



Chandigarh Engineering College Jhanjeri

Mohali-140307

Department of Artificial Intelligence (AI) and Data Sciences

Experiment-2

Experiment Name:

Data Definition & Manipulation (DDL & DML)

Objective:

To create tables using DDL commands and perform data insertion, updation, and deletion using DML commands

Prerequisites:

- Basic knowledge of SQL syntax.
- Understanding of database concepts (tables, rows, columns).

Key Terms: SQL Data types

System requirements:

- 4-8 GB RAM, 10-15 disk space, Windows OS
- MySQL Workbench 8.0 CE

Theory and Application:

SQL Database Data Types

DATA TYPES represents the type of data an object is holding. Data Types are defined for columns of a table, local/global variables, input/output arguments of procedures etc..

Each database system (MS SQL Server, MYSQL, DB2, Oracle etc.) have its own long list of data types but several data types are common in most of them. This article will list down common data types across various database systems.

Number Data Type

Few numeric data type has syntax of data_type(x). Here x is meant for precision value.

Date Time Data Type

- **datetime:** This data type stores both date and time values together. It supports dates ranging from **01/01/1753** to **12/31/9999** and requires **8 bytes** of storage. In some database systems, this data type is also referred to as a **timestamp**.
- **date:** This data type is used to store only the date component, without any time information.



- **time:** This data type stores only time-related information, without including the date.
- Some numeric data types follow the syntax **data_type(x)**, where **x** specifies the precision or size of the value.

String Data Type

- **char(x):** This data type stores fixed-length character data and is padded with spaces to match the specified length. Here, x defines the exact number of characters to be stored.
- **varchar(x):** Short for *variable characters*, this data type stores character data of variable length and does not add extra spaces. The value of x specifies the maximum number of characters allowed.
- **text:** This data type is used to store large or lengthy textual data.
- Some numeric data types use the syntax **data_type(x)**, where **x** represents the precision of the value.

Other Data Type

- **blob:** Binary Large Object. This type is used to store large amount of binary date such as images or other type of files.
- **money:** In few databases also termed as currency. The type is used to storage money/currency information
- **binary:** The data type is used to store information in binary string format.



Chandigarh Engineering College Jhanjeri
Mohali-140307
Department of Artificial Intelligence (AI) and Data Sciences

1. CREATE & USE DATABASE

Syntax:

```
CREATE DATABASE d2;  
USE d2;
```

A screenshot of the SQL Server Management Studio (SSMS) interface. The left pane shows the Object Explorer with the 'SCHEMAS' node expanded, displaying databases like 'd1', 'sakila', 'sample', 'sys', and 'world'. The central pane contains two query panes: Q1 and Q2. Q1 has the following content:

```
1 • CREATE database d2;  
2 • USE d2;
```

The right pane shows the results of the execution. A message box is displayed with the text: "Automatic context help is disabled. Use the toolbar to manually get help for the current caret position or to toggle automatic help." Below this, the output window shows the following log entries:

#	Time	Action	Message	Duration / Fetch
1	19:14:29	CREATE database d2	1 row(s) affected	0.000 sec
2	19:14:29	USE d2	0 row(s) affected	0.000 sec

Figure 1: CREATE & USE Database



2. CREATE TABLE

Syntax:

```
CREATE TABLE Building ( building_id INT AUTO_INCREMENT PRIMARY KEY, bname  
VARCHAR(20) NOT NULL UNIQUE, location VARCHAR(50) DEFAULT 'Unknown' );
```

The screenshot shows the MySQL Workbench interface. In the SQL editor tab titled 'SQL File 4', three SQL statements are listed:

```
1 • CREATE database d2j  
2 • USE d2j  
3 • CREATE TABLE Building(building_id int AUTO_INCREMENT PRIMARY KEY, bname varchar(20) NOT NULL UNIQUE, location varchar(50) DEFAULT 'Unknown');
```

In the output pane, the results of these statements are shown:

#	Time	Action	Message	Duration / Fetch
1	19:14:29	CREATE database d2j	1 row(s) affected	0.000 sec
2	19:14:29	USE d2j	0 row(s) affected	0.000 sec
3	19:18:32	CREATE TABLE Building(building_id int AUTO_INCREMENT PRIMARY KEY, bname varchar(20) NOT NULL UNIQUE, location varchar(50) DEFAULT 'Unknown');	0 row(s) affected	0.016 sec

Figure 2: CREATE TABLE

Creates Building table with PRIMARY KEY, AUTO_INCREMENT, NOT NULL, UNIQUE, DEFAULT value.



