## NIRAJ NANDISH

## 191IT234 | IT252 | 10/02/2021

# IT252

# Database Management System

## Assignment IV

## Manufacturers and Products schema

1. Compute the average price of all products with manufacturer code equal to 2.

Text

Description automatically generated> SELECT AVG(Price) FROM Products WHERE Manufacturer = 2;

1. Compute the number of products with a price larger than or equal to $180.

Text

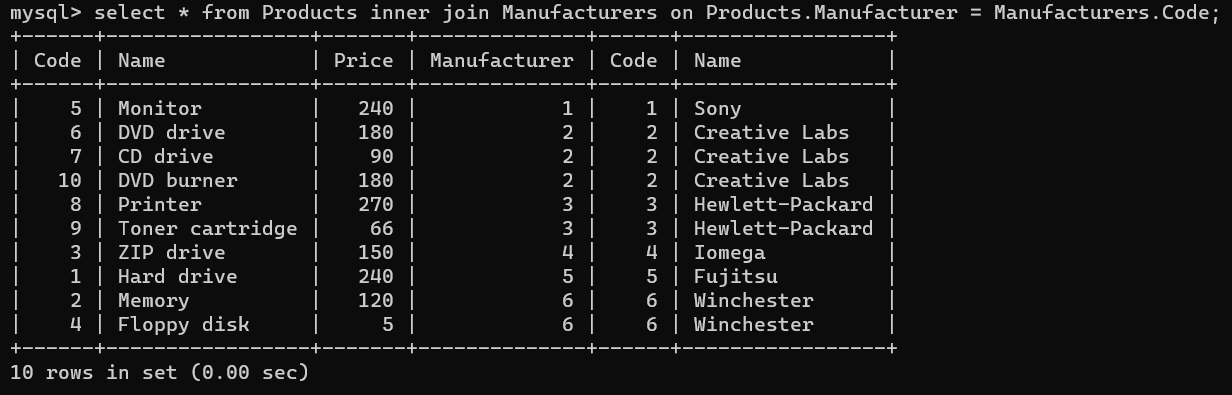
Description automatically generated> SELECT COUNT(\*) AS “Product Count” FROM Products WHERE Price >= 180;

1. Select the name and price of all products with a price larger than or equal to $180, and sort first by price (in descending order), and then by name (in ascending order).

Text

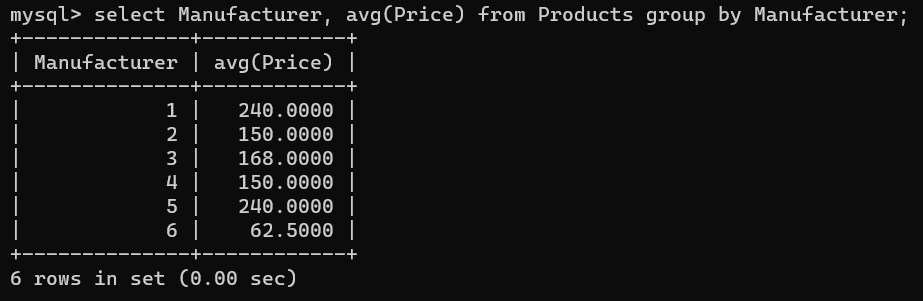
Description automatically generated> SELECT Name, Price FROM Products WHERE Price >= 180 ORDER BY Price DESC, Name ASC;

