Oracle SQL Developer to use https://www.oracle.com/database/sqldeveloper/technologies/download/

Oracle Data Modeler to use https://www.oracle.com/database/sqldeveloper/technologies/sql-data-modeler/

Propose a case study like the cases from the required textbooks. The development process should follow the typical SDLC process:

- Planning
- Analysis
- Logical Design
- Physical Design
- Implementation
- Maintenance (excluded from the project)

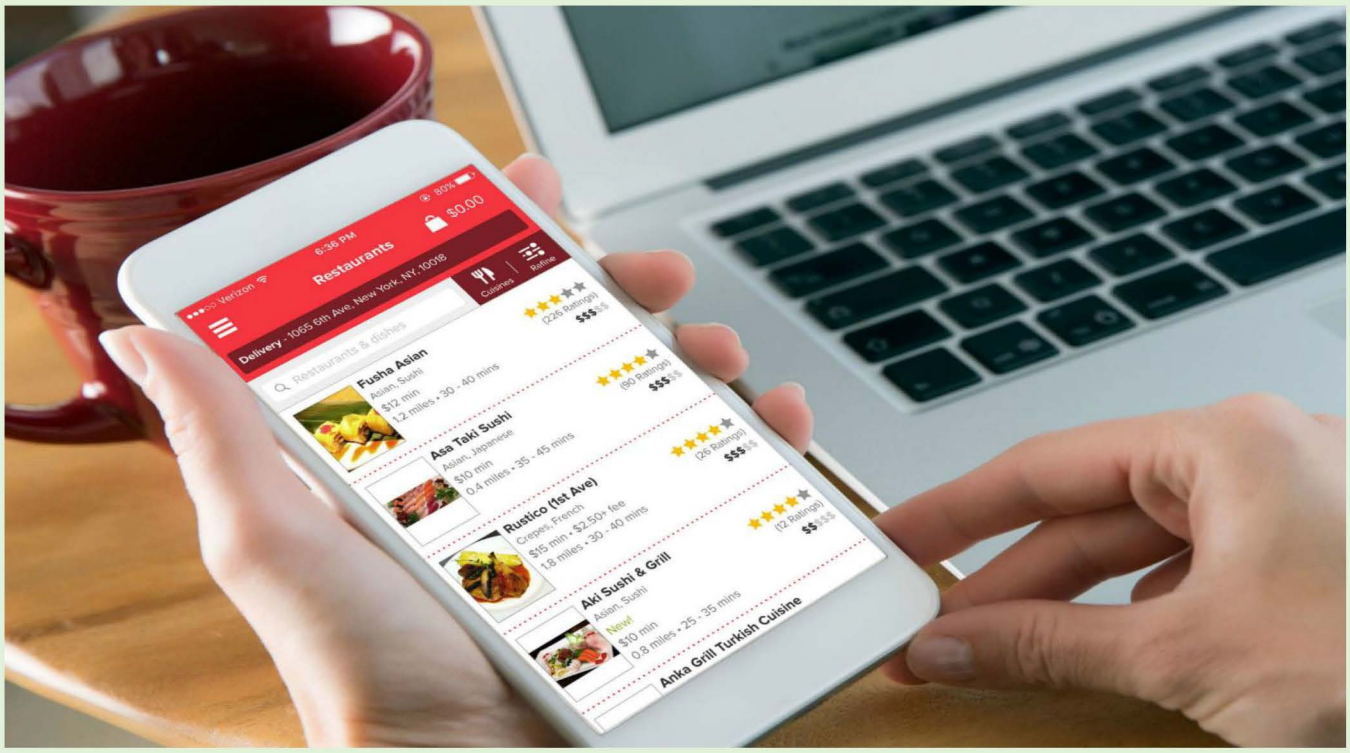
The final submission should include the following components as developed and refined throughout the semester:

1. **Project Planning (10%):** description of the business, mission, information systems architecture, and database problems to solve (objectives)
2. **Analysis Documents (20%):** process decomposition or workflows, business function-to-data entity matrix, business rules, and a high-level conceptual model (enterprise ERD) developed with Oracle Data Modeler
3. **Logical Design (20%):** Detailed ERD or enhanced ERD, normalized to the third normal form or higher (minimum 10 entities)
4. **Physical Design (20%):** Refined physical model with appropriate field data types; add appropriate indexes and check constraints; generate DDL and create the physical database using Oracle SQL Developer *https://www.oracle.com/database/sqldeveloper/technologies/download/*
5. **Implementation and Testing (20%):** Create script to load test data; create views, stored procedures, functions, and triggers (minimum 8 code objects) to use the database; test database and code thoroughly.
6. **Presentation (10%):** Prepare a separate 10-minimum PowerPoint presentation and give an oral presentation at the last class.
7. Show picture of Logical and Relational of database using Oracle Data Modeler
8. **Final Submission:** Document the above project requirements in a Word document.

SEE SAMPLE PROJECT and POWERPOINT BELOW (DO NOT COPY)

YOUR DELIVERY PARTNER

Course	
Title	Go-Food Food Delivery
SOP Version	V 1.0
Last Updated on	
By	



GO-Food Food Delivery Business Project

Technology Description:

This is a database for a website where customers can place order for food from different restaurants near by or customer also can place food order for their friends and family as well from any location but must be in the same country. Administrator can add or remove any customers as well as employee based on their performance. For example, if a restaurant failed to provide best service to the customer includes, maintain food quality, delivering on time, packing, provide necessary utensils/plates/cups etc., administrator can remove the restaurant from contact. Also if any customer tries to commit any fraudulent activity, such as after receiving food complain as never received food, misbehaving with the delivery person, using stolen credit card, customers account can be banned. Also for the delivery drivers, if not follow the company guidelines, might get banned from accessing the app.

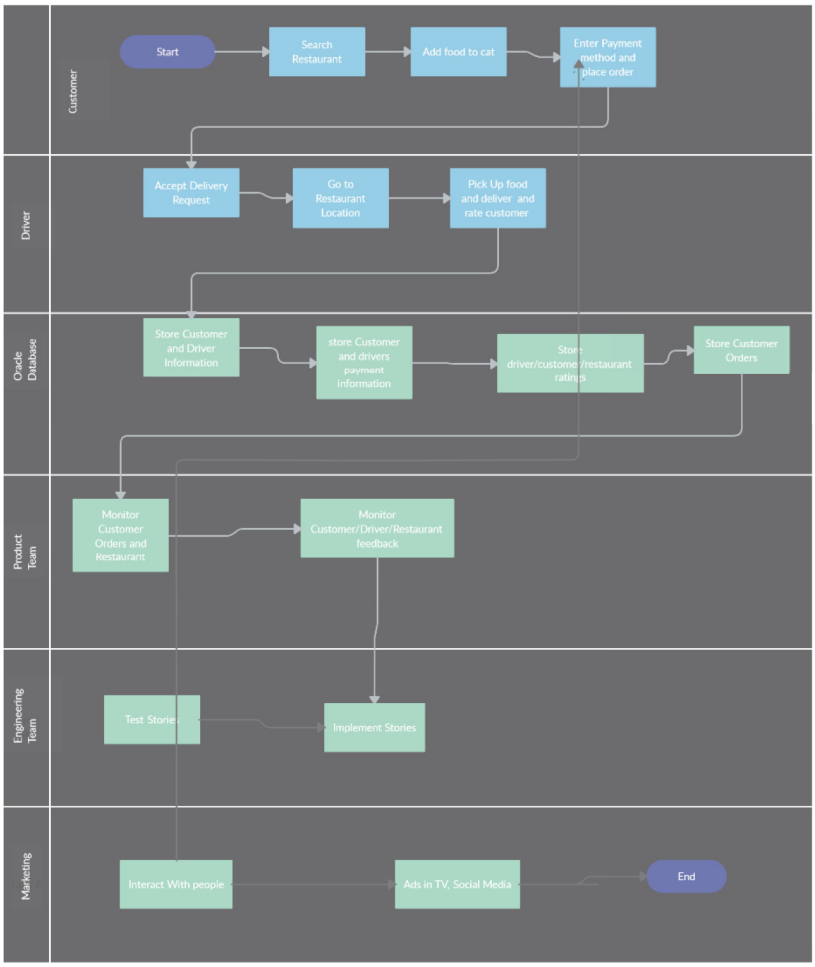
Problem Statement:

There needs to be a website as well as app where customer can place order online, browse their favorite restaurant , add favorite food to their cart and write reviews about their orders/ restaurants/delivery driver. Also an app for the drivers to receive food order, as well as rate customers and restaurants. Also there needs to be a secured database where customers and drivers can store their credit card information for transactions.

