SVKM’s NMIMS
School of Technology Management & Engineering, Chandigarh
A.Y. 2023 - 24
Course: Database Management Systems

Project Report

Program	MBA TECH CE	
Semester	IV	
Name of the Project:	GYM MANAGEMENT	
Details of Project Members		
Batch	Roll No.	Name
D1	N019	SAMKIT SANGHVI
D1	N025	VANSH ISRANI
Date of Submission: 12-03-2024		

Contribution of each project Members:

Roll No.	Name:	Contribution
N019	Samkit Sanghvi	SCHEMA, EER, SQL QUERIES, REPORT
N025	Vansh Israni	SCHEMA, EER, SQL QUERIES, REPORT

Github link of your project:

Note:

1. Create a readme file if you have multiple files
2. All files must be properly named (Example:R004_DBMSProject)
3. Submit all relevant files of your work (Report, all SQL files, Any other files)
4. **Plagiarism is highly discouraged (Your report will be checked for plagiarism)**

Rubrics for the Project evaluation:

First phase of evaluation: Innovative Ideas (5 Marks) Design and Partial implementation (5 Marks)	10 marks
Final phase of evaluation Implementation, presentation and viva, Self-Learning and Learning Beyond classroom	10 marks

Project Report on

GYM MANAGEMENT

by

Samkit Sanghvi, Roll number: N019

Vansh Israni, Roll number: N025

Course: DBMS

AY: 2023-24

Table of Contents

I. Storyline

A well-organized Gym Management Database is like a central hub where all important information about the fitness center is stored. It keeps track of members' details, such as their contact information, membership plans, and when they joined. This helps the gym manage subscriptions, billing, and stay in touch with members easily.

The database also keeps track of the gym's equipment, like what types of machines it has, how many, and when they need maintenance. This helps the gym keep its equipment in good shape and available for members to use.

Additionally, the database can include class schedules, who's teaching them, and how members can sign up for them. This makes it easier for members to find and join classes, and for the gym to keep track of everything.

Basically, a Gym Management Database helps gyms run smoothly, makes things easier for members, and helps the gym make smart decisions based on data.

II. Components of Database Design

Entities and Attributes:

Member:
Attributes: (member_id INT PRIMARY KEY, member_phone_no INT NULL, member_email VARCHAR(50) NULL, member_date_joined INT NULL, address VARCHAR(100) NULL, member_type VARCHAR(15) NULL)

Admin
Attributes: (admin_id INT PRIMARY KEY, phone_no INT NULL, email VARCHAR(25) NULL, name VARCHAR(20) NULL)

Membership
Attributes: (membership_id INT PRIMARY KEY, membership_type VARCHAR(10) NULL, membership_fee INT NULL, membership_till DATE NULL, member_id INT NULL, class_id INT NULL)

Payment
Attributes: (payment_id INT PRIMARY KEY, payment_amount INT NULL, payment_mode VARCHAR(20) NULL, member_id INT NULL, membership_id INT NULL)

Workers
Attributes: (worker_id INT PRIMARY KEY, phone_no VARCHAR(50) NULL, worker_type VARCHAR(20) NULL)

Promotional_Activities
Attributes: (promotional_id INT PRIMARY KEY, offer_id INT NULL, offer_type VARCHAR(20) NULL, files_uploaded VARCHAR(30) NULL)
Equipment
Attributes: (equipment_id INT PRIMARY KEY, equipment_type VARCHAR(20) NULL, maintainance_date DATE NULL, class_id INT)

Class
Attributes: (class_id INT PRIMARY KEY, class_name VARCHAR(50) NULL, trainer_id INT NULL)

Workout_Plans
Attributes: (workout_id INT PRIMARY KEY, class_id INT NULL, trainer_id INT NULL)

Trainer
Attributes: (trainer_id INT PRIMARY KEY, trainer_name VARCHAR(50) NULL, trainer_specialization VARCHAR(50) NULL, trainer_phone_no VARCHAR(50) NULL, trainer_email VARCHAR(50) NULL)

Feedback
Attributes: (feedback_id INT PRIMARY KEY, member_id INT, membership_id INT, comment VARCHAR(50) NULL)

Relationships:

Membership and Member:

Cardinality: One membership can have many member.

Participation: Every member must have a membership.

Membership and Payment:

Cardinality: One membership may have multiple payments associated with it.

Participation: Each payment corresponds to a single membership.

Member and Feedback:

Cardinality: One member can provide multiple feedback entries.

Participation: Each feedback entry is provided by a single member.

Class and Trainer:

Cardinality: One class is conducted by a single trainer.

Participation:Each trainer can conduct multiple class

Equipment and Class:

Cardinality: Equipment can be used in specific classes.

Participation:Each class may utilize multiple pieces of equipment.

Trainer and Feedback:

Cardinality: A trainer may receive feedback from multiple members.

Participation: Each feedback entry is associated with a single trainer.

Trainer and Class:

Cardinality: Each class is conducted by a single trainer.

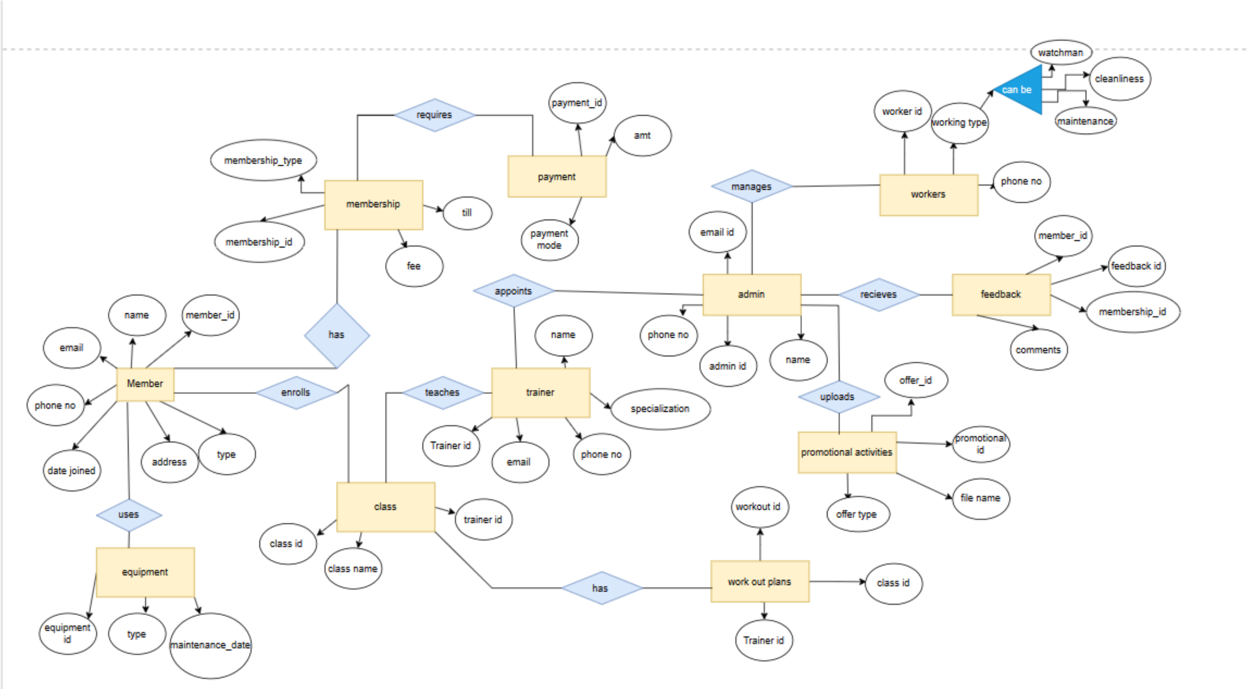
Participation: Each trainer can conduct multiple classes.

Worker and Maintenance:

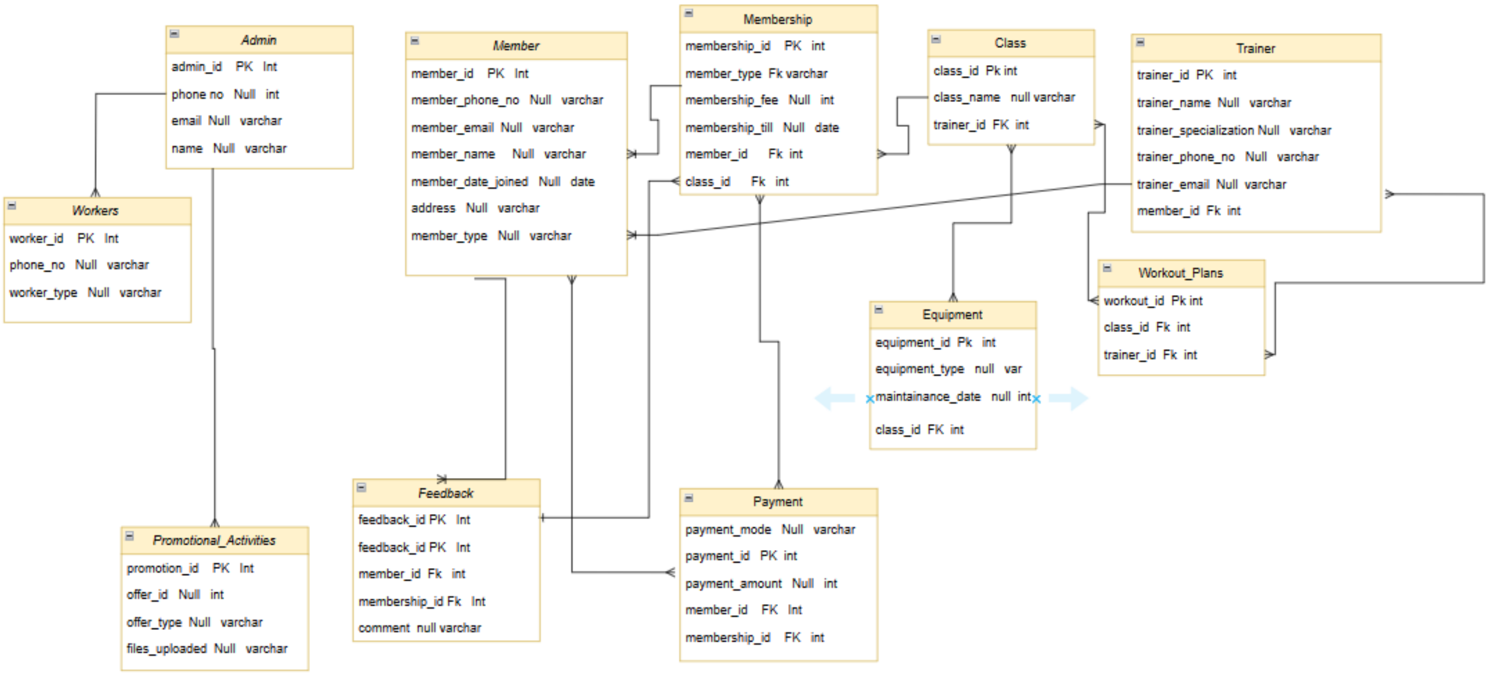
Cardinality: Workers are responsible for the maintenance of gym equipment.

Participation: Each piece of equipment may require maintenance by multiple workers.

III. Entity Relationship Diagram



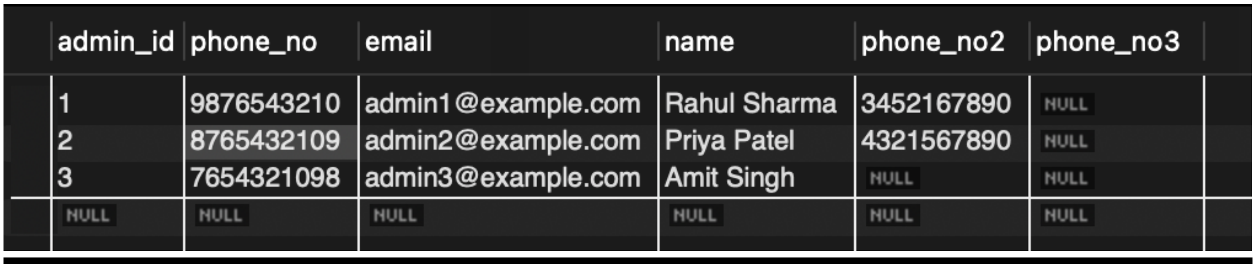
IV. Relational Model



V. Normalization

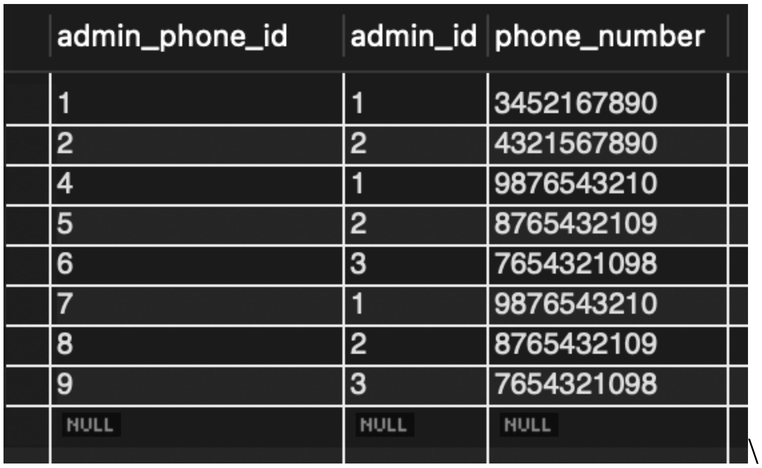
Perform normalization (1NF, 2NF, 3NF, BCNF) as applicable for the entire database.

