SPEARS
SCHOOL OF BUSINESS

A
PROJECT REPORT
ON

"FOOD DELIVERY DATABASE SYSTEM"

MSIS-5643

SPEARS
SCHOOL OF BUSINESS

A
PROJECT REPORT
ON

"FOOD DELIVERY DATABASE SYSTEM"

SUBMITTED BY

RAVALI MUSTY- A20101635

MRUNALI KADAM-A20114076

BHAVANA BAHETI- A20115372

CHAKRADHAR PATHI- A20096739

UNDER THE GUIDANCE OF
Dr. ALI AMIRI

A REPORT SUBMITTED IN PART FULFILMENT OF THE PROJECT
AND COURSE **MSIS-5643**

Table of Contents

Executive Summary

Business Description:

Our Database is a Food Delivery Database system which performs orders as per customer requests and has information about orders, delivery, customer details, and menu form where the customer's order the food. Once the customer places an order, employees receive the orders and only after the payment is done, he updates the status as done. The customer can then track the status of his order by checking the status information from the portal which he logs in. Once the order is prepared, the delivery team picks up the food and delivers it to the customer and updates the status as 'Delivered'. The administrator can add and edit food items in the menu and he can also fetch the details of total orders or total sales made in a day. Based on the total amount of sales made in a day, the Manager can assess the progress of his business.

Type of User:

It is used by administrators of our database to keep track on the customer order requests and deal with the orders and delivery. Our employees receive the orders, Delivery team takes care of the delivery of the food items to the customer, customers can track the order status. The Manager of the business takes care of the total sales in a day and number of orders.

Features and Functions:

Manage the customers and their orders, track the order by the delivery status and maintaining the details of the price of food items. We can also keep track of the total sales and total number of orders generated in a single day.

Business Rules:

- A customer can place zero, one or many orders.
- A customer address can be associated with only one address and one phone number.
- The customer must have valid login details.
- Every order can have zero or multiple deliveries depending on the quantity of products.
- One order can have multiple order items
- Multiple order item id can have one order id.
- One order item can have one or more foods.
- Each food item may have one or more order items.
- A delivery date cannot be prior to the order date.

Data Dictionary

Entity Name	Description	Aliases	Occurrence
Customer_detail	It describes all the customers Information.	Order_detail	Each customer places 0 or many orders.
Delivery	It describes all the delivery details.	Order_detail	Every order can have 0 or more deliveries depending on the quantity of food items.
Menu	It describes the food items.	Order_item	Each food item may have 0 or more order items.
Order_item	It describes the order items depending on the quantity and the total prices.	Order_detail	Each order may have 1 or many order items.

Entity name	Multiplicity	Relationship	Multiplicity	Entity name
Customers	0:1	places	0: *	orders
Order_item	0: *	has	0:1	Orders
Menu	0:1	Fetched	0: *	Order_items
order	0:1	reserves	0: *	Delivery

Entity name	Attributes	Description	Data type and length	Nulls	Multivalued
Customer_Detail	Cust_ID	It shows the Id of the customer	Number[10]	No	No

	First_Name	First name of the customer	Varchar2[25]	No	No
	Last_Name	Last name of the customer	Varchar2[25]	No	No
	Login_ID	Login ID of the customer	Varchar2[25]	No	No
	Login_Password	Login password of the customer to login	Varchar2[15]	No	No
	Email_ID	Email id of the customer	Varchar2[20]	No	No
	House_Number	House Number of the customer	Varchar2[5]	No	No
	Street	Street in which the customer lives.	Varchar2[25]	No	No
	City	City in which the customer is located	Varchar2[25]	No	No
	State	State in which the customer is located	Varchar2[25]	No	No
	Phone_number	Gives the contact number of the customer	Number[10]	No	No
	Zipcode	Zip code of the area where customer resides.	Number[9]	No	No
Order_Detail	Order_ID	It shows the order id	Varchar2[10]	No	No
	Cust_ID	It shows the customer id	Number[10]	Yes	No
	Order_Date	It shows the date of order placed	Timestamp	No	No
	Order_Status	It gives the status of order placed	Varchar2[25]	No	No
Delivery	Delivery_ID	It shows the delivery id	Varchar2[10]	No	No
	Order_ID	It shows the order id	Varchar2[10]	Yes	No