Technology Process Analysis:

No.	Position	Description
1	Drivers	Receive customer orders through app, pickup food from restaurant, and deliver it to the customer location
2	Customers	Can place order through app and website and track their orders
3	Website Administrator	Maintain the website for to make reliable for customers and making sure website is working based on the expectation
4	Support team	Teams that listen to customer and drivers complain and concerns about the order
5	Product team	Teams that add more popular restaurant and keep track of the most likable food by the customer and suggest it to the popular list
6	Marketing team	Team that is in charge of advertising about the GO-Food company, encourage people to use our service
7	Engineering Team	Team implements requirements based on the customer and delivery agent demands as well as restaurant needs
8	Oracle database	Information about the customer, delivery personnel, restaurant , credit card information will be store in the database.

Swimline:

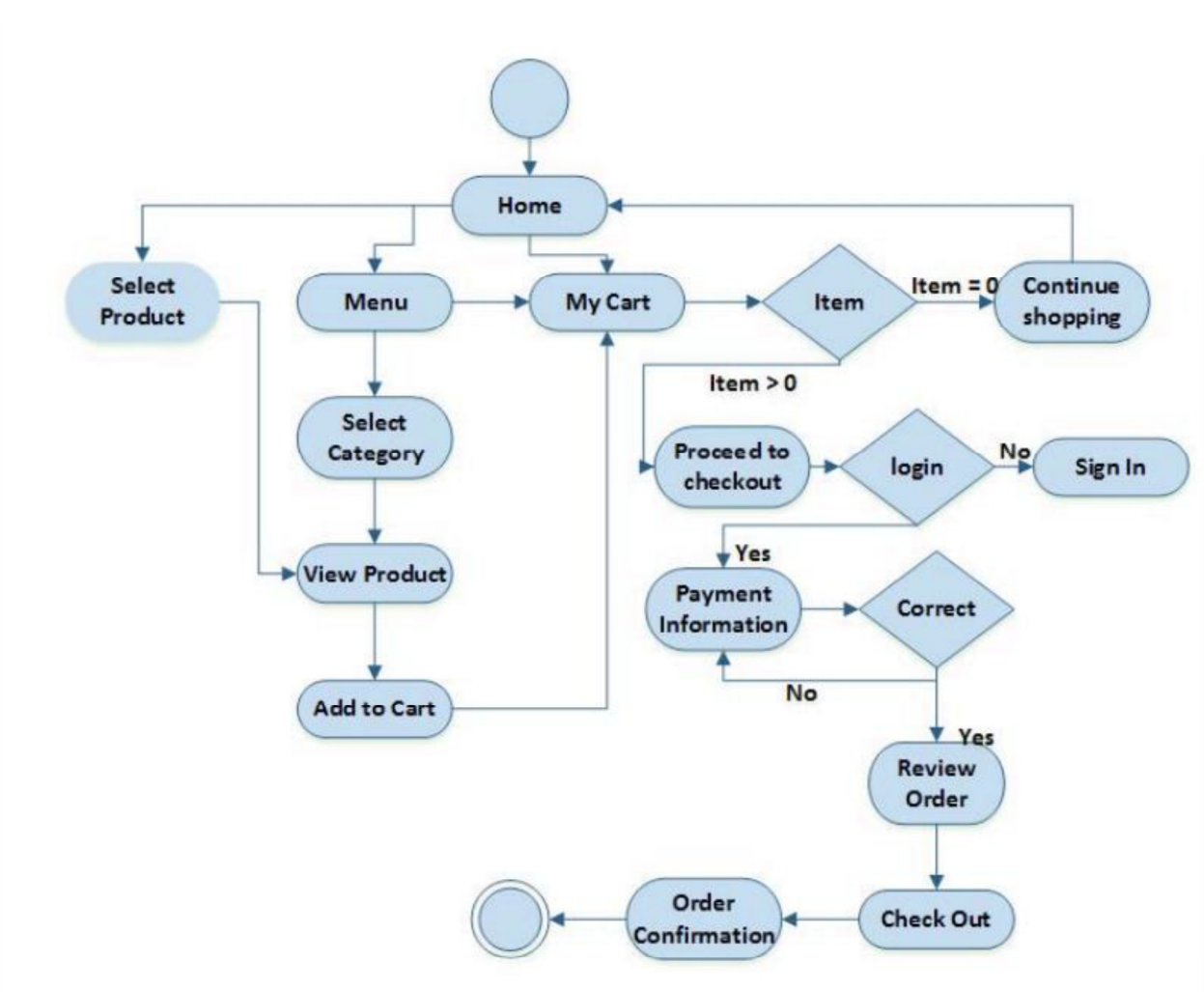

Customer
Start
Search Restaurant
Add food to cat
Enter Payment method and place order
Driver
Accept Delivery Request
Go to Restaurant Location
Pick Up food and deliver and rate customer
Oracle Database
Store Customer and Driver Information
store Customer and drivers payment information
Store driver/customer/restaurant ratings
Store Customer Orders
Product Team
Monitor Customer Orders and Restaurant
Monitor Customer/Driver/Restaurant feedback
Engineering Team
Test Stories
Implement Stories
Marketing
Interact With people
Ads in TV, Social Media
End

Requirements:

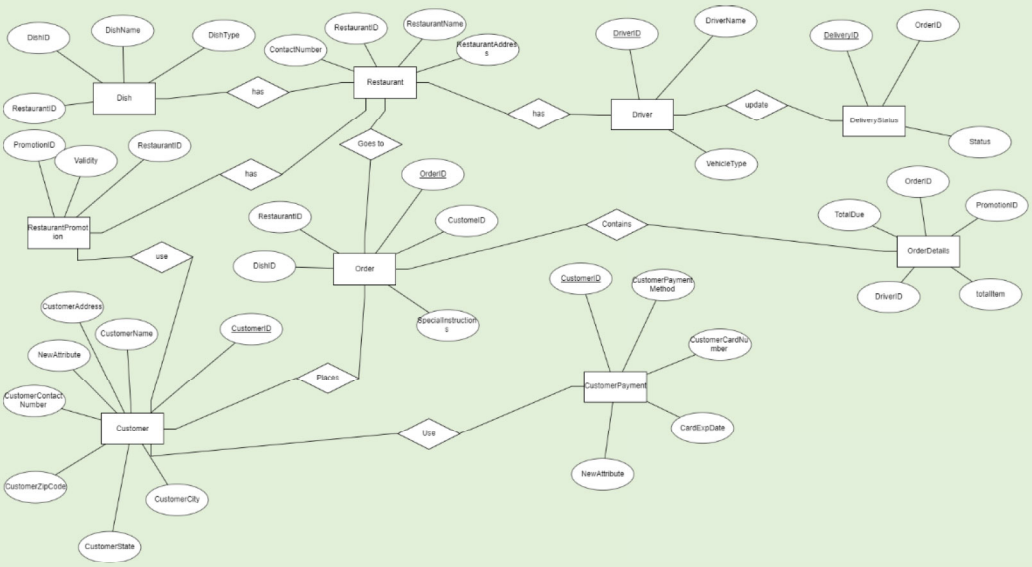
No	Functionality
1	Login for Customers/Drivers
2	Log out for Customers/Drivers
3	Register Customers/Drivers
4	Payment Processing
5	Storing Customer/ Drivers/ Restaurant ratings
6	Validating Complain
7	Display customers total orders
8	Filter by Highly Rated Restaurant
9	Search By Restaurant name
10	Search by food type
11	Selecting Highly rated drivers

Database Design

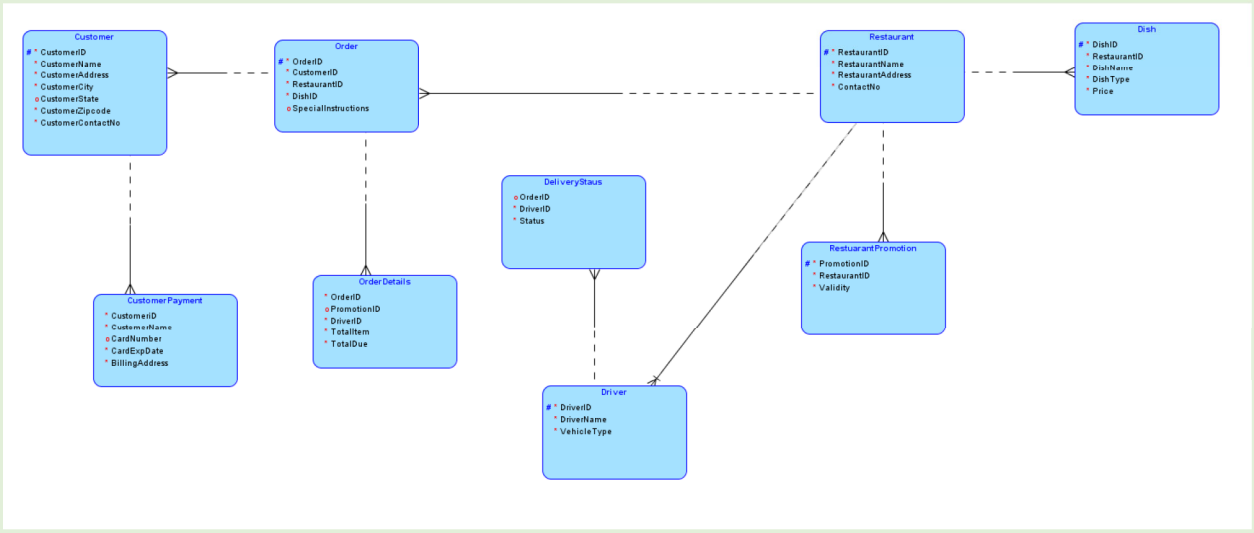
Activity Diagram:



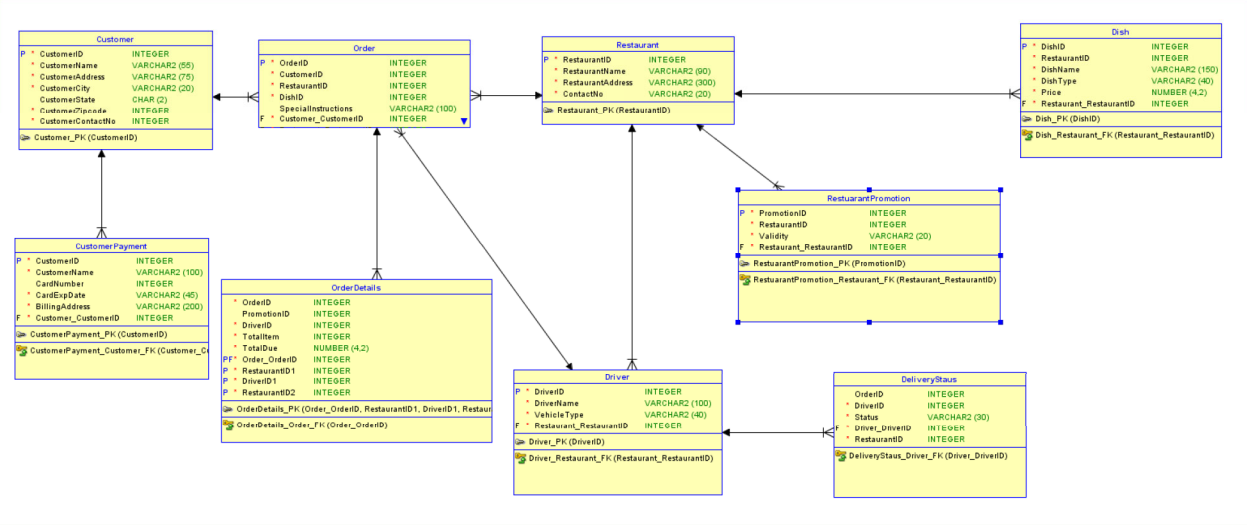
ER-Diagram:



Logical Design:



Relational Design:



Database Code

```
CREATE TABLE customers (
    customerid        INTEGER NOT NULL,
    customername      VARCHAR2(55) NOT NULL,
    customeraddress   VARCHAR2(75) NOT NULL,
    customercity      VARCHAR2(20) NOT NULL,
    customerstate     CHAR(2),
    customerzipcode   INTEGER NOT NULL,
    customercontactno  INTEGER NOT NULL
);
```

```
CREATE TABLE customerpayments (
    customerid          INTEGER NOT NULL,
    customername        VARCHAR2(100) NOT NULL,
    cardnumber          INTEGER,
    cardexpdate         VARCHAR2(45) NOT NULL,
    billingaddress      VARCHAR2(200) NOT NULL
);

CREATE TABLE deliverystaus (
    orderid         INTEGER,
    driverid        INTEGER NOT NULL,
    status          VARCHAR2(30) NOT NULL
);

CREATE TABLE dishes (
    dishid              INTEGER NOT NULL,
    restaurantid        INTEGER NOT NULL,
    dishname            VARCHAR2(150) NOT NULL,
    dishtype            VARCHAR2(40) NOT NULL,
    price               NUMBER(4, 2) NOT NULL
```

```
);
CREATE TABLE drivers (
    driverid              INTEGER NOT NULL,
    drivername             VARCHAR2(100) NOT NULL,
    vehicletype            VARCHAR2(40) NOT NULL
);

CREATE TABLE Orders (
    orderid               INTEGER NOT NULL,
    customerid             INTEGER NOT NULL,
    restaurantid           INTEGER NOT NULL,
    dishid                INTEGER NOT NULL,
    specialinstructions     VARCHAR2(100) NULL
);
CREATE TABLE orderdetails (
    orderid       INTEGER NOT NULL,
    promotionid    INTEGER,
    driverid       INTEGER NOT NULL,
    totalitem      INTEGER NOT NULL,
    totaldue       NUMBER(4, 2) NOT NULL
);
```

```
CREATE TABLE restaurants (
    restaurantid       INTEGER NOT NULL,
    restaurantname     VARCHAR2(90) NOT NULL,
    restaurantaddress  VARCHAR2(300) NOT NULL,
    contactno          VARCHAR2(20) NOT NULL
);

CREATE TABLE restaurantpromotions (
    PromotionID            INTEGER NOT NULL,
    RestaurantID           INTEGER NOT NULL,
    Validity               VARCHAR2(20) NOT NULL,
    PromotionType VARCHAR(50) NOT NULL
);
--- Adding primary key and Foreign key
ALTER TABLE CUSTOMERS ADD PRIMARY KEY (CUSTOMERID);
ALTER TABLE ORDERS ADD PRIMARY KEY (ORDERID);
ALTER TABLE DISHES ADD PRIMARY KEY (DISHID);
ALTER TABLE RESTAURANTS ADD PRIMARY KEY (RESTAURANTID);
ALTER TABLE DRIVERS ADD PRIMARY KEY (DRIVERID);
```

```
ALTER TABLE RESTUARANTPROMOTIONS ADD PRIMARY KEY
(PROMOTIONID);

ALTER TABLE ORDERS ADD FOREIGN KEY (CUSTOMERID)
REFERENCES CUSTOMERS (CUSTOMERID);

ALTER TABLE ORDERS ADD FOREIGN KEY (RESTAURANTID)
REFERENCES RESTAURANTS(RESTAURANTID);

ALTER TABLE ORDERS ADD FOREIGN KEY (DISHID) REFERENCES
DISHES(DISHID);

ALTER TABLE DISHES ADD FOREIGN KEY (RESTAURANTID)
REFERENCES RESTAURANTS (RESTAURANTID);

ALTER TABLE RESTUARANTPROMOTIONS ADD FOREIGN KEY
(RESTAURANTID) REFERENCES RESTAURANTS(RESTAURANTID);

ALTER TABLE ORDERDETAILS ADD FOREIGN KEY (ORDERID)
REFERENCES ORDERS(ORDERID);

ALTER TABLE ORDERDETAILS ADD FOREIGN KEY (PROMOTIONID)
REFERENCES RESTUARANTPROMOTIONS(PROMOTIONID);

ALTER TABLE ORDERDETAILS ADD FOREIGN KEY (DRIVERID)
REFERENCES DRIVERS(DRIVERID);

ALTER TABLE DELIVERYSTAUS ADD FOREIGN KEY (ORDERID)
REFERENCES ORDERS(ORDERID);

ALTER TABLE DELIVERYSTAUS ADD FOREIGN KEY (DRIVERID)
REFERENCES DRIVERS(DRIVERID);
```

Inserting Data

Customer Data Insertion:

Insert into customers values (10000,'Kassy Allana', '69 Summit Court', 'Cookeville', 'TN',38501,6056987233);

Insert into customers values (10001,'Rachel Sierra', '6 Sheffield Drive', 'Orange Park', 'FL', 32065,7858196588);

Insert into customers values (10002,'Roberta Eric', '8094 Wild Rose Street', 'Rockville Centre', 'NY', 11570,8313022175);

Insert into customers values (10003,'Houston Radcliff', '80 Woodsman Road', 'East Meadow', 'NY', 11554,8644392049);

Insert into customers values (10004,'Vanessa Jerri', '400 South Sheffield Rd.', 'Wheeling', 'WV', 26003,6069964865);

Restaurants Data Insertion:

Insert into restaurants values (20001,'The Arctic Willow','354 Riverview St.Wappingers Falls, NY 12590' , 7737663006);

Insert into restaurants values (20002,'The Abacus Pantry','36 North Albany Lane, District Heights, MD 20747' , 6076922813);

Insert into restaurants values (20003,'The Savory Factory','69 Peninsula Ave., Oakland Gardens, NY 11364' , 4844208404);

Insert into restaurants values (20004,'The Minty Canteen','7408 Walt Whitman Street, Glastonbury, CT 06033' , 9174866382);

