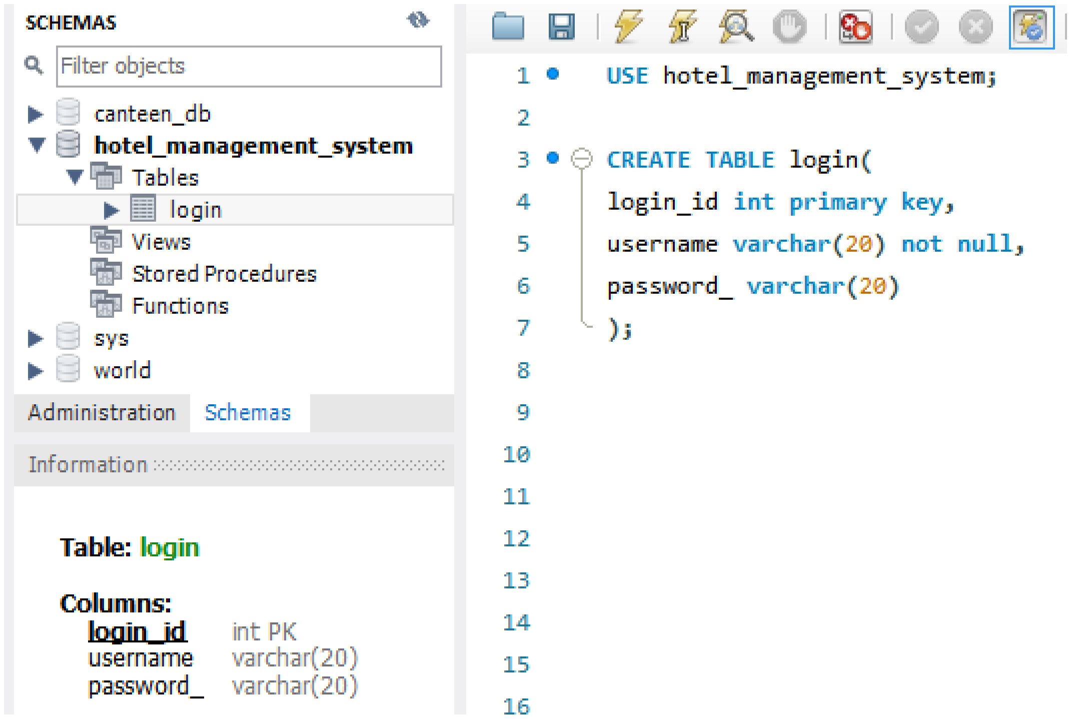
Expt 3 :

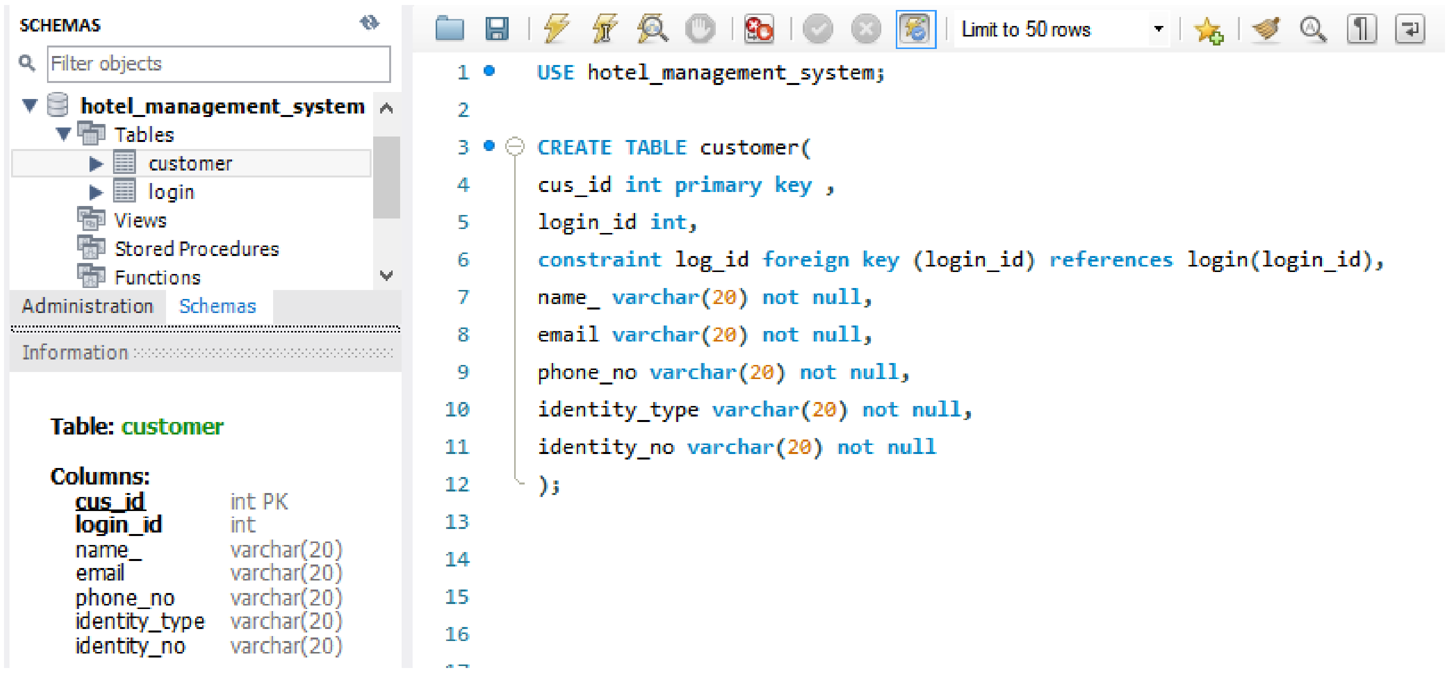


```
USE hotel_management_system;

CREATE TABLE login(
login_id int primary key,
username varchar(20) not null,
password_ varchar(20)
);
```

Table: login

Columns:

login_id	int PK
username	varchar(20)
password_	varchar(20)

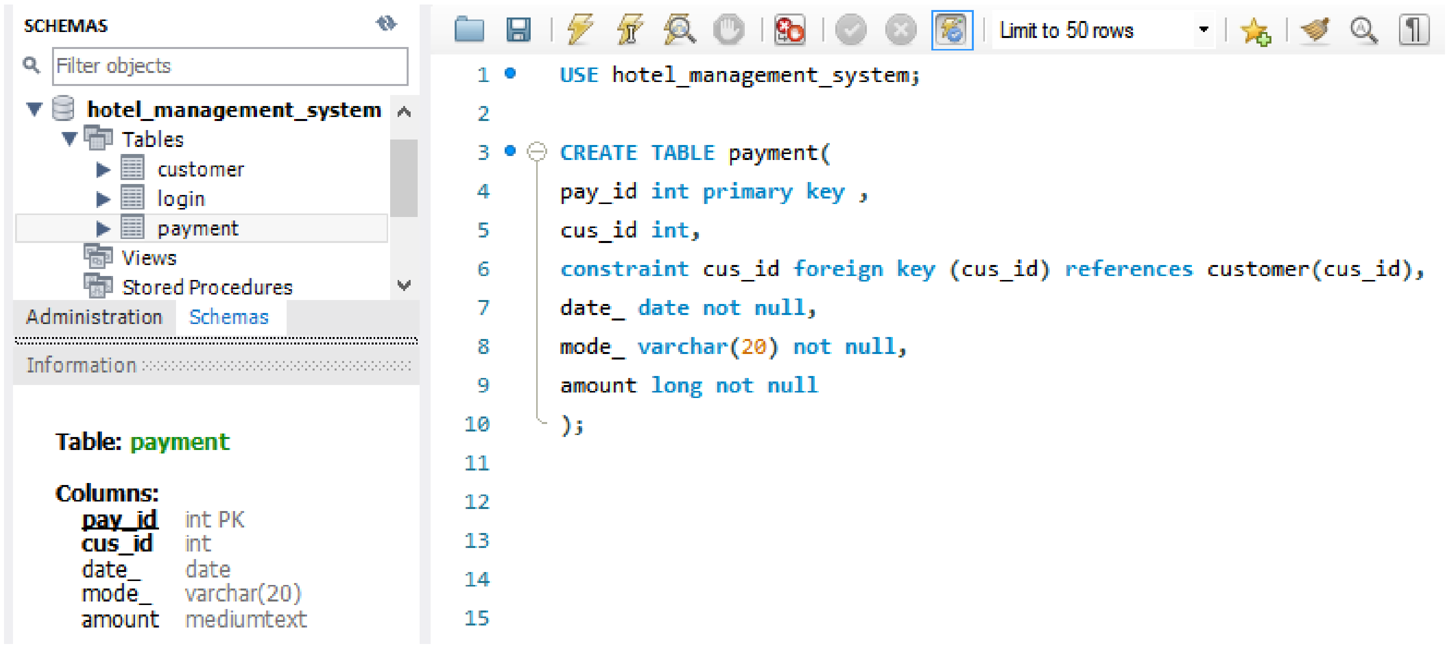


```
USE hotel_management_system;

CREATE TABLE customer(
cus_id int primary key ,
login_id int,
constraint log_id foreign key (login_id) references login(login_id),
name_ varchar(20) not null,
email varchar(20) not null,
phone_no varchar(20) not null,
identity_type varchar(20) not null,
identity_no varchar(20) not null
);
```

Table: customer

Columns:

cus_id	int PK
login_id	int
name_	varchar(20)
email	varchar(20)
phone_no	varchar(20)
identity_type	varchar(20)
identity_no	varchar(20)

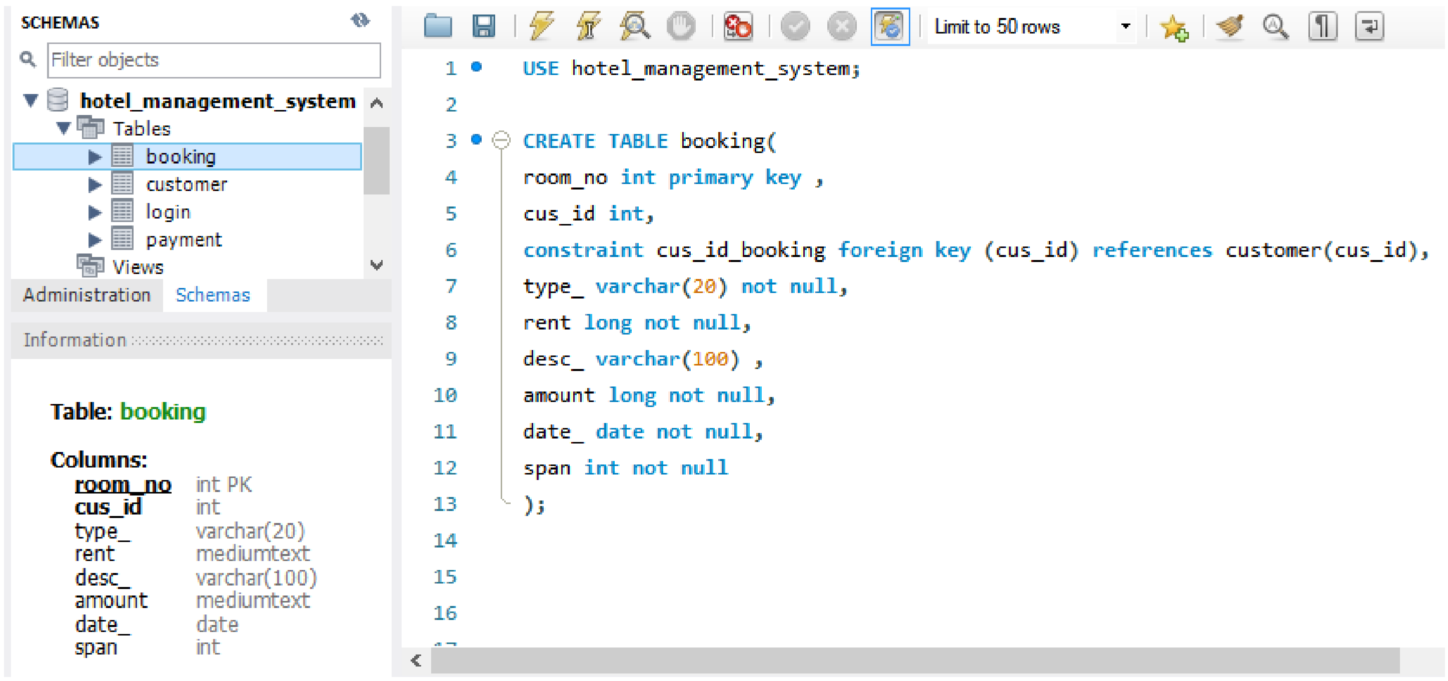


```sql
USE hotel_management_system;

CREATE TABLE payment(
pay_id int primary key ,
cus_id int,
constraint cus_id foreign key (cus_id) references customer(cus_id),
date_ date not null,
mode_ varchar(20) not null,
amount long not null
);
```

Table: payment

Columns:

pay_id	int PK
cus_id	int
date_	date
mode_	varchar(20)
amount	mediumtext



```sql
USE hotel_management_system;

CREATE TABLE booking(
room_no int primary key ,
cus_id int,
constraint cus_id_booking foreign key (cus_id) references customer(cus_id),
type_ varchar(20) not null,
rent long not null,
desc_ varchar(100) ,
amount long not null,
date_ date not null,
span int not null
);
```

Table: booking

Columns:

room_no	int PK
cus_id	int
type_	varchar(20)
rent	mediumtext
desc_	varchar(100)
amount	mediumtext
date_	date
span	int

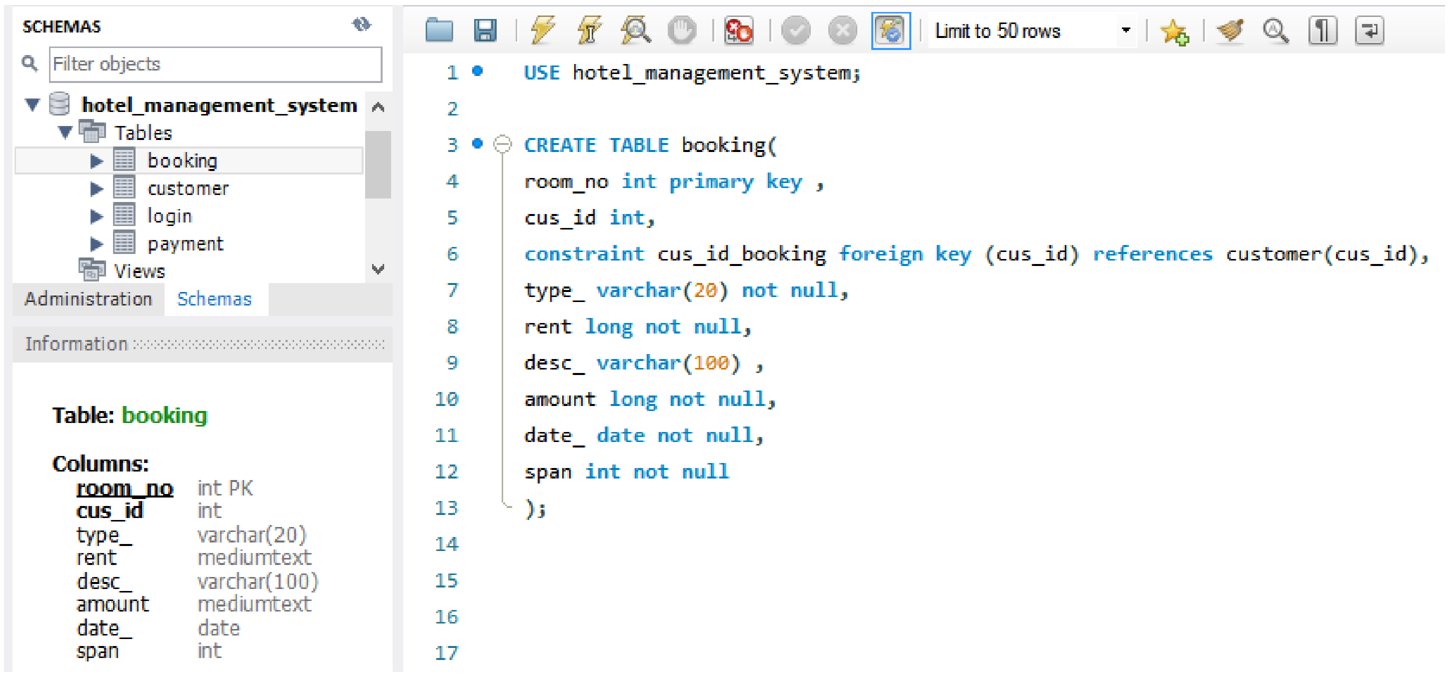


Table: booking

Columns:

room_no	int PK
cus_id	int
type_	varchar(20)
rent	mediumtext
desc_	varchar(100)
amount	mediumtext
date_	date
span	int

```sql
USE hotel_management_system;

CREATE TABLE booking(
room_no int primary key ,
cus_id int,
constraint cus_id_booking foreign key (cus_id) references customer(cus_id),
type_ varchar(20) not null,
rent long not null,
desc_ varchar(100) ,
amount long not null,
date_ date not null,
span int not null
);
```

