DBMS Project Report

on

Food Manufacturing and Inventory

Designed by:

Thoparam Adithya Vardhan-21CSB0B60

Elagonda Vivek-21CSB0B15

Submitted to:

Dr. T Ramakrishnudu

CONTENTS:-

- ➢ **PROBLEM STATEMENT**
- ➢ **ER DIAGRAM**
- ➢ **NORMALISATION**
- ➢ **RELATIONS**
- ➢ **RELATIONAL SCHEMA**
- ➢ **TABLES CREATION**
- ➢ **TABLES DISPLAY**
- ➢ **SQL QUERIES**

PROBLEM STATEMENT:

This database that we have constructed in our project basically gives a broad overview about how a food manufacturing company works, in general there are many companies in this business, we have analysed how the data is managed in those companies and have constructed an efficient DB.

This database consists of only essential components (entities) that covers almost all features of a efficient food manufacturing Company ,giving an edge to

●to have data of the distributors of the company who are distributing products.

● the ingredients used in the process of manufacturing.

● different types of products produced by the company.

● the types of employees working in the company and their working hours.

It mainly focuses on the entities and the relationships between them with all the key constraints and participation constraints.

ASSUMPTIONS:-

1) A particular ingredient is supplied by only one supplier but a supplier may supply more than one ingredients.
2) Only two types of workers are present one type of them are managers and other are general workers.
3) All the products manufactured and ingredients bought are under the supervision of the mangers.
4) A distributor may order more than one product and also a product may be ordered by more than one distributor.

