

**Week 7 Module 10.1 Case Study Milestone #2**

Joel Atkinson, Zachary Baker, Kyle Klausen, Juan Macias Vasquez, Brittany Perry-Morgan

Bellevue University

CSD310-H323 Database Development and Use (2255-DD)

**Adam Bailey**

July 13<sup>th</sup>, 2025

## Module 10.1 Case Study Milestone #2

### Group Name:

Group 1 The best group!

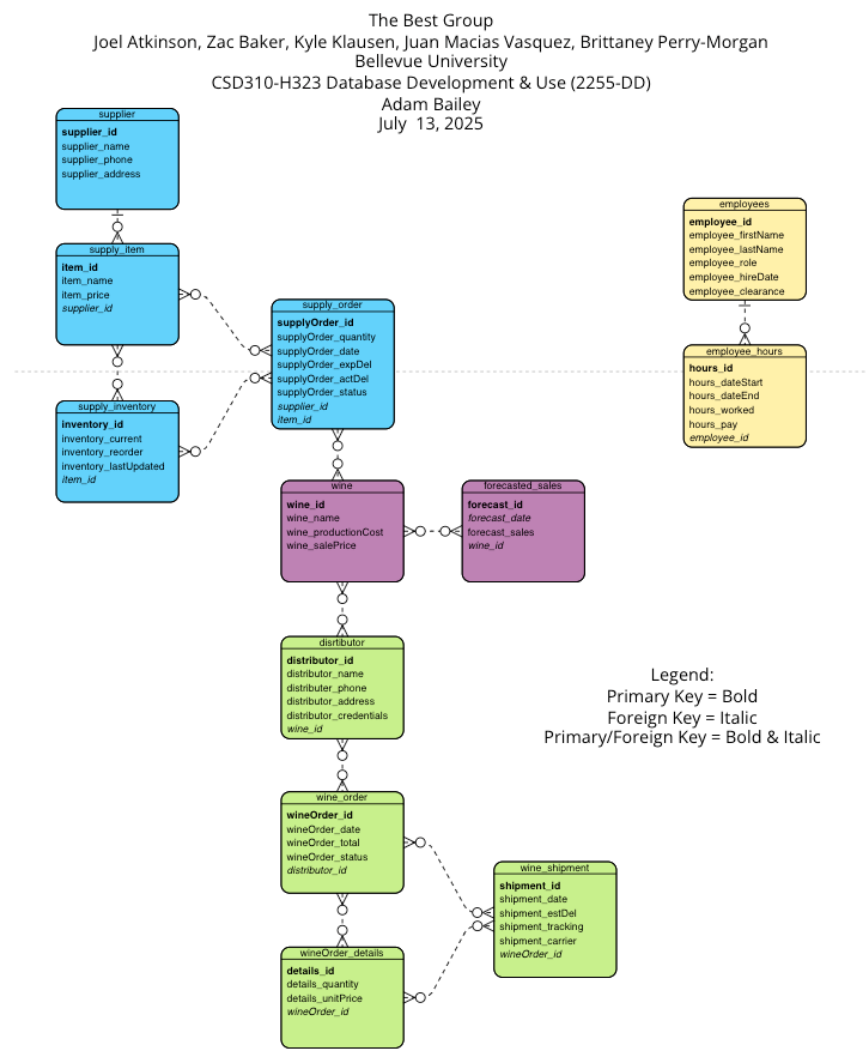
### Team Members:

Joel Atkinson, Zachary Baker, Kyle Klausen, Juan Macias Vasquez, Brittane Perry-Morgan

