README

Business Problem

Currently, companies trying to build a database server have never been trying to make a profit-making strategy through E-Commercial, so there are a limited number of consumers, and they have difficulty in purchasing goods and it is time-consuming.

Therefore, the purpose of this project is to generate more profits by selling goods to a wider customer base through E-Commercial.

Potential Approach

Based on the Oracle SQL Developer program, we will create tables based on the characteristics of the information contained in PL/SQL. The total number of the table is six, which are **Product**, **Customers**, **Category**, **OrderDetail**, **Orders**, and **Shipping** in the Entity-Relationship Diagram (ERD).

To illustrate the sequence of operations, we first create a Sequence and insert the values into the six tables created and the attributes each table has. Next, we will set up the Index and Trigger, test them, and make two Procedures and Functions for convenience.

Related SQL Work:

- Sequences
- Create six tables
- Insert values
- Trigger
- Procedure
- Function

Once Oracle SQL scripts are made, we will use JavaFX, which is a Java library using which we can develop Rich Internet Applications.

What is JavaFX?

JavaFX is a Java library that allows users to build Rich Internet Applications. This application used by JavaFX library can run consistently across multiple platforms and on various devices such as Desktop, Mobile Phones, Tablets, TVs, etc. Moreover, JavaFX also provides interfaces using which developers can combine graphics animation and UI control.

** More Details: https://www.tutorialspoint.com/javafx/javafx_overview.htm

Entity-Relationship Diagram (ERD)

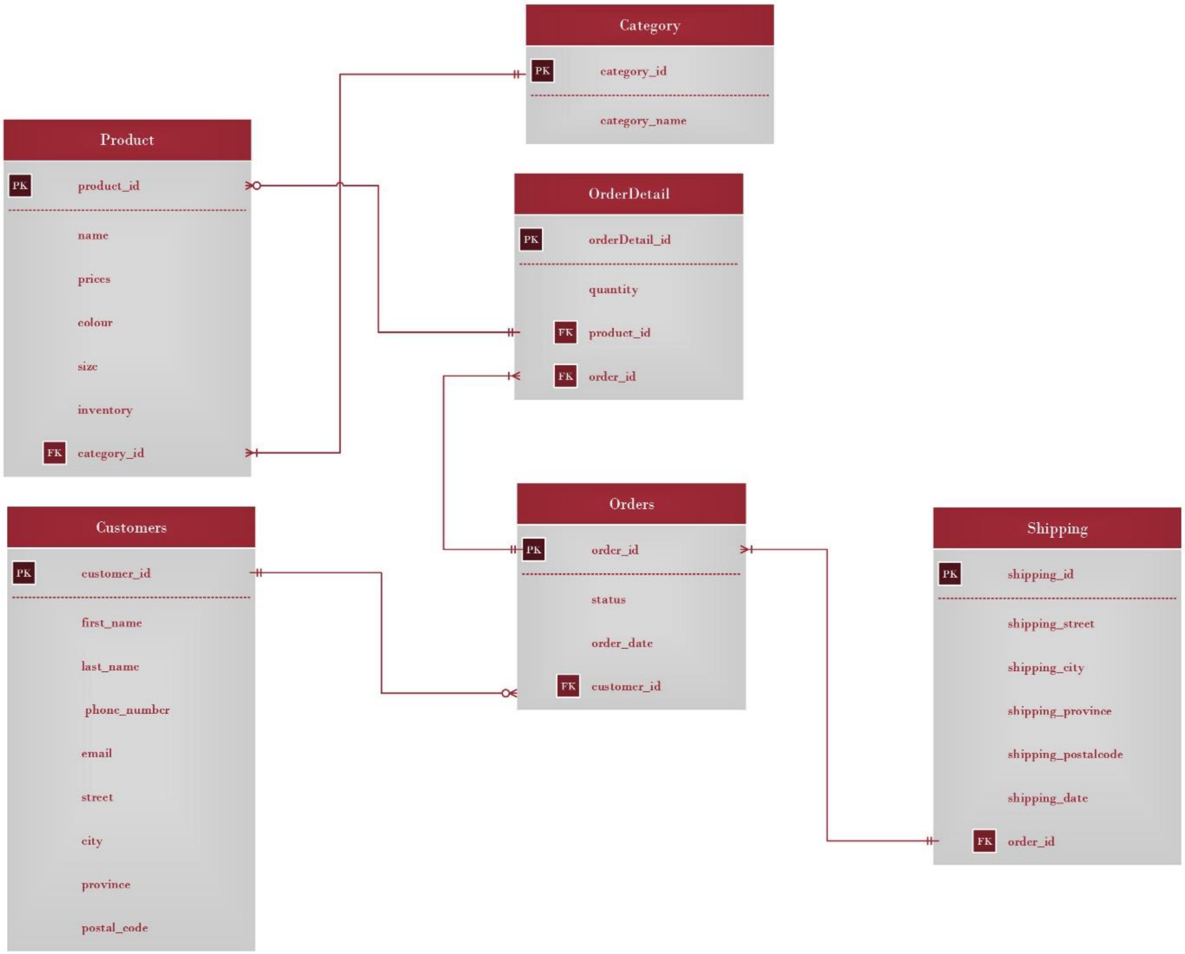
What is Entity-Relationship Diagram (ERD)?