ER DIAGRAM:
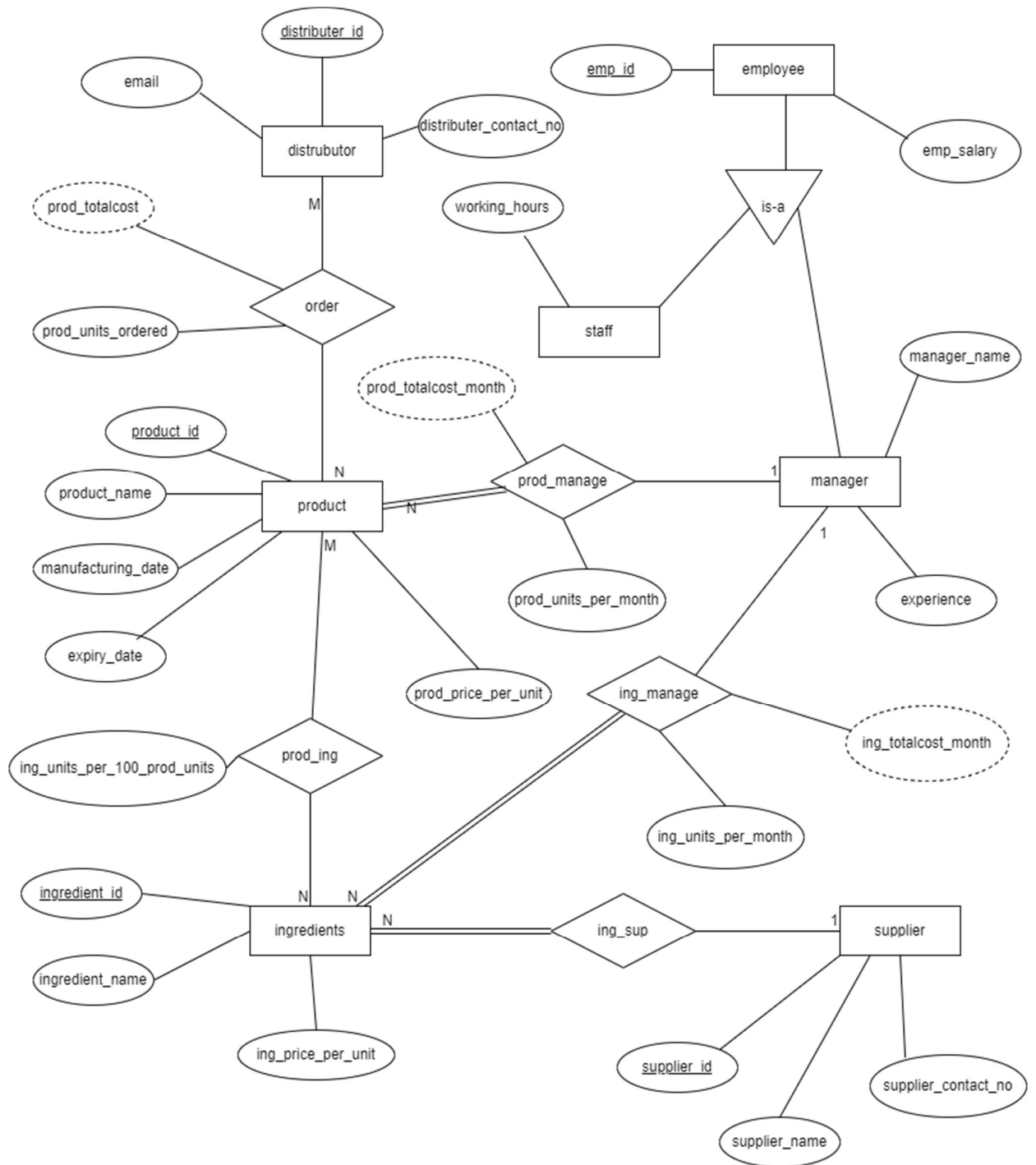
Note :-

Let's say that a table is in 1NF if it satisfies the below condition :-

" If there are no multivalued attributes , composite attributes , all columns have unique names and here the order in which data will be stored doesn't matter. "

Lets name this condition as 1NF condition.

NORMALISATION:-

ENTITY:- SUPPLIER

ATTRUBUTES:-

- Supplier_id
- Supplier_name
- Supplier_contact_no

FUNCTIONAL DEPENDENCIES :-

- Supplier_id ->supplier_name
- Supplier_id->supplier_contact_no

1 NF :- this relation satisfies 1NF condition. Therefore it is in 1st normal form.

2NF :- As we can see in the above functional dependencies the determinant (X in X ➟ Y) is a single attribute i.e, all our candidate keys are single attributes . so , there exists no proper subset of candidate keys.Therefore there is no chance of existence of partial dependency . Hence this table is in 2NF.

3 NF :- In this table all functional dependencies are from candidate key(prime attribute) to non prime attributes . Therefore there is no transitive dependency. Hence this table is in 3rd normal form.

BCNF :- In all the above modified tables , only the superkeys are determining all other attributes . Hence, we can say that the table is in BCNF .

ENTITY:- employee

ATTRUBUTES:-

- emp_id
- emp_salary

FUNCTIONAL DEPENDENCIES :-

- emp_id->emp_salary

1 NF :- this relation satisfies 1NF condition. Therefore it is in 1st normal form.

2NF :- As we can see in the above functional dependencies the determinant (X in X ➞ Y) is a single attribute i.e, all our candidate keys are single attributes . so , there exists no proper subset of candidate keys.Therefore there is no chance of existence of partial dependency . Hence this table is in 2NF.

3 NF :- In this table all functional dependencies are from candidate key(prime attribute) to non prime attributes . Therefore there is no transitive dependency. Hence this table is in 3rd normal form.

BCNF :- In all the above modified tables , only the superkeys are determining all other attributes . Hence, we can say that the table is in BCNF .

ENTITY:- staff

ATTRUBUTES:-

- emp_id
- working_hours

FUNCTIONAL DEPENDENCIES :-

emp_id->working_hours

1 NF :- this relation satisfies 1NF condition. Therefore it is in 1st normal form.

2NF :- As we can see in the above functional dependencies the determinant (X in X ➜ Y) is a single attribute i.e, all our candidate keys are single attributes . so , there exists no proper subset of candidate keys.Therefore there is no chance of existence of partial dependency . Hence this table is in 2NF.

3 NF :- In this table all functional dependencies are from candidate key(prime attribute) to non prime attributes . Therefore there is no transitive dependency. Hence this table is in 3rd normal form.