Admin Table before 1NF and 2NF



admin_id	phone_no	email	name	phone_no2	phone_no3
1	9876543210	admin1@example.com	Rahul Sharma	3452167890	NULL
2	8765432109	admin2@example.com	Priya Patel	4321567890	NULL
3	7654321098	admin3@example.com	Amit Singh	NULL	NULL
NULL	NULL	NULL	NULL	NULL	NULL

```
CREATE TABLE Admin_PhoneNumbers (
   admin_phone_id INT AUTO_INCREMENT PRIMARY KEY,
   admin_id INT,
   phone_number VARCHAR(50),
   FOREIGN KEY (admin_id) REFERENCES Admin(admin_id)
);

INSERT INTO Admin_PhoneNumbers (admin_id, phone_number)
SELECT admin_id, phone_no2 FROM Admin WHERE phone_no2 IS NOT
NULL;

INSERT INTO Admin_PhoneNumbers (admin_id, phone_number)
SELECT admin_id, phone_no3 FROM Admin WHERE phone_no3 IS NOT
NULL;

INSERT INTO Admin_PhoneNumbers (admin_id, phone_number)
SELECT admin_id, phone_no FROM Admin;

ALTER TABLE Admin DROP COLUMN phone_no;

ALTER TABLE Admin DROP COLUMN phone_no2, DROP COLUMN
phone_no3;

SELECT * FROM Admin_PhoneNumbers;
```



admin_phone_id	admin_id	phone_number
1	1	3452167890
2	2	4321567890
4	1	9876543210
5	2	8765432109
6	3	7654321098
7	1	9876543210
8	2	8765432109
9	3	7654321098
NULL	NULL	NULL

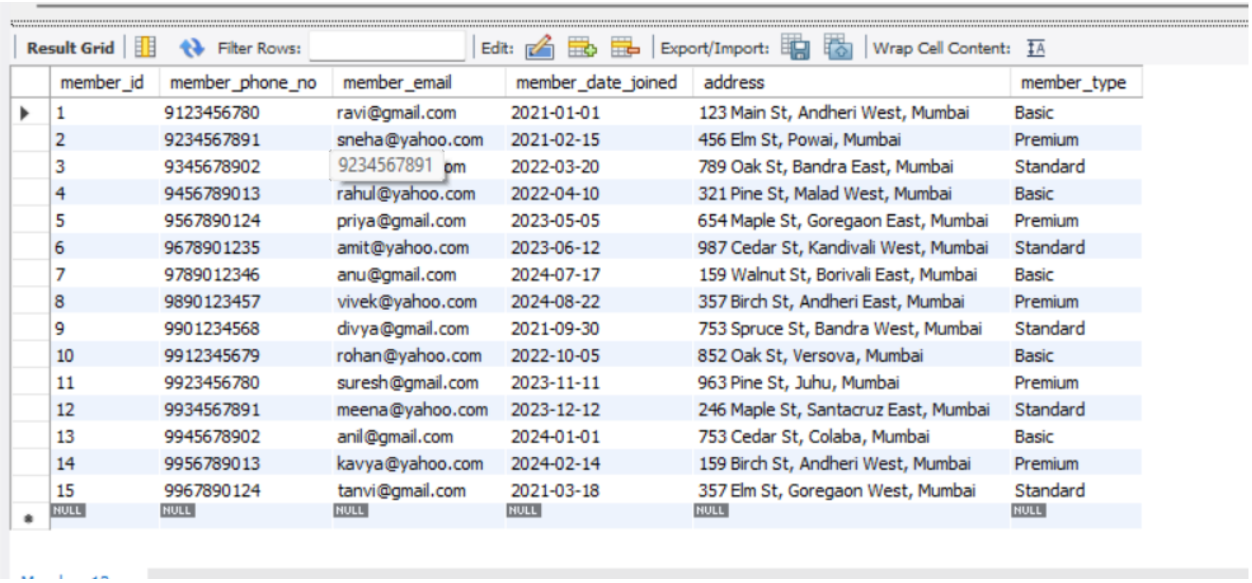
VI. SQL Queries

- **Create the tables & Populate the tables**

```
CREATE DATABASE GYM_MANAGEMENT;

USE GYM_MANAGEMENT;
```

1)

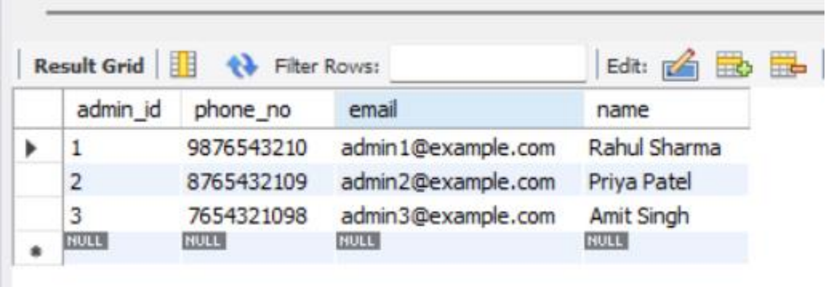
```
CREATE TABLE Member(
member_id INT PRIMARY KEY,
member_phone_no VARCHAR(50) NULL,
member_email VARCHAR(50) NULL,
member_date_joined DATE NULL,
address VARCHAR(100) NULL,
member_type VARCHAR(15) NULL
);
INSERT INTO Member VALUES
(1, '9123456780', 'ravi@gmail.com', '2021-01-01', '123 Main St, Andheri West, Mumbai',
'Basic'),
(2, '9234567891', 'sneha@yahoo.com', '2021-02-15', '456 Elm St, Powai, Mumbai',
'Premium'),
(3, '9345678902', 'neha@gmail.com', '2022-03-20', '789 Oak St, Bandra East, Mumbai',
'Standard'),
(4, '9456789013', 'rahul@yahoo.com', '2022-04-10', '321 Pine St, Malad West, Mumbai',
'Basic'),
(5, '9567890124', 'priya@gmail.com', '2023-05-05', '654 Maple St, Goregaon East, Mumbai',
'Premium'),
(6, '9678901235', 'amit@yahoo.com', '2023-06-12', '987 Cedar St, Kandivali West, Mumbai',
'Standard'),
(7, '9789012346', 'anu@gmail.com', '2024-07-17', '159 Walnut St, Borivali East, Mumbai',
'Basic'),
(8, '9890123457', 'vivek@yahoo.com', '2024-08-22', '357 Birch St, Andheri East, Mumbai',
'Premium'),
(9, '9901234568', 'divya@gmail.com', '2021-09-30', '753 Spruce St, Bandra West, Mumbai',
'Standard'),
(10, '9912345679', 'rohan@yahoo.com', '2022-10-05', '852 Oak St, Versova, Mumbai',
'Basic'),
(11, '9923456780', 'suresh@gmail.com', '2023-11-11', '963 Pine St, Juhu, Mumbai',
'Premium'),
(12, '9934567891', 'meena@yahoo.com', '2023-12-12', '246 Maple St, Santacruz East, Mumbai', 'Standard'),
(13, '9945678902', 'anil@gmail.com', '2024-01-01', '753 Cedar St, Colaba, Mumbai', 'Basic'),
(14, '9956789013', 'kavya@yahoo.com', '2024-02-14', '159 Birch St, Andheri West, Mumbai', 'Premium'),
(15, '9967890124', 'tanvi@gmail.com', '2021-03-18', '357 Elm St, Goregaon West, Mumbai', 'Standard');
```



member_id	member_phone_no	member_email	member_date_joined	address	member_type
1	9123456780	ravi@gmail.com	2021-01-01	123 Main St, Andheri West, Mumbai	Basic
2	9234567891	sneha@yahoo.com	2021-02-15	456 Elm St, Powai, Mumbai	Premium
3	9345678902	9234567891 om	2022-03-20	789 Oak St, Bandra East, Mumbai	Standard
4	9456789013	rahul@yahoo.com	2022-04-10	321 Pine St, Malad West, Mumbai	Basic
5	9567890124	priya@gmail.com	2023-05-05	654 Maple St, Goregaon East, Mumbai	Premium
6	9678901235	amit@yahoo.com	2023-06-12	987 Cedar St, Kandivali West, Mumbai	Standard
7	9789012346	anu@gmail.com	2024-07-17	159 Walnut St, Borivali East, Mumbai	Basic
8	9890123457	vivek@yahoo.com	2024-08-22	357 Birch St, Andheri East, Mumbai	Premium
9	9901234568	divya@gmail.com	2021-09-30	753 Spruce St, Bandra West, Mumbai	Standard
10	9912345679	rohan@yahoo.com	2022-10-05	852 Oak St, Versova, Mumbai	Basic
11	9923456780	suresh@gmail.com	2023-11-11	963 Pine St, Juhu, Mumbai	Premium
12	9934567891	meena@yahoo.com	2023-12-12	246 Maple St, Santacruz East, Mumbai	Standard
13	9945678902	anil@gmail.com	2024-01-01	753 Cedar St, Colaba, Mumbai	Basic
14	9956789013	kavya@yahoo.com	2024-02-14	159 Birch St, Andheri West, Mumbai	Premium
15	9967890124	tanvi@gmail.com	2021-03-18	357 Elm St, Goregaon West, Mumbai	Standard
NULL	NULL	NULL	NULL	NULL	NULL

2)

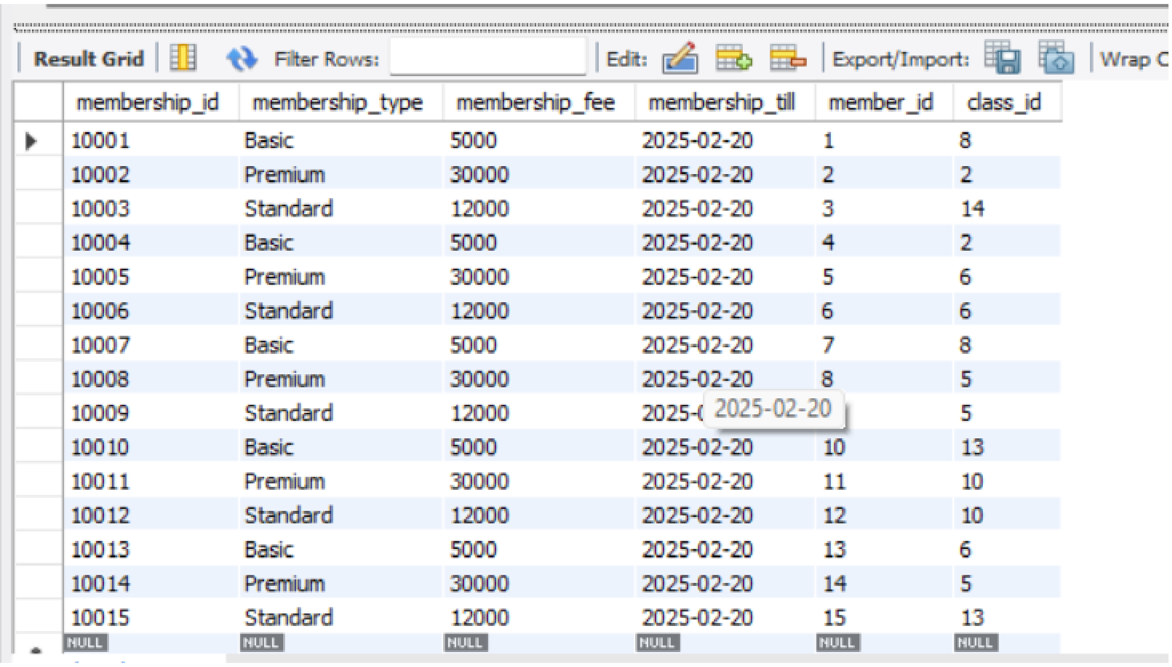
```
CREATE TABLE Admin(
admin_id INT PRIMARY KEY,
phone_no VARCHAR(50) NULL,
email VARCHAR(25) NULL,
name VARCHAR(20) NULL
);
INSERT INTO Admin VALUES
(1, '9876543210', 'admin1@example.com', 'Rahul Sharma'),
(2, '8765432109', 'admin2@example.com', 'Priya Patel'),
(3, '7654321098', 'admin3@example.com', 'Amit Singh');
```



admin_id	phone_no	email	name
1	9876543210	admin1@example.com	Rahul Sharma
2	8765432109	admin2@example.com	Priya Patel
3	7654321098	admin3@example.com	Amit Singh
NULL	NULL	NULL	NULL

3)