Chandigarh Engineering College Jhanjeri

Mohali-140307

Department of Artificial Intelligence (AI) and Data Sciences

```
CREATE TABLE room1 (room_no INT AUTO_INCREMENT PRIMARY KEY, Chair INT NOT NULL, Bench INT NOT NULL, building_id INT, FOREIGN KEY (building_id) REFERENCES Building(building_id));
```

The screenshot shows a SQL Server Management Studio (SSMS) interface. The top menu bar includes File, Edit, View, Query, Database, Server, Tools, Scripting, and Help. Below the menu is a toolbar with various icons. The main area contains two tabs: Q1 and Q2, with Q1 active. The query window displays the following SQL script:

```
1 • CREATE database d2;
2 • USE d2;
3 • CREATE TABLE Building(building_id int AUTO_INCREMENT PRIMARY KEY, bname varchar(20) NOT NULL UNIQUE, location varchar(50) DEFAULT 'Unknown');
4 • CREATE TABLE room1(room_no int AUTO_INCREMENT PRIMARY KEY, Chair int NOT NULL, Bench int NOT NULL, building_id int, FOREIGN KEY (building_id)
5 references Building(building_id));
6
```

To the right of the query window, there is a context help panel with the message: "Automatic context help is disabled. Use the toolbar to manually get help for the current caret position or to toggle automatic help." Below the query window is an output window titled "Action Output" which lists the execution details:

#	Time	Action	Message	Duration / Fetch
1	19:14:29	CREATE database d2	1 row(s) affected	0.000 sec
2	19:14:29	USE d2	0 row(s) affected	0.000 sec
3	19:18:32	CREATE TABLE Building(building_id int AUTO_INCREMENT PRIMARY KEY, bname varchar(20) NOT NULL UNIQUE, location varchar(50) DEFAULT 'Unknown');	0 row(s) affected	0.016 sec
4	19:24:31	CREATE TABLE room1(room_no int AUTO_INCREMENT PRIMARY KEY, Chair int NOT NULL, Bench int NOT NULL, building_id int, FOREIGN KEY (building_id) REFERENCES Building(building_id));	0 row(s) affected	0.047 sec

Figure 3: CREATE room1 TABLE

Creates room1 with NOT NULL, DEFAULT and a FOREIGN KEY referencing building.



Chandigarh Engineering College Jhanjeri

Mohali-140307

Department of Artificial Intelligence (AI) and Data Sciences

3. INSERT

Syntax

```
INSERT INTO Building (name, location) VALUES ('Block 1', 'Campus A'), ('Block 2', 'Campus B');
```

```
INSERT INTO room1 (Chair, Bench, building_id) VALUES(20, 10, 1), (25, 16, 2);
```

A screenshot of the SQL Server Management Studio (SSMS) interface. The menu bar includes File, Edit, View, Query, Database, Server, Tools, Scripting, and Help. The toolbar has various icons for file operations like Open, Save, and Print. The main window shows two queries: Q1 and Q2. Q1 contains the script for creating the 'Building' table and establishing a foreign key relationship with the 'room1' table. Q2 contains the script for inserting data into the 'Building' and 'room1' tables. A status bar at the bottom right indicates 'Automatic context help is disabled. Use the toolbar to manually get help for the current caret position or to toggle automatic help.'

```
File Edit View Query Database Server Tools Scripting Help  
Q1 Q2 SQL File 4* ×  
CREATE database d2;  
USE d2;  
CREATE TABLE Building(building_id int AUTO_INCREMENT PRIMARY KEY, bname varchar(20) NOT NULL UNIQUE, location varchar(50) DEFAULT 'Unknown');  
CREATE TABLE room1(room_no int AUTO_INCREMENT PRIMARY KEY, Chair int NOT NULL, Bench int NOT NULL, building_id int, FOREIGN KEY (building_id)  
REFERENCES Building(building_id));  
INSERT INTO Building(bname,location) VALUES ('Block 1','Campus A'),('Block 2','Campus B');  
INSERT INTO room1(Chair, Bench, building_id) VALUES (20,10,1),(25,16,2);
```

Figure 4: INSERT



4. SELECT

Syntax:

SELECT * FROM Building;