BCNF :- In all the above modified tables , only the superkeys are determining all other attributes . Hence, we can say that the table is in BCNF .

ENTITY:- manager

ATTRUBUTES:-

- emp_id
- experience
- manager_hours

FUNCTIONAL DEPENDENCIES :-

- emp_id->experience
- emp_id->manager_hours

1 NF :- this relation satisfies 1NF condition. Therefore it is in 1st normal form.

2NF :- As we can see in the above functional dependencies the determinant (X in X � Y) is a single attribute i.e, all our candidate keys are single attributes . so , there exists no proper subset of candidate keys.Therefore there is no chance of existence of partial dependency . Hence this table is in 2NF.

3 NF :- In this table all functional dependencies are from

candidate key(prime attribute) to non prime attributes .
Therefore there is no transitive dependency. Hence this
table is in 3rd normal form.

BCNF :- In all the above modified tables , only the
superkeys are determining all other attributes . Hence, we
can say that the table is in BCNF .

ENTITY:- **product**

ATTRUBUTES:-

- product_id
- product_name
- manufacturing_date
- expiry_date
- prod_unit_per_month
- prod_price_per_month
- prod_totalcost_month
- emp_id

FUNCTIONAL DEPENDENCIES :-

- product_id->product_name
- product_id->manufacturing_date
- product_id->expiry_date
- product_id->prod_units_per_month
- product_id->prod_price_per_month
- product_id->prod_totalcost_month
- product_id->emp_id

1 NF :- this relation satisfies 1NF condition. Therefore it is in 1st normal form.

2NF :- As we can see in the above functional dependencies the determinant (X in X ➡ Y) is a single attribute i.e, all our candidate keys are single attributes . so , there exists no proper subset of candidate keys.Therefore there is no chance of existence of partial dependency . Hence this table is in 2NF.

3 NF :- In this table all functional dependencies are from candidate key(prime attribute) to non prime attributes . Therefore there is no transitive dependency. Hence this table is in 3rd normal form.

BCNF :- In all the above modified tables , only the superkeys are determining all other attributes . Hence, we can say that the table is in BCNF .

ENTITY:- ingredients

ATTRUBUTES:-

- ingredient_id
- ingredient_name
- ing_price_per_unit
- supplier_id
- emp_id
- ing_units_per_month
- ing_totalcost_month

FUNCTIONAL DEPENDENCIES :-

- Ingredient_id->ingredient_name
- Ingredient_id->ing_price_per_unit
- Ingredient_id->supplier_id
- Ingredient_id->emp_id
- Ingredient_id->ing_units_per_month

- Ingredient_id->ing_totalcost_month

1 NF :- this relation satisfies 1NF condition. Therefore it is in 1st normal form.

2NF :- As we can see in the above functional dependencies the determinant (X in X ➞ Y) is a single attribute i.e, all our candidate keys are single attributes . so , there exists no proper subset of candidate keys.Therefore there is no chance of existence of partial dependency . Hence this table is in 2NF.

3 NF :- In this table all functional dependencies are from candidate key(prime attribute) to non prime attributes . Therefore there is no transitive dependency. Hence this table is in 3rd normal form.

BCNF :- In all the above modified tables , only the superkeys are determining all other attributes . Hence, we can say that the table is in BCNF .

ENTITY:- prod_ing

ATTRUBUTES:-

- product_id
- ingredient_id
- ing_units_per_100_prod_units

FUNCTIONAL DEPENDENCIES :-

- Product_id,ingredient_id->ing_units_per_100_prod_ing

1 NF :- this relation satisfies 1NF condition. Therefore it is in 1st normal form.

2NF :- As we can see in the above functional dependencies the determinant (X in X �డ Y) is a single attribute i.e, all our candidate keys are single attributes . so , there exists no proper subset of candidate

keys.Therefore there is no chance of existence of partial dependency . Hence this table is in 2NF.

3 NF :- In this table all functional dependencies are from candidate key(prime attribute) to non prime attributes . Therefore there is no transitive dependency. Hence this table is in 3rd normal form.

