**Group members:**

| **Name** | **Email** |
| --- | --- |
| Nick Babcock | nb987@drexel.edu |
| Samyak Jain | sj976@drexel.edu |
| Mithila Chintala | mc4343@drexel.edu |
| Ibrahim Alabi | ia426@drexel.edu |
| Nawaf Alhumaid | nka39@drexel.edu |

**Project Description**

In today’s world, the e-commerce sector of any business is arguably the most important and can be a huge contributor to one’s success. With technology getting better every year and with the situation in our world today, people are purchasing goods and services online more than ever before. This high volume of users and transactions creates a demand for a highly functional system or database that can be dependable and easily accessible in order to have a positive impact on one’s business.

Our database will help manage the online company Flipkart and needs to efficiently keep track of the customers browsing our sellers products along with each order they make. Each customer has an id, name, phone number, email, age, and address. Also, we need to track our sellers' information. Each seller has a unique identifier, name, phone number and email. The product information should be updated in real time to display current inventory, name, price, descriptions, and id.

For each transaction that is made, our database will record the type of transaction, the product(s), the customer, and the cost associated with it. It will also update our current inventory as necessary. Tracking orders is essential to updating stock, being transparent with a customer’s order information, and allowing for returns to be easily processed. Each order has an order id, status, total, and date and shows the quantities of products which were ordered. We also need to keep track of product information in order to enhance the management of sellers inventory.

Each order made in our system will automatically generate an invoice which will show a summary of the products ordered, the date it was made, and the total amount. This invoice can then have a payment made from the customer that will allow different types of payments such as credit cards or gift cards. After the order is made and an invoice is generated, this information is then sent over to our delivery company who will ultimately deliver the products to the customer.

The more efficient work our database does in regards to updating inventory and tracking purchases and shipments, the less work our management team needs to do manually. This should minimize the aspects of human error and time management, and thus maximize the profitability of our business.

We don’t want to include a sophisticated inventory management in our database. We just want to include the number of available stock of each product and display this to our customers. We also don’t interested in keep tracking of the shipments history and geographic locations of each delivery. We will update the order as it is out for delivery and when it is delivered, but a more detailed tracking will be unnecessary.

Some queries that we are interested in would be ones that allow us to have detailed reports about the total sales of our sellers. We want to focus on the categories and the sellers or each product. Also, we want to have reports for the delivery companies in order to track the delivered orders so we can improve the performance and the quality of the service.

**The Database Requirements**

We need to track the sellers, products, categories, orders, customers, payments, invoices, and delivery companies. We want to make sure our database is designed following these constraints:

* The seller doesn’t have to sell any products, but the seller can sell many products. On the other hand, a product is limited to only one seller.
* The product can belong to many categories and each category can have many products. But a product doesn’t have to belong to any category. It is also possible that a category does not contain any product.
* There must be at least one product or many for every order also, we want to capture the product quantity and the price at the time of the purchase. Orders might be assigned to a single delivery company. The delivery company can deliver many orders or none.
* For any customer, they may place zero orders, but they can also place many orders. But every individual order can only be placed by a single customer.
* For every customer, they may make zero payments but can also make many payments. Each payment can only be made by just one customer.
* An order can only have one invoice and each invoice only has one order.
* Every invoice has at least one payment but an invoice can include many payments. But each payment only has or belongs to one invoice.

**CONCEPTUAL DESIGN**

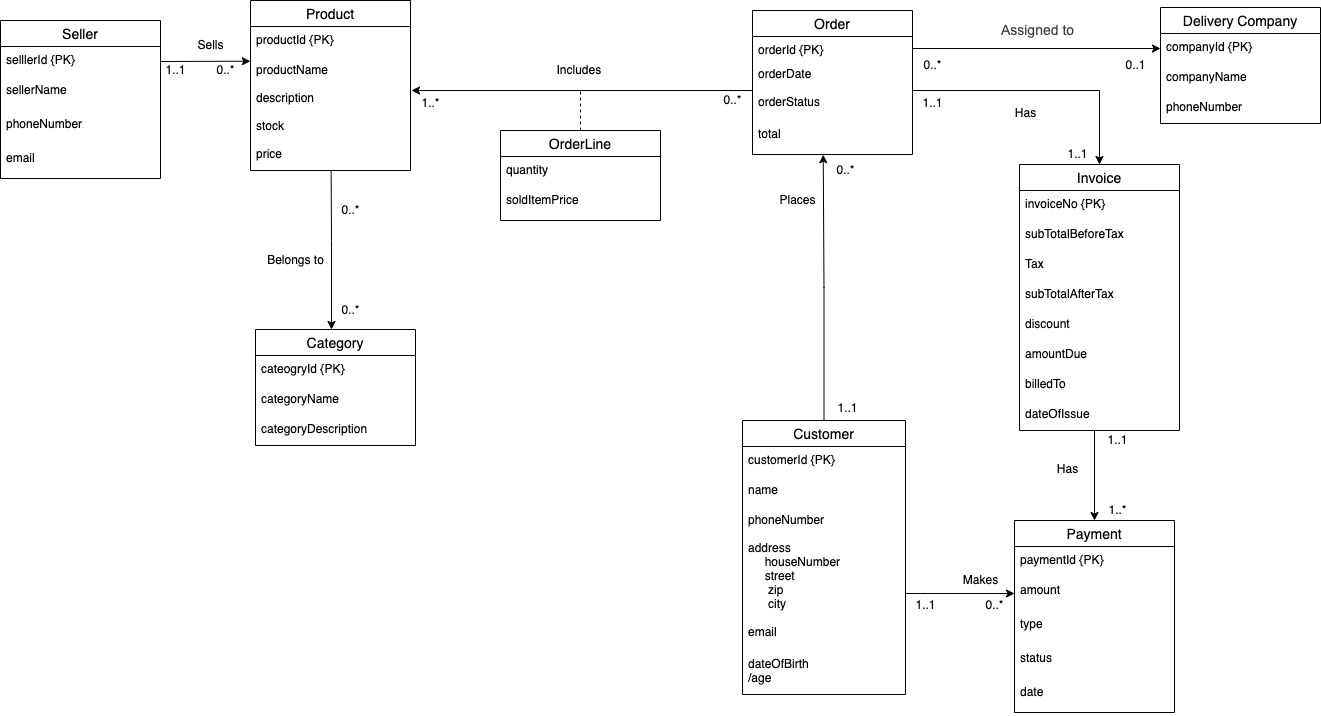
**ER Diagram**

Diagram

Description automatically generated

**PART II**

**UPDATED ERD**



**Relational Schema**

Relational Schema 3NF:

All tables are 1NF because all attributes are atomic and have no repeating groups, as well as all rows, are unique identifiers.

All tables are in 2NF because it has a single attribute of a candidate key, and they are fully functional dependencies.

All tables are in 3NF because it has no transitive dependencies.

* Seller(sellerId, sellerName, phoneNumber, email)
* Product(productId, productName, description, stock, price, sellerId ) **sellerId references sellerId from table Seller**
* Category(categoryId, categoryName, categoryDescription)
* Product\_In\_Category(productId, categoryId) **productId and categoryId references productId and cateogryId from tables Product and Category respectively.**
* Orders(orderId, orderDate, orderStatus, total, customerId, companyId) **customerId and companyId references customerId and companyId from tables Customer and Delivery\_Company respectively**
* Product\_In\_Order(productId, orderId, quantity, soldItemPrice) **productId and order Id references productId and orderId from tables Product and Orders respectively.**
* Customer(customerId, name, phoneNumber, addressHouseNumber, addressStreet, addressZip, addressCity, email, dateOfBirth)
* Payment(paymentId, amount, payment\_type, status, payment\_date, customerId, invoiceNo) **customerId and invoiceNo references customerId and invoiceNo from table Customer and Invoice respectively.**
* Delivery\_Company(companyId, companyName, phoneNumber)
* Invoice(invoiceNo, subTotalBeforeTax, Tax, SubTotalAfterTax, discount, amountDue, billedTo, dateOfIssue, orderId) **orderId references orderId from table Order.**

**Data Dictionary**

**Product:** Contains information about the product

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Column** | **Description** | **DataType** | **Domain** | **Nullable** | **PK** | **FK** |
| PRODUCTID | ID of product | CHAR(12) | All | No | Yes | No |
| PRODUCTNAME | Name of product | VARCHAR(25) | All | No | No | No |
| DESCRIPTION | Product description | VARCHAR(50) | All | Yes | No | No |
| STOCK | # available | NUMBER | All | Yes | No | No |
| PRICE | price | NUMBER | All | No | No | No |
| SELLERID | ID of seller | CHAR(12) | All | Yes | No | Yes |

|  |  |  |
| --- | --- | --- |
| **Constraint** | **Type** | **On Delete** |
| PRODUCT\_PK | Primary Key | - |
| PRODUCT\_FK\_SELLERID | Foreign Key | SET NULL |
| PRODUCT\_NN\_PRODUCTNAME | NOT NULL | - |
| PRODUCT\_NN\_PRICE | NOT NULL | - |

**Seller:** Contains information about the product’s seller

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Column** | **Description** | **DataType** | **Domain** | **Nullable** | **PK** | **FK** |
| SELLERID | ID of seller | CHAR(12) | All | No | Yes | No |
| SELLERNAME | Name of seller | VARCHAR(25) | All | Yes | No | No |
| PHONENUMBER | Seller phone # | CHAR(10) | All | Yes | No | No |
| EMAIL | Seller email | VARCHAR(25) | All | No | No | No |

|  |  |  |
| --- | --- | --- |
| **Constraint** | **Type** | **On Delete** |
| SELLER\_PK | Primary Key | - |
| SELLER\_NN\_EMAIL | NOT NULL | - |

**Category:** Contains information about the product’s category

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Column** | **Description** | **DataType** | **Domain** | **Nullable** | **PK** | **FK** |
| CATEGORYID | ID of category | CHAR(12) | All | No | Yes | No |
| CATEGORYNAME | Category name | VARCHAR(12) | All | No | No | No |
| CATEGORYDESCRIPTION | Category description | VARCHAR(50) | All | Yes | No | No |

|  |  |  |
| --- | --- | --- |
| **Constraint** | **Type** | **On Delete** |
| CATEGORY\_PK | Primary Key | - |
| CATEGORY\_NN\_CATEGORYNAME | NOT NULL | - |

