

TABLES:

ORDER

ORDER DETAILS

CUSTOMER

PAYMENT

USER

RATING

MENU

ER DIAGRAM

(Entity Relationship Diagram)

CREATION OF DATABASE

/*Table1*/

```
CREATE TABLE ORDER_DETAILS (  
ID VARCHAR(50),  
order_id VARCHAR(50),  
menu_id VARCHAR(50) UNIQUE,  
amount INT,  
no_of_serving INT,  
PRIMARY KEY(ID));
```

/*TABLE2*/

```
CREATE TABLE USER (  
ID VARCHAR(50),  
FULLNAME VARCHAR(50),  
EMAIL_ADDRESS VARCHAR(50) UNIQUE,  
CONTACT VARCHAR(13) CHECK (CONTACT >=0),  
USERNAME VARCHAR(20),  
PASSWORD VARCHAR(8),  
PRIMARY KEY(ID));
```

/*PRIMARY KEY CONSTRAINT*/

/*TABLE3*/

```
CREATE TABLE PAYMENT (  
ID VARCHAR(50),  
order_id VARCHAR(50) UNIQUE,  
amount INT,  
paid_by CHAR(50),  
date_ DATE NOT NULL,  
PRIMARY KEY(ID));
```

/*CHECK CONSTRAINT*/

/*TABLE4*/

```
CREATE TABLE CUSTOMER (  
ID VARCHAR(50),  
NAME CHAR(50),  
EMAIL VARCHAR(50) UNIQUE,  
ACCOUNT_STATUS INT,  
USERNAME VARCHAR(20),  
PASSWORD VARCHAR(10),  
PHONE_NO VARCHAR(13) CHECK (PHONE_NO >=0),  
PRIMARY KEY(ID));
```

/*UNIQUE CONSTARINT*/

/*TABLE5*/

```
CREATE TABLE RATING(  
ID VARCHAR(50),  
score DECIMAL(10,2),  
menu_id VARCHAR(50) UNIQUE,  
customer_id VARCHAR(50) UNIQUE,
```

```
date_recorded DATE,  
remarks VARCHAR(50),  
PRIMARY KEY(ID));
```

/*FOREIGN KEY CONSTRAINT*/

/*TABLE6*/

```
CREATE TABLE MENU (  
ID VARCHAR(50),  
price INT,  
type_id VARCHAR(50) PRIMARY KEY,  
name_ VARCHAR(50),  
status_ INT,  
FOREIGN KEY (ID) REFERENCES USER(ID)  
);
```

/*NOT NULL CONSTRAINT*/

/*TABLE7*/

```
CREATE TABLE ORDER_  
ID VARCHAR(50),  
customer_id VARCHAR(50),  
total_amount INT,  
order_status VARCHAR(50),  
processed_by VARCHAR(50),  
order_date DATE NOT NULL,  
PRIMARY KEY(ID));
```

➤ DDL

(COMMANDS: CREATE ALTER DROP VIEW RENAME)

• Alter

Adding a new column

ALTER TABLE USER

ADD DOB DATE;

SELECT * FROM USER;

ID	FULLNAME	EMAIL_ADDRESS	CONTACT	USERNAME	PASSWORD	DOB
3	SNEHAL	SP23@GMAIL.COM	9891100001	S23	SNE	
4	SAKSHI	SJ3@GMAIL.COM	9888100001	S24	SAK	
1	JIA	FIA002@GMAIL.COM	9771100001	S26	12J	
2	MAHIKA	K2@GMAIL.COM	8891400001	G23	6GS	
5	SHIVANI	M2F3@GMAIL.COM	7791900001	Y23	YEN	

• DROP

Dropping a column

ALTER TABLE USER

DROP DOB;

SELECT * FROM USER;