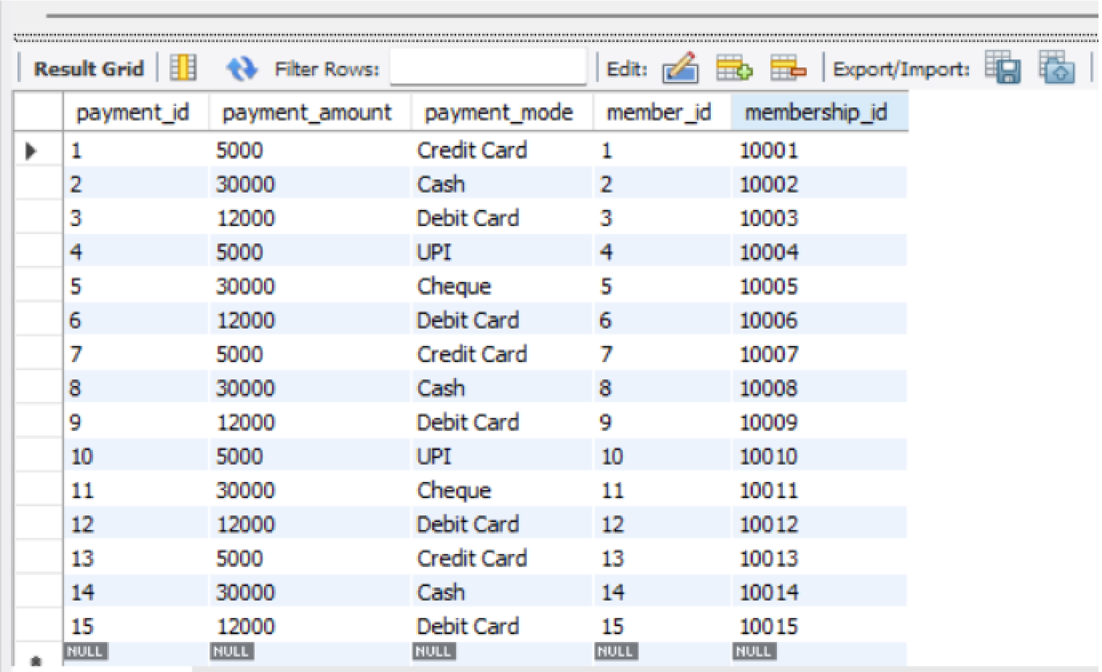
```
CREATE TABLE Membership(
membership_id INT PRIMARY KEY,
membership_type VARCHAR(10) NULL,
membership_fee INT NULL,
membership_till DATE NULL,
member_id INT NULL,
class_id INT NULL
);
INSERT INTO Membership VALUES
(10001, 'Basic', 5000, '2025-02-20', 1, 8),
(10002, 'Premium', 30000, '2025-02-20', 2, 2),
(10003, 'Standard', 12000, '2025-02-20', 3, 14),
(10004, 'Basic', 5000, '2025-02-20', 4, 2),
(10005, 'Premium', 30000, '2025-02-20', 5, 6),
(10006, 'Standard', 12000, '2025-02-20', 6, 6),
(10007, 'Basic', 5000, '2025-02-20', 7, 8),
(10008, 'Premium', 30000, '2025-02-20', 8, 5),
(10009, 'Standard', 12000, '2025-02-20', 9, 5),
(10010, 'Basic', 5000, '2025-02-20', 10, 13),
(10011, 'Premium', 30000, '2025-02-20', 11, 10),
(10012, 'Standard', 12000, '2025-02-20', 12, 10),
(10013, 'Basic', 5000, '2025-02-20', 13, 6),
(10014, 'Premium', 30000, '2025-02-20', 14, 5),
(10015, 'Standard', 12000, '2025-02-20', 15, 13);
```



membership_id	membership_type	membership_fee	membership_till	member_id	class_id
10001	Basic	5000	2025-02-20	1	8
10002	Premium	30000	2025-02-20	2	2
10003	Standard	12000	2025-02-20	3	14
10004	Basic	5000	2025-02-20	4	2
10005	Premium	30000	2025-02-20	5	6
10006	Standard	12000	2025-02-20	6	6
10007	Basic	5000	2025-02-20	7	8
10008	Premium	30000	2025-02-20	8	5
10009	Standard	12000	2025-(2025-02-20		5
10010	Basic	5000	2025-02-20	10	13
10011	Premium	30000	2025-02-20	11	10
10012	Standard	12000	2025-02-20	12	10
10013	Basic	5000	2025-02-20	13	6
10014	Premium	30000	2025-02-20	14	5
10015	Standard	12000	2025-02-20	15	13
NULL	NULL	NULL	NULL	NULL	NULL

4)

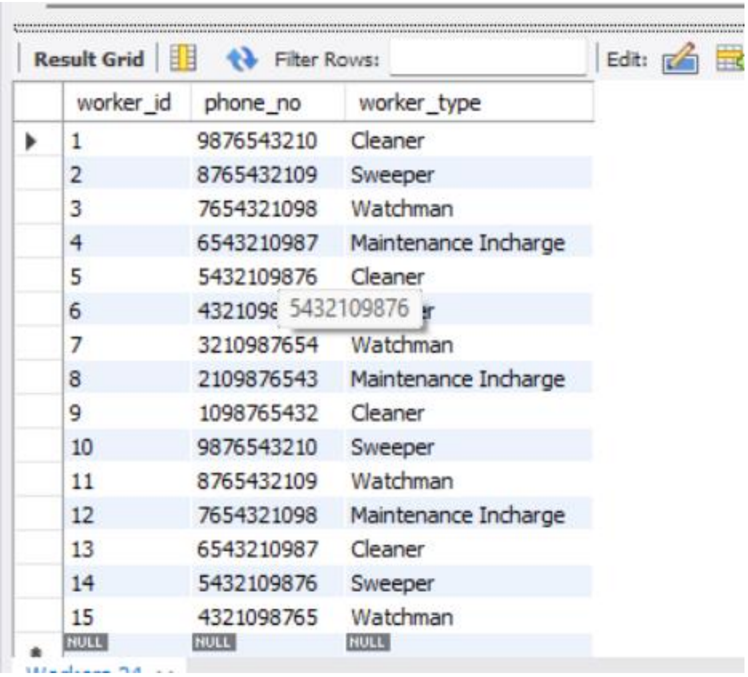
```
CREATE TABLE Payment(
payment_id INT PRIMARY KEY,
payment_amount INT NULL,
payment_mode VARCHAR(20) NULL,
member_id INT NULL,
membership_id INT NULL
);
INSERT INTO Payment VALUES
(1, 5000, 'Credit Card', 1, 10001),
(2, 30000, 'Cash', 2, 10002),
(3, 12000, 'Debit Card', 3, 10003),
(4, 5000, 'UPI', 4, 10004),
(5, 30000, 'Cheque', 5, 10005),
(6, 12000, 'Debit Card', 6, 10006),
(7, 5000, 'Credit Card', 7, 10007),
(8, 30000, 'Cash', 8, 10008),
(9, 12000, 'Debit Card', 9, 10009),
(10, 5000, 'UPI', 10, 10010),
(11, 30000, 'Cheque', 11, 10011),
(12, 12000, 'Debit Card', 12, 10012),
(13, 5000, 'Credit Card', 13, 10013),
(14, 30000, 'Cash', 14, 10014),
(15, 12000, 'Debit Card', 15, 10015);
```



	payment_id	payment_amount	payment_mode	member_id	membership_id
▸	1	5000	Credit Card	1	10001
	2	30000	Cash	2	10002
	3	12000	Debit Card	3	10003
	4	5000	UPI	4	10004
	5	30000	Cheque	5	10005
	6	12000	Debit Card	6	10006
	7	5000	Credit Card	7	10007
	8	30000	Cash	8	10008
	9	12000	Debit Card	9	10009
	10	5000	UPI	10	10010
	11	30000	Cheque	11	10011
	12	12000	Debit Card	12	10012
	13	5000	Credit Card	13	10013
	14	30000	Cash	14	10014
	15	12000	Debit Card	15	10015
*	NULL	NULL	NULL	NULL	NULL

5)

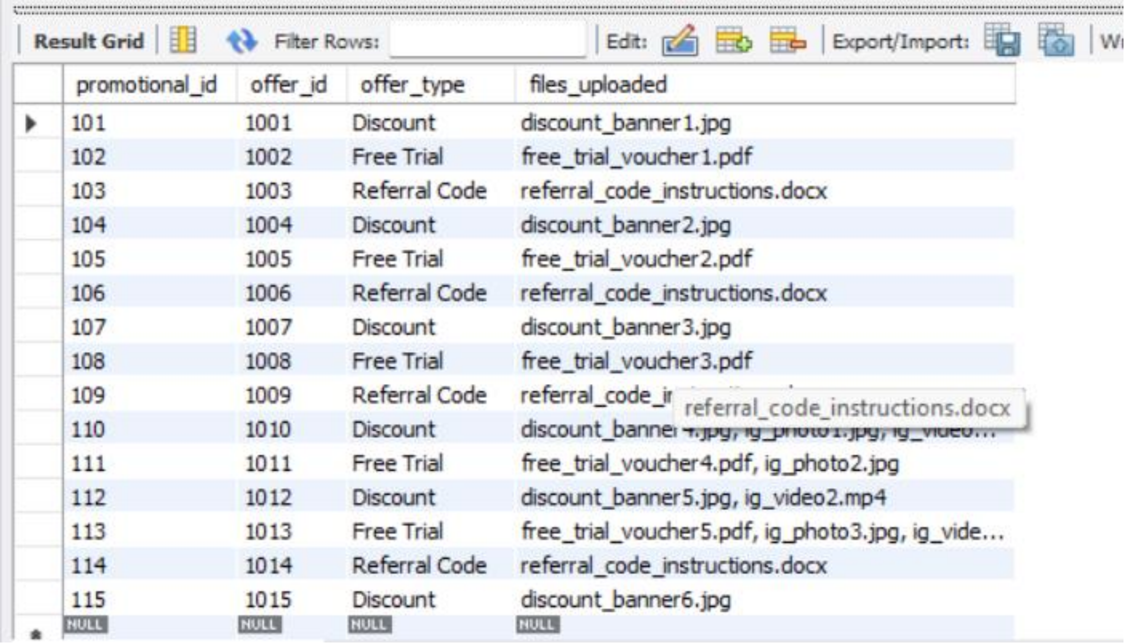
```
CREATE TABLE Workers(
worker_id INT PRIMARY KEY,
phone_no VARCHAR(50) NULL,
worker_type VARCHAR(20) NULL
);
INSERT INTO Workers VALUES
(1, '9876543210', 'Cleaner'),
(2, '8765432109', 'Sweeper'),
(3, '7654321098', 'Watchman'),
(4, '6543210987', 'Maintenance Incharge'),
(5, '5432109876', 'Cleaner'),
(6, '4321098765', 'Sweeper'),
(7, '3210987654', 'Watchman'),
(8, '2109876543', 'Maintenance Incharge'),
(9, '1098765432', 'Cleaner'),
(10, '9876543210', 'Sweeper'),
(11, '8765432109', 'Watchman'),
(12, '7654321098', 'Maintenance Incharge'),
(13, '6543210987', 'Cleaner'),
(14, '5432109876', 'Sweeper'),
(15, '4321098765', 'Watchman');
```



worker_id	phone_no	worker_type
1	9876543210	Cleaner
2	8765432109	Sweeper
3	7654321098	Watchman
4	6543210987	Maintenance Incharge
5	5432109876	Cleaner
6	4321098 5432109876	r
7	3210987654	Watchman
8	2109876543	Maintenance Incharge
9	1098765432	Cleaner
10	9876543210	Sweeper
11	8765432109	Watchman
12	7654321098	Maintenance Incharge
13	6543210987	Cleaner
14	5432109876	Sweeper
15	4321098765	Watchman
NULL	NULL	NULL

6)

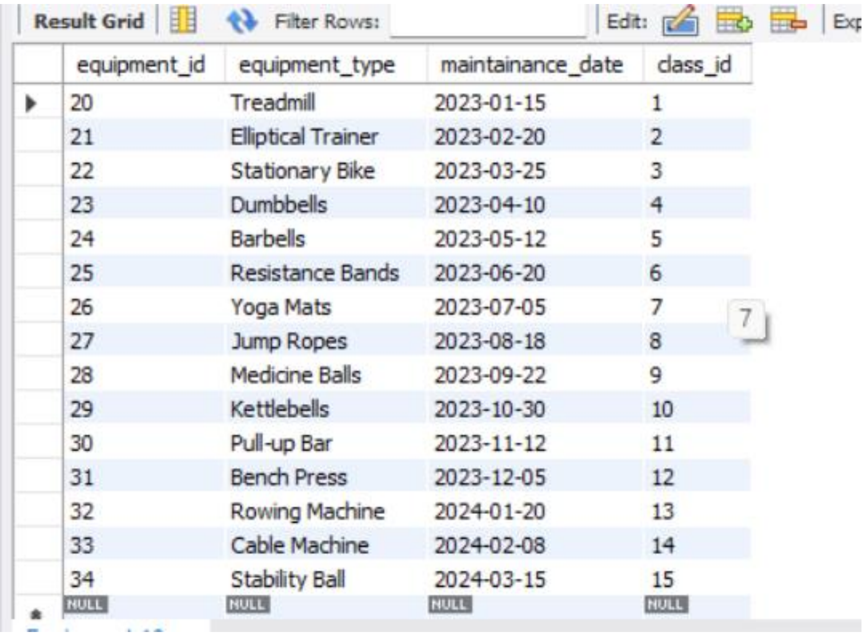
```
CREATE TABLE Promotional_Activities(
promotional_id INT PRIMARY KEY,
offer_id INT NULL,
offer_type VARCHAR(20) NULL,
files_uploaded VARCHAR(60) NULL
);
INSERT INTO Promotional_Activities VALUES
(101, 1001, 'Discount', 'discount_banner1.jpg'),
(102, 1002, 'Free Trial', 'free_trial_voucher1.pdf'),
(103, 1003, 'Referral Code', 'referral_code_instructions.docx'),
(104, 1004, 'Discount', 'discount_banner2.jpg'),
(105, 1005, 'Free Trial', 'free_trial_voucher2.pdf'),
(106, 1006, 'Referral Code', 'referral_code_instructions.docx'),
(107, 1007, 'Discount', 'discount_banner3.jpg'),
(108, 1008, 'Free Trial', 'free_trial_voucher3.pdf'),
(109, 1009, 'Referral Code', 'referral_code_instructions.docx'),
(110, 1010, 'Discount', 'discount_banner4.jpg, ig_photo1.jpg, ig_video1.mp4'),
(111, 1011, 'Free Trial', 'free_trial_voucher4.pdf, ig_photo2.jpg'),
(112, 1012, 'Discount', 'discount_banner5.jpg, ig_video2.mp4'),
(113, 1013, 'Free Trial', 'free_trial_voucher5.pdf, ig_photo3.jpg, ig_video3.mp4'),
(114, 1014, 'Referral Code', 'referral_code_instructions.docx'),
(115, 1015, 'Discount', 'discount_banner6.jpg');
```