	Delivery_Datetime	It shows the delivery date time.	Timestamp	No	No
	Delivery_Status	It shows the date its delivered	Varchar2[15]	No	No
Order_item	Order_item_id	It shows the order item id	Number[10]	No	No
	Food_ID	It gives food item id	Number[10]	Yes	No
	Order_ID	It shows order id	Varchar2[10]	No	No
	Quantity	It shows the quantity	Number[2]	No	No
Menu	Food_ID	It gives food item id	Number[10]	No	No
	Food_Name	It gives name of food item	Varchar2[150]	No	No
	Food_Price	It shows the food item price	Number[4,2]	No	No

List of Entities

- Customer_Detail
- Order_Detail
- Delivery
- Order_Item
- Menu

List of Attributes & Constraints

- **Customer_Detail:**

1. Cust_Id – NUMBER(10)
2. First_Name – VARCHAR2(25)
3. Last_Name - VARCHAR2(25)
4. Login_ID – VARCHAR2(25)
5. Login_Password – VARCHAR2(25)
6. Email_ID - VARCHAR2(25)
7. City - VARCHAR2(25)
8. State - VARCHAR2(25)
9. Phone_Number – NUMBER(10)
10. Zipcode - NUMBER(9)

- **Order_Detail:**

1. Order_ID - VARCHAR2(10)
2. Cust_ID – NUMBER(10)
3. Order_Date – TIMESTAMP
4. Order_Status – VARCHAR2(25)

- **Menu:**

1. Food_ID - NUMBER(10)
2. Food_Name – VARCHAR2(150)

3. Food_Price – NUMBER(4,2)

- **Order_Item:**

1. Order_item_ID - NUMBER(10)
2. Food_ID - NUMBER(10)
3. Order_ID – VARCHAR2(10)
4. Quantity - NUMBER(2)

- **Delivery:**

1. Delivery_ID – VARCHAR2(10)
2. Order_ID - VARCHAR2(10)
3. Delivery_Datetime - TIMESTAMP
4. Delivery_Status – VARCHAR2(15)

Primary & Foreign Keys

- **Primary Keys:**

1. Customers: - Cust_ID
2. Menu: - Food_ID
3. Order_Detail: - Order_ID
4. Delivery: - Delivery_ID

5. Order_Item: - order_item_ID

- **Foreign Keys:**

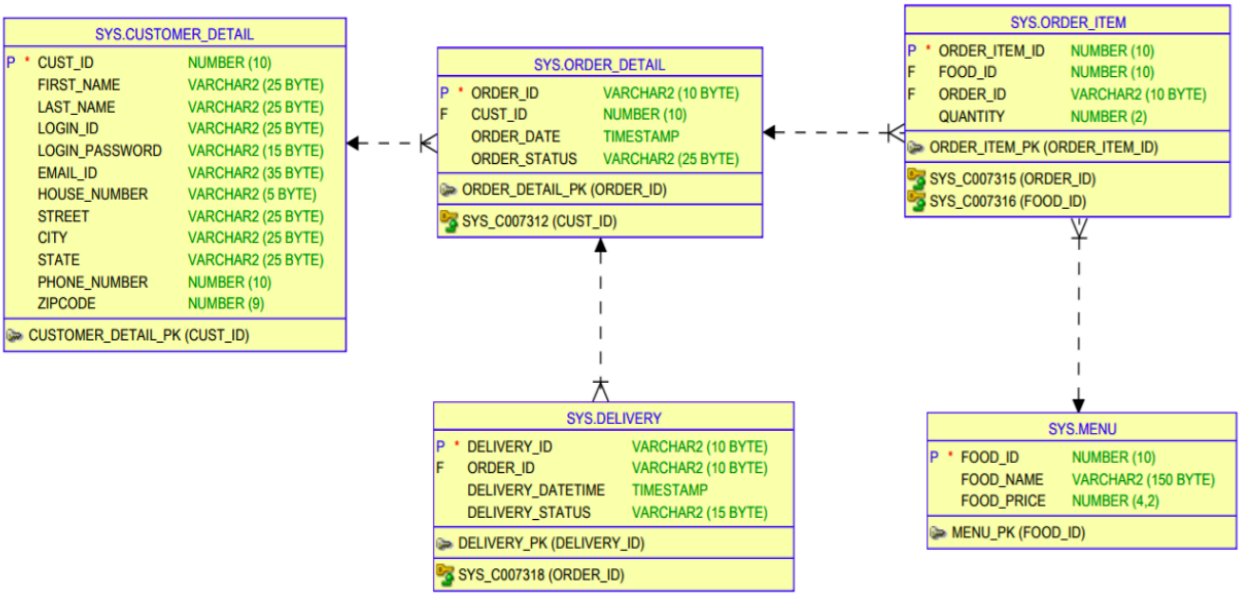
1. Order_Detail: - Cust_ID
2. Delivery: - order_ID
3. Order_item: - (Food_ID, order_ID)