Insert into restaurants values (20005,'The Mellow Blossom','71 Euclid St., Reno, NV 89523' , 4422007027);

Dishes Data Insertion:

Insert into dishes (DISHID,RESTAURANTID,DISHNAME,DISHTYPE,PRICE) values (30001,20001,'Steamed Fennel Orange Shrimps','Sea Food',21.99);

Insert into dishes (DISHID,RESTAURANTID,DISHNAME,DISHTYPE,PRICE) values (30002,20002,'Gentle-Fried Coconut Bread','Bread',5.99);

Insert into dishes (DISHID,RESTAURANTID,DISHNAME,DISHTYPE,PRICE) values (30003,20003,'Sautéed Pepper Mango Gratin','Vegeterian',9.99);

Insert into dishes (DISHID,RESTAURANTID,DISHNAME,DISHTYPE,PRICE) values (30004,20004,'Cannellini and mango soup','Vegeterian',8.99);

Insert into dishes (DISHID,RESTAURANTID,DISHNAME,DISHTYPE,PRICE) values (30005,20005,'Beef tenderloin','Meat',25.99);

Customer Payments Data Insertion:

Insert into customerPayments (Customerid, CustomerName, Cardnumber, CardExpDate, BillingAddress) values (10000,'Kassy Allana',365620128238544,'05/25','69 Summit Court,Cookeville, TN 38501');

Insert into customerPayments (Customerid, CustomerName, Cardnumber, CardExpDate, BillingAddress) values (10001,'Rachel Sierra',371296260356014,'04/21','6 Sheffield Drive,Orange Park, FL 32065');

Insert into customerPayments (Customerid, CustomerName, Cardnumber, CardExpDate, BillingAddress) values (10002,'Roberta Eric',361235758065671,'05/23','Rockville Centre, NY 11570');

Insert into customerPayments (Customerid, CustomerName, Cardnumber, CardExpDate, BillingAddress) values (10003,'Houston Radcliff',361192083678373,'01/22','80 Woodsman Road,East Meadow, NY 11554');

Insert into customerPayments (Customerid, CustomerName, Cardnumber, CardExpDate, BillingAddress) values (10004,'Vanessa Jerri',374234880858652,'09/21','400 South Sheffield Rd,Wheeling, WV 26003');

Orders Data Insertion:

Insert into ORDERS (OrderID, CustomerID, RestaurantID, DishID, SpecialInstructions) Values (40001,10000,20001,30001, 'No onion Please');

Insert into ORDERS (OrderID, CustomerID, RestaurantID, DishID, SpecialInstructions) Values (40002,10001,20002,30002, 'No tomato Please');

Insert into ORDERS (OrderID, CustomerID, RestaurantID, DishID, SpecialInstructions) Values (40003,10002,20003,30003, 'N/A');

Insert into ORDERS (OrderID, CustomerID, RestaurantID, DishID, SpecialInstructions) Values (40004,10003,20004,30004, 'No Mastard Please');

Insert into ORDERS (OrderID, CustomerID, RestaurantID, DishID, SpecialInstructions) Values (40005,10004,20005,30005, 'No sauce Please');

Drivers Data Insertion:

Insert into Drivers (DriverID, DriverName, VehicleType) values (50001,'Daryl Buchanan','Car');

Insert into Drivers (DriverID, DriverName, VehicleType) values (50002,'Dawn Collins','BiCycle');

Insert into Drivers (DriverID, DriverName, VehicleType) values (50003,'Jacquelyn Bennett','MotorBike');

Insert into Drivers (DriverID, DriverName, VehicleType) values (50004,'Lila Simon','Car');

Insert into Drivers (DriverID, DriverName, VehicleType) values (50005,'Carroll Stevens','Car');

RestaurantPromotions Data Insertion:

Insert into RestaurantPromotions Values (60001,20001,'12/20/2020','10% off next delivery');

Insert into RestaurantPromotions Values (60002,20002,'12/20/2020','10% off next delivery');

Insert into RestaurantPromotions Values (60003,20003,'01/20/2021','15% off next delivery');

Insert into RestaurantPromotions Values (60004,20004,'01/13/2021','15% off next delivery');

Insert into RestaurantPromotions Values (60005,20005,'02/10/2021','25% off next delivery');

Insert into RestaurantPromotions Values (60006,20005,'12/28/2020','30% off next delivery');

OrderDetails Data Insertion:

Insert into OrderDetails Values (40001,60001,50001,4,65.00);

Insert into OrderDetails Values (40002,60002,50002,1,20.00);

Insert into OrderDetails Values (40003,60003,50003,7,105.90);

Insert into OrderDetails Values (40004,60004,50004,9,165.87);

Insert into OrderDetails Values (40005,60005,50005,3,48.45);

Delivery Status Data Insertion:

Insert into DeliveryStatus Values (40001,50001,'Delivered');

Insert into DeliveryStatus Values (40002,50002,'Prepared');

Insert into DeliveryStatus Values (40003,50003,'Delivered');

Insert into DeliveryStatus Values (40004,50004,'Delivered');

Insert into DeliveryStatus Values (40005,50005,'Delayed');

Joining Multiple table to show relations

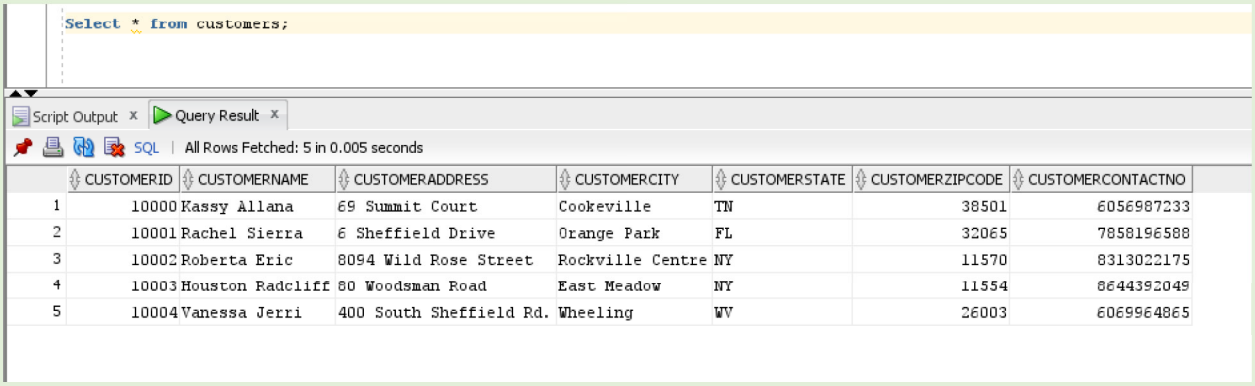
Select customers.Customerid,customername, orders.orderID, driverID from customers,orders,orderdetails where

customers.customerid = orders.customerid and

orders.orderid = orderdetails.orderid;

Table Data

Customer Table Data:



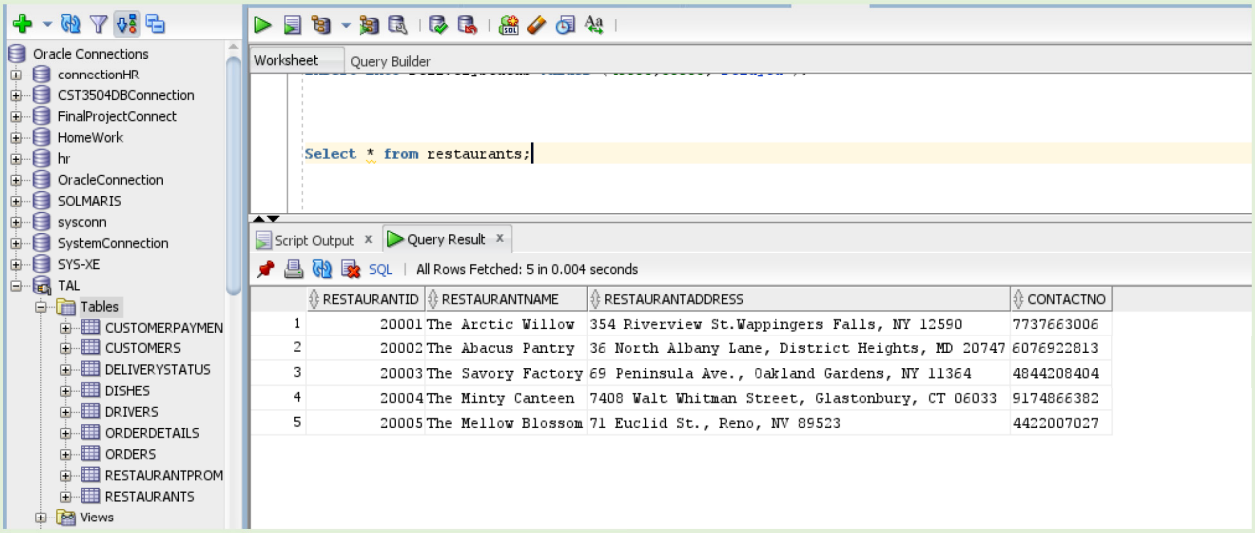
```
Select * from customers;
```

Script Output | Query Result

SQL | All Rows Fetched: 5 in 0.005 seconds

	CUSTOMERID	CUSTOMERNAME	CUSTOMERADDRESS	CUSTOMERCITY	CUSTOMERSTATE	CUSTOMERZIPCODE	CUSTOMERCONTACTNO
1	10000	Kassy Allana	69 Summit Court	Cookeville	TN	38501	6056987233
2	10001	Rachel Sierra	6 Sheffield Drive	Orange Park	FL	32065	7858196588
3	10002	Roberta Eric	8094 Wild Rose Street	Rockville Centre	NY	11570	8313022175
4	10003	Houston Radcliff	80 Woodsman Road	East Meadow	NY	11554	8644392049
5	10004	Vanessa Jerri	400 South Sheffield Rd.	Wheeling	WV	26003	6069964865