1. Select all the data from the products, including all the data for each product's manufacturer.

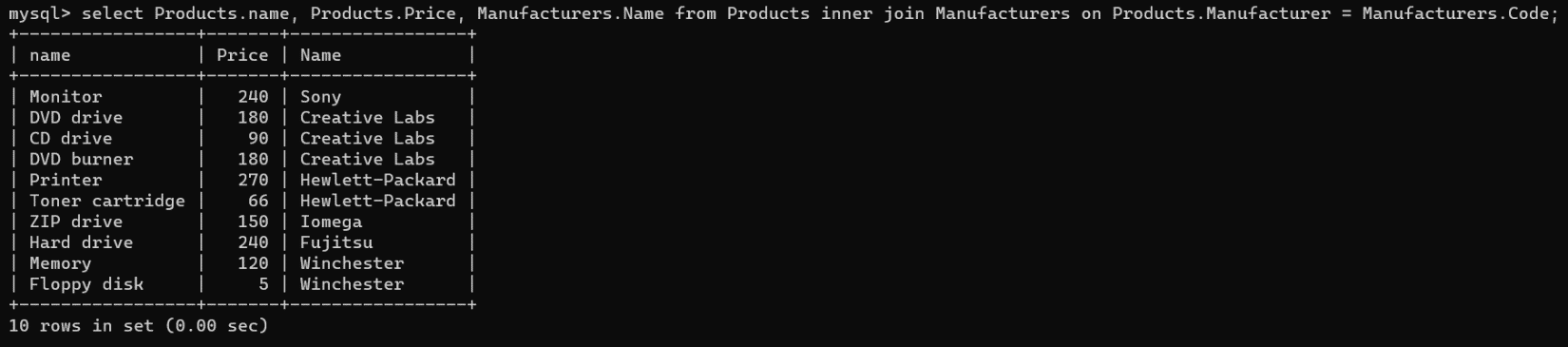
> SELECT \* FROM Products INNER JOIN Manufacturers ON Products.Manufacturer = Manufacturers.Code;

1. Select the average price of each manufacturer's products, showing only the manufacturer's code.

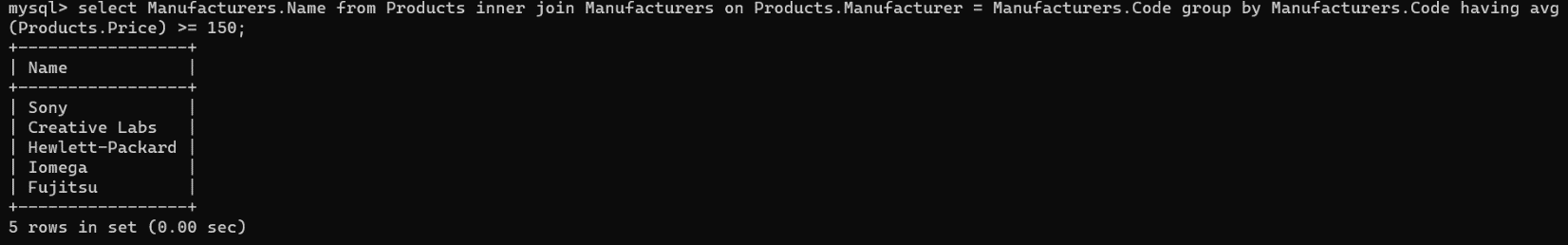
> SELECT Manufacturer, AVG(Price) FROM Products GROUP BY Manufacturer;



1. Select the product name, price, and manufacturer name of all the products.

> SELECT Products.Name, Products.Price, Manufacturers.Name FROM Products INNER JOIN Manufacturers ON Products.Manufacturer = Manufacturers.Code;

1. Select the names of manufacturer whose products have an average price larger than or equal to $150.

> SELECT Manufacturers.Name FROM Products INNER JOIN Manufacturers ON Products.Manufacturer = Manufacturers.Code GROUP BY Manufacturers.Code HAVING AVG(Products.Price) >= 150;

1. Select the name and price of the cheapest product.

Text

Description automatically generated> SELECT Name, Price FROM Products ORDER BY Price ASC LIMIT 1;

1. Select the name of each manufacturer along with the name and price of its most expensive product.

Text

Description automatically generated> SELECT M.Name AS “Manufacturer Name”, P.Name AS “Product Name”, P.Price FROM Products P INNER JOIN Manufacturers M ON M.Code = P.Manufacturer AND P.Price = (SELECT MAX(P.Price) FROM Products P WHERE P.Manufacturer = M.Code);

1. Apply a 10% discount to all products with a price larger than or equal to $120.

> UPDATE Products SET Price = Price \* 0.9 WHERE Price >= 120;

A picture containing text

Description automatically generated

## Departments and Employees schema

1. Select the number of employees in each department (you only need to show the department code and the number of employees).

