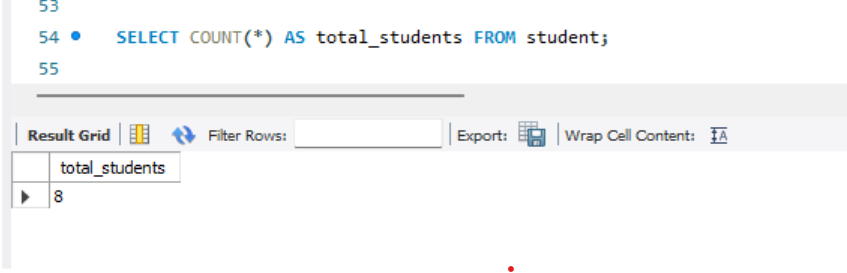
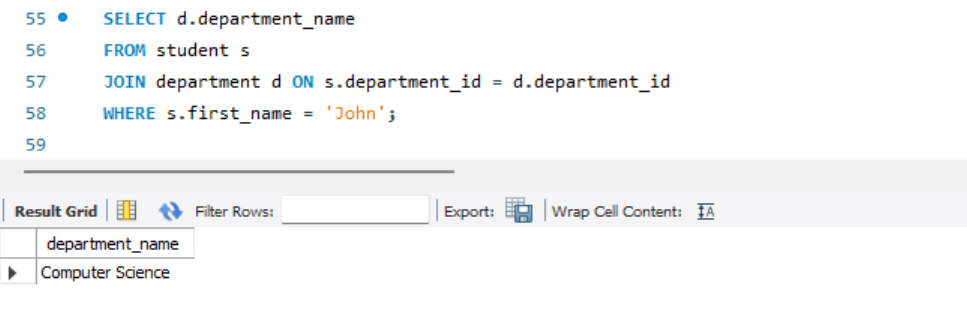
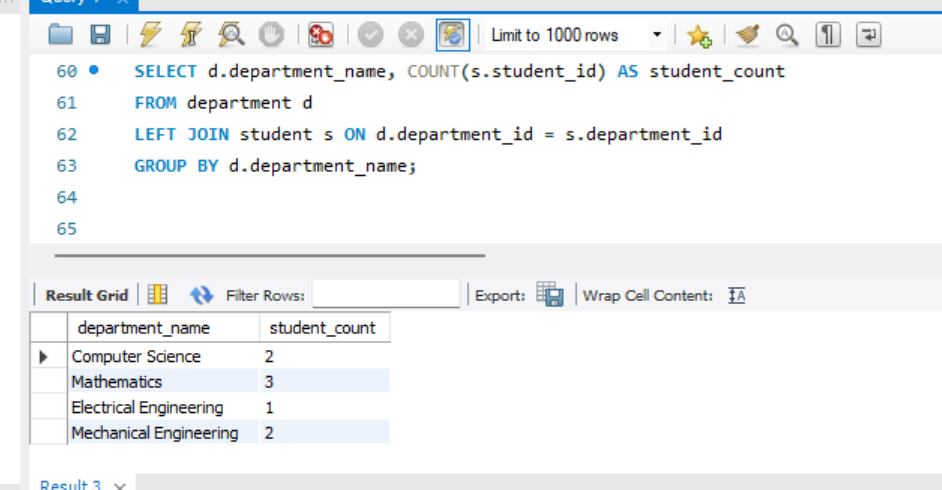
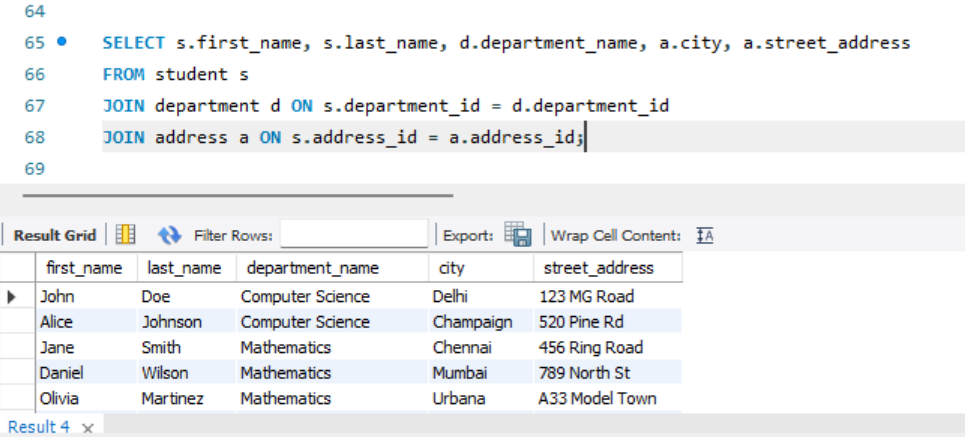
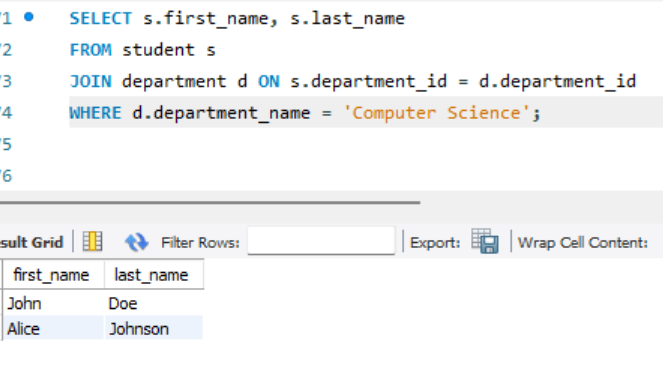
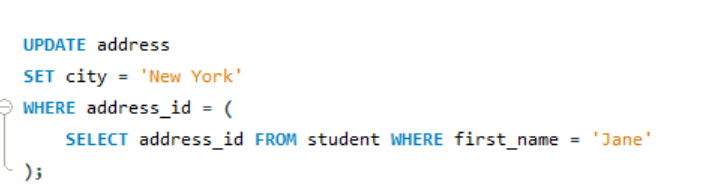
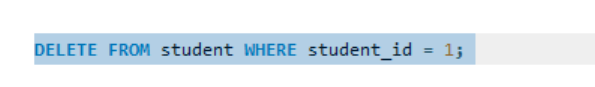
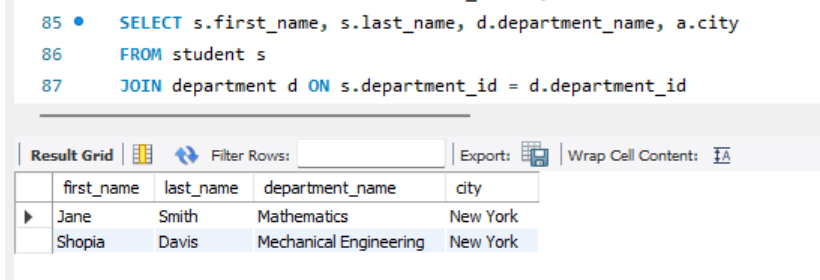
