATE (:Def\_Flight { def\_airline: " A8", date: "2022-03-8", distance: 200, duration: 308})

CREATE (:Def\_Flight { def\_airline: " A9", date: "2022-03-9", distance: 300, duration: 408})

CREATE (:Def\_Flight { def\_airline: " A10", date: "2022-03-10", distance: 309, duration: 308})

**Output:**

Added 10 labels, created 10 nodes, set 40 properties, completed after 21 ms.

1. **Ticket nodes:**
2. **Cypher:**

CREATE (:Def\_Ticket {ticket\_class: "Business Class", price: 2000})

CREATE (:Def\_Ticket {ticket\_class: "Economy Class", price: 500})

1. **Ticket & Flights Relation:**
2. **Cypher:**

MATCH (a:Def\_Flight {date: "2022-03-1"})

MATCH (b:Def\_Ticket {class: " Business Class "})

CREATE (a)-[:HAS]->(b)

Output: Created relationship

1. **Cypher:**

MATCH (a:Def\_Flight {date: "2022-03-1"})

MATCH (b:Def\_Ticket {class: " Economy Class "})

CREATE (a)-[:HAS]->(b)

Output: Created relationship

1. **Cypher:**

MATCH (a:Def\_Flight {date: "2022-03-2"})

MATCH (b:Def\_Ticket {class: " Business Class "})

CREATE (a)-[:HAS]->(b)

Output: Created relationship

1. **Cypher:**

MATCH (a:Def\_Flight {date: "2022-03-2"})

MATCH (b:Def\_Ticket {class: " Economy Class "})

CREATE (a)-[:HAS]->(b)

Output: Created relationship

1. **Cypher:**

MATCH (a:Def\_Flight {date: "2022-03-3"})

MATCH (b:Def\_Ticket {class: " Business Class "})

CREATE (a)-[:HAS]->(b)

Output: Created relationship

1. **Cypher:**

MATCH (a:Def\_Flight {date: "2022-03-3"})

MATCH (b:Def\_Ticket {class: " Economy Class "})

CREATE (a)-[:HAS]->(b)

Output: Created relationship

1. **Cypher:**

MATCH (a:Def\_Flight {date: "2022-03-4"})

MATCH (b:Def\_Ticket {class: " Business Class "})

CREATE (a)-[:HAS]->(b)

Output: Created relationship

1. **Cypher:**

MATCH (a:Def\_Flight {date: "2022-03-4"})

MATCH (b:Def\_Ticket {class: " Economy Class "})

CREATE (a)-[:HAS]->(b)

Output: Created relationship

1. **Cypher:**

MATCH (a:Def\_Flight {date: "2022-03-5"})

MATCH (b:Def\_Ticket {class: " Business Class "})

CREATE (a)-[:HAS]->(b)

Output: Created relationship

1. **Cypher:**

MATCH (a:Def\_Flight {date: "2022-03-5"})

MATCH (b:Def\_Ticket {class: " Economy Class "})

CREATE (a)-[:HAS]->(b)

1. **Cypher:**

MATCH (a:Def\_Flight {date: "2022-03-6"})

MATCH (b:Def\_Ticket {class: " Business Class "})

CREATE (a)-[:HAS]->(b)

Output: Created relationship

1. **Cypher:**

MATCH (a:Def\_Flight {date: "2022-03-6"})

MATCH (b:Def\_Ticket {class: " Economy Class "})

CREATE (a)-[:HAS]->(b)

Output: Created relationship

1. **Cypher:**

MATCH (a:Def\_Flight {date: "2022-03-7"})

MATCH (b:Def\_Ticket {class: " Business Class "})

CREATE (a)-[:HAS]->(b)

Output: Created relationship

1. **Cypher:**

MATCH (a:Def\_Flight {date: "2022-03-7"})

MATCH (b:Def\_Ticket {class: " Economy Class "})

CREATE (a)-[:HAS]->(b)

Output: Created relationship

1. **Cypher:**

MATCH (a:Def\_Flight {date: "2022-03-8"})

MATCH (b:Def\_Ticket {class: " Business Class "})

CREATE (a)-[:HAS]->(b)

Output: Created relationship

1. **Cypher:**

MATCH (a:Def\_Flight {date: "2022-03-8"})

MATCH (b:Def\_Ticket {class: " Economy Class "})

CREATE (a)-[:HAS]->(b)

Output: Created relationship

1. **Cypher:**

MATCH (a:Def\_Flight {date: "2022-03-9"})

MATCH (b:Def\_Ticket {class: " Business Class "})

CREATE (a)-[:HAS]->(b)

Output: Created relationship

1. **Cypher:**

MATCH (a:Def\_Flight {date: "2022-03-9"})

MATCH (b:Def\_Ticket {class: " Economy Class "})

CREATE (a)-[:HAS]->(b)