Text

Description automatically generated> SELECT Department, COUNT(\*) AS “Count of Employees” FROM Employees GROUP BY Department;

1. Select the name and last name of each employee, along with the name and budget of the employee's department.

Text

Description automatically generated> SELECT E.Name, E.LastName, D.Name, D.Budget FROM Employees E INNER JOIN Departments D on E.Department = D.Code;

1. Select the name and last name of employees working for departments with a budget greater than $60,000.

> SELECT E.Name, E.LastName FROM Employees E INNER JOIN Departments D ON E.Department = D.Code WHERE D.Budget > 60000;

Text

Description automatically generated

1. Select the departments with a budget larger than the average budget of all the departments.

> SELECT \* FROM Departments WHERE Budget > (SELECT AVG(Budget) FROM Departments);

Text

Description automatically generated

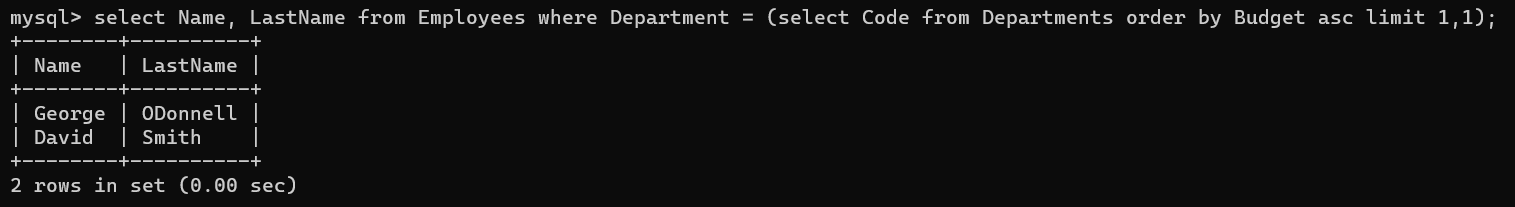
1. Select the names of departments with more than two employees.

> SELECT Name FROM Departments WHERE Code in (SELECT Department FROM Employees GROUP BY Department HAVING COUNT(\*) > 2);

Text

Description automatically generated

1. Select the name and last name of employees working for departments with second lowest budget.

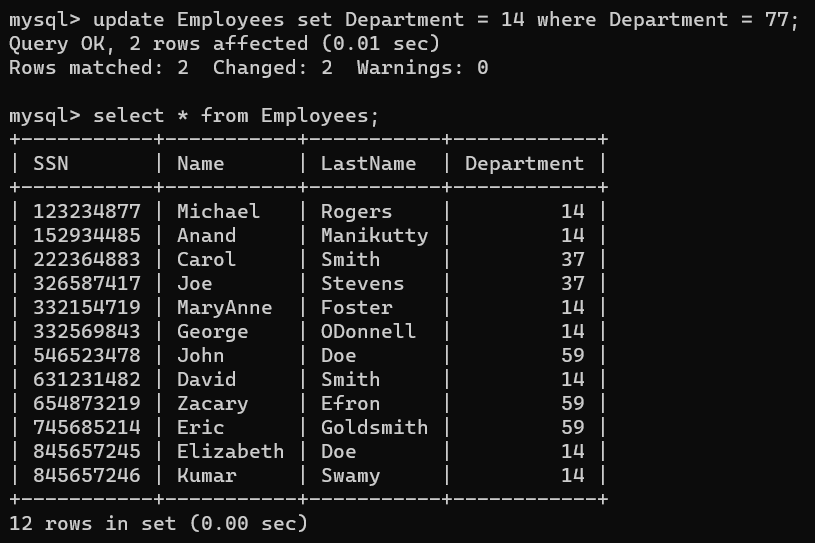
> SELECT Name, LastName FROM Employees WHERE Department = (SELECT Code FROM Departments ORDER BY Budget ASC LIMIT 1,1);