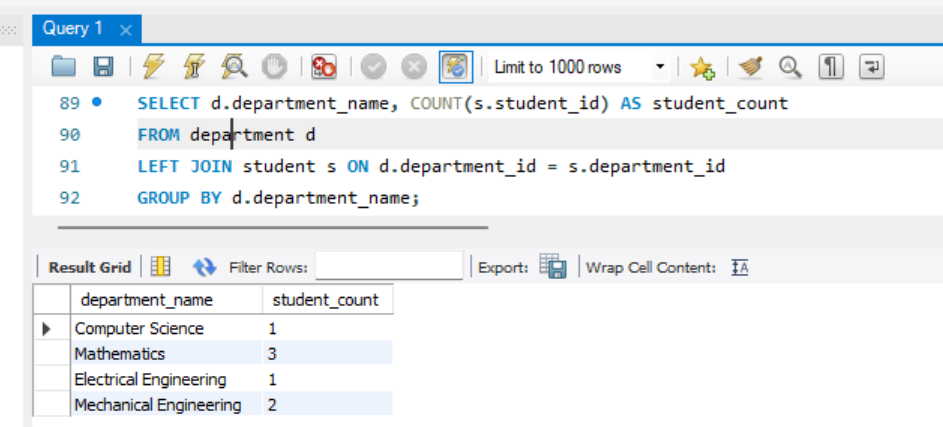
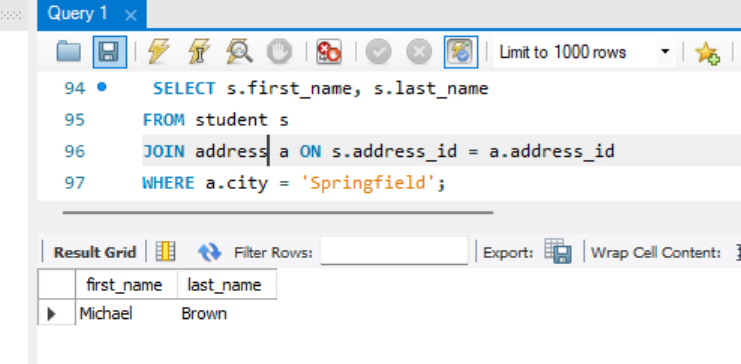
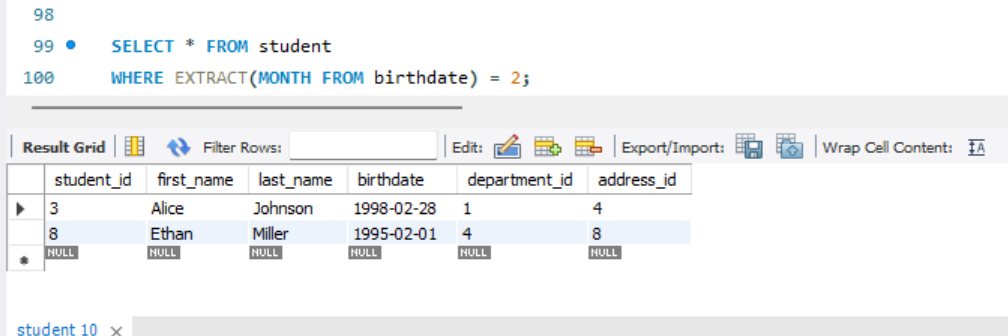
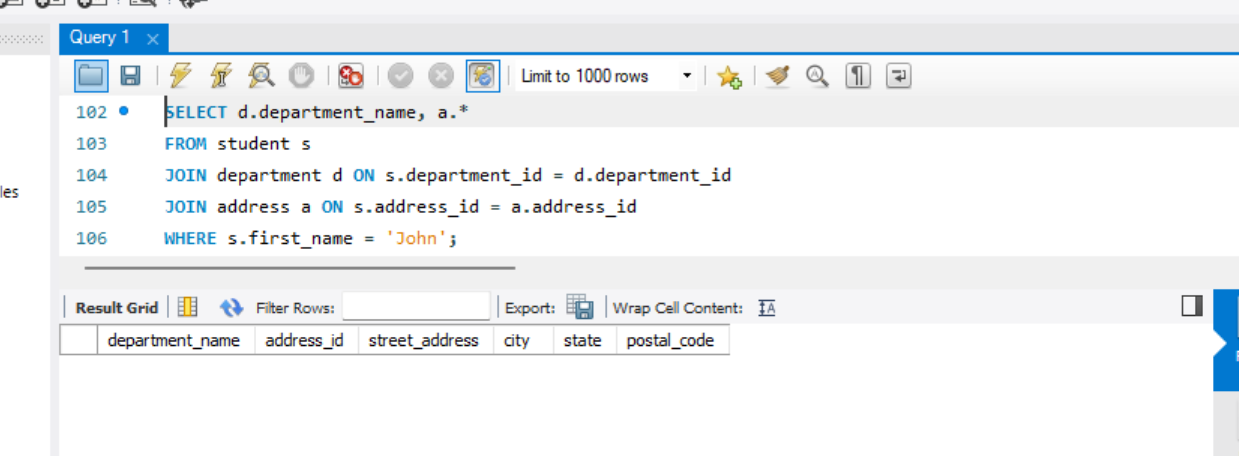
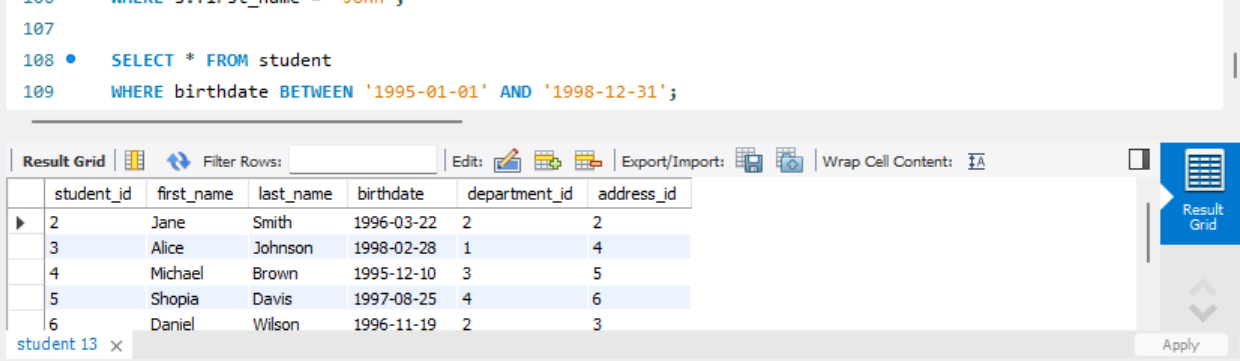
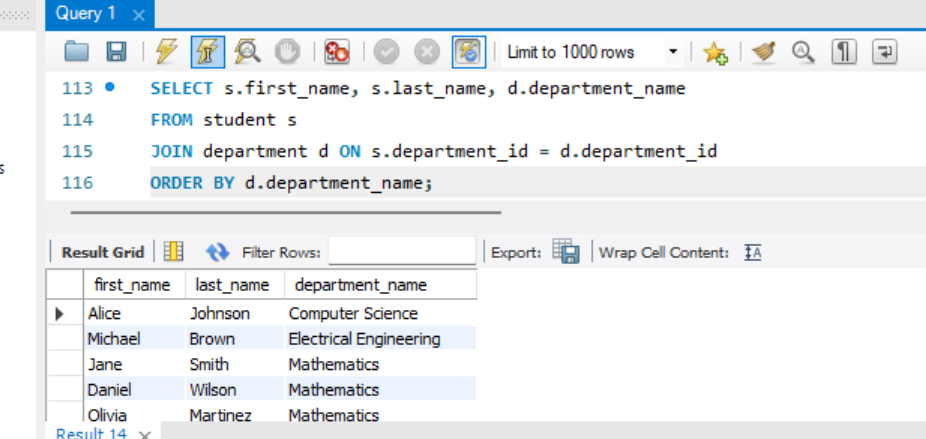
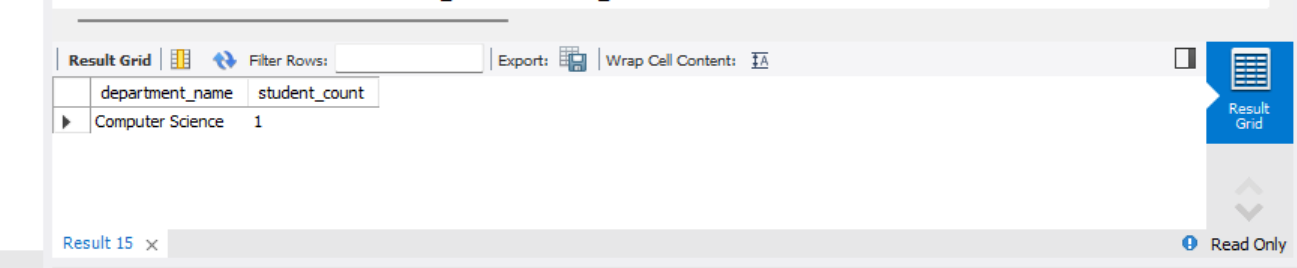
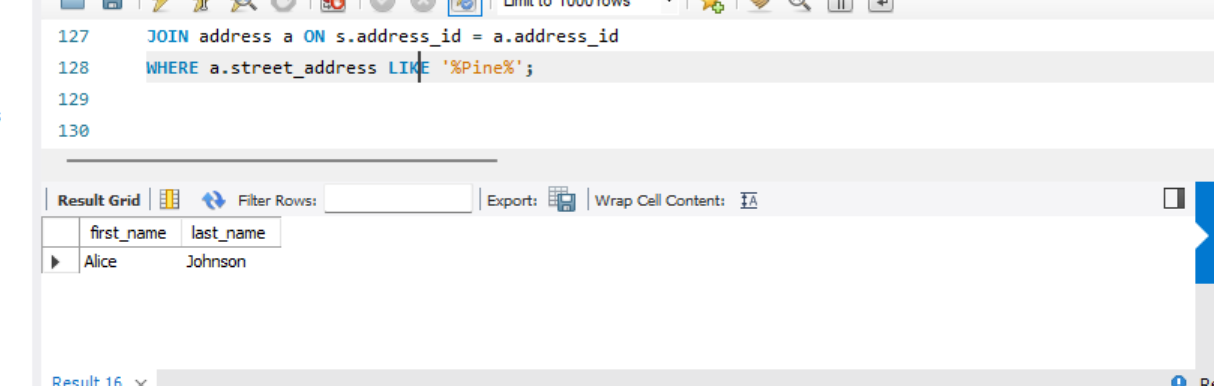
