

Title page:

# **MY SQL DATABASE CREATION**

For

## **(Car Parking System)**

Prepared by

**Mohsin (2022-ag-7691)**

**Anas Mustafa(2022-ag-7706)**

**Hateem Farooq(2022-ag-7692)**

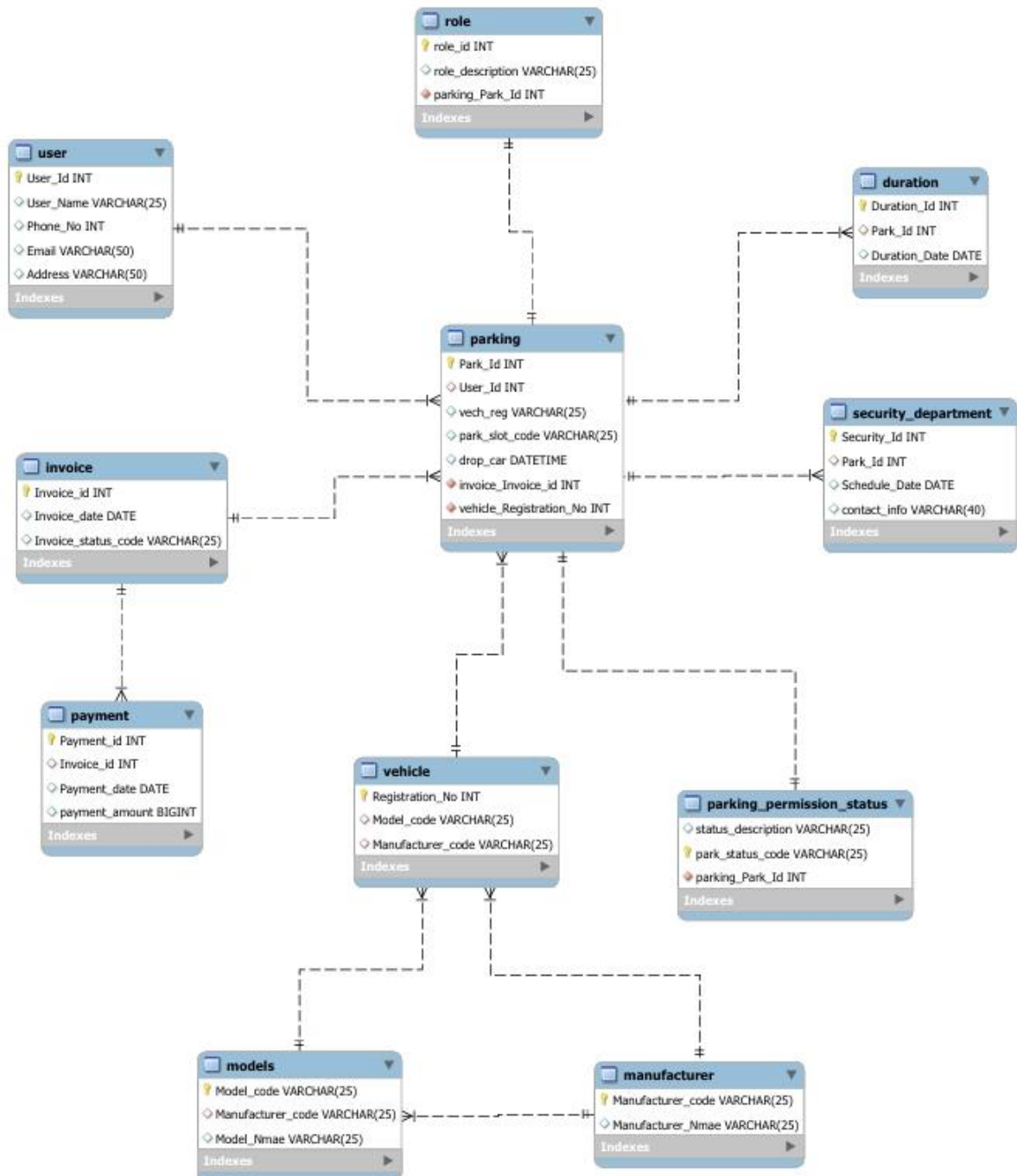
**Abdullah Siddique(2002-ag-7694)**

**Degree (BSCS)**

**Section (A)**

**Mam Raheela Naseem**

# Previous assignment ER Model



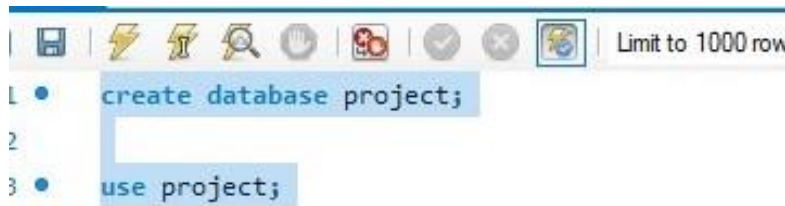
# 1.The Table satisfy 1NF,2NF & 3NF

## 2.My SQL Database Creation

create database project;

use assignment;

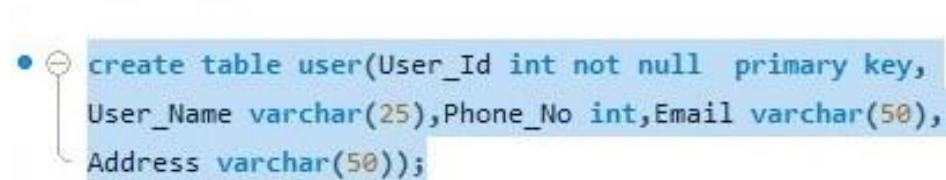
select database();



## 3.Table Creation

### User Table:

create table user(User\_Id int not null primary key,  
User\_Name varchar(25),Phone\_No int,Email varchar(50),  
Address varchar(50));



### Parking Table:

create table parking(Park\_Id int not null primary key,  
User\_Id int,vech\_reg varchar(25),  
park\_slot\_code varchar(25), drop\_car datetime,index or\_ind (User\_Id),  
constraint fk\_user foreign key(User\_Id) references user (User\_Id)  
ON DELETE CASCADE ON UPDATE RESTRICT );

```
create table parking(Park_Id int not null primary key,  
User_Id int,vech_reg varchar(25),park_slot_code varchar(25),  
drop_car datetime,index or_ind (User_Id),  
constraint fk_user foreign key(User_Id) references user (User_Id)  
ON DELETE CASCADE ON UPDATE RESTRICT );
```

### Parking Permission Status Table:

```
create table parking_permission_status(status_description varchar(25),  
park_status_code varchar(25) not null primary key);
```

```
create table parking_permission_status(status_description varchar(25),  
park_status_code varchar(25) not null primary key);
```

### Role Table:

```
create table role(role_id int not null primary key,  
role_description varchar(25));
```

```
select * from parking_permission_status;  
create table role(role_id int not null primary key, role_description varchar(25));  
insert into role values
```

### Invoice Table:

```
create table Invoice(Invoice_id int not null primary key,  
Invoice_date date,Invoice_status_code varchar(25));
```

```
create table Invoice(Invoice_id int not null primary key,  
Invoice_date date,Invoice_status_code varchar(25));  
insert into Invoice values
```