Referential Integrity Constraints

The referential integrity constraints can be observed in the following attributes.

1. Cust_id from ORDER_DETAIL refers to cust_id of CUSTOMER_DETAIL.
2. Food_id from ORDER_ITEM refers to Food_id of MENU.
3. Order_id from ORDER_ITEM refers to order_id of ORDER_DETAIL.
4. Order_id from DELIVERY refers to order_id of ORDER_DETAIL

ER Model



In Entity-Relationship diagram (ERD), every relation between the entities are described.

- Customer --- Places --- Order

The customer can place several orders; the customer entity holds the attributes describing the properties of customer. The Order entity holds attributes concerning the details related to order such as Order date and time along with the status of the order.

- Order --- has --- Order item

An order has several order items; The Order item entity entails the attributes concerning the food items and the quantities of each of these food items.

- Menu--- Fetched from --- Order Item

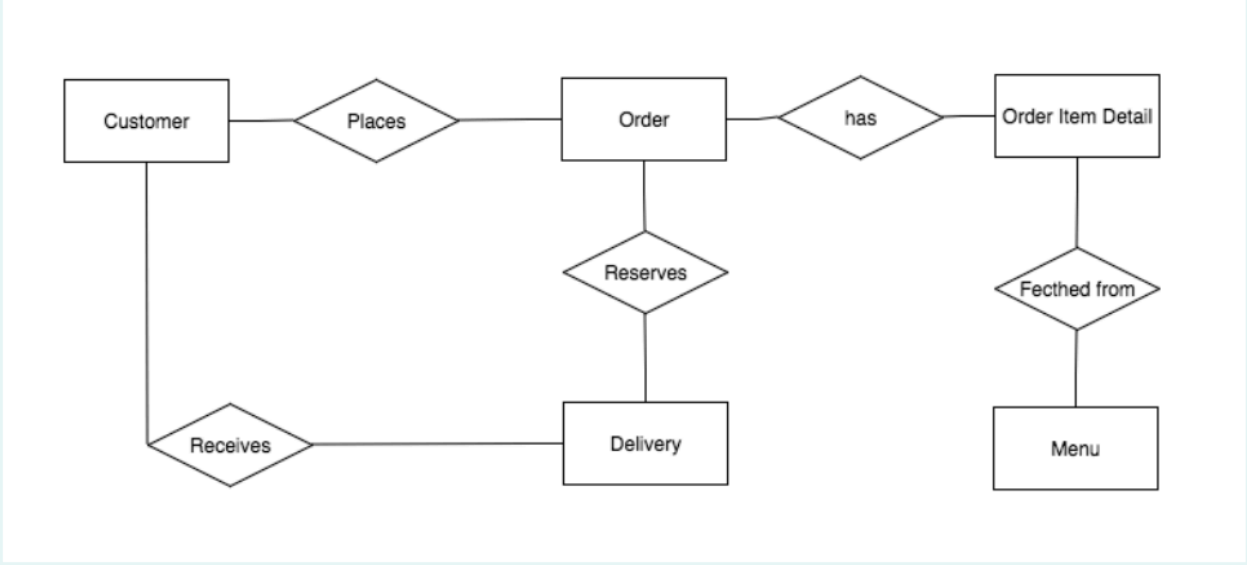
The Order Item is a compilation of zero, one or more food items. An order item contains the details of each food item and the quantity in which it has been ordered. All the food item details are being fetched from the Menu entity which has the information about the food item ID and food item description.

