EASTERN INTERNATIONAL UNIVERSITY

SCHOOL OF COMPUTING

AND INFORMATION TECHNOLOGY

♑🕮♒

Practice Assignment 5

Practice Assignment – Quarter 2, 2024-2025

Course Name: Database

Course Code: CSE 301

Student's Full Name:

Student ID:

***Instruction*:**

** Students are allowed to write their answers (like SQL queries, Screen shot of outputs, etc.) in word file (Answer sheet) provided by instructor. After finishing the assignment, students must convert the word file (Answer sheet) into a PDF file. Finally, students upload the file in Moodle.*

I. Using the database 'SaleManagement' with following schema below:

Clients(**Client_Number**, Client_Name, Address, City, Pincode, Province, Amount_Paid, Amount_Due)

Product(**Product_Number**, Product_Name, Quantity_On_Hand, Quantity_Sell, Sell_Price, Cost_Price)

Salesman (**Salesman_Number**, Salesman _Name, Address, City, Pincode, Province, Salary, Sales_Target, Target_Achieve, Phone)

Salesorder(**Order_Number**, Order_Date**, Client_Number**, **Salesman_Number**, Delivery_Status, Delivery_Date, Order_Status)

Salesorderdetails(**Order_Number, Product_Number**, Order_Quantity)

II. Insert more values data below:

Salesman

('S007','Quang','Chanh My','Da Lat',700032,'Lam Dong',25000,90,95,'0900853487')
('S008','Hoa','Hoa Phu','Thu Dau Mot',700051,'Binh Duong',13500,50,75,'0998213659')

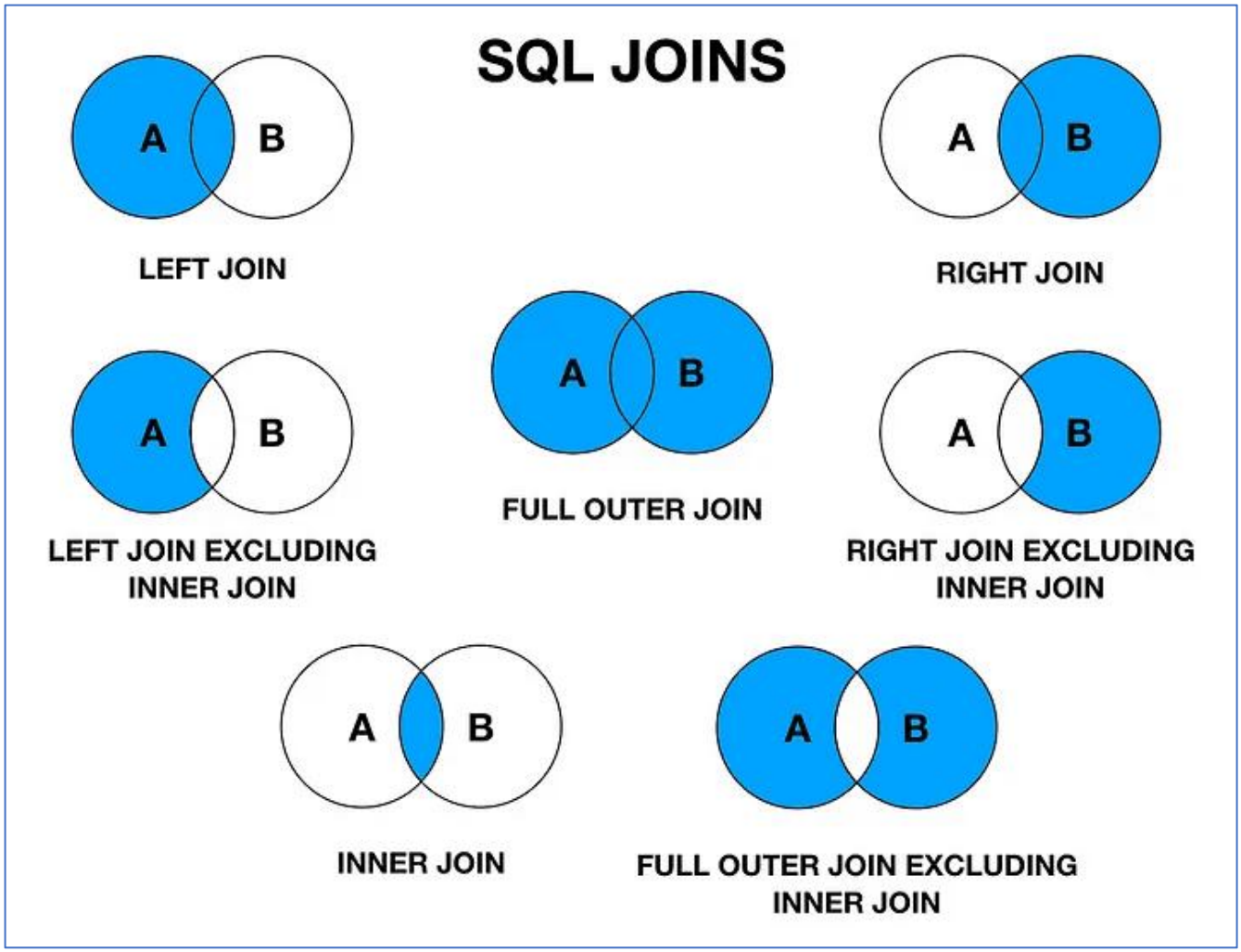
Salesorder

('O20015','2022-05-12','C108','S007','On Way', '2022-05-15','Successful')
('O20016','2022-05-16','C109','S008','Ready to Ship',null,'In Process')

