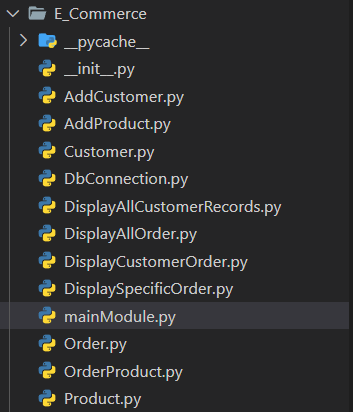
**E-Commerce Assignment**

**By - Ashwin Harish P**

**Directory Structure**

****

**Database**

CREATE DATABASE Ecommerce;

USE Ecommerce

**1.Customer Table**

CREATE TABLE Customer(

cid INT PRIMARY KEY IDENTITY(1,1),

cname VARCHAR(30),

caddress VARCHAR(50),

cmob VARCHAR(13)

);

**2.Product Table**

CREATE TABLE Product(

pid INT PRIMARY KEY IDENTITY(1,1),

pname VARCHAR(30),

pdesc VARCHAR(50),

price FLOAT

);

**3.Orders Table**

CREATE TABLE Orders(

oid INT PRIMARY KEY,

odate DATE,

cid INT,

pid INT,

qty INT,

price FLOAT,

total\_amt FLOAT,

FOREIGN KEY(cid) REFERENCES Customer(cid),

FOREIGN KEY(pid) REFERENCES Product(pid)

);

**4.Cart Table**

CREATE TABLE Cart(

oid INT,

odate DATE,

cid INT,

pid INT,

qty INT,

price FLOAT,

total\_amt FLOAT,

FOREIGN KEY(cid) REFERENCES Customer(cid),

FOREIGN KEY(pid) REFERENCES Product(pid)

);

**Python Code**

**\_\_init\_\_.py**

from .AddCustomer import addCustomer

from .AddProduct import addProduct

from .DisplayAllCustomerRecords import displayAllCustomerRecords

from .DisplayAllOrder import displayAllOrders

from .DisplaySpecificOrder import displaySpecificOrder

from .OrderProduct import orderProduct

from .DbConnection import getDBConnection

from .DisplayCustomerOrder import displayCustomerOrder

**DbConnection.py**

import pyodbc

def getDBConnection():

    driver\_name = "SQL Server"

    server = "LAPTOP-2Q84MA0J\SQLEXPRESS"

    databse = "Ecommerce"

    username = ""

    password = ""

    connectionString = f"Driver={driver\_name}; Server={server}; Database={databse}; UID={username}; PWD={password};"

    try:

        con = pyodbc.connect(connectionString)

        return con

    except Exception as Error:

        print("Error in the database connection", Error)

        return None

**mainModule.py**

from AddCustomer import addCustomer

from AddProduct import addProduct

from DisplayAllCustomerRecords import displayAllCustomerRecords

from DisplayAllOrder import displayAllOrder

from DisplaySpecificOrder import displaySpecificOrder

from OrderProduct import orderProduct

from DbConnection import getDBConnection

from DisplayCustomerOrder import displayCustomerOrder

con = getDBConnection()

if con:

    try:

        cursor = con.cursor()

    except:

        print("Error in Connection")

print("\n\t\tWelcome to E-Commerce Application")

print("\n\t\t Menu")

print("\n 1. Add Customer")

print("\n 2. Add Product")

print("\n 3. Order Product")

print("\n 4. Display All Orders")

print("\n 5. Display Specific Order")

print("\n 6. Display All Customer Records")

print("\n 7. Display Specific Customer Orders")

print("\n 8. Exit")

while True:

    choice = int(input("\n Enter your choice: "))

    if choice == 1:

        addCustomer()

    elif choice == 2:

        addProduct()

    elif choice == 3:

        orderProduct()

    elif choice == 4:

        displayAllOrder()

    elif choice == 5:

        displaySpecificOrder()

    elif choice == 6:

        displayAllCustomerRecords()

    elif choice == 7:

        displayCustomerOrder()

    elif choice == 8:

        print("Exiting from menu...\n")

        break

    else:

        print("!!! Invalid Choice !!!")