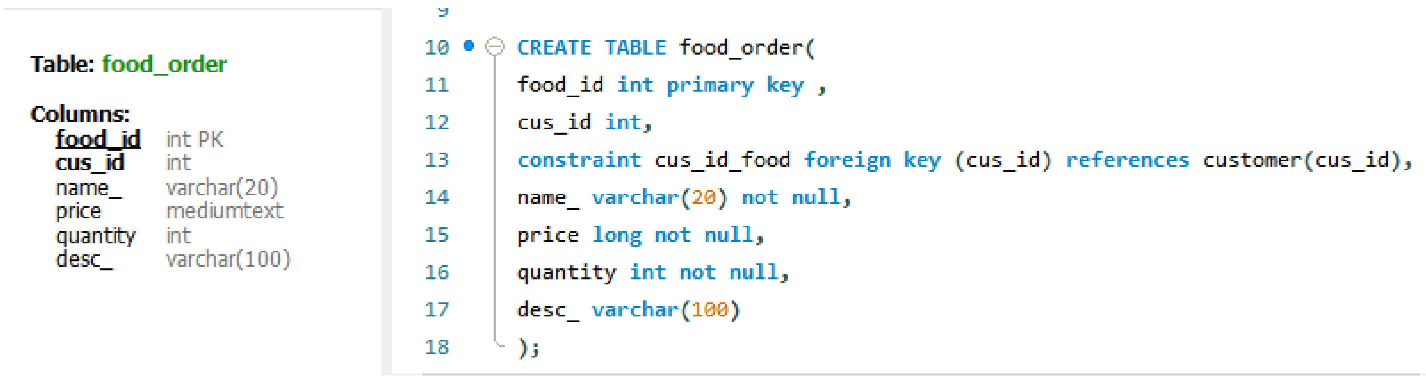


Table: food_order

Columns:

food_id	int PK
cus_id	int
name_	varchar(20)
price	mediumtext
quantity	int
desc_	varchar(100)

```sql
CREATE TABLE food_order(
food_id int primary key ,
cus_id int,
constraint cus_id_food foreign key (cus_id) references customer(cus_id),
name_ varchar(20) not null,
price long not null,
quantity int not null,
desc_ varchar(100)
);
```

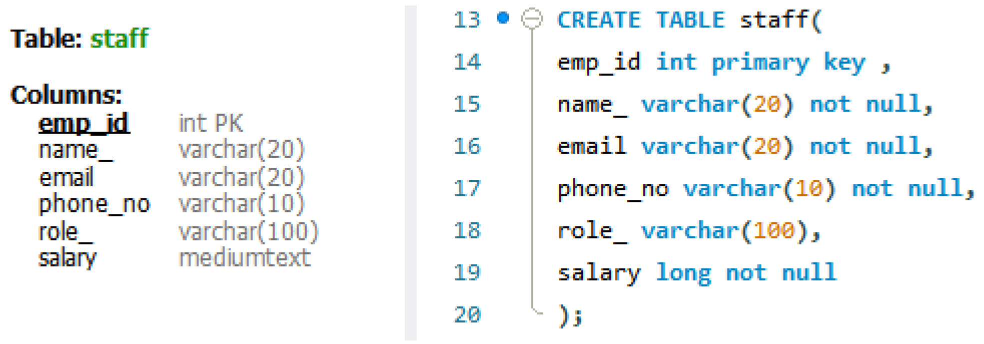


Table: staff

Columns:

emp_id	int PK
name_	varchar(20)
email	varchar(20)
phone_no	varchar(10)
role_	varchar(100)
salary	mediumtext

```sql
CREATE TABLE staff(
emp_id int primary key ,
name_ varchar(20) not null,
email varchar(20) not null,
phone_no varchar(10) not null,
role_ varchar(100),
salary long not null
);
```

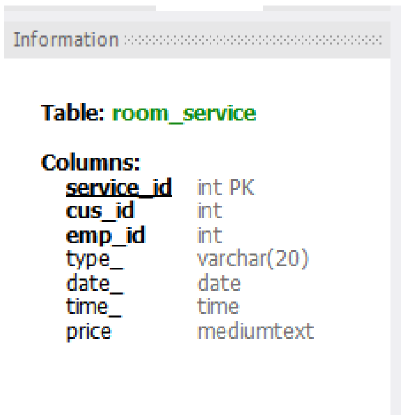
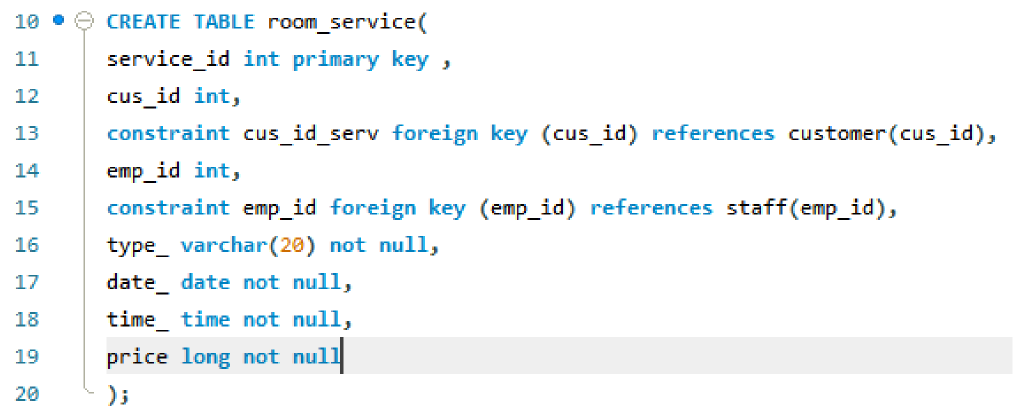

Information

Table: room_service

Columns:

service_id	int PK
cus_id	int
emp_id	int
type_	varchar(20)
date_	date
time_	time
price	mediumtext


```
CREATE TABLE room_service(
service_id int primary key ,
cus_id int,
constraint cus_id_serv foreign key (cus_id) references customer(cus_id),
emp_id int,
constraint emp_id foreign key (emp_id) references staff(emp_id),
type_ varchar(20) not null,
date_ date not null,
time_ time not null,
price long not null
);
```

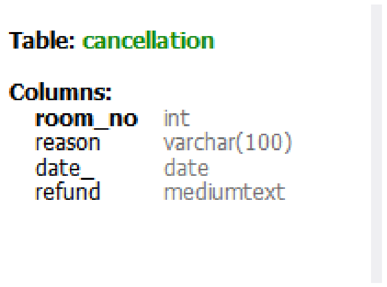
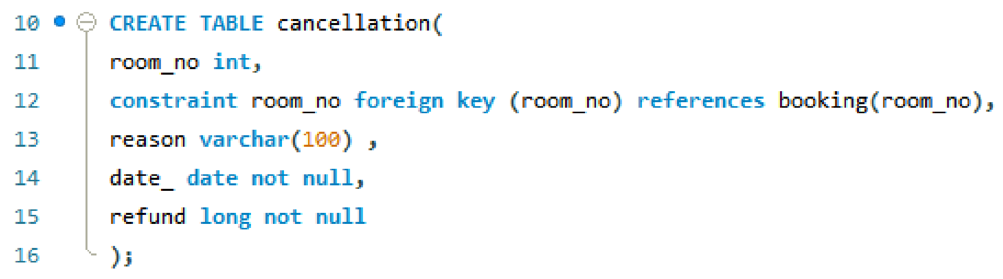


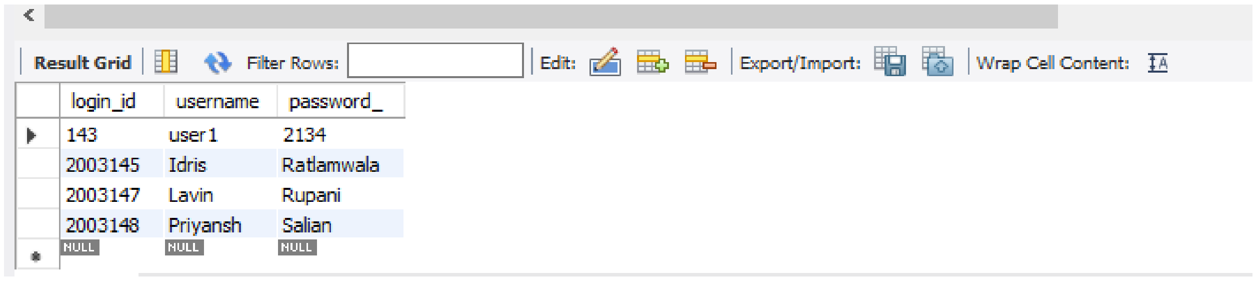
Table: cancellation

Columns:

room_no	int
reason	varchar(100)
date_	date
refund	mediumtext

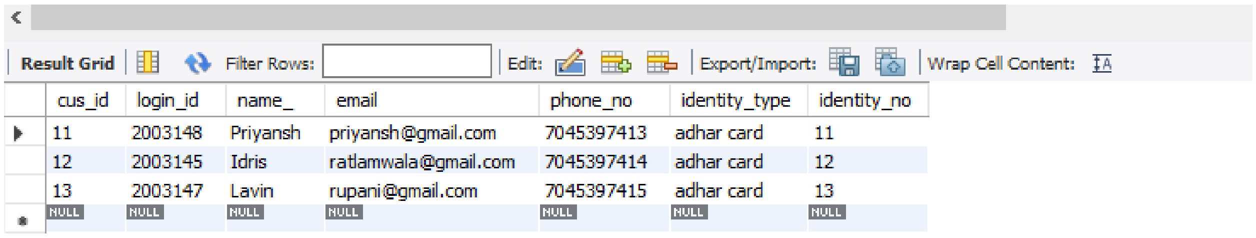

```
CREATE TABLE cancellation(
room_no int,
constraint room_no foreign key (room_no) references booking(room_no),
reason varchar(100) ,
date_ date not null,
refund long not null
);
```

Expt 4 :



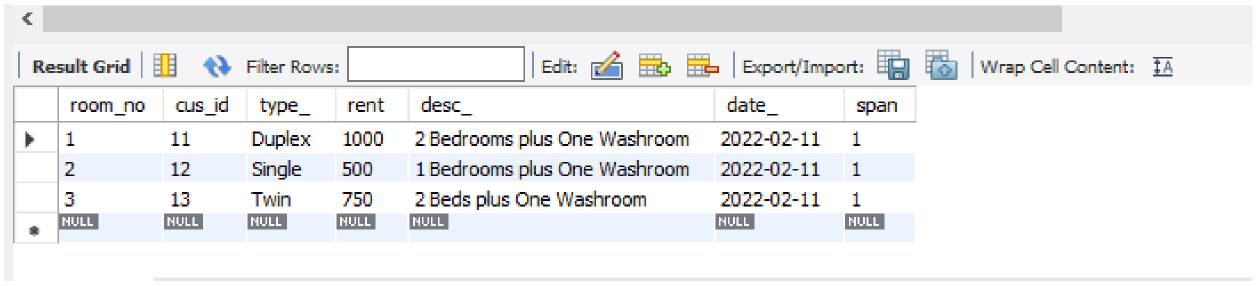
Result Grid | Filter Rows: | Edit: | Export/Import: | Wrap Cell Content:

login_id	username	password_
143	user1	2134
2003145	Idris	Ratlamwala
2003147	Lavin	Rupani
2003148	Priyansh	Salian
NULL	NULL	NULL



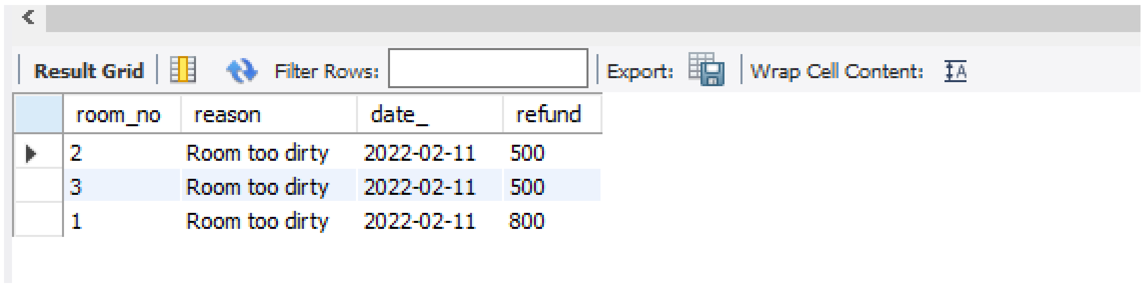
Result Grid | Filter Rows: | Edit: | Export/Import: | Wrap Cell Content:

cus_id	login_id	name_	email	phone_no	identity_type	identity_no
11	2003148	Priyansh	priyansh@gmail.com	7045397413	adhar card	11
12	2003145	Idris	ratlamwala@gmail.com	7045397414	adhar card	12
13	2003147	Lavin	rupani@gmail.com	7045397415	adhar card	13
NULL	NULL	NULL	NULL	NULL	NULL	NULL



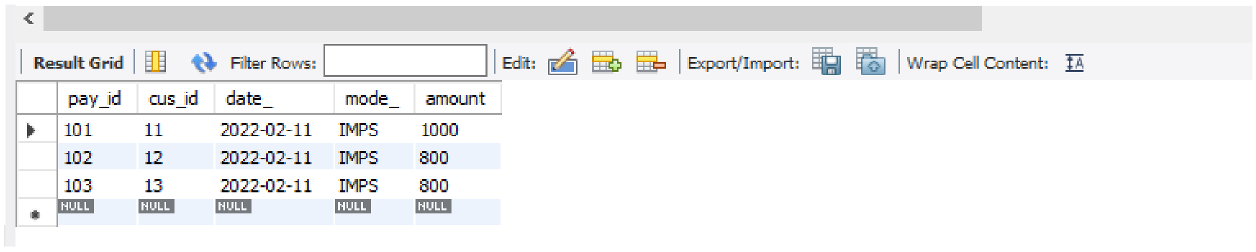
Result Grid | Filter Rows: | Edit: | Export/Import: | Wrap Cell Content:

room_no	cus_id	type_	rent	desc_	date_	span
1	11	Duplex	1000	2 Bedrooms plus One Washroom	2022-02-11	1
2	12	Single	500	1 Bedrooms plus One Washroom	2022-02-11	1
3	13	Twin	750	2 Beds plus One Washroom	2022-02-11	1
NULL	NULL	NULL	NULL	NULL	NULL	NULL

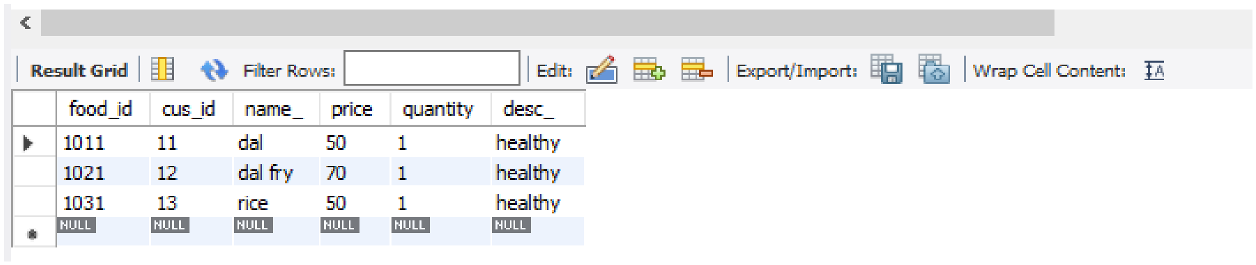


Result Grid | Filter Rows: | Export: | Wrap Cell Content:

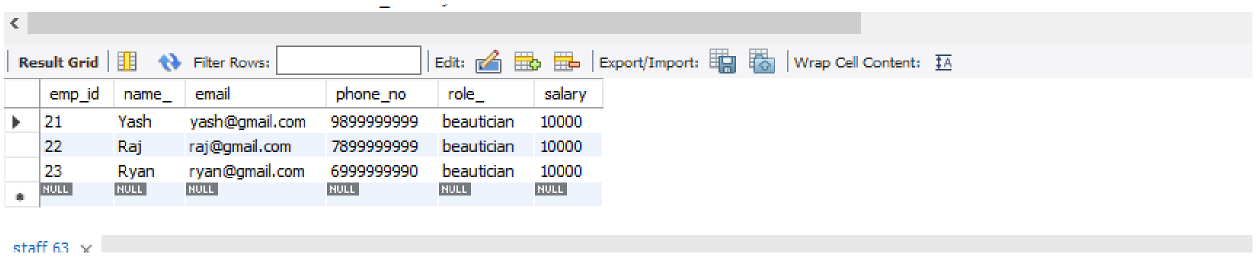
room_no	reason	date_	refund
2	Room too dirty	2022-02-11	500
3	Room too dirty	2022-02-11	500
1	Room too dirty	2022-02-11	800



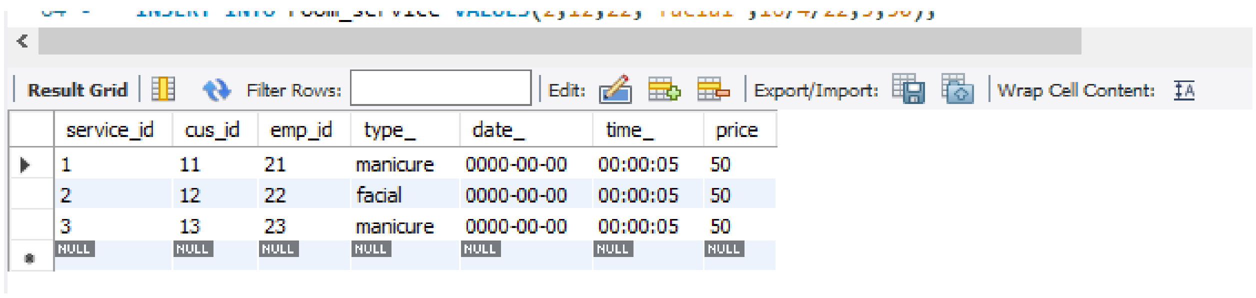
pay_id	cus_id	date_	mode_	amount
101	11	2022-02-11	IMPS	1000
102	12	2022-02-11	IMPS	800
103	13	2022-02-11	IMPS	800
NULL	NULL	NULL	NULL	NULL



food_id	cus_id	name_	price	quantity	desc_
1011	11	dal	50	1	healthy
1021	12	dal fry	70	1	healthy
1031	13	rice	50	1	healthy
NULL	NULL	NULL	NULL	NULL	NULL



emp_id	name_	email	phone_no	role_	salary
21	Yash	yash@gmail.com	9899999999	beautician	10000
22	Raj	raj@gmail.com	7899999999	beautician	10000
23	Ryan	ryan@gmail.com	6999999990	beautician	10000
NULL	NULL	NULL	NULL	NULL	NULL

