

Data Base Management

Students will learn and practice creating and managing databases. They will also gain experience writing and executing queries to retrieve the desired results. Using MySQL, students will work on creating, editing, and managing structured databases.

Here is a sample of the work completed by our students.

The screenshot shows a MySQL IDE interface with the following SQL code:

```
1 CREATE TABLE client_master (ClientNO int(10) primary key, Name varchar(15), Address varchar(20), City varchar(10), State varchar(10), bal_due  
2 show tables;  
3 insert into client_master values(1, 'akash', 'JP Nagar', 'Bangalore', 'Karnataka', 1000);  
4 insert into client_master values(2, 'Vilok', 'ITPL', 'Bangalore', 'Karnataka', 2200);  
5 insert into client_master values(3, 'Prashant', 'Indranagar', 'Bangalore', 'Karnataka', 1300);  
6 insert into client_master values(4, 'Sujay', 'yelanka', 'Bangalore', 'Karnataka', 6000);  
7 insert into client_master values(5, 'Vishua', 'MG road', 'Bangalore', 'Karnataka', 2500);  
8 desc client_master;  
9 select * from client_master;  
10 SELECT name FROM client_master WHERE bal_due > 2000;  
11 UPDATE client_master set bal_due = 500 WHERE ClientNO = 132;  
12 select * from client_master;  
13 ALTER TABLE client_master ADDNO 10 client_master;  
14 SELECT bal_due AS BALANCE FROM client_master;
```

The output shows the table structure and data:

Field	Type	Null	Key	Default	Extra
ClientNO	int	NO	PK		
Name	varchar(15)	YES			
Address	varchar(20)	YES			
City	varchar(10)	YES			
State	varchar(10)	YES			
bal_due	int	YES			

ClientNO	Name	Address	City	State	bal_due
1	Akash	JP Nagar	Bangalore	Karnataka	1000
2	Vilok	ITPL	Bangalore	Karnataka	2200
3	Prashant	Indranagar	Bangalore	Karnataka	1300
4	Sujay	yelanka	Bangalore	Karnataka	6000
5	Vishua	MG road	Bangalore	Karnataka	2500

The screenshot shows the output of the query:

```
name |  
-----  
Sujay |  
-----  
ClientNO | Name | Address | City | State | bal_due |  
-----  
1 | Akash | JP Nagar | Bangalore | Karnataka | 1000 |  
2 | Vilok | ITPL | Bangalore | Karnataka | 2200 |  
3 | Prashant | Indranagar | Bangalore | Karnataka | 1300 |  
4 | Sujay | yelanka | Bangalore | Karnataka | 6000 |  
5 | Vishua | MG road | Bangalore | Karnataka | 2500 |  
-----  
BALANCE |  
-----  
1000 |  
2200 |  
1300 |  
6000 |  
2500 |  
-----
```

```
deptno INT PRIMARY KEY,
dname VARCHAR(50),
loc VARCHAR(50)
);
CREATE TABLE Emp (
empno INT PRIMARY KEY,
ename VARCHAR(50),
job VARCHAR(50),
mgr INT,
hiredate DATE,
sal DECIMAL(10,2),
comm DECIMAL(10,2) DEFAULT 0,
deptno INT,
FOREIGN KEY (deptno) REFERENCES Dept(deptno)
);
INSERT INTO Dept (deptno, dname, loc)
VALUES
(101, 'Finance', 'Sydney'),
(102, 'Audit', 'Melbourne'),
(103, 'Marketing', 'Perth'),
(104, 'Production', 'Brisbane'),
(105, 'HR', 'Hyderabad');
INSERT INTO Emp (empno, ename, job, hiredate, sal, deptno)
VALUES
(69319, 'MAYLIND', 'PRESIDENT', '1991-11-18', 6000.00, 101),
(66928, 'BLAKE', 'MANAGER', '1991-06-09', 2750.00, 103),
(67932, 'CLARE', 'MANAGER', '1991-11-18', 2550.00, 101),
(65646, 'JONES', 'MANAGER', '1991-04-02', 2987.00, 103),
(67858, 'SCARLET', 'ANALYST', '1997-04-19', 3100.00, 102),
(69062, 'FRANK', 'ANALYST', '1991-12-03', 3100.00, 102),
(63679, 'SANDRINE', 'CLERK', '1990-12-18', 500.00, 102),
(64989, 'ADELYN', 'SALESMAN', '1991-02-20', 1700.00, 103);
select * from Emp;
UPDATE Emp
SET sal = sal * 1.15
WHERE DATEDIFF(CURRENT_DATE, hiredate) / 365 > 10;
select * from Emp;
DELETE FROM Emp
WHERE DATEDIFF(CURRENT_DATE, hiredate) / 365 >= 30;
select * from Emp;
```