Salesorderdetails

('O20015','P1008',15),
('O20015','P1007',10),
('O20016','P1007',20);
('O20016','P1003',5);

III. Using Joining table to combine rows from more than one table.



Read more at: https://tinyurl.com/SQL-AllJoinTypes

Supported Types of Joins in MySQL

INNER JOIN: Returns records that have matching values in both tables

LEFT JOIN: Returns all records from the left table, and the matched records from the right table

RIGHT JOIN: Returns all records from the right table, and the matched records from the left table

CROSS JOIN: Returns all records from both tables

1. Display the clients (name) who lives in same city.

```
21
22      -- 1.Display the clients (name) who lives in same city.
23      SELECT c1.Client_Name, c1.City
24      FROM Clients c1
25      WHERE c1.City IN (
26          SELECT City
27          FROM Clients
28          GROUP BY City
29          HAVING COUNT(*) > 1
```

esult Grid | Filter Rows: | Export: | Wrap Cell Content:

Client_Name	City
Mai Xuan	Dai An
Le Xuan	Thu Dau Mot
Trinh Huu	Da Lat
Tran Tuan	Thu Dau Mot
Ho Nhu	Hanoi
Tran Hai	Ho Chi Minh
Nguyen Thanh	Dai An
Nguyen Sy	Da Lat
Duong Thanh	Ho Chi Minh
Tran Minh	Hanoi

2. Display city, the client names and salesman names who are lives in “Thu Dau Mot” city.

```
30      );
31
32      -- 2.Display city, the client names and salesman names who are lives in “Thu Dau Mot” city.
33      SELECT c.City, c.Client_Name, s.Salesman_Name
34      FROM Clients c, Salesman s
35      WHERE c.City = 'Thu Dau Mot'
36      AND s.City = 'Thu Dau Mot';
37
38      -- 3.Display client name, client number, order number, salesman number, and product number for each or
39      SELECT c.Client_Name, c.Client_Number, so.Order_Number, so.Salesman_Number, sod.Product_Number
40      FROM Clients c
41      JOIN SalesOrder so ON c.Client_Number = so.Client_Number
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

City	Client_Name	Salesman_Name
Thu Dau Mot	Le Xuan	Hoa
Thu Dau Mot	Le Xuan	Deb
Thu Dau Mot	Le Xuan	Khoa
Thu Dau Mot	Tran Tuan	Hoa
Thu Dau Mot	Tran Tuan	Deb
Thu Dau Mot	Tran Tuan	Khoa

3. Display client name, client number, order number, salesman number, and product number for each order.

```
38      -- 3.Display client name, client number, order number, salesman number, and product number for each order
39 •    SELECT c.Client_Name, c.Client_Number, so.Order_Number, so.Salesman_Number, sod.Product_Number
40      FROM Clients c
41      JOIN SalesOrder so ON c.Client_Number = so.Client_Number
42      JOIN SalesOrderDetails sod ON so.Order_Number =sod.Order_Number;
43
44      -- 4.Find each order (client_number, client_name, order_number) placed by each client.
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

Client_Name	Client_Number	Order_Number	Salesman_Number	Product_Number
Mai Xuan	C101	O20001	S003	P1001
Mai Xuan	C101	O20001	S003	P1002
Mai Xuan	C101	O20005	S003	P1001
Mai Xuan	C101	O20005	S003	P1008
Mai Xuan	C101	O20005	S003	P1002
Mai Xuan	C101	O20009	S004	P1008
Le Xuan	C102	O20002	S003	P1007
Le Xuan	C102	O20008	S006	P1004
Trinh Huu	C103	O20003	S002	P1003
Tran Tuan	C104	O20004	S003	P1004
Ho Nhu	C105	O20006	S005	P1002
Tran Hai	C106	O20007	S001	P1005
Tran Hai	C106	O20010	S006	P1006
Tran Hai	C106	O20010	S006	P1001
Nguyen Th...	C107	O20011	S005	P1007
Nguyen Sy	C108	O20012	S004	P1005
Nguyen Sy	C108	O20012	S004	P1001
Nguyen Sy	C108	O20015	S007	P1008
Nguyen Sy	C108	O20015	S007	P1007
Duong Thanh	C109	O20013	S001	P1006
Duong Thanh	C109	O20016	S008	P1007
Duong Thanh	C109	O20016	S008	P1003
Tran Minh	C110	O20014	S001	P1002

