**Team Name:** Blue Team

**Team Members:** Riley Dorrington, Robin Tageant, Brianne Hulgan, Andy Nguyen, Joseph Black

**Case Study:** Bacchus Winery

**Database Name:** bacchus\_winery

[**Create & Populate Code:**](https://docs.google.com/document/d/1A9nUR1Ce2rf_hlVVRZPc3WoAUyIeYQykhCg_qpns8s0/edit)

[**Queries Code:**](https://docs.google.com/document/d/1LTOArMbN4gNE2D0VnQeCj8VBEtfJ0FNgpf41rOG7EhI/edit)

[**Powerpoint:**](https://docs.google.com/presentation/d/1hwDQ2tSsZ5GxAe4nUHiXYmPDYyhdqdJTVl67u2A_4cA/edit#slide=id.p)

[**Database Tables**](https://docs.google.com/spreadsheets/d/1VqV_VlimUeIzK25mk5oyYJ1rHdlSGUj3sxTf6JbJrOI/edit?usp=sharing)

**Business Rules:**

A Company has many Employees

A Company has many Distributors

A Distributor has many Orders

A Company has many Suppliers

A Supplier has many Supplies

A Company has many Products

Ship Date must be after Order Date

Delivery Date must be After Order and Ship Date

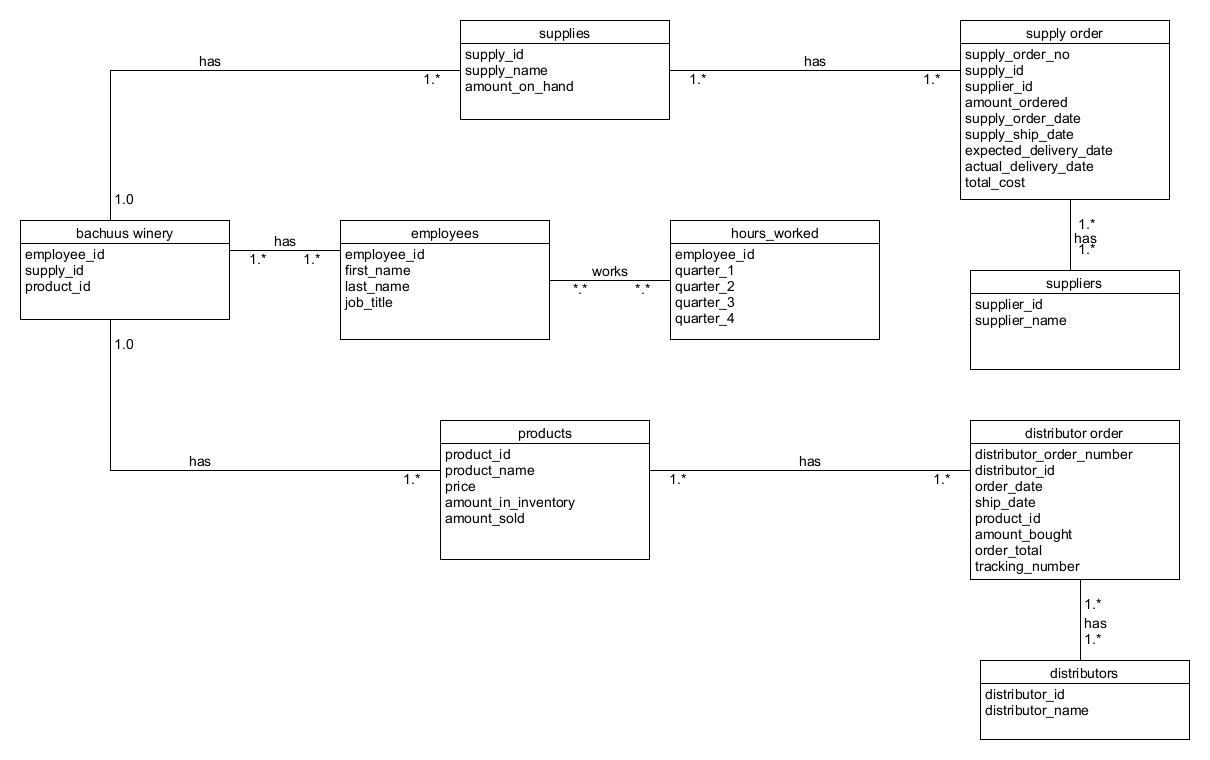
Total Prices are Calculated Automatically

We have set distributors for products to be sold to

We have set suppliers to buy supplies from

No Tables have empty fields.

**ORD:**



**Workers**

Stan and Davis Bacchus - Owners - inherited from father George

Janet Collins - Finances/Payroll

Roz Murphy - Marketing - has an assistant ⇒ Bob Ulrich

Henry Doyle - Manages production line and 20 employees

Maria Costanza - In charge of distribution

**Product**

They make 4 types of wine, Merlot, Cabernet, Chablis and Chardonnay, and grow their own grapes.

**3 Suppliers Ship Every Month or so**

1. Bottles and Corks
2. Labels and Boxes
3. Vats and Tubing

**Wants/Needs**

Stan and Davis are responsible for keeping track of shipments, they want a more efficient way of keeping track of supply inventory, and want a better way to order supplies, (over the internet).

Maria wants the distributors to be able to order online, and track shipments.