**Product in Category:** Contains information about the product’s in a category

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Column** | **Description** | **DataType** | **Domain** | **Nullable** | **PK** | **FK** |
| PRODUCTID | ID of product | CHAR(12) | All | No | No | Yes |
| CATEGORYID | ID of category | CHAR(12) | All | No | No | Yes |

|  |  |  |
| --- | --- | --- |
| **Constraint** | **Type** | **On Delete** |
| PIC\_FK\_PRODUCTID | Foreign Key | CASCADE |
| PIC\_FK\_CATEGORYID | Foreign Key | CASCADE |
| PIC\_PK | Primary Key | - |

**Customer:** Contains information about the customer

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Column** | **Description** | **DataType** | **Domain** | **Nullable** | **PK** | **FK** |
| CUSTOMERID | ID of customer | CHAR(12) | All | No | Yes | No |
| NAME | Name of customer | VARCHAR(25) | All | Yes | No | No |
| PHONENUMBER | Customer phone # | CHAR(10) | All | Yes | No | No |
| ADDRESSHOUSENUMBER | House number | VARCHAR(12) | All | Yes | No | No |
| ADDRESSSTREET | Street name | VARCHAR(25) | All | Yes | No | No |
| ADDRESSZIP | Zip code | NUMBER | All | Yes | No | No |
| ADDRESSCITY | City | VARCHAR(12) | All | Yes | No | No |
| EMAIL | Customer email | VARCHAR(25) | All | No | No | No |
| DATEOFBIRTH | Customer DOB | DATE | All | Yes | No | No |

|  |  |  |
| --- | --- | --- |
| **Constraint** | **Type** | **On Delete** |
| CUSTOMER\_PK | Primary Key | - |
| CUSTOMER\_NN\_EMAIL | NOT NULL | - |

**Orders:** Contains information about the customer’s order

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Column** | **Description** | **DataType** | **Domain** | **Nullable** | **PK** | **FK** |
| ORDERID | ID of order | CHAR(12) | All | No | Yes | No |
| ORDERDATE | Date of order | DATE | All | No | No | No |
| ORDERSTATUS | Status of order | VARCHAR(3) | ‘P’, ‘C’, ‘OFD’, ‘D’ | No | No | No |
| TOTAL | Order total | NUMBER | All | No | No | No |
| CUSTOMERID | ID of customer | CHAR(12) | All | Yes | No | Yes |
| COMPANYID | Id of delivery company | CHAR(12) | All | Yes | No | Yes |

**NOTE:** In ‘ORDERSTATUS’, ‘P’ = Pending, ‘C’ = Confirmed, ‘OFD’ = OutForDelivery, ‘D’ = Delivered

|  |  |  |
| --- | --- | --- |
| **Constraint** | **Type** | **On Delete** |
| ORDER\_PK | Primary Key | - |
| ORDER\_FK\_CUSTOMERID | Foreign Key | SET NULL |
| ORDER\_FK\_COMPANYID | Foreign Key | SET NULL |
| ORDER\_NN\_ORDERDATE | NOT NULL | - |
| ORDER\_NN\_ORDERSTATUS | NOT NULL | - |
| ORDER\_NN\_TOTAL | NOT NULL | - |

**Product in Order:** Contains information about the product in the customer’s order

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Column** | **Description** | **DataType** | **Domain** | **Nullable** | **PK** | **FK** |
| PRODUCTID | ID of product | CHAR(12) | All | No | No | Yes |
| ORDERID | ID of order | CHAR(12) | All | No | No | Yes |
| QUANTITY | # of products in order | NUMBER | All | No | No | No |
| SOLDITEMPRICE | Price of product ordered | NUMBER | All | No | No | No |

|  |  |  |
| --- | --- | --- |
| **Constraint** | **Type** | **On Delete** |
| PIO\_FK\_PRODUCTID | Foreign Key | CASCADE |
| PIO\_FK\_CATEGORYID | Foreign Key | CASCADE |
| PIO\_PK | Primary Key | - |
| PIO\_NN\_QUANTITY | NOT NULL | - |
| PIO\_NN\_SOLDITEMPRICE | NOT NULL | - |

**Delivery Company:** Contains information about the order’s delivery company

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Column** | **Description** | **DataType** | **Domain** | **Nullable** | **PK** | **FK** |
| COMPANYID | ID of delivery company | CHAR(12) | All | No | Yes | No |
| COMPANYNAME | Name of delivery company | VARCHAR(25) | All | No | No | No |
| PHONENUMBER | Delivery company phone number | CHAR(10) | All | Yes | No | No |

|  |  |  |
| --- | --- | --- |
| **Constraint** | **Type** | **On Delete** |
| DELIVERYCOMPANY\_PK | Primary Key | - |
| DELIVERYCOMPANY\_NN\_COMPANYNAME | NOT NULL | - |

**Invoice:** Contains information about the order

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Column** | **Description** | **DataType** | **Domain** | **Nullable** | **PK** | **FK** |
| INVOICENO | Number of invoice | CHAR(12) | All | No | Yes | No |
| SUBTOTALBEFORETAX | Before tax total | NUMBER | All | Yes | No | No |
| TAX | Tax amount | NUMBER | All | Yes | No | No |
| SUBTOTALAFTERTAX | After tax total | NUMBER | All | Yes | No | No |
| DISCOUNT | Discount amount | NUMBER | All | Yes | No | No |
| AMOUNTDUE | Total amount due | NUMBER | All | Yes | No | No |
| BILLEDTO | Customer billed | VARCHAR(25) | All | Yes | No | No |
| DATEOFISSUE | Date of invoice | DATE | All | Yes | No | No |
| ORDERID | ID of order | CHAR(12) | All | Yes | No | Yes |

|  |  |  |
| --- | --- | --- |
| **Constraint** | **Type** | **On Delete** |
| INVOICE\_PK | Primary Key | - |
| INVOICE\_FK\_ORDERID | Foreign Key | SET NULL |

**Payment:** Contains information about the customer’s payment for the generated invoice

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Column** | **Description** | **DataType** | **Domain** | **Nullable** | **PK** | **FK** |
| PAYMENTID | ID of payment | CHAR(12) | All | No | Yes | No |
| AMOUNT | Amount payed | NUMBER | All | Yes | No | No |
| PAYMENTTYPE | Type of payment | VARCHAR(12) | All | Yes | No | No |
| STATUS | Status of payment | VARCHAR(12) | All | Yes | No | No |
| PAYMENTDATE | Date of payment | DATE | All | Yes | No | No |
| CUSTOMERID | Customer ID | CHAR(12) | All | Yes | No | Yes |
| INVOICENO | Invoice number | CHAR(12) | All | Yes | No | Yes |

|  |  |  |
| --- | --- | --- |
| **Constraint** | **Type** | **On Delete** |
| PAYMENT\_PK | Primary Key | - |
| CUSTOMERID\_FK | Foreign Key | SET NULL |
| INVOICENO\_FK | Foreign Key | SET NULL |

**Database Implementation “ DDL”**

**- Create Seller table**

CREATE TABLE Seller(

sellerId CHAR(12) CONSTRAINT seller\_pk PRIMARY KEY,

sellerName VARCHAR(25),

phoneNumber CHAR(10),

email VARCHAR(25) CONSTRAINT seller\_nn\_email NOT NULL

);

**- Create Category table**

CREATE TABLE Category(

categoryId CHAR(12) CONSTRAINT category\_pk PRIMARY KEY,

categoryName VARCHAR(12) CONSTRAINT category\_nn\_categoryname NOT NULL,

categoryDescription VARCHAR(50)

);

**- Create Customer table**

CREATE TABLE Customer(

customerId CHAR(12) CONSTRAINT customer\_pk PRIMARY KEY,

name VARCHAR(25),

phoneNumber CHAR(10),

addressHouseNumber VARCHAR(12),

addressStreet VARCHAR(25),

addressZip NUMBER,

addressCity VARCHAR(12),

email VARCHAR(25) CONSTRAINT customer\_nn\_email NOT NULL,

dateOfBirth DATE

);

**- Create Delivery Company table**

CREATE TABLE Delivery\_Company(

companyId CHAR(12) CONSTRAINT deliverycompany\_pk PRIMARY KEY,

companyName VARCHAR(25) CONSTRAINT deliverycompany\_nn\_companyname NOT NULL,

phoneNumber NUMBER

);

**- Create Product table**

CREATE TABLE Product(

productId CHAR(12) CONSTRAINT product\_pk PRIMARY KEY,

productName VARCHAR(25) CONSTRAINT product\_nn\_productname NOT NULL,

description VARCHAR(50),

stock NUMBER,

price NUMBER CONSTRAINT product\_nn\_price NOT NULL,

sellerId CHAR(12) CONSTRAINT product\_fk\_sellerid REFERENCES Seller(sellerId) ON DELETE SET NULL

);

**- Create Product In Category table**

CREATE TABLE Product\_In\_Category(

productId CHAR(12) CONSTRAINT pic\_fk\_productid REFERENCES Product(productId) ON DELETE CASCADE,

categoryId CHAR(12) CONSTRAINT pic\_fk\_categoryid REFERENCES Category(categoryId) ON DELETE CASCADE ,

CONSTRAINT pic\_pk PRIMARY KEY (productId, categoryId)

);

**- Create Orders table**

CREATE TABLE Orders(

orderId CHAR(12) CONSTRAINT order\_pk PRIMARY KEY,

orderDate DATE CONSTRAINT order\_nn\_orderdate NOT NULL,

orderStatus VARCHAR(3) CONSTRAINT order\_nn\_orderstatus NOT NULL,

total NUMBER CONSTRAINT order\_nn\_total NOT NULL,

customerId CHAR(12) CONSTRAINT order\_fk\_customerid REFERENCES Customer(customerId) ON DELETE SET NULL ,

companyId CHAR(12) CONSTRAINT order\_fk\_companyid REFERENCES Delivery\_Company(companyId) ON DELETE SET NULL

);

