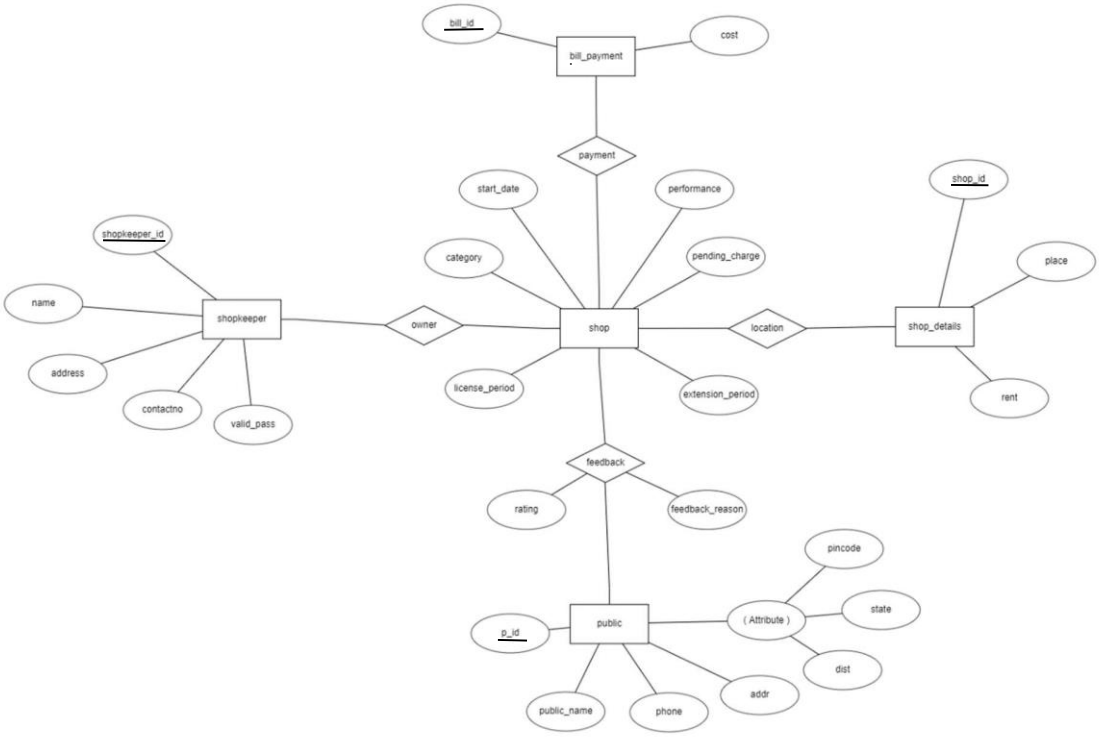
MARKET SHOP

ER DIAGRAM:



In this database it consists of total 5 entities:
1.bill_payment

2.shopkeeper

3.public

4.shop

5.shop_details

It also consists of relation:

1.feedback 4.payment

2.location

3.owner

Relationship:

Owner:
We have one to many relation ship for shopkeeper, shop.

Location:

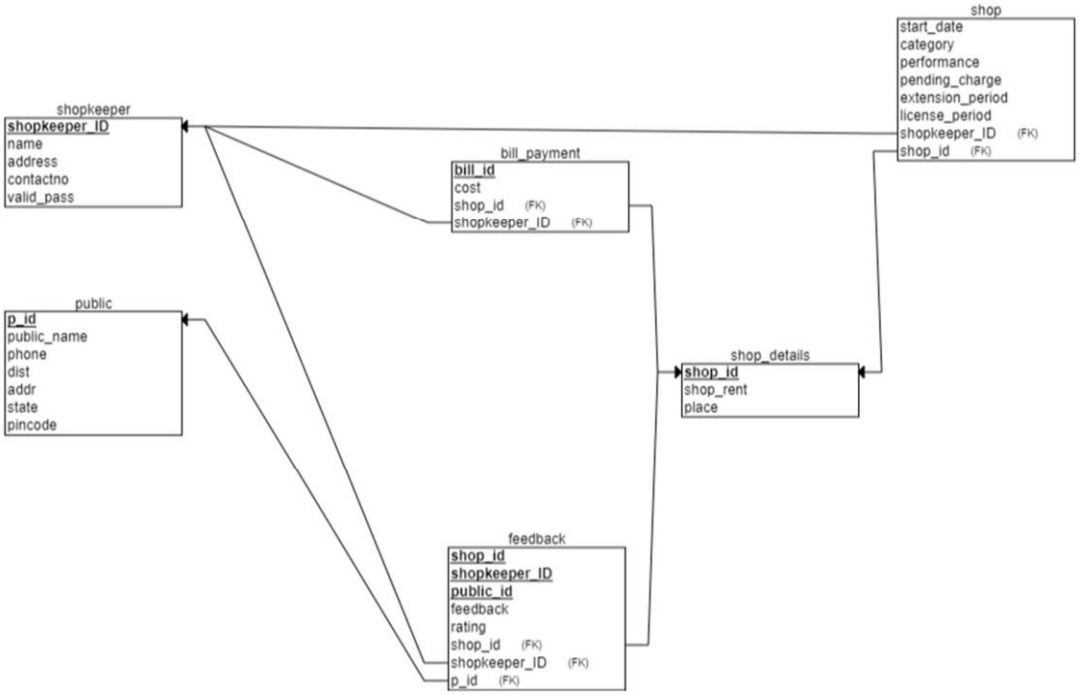
We have many to one relation ship in shop, shop_details.

