

INF230/Assignment_5/Assignment5.sql

```
1  #Group number: G05_Tues_10
2  #Group members: Peder Ørmen Bukaasen, Mohammed Idris Omar, Bård Tollef Pedersen.
3
4  --@block Task 1
5  SELECT E.EmployeeID,
6         E.FirstName,
7         COUNT(0.OrderID) AS OrderQuantity
8  FROM employees E
9       JOIN orders 0 ON 0.EmployeeID = E.EmployeeID
10 GROUP BY E.EmployeeID
11 ORDER BY OrderQuantity DESC;
12
13 --@block Task 2
14 SELECT DISTINCT C1.*
15 FROM customers C1,
16      customers C2
17 WHERE C1.PostalCode = C2.PostalCode
18       AND C1.CustomerID <> C2.CustomerID
19 ORDER BY C1.Country;
20
21 --@block Task 3
22 SELECT P.ProductName,
23        P.ProductID,
24        COUNT(0.OrderID) AS OrderQuantity
25 FROM products P
26      JOIN orderdetails 0 ON P.ProductID = 0.ProductID
27 GROUP BY P.ProductID
28 HAVING OrderQuantity > 50
29 ORDER BY OrderQuantity DESC;
30
31 --@block Task 4
32 SELECT P.ProductID,
33        P.ProductName,
34        P.UnitsInStock,
35        SUM(0.Quantity) AS Units
36 FROM products P
37      JOIN orderdetails 0 ON P.ProductID = 0.ProductID
38 WHERE P.discontinued = 1
39 GROUP BY P.ProductID
40 HAVING Units > 50;
41
42 --@block Task 5
43 SELECT C.ContactName,
44        C.Country
45 FROM customers C
46      LEFT JOIN suppliers S ON C.Country = S.Country
47 WHERE S.SupplierID IS NULL;
48
49 --@block Task 6
50 SELECT C.ContactName,
51        COUNT(0.OrderID) AS OrderQuantity
52 FROM customers C
53      JOIN orders 0 ON C.CustomerID = 0.CustomerID
54 WHERE YEAR(0.OrderDate) = 1994
```

```

55 GROUP BY C.CustomerID
56 ORDER BY OrderQuantity DESC;
57
58 --@block Task 7
59 SELECT EmployeeID,
60         CONCAT(FirstName, ' ', LastName) AS employee_name,
61         CASE
62             WHEN EmployeeID % 2 = 0 THEN 'Even'
63             ELSE 'Odd'
64         END AS 'EmployeeID status'
65 FROM employees;
66
67 --@block Task 8
68 SELECT EmployeeID,
69         FirstName
70 FROM employees
71 WHERE EmployeeID = ALL(
72     SELECT EmployeeID
73     FROM orders
74     WHERE Freight > 1000
75 );
76
77 --@block Task 9
78 SELECT EmployeeID,
79         CONCAT(UPPER(FirstName), ' ', LOWER(LastName)) AS employee_name
80 FROM employees
81 WHERE EmployeeID = ANY(
82     SELECT EmployeeID
83     FROM orders
84     WHERE Freight > 800
85 );
86
87 --@block Task 10
88 SELECT C.ContactName,
89         AVG(OD.Quantity) AS AvgPurchaseQuantity
90 FROM customers C
91     JOIN orders O ON C.CustomerID = O.CustomerID
92     JOIN orderdetails OD ON O.OrderID = OD.OrderID
93     JOIN products P ON OD.ProductID = P.ProductID
94     JOIN categories CA ON P.CategoryID = CA.CategoryID
95 WHERE CA.CategoryName = 'Beverages'
96 GROUP BY C.CustomerID
97 ORDER BY AvgPurchaseQuantity;
98
99 --@block Creating the table
100 CREATE TABLE employee_work_hours (
101     employee_id INT NOT NULL,
102     login_hours INT NOT NULL,
103     working_date date,
104     PRIMARY KEY (employee_id)
105 );
106
107 --@block Testing the table
108 SELECT *
109 from employee_work_hours;
110
111 --@block Task 11

```

```

112 INSERT INTO employee_work_hours(employee_id, login_hours, working_date)
113 VALUES (1, 4, '2022-02-14'),
114         (2, 8, '2022-02-15'),
115         (3, 2, '2022-02-10'),
116         (4, 7, '2022-02-10'),
117         (8, 5, '2022-02-13'),
118         (9, 4, '2022-02-14');
119
120 --@block Task 12
121 SELECT E.FirstName,
122        EW.login_hours
123 FROM employees E
124      JOIN employee_work_hours EW ON E.EmployeeID = EW.employee_id
125 WHERE EW.login_hours > 2
126        AND (
127            E.TitleOfCourtesy IN ('Mrs.', 'Ms.')
128            OR E.Notes = 'she'
129        );
130
131 --@block Task 13
132 DELETE FROM employee_work_hours
133 WHERE employee_id IN (
134     SELECT EmployeeID
135     FROM employees
136     WHERE BirthDate BETWEEN '1950-01-01' AND '1960-05-27'
137 );
138
139 --@block Deleting the table
140 DROP TABLE employee_work_hours;
141
142 --@block Task 14
143 SELECT orders.CustomerID, -- Select the information we want to show
144        customers.ContactName,
145        COUNT(orderdetails.OrderID) AS total_orders -- Count the number of orders and
146        name the column total_orders
147 FROM customers -- Select the table we want to get information from
148      INNER JOIN orders ON customers.CustomerID = orders.CustomerID -- Join the
149      tables, to get the information from multiple tables
150      INNER JOIN orderdetails ON orders.OrderID = orderdetails.OrderID -- Inner join
151      means that we only get the information that is in both tables
152 GROUP BY orders.CustomerID -- Group the information by the customerID
153 ORDER BY total_orders DESC; -- Sort the information by the total_orders column in
154 descending order
155
156 --@block Task 15
157 SELECT ROUND(POW(EXP(SUM(LOG(UnitPrice))), 1 / COUNT(*)), 2) AS GM1_UnitPrice,
158        ROUND(EXP(SUM(LOG(UnitPrice)) / COUNT(*)), 2) AS GM2_UnitPrice
159 FROM products;

```