**- Create Product In Orders table**

CREATE TABLE Product\_In\_Orders(

productId CHAR(12) CONSTRAINT pio\_fk\_productid REFERENCES Product(productId) ON DELETE CASCADE,

orderId CHAR(12) CONSTRAINT pio\_fk\_categoryid REFERENCES Orders(orderId) ON DELETE CASCADE,

CONSTRAINT pio\_pk PRIMARY KEY (productId, orderId),

quantity NUMBER CONSTRAINT pio\_nn\_quantity NOT NULL,

soldItemPrice NUMBER CONSTRAINT pio\_nn\_solditemprice NOT NULL

);

**- Create Invoice table**

CREATE TABLE Invoice(

invoiceNo CHAR(12) CONSTRAINT invoice\_pk PRIMARY KEY,

subTotalBeforeTax NUMBER,

Tax NUMBER,

subTotalAfterTax NUMBER,

discount NUMBER,

amountDue NUMBER,

billedTo VARCHAR(25),

dateOfIssue DATE,

orderId CHAR(12) CONSTRAINT invoice\_fk\_orderid REFERENCES Orders(orderId) ON DELETE SET NULL

);

**- Create Payment table**

CREATE TABLE Payment(

paymentId CHAR(12) CONSTRAINT payment\_pk PRIMARY KEY,

amount NUMBER,

payment\_type VARCHAR(12),

status VARCHAR(12),

payment\_date DATE,

customerId CHAR(12) CONSTRAINT customerid\_fk REFERENCES Customer(customerId) ON DELETE SET NULL,

invoiceNo CHAR(12) CONSTRAINT invoiceno\_fk REFERENCES Invoice(invoiceNo) ON DELETE SET NULL

);

**Data Inserts “DML”**

**- Seller table (SAMYAK)**

* INSERT INTO Seller(sellerId, sellerName, phoneNumber, email) VALUES('123456789', 'Robert', '10000001', 'robert@gmail.com');
* INSERT INTO Seller(sellerId, sellerName, phoneNumber, email) VALUES('234567890', 'Pattinson', '10000002', 'pattinson@gmail.com');
* INSERT INTO Seller(sellerId, sellerName, phoneNumber, email) VALUES('345678901', 'Paul', '10000003', 'paul@gmail.com');
* INSERT INTO Seller(sellerId, sellerName, phoneNumber, email) VALUES('456789012', 'Dano', '10000004', 'dano@gmail.com');
* INSERT INTO Seller(sellerId, sellerName, phoneNumber, email) VALUES('567890123', 'Zoe', '10000005', 'zoe@gmail.com');

-------

**- Category table (SAMYAK)**

* INSERT INTO Category(categoryId, categoryName, categoryDescription) VALUES('1', 'Dry Skincare', 'For Dry Skin');
* INSERT INTO Category(categoryId, categoryName, categoryDescription) VALUES('2', 'Haircare', 'For Silky Hair');
* INSERT INTO Category(categoryId, categoryName, categoryDescription) VALUES('3', 'Toys', 'Children’s toys');
* INSERT INTO Category(categoryId, categoryName, categoryDescription) VALUES('4', 'Skincare', 'For Smooth Skin');
* INSERT INTO Category(categoryId, categoryName, categoryDescription) VALUES('5', 'Tan Skincare', 'For Tan Skin');

--------

**- Customer table (MITHILA)**

* INSERT INTO Customer(customerId, name, phoneNumber,addressHouseNumber, addressStreet, addressZip, addressCity, email, dateOfBirth)

VALUES('12345678', 'Jack', '1234567890', '45', '34th Chestnut', 10904, 'Philadelphia', 'jack@gmail', '09-JUN-1997');

* INSERT INTO Customer(customerId, name, phoneNumber,addressHouseNumber, addressStreet, addressZip, addressCity, email, dateOfBirth)

VALUES('12345677', 'Colin', '1234566666', '24', '33rd Walnut', 10945,'Philadelphia','colin@gmail', '01-JUL-1969');

* INSERT INTO Customer(customerId, name, phoneNumber,addressHouseNumber, addressStreet, addressZip, addressCity, email, dateOfBirth)

VALUES('12345688', 'Jaffrey', '1234577777', '63', '32nd Chestnut', 10974,'Philadelphia','jaffrey@gmail', '22-DEC-1988');

* INSERT INTO Customer(customerId, name, phoneNumber,addressHouseNumber, addressStreet, addressZip, addressCity, email, dateOfBirth)

VALUES('12345666', 'Barry', '1234588888', '73', '31st Spruce', 10910,'NYC','barry@gmail', '30-MAY-1999');

* INSERT INTO Customer(customerId, name, phoneNumber,addressHouseNumber, addressStreet, addressZip, addressCity, email, dateOfBirth)

VALUES('12345555', 'Andy', '1234599999', '987', '30th Walnut', 10908,'Philadelphia','andy@gmail', '15-OCT-1981');

---------

**- Delivery Company table (MITHILA)**

* INSERT INTO Delivery\_Company(companyId, companyName, phoneNumber) VALUES('1212121212', 'Neutrogena', '1111111111');
* INSERT INTO Delivery\_Company(companyId, companyName, phoneNumber) VALUES('1313131313', 'Panteen', '2222222222');
* INSERT INTO Delivery\_Company(companyId, companyName, phoneNumber) VALUES('1414141414', 'DHL', '3333333333');
* INSERT INTO Delivery\_Company(companyId, companyName, phoneNumber) VALUES('1515151515', 'BathandBody', '4444444444');
* INSERT INTO Delivery\_Company(companyId, companyName, phoneNumber) VALUES('1616161616', 'The Body Shop', '5555555555');

--------

**- Product table (NICK)**

* INSERT INTO Product(productId, productName, description, stock, price, sellerId) VALUES('1122334455', 'Bodywash', 'Glowing Skin', 20, 7.99, '123456789');
* INSERT INTO Product(productId, productName, description, stock, price, sellerId) VALUES('2233445566', 'Shampoo', 'Anti Hairfall', 30, 8.99, '234567890');
* INSERT INTO Product(productId, productName, description, stock, price, sellerId) VALUES('3344556677', 'Conditioner', 'Shining Hair', 40, 8.99, '345678901');
* INSERT INTO Product(productId, productName, description, stock, price, sellerId) VALUES('4455667788', 'Moisturizer', 'Soft and Healthy', 50, 5.99, '456789012');
* INSERT INTO Product(productId, productName, description, stock, price, sellerId) VALUES('5566778899', 'Sunscreen', 'Damage Control', 60, 4.99, '567890123');

--------

**- Product In Category table (NICK)**

* INSERT INTO Product\_In\_Category(productId, categoryId) VALUES('1122334455', 1);
* INSERT INTO Product\_In\_Category(productId, categoryId) VALUES('2233445566', 2);
* INSERT INTO Product\_In\_Category(productId, categoryId) VALUES('3344556677', 3);
* INSERT INTO Product\_In\_Category(productId, categoryId) VALUES('4455667788', 4);
* INSERT INTO Product\_In\_Category(productId, categoryId) VALUES('5566778899', 5);

--------

**- Order table (NAWAF)**

* INSERT INTO Orders(orderId, orderDate, orderStatus, total, customerId, companyId) VALUES('1000000001', '01-JAN-2000', 'D', 21.99, '12345678', '1212121212');
* INSERT INTO Orders(orderId, orderDate, orderStatus, total, customerId, companyId) VALUES('2000000002', '02-FEB-2001', 'OFD', 19.99, '12345677', '1313131313');
* INSERT INTO Orders(orderId, orderDate, orderStatus, total, customerId, companyId) VALUES('3000000003', '03-MAR-2002', 'D', 9.99, '12345688', '1414141414');
* INSERT INTO Orders(orderId, orderDate, orderStatus, total, customerId, companyId) VALUES('4000000004', '01-APR-2003', 'P', 13.99, '12345666', '1515151515');
* INSERT INTO Orders(orderId, orderDate, orderStatus, total, customerId, companyId) VALUES('5000000005', '01-MAY-2004', 'C', 11.99, '12345555', '1616161616');

--------

**- Product In Order table (NAWAF)**

* INSERT INTO Product\_In\_Orders(productId, orderId, quantity, soldItemPrice) VALUES('1122334455', '1000000001', 2, 7.99);
* INSERT INTO Product\_In\_Orders(productId, orderId, quantity, soldItemPrice) VALUES('2233445566', '2000000002', 3, 8.99);
* INSERT INTO Product\_In\_Orders(productId, orderId, quantity, soldItemPrice) VALUES('3344556677', '3000000003', 4, 8.99);
* INSERT INTO Product\_In\_Orders(productId, orderId, quantity, soldItemPrice) VALUES('4455667788', '4000000004', 5, 5.99);
* INSERT INTO Product\_In\_Orders(productId, orderId, quantity, soldItemPrice) VALUES('5566778899', '5000000005', 6, 4.99);

----------

**- Invoice table (IBRAHIM)**

* INSERT INTO Invoice(invoiceNo, subTotalBeforeTax, Tax, SubTotalAfterTax, discount, amountDue, billedTo, dateOfIssue, orderId) VALUES('1000000000', 9.99, 2.99, 12.98, 1, 11.98, 'Jack', '02-JAN-2021', '1000000001');
* INSERT INTO Invoice(invoiceNo, subTotalBeforeTax, Tax, SubTotalAfterTax, discount, amountDue, billedTo, dateOfIssue, orderId) VALUES('2000000000', 19.99, 2.99, 22.98, 1, 21.98, 'Colin', '03-JUN-2021', '2000000002');
* INSERT INTO Invoice(invoiceNo, subTotalBeforeTax, Tax, SubTotalAfterTax, discount, amountDue, billedTo, dateOfIssue, orderId) VALUES('3000000000', 29.99, 2.99, 32.98, 1, 31.98, 'Jaffrey', '04-APR-2021', '3000000003');
* INSERT INTO Invoice(invoiceNo, subTotalBeforeTax, Tax, SubTotalAfterTax, discount, amountDue, billedTo, dateOfIssue, orderId) VALUES('4000000000', 39.99, 2.99, 42.98, 1, 41.98, 'Barry', '05-SEP-2021', '4000000004');
* INSERT INTO Invoice(invoiceNo, subTotalBeforeTax, Tax, SubTotalAfterTax, discount, amountDue, billedTo, dateOfIssue, orderId) VALUES('5000000000', 49.99, 2.99, 52.98, 1, 51.98, 'Andy', '06-OCT-2021', '5000000005');