Restaurants Table Data:



Oracle Connections
- connectionHR
- CST3504DBConnection
- FinalProjectConnect
- HomeWork
- hr
- OracleConnection
- SOLMARIS
- sysconn
- SystemConnection
- SYS-XE
- TAL
  - Tables
    - CUSTOMERPAYMEN
    - CUSTOMERS
    - DELIVERYSTATUS
    - DISHES
    - DRIVERS
    - ORDERDETAILS
    - ORDERS
    - RESTAURANTPROM
    - RESTAURANTS
  - Views

Worksheet | Query Builder

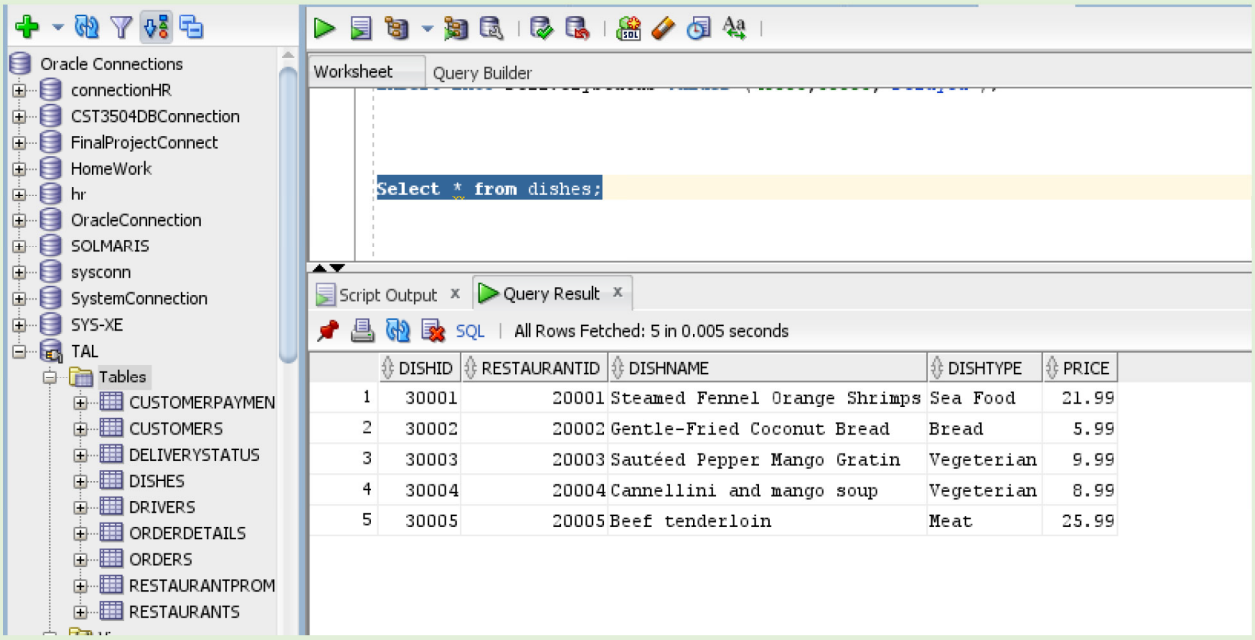
```
Select * from restaurants;
```

Script Output | Query Result

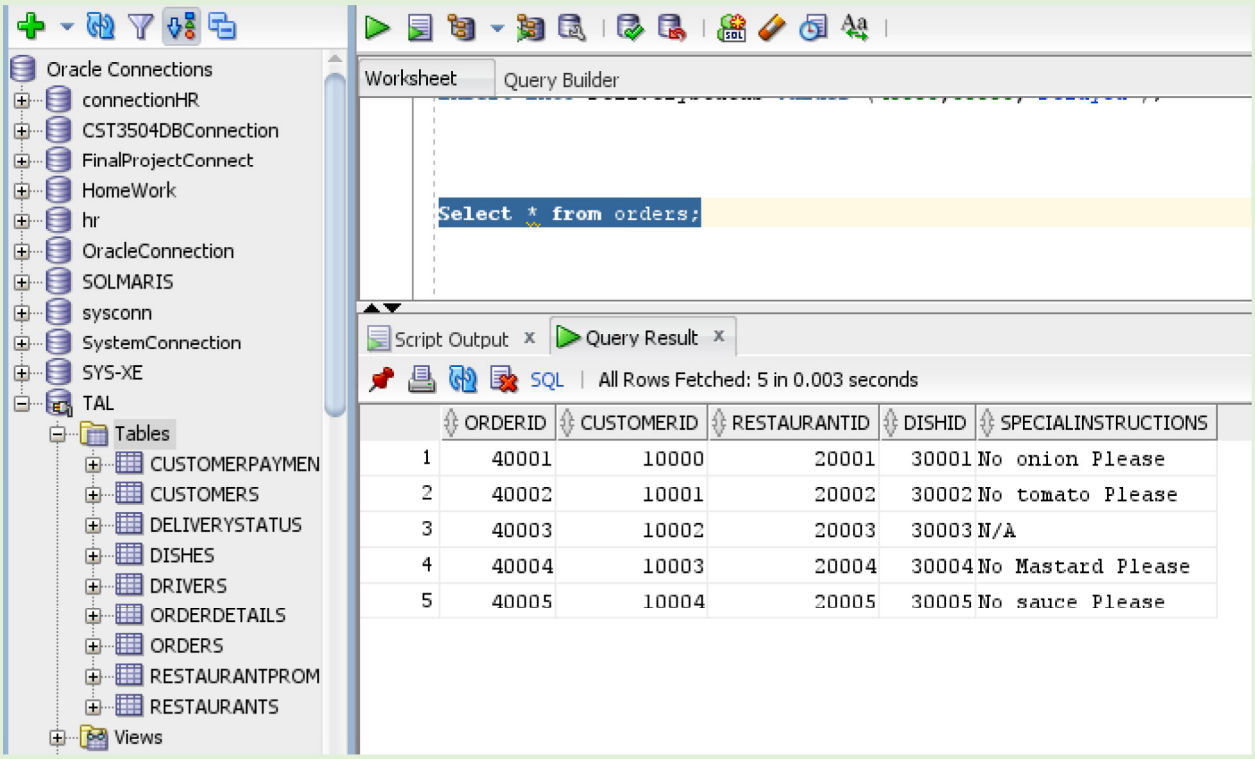
SQL | All Rows Fetched: 5 in 0.004 seconds

	RESTAURANTID	RESTAURANTNAME	RESTAURANTADDRESS	CONTACTNO
1	20001	The Arctic Willow	354 Riverview St.Wappingers Falls, NY 12590	7737663006
2	20002	The Abacus Pantry	36 North Albany Lane, District Heights, MD 20747	6076922813
3	20003	The Savory Factory	69 Peninsula Ave., Oakland Gardens, NY 11364	4844208404
4	20004	The Minty Canteen	7408 Walt Whitman Street, Glastonbury, CT 06033	9174866382
5	20005	The Mellow Blossom	71 Euclid St., Reno, NV 89523	4422007027