SELECT * FROM room1;

```
Q1 Q2 SQL File 4* 
CREATE database d2;
USE d2;
CREATE TABLE Building(building_id int AUTO_INCREMENT PRIMARY KEY, bname varchar(20) NOT NULL UNIQUE, location varchar(50) DEFAULT 'Unknown');
CREATE TABLE room1(room_no int AUTO_INCREMENT PRIMARY KEY, Chair int NOT NULL, Bench int NOT NULL, building_id int, FOREIGN KEY (building_id) REFERENCES Building(building_id));
INSERT INTO Building(bname,location) VALUES ('Block 1','Campus A'),('Block 2','Campus 2');
INSERT INTO room1(Chair, Bench, building_id) VALUES (20,10,1),(25,16,2);
SELECT * FROM Building;
```

Result Grid | Filter Rows: | Edit: | Export/Import: | Wrap Cell Content: | Result Grid | Form Editor | Field Types |

building_id	bname	location
1	Block 1	Campus A
2	Block 2	Campus 2
*	HULL	HULL

Building 1 x Apply Revert

```
File Edit View Query Database Server Tools Scripting Help
Q1 Q2 SQL File 4* 
USE d2;
CREATE TABLE Building(building_id int AUTO_INCREMENT PRIMARY KEY, bname varchar(20) NOT NULL UNIQUE, location varchar(50) DEFAULT 'Unknown');
CREATE TABLE room1(room_no int AUTO_INCREMENT PRIMARY KEY, Chair int NOT NULL, Bench int NOT NULL, building_id int, FOREIGN KEY (building_id) REFERENCES Building(building_id));
INSERT INTO Building(bname,location) VALUES ('Block 1','Campus A'),('Block 2','Campus 2');
INSERT INTO room1(Chair, Bench, building_id) VALUES (20,10,1),(25,16,2);
SELECT * FROM Building;
SELECT * FROM room1;
```

Result Grid | Filter Rows: | Edit: | Export/Import: | Wrap Cell Content: | Result Grid | Form Editor | Field Types |

room_no	Chair	Bench	building_id
1	20	10	1
2	25	16	2
*	HULL	HULL	HULL

room1 2 x Apply Revert

Figure 5: SELECT



Chandigarh Engineering College Jhanjeri Mohali-140307

Department of Artificial Intelligence (AI) and Data Sciences

5. UPDATE

Syntax:

```
UPDATE room1 SET chair = 22 WHERE room_id = 1;
```

The screenshot shows the SQL Server Management Studio interface. The top menu bar includes File, Edit, View, Query, Database, Server, Tools, Scripting, Help. Below the menu is a toolbar with various icons. The main area has two tabs: Q1 and Q2. Tab Q1 contains the SQL query:

```
1 • CREATE database d2;
2 • USE d2;
3 • CREATE TABLE Building(building_id int AUTO_INCREMENT PRIMARY KEY, bname varchar(20) NOT NULL UNIQUE, location varchar(50) DEFAULT 'Unknown');
4 • CREATE TABLE room1(room_no int AUTO_INCREMENT PRIMARY KEY, Chair int NOT NULL, Bench int NOT NULL, building_id int, FOREIGN KEY (building_id)
5   references Building(building_id));
6 • INSERT INTO Building(bname,location) VALUES ('Block 1','Campus A'), ('Block 2','Campus 2');
7 • INSERT INTO room1(Chair, Bench, building_id) VALUES (20,10,1),(25,16,2);
8 • SELECT * FROM Building;
9 • SELECT * FROM room1;
10 • UPDATE room1 SET Chair=22 WHERE room_no= 1;
```

 Tab Q2 is labeled "SQL File 4*". On the right side, there is a "SQLAdditions" panel with the message: "Automatic context help is disabled. Use the toolbar to manually get help for the current caret position or to toggle automatic help." Below the main area is an "Output" window showing the execution results:

Action Output	#	Time	Action	Message	Duration / Fetch
	1	19:32:23	UPDATE room1 SET Chair=22 WHERE room_no= 1	1 row(s) affected Rows matched: 1 Changed: 1 Warnings: 0	0.000 sec