**AddCustomer.py**

from DbConnection import getDBConnection

def addCustomer():

    con = getDBConnection()

    print("\t\t\nEnter Customer Details")

    cname = input("Enter Your Name: ")

    caddress = input("Enter Your Address: ")

    cmob = input("Enter Your Mobile Number: ")

    if con:

        try:

            cursor = con.cursor()

            query = "INSERT INTO Customer(cname, caddress, cmob) VALUES(?, ?, ?)"

            cursor.execute(query, (cname, caddress, cmob))

            con.commit()

            print("\nCustomer Added")

        except Exception as Error:

            print("Error:", Error)

    else:

        print("No Connection Created")

**AddProduct.py**

from DbConnection import getDBConnection

def addProduct():

    con = getDBConnection()

    print("\t\t\nEnter Product Details")

    pname = input("Enter Poduct Name: ")

    pdesc = input("Enter Product Descriptoin: ")

    price = float(input("Enter Your Product Price: "))

    if con:

        try:

            cursor = con.cursor()

            query = "INSERT INTO Product(pname, pdesc, price) VALUES(?, ?, ?)"

            cursor.execute(query, (pname, pdesc, price))

            con.commit()

            print("\nProduct Added")

        except Exception as Error:

            print("Error:", Error)

    else:

        print("No Connection Created")

**Customer.py**

class Customer:

    def \_\_init\_\_(self, cid, cname, caddress, cmob):

        self.cid = cid

        self.cname = cname

        self.caddress = caddress

        self.cmob = cmob

    def getCustomer(self):

        print("\nCustomer Details:")

        print("Customer ID: ", self.cid)

        print("Customer Name: ", self.cname)

        print("Customer Address: ", self.caddress)

        print("Customer Mobile: ", self.cmob)

        print("-" \* 50)

**DisplayAllCustomerRecords.py**

from DbConnection import getDBConnection

from Customer import Customer

def displayAllCustomerRecords():

    con = getDBConnection()

    cursor = con.cursor()

    query = "SELECT cid, cname, caddress, cmob FROM Customer"

    cursor.execute(query)

    customers = cursor.fetchall()

    if not customers:

        print("\n !!! No customer records found !!!.")

        return

    print("\n\t\tAll Customer Records\n")

    for row in customers:

        cid, cname, caddress, cmob = row

        customer = Customer(cid, cname, caddress, cmob)

        customer.getCustomer()

**DisplayAllOrder.py**

from DbConnection import getDBConnection

from Order import Order

def displayAllOrder():

    con = getDBConnection()

    cursor = con.cursor()

    query = """

    SELECT o.oid, o.odate, o.cid, c.cname, c.caddress, c.cmob, o.pid, p.pname, p.pdesc, o.price, o.qty, o.total\_amt

    FROM Orders o

    JOIN Customer c

    ON o.cid = c.cid

    JOIN Product p

    ON o.pid = p.pid

    """

    cursor.execute(query)

    orders = cursor.fetchall()

    if not orders:

        print("\nNo orders found.")

        return

    print("\n\t\tAll Orders\n")

    for row in orders:

        order = Order(\*row)

        order.displayOrder()

**DisplayCustomerOrder.py**

from DbConnection import getDBConnection

from Customer import Customer

from Order import Order

def displayCustomerOrder():

    con = getDBConnection()

    cursor = con.cursor()

    print("\n Displaying the Customer Order Details")

    cid = int(input("\nEnter Customer id: "))

    query = """

    SELECT c.cid, c.cname, c.caddress, c.cmob, o.oid, o.odate, o.cid,

           o.pid, p.pname, p.pdesc, o.price, o.qty, o.total\_amt

    FROM Orders o

    JOIN Customer c ON o.cid = c.cid

    JOIN Product p ON o.pid = p.pid

    WHERE c.cid = ?

    """

    cursor.execute(query, (cid,))

    orders = cursor.fetchall()

    if not orders:

        print(f"\nNo customer order found with Customer ID: {cid}")

        return

    specificCustomer = Customer(orders[0][0], orders[0][1], orders[0][2], orders[0][3])

    specificCustomer.getCustomer()

    for order in orders:

        specificOrder = Order(

            order[4], order[5],

            specificCustomer.cid, specificCustomer.cname, specificCustomer.caddress, specificCustomer.cmob,

            order[6], order[7], order[8],

            order[9], order[10], order[11]

        )

        specificOrder.displayOrder()

**DisplaySpecificOrder.py**

from DbConnection import getDBConnection

from Order import Order

def displaySpecificOrder():

    con = getDBConnection()

    cursor = con.cursor()

    oid = int(input("Enter Order ID to display: "))

    query = """

    SELECT o.oid, o.odate, o.cid, c.cname, c.caddress, c.cmob,

           o.pid, p.pname, p.pdesc, o.price, o.qty, o.total\_amt

    FROM Orders o

    JOIN Customer c ON o.cid = c.cid

    JOIN Product p ON o.pid = p.pid

    WHERE o.oid = ?