--------  
**- Payment table (IBRAHIM)**

* INSERT INTO Payment(paymentId, amount, payment\_type, status, payment\_date, customerId, invoiceNo) VALUES('1000000001', 10.99, 'Cash', 'Paid', '09-DEC-2021', '12345678', '1000000000');
* INSERT INTO Payment(paymentId, amount, payment\_type, status, payment\_date, customerId, invoiceNo) VALUES('2000000002', 20.99, 'Debit', 'Paid', '08-NOV-2021', '12345677', '2000000000');
* INSERT INTO Payment(paymentId, amount, payment\_type, status, payment\_date, customerId, invoiceNo) VALUES('3000000003', 30.99, 'Credit', 'Paid', '07-OCT-2021', '12345688', '3000000000');
* INSERT INTO Payment(paymentId, amount, payment\_type, status, payment\_date, customerId, invoiceNo) VALUES('4000000004', 40.99, 'Cash', 'Paid', '06-SEP-2021', '12345666', '4000000000');
* INSERT INTO Payment(paymentId, amount, payment\_type, status, payment\_date, customerId, invoiceNo) VALUES('5000000005', 50.99, 'Cash', 'Paid', '05-AUG-2021', '12345555', '5000000000');

**Data Queries**

**How many orders do we have?**

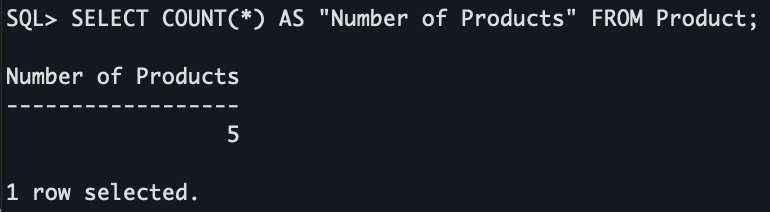
* SELECT COUNT(\*) AS "Number of Orders" FROM Orders;

Text

Description automatically generated

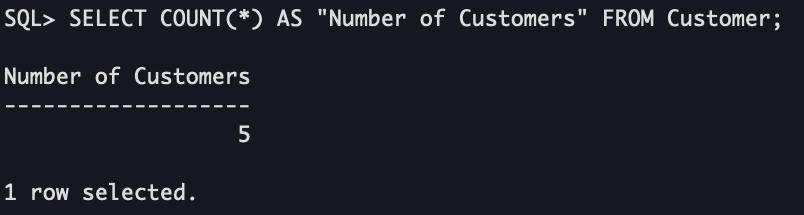
**How many Products do we have?**

* SELECT COUNT(\*) AS "Number of Products" FROM Product;



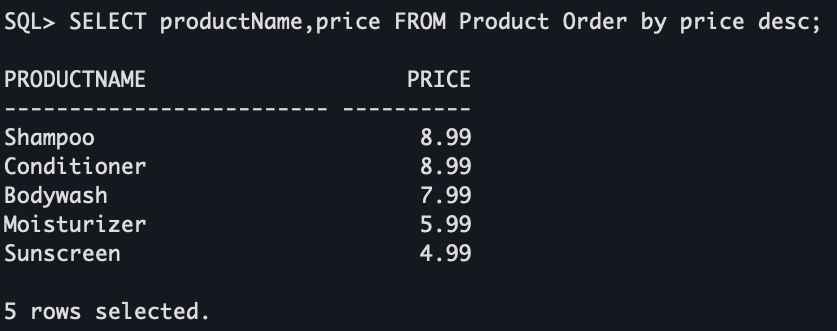
**How many Customers do we have?**

* SELECT COUNT(\*) AS "Number of Customers" FROM Customer;



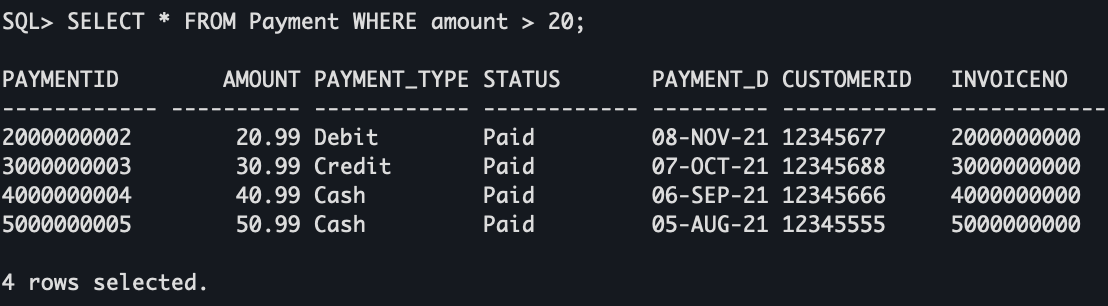
**What is the Products that were ordered for the highest price?**

* SELECT productName,price FROM Product Order by price desc;



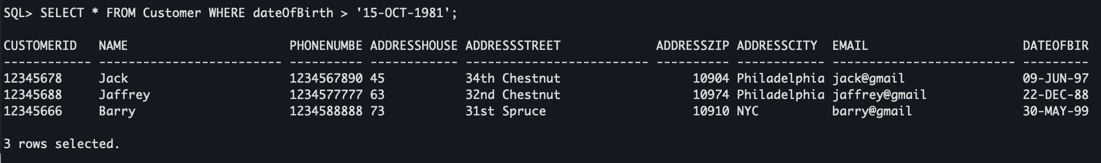
**Show me the Payments greater than $20**

* SELECT \* FROM Payment WHERE amount > 20;



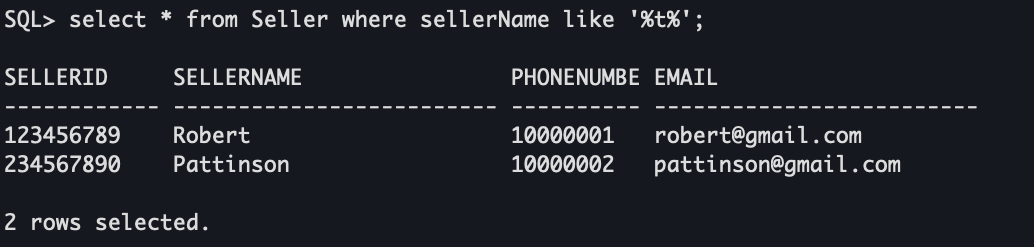
**Show me the Customers born after 15-OCT-1981**

* SELECT \* FROM Customer WHERE dateOfBirth > '15-OCT-1981';



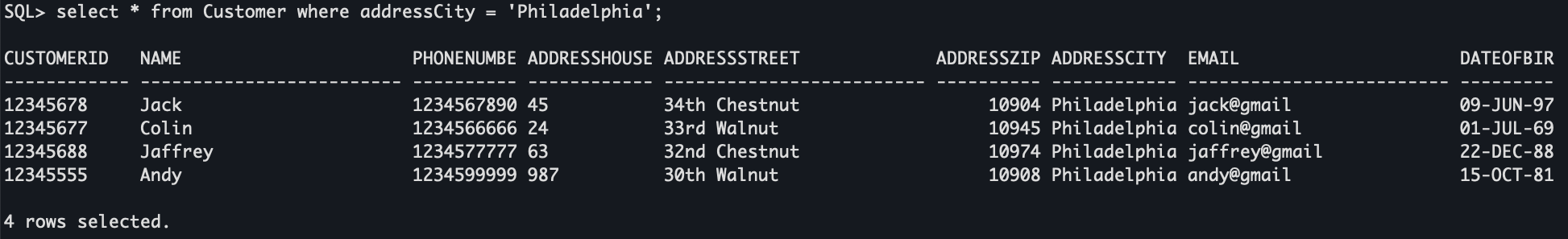
**Show me the seller with a name like t?**

* select \* from Seller where sellerName like '%t%';



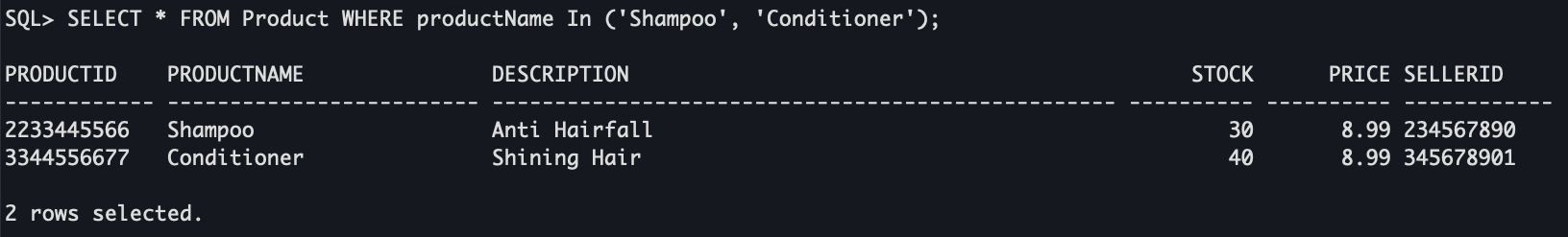
**Show me the customer we have in Philadelphia?**

* select \* from Customer where addressCity = 'Philadelphia';



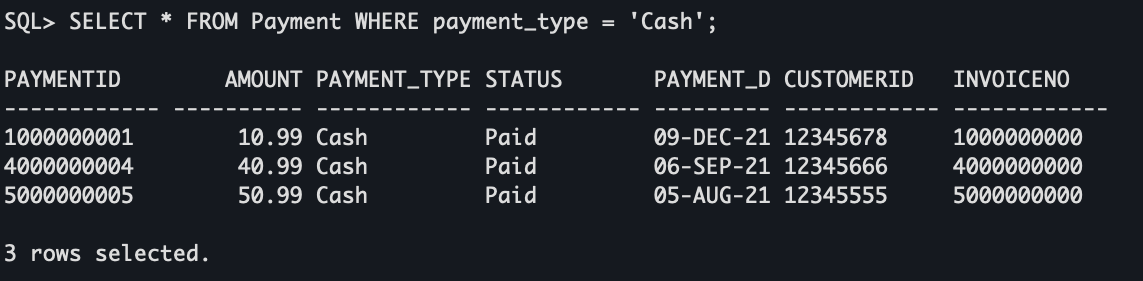
**Show me Shampoo or Conditioner Products?**