## Payment Table:

```
create table Payment(Payment_id int not null primary key,Invoice_id int ,
Payment_date date,payment_amount bigint,index or_ind(Invoice_id),
constraint fk_Invoice foreign key(Invoice_id) references Invoice (Invoice_id)
on delete cascade on update restrict);
```

```
select * from invoice;
> create table Payment(Payment_id int not null primary key,Invoice_id int ,
Payment_date date,payment_amount bigint,index or_ind(Invoice_id),
constraint fk_Invoice foreign key(Invoice_id) references Invoice (Invoice_id)
on delete cascade on update restrict);
```

## Duration Table:

```
create table Duration(Duration_Id int not null primary key,Park_Id int,
Duration_Date date,index or_ind (Park_Id),
constraint fk_parking foreign key(Park_Id) references parking(Park_Id)
ON DELETE CASCADE ON UPDATE RESTRICT );
```

```
create table Duration(Duration_Id int not null primary key,Park_Id int,
Duration_Date date,index or_ind (Park_Id),
constraint fk_parking foreign key(Park_Id) references parking(Park_Id)
ON DELETE CASCADE ON UPDATE RESTRICT );
```

## Security Department Table:

```
create table Security_department(Security_Id int not null primary key,Park_Id int, Schedule_Date
date,contact_info varchar(40), index or_ind (Park_Id), constraint fk_parking1 foreign key(Park_Id)
references parking(Park_Id) ON DELETE CASCADE ON UPDATE RESTRICT );
```

```

create table Security_department(Security_Id int not null primary key,Park_Id int,
Schedule_Date date,contact_info varchar(40), index or_ind (Park_Id),
constraint fk_parking1 foreign key(Park_Id) references parking(Park_Id)
ON DELETE CASCADE ON UPDATE RESTRICT );
insert into Security_department values

```

## Manufacture Table:

create table Manufacturer(Manufacturer\_code varchar(25) not null primary key,  
Manufacturer\_Nmae varchar(25));

```

create table Manufacturer(Manufacturer_code varchar(25) not null primary key,
Manufacturer_Nmae varchar(25));
insert into Manufacturer values

```

## Model Table:

create table Models(Model\_code varchar(25) not null primary key, Manufacturer\_code  
varchar(25),Model\_Nmae varchar(25),index or\_ind (Manufacturer\_code), constraint  
fk\_Manufacturer foreign key(Manufacturer\_code) references Manufacturer(Manufacturer\_code)  
ON DELETE CASCADE ON UPDATE RESTRICT);

```

create table Models(Model_code varchar(25) not null primary key,
Manufacturer_code varchar(25),Model_Nmae varchar(25),index or_ind (Manufacturer_code),
constraint fk_Manufacturer foreign key(Manufacturer_code) references Manufacturer(Manufacturer_code)
ON DELETE CASCADE ON UPDATE RESTRICT);

```

## Vehicle Table:

create table Vehicle(Registration\_No int not null primary key, Model\_code varchar(25),  
Manufacturer\_code varchar(25),index or\_ind (Manufacturer\_code), constraint fk\_Manufacturer1  
foreign key(Manufacturer\_code) references Manufacturer(Manufacturer\_code) ON DELETE  
CASCADE ON UPDATE RESTRICT,index or\_ind1 (Model\_code), constraint fk\_Models foreign  
key(Model\_code) references Models(Model\_code) ON DELETE CASCADE ON UPDATE RESTRICT);



```

1 create table Vehicle(Registration_No int not null primary key, Model_code varchar(25),
2 Manufacturer_code varchar(25),index or_ind1 (Manufacturer_code),
3 constraint fk_Manufacturer1 foreign key(Manufacturer_code) references Manufacturer(Manufacturer_code)
4 ON DELETE CASCADE ON UPDATE RESTRICT,index or_ind1 (Model_code),
5 constraint fk_Models foreign key(Model_code) references Models(Model_code)
6 ON DELETE CASCADE ON UPDATE RESTRICT);

```

## 4.Data Population

### User Table:

insert into user values(1001,'Bilal Arshad',04195678,'BilalArshad @ gmail.com','Peoples Colony 1 Street 1 FSD'), (1002,'Ali Arshad',04177899,'AliArshad @ gmail.com','Peoples Colony 5 Street 9 FSD'), (1003,'Aleena Akhtar',04195101,'AleenaAkhtar @ gmail.com','Shadman Colony 1 Street 1 FSD'), (1004,'Wjeeha Ali',04177678,'WjeehaAli @ gmail.com','GM Abad Street 5 FSD'), (1005,'Abdullah Ali',04196788,'AbduulahAli @ gmail.com','Islam Pur Street 5 FSD');

```

1 / Address varchar(50));
2
3 8 • insert into user values(1001,'Bilal Arshad',04195678,'BilalArshad @ gmail.com','Peoples Colony
4 (1002,'Ali Arshad',04177899,'AliArshad @ gmail.com','Peoples Colony 5 Street 9 FSD'),
5 (1003,'Aleena Akhtar',04195101,'AleenaAkhtar @ gmail.com','Shadman Colony 1 Street 1 FSD'),
6 (1004,'Wjeeha Ali',04177678,'WjeehaAli @ gmail.com','GM Abad Street 5 FSD'),
7 (1005,'Abdullah Ali',04196788,'AbduulahAli @ gmail.com','Islam Pur Street 5 FSD');
8
9 13 • select *from user;

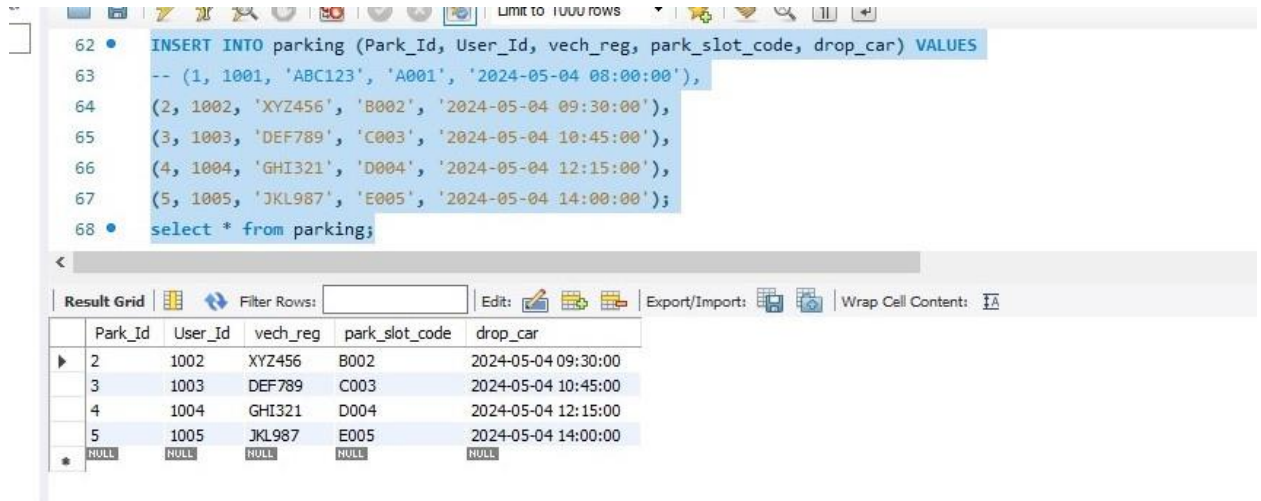
```