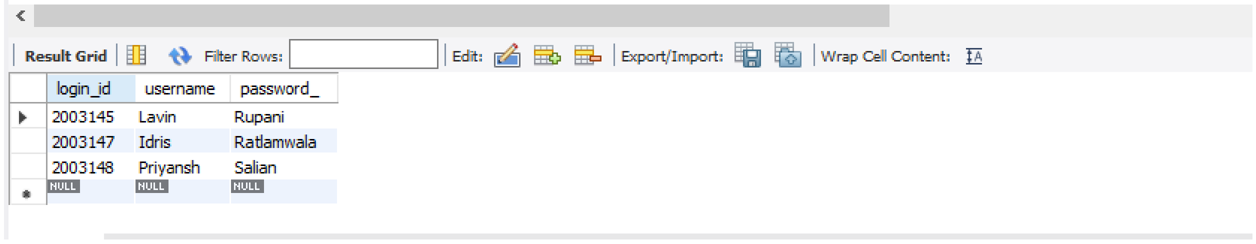


```
64 •    INSERT INTO room_service VALUES(2,12,22,'facial',10/4/22,5,50);
```

service_id	cus_id	emp_id	type_	date_	time_	price
1	11	21	manicure	0000-00-00	00:00:05	50
2	12	22	facial	0000-00-00	00:00:05	50
3	13	23	manicure	0000-00-00	00:00:05	50
NULL	NULL	NULL	NULL	NULL	NULL	NULL

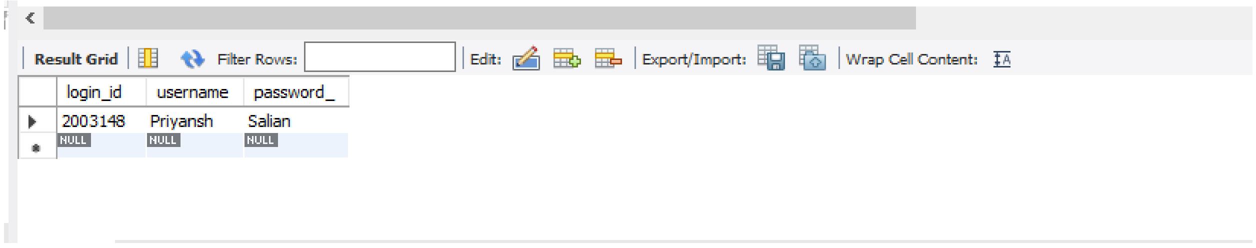
Expt 5 :

1) Print full table -select



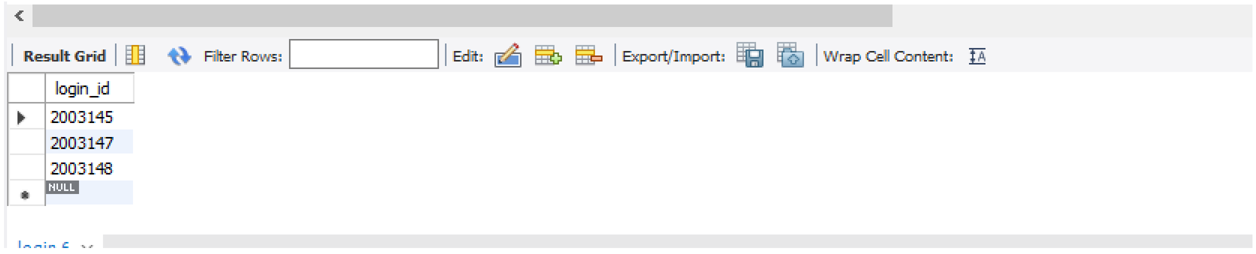
login_id	username	password_
2003145	Lavin	Rupani
2003147	Idris	Ratlamwala
2003148	Priyansh	Salian
NULL	NULL	NULL

2) Print only few attributes



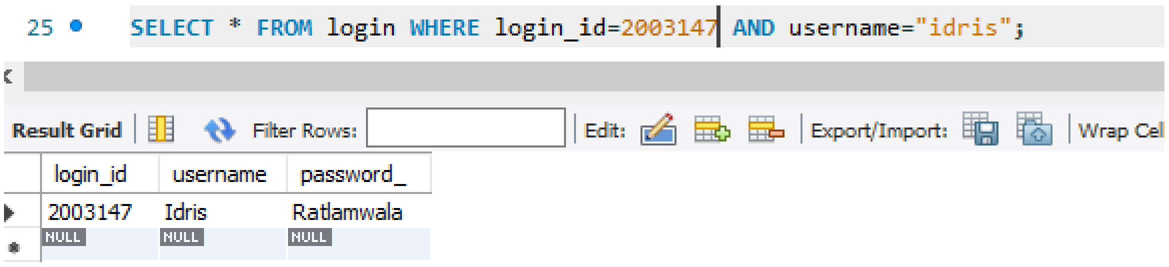
login_id	username	password_
2003148	Priyansh	Salian
NULL	NULL	NULL

3) Select conditional tuples



login_id
2003145
2003147
2003148
NULL

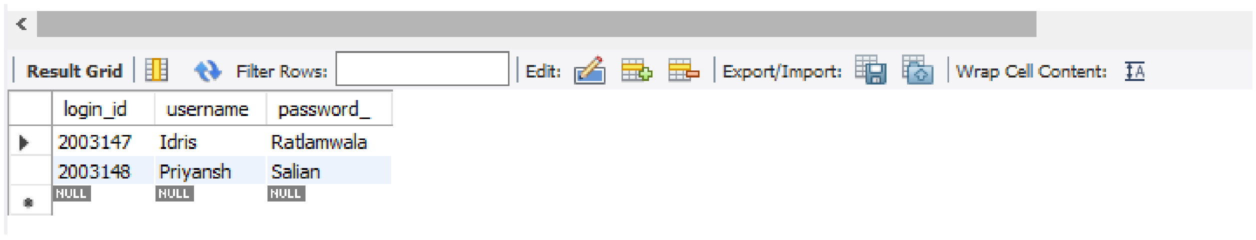
4) Where clause with AND condition



```
SELECT * FROM login WHERE login_id=2003147 AND username="idris";
```

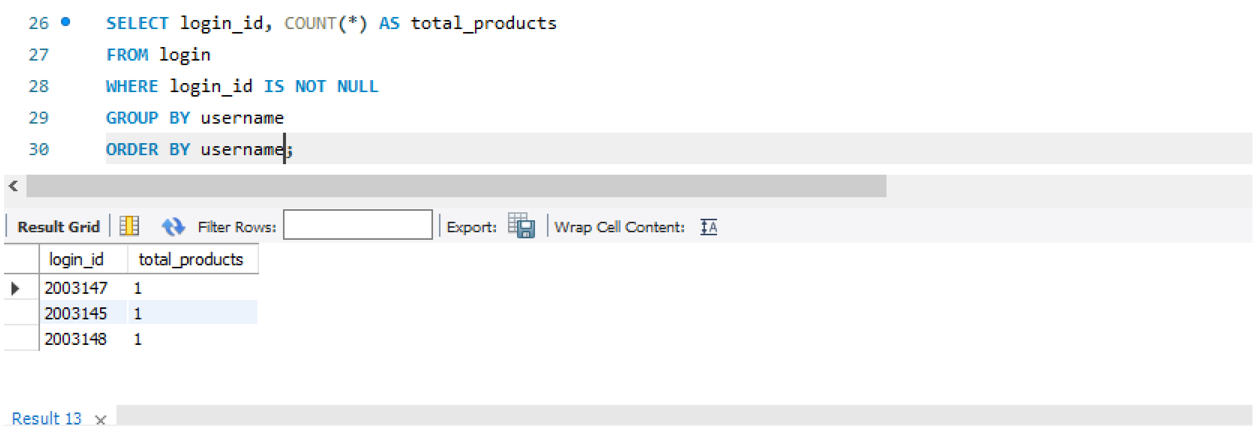
login_id	username	password_
2003147	Idris	Ratlamwala
NULL	NULL	NULL

5) Where clause with OR condition



	login_id	username	password_
▶	2003147	Idris	Ratlamwala
	2003148	Priyansh	Salian
✱	NULL	NULL	NULL

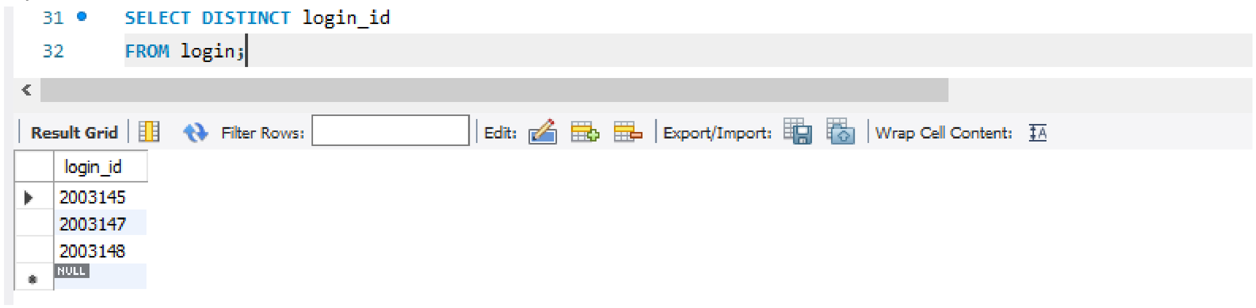
6) Order by



```
SELECT login_id, COUNT(*) AS total_products
FROM login
WHERE login_id IS NOT NULL
GROUP BY username
ORDER BY username;
```

	login_id	total_products
▶	2003147	1
	2003145	1
	2003148	1

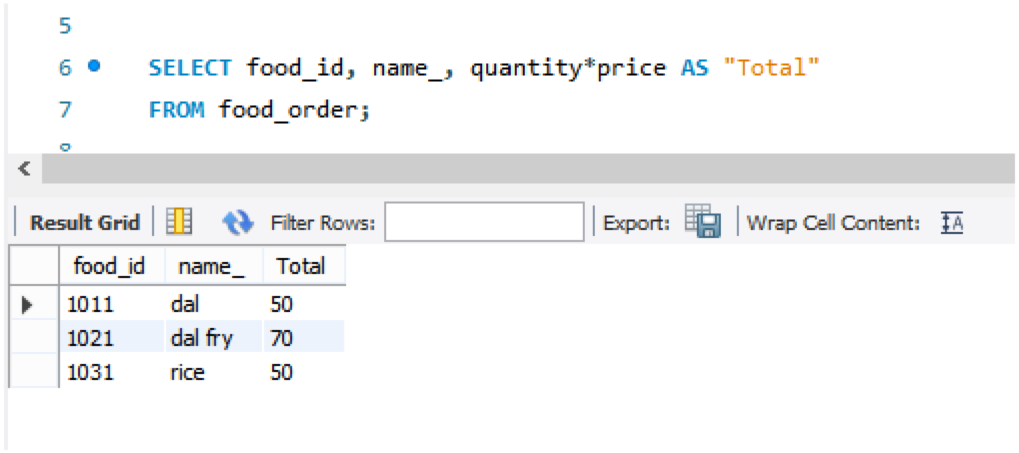
7) Distinct



```
SELECT DISTINCT login_id
FROM login;
```

	login_id
▶	2003145
	2003147
	2003148
✱	NULL

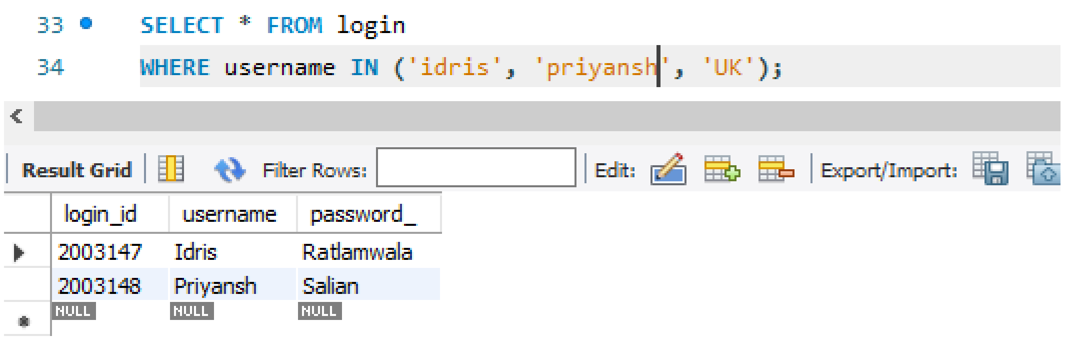
8) Calculations in select



```
SELECT food_id, name_, quantity*price AS "Total"
FROM food_order;
```

	food_id	name_	Total
▶	1011	dal	50
	1021	dal fry	70
	1031	rice	50

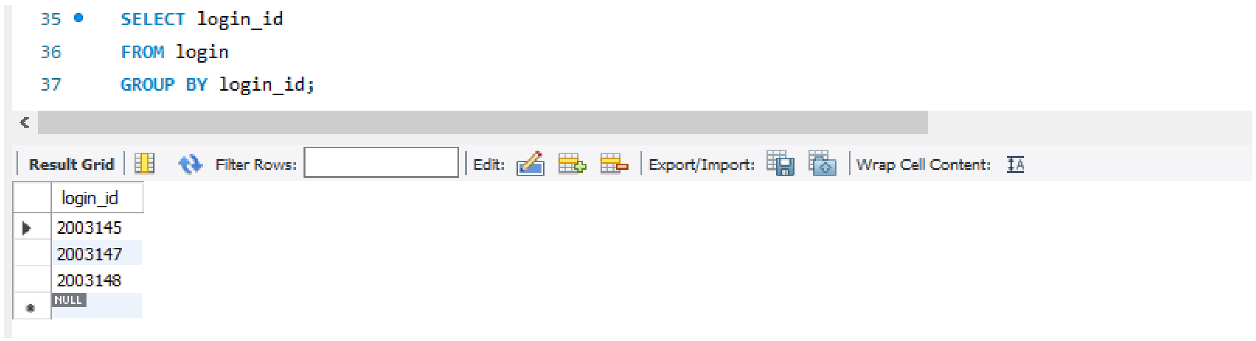
9) Select with in clause



```sql
SELECT * FROM login
WHERE username IN ('idris', 'priyansh', 'UK');
```

login_id	username	password_
2003147	Idris	Ratlamwala
2003148	Priyansh	Salian
NULL	NULL	NULL

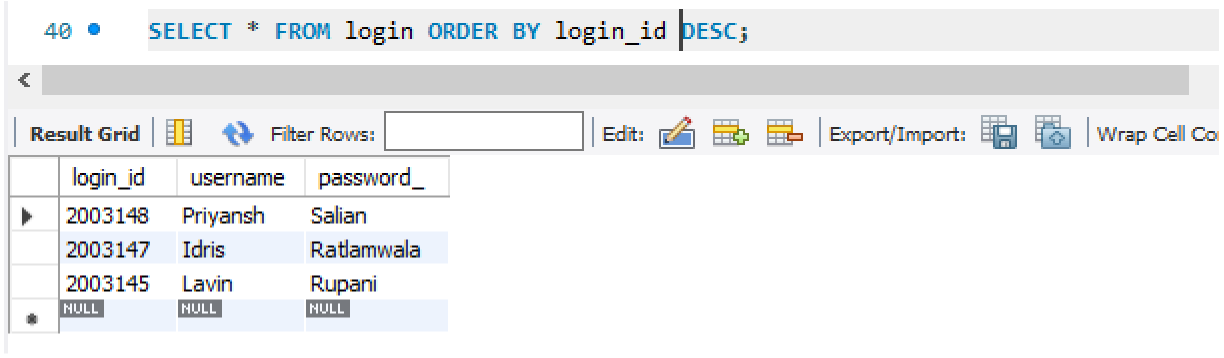
10) Sorting- asc



```sql
SELECT login_id
FROM login
GROUP BY login_id;
```

login_id
2003145
2003147
2003148
NULL

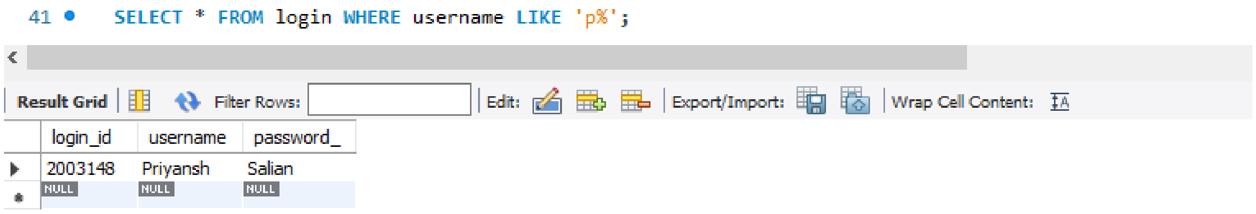
11) Sorting- desc



```sql
SELECT * FROM login ORDER BY login_id DESC;
```

login_id	username	password_
2003148	Priyansh	Salian
2003147	Idris	Ratlamwala
2003145	Lavin	Rupani
NULL	NULL	NULL

