# E-Commerce Website Database Management

Group 10
Anshita Aishwarya
Vignesh Selvanayagam

(669) 279-9234 (Tel of Student 1)
(857) 437-2609 (Tel of Student 2)

aishwarya.a@northeastern.edu
selvanayagam.v@northeastern.edu

Percentage of Effort Contributed by Student1: 50%

Percentage of Effort Contributed by Student2: 50%

Signature of Student 1: 

Signature of Student 2: 

Submission Date: 12/10/2022

# Table of Contents

# E-commerce Website Database Management

In recent times, online retail platforms have gained popularity over traditional stores as the shopping behaviors are changing. It is the platform for buying and selling goods and services, or the transmitting of funds or data, over an electronic network (primarily the Internet). Some of the businesses offered by E-commerce include online shopping, online marketplace, B2B buying and selling, pretail services, online financial exchanges, and marketing. The main objective of our project is to develop an e-commerce platform model for a customer where they can register or buy a product i.e., it allows registered users to view and purchase the required products immediately. Also, from the supplier's standpoint, it allows administrators to view order transaction details. However, for the project implementation purposes, in the model, we will restrict the model to online shopping for one product like electronics, clothes, etc.

## Theory for E-Commerce:

The E-commerce model should support any customer to register or buy a product. So, it should have a registration window that allows the user to enter their personal information like name, bank name, and bank account number to get themselves registered with the platform. Once they have registered, the platform should generate unique user IDs and customer IDs for each customer. Also, the customers should be given the option to set up their own passwords.

In our project, the e-commerce platform should have a listing of all products i.e., the list of all products being sold through the platform. The list must be classified according to many criteria like prices, ratings, etc. The user should be able to buy any number of products. An invoice must be generated as soon as a purchase is made with the customer details in it. The most important feature is that the same product can be sold by multiple suppliers and the customer must be given the option to choose according to availability.

This project uses databases that can be used for tracking the transactions and also can be used by the marketing team to track the traffic, understand the demand, acquire potential customers and think about ways to retain existing customers.

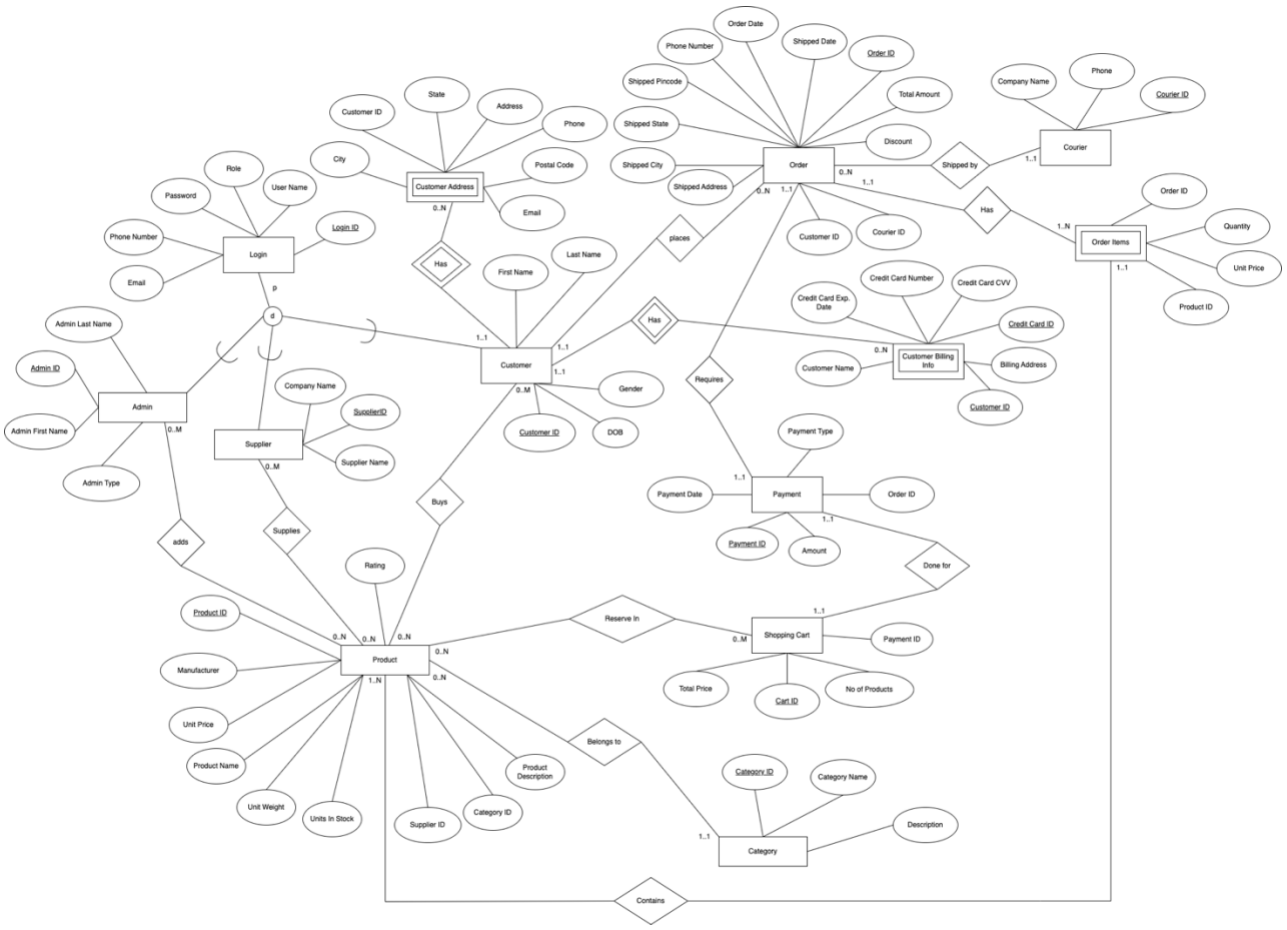
## Database Description:

1) Any user can register on the website and view the available products. Only registered users can purchase the products available. The **login** user will need to input information such as name, email address, date of birth, phone number, etc.
2) When a customer makes the first purchase, new information related to the **payment** such as billing address, shipping address, card and/or bank details etc., will be added to the database.
3) Each customer can place several **orders** at a single time. The customer can add multiple products (at least one) to a **shopping cart**.
4) Each order will generate an **order items** entry which will hold the information of all the products that have been bought in each order being placed.
5) Each product can be sold by several **suppliers**. One supplier can be linked to multiple products. The user can choose the supplier's name while making a purchase.

