Module 11 .1

Tatiana Tan

### **Report 1: Wine Inventory and Price Report**

**Description:** This report shows the current inventory of wines along with their prices. This helps in tracking wine availability and pricing, which is crucial for pricing strategies and stock management.

**SQL Query:**

SELECT WineName, Type, Price  
FROM Wine;

**Python Script:**

python

Copy code

import mysql.connector  
from tabulate import tabulate  
  
# Connect to the MySQL database  
connection = mysql.connector.connect(  
 host='localhost',  
 user='root',  
 password='Peppabeppa12.',  
 database='BacchusWinery'  
)  
  
cursor = connection.cursor()  
  
# Function to display Wine Inventory Report  
def display\_wine\_inventory():  
 cursor.execute("SELECT WineName, Type, Price FROM Wine")  
 rows = cursor.fetchall()  
 headers = [i[0] for i in cursor.description]  
 print("\nWine Inventory and Price Report")  
 print(tabulate(rows, headers, tablefmt='psql'))  
  
# Display Wine Inventory Report  
display\_wine\_inventory()  
  
# Close the connection  
cursor.close()  
connection.close()

### **Report 2: Supplier Supply Summary Report**

**Description:** This report provides a summary of supplies from each supplier, including supply types and quantities. It helps in understanding supplier contributions and managing supplier relationships.

**SQL Query:**

SELECT s.SupplierName, sp.SupplyType, sp.Quantity  
FROM Supplier s  
JOIN Supply sp ON s.SupplierID = sp.SupplierID;

**Python Script:**

python

Copy code

import mysql.connector  
from tabulate import tabulate  
  
# Connect to the MySQL database  
connection = mysql.connector.connect(  
 host='localhost',  
 user='root',  
 password='Peppabeppa12.',  
 database='BacchusWinery'  
)  
  
cursor = connection.cursor()  
  
# Function to display Supplier Supply Summary Report  
def display\_supplier\_supply\_summary():  
 cursor.execute("""  
 SELECT s.SupplierName, sp.SupplyType, sp.Quantity  
 FROM Supplier s  
 JOIN Supply sp ON s.SupplierID = sp.SupplierID  
 """)  
 rows = cursor.fetchall()  
 headers = [i[0] for i in cursor.description]  
 print("\nSupplier Supply Summary Report")  
 print(tabulate(rows, headers, tablefmt='psql'))  
  
# Display Supplier Supply Summary Report  
display\_supplier\_supply\_summary()  
  
# Close the connection  
cursor.close()  
connection.close()

### **Report 3: Order Details Report**

**Description:** This report details all orders, including the distributor, order date, wine, and quantity. It helps in tracking orders and analyzing sales trends.

**SQL Query:**

SELECT o.OrderID, o.OrderDate, d.DistributorName, w.WineName, od.Quantity  
FROM Orders o  
JOIN Distributor d ON o.DistributorID = d.DistributorID  
JOIN OrderDetails od ON o.OrderID = od.OrderID  
JOIN Wine w ON od.WineID = w.WineID;

**Python Script:**

python

Copy code

import mysql.connector  
from tabulate import tabulate  
  
# Connect to the MySQL database  
connection = mysql.connector.connect(  
 host='localhost',  
 user='root',  
 password='Peppabeppa12.',  
 database='BacchusWinery'  
)  
  
cursor = connection.cursor()  
  
# Function to display Order Details Report  
def display\_order\_details():  
 cursor.execute("""  
 SELECT o.OrderID, o.OrderDate, d.DistributorName, w.WineName, od.Quantity  
 FROM Orders o  
 JOIN Distributor d ON o.DistributorID = d.DistributorID  
 JOIN OrderDetails od ON o.OrderID = od.OrderID  
 JOIN Wine w ON od.WineID = w.WineID  
 """)  
 rows = cursor.fetchall()  
 headers = [i[0] for i in cursor.description]  
 print("\nOrder Details Report")  
 print(tabulate(rows, headers, tablefmt='psql'))  
  
# Display Order Details Report  
display\_order\_details()  
  
# Close the connection  
cursor.close()  
connection.close()

A black screen with blue lines

Description automatically generatedA screenshot of a computer

Description automatically generatedA screenshot of a computer

Description automatically generated