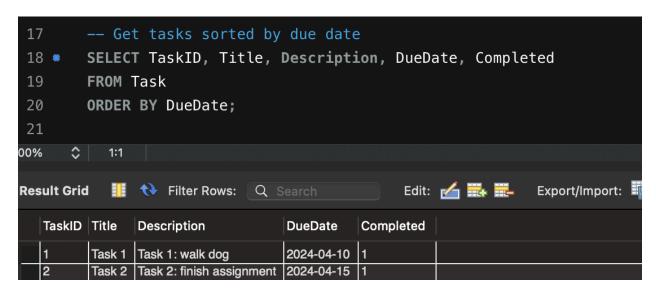
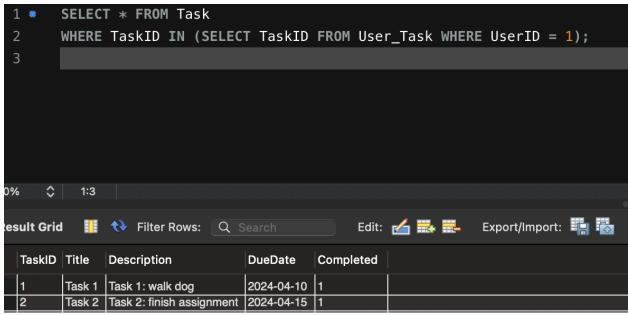
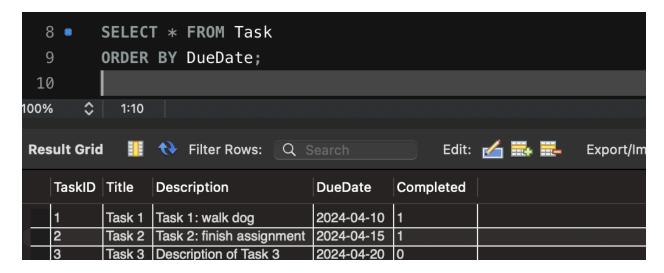
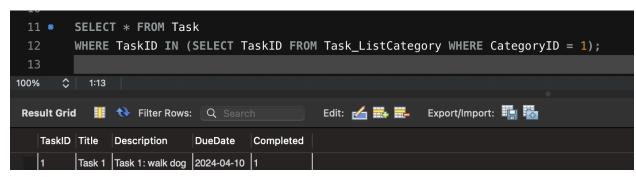
COP 4703

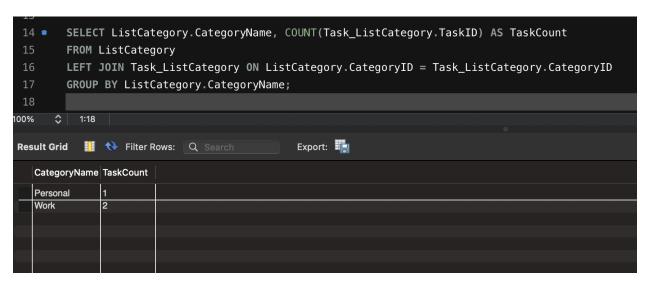
Task 3











My Sql Code:

-- Create User table

CREATE TABLE User (

UserID INT AUTO_INCREMENT PRIMARY KEY,

```
Username VARCHAR(50) NOT NULL,
  Password VARCHAR(50) NOT NULL,
  Email VARCHAR(100) NOT NULL
);
-- Create Task table
CREATE TABLE Task (
  TaskID INT AUTO INCREMENT PRIMARY KEY,
  Title VARCHAR(100) NOT NULL,
  Description TEXT,
  DueDate DATE,
  Completed BOOLEAN DEFAULT 0
);
-- Create ListCategory table
CREATE TABLE ListCategory (
  CategoryID INT AUTO INCREMENT PRIMARY KEY,
 CategoryName VARCHAR(50) NOT NULL
);
-- Create User_Task table for many-to-many relationship between User and Task
CREATE TABLE User Task (
  UserID INT,
```

```
TaskID INT,
  PRIMARY KEY (UserID, TaskID),
  FOREIGN KEY (UserID) REFERENCES User(UserID),
  FOREIGN KEY (TaskID) REFERENCES Task(TaskID)
);
-- Create Task ListCategory table for many-to-many relationship between Task and ListCategory
CREATE TABLE Task ListCategory (
  TaskID INT,
  CategoryID INT,
  PRIMARY KEY (TaskID, CategoryID),
  FOREIGN KEY (TaskID) REFERENCES Task(TaskID),
  FOREIGN KEY (CategoryID) REFERENCES ListCategory(CategoryID)
);
-- Insert sample data into User table
INSERT INTO User (UserID, Username, Password, Email) VALUES
(1, 'user1', 'password1', 'user1@example.com'),
(2, 'user2', 'password2', 'user2@example.com');
-- Insert sample data into Task table
INSERT INTO Task (TaskID, Title, Description, DueDate, Completed) VALUES
(1, 'Task 1', 'Task 1: walk the dog', '2024-04-10', false),
(2, 'Task 2', Task 2: finish assignment', '2024-04-15', true);
```

```
-- Insert sample data into ListCategory table
INSERT INTO ListCategory (CategoryID, CategoryName) VALUES
(1, 'Personal'),
(2, 'Work');
-- Insert sample data into User_Task table
INSERT INTO User Task (UserID, TaskID) VALUES
(1, 1),
(1, 2),
(2, 2);
-- Insert sample data into Task_ListCategory table
INSERT INTO Task_ListCategory (TaskID, CategoryID) VALUES
(1, 1),
(1, 2),
(2, 2);
SELECT * FROM Task
WHERE TaskID IN (SELECT TaskID FROM User_Task WHERE UserID = 1);
UPDATE Task
SET Completed = 1
WHERE TaskID = 1;
SELECT * FROM Task
```

ORDER BY DueDate;

SELECT * FROM Task

WHERE TaskID IN (SELECT TaskID FROM Task_ListCategory WHERE CategoryID = 1);

SELECT ListCategory.CategoryName, COUNT(Task_ListCategory.TaskID) AS TaskCount

FROM ListCategory

LEFT JOIN Task_ListCategory ON ListCategory.CategoryID = Task_ListCategory.CategoryID GROUP BY ListCategory.CategoryName;