ID	FULLNAME	EMAIL_ADDRESS	CONTACT	USERNAME	PASSWORD
3	SNEHAL	SP23@GMAIL.COM	9891100001	S23	SNE
4	SAKSHI	SJ3@GMAIL.COM	9888100001	S24	SAK
1	JIA	FIA002@GMAIL.COM	9771100001	S26	12J
2	MAHIKA	K2@GMAIL.COM	8891400001	G23	6GS
5	SHIVANI	M2F3@GMAIL.COM	7791900001	Y23	YEN

Adding a new constraint

ALTER TABLE USER

ADD CONSTRAINT P_USER PRIMARY KEY (ID);

SELECT * FROM USER;

	ID	FULLNAME	EMAIL_ADDRESS	CONTACT	USERNAME	PASSWORD
▶	1	JIA	FIA002@GMAIL.COM	9771100001	S26	12J
	2	MAHIKA	K2@GMAIL.COM	8891400001	G23	6GS
	3	SNEHAL	SP23@GMAIL.COM	9891100001	S23	SNE
	4	SAKSHI	SJ3@GMAIL.COM	9888100001	S24	SAK
	5	SHIVANI	M2F3@GMAIL.COM	7791900001	Y23	YEN
*	HULL	HULL	HULL	HULL	HULL	HULL

Dropping constraint

ALTER TABLE USER

DROP CONSTRAINT P_USER;

SELECT * FROM USER;

ID	FULLNAME	EMAIL_ADDRESS	CONTACT	USERNAME	PASSWORD
3	SNEHAL	SP23@GMAIL.COM	9891100001	S23	SNE
4	SAKSHI	SJ3@GMAIL.COM	9888100001	S24	SAK
1	JIA	FIA002@GMAIL.COM	9771100001	S26	12J
2	MAHIKA	K2@GMAIL.COM	8891400001	G23	6GS
5	SHIVANI	M2F3@GMAIL.COM	7791900001	Y23	YEN

Modify column

ALTER TABLE USER

ALTER COLUMN FULLNAME CHAR(50);

SELECT * FROM USER;

ID	FULLNAME	EMAIL_ADDRESS	CONTACT	USERNAME	PASSWORD
3	SNEHAL	SP23@GMAIL.COM	9891100001	S23	SNE
4	SAKSHI	SJ3@GMAIL.COM	9888100001	S24	SAK
1	JIA	FIA002@GMAIL.COM	9771100001	S26	12J
2	MAHIKA	K2@GMAIL.COM	8891400001	G23	6GS
5	SHIVANI	M2F3@GMAIL.COM	7791900001	Y23	YEN

• VIEW

CREATE VIEW USER_VIEW AS

SELECT ID, FULLNAME, USERNAME, CONTACT

FROM USER

WHERE ID>1;

SELECT * FROM USER_VIEW;

ID	FULLNAME	USERNAME	CONTACT
2	MAHIKA	G23	8891400001
3	SNEHAL	S23	9891100001
4	SAKSHI	S24	9888100001
5	SHIVANI	Y23	7791900001
6	ABHI	AG2	9121100001
7	NIK	NNI	8127002201

• RENAME

INSERT INTO RATING (MENU_ID, ID, SCORE, REMARKS, DATE_RECORDED, CUSTOMER_ID) VALUES ("E12", "2", 4, "NICE", "2013-03-12", "122"), ("F31", "5", 3.5, "GOOD", "2022-09-11", "34U");

SELECT * FROM RATING;

ALTER TABLE RATING

RENAME TO RATING_TABLE_;

SELECT * FROM RATING_TABLE_

WHERE MENU_ID>5;

ID	score	menu_id	customer_id	date_recorded	remarks
2	4	12	122	2013-03-12	NICE
5	3.5	31	34	2022-09-11	GOOD

➤ DML

(COMMANDS: SELECT MERGE INSERT DELETE UPDATE)