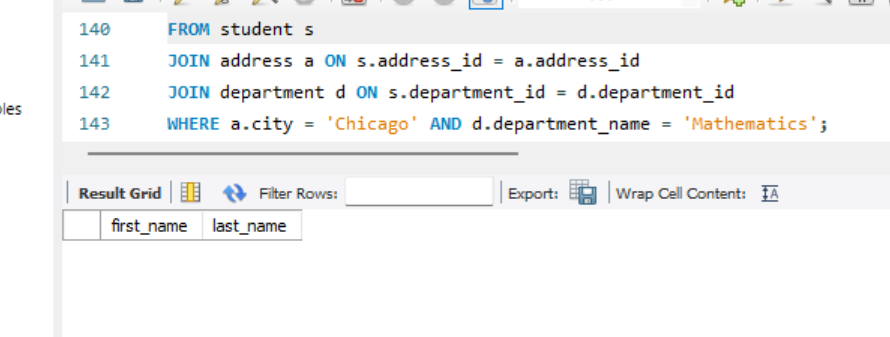
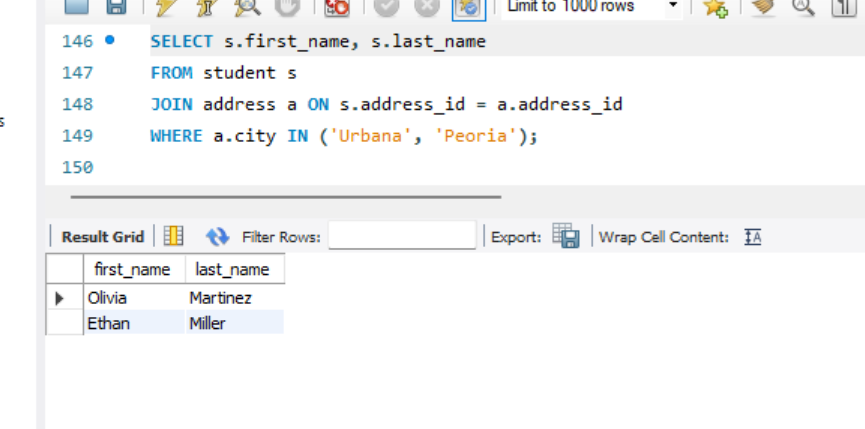
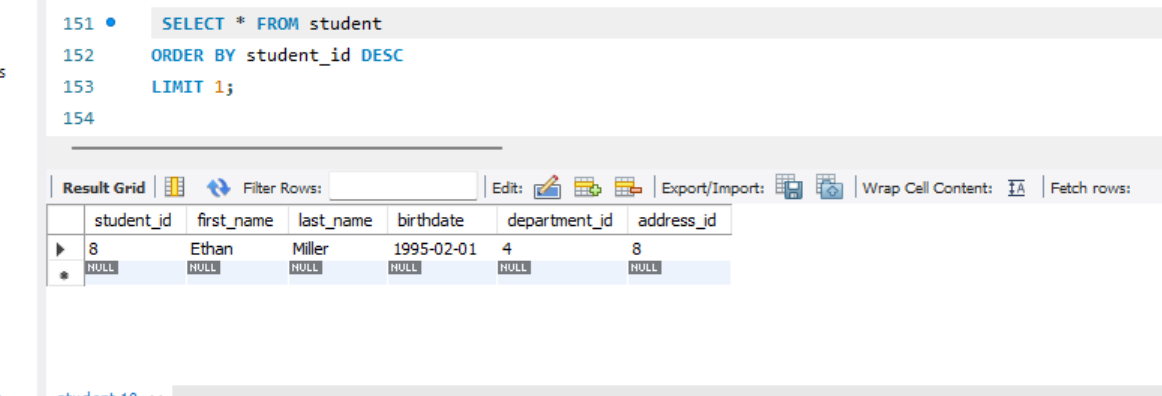
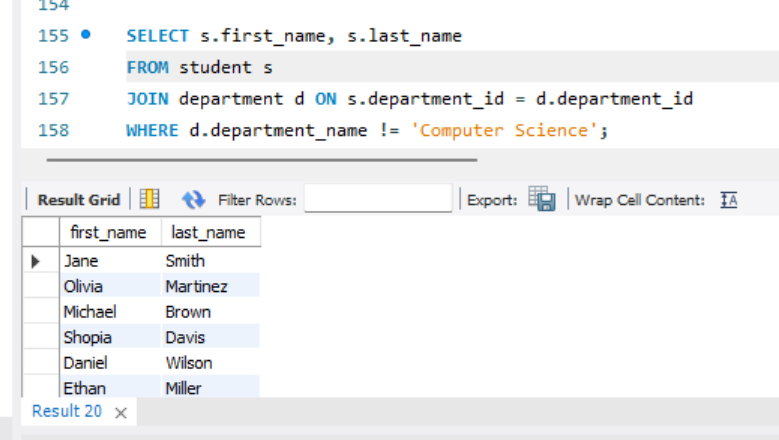
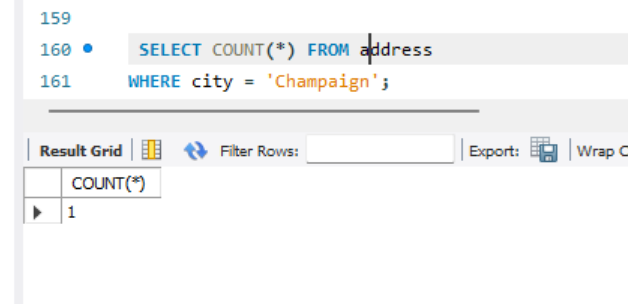
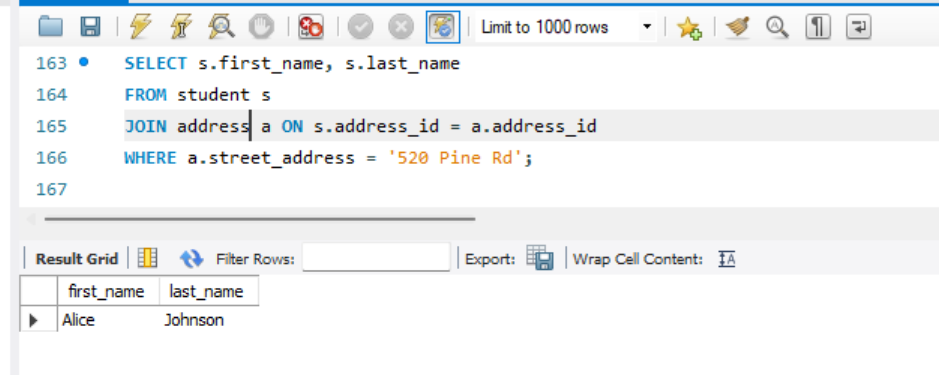
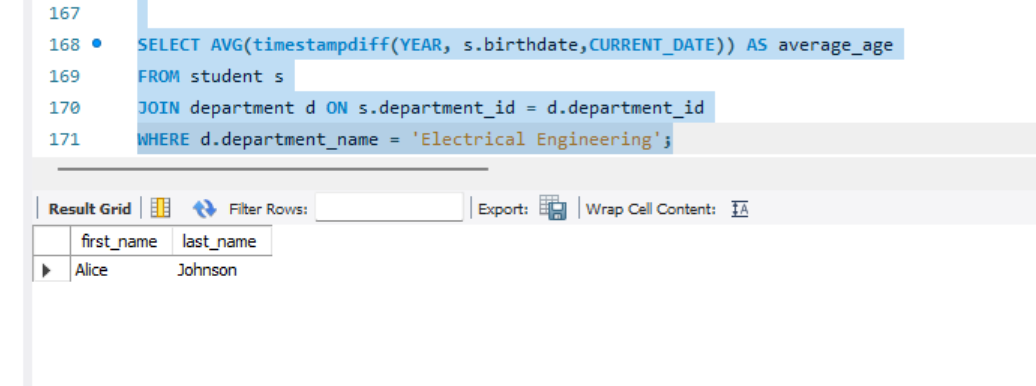
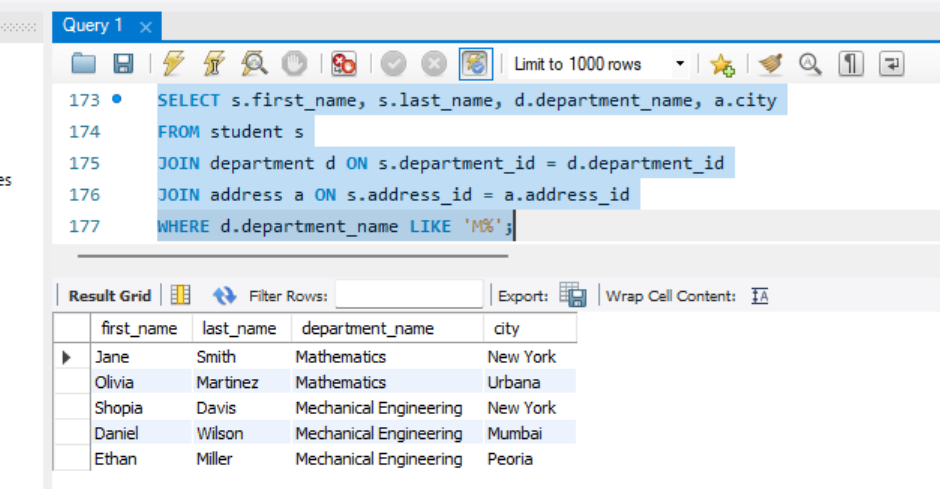