Result Grid

User_Id	User_Name	Phone_No	Email	Address
1002	Ibrahim	4177878	AliAshraf @ gmail.com	Peoples Colony 2 Street 7 FSD
1003	Aleena Akhtar	4195101	AleenaAkhtar @ gmail.com	Shadman Colony 1 Street 1 FSD
1004	Wjeeha Ali	4177678	WjeehaAli @ gmail.com	GM Abad Street 5 FSD
1005	Abdullah Ali	4196788	AbduulahAli @ gmail.com	Islam Pur Street 5 FSD
NULL	NULL	NULL	NULL	NULL

## Parking Table:

INSERT INTO parking (Park\_Id, User\_Id, vech\_reg, park\_slot\_code, drop\_car) VALUES -- (1, 1001, 'ABC123', 'A001', '2024-05-04 08:00:00'), (2, 1002, 'XYZ456', 'B002', '2024-05-04 09:30:00'), (3, 1003, 'DEF789', 'C003', '2024-05-04 10:45:00'), (4, 1004, 'GHI321', 'D004', '2024-05-04 12:15:00'), (5, 1005, 'JKL987', 'E005', '2024-05-04 14:00:00'); select \* from parking;



The screenshot shows a database query editor with a SQL statement and its results. The SQL statement is:

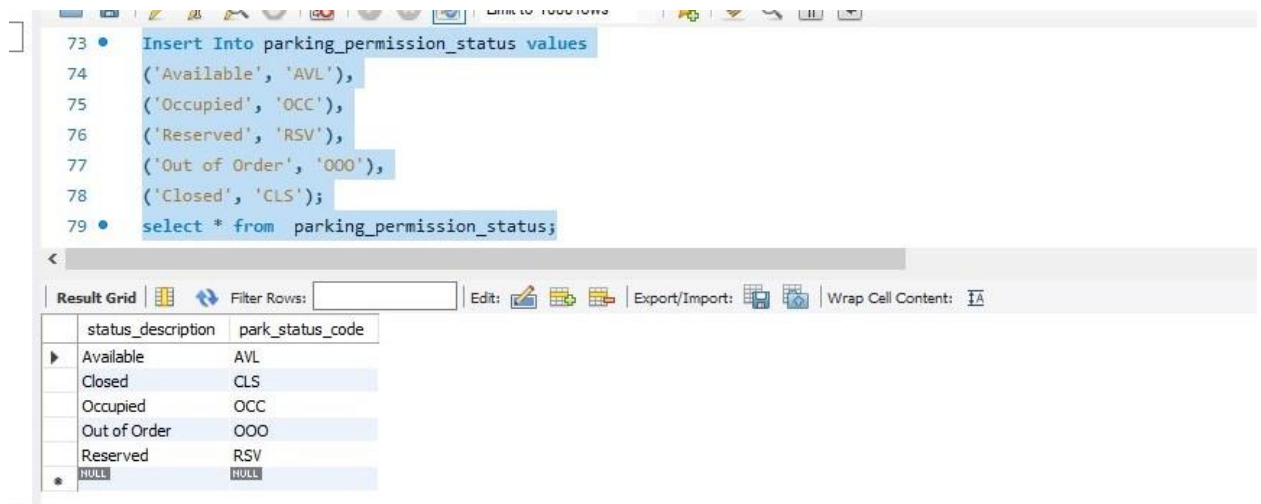
```
INSERT INTO parking (Park_Id, User_Id, vech_reg, park_slot_code, drop_car) VALUES -- (1, 1001, 'ABC123', 'A001', '2024-05-04 08:00:00'), (2, 1002, 'XYZ456', 'B002', '2024-05-04 09:30:00'), (3, 1003, 'DEF789', 'C003', '2024-05-04 10:45:00'), (4, 1004, 'GHI321', 'D004', '2024-05-04 12:15:00'), (5, 1005, 'JKL987', 'E005', '2024-05-04 14:00:00'); select * from parking;
```

The results are displayed in a table grid with the following columns: Park\_Id, User\_Id, vech\_reg, park\_slot\_code, and drop\_car. The data is as follows:

Park_Id	User_Id	vech_reg	park_slot_code	drop_car
2	1002	XYZ456	B002	2024-05-04 09:30:00
3	1003	DEF789	C003	2024-05-04 10:45:00
4	1004	GHI321	D004	2024-05-04 12:15:00
5	1005	JKL987	E005	2024-05-04 14:00:00
NULL	NULL	NULL	NULL	NULL

## Parking Permission Status Table:

Insert Into parking\_permission\_status values ('Available', 'AVL'), ('Occupied', 'OCC'), ('Reserved', 'RSV'), ('Out of Order', 'OOO'), ('Closed', 'CLS'); select \* from parking\_permission\_status;



The screenshot shows a database query editor with a SQL statement and its results. The SQL statement is:

```
Insert Into parking_permission_status values ('Available', 'AVL'), ('Occupied', 'OCC'), ('Reserved', 'RSV'), ('Out of Order', 'OOO'), ('Closed', 'CLS'); select * from parking_permission_status;
```

The results are displayed in a table grid with the following columns: status\_description and park\_status\_code. The data is as follows:

status_description	park_status_code
Available	AVL
Closed	CLS
Occupied	OCC
Out of Order	OOO
Reserved	RSV
NULL	NULL



## Role Table:

insert into role values (1, 'Administrator'), (2, 'Manager'), (3, 'Supervisor'), (4, 'Employee'), (5, 'Intern'); select \* from role;

The screenshot shows the MySQL Workbench interface. The left sidebar displays the 'SHEMAS' tree with a 'practice' database selected. The main editor window shows the following SQL script:

```
81 • insert into role values
82 (1, 'Administrator'),
83 (2, 'Manager'),
84 (3, 'Supervisor'),
85 (4, 'Employee'),
86 (5, 'Intern');
87 • select * from role;
```

The 'Result Grid' below the script shows the output of the 'select \* from role;' query:

role_id	role_description
1	Administrator
2	Manager
3	Supervisor
4	Employee
5	Intern

The 'Output' pane at the bottom shows the execution log with the following entries:

#	Time	Action	Message	Duration / Fetch
5	14:07:33	create table parking_permission_status(status_description varchar(25), park_status_code var...	0 row(s) affected	0.156 sec
6	14:07:40	insert into parking_permission_status values ('Available', 'AVL'), ('Occupied', 'OCC'), ('Reserv...	5 row(s) affected Records: 5 Duplicates: 0 Warnings: 0	0.015 sec
7	14:07:40	select * from parking_permission_status LIMIT 0, 1000	5 row(s) returned	0.000 sec / 0.000 sec
8	14:08:26	create table role(role_id int not null primary key, role_description varchar(25))	0 row(s) affected	0.141 sec
9	14:08:35	insert into role values (1, 'Administrator'), (2, 'Manager'), (3, 'Supervisor'), (4, 'Employee'), (5, 'I...	5 row(s) affected Records: 5 Duplicates: 0 Warnings: 0	0.016 sec
10	14:08:35	select * from role LIMIT 0, 1000	5 row(s) returned	0.000 sec / 0.000 sec

## Invoice Table:

create table Invoice(Invoice\_id int not null primary key, Invoice\_date date, Invoice\_status\_code varchar(25)); insert into Invoice values (1, '2024-05-01', 'PAID'), (2, '2024-05-02', 'PENDING'), (3, '2024-05-03', 'PAID'), (4, '2024-05-04', 'PENDING'), (5, '2024-05-05', 'PAID'); select \* from Invoice;