Company - Company Name (char)

Employee - Employee Id (int), First Name (char) , Last Name (char), Job Title (char), Hours Worked (int)

Products - Product Id (int), Product Name (char), Price (float), Amount in inventory (int) , Amount Sold (int)

Distributor Orders - Distributor Order Id (int), Distributor Id (int), Order Date (date), Ship Date (date), Product Id (int), Amount Bought (int), Total Price (float), Tracking Number (char)

Distributor - Distributor Id (int), Distributor Name (char)

Supplies - Supply Id (int), Supply Name (char), Amount on Hand (int)

Supply order - Supply Order Number (int), Supplier Id (int), Supply Id (int), Amount Ordered (int), Total Cost (float), Supply Order Date (date), Supply Ship Date (date), Expected Delivery Date (date), Actual Delivery Date (date)

Supplier - Supplier Id (int), Supplier Name (char)

**Determine state of inventory, distribution, and employees.**

**Questions to answer:**

* *Are all suppliers delivering on time? Is there a large gap between expected delivery and actual delivery? A month by month report should show problem areas.*
* *The wine distribution, are all wines selling as they thought? Is one wine not selling? Which distributor carries which wine?*
* *Employee time. During the last four quarters, how many hours did each employee work?*

**Record # 1:**

**Questions:** Are all suppliers delivering on time? Is there a large gap between expected delivery and actual delivery? A month by month report should show problem areas.

**Description: This record will run a query showing the expected delivery dates and the actual delivery dates of each supply order.**

**Code:**

cursor.execute("SELECT SUPPLY\_ORDER\_NUMBER, EXPECTED\_DELIVERY\_DATE, ACTUAL\_DELIVERY\_DATE FROM SUPPLY\_ORDERS")

delivery\_Dates = cursor.fetchall()

print("-- Expected vs. Actual Delivery Date --")

for row in delivery\_Dates:

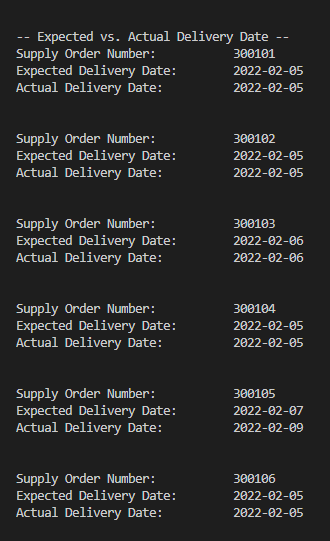
print("Supply Order Number: {}".format(row[0]))

print("Expected Delivery Date: {}".format(row[1]))

print("Actual Delivery Date: {}".format(row[2]))

print("\n")

**Output:**



**Record #2:**

**Questions:** The wine distribution, are all wines selling as they thought? Is one wine not selling? Which distributor carries which wine?

**Description: This record creates a join from the “Products” table, “Distributor” table and the “distributor order” table and runs a query displaying the distributor, the wine name and the amount the wine has sold.**

**Code:**

# Select the distributor name, wine name, and the amount of wine sold

cursor.execute('''SELECT DISTRIBUTOR.DISTRIBUTOR\_NAME, PRODUCTS.PRODUCT\_NAME, DISTRIBUTOR\_ORDERS.AMOUNT\_BOUGHT

FROM PRODUCTS INNER JOIN DISTRIBUTOR\_ORDERS ON PRODUCTS.PRODUCT\_ID = DISTRIBUTOR\_ORDERS.PRODUCT\_ID INNER JOIN

DISTRIBUTOR ON DISTRIBUTOR\_ORDERS.DISTRIBUTOR\_ID = DISTRIBUTOR.DISTRIBUTOR\_ID''')

# Storing the results of the query in a variable

wine\_sold = cursor.fetchall()

print("-- Wines Sold --")

# Looping through the results of the query

for row in wine\_sold:

print("Distributor: {}".format(row[0]))

print("Wine: {}".format(row[1]))

print("Amount Sold: {}".format(row[2]))

print("\n")

**Output:**

****

**Record #3:**

**Question:** Employee time. During the last four quarters, how many hours did each employee work?

**Description: This record creates a join combining the “employee” and “hours worked” tables to display each employee and how many hours they have worked over the last four quarters.**

**Code:**

# select the employee name and the numbers of hours worked per quarter

cursor.execute('''SELECT EMPLOYEE.FIRST\_NAME, EMPLOYEE.LAST\_NAME, HOURS\_WORKED.Q1, HOURS\_WORKED.Q2, HOURS\_WORKED.Q3, HOURS\_WORKED.Q4

FROM EMPLOYEE INNER JOIN HOURS\_WORKED ON EMPLOYEE.EMPLOYEE\_ID = HOURS\_WORKED.EMPLOYEE\_ID''')

# Storing the results of the query in a variable

hours\_worked = cursor.fetchall()

print("-- Hours Worked --")

# Looping through the results of the query

for row in hours\_worked:

print("Employee: {}".format(row[0]) + " " + row[1])

print("Q1: {}".format(row[2]) + " hours")

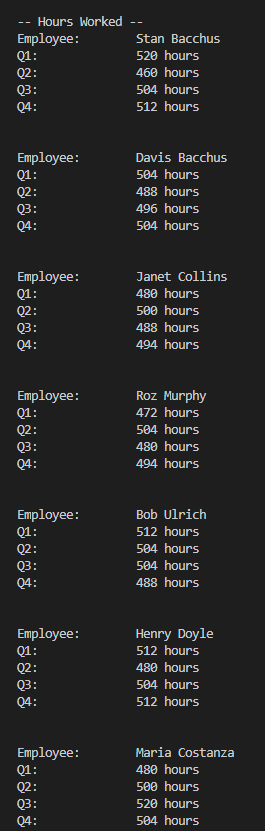
print("Q2: {}".format(row[3]) + " hours")

print("Q3: {}".format(row[4]) + " hours")

print("Q4: {}".format(row[5]) + " hours")

print("\n")

**Output:**

****