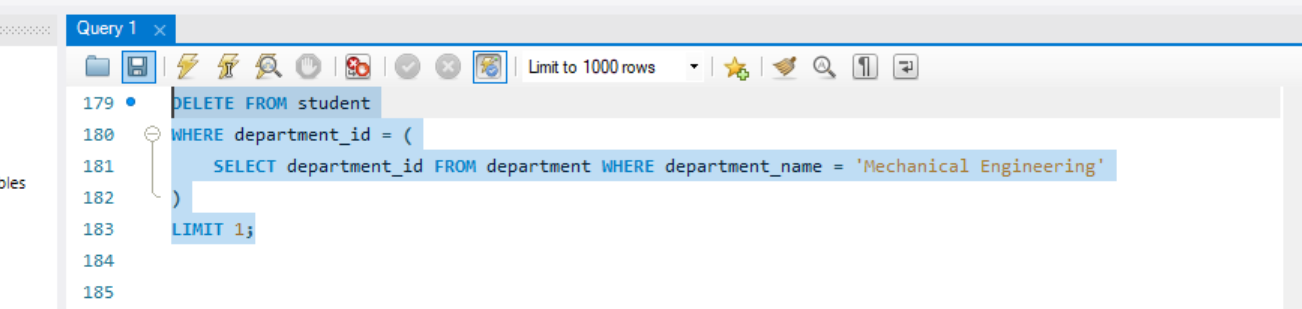
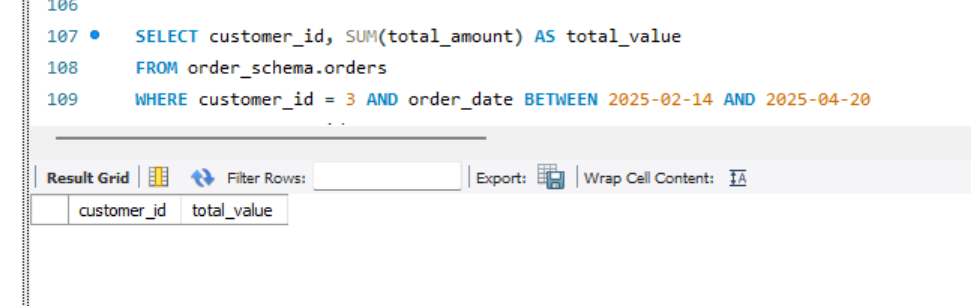
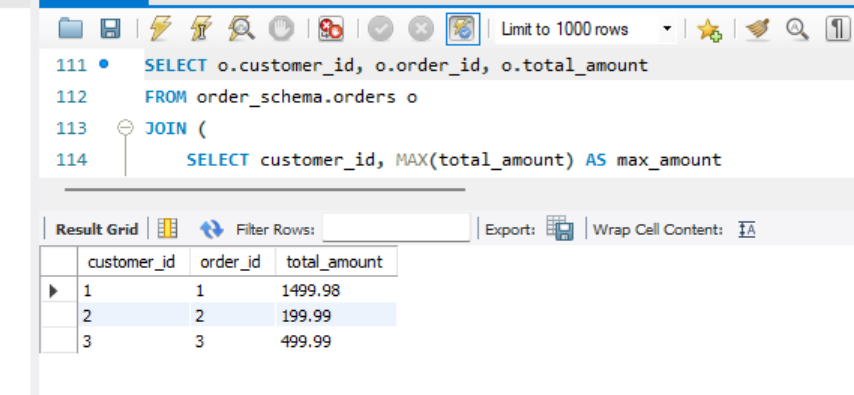
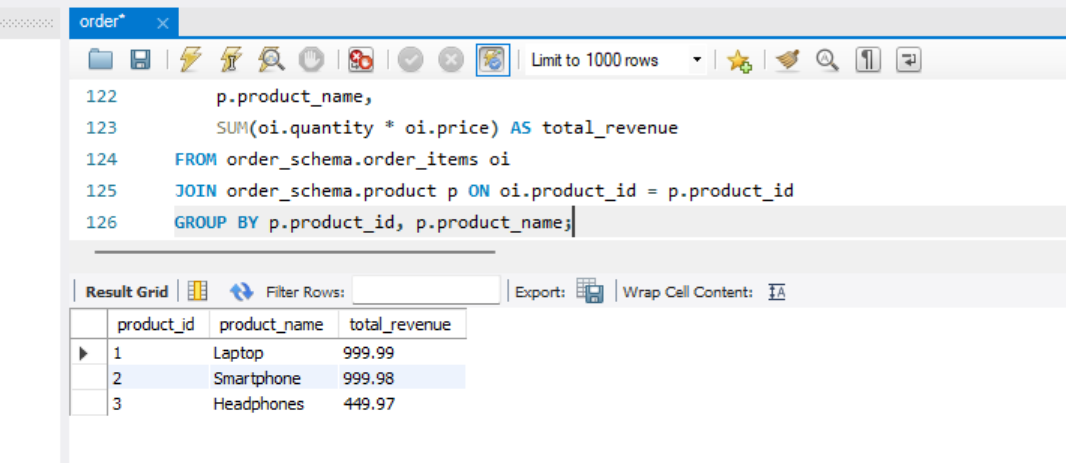
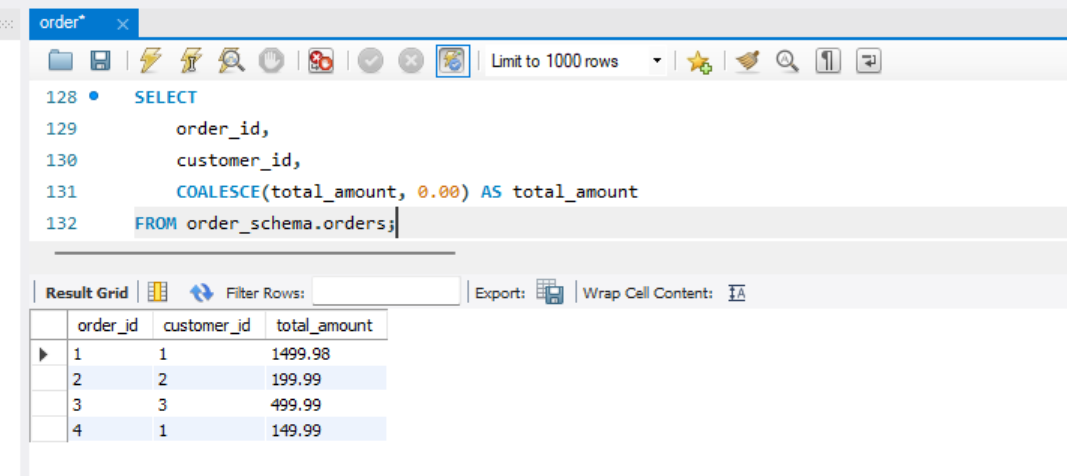
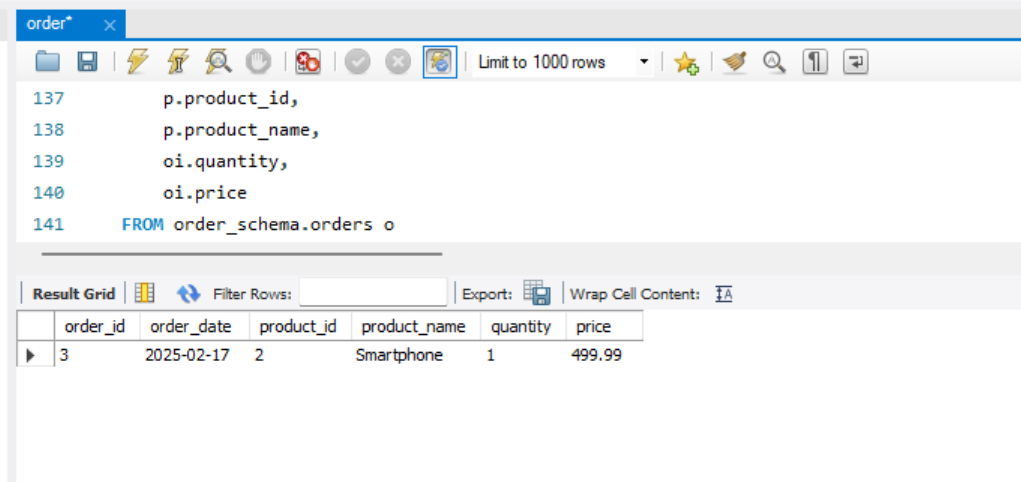
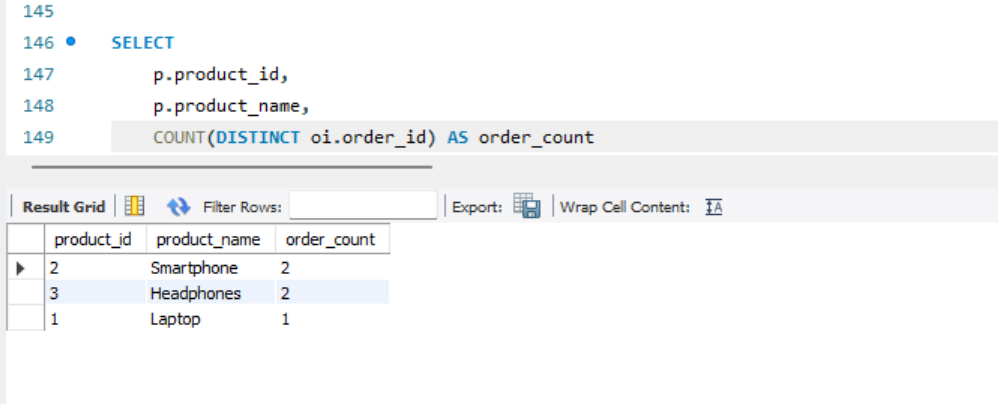
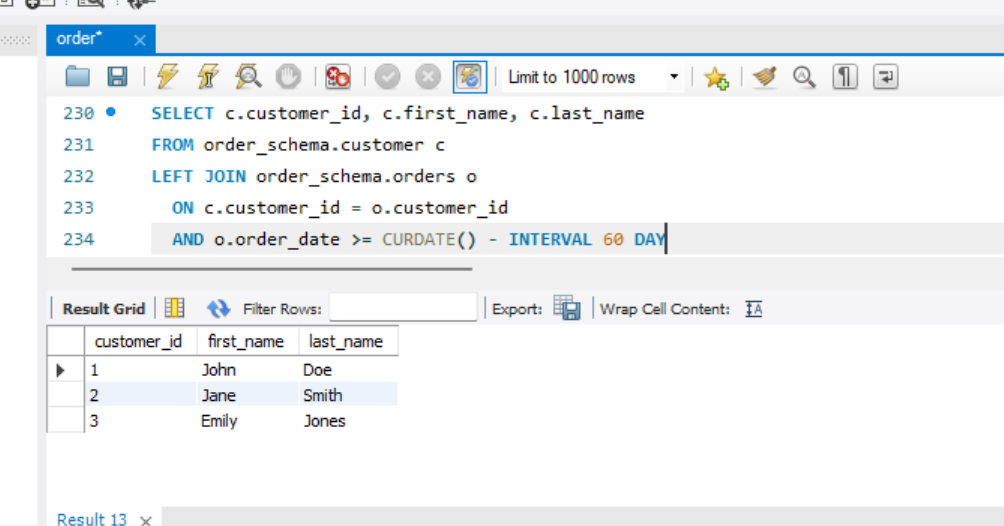