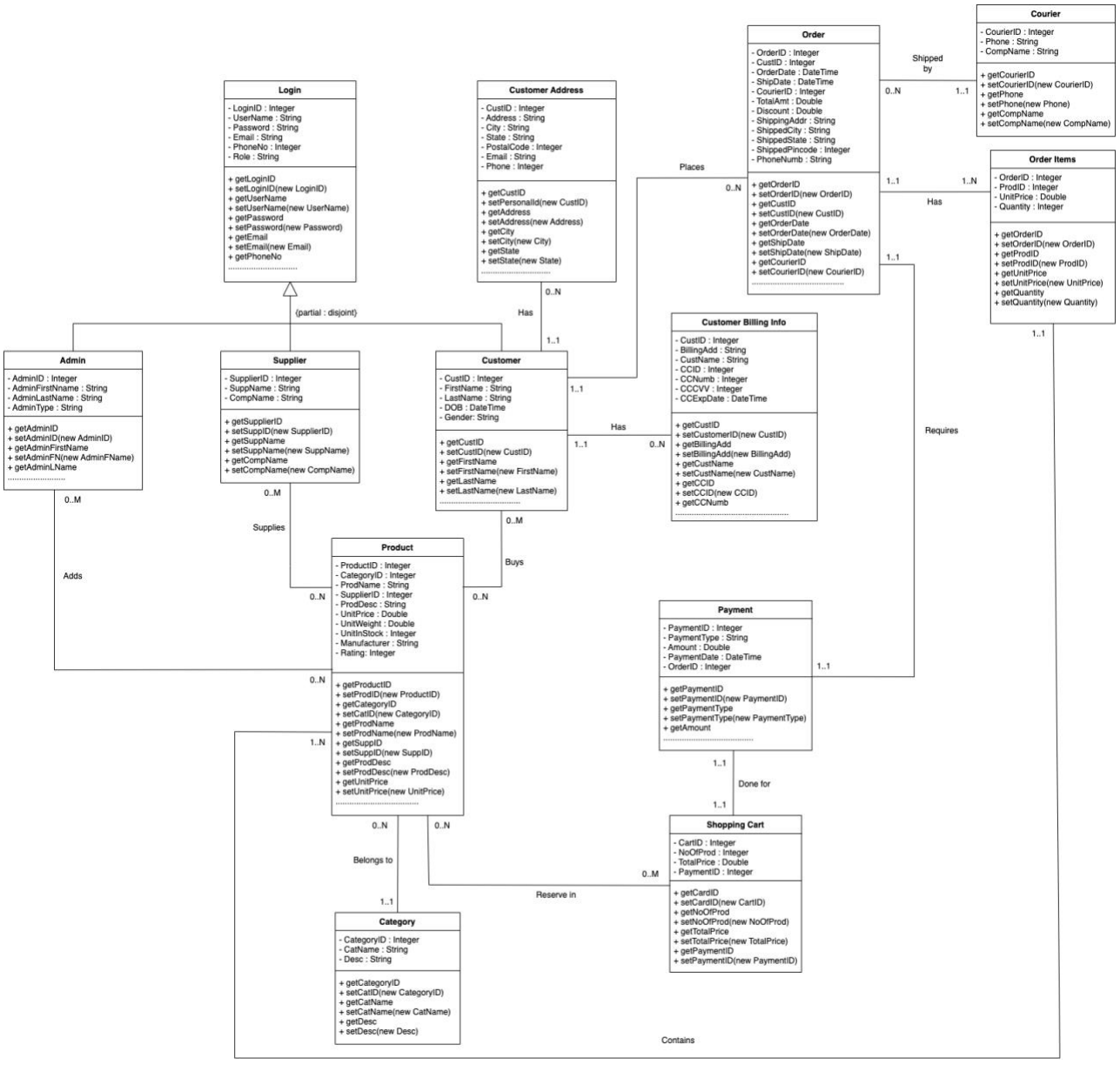
6) The **products** sold on the website will contain information such as item number, name, category, availability in stock or not, customer rating, etc.
7) Each product will belong to a certain **category** depending on the type of item being sold. One productcan belong to a single category.
8) Each order can be assigned to a **courier** service that takes care of shipping the products to the customer.
9) 3 types of roles (supplier, customer, and admins) with different privileges granted will be available on the website.
    a. **Suppliers** are the source of the range of products available.
    b. Once the visitor decides to purchase items, they need to register on the website. Such users then become **customers**.
    c. **Admins**, on the other hand, will have special privileges apart from the other roles. This will include adding new products, editing, or removing available products, marking products asout of stock, etc.

# Conceptual Modelling:

## *EER Diagram:*



***UML Diagram:***




# Logical Modelling:

Mapping the conceptual model to the relational model:

- **Login** (<u>LoginID</u>, Username, Password, PhoneNumber, Email, Role)
  - **Primary key**: LoginID

- **Admin** (*<u>AdminID</u>*, AdminFirstName, AdminLastName, AdminType)
  - **Relationship**: Partially derived from Login
  - **Foreign keys**: AdminID → LoginID from Login
  - **NOT NULL**: AdminID

- **Supplier** (*SupplierID*, SupplierName, CompanyName)
  - **Relationship**: Partially derived from Login
  - **Foreign keys**: SupplierID → LoginID from Login
  - **NOT NULL**: SupplierID

- **Customer** (*CustomerID*, FirstName, LastName, DOB, Gender)
  - **Relationship**: Partially derived from Login, 1-N relationship with CustomerAddress, 1-N relationship with CustomerBillingInfo, 1-N relationship with Order
  - **Foreign keys**: CustomerID → LoginID from Login
  - **NOT NULL**: CustomerID

- **CustomerBillingInfo** (CreditCardID, *CustomerID*, CustomerName, CreditCardNo,CreditCardCVV, CreditCardExp, BillingAddress)
  - **Relationship**: 1-N relationship with CustomerBillingInfo
  - **Primary key**: CreditCardID
  - **Foreign keys**: CustomerID from Customer
  - **NOT NULL**: CustomerID

- **adds** (*AdminID*, *ProductID*)
  - **Relationship**: Relationship between Admin and Product
  - **Foreign keys**: AdminID from Admin and ProductID from Product
  - **NOT NULL**: AdminID, ProductID

- **supplies** (*SupplierID*, *ProductID*)
  - **Relationship**: Relationship between Supplier and Product
  - **Foreign keys**: SupplierID from Supplier and ProductID from Product
  - **NOT NULL**: SupplierID, ProductID

- **buys** (*CustomerID*, *ProductID*)
  - **Relationship**: Relationship between Customer and Product
  - **Foreign keys**: CustomerID from Customer and ProductID from Product
  - **NOT NULL**: CustomerID, ProductID

- **CustomerAddress** (*CustomerID*, Address, City, State, PostalCode, Email, Phone)
  - **Relationship**: 1-N relationship with Customer
  - **Foreign keys**: CustomerID from Customer
  - **NOT NULL**: CustomerID

- **Orders** (OrderID, OrderDate, *CustomerID*, TotalAmount, Discount, *CourierID*, ShippedDate, ShippedAddress, ShippedCity, ShippedPincode, PhoneNumber)
  - **Relationship**: 1-N relationship with Customer, 1-N relationship with Courier, 1-N relationship with Order Details, 1-1 relationship with Payment
  - **Primary key**: OrderID
  - **Foreign keys**: CustomerID from Customer, CourierID from Courier
  - **NOT NULL**: CustomerID, CourierID

- **Courier** (CourierID, CompanyName, Phone)
  - **Relationship**: 1-N relationship with Orders
  - **Primary key**: CourierID
  - **Foreign keys**: -------

- **OrderItems** (*OrderID*, *ProductID*, Quantity, UnitPrice)
  - **Relationship**: 1-N relationship with Orders
  - **Foreign keys**: OrderID from Orders, ProductID from Product

- **Payment** (PaymentID, PaymentType, Amount, PaymentDate, *OrderID*)
  - **Relationship**: 1-1 relationship with Orders
  - **Primary key**: PaymentID
  - **Foreign keys**: OrderID from Orders
  - **NOT NULL**: OrderID

- **ShoppingCart** (CartID, NoOfProducts, TotalPrice, *PaymentID*)
  - **Relationship**: 1-1 relationship with Payment and M-N relationship with Product
  - **Primary key**: CartID
  - **Foreign keys**: PaymentID from Payment
  - **NOT NULL**: PaymentID

- **reserved** (*CartID*, *ProductID*)
  - **Relationship**: Relationship between ShoppingCart and Product
  - **Foreign keys**: CartID from ShoppingCart, ProductID from Product
  - **NOT NULL**: CartID, ProductID

- **Product** (ProductID, ProductName, Description, UnitPrice, UnitWeight, Manufacturer,UnitsInStock, Rating, *CategoryID*)
  - **Relationship**: N-M relationship with ShoppingCart, 1-N relationship with Category, N-M relationships with Admin, Supplier, Customer and OrderDetails
  - **Primary key**: ProductID
  - **Foreign keys**: CategoryID from Category
  - **NOT NULL**: CategoryID

- **Category** (CategoryID, CategoryName, Description)
  - **Relationship**: 1-N relationship with Product
  - **Primary key**: Category_ID
  - **Foreign keys**: ------