Students also gained hands-on experience working with MongoDB and Neo4j. They explored MongoDB's flexible, document-based database structure, learning how to store and manage unstructured data effectively. In addition, students worked with Neo4j, a graph database, to understand and model complex relationships between data. This exposure helped them broaden their database management skills beyond traditional relational databases.

```
script.js
1 use DBKCLASS;
2
3 db.student.insertMany([
4   { Name: "Jack", Major: "Biology", GPA: 3.5 },
5   { Name: "Claire", Major: "Marketing", GPA: 4.0 },
6   { Name: "Ethan", Major: "Astronomy", GPA: 3.7, Grades: [90, 88, 85, 78] },
7   { Name: "Kate", Major: "Sociology", GPA: 3.2, Contact: { Phone: "3333-3333", Email: "Students@gmail.com" } },
8   { Name: "Phil", Major: "Biology", GPA: 2.5, StartDate: "2024-08-23" },
9   { Name: "Mike", Major: "Computer Science", GPA: 2.7 },
10  { Name: "Andre", Major: "Maths", GPA: 4.8, Awards: "star player" }]);
11
12
13 db.student.updateMany(
14   { Major: "Biology" },
15   { $set: { Major: "Bio" } });
16
17 db.student.updateMany(
18   { GPA: { $lt: 3 } },
19   { $set: { GPA: 3.8 } });
20
21 db.student.insertOne(
22   Name: "Master",
23   Age: 22,
24   Hobbies: ["Painting", "Fishing", "Travelling"]);
25
26 db.student.deleteMany({ GPA: { $gt: 3.5 } });
27
28 db.student.find().pretty();
```

Lab Test 2 on MongoDB

Vishwa V 15/15

Screenshot 2025-03-07 161244.png

```
script.js 43b447p8q
1 use DBKCLASS;
2
3 db.student.insertMany([
4   { Name: "Jack", Major: "Biology", GPA: 3.5 },
5   { Name: "Claire", Major: "Marketing", GPA: 4.0 },
6   { Name: "Ethan", Major: "Astronomy", GPA: 3.7, Grades: [90, 88, 95, 78] },
7   { Name: "Kate", Major: "Sociology", GPA: 3.2, Contact: { Phone: "333-3333", Email: "Students@mail.com" } },
8   { Name: "Phil", Major: "Biology", GPA: 2.5, StartDate: "2024-08-23" },
9   { Name: "Mike", Major: "Computer Science", GPA: 2.7 },
10  { Name: "Andre", Major: "Maths", GPA: 4.0, Awards: "Star player" }]);
11
12
13 db.student.updateMany(
14   { Major: "Biology" },
15   { $set: { Major: "Bio" } });
16
17 db.student.updateMany(
18   { GPA: { $lt: 3 } },
19   { $set: { GPA: 3.0 } });
20
21 db.student.insertOne({
22   Name: "Nester",
23   Age: 22,
24   Hobbies: ["Painting", "Fishing", "Travelling"]});
25
26 db.student.deleteMany({ GPA: { $gt: 3.5 } });
27
28 db.student.find().pretty();
```

Files

Turned in on Mar 11, 6:57 PM

See history

- Screenshot 2025-03-...
- Screenshot 2025-03-07 1613...
- Screenshot 2025-03-07 1613...

Grade

15/15

Private comments

Add private comment...

Post

Lab Test 2 on MongoDB

Thousif Pasha 15/15

neo4j1.png

Files

Turned in on Mar 7, 4:46 PM

See history

- mongodb.js
- Screenshot 2025-03-07 1636...
- Screenshot 2025-03-07 1646...
- neo4j2.png
- neo4j1.png

Grade

15/15

Private comments

Add private comment...

Post

Lab Test 2 on MongoDB

Thousif Pasha 15/15

neo4j2.png

Files

Turned in on Mar 7, 4:46 PM

See history

mongodb.js

Screenshot 2025-03-07 1636...

Screenshot 2025-03-07 1646...

neo4j2.png

neo4j1.png

Grade

15/15

Private comments

Add private comment...

Post

Lab Test 2 on MongoDB

Sumodh R 15/15

graph.png

Files

Turned in on Mar 7, 4:41 PM

See history

main.js

graph.png

Grade

15/15

Private comments

Add private comment...

Post

Using MySQL, MongoDB, and Neo4j in their training provides students with a comprehensive understanding of different database technologies. Each platform offers unique features—MySQL for structured relational data, MongoDB for flexible document storage, and Neo4j for graph-based relationships. By working with multiple database applications, students gain valuable hands-on experience that prepares them to handle a wide range of real-world data challenges. This diverse exposure not only enhances their technical skills but also makes them more adaptable and competitive in the evolving job market.