Figure 6: UPDATE

6. DELETE

Syntax:

```
DELETE FROM room1 WHERE room_id = 2;
```

The screenshot shows the SQL Server Management Studio interface. The top menu bar includes File, Edit, View, Query, Database, Server, Tools, Scripting, Help. Below the menu is a toolbar with various icons. The main area has two tabs: Q1 and Q2. Tab Q1 contains the SQL query:

```
1 • CREATE database d2;
2 • USE d2;
3 • CREATE TABLE Building(building_id int AUTO_INCREMENT PRIMARY KEY, bname varchar(20) NOT NULL UNIQUE, location varchar(50) DEFAULT 'Unknown');
4 • CREATE TABLE room1(room_no int AUTO_INCREMENT PRIMARY KEY, Chair int NOT NULL, Bench int NOT NULL, building_id int, FOREIGN KEY (building_id)
5   references Building(building_id));
6 • INSERT INTO Building(bname,location) VALUES ('Block 1','Campus A'), ('Block 2','Campus 2');
7 • INSERT INTO room1(Chair, Bench, building_id) VALUES (20,10,1),(25,16,2);
8 • SELECT * FROM Building;
9 • SELECT * FROM room1;
10 • UPDATE room1 SET Chair=22 WHERE room_no= 1;
11 • DELETE FROM room1 WHERE room_no=2;
```

 Tab Q2 is labeled "SQL File 4*". On the right side, there is a "SQLAdditions" panel with the message: "Automatic context help is disabled. Use the toolbar to manually get help for the current caret position or to toggle automatic help." Below the main area is an "Output" window showing the execution results:

Action Output	#	Time	Action	Message	Duration / Fetch
	1	19:32:23	UPDATE room1 SET Chair=22 WHERE room_no= 1	1 row(s) affected Rows matched: 1 Changed: 1 Warnings: 0	0.000 sec
	2	19:33:30	DELETE FROM room1 WHERE room_no=2	1 row(s) affected	0.000 sec

Figure 7: DELETE



Chandigarh Engineering College Jhanjeri

Mohali-140307

Department of Artificial Intelligence (AI) and Data Sciences

7. ALTER

Syntax:

ALTER TABLE room1 ADD COLUMN projector BOOLEAN DEFAULT FALSE;

ALTER TABLE room1 MODIFY COLUMN bench INT NOT NULL;

The screenshot shows the SQL Server Management Studio interface. In the top-left pane, there are two tabs: Q1 and Q2. The Q2 tab is active and contains a SQL file named 'SQL File 4'. The code in the editor is as follows:

```
File Edit View Query Database Server Tools Scripting Help
Q1 Q2 SQL File 4*
[SQL Editor toolbar]
6 • INSERT INTO Building(bname,location) VALUES ('Block 1','Campus 1'),('Block 2','Campus 2');
7 • INSERT INTO room1(Chair, Bench, building_id) VALUES (20,10,1),(25,16,2);
8 • SELECT * FROM Building;
9 • SELECT * FROM room1;
10 • UPDATE room1 SET Chair=22 WHERE room_no= 1;
11 • DELETE FROM room1 WHERE room_no=2;
12 • ALTER TABLE room1 ADD COLUMN projector BOOLEAN DEFAULT FALSE;
13 • ALTER TABLE room1 MODIFY COLUMN bench INT NOT NULL;
```

To the right of the editor, a message box displays: "Automatic context help is disabled. Use the toolbar to manually get help for the current caret position or to toggle automatic help."

In the bottom-left pane, there is a table named 'room1' with columns: room_no, Char, bench, building_id, and projector. The data is:

room_no	Char	bench	building_id	projector
1	22	10	1	0
*	MAX	MAX	MAX	MAX

In the bottom-right pane, the 'Output' window shows the execution log for the statements:

#	Time	Action	Message	Duration / Fetch
1	19:32:23	UPDATE room1 SET Chair=22 WHERE room_no= 1	1 row(s) affected Rows matched: 1 Changed: 1 Warnings: 0	0.000 sec
2	19:33:30	DELETE FROM room1 WHERE room_no=2	1 row(s) affected	0.000 sec
3	19:34:56	ALTER TABLE room1 ADD COLUMN projector BOOLEAN DEFAULT FALSE	0 row(s) affected Records: 0 Duplicates: 0 Warnings: 0	0.016 sec
4	19:35:00	ALTER TABLE room1 MODIFY COLUMN bench INT NOT NULL	0 row(s) affected Records: 0 Duplicates: 0 Warnings: 0	0.016 sec
5	19:35:17	SELECT * FROM room1 LIMIT 0, 1000	1 row(s) returned	0.000 sec / 0.000 sec

Figure 8: ALTER

8. RENAME

Syntax:

RENAME TABLE room1 TO classroom1;