## Implementation in MySQL:

The database model has been translated to the database schema in MySQL. Below are few screenshots of the schema used to perform this implementation:

```
CREATE SCHEMA 'ecommerce_DB'
DEFAULT CHARACTER SET utf8mb4
COLLATE utf8mb4_unicode_ci;
```

```sql
CREATE TABLE `Login` (
  `LoginID` int NOT NULL DEFAULT '111',
  `Username` varchar(45) NOT NULL,
  `Password` varchar(45) NOT NULL,
  `PhoneNumber` varchar(15) NOT NULL,
  `Email` varchar(45) NOT NULL,
  `Role` varchar(45) NOT NULL,
  PRIMARY KEY (`LoginID`),
  UNIQUE KEY `Password_UNIQUE` (`Password`),
  UNIQUE KEY `PhoneNumber_UNIQUE` (`PhoneNumber`),
  UNIQUE KEY `Email_UNIQUE` (`Email`),
  UNIQUE KEY `LoginID_UNIQUE` (`LoginID`)
);
```

```sql
CREATE TABLE `Product` (
  `ProductID` int NOT NULL,
  `ProductName` varchar(100) NOT NULL,
  `Description` varchar(500) NOT NULL,
  `UnitPrice` varchar(45) NOT NULL,
  `UnitWeight` varchar(45) NOT NULL,
  `Manufacturer` varchar(100) NOT NULL,
  `UnitsInStock` varchar(45) NOT NULL,
  `Rating` decimal(5,0) DEFAULT NULL,
  `CategoryID` int NOT NULL,
  PRIMARY KEY (`ProductID`),
  UNIQUE KEY `ProductID_UNIQUE` (`ProductID`)
);
```

## ***Data Generation:***

After creating the tables, we used the 'https://generatedata.com/generator' to generate the Data for our database. We specified data types and attribute type names for each attribute type of every table in the model to create data for corresponding table. Below are few screenshots of the sample data generated.

| CustomerID | FirstName | LastName | DOB | Gender |
|---|---|---|---|---|
| 7 | Sade S | Johnson | 2/11/1968 | Other |
| 9 | Sage F | Hopkins | 5/27/1990 | Male |
| 10 | Brielle O | Pugh | 4/15/1972 | Other |
| 17 | Orla J | Wilkins | 5/13/2017 | Other |
| 18 | Aidan S | Nolan | 5/1/1996 | Male |
| 23 | Hasad Y | Campbell | 11/13/1961 | Other |
| 26 | Griffin S | Craft | 3/29/1971 | Male |
| 30 | Francis C | Schultz | 12/13/1966 | Male |
| 31 | Justina B | Saunders | 5/20/2007 | Female |
| 32 | Quentin I | Hood | 9/2/1969 | Female |

| OrderID | OrderDate | CustomerID | TotalAmount | Discount | CourierID | ShippedDate | ShippedAddress | ShippedCity | ShippedState | Pincode |
|---|---|---|---|---|---|---|---|---|---|---|
| 1000 | 3/7/22 | 141 | 82,414.69 | 460.64 | 659 | 3/24/22 | 500 Pine Street | Seattle | WA | 98101 |
| 1001 | 12/31/20 | 126 | 41,442.17 | 746.48 | 670 | 8/18/22 | 401 N.E. Northgate Way | Seattle | WA | 98125 |
| 1002 | 5/9/22 | 207 | 43,150.66 | 199.34 | 660 | 3/27/22 | 17200 Southcenter Parkway, Suite 110 | Tukwila | WA | 98188 |
| 1003 | 11/4/21 | 289 | 74,999.57 | 118.22 | 664 | 3/12/22 | 100 Bellevue Square | Bellevue | WA | 98004 |
| 1004 | 4/8/22 | 22 | 69,934.29 | 581.22 | 664 | 12/27/21 | 100 Southcenter Shopping Center | Tukwila | WA | 98188 |
| 1005 | 6/27/21 | 83 | 62,546.08 | 84.2 | 658 | 5/11/21 | 4502 S. Steele Street, Suite 800 | Tacoma | WA | 98409 |
| 1006 | 1/19/22 | 288 | 84,032.42 | 162.24 | 666 | 5/13/22 | 828 W. Main Avenue | Spokane | WA | 99201 |
| 1007 | 10/28/21 | 5 | 63,436.60 | 872.4 | 662 | 2/2/22 | 3200 184th Street SW | Lynnwood | WA | 98037 |
| 1008 | 11/30/20 | 110 | 6,695.10 | 903.65 | 659 | 3/6/22 | 680 E. Northern Lights Boulevard | Anchorage | AK | 99503 |
| 1009 | 7/5/22 | 182 | 33,976.28 | 732.06 | 659 | 7/1/21 | 603 D Street | Anchorage | AK | 99501 |

| ProductID | ProductName | Description | UnitPrice | UnitWeight (o | Manufacturer | UnitsinStock | Rating | CategoryID |
|---|---|---|---|---|---|---|---|---|
| 1500 | Watch Box Display Case Organizer Glass | Double Layer Design with 8 watch slots; glass is made of high-quality real glas | 32.99 | 4.17 | TomCare | 23 | 4.7 | 406 |
| 1501 | SanDisk 128GB Ultra Flair USB 3.0 Flash Drive | High-speed USB 3.0 performance of up to 150MB/s; 1.2GB MPEG-4 video trai | 14.29 | 0.1 | SanDisk | 79 | 3.5 | 409 |
| 1502 | Yamaha YHT-5960U Home Theater System | 8K60B HDMI 2.1 with HDCP 2.3 and eARC (4 in/1 out); Dolby Vision, Hybrid | 779.95 | 54 | Yamaha | 1 | 4.1 | 415 |
| 1503 | Samsung BAR Plus 64GB - 300MB/s USB 3.1 Flash Drive | Reliable and secure storage for your photos, videos, music, and files; Rugged n | 22.99 | 0.12 | Samsung | 25 | 4.7 | 409 |
| 1504 | Powermax 100-Count AAA Batteries | Ultra Long Lasting Alkaline Battery, 10-Year Shelf Life, Recloseable Packaging | 21.99 | 2.6 | Powermax | 14 | 2.9 | 413 |
| 1505 | Echo Dot Smart Speaker with Alexa | Echo Dot with Alexa provides clear vocals, deep bass and vibrant sound. Play n | 24.99 | 12 | Amazon | 78 | 4.4 | 403 |
| 1506 | VAJUN Portable Ergonomic Laptop Desk Stand | Creative portable design ,lightweight; compatible with: MacBook Air/ Pro, Goo | 39.99 | 1.32 | VAJUN | 33 | 3.9 | 408 |
| 1507 | uVlogger - Android Compatible Vlogging Kit Accessories | Complete smartphone video kit with Shotgun Microphone, Ultra bright LED30 | 79.96 | 1.79 | Movo | 82 | 3.9 | 402 |
| 1508 | LG LHD657 Bluetooth 5.1-Channel Home Theater System | Region Free Home Theater System with PAL/NTSC Support; Audio L/R Yes, C | 399.99 | 50 | LG | 53 | 4.2 | 415 |

| CustomerID | BillingAddress | City | State | PostalCode | Email | phone |
|---|---|---|---|---|---|---|
| 7 | 897 Long Airport Avenue | NYC | NY | 10022 | tincidunt.donec@aol.org | (175) 378-1647 |
| 9 | 361 Furth Circle | San Diego | CA | 91217 | commodo.ipsum@hotmail.couk | (361) 301-3456 |
| 10 | 39323 Spinnaker Dr. | Cambridge | MA | 51247 | ipsum.leo.elementum@google.net | (181) 545-1726 |
| 17 | 3758 North Pendale Street | White Plains | NY | 24067 | nisl.maecenas@google.couk | (734) 216-1628 |
| 18 | 4575 Hillside Dr. | New Bedford | MA | 50553 | purus.mauris@yahoo.edu | (886) 654-1362 |
| 23 | 7734 Strong St. | San Francisco | CA | 94217 | in.faucibus@yahoo.net | (564) 681-1052 |
| 26 | 7476 Moss Rd. | Newark | NJ | 94019 | ut.molestie@protonmail.edu | (786) 829-3623 |
| 30 | 782 First Street | Philadelphia | PA | 71270 | vitae.aliquam.eros@aol.com | (485) 372-6527 |
| 31 | 6047 Douglas Av. | Los Angeles | CA | 91003 | egestas.duis@yahoo.ca | (472) 599-8531 |

| CategoryID | CategoryName | Description |
|---|---|---|
| 400 | Headphone | A pair of padded speakers which you wear over your ears in order to listen to a radio or recorded music. |
| 401 | Phone | Portable device for connecting to a telecommunications network in order to transmit and receive voice, video, or other data. |
| 402 | Phone Accessories | Any hardware that is not integral to the operation of a mobile smartphone as designed by the manufacturer |
| 403 | Speaker | Device for converting electrical energy into acoustical signal energy that is radiated into a room or open air. |
| 404 | Charging Cable | A cable used to charge a smartphone or other portable device. |
| 405 | Watch | A portable timepiece intended to be carried or worn by a person. |