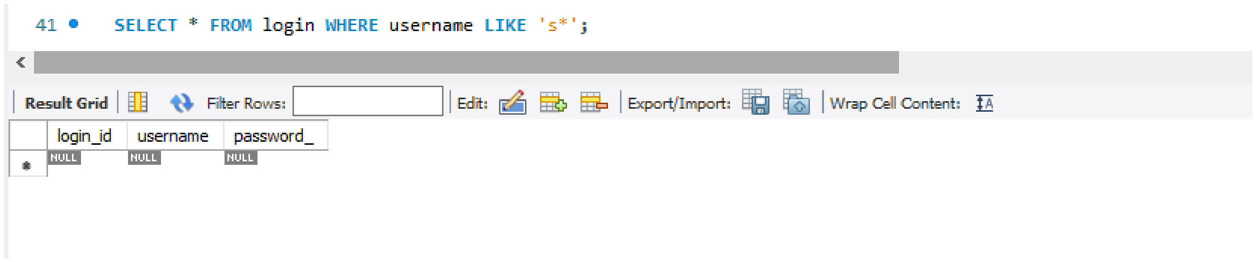
12) Sting Matching- %



```sql
SELECT * FROM login WHERE username LIKE 'p%';
```

login_id	username	password_
2003148	Priyansh	Salian
NULL	NULL	NULL

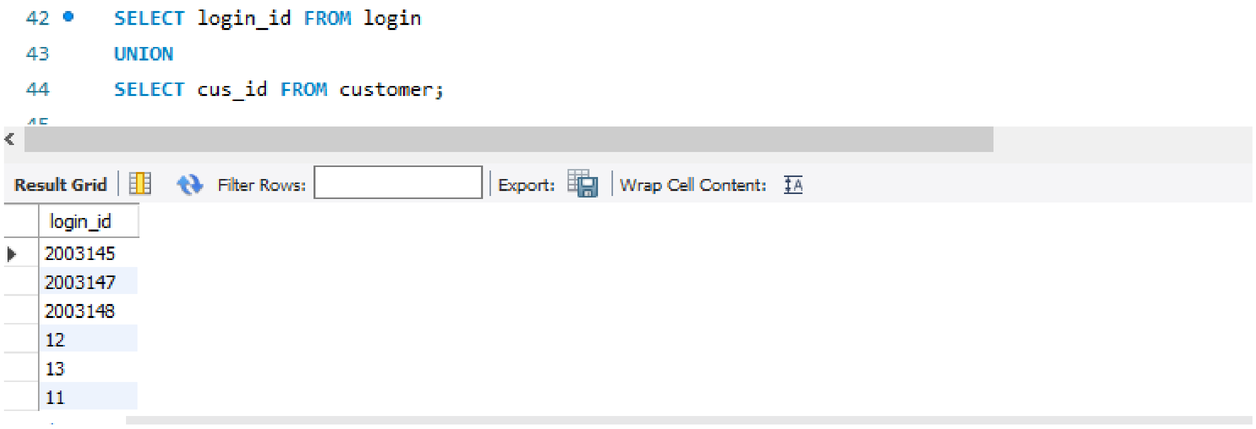
13) Sting Matching- *



```sql
SELECT * FROM login WHERE username LIKE 's*';
```

login_id	username	password_
NULL	NULL	NULL

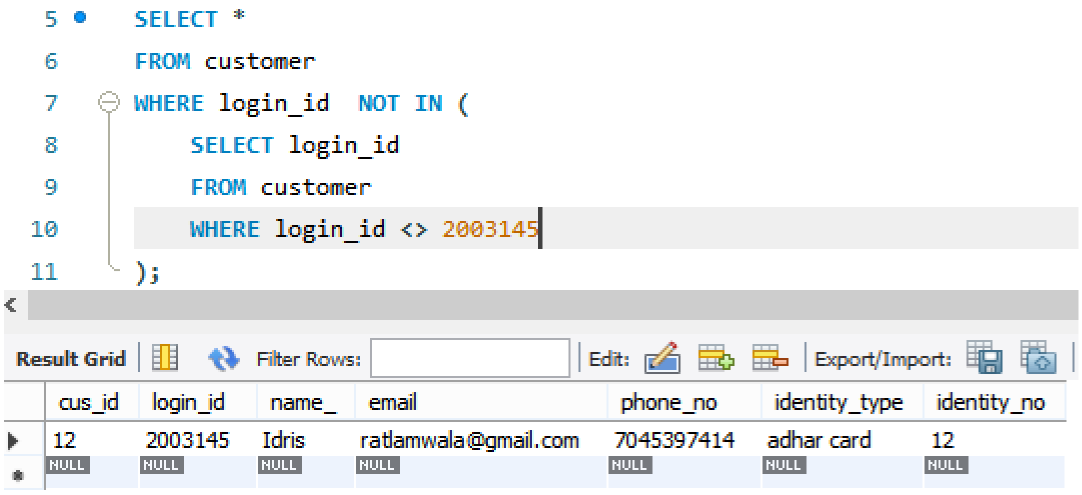
14) Union



```sql
SELECT login_id FROM login
UNION
SELECT cus_id FROM customer;
```

login_id
2003145
2003147
2003148
12
13
11

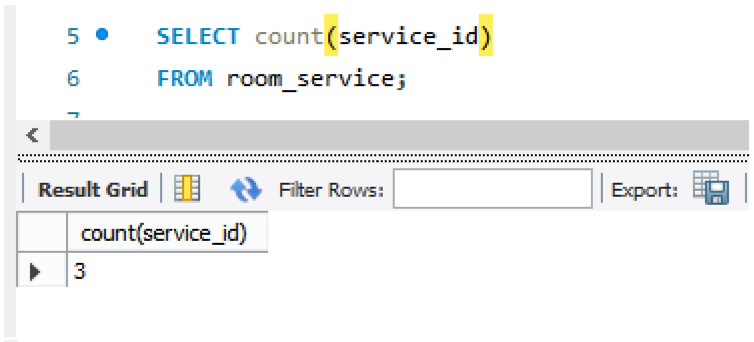
16) Difference



```sql
SELECT *
FROM customer
WHERE login_id  NOT IN (
    SELECT login_id
    FROM customer
    WHERE login_id <> 2003145
);
```

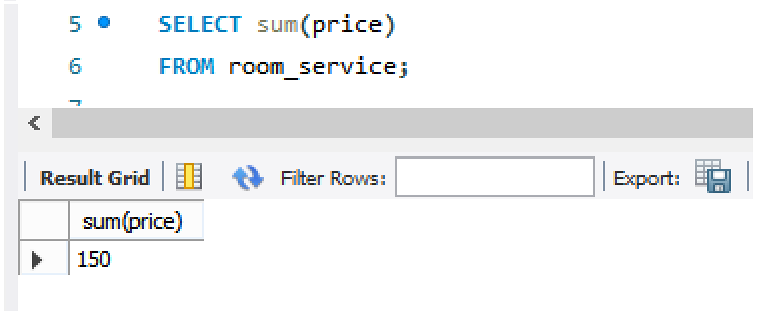
cus_id	login_id	name_	email	phone_no	identity_type	identity_no
12	2003145	Idris	ratlamwala@gmail.com	7045397414	adhar card	12
NULL	NULL	NULL	NULL	NULL	NULL	NULL

17) Aggregate Function (Any two)



```sql
SELECT count(service_id)
FROM room_service;
```

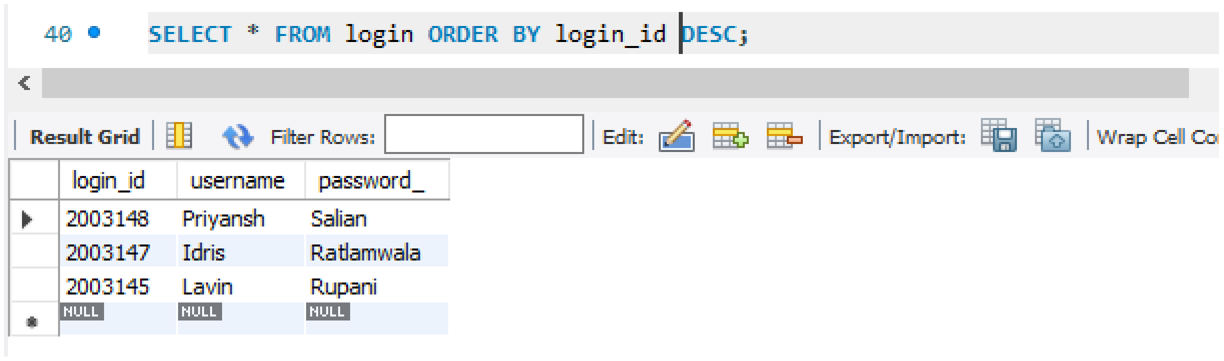
```
count(service_id)
3
```



```sql
SELECT sum(price)
FROM room_service;
```

```
sum(price)
150
```

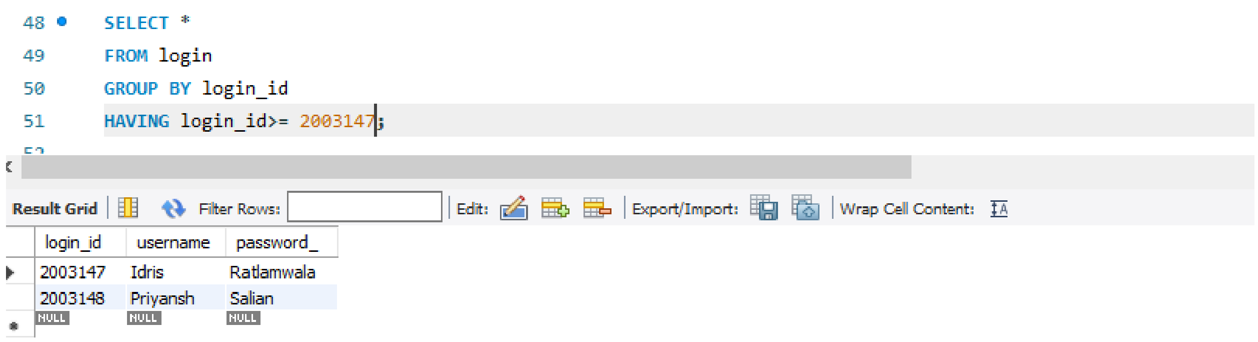
18) Group by



```sql
SELECT * FROM login ORDER BY login_id DESC;
```

```
login_id  username  password_
2003148   Priyansh  Salian
2003147   Idris     Ratlamwala
2003145   Lavin     Rupani
NULL      NULL      NULL
```

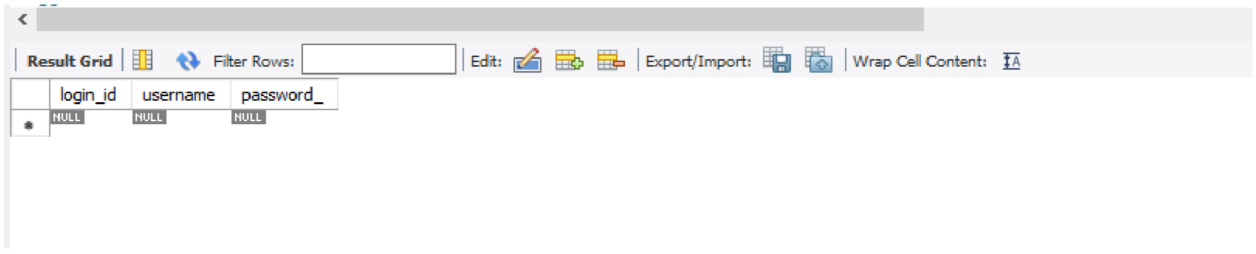
19) Group by having



```sql
SELECT *
FROM login
GROUP BY login_id
HAVING login_id>= 2003147;
```

```
login_id  username  password_
2003147   Idris     Ratlamwala
2003148   Priyansh  Salian
NULL      NULL      NULL
```

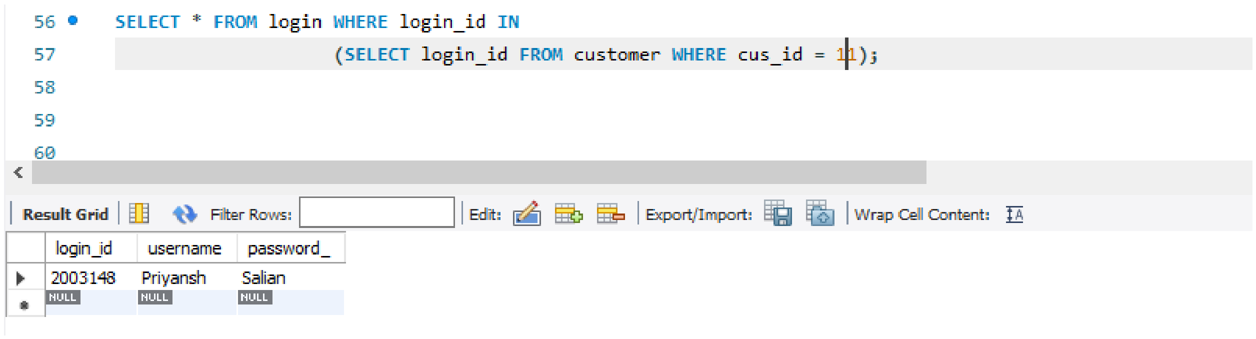
20) Query with Null value



	login_id	username	password_
*	NULL	NULL	NULL

Expt 6 :

1) in

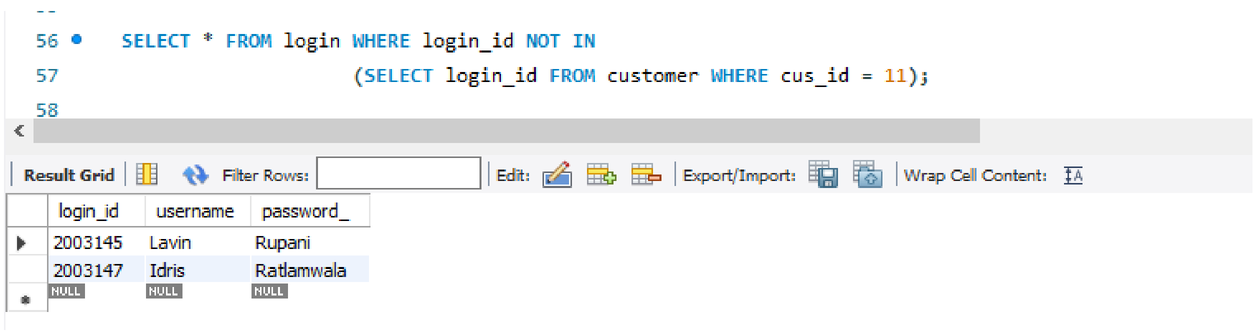


```
SELECT * FROM login WHERE login_id IN
                (SELECT login_id FROM customer WHERE cus_id = 11);
```

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

login_id	username	password_
2003148	Priyansh	Salian
NULL	NULL	NULL

2) not in

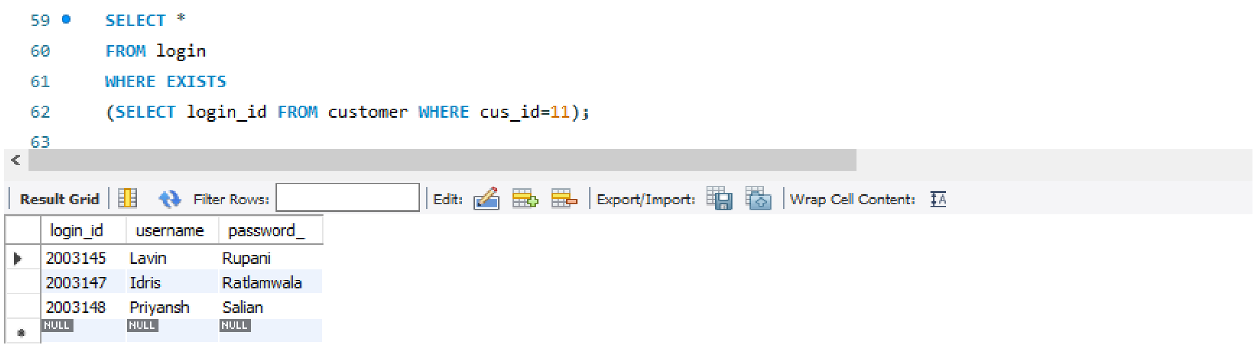


```
--
SELECT * FROM login WHERE login_id NOT IN
                (SELECT login_id FROM customer WHERE cus_id = 11);
```

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

login_id	username	password_
2003145	Lavin	Rupani
2003147	Idris	Ratlamwala
NULL	NULL	NULL

3) exists

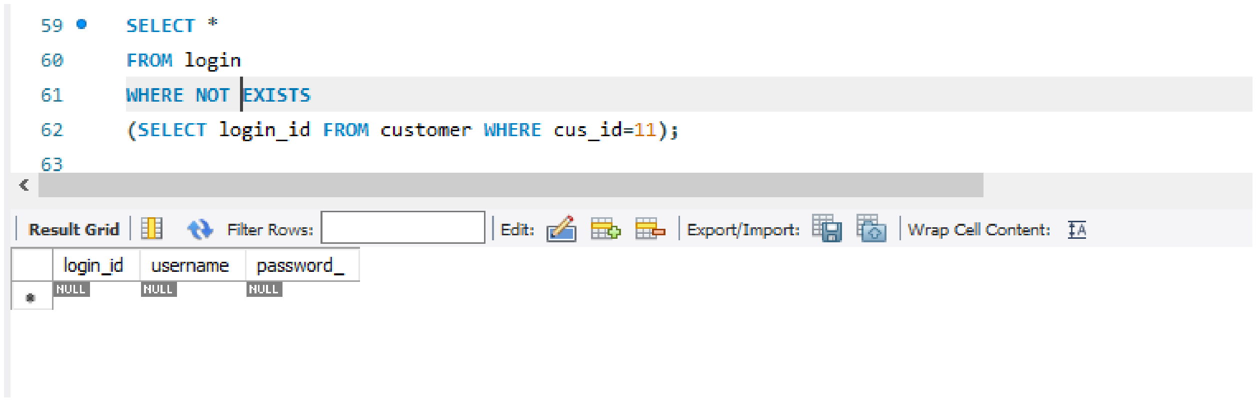


```
SELECT *
FROM login
WHERE EXISTS
(SELECT login_id FROM customer WHERE cus_id=11);
```

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

login_id	username	password_
2003145	Lavin	Rupani
2003147	Idris	Ratlamwala
2003148	Priyansh	Salian
NULL	NULL	NULL