Result Grid
Form Editor
Field Types
Query Stats
Execution Plan

4. Find each order (client_number, client_name, order_number) placed by each client.

```
43
44      -- 4.Find each order (client_number, client_name, order_number) placed by each client.
45      SELECT c.Client_Number, c.Client_Name, so.Order_Number
46      FROM Clients c
47      JOIN SalesOrder so ON c.Client_Number = so.Client_Number;
48
49      -- 5.Display the details of clients (client_number, client_name) and the number of orders which is paid by
50      SELECT c.Client_Number, c.Client_Name, COUNT(so.Order_Number) AS NumberOfOrders
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

Client_Number	Client_Name	Order_Number
C101	Mai Xuan	O20001
C101	Mai Xuan	O20005
C101	Mai Xuan	O20009
C102	Le Xuan	O20002
C102	Le Xuan	O20008
C103	Trinh Huu	O20003
C104	Tran Tuan	O20004
C105	Ho Nhu	O20006
C106	Tran Hai	O20007
C106	Tran Hai	O20010
C107	Nguyen Th...	O20011
C108	Nguyen Sy	O20012
C108	Nguyen Sy	O20015
C109	Duong Thanh	O20013
C109	Duong Thanh	O20016
C110	Tran Minh	O20014

Result Grid | Form Editor | Field Types | Query Stats | Execution Plan

5. Display the details of clients (client_number, client_name) and the number of orders which is paid by them.

```
49      -- 5.Display the details of clients (client_number, client_name) and the number of orders which is paid by
50      SELECT c.Client_Number, c.Client_Name, COUNT(so.Order_Number) AS NumberOfOrders
51      FROM Clients c
52      JOIN SalesOrder so ON c.Client_Number = so.Client_Number
53      WHERE so.Order_Status = 'Successful'
54      GROUP BY c.Client_Number, c.Client_Name;
55
56      -- 6.Display the details of clients (client_number, client_name) who have paid for more than 2 orders.
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

Client_Number	Client_Name	NumberOfOrders
C101	Mai Xuan	2
C103	Trinh Huu	1
C104	Tran Tuan	1
C106	Tran Hai	2
C108	Nguyen Sy	2
C110	Tran Minh	1

Result Grid | Form Editor

6. Display the details of clients (client_number, client_name) who have paid for more than 2 orders.

```
55
56      -- 6.Display the details of clients (client_number, client_name) who have paid for more than 2 orders.
57 •    SELECT c.Client_Number, c.Client_Name
58      FROM Clients c
59      JOIN SalesOrder so ON c.Client_Number = so.Client_Number
60      WHERE so.Order_Status = 'Successful'
61      GROUP BY c.Client_Number, c.Client_Name
62      HAVING COUNT(so.Order_Number) > 2;
63
64      -- 7.Display details of clients who have paid for more than 1 order in descending order of client_number
65 •    SELECT c.Client_Number, c.Client_Name, COUNT(so.Order_Number) AS OrderCount
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

Client_Number	Client_Name

7. Display details of clients who have paid for more than 1 order in descending order of client_number.

Setup

```
63
64      -- 7.Display details of clients who have paid for more than 1 order in descending order of client_number
65 •    SELECT c.Client_Number, c.Client_Name, COUNT(so.Order_Number) AS OrderCount
66      FROM Clients c
67      JOIN SalesOrder so ON c.Client_Number = so.Client_Number
68      WHERE so.Order_Status = 'Successful'
69      GROUP BY c.Client_Number, c.Client_Name
70      HAVING COUNT(so.Order_Number) > 1
71      ORDER BY c.Client_Number DESC;
72
73      -- 8.Find the salesman names who sells more than 20 products.
74 •    SELECT s.Salesman_Name, SUM(sod.Order_Quantity) AS TotalSold
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

Client_Number	Client_Name	OrderCount
C108	Nguyen Sy	2
C106	Tran Hai	2
C101	Mai Xuan	2

Result Grid

8. Find the salesman names who sells more than 20 products.

```
72
73      -- 8.Find the salesman names who sells more than 20 products.
74 •    SELECT s.Salesman_Name, SUM(sod.Order_Quantity) AS TotalSold
75      FROM Salesman s
76      JOIN SalesOrder so ON s.Salesman_Number = so.Salesman_Number
77      JOIN SalesOrderDetails sod ON so.Order_Number = sod.Order_Number
78      GROUP BY s.Salesman_Number, s.Salesman_Name
79      HAVING SUM(sod.Order_Quantity) > 20;
80
81      -- 9.Display the client information (client_number, client_name) a
82 •    SELECT c.Client_Number, c.Client_Name, so.Order_Number
83      FROM Clients c
84      JOIN SalesOrder so ON c.Client_Number = so.Client_Number
85      WHERE so.Order_Status = 'Cancelled';
86
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

Salesman_Name	TotalSold
Huu	36
Khoa	59
Tin	28
Quang	25
Hoa	25

9. Display the client information (client_number, client_name) and order number of those clients who have order status is cancelled.

```
80
81      -- 9.Display the client information (client_number, client_name) and order number of those clients wh
82 •    SELECT c.Client_Number, c.Client_Name, so.Order_Number
83      FROM Clients c
84      JOIN SalesOrder so ON c.Client_Number = so.Client_Number
85      WHERE so.Order_Status = 'Cancelled';
86
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

Client_Number	