* SELECT \* FROM Product WHERE productName In ('Shampoo', 'Conditioner');



**Show me Cash Payments?**

* SELECT \* FROM Payment WHERE payment\_type = 'Cash';



**Total sales per Category?**

* select

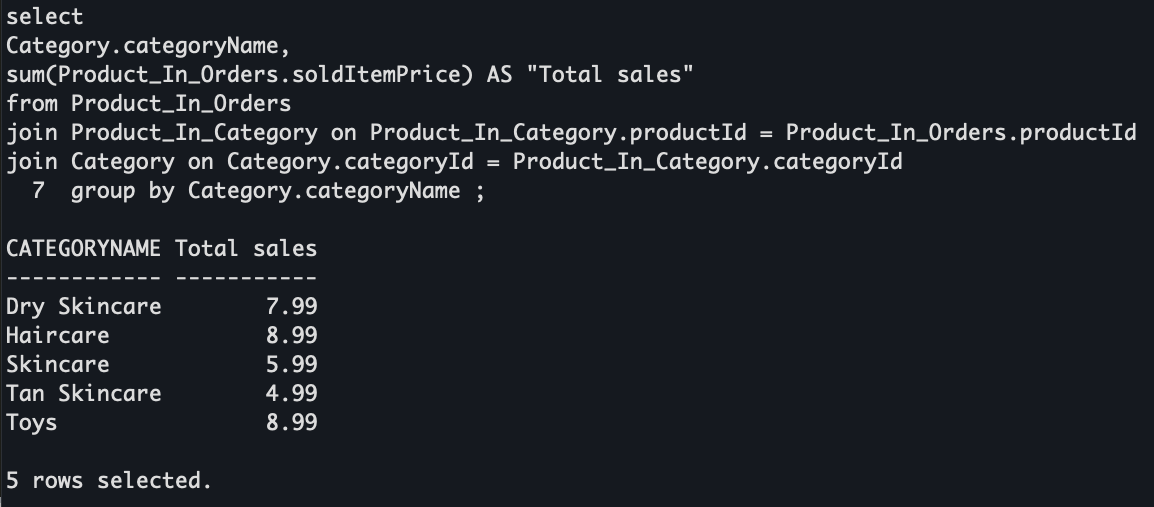
Category.categoryName,

sum(Product\_In\_Orders.soldItemPrice) AS "Total sales"

from Product\_In\_Orders

join Product\_In\_Category on Product\_In\_Category.productId = Product\_In\_Orders.productId

join Category on Category.categoryId = Product\_In\_Category.categoryId

group by Category.categoryName ;

**Customers orders?**

* SELECT

Customer.customerId,

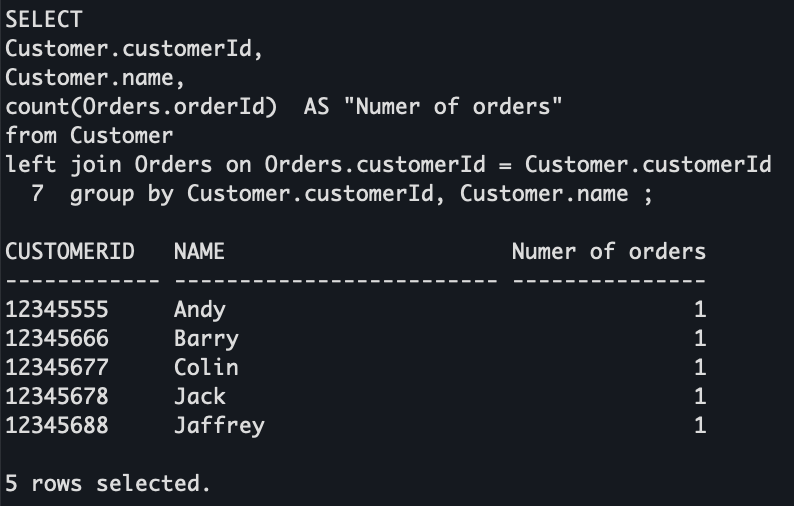
Customer.name,

count(Orders.orderId) AS "Numer of orders"

from Customer

left join Orders on Orders.customerId = Customer.customerId

group by Customer.customerId, Customer.name;



**Total sales per seller?**

* select

Seller.sellerName,

sum(Product\_In\_Orders.soldItemPrice) AS "Total sales"

from Seller

join Product on Seller.sellerId = Product.sellerId

join Product\_In\_Orders on Product\_In\_Orders.productId = Product.productId

group by Seller.sellerName ;



**Number of products per Category**

* select

Category.categoryName,

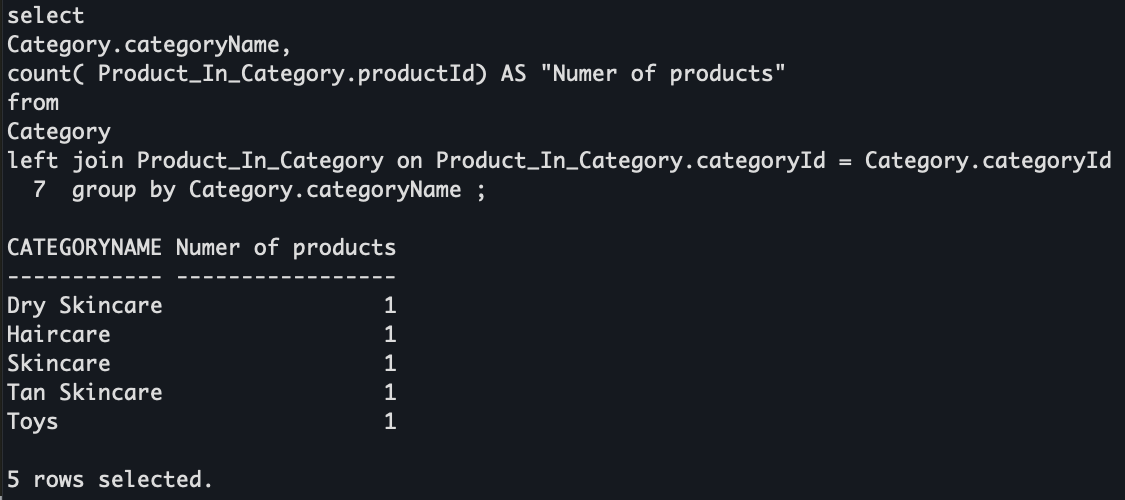
count( Product\_In\_Category.productId) AS "Numer of products"

from

Category

left join Product\_In\_Category on Product\_In\_Category.categoryId = Category.categoryId

group by Category.categoryName ;



**Number of delivered orders per Delivery Company**

* SELECT

Delivery\_Company.companyName,

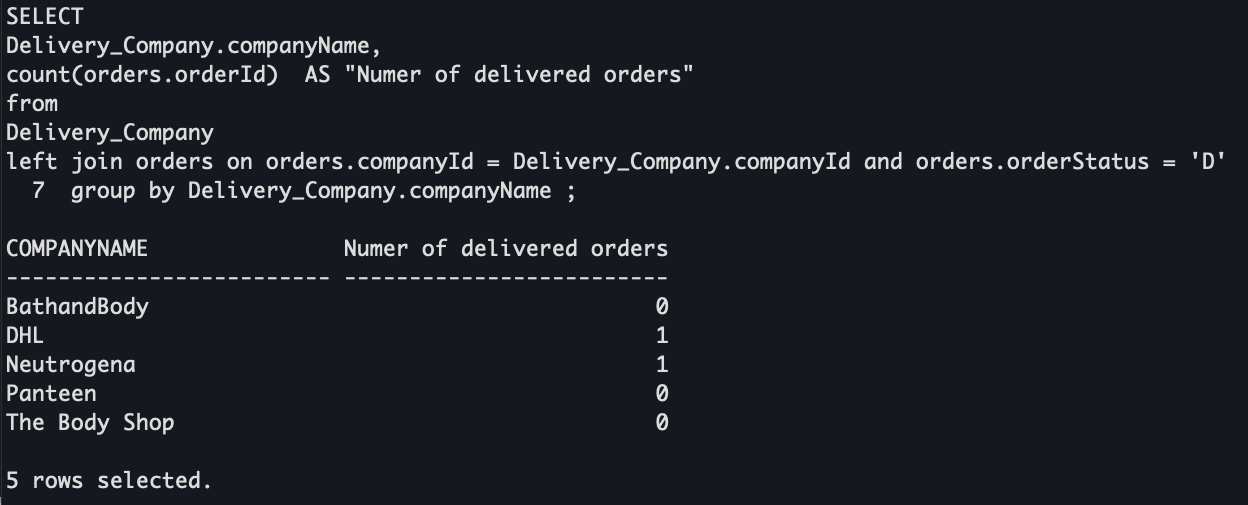
count(orders.orderId) AS "Numer of delivered orders"

from

Delivery\_Company

left join orders on orders.companyId = Delivery\_Company.companyId and orders.orderStatus = 'D'

group by Delivery\_Company.companyName ;