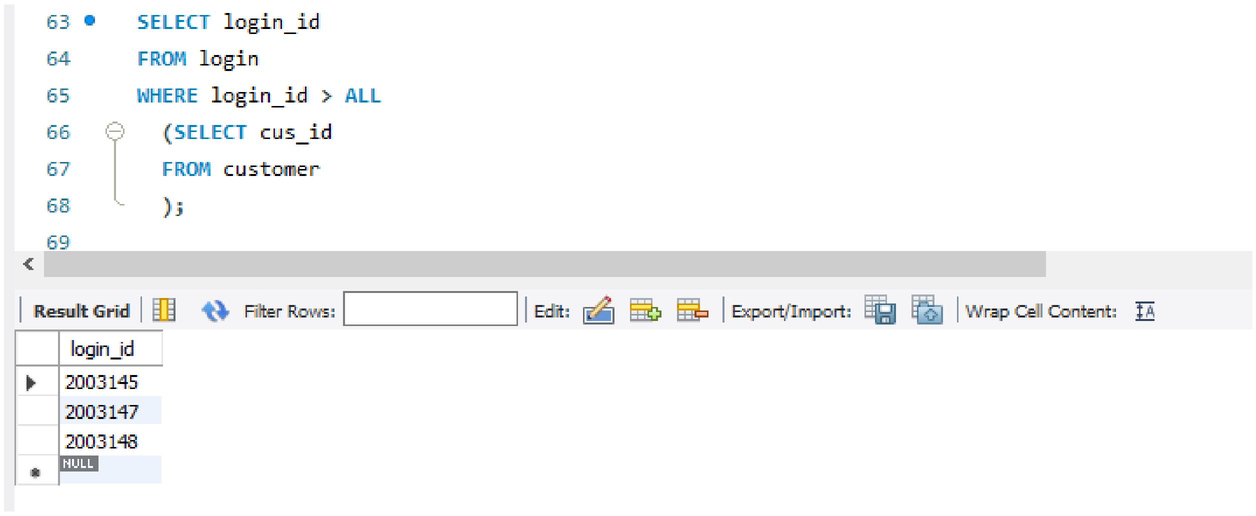
4) not exists



```
SELECT *
FROM login
WHERE NOT EXISTS
(SELECT login_id FROM customer WHERE cus_id=11);
```

	login_id	username	password_
*	NULL	NULL	NULL

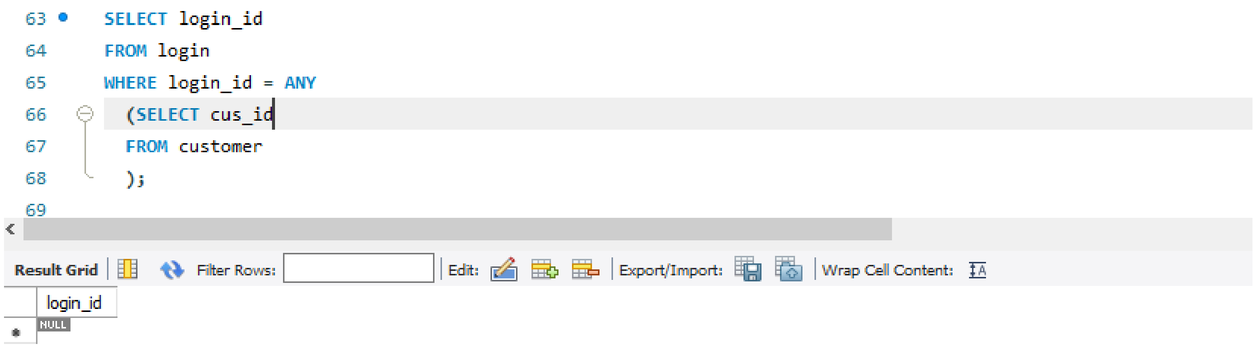
5) All



```
SELECT login_id
FROM login
WHERE login_id > ALL
  (SELECT cus_id
  FROM customer
  );
```

	login_id
▶	2003145
	2003147
	2003148
*	NULL

6) Any/ some

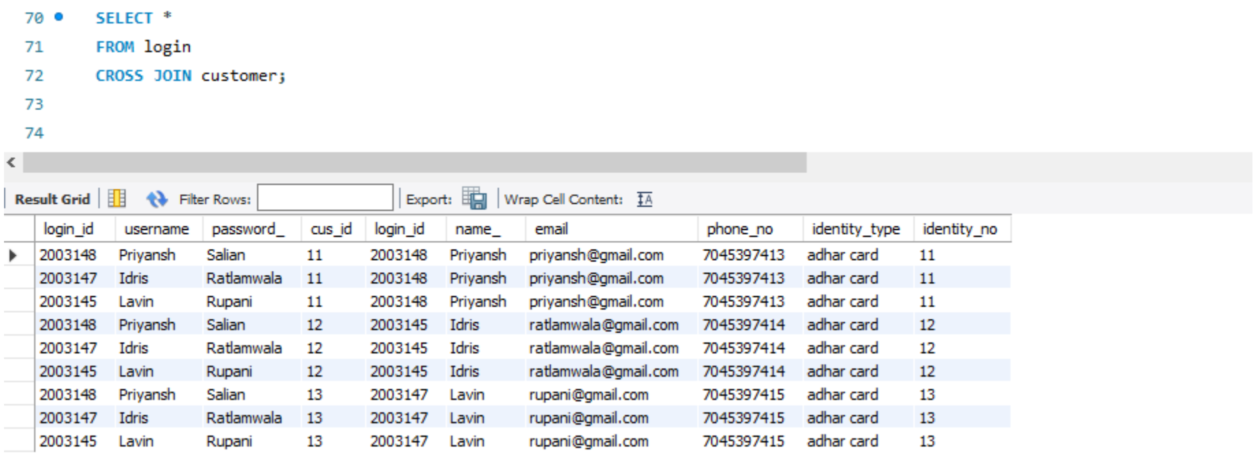


```
SELECT login_id
FROM login
WHERE login_id = ANY
  (SELECT cus_id
  FROM customer
  );
```

	login_id
*	NULL

Expt 7 :

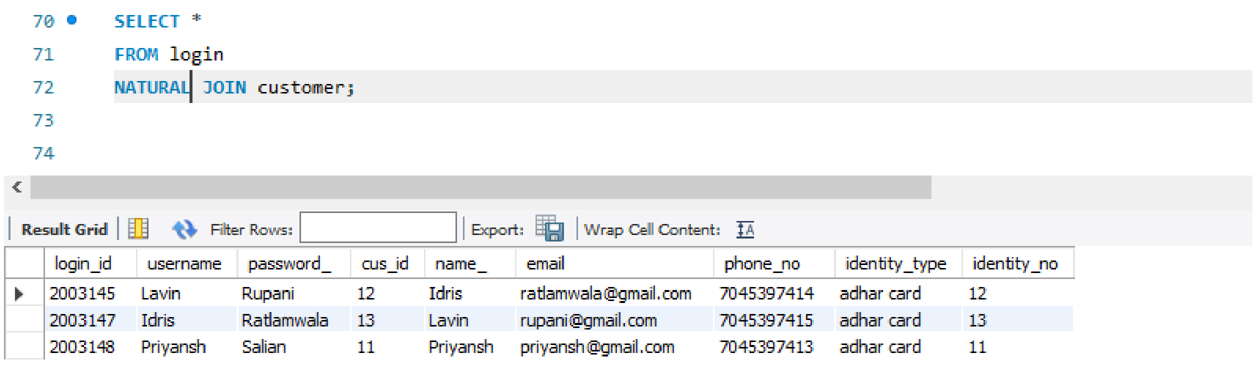
1) Cross Join



```sql
SELECT *
FROM login
CROSS JOIN customer;
```

login_id	username	password_	cus_id	login_id	name_	email	phone_no	identity_type	identity_no
2003148	Priyansh	Salian	11	2003148	Priyansh	priyansh@gmail.com	7045397413	adhar card	11
2003147	Idris	Ratlamwala	11	2003148	Priyansh	priyansh@gmail.com	7045397413	adhar card	11
2003145	Lavin	Rupani	11	2003148	Priyansh	priyansh@gmail.com	7045397413	adhar card	11
2003148	Priyansh	Salian	12	2003145	Idris	ratlamwala@gmail.com	7045397414	adhar card	12
2003147	Idris	Ratlamwala	12	2003145	Idris	ratlamwala@gmail.com	7045397414	adhar card	12
2003145	Lavin	Rupani	12	2003145	Idris	ratlamwala@gmail.com	7045397414	adhar card	12
2003148	Priyansh	Salian	13	2003147	Lavin	rupani@gmail.com	7045397415	adhar card	13
2003147	Idris	Ratlamwala	13	2003147	Lavin	rupani@gmail.com	7045397415	adhar card	13
2003145	Lavin	Rupani	13	2003147	Lavin	rupani@gmail.com	7045397415	adhar card	13

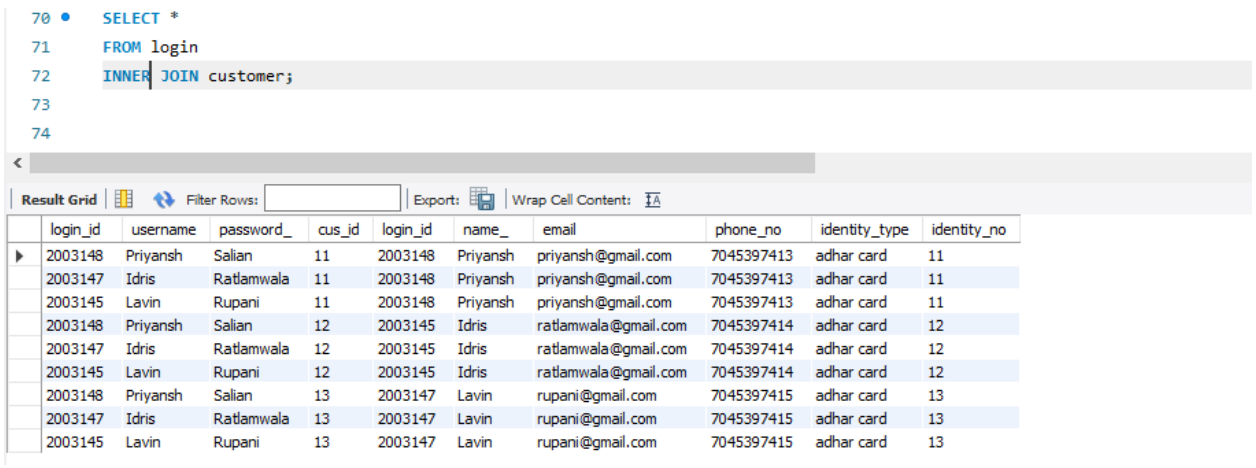
2) Natural Join



```sql
SELECT *
FROM login
NATURAL JOIN customer;
```

login_id	username	password_	cus_id	name_	email	phone_no	identity_type	identity_no
2003145	Lavin	Rupani	12	Idris	ratlamwala@gmail.com	7045397414	adhar card	12
2003147	Idris	Ratlamwala	13	Lavin	rupani@gmail.com	7045397415	adhar card	13
2003148	Priyansh	Salian	11	Priyansh	priyansh@gmail.com	7045397413	adhar card	11

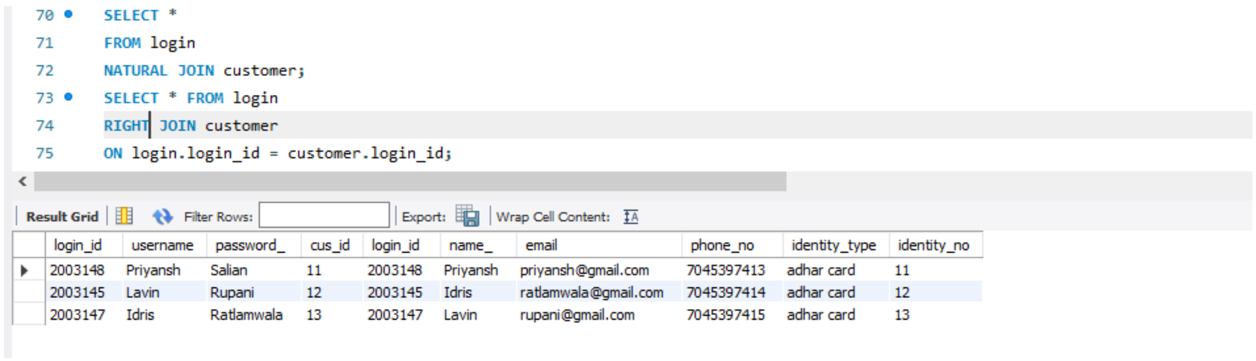
3) Inner /join



```sql
SELECT *
FROM login
INNER JOIN customer;
```

login_id	username	password_	cus_id	login_id	name_	email	phone_no	identity_type	identity_no
2003148	Priyansh	Salian	11	2003148	Priyansh	priyansh@gmail.com	7045397413	adhar card	11
2003147	Idris	Ratlamwala	11	2003148	Priyansh	priyansh@gmail.com	7045397413	adhar card	11
2003145	Lavin	Rupani	11	2003148	Priyansh	priyansh@gmail.com	7045397413	adhar card	11
2003148	Priyansh	Salian	12	2003145	Idris	ratlamwala@gmail.com	7045397414	adhar card	12
2003147	Idris	Ratlamwala	12	2003145	Idris	ratlamwala@gmail.com	7045397414	adhar card	12
2003145	Lavin	Rupani	12	2003145	Idris	ratlamwala@gmail.com	7045397414	adhar card	12
2003148	Priyansh	Salian	13	2003147	Lavin	rupani@gmail.com	7045397415	adhar card	13
2003147	Idris	Ratlamwala	13	2003147	Lavin	rupani@gmail.com	7045397415	adhar card	13
2003145	Lavin	Rupani	13	2003147	Lavin	rupani@gmail.com	7045397415	adhar card	13

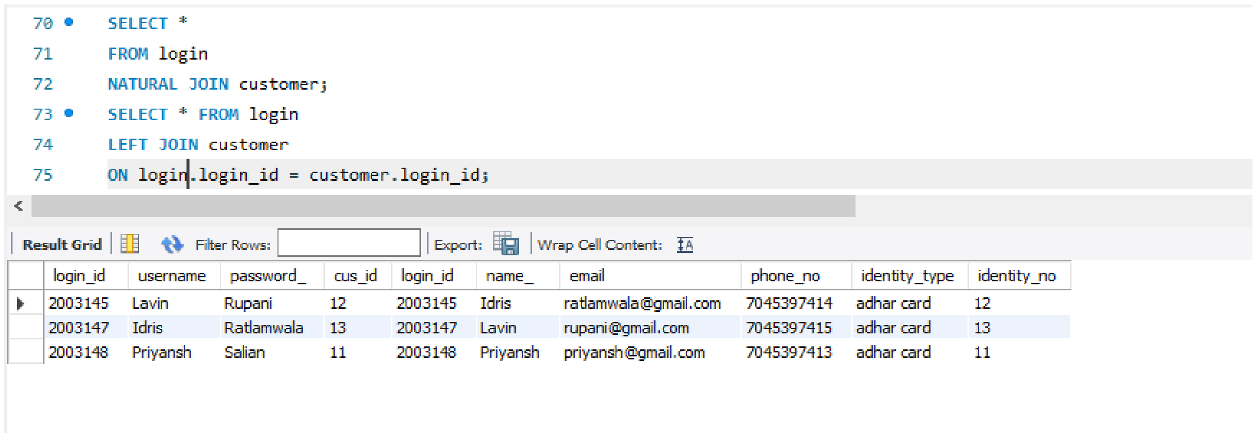
4) Right outer Join



```
SELECT *
FROM login
NATURAL JOIN customer;
SELECT * FROM login
RIGHT JOIN customer
ON login.login_id = customer.login_id;
```

login_id	username	password_	cus_id	login_id	name_	email	phone_no	identity_type	identity_no
2003148	Priyansh	Salian	11	2003148	Priyansh	priyansh@gmail.com	7045397413	adhar card	11
2003145	Lavin	Rupani	12	2003145	Idris	ratlamwala@gmail.com	7045397414	adhar card	12
2003147	Idris	Ratlamwala	13	2003147	Lavin	rupani@gmail.com	7045397415	adhar card	13

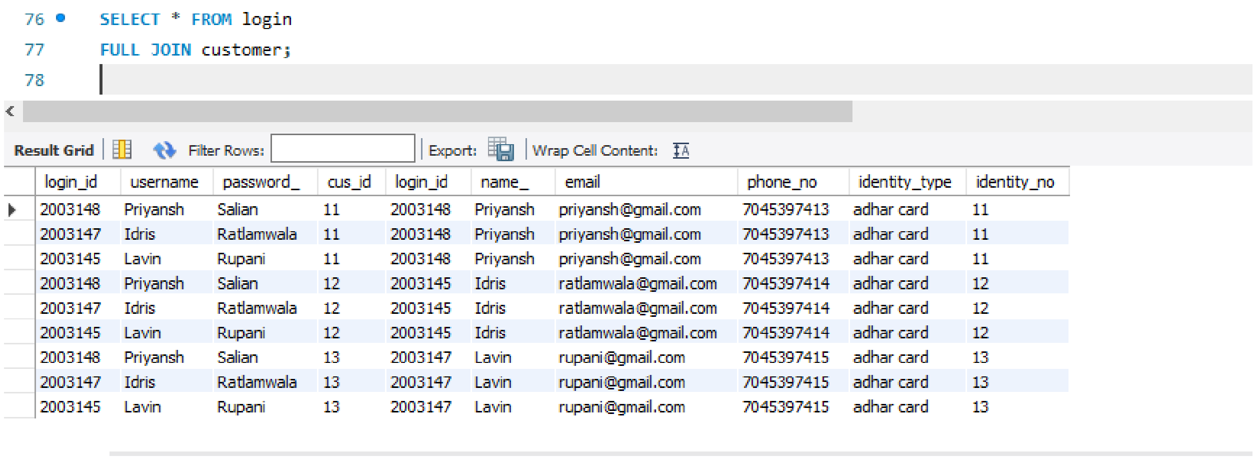
5) Left Outer Join



```
SELECT *
FROM login
NATURAL JOIN customer;
SELECT * FROM login
LEFT JOIN customer
ON login.login_id = customer.login_id;
```

login_id	username	password_	cus_id	login_id	name_	email	phone_no	identity_type	identity_no
2003145	Lavin	Rupani	12	2003145	Idris	ratlamwala@gmail.com	7045397414	adhar card	12
2003147	Idris	Ratlamwala	13	2003147	Lavin	rupani@gmail.com	7045397415	adhar card	13
2003148	Priyansh	Salian	11	2003148	Priyansh	priyansh@gmail.com	7045397413	adhar card	11

