1.

Company Name – Uber Eats

Website – www.ubereats.com

Main business – An online platform that links customers with the restaurant of their choosing for online meal delivery.

Description - In 2014, Uber introduced its online meal ordering and delivery service, Uber Eats. Meal couriers can deliver meals on foot, by bicycle, scooter, or automobile. By 2021, it will run in over 6,000 cities across 45 nations. To increase their customer base and provide delivery services, restaurants collaborate with Uber Eats.

The process of delivering food is carried out by Uber drivers, and they sometimes just require a short amount of time to complete a single delivery. Drivers with UberEats collect food from restaurants chosen by the customers and deliver their orders. UberEats takes care of all the meal orders, so the driver only needs to enter the restaurant, pick up the food, and then go to the customer's location. The driver may deliver the order to the door or curbside. In locations where Uber Eats is available, Uber will regularly work with several restaurants to provide meals to its customers.

Scenario – a customer opens the Ubereats app, and logs in to their account. They then search for the restaurants nearby in terms of their food price and ratings. They browse and choose a restaurant menu and order food by entering their delivery address. The customer reviews the order and total amount and confirms it. The Uber Eats app generates a new order, assigns a unique OrderID, and links it with the CustomerID. Afterward, it also assigns a driver (DriverID) for the order, The available drivers will accept the order and update the order status as the driver assigned. The order will be timestamped. the restaurant will receive the order and starts preparing it. Then informs the driver once the order is prepared. The driver picks up the order and uses the navigation tools provided by the app to reach the location of the customer. Once delivered, the driver marks it as “Order Delivered” The uber eats charges the customer’s preferred payment method for the total order amount, the customer can choose to pay cash or card in the beginning itself. Finally, the customer receives an order completion notification and has the option to rate the driver and restaurant. The order status in the uber eats app changes to “Completed”.

The drivers, customers, restaurants, and vehicles are the main use cases involved in this.

2.

Customer [Customer_ID (PK), Name, Email, Phone, Address]

Restaurant [Restaurant_ID (PK), Name, Cuisine, Email, Phone, Address, Rating]

Driver [Driver_ID (PK), Name, License, Vehicle_Number, Vehicle_type, Phone, Number_of_orders_completed, Rating]

DriverVehicle [DriverVehicle_ID(PK), Driver_ID(FK), Vehicle_ID(FK), StartDate, EndDate]

Vehicle [Vehicle (PK), Vehicle_Number, Vehicle_Type]

Order_Invoice [Order_ID (PK), Item_Ordered_ID (FK), Customer_ID (FK), Restaurant_ID (FK), Driver_ID (FK), Payment_ID(FK), Order_Date, Amount_Paid]

Menu [Menu_Item_ID (PK), Restaurant_ID (FK), Menu_Item_Name, Menu_Item_Description, Menu_Item_Price]

Item_Ordered [Item_Ordered_ID (PK), Menu_Item_ID (FK), Quantity]

Payment [Payment_ID(PK), Payment_Method (FK), Customer_ID(FK)]

3.

Customer entity

- 1NF – satisfies it, because there are no repeating groups, all the attributes (Customer_ID (PK), Name, Email, Phone, Address) contain atomic values.
- 2NF – satisfies it because there is no partial dependency, all of the non-key attributes (Name, Email, Phone, Address) are directly connected to one primary key (Customer_ID).
- 3NF – satisfies it because there is no transitive dependency, none of the non-key attributes (Name, Email, Phone, Address) have a dependency on each other. They are all directly dependent on the PK (Customer_ID). There is no indirect relationship among these attributes.

Restaurant entity

- 1NF – satisfies it, because there are no repeating groups, all the attributes (Restaurant_ID (PK), Name, Cuisine, Email, Phone, Address, Rating) contain atomic values.
- 2NF – satisfies it because there is no partial dependency, all of the non-key attributes (Name, Cuisine, Email, Phone, Address, Rating) are directly connected to one primary key (Restaurant_ID).
- 3NF – satisfies it because there is no transitive dependency, none of the non-key attributes (Name, Email, Phone, Address) have a dependency on each other. They are all directly dependent on the PK (Customer_ID). There is no indirect relationship among these attributes.

Driver entity

- 1NF – satisfies it, because there are no repeating groups, all the attributes (Driver_ID (PK), Name, License, Vehicle_Number, Vehicle_type, Phone, Number_of_orders_completed, rating) contain atomic values.
- 2NF – satisfies it because there is no partial dependency, all of the non-key attributes (Name, License, Vehicle_Number, Vehicle_type, Phone, Number_of_orders_completed, rating) are directly connected to one primary key (Driver_ID).
- 3NF – satisfies it because there is no transitive dependency, none of the non-key attributes (Name, License, Vehicle_Number, Vehicle_type, Phone, Number_of_orders_completed) have a dependency on each other. They are all directly dependent on the PK (Driver_ID). There is no indirect relationship among these attributes.