### Chosen Case Study:

Bacchus Winery

### Updates ERD



## Python Code

```
#Team Members: Joel Atkinson, Zachary Baker, Kyle Klausen, Juan Macias Vasquez,
Brittaney Perry-Morgan
# Date 07/13/2025
# Module 10.2 Group1 Winery Database
# Grouptables.py

import mysql.connector

# Connection configuration
config = {
    'user': 'root',
    'password': 'Powerful@2025', #Change to your own password
    'host': 'localhost'
}

def execute_sql_file(cursor, filename):
    with open(filename, 'r') as file:
        lines = file.readlines()

    statement = ''
    for line in lines:
        stripped = line.strip()

        if not stripped or stripped.startswith('--') or stripped.startswith('#'):
            continue

        statement += ' ' + stripped

        if stripped.endswith(';'):
            try:
                # Optional fix for subqueries with multiple results
                if "SELECT" in statement.upper() and "FROM" in statement.upper():
                    if "employee_id FROM employees WHERE employee_firstName" in
statement:
                        # Auto-fix: add LIMIT 1 to subqueries that cause errors
                        statement = statement.replace("'", "' LIMIT 1)") # crude
            except:
                pass

    but works here

    cursor.execute(statement)
    except mysql.connector.Error as err:
        print("Error executing SQL:")
        print(statement)
        print("MySQL Error:", err)
    statement = ''

def get_table_names(cursor):
    cursor.execute("SHOW TABLES")
    return [row[0] for row in cursor.fetchall()]

def get_primary_key(cursor, table):
    cursor.execute(f"SHOW KEYS FROM {table} WHERE Key_name = 'PRIMARY'")
    result = cursor.fetchone()
    return result[4] if result else None

def delete_duplicates(cursor, table, primary_key):
```

```

cursor.execute(f"SHOW COLUMNS FROM {table}")
columns = [row[0] for row in cursor.fetchall()]
non_pk_columns = [col for col in columns if col != primary_key]

if not non_pk_columns:
    return

match_conditions = " AND ".join([f"t1.{col} = t2.{col}" for col in
non_pk_columns])

sql = f"""
DELETE t1 FROM {table} t1
INNER JOIN {table} t2
WHERE t1.{primary_key} > t2.{primary_key}
AND {match_conditions}
"""
try:
    cursor.execute(sql)
except mysql.connector.Error as err:
    print(f"Error deduplicating {table}: {err}")

def display_table(cursor, table):
    cursor.execute(f"SELECT * FROM {table}")
    rows = cursor.fetchall()
    columns = [desc[0] for desc in cursor.description]

    print(f"\n--- {table.upper()} ---")
    print(" | ".join(columns))
    print("-" * 60)
    for row in rows:
        print(" | ".join(str(val) for val in row))
    print(f"Total Rows: {len(rows)}")

def main():
    conn = None
    cursor = None
    try:
        conn = mysql.connector.connect(**config)
        cursor = conn.cursor()

        # Ensure database exists and select it
        cursor.execute("CREATE DATABASE IF NOT EXISTS Winery")
        cursor.execute("USE Winery")

        # Execute all SQL statements from file
        execute_sql_file(cursor, 'tables.sql')
        conn.commit()

        # Confirm use of Winery
        cursor.execute("USE Winery")

        # Deduplicate and show tables
        tables = get_table_names(cursor)
        for table in tables:
            pk = get_primary_key(cursor, table)
            if pk:
                delete_duplicates(cursor, table, pk)
            conn.commit()
    
```

```

        for table in tables:
            display_table(cursor, table)

    except mysql.connector.Error as err:
        print(f"Database error: {err}")
    finally:
        if cursor:
            cursor.close()
        if conn and conn.is_connected():
            conn.close()

if __name__ == "__main__":
    main()

```

Screen shots of code running

The screenshot shows a Visual Studio Code window with a Python script named `GroupTables.py` running. The script is connected to a MySQL database and is performing the following actions:

- Connecting to the database.
- Dropping the database `users` if it exists.
- Creating a new database `users`.
- Granting all privileges on the `users` database to the user `movies_user`.
- Setting foreign key checks.
- Dropping tables `users`, `movies_user`, `employees`, `hours`, `distros`, `distro_credentials`, `distro_addresses`, `distro_phones`, `distro_units`, `employees`, `hours`, `distros`, `distro_credentials`, `distro_addresses`, `distro_phones`, `distro_units` if they exist.
- Creating the `DISTRIBUTOR` table with columns: `distributor_id`, `distributor_name`, `distributor_phone`, `distributor_address`, `distributor_credentials`, `distributor_unit`.
- Creating the `EMPLOYEE_HOURS` table with columns: `hours_id`, `hours_dateStart`, `hours_dateEnd`, `hours_worked`, `hours_pay`, `employee_id`.
- Inserting data into the `DISTRIBUTOR` table.