***Database Population:***

We have used insert queries to load the data in the database. Below are few screenshots of how the populated database looks like:

*Login*

| LoginID | Username | Password | PhoneNumber | Email | Role |
|---|---|---|---|---|---|
| 1 | Bernard U. Hoover | LDE55RPE4SI | (275) 232-1527 | tempus.non@protonmail.edu | Admin |
| 2 | Luke X. Hodges | QWG53XUS3ZO | (238) 528-2316 | nonummy.ac.feugiat@hotmail.org | Supplier |
| 3 | Faith J. Keller | LQE51VUD2FJ | (555) 863-8368 | et.lacinia@google.edu | Supplier |
| 4 | Darrel G. Silva | LDI19WHY2JJ | (971) 756-6623 | lorem.donec.elementum@hotmail.couk | Supplier |
| 5 | Alden X. Short | GVR74VNR6XQ | (667) 814-7774 | molestie@outlook.ca | Supplier |
| 6 | Brenden J. Swanson | JUQ84DVM7CI | (468) 986-1967 | vulputate.posuere.vulputate@outlook.com | Admin |
| 7 | Sade S. Johnson | HXE55FOC3ZO | (711) 917-0963 | malesuada.fames@hotmail.couk | Customer |
| 8 | Kylee H. Perkins | JBV89CHK6XH | (560) 945-4529 | feugiat.metus@icloud.couk | Supplier |
| 9 | Sage F. Hopkins | GEI57TXX4VJ | (725) 841-2246 | massa.quisque@protonmail.couk | Customer |
| 10 | Brielle O. Pugh | BMK68PXO6NC | (761) 287-9756 | ipsum.donec@google.ca | Customer |
| 11 | Joseph X. Phelps | SKW62WHY5XU | (384) 677-4352 | sapien.cursus@icloud.ca | Supplier |
| 12 | Ivan G. Lindsey | VBE02OFW5CO | (656) 354-4442 | leo.vivamus@icloud.edu | Supplier |
| 13 | Theodore H. Hester | HHP78QIS7GP | (611) 957-2677 | ac.orci@hotmail.ca | Supplier |
| 14 | Leroy H. Witt | ACF28LIN6KJ | (276) 222-9795 | nunc@protonmail.org | Admin |
| 15 | Cooper Y. Washing… | XGR80HYH3TD | (527) 829-7294 | sem@aol.net | Supplier |
| 16 | Heidi G. Fields | DYR57RKJ3PZ | (572) 829-1396 | nam@google.net | Supplier |
| 17 | Orla J. Wilkins | THA60GTP3FF | (444) 605-4778 | facilisis@yahoo.edu | Customer |
| 18 | Aidan S. Nolan | JBC75ZVE6RJ | (725) 962-1862 | sollicitudin@google.org | Customer |
| 19 | Zoe K. Richards | XOK55NNJ8LK | (492) 733-2622 | dignissim.pharetra@icloud.edu | Supplier |
| 20 | Jenna I. Sellers | QEJ83HLA2LQ | (215) 967-6724 | at.fringilla.purus@aol.net | Supplier |

*Customer*

| CustomerID | Address | City | State | PostalCode | Email | Phone |
|---|---|---|---|---|---|---|
| 7 | 897 Long Airport Avenue | NYC | NY | 10022 | tincidunt.donec@aol.org | (175) 378-1647 |
| 9 | 361 Furth Circle | San Diego | CA | 91217 | commodo.ipsum@hotmail.couk | (361) 301-3456 |
| 10 | 39323 Spinnaker Dr. | Cambridge | MA | 51247 | ipsum.leo.elementum@google.net | (181) 545-1726 |
| 17 | 3758 North Pendale Street | White Plains | NY | 24067 | nisl.maecenas@google.couk | (734) 216-1628 |
| 18 | 4575 Hillside Dr. | New Bedford | MA | 50553 | purus.mauris@yahoo.edu | (886) 654-1362 |
| 23 | 7734 Strong St. | San Francisco | CA | 94217 | in.faucibus@yahoo.net | (564) 681-1052 |
| 26 | 7476 Moss Rd. | Newark | NJ | 94019 | ut.molestie@protonmail.edu | (786) 829-3623 |
| 30 | 782 First Street | Philadelphia | PA | 71270 | vitae.aliquam.eros@aol.com | (485) 372-6527 |
| 31 | 6047 Douglas Av. | Los Angeles | CA | 91003 | egestas.duis@yahoo.ca | (472) 599-8531 |
| 32 | 8616 Spinnaker Dr. | Boston | MA | 51003 | euismod.enim@google.edu | (767) 822-4546 |
| 34 | 2304 Long Airport Avenue | Nashua | NH | 62005 | molestie.dapibus@google.edu | (637) 458-1316 |

*Orders*

| OrderID | OrderDate | CustomerID | TotalAmount | Discount | CourierID | ShippedDate | ShippedAddress | ShippedCity | ShippedState | Shippec |
|---|---|---|---|---|---|---|---|---|---|---|
| 1000 | 2022-03-07 00:00:00 | 7 | 82415 | 461 | 659 | 2022-03-24 00:00:00 | 500 Pine Street | Seattle | WA | 98101 |
| 1001 | 2020-12-31 00:00:00 | 9 | 41442 | 746 | 670 | 2022-08-18 00:00:00 | 401 N.E. Northgate Way | Seattle | WA | 98125 |
| 1002 | 2022-05-09 00:00:00 | 10 | 43151 | 199 | 660 | 2022-03-27 00:00:00 | 17200 Southcenter Parkway, Suite 110 | Tukwila | WA | 98188 |
| 1003 | 2021-11-04 00:00:00 | 17 | 75000 | 118 | 664 | 2022-03-12 00:00:00 | 100 Bellevue Square | Bellevue | WA | 98004 |
| 1004 | 2022-04-08 00:00:00 | 18 | 69934 | 581 | 664 | 2021-12-27 00:00:00 | 100 Southcenter Shopping Center | Tukwila | WA | 98188 |
| 1005 | 2021-06-27 00:00:00 | 23 | 62546 | 84 | 658 | 2021-05-11 00:00:00 | 4502 S. Steele Street, Suite 800 | Tacoma | WA | 98409 |
| 1006 | 2022-01-19 00:00:00 | 26 | 84032 | 162 | 666 | 2022-05-13 00:00:00 | 828 W. Main Avenue | Spokane | WA | 99201 |
| 1007 | 2021-10-28 00:00:00 | 30 | 63437 | 872 | 662 | 2022-02-02 00:00:00 | 3200 184th Street SW | Lynnwood | WA | 98037 |
| 1008 | 2020-11-30 00:00:00 | 31 | 6695 | 904 | 659 | 2022-03-06 00:00:00 | 680 E. Northern Lights Boulevard | Anchorage | AK | 99503 |
| 1009 | 2022-07-05 00:00:00 | 32 | 33976 | 732 | 659 | 2021-07-01 00:00:00 | 603 D Street | Anchorage | AK | 99501 |
| 1010 | 2021-06-07 00:00:00 | 34 | 28768 | 764 | 652 | 2021-10-12 00:00:00 | 3920 124th Street SE | Bellevue | WA | 98006 |
| 1011 | 2020-11-21 00:00:00 | 36 | 10051 | 790 | 669 | 2021-10-22 00:00:00 | 19500 Alderwood Mall Parkway | Lynnwood | WA | 98036 |
| 1012 | 2021-08-03 00:00:00 | 37 | 33779 | 288 | 661 | 2020-11-13 00:00:00 | 400 Pine Street, Suite 36 | Seattle | WA | 98101 |
| 1013 | 2020-09-07 00:00:00 | 38 | 60271 | 813 | 669 | 2022-08-27 00:00:00 | 1101 Outlet Collection DR SW | Auburn | WA | 98001 |
| 1014 | 2022-03-13 00:00:00 | 42 | 46698 | 823 | 670 | 2021-03-31 00:00:00 | 701 S.W. Broadway | Portland | OR | 97205 |
| 1015 | 2021-04-05 00:00:00 | 49 | 72346 | 426 | 662 | 2021-09-12 00:00:00 | 8930 S.E. Sunnyside Road | Clackamas | OR | 97015 |
| 1016 | 2022-01-19 00:00:00 | 50 | 93498 | 200 | 673 | 2021-02-14 00:00:00 | 11900 S.E. 82nd Avenue | Happy Valley | OR | 97086 |