The screenshot shows the MySQL Workbench interface. The left sidebar displays the 'SHEMAS' tree with a 'practice' database selected. The main editor window shows the following SQL script:

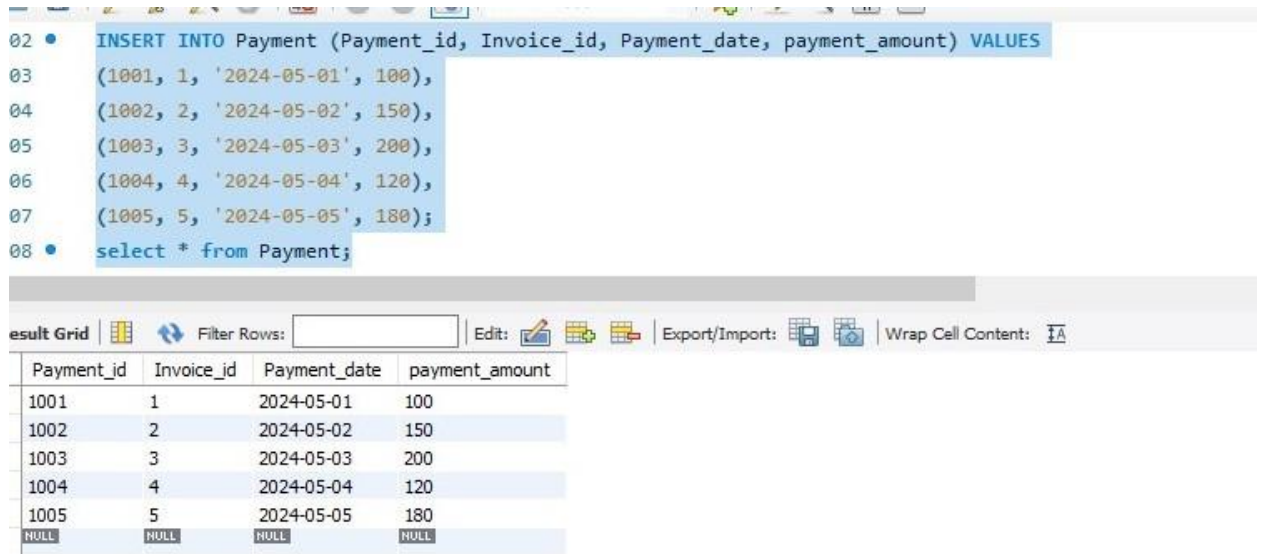
```
91 • insert into Invoice values
92 (1, '2024-05-01', 'PAID'),
93 (2, '2024-05-02', 'PENDING'),
94 (3, '2024-05-03', 'PAID'),
95 (4, '2024-05-04', 'PENDING'),
96 (5, '2024-05-05', 'PAID');
97 • select * from Invoice;
```

The 'Result Grid' below the script shows the output of the 'select \* from Invoice;' query:

Invoice_id	Invoice_date	Invoice_status_code
1	2024-05-01	PAID
2	2024-05-02	PENDING
3	2024-05-03	PAID
4	2024-05-04	PENDING
5	2024-05-05	PAID
NULL	NULL	NULL

## Payment Table:

INSERT INTO Payment (Payment\_id, Invoice\_id, Payment\_date, payment\_amount) VALUES (1001, 1, '2024-05-01', 100), (1002, 2, '2024-05-02', 150), (1003, 3, '2024-05-03', 200), (1004, 4, '2024-05-04', 120), (1005, 5, '2024-05-05', 180); select \* from Payment;



The screenshot shows a SQL IDE interface. The top pane contains the following SQL code:

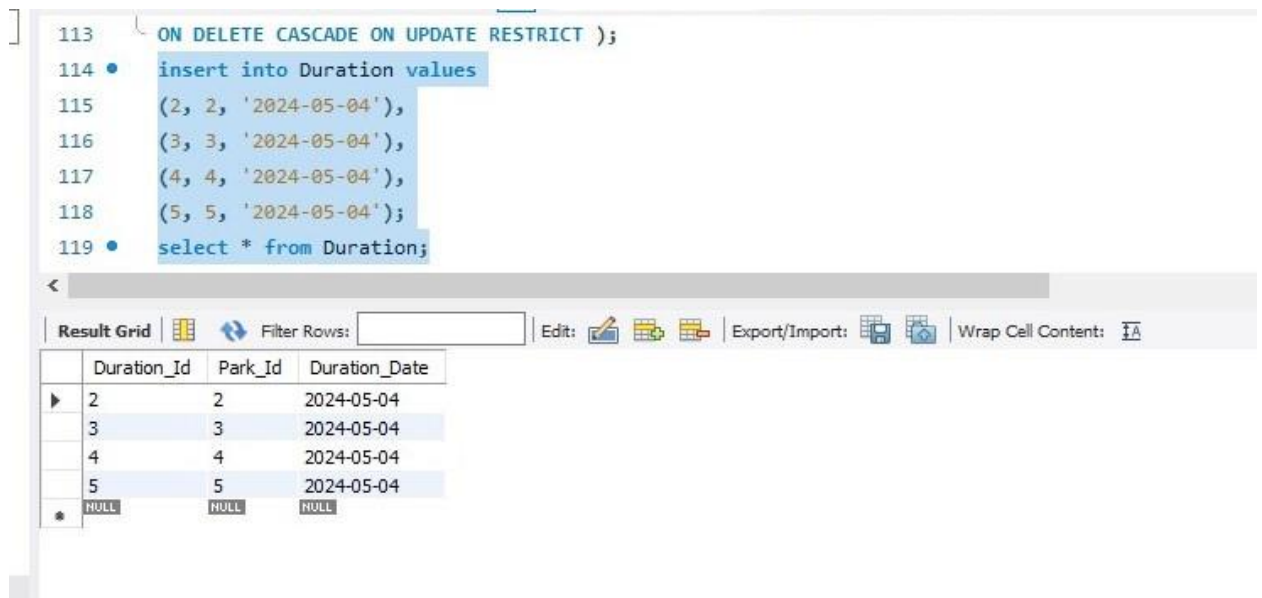
```
02 • INSERT INTO Payment (Payment_id, Invoice_id, Payment_date, payment_amount) VALUES
03     (1001, 1, '2024-05-01', 100),
04     (1002, 2, '2024-05-02', 150),
05     (1003, 3, '2024-05-03', 200),
06     (1004, 4, '2024-05-04', 120),
07     (1005, 5, '2024-05-05', 180);
08 • select * from Payment;
```

The bottom pane displays the 'result Grid' with the following data:

Payment_id	Invoice_id	Payment_date	payment_amount
1001	1	2024-05-01	100
1002	2	2024-05-02	150
1003	3	2024-05-03	200
1004	4	2024-05-04	120
1005	5	2024-05-05	180
NULL	NULL	NULL	NULL

## Duration Table:

insert into Duration values (1, 1, '2024-05-04'), (2, 2, '2024-05-04'), (3, 3, '2024-05-04'), (4, 4, '2024-05-04'), (5, 5, '2024-05-04'); select \* from Duration;



The screenshot shows a SQL IDE interface. The top pane contains the following SQL code:

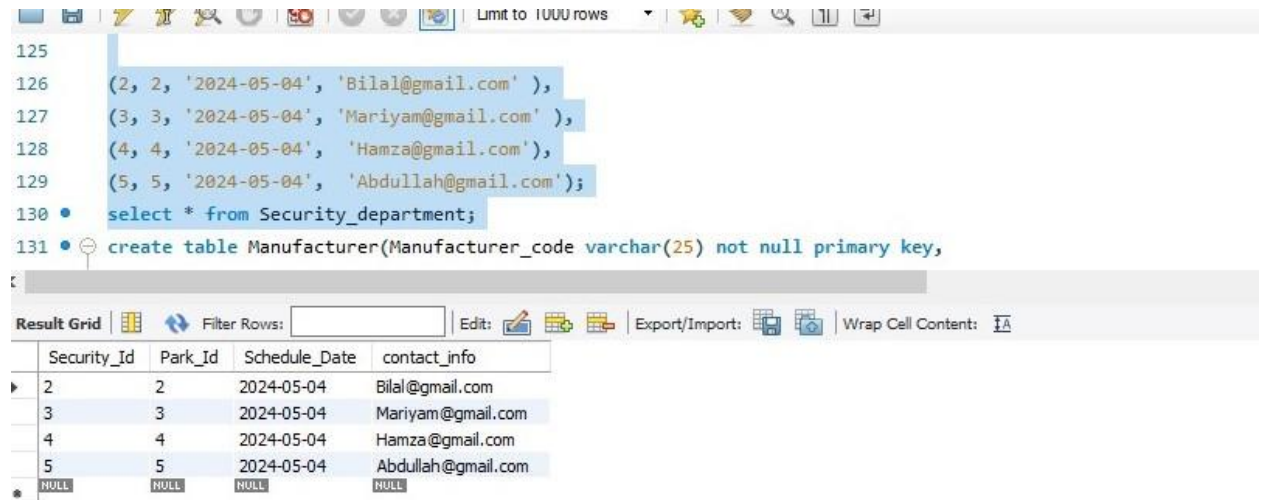
```
113 ON DELETE CASCADE ON UPDATE RESTRICT );
114 • insert into Duration values
115     (2, 2, '2024-05-04'),
116     (3, 3, '2024-05-04'),
117     (4, 4, '2024-05-04'),
118     (5, 5, '2024-05-04');
119 • select * from Duration;
```

The bottom pane displays the 'Result Grid' with the following data:

Duration_Id	Park_Id	Duration_Date
2	2	2024-05-04
3	3	2024-05-04
4	4	2024-05-04
5	5	2024-05-04
NULL	NULL	NULL

## Security Department Table:

insert into Security\_department values (1, 1, '2024-05-04', 'Ali@gmail.com'), (2, 2, '2024-05-04', 'Bilal@gmail.com' ), (3, 3, '2024-05-04', 'Mariyam@gmail.com' ), (4, 4, '2024-05-04', 'Hamza@gmail.com'), (5, 5, '2024-05-04', 'Abdullah@gmail.com'); select \* from Security\_department;



The screenshot shows a database IDE with a SQL editor and a result grid. The SQL editor contains the following code:

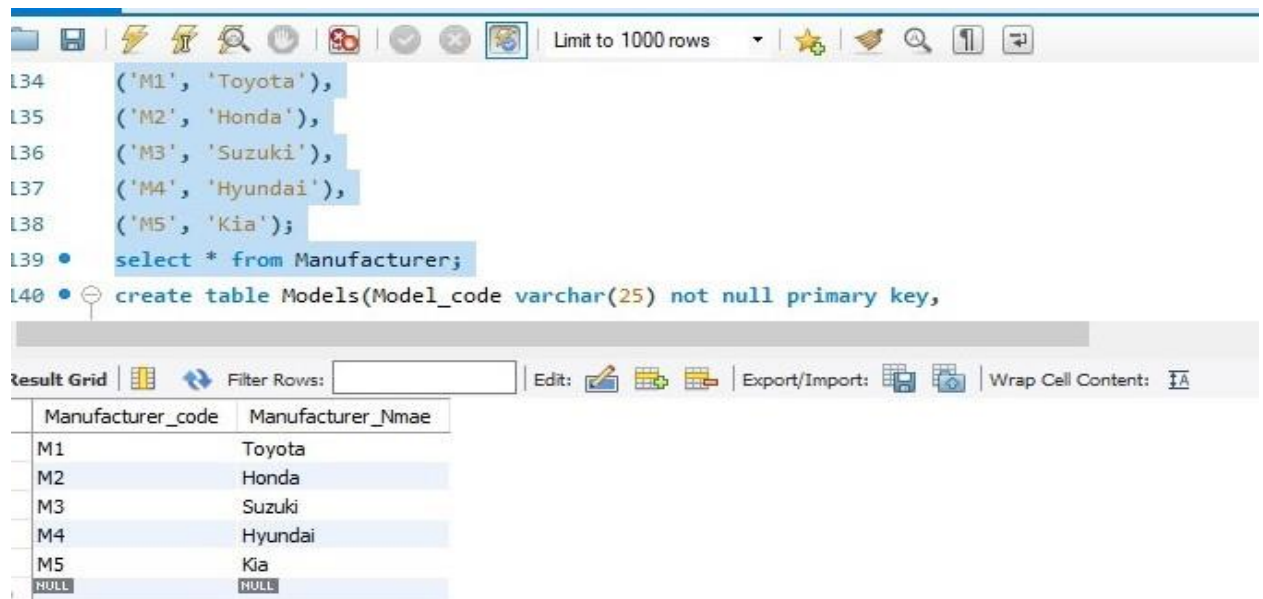
```
125
126 (2, 2, '2024-05-04', 'Bilal@gmail.com' ),
127 (3, 3, '2024-05-04', 'Mariyam@gmail.com' ),
128 (4, 4, '2024-05-04', 'Hamza@gmail.com'),
129 (5, 5, '2024-05-04', 'Abdullah@gmail.com');
130 • select * from Security_department;
131 • create table Manufacturer(Manufacturer_code varchar(25) not null primary key,
```

The result grid displays the data for the Security\_department table:

Security_Id	Park_Id	Schedule_Date	contact_info
2	2	2024-05-04	Bilal@gmail.com
3	3	2024-05-04	Mariyam@gmail.com
4	4	2024-05-04	Hamza@gmail.com
5	5	2024-05-04	Abdullah@gmail.com
NULL	NULL	NULL	NULL

## Manufacture Table:

insert into Manufacturer values ('M1', 'Toyota'), ('M2', 'Honda'), ('M3', 'Suzuki'), ('M4', 'Hyundai'), ('M5', 'Kia'); select \* from Manufacturer;



The screenshot shows a database IDE with a SQL editor and a result grid. The SQL editor contains the following code:

```
L34 ('M1', 'Toyota'),
L35 ('M2', 'Honda'),
L36 ('M3', 'Suzuki'),
L37 ('M4', 'Hyundai'),
L38 ('M5', 'Kia');
L39 • select * from Manufacturer;
L40 • create table Models(Model_code varchar(25) not null primary key,
```