6) Full Outer Join



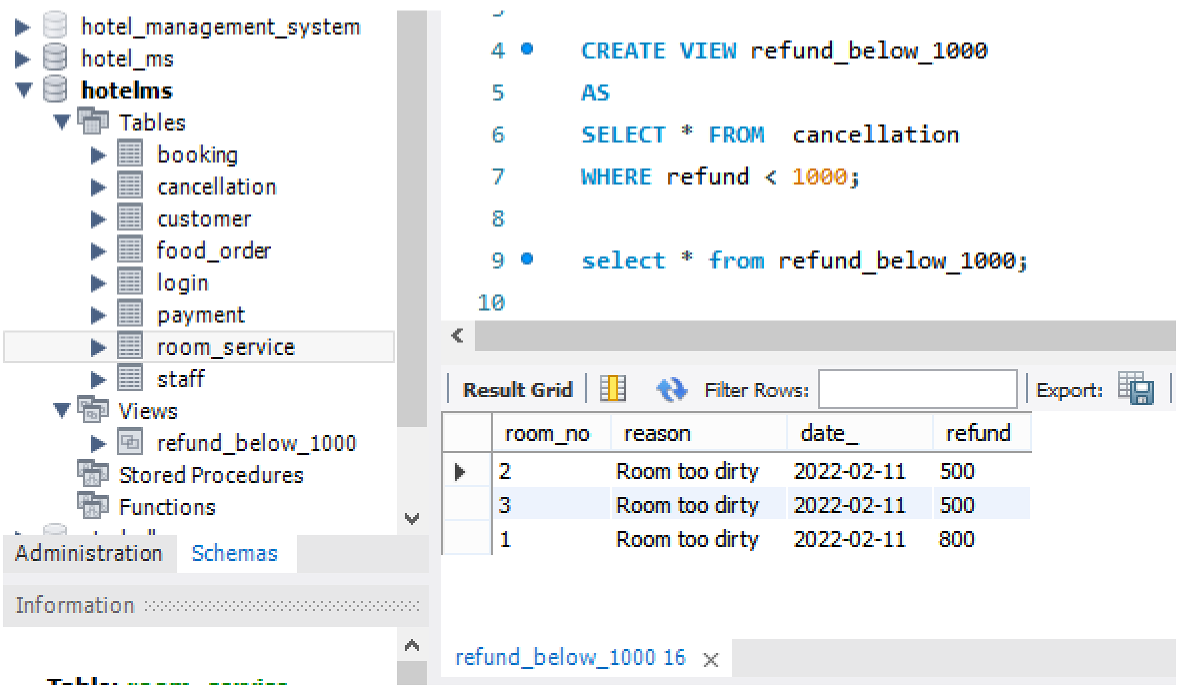
```
SELECT * FROM login
FULL JOIN customer;
```

login_id	username	password_	cus_id	login_id	name_	email	phone_no	identity_type	identity_no
2003148	Priyansh	Salian	11	2003148	Priyansh	priyansh@gmail.com	7045397413	adhar card	11
2003147	Idris	Ratlamwala	11	2003148	Priyansh	priyansh@gmail.com	7045397413	adhar card	11
2003145	Lavin	Rupani	11	2003148	Priyansh	priyansh@gmail.com	7045397413	adhar card	11
2003148	Priyansh	Salian	12	2003145	Idris	ratlamwala@gmail.com	7045397414	adhar card	12
2003147	Idris	Ratlamwala	12	2003145	Idris	ratlamwala@gmail.com	7045397414	adhar card	12
2003145	Lavin	Rupani	12	2003145	Idris	ratlamwala@gmail.com	7045397414	adhar card	12
2003148	Priyansh	Salian	13	2003147	Lavin	rupani@gmail.com	7045397415	adhar card	13
2003147	Idris	Ratlamwala	13	2003147	Lavin	rupani@gmail.com	7045397415	adhar card	13
2003145	Lavin	Rupani	13	2003147	Lavin	rupani@gmail.com	7045397415	adhar card	13

Expt 8 :

A] Implementation of views :

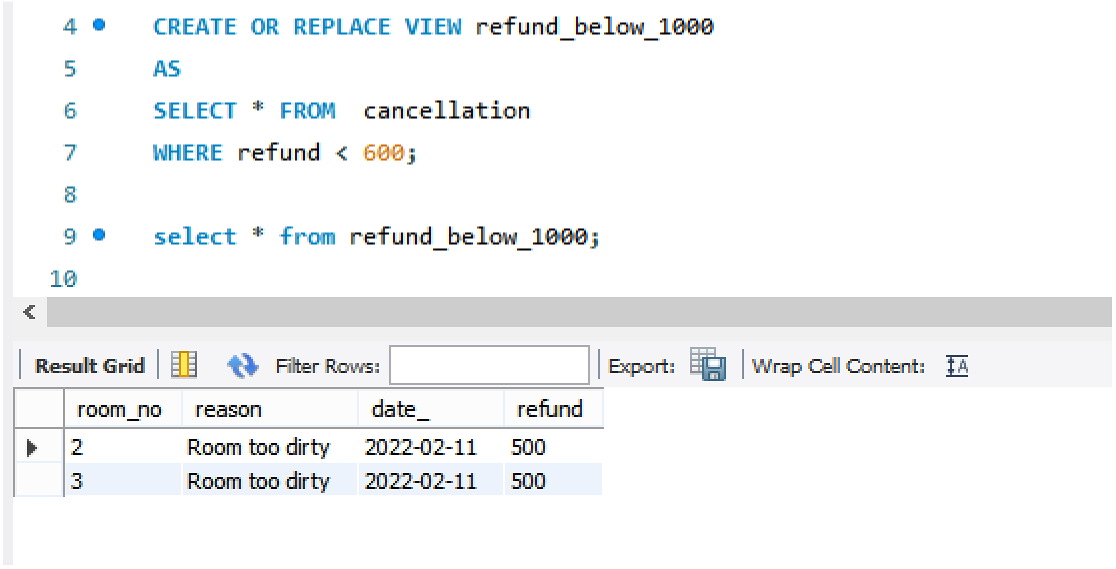
1. Creating a view



```
CREATE VIEW refund_below_1000
AS
SELECT * FROM  cancellation
WHERE refund < 1000;

select * from refund_below_1000;
```

```
room_no  reason          date_       refund
2        Room too dirty  2022-02-11  500
3        Room too dirty  2022-02-11  500
1        Room too dirty  2022-02-11  800
```

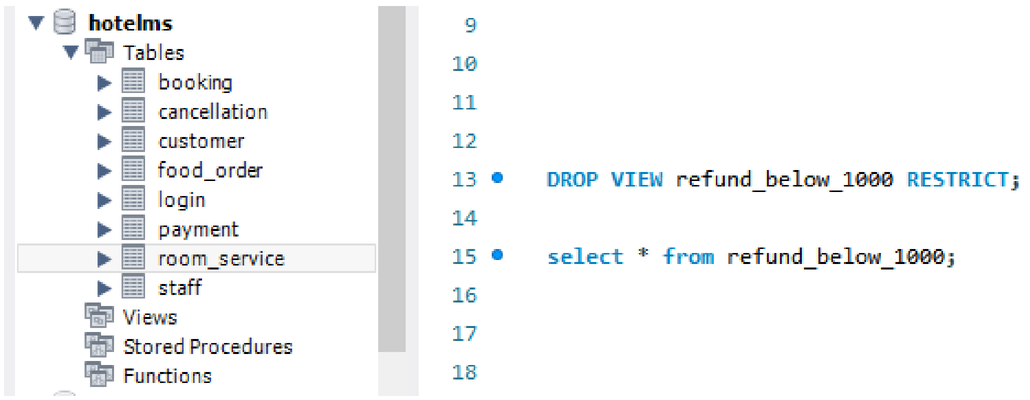
2. Modifing a view



```
CREATE OR REPLACE VIEW refund_below_1000
AS
SELECT * FROM  cancellation
WHERE refund < 600;

select * from refund_below_1000;
```

```
room_no  reason          date_       refund
2        Room too dirty  2022-02-11  500
3        Room too dirty  2022-02-11  500
```

3. Dropping a view

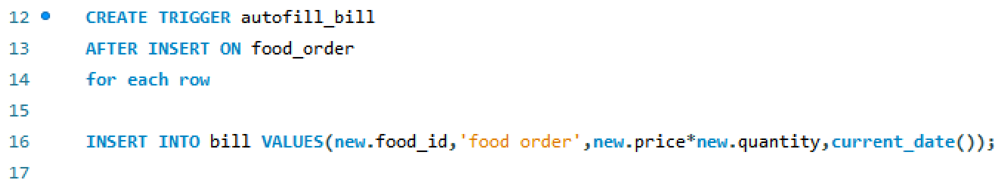


hotelms
Tables
booking
cancellation
customer
food_order
login
payment
room_service
staff
Views
Stored Procedures
Functions

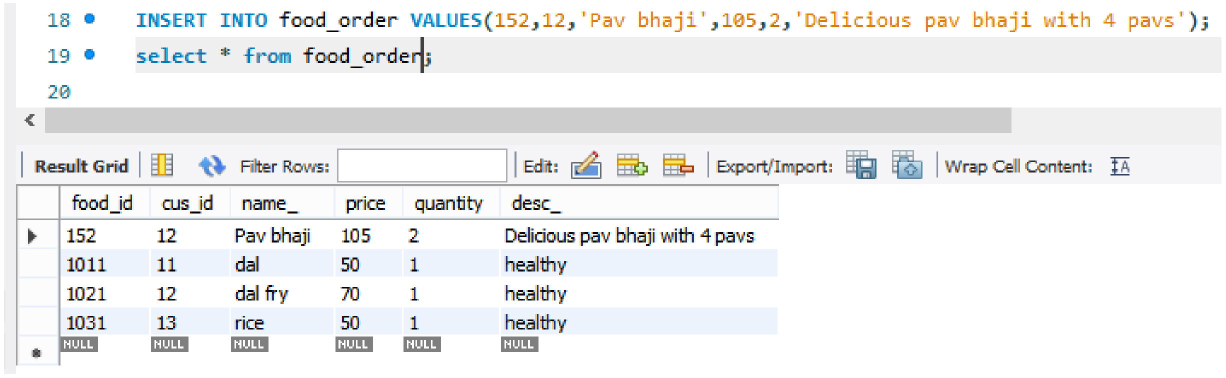
```
DROP VIEW refund_below_1000 RESTRICT;

select * from refund_below_1000;
```

B] Implementation of triggers :

1. Create trigger

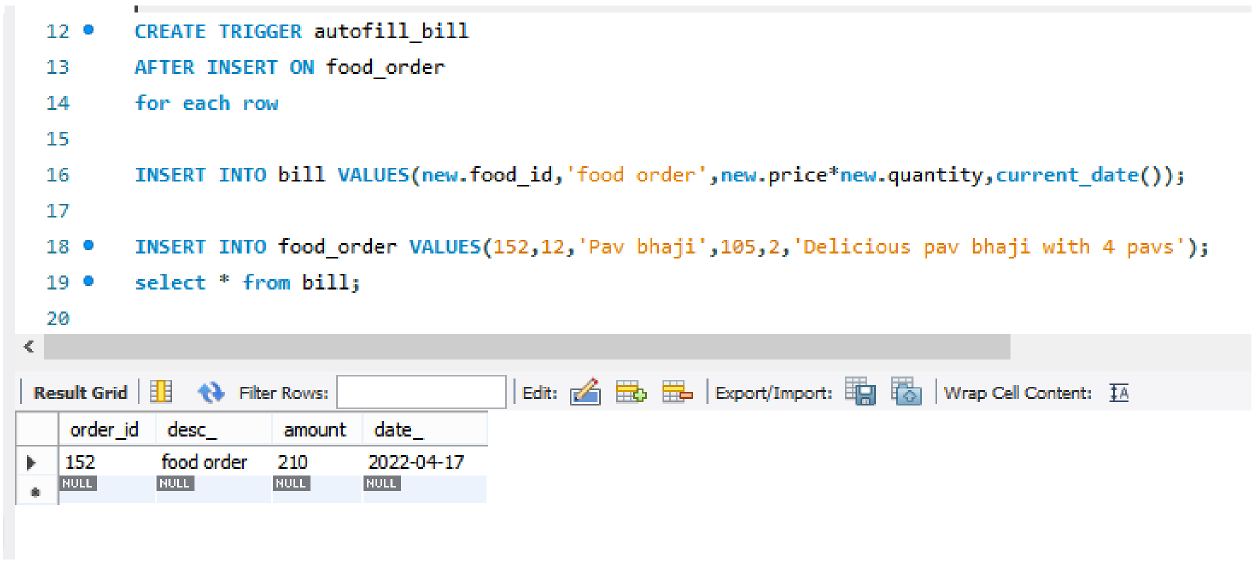


```
CREATE TRIGGER autofill_bill
AFTER INSERT ON food_order
for each row

INSERT INTO bill VALUES(new.food_id,'food order',new.price*new.quantity,current_date());
```



```
INSERT INTO food_order VALUES(152,12,'Pav bhaji',105,2,'Delicious pav bhaji with 4 pavs');
select * from food_order;
```

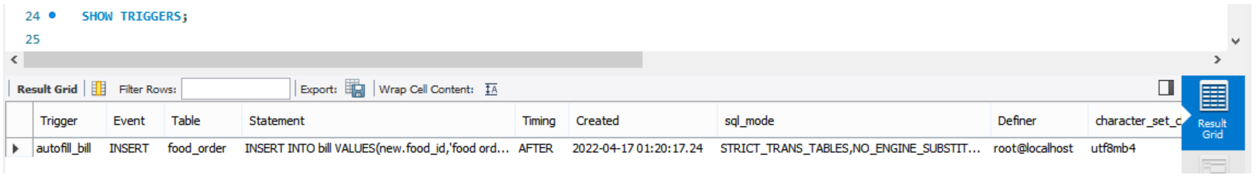
food_id	cus_id	name_	price	quantity	desc_
152	12	Pav bhaji	105	2	Delicious pav bhaji with 4 pavs
1011	11	dal	50	1	healthy
1021	12	dal fry	70	1	healthy
1031	13	rice	50	1	healthy
NULL	NULL	NULL	NULL	NULL	NULL



```
CREATE TRIGGER autofill_bill
AFTER INSERT ON food_order
for each row

INSERT INTO bill VALUES(new.food_id,'food order',new.price*new.quantity,current_date());

INSERT INTO food_order VALUES(152,12,'Pav bhaji',105,2,'Delicious pav bhaji with 4 pavs');
select * from bill;
```

order_id	desc_	amount	date_
152	food order	210	2022-04-17
NULL	NULL	NULL	NULL

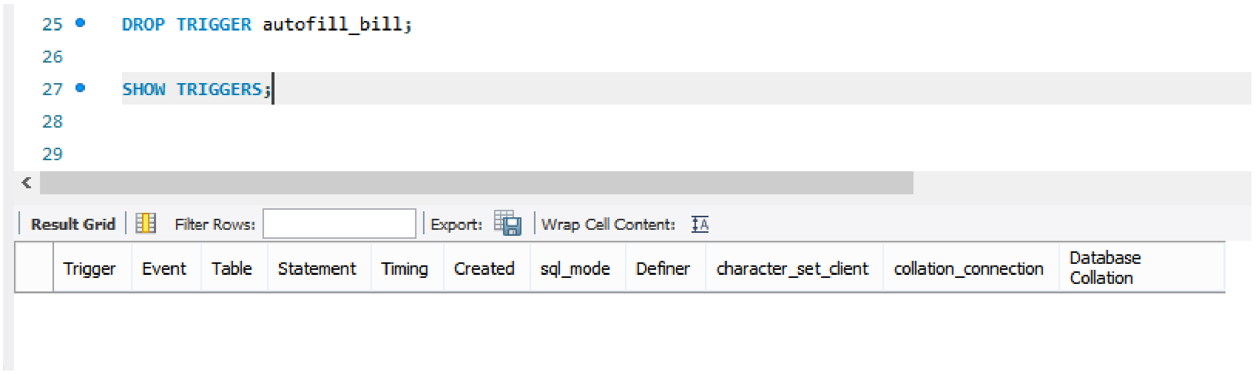
2. Data dictionary for triggers



```
SHOW TRIGGERS;
```

Trigger	Event	Table	Statement	Timing	Created	sql_mode	Definer	character_set_c
autofill_bill	INSERT	food_order	INSERT INTO bill VALUES(new.food_id,'food ord...	AFTER	2022-04-17 01:20:17.24	STRICT_TRANS_TABLES,NO_ENGINE_SUBSTIT...	root@localhost	utf8mb4