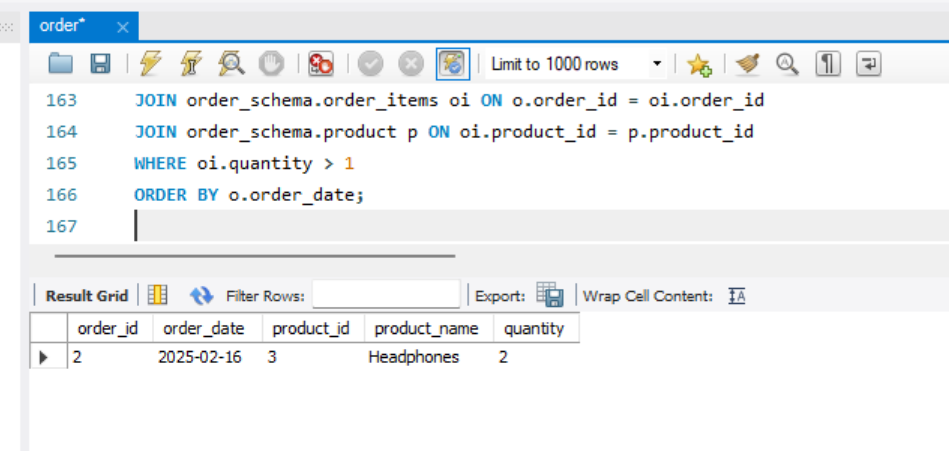
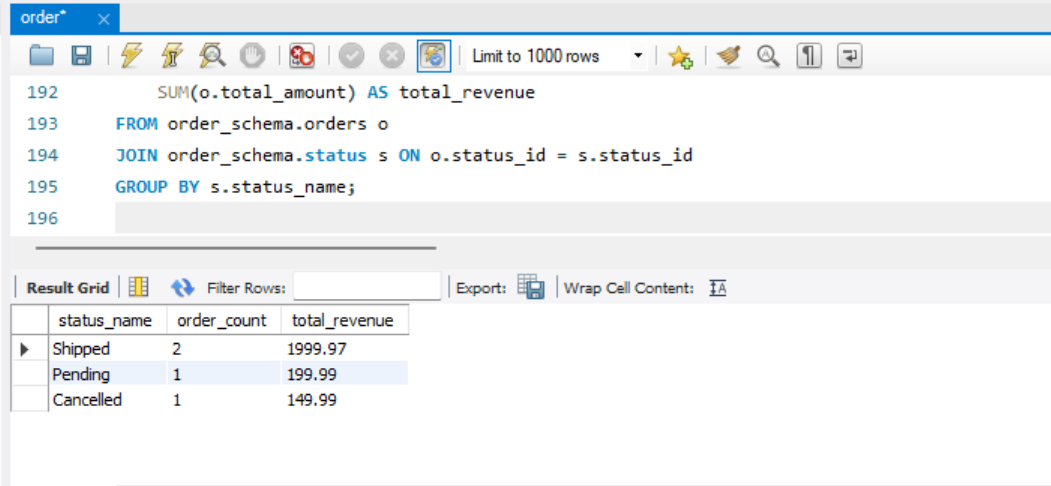
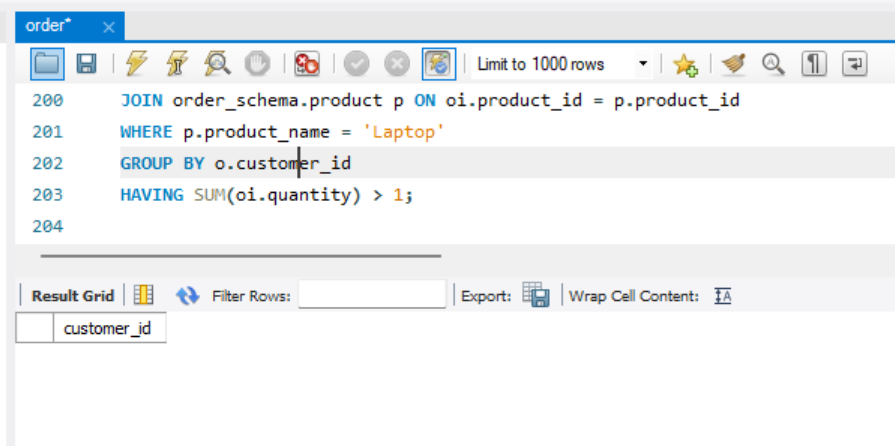
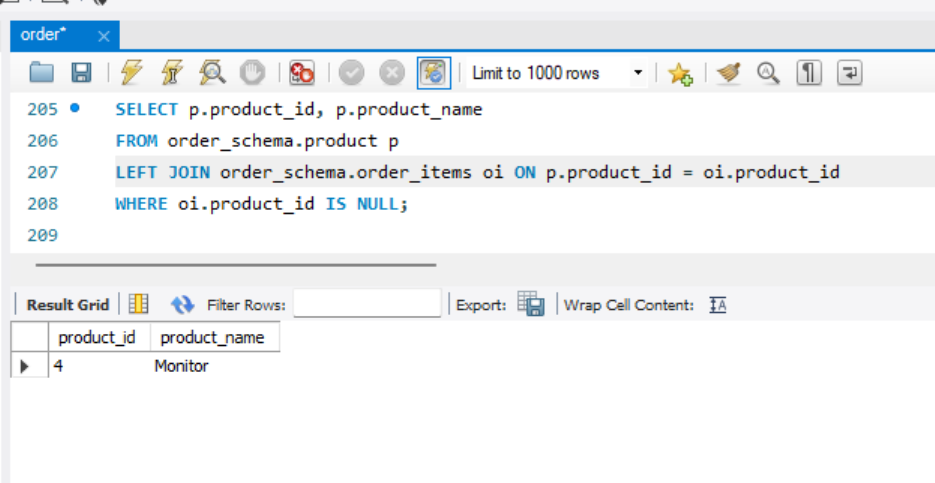
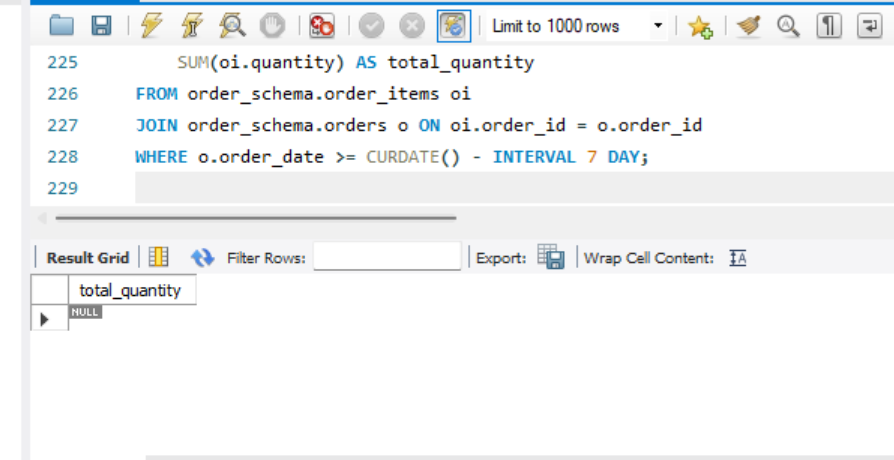
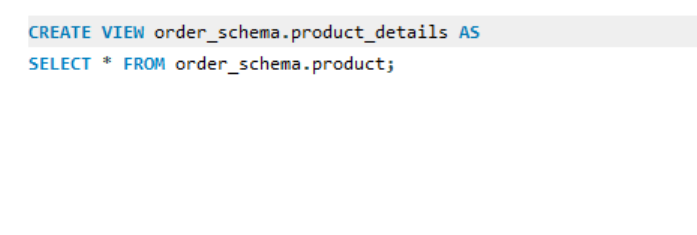
1. Create the tables below in the database. Use foreign keys and primary keys as required.
2. Create a table called as student with the following columns student\_id, first\_name,  last\_name ,birthdate , department\_id  ,address\_id .
3. Create Address table with following columns address\_id , street\_address, city, State, postal\_code
4. Create department table department\_id, department name. Make sure you are using the right data type against all the columns.
5. Use Sample data from [sampledata.txt](https://hs2solutions-my.sharepoint.com/:t:/g/personal/iqbal_sarkar_bounteous_com/ET8VdCEdjTJNoaWCrFPFxfwBq9bpLp9HkT7ycwnltym8qA?e=xTKw8M) to insert data into the database
6. Write a query to find the total number of students.    
   