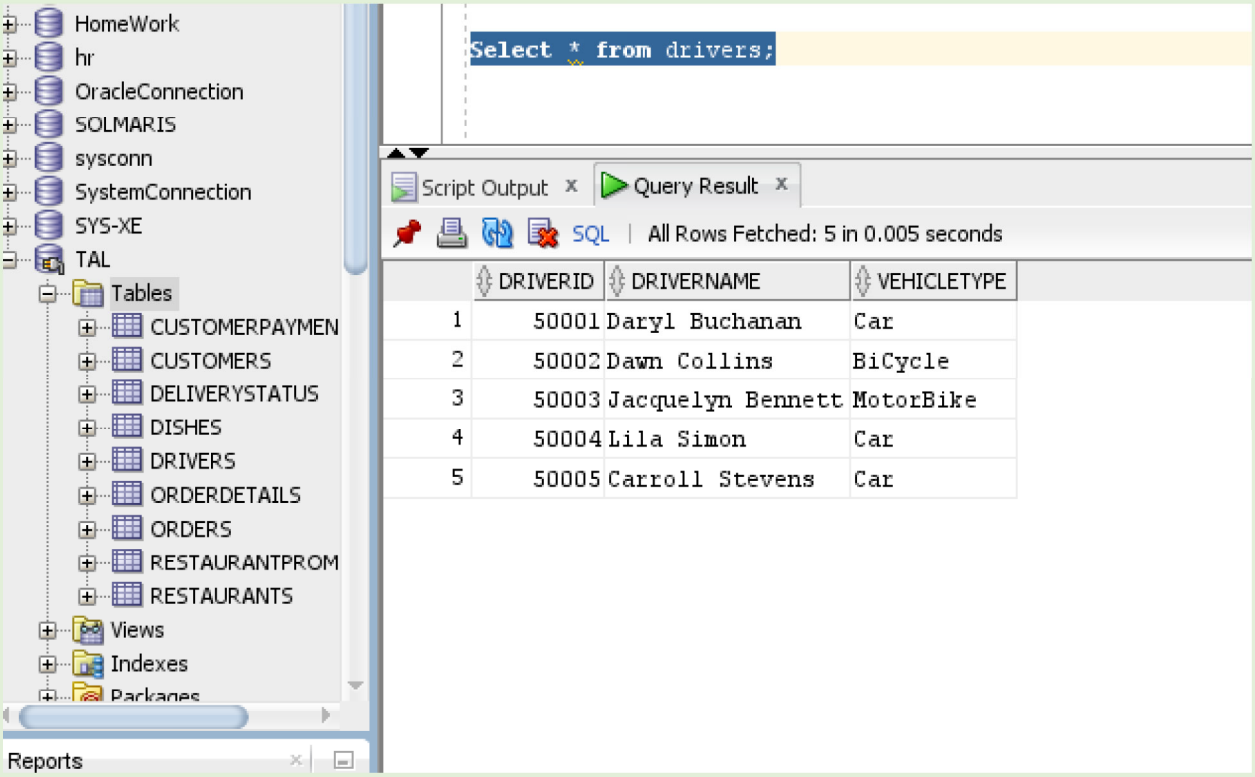
Dishes Table Data:



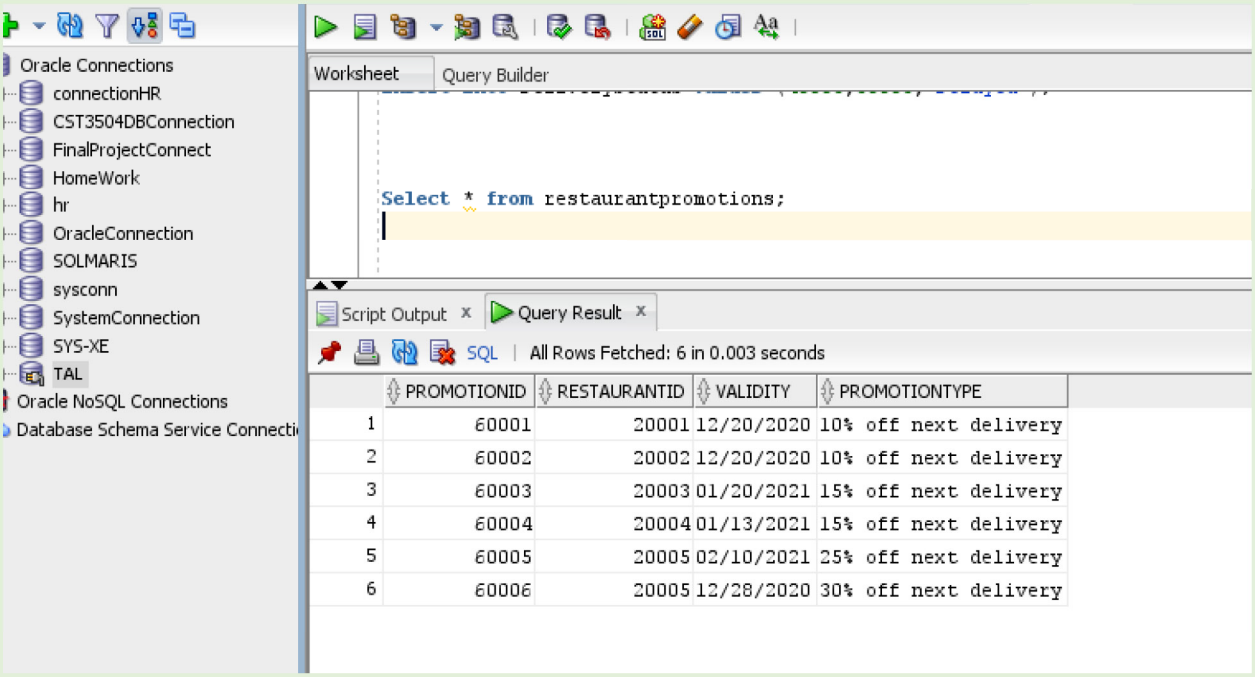
Orders table Data:



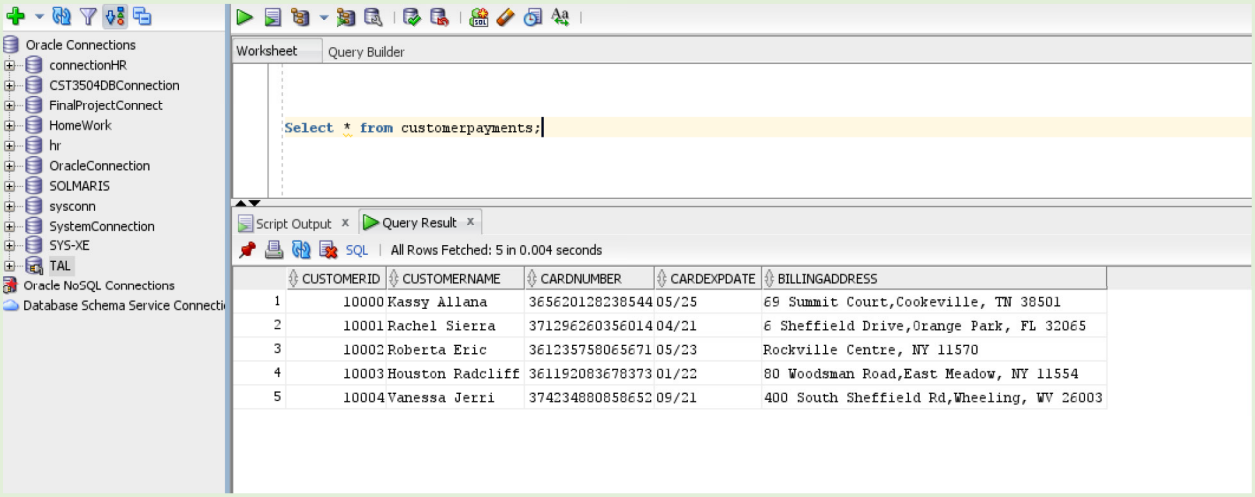
Drivers table Data:



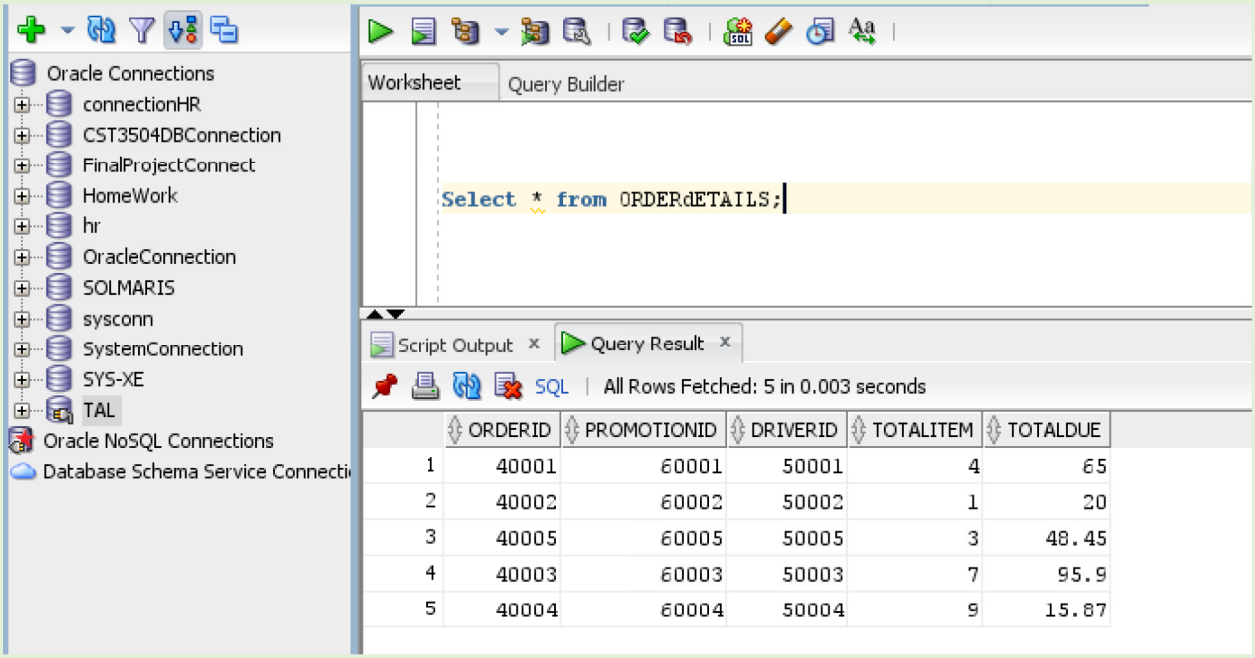
RestaurantPromotions table data:



