SQL Schema Creation and Queries

RISHI KIRAN

Creating Tables

- CREATE TABLE department (
- department_id INT PRIMARY KEY,
- department_name VARCHAR(255) NOT NULL
-);

- CREATE TABLE year (
- year_id INT PRIMARY KEY,
- year_name VARCHAR(50) NOT NULL
-);

- CREATE TABLE students (
- student_id INT PRIMARY KEY AUTO_INCREMENT,
- student_name VARCHAR(255) NOT NULL,
- department_id INT,
- year_id INT,
- FOREIGN KEY (department_id) REFERENCES department(department_id),
- FOREIGN KEY (year_id) REFERENCES year(year_id)
-);

Inserting Data

- INSERT INTO department (department_id, department_name) VALUES
- (66, 'CSE'), (67, 'ECE'), (68, 'ME'), (69, 'CE'), (70, 'EE');

- INSERT INTO year (year_id, year_name) VALUES
- (21, 'First Year'), (22, 'Second Year'), (23, 'Third Year'), (24, 'Fourth Year');

- INSERT INTO students (student_name, department_id, year_id) VALUES
- ('Alice', 66, 21), ('Bob', 66, 21), ('Charlie', 66, 21), ('David', 66, 21), ('Eve', 66, 21),
- ('Frank', 67, 21), ('Grace', 67, 21), ('Heidi', 67, 21), ('Ivan', 67, 21), ('Judy', 67, 21),
- ('Mallory', 68, 21), ('Niaj', 68, 21), ('Olivia', 68, 21), ('Peggy', 68, 21), ('Sybil', 68, 21),
- ('Trent', 69, 21), ('Victor', 69, 21), ('Walter', 69, 21), ('Xander', 69, 21), ('Yvonne', 69, 21),
- ('Zara', 70, 21), ('Bill', 70, 21), ('Carl', 70, 21), ('Dylan', 70, 21), ('Ethan', 70, 21);

Query 1: Display Students from CSE Department

- select * from students where students.department_id=(select department_id from department where department_name="CSE");

```
+------------+--------------+---------------+---------+
| student_id | student_name | department_id | year_id |
+------------+--------------+---------------+---------+
|          5 | Jignes       |            66 |      21 |
|          6 | Rohith       |            66 |      22 |
|         32 | Rishi        |            66 |      21 |
|         44 | Nitin        |            66 |      23 |
|         58 | Laxman       |            66 |      22 |
+------------+--------------+---------------+---------+
```

Query 2: Display Only Department Names

- select distinct department_name from students join department on students.department_id = department.department_id;

```
+----------------------+
| department_name      |
+----------------------+
| CSE                  |
| EC                   |
| EEE                  |
+----------------------+
```

Query 3: Display Students Sorted by Department and First Name

- select * from students,department order by department.department_name,students.student_nam;

```
+------------+--------------+---------------+----------+--------------+-----------
-----+
| student_id | student_name | department_id | year_id | department_id |
department_name |
+------------+--------------+---------------+----------+--------------+-----------
-----+
|          9 | Jignes       |            68 |      21 |            66 | CSE
|
|          5 | Jignes       |            66 |      21 |            66 | CSE
|
|          8 | Jignes       |            67 |      21 |            66 | CSE
|
|         60 | Laxman       |            68 |      22 |            66 | CSE
|
|         59 | Laxman       |            67 |      22 |            66 | CSE
|
|         58 | Laxman       |            66 |      22 |            66 | CSE
|
|         46 | Nitin        |            68 |      23 |            66 | CSE
|
```

```
|  44 | Nitin    |              66 |   23 |             66 | CSE
|  45 | Nitin    |              67 |   23 |             66 | CSE
|  30 | Rishi    |              68 |   21 |             66 | CSE
|  32 | Rishi    |              66 |   21 |             66 | CSE
|  33 | Rishi    |              67 |   21 |             66 | CSE
|   6 | Rohith   |              66 |   22 |             66 | CSE
|   7 | Rohith   |              67 |   22 |             66 | CSE
|   4 | Rohith   |              68 |   22 |             66 | CSE
|   8 | Jignes   |              67 |   21 |             67 | EC
|   9 | Jignes   |              68 |   21 |             67 | EC
|   5 | Jignes   |              66 |   21 |             67 | EC
|  58 | Laxman   |              66 |   22 |             67 | EC
|  59 | Laxman   |              67 |   22 |             67 | EC
|  60 | Laxman   |              68 |   22 |             67 | EC
|  46 | Nitin    |              68 |   23 |             67 | EC
|  45 | Nitin    |              67 |   23 |             67 | EC
|  44 | Nitin    |              66 |   23 |             67 | EC
|  32 | Rishi    |              66 |   21 |             67 | EC
```

33	Rishi		67	21	67	EC
30	Rishi		68	21	67	EC
4	Rohith		68	22	67	EC
6	Rohith		66	22	67	EC
7	Rohith		67	22	67	EC
8	Jignes		67	21	68	EEE
5	Jignes		66	21	68	EEE
9	Jignes		68	21	68	EEE
58	Laxman		66	22	68	EEE
59	Laxman		67	22	68	EEE
60	Laxman		68	22	68	EEE
46	Nitin		68	23	68	EEE
45	Nitin		67	23	68	EEE
44	Nitin		66	23	68	EEE
30	Rishi		68	21	68	EEE
32	Rishi		66	21	68	EEE
33	Rishi		67	21	68	EEE
7	Rohith		67	22	68	EEE
4	Rohith		68	22	68	EEE
6	Rohith		66	22	68	EEE

+-----------+---------------+----------------+----------+----------------+-----------
-----

Introduction to MongoDB with MongoDB Atlas

- MongoDB is a popular NoSQL database that provides high performance, high availability, and easy scalability. It uses a flexible document model, allowing you to store data in JSON-like documents with dynamic schemas.

