Name: Aradhya Phutak

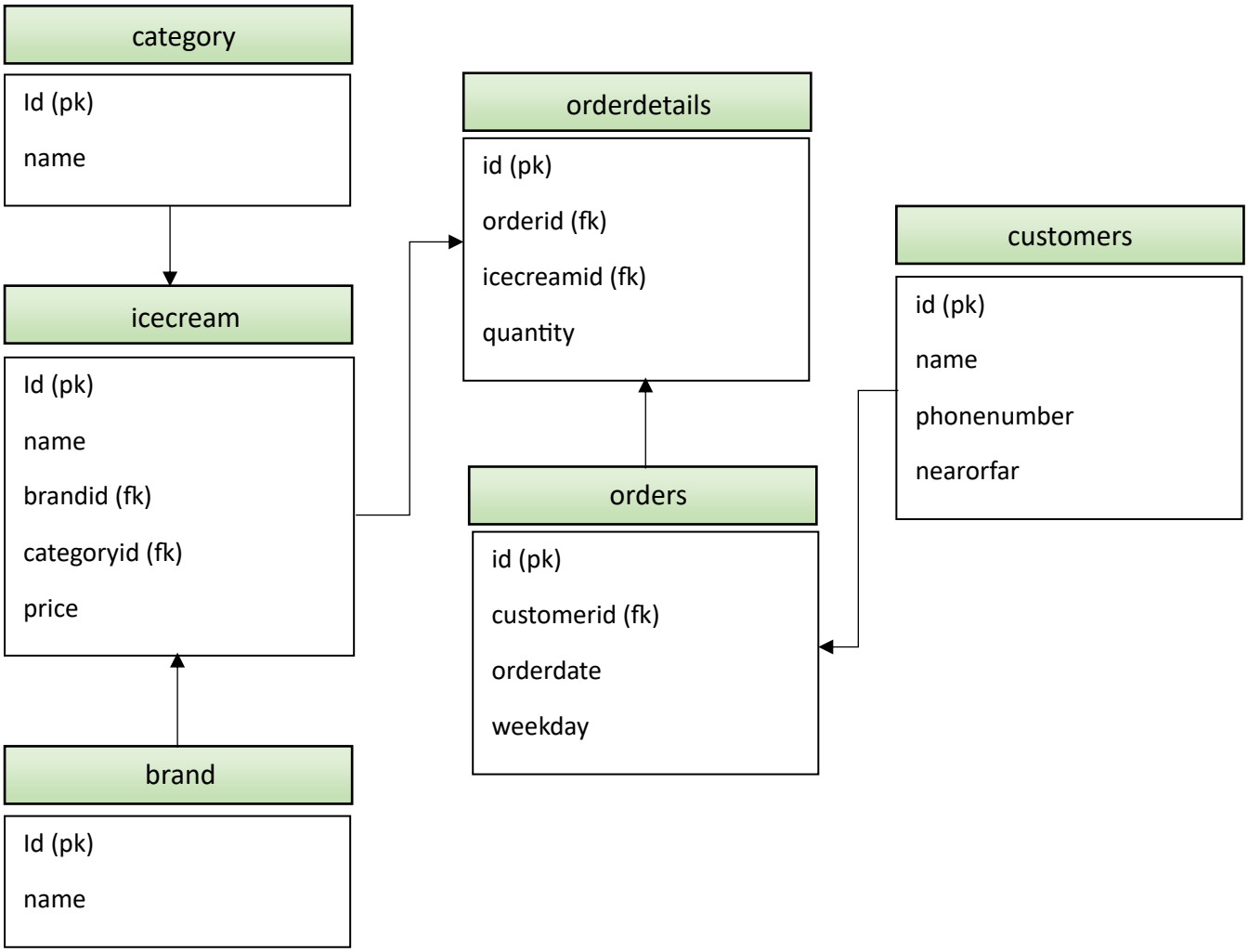
SQL Project

Begin by providing a brief background of an ice-cream shop that led to the need for the SQL based solution. This could be an overview of the business environment, the challenges the organization was facing, or the specific needs that prompted the analysis.

Objectives:

Clearly define the objectives of the case study. Outline what the case study aims to achieve, such as improving decision-making processes, enhancing data accessibility, or optimizing business operations through SQL queries.

