

AutoManager: Vehicle & Sales Management System

DATABASE OVERVIEW

Database Name: carshowroom

BASIC DATABASE COMMANDS

SHOW DATABASES;

USE carshowroom;

SHOW TABLES;

DESC customers;

DESC salespeople;

DESC vehicles;

```
mysql> SHOW TABLES;
```

| Tables_in_carshowroom |
|-----------------------|
| customers |
| employees |
| salespeople |
| vehicles |

4 rows in set (0.01 sec)

```
mysql> DESC customers;
```

| Field | Type | Null | Key | Default | Extra |
|----------------|-------------|------|-----|---------|-------|
| customer_id | int | NO | PRI | NULL | |
| customer_name | varchar(20) | YES | | NULL | |
| phone_no | int | YES | | NULL | |
| driver_licence | varchar(20) | YES | | NULL | |

4 rows in set (0.01 sec)

```
mysql> DESC salespeople;
```

| Field | Type | Null | Key | Default | Extra |
|-------------------|-------------|------|-----|---------|-------|
| sales_person_id | int | NO | PRI | NULL | |
| sales_person_name | varchar(20) | YES | | NULL | |
| hire_date | date | YES | | NULL | |
| commision_rate | int | YES | | NULL | |

4 rows in set (0.00 sec)

```
mysql> DESC vehicles;
```

| Field | Type | Null | Key | Default | Extra |
|------------|---------------|------|-----|---------|----------------|
| vehicle_id | int | NO | PRI | NULL | auto_increment |
| brand | varchar(30) | YES | | NULL | |
| model | varchar(30) | YES | | NULL | |
| fuel_type | varchar(15) | YES | | NULL | |
| price | decimal(10,2) | YES | | NULL | |
| stock | int | YES | | NULL | |

```
6 rows in set (0.00 sec)
```

```
mysql> DESC employees;
```

| Field | Type | Null | Key | Default | Extra |
|-------------|-------------|------|-----|---------|----------------|
| employee_id | int | NO | PRI | NULL | auto_increment |
| emp_name | varchar(50) | YES | | NULL | |
| designation | varchar(30) | YES | | NULL | |
| phone | varchar(15) | YES | | NULL | |

```
4 rows in set (0.00 sec)
```

DDL – DATA DEFINITION LANGUAGE

1. CREATE TABLE

```
CREATE TABLE customers (  
  customer_id INT PRIMARY KEY,  
  customer_name VARCHAR(20),  
  phone_no INT,  
  driver_licence VARCHAR(20)  
);
```

2.Salespeople

```
CREATE TABLE salespeople (  
  sales_person_id INT PRIMARY KEY,  
  sales_person_name VARCHAR(20),
```

```
hire_date DATE,  
commision_rate INT  
);
```

3.Vehicles

```
CREATE TABLE vehicles (  
VIN VARCHAR(10) PRIMARY KEY,  
model VARCHAR(20),  
status VARCHAR(20),  
price INT  
);
```

3.2 ALTER TABLE

```
mysql> ALTER TABLE vehicles ADD manufacture_year INT;  
Query OK, 0 rows affected (0.15 sec)  
Records: 0 Duplicates: 0 Warnings: 0
```

3.3 DROP TABLE

```
DROP TABLE vehicles;
```

3.4 TRUNCATE TABLE

```
TRUNCATE TABLE customers;
```

4. DML – DATA MANIPULATION LANGUAGE

4.1 INSERT

```
mysql> INSERT INTO customers VALUES (1, 'Rahul', 987654321, 'OL12345');  
Query OK, 1 row affected (0.02 sec)  
  
mysql> INSERT INTO salespeople VALUES (101, 'Anita', '2023-06-10', 5);  
Query OK, 1 row affected (0.02 sec)  
  
mysql> INSERT INTO vehicles VALUES ('VIN001', 'Hyundai i20', 'Available', 800000);  
ERROR 1136 (21501): Column count doesn't match value count at row 1  
mysql> INSERT INTO vehicles VALUES ('001', 'Hyundai i20', 'Available', 800000);  
ERROR 1136 (21501): Column count doesn't match value count at row 1  
mysql> INSERT INTO vehicles VALUES ('VIN001', 'Hyundai i20', 'Available', 800000);  
ERROR 1136 (21501): Column count doesn't match value count at row 1  
mysql> INSERT INTO vehicles (VIN, model, status, price)  
-> VALUES ('VIN001', 'Hyundai i20', 'Available', 800000);  
Query OK, 1 row affected (0.03 sec)
```