- **SELECT**

```
SELECT ID, AMOUNT, ORDER_ID
FROM PAYMENT;
```

ID	amount	order_id
1	2300	43J2
2	1050	58S2
5	8402	62H9
3	590	JH73
4	2300	GHWO

- **MERGE**

```
MERGE INTO USER AS U
USING CUSTOMER AS CUST
ON U.ID = CUST.ID
WHEN MATCHED THEN
    UPDATE SET U.CONTACT = CUST.CONTACT, U.USERNAME = U.USERNAME
WHEN NOT MATCHED THEN
    INSERT (ID, CONTACT, USERNAME)
    VALUES (CUST.ID, CUST.CONTACT, CUST.USERNAME)
WHEN NOT MATCHED BY CUST THEN
    DELETE;
```

- **Insert**

```
INSERT INTO ORDER_DETAILS (ID,order_id,menu_id,amount,no_of_serving) VALUES
('1',"32S2","119O",1240,5),('2',"33S2","1195",240,2),('5',"34S2","12OO",1110,3),('4',"35S2","1F9O",140,1),('3',"36S2","109O",5240,7);
```

```
SELECT * FROM ORDER_DETAILS;
```

ID	order_id	menu_id	amount	no_of_serving
1	32S2	119O	1240	5
2	33S2	1195	240	2
5	34S2	12OO	1110	3
4	35S2	1F9O	140	1
3	36S2	109O	5240	7

```
INSERT INTO USER (ID,FULLNAME,EMAIL_ADDRESS,CONTACT,USERNAME,PASSWORD) VALUES
(3,"SNEHAL","SP23@GMAIL.COM",9891100001,"S23","SNE"),(4,"SAKSHI","SJ3@GMAIL.COM",9888100001,"S24","SAK"),(1,"JIA","FIA002@GMAIL.COM",9771100001,"S26","12J"),(2,"MAHIKA","K2@GMAIL.COM",8891400001,"G23","6GS"),(5,"SHIVANI","M2F3@GMAIL.COM",7791900001,"Y23","YEN");
```

ONLINE-FOOD-ORDERING-SYSTEM

SELECT * FROM USER;

ID	FULLNAME	EMAIL_ADDRESS	CONTACT	USERNAME	PASSWORD
3	SNEHAL	SP23@GMAIL.COM	9891100001	S23	SNE
4	SAKSHI	SJ3@GMAIL.COM	9888100001	S24	SAK
1	JIA	FIA002@GMAIL.COM	9771100001	S26	12J
2	MAHIKA	K2@GMAIL.COM	8891400001	G23	6GS
5	SHIVANI	M2F3@GMAIL.COM	7791900001	Y23	YEN

INSERT INTO PAYMENT (ID,order_id,amount,paid_by,date_) VALUES (1,"43J2",2300,"DEBIT CARD","2023-09-21"),(2,"58S2",1050,"UPI","2023-05-01"),(5,"62H9",8402,"DEBIT CARD","2022-10-11"),(3,"JH73",590,"CASH","2023-12-02"),(4,"GHW0",2300,"CREDIT CARD","2021-02-28");
SELECT * FROM PAYMENT;

ID	order_id	amount	paid_by	date_
1	43J2	2300	DEBIT CARD	2023-09-21
2	58S2	1050	UPI	2023-05-01
5	62H9	8402	DEBIT CARD	2022-10-11
3	JH73	590	CASH	2023-12-02
4	GHW0	2300	CREDIT CARD	2021-02-28

INSERT INTO CUSTOMER (ID,NAME,EMAIL,PHONE_NO,USERNAME, PASSWORD,ACCOUNT_STATUS) VALUES (2,"MAHIKA","K2@GMAIL.COM",8891400001,"G23","6GS",52000),(3,"SNEHAL","SP23@GMAIL.COM",9891100001,"S23","SNE",26500),(4,"SAKSHI","SJ3@GMAIL.COM",9888100001,"S24","SAK",27699),(5,"SHIVANI","M2F3@GMAIL.COM",7791900001,"Y23","YEN",15446),(1,"JIA","FIA002@GMAIL.COM",9771100001,"S26","12J",23134);
SELECT * FROM CUSTOMER;