The screenshot shows the SQL Server Management Studio interface. In the top-left pane, there are two tabs: Q1 and Q2. The Q2 tab is active and contains a SQL file named 'SQL File 4'. The code in the editor is as follows:

```
File Edit View Query Database Server Tools Scripting Help
Q1 Q2 SQL File 4*
[SQL Editor toolbar]
1 • CREATE database d2;
2 • USE d2;
3 • CREATE TABLE Building(building_id int AUTO_INCREMENT PRIMARY KEY, bname varchar(20) NOT NULL UNIQUE, location varchar(50) DEFAULT 'Unknown');
4 • CREATE TABLE room1(room_no int AUTO_INCREMENT PRIMARY KEY, Chair int NOT NULL, Bench int NOT NULL, building_id int, FOREIGN KEY (building_id)
5 references Building(building_id));
6 • INSERT INTO Building(bname,location) VALUES ('Block 1','Campus A'),('Block 2','Campus 2');
7 • INSERT INTO room1(Chair, Bench, building_id) VALUES (20,10,1),(25,16,2);
8 • SELECT * FROM Building;
9 • SELECT * FROM room1;
10 • UPDATE room1 SET Chair=22 WHERE room_no= 1;
11 • DELETE FROM room1 WHERE room_no=2;
12 • ALTER TABLE room1 ADD COLUMN projector BOOLEAN DEFAULT FALSE;
13 • ALTER TABLE room1 MODIFY COLUMN bench INT NOT NULL;
14 • RENAME TABLE room1 TO classroom1;
```

To the right of the editor, a message box displays: "Automatic context help is disabled. Use the toolbar to manually get help for the current caret position or to toggle automatic help."

In the bottom-left pane, the 'Output' window shows the execution log for the statements:

#	Time	Action	Message	Duration / Fetch
2	19:33:30	DELETE FROM room1 WHERE room_no=2	1 row(s) affected	0.000 sec
3	19:34:56	ALTER TABLE room1 ADD COLUMN projector BOOLEAN DEFAULT FALSE	0 row(s) affected Records: 0 Duplicates: 0 Warnings: 0	0.016 sec
4	19:35:00	ALTER TABLE room1 MODIFY COLUMN bench INT NOT NULL	0 row(s) affected Records: 0 Duplicates: 0 Warnings: 0	0.016 sec
5	19:35:17	SELECT * FROM room1 LIMIT 0, 1000	1 row(s) returned	0.000 sec / 0.000 sec
6	19:35:49	RENAME TABLE room1 TO classroom1	0 row(s) affected	0.000 sec

Figure 9: RENAME



9. TRUNCATE

Syntax:

```
TRUNCATE TABLE classroom1;
```

The screenshot shows the SQL Server Management Studio interface. The top menu bar includes File, Edit, View, Query, Database, Server, Tools, Scripting, Help, and a toolbar with various icons. Below the menu is a tab bar with Q1, Q2, and SQL File 4*. The main pane displays a script of SQL commands. The command being executed is TRUNCATE TABLE classroom1;. The bottom pane shows the output of the command, indicating it was run at 19:37:12 and affected 0 rows.

```
File Edit View Query Database Server Tools Scripting Help
Q1 Q2 SQL File 4* ×
Limit to 1000 rows
1 • CREATE database d2;
2 • USE d2;
3 • CREATE TABLE Building(building_id int AUTO_INCREMENT PRIMARY KEY, bname varchar(20) NOT NULL UNIQUE, location varchar(50) DEFAULT 'Unknown');
4 • CREATE TABLE room1(room_no int AUTO_INCREMENT PRIMARY KEY, Chair int NOT NULL, Bench int NOT NULL, building_id int, FOREIGN KEY (building_id)
5 references Building(building_id));
6 • INSERT INTO Building(bname,location) VALUES ('Block 1','Campus A'), ('Block 2','Campus 2');
7 • INSERT INTO room1(Chair, Bench, building_id) VALUES (20,10,1),(25,16,2);
8 • SELECT * FROM Buildings;
9 • SELECT * FROM classroom1;
10 • UPDATE room1 SET Chair=22 WHERE room_no= 1;
11 • DELETE FROM room1 WHERE room_no=2;
12 • ALTER TABLE room1 ADD COLUMN projector BOOLEAN DEFAULT FALSE;
13 • ALTER TABLE room1 MODIFY COLUMN bench INT NOT NULL;
14 • RENAME TABLE room1 TO classroom1;
15 • TRUNCATE TABLE classroom1;
```

Output

#	Time	Action	Message	Duration / Fetch
1	19:37:12	TRUNCATE TABLE classroom1	0 row(s) affected	0.047 sec

Figure 10: TRUNCATE