DriverVehicle entity

- 1NF – satisfies it, because there are no repeating groups, all the attributes (DriverVehicle_ID(PK), Driver_ID(FK), Vehicle_ID(FK), StartDate, EndDate) contain atomic values
- 2NF – satisfies it because there is no partial dependency, all of the non-key attributes (StartDate, EndDate) are directly connected to one primary key (DriverVehicle_ID).
- 3NF – satisfies it because there is no transitive dependency, none of the non-key attributes (StartDate, EndDate) have a dependency on each other. They are all directly dependent on the PK (DriverVehicle_ID). There is no indirect relationship among these attributes.

Vehicle entity

- 1NF – satisfies it, because there are no repeating groups, all the attributes (Vehicle (PK), Vehicle_Number, Vehicle_Type) contain atomic values
- 2NF – satisfies it because there is no partial dependency, all of the non-key attributes (Vehicle_Number, Vehicle_Type) are directly connected to one primary key (Vehicle_ID).
- 3NF – satisfies it because there is no transitive dependency, none of the non-key attributes (Vehicle_Number, Vehicle_Type) have a dependency on each other. They are all directly dependent on the PK (Vehicle_ID). There is no indirect relationship among these attributes.

Order_Invoice entity

- 1NF – satisfies it, because there are no repeating groups, all the attributes (Order_ID (PK), Item_Ordered_ID (FK), Customer_ID (FK), Restaurant_ID (FK), Driver_ID (FK), Payment_ID(FK), Order_Date, Amount_Paid) contain atomic values.
- 2NF – satisfies it because there is no partial dependency, all of the non-key attributes (Order_Date, Amount_Paid) are directly connected to one primary key (Order_ID(PK), Item_Ordered_ID (FK), Customer_ID (FK), Restaurant_ID (FK), Driver_ID (FK)) are all attributes that connect the other entities.
- 3NF – satisfies it because there is no transitive dependency, none of the non-key attributes (Order_Date, Amount_Paid) have a dependency on each other. Item_Ordered_ID (FK), Customer_ID (FK), Restaurant_ID (FK), and Driver_ID (FK) are all foreign keys related to the primary key of their respective entities. They are all directly dependent on the PK (Order_ID). There is no indirect relationship among these attributes.

Menu entity

- 1NF – satisfies it, because there are no repeating groups, all the attributes (Menu_Item_ID (PK), Restaurant_ID (FK), Menu_Item_Name, Menu_Item_Description, Menu_Item_Price) contain atomic values.
- 2NF – satisfies it because there is no partial dependency, all of the non-key attributes (Menu_Item_Name, Menu_Item_Description, Menu_Item_Price) are directly connected to one primary key (Menu_Item_ID).
- 3NF – satisfies it because there is no transitive dependency, none of the non-key attributes (Menu_Item_Name, Menu_Item_Description, Menu_Item_Price) have a dependency on each other. They are all directly dependent on the PK (Menu_Item_ID). The foreign key (Restaurant_ID (FK)) is related to the respective table it belongs. There is no indirect relationship among these attributes.