CustomerPayment Table Data:

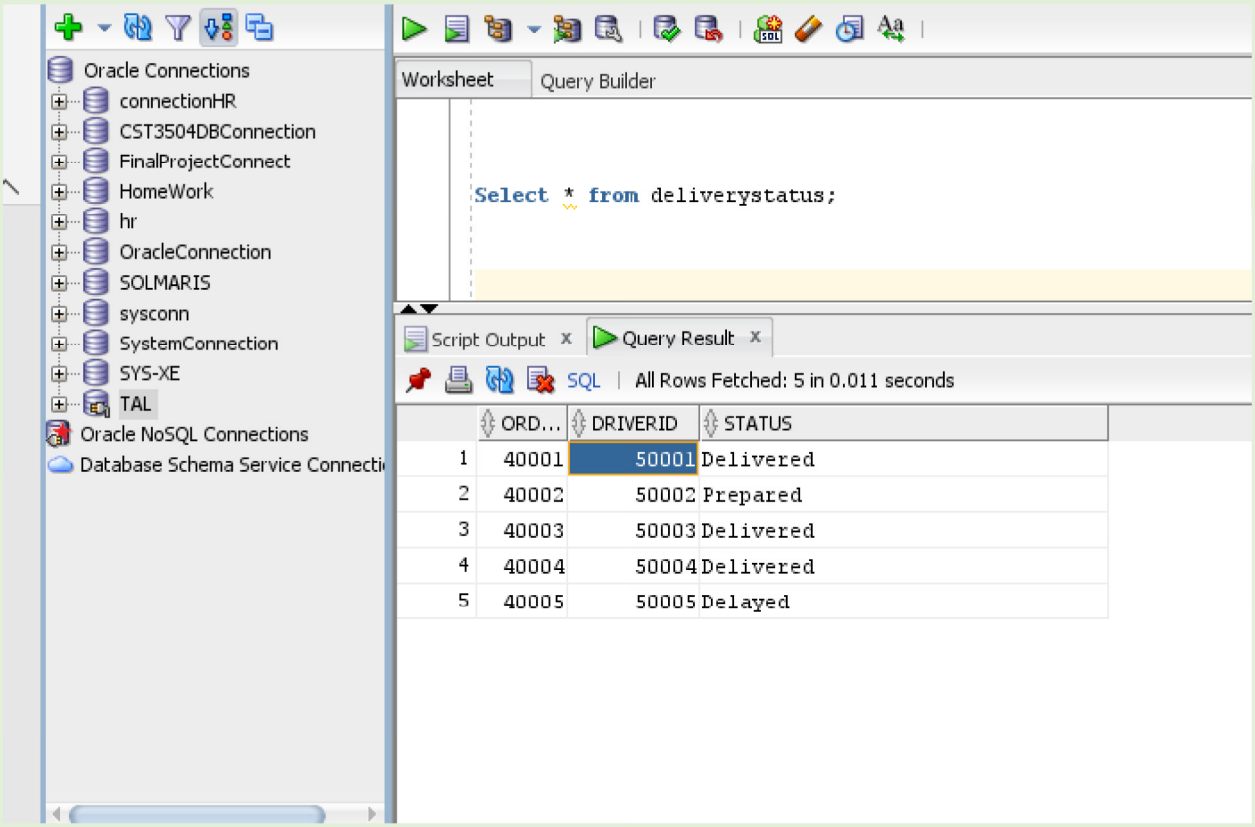


OrderDetails table Data:

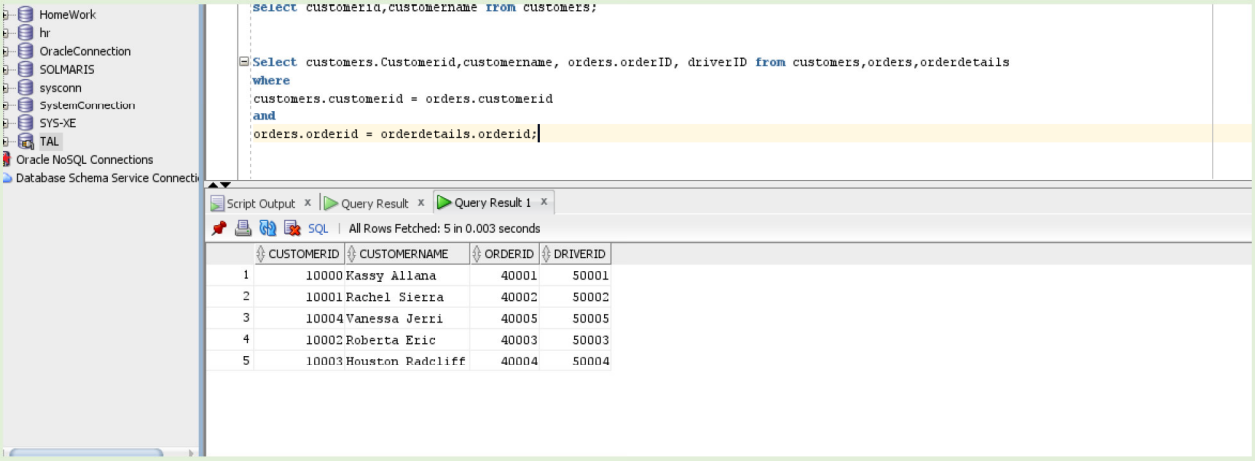


	ORDERID	PROMOTIONID	DRIVERID	TOTALITEM	TOTALDUE
1	40001	60001	50001	4	65
2	40002	60002	50002	1	20
3	40005	60005	50005	3	48.45
4	40003	60003	50003	7	95.9
5	40004	60004	50004	9	15.87

DeliveryStatus data Report:



Joining tables to Display Record



GO FOOD DELIVERY

YOUR DELIVERY PARTNER

BY

Table of Content

Who We are

- GoFood Started their journey during the global Covid-19 Crisis of 2020 where the majority of restaurant delivery aggregators and delivery service providers refused to reduce their unreasonable commissions, putting a massive strain on restaurants already struggling with reduced business.

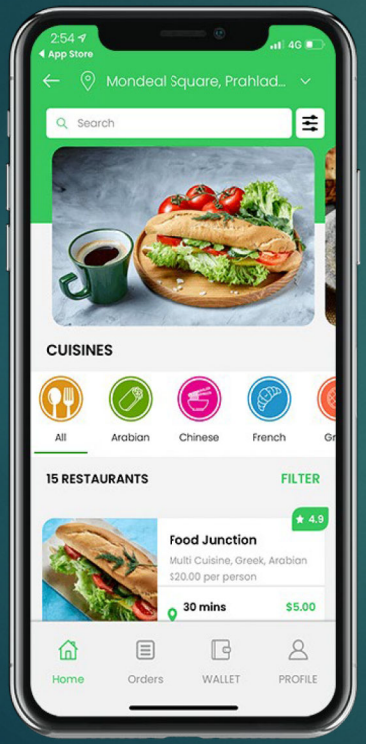
- More than 500 restaurant owners representing 4,000 restaurants in the USA decided to come together to create their own delivery platform and GoFood was born.