BCNF :- In all the above modified tables , only the superkeys are determining all other attributes . Hence, we can say that the table is in BCNF .

ENTITY:- distributor

ATTRUBUTES:-

- Distributor_id
- Email
- Distributor_contact_no

FUNCTIONAL DEPENDENCIES :-

- Distributor_id->email
- Distributor_id->distributor_contact_no

1 NF :- this relation satisfies 1NF condition. Therefore it is in 1st normal form.

2NF :- As we can see in the above functional dependencies the determinant (X in X ➡ Y) is a single attribute i.e, all our candidate keys are single attributes . so , there exists no proper subset of candidate keys.Therefore there is no chance of existence of partial dependency . Hence this table is in 2NF.

3 NF :- In this table all functional dependencies are from candidate key(prime attribute) to non prime attributes . Therefore there is no transitive dependency. Hence this table is in 3rd normal form.

BCNF :- In all the above modified tables , only the superkeys are determining all other attributes . Hence, we

can say that the table is in BCNF .

ENTITY:- order

ATTRUBUTES:-

- <u>Distributor_id</u>
- <u>Product_id</u>
- Prod_units_ordered
- Prod_totalcost

FUNCTIONAL DEPENDENCIES :-

- Distributor_id->email
- Distributor_id->distributor_contact_no

1 NF :- this relation satisfies 1NF condition. Therefore it is in 1st normal form.

2NF :- As we can see in the above functional dependencies the determinant (X in X ➡ Y) is a single attribute i.e, all our candidate keys are single attributes . so , there exists no proper subset of candidate keys.Therefore there is no chance of existence of partial
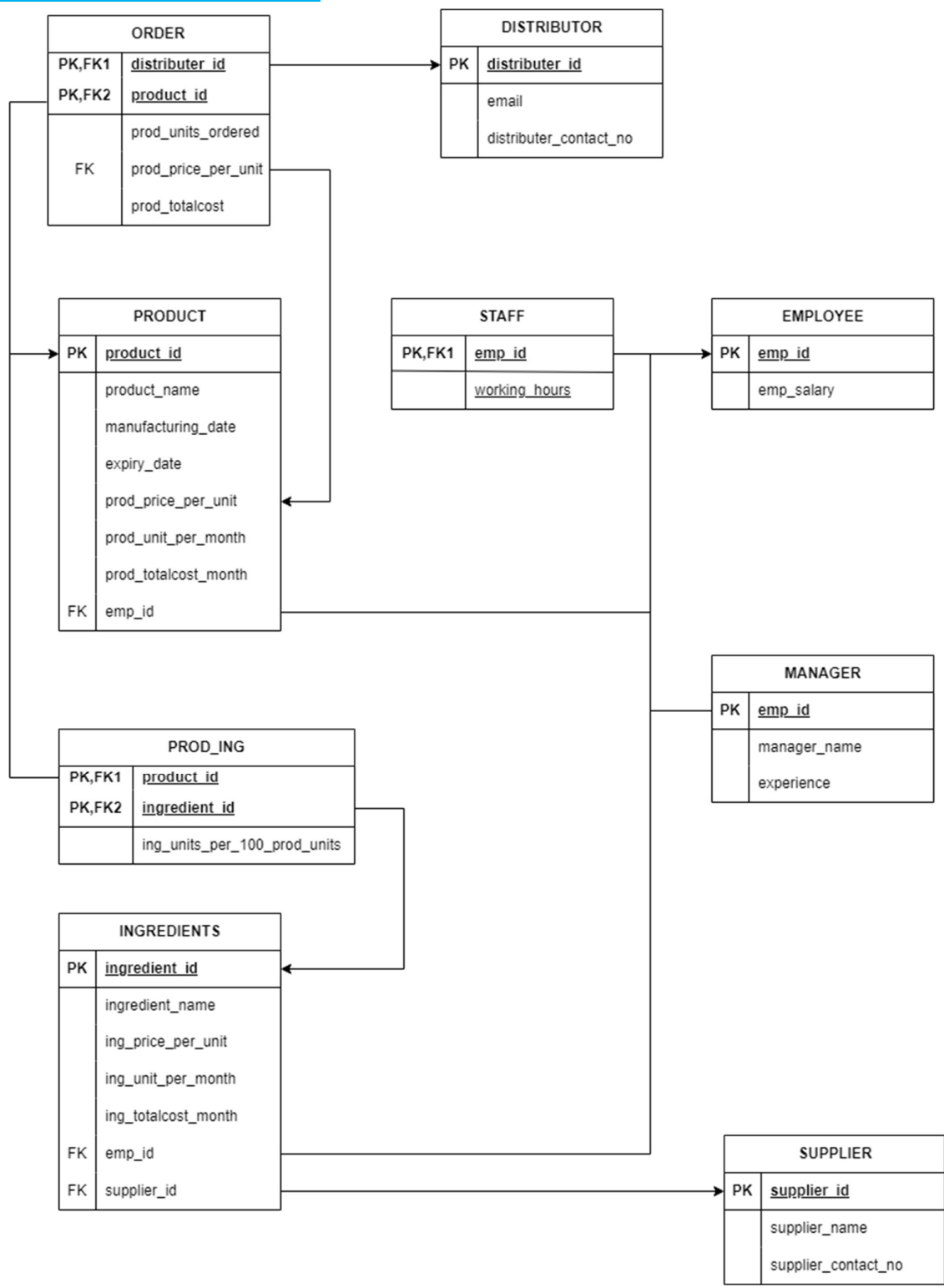
dependency . Hence this table is in 2NF.