*OrderItems*

| OrderID | ProductID | UnitPrice | Quantity |
|---|---|---|---|
| 1000 | 1549 | 2300 | 32 |
| 1001 | 1503 | 23 | 56 |
| 1002 | 1559 | 930 | 20 |
| 1002 | 1545 | 2697 | 10 |
| 1003 | 1523 | 250 | 39 |
| 1003 | 1544 | 48 | 39 |
| 1004 | 1567 | 152 | 6 |
| 1005 | 1509 | 32 | 51 |
| 1005 | 1553 | 449 | 27 |

*Payment*

| PaymentID | PaymentType | PaymentDate | Amount | OrderID |
|---|---|---|---|---|
| 700 | UPI | 2021-05-21 | 4706 | 1000 |
| 701 | Bank Transfer | 2022-09-24 | 4557 | 1001 |
| 702 | Cash on Delivery | 2022-11-16 | 4170 | 1002 |
| 703 | Credit Card | 2021-10-07 | 1142 | 1003 |
| 704 | Bank Transfer | 2021-09-11 | 3676 | 1004 |
| 705 | UPI | 2020-11-12 | 7184 | 1005 |
| 706 | Cash on Delivery | 2021-03-08 | 4635 | 1006 |
| 707 | UPI | 2021-03-15 | 6848 | 1007 |
| 708 | UPI | 2022-01-31 | 8514 | 1008 |
| 709 | UPI | 2022-07-24 | 858 | 1009 |
| 710 | UPI | 2022-10-16 | 4825 | 1010 |
| 711 | UPI | 2022-03-29 | 9079 | 1011 |
| 712 | Cash on Delivery | 2022-01-06 | 9081 | 1012 |

*Product*

| ProductID | ProductName | Description | UnitPrice | UnitWeight | Manufactur... | UnitsInStock | Rating | CategoryID |
|---|---|---|---|---|---|---|---|---|
| 1500 | Watch Box Display Case Organizer Glass | Double Layer Design with 8 watch slots; glass is... | 32.99 | 4.17 | TomCare | 23 | 5 | 406 |
| 1501 | SanDisk 128GB Ultra Flair USB 3.0 Flash Drive | High-speed USB 3.0 performance of up to 150... | 14.29 | 0.1 | SanDisk | 79 | 4 | 409 |
| 1502 | Yamaha YHT-5960U Home Theater System | 5.1-channel 80-Watt powerful surround sound s... | 779.95 | 54 | Yamaha | 1 | 4 | 415 |
| 1503 | Samsung BAR Plus 64GB - 300MB/s USB 3.1 F... | Reliable and secure storage for your photos, vid... | 22.99 | 0.12 | Samsung | 25 | 5 | 409 |
| 1504 | Powermax 100-Count AAA Batteries | Ultra Long Lasting Alkaline Battery, 10-Year She... | 21.99 | 2.6 | Powermax | 14 | 3 | 413 |
| 1505 | Echo Dot Smart Speaker with Alexa | Echo Dot with Alexa provides clear vocals, deep... | 24.99 | 12 | Amazon | 78 | 4 | 403 |
| 1506 | VAJUN Portable Ergonomic Laptop Desk Stand | Creative portable design ,lightweight; compatibl... | 39.99 | 1.32 | VAJUN | 33 | 4 | 408 |
| 1507 | uVlogger - Android Compatible Vlogging Kit Acc... | Complete smartphone video kit with Shotgun Mi... | 79.96 | 1.79 | Movo | 82 | 4 | 402 |
| 1508 | LG LHD657 Bluetooth 5.1-Channel Home Theat... | Region Free Home Theater System with PAL/N... | 399.99 | 50 | LG | 53 | 4 | 415 |
| 1509 | AprilStar 1TB USB Flash Drive | Compatible with most systems such as Win XP/... | 32.48 | 0.15 | AprilStar | 63 | 4 | 409 |
| 1510 | Amazon Fire TV 43-inch | Brilliant 4K vivid Ultra HD; Fire TV Alexa Voice... | 369.99 | 15.5 | Amazon | 3 | 5 | 414 |
| 1511 | Samsung 65-Inch Class Crystal Smart TV | 4K UHD AU8000 Series HDR, 3 HDMI Ports, M... | 647.99 | 46.1 | Samsung | 21 | 5 | 414 |
| 1512 | iPhone Charger 3 Pack 10 ft. | iPhone charger with fast charging; cable nylon b... | 22.99 | 6.4 | Apple | 36 | 5 | 404 |
| 1513 | Xenvo Prod Lens Kit for iPhone and Android | Macro and Wide Angle Lens with LED Light and... | 39.99 | 3.65 | Xenvo | 12 | 4 | 402 |
| 1514 | Amazon Basics 48 Pack AA High-Performance... | 48-pack of 1.5 volt AA alkaline batteries; 10-yea... | 15.67 | 2.54 | Amazon Ba... | 85 | 3 | 412 |
| 1515 | Tzowla Laptop Backpack for Men Women | Travel Work Security, Anti Theft, Water Resistan... | 36.89 | 0.78 | Tzowla | 44 | 4 | 408 |

***MySQL Queries:***

**# No of orders and average sales per week**

```
SELECT YEAR(OrderDate) AS Year, WEEK(OrderDate) AS Week,
COUNT(OrderID) AS Num_of_orders,
AVG(TotalAmount) AS Average_Sales
FROM Orders
GROUP BY Year, Week
ORDER BY Num_of_orders DESC;
```

| Year | Week | Num_of_orders | Average_Sales |
|---|---|---|---|
| 2022 | 3 | 5 | 70842.8000 |
| 2021 | 31 | 5 | 40660.0000 |
| 2020 | 48 | 4 | 38386.2500 |
| 2021 | 23 | 4 | 46025.0000 |
| 2021 | 6 | 4 | 45747.7500 |
| 2021 | 36 | 4 | 52141.5000 |
| 2021 | 11 | 4 | 69175.5000 |
| 2021 | 8 | 4 | 51323.0000 |
| 2022 | 37 | 4 | 71635.7500 |

**# Most popular Category**

```
SELECT ct.CategoryName AS Category, COUNT(oi.OrderID) AS Total_Orders
FROM Category ct
INNER JOIN Product p
ON ct.CategoryID = p.CategoryID
INNER JOIN OrderItems oi
ON p.ProductID = oi.ProductID
GROUP BY p.CategoryID
ORDER BY Total_Orders DESC;
```

| Category | Total_Orders |
|---|---|
| Speaker | 27 |
| TV | 27 |
| Home Theater | 25 |
| Phone Accessories | 20 |
| Monitor | 20 |
| Phone | 18 |
| Pen Drive | 16 |
| Headphone | 16 |