	promotional_id	offer_id	offer_type	files_uploaded
▶	101	1001	Discount	discount_banner1.jpg
	102	1002	Free Trial	free_trial_voucher1.pdf
	103	1003	Referral Code	referral_code_instructions.docx
	104	1004	Discount	discount_banner2.jpg
	105	1005	Free Trial	free_trial_voucher2.pdf
	106	1006	Referral Code	referral_code_instructions.docx
	107	1007	Discount	discount_banner3.jpg
	108	1008	Free Trial	free_trial_voucher3.pdf
	109	1009	Referral Code	referral_code_instructions.docx
	110	1010	Discount	discount_banner4.jpg, ig_photo1.jpg, ig_video…
	111	1011	Free Trial	free_trial_voucher4.pdf, ig_photo2.jpg
	112	1012	Discount	discount_banner5.jpg, ig_video2.mp4
	113	1013	Free Trial	free_trial_voucher5.pdf, ig_photo3.jpg, ig_vide…
	114	1014	Referral Code	referral_code_instructions.docx
	115	1015	Discount	discount_banner6.jpg
*	NULL	NULL	NULL	NULL

7)

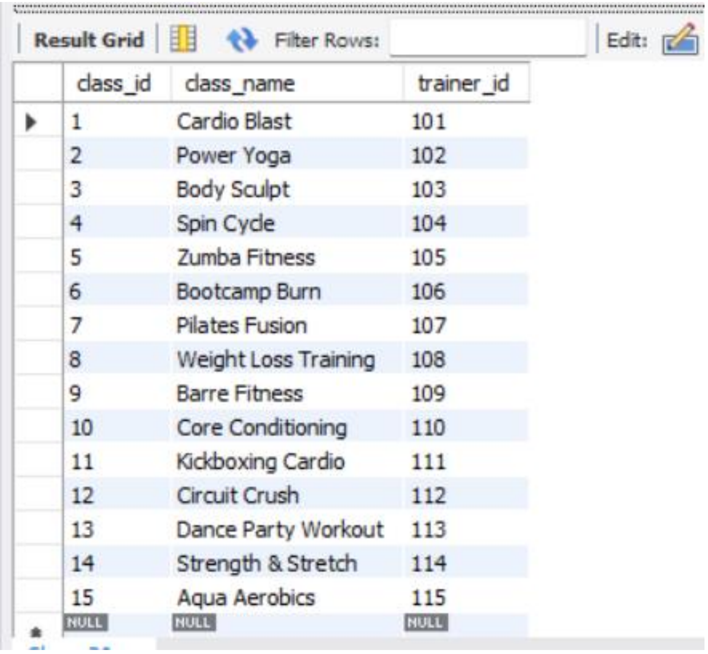
```
CREATE TABLE Equipment(
equipment_id INT PRIMARY KEY,
equipment_type VARCHAR(20) NULL,
maintainance_date DATE NULL,
class_id INT
);
INSERT INTO Equipment VALUES
(20, 'Treadmill', '2023-01-15', 1),
(21, 'Elliptical Trainer', '2023-02-20', 2),
(22, 'Stationary Bike', '2023-03-25', 3),
(23, 'Dumbbells', '2023-04-10', 4),
(24, 'Barbells', '2023-05-12', 5),
(25, 'Resistance Bands', '2023-06-20', 6),
(26, 'Yoga Mats', '2023-07-05', 7),
(27, 'Jump Ropes', '2023-08-18', 8),
(28, 'Medicine Balls', '2023-09-22', 9),
(29, 'Kettlebells', '2023-10-30', 10),
(30, 'Pull-up Bar', '2023-11-12', 11),
(31, 'Bench Press', '2023-12-05', 12),
(32, 'Rowing Machine', '2024-01-20', 13),
(33, 'Cable Machine', '2024-02-08', 14),
(34, 'Stability Ball', '2024-03-15', 15);
```



equipment_id	equipment_type	maintainance_date	class_id
20	Treadmill	2023-01-15	1
21	Elliptical Trainer	2023-02-20	2
22	Stationary Bike	2023-03-25	3
23	Dumbbells	2023-04-10	4
24	Barbells	2023-05-12	5
25	Resistance Bands	2023-06-20	6
26	Yoga Mats	2023-07-05	7
27	Jump Ropes	2023-08-18	8
28	Medicine Balls	2023-09-22	9
29	Kettlebells	2023-10-30	10
30	Pull-up Bar	2023-11-12	11
31	Bench Press	2023-12-05	12
32	Rowing Machine	2024-01-20	13
33	Cable Machine	2024-02-08	14
34	Stability Ball	2024-03-15	15
NULL	NULL	NULL	NULL

8)

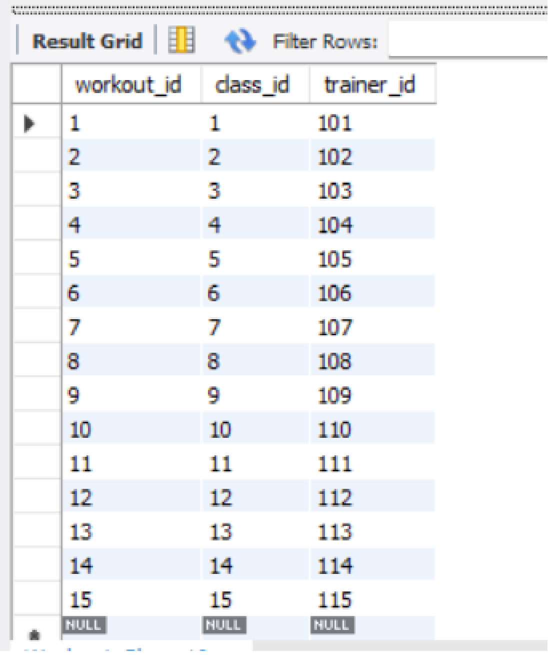
```
CREATE TABLE Class(
class_id INT PRIMARY KEY,
class_name VARCHAR(50) NULL,
trainer_id INT NULL
);
INSERT INTO Class VALUES
(1, 'Cardio Blast', 101),
(2, 'Power Yoga', 102),
(3, 'Body Sculpt', 103),
(4, 'Spin Cycle', 104),
(5, 'Zumba Fitness', 105),
(6, 'Bootcamp Burn', 106),
(7, 'Pilates Fusion', 107),
(8, 'Weight Loss Training', 108),
(9, 'Barre Fitness', 109),
(10, 'Core Conditioning', 110),
(11, 'Kickboxing Cardio', 111),
(12, 'Circuit Crush', 112),
(13, 'Dance Party Workout', 113),
(14, 'Strength & Stretch', 114),
(15, 'Aqua Aerobics', 115);
```



class_id	class_name	trainer_id
1	Cardio Blast	101
2	Power Yoga	102
3	Body Sculpt	103
4	Spin Cycle	104
5	Zumba Fitness	105
6	Bootcamp Burn	106
7	Pilates Fusion	107
8	Weight Loss Training	108
9	Barre Fitness	109
10	Core Conditioning	110
11	Kickboxing Cardio	111
12	Circuit Crush	112
13	Dance Party Workout	113
14	Strength & Stretch	114
15	Aqua Aerobics	115
NULL	NULL	NULL

9)

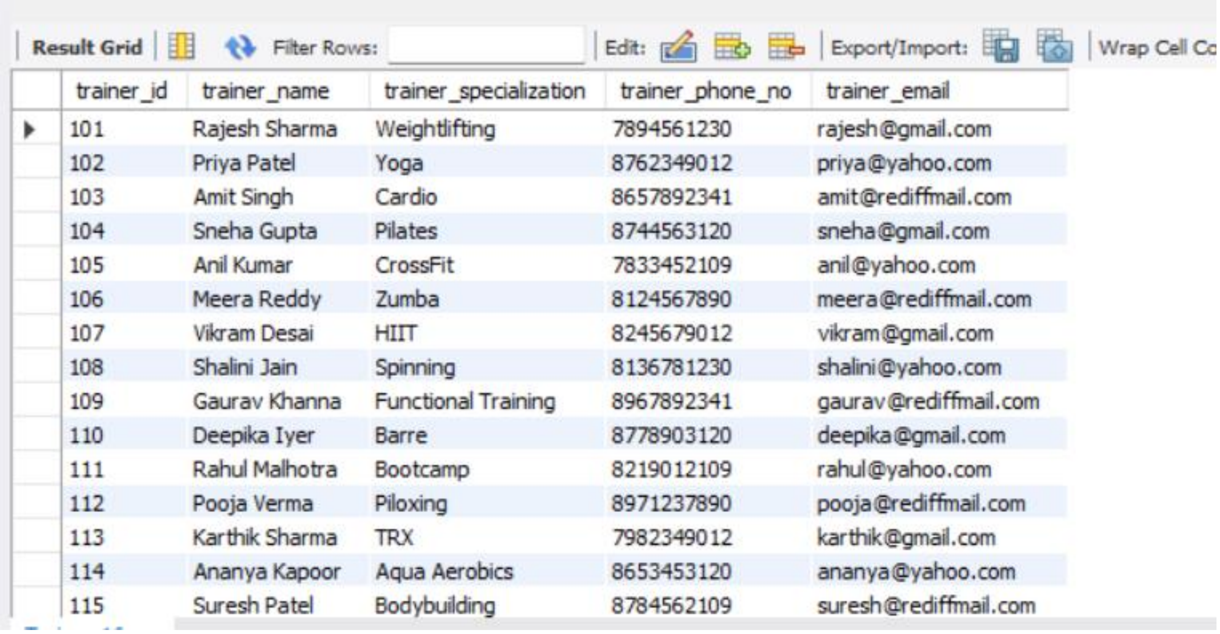
```
CREATE TABLE Workout_Plans(
workout_id INT PRIMARY KEY,
class_id INT NULL,
trainer_id INT NULL
);
INSERT INTO Workout_Plans VALUES
(1, 1, 101),
(2, 2, 102),
(3, 3, 103),
(4, 4, 104),
(5, 5, 105),
(6, 6, 106),
(7, 7, 107),
(8, 8, 108),
(9, 9, 109),
(10, 10, 110),
(11, 11, 111),
(12, 12, 112),
(13, 13, 113),
(14, 14, 114),
(15, 15, 115);
```



workout_id	class_id	trainer_id
1	1	101
2	2	102
3	3	103
4	4	104
5	5	105
6	6	106
7	7	107
8	8	108
9	9	109
10	10	110
11	11	111
12	12	112
13	13	113
14	14	114
15	15	115
NULL	NULL	NULL

10)

```
CREATE TABLE Trainer(
trainer_id INT PRIMARY KEY,
trainer_name VARCHAR(50) NULL,
trainer_specialization VARCHAR(50) NULL,
trainer_phone_no VARCHAR(50) NULL,
trainer_email VARCHAR(50) NULL
);
INSERT INTO Trainer VALUES
(101, 'Rajesh Sharma', 'Weightlifting', '7894561230', 'rajesh@gmail.com'),
(102, 'Priya Patel', 'Yoga', '8762349012', 'priya@yahoo.com'),
(103, 'Amit Singh', 'Cardio', '8657892341', 'amit@rediffmail.com'),
(104, 'Sneha Gupta', 'Pilates', '8744563120', 'sneha@gmail.com'),
(105, 'Anil Kumar', 'CrossFit', '7833452109', 'anil@yahoo.com'),
(106, 'Meera Reddy', 'Zumba', '8124567890', 'meera@rediffmail.com'),
(107, 'Vikram Desai', 'HIIT', '8245679012', 'vikram@gmail.com'),
(108, 'Shalini Jain', 'Spinning', '8136781230', 'shalini@yahoo.com'),
(109, 'Gaurav Khanna', 'Functional Training', '8967892341', 'gaurav@rediffmail.com'),
(110, 'Deepika Iyer', 'Barre', '8778903120', 'deepika@gmail.com'),
(111, 'Rahul Malhotra', 'Bootcamp', '8219012109', 'rahul@yahoo.com'),
(112, 'Pooja Verma', 'Piloxing', '8971237890', 'pooja@rediffmail.com'),
(113, 'Karthik Sharma', 'TRX', '7982349012', 'karthik@gmail.com'),
(114, 'Ananya Kapoor', 'Aqua Aerobics', '8653453120', 'ananya@yahoo.com'),
(115, 'Suresh Patel', 'Bodybuilding', '8784562109', 'suresh@rediffmail.com');
```