- MongoDB Atlas is the global cloud database service for MongoDB, providing fully managed databases on AWS, Azure, and Google Cloud. It offers features like automated backups, monitoring, and security features like encryption at rest and in transit.

Gen_AI.year

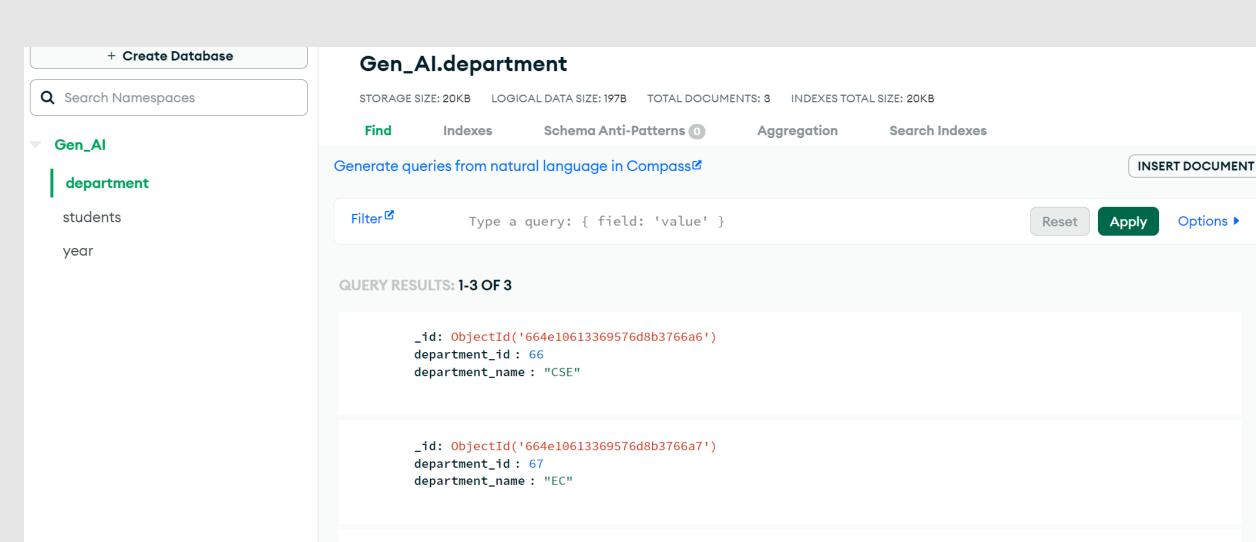STORAGE SIZE: 20KB   LOGICAL DATA SIZE: 169B   TOTAL DOCUMENTS: 3   INDEXES TOTAL SIZE: 20KB

Find   Indexes   Schema Anti-Patterns   Aggregation   Search Indexes

Generate queries from natural language in Compass⧉   INSERT DOCUMENT

Filter⧉   Type a query: { field: 'value' }   Reset   Apply   Options ▶

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

```
_id: ObjectId('664e10fb3369576d8b389730')
year_id : 21
year_name : "First"
```

```
_id: ObjectId('664e10fb3369576d8b389731')
year_id : 22
year_name : "Second"
```

```
_id: ObjectId('664e10fb3369576d8b389732')
year_id : 23
year_name : "Third"
```

Gen_AI.students

STORAGE SIZE: 4KB   LOGICAL DATA SIZE: 0B   TOTAL DOCUMENTS: 0   INDEXES TOTAL SIZE: 4KB

| **Find** | Indexes | Schema Anti-Patterns ⓪ | Aggregation | Search Indexes |

Generate queries from natural language in Compass⧉   **INSERT DOCUMENT**

Filter⧉   Type a query: { field: 'value' }   Reset   Apply   Options ▸

QUERY RESULTS: 1-15 OF 15

```
        _id: ObjectId('664e113e3369576d8b392167')
        student_id : 32
        student_name : "Rishi"
        department_id : 66
        year_id : 21


        _id: ObjectId('664e113e3369576d8b392168')
        student_id : 6
        student_name : "Rohith"
        department_id : 66
        year_id : 22
```

Create Database

Search Namespaces

▾ **Gen_AI**

    department

    students

    year

Gen_AI.department

Find   Indexes   Schema Anti-Patterns **0**   Aggregation   Search Indexes

Generate queries from natural language in Compass⧉   INSERT DOCUMENT

Filter⧉   Type a query: { field: 'value' }   Reset Apply Options ▶

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

```
        _id: ObjectId('664e10613369576d8b3766a6')
        department_id : 66
        department_name : "CSE"


        _id: ObjectId('664e10613369576d8b3766a7')
        department_id : 67
        department_name : "EC"


        _id: ObjectId('664e10613369576d8b3766a8')
        department_id : 68
        department_name : "EEE"
```