3 NF :- In this table all functional dependencies are from candidate key(prime attribute) to non prime attributes . Therefore there is no transitive dependency. Hence this table is in 3rd normal form.

BCNF :- In all the above modified tables , only the superkeys are determining all other attributes . Hence, we can say that the table is in BCNF .

RELATIONSHIPS:

S.NO	ENTITY1	ENTITY2	RELATIONSHIP NAME	RELATION	EXPLANATION
1.	supplier	Ingredients	Ing_sup	One-to-many	A supplier may supply multiple products but an ingredient is supplied by a single supplier
2.	Ingredients	Products	Prod_ing	Many-to-many	An ingredient may used in making multiple products and a products require multiple ingredients to make
3.	Product	Distributer	Order	Many-to-many	A distributer may order multiple products and a product may be ordered by multiple distributers
4.	Ingredients	Manager	Ing_manage	Many-to-one	Multiple ingredients are managed by a single manager but a product is managed by single manager
5.	Product	Manager	Prod_manage	Many-to-one	Multiple products managed by a single manager but a product is manged by single manager

Relational schema:

ORDER

PK,FK1	distributer_id
PK,FK2	product_id
FK	prod_units_ordered
	prod_price_per_unit
	prod_totalcost

DISTRIBUTOR

PK	distributer_id
	email
	distributer_contact_no

PRODUCT

PK	product_id
	product_name
	manufacturing_date
	expiry_date
	prod_price_per_unit
	prod_unit_per_month
	prod_totalcost_month
FK	emp_id

STAFF

PK,FK1	emp_id
	working_hours

EMPLOYEE

PK	emp_id
	emp_salary

MANAGER

PK	emp_id
	manager_name
	experience

PROD_ING

PK,FK1	product_id
PK,FK2	ingredient_id
	ing_units_per_100_prod_units

INGREDIENTS

PK	ingredient_id
	ingredient_name
	ing_price_per_unit
	ing_unit_per_month
	ing_totalcost_month
FK	emp_id
FK	supplier_id

SUPPLIER

PK	supplier_id
	supplier_name
	supplier_contact_no

Creating tables:

1)SUPPLIER

create table supplier(

supplier_id number not null,

supplier_name varchar(50) not null,

supplier_contact_no number not null,

primary key(supplier_id)

);

2)EMPLOYEE

create table employee(

emp_id number not null,

emp_salary number not null,

primary key(emp_id)