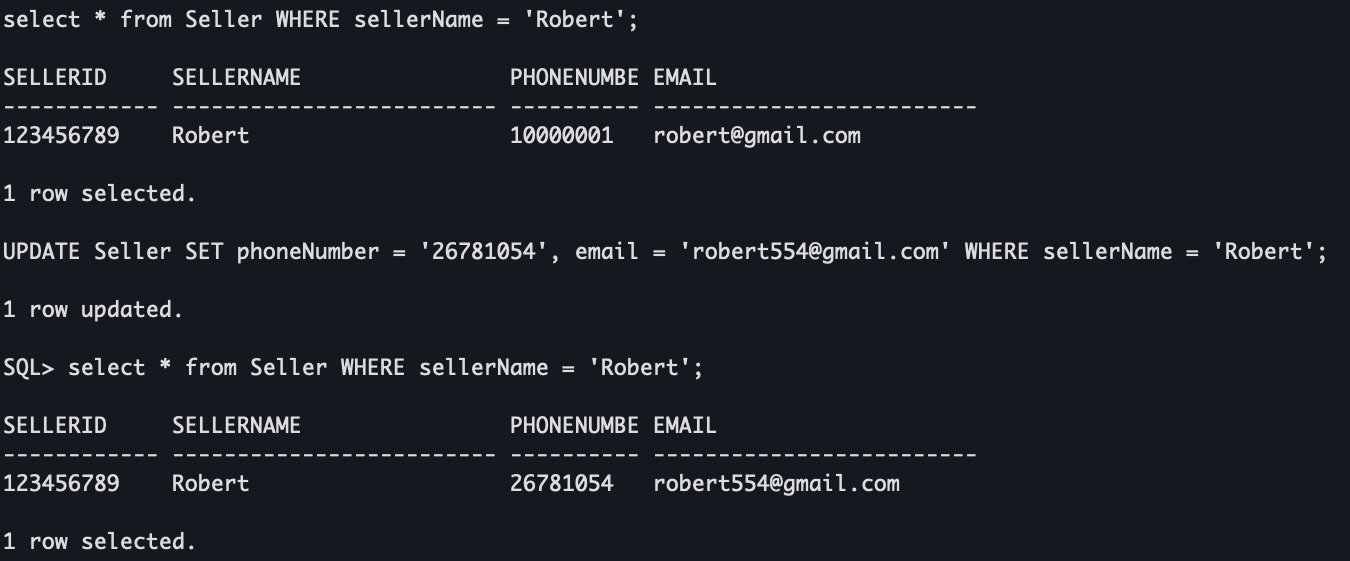
**Data Manipulations**

Update

**Nick**

select \* from Seller WHERE sellerName = 'Robert';

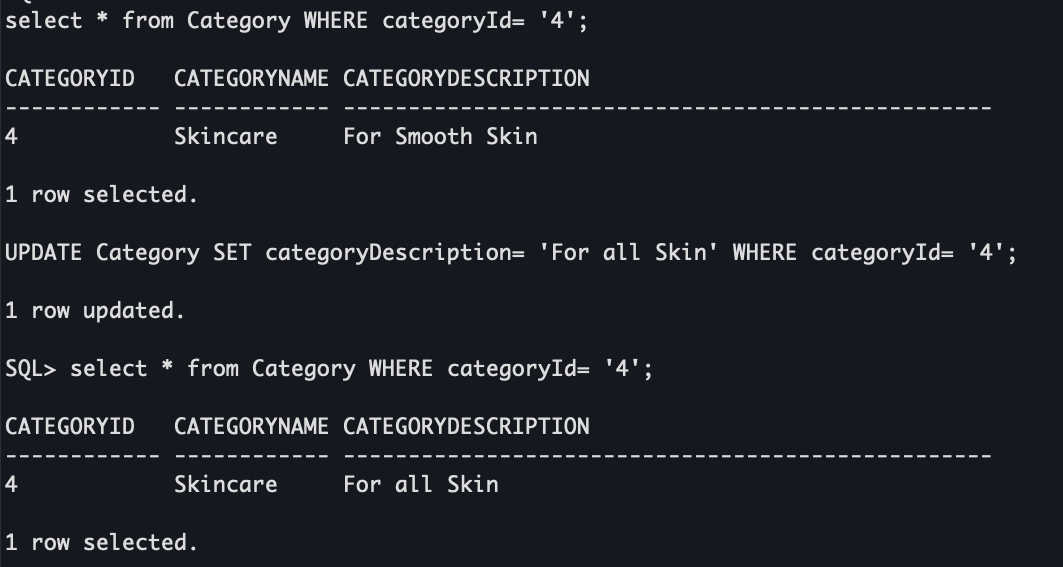
UPDATE Seller SET phoneNumber = '26781054', email = 'robert554@gmail.com' WHERE sellerName = 'Robert';

select \* from Seller WHERE sellerName = 'Robert'; 

**Nawaf**

select \* from Category WHERE categoryId= '4';

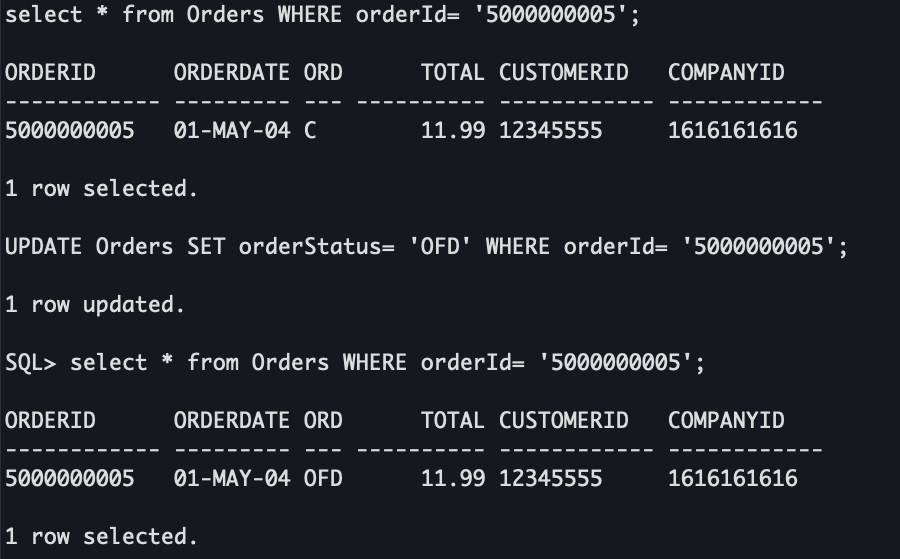
UPDATE Category SET categoryDescription= 'For all Skin' WHERE categoryId= '4';

select \* from Category WHERE categoryId= '4'; 

**Samyak**

select \* from Orders WHERE orderId= '5000000005';

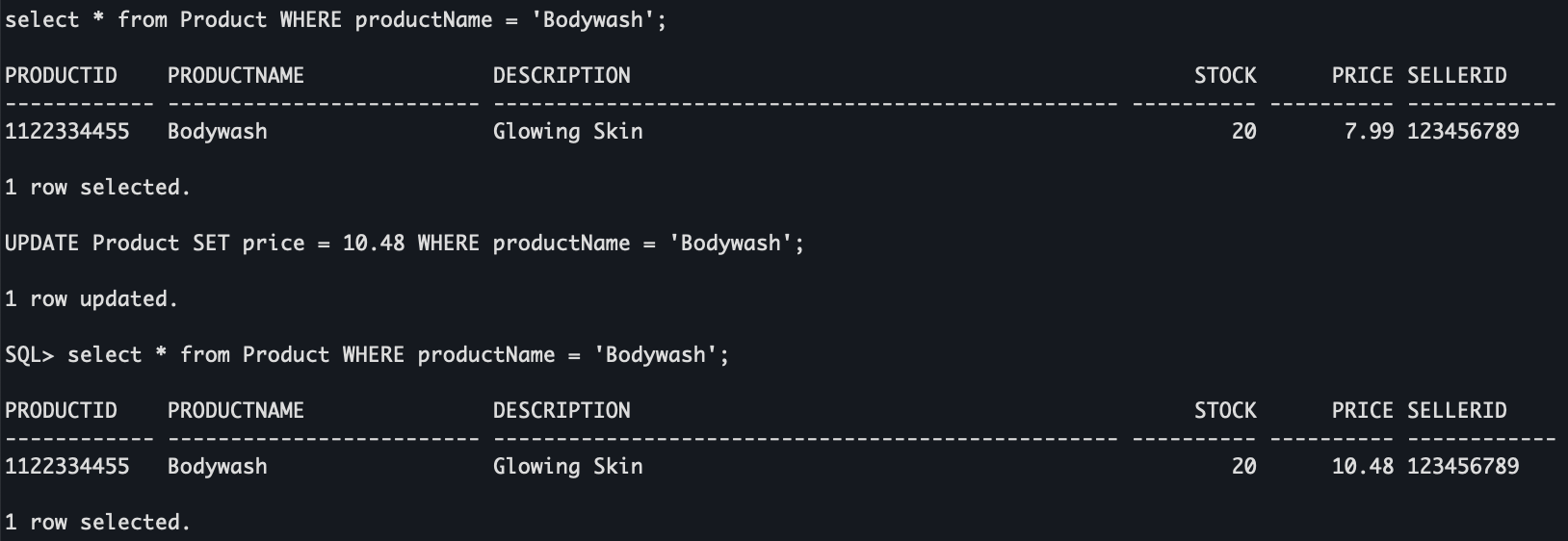
UPDATE Orders SET orderStatus= 'OFD' WHERE orderId= '5000000005';

select \* from Orders WHERE orderId= '5000000005'; 

**Ibrahim**

select \* from Product WHERE productName = 'Bodywash';

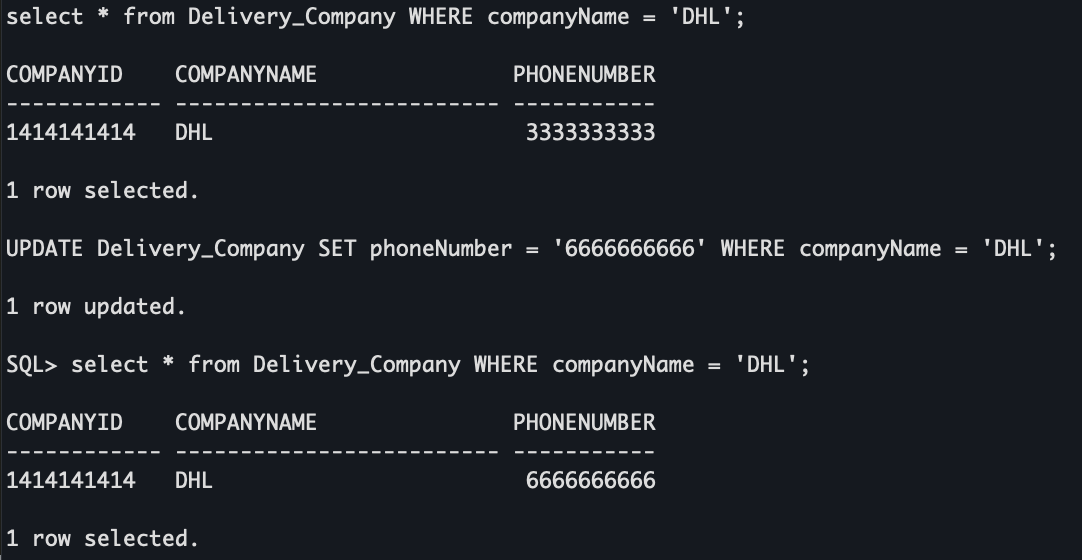
UPDATE Product SET price = 10.48 WHERE productName = 'Bodywash';

select \* from Product WHERE productName = 'Bodywash';