Output: Created relationship

1. **Cypher:**

MATCH (a:Def\_Flight {date: "2022-03-10"})

MATCH (b:Def\_Ticket {class: " Business Class "})

CREATE (a)-[:HAS]->(b)

Output: Created relationship

1. **Cypher:**

MATCH (a:Def\_Flight {date: "2022-03-10"})

MATCH (b:Def\_Ticket {class: " Economy Class "})

CREATE (a)-[:HAS]->(b)

1. **Flights and Airports Relation**
2. **Cypher:**

MATCH (a:Flight {date: "2022-03-1"})

MATCH (c:Airport {name: " Qatar "})

CREATE (a)-[:ORIGIN]->(c)

Output: Created relationship

1. **Cypher:**

MATCH (a:Flight {date: "2022-03-2"})

MATCH (c:Airport {name: " Saudi Air "})

CREATE (a)-[:ORIGIN]->(c)

Output: Created relationship

1. **Cypher:**

MATCH (a:Flight {date: "2022-03-3"})

MATCH (c:Airport {name: " SereneAir "})

CREATE (a)-[:ORIGIN]->(c)

Output: Created relationship

1. **Cypher:**

MATCH (a:Flight {date: "2022-03-4"})

MATCH (c:Airport {name: "AirSial"})

CREATE (a)-[:ORIGIN]->(c)

Output: Created relationship

1. **Cypher:**

MATCH (a:Flight {date: "2022-03-5"})

MATCH (c:Airport {name: "FlyToday"})

CREATE (a)-[:ORIGIN]->(c)

Output: Created relationship

1. **Cypher:**

MATCH (a:Flight {date: "2022-03-6"})

MATCH (c:Airport {name: "Qatar"})

CREATE (a)-[:ORIGIN]->(c)

1. **Cypher:**

MATCH (a:Flight {date: "2022-03-7"})

MATCH (c:Airport {name: " Saudi Air "})

CREATE (a)-[:ORIGIN]->(c)

Output: Created relationship

1. **Cypher:**

MATCH (a:Flight {date: "2022-03-8"})

MATCH (c:Airport {name: " SereneAir "})

CREATE (a)-[:ORIGIN]->(c)

Output: Created relationship

1. **Cypher:**

MATCH (a:Flight {date: "2022-03-9"})

MATCH (c:Airport {name: " AirSial "})

CREATE (a)-[:ORIGIN]->(c)

Output: Created relationship

1. **Cypher:**

MATCH (a:Flight {date: "2022-03-10"})

MATCH (c:Airport {name: " FlyToday "})

CREATE (a)-[:ORIGIN]->(c)

Output: Created relationship

**Cypher for Destination**

DESTINATION

1. **Cypher:**

MATCH (a:Flight {date: "2022-03-1"})

MATCH (c:Airport {name: " Qatar "})

CREATE (a)-[: DESTINATION]->(c)

Output: Created relationship

1. **Cypher:**

MATCH (a:Flight {date: "2022-03-2"})

MATCH (c:Airport {name: " Saudi Air "})

CREATE (a)-[: DESTINATION]->(c)

Output: Created relationship

1. **Cypher:**

MATCH (a:Flight {date: "2022-03-3"})

MATCH (c:Airport {name: " SereneAir "})

CREATE (a)-[: DESTINATION]->(c)

Output: Created relationship

1. **Cypher:**

MATCH (a:Flight {date: "2022-03-4"})

MATCH (c:Airport {name: "AirSial"})

CREATE (a)-[: DESTINATION]->(c)

Output: Created relationship

1. **Cypher:**

MATCH (a:Flight {date: "2022-03-5"})

MATCH (c:Airport {name: "FlyToday"})

CREATE (a)-[: DESTINATION]->(c)

Output: Created relationship

1. **Cypher:**

MATCH (a:Flight {date: "2022-03-6"})

MATCH (c:Airport {name: "Qatar"})

CREATE (a)-[: DESTINATION]->(c)

Output: Created relationship

1. **Cypher:**

MATCH (a:Flight {date: "2022-03-7"})

MATCH (c:Airport {name: " Saudi Air "})

CREATE (a)-[: DESTINATION]->(c)

Output: Created relationship

1. **Cypher:**

MATCH (a:Flight {date: "2022-03-8"})

MATCH (c:Airport {name: " SereneAir "})

CREATE (a)-[: DESTINATION]->(c)

Output: Created relationship

1. **Cypher:**

MATCH (a:Flight {date: "2022-03-9"})

MATCH (c:Airport {name: " AirSial "})

CREATE (a)-[: DESTINATION]->(c)

Output: Created relationship

1. **Cypher:**

MATCH (a:Flight {date: "2022-03-10"})

MATCH (c:Airport {name: " FlyToday "})

CREATE (a)-[: DESTINATION]->(c)