It shows the relationships of entity sets stored in a database. An entity set is a collection of similar entities. With entities' attributes and relationships between them, ERD illustrates the logical structure of databases.

** More Details: https://www.smartdraw.com/entity-relationship-diagram/

We have created 6 tables with attributes (including Primary Key & Foreign Key):

- Product – product_id (PK), name, prices, colour, size, inventory, category_id (FK)
- Customers – customer_id (PK), first_name, last_name, phone_number, email, street, city, province, postal_code
- Category – category_id (PK), category_name
- OrderDetail – orderDetail_id (PK), quantity, product_id (FK), order_id (FK)
- Orders – order_id (PK), status, order_date, customer_id (FK)
- Shipping – shipping_id (PK), shipping_street, shipping_city, shipping_province, shipping_postalcode, shipping_date, order_id (FK)



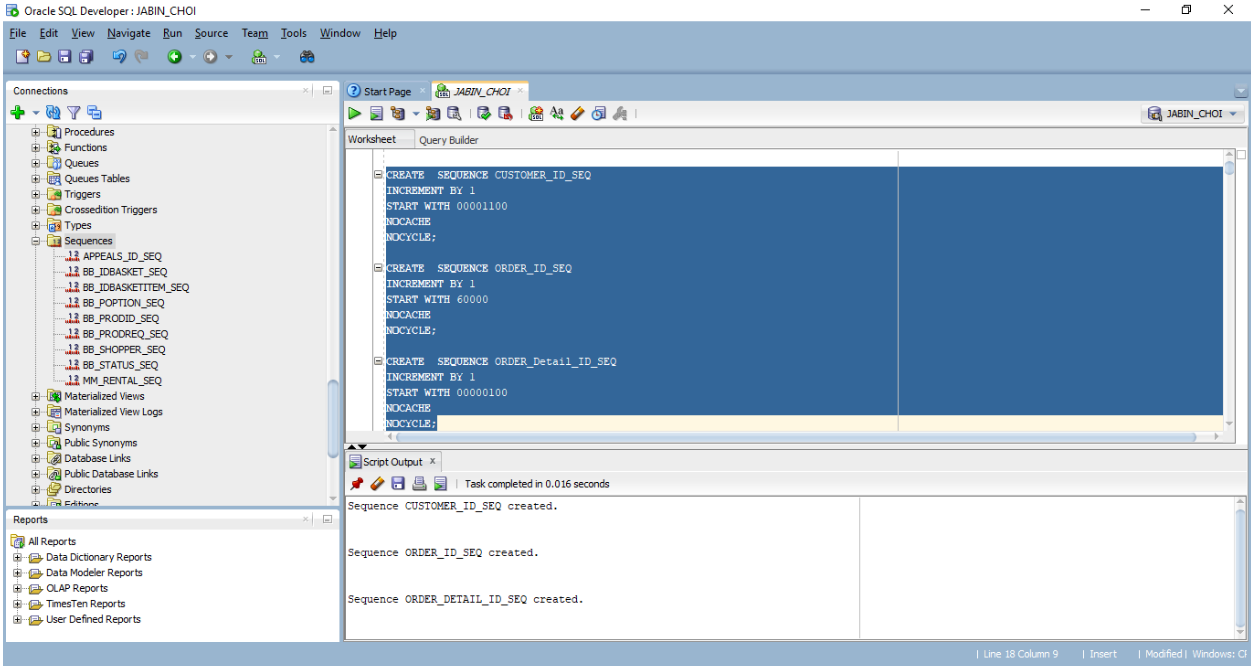
SQL Queries

Sequences

```
CREATE  SEQUENCE CUSTOMER_ID_SEQ
INCREMENT BY 1
START WITH 00001100
NOCACHE
NOCYCLE;

CREATE  SEQUENCE ORDER_ID_SEQ
INCREMENT BY 1
START WITH 60000
NOCACHE
NOCYCLE;

CREATE  SEQUENCE ORDER_Detail_ID_SEQ
INCREMENT BY 1
START WITH 00000100
NOCACHE
NOCYCLE;
```



```
CREATE  SEQUENCE CUSTOMER_ID_SEQ
INCREMENT BY 1
START WITH 00001100
NOCACHE
NOCYCLE;

CREATE  SEQUENCE ORDER_ID_SEQ
INCREMENT BY 1
START WITH 60000
NOCACHE
NOCYCLE;

CREATE  SEQUENCE ORDER_Detail_ID_SEQ
INCREMENT BY 1
START WITH 00000100
NOCACHE
NOCYCLE;
```

Task completed in 0.016 seconds

```
Sequence CUSTOMER_ID_SEQ created.