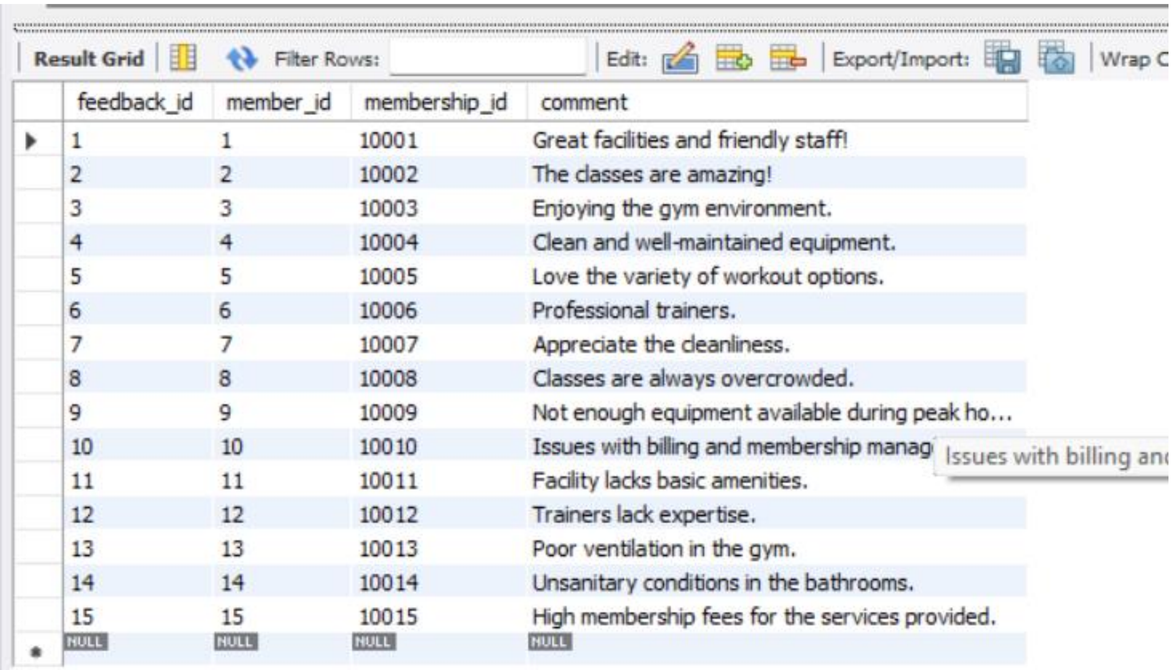


trainer_id	trainer_name	trainer_specialization	trainer_phone_no	trainer_email
101	Rajesh Sharma	Weightlifting	7894561230	rajesh@gmail.com
102	Priya Patel	Yoga	8762349012	priya@yahoo.com
103	Amit Singh	Cardio	8657892341	amit@rediffmail.com
104	Sneha Gupta	Pilates	8744563120	sneha@gmail.com
105	Anil Kumar	CrossFit	7833452109	anil@yahoo.com
106	Meera Reddy	Zumba	8124567890	meera@rediffmail.com
107	Vikram Desai	HIIT	8245679012	vikram@gmail.com
108	Shalini Jain	Spinning	8136781230	shalini@yahoo.com
109	Gaurav Khanna	Functional Training	8967892341	gaurav@rediffmail.com
110	Deepika Iyer	Barre	8778903120	deepika@gmail.com
111	Rahul Malhotra	Bootcamp	8219012109	rahul@yahoo.com
112	Pooja Verma	Piloxing	8971237890	pooja@rediffmail.com
113	Karthik Sharma	TRX	7982349012	karthik@gmail.com
114	Ananya Kapoor	Aqua Aerobics	8653453120	ananya@yahoo.com
115	Suresh Patel	Bodybuilding	8784562109	suresh@rediffmail.com

11)

```
CREATE TABLE Feedback(
feedback_id INT PRIMARY KEY,
member_id INT,
membership_id INT,
comment VARCHAR(50) NULL
);
INSERT INTO Feedback (feedback_id, member_id, membership_id, comment)VALUES
(1, 1, 10001, 'Great facilities and friendly staff!'),
(2, 2, 10002, 'The classes are amazing!'),
(3, 3, 10003, 'Enjoying the gym environment.'),
(4, 4, 10004, 'Clean and well-maintained equipment.'),
(5, 5, 10005, 'Love the variety of workout options.'),
(6, 6, 10006, 'Professional trainers.'),
(7, 7, 10007, 'Appreciate the cleanliness.'),
(8, 8, 10008, 'Classes are always overcrowded.'),
(9, 9, 10009, 'Not enough equipment available during peak hours.'),
(10, 10, 10010, 'Issues with billing and membership management.'),
(11, 11, 10011, 'Facility lacks basic amenities.'),
(12, 12, 10012, 'Trainers lack expertise.'),
(13, 13, 10013, 'Poor ventilation in the gym.'),
(14, 14, 10014, 'Unsanitary conditions in the bathrooms.'),
```

(15, 15, 10015, 'High membership fees for the services provided.');



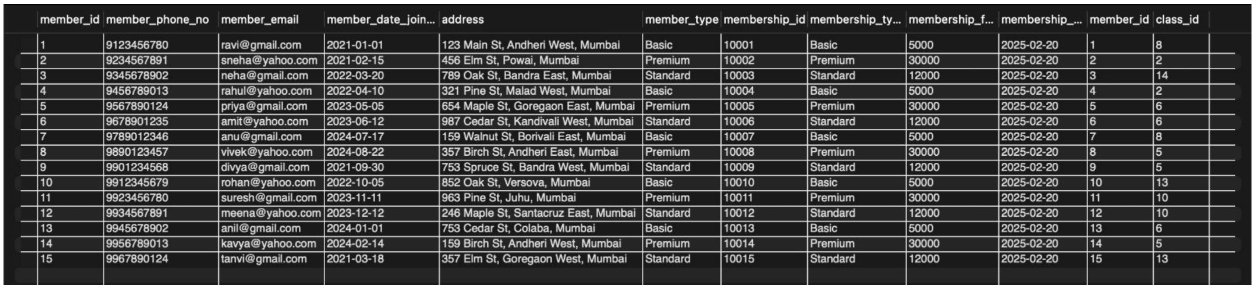
feedback_id	member_id	membership_id	comment
1	1	10001	Great facilities and friendly staff!
2	2	10002	The classes are amazing!
3	3	10003	Enjoying the gym environment.
4	4	10004	Clean and well-maintained equipment.
5	5	10005	Love the variety of workout options.
6	6	10006	Professional trainers.
7	7	10007	Appreciate the cleanliness.
8	8	10008	Classes are always overcrowded.
9	9	10009	Not enough equipment available during peak ho...
10	10	10010	Issues with billing and membership manag
11	11	10011	Facility lacks basic amenities.
12	12	10012	Trainers lack expertise.
13	13	10013	Poor ventilation in the gym.
14	14	10014	Unsanitary conditions in the bathrooms.
15	15	10015	High membership fees for the services provided.
NULL	NULL	NULL	NULL

- ## SQL queries

1)

Retrieve all members along with their membership details

SELECT Member., Membership. FROM Member JOIN Membership ON Member.member_id = Membership.member_id;

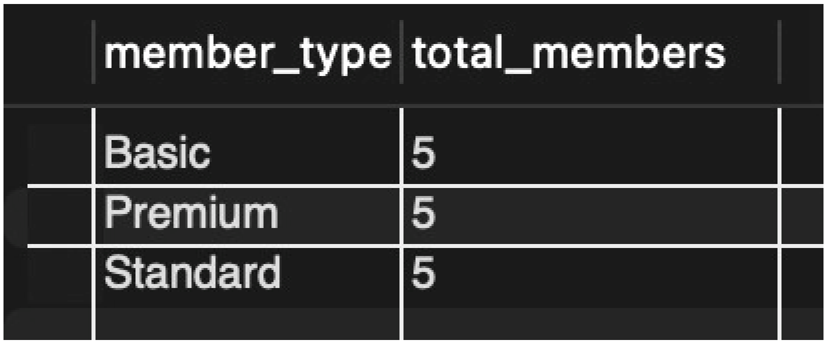


member_id	member_phone_no	member_email	member_date_join...	address	member_type	membership_id	membership_ty...	membership_f...	membership_...	member_id	class_id
1	9123456780	ravi@gmail.com	2021-01-01	123 Main St, Andheri West, Mumbai	Basic	10001	Basic	5000	2025-02-20	1	8
2	9234567891	sneha@yahoo.com	2021-02-15	456 Elm St, Powai, Mumbai	Premium	10002	Premium	30000	2025-02-20	2	2
3	9345678902	neha@gmail.com	2022-03-20	789 Oak St, Bandra East, Mumbai	Standard	10003	Standard	12000	2025-02-20	3	14
4	9456789013	rahul@yahoo.com	2022-04-10	321 Pine St, Malad West, Mumbai	Basic	10004	Basic	5000	2025-02-20	4	2
5	9567890124	priya@gmail.com	2023-05-05	654 Maple St, Goregaon East, Mumbai	Premium	10005	Premium	30000	2025-02-20	5	6
6	9678901235	amit@yahoo.com	2023-06-12	987 Cedar St, Kandivali West, Mumbai	Standard	10006	Standard	12000	2025-02-20	6	6
7	9789012346	anu@gmail.com	2024-07-17	159 Walnut St, Borivali East, Mumbai	Basic	10007	Basic	5000	2025-02-20	7	8
8	9890123457	vivek@yahoo.com	2024-08-22	357 Birch St, Andheri East, Mumbai	Premium	10008	Premium	30000	2025-02-20	8	5
9	9901234568	divya@gmail.com	2021-09-30	753 Spruce St, Bandra West, Mumbai	Standard	10009	Standard	12000	2025-02-20	9	5
10	9912345679	rohan@yahoo.com	2022-10-05	852 Oak St, Versova, Mumbai	Basic	10010	Basic	5000	2025-02-20	10	13
11	9923456780	suresh@gmail.com	2023-11-11	963 Pine St, Juhu, Mumbai	Premium	10011	Premium	30000	2025-02-20	11	10
12	9934567891	meena@yahoo.com	2023-12-12	246 Maple St, Santacruz East, Mumbai	Standard	10012	Standard	12000	2025-02-20	12	10
13	9945678902	anil@gmail.com	2024-01-01	753 Cedar St, Colaba, Mumbai	Basic	10013	Basic	5000	2025-02-20	13	6
14	9956789013	kavya@yahoo.com	2024-02-14	159 Birch St, Andheri West, Mumbai	Premium	10014	Premium	30000	2025-02-20	14	5
15	9967890124	tanvi@gmail.com	2021-03-18	357 Elm St, Goregaon West, Mumbai	Standard	10015	Standard	12000	2025-02-20	15	13

2)

Get the total count of members in each membership type

SELECT member_type, COUNT(*) AS total_members FROM Member GROUP BY member_type;

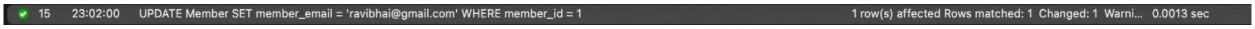


member_type	total_members
Basic	5
Premium	5
Standard	5

3)

change email address using member id

UPDATE Member SET member_email = 'ravibhai@gmail.com' WHERE member_id = 1;

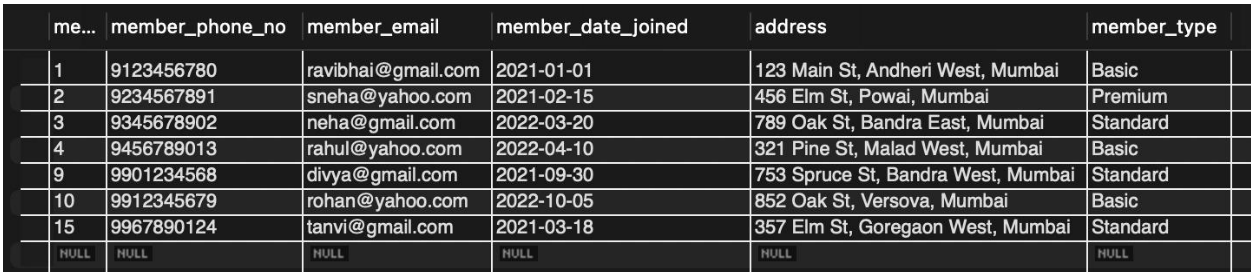


```
✓ 15    23:02:00    UPDATE Member SET member_email = 'ravibhai@gmail.com' WHERE member_id = 1    1 row(s) affected Rows matched: 1  Changed: 1  Warni...  0.0013 sec
```

4)

retrieve members that joined between two certain dates

SELECT * FROM Member WHERE member_date_joined BETWEEN '2020-01-25 00:00:00' AND '2022-12-25 23:59:59';

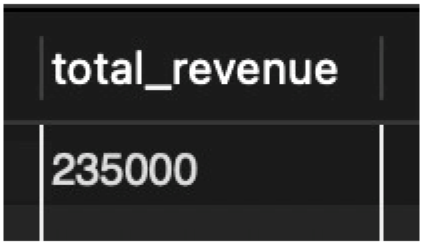


me...	member_phone_no	member_email	member_date_joined	address	member_type
1	9123456780	ravibhai@gmail.com	2021-01-01	123 Main St, Andheri West, Mumbai	Basic
2	9234567891	sneha@yahoo.com	2021-02-15	456 Elm St, Powai, Mumbai	Premium
3	9345678902	neha@gmail.com	2022-03-20	789 Oak St, Bandra East, Mumbai	Standard
4	9456789013	rahul@yahoo.com	2022-04-10	321 Pine St, Malad West, Mumbai	Basic
9	9901234568	divya@gmail.com	2021-09-30	753 Spruce St, Bandra West, Mumbai	Standard
10	9912345679	rohan@yahoo.com	2022-10-05	852 Oak St, Versova, Mumbai	Basic
15	9967890124	tanvi@gmail.com	2021-03-18	357 Elm St, Goregaon West, Mumbai	Standard
NULL	NULL	NULL	NULL	NULL	NULL