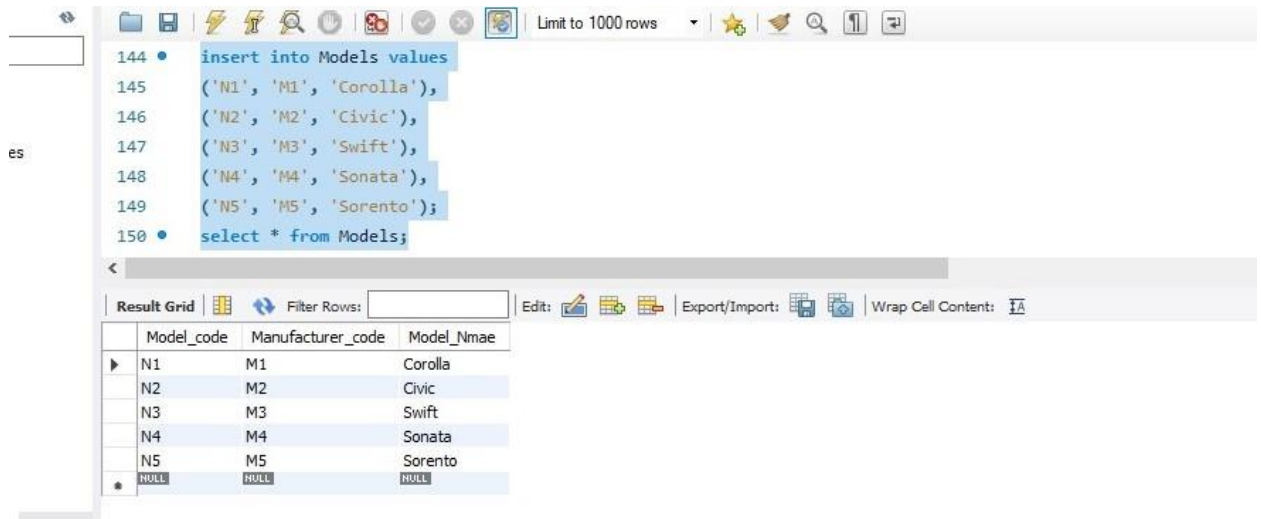
The result grid displays the data for the Manufacturer table:

Manufacturer_code	Manufacturer_Nmae
M1	Toyota
M2	Honda
M3	Suzuki
M4	Hyundai
M5	Kia
NULL	NULL

Manufacturer 10 x

## Model Table:

insert into Models values ('N1', 'M1', 'Corolla'), ('N2', 'M2', 'Civic'), ('N3', 'M3', 'Swift'), ('N4', 'M4', 'Sonata'), ('N5', 'M5', 'Sorento'); select \* from Models;



The screenshot shows a database IDE with a SQL editor and a result grid. The SQL editor contains the following code:

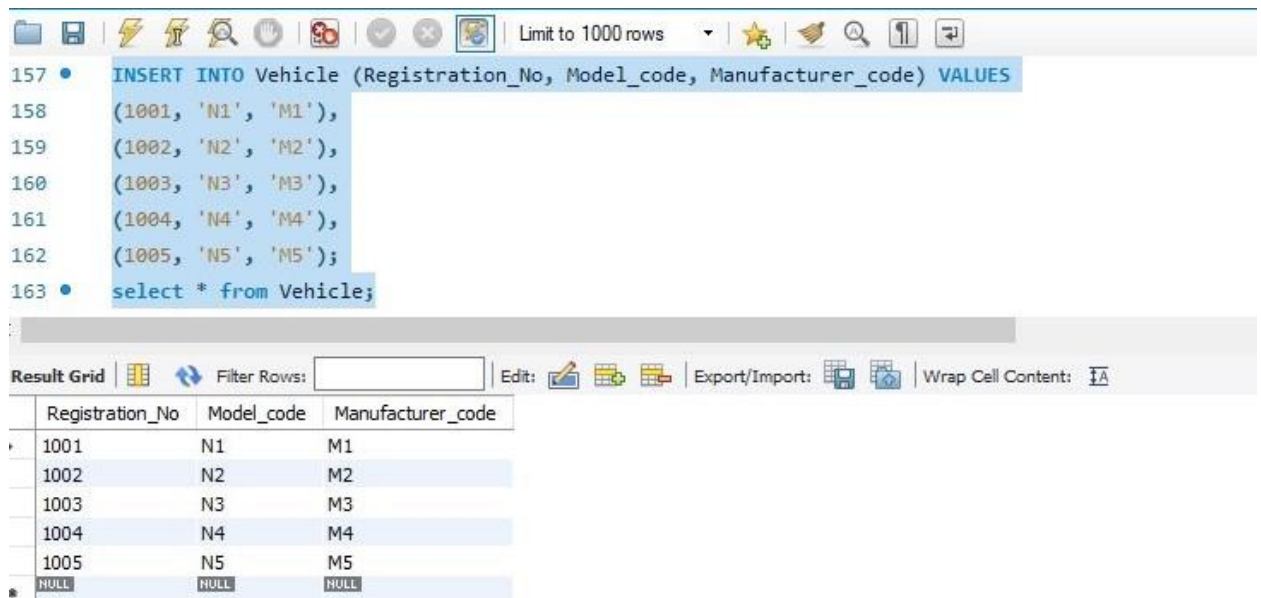
```
144 • insert into Models values
145 ('N1', 'M1', 'Corolla'),
146 ('N2', 'M2', 'Civic'),
147 ('N3', 'M3', 'Swift'),
148 ('N4', 'M4', 'Sonata'),
149 ('N5', 'M5', 'Sorento');
150 • select * from Models;
```

The result grid displays the following data:

Model_code	Manufacturer_code	Model_Nmae
N1	M1	Corolla
N2	M2	Civic
N3	M3	Swift
N4	M4	Sonata
N5	M5	Sorento
NULL	NULL	NULL

## Vehicle Table:

INSERT INTO Vehicle (Registration\_No, Model\_code, Manufacturer\_code) VALUES (1001, 'N1', 'M1'), (1002, 'N2', 'M2'), (1003, 'N3', 'M3'), (1004, 'N4', 'M4'), (1005, 'N5', 'M5'); select \* from Vehicle;



The screenshot shows a database IDE with a SQL editor and a result grid. The SQL editor contains the following code:

```
157 • INSERT INTO Vehicle (Registration_No, Model_code, Manufacturer_code) VALUES
158 (1001, 'N1', 'M1'),
159 (1002, 'N2', 'M2'),
160 (1003, 'N3', 'M3'),
161 (1004, 'N4', 'M4'),
162 (1005, 'N5', 'M5');
163 • select * from Vehicle;
```

The result grid displays the following data:

Registration_No	Model_code	Manufacturer_code
1001	N1	M1
1002	N2	M2
1003	N3	M3
1004	N4	M4
1005	N5	M5
NULL	NULL	NULL

# 5.View Creation

```
create view user_view As
select User_Id ,User_Name
from user;
select * from user_view;
```

The screenshot displays the SQL Developer interface. The top pane shows a SQL script for creating a view and querying it. The bottom pane shows the execution results, including a table of user data and an action log.