- Delivery --- Reserves --- Order

A delivery reserves the order and ever every order can have zero or multiple deliveries depending on the quantity of products. The delivery entity comprises delivery status and delivery time information.

Relational Model

The below diagram shows the relational schema of the database. It is derived from the Entity-Relationship diagram.



Creating Tables of FOOD DELIVERY Database

- **Customers Table**

```
select * from customer_detail;
```

Script Output | Query Result

SQL | All Rows Fetched: 10 in 0.004 seconds

	CUST_ID	FIRST_NAME	LAST_NAME	LOGIN_ID	LOGIN_PASSWORD	EMAIL_ID	HOUSE_NUMBER	STREET	CITY	STATE
1	1	Williams	Brown	WilliamsB	Hello@1234	williambrown@gmail.com	81S	WIND STREET	Stillwater	Oklahoma
2	2	Sophia	Smith	Sophsmi	Password123	sophia1999@gmail.com	81S	NORTH DUNKAN STREET	Stillwater	Oklahoma
3	3	Olia	Miller	Oliamill	December19!	chatwithme@yahoo.com	25w	STREET	Stillwater	Oklahoma
4	4	Emma	Johnson	Emmaooh	Hello865	emmaread@gmail.com	30T	WALL STREET	Stillwater	Oklahoma
5	5	Ava	Jones	jonesa	querty@1234	avajones888@gmail.com	89W	BROADWAY	NEW YORK	Oklahoma
6	6	Jackson	Davis	davidjack	3rjslla7qe	davidkacl23@gmail.com	70W	BOURBOURN STREET	Stillwater	Oklahoma
7	7	Liam	Williams	liwilliam	google	williamnora@gmail.com	101	LASVEGAS STREET	Stillwater	Oklahoma
8	8	Noah	Brown	brownnah	zxcvbnm	impeccablebrown@gmail.com	05O	Lombard STREET	Stillwater	Oklahoma
9	9	Aiden	Wilson.	aidenwils	zxcvbnm	wilsonaiden@gmail.com	92S	Rodeo STREET	Stillwater	Oklahoma
10	10	Logan	Hernandez	hernand	123qwe	hearnandezgreat@gmail.com	1010T	Sunset STREET	Stillwater	Oklahoma

- **Menu Table**

```
select * from menu;
```

Script Output × | Query Result ×

SQL | All Rows Fetched: 23 in 0.004 seconds

	FOOD_ID	FOOD_NAME	FOOD_PRICE
1	111111	Hakka Noodles	10.5
2	111112	Chow Mein	8
3	111113	Noodles Soup	12
4	111114	Fried Rice	14
5	111115	Braised Fork	15
6	111116	Chicken oriental salad	16
7	111117	Hot and Sour Soup	14
8	111118	Munchow	5
9	111119	Lemon pepper	10
10	111120	Thai Vegetable Red Curry Steamed Rice	10
11	111121	Thai Chicken Red Currry	10
12	111122	Spring Rolls - VEG	8
13	111123	Veg Dragon rolls	8
14	111124	Crispy peking baby corn	8
15	111125	Veg fried dumplings	8
16	111126	Crispy Crunchy Spinach - Veg	10
17	111127	Golden Fried Baby Corn	10
18	111128	Stir fried broccoli	12