5)

Calculate Total Revenue from Membership Fees

SELECT SUM(membership_fee) AS total_revenue FROM Membership;

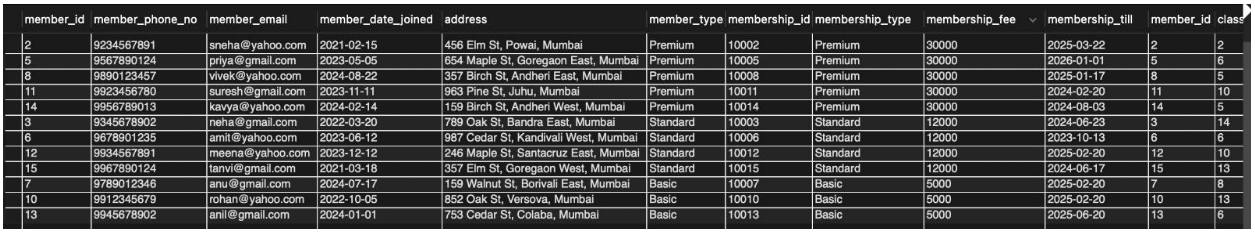


total_revenue
235000

6)

get membership details of members with active membership

```
SELECT Member., Membership. FROM Member JOIN Membership ON
Member.member_id = Membership.member_id WHERE Membership.membership_till
>= '2023-03-11 23:59:59';
```

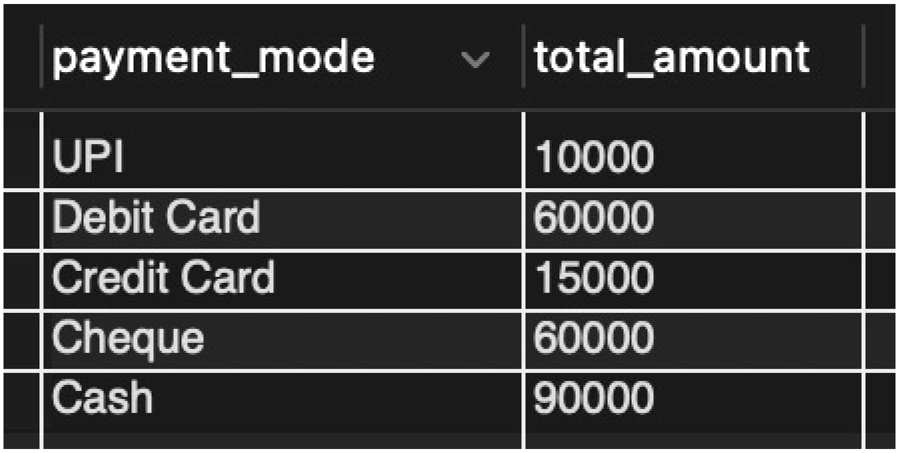


member_id	member_phone_no	member_email	member_date_joined	address	member_type	membership_id	membership_type	membership_fee	membership_till	member_id	class
2	9234567891	sneha@yahoo.com	2021-02-15	456 Elm St, Powai, Mumbai	Premium	10002	Premium	30000	2025-03-22	2	2
5	9567890124	priya@gmail.com	2023-05-05	654 Maple St, Goregaon East, Mumbai	Premium	10005	Premium	30000	2026-01-01	5	6
8	9890123457	vivek@yahoo.com	2024-08-22	357 Birch St, Andheri East, Mumbai	Premium	10008	Premium	30000	2025-01-17	8	5
11	9923456780	suresh@gmail.com	2023-11-11	963 Pine St, Juhu, Mumbai	Premium	10011	Premium	30000	2024-02-20	11	10
14	9956789013	kavya@yahoo.com	2024-02-14	159 Birch St, Andheri West, Mumbai	Premium	10014	Premium	30000	2024-08-03	14	5
3	9345678902	neha@gmail.com	2022-03-20	789 Oak St, Bandra East, Mumbai	Standard	10003	Standard	12000	2024-06-23	3	14
6	9678901235	amit@yahoo.com	2023-06-12	987 Cedar St, Kandivali West, Mumbai	Standard	10006	Standard	12000	2023-10-13	6	6
12	9934567891	meena@yahoo.com	2023-12-12	246 Maple St, Santacruz East, Mumbai	Standard	10012	Standard	12000	2025-02-20	12	10
15	9967890124	tanvi@gmail.com	2021-03-18	357 Elm St, Goregaon West, Mumbai	Standard	10015	Standard	12000	2024-06-17	15	13
7	9789012346	anu@gmail.com	2024-07-17	159 Walnut St, Borivali East, Mumbai	Basic	10007	Basic	5000	2025-02-20	7	8
10	9912345679	rohan@yahoo.com	2022-10-05	852 Oak St, Versova, Mumbai	Basic	10010	Basic	5000	2025-02-20	10	13
13	9945678902	anil@gmail.com	2024-01-01	753 Cedar St, Colaba, Mumbai	Basic	10013	Basic	5000	2025-06-20	13	6

7)

Get the total amount of payments made through each payment mode.

```
SELECT payment_mode, SUM(payment_amount) AS total_amount FROM Payment
GROUP BY payment_mode;
```

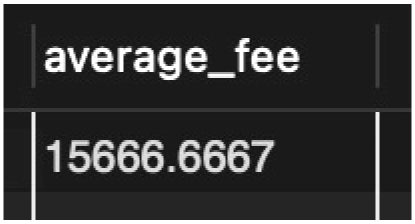


payment_mode	total_amount
UPI	10000
Debit Card	60000
Credit Card	15000
Cheque	60000
Cash	90000

8)

Find the average membership fee paid by members.

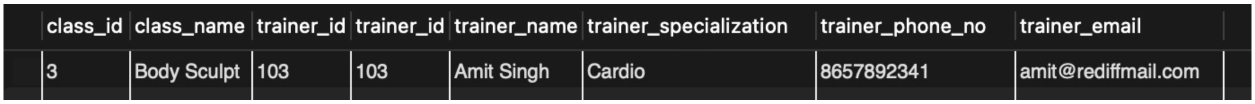
```
SELECT AVG(membership_fee) AS average_fee FROM Membership;
```



average_fee
15666.6667

9)

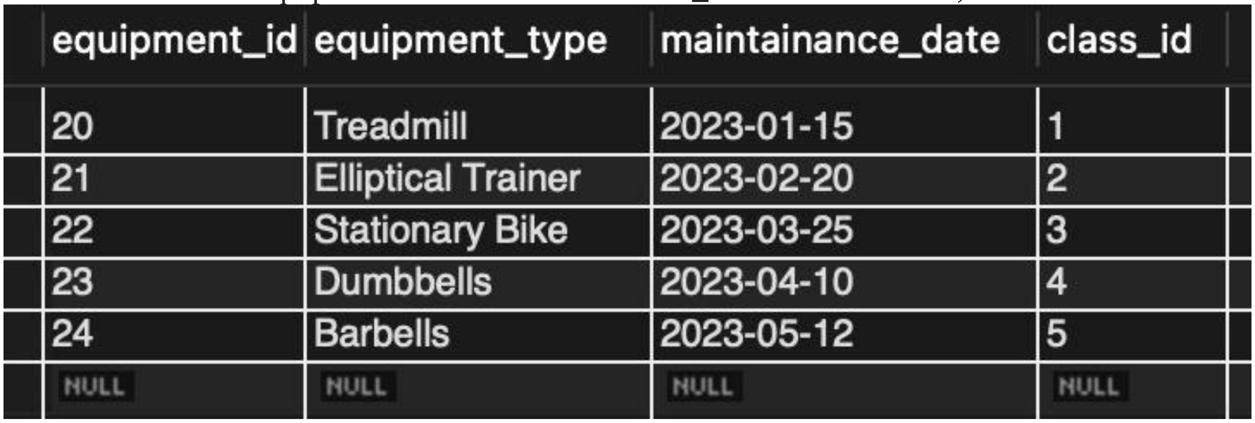
Identify the classes where the trainer specializes in Cardio.
SELECT * FROM Class c JOIN Trainer t ON c.trainer_id = t.trainer_id WHERE t.trainer_specialization = 'Cardio';



class_id	class_name	trainer_id	trainer_id	trainer_name	trainer_specialization	trainer_phone_no	trainer_email
3	Body Sculpt	103	103	Amit Singh	Cardio	8657892341	amit@rediffmail.com

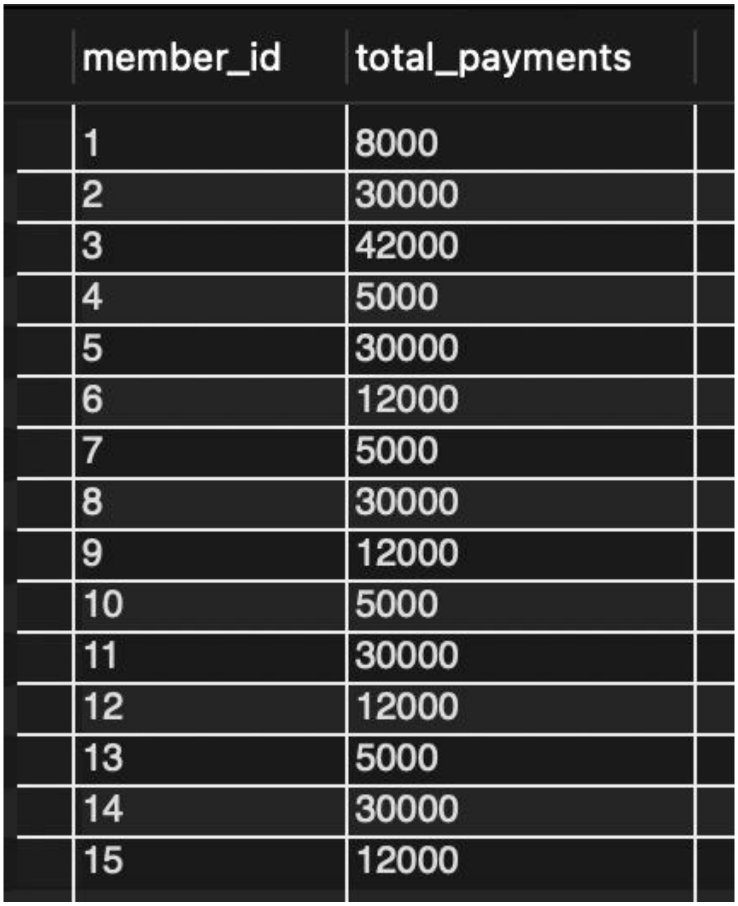
10)

List all equipment that requires maintenance before a certain date.
SELECT * FROM Equipment WHERE maintainance_date < '2023-06-01';



equipment_id	equipment_type	maintainance_date	class_id
20	Treadmill	2023-01-15	1
21	Elliptical Trainer	2023-02-20	2
22	Stationary Bike	2023-03-25	3
23	Dumbbells	2023-04-10	4
24	Barbells	2023-05-12	5
NULL	NULL	NULL	NULL

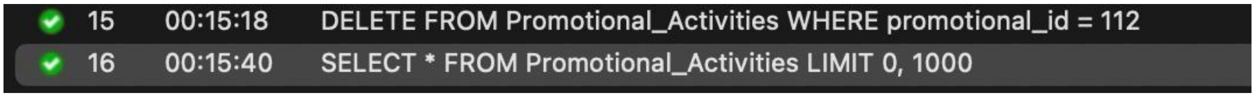
11)
Find the total amount of payments made by each member.
SELECT member_id, SUM(payment_amount) AS total_payments FROM Payment GROUP BY member_id;