**SQL Script:**

```
7 Address varchar(50));
8 insert into user values(1001,'Bilal Arshad',04195678,'BilalArshad @ gmail.com','Peoples Colony 1 Street 1 FSD'),
9 (100,'Bilal Arshad',04177878,'AliAshraf @ gmail.com','Peoples Colony 2 Street 7 FSD'),
10 (1003,'Aleena Akhtar',04195101,'AleenaAkhtar @ gmail.com','Shadman Colony 1 Street 1 FSD'),
11 (1004,'Wjeeha Ali',04177678,'WjeehaAli @ gmail.com','GH Abad Street 5 FSD'),
12 (1005,'Abdullah Ali',04196788,'AbduulahAli @ gmail.com','Islam Pur Street 5 FSD');
13 create view user_view As
14 select User_Id ,User_Name
15 from user;
16 select * from user_view;
```

**Result Grid:**

User_Id	User_Name
101	zahid
102	ahmed
103	farhan
104	abdullah
105	irfan

**Action Output:**

#	Time	Action	Message	Duration / Fetch
1	17:23:32	use project	0 row(s) affected	0.000 sec
2	17:23:57	create view user_view As select User_Id ,User_Name from user	Error Code: 1050. Table 'user_view' already exists	0.032 sec
3	17:24:45	select * from user_view LIMIT 0, 1000	5 row(s) returned	0.015 sec / 0.000 sec

```
create view parking_view1 As
select Park_Id,User_Id ,vech_reg
from parking
where Park_Id is not null with check option ;
select * from parking_view1;
```

Query 1 sql project\* SQL File 4\*

```

65 (4, 1004, 'GHI321', 'D004', '2024-05-04 12:15:00'),
66 (5, 1005, 'JKL987', 'E005', '2024-05-04 14:00:00');
67 • select * from parking;
68 • create view parking_view1 As
69 select Park_Id,User_Id ,vech_reg
70 from parking
71 where Park_Id is not null with check option ;
72 • select * from parking_view1;
73
74

```

Result Grid

Park_Id	User_Id	vech_reg
2	1002	XYZ456
3	1003	DEF789
4	1004	GHI321
5	1005	JKL987

parking\_view1 5 x

Output

Action Output

#	Time	Action	Message	Duration / Fetch
18	17:38:34	INSERT INTO parking (Park_Id,User_Id ,vech_reg, park_slot_code, drop_car) VALUES (1, 1001, 'ABC123', '...	Error Code: 1062. Duplicate entry '2' for key 'parking.PRIMARY'	0.000 sec
19	17:38:41	select * from parking LIMIT 0, 1000	4 row(s) returned	0.000 sec / 0.000 sec
20	17:38:59	select * from parking LIMIT 0, 1000	4 row(s) returned	0.000 sec / 0.000 sec
21	17:40:28	create view parking_view1 As select Park_Id,User_Id ,vech_reg from parking where Park_Id is not null with c...	0 row(s) affected	0.016 sec
22	17:40:43	select * from parking_view1 LIMIT 0, 1000	4 row(s) returned	0.000 sec / 0.000 sec

DELETE FROM parking\_view1

WHERE Park\_Id = 3;

select \* from parking\_view1;

Query 1 sql project\* SQL File 4\*

```

68 • create view parking_view1 As
69 select Park_Id,User_Id ,vech_reg
70 from parking
71 where Park_Id is not null with check option ;
72 • select * from parking_view1;
73 • DELETE FROM parking_view1
74 WHERE Park_Id = 3;
75 • select * from parking_view1;
76
77

```

Result Grid

Park_Id	User_Id	vech_reg
2	1002	XYZ456
4	1004	GHI321
5	1005	JKL987

parking\_view1 6 x

Output

Action Output

#	Time	Action	Message	Duration / Fetch
21	17:40:28	create view parking_view1 As select Park_Id,User_Id ,vech_reg from parking where Park_Id is not null with c...	0 row(s) affected	0.016 sec
22	17:40:43	select * from parking_view1 LIMIT 0, 1000	4 row(s) returned	0.000 sec / 0.000 sec
23	17:44:02	DELETE FROM parking_view WHERE Park_Id = 3	Error Code: 1146. Table 'project1.parking_view' doesn't exist	0.000 sec
24	17:44:20	DELETE FROM parking_view1 WHERE Park_Id = 3	1 row(s) affected	0.000 sec
25	17:44:45	select * from parking_view1 LIMIT 0, 1000	3 row(s) returned	0.000 sec / 0.000 sec



## 6. Transactions

Commit:

```
SET autocommit=0;
```

```
START TRANSACTION;
```

```
UPDATE payment
```

```
SET payment_amount=200
```

```
WHERE payment_id=1005;
```

```
COMMIT;
```

```
127 • COMMIT;
128 • START TRANSACTION;
129 • UPDATE payment
130 • SET payment_amount=200
131 • WHERE payment_id=1005;
132 • select * from payment;
133 • commit;
```

Result Grid

	Payment_id	Invoice_id	Payment_date	payment_amount
▶	1001	1	2024-05-01	100
	1002	2	2024-05-02	150
	1003	3	2024-05-03	200
	1004	4	2024-05-04	120
	1005	5	2024-05-05	200

payment 24 x

Output

Action Output

#	Time	Action	Message
✓ 149	15:17:22	UPDATE payment SET payment_amount=200 WHERE payment_id=1005	0 row(s) affected Rows matched: 1 Changed: 0 Warnings: 0
✓ 150	15:17:22	select * from payment LIMIT 0, 1000	5 row(s) returned
✓ 151	15:17:22	commit	0 row(s) affected

Rollback:

```
START TRANSACTION;
```

```
UPDATE models
```

```
SET model_name= 'Lexus'
```

```
WHERE manufacturer_code= 'MS';
```

```
Rollback;
```

```

128 • START TRANSACTION;
129 • UPDATE models
130 • SET Model_Nmae='Lexus'
131 • WHERE Manufacturer_code='M5';
132 • select * from models;
133 • rollback;
134 • select * from models;

```

Result Grid | Filter Rows: | Edit: | Export/Import: | Wrap Cell Content: |

Model_code	Manufacturer_code	Model_Nmae
N1	M1	Corolla
N2	M2	Civic
N3	M3	Swift
N4	M4	Sonata
N5	M5	Lexus

models 21 x

Output

Action Output

#	Time	Action	Message
✓ 139	15:13:58	START TRANSACTION	0 row(s) affected
✓ 140	15:13:58	UPDATE models SET Model_Nmae='Lexus' WHERE Manufacturer_code='M5'	1 row(s) affected Rows matched: 1 Changed: 1 Warnings: 0
✓ 141	15:13:58	select * from models LIMIT 0, 1000	5 row(s) returned

## 7.Advanced SQL Queries

Delete:

delete from user

where User\_id=1001;

```

10 (1003,'Aleena Akhtar','04195101','AleenaAkhtar @ gmail.com','Shadman Colony 1 Street 1 FSD'),
11 (1004,'Wjeeha Ali','04177678','WjeehaAli @ gmail.com','GM Abad Street 5 FSD'),
12 (1005,'Abdullah Ali','04196788','AbduulahAli @ gmail.com','Islam Pur Street 5 FSD');
13 delete from user
14 where User_id=1001;
15 • select * from user;
16 • update user
17 set User_id=1002

```

Limit to 1000 rows

Result Grid | Filter Rows: | Edit: | Export/Import: | Wrap Cell Content: |

User_Id	User_Name	Phone_No	Email	Address
1002	Bilal Arshad	4177878	AliAshraf @ gmail.com	Peoples Colony 2 Street 7 FSD
1003	Aleena Akhtar	4195101	AleenaAkhtar @ gmail.com	Shadman Colony 1 Street 1 FSD
1004	Wjeeha Ali	4177678	WjeehaAli @ gmail.com	GM Abad Street 5 FSD
1005	Abdullah Ali	4196788	AbduulahAli @ gmail.com	Islam Pur Street 5 FSD
* NULL	NULL	NULL	NULL	NULL