Sequence ORDER_ID_SEQ created.


Sequence ORDER_DETAIL_ID_SEQ created.
```

Create tables

```
--/Customer/
CREATE TABLE Customer
( customer_id  NUMBER(8) Default CUSTOMER_ID_SEQ.NEXTVAL,
  first_name VARCHAR2(20),
  last_name VARCHAR2(20),
  phone_number VARCHAR2 (15),
  email VARCHAR2(50),
  street VARCHAR2(50),
  city VARCHAR2(20),
  province CHAR(2),
  postalCode CHAR(7),
  CONSTRAINT customer_customerID_pk PRIMARY KEY ( customer_id )
);

--/Category/
CREATE TABLE Category
( category_id NUMBER(8) PRIMARY KEY,
  category_name VARCHAR(20)
);

--/Product/
CREATE TABLE Product
( product_id NUMBER(8) PRIMARY KEY,
  product_name VARCHAR2(20),
  prices NUMBER(8, 2),
  colour VARCHAR2(20),
  product_size VARCHAR2(10),
  inventory VARCHAR2(10),
  category_id NUMBER(8) REFERENCES Category ( category_id )
);

--/Orders/
CREATE TABLE Orders
( order_id NUMBER(8) Default ORDER_ID_SEQ.NEXTVAL PRIMARY KEY,
  status VARCHAR2(10),
  order_date DATE,
  customer_id NUMBER(8) REFERENCES Customer ( customer_id )
  );

--/Orders detail/
CREATE TABLE OrderDetail
( orderDetail_id NUMBER(8) Default ORDER_Detail_ID_SEQ.NEXTVAL PRIMARY KEY,

  quantity NUMBER(3),
  order_id NUMBER(8) REFERENCES orders ( order_id ),
  product_id NUMBER(8) REFERENCES Product ( product_id )
);

--/Shipping/
CREATE TABLE Shipping
( shipping_id NUMBER(8)PRIMARY KEY,
  shipping_street VARCHAR2(50),
  shipping_city VARCHAR2(20),
  shipping_province CHAR(2),
  shipping_postalcode CHAR(7),
  shipping_date DATE,
  order_id NUMBER(8) REFERENCES Orders (order_id)
);
```