Dataflow of the database:



-- Creating the database, tables and inserting the values in the tables

```
CREATE DATABASE icecreamshop;
USE icecreamshop;

CREATE TABLE category (
id INT PRIMARY KEY,
name VARCHAR(50) NOT NULL
);

INSERT INTO category (id, name)
VALUES
(1,"cone"),
(2,"cup"),
(3,"stick/kulfi"),
(4,"packet"),
(5,"others")
;


CREATE TABLE brand (
id INT PRIMARY KEY,
name VARCHAR(50) NOT NULL
);

INSERT INTO brand (id, name)
VALUES
(1,"creamery"),
(2,"havmor"),
(3,"motherDairy"),
```

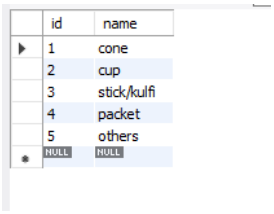
```
(4,"vadilal"),
(5,"frozenFrenzy"),
(6,"berryBliss")
;


CREATE TABLE iceCream (
   id INT NOT NULL,
   name VARCHAR(50) NOT NULL,
   brandid INT NOT NULL,
   categoryid INT NOT NULL,
   price INT NOT NULL,
   PRIMARY KEY (id),
   FOREIGN KEY (brandid) REFERENCES brand (id),
   FOREIGN KEY (categoryid) REFERENCES category (id)
);

-- data has been imported through "table data import wizard"

CREATE TABLE customer (
id INT PRIMARY KEY,
name VARCHAR(50) NOT NULL,
phoneNumber INT NOT NULL,
nearorfar VARCHAR(50) NOT NULL
);

-- data has been imported through "table data import wizard"
```

```
CREATE TABLE orders (
    id INT NOT NULL,
    customerid INT NOT NULL,
    orderDate DATE NOT NULL,
    weekday INT NOT NULL,
    PRIMARY KEY (id),
    FOREIGN KEY (customerid) REFERENCES customer (id)
);

-- data has been imported through "table data import wizard"

CREATE TABLE ordersdetails (
    id INT NOT NULL,
    orderid INT NOT NULL,
    icecreamid INT NOT NULL,
    quantity INT NOT NULL,
    PRIMARY KEY (id),
    FOREIGN KEY (orderid) REFERENCES orders (id),
    FOREIGN KEY (icecreamid) REFERENCES icecream (id)
);

-- data has been imported through "table data import wizard"
```

-- Queries for the tables

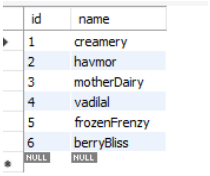
- All the tables

SELECT * FROM category;



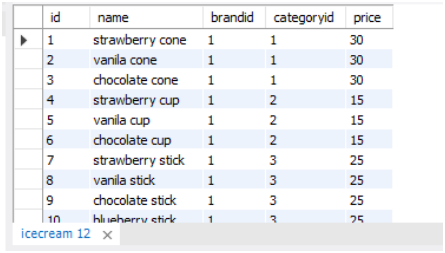
id	name
1	cone
2	cup
3	stick/kulfi
4	packet
5	others
NULL	NULL

SELECT * FROM brand;



id	name
1	creamery
2	havmor
3	motherDairy
4	vadilal
5	frozenFrenzy
6	berryBliss
NULL	NULL

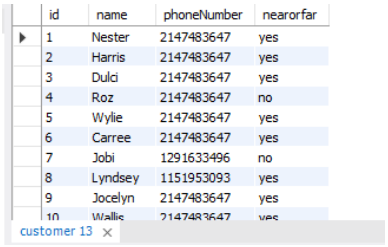
SELECT * FROM icecream;



id	name	brandid	categoryid	price
1	strawberry cone	1	1	30
2	vanila cone	1	1	30
3	chocolate cone	1	1	30
4	strawberry cup	1	2	15
5	vanila cup	1	2	15
6	chocolate cup	1	2	15
7	strawberry stick	1	3	25
8	vanila stick	1	3	25
9	chocolate stick	1	3	25
10	blueberry stick	1	3	25

icecream 12

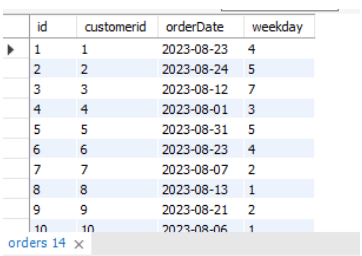
SELECT * FROM customer;



id	name	phoneNumber	nearorfar
1	Nester	2147483647	yes
2	Harris	2147483647	yes
3	Dulci	2147483647	yes
4	Roz	2147483647	no
5	Wylie	2147483647	yes
6	Carree	2147483647	yes
7	Jobi	1291633496	no
8	Lyndsey	1151953093	yes
9	Jocelyn	2147483647	yes
10	Wallis	2147483647	yes

customer 13

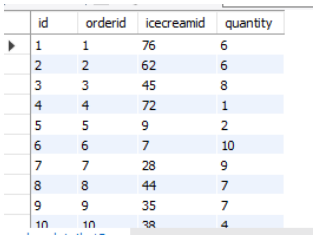
SELECT * FROM orders;



id	customerid	orderDate	weekday
1	1	2023-08-23	4
2	2	2023-08-24	5
3	3	2023-08-12	7
4	4	2023-08-01	3
5	5	2023-08-31	5
6	6	2023-08-23	4
7	7	2023-08-07	2
8	8	2023-08-13	1
9	9	2023-08-21	2
10	10	2023-08-06	1

orders 14

SELECT * FROM ordersdetails;