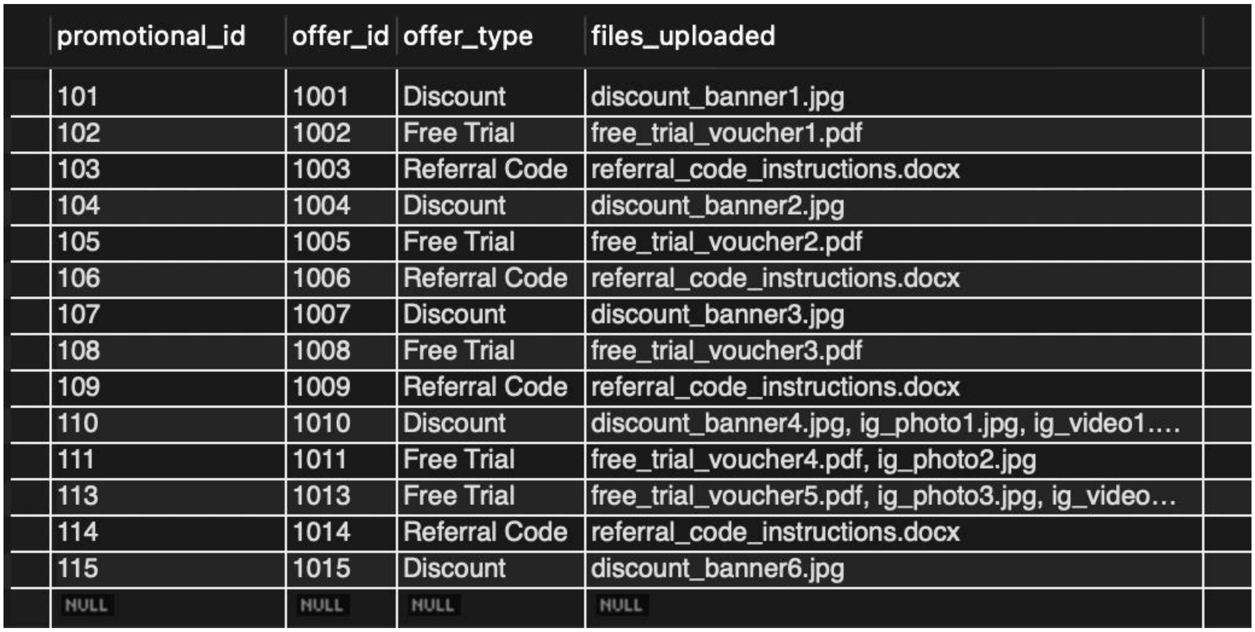
member_id	total_payments
1	8000
2	30000
3	42000
4	5000
5	30000
6	12000
7	5000
8	30000
9	12000
10	5000
11	30000
12	12000
13	5000
14	30000
15	12000

12)
Delete a specific promotional activity:

DELETE FROM Promotional_Activities WHERE promotional_id = 112;
SELECT * FROM Promotional_Activities;



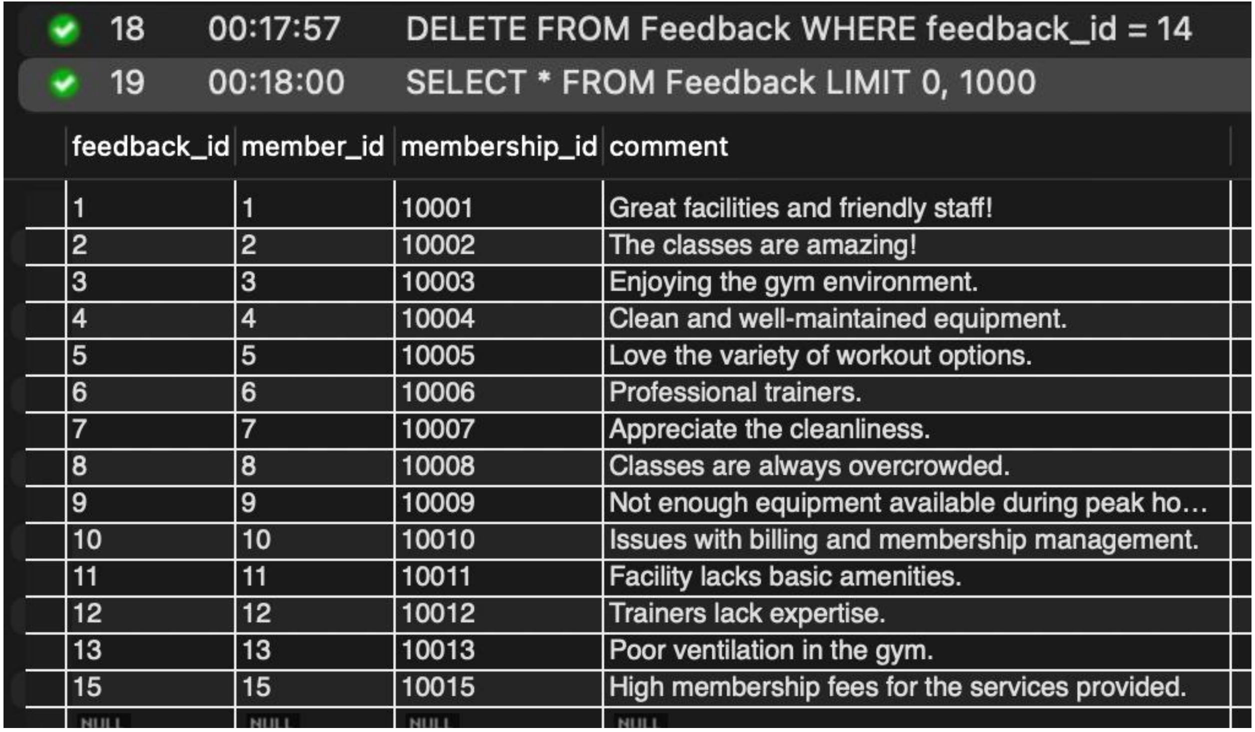
```
✓  15   00:15:18   DELETE FROM Promotional_Activities WHERE promotional_id = 112
✓  16   00:15:40   SELECT * FROM Promotional_Activities LIMIT 0, 1000
```



promotional_id	offer_id	offer_type	files_uploaded
101	1001	Discount	discount_banner1.jpg
102	1002	Free Trial	free_trial_voucher1.pdf
103	1003	Referral Code	referral_code_instructions.docx
104	1004	Discount	discount_banner2.jpg
105	1005	Free Trial	free_trial_voucher2.pdf
106	1006	Referral Code	referral_code_instructions.docx
107	1007	Discount	discount_banner3.jpg
108	1008	Free Trial	free_trial_voucher3.pdf
109	1009	Referral Code	referral_code_instructions.docx
110	1010	Discount	discount_banner4.jpg, ig_photo1.jpg, ig_video1.…
111	1011	Free Trial	free_trial_voucher4.pdf, ig_photo2.jpg
113	1013	Free Trial	free_trial_voucher5.pdf, ig_photo3.jpg, ig_video…
114	1014	Referral Code	referral_code_instructions.docx
115	1015	Discount	discount_banner6.jpg
NULL	NULL	NULL	NULL

13)
Delete a specific feedback comment:

DELETE FROM Feedback WHERE feedback_id = 14;
SELECT * FROM Feedback;



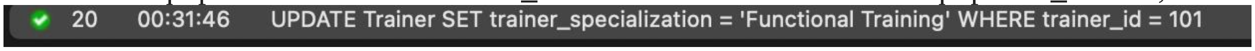
```
18   00:17:57   DELETE FROM Feedback WHERE feedback_id = 14
19   00:18:00   SELECT * FROM Feedback LIMIT 0, 1000
```

feedback_id	member_id	membership_id	comment
1	1	10001	Great facilities and friendly staff!
2	2	10002	The classes are amazing!
3	3	10003	Enjoying the gym environment.
4	4	10004	Clean and well-maintained equipment.
5	5	10005	Love the variety of workout options.
6	6	10006	Professional trainers.
7	7	10007	Appreciate the cleanliness.
8	8	10008	Classes are always overcrowded.
9	9	10009	Not enough equipment available during peak ho…
10	10	10010	Issues with billing and membership management.
11	11	10011	Facility lacks basic amenities.
12	12	10012	Trainers lack expertise.
13	13	10013	Poor ventilation in the gym.
15	15	10015	High membership fees for the services provided.
NULL	NULL	NULL	NULL

14)

Update the maintenance date of a specific equipment:

UPDATE Equipment SET maintainance_date = '2024-06-01' WHERE equipment_id = 14;

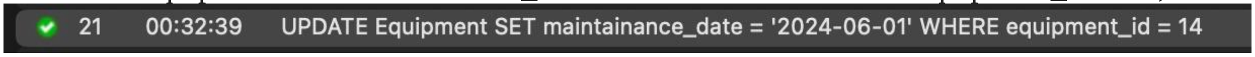


```
✓ 20    00:31:46    UPDATE Trainer SET trainer_specialization = 'Functional Training' WHERE trainer_id = 101
```

15)

Update the maintenance date of a specific equipment:

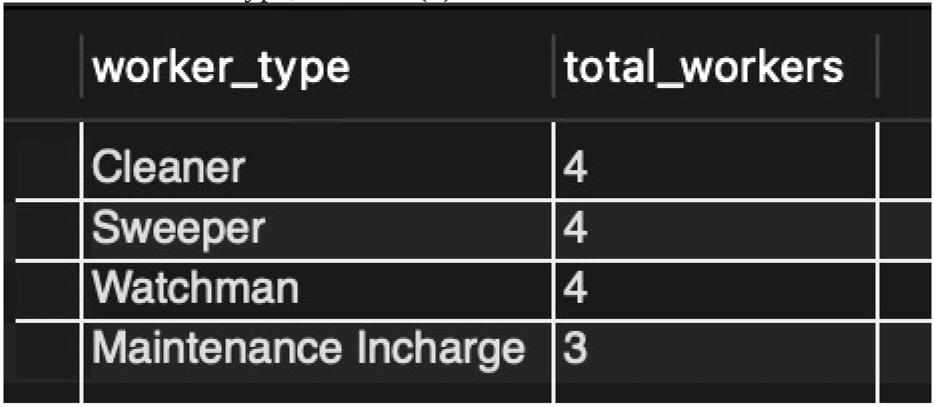
UPDATE Equipment SET maintainance_date = '2024-06-01' WHERE equipment_id = 14;



```
✓ 21    00:32:39    UPDATE Equipment SET maintainance_date = '2024-06-01' WHERE equipment_id = 14
```

16)
Get the total number of workers employed in each worker type.

SELECT worker_type, COUNT(*) AS total_workers FROM Workers GROUP BY worker_type;

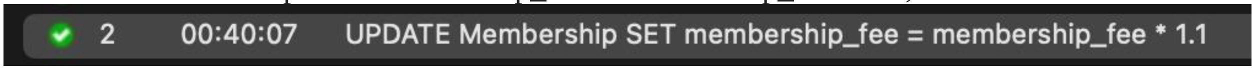


worker_type	total_workers
Cleaner	4
Sweeper	4
Watchman	4
Maintenance Incharge	3

17)

Increment the membership fee by 10% for all members:

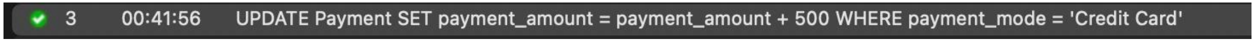
UPDATE Membership SET membership_fee = membership_fee * 1.1;



```
✓ 2    00:40:07    UPDATE Membership SET membership_fee = membership_fee * 1.1
```

18)
Increment the payment amount by 500 for all payments made by Credit Card:

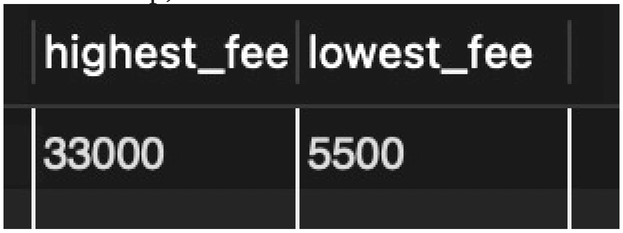
UPDATE Payment SET payment_amount = payment_amount + 500 WHERE payment_mode = 'Credit Card';



```
✓ 3    00:41:56    UPDATE Payment SET payment_amount = payment_amount + 500 WHERE payment_mode = 'Credit Card'
```

19)
Find the membership types with the highest and lowest membership fees.

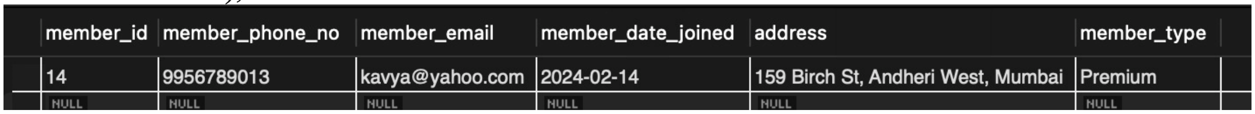
SELECT MAX(membership_fee) AS highest_fee, MIN(membership_fee) AS lowest_fee FROM Membership;



highest_fee	lowest_fee
33000	5500

20)
List all members who have not provided any feedback.

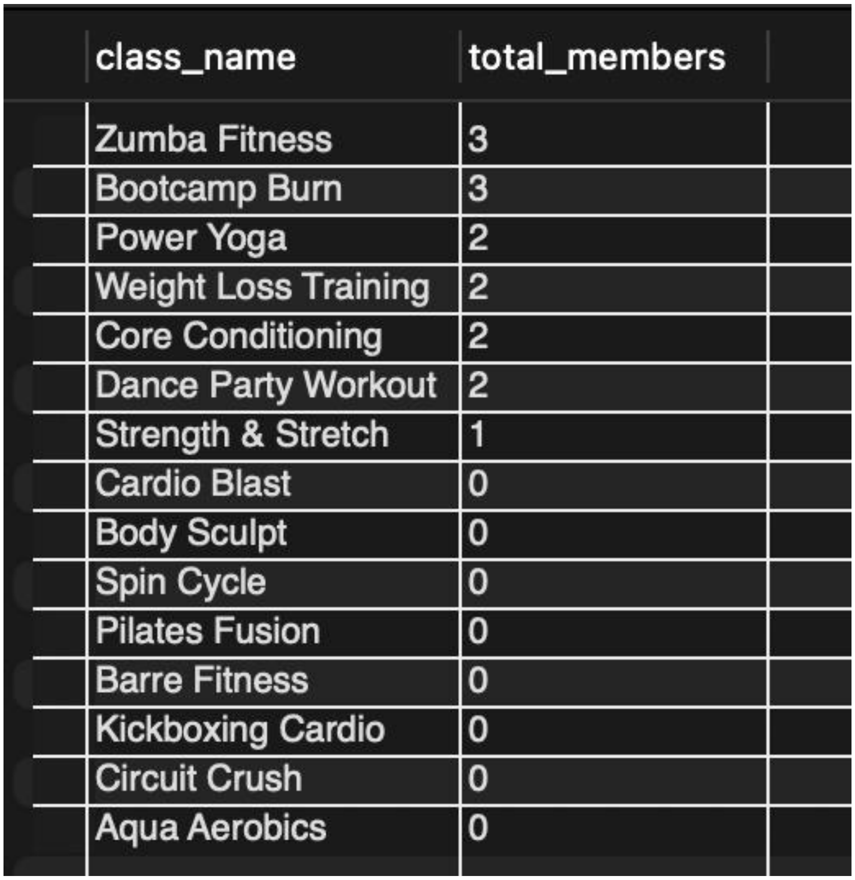
SELECT * FROM Member WHERE member_id NOT IN (SELECT DISTINCT member_id FROM Feedback);



member_id	member_phone_no	member_email	member_date_joined	address	member_type
14	9956789013	kavya@yahoo.com	2024-02-14	159 Birch St, Andheri West, Mumbai	Premium
NULL	NULL	NULL	NULL	NULL	NULL

21)
Retrieve the classes with the highest number of members.