Insert Values

```
--/Insert Customer Values/
INSERT INTO CUSTOMER
VALUES (Default, 'Louis', 'Russell', '(647)-470-8867','sl341928@gmail.com','236 Olive St.','Donnacona', 'QC','G3M S8A');
INSERT INTO CUSTOMER
VALUES (Default,'Ramon','Hunter','(416)-778-3524','34567thred@gmail.com','7063 Oakland Drive','Moores Mills', 'NB','E5A M3P');
INSERT INTO CUSTOMER
VALUES (Default,'Terrance','Kim','(647)-445-6454','shuhhth12438@gmail.com','8866 Kingston Rd.','Sainte-Geneviève', 'QC','H9H V6P');
INSERT INTO CUSTOMER
VALUES (Default,'Doyle','Padilla','(647)-889-5643','wanqilimen9856@gmail.com','592 Addison Avenue','Concord', 'ON','L4K C9Y');
INSERT INTO CUSTOMER
VALUES (Default,'Jacob','Collier','(647)-998-4756','bu5634238@gmail.com','9957 North Andover Ave.','Saint-Colomban', 'QC','J5K T9Y');
INSERT INTO CUSTOMER
VALUES (Default,'Gina','Caldwell','(416)-478-7860','gang_tuir45@gmail.com','668 E. Wood St.','Penetanguishene', 'ON','L9M K6Y');
INSERT INTO CUSTOMER
VALUES (Default,'Inez','Schneider','(647)-990-8987','syy_dd876@gmail.com','979 North Lancaster Street','Orillia', 'ON','L3V C1Y');
INSERT INTO CUSTOMER
VALUES (Default,'Sandy','Spencer','(647)-100-8898','0987guanlee@gmail.com','7071 Thatcher Street','Lantz', 'NS','B2S G1J');
INSERT INTO CUSTOMER
VALUES (Default,'Brenda','Hopkins','(647)-470-8767','mindguuwn@gmail.com','703 N. West Ave.','Ste. Anne', 'MB','R5H T4T');
INSERT INTO CUSTOMER
VALUES (Default,'Salvatore','Cooper','(416)-487-1299','priminyy_cas@gmail.com','9004 Branch Lane','Keswick', 'ON','L4P S3G');
```

```
--/Insert Category Values/
INSERT INTO Category
VALUES (00001001,'Dresses');
INSERT INTO Category
VALUES (00001002,'Pants');
INSERT INTO Category
VALUES (00001003,'Blouses');
INSERT INTO Category
VALUES (00001004,'Sweaters');
INSERT INTO Category
VALUES (00001005,'T-Shirts');
INSERT INTO Category
VALUES (00001006,'Skirts');
INSERT INTO Category
VALUES (00001007,'Jackets');
INSERT INTO Category
VALUES (00001008,'Denim');
INSERT INTO Category
VALUES (00001009,'Shoes');
INSERT INTO Category
VALUES (00001010,'Accessories');

--/Insert Product Values/
INSERT INTO product
VALUES (20001, 'Talence Blouse','87.00','white','XXS', '13', 00001003);
INSERT INTO product
VALUES (20002, 'Best Primrose Skirt','67.00','Ashen','S','11', 00001006);
INSERT INTO product
VALUES (20003, 'Best Weller Dress','120.00','Oak','M','10', 00001001);
INSERT INTO product
VALUES (20004, 'Howley Jeans','78.00','Reddish','XXS','8', 00001008);
INSERT INTO product
VALUES (20005, 'Fitted Mini Dress','120.00','Cordovan','S','24', 00001001);
INSERT INTO product
VALUES (20006, 'Coleridge T-Shirt','35.00','Grey','S','32', 00001005);
INSERT INTO product
VALUES (20007, 'Free Subah Pant','140.00','Fatigue','L','11', 00001002);
INSERT INTO product
VALUES (20008,'Best Carly Dress','127.00','Cardamon','M', '8',00001001);
INSERT INTO product
VALUES (20009, 'Free Jamilla T-Shirt','30.00','Dew Blue','XS','13', 00001003);
INSERT INTO product
VALUES (20010, 'Lance Cardigan','225.00','White','XXS','2', 00001004);
INSERT INTO product
VALUES (20011, 'Gap hoody', '15.00', 'Dark Gray', 'S', '0', 00001005);
```

```
--/Insert Orders Values/
INSERT INTO Orders
VALUES (Default,'Y','09-SEP-17', 1100);
INSERT INTO Orders
VALUES (Default,'Y','09-OCT-17', 1100);