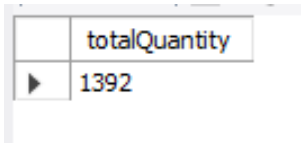


id	orderid	icecreamid	quantity
1	1	76	6
2	2	62	6
3	3	45	8
4	4	72	1
5	5	9	2
6	6	7	10
7	7	28	9
8	8	44	7
9	9	35	7
10	10	38	4

-- Sales overview

- total number of orders throughout the month

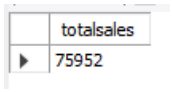
```
SELECT SUM(quantity) AS totalQuantity FROM ordersdetails;
```



totalQuantity
1392

- total sales throughout the month

```
SELECT SUM(od.quantity*ic.price) AS totalsales
FROM ordersdetails od
JOIN icecream ic ON ic.id = od.icecreamid;
```



totalsales
75952

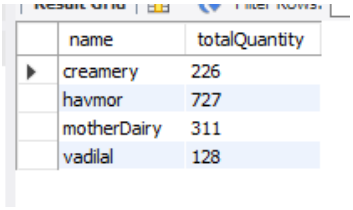
-- Remark:

-- in the given month the shop has acquired the total sales of Rs. 75,952 while selling total 1392 products of the ice creams

-- Brand wise Sales overview

-- quantity

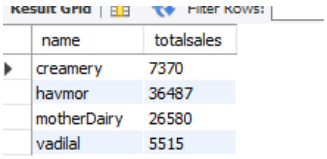
```
SELECT b.name, SUM(od.quantity) AS totalQuantity
FROM ordersdetails od
JOIN icecream ic ON ic.id = od.icecreamid
JOIN brand b ON b.id = ic.brandid
GROUP BY b.id;
```



name	totalQuantity
creamery	226
havmor	727
motherDairy	311
vadilal	128

-- Sales

```
SELECT b.name, SUM(od.quantity*ic.price) AS totalsales
FROM ordersdetails od
JOIN icecream ic ON ic.id = od.icecreamid
JOIN brand b ON b.id = ic.brandid
GROUP BY b.id;
```



name	totalsales
creamery	7370
havmor	36487
motherDairy	26580
vadilal	5515

-- Category wise Sales overview

-- quantity

```
SELECT c.name, SUM(od.quantity) AS totalQuantity
FROM ordersdetails od
JOIN icecream ic ON ic.id = od.icecreamid
JOIN category c ON c.id = ic.categoryid
GROUP BY c.id;
```

name	totalQuantity
cone	416
cup	452
stick/kulfi	328
packet	98
others	98

-- sales

```
SELECT c.name, SUM(od.quantity*ic.price) AS totalsales
FROM ordersdetails od
JOIN icecream ic ON ic.id = od.icecreamid
JOIN category c ON c.id = ic.categoryid
GROUP BY c.id;
```

name	totalsales
cone	14495
cup	12550
stick/kulfi	12775
packet	32362
others	3770

-- Locality wise Sales overview (near or far)

-- quantity

```
SELECT c.nearorfar, SUM(od.quantity) AS Totalquantity
FROM ordersdetails od
JOIN orders o ON o.id = od.orderid
JOIN customer c ON c.id = o.customerid
GROUP BY nearorfar;
```

nearorfar	Totalquantity
no	382
yes	1010

-- sales

```
SELECT c.nearorfar, SUM(od.quantity*ic.price) AS totalsales
FROM ordersdetails od
JOIN orders o ON o.id = od.orderid
```

JOIN customer c ON c.id = o.customerid

JOIN icecream ic ON ic.id = od.icecreamid

GROUP BY c.nearorfar;

	nearorfar	totalsales
▶	no	15895
	yes	60057

-- havmor has the most number of the sales to figure out the best category in it we have

SELECT b.name, c.name, SUM(od.quantity) AS totalorders, SUM(od.quantity*ic.price) AS Sales

FROM brand b

JOIN icecream ic ON ic.brandid = b.id

JOIN category c ON c.id = ic.categoryid

JOIN ordersdetails od ON od.icecreamid = ic.id

WHERE b.name = "havmor"

GROUP BY c.name;

	name	name	totalorders	Sales
▶	havmor	cone	235	7560
	havmor	cup	246	7035
	havmor	others	25	1245
	havmor	packet	40	11862
	havmor	stick/kulfi	181	8785