**# Different Payment Methods**

```
SELECT
   COUNT(CASE WHEN PaymentType IN ('UPI', 'Bank Transfer', 'Credit Card') THEN 1 ELSE NULL
END) AS Online_Payment,
   COUNT(CASE WHEN PaymentType = 'Cash on Delivery' THEN 1 ELSE NULL END) AS
Cash_Payment
FROM Payment;
```

| Online_Payment | Cash_Payment |
|---|---|
| 157 | 67 |

**# Customers who placed orders in 2022 and order amount less than or equal to 10000**

```
SELECT c.CustomerID, c.FirstName, c.LastName
FROM Customer c
INNER JOIN Orders o
ON c.CustomerID = o.CustomerID
WHERE YEAR(o.OrderDate) = '2022'
AND o.CustomerID NOT IN
   (SELECT CustomerID
   FROM Orders
   WHERE YEAR(OrderDate) = '2021')
   GROUP BY CustomerID
   HAVING SUM(TotalAmount) > 10000;
```

| Cus... | FirstName | LastName |
|---|---|---|
| 18 | Aidan S | Nolan |
| 26 | Griffin S | Craft |
| 36 | Cherokee N | James |
| 50 | Nyssa U | Erickson |
| 71 | Montana F | Mendez |
| 74 | Howard Y | Mayo |

**# Products with at least 5 orders**

```
use ecommerce_DB;
SELECT p.ProductID, p.ProductName
FROM Product p
WHERE (SELECT COUNT(*)
   FROM OrderItems oi
   WHERE oi.ProductID = p.ProductID
   GROUP BY oi.ProductID) >= 5;
```

| ProductID | ProductName |
|---|---|
| 1500 | Watch Box Display Case Organizer Glass |
| 1501 | SanDisk 128GB Ultra Flair USB 3.0 Flash Drive |
| 1502 | Yamaha YHT-5960U Home Theater System |
| 1503 | Samsung BAR Plus 64GB - 300MB/s USB 3.1 F… |
| 1507 | uVlogger - Android Compatible Vlogging Kit Acc… |
| 1508 | LG LHD657 Bluetooth 5.1-Channel Home Theat… |

**# Customers with Order who received discount greater than unit price of any item**

```
use ecommerce_DB;
SELECT c.CustomerID, c.FirstName, c.LastName
FROM Customer c
where c.CustomerID IN (SELECT o.CustomerID
         FROM Orders o, OrderItems oi
```

```
WHERE o.OrderID = oi.OrderID
AND o.Discount >= ANY (SELECT UnitPrice
        FROM OrderItems));
```

| Cus... | FirstName | LastName |
|---|---|---|
| 7 | Sade S | Johnson |
| 9 | Sage F | Hopkins |
| 10 | Brielle O | Pugh |
| 17 | Orla J | Wilkins |
| 18 | Aidan S | Nolan |
| 23 | Hasad Y | Campbell |

**# Customers living in MA with Creditcard expiry more than 2 years**

```
use ecommerce_DB;
SELECT c.CustomerID, c.FirstName, c.LastName
FROM Customer c, CustomerAddress ca
WHERE c.CustomerID = ca.CustomerID
AND ca.State = 'MA'
UNION
SELECT c.CustomerID, c.FirstName, c.LastName
FROM Customer c, CustomerBillingInfo cbi
WHERE c.CustomerID = cbi.CustomerID
AND YEAR(cbi.CreditCardExp) - YEAR(CURDATE()) >= 2;
```

| CustomerID | FirstName | LastName |
|---|---|---|
| 10 | Brielle O | Pugh |
| 18 | Aidan S | Nolan |
| 32 | Quentin I | Hood |
| 36 | Cherokee N | James |
| 50 | Nyssa U | Erickson |
| 51 | Doris L | Benjamin |
| 62 | Rebecca T | Jennings |

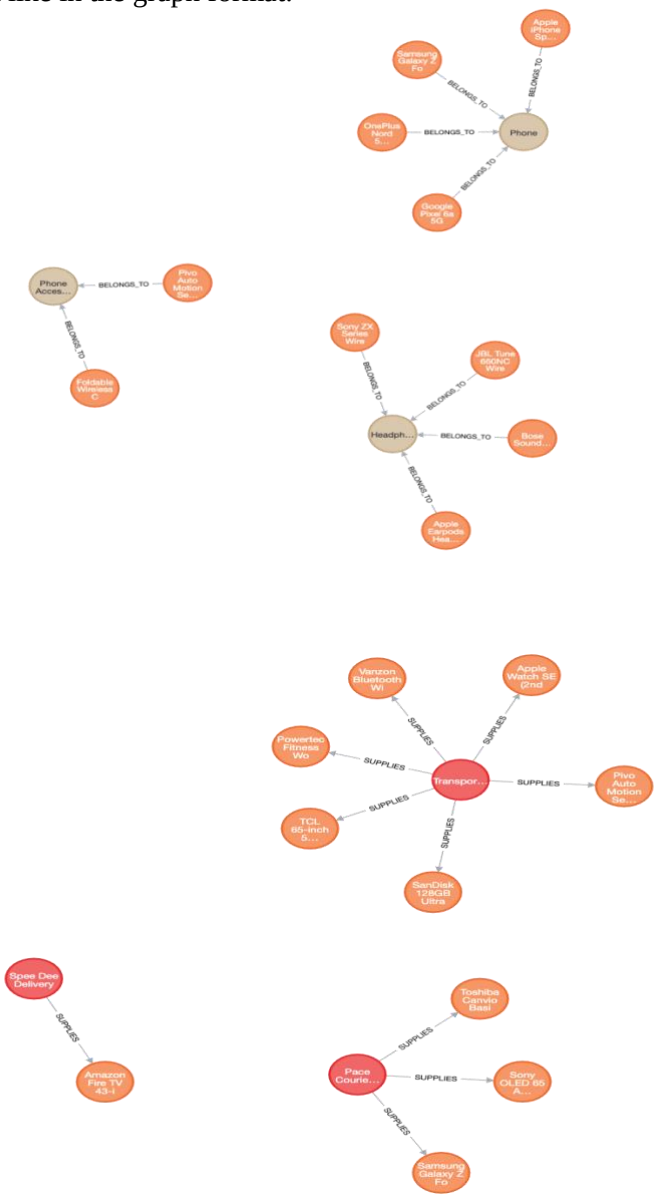
**# Top selling products**

```
use ecommerce_DB;
SELECT p.ProductName AS Product, COUNT(oi.OrderID) AS Total, oi.ProductID
FROM Product p
INNER JOIN OrderItems oi
ON p.ProductID = oi.ProductID
GROUP BY oi.ProductID
ORDER BY Total DESC
limit 5;
```

| Product | Total | ProductID |
|---|---|---|
| Yamaha YHT-5960U Home Theater System | 12 | 1502 |
| Watch Box Display Case Organizer Glass | 9 | 1500 |
| Powertec Fitness Workbench Accessory Rack | 8 | 1567 |
| Dual Electronics High Performance Speakers | 8 | 1544 |
| LG LHD657 Bluetooth 5.1-Channel Home Theat… | 8 | 1508 |

## Implementation in NoSQL:

A small part of the database model that we have created has been implemented in the Neo4j application as part of NoSQL implementation. This is a graph database that helps to easily visualize the relationship between different tables. Out of the 13 tables, we have implemented 5 important tables including Customer, Orders, Product, Category and Supplier. Cypher queries have been written to create these tables. Further, the relationships between these tables have been defined. Below are few screenshots of how these relationships look like in the graph format:




### *Cypher Queries:*

**# List of Apple products available**

```
MATCH (p:Product)-[:CONTAINS]-(ProductID)
WHERE p.Manufacturer = 'Apple'
RETURN p.ProductName AS Apple_Product;
```

| | Apple_Product |
|---|---|
| 1 | "Apple iPhone XS, Space Gray" |
| 2 | "Apple iPhone XS, Space Gray" |