Feedback:
We have many to many relationship in shop, customer

Payment:
We have one to many relationship in shop ,bill_payment

//

Relational Schema:



//

TABLES:

In this we require 6 tables

1.public:
It contains the details of the customer such as name,address etc.,

Sql query: create table public(p_id int NOT NULL PRIMARY KEY, public_name varchar(50),phone bigint,addr varchar(50),dist varchar(50),state varchar(50),pincode int);

describe public;

+-------------+-------------+------+-----+---------+-------+

| Field | Type | Null | Key | Default | Extra |

+-------------+-------------+------+-----+---------+-------+

| p_id | int(11) | NO | PRI | NULL | |

| public_name | varchar(50) | YES | | NULL | |

| phone | bigint(20) | YES | | NULL | |

| addr | varchar(50) | YES | | NULL | | | dist

| varchar(50) | YES | | NULL | |

| state | varchar(50) | YES | | NULL | |

| pincode | int(11) | YES | | NULL | |

+-------------+-------------+------+-----+---------+-------+

2.shopkeeper:
It contains details of the shopkeeper such as name,shopkeeper id, address etc.,

Mysql query: CREATE TABLE shopkeeper(shopkeeper_ID INT NOT NULL PRIMARY KEY,name varchar(50),address varchar(50),valid_pass varchar(50),contactno varchar(11));

+---------------+-------------+------+-----+---------+-------+

| Field | Type | Null | Key | Default | Extra |

+---------------+-------------+------+-----+---------+-------+

| shopkeeper_ID | int(11) | NO | PRI | NULL | |

| name | varchar(50) | YES | | NULL | |

| address | varchar(50) | YES | | NULL | |

| valid_pass | varchar(50) | YES | | NULL | |

| contactno | varchar(11) | YES | | NULL | |

+---------------+-------------+------+-----+---------+-------+

3.shop_details:
It contains the details of the shops such as owner name,shop id etc.,

Mysql query: create table shop_details(shop_id int NOT NULL PRIMARY KEY ,shop_rent int, place varchar(20));

+-----------+-------------+------+-----+---------+-------+

| Field | Type | Null | Key | Default | Extra |

+-----------+-------------+------+-----+---------+-------+

| shop_id | int(11) | NO | PRI | NULL | |

| shop_rent | int(11) | YES | | NULL | |

| place | varchar(20) | YES | | NULL | |

+-----------+-------------+------+-----+---------+-------+

4.bill_payment:

It contains the pending bills has to be cleared by the shop.

Mysql Query: create table bill_payment(bill_id int NOT NULL , cost int,shop_id int ,shopkeeper_id int, PRIMARY KEY(bill_id),FOREIGN KEY(shop_id) REFERENCES shop_details(shop_id), FOREIGN
KEY(shopkeeper_id) REFERENCES shopkeeper(shopkeeper_id));+---------------+---------+------+-----+---------+-------+

| Field | Type | Null | Key | Default | Extra |

+---------------+---------+------+-----+---------+-------+

| bill_id | int(11) | NO | PRI | NULL | |

| cost | int(11) | YES | | NULL | |

| shop_id | int(11) | YES | MUL | NULL | |

| shopkeeper_id | int(11) | YES | MUL | NULL | |

+---------------+---------+------+-----+---------+-------+

5. shop:
It contains the details of the respective shop.