Import records from an external file

## Update

update user

set User\_name='Ibrahim'

where User\_id=1002;

```
19
20 • update user
21   set User_name='Ibrahim'
22   where User_id=1002;
23 • select * from user;
24 • select User_Name,Phone_No,vech_reg
25   from user,parking
26   where user.User_id=parking.Park_id;
```

Result Grid

	User_Id	User_Name	Phone_No	Email	Address
▶	1002	Ibrahim	4177878	AliAshraf @ gmail.com	Peoples Colony 2 Street 7 FSD
	1003	Aleena Akhtar	4195101	AleenaAkhtar @ gmail.com	Shadman Colony 1 Street 1 FSD
	1004	Wjeeha Ali	4177678	WjeehaAli @ gmail.com	GM Abad Street 5 FSD
	1005	Abdullah Ali	4196788	AbduulahAli @ gmail.com	Islam Pur Street 5 FSD
*	NULL	NULL	NULL	NULL	NULL

## Join:

select User\_Name,Phone\_No,vech\_reg

from user,parking

where user.User\_id=parking.User\_i

```
20 • update user
21   set User_name='Ibrahim'
22   where User_id=1002;
23 • select * from user;
24 • select User_Name,Phone_No,vech_reg
25   from user,parking
26   where user.User_id=parking.User_id;
27 • select User_Name,park_slot_code,vech_reg
```

Limit to 1000 rows

Result Grid

	User_Name	Phone_No	vech_reg
▶	Ibrahim	4177878	XYZ456
	Aleena Akhtar	4195101	DEF789
	Wjeeha Ali	4177678	GHI321
	Abdullah Ali	4196788	JKL987

Inner Join:

```
select User_Name,park_slot_code,vech_reg
```

```
from user
```

```
inner join parking
```

```
on user.User_id=parking.User_id;
```

```
26   where user.User_id=parking.User_id;
27   • select User_Name,park_slot_code,vech_reg
28   from user
29   inner join parking
30   on user.User_id=parking.User_id;
31   • select User_Name,park_slot_code,vech_reg,phone_
32   from user
33   left join parking
```

< Result Grid Filter Rows:  Export: Wrap

	User_Name	park_slot_code	vech_reg
▶	Ibrahim	B002	XYZ456
	Aleena Akhtar	C003	DEF789
	Wjeeha Ali	D004	GHI321
	Abdullah Ali	E005	JKL987

Left Join:

```
select User_Name,park_slot_code,vech_reg,phone_no
```

```
from user
```

```
left join parking on user.User_id=parking.User_id;
```

Limit to 1000 rows ▼

```
28   from user
29   inner join parking
30   on user.User_id=parking.User_id;
31   • select User_Name,park_slot_code,vech_reg,phone_no
32   from user
33   left join parking
34   on user.User_id=parking.User_id;
35   • select User_Name,park_slot_code,vech_reg,phone_no,Email
```

< Result Grid Filter Rows:  Export: Wrap Cell Content

	User_Name	park_slot_code	vech_reg	phone_no
▶	Ibrahim	B002	XYZ456	4177878
	Aleena Akhtar	C003	DEF789	4195101
	Wjeeha Ali	D004	GHI321	4177678
	Abdullah Ali	E005	JKL987	4196788

Right Join:

select User\_Name,park\_slot\_code,vech\_reg,phone\_no,Email

from user

right join parking

on user.User\_id=parking.User\_id;

```
32  from user
33  left join parking
34  on user.User_id=parking.User_id;
35  • select User_Name,park_slot_code,vech_reg,phone_no,Email
36  from user
37  right join parking
38  on user.User_id=parking.User_id;
39  • select count(*)
```

User_Name	park_slot_code	vech_reg	phone_no	Email
Ibrahim	B002	XYZ456	4177878	AliAshraf @ gmail.com
Aleena Akhtar	C003	DEF789	4195101	AleenaAkhtar @ gmail.com
Wjeeha Ali	D004	GHI321	4177678	WjeehaAli @ gmail.com
Abdullah Ali	E005	JKL987	4196788	Ali @ gmail.com

Aggregate Function:

Count(\*):

select count(\*)

From user

where User\_name='Bilal Arshad';

```
35  • select User_Name,park_slot_code,vech_reg,phone_no,Email
36  from user
37  right join parking
38  on user.User_id=parking.User_id;
39  • select count(*)
40  From user
41  where User_name='Bilal Arshad';
42  • select count(User_id)
```

count(*)
0



Count(Column\_Name):

```
select count(User_id)
```

```
from user;
```

The screenshot shows the MySQL Workbench interface. The query editor contains the following SQL code:

```
41 where User_name='Bilal Arshad';
42 select count(User_id)
43 from user;
44 select count( distinct User_name)
45 from user;
46 select avg(payment_amount)
47 from payment;
48 select sum(payment_amount)
```

The 'Result Grid' tab is active, showing the result of the query. The result is a single row with the value 4.

count(User_id)
4

The bottom status bar indicates 'Query Completed'.

Count( Distinct Column\_Name):

```
select count( distinct User_name)
```

```
from user;
```

The screenshot shows the MySQL Workbench interface. The query editor contains the following SQL code:

```
41 where User_name='Bilal Arshad';
42 select count(User_id)
43 from user;
44 select count( distinct User_name)
45 from user;
46 select avg(payment_amount)
47 from payment;
48 select sum(payment_amount)
```

The 'Result Grid' tab is active, showing the result of the query. The result is a single row with the value 4.

count( distinct User_name)
4



AVG :

```
select avg(payment_amount)
from payment;
```

```
46 • select avg(payment_amount)
47   from payment;
48 • select sum(payment_amount)
49   from payment;
50 • select max(payment_amount)
51   from payment;
52 • select min(payment_amount)
53   from payment;
```

<

Result Grid | | Filter Rows:  | Export: | Wrap Cell Content:

	avg(payment_amount)
▶	150.0000

SUM:

```
select sum(payment_amount)
from payment;
```

```
46 • select avg(payment_amount)
47   from payment;
48 • select sum(payment_amount)
49   from payment;
50 • select max(payment_amount)
51   from payment;
52 • select min(payment_amount)
53   from payment;
```

<

Result Grid | | Filter Rows:  | Export: | Wrap Cell Content:

	sum(payment_amount)
▶	750





[Toggle w](#)

MIN:

```
select min(payment_amount)
from payment;
```

```
48 • select sum(payment_amount)
49   from payment;
50 • select max(payment_amount)
51   from payment;
52 • select min(payment_amount)
53   from payment;
54 • select * from user;
55
```

<

Result Grid |   Filter Rows:  | Export:  | Wrap Cell Content: 




min(payment_amount)
100

MAX:

```
select max(payment_amount)
from payment;
```

```
46 • select avg(payment_amount)
47   from payment;
48 • select sum(payment_amount)
49   from payment;
50 • select max(payment_amount)
51   from payment;
52 • select min(payment_amount)
53   from payment;
```

<

Result Grid |   Filter Rows:  | Export:  | Wra

max(payment_amount)
200