

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 ORDER BY Units DESC;
42
43 --@block Task 5
44 SELECT C.ContactName,
45        C.Country
46 FROM customers C
47      LEFT JOIN suppliers S ON C.Country = S.Country
48 WHERE S.SupplierID IS NULL;
49
50 --@block Task 6
51 SELECT C.ContactName,
52        COUNT(0.OrderID) AS OrderQuantity
53 FROM customers C
54      JOIN orders 0 ON C.CustomerID = 0.CustomerID
```

```

55 WHERE YEAR(O.OrderDate) = 1994
56 GROUP BY C.CustomerID
57 ORDER BY OrderQuantity DESC;
58
59 --@block Task 7
60 SELECT EmployeeID,
61         CONCAT(FirstName, ' ', LastName) AS employee_name,
62         CASE
63             WHEN EmployeeID % 2 = 0 THEN 'Even'
64             ELSE 'Odd'
65         END AS 'EmployeeID status'
66 FROM employees;
67
68 --@block Task 8
69 SELECT EmployeeID,
70         FirstName
71 FROM employees
72 WHERE EmployeeID = ALL(
73     SELECT EmployeeID
74     FROM orders
75     WHERE Freight > 1000
76 );
77
78 --@block Task 9
79 SELECT EmployeeID,
80         CONCAT(UPPER(FirstName), ' ', LOWER(LastName)) AS employee_name
81 FROM employees
82 WHERE EmployeeID = ANY(
83     SELECT EmployeeID
84     FROM orders
85     WHERE Freight > 800
86 );
87
88 --@block Task 10
89 SELECT C.ContactName,
90         AVG(OD.Quantity) AS AvgPurchaseQuantity
91 FROM customers C
92     JOIN orders O ON C.CustomerID = O.CustomerID
93     JOIN orderdetails OD ON O.OrderID = OD.OrderID
94     JOIN products P ON OD.ProductID = P.ProductID
95     JOIN categories CA ON P.CategoryID = CA.CategoryID
96 WHERE CA.CategoryName = 'Beverages'
97 GROUP BY C.CustomerID
98 ORDER BY AvgPurchaseQuantity DESC;
99
100 --@block Creating the table
101 CREATE TABLE employee_work_hours (
102     employee_id INT NOT NULL,
103     login_hours INT NOT NULL,
104     working_date date,
105     PRIMARY KEY (employee_id)
106 );
107
108 --@block Testing the table
109 SELECT *
110 from employee_work_hours;
111

```

```

112 --@block Task 11
113 INSERT INTO employee_work_hours(employee_id, login_hours, working_date)
114 VALUES (1, 4, '2022-02-14'),
115         (2, 8, '2022-02-15'),
116         (3, 2, '2022-02-10'),
117         (4, 7, '2022-02-10'),
118         (8, 5, '2022-02-13'),
119         (9, 4, '2022-02-14');
120
121 --@block Task 12
122 SELECT E.FirstName,
123        EW.login_hours
124 FROM employees E
125      JOIN employee_work_hours EW ON E.EmployeeID = EW.employee_id
126 WHERE EW.login_hours > 2
127        AND (
128            E.TitleOfCourtesy IN ('Mrs.', 'Ms.')
129            OR E.Notes = 'she'
130        );
131
132 --@block Task 13
133 DELETE FROM employee_work_hours
134 WHERE employee_id IN (
135     SELECT EmployeeID
136     FROM employees
137     WHERE BirthDate BETWEEN '1950-01-01' AND '1960-05-27'
138 );
139
140 --@block Deleting the table
141 DROP TABLE employee_work_hours;
142
143 --@block Task 14
144 SELECT orders.CustomerID, -- Select the information we want to show
145        customers.ContactName,
146        COUNT(orderdetails.OrderID) AS total_orders -- Count the number of orders and
147        name the column total_orders
148 FROM customers -- Select the table we want to get information from
149      INNER JOIN orders ON customers.CustomerID = orders.CustomerID -- Join the
150      tables, to get the information from multiple tables
151      INNER JOIN orderdetails ON orders.OrderID = orderdetails.OrderID -- Inner join
152      means that we only get the information that is in both tables
153 GROUP BY orders.CustomerID -- Group the information by the customerID
154 ORDER BY total_orders DESC; -- Sort the information by the total_orders column in
155 descending order
156
157 --@block Task 15
158 SELECT ROUND(POW(EXP(SUM(LOG(UnitPrice))), 1 / COUNT(*)), 2) AS GM1_UnitPrice,
159        ROUND(EXP(SUM(LOG(UnitPrice)) / COUNT(*)), 2) AS GM2_UnitPrice
160 FROM products;

```