Mysql Query:
create table shop(shop_id int,shopkeeper_id int,start_date date,category varchar(20),license_period smallint,extension_period smallint,pending_charge int,performance float,foreign key(shop_id) REFERENCES shop_details(shop_id),FOREIGN KEY(shopkeeper_id) REFERENCES shopkeeper(shopkeeper_id));

+------------------+-------------+------+-----+---------+-------+

| Field | Type | Null | Key | Default | Extra |

+------------------+-------------+------+-----+---------+-------+

| shop_id | int(11) | YES | MUL | NULL | |

| shopkeeper_id | int(11) | YES | MUL | NULL | |

| start_date | date | YES | | NULL | |

| category | varchar(20) | YES | | NULL | |

| license_period | smallint(6) | YES | | NULL | |

| extension_period | smallint(6) | YES | | NULL | |

| pending_charge | int(11) | YES | | NULL | |

| performance | float | YES | | NULL | |

+------------------+-------------+------+-----+---------+-------+

6.Feedback:
It contains the feedback of the customer.

Mysql Query:

create table feedback(shop_id int,shopkeeper_ID int,public_id int,feedback text,rating smallint,FOREIGN KEY(public_id) REFERENCES public(public_id));

+---------------+-------------+------+-----+---------+-------+

| Field | Type | Null | Key | Default | Extra |

+---------------+-------------+------+-----+---------+-------+

| shop_id | int(11) | YES | | NULL | |

| shopkeeper_ID | int(11) | YES | | NULL | |

| public_id | int(11) | YES | | NULL | |

| feedback | text | YES | | NULL | |

| rating | smallint(6) | YES | | NULL | |

+---------------+-------------+------+-----+---------+-------+

//

Triggers:

1.update_rating:

It will trigger when we insert a feedback in feedback table and it will the update the rating of the respective shop_id.

Delimiter $$ create trigger update_rating AFTER INSERT ON feedback FOR EACH

ROW

BEGIN

DECLARE avg_rating FLOAT DEFAULT 0;

SELECT SUM(rating)/count(*) INTO avg_rating FROM feedback where feedback.shopkeeper_ID=new.shopkeeper_ID AND feedback.shop_id=new.shop_id;

UPDATE shop SET Performance=avg_rating where shop.shopkeeper_id=new.shopkeeper_id AND shop.shop_id=new.shop_id;

END $$ DELIMITER

;

Verifying trigger:

select * from shop where shop_id=501;

+---------+---------------+------------+--------------+----------------+------------------+----------------+-------------+

| shop_id | shopkeeper_id | start_date | category | license_period | extension_period | pending_charge | performance |

+---------+---------------+------------+--------------+----------------+------------------+----------------+-------------+

| 501 | 701 | 2022-09-20 | nescafe | 100 | 60 | 7600 | 2 |

+---------+---------------+------------+--------------+----------------+------------------+----------------+-------------+

MariaDB [miniproject]> insert into feedback values(501,701,102,'good',5);

select * from shop where shop_id=501;

+---------+---------------+------------+--------------+----------------+------------------+----------------+-------------+

| shop_id | shopkeeper_id | start_date | category | license_period | extension_period | pending_charge | performance |

+---------+---------------+------------+--------------+----------------+------------------+----------------+-------------+

| 501 | 701 | 2022-09-20 | nescafe | 100 | 60 | 7600 | 3.5 |

Mysql query: insert into feedback values('502','703','109','nice',2);

2.after_billpayment:

It will trigger when the new bill has paid and updated in the bill_payment table and it will update the pending charges of the shop accordingly.