INSERT INTO Orders
VALUES (Default,'Y','19-JAn-18', 1100);
INSERT INTO Orders
VALUES (Default,'Y','09-OCT-17', 1101);
INSERT INTO Orders
VALUES (Default,'Y','10-OCT-17', 1102);
INSERT INTO Orders
VALUES (Default,'N','11-OCT-17', 1103);
INSERT INTO Orders
VALUES (Default,'Y','12-OCT-17', 1104);
INSERT INTO Orders
VALUES (Default,'Y','12-NOV-17', 1105);
INSERT INTO Orders
VALUES (Default,'N','13-NOV-17', 1106);
INSERT INTO Orders
VALUES (Default,'Y','15-NOV-17', 1107);
INSERT INTO Orders
VALUES (Default,'N','15-NOV-17', 1108);
INSERT INTO Orders
VALUES (Default,'N','15-NOV-17', 1109);

--/Insert OrderDetail/
INSERT INTO OrderDetail
VALUES (Default, 1,60012,  20001);
INSERT INTO OrderDetail
VALUES (Default, 2,60012,  20008);
INSERT INTO OrderDetail
VALUES (Default, 1,60012,  20005);
INSERT INTO OrderDetail
VALUES (Default, 4,60013,  20004);
INSERT INTO OrderDetail
VALUES (Default,1, 60013, 20002);
INSERT INTO OrderDetail
VALUES (Default,3, 60014, 20008);
INSERT INTO OrderDetail
VALUES (Default,1, 60014, 20005);
INSERT INTO OrderDetail
VALUES (Default,1, 60014, 20006);
INSERT INTO OrderDetail
VALUES (Default,1, 60015, 20002);
INSERT INTO OrderDetail
VALUES (Default,5, 60016, 20002);
INSERT INTO OrderDetail
```

```
VALUES (Default,5, 60016, 20004);
INSERT INTO OrderDetail
VALUES (Default,5, 60017, 20006);
INSERT INTO OrderDetail
VALUES (Default,3, 60018, 20007);
INSERT INTO OrderDetail
VALUES (Default,2, 60019, 20002);
INSERT INTO OrderDetail
VALUES (Default,5, 60020, 20002);
INSERT INTO OrderDetail
VALUES (Default,3, 60021, 20007);
INSERT INTO OrderDetail
VALUES (Default,2, 60022, 20005);
INSERT INTO OrderDetail
VALUES (Default,3, 60022, 20002);
INSERT INTO OrderDetail
VALUES (Default,2,60022, 20001);
INSERT INTO OrderDetail
VALUES (Default,2, 60023, 20009);

/* Insert Shipping Values */
INSERT INTO Shipping
VALUES (111221,'236 Olive St.','Donnacona', 'QC','G3M S8A','11-SEP-17',60012);
INSERT INTO Shipping
VALUES (111222,'7071 Thatcher Street','Lantz', 'NS','B2S G1J','11-OCT-17',60013);
INSERT INTO Shipping
VALUES (111223,'236 Olive St.','Donnacona','QC','G3M S8A','12-OCT-17',60013);
INSERT INTO Shipping
VALUES (111224,'668 E. Wood St.','Penetanguishene', 'ON','L9M K6Y','12-OCT-17',60014);
INSERT INTO Shipping
VALUES (111225,'8866 Kingston Rd.','Sainte-Geneviève', 'QC','H9H V6P','14-OCT-17',60015);
INSERT INTO Shipping
VALUES (111226,'2680 Lockhart Drive','Barrie', 'ON','L4M 3B1','15-NOV-17',60016);
INSERT INTO Shipping
VALUES (111227,'393 James Street','St Catharines', 'ON','L2R 3H6','15-NOV-17',60017);
INSERT INTO Shipping
VALUES (111228,'592 Addison Avenue','Concord', 'ON','L4K C9Y','16-NOV-17',60018);
INSERT INTO Shipping
VALUES (111229,'703 N. West Ave.','Ste. Anne', 'MB','R5H T4T','16-NOV-17',60019);
INSERT INTO Shipping
VALUES (111210,'9004 Branch Lane','Keswick', 'ON','L4P S3G','16-NOV-17',60020);
INSERT INTO Shipping
VALUES (111211,'263 Progress Ave','Barbie', 'ON','L4P S3G','16-NOV-17',60021);
INSERT INTO Shipping
VALUES (111212,'92 Markham Road','Nakeov', 'ON','4YP SO9','21-NOV-15',60022);
INSERT INTO Shipping
VALUES (111213,'945 Dundas St','Polfea', 'ON','ML1 IH3','6-Jan-18',60023);
INSERT INTO Shipping
```

```
VALUES (111214,'43848 Steels','Barbie', 'ON','M1G 3T8','16-Dec-17',60023);
```