-- sales based on the ice cream products

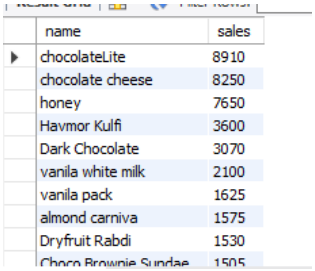
SELECT ic.name, SUM(ic.price*od.quantity) AS sales

FROM icecream ic

JOIN ordersdetails od ON od.icecreamid = ic.id

GROUP BY ic.name

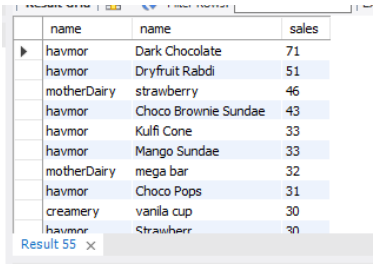
ORDER BY sales DESC;



name	sales
chocolateLite	8910
chocolate cheese	8250
honey	7650
Havmor Kulfi	3600
Dark Chocolate	3070
vanila white milk	2100
vanila pack	1625
almond carniva	1575
Dryfruit Rabdi	1530
Choco Brownie Sundae	1505

-- quantity based on the ice cream products

```
SELECT b.name, ic.name, SUM(od.quantity) AS sales

FROM icecream ic

JOIN ordersdetails od ON od.icecreamid = ic.id

JOIN brand b ON b.id = ic.brandid

GROUP BY ic.name

ORDER BY sales DESC;
```

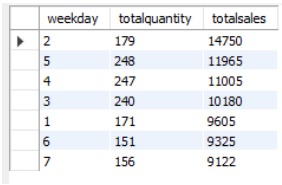


name	name	sales
havmor	Dark Chocolate	71
havmor	Dryfruit Rabdi	51
motherDairy	strawberry	46
havmor	Choco Brownie Sundae	43
havmor	Kulfi Cone	33
havmor	Mango Sundae	33
motherDairy	mega bar	32
havmor	Choco Pops	31
creamery	vanila cup	30
havmor	Strawberr	30

Result 55

-- sales and quantity based on the weekday

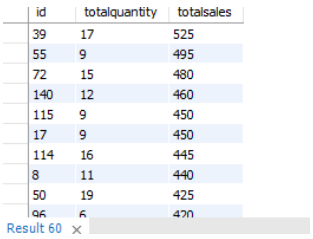
```
SELECT o.weekday, SUM(od.quantity) AS totalquantity, SUM(od.quantity*ic.price) AS totalsales

FROM ordersdetails od

JOIN orders o ON od.orderid = o.id

JOIN icecream ic ON ic.id = od.icecreamid

GROUP BY weekday

ORDER BY totalsales DESC;
```



weekday	totalquantity	totalsales
2	179	14750
5	248	11965
4	247	11005
3	240	10180
1	171	9605
6	151	9325
7	156	9122

-- BEST customers

```
SELECT cm.id, SUM(od.quantity) AS totalquantity, SUM(od.quantity*ic.price) AS totalsales
FROM ordersdetails od
JOIN orders o ON od.orderid = o.id
JOIN icecream ic ON ic.id = od.icecreamid
JOIN customer cm ON cm.id = o.customerid
GROUP BY cm.id
ORDER BY totalsales DESC;
```

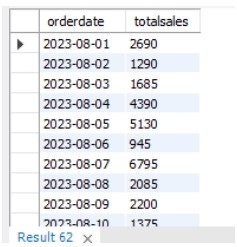


id	totalquantity	totalsales
39	17	525
55	9	495
72	15	480
140	12	460
115	9	450
17	9	450
114	16	445
8	11	440
50	19	425
96	6	420

Result 60

-- Sales in the date range

```
SELECT o.orderdate, SUM(od.quantity*ic.price) AS totalsales
FROM ordersdetails od
JOIN orders o ON o.id = od.orderid
JOIN icecream ic ON ic.id = od.icecreamid
WHERE o.orderdate BETWEEN '2023-08-01'
AND '2023-08-10'
GROUP BY orderdate
ORDER BY orderdate;
```



orderdate	totalsales
2023-08-01	2690
2023-08-02	1290
2023-08-03	1685
2023-08-04	4390
2023-08-05	5130
2023-08-06	945
2023-08-07	6795
2023-08-08	2085
2023-08-09	2200
2023-08-10	1375

Result 62

CONCLUSION:

In this case study, we explored the design and implementation of a relational database using SQL to address the needs of icecream shop. Through a structured approach, we identified key entities, defined their relationships, and ensured data integrity through appropriate constraints and normalization.