ID	NAME	EMAIL	ACCOUNT_STATUS	USERNAME	PASSWORD	PHONE_NO
2	MAHIKA	K2@GMAIL.COM	52000	G23	6GS	8891400001
3	SNEHAL	SP23@GMAIL.COM	26500	S23	SNE	9891100001
4	SAKSHI	SJ3@GMAIL.COM	27699	S24	SAK	9888100001
5	SHIVANI	M2F3@GMAIL.COM	15446	Y23	YEN	7791900001
1	JIA	FIA002@GMAIL.COM	23134	S26	12J	9771100001

- **Update**

Update (using with)

UPDATE ORDER_DETAILS

SET amount = amount * 1.10

WHERE ID = "3";

SELECT * FROM ORDER_DETAILS;

ID	order_id	menu_id	amount	no_of_serving
1	32S2	119O	1240	5
2	33S2	1195	240	2
5	34S2	120O	1110	3
4	35S2	1F9O	140	1
3	36S2	109O	5764.0000000000001	7

Update(without where)

UPDATE ORDER_DETAILS

SET amount = amount * 1.10;

SELECT * FROM ORDER_DETAILS;

ID	order_id	menu_id	amount	no_of_serving
1	32S2	119O	1364	5
2	33S2	1195	264	2
5	34S2	120O	1221	3
4	35S2	1F9O	154	1
3	36S2	109O	6340.40000000000015	7

- Delete**

Delete (with where)

DELETE FROM ORDER_DETAILS

WHERE amount < 1000;

SELECT * FROM ORDER_DETAILS;

ID	order_id	menu_id	amount	no_of_serving
1	32S2	119O	1364	5
5	34S2	120O	1221	3
3	36S2	109O	6340.40000000000015	7

Delete(without where)

DELETE FROM ORDER_DETAILS;

SELECT * FROM ORDER_DETAILS;

SQL query successfully executed. However, the result set is empty.

Select (using where)

Select amount,Order_id FROM PAYMENT
WHERE AMOUNT>1000;

amount	order_id
2300	43J2
1050	58S2
8402	62H9
2300	GHWO

Select (without where)

Select amount,Order_id FROM PAYMENT

amount	order_id
2300	43J2
1050	58S2
8402	62H9
590	JH73
2300	GHWO

➤ Boolean operators

• AND

SELECT * FROM USER
WHERE ID = 3 AND PASSWORD = "SNE";

ID	FULLNAME	EMAIL_ADDRESS	CONTACT	USERNAME	PASSWORD
3	SNEHAL	SP23@GMAIL.COM	9891100001	S23	SNE

• OR

SELECT * FROM USER
WHERE ID = 3 OR PASSWORD = "SAK";

ID	FULLNAME	EMAIL_ADDRESS	CONTACT	USERNAME	PASSWORD
3	SNEHAL	SP23@GMAIL.COM	9891100001	S23	SNE
4	SAKSHI	SJ3@GMAIL.COM	9888100001	S24	SAK

(WITHOUT)

SELECT * FROM USER;

ID	FULLNAME	EMAIL_ADDRESS	CONTACT	USERNAME	PASSWORD
3	SNEHAL	SP23@GMAIL.COM	9891100001	S23	SNE
4	SAKSHI	SJ3@GMAIL.COM	9888100001	S24	SAK
1	JIA	FIA002@GMAIL.COM	9771100001	S26	12J
2	MAHIKA	K2@GMAIL.COM	8891400001	G23	6GS
5	SHIVANI	M2F3@GMAIL.COM	7791900001	Y23	YEN

• NOT

SELECT * FROM USER
WHERE NOT ID = 3;