3. Dropping triggers

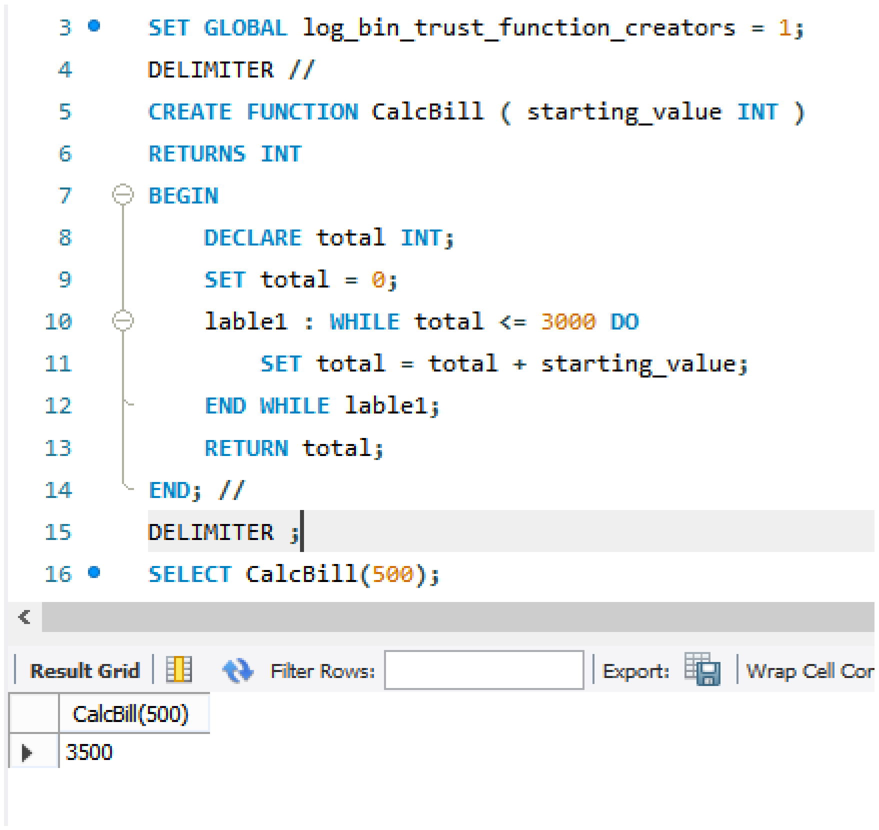


```
DROP TRIGGER autofill_bill;

SHOW TRIGGERS;
```

Trigger	Event	Table	Statement	Timing	Created	sql_mode	Definer	character_set_client	collation_connection	Database Collation

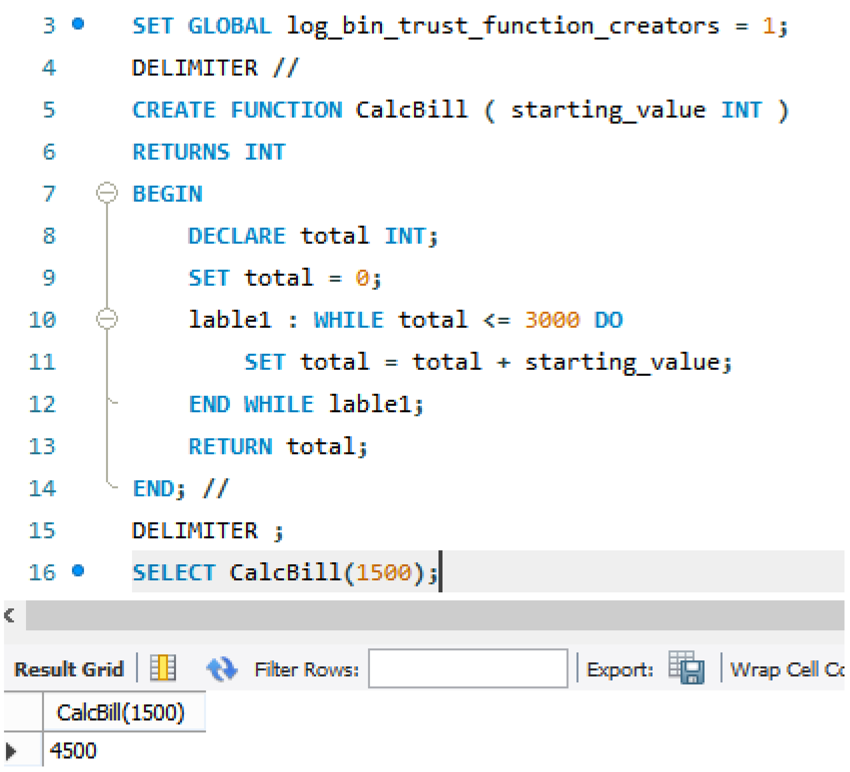
Expt 9 :

Functions :



```sql
SET GLOBAL log_bin_trust_function_creators = 1;
DELIMITER //
CREATE FUNCTION CalcBill ( starting_value INT )
RETURNS INT
BEGIN
    DECLARE total INT;
    SET total = 0;
    lable1 : WHILE total <= 3000 DO
        SET total = total + starting_value;
    END WHILE lable1;
    RETURN total;
END; //
DELIMITER ;
SELECT CalcBill(500);
```

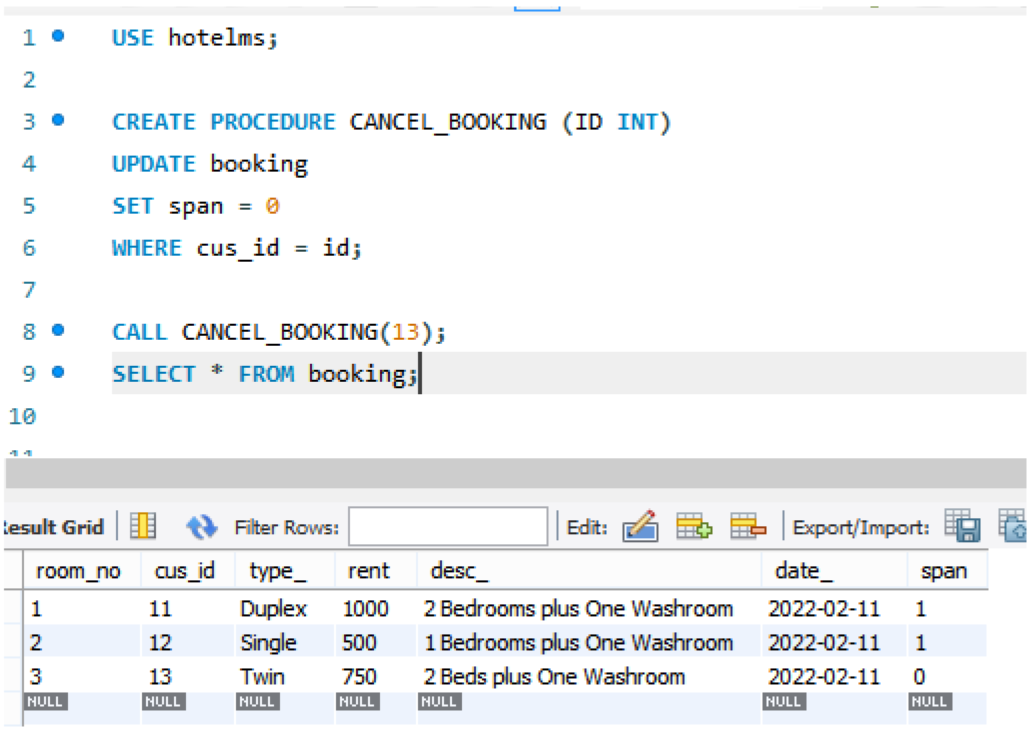
```
CalcBill(500)
3500
```



```sql
SET GLOBAL log_bin_trust_function_creators = 1;
DELIMITER //
CREATE FUNCTION CalcBill ( starting_value INT )
RETURNS INT
BEGIN
    DECLARE total INT;
    SET total = 0;
    lable1 : WHILE total <= 3000 DO
        SET total = total + starting_value;
    END WHILE lable1;
    RETURN total;
END; //
DELIMITER ;
SELECT CalcBill(1500);
```

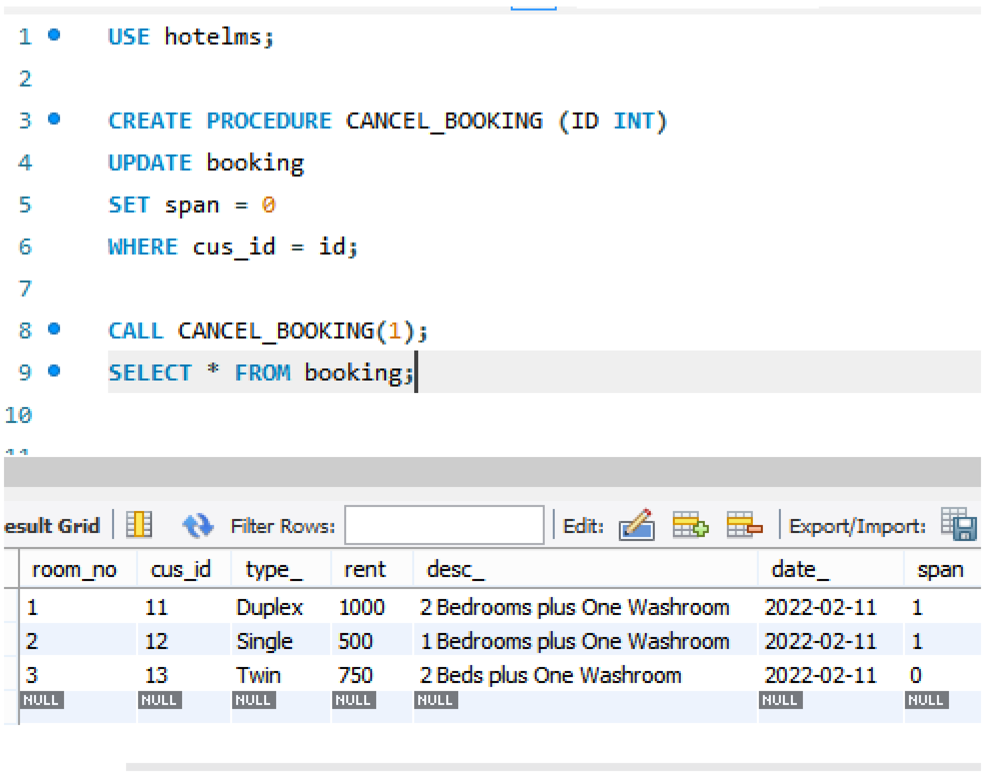
```
CalcBill(1500)
4500
```

Procedures :



```
USE hotelms;

CREATE PROCEDURE CANCEL_BOOKING (ID INT)
UPDATE booking
SET span = 0
WHERE cus_id = id;

CALL CANCEL_BOOKING(13);
SELECT * FROM booking;
```

room_no	cus_id	type_	rent	desc_	date_	span
1	11	Duplex	1000	2 Bedrooms plus One Washroom	2022-02-11	1
2	12	Single	500	1 Bedrooms plus One Washroom	2022-02-11	1
3	13	Twin	750	2 Beds plus One Washroom	2022-02-11	0
NULL	NULL	NULL	NULL	NULL	NULL	NULL



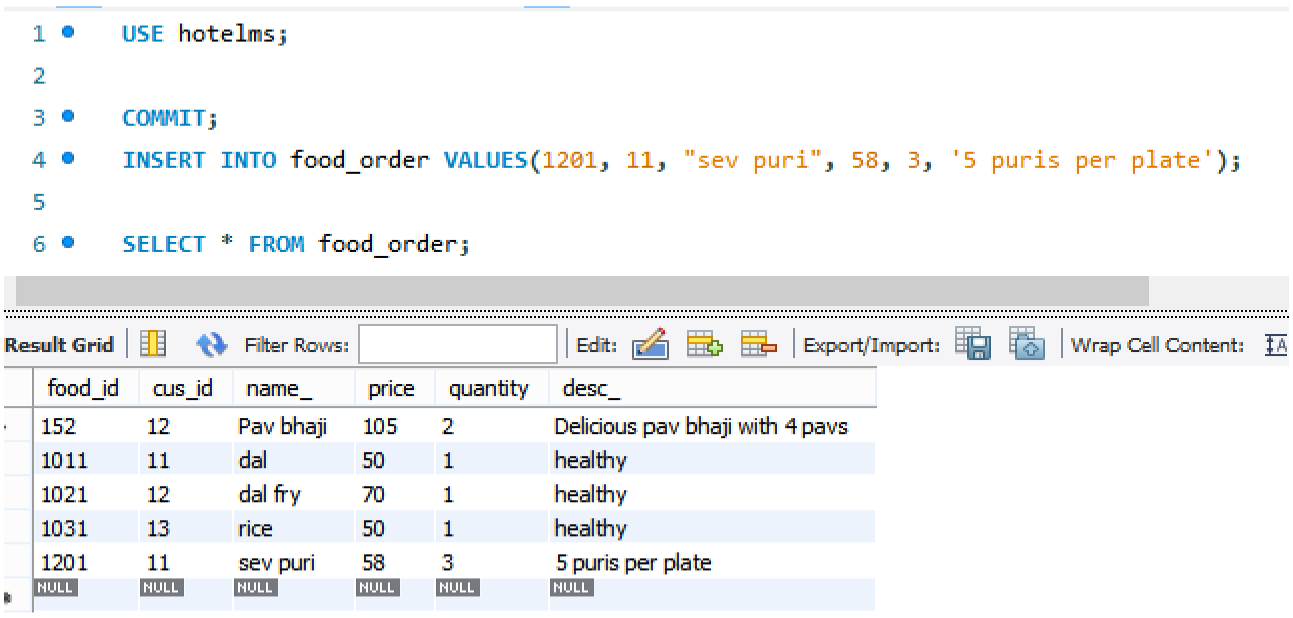
```
USE hotelms;

CREATE PROCEDURE CANCEL_BOOKING (ID INT)
UPDATE booking
SET span = 0
WHERE cus_id = id;

CALL CANCEL_BOOKING(1);
SELECT * FROM booking;
```

room_no	cus_id	type_	rent	desc_	date_	span
1	11	Duplex	1000	2 Bedrooms plus One Washroom	2022-02-11	1
2	12	Single	500	1 Bedrooms plus One Washroom	2022-02-11	1
3	13	Twin	750	2 Beds plus One Washroom	2022-02-11	0
NULL	NULL	NULL	NULL	NULL	NULL	NULL

Expt 10 :

TCL :

1. Commit :

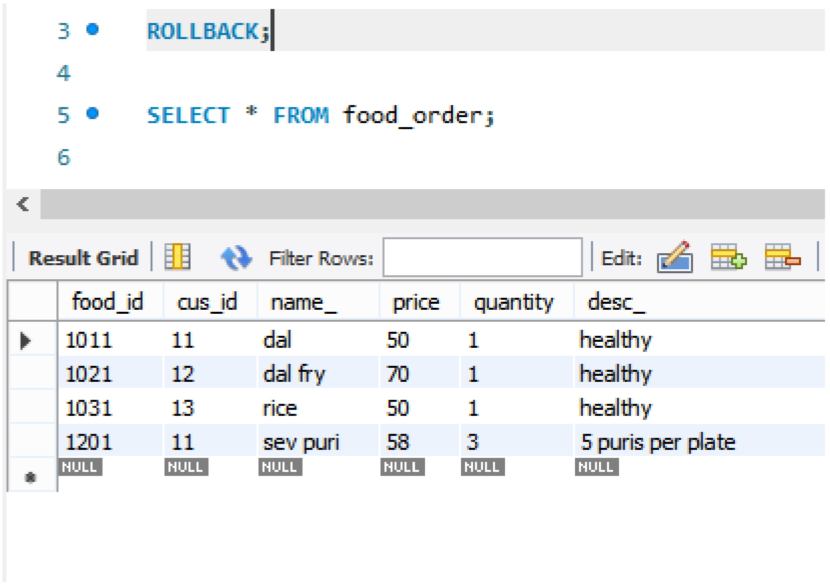


```
USE hotelms;

COMMIT;
INSERT INTO food_order VALUES(1201, 11, "sev puri", 58, 3, '5 puris per plate');

SELECT * FROM food_order;
```

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

food_id	cus_id	name_	price	quantity	desc_
152	12	Pav bhaji	105	2	Delicious pav bhaji with 4 pavs
1011	11	dal	50	1	healthy
1021	12	dal fry	70	1	healthy
1031	13	rice	50	1	healthy
1201	11	sev puri	58	3	5 puris per plate
NULL	NULL	NULL	NULL	NULL	NULL

2. Rollback:

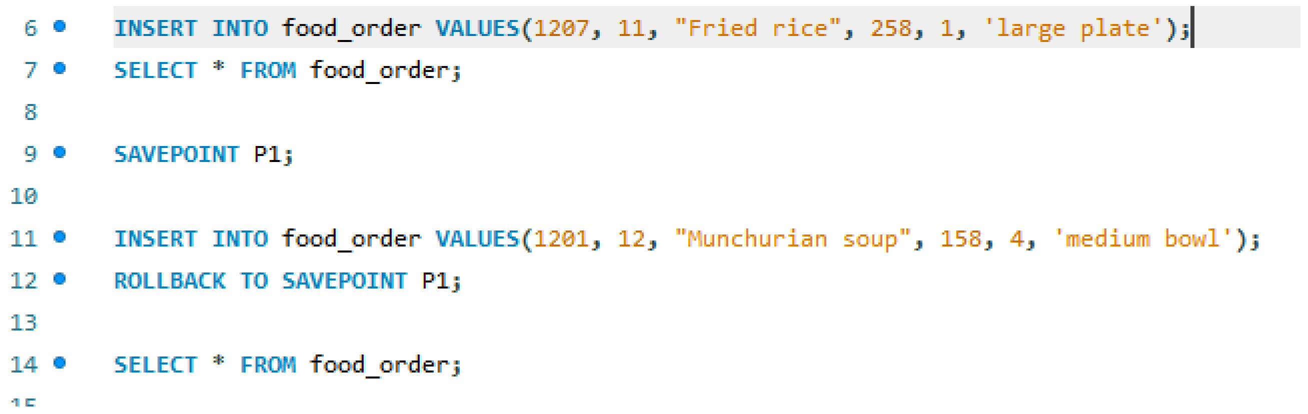


```
ROLLBACK;

SELECT * FROM food_order;

```

Result Grid | Filter Rows: | Edit:

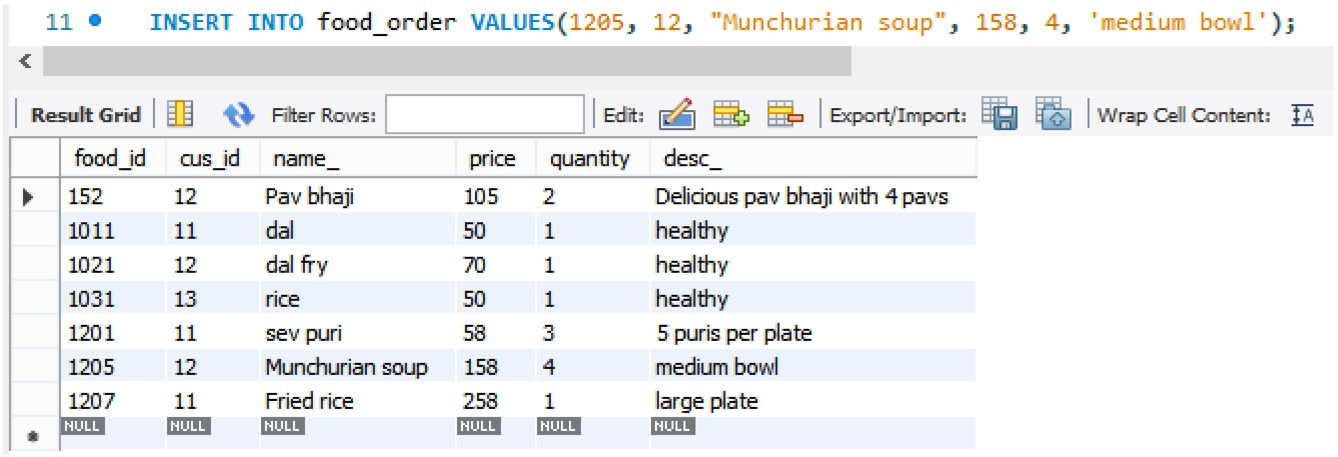
food_id	cus_id	name_	price	quantity	desc_
1011	11	dal	50	1	healthy
1021	12	dal fry	70	1	healthy
1031	13	rice	50	1	healthy
1201	11	sev puri	58	3	5 puris per plate
NULL	NULL	NULL	NULL	NULL	NULL

3. Savepoint:



```sql
INSERT INTO food_order VALUES(1207, 11, "Fried rice", 258, 1, 'large plate');
SELECT * FROM food_order;

SAVEPOINT P1;

INSERT INTO food_order VALUES(1201, 12, "Munchurian soup", 158, 4, 'medium bowl');
ROLLBACK TO SAVEPOINT P1;

SELECT * FROM food_order;
```

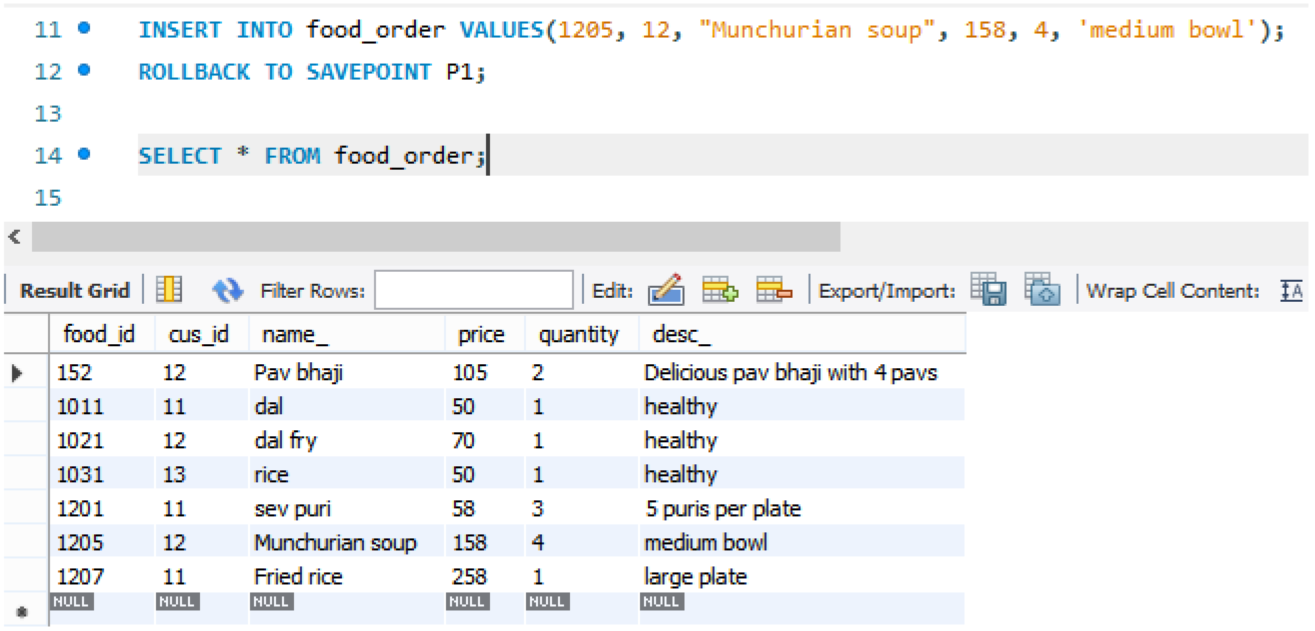
Before savepoint



```sql
INSERT INTO food_order VALUES(1205, 12, "Munchurian soup", 158, 4, 'medium bowl');
```

food_id	cus_id	name_	price	quantity	desc_
152	12	Pav bhaji	105	2	Delicious pav bhaji with 4 pavs
1011	11	dal	50	1	healthy
1021	12	dal fry	70	1	healthy
1031	13	rice	50	1	healthy
1201	11	sev puri	58	3	5 puris per plate
1205	12	Munchurian soup	158	4	medium bowl
1207	11	Fried rice	258	1	large plate
NULL	NULL	NULL	NULL	NULL	NULL

After reverting to savepoint

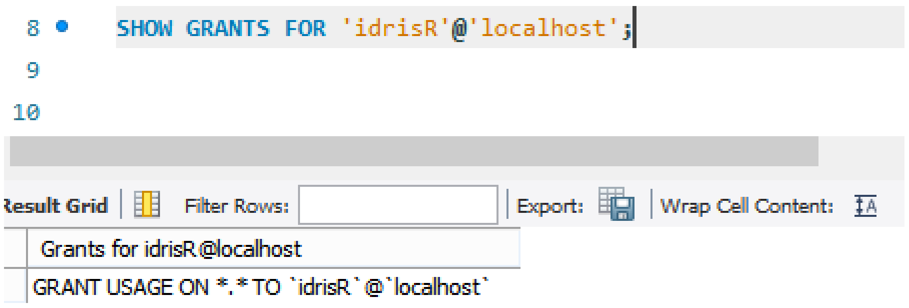


```sql
INSERT INTO food_order VALUES(1205, 12, "Munchurian soup", 158, 4, 'medium bowl');
ROLLBACK TO SAVEPOINT P1;

SELECT * FROM food_order;
```

food_id	cus_id	name_	price	quantity	desc_
152	12	Pav bhaji	105	2	Delicious pav bhaji with 4 pavs
1011	11	dal	50	1	healthy
1021	12	dal fry	70	1	healthy
1031	13	rice	50	1	healthy
1201	11	sev puri	58	3	5 puris per plate
1205	12	Munchurian soup	158	4	medium bowl
1207	11	Fried rice	258	1	large plate
NULL	NULL	NULL	NULL	NULL	NULL

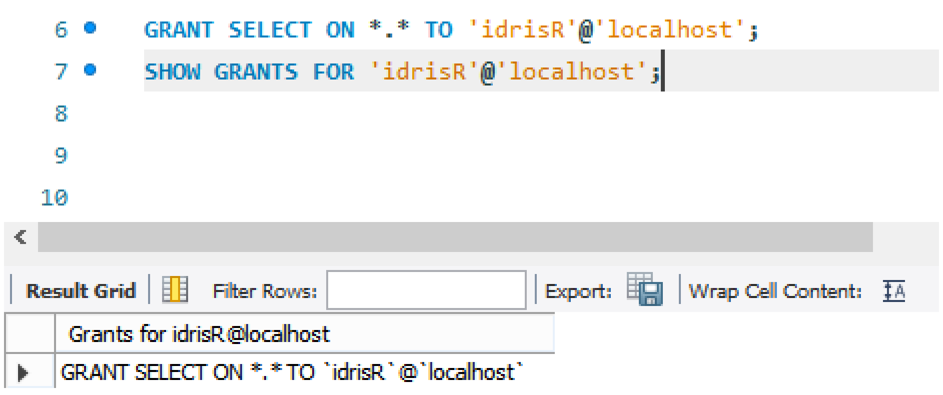
DCL :

1. **Grant :**



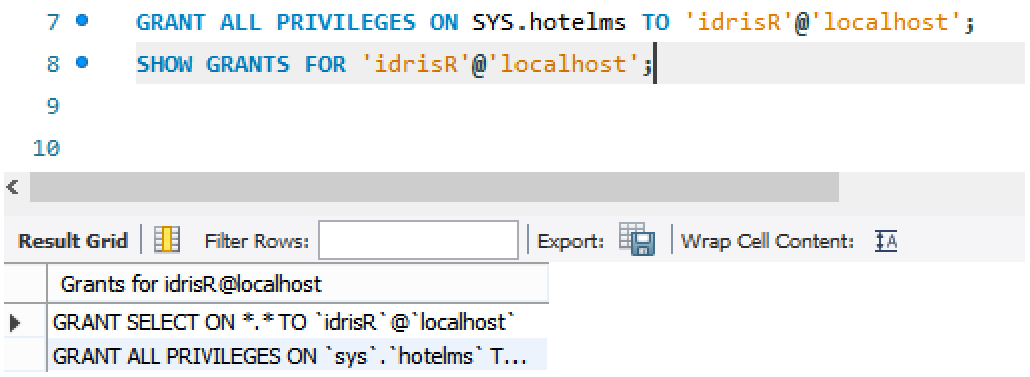
```sql
SHOW GRANTS FOR 'idrisR'@'localhost';
```

```
Grants for idrisR@localhost
GRANT USAGE ON *.* TO `idrisR`@`localhost`
```



```sql
GRANT SELECT ON *.* TO 'idrisR'@'localhost';
SHOW GRANTS FOR 'idrisR'@'localhost';
```

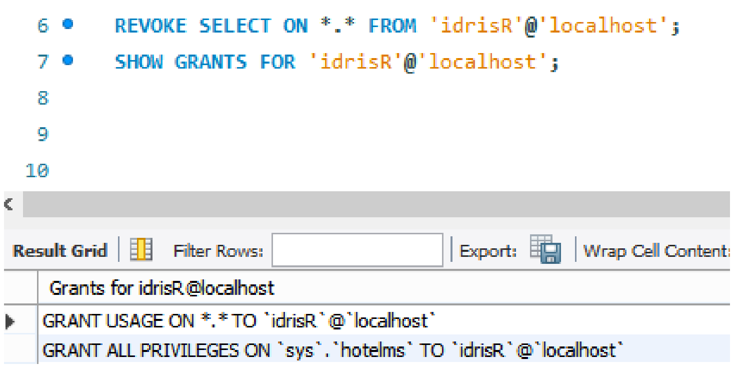
```
Grants for idrisR@localhost
GRANT SELECT ON *.* TO `idrisR`@`localhost`
```



```sql
GRANT ALL PRIVILEGES ON SYS.hotelms TO 'idrisR'@'localhost';
SHOW GRANTS FOR 'idrisR'@'localhost';
```

```
Grants for idrisR@localhost
GRANT SELECT ON *.* TO `idrisR`@`localhost`
GRANT ALL PRIVILEGES ON `sys`.`hotelms` T...
```

2. **Revoke :**

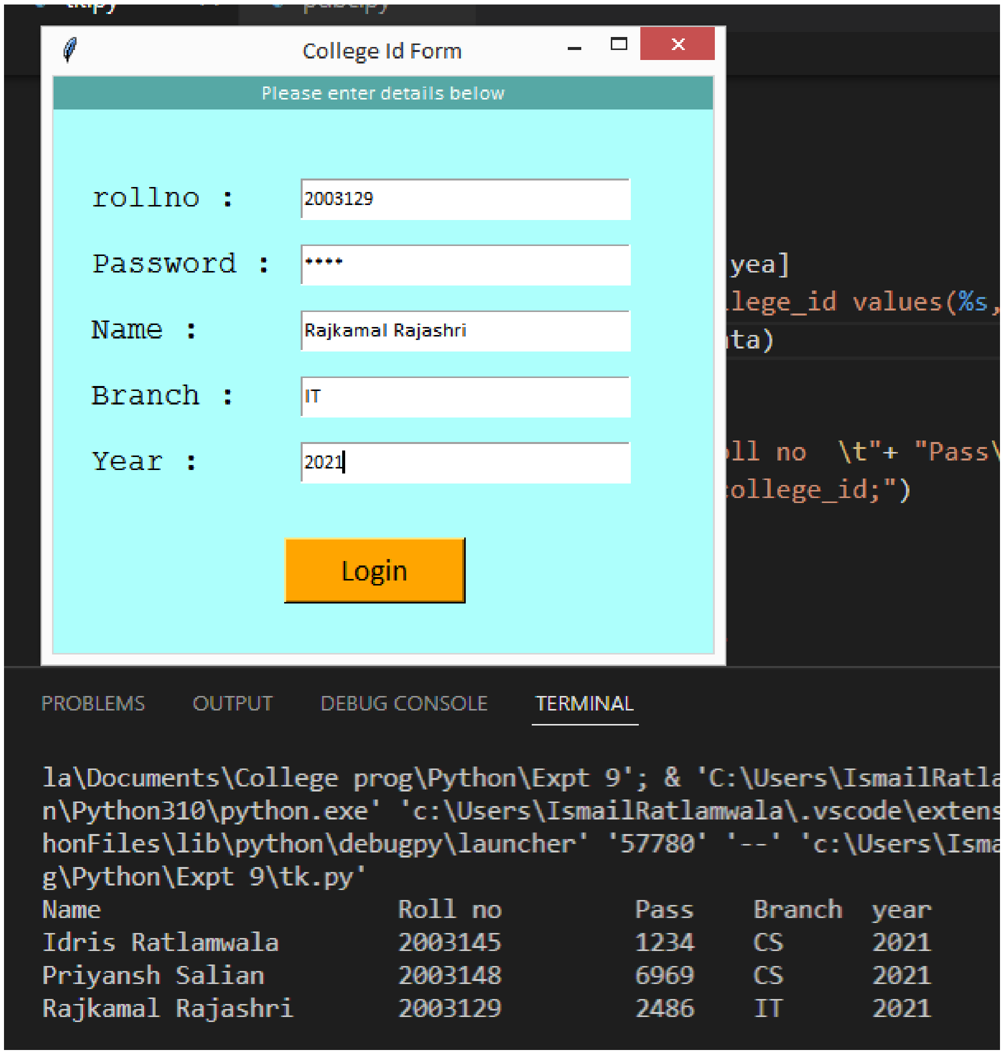


```sql
REVOKE SELECT ON *.* FROM 'idrisR'@'localhost';
SHOW GRANTS FOR 'idrisR'@'localhost';
```

```
Grants for idrisR@localhost
GRANT USAGE ON *.* TO `idrisR`@`localhost`
GRANT ALL PRIVILEGES ON `sys`.`hotelms` TO `idrisR`@`localhost`
```

Expt 11 :

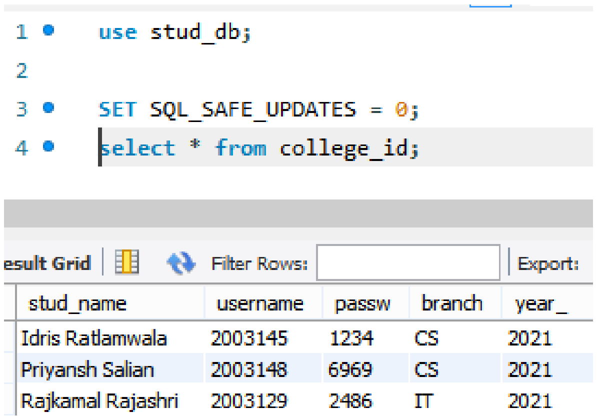
```
import mysql.connector
from tkinter import *
from tkinter import messagebox
def login():
    uname=rollno.get()
    pwd=password.get()
    nam=name.get()
    yea=int(year.get())
    bran=branch.get()

    try:
        conn =
mysql.connector.connect(host='localhost',user='root',password='1234',db='stud
_db')
        cursor = conn.cursor()
        create_query = '''create table if not exists college_id
        (stud_name varchar(50),
        username varchar(50),
        passw varchar(20),
        branch varchar(20),
        year_ int
        );'''
        cursor.execute(create_query)

        data = [nam, uname, pwd, bran, yea]
        insert_query = "insert into college_id values(%s,%s,%s,%s,%s)"
        cursor.execute(insert_query, data)
        conn.commit()

        print("Name\t"+ "Roll no  \t"+ "Pass\t" +"Branch\t" +"year")
        cursor.execute("select * from college_id;")
        data = cursor.fetchall()
        for record in data :
            for value in record :
                print(value, end=" \t")
            print()

    except Exception as e :
        print(e)

    if uname=='' or pwd==''or nam==''or yea==''or bran=='':
       messagebox.showerror('Error', 'Plese enter all details')
    else:
        messagebox.showinfo('Successful', 'Your data is saved\nsuccessfully
!')
```



rollno :	2003129
Password :	****
Name :	Rajkamal Rajashri
Branch :	IT
Year :	2021

```
yea]
lege_id values(%s,
ta)

ll no  \t"+ "Pass\
ollege_id;")
```

```
la\Documents\College prog\Python\Expt 9'; & 'C:\Users\IsmailRatla
n\Python310\python.exe' 'c:\Users\IsmailRatlamwala\.vscode\extens
honFiles\lib\python\debugpy\launcher' '57780' '--' 'c:\Users\Isma
g\Python\Expt 9\tk.py'
Name                    Roll no         Pass    Branch   year
Idris Ratlamwala        2003145         1234    CS       2021
Priyansh Salian         2003148         6969    CS       2021
Rajkamal Rajashri       2003129         2486    IT       2021
```



```sql
use stud_db;

SET SQL_SAFE_UPDATES = 0;
select * from college_id;
```

stud_name	username	passw	branch	year_
Idris Ratlamwala	2003145	1234	CS	2021
Priyansh Salian	2003148	6969	CS	2021
Rajkamal Rajashri	2003129	2486	IT	2021