**Mithila**

select \* from Delivery\_Company WHERE companyName = 'DHL';

UPDATE Delivery\_Company SET phoneNumber = '6666666666' WHERE companyName = 'DHL';

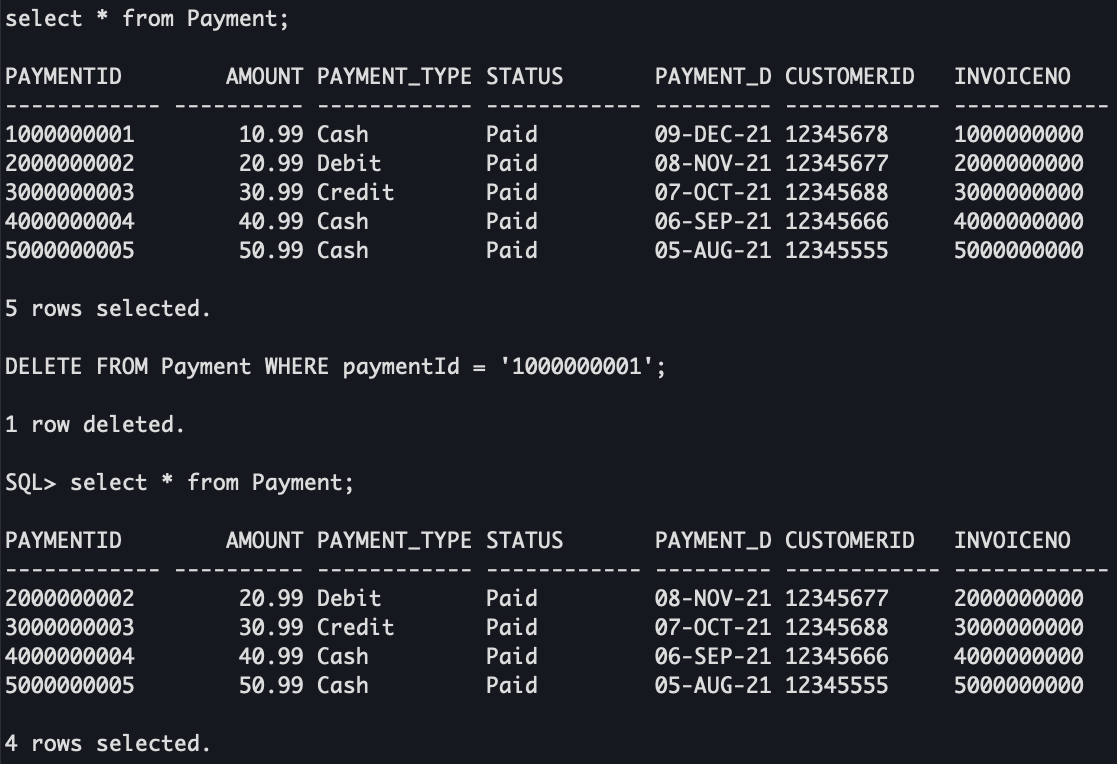
select \* from Delivery\_Company WHERE companyName = 'DHL';

Delete

**Nick**

select \* from Payment;

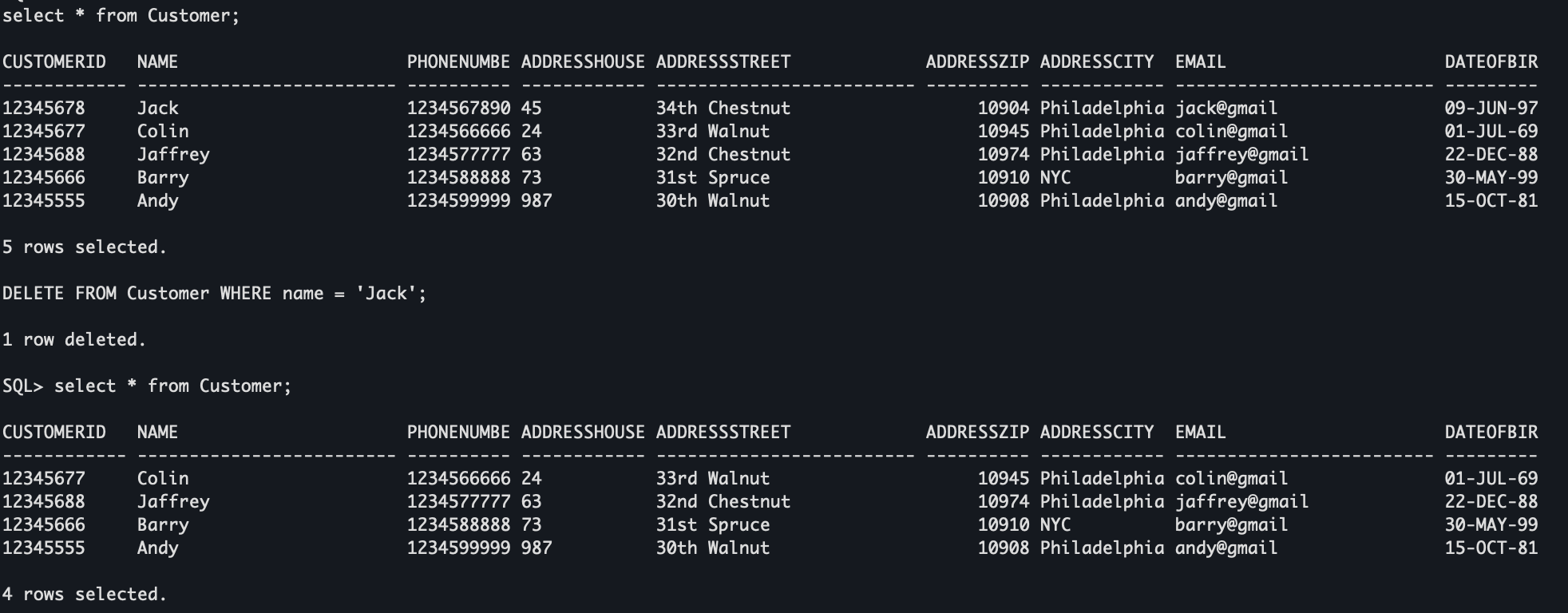
DELETE FROM Payment WHERE paymentId = '1000000001';

select \* from Payment;

**Mithila**

select \* from Customer;

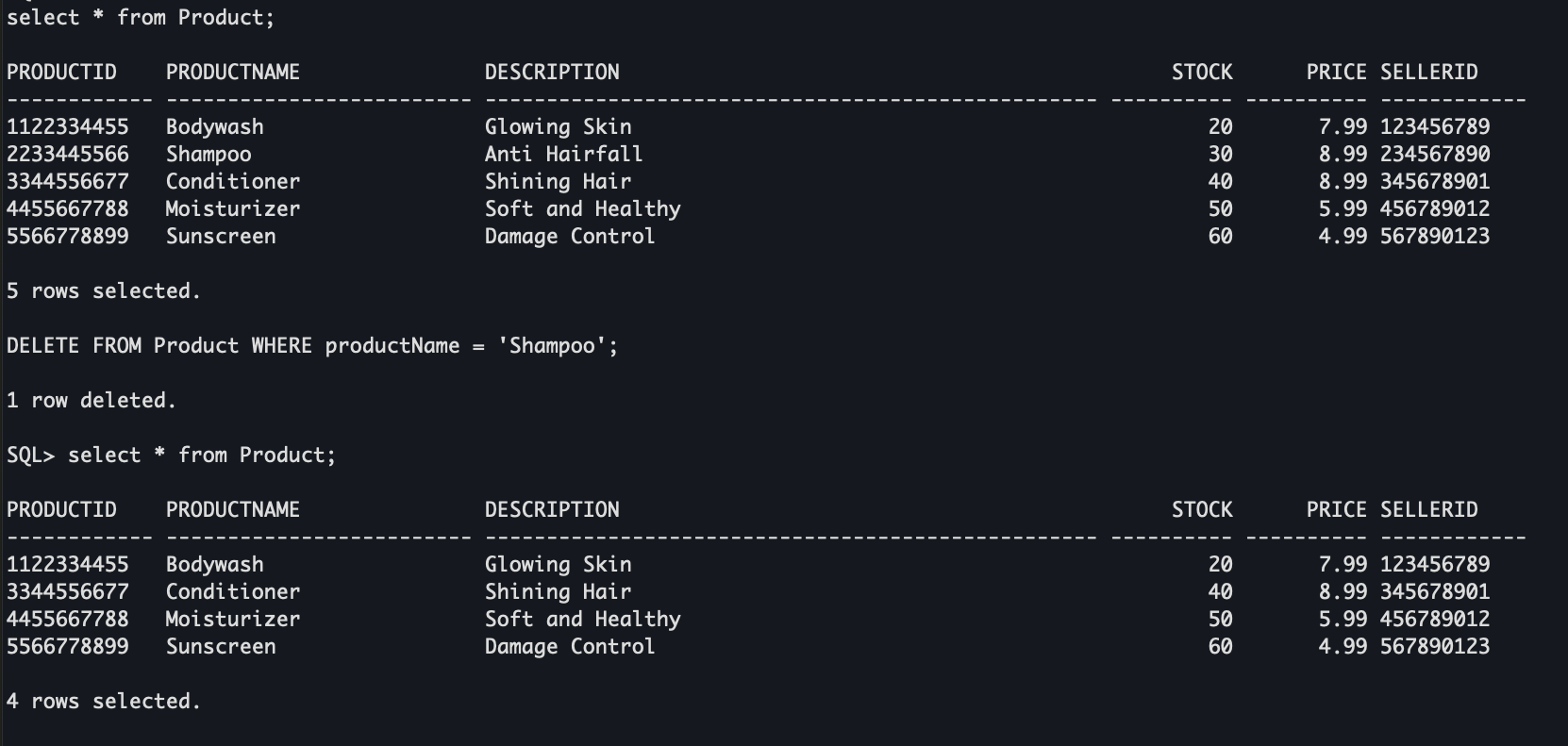
DELETE FROM Customer WHERE name = 'Jack';

select \* from Customer;