    """

    cursor.execute(query, (oid,))

    order = cursor.fetchone()

    if not order:

        print(f"\nNo order found with Order ID: {oid}")

        return

    print("\n\t\tOrder Details\n")

    specific\_order = Order(\*order)

    specific\_order.displayOrder()

**Order.py**

from DbConnection import getDBConnection

from Customer import Customer

from Product import Product

class Order(Customer, Product):

    def \_\_init\_\_(self, oid, odate, cid, cname, caddress, cmob, pid, pname, pdesc, price, qty, total\_amt):

        Customer.\_\_init\_\_(self, cid, cname, caddress, cmob)

        Product.\_\_init\_\_(self, pid, pname, pdesc, price)

        self.oid = oid

        self.odate = odate

        self.qty = qty

        self.total\_amt = total\_amt

    def displayOrder(self):

        print("\n\t Product Details\n")

        print("Product ID: ", self.pid)

        print("Product Name: ", self.pname)

        print("Product Description: ", self.pdesc)

        print("Price per Unit: ", self.price)

        print("Quantity: ", self.qty)

        print("Total Amount: ", self.total\_amt)

        print("\n")

        print("\n\t\tOrder Details\n")

        print("Order ID: ", self.oid)

        print("Order Date: ", self.odate)

        print("-" \* 50)

**orderProduct.py**

from DbConnection import getDBConnection

from Product import Product

import random

from datetime import datetime

def orderProduct():

    con = getDBConnection()

    cursor = con.cursor()

    print("\t\t\nOrder Product")

    oid = random.randint(1000, 9999)

    odate = str(datetime.today().strftime('%Y-%m-%d'))

    cid  =int(input("Enter Customer id: "))

    pid = int(input("Enter product id: "))

    product = Product(pid, None, None, None)

    isProductExist = product.getProductID(pid)

    if isProductExist == 0:

        print("!!! Product is not there in the inventory !!!")

    else:

        qty = int(input("Enter Product Quantity: "))

        price = product.getPrice()

        total\_Amt = price \* qty

        print("\nTotal Price: ", total\_Amt)

        orderChoice = input("Are you sure to place this order?(Y/N)").upper()

        if orderChoice == "Y":

            orderQuerry = "INSERT INTO Orders(oid, odate, cid, pid, qty, price, total\_Amt) VALUES(?, ?, ?, ?, ?, ?, ?)"

            cursor.execute(orderQuerry, (oid, odate, cid, pid, qty, price, total\_Amt))

            con.commit()

            print("\nOrder Placed Sucessfully")

        else:

            orderQuerry = "INSERT INTO Cart(oid, odate, cid, pid, qty, price, total\_Amt) VALUES(?, ?, ?, ?, ?, ?, ?)"

            cursor.execute(orderQuerry, (oid, odate, cid, pid, qty, price, total\_Amt))

            con.commit()

            print("Item added in the cart")

**Product.py**

from DbConnection import getDBConnection

class Product:

    def \_\_init\_\_(self, pid, pname, pdesc, price):

        self.pid = pid

        self.pname = pname

        self.pdesc = pdesc

        self.price = price

    def getProduct(self):

        print("Product ID: ", self.pid)

        print("Product Name: ", self.pname)

        print("Product Description: ", self.pdesc)

        print("Product Price: ", self.price)

    def getPrice(self):

        con = getDBConnection()

        if con:

            try:

                cursor = con.cursor()

                query = "SELECT price FROM Product WHERE pid = ?"

                cursor.execute(query, (self.pid,))

                result = cursor.fetchone()

                return result[0]

            except Exception as Error:

                print("Error fetching price:", Error)

    def getProductID(self, pid):

        con = getDBConnection()

        if con:

            try:

                cursor = con.cursor()

                check\_query = "SELECT COUNT(\*) FROM Product WHERE pid = ?"

                cursor.execute(check\_query, (pid,))

                result = cursor.fetchone()

                return result[0] if result else 0

            except Exception as Error:

                print("Error Occoured", Error)