- **Orders Table**

```
select * from order_detail;
```

Script Output | Query Result

SQL | All Rows Fetched: 24 in 0.01 seconds

	ORDER_ID	CUST_ID	ORDER_DATE	ORDER_STATUS
1	1111	1	10-OCT-17	Done
2	1112	2	10-OCT-17	Done
3	1113	3	10-DEC-17	Done
4	1114	5	13-OCT-17	Done
5	1115	6	13-OCT-17	Done
6	1116	4	13-OCT-17	Done
7	1117	7	13-OCT-17	Done
8	1118	8	14-OCT-17	Done
9	1119	9	14-OCT-17	Done
10	1120	10	14-OCT-17	Done
11	1121	10	14-OCT-17	Done
12	1122	10	14-OCT-17	Done
13	1123	2	15-OCT-17	Done
14	1124	1	15-OCT-17	Done
15	1125	3	15-OCT-17	Done
16	1126	4	15-OCT-17	Done
17	1127	6	15-OCT-17	Done
18	1128	6	15-OCT-17	Done

- **Delivery Table**

```
select * from delivery;
```

Script Output | Query Result

SQL | All Rows Fetched: 19 in 0.004 seconds

	DELIVERY_ID	ORDER_ID	DELIVERY_DATETIME	DELIVERY_STATUS
1	D1	1111	10-OCT-17 12.22.23.000000000 PM	Delivered
2	D2	1112	10-OCT-17 02.28.23.000000000 PM	Delivered
3	D3	1113	12-OCT-17 03.22.23.000000000 PM	Delivered
4	D4	1114	13-OCT-17 10.22.23.000000000 AM	Delivered
5	D5	1115	13-OCT-17 09.22.23.000000000 PM	Delivered
6	D6	1116	13-OCT-17 10.22.23.000000000 PM	Delivered
7	D7	1117	13-OCT-17 10.24.23.000000000 PM	Delivered
8	D8	1118	14-OCT-17 03.22.23.000000000 PM	Delivered
9	D9	1119	14-OCT-17 04.10.00.000000000 PM	Delivered
10	D10	1120	14-OCT-17 09.15.00.000000000 AM	Delivered
11	D11	1121	14-OCT-17 11.22.05.000000000 AM	Delivered
12	D12	1122	14-OCT-17 01.05.00.000000000 PM	Delivered
13	D13	1123	15-OCT-17 10.01.03.000000000 AM	Delivered
14	D14	1124	15-OCT-17 11.22.23.000000000 AM	Delivered
15	D15	1125	15-OCT-17 12.10.03.000000000 PM	Delivered
16	D16	1126	15-OCT-17 02.22.51.000000000 PM	On the way
17	D17	1127	15-OCT-17 04.22.23.000000000 PM	On the way
18	D18	1128	15-OCT-17 04.35.23.000000000 PM	On the way

- **Order Items Table**

```
select * from order_item;
```

Script Output | Query Result | SQL | All Rows Fetched: 25 in 0.003 seconds

	ORDER_ITEM_ID	FOOD_ID	ORDER_ID	QUANTITY
1	100	111111	1134	1
2	101	111112	1133	2
3	102	111113	1133	1
4	103	111114	1132	2
5	104	111115	1131	5
6	105	111116	1130	1
7	106	111117	1129	1
8	107	111118	1128	2
9	108	111119	1127	4
10	109	111129	1126	6
11	110	111125	1125	1
12	111	111118	1124	2
13	112	111124	1123	1
14	113	111132	1122	4
15	114	111125	1121	1
16	115	111112	1120	2
17	116	111112	1119	3
18	117	111118	1118	1

Query Operations on Database

- **Display all orders that are "In Progress" Status.**

```
select * from order_detail
where order_status='In Progress';
```

Script Output | Query Result

SQL | All Rows Fetched: 5 in 0.001 seconds

	ORDER_ID	CUST_ID	ORDER_DATE	ORDER_STATUS
1	1130	1	15-OCT-17	In Progress
2	1131	8	15-OCT-17	In Progress
3	1132	1	15-OCT-17	In Progress
4	1133	9	15-OCT-17	In Progress
5	1134	10	15-OCT-17	In Progress