ID	FULLNAME	EMAIL_ADDRESS	CONTACT	USERNAME	PASSWORD
4	SAKSHI	SJ3@GMAIL.COM	9888100001	S24	SAK
1	JIA	FIA002@GMAIL.COM	9771100001	S26	12J
2	MAHIKA	K2@GMAIL.COM	8891400001	G23	6GS
5	SHIVANI	M2F3@GMAIL.COM	7791900001	Y23	YEN

➤ Logical operators

• BETWEEN

SELECT * FROM USER
WHERE ID BETWEEN 2 AND 4;

ID	FULLNAME	EMAIL_ADDRESS	CONTACT	USERNAME	PASSWORD
2	MAHIKA	K2@GMAIL.COM	8891400001	G23	6GS
3	SNEHAL	SP23@GMAIL.COM	9891100001	S23	SNE
4	SAKSHI	SJ3@GMAIL.COM	9888100001	S24	SAK

- **IN**

SELECT * FROM USER

WHERE ID IN (2,1,3);

ID	FULLNAME	EMAIL_ADDRESS	CONTACT	USERNAME	PASSWORD
1	JIA	FIA002@GMAIL.COM	9771100001	S26	12J
2	MAHIKA	K2@GMAIL.COM	8891400001	G23	6GS
3	SNEHAL	SP23@GMAIL.COM	9891100001	S23	SNE

- **LIKE**

SELECT * FROM USER

WHERE FULLNAME LIKE "S%";

ID	FULLNAME	EMAIL_ADDRESS	CONTACT	USERNAME	PASSWORD
3	SNEHAL	SP23@GMAIL.COM	9891100001	S23	SNE
4	SAKSHI	SJ3@GMAIL.COM	9888100001	S24	SAK
5	SHIVANI	M2F3@GMAIL.COM	7791900001	Y23	YEN

- **Exists**

SELECT FULLNAME

FROM USER

WHERE EXISTS (

SELECT EMAIL_ADDRESS

FROM CUSTOMER

WHERE USER.id = CUSTOMER.d);

	ID	FULLNAME	EMAIL_ADDRESS	CONTACT	USERNAME	PASSWORD
▶	1	JIA	FIA002@GMAIL.COM	9771100001	S26	12J
	2	MAHIKA	K2@GMAIL.COM	8891400001	G23	6GS
	3	SNEHAL	SP23@GMAIL.COM	9891100001	S23	SNE
	4	SAKSHI	SJ3@GMAIL.COM	9888100001	S24	SAK
	5	SHIVANI	M2F3@GMAIL.COM	7791900001	Y23	YEN

➤ Inbuilt functions

- **Sum**

SELECT SUM(amount) AS net_worth FROM payment;

net_worth
14642

- **MAX**

SELECT MAX(amount) AS MAX_AMOUNT FROM payment;

MAX_AMOUNT
8402

- **MIN**

SELECT MIN(amount) AS MIN_AMOUNT FROM payment;

MIN_AMOUNT
590

- **AVG**

SELECT AVG(amount) AS AVG_AMOUNT FROM payment;

AVG_AMOUNT
2928.4

- **DATE**

SELECT DATEDIFF("2021-02-28","2023-05-01") AS DATE_DIFFERENCE;

DATE_DIFFERENCE
-792



- **TRIM**

SELECT TRIM(' DEBIT , CARD ') AS trimmed_string;

trimmed_string
DEBIT , CARD

- **Having and group by clause**

```
SELECT ID, SUM(amount) AS total_order_value
```

```
FROM payment
```

```
GROUP BY ID
```

```
HAVING SUM(amount) > 1000;
```

ID	total_order_value
1	2300
2	1050
4	2300
5	8402

➤ JOINS

- **INNER JOIN**

```
SELECT USER.FULLNAME, USER.USERNAME, PAYMENT.AMOUNT
```

```
FROM USER
```

```
INNER JOIN PAYMENT ON USER.ID = PAYMENT.ID;
```

