

“Heaven’s Light is Our Guide”

Rajshahi University of Engineering & Technology, Rajshahi



Department of Electrical & Computer Engineering

Course Code : ECE 2216

Course Title : Database Systems Sessional

Submission Date : 22-09-24

<p>Submitted to,</p> <p>Oishi Jyoti Assistant Professor Electrical & Computer Engineering, RUET</p>	<p>Submitted by,</p> <p>Shaikh Golam Rabbani 2110052 Electrical & Computer Engineering, RUET</p>
---	--

Experiment No: 01

1.1 Experiment Name: Introduction to Database System and Design and Implementation of a Student Database System for Odd and Even Batches

1.2 Objectives:

- 1.2.1 Design and implement a structured database system using SQL, focusing on DDL and DML operations.
- 1.2.2 Gain practical experience in creating, modifying, and managing database schemas with DDL commands such as CREATE, ALTER, DROP, and TRUNCATE.
- 1.2.3 Learn to manipulate data within database structures using DML commands like INSERT, UPDATE, DELETE, and SELECT.
- 1.2.4 Understand how to define database tables and enforce constraints within a database system.
- 1.2.5 Perform basic data handling tasks to manage real-world applications using SQL.

1.3 Theory

A Database System is a structured collection of data that is stored and accessed electronically. It allows users to efficiently manage, organize, and retrieve large volumes of information. A Database Management System (DBMS) is software that provides the tools to interact with the database, ensuring data is organized, secure, and accessible. Databases are essential in modern applications as they offer a systematic way to store data, enforce integrity, and enable complex queries. Common types of databases include relational databases (like MySQL, PostgreSQL) that use tables to store data in rows and columns, and NoSQL databases that handle unstructured data. (*Database - Wikipedia*, n.d.; Silberschatz et al., 2003)

Key concepts in databases include:

- Data Definition Language (DDL): Commands such as CREATE, ALTER, and DROP for defining and modifying database structures.
- Data Manipulation Language (DML): Commands like INSERT, UPDATE, DELETE, and SELECT for managing and retrieving data. (*Database Systems - an Overview / ScienceDirect Topics*, n.d.)

The Design and Implementation of a Student Database System for Odd and Even Batches involves creating two tables to manage student information based on batch numbers. Each table stores details such as Student ID, Name, Contact, Blood Group, Major Subject, and Obtained Marks. Key steps include defining the tables using SQL CREATE commands, inserting student data with INSERT, and performing updates, deletions, or queries using UPDATE, DELETE, and SELECT commands. Constraints like primary keys and NOT NULL ensure data integrity. This system demonstrates practical database management,

organizing student data by batch, and utilizing SQL for effective data handling and querying.

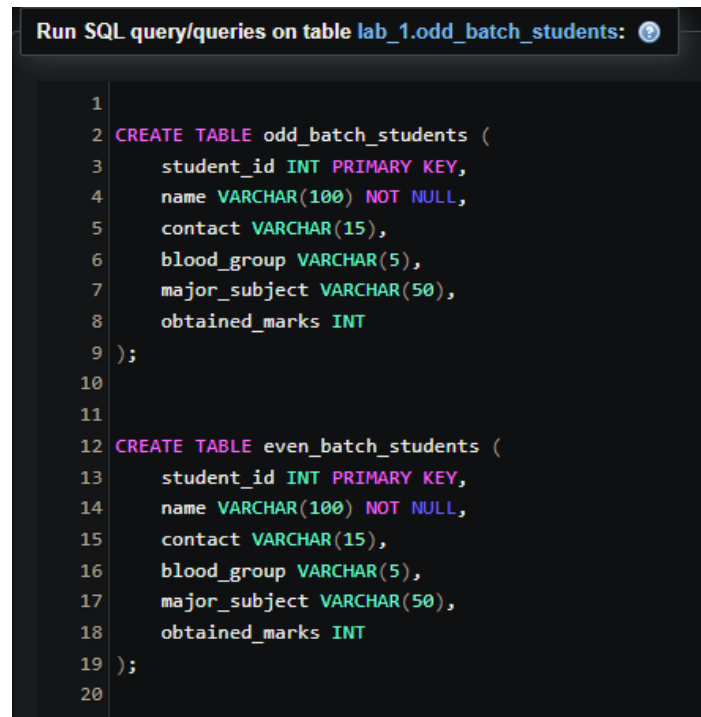
1.4 Required Apparatus:

1.4.1 XAMPP Computer Program

1.5 Code Screenshot and Output:

Creating Table:

```
1. CREATE TABLE odd_batch_students (  
2.     student_id INT PRIMARY KEY,  
3.     name VARCHAR(100) NOT NULL,  
4.     contact VARCHAR(15),  
5.     blood_group VARCHAR(5),  
6.     major_subject VARCHAR(50),  
7.     obtained_marks INT  
8. );  
9.  
10.  
11. CREATE TABLE even_batch_students (  
12.     student_id INT PRIMARY KEY,  
13.     name VARCHAR(100) NOT NULL,  
14.     contact VARCHAR(15),  
15.     blood_group VARCHAR(5),  
16.     major_subject VARCHAR(50),  
17.     obtained_marks INT  
18. );  
19.
```



The screenshot shows a SQL query runner interface with a dark background. At the top, a header bar reads "Run SQL query/queries on table lab_1.odd_batch_students: ?". Below this, the SQL code from the previous block is displayed in a monospaced font with syntax highlighting. The code is numbered on the left margin from 1 to 20. The code defines two tables: 'odd_batch_students' and 'even_batch_students', both with columns for 'student_id' (primary key), 'name', 'contact', 'blood_group', 'major_subject', and 'obtained_marks'.

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
<input type="checkbox"/>	1 student_id	int(11)			No	None			Change Drop More
<input type="checkbox"/>	2 name	varchar(100)	utf8mb4_general_ci		No	None			Change Drop More
<input type="checkbox"/>	3 contact	varchar(15)	utf8mb4_general_ci		Yes	NULL			Change Drop More
<input type="checkbox"/>	4 blood_group	varchar(5)	utf8mb4_general_ci		Yes	NULL			Change Drop More
<input type="checkbox"/>	5 major_subject	varchar(50)	utf8mb4_general_ci		Yes	NULL			Change Drop More
<input type="checkbox"/>	6 obtained_marks	int(11)			Yes	NULL			Change Drop More

Data Insertion:

```

1. INSERT INTO odd_batch_students (student_id, name, contact, blood_group, major_subject,
obtained_marks)
2. VALUES
3. (2110051, 'Srity', '12345', 'A+', 'Computer Science', 85),
4. (2110053, 'Sharif', '12345', 'B+', 'Mathematics', 78),
5. (2110055, 'Fahim', '12345', 'O-', 'Physics', 91);
6.
7. INSERT INTO even_batch_students (student_id, name, contact, blood_group, major_subject,
obtained_marks)
8. VALUES
9. (2110052, 'Rabbani', '12345', 'AB+', 'Biology', 88),
10. (2110054, 'Esha', '12345', 'A-', 'Chemistry', 76),
11. (2110056, 'Prattay', '12345', 'B-', 'Engineering', 92);
12.

```

				student_id	name	contact	blood_group	major_subject	obtained_marks
<input type="checkbox"/>	Edit	Copy	Delete	2110051	Srity	12345	A+	Computer Science	85
<input type="checkbox"/>	Edit	Copy	Delete	2110053	Sharif	12345	B+	Mathematics	78
<input type="checkbox"/>	Edit	Copy	Delete	2110055	Fahim	12345	O-	Physics	91
<input type="checkbox"/>	Check all	With selected:	Edit	Copy	Delete	Export			
				student_id	name	contact	blood_group	major_subject	obtained_marks
<input type="checkbox"/>	Edit	Copy	Delete	2110052	Rabbani	12345	AB+	Biology	88
<input type="checkbox"/>	Edit	Copy	Delete	2110054	Esha	12345	A-	Chemistry	76
<input type="checkbox"/>	Edit	Copy	Delete	2110056	Prattay	12345	B-	Engineering	92
<input type="checkbox"/>	Check all	With selected:	Edit	Copy	Delete	Export			

Setting and Updating Data:

```

1. UPDATE odd_batch_students
2. SET contact = '55555';
3. UPDATE even_batch_students
4. SET contact = '55555';
5.

```

```

1. UPDATE odd_batch_students
2. SET major_subject = 'Information Technology', obtained_marks = 90
3. WHERE student_id = 2110051;
4.
5. UPDATE even_batch_students
6. SET major_subject = 'Environmental Science', obtained_marks = 92
7. WHERE student_id = 2110052;
8.