);

3)STAFF

```
create table staff(
emp_id number references employee(emp_id),
working_hours number ,
primary key(emp_id)
);
```

4)MANAGER

```
create table manager(
emp_id number references employee(emp_id),
manager_name varchar(50),
experience varchar(50)
primary key(emp_id)
);
```

5)PRODUCT

```
create table product(
product_id number not null,
product_name varchar(50) not null,
```

```
manufacturing_date date,

expiry_date date,

prod_price_per_unit number not null,

prod_unit_per_month number not null,

prod_totalcost_month number not null,

emp_id number references employee(emp_id),

primary key(product_id)

);
```

6)INGREDIENTS

```
create table ingredients(

ingredient_id number not null,

ingredient_name varchar(50) not null,

ing_price_per_unit number not null,

ing_units_per_month number not null,

ing_totalcost_month number not null,

emp_id number references employee(emp_id),

supplier_id number references supplier(supplier_id),

primary id(ingredient_id)
```

);

7)PROD_ING

```
create table prod_ing(
product_id number references product(product_id),
ingredient_id number references ingredients(ingredient_id),
ing_units_per_100_prod_units number not null,
primary key(product_id,ingredient_id)
);
```

8)DISTRIBUTOR

```
create table distributor(
distributor_id number not null,
email varchar(50) not null,
distributor_contact_no number not null,
primary key(distributor_id)
);
```

9)ORDER

```
create table order(

distributor_id number references
distributor(distributor_id),

product_id number references product(product_id),

prod_units_ordered number not null,

prod_totalcost number not null,

primary key(distributor_id,product_id)
);
```

INSERTING DATA INTO VALUES:-

SUPPLIER:-

insert into supplier values(101,'dinesh',9998887810);

insert into supplier values(102,'rakesh',9998887811);

insert into supplier values(103,'suresh',9998887812);

insert into supplier values(104,'ramesh',9998887813);

insert into supplier values(105,'arjun',9998887814);

insert into supplier values(106,'vijay',9998887815);

EMPLOYEE:-

insert into employee values(501,100000);

insert into employee values(502,150000);

insert into employee values(503,50000);

insert into employee values(504,60000);

insert into employee values(505,40000);

insert into employee values(506,80000);

insert into employee values(507,90000);

insert into employee values(508,20000);

insert into employee values(509,15000);

insert into employee values(510,30000);

STAFF:-

insert into staff values(503,6);

insert into staff values(504,5);

insert into staff values(505,4);

insert into staff values(506,8);

insert into staff values(507,4);

insert into staff values(508,6);

insert into staff values(504,7);

insert into staff values(510,8);

MANAGER:-

insert into manager values(501,'krishna','15years');

insert into manager values(502,'Hari','20years');

PRODUCT:-

insert into product values(301,'prod1','2021-01-01','2024-01-01',6000,20,120000,502);

insert into product values(302,'prod2','2021-02-02','2024-02-02',8000,10,80000,502);

insert into product values(303,'prod3','2021-03-03','2024-03-03',10000,8,80000,502);

insert into product values(304,'prod4','2021-04-04','2024-04-04',14000,6,84000,502);

insert into product values(305,'prod5','2021-05-05','2024-05-05',6000,11,66000,502);

insert into product values(306,'prod6','2021-06-06','2024-06-06',7000,15,105000,502);

insert into product values(307,'prod7','2021-07-07','2024-07-07',9000,14,126000,502);

insert into product values(308,'prod8','2021-08-08','2024-08-08',11000,12,132000,502);

INGREDIENTS:-

insert into ingredients values(201,'ing1',1000,20,20000,501,101);

insert into ingredients values(202,'ing2',500,10,50000,501,102);

insert into ingredients values(203,'ing3',800,15,12000,501,103);

insert into ingredients values(204,'ing4',200,20,4000,501,103);

insert into ingredients values(205,'ing5',1100,15,16500,501,104);

insert into ingredients values(206,'ing6',1200,20,24000,501,105);

insert into ingredients
values(207,'ing7',900,10,9000,501,105);

insert into ingredients
values(208,'ing8',800,15,12000,501,106);

insert into ingredients
values(209,'ing9',700,20,14000,501,102);

insert into ingredients
values(210,'ing10',500,10,5000,501,101);

insert into ingredients
values(211,'ing11',400,10,4000,501,104);