FULLNAME	USERNAME	amount
SNEHAL	S23	590
SAKSHI	S24	2300
JIA	S26	2300
MAHIKA	G23	1050
SHIVANI	Y23	8402

- **EQUI JOIN (=)**

```
SELECT USER.FULLNAME, USER.USERNAME, PAYMENT.AMOUNT
```

```
FROM USER
```

```
INNER JOIN PAYMENT ON USER.ID = PAYMENT.ID;
```

FULLNAME	USERNAME	amount
SNEHAL	S23	590
SAKSHI	S24	2300
JIA	S26	2300
MAHIKA	G23	1050
SHIVANI	Y23	8402

- **NATURAL JOIN**

```
SELECT USER.FULLNAME, USER.USERNAME, PAYMENT.AMOUNT
```

```
FROM USER
```

```
NATURAL JOIN PAYMENT;
```

FULLNAME	USERNAME	amount
SNEHAL	S23	590
SAKSHI	S24	2300
JIA	S26	2300
MAHIKA	G23	1050
SHIVANI	Y23	8402

- **LEFT OUTER JOIN**

```
INSERT INTO USER (ID,FULLNAME,EMAIL_ADDRESS,CONTACT,USERNAME,PASSWORD) VALUES  
(7,"ABHI","AB7@GMAIL.COM",9121100001,"AG2","87G");
```

```
INSERT INTO USER (ID,FULLNAME,EMAIL_ADDRESS,CONTACT,USERNAME,PASSWORD) VALUES
(19,"NIK","NB7@EMAIL.COM",8127002201,"NNI","G9JK");
```

```
SELECT USER.FULLNAME, USER.USERNAME, PAYMENT.AMOUNT
```

```
FROM USER
```

```
LEFT OUTER JOIN PAYMENT ON USER.ID = PAYMENT.ID;
```

FULLNAME	USERNAME	amount
SNEHAL	S23	590
SAKSHI	S24	2300
JIA	S26	2300
MAHIKA	G23	1050
SHIVANI	Y23	8402
ABHI	AG2	
NIK	NNI	

- **RIGHT OUTER JOIN**

```
INSERT INTO PAYMENT (ID,order_id,amount,paid_by,date_) VALUES (8,"49R2",1300,"CREDIT
CARD","2022-09-21");
```

```
INSERT INTO PAYMENT (ID,order_id,amount,paid_by,date_) VALUES (9,"11Y2",3300,"DEBIT
CARD","2022-04-01");
```

```
SELECT USER.FULLNAME, USER.USERNAME, PAYMENT.AMOUNT
```

```
FROM USER
```

```
RIGHT OUTER JOIN PAYMENT ON USER.ID = PAYMENT.ID;
```

	FULLNAME	USERNAME	AMOUNT
▶	JIA	S26	2300
	MAHIKA	G23	1050
	SNEHAL	S23	590
	SAKSHI	S24	2300
	SHIVANI	Y23	8402
	NULL	NULL	1300
	NULL	NULL	3300

- **FULL OUTER JOIN**

```
SELECT USER.FULLNAME, USER.USERNAME, PAYMENT.AMOUNT
```

```
FROM USER
```

```
FULL OUTER JOIN PAYMENT ON USER.ID = PAYMENT.ID;
```


ONLINE-FOOD-ORDERING-SYSTEM

	ID	order_id	amount	paid_by	date_
▶	1	43J2	2300	DEBIT CARD	2023-09-21
	2	58S2	1050	UPI	2023-05-01
	3	JH73	590	CASH	2023-12-02
	4	GHWO	2300	CREDIT CARD	2021-02-28
	5	62H9	8402	DEBIT CARD	2022-10-11
	8	49R2	1300	CREDIT CARD	2022-09-21
	9	11Y2	3300	DEBIT CARD	2022-04-01
*	NULL	NULL	NULL	NULL	NULL