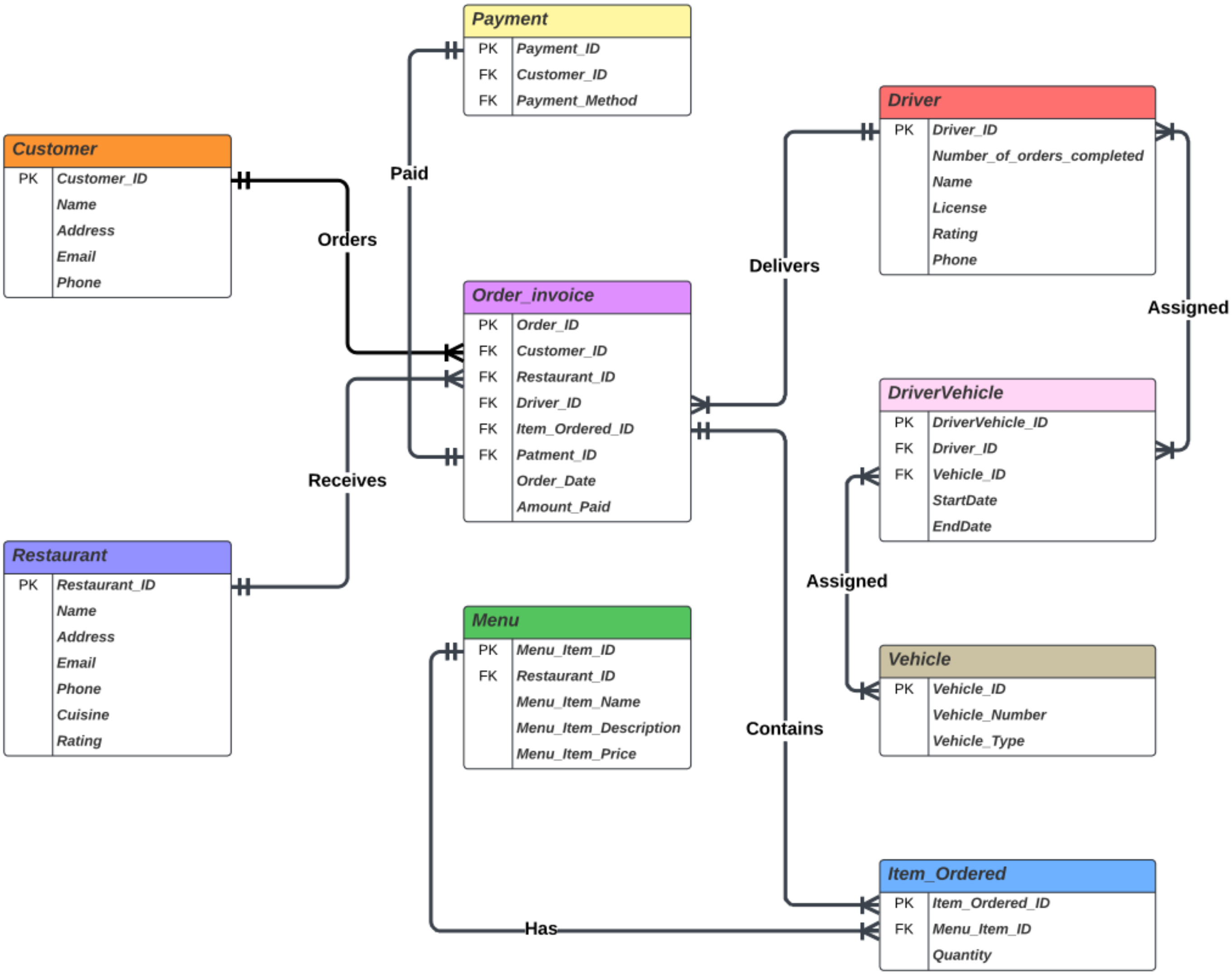
ItemOrdered entity

- 1NF – satisfies it, because there are no repeating groups, all the attributes (Item_Ordered_ID (PK), Menu_Item_ID (FK), Quantity) contain atomic values.
- 2NF – satisfies it because there is no partial dependency, the non-key attribute (Quantity) is directly connected to one primary key (Item_Ordered_ID).
- 3NF – satisfies it because there is no transitive dependency, the non-key attributes (Quantity) don't have a dependency. It is directly dependent on the PK (Item_Ordered_ID).

Payment entity

- 1NF – satisfies it, because there are no repeating groups, all the attributes (Payment_ID(PK), Payment_Method (FK), Customer_ID(FK)) contain atomic values.
- 2NF – satisfies it because there is no partial dependency, there are no non-key attributes in this entity so no partial dependency, and both the foreign keys (Payment_Method (FK), Customer_ID(FK))) are directly dependent on PK.
- 3NF – satisfies it because there is no transitive dependency, there are no non-key attributes in this entity so no chance of having a transitive dependency. The table is now in 3NF.

4.


Payment
PK Payment_ID
FK Customer_ID
FK Payment_Method
Customer
PK Customer_ID
Name
Address
Email
Phone
Driver
PK Driver_ID
Number_of_orders_completed
Name
License
Rating
Phone
Paid
Orders
Delivers
Assigned
Order_invoice
PK Order_ID
FK Customer_ID
FK Restaurant_ID
FK Driver_ID
FK Item_Ordered_ID
FK Patment_ID
Order_Date
Amount_Paid
DriverVehicle
PK DriverVehicle_ID
FK Driver_ID
FK Vehicle_ID
StartDate
EndDate
Receives
Restaurant
PK Restaurant_ID
Name
Address
Email
Phone
Cuisine
Rating
Assigned
Menu
PK Menu_Item_ID
FK Restaurant_ID
Menu_Item_Name
Menu_Item_Description
Menu_Item_Price
Vehicle
PK Vehicle_ID
Vehicle_Number
Vehicle_Type
Contains
Item_Ordered
PK Item_Ordered_ID
FK Menu_Item_ID
Quantity
Has