The output of the script shows the table structures and the data inserted into the `DISTRIBUTOR` table:

```

--- DISTRIBUTOR ---
distributor_id | distributor_name | distributor_phone | distributor_address | distributor_credentials | distributor_unit
-----
1 | WineBrothers | 402.154.1515 | 1231 Monroe Street | License#A12 | 520 | 1
2 | GoldenDistribution | 402.789.5465 | 5438 Jefferson Street | License#B22 | 613 | 2
3 | QualityWine | 402.789.5465 | 5438 Jefferson Street | License#C33 | 321 | 4
Total Rows: 3

```

The output also shows the structure of the `EMPLOYEE_HOURS` table:

```

--- EMPLOYEE_HOURS ---
hours_id | hours_dateStart | hours_dateEnd | hours_worked | hours_pay | employee_id
-----
1 | 2025-04-01 | 2025-06-30 | 412 | 30 | 1
2 | 2025-04-01 | 2025-06-30 | 405 | 30 | 2
3 | 2025-04-01 | 2025-06-30 | 410 | 25 | 3
4 | 2025-04-01 | 2025-06-30 | 377 | 25 | 4
5 | 2025-04-01 | 2025-06-30 | 450 | 18 | 5
6 | 2025-04-01 | 2025-06-30 | 410 | 25 | 6
7 | 2025-04-01 | 2025-06-30 | 433 | 25 | 7
8 | 2025-01-01 | 2025-03-31 | 432 | 30 | 1
9 | 2025-01-01 | 2025-03-31 | 445 | 30 | 2
10 | 2025-01-01 | 2025-03-31 | 384 | 25 | 3
11 | 2025-01-01 | 2025-03-31 | 412 | 25 | 4
12 | 2025-01-01 | 2025-03-31 | 398 | 18 | 5
13 | 2025-01-01 | 2025-03-31 | 458 | 25 | 6
14 | 2025-01-01 | 2025-03-31 | 431 | 25 | 7
15 | 2024-10-01 | 2024-12-31 | 455 | 30 | 1
16 | 2024-10-01 | 2024-12-31 | 441 | 30 | 2
17 | 2024-10-01 | 2024-12-31 | 405 | 25 | 3