How Does our app Work

- Its Simple. Download the app or go to gofood.com
- Search your favorite Restaurant
- Select the Food
- Select the Payment Method
- Place your order

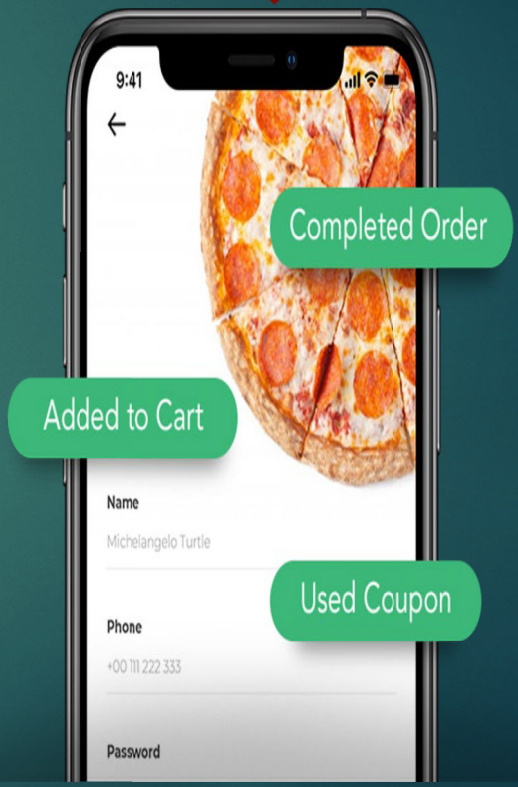
App Demo

Search Your favorite Restaurant



Search Your favorite Food

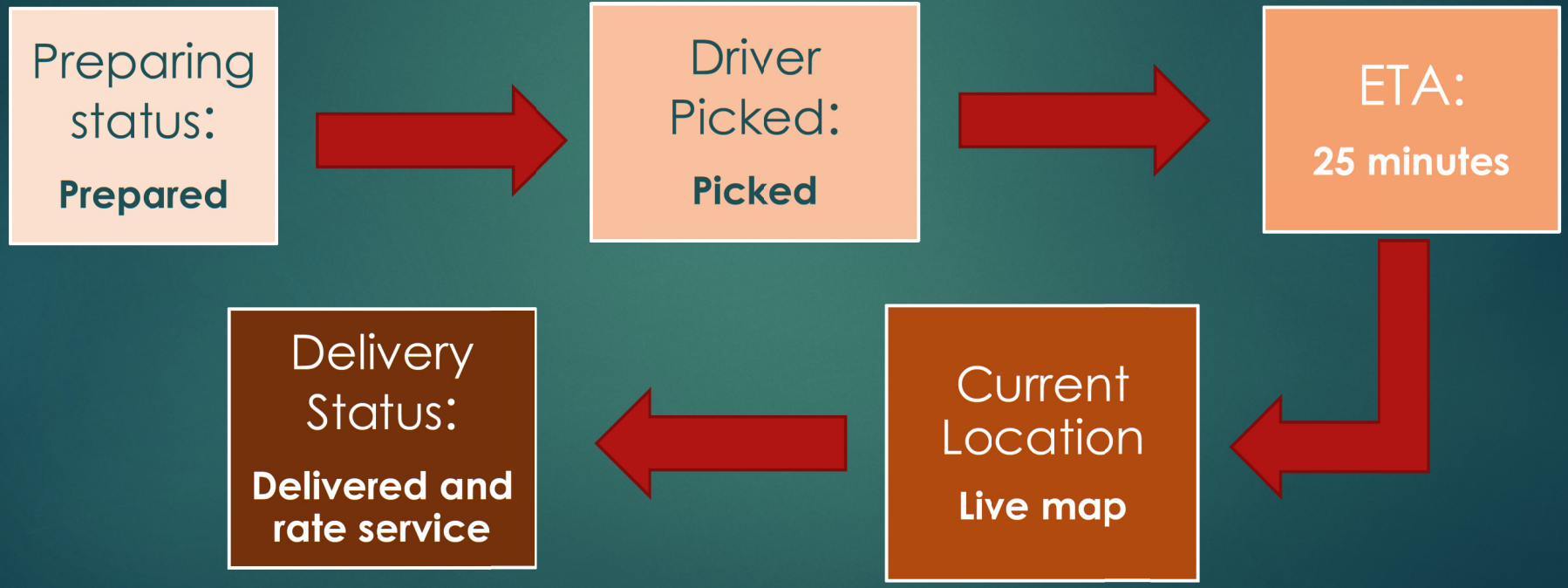
Complete Order



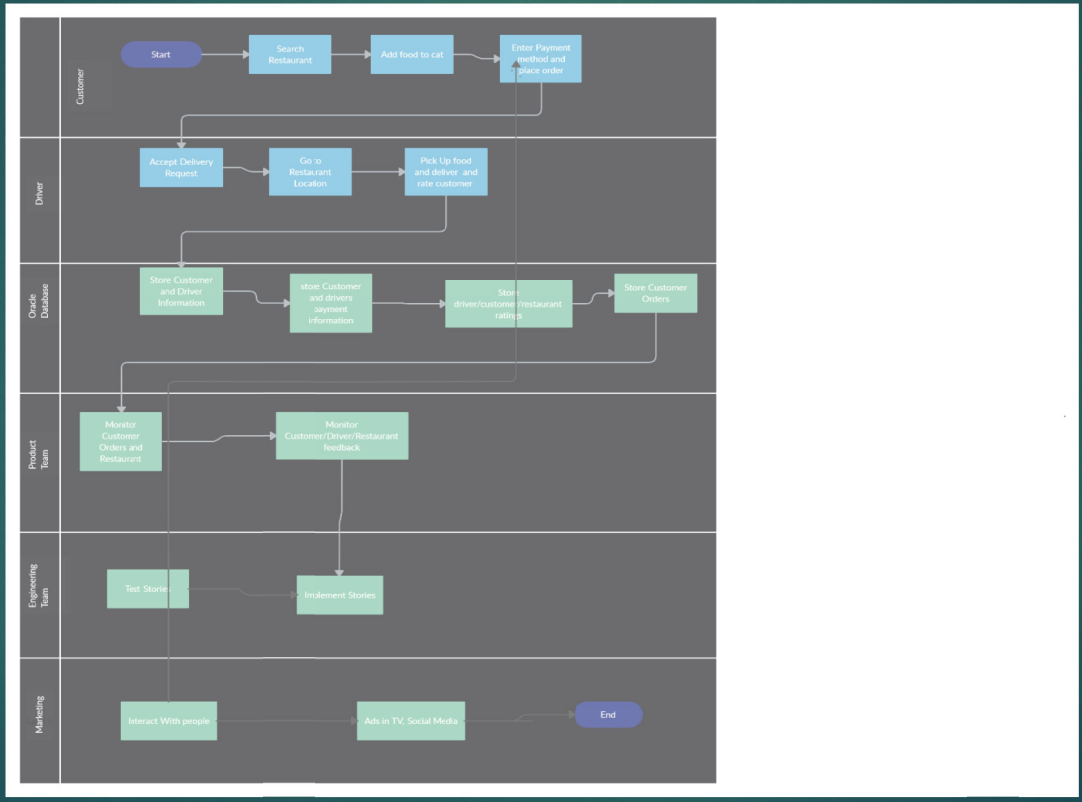
Delivery Tracking System

- Check the status of preparing food
- Check if the driver picked the food
- Check live Estimated arrival time
- Track drivers live location
- Get food and rate the service

Delivery Tracking System Demo



Swim line


Customer
Start
Search Restaurant
Add food to cat
Enter Payment method and place order
Driver
Accept Delivery Request
Go to Restaurant Location
Pick Up food and deliver and rate customer
Oracle Database
Store Customer and Driver Information
store Customer and drivers payment information
Store driver/customer/restaurant ratings
Store Customer Orders
Product Team
Monitor Customer Orders and Restaurant
Monitor Customer/Driver/Restaurant feedback
Engineering Team
Test Stories
Implement Stories
Marketing
Interact With people
Ads in TV, Social Media
End

Why We are

- GoFood will give you access to more than 2,000 restaurants at launch
- with the number expected to grow to over 4,000 by the end of 2020.
- As a marketplace and optional last mile provider, GoFood will never charge a delivery fee directly from the customer.
- Instead, all delivery fees go directly to the restaurant.

Delivery Security

- Your Food is safe and secure in our hand
- Every driver must wear gloves and change gloves before and after delivery
- Your contact number will never be public to driver or restaurant
- Gives you the best food service
- Priority service for orders greater than $20
- Always rate driver and restaurant about your delivery Experience

Benefits Of using our Service

- Promotions added to your account after each delivery
- Earned point for every dollar spent
- Low cost health insurance for long term members
- Referral bonus
- Use GoFood points to buy travel tickets

Conclusion

- GoFood is for everyone. We will have locally home-grown restaurants, owner operated restaurants in addition to local, regional and international chains. We are also in the process of developing a loyalty program, which you can use for discounts and special offers with participating restaurants. By using GoFood, you're also supporting your favorite local restaurants directly, by ensuring they do not pay high commissions to third party aggregators, which in turn leads to better quality and services. It's a win-win.