4.2 SELECT

```
mysql> SELECT * FROM customers;
+-----+-----+-----+-----+
| customer_id | customer_name | phone_no | driver_licence |
+-----+-----+-----+-----+
|          5 | Kiran         | 909090909 | DL90909        |
+-----+-----+-----+-----+
1 row in set (0.00 sec)

mysql> SELECT * FROM salespeople;
+-----+-----+-----+-----+
| sales_person_id | sales_person_name | hire_date | commision_rate |
+-----+-----+-----+-----+
|          105 | Suman            | 2024-02-20 | 8              |
+-----+-----+-----+-----+
1 row in set (0.00 sec)
```

4.3 UPDATE

```
mysql> SELECT model, price FROM vehicles;UPDATE vehicles
+-----+-----+
| model   | price |
+-----+-----+
| Hyundai i20 | 800000 |
+-----+-----+
1 row in set (0.00 sec)

    -> SET status = 'Sold'
    -> WHERE VIN = 'VIN001';
Query OK, 1 row affected (0.01 sec)
Rows matched: 1  Changed: 1  Warnings: 0
```

4.4 DELETE

```
mysql> DELETE FROM vehicles
      -> WHERE VIN = 'VIN107';
Query OK, 1 row affected (0.01 sec)
```

```
mysql> SELECT * FROM vehicles;
```

| VIN | model | status | price |
|--------|---------------|-----------|---------|
| VIN105 | Toyota Glanza | Available | 780000 |
| VIN106 | Hyundai Venue | Sold | 900000 |
| VIN108 | Honda City | Available | 1150000 |
| VIN109 | Tata Nexon | Available | 980000 |

```
4 rows in set (0.00 sec)
```

5. DQL – DATA QUERY LANGUAGE

5.1 AGGREGATE FUNCTIONS

```
mysql> SELECT COUNT(VIN) AS total_vehicles FROM vehicles;
```

| total_vehicles |
|----------------|
| 1 |

```
1 row in set (0.01 sec)
```

```
mysql> SELECT AVG(price) AS avg_price FROM vehicles;
```

| avg_price |
|-------------|
| 880000.0000 |

```
1 row in set (0.03 sec)
```

5.2 GROUP BY & ORDER BY

```
mysql> SELECT status, COUNT(VIN)
      -> FROM vehicles
      -> GROUP BY status
      -> ORDER BY status;
```

| status | COUNT(VIN) |
|--------|------------|
| Sold | 1 |

```
1 row in set (0.08 sec)
```

```
mysql> |
```

6. TCL – TRANSACTION CONTROL

LANGUAGE 6.1 COMMIT

```
mysql> START TRANSACTION;
Query OK, 0 rows affected (0.01 sec)

mysql> UPDATE vehicles SET status = 'Sold' WHERE VIN = 'VIN002';
Query OK, 0 rows affected (0.00 sec)
Rows matched: 0 Changed: 0 Warnings: 0

mysql> COMMIT;
Query OK, 0 rows affected (0.00 sec)
```

6.2 ROLLBACK

```
mysql> START TRANSACTION;
Query OK, 0 rows affected (0.00 sec)

mysql> DELETE FROM vehicles WHERE VIN = 'VIN003';
Query OK, 0 rows affected (0.00 sec)

mysql> ROLLBACK;
Query OK, 0 rows affected (0.00 sec)
```

6.3 SAVEPOINT

```
mysql> START TRANSACTION;
Query OK, 0 rows affected (0.00 sec)

mysql> SAVEPOINT sp1;
Query OK, 0 rows affected (0.00 sec)

mysql> UPDATE vehicles SET price = 900000 WHERE VIN = 'VIN004';
Query OK, 0 rows affected (0.00 sec)
Rows matched: 0 Changed: 0 Warnings: 0

mysql> ROLLBACK TO sp1;
Query OK, 0 rows affected (0.00 sec)

mysql> COMMIT;
Query OK, 0 rows affected (0.00 sec)
```

7. JOINS

7.1 INNER JOIN

```
mysql> SELECT c.customer_name, s.sales_person_name
-> FROM customers c
-> INNER JOIN salespeople s
-> ON c.customer_id = s.sales_person_id;
Empty set (0.00 sec)
```

7.2 LEFT JOIN

SELECT c.customer_name, v.model

FROM customers c

LEFT JOIN vehicles v

ON c.customer_id = v.price;

7.3 RIGHT JOIN

SELECT s.sales_person_name, v.model

FROM salespeople s
RIGHT JOIN vehicles v
ON s.sales_person_id = v.price;

8. VIEW

CREATE VIEW available_vehicles AS
SELECT model, price
FROM vehicles
WHERE status = 'Available';

9. TRIGGER

```
mysql> CREATE TRIGGER set_default_status  
-> BEFORE INSERT ON vehicles  
-> FOR EACH ROW  
-> SET NEW.status = 'Available';  
Query OK, 0 rows affected (0.02 sec)
```