- **Number of orders placed by each customer.**

```
select customer_detail.cust_id,first_name,last_name, count(order_id) AS "Number of Orders"
from customer_detail,order_detail
where customer_detail.cust_id=order_detail.cust_id
group by customer_detail.cust_id,first_name,last_name;
```

Script Output | Query Result

SQL | All Rows Fetched: 10 in 0.004 seconds

	CUST_ID	FIRST_NAME	LAST_NAME	Number of Orders
1	5	Ava	Jones	1
2	10	Logan	Hernandez	4
3	8	Noah	Brown	2
4	4	Emma	Johnson	2
5	7	Liam	Williams	2
6	9	Aiden	Wilson.	2
7	1	Williams	Brown	4
8	2	Sophia	Smith	2
9	3	Olia	Miller	2
10	6	Jackson	Davis	3

- **Increase the price of Fried Rice by 20 percent.**

```
select * from menu;
```

Script Output | Query Result

SQL | All Rows Fetched: 23 in 0.001 seconds

	FOOD_ID	FOOD_NAME	FOOD_PRICE
1	111111	Hakka Noodles	10.5
2	111112	Chow Mein	8
3	111113	Noodles Soup	12
4	111114	Fried Rice	14
5	111115	Braised Fork	15
6	111116	Chicken oriental salad	16
7	111117	Hot and Sour Soup	14
8	111118	Munchow	5
9	111119	Lemon pepper	10
10	111120	Thai Vegetable Red Curry Steamed Rice	10
11	111121	Thai Chicken Red Curry	10
12	111122	Spring Rolls - VEG	8
13	111123	Veg Dragon rolls	8
14	111124	Crispy peking baby corn	8

```
update menu set food_price=1.2*(food_price)
where food_name='Fried Rice';

select * from menu;
```

Script Output | Query Result | Query Result 1 | Query Result 2

SQL | All Rows Fetched: 23 in 0.001 seconds

	FOOD_ID	FOOD_NAME	FOOD_PRICE
1	111111	Hakka Noodles	10.5
2	111112	Chow Mein	8
3	111113	Noodles Soup	12
4	111114	Fried Rice	16.8
5	111115	Braised Fork	15
6	111116	Chicken oriental salad	16
7	111117	Hot and Sour Soup	14
8	111118	Munchow	5
9	111119	Lemon pepper	10
10	111120	Thai Vegetable Red Curry Steamed Rice	10
11	111121	Thai Chicken Red Currry	10
12	111122	Spring Rolls - VEG	8
13	111123	Veg Dragon rolls	8
14	111124	Crispy peking baby corn	8

- **What is the delivery status of the customer who stays in "Rodeo Street"?**

```
select x.cust_id, first_name,delivery_status
from customer_detail x, order_detail y, delivery z
where x.cust_id=y.cust_id AND y.order_id=z.order_id
AND street='Rodeo STREET';
```

Script Output | Query Result | Query Result 1 | Query Result 2

SQL | All Rows Fetched: 1 in 0.005 seconds

	CUST_ID	FIRST_NAME	DELIVERY_STATUS
1	9	Aiden	Delivered

- **Below view will give us the total order amount.**

```
CREATE VIEW TOTAL_ORDER_AMOUNT AS
SELECT OD.ORDER_ID,M.FOOD_PRICE, OT.QUANTITY, round((FOOD_PRICE*QUANTITY),2) AS ORDER_AMOUNT
FROM MENU M, ORDER_ITEM OT ,ORDER_DETAIL OD
WHERE OD.ORDER_ID= OT.ORDER_ID AND OT.FOOD_ID=M.FOOD_ID;

SELECT * FROM TOTAL_ORDER_AMOUNT;
```