```

Run SQL query/queries on table lab_1.even_b

```

1 UPDATE odd_batch_students
2 SET contact = '55555';
3 UPDATE even_batch_students
4 SET contact = '55555';
5

```

	student_id	name	contact	blood_group	major_subject	obtained_marks
<input type="checkbox"/> Edit Copy Delete	2110052	Rabbani	55555	AB+	Environmental Science	92
<input type="checkbox"/> Edit Copy Delete	2110054	Esha	55555	A-	Chemistry	76
<input type="checkbox"/> Edit Copy Delete	2110056	Prattay	55555	B-	Engineering	92

Deleting Values:

```

1. DELETE FROM odd_batch_students
2. WHERE student_id = 2110053;
3.
4. INSERT INTO odd_batch_students (student_id, name, contact, blood_group, major_subject,
obtained_marks)
5. VALUES (2110053, 'Tahsin', '55555', 'O+', 'Mathematics', 82);
6.
7. DELETE FROM even_batch_students
8. WHERE student_id = 2110054;
9. INSERT INTO even_batch_students (student_id, name, contact, blood_group, major_subject,
obtained_marks)
10. VALUES (2110054, 'Galib', '55555', 'A+', 'Physics', 87);
11.

```

Extra options

	student_id	name	contact	blood_group	major_subject	obtained_marks
<input type="checkbox"/> Edit Copy Delete	2110052	Rabbani	55555	AB+	Environmental Science	92
<input type="checkbox"/> Edit Copy Delete	2110054	Galib	55555	A+	Physics	87
<input type="checkbox"/> Edit Copy Delete	2110056	Prattay	55555	B-	Engineering	92

☐ Check all
 With selected:
 ☐ Edit
 ☐ Copy
 ☐ Delete
 ☐ Export

Adding, Dropping and Truanting Table:

1. ALTER TABLE odd_batch_students
2. ADD email VARCHAR(100);
3. DROP TABLE even_batch_students;
4. TRUNCATE TABLE odd_batch_students;
- 5.

The screenshot shows the MySQL Workbench interface. At the top, a green status bar indicates: "✓ MySQL returned an empty result set (i.e. zero rows). (Query took 0.0002 seconds.)". Below this, the SQL query is displayed: `SELECT * FROM `odd_batch_students``. A toolbar contains buttons for Profiling, Edit inline, Edit, Explain SQL, Create PHP code, and Refresh. Below the toolbar, a table header is visible with columns: student_id, name, contact, blood_group, major_subject, obtained_marks, and email. A "Query results operations" dropdown menu is open, showing a "Create view" button. To the right of the table header, a red error box displays the message: "#1146 - Table 'lab_1.even_batch_students' doesn't exist". Below the table header, there is a "Bookmark this SQL query" button and a "Label:" input field. A checkbox labeled "Let every user access this bookmark" is also present. At the bottom, a red error bar displays the message: "#1146 - Table 'lab_1.even_batch_students' doesn't exist".

1.6 Procedure:

- 1.6.1 First, we start XAMPP and Access PHPMysqlAdmin
- 1.6.2 We create a new database and create tables using SQL DDL
- 1.6.3 Then we insert and manipulate data using DML
- 1.6.4 Then we delete data, set and update values
- 1.6.5 Finally, we alter, drop and delete a database

1.7 Discussions:

In conclusion, designing and implementing a structured database system using SQL provides invaluable hands-on experience in managing and manipulating data effectively. Through the use of Data Definition Language (DDL) for creating and modifying database schemas, as well as Data Manipulation Language (DML) for inserting, updating, and querying data, students gain essential skills that are crucial for real-world database management. This exercise not only reinforces theoretical knowledge but also equips students with practical tools to handle data integrity, enforce constraints, and perform efficient data handling tasks, preparing them for future challenges in database applications.

1.8 Precautions:

When managing databases, we ensure regular backups to prevent data loss and use strong passwords to protect access. We also limit user privileges to reduce unauthorized access, and implement constraints to maintain data integrity. We stay cautious with DDL and DML commands, as they can lead to irreversible changes. Lastly, we keep our database software updated to safeguard against vulnerabilities.

1.9 References:

Database - Wikipedia. (n.d.). Retrieved September 22, 2024, from <https://en.wikipedia.org/wiki/Database>

Database Systems - an overview / ScienceDirect Topics. (n.d.). Retrieved September 22, 2024, from <https://www.sciencedirect.com/topics/social-sciences/database-systems>

Silberschatz, A., Korth, H. F., & Sudarshan, S. (2003). Database Systems. *Encyclopedia of Information Systems*, 411–422. <https://doi.org/10.1016/B0-12-227240-4/00028-9>