PROD_ING:-

insert into prod_ing values(301,201,105);

insert into prod_ing values(301,204,109);

insert into prod_ing values(301,206,110);

insert into prod_ing values(302,202,120);

insert into prod_ing values(302,205,104);

insert into prod_ing values(302,203,103);

```sql
insert into prod_ing values(303,210,110);

insert into prod_ing values(303,207,125);

insert into prod_ing values(304,208,130);

insert into prod_ing values(304,211,165);

insert into prod_ing values(304,209,114);

insert into prod_ing values(305,201,119);

insert into prod_ing values(305,204,108);

insert into prod_ing values(306,202,121);

insert into prod_ing values(306,205,109);

insert into prod_ing values(306,206,111);

insert into prod_ing values(307,203,113);

insert into prod_ing values(307,207,112);

insert into prod_ing values(308,208,117);

insert into prod_ing values(308,209,119);

insert into prod_ing values(308,210,121);
```

DISTRIBUTOR:-

insert into distributor values(401,'dist1@gmail.com',9998887771);

insert into distributor values(402,'dist2@gmail.com',9998887772);

insert into distributor values(403,'dist3@gmail.com',9998887773);

insert into distributor values(404,'dist4@gmail.com',9998887774);

insert into distributor values(405,'dist5@gmail.com',9998887775);

ORDER:-

insert into order values(401,301,20,100000);

insert into order values(401,303,10,200000);

insert into order values(402,302,30,300000);

insert into order values(402,304,40,400000);

insert into order values(402,305,50,500000);

```
insert into order values(403,301,15,600000);
insert into order values(403,302,60,700000);
insert into order values(404,303,30,800000);
insert into order values(404,306,20,900000);
insert into order values(404,307,20,200000);
insert into order values(404,308,15,500000);
insert into order values(405,304,10,900000);
insert into order values(405,305,25,800000);
insert into order values(405,306,10,700000);
```

Tables Display:

1)SUPPLIER

select * from supplier;

	SUPPLIER_ID	SUPPLIER_NAME	SUPPLIER_CONTACT_NO
1	101	dinesh	9998887810
2	102	rakesh	9998887811
3	103	suresh	9998887812
4	104	ramesh	9998887813
5	105	arjun	9998887814
6	106	vijay	9998887815

2)EMPLOYEE

select * from employee ;

	EMP_ID	EMP_SALARY
1	501	100000
2	502	150000
3	503	50000
4	504	60000
5	505	40000
6	506	80000
7	507	90000
8	508	20000
9	509	15000
10	510	30000

3)STAFF:

select * from staff;

	EMP_ID	WORKING_HOURS
1	503	6
2	504	5
3	505	4
4	506	8
5	507	4
6	508	6
7	510	8

4)MANAGER

select * from manager;

	EMP_ID	MANAGER_NAME	EXPERIENCE
1	501	krishna	15years
2	502	Hari	20years

5)PRODUCT

select * from product;

	PRODUCT_ID	PRODUCT_NAME	MANUFACTURING_DATE	EXPIRY_DATE	PROD_PRICE_PER_UNIT	PROD_UNIT_PER_MONTH	PROD_TOTALCOST_MONTH	EMP_ID
1	301	prod1	2021-01-01	2024-01-01	6000	20	120000	502
2	302	prod2	2021-02-02	2024-02-02	8000	10	80000	502
3	303	prod3	2021-03-03	2024-03-03	10000	8	80000	502
4	304	prod4	2021-04-04	2024-04-04	14000	6	84000	502
5	305	prod5	2021-05-05	2024-05-05	6000	11	66000	502
6	306	prod6	2021-06-06	2024-06-06	7000	15	105000	502
7	307	prod7	2021-07-07	2024-07-07	9000	14	126000	502
8	308	prod8	2021-08-08	2024-08-08	11000	12	132000	502