Script Output | Query Result | Query Result 1 | Query Result 2 | Query Result 3

SQL | All Rows Fetched: 25 in 0.007 seconds

	ORDER_ID	FOOD_PRICE	QUANTITY	ORDER_AMOUNT
1	1134	10.5	1	10.5
2	1133	8	2	16
3	1133	12	1	12
4	1132	16.8	2	33.6
5	1131	15	5	75
6	1130	16	1	16
7	1129	14	1	14
8	1128	5	2	10
9	1127	10	4	40
10	1126	15	6	90
11	1125	8	1	8
12	1124	5	2	10
13	1123	8	1	8
14	1122	10	4	40

- **Below view will give the Total sale for a day.**

```
CREATE VIEW TOTAL_SALE_OF_DAY AS
SELECT CAST(OD.ORDER_DATE AS DATE) AS ORDER_DATE ,
round((SUM(FOOD_PRICE*QUANTITY)),2) AS TOTAL_SALE_OF_DAY
FROM MENU M, ORDER_ITEM OT ,ORDER_DETAIL OD
WHERE OD.ORDER_ID= OT.ORDER_ID AND OT.FOOD_ID=M.FOOD_ID group by OD.ORDER_DATE
ORDER BY ORDER_DATE;

SELECT * FROM TOTAL_SALE_OF_DAY;
```

Script Output | Query Result | Query Result 1 | Query Result 2 | Query Result 3

SQL | All Rows Fetched: 24 in 0.007 seconds

	ORDER_DATE	TOTAL_SALE_OF_DAY
1	10-OCT-17	8
2	10-OCT-17	16
3	13-OCT-17	30
4	13-OCT-17	20
5	13-OCT-17	10
6	13-OCT-17	20
7	14-OCT-17	16
8	14-OCT-17	8
9	14-OCT-17	40
10	14-OCT-17	5
11	14-OCT-17	24
12	15-OCT-17	8
13	15-OCT-17	8

- **Below view will give us the delivered order details.**

```
CREATE VIEW DELIVERED_ORDERS AS
SELECT D.ORDER_ID,D.DELIVERY_STATUS,CAST(OD.ORDER_DATE AS DATE) AS ORDER_DATE
FROM DELIVERY D, ORDER_DETAIL OD
WHERE D.DELIVERY_STATUS= 'Delivered' and OD.ORDER_ID= D.ORDER_ID
ORDER BY ORDER_ID, ORDER_DATE ;

select * from delivered_orders;
```

Script Output | Query Result | Query Result 1 | Query Result 2 | Query Result 3

SQL | All Rows Fetched: 15 in 0.005 seconds

	ORDER_ID	DELIVERY_STATUS	ORDER_DATE
1	1111	Delivered	10-OCT-17
2	1112	Delivered	10-OCT-17
3	1113	Delivered	10-DEC-17
4	1114	Delivered	13-OCT-17
5	1115	Delivered	13-OCT-17
6	1116	Delivered	13-OCT-17
7	1117	Delivered	13-OCT-17
8	1118	Delivered	14-OCT-17
9	1119	Delivered	14-OCT-17
10	1120	Delivered	14-OCT-17
11	1121	Delivered	14-OCT-17
12	1122	Delivered	14-OCT-17
13	1123	Delivered	15-OCT-17

- **Below view will give us "on the way" order details.**

```
CREATE VIEW ON_THE_WAY_ORDERS AS
SELECT D.ORDER_ID,D.DELIVERY_STATUS,CAST(OD.ORDER_DATE AS DATE) AS ORDER_DATE
FROM DELIVERY D, ORDER_DETAIL OD
WHERE D.DELIVERY_STATUS= 'On the way' and OD.ORDER_ID= D.ORDER_ID
ORDER BY ORDER_ID, ORDER_DATE ;

SELECT * FROM ON_THE_WAY_ORDERS;
```

Script Output | Query Result | Query Result 1 | Query Result 2 | Query Result 3

SQL | All Rows Fetched: 4 in 0.331 seconds

	ORDER_ID	DELIVERY_STATUS	ORDER_DATE
1	1126	On the way	15-OCT-17
2	1127	On the way	15-OCT-17
3	1128	On the way	15-OCT-17
4	1129	On the way	15-OCT-17