```

```
File Edit View Git Project Build Debug Test Analyze Tools Extensions Window Help Search GroupTables
Process: [24416] GroupTables.py
tables.sql x GroupTables.py
1 -- Connecting-- to dat
2
3 -- drop database user a
4 DROP USER IF EXISTS 'wi
5
6 -- create movies_user a
7 CREATE USER 'wine_emplo
8
9 -- grant all privileges
10 GRANT ALL PRIVILEGES ON
11
12 -- SET FOREIGN_KEY_CHECK
13 SET FOREIGN_KEY_CHECKS
14
15 -- drop tables if they
16 DROP TABLE IF EXISTS su
17 DROP TABLE IF EXISTS su
18 DROP TABLE IF EXISTS su
19 DROP TABLE IF EXISTS wi
20 DROP TABLE IF EXISTS fo
21 DROP TABLE IF EXISTS di
22 DROP TABLE IF EXISTS wi
23 DROP TABLE IF EXISTS wi
24 DROP TABLE IF EXISTS wi
25 DROP TABLE IF EXISTS wi
26 DROP TABLE IF EXISTS em
27 DROP TABLE IF EXISTS em
28
100% No issues found
Autos Search (Ctrl+E)
Name Value
Type
Autos Locals Threads Modules Watch 1
Ready
C:\Users\JoshN\AppData\Loc...
C:\Users\JoshN\AppData\Loc...
12 | 2025-01-01 | 2025-03-31 | 398 | 18 | 5
13 | 2025-01-01 | 2025-03-31 | 458 | 25 | 6
14 | 2025-01-01 | 2025-03-31 | 431 | 25 | 7
15 | 2024-10-01 | 2024-12-31 | 455 | 30 | 1
16 | 2024-10-01 | 2024-12-31 | 441 | 30 | 2
17 | 2024-10-01 | 2024-12-31 | 405 | 25 | 3
18 | 2024-10-01 | 2024-12-31 | 395 | 25 | 4
19 | 2024-10-01 | 2024-12-31 | 320 | 18 | 5
20 | 2024-10-01 | 2024-12-31 | 415 | 25 | 6
21 | 2024-10-01 | 2024-12-31 | 441 | 25 | 7
22 | 2024-07-01 | 2024-09-30 | 348 | 30 | 1
23 | 2024-07-01 | 2024-09-30 | 378 | 30 | 2
24 | 2024-07-01 | 2024-09-30 | 422 | 25 | 3
25 | 2024-07-01 | 2024-09-30 | 436 | 25 | 4
26 | 2024-07-01 | 2024-09-30 | 404 | 18 | 5
27 | 2024-07-01 | 2024-09-30 | 379 | 25 | 6
28 | 2024-07-01 | 2024-09-30 | 429 | 25 | 7
Total Rows: 28
--- EMPLOYEES ---
employee_id | employee_firstName | employee_lastName | employee_role | employee_hireDate | employee_clearance
-----
1 | Stan | Bacchus | Co-CEO | 2023-07-12 | True
2 | Davis | Bacchus | Co-CEO | 2023-07-12 | True
3 | Janet | Collins | Finance Manager | 2023-12-22 | True
4 | Roz | Murphy | Marketing Manager | 2023-12-23 | True
5 | Bob | Ulrich | Marketing Associate | 2024-01-05 | False
6 | Henry | Doyle | Production Manager | 2023-08-07 | True
7 | Maria | Costanza | Distribution Manager | 2024-02-15 | True
Total Rows: 7
** Visual Studio 2022 Developer PowerShell v17.14.7
** Copyright (c) 2025 Microsoft Corporation
PS C:\Users\JoshN\Downloads\SchoolBelvue\Summer2025\Database Development and Use\Week 7\Module 10\GroupTables>
```

```
File Edit View Git Project Build Debug Test Analyze Tools Extensions Window Help Search GroupTables
Process: [24416] GroupTables.py
tables.sql x GroupTables.py
1 #Team Members: Joel Atk
2 # Date 07/13/2025
3 # Module 10.2 Group1 Wi
4 # GroupTables.py
5
6 import mysql.connector
7
8 # Connection configurat
9 config = {
10     'user': 'root',
11     'password': 'Powerf
12     'host': 'localhost'
13 }
14
15 def execute_sql_file(cu
16     with open(filename,
17         lines = file.re
18
19     statement = ''
20     for line in lines:
21         stripped = line
22         if not stripped
23             continue
24
25
100% No issues found
Autos Search (Ctrl+E)
Name Value
Type
Autos Locals Threads Modules Watch 1
Ready
C:\Users\JoshN\AppData\Loc...
C:\Users\JoshN\AppData\Loc...
--- FORECASTED_SALES ---
forecast_id | forecast_date | forecast_sales | wine_id
-----
1 | 2025-07-12 | 6000 | 1
2 | 2025-07-12 | 5100 | 2
3 | 2025-07-12 | 9600 | 3
4 | 2025-07-12 | 8000 | 4
Total Rows: 4
--- SUPPLIER ---
supplier_id | supplier_name | supplier_phone | supplier_address
-----
1 | SuppliesInc | 402.888.1212 | 1234 Maple Street
2 | ContainerGalore | 402.951.3000 | 575 Washington Boulevard
3 | WinerySupplies | 402.772.2315 | 9000 Blondo Street
Total Rows: 3
--- SUPPLY_INVENTORY ---
inventory_id | inventory_current | inventory_reorder | inventory_lastUpdated | item_id
-----
