Here's a **complete MySQL practice session** with step-by-step explanation, including:

- **Database creation**
- **Table creation**
- **INSERT, SELECT, UPDATE, DELETE operations**
- **Column ADD, MODIFY, DROP**
- **AUTO_INCREMENT primary key**
- **Proper code with output and Hinglish explanation**

🔶 Step 1: Create Database

```
sql
CopyEdit
CREATE DATABASE school;
```

🧠 **Explanation**: Ek naya database `school` banaya gaya jisme tables create karenge.

📩 **Output**:

```
graphql
CopyEdit
Query OK, 1 row affected (0.01 sec)
```

🔶 Step 2: Use the Database

```
sql
CopyEdit
USE school;
```

🧠 **Explanation**: Ab `school` database ke andar kaam karenge.

📩 **Output**:

```
nginx
CopyEdit
Database changed
```

🔶 Step 3: Create Table `students`

```
sql
CopyEdit
CREATE TABLE students (
```

```
    roll INT PRIMARY KEY,
    name TEXT,
    marks INT
);
```

🧠 **Explanation**: `students` table banaya jisme:

- `roll` ek unique roll number h (primary key)
- `name` student ka naam
- `marks` uske marks

📩 **Output**:

```
graphql
CopyEdit
Query OK, 0 rows affected (0.01 sec)
```

🔶 Step 4: Insert Data

```
sql
CopyEdit
INSERT INTO students VALUES
(101, "ravi", 90),
(102, "ram", 99),
(103, "harish", 60),
(104, "yash", 50);
```

🧠 **Explanation**: 4 students ke data insert kiye gaye.

📩 **Output**:

```
graphql
CopyEdit
Query OK, 4 rows affected (0.01 sec)
Records: 4  Duplicates: 0  Warnings: 0
```

🔶 Step 5: Display Data

```
sql
CopyEdit
SELECT * FROM students;
```

📩 **Output**:

```
pgsql
CopyEdit
```

```
+------+--------+-------+
| roll | name   | marks |
+------+--------+-------+
|  101 | ravi   |    90 |
|  102 | ram    |    99 |
|  103 | harish |    60 |
|  104 | yash   |    50 |
+------+--------+-------+
```

🔶 Step 6: Add a Column

```
sql
CopyEdit
ALTER TABLE students ADD COLUMN phone TEXT;
```

🧠 **Explanation**: Ek naya column `phone` add kiya gaya students table mein.

📩 **Output**:

```
graphql
CopyEdit
Query OK, 0 rows affected (0.02 sec)
```

🔶 Step 7: Rename Column

```
sql
CopyEdit
ALTER TABLE students RENAME COLUMN phone TO contact;
```

🧠 **Explanation**: `phone` column ka naam `contact` rakha gaya.

📩 **Output**:

```
graphql
CopyEdit
Query OK, 0 rows affected (0.01 sec)
```

🔶 Step 8: Drop a Column

```
sql
CopyEdit
ALTER TABLE students DROP COLUMN contact;
```

🧠 **Explanation**: `contact` column delete kar diya.

📤 **Output**:

```
graphql
CopyEdit
Query OK, 0 rows affected (0.01 sec)
```

🔶 Step 9: Update Marks

```
sql
CopyEdit
UPDATE students SET marks = marks + 5;
```

🧠 **Explanation**: Sabhi students ke marks mein +5 add kiya gaya.

📤 **Output**:

```
graphql
CopyEdit
Query OK, 4 rows affected (0.01 sec)
```

🔶 Step 10: Conditional Update

```
sql
CopyEdit
UPDATE students SET marks = marks - 15 WHERE roll = 104;
```

🧠 **Explanation**: Roll number 104 wale student ke marks 15 kam kiye.

📤 **Output**:

```
graphql
CopyEdit
Query OK, 1 row affected (0.01 sec)
```

🔶 Step 11: Delete One Row

```
sql
CopyEdit
DELETE FROM students WHERE roll = 104;
```

🧠 **Explanation**: Roll 104 wale student ko delete kar diya.

📤 **Output**:

```
graphql
CopyEdit
Query OK, 1 row affected (0.00 sec)
```

🔶 Step 12: Delete All Rows

```
sql
CopyEdit
DELETE FROM students;
```

🧠 **Explanation**: Table `students` ka saara data delete kar diya.

📩 **Output**:

```
graphql
CopyEdit
Query OK, 3 rows affected (0.00 sec)
```

🔶 Step 13: Insert Again

```
sql
CopyEdit
INSERT INTO students VALUES (101, "rahul", 90);
```

🧠 **Explanation**: Ek naya student insert kiya.

📩 **Output**:

```
graphql
CopyEdit
Query OK, 1 row affected (0.00 sec)
```

🔶 Step 14: Create Table `emp` with AUTO_INCREMENT

```
sql
CopyEdit
CREATE TABLE emp (
    id INT AUTO_INCREMENT PRIMARY KEY,
    name TEXT
);
```

🧠 **Explanation**: Ek `emp` table banayi jisme `id` auto increment hoga aur primary key hai.

📩 **Output**:

```
graphql
CopyEdit
Query OK, 0 rows affected (0.01 sec)
```

🔶 Step 15: Insert into `emp`

```
sql
CopyEdit
INSERT INTO emp (name) VALUES ("ravi"), ("ram"), ("suresh");
```

📩 Output:

```
graphql
CopyEdit
Query OK, 3 rows affected (0.01 sec)
```

🔶 Step 16: Display Data from `emp`

```
sql
CopyEdit
SELECT * FROM emp;
```

📩 Output:

```
pgsql
CopyEdit
+----+--------+
| id | name   |
+----+--------+
|  1 | ravi   |
|  2 | ram    |
|  3 | suresh |
+----+--------+
```

🔶 Step 17: Delete a Row from `emp`

```
sql
CopyEdit
DELETE FROM emp WHERE id = 3;
```

📩 Output:

```
graphql
CopyEdit
Query OK, 1 row affected (0.01 sec)
```

🔶 Step 18: Insert Again

```
sql
CopyEdit
INSERT INTO emp (name) VALUES ("ram yash");
```

📩 Output:

```
graphql
CopyEdit
Query OK, 1 row affected (0.00 sec)
```

🔶 Step 19: Final Table Output

```
sql
CopyEdit
SELECT * FROM emp;
```

📩 Output:

```
pgsql
CopyEdit
+----+----------+
| id | name     |
+----+----------+
|  1 | ravi     |
|  2 | ram      |
|  4 | ram yash |
+----+----------+
```

🔶 Step 20: Delete All from `emp`

```
sql
CopyEdit
DELETE FROM emp;
```

📩 Output:

```
graphql
CopyEdit
Query OK, 3 rows affected (0.00 sec)
```

🔶 Step 21: Insert Again After Delete

```
sql
CopyEdit
INSERT INTO emp (name) VALUES ("ram yash");
```

📤 **Output**:

```
graphql
CopyEdit
Query OK, 1 row affected (0.00 sec)
```

Final `emp` Table:

```
sql
CopyEdit
SELECT * FROM emp;
```

📤 **Output**:

```
pgsql
CopyEdit
+----+----------+
| id | name     |
+----+----------+
|  5 | ram yash |
+----+----------+
```