- **Below view will give us in-progress order details.**

```
CREATE VIEW INPROGRESS_ORDERS AS
SELECT OD.ORDER_ID,OD.CUST_ID,M.FOOD_NAME, OD.ORDER_STATUS
FROM MENU M, ORDER_ITEM OT ,ORDER_DETAIL OD
WHERE OD.ORDER_ID= OT.ORDER_ID AND OT.FOOD_ID=M.FOOD_ID AND OD.ORDER_STATUS='In Progress'
ORDER BY ORDER_ID,CUST_ID;

SELECT * FROM INPROGRESS_ORDERS;
```

Script Output | Query Result | Query Result 1 | Query Result 2 | Query Result 3 | Query Result 4

SQL | All Rows Fetched: 6 in 0.011 seconds

	ORDER_ID	CUST_ID	FOOD_NAME	ORDER_STATUS
1	1130	1	Chicken oriental salad	In Progress
2	1131	8	Braised Fork	In Progress
3	1132	1	Fried Rice	In Progress
4	1133	9	Chow Mein	In Progress
5	1133	9	Noodles Soup	In Progress
6	1134	10	Hakka Noodles	In Progress

- **Below view will give us details about number of times each customer ordered food**

```
CREATE VIEW ORDER_COUNT AS
SELECT COUNT(OD.ORDER_ID) AS ORDER_COUNT,C.CUST_ID, C.FIRST_NAME, C.LAST_NAME,C.EMAIL_ID, C.PHONE_NUMBER, C.HOUSE_NUMBER,C.STREET,
FROM CUSTOMER_DETAIL C, ORDER_DETAIL OD
WHERE OD.CUST_ID= C.CUST_ID
GROUP BY C.CUST_ID, C.FIRST_NAME, C.LAST_NAME, C.CUST_ID, C.EMAIL_ID,
C.PHONE_NUMBER, C.HOUSE_NUMBER, C.STREET, C.CITY, C.STATE,
C.ZIPCODE
ORDER BY ORDER_COUNT DESC;

SELECT * FROM ORDER_COUNT
```

Script Output | Query Result | Query Result 1 | Query Result 2 | Query Result 3 | Query Result 4 | Query Result 5

SQL | All Rows Fetched: 10 in 0.013 seconds

	ORDER_COUNT	CUST_ID	FIRST_NAME	LAST_NAME	EMAIL_ID	PHONE_NUMBER	HOUSE_NUMBER	STREET	CITY
1	4	1	Williams	Brown	williambrown@gmail.com	4056125824	81S	WIND STREET	Stillwater
2	4	10	Logan	Hernandez	hearnandezgreat@gmail.com	4056123333	1010T	Sunset STREET	Stillwater
3	3	6	Jackson	Davis	davidkacl23@gmail.com	4056635865	70W	BOURBOURN STREET	Stillwater
4	2	8	Noah	Brown	impeccablebrown@gmail.com	2056125824	05O	Lombard STREET	Stillwater
5	2	4	Emma	Johnson	emmaread@gmail.com	4056125850	30T	WALL STREET	Stillwater
6	2	9	Aiden	Wilson.	wilsonaiden@gmail.com	4056121111	92S	Rodeo STREET	Stillwater
7	2	2	Sophia	Smith	sophia1999@gmail.com	4056705824	81S	NORTH DUNKAN STREET	Stillwater
8	2	7	Liam	Williams	williamnora@gmail.com	4056635999	101	LASVEGAS STREET	Stillwater
9	2	3	Olia	Miller	chatwithme@yahoo.com	4056236724	25w	STREET	Stillwater
10	1	5	Ava	Jones	avajones888@gmail.com	4056635850	89W	BROADWAY	NEW YORK

Applications

Administrator side database is assimilated into every company that runs on the principle of Food delivery systems like Panda Express, Tapingo or dealing with customers.

Conclusion

References

- https://www.oracle.com/downloads/index.html
- https://www.w3schools.com/sql/DEfaULT.asP
- https://www.tutorialspoint.com/sql/