Delimiter $$ create trigger after_billpayment AFTER INSERT on

bill_payment

FOR EACH ROW

BEGIN

UPDATE shop set pending_charge=pending_charge-new.cost where shop.shopkeeper_id=new.shopkeeper_id and shop.shop_id=new.shop_id;

END $$ Delimiter;

Verifying trigger: select * from shop where shop_id=501;

+---------+---------------+------------+--------------+----------------+------------------+----------------+-------------+

| shop_id | shopkeeper_id | start_date | category | license_period | extension_period | pending_charge | performance |

+---------+---------------+------------+--------------+----------------+------------------+----------------+-------------+

| 501 | 701 | 2022-09-20 | nescafe | 100 | 60 | 10000 | 3.5 | +---------+---------------+------------+------------

--+----------------+------------------+----------------+-------------+

insert into bill_payment values(906,5050,505,706); select * from

shop;

+---------+---------------+------------+--------------+----------------+------------------+----------------+-------------+

| shop_id | shopkeeper_id | start_date | category | license_period | extension_period | pending_charge | performance |

+---------+---------------+------------+--------------+----------------+------------------+----------------+-------------+

| 501 | 701 | 2022-09-20 | nescafe | 100 | 60 | 7600 | 3.5 |

+---------+---------------+------------+--------------+----------------+------------------+----------------+-------------+

mysql query: insert into bill_payment values(911,1000,501,701);

//

Queries:

1.Details of shop details of different areas of the campus:

Mysql:select * from shop NATURAL JOIN shop_details;

+---------+---------------+------------+--------------+----------------+------------------+----------------+-------------+-----------+-----------+

| shop_id | shopkeeper_id | start_date | category | license_period | extension_period | pending_charge | performance | shop_rent | place |

+---------+---------------+------------+--------------+----------------+------------------+----------------+-------------+-----------+-----------+

| 501 | 701 | 2022-09-20 | nescafe | 100 | 60 | 7600 | 3 | 5000 | kalam |

| 502 | 703 | 2022-09-22 | generalstore | 150 | 30 | 14000 | 3 | 6000 | Foodcourt |

| 503 | 704 | 2022-07-24 | foodstall | 100 | 30 | 9000 | 3 | 6000 | Foodcourt |

| 504 | 705 | 2022-11-16 | generalstore | 100 | 30 | 10960 | 3.5 | 7000 | bquaters | +---------+--------
-------+------------+--------------+----------------+------------------+----------------+-------------+-----------+-----------+

2.Details of shopkeepers :

It contains the details of the shopkeeper like name ,address and mainly security pass.

mysql query: select * from shopkeeper;

+---------------+------+---------+------------+------------+

| shopkeeper_ID | name | address | valid_pass | contactno |

+---------------+------+---------+------------+------------+

| 701 | sk_1 | s_ad_1 | yes | 1234565891 |

| 702 | sk_2 | s_ad_2 | yes | 1234565892 |

| 703 | sk_3 | s_ad_3 | yes | 1234565893 |

| 704 | sk_4 | s_ad_4 | yes | 1234565894 |

| 705 | sk_5 | s_ad_5 | yes | 1234565895 |

| 706 | sk_6 | s_ad_6 | yes | 1234565896 |

| 707 | sk_7 | s_ad_7 | yes | 1234565897 |

| 708 | sk_8 | s_ad_8 | yes | 1234565898 |

+---------+------+---------+------------+------------+

Mysql query: select shopkeeper_ID,valid_pass from shopkeeper;

+---------+------------+

| shopkeeper_ID | valid_pass |

+---------+------------+

| 701 | yes |

| 702 | yes |

| 703 | yes |

| 704 | yes |

| 705 | yes |

| 706 | yes |

| 707 | yes |

| 708 | yes |

+---------+------------+

3.Pending charges from each shops:

We can also show the shops with pending bills and total anount of pending_charges

Mysql query: select shop_id,pending_charge from shop;

+---------+----------------+

| shop_id | pending_charge |

+---------+----------------+

| 501 | 7600 |

| 502 | 14000 |

| 503 | 9000 |

| 504 | 10960 |

+---------+----------------+

4.Summary of performance of the shop:

It shows the shop_id and shopkeeper id and mainly performance of the shop which is given by the

Mysql query: select shop_id,shopkeeper_id,performance from shop; +---------+---------------+--------

-----+

| shop_id | shopkeeper_id | performance |

+---------+---------------+-------------+

| 501 | 701 | 3 |

| 502 | 703 | 3 |

| 503 | 704 | 3 |

| 504 | 705 | 3.5 |

+---------+---------------+-------------+

5. Reminders for expiring license agreement period:

It shows the list of the shops whose license will expire within a month

Mysql query : SELECT shopkeeper_id, shop_id FROM Shop WHERE DATEDIFF (DATE ADD (Start Date,
INTERVAL (License_Period+Extension_Period)MONTH), CURDATE()) <<= 30;