6)INGREDIENTS

select * from ingredients;

	INGREDIENT_ID	INGREDIENT_NAME	ING_PRICE_PER_UNIT	ING_UNITS_PER_MONTH	ING_TOTALCOST_MONTH	EMP_ID	SUPPLIER_ID
1	201	ing1	1000	20	20000	501	101
2	202	ing2	500	10	50000	501	102
3	203	ing3	800	15	12000	501	103
4	204	ing4	200	20	4000	501	103
5	205	ing5	1100	15	16500	501	104
6	206	ing6	1200	20	24000	501	105
7	207	ing7	900	10	9000	501	105
8	208	ing8	800	15	12000	501	106
9	209	ing9	700	20	14000	501	102
10	210	ing10	500	10	5000	501	101
11	211	ing11	400	10	4000	501	104

7)PROD_ING

select * from prod_ing;

	PRODUCT_ID	INGREDIENT_ID	ING_UNITS_PER_100_PROD_UNITS
1	301	201	105
2	301	204	109
3	301	206	110
4	302	202	120
5	302	205	104
6	302	203	103
7	303	210	110
8	303	207	125
9	304	208	130
10	304	211	165
11	304	209	114
12	305	201	119
13	305	204	108
14	306	202	121
15	306	205	109
16	306	206	111
17	307	203	113
18	307	207	112
19	308	208	117
20	308	209	119
21	308	210	121

8)DISTRIBUTOR

select * from distributor;

	DISTRIBUTOR_ID	EMAIL	DISTRIBUTOR_CONTACT_NO
1	401	dist1@gmail.com	9998887771
2	402	dist2@gmail.com	9998887772
3	403	dist3@gmail.com	9998887773
4	404	dist4@gmail.com	9998887774
5	405	dist5@gmail.com	9998887775

9)ORDER1

select * from order1;

	DISTRIBUTOR_ID	PRODUCT_ID	PROD_UNITS_ORDERED	PROD_TOTALCOST
1	401	301	20	100000
2	401	303	10	200000
3	402	302	30	300000
4	402	304	40	400000
5	402	305	50	500000
6	403	301	15	600000
7	403	302	60	700000
8	404	303	30	800000
9	404	306	20	900000
10	404	307	20	200000
11	404	308	15	500000
12	405	304	10	900000
13	405	305	25	800000
14	405	306	10	700000

SQL Queries:-

1)Write a SQL query to display email and contact number of distributor who ordered product with product_id=301?

Query:

Select distributor_id,email, distributor_contact_no from order1 natural join distributor where product_id=301;

	DISTRIBUTOR_ID	EMAIL	DISTRIBUTOR_CONTACT_NO
1	401	dist1@gmail.com	9998887771
2	403	dist3@gmail.com	9998887773

2) Write an SQL query to display ingredients supplied by supplier 'arjun'.

Query:

Select supplier_name,ingredient_id,ingredient_name from ingredients natural join supplier where supplier_name= 'arjun';

	SUPPLIER_NAME	INGREDIENT_ID	INGREDIENT_NAME
1	arjun	206	ing6
2	arjun	207	ing7

3)Write an SQL query to display the total amount the company spends in buying ingredients.

Query:

Select sum(ing_totalcost_month) as totalcost_of_purchase from ingredients;

	TOTALCOST_OF_PURCHASE
1	170500

4)Write an SQL query to display the cost of purchase of goods by distributor with id number 404.

Query:

Select sum(prod_totalcost) as purchase_cost from order where distributor_id=404;

	PURCHASE_COST
1	2400000

5)Write an SQL query to display total units of ingredients supplied by each supplier.

Query:

Select supplier_id,sum(ing_units_per_month) as units_supplied from ingredients natural join supplier group by supplier_id;

	SUPPLIER_ID	UNITS_SUPPLIED
1	105	30
2	104	25
3	101	30
4	103	35
5	102	30
6	106	15

Thank you