1. Reduce the budget of all departments by 10%.

Text

Description automatically generated> UPDATE Departments SET Budget = Budget \* 0.9;

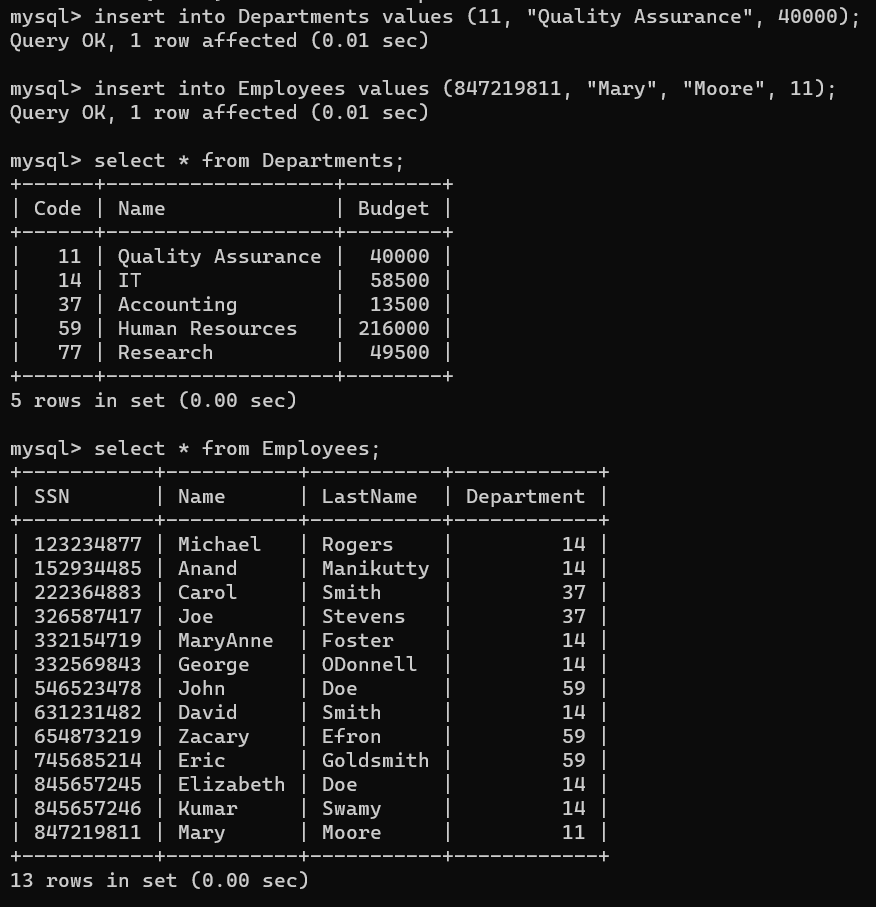
1. Reassign all employees from the Research department (code 77) to the IT department (code 14).

> UPDATE Employees SET Department = 14 WHERE Department = 77;

1. Add a new department called "Quality Assurance", with a budget of $40,000 and departmental code 11. And Add an employee called "Mary Moore" in that department, with SSN 847-21-9811.

> INSERT INTO Departments VALUES (11, “Quality Assurance”, 40000);

> INSERT INTO Employees VALUES (847219811, “Mary”, “Moore”, 11);



1. Delete from the table all employees who work in departments with a budget greater than or equal to $60,000.

> DELETE FROM Employees WHERE Department IN (SELECT Code FROM Departments WHERE Budget >= 60000);

A picture containing text

Description automatically generated