Output: Created relationship

**B2. Entering Queries for Graph Databased**

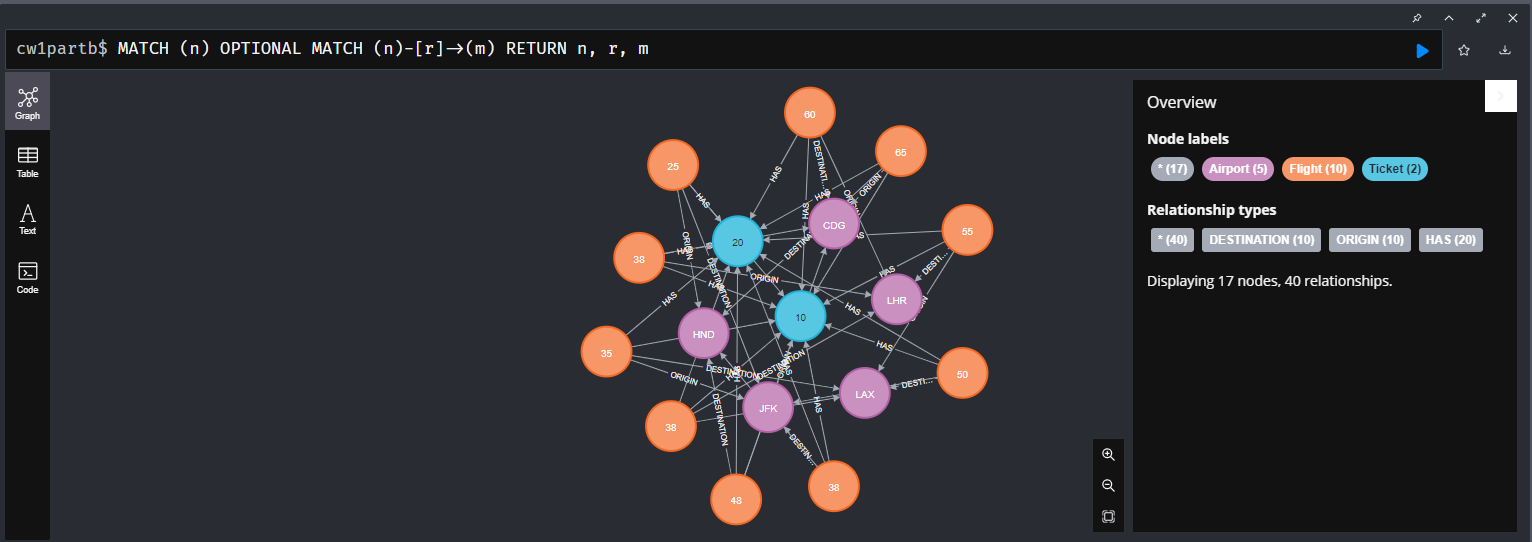
1. **Query 1:**

MATCH (a)

OPTIONAL MATCH (a)-[b]->(c)

RETURN a, b, c

**Output:**



1. **Query 2:**

MATCH (c:Def\_Airport {name: "Qatar"})<-[:ORIGIN]-(a:Def\_Flight)

RETURN a.def\_airline, a.date, a.distance, a.duration

**Output:**

╒════════════╤════════════╤════════════╤════════════╕

│"a.airline" │"a.date" │"a.distance"│"a.duration"│

╞════════════╪════════════╪════════════╪════════════╡

│"A1"│"2022-03-1" │150 │500 │

├────────────┼────────────┼────────────┼────────────┤

│"A6"│"2022-03-6" │105 │305 │

└────────────┴────────────┴────────────┴────────────┘

1. **Query 3:**

MATCH (origin:Def\_Airport {name: "Qatar"})<-[:ORIGIN]-(a:Def\_Flight)-[:DESTINATION]->(destination:Def\_Airport)

RETURN destination.name

**Output:**

╒══════════════════╕

│"destination.name"│

╞══════════════════╡

│"Saudi Air" │

├──────────────────┤

│"Saudi Air" │

└──────────────────┘

1. **Query 4:**

MATCH (a:Def\_Flight {def\_airline: "A1"})

CREATE (a)-[:HAS]->(:Def\_Ticket {class: " Business Class ", price: 2000})

**Output:**

Added 1 label, created 1 node, set 2 properties, created 1 relationship

**Query 5:**

MATCH (destination:Def\_Airport {name: "Saudi Air"})<-[to:DESTINATION]-(a:Def\_Flight)-[has:HAS]->(b:Def\_Ticket)

RETURN b.class, b.price

**Output:**

╒════════════╤═════════╕

│"b.lass" │"b.price" │

╞════════════╪═════════╡

│"Economy Class" │500 │

├────────────┼─────────┤

│"Business Class" │2000 │

├────────────┼─────────┤

│"Business Class" │2000 │

├────────────┼─────────┤

│"Economy Class" │500 │

├────────────┼─────────┤

│"Business Class" │2000 │

└────────────┴─────────┘