SELECT c.class_name, COUNT(m.member_id) AS total_members FROM Class c LEFT JOIN Membership ms ON c.class_id = ms.class_id LEFT JOIN Member m ON ms.member_id = m.member_id GROUP BY c.class_name ORDER BY total_members DESC;

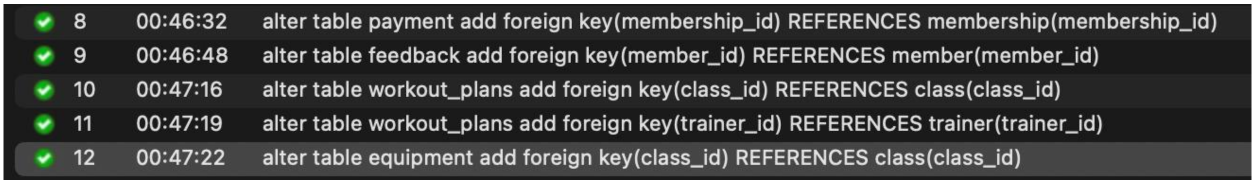


class_name	total_members
Zumba Fitness	3
Bootcamp Burn	3
Power Yoga	2
Weight Loss Training	2
Core Conditioning	2
Dance Party Workout	2
Strength & Stretch	1
Cardio Blast	0
Body Sculpt	0
Spin Cycle	0
Pilates Fusion	0
Barre Fitness	0
Kickboxing Cardio	0
Circuit Crush	0
Aqua Aerobics	0

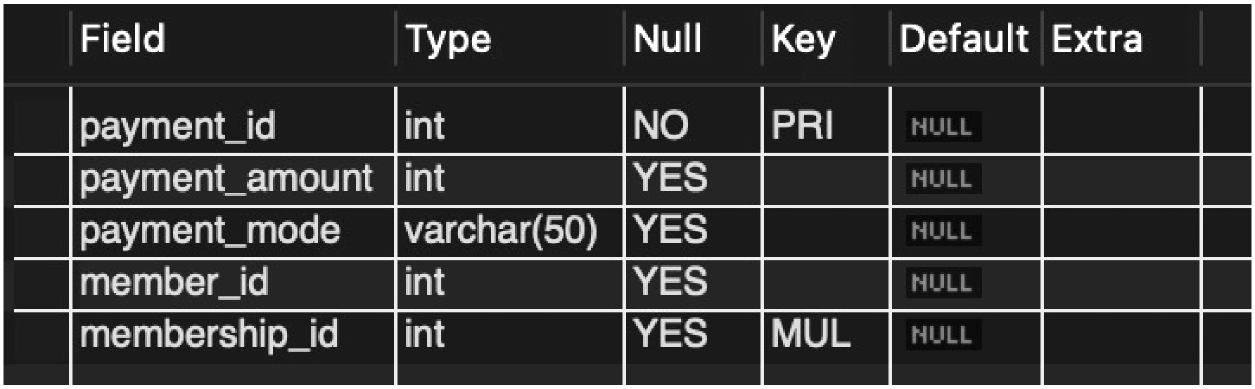
22)

Show Foreign Keys:
alter table payment add foreign key(membership_id) REFERENCES membership(membership_id);
alter table feedback add foreign key(member_id) REFERENCES member(member_id);
alter table workout_plans add foreign key(class_id) REFERENCES class(class_id);
alter table workout_plans add foreign key(trainer_id) REFERENCES trainer(trainer_id);
alter table equipment add foreign key(class_id) REFERENCES class(class_id);

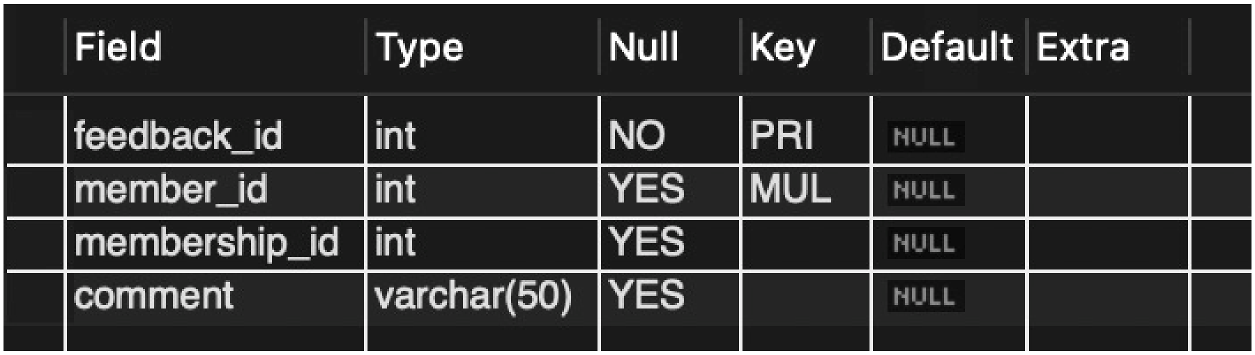
DESC PAYMENT;
DESC FEEDBACK;
DESC WORKOUT_PLANS;
DESC EQUIPMENT;



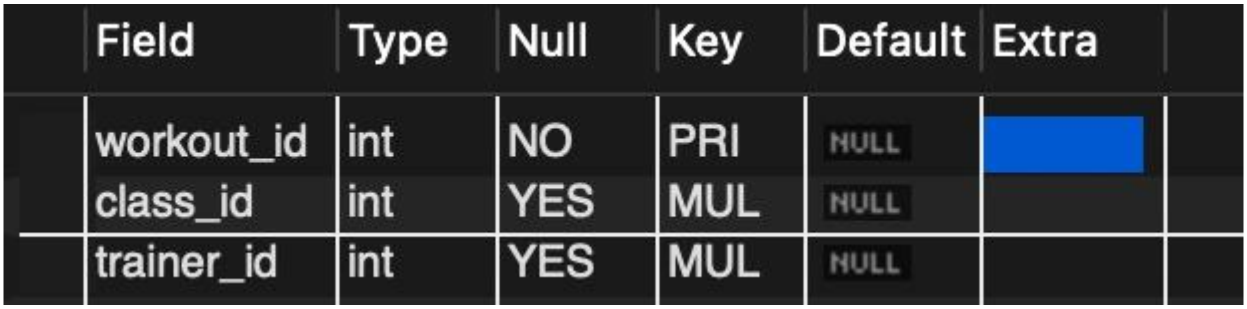
```
✓  8    00:46:32   alter table payment add foreign key(membership_id) REFERENCES membership(membership_id)
✓  9    00:46:48   alter table feedback add foreign key(member_id) REFERENCES member(member_id)
✓  10   00:47:16   alter table workout_plans add foreign key(class_id) REFERENCES class(class_id)
✓  11   00:47:19   alter table workout_plans add foreign key(trainer_id) REFERENCES trainer(trainer_id)
✓  12   00:47:22   alter table equipment add foreign key(class_id) REFERENCES class(class_id)
```



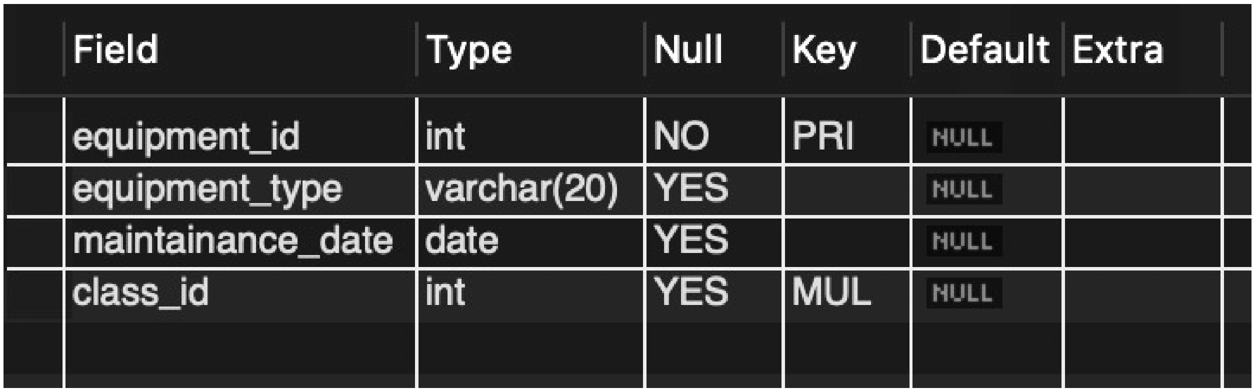
Field	Type	Null	Key	Default	Extra
payment_id	int	NO	PRI	NULL	
payment_amount	int	YES		NULL	
payment_mode	varchar(50)	YES		NULL	
member_id	int	YES		NULL	
membership_id	int	YES	MUL	NULL	



Field	Type	Null	Key	Default	Extra
feedback_id	int	NO	PRI	NULL	
member_id	int	YES	MUL	NULL	
membership_id	int	YES		NULL	
comment	varchar(50)	YES		NULL	



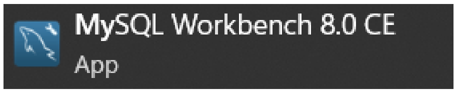
Field	Type	Null	Key	Default	Extra
workout_id	int	NO	PRI	NULL	
class_id	int	YES	MUL	NULL	
trainer_id	int	YES	MUL	NULL	



Field	Type	Null	Key	Default	Extra
equipment_id	int	NO	PRI	NULL	
equipment_type	varchar(20)	YES		NULL	
maintainance_date	date	YES		NULL	
class_id	int	YES	MUL	NULL	

VI. Project demonstration

Software used is MY SQL WORKBENCH and Draw.io



VII. Self -Learning beyond classroom

:

This project served as an immersive learning experience in database design and SQL, offering practical application of theoretical concepts within the context of gym management. Through creating tables for various entities such as members, trainers, classes, and equipment, I gained insight into database schema design and normalization techniques , which are essential for ensuring data integrity and minimizing redundancy.

Working on a gym management project was interesting because it involved things we all know, like memberships, classes, and workout details. Fixing mistakes and constantly improving the system helped me understand how gyms really work.

Creating primary and foreign keys in tables links them together, making sure the database structure stays organized and data remains connected. Writing SQL queries to add, get, or change data improved my ability to search and manage information efficiently.

VIII. Learning from the Project

The Gym Management project using SQL likely provided several valuable learning opportunities:

One of them is how to design a database schema, including entity relationships, normalization, and data integrity constraints, it also increased our knowledge regarding SQL and its syntax.

Implementing error handling mechanisms and data validation rules to maintain data integrity and handle unexpected scenarios gracefully.

By working on a real-world project scenario, you gained practical experience that goes beyond theoretical knowledge. This hands-on experience provided valuable insights into the challenges and considerations involved in developing and managing database systems in real-life scenarios.

Managing the Gym Management project involved planning, organization, and collaboration with team members or stakeholders. You learned project management skills such as requirement analysis, task prioritization, and effective communication.

IX. Challenges Faced

Challenges encountered in a Gym Management project using SQL are as follows:

Normalization Challenges: Addressing dependencies and anomalies while ensuring data integrity through normalization.

Error handling: Managing errors, including constraint violations or syntax issues, that arise during database operations necessitates robust error handling mechanisms. These mechanisms are crucial for offering informative feedback to users and preserving data integrity.

Complex Queries: Writing and optimizing SQL queries for various operations, including data retrieval, updates, and aggregation.

Managing Relationships Between Entities: Establishing and managing relationships between different entities such as members, classes, trainers, and equipment requires careful planning. Determining the appropriate type of relationships (one-to-one, one-to-many, many-to-many) and enforcing referential integrity constraints can be challenging.

X. Conclusion

The project provided a valuable opportunity to apply theoretical knowledge gained in SQL and DBMS courses to a real-world scenario. It reinforced concepts such as data modeling, query optimization, and database management in a practical context.

I learned how to organize and handle complicated connections between different types of information in a database. Knowing how to structure data relationships, keep information organized, and ensure everything stays accurate was really important for keeping the database working smoothly.

In essence, the Gym Management project served as a catalyst for holistic learning, fostering technical proficiency, problem-solving skills, and a deeper appreciation for the practical applications of SQL, DBMS, and software development principles.