1 | 5122 | 500 | 2025-07-12 | 1
2 | 2400 | 2000 | 2025-07-12 | 2
3 | 6124 | 1200 | 2025-07-12 | 3
4 | 1023 | 515 | 2025-07-12 | 4
5 | 820 | 110 | 2025-07-12 | 5
6 | 4000 | 100 | 2025-07-12 | 6
Total Rows: 6
--- SUPPLY_ITEM ---
** Visual Studio 2022 Developer PowerShell v17.14.7
** Copyright (c) 2025 Microsoft Corporation
PS C:\Users\JoshN\Downloads\SchoolBelvue\Summer2025\Database Development and Use\Week 7\Module 10\GroupTables>
```

```
File Edit View Git Project Build Debug Test Analyze Tools Extensions Window Help Search GroupTables
Process: [24416] GroupTables.py
tables.sql GroupTables.py
1 #Team Members: Joel Atk
2 # Date 07/13/2025
3 # Module 10.2 Group1 Wi
4 # GroupTables.py
5
6 import mysql.connector
7
8 # Connection configurat
9
10 config = {
11     'user': 'root',
12     'password': 'Powerf
13     'host': 'localhost'
14 }
15
16 def execute_sql_file(cu
17     with open(filename,
18         lines = file.re
19
20 statement = ''
21 for line in lines:
22     stripped = line
23     if not stripped
24         continue
25
--- SUPPLY_ITEM ---
item_id | item_name | item_price | supplier_id
-----
1 | Bottle | 5 | 1
2 | Cork | 1 | 1
3 | Label | 2 | 2
4 | Box | 7 | 2
5 | Vat | 22 | 3
6 | Tubing | 10 | 3
Total Rows: 6
--- SUPPLY_ORDER ---
supplyOrder_id | supplyOrder_quantity | supplyOrder_date | supplyOrder_expDel | supplyOrder_actDel | supplyOrder_status
| supplier_id | item_id
-----
1 | 120 | 2025-05-12 | 2025-06-12 | 2025-06-20 | Delivered | 1 | 1
2 | 200 | 2025-05-12 | 2025-06-12 | 2025-06-20 | Delivered | 1 | 2
3 | 312 | 2025-05-12 | 2025-06-04 | 2025-06-04 | Delivered | 2 | 3
4 | 55 | 2025-05-12 | 2025-06-04 | 2025-06-04 | Delivered | 2 | 4
5 | 10 | 2025-05-12 | 2025-06-10 | 2025-06-11 | Delivered | 3 | 5
6 | 42 | 2025-05-12 | 2025-06-10 | 2025-06-11 | Delivered | 3 | 6
Total Rows: 6
--- WINE ---
wine_id | wine_name | wine_productionCost | wine_salePrice
-----
1 | Merlot | 8 | 12
2 | Cabernet | 5 | 10
3 | Chablis | 10 | 19
-----
** Visual Studio 2022 Developer PowerShell v17.14.7
** Copyright (c) 2025 Microsoft Corporation
PS C:\Users\joshn\Downloads\SchoolBellevue\Summer2025\Database Development and Use\Week 7\Module 10\Grou
pTables>
```

```
File Edit View Git Project Build Debug Test Analyze Tools Extensions Window Help Search GroupTables
Process: [24416] GroupTables.py
tables.sql GroupTables.py
1 #Team Members: Joel Atk
2 # Date 07/13/2025
3 # Module 10.2 Group1 Wi
4 # GroupTables.py
5
6 import mysql.connector
7
8 # Connection configurat
9
10 config = {
11     'user': 'root',
12     'password': 'Powerf
13     'host': 'localhost'
14 }
15
16 def execute_sql_file(cu
17     with open(filename,
18         lines = file.re
19
20 statement = ''
21 for line in lines:
22     stripped = line
23     if not stripped
24         continue
25
1 Merlot | 8 | 12
2 Cabernet | 5 | 10
3 Chablis | 10 | 19
4 Chardonnay | 9 | 20
Total Rows: 4
--- WINE_ORDER ---
wineOrder_id | wineOrder_date | wineOrder_total | wineOrder_status | distributor_id
-----
1 | 2025-07-03 | 8600 | Shipping | 1
2 | 2025-07-06 | 9850 | Shipping | 2
3 | 2025-07-06 | 3500 | Shipping | 3
Total Rows: 3
--- WINE_SHIPMENT ---
shipment_id | shipment_date | shipment_estDel | shipment_tracking | shipment_carrier | wineOrder_id
-----
1 | 2025-07-13 | 2025-07-23 | 1554894321 | None | 1
2 | 2025-07-16 | 2025-07-26 | 1557414354 | None | 2
3 | 2025-07-13 | 2025-07-23 | 1559994322 | None | 3
Total Rows: 3
--- WINEORDER_DETAILS ---
details_id | details_quantity | details_unitPrice | wineOrder_id
-----
1 | 700 | 12 | 1
2 | 940 | 15 | 2
3 | 350 | 19 | 3
Total Rows: 3
Press any key to continue . . . |
-----
** Visual Studio 2022 Developer PowerShell v17.14.7
** Copyright (c) 2025 Microsoft Corporation
PS C:\Users\joshn\Downloads\SchoolBellevue\Summer2025\Database Development and Use\Week 7\Module 10\Grou
pTables>
```