**# Top 5 categories (based on no. of products available)**

```
MATCH(p:Product)-[:BELONGS_TO]-(Ct: Category)
RETURN Ct.CategoryID AS CatID,Ct.CategoryName AS CatName,Count(Ct.CategoryID)
AS No_Of_Products
Order By No_Of_Products DESC LIMIT 5
```

| | CatID | CatName | No_Of_Products |
|---|---|---|---|
| 1 | "414" | "TV" | 6 |
| 2 | "402" | "Phone Accessories " | 5 |
| 3 | "403" | "Speaker" | 5 |
| 4 | "401" | "Phone" | 4 |
| 5 | "400" | "Headphone" | 4 |

**# Top 5 suppliers (based on total worth of products shipped)**

```
MATCH(s:Supplier)-[:SUPPLIES]-(p:Product)
RETURN s.CompanyName AS Supplier_Name,Count(p.ProductID) AS TotalItem,
round(Sum(p.UnitPrice),2) AS TotalAmt
Order By TotalAmt DESC LIMIT 5
```

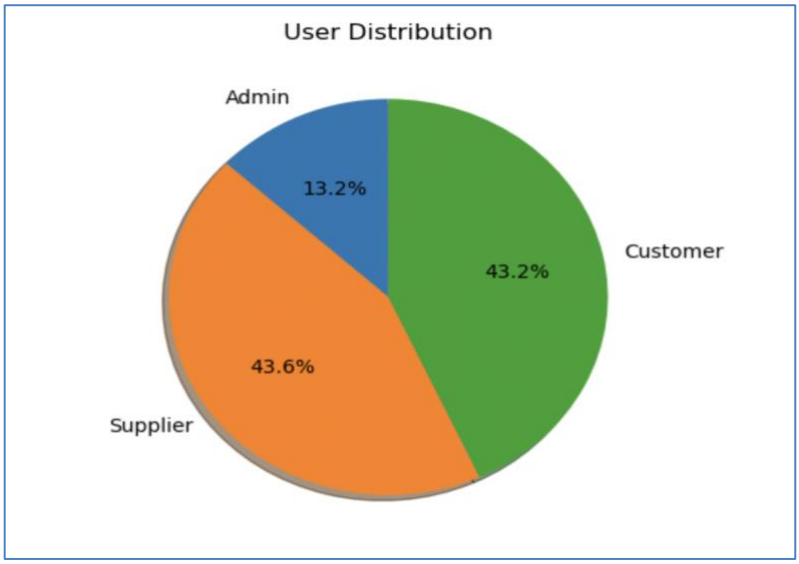
| | Supplier_Name | TotalItem | TotalAmt |
|---|---|---|---|
| 1 | "USPS" | 5 | 1321.82 |
| 2 | "Transportes Pitic" | 6 | 1186.4 |
| 3 | "Flat Export" | 5 | 1151.95 |
| 4 | "Sendle" | 4 | 1048.95 |
| 5 | "OnTrac Inc." | 3 | 1024.44 |

## Database Access Via Python (Application):

The database is accessed using Python and visualization of analyzed data is shown below. The connection of MySQL to Python is done using 'mysql.connector' and using 'mycursor.execute' to execute the query through pyhton followed by converting the list into a data frame using pandas library and using matplotlib and seaborn to plot the graphs for the analytics.

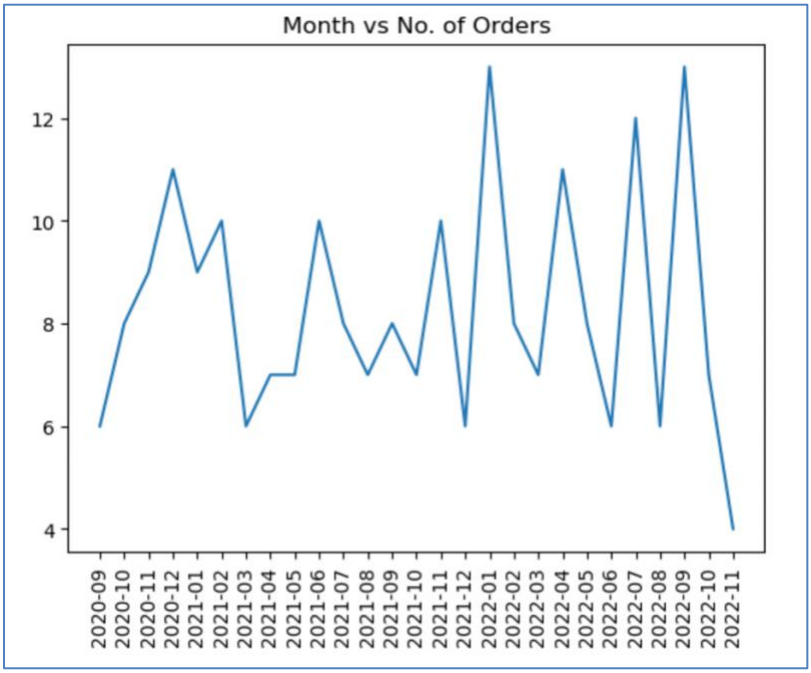
***Plot 1: Pie Chart for different type of user (Logins) in percentage***

mycursor.execute ("SELECT Role, COUNT(*) FROM Login GROUP BY Role")



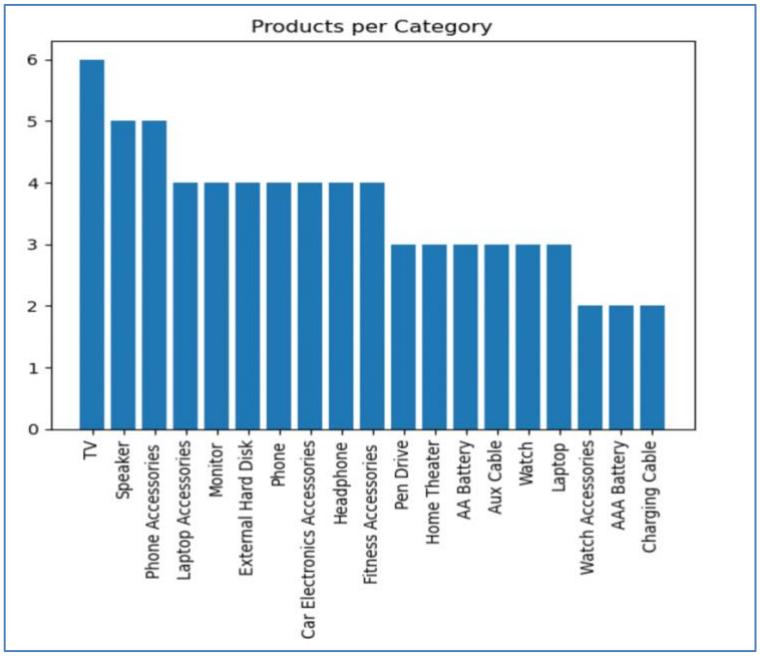

***Plot 2: Line Graph of orders per month of every year***

mycursor.execute ("SELECT DATE_FORMAT(OrderDate, '%Y-%m') AS Order_Date, COUNT(*) FROM Orders GROUP BY Order_Date ORDER BY Order_Date")