**Ibrahim**

select \* from Product;

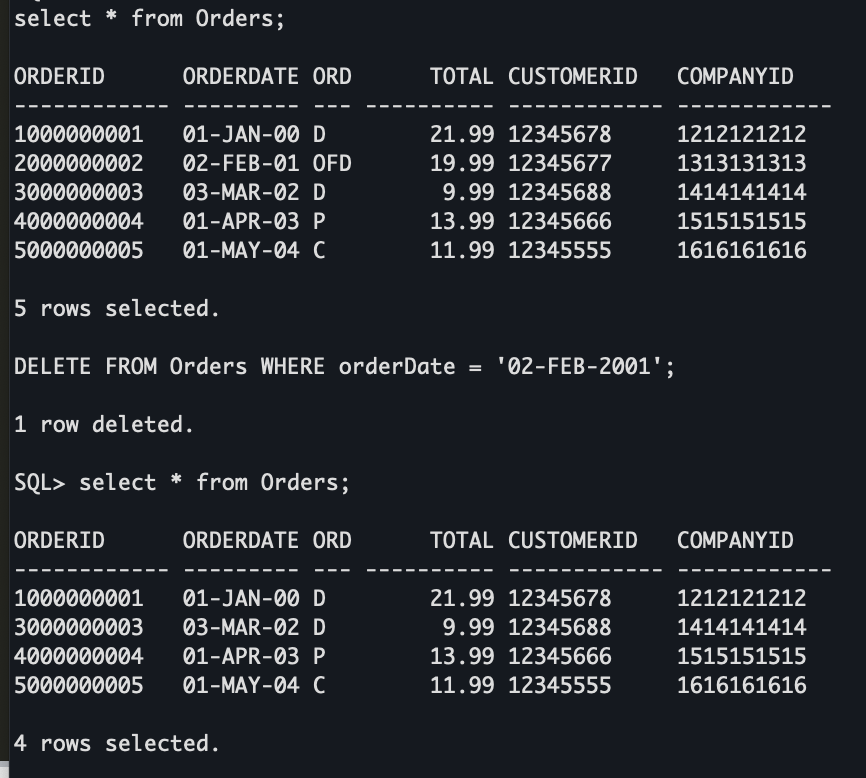
DELETE FROM Product WHERE productName = 'Shampoo';

select \* from Product;

**Nawaf**

select \* from Orders;

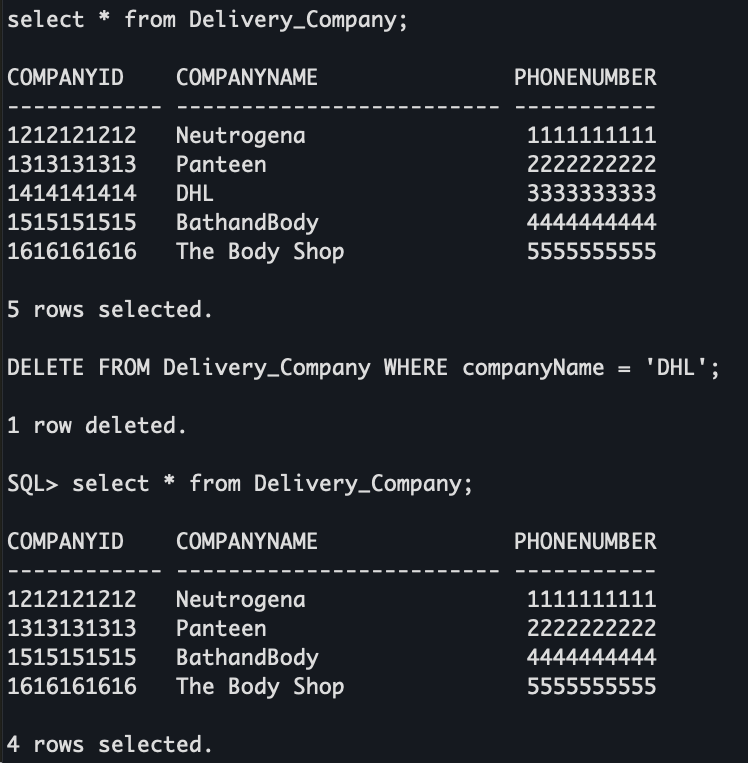
DELETE FROM Orders WHERE orderDate = '02-FEB-2001';

select \* from Orders;

**Samyak**

select \* from Delivery\_Company;

DELETE FROM Delivery\_Company WHERE companyName = 'DHL';

select \* from Delivery\_Company;

**Summary**

Our database has now been implemented to help manage the online e-commerce company Flipkart. As a seller signs up an account on this site, they are assigned a seller ID, which is the main way the website will keep track of the products they sell. Each product listed on this site will be attached to its seller since the seller ID will be included in the product’s table. Each product is now listed as part of a specific category as we created a table to keep track of that information. You can now easily display which products are part of what category. Each order is tracked by which customer placed that order, what company will deliver the order, when it was placed, and what the total amount was. You can also display which products are associated with each order. Every customer is registered in this database with a unique customer ID which is tied to their address, email, phone number, and date of birth. Every payment made on this website is associated with the customer who made the payment, what invoice it will be included on, how much the payment was for, and what type of payment was made (cash, credit, debit, etc.). A detailed invoice will be generated for each order which will show the total amount, the tax applied, if a discount was used, and the date it was issued. The delivery companies used by this website are also part of the database as we would like to keep track of what companies our sellers are using to deliver their product.

As mentioned before, most of this information will be automatically generated and updated as a person signs up to be a seller or a customer. Each seller will have to manually update a product and assign it to a category, but the product ID will be generated automatically. They will also have to manually enter which delivery company they want to use. Each customer will have to manually input their information, but a customer ID will be automatically generated. An order will be generated as a customer confirms their purchase, which is when an invoice will be generated.

This database will allow our client company to employ less people which can help generate more revenue. It can also eliminate the possibility of human error as most of the information in our database is automatically generated. Less errors within the management team means a more efficient way of managing the company. This database should be very easily manageable by the company, and it should also be very user friendly for the customer’s to interact with. A smooth customer interaction means a more customers will enjoy their experience on the website and will hopefully return to buy more products. Management will need to monitor the sellers’ listings and make sure that they are keeping up with entering the necessary data to allow a customer to have a positive experience. Other than this and monitoring any potential errors or bugs within the system, management should not have to extensively manage the database.

**Appendix**

Each group member put an equal amount of effort into creating this database and typing up this report. The work was divided up as follows:

1. Project Description – All
2. Requirements Specification – All
3. Conceptual Design – All
4. Relational Schema – Samyak, Nawaf, Ibrahim
5. Data Dictionary – Nick, Ibrahim
6. Database Implementaion – Samyak, Mithila
7. Data – All (Refer to ‘Data Inserts’ section for specifics)
8. Data Queries - All (Refer to ‘Data Queries’ section for specifics)
9. Data Manipulation - All (Refer to ‘Data Queries’ section for specifics)
10. Summary – Nick, Nawaf
11. Appendix – Mithila

**Database Screen-shots**

Text

Description automatically generated with medium confidence

*Screenshot of Seller Table after inserts*

Text

Description automatically generated

*Screenshot of Category Table after inserts*

Text

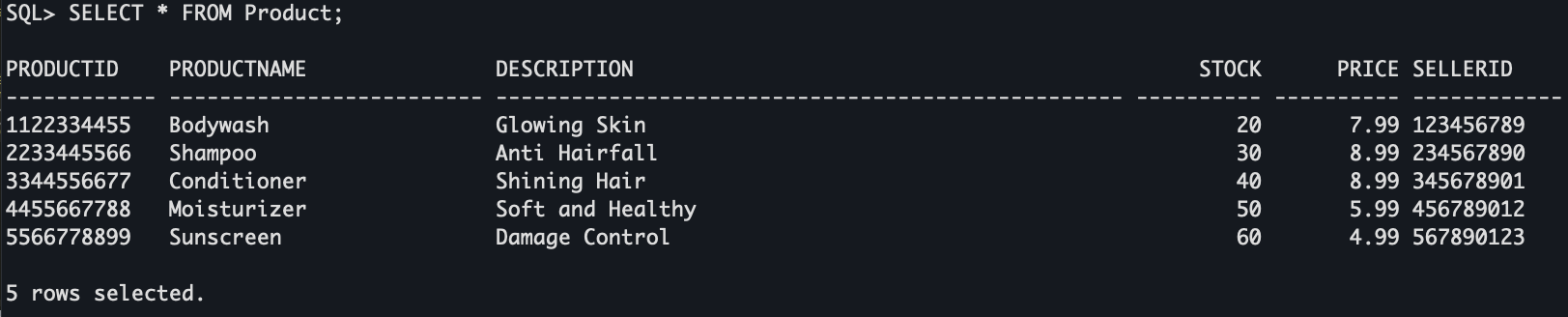
Description automatically generated with medium confidence

*Screenshot of Customer Table after inserts*

Text

Description automatically generated

*Screenshot of Delivery Company Table after inserts*



*Screenshot of Product Table after inserts*

Text

Description automatically generated with low confidence

*Screenshot of Product In Category Table after inserts*

Text

Description automatically generated

*Screenshot of Orders Table after inserts*

Text

Description automatically generated with low confidence

*Screenshot of Product in Orders Table after inserts*

**A picture containing calendar

Description automatically generated**

*Screenshot of Invoice Table after inserts*

**Table

Description automatically generated with medium confidence**

*Screenshot of Payment Table after inserts*