Index

```
CREATE INDEX Order_Status
ON Orders (status);

 CREATE INDEX Product_info
ON Product (product_name,prices);

 CREATE INDEX Customer_info
ON Customer (first_name,last_name,email);

 CREATE INDEX Shipping_info
ON Shipping (shipping_street,shipping_city,shipping_province,shipping_postalcode);
```

Trigger

```
--/Trigger 01/
CREATE OR REPLACE TRIGGER
trg_before_order_insert
BEFORE INSERT
        on orders
        FOR EACH ROW
DECLARE
BEGIN
        -- Allow only past date of shipping
        IF(:new.order_date > sysdate) THEN
                RAISE_APPLICATION_ERROR(-20000,'Date of order can not be Future date.');
        END IF;
END;


--/Trigger 02/
CREATE OR REPLACE TRIGGER limited_quantity
BEFORE INSERT ON orderdetail
for each row
DECLARE
BEGIN
        IF :new.quantity > 50 THEN
                RAISE_APPLICATION_ERROR(-20105,'cant order more than 50 product for once, please
contact to the assistant to get more info');
        END IF;
END;
```

Procedures

```
/*Procedure 01*/
--/Check the availability of the item/
CREATE OR REPLACE
PROCEDURE stock_available
( prod_id IN NUMBER,
  prod_qty IN NUMBER
)

IS
        stock_num Product.product_id%TYPE;
BEGIN
        SELECT inventory
        INTO stock_num
        FROM Product
        where product_id = prod_id;

        IF prod_qty > stock_num THEN
                RAISE_APPLICATION_ERROR(-20000, 'Sorry. Not Enough in stock now.' ||
                        'Requested order quantity = ' || prod_qty || 'Current stock amount = ' ||
                        stock_num);
        END IF;

EXCEPTION
        WHEN NO_DATA_FOUND THEN
                DBMS_OUTPUT.PUT_LINE('No Stock Found.');
END;


/*Test the Procedure 01*/
BEGIN
        stock_available(20006, 20);
END;
```

```
/*Procedure 02*/
--/Checking shipping status/
CREATE OR REPLACE
PROCEDURE check_status
( ord_id IN NUMBER,
  display_status OUT BOOLEAN
)
IS
        shipping_status Orders.status%TYPE;
BEGIN
        SELECT status
        INTO shipping_status
        FROM Orders
        WHERE ord_id = order_id;

        IF shipping_status = 'Y' THEN
                display_status := TRUE;
        ELSE display_status := FALSE;
        END IF;

        DBMS_OUTPUT.PUT_LINE('Is the selected item in transit ?' || ' ' ||
                                CASE display_status
                                WHEN TRUE THEN 'Yes. In transit.'
                                ELSE 'Not Yet.'
                                END);
END;
```

```
/*Test the Procedure 02*/
DECLARE
        testing BOOLEAN;
BEGIN
        check_status(60018, testing);
END;
```

Functions

```
--/Function 01/
CREATE OR REPLACE
FUNCTION order_amt_cal
(o_id IN number,
p_price IN number,
p_qty IN number
)
RETURN NUMBER
IS
        total_price NUMBER(8,2);
BEGIN
        SELECT p_price*p_qty*1.13 INTO total_price
        FROM product JOIN orderdetail USING(product_id)
        GROUP BY order_id
        Having order_id=o_id;
        RETURN total_price;
END;
```

```
/*Test Function 01*/
SELECT SUM(order_amt_cal(60023,prices,quantity))"Order amount" from product JOIN orderdetail
USING(product_id)
GROUP BY order_id
HAVING order_id=60023;
```

```
--/Function 02/
CREATE OR REPLACE FUNCTION num_orders_sf
(p_customer_ID IN number
)
RETURN number
IS
        or_count NUMBER(10);
BEGIN
        SELECT COUNT(*)
        INTO or_count
        FROM orders
        WHERE status = 'Y' and CUSTOMER_ID = p_customer_ID
        GROUP BY customer_id;
        RETURN (or_count);
END;
```

```
/*Test Function 02*/
DECLARE
	test_customer_id NUMBER(8,0):=1110;
BEGIN
	DBMS_OUTPUT.PUT_LINE('The total number of orders is:'|| ' '
||num_orders_sf(test_customer_id));
END;
```