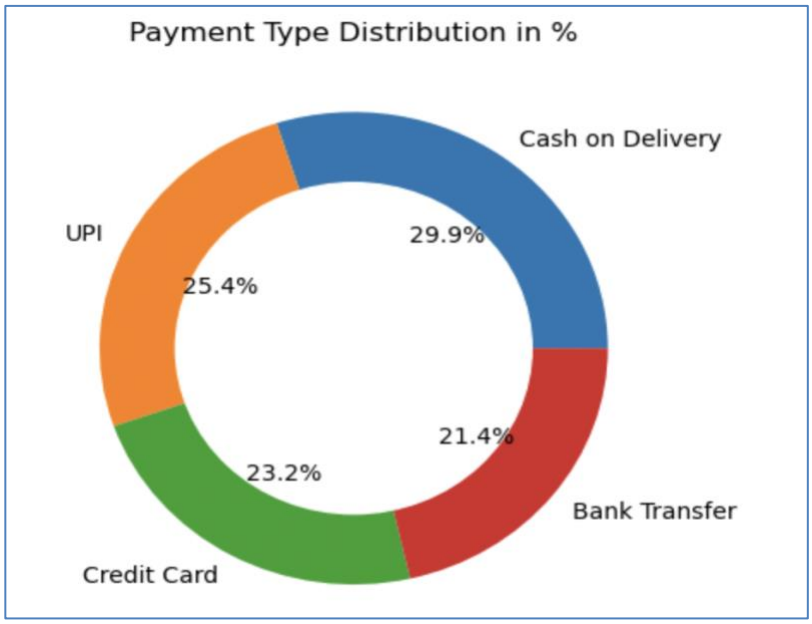
***Plot 3: Bar Graph of no. of products per category***

mycursor.execute ("SELECT C.CategoryName, COUNT(P.ProductID) FROM Product P, Category C WHERE P.CategoryID = C.CategoryID GROUP BY C.CategoryID ORDER BY COUNT(P.ProductID) DESC")



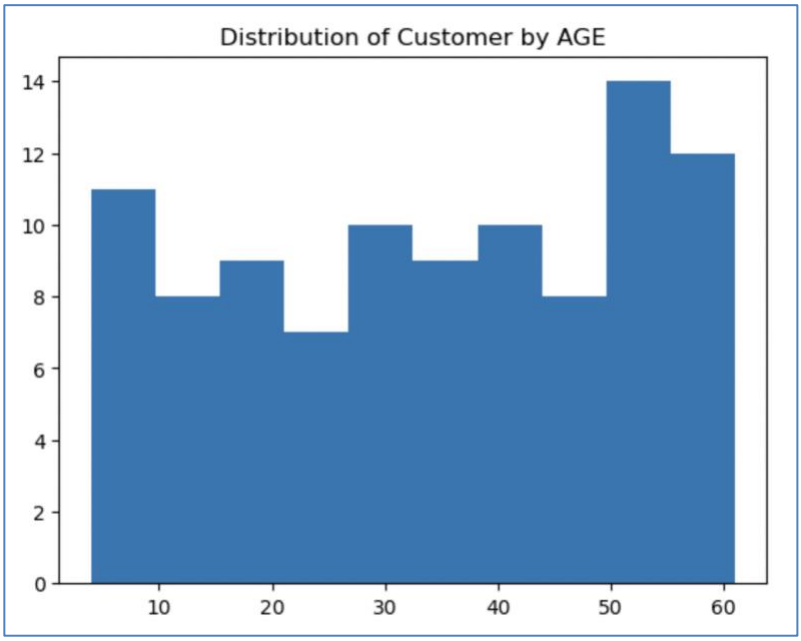

***Plot 4: Donut Chart for payment type distribution in percentage***

mycursor.execute ("SELECT PaymentType, COUNT(PaymentType) FROM Payment GROUP BY PaymentType ORDER BY COUNT(PaymentType) DESC")



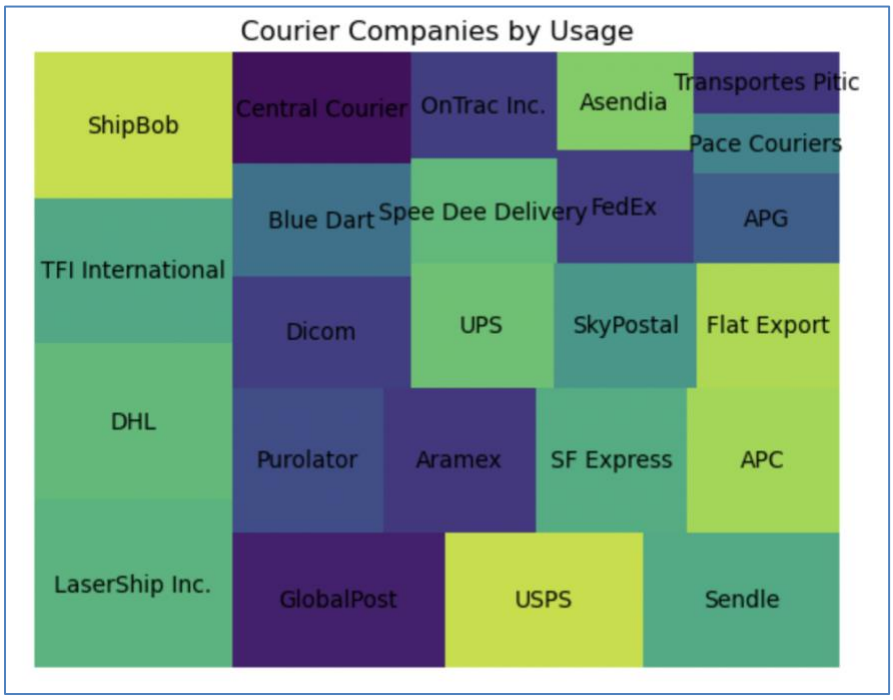
***Plot 5: Histogram of distribution of customers by age***

mycursor.execute ("SELECT DATE_FORMAT(FROM_DAYS(DATEDIFF(now(), DOB)), '%Y')+0 FROM Customer AS AGE")



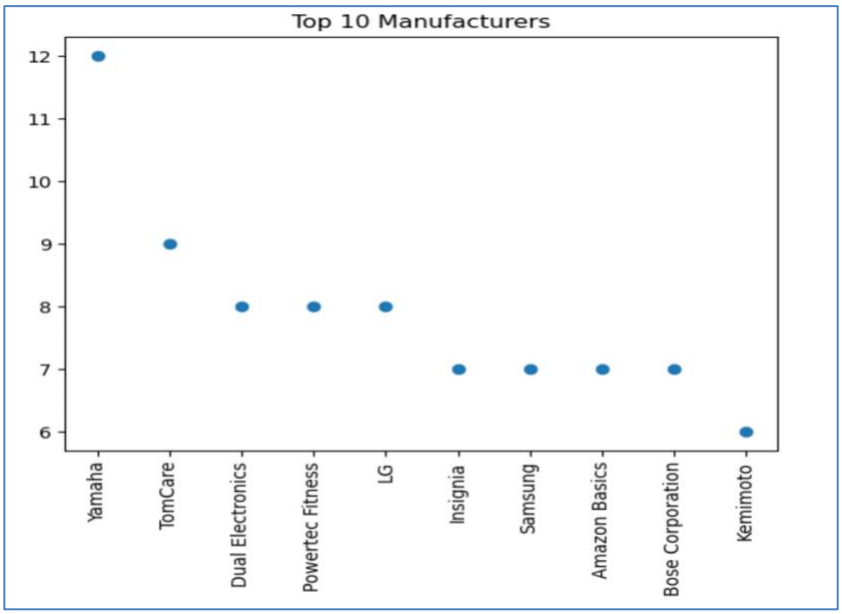

***Plot 6: Tree Map of courier companies preferred***

mycursor.execute ("SELECT C.CompanyName, count(O.OrderID) FROM Orders O, Courier C WHERE O.CourierID = C.CourierID GROUP BY C.CourierID ORDER BY COUNT(O.OrderID) DESC")



***Plot 7: Scatter Plot of top 10 manufacturers (most products sold)***

mycursor.execute ("SELECT P.ProductName, P.Manufacturer, COUNT(O.OrderID) FROM OrderItems O, Product P WHERE O.ProductID = P.ProductID GROUP BY P.ProductID ORDER BY count(O.OrderID) DESC LIMIT 10 ")




***Plot 8: Box Plot for distribution of Unit Price by each category***

mycursor.execute ("SELECT C.CategoryName, CAST(P.UnitPrice as unsigned) FROM Category C, Product P WHERE C.CategoryID = P.CategoryID ")