7. Write a query to find which department john belongs to.   
   
8. List All Departments with Their Number of Students (Including Departments with No Students)   
   
9. Select all students with their department and address.    
   
10. Find all students who are in the 'Computer Science' department    
    
11. Update Jane’s city name to New York.    
    
12. Delete a student from the student table.   
    
13. Select all students with their department and address in New York.    
    
14. Count how many students are in each department    
    
15. Find students who live in 'Springfield'    
    
16. Select students whose birthday falls in February    
    
17. Get the department and address details for a specific student, example john    
    
18. Find all students who are born within 1995 to 1998    
    
19. List all students and their corresponding department names, sorted by department    
    
20. Find the number of students in each department who are living in 'Champaign'    
    
21. Retrieve the names of students who live on 'Pine' street    
    
22. Update the department of a student with student\_id = 6 to 'Mechanical Engineering'    
    
23. Find the student(s) who live in the city 'Chicago' and are in the 'Mathematics' department    
    
24. List all students who have an address in 'Urbana' or 'Peoria'    
    
25. Find the student with the highest student\_id    
    
26. Find all students who are not in the 'Computer Science' department    
    
27. Count the total number of addresses in the 'Champaign' city    
    
28. Find the name of the student who lives at '520 Pine Rd'    
    
29. Get the average age of students in the 'Electrical Engineering' department    
    
30. List the students, their department, and the city where they live, but only for those in departments starting with 'M'    
      
    
31. Delete a student from the 'Mechanical Engineering' department  
    
32. Get the Total Value of Orders for a Given Customer in a Specific Time Period    
    
33. Find the Most Expensive Order by Customer    
    
34. Find the Total Revenue for Each Product Based on Orders    
    
35. Write a query to retrieve the order ID, customer ID, and the total amount of each order. If the total amount is null, display '0.00' instead.   
    
36. Retrieve the Order History of a Specific Customer Along with Product Details    
    

1. Get the Top 5 Products with the Highest Number of Orders.   
   
2. Get the Customers Who Have Not Placed Any Orders in the Last 60 Days  
   
3. List the Orders with Products Ordered More Than Once, Sorted by Order Date    
   
4. Retrieve the Number of Orders and Total Revenue for Each Status  
   
5. Find Customers Who Have Ordered More Than a Specific Product (e.g., "Laptop")    
   
6. Find the Products That Have Never Been Ordered    
   
7. Get the Total Quantity of Products Ordered in the Last 7 Days    
   
8. Create a view named product\_details that includes all columns from the product table.   
   
9. Create a view named order\_summary that includes the order\_id, customer\_id, order\_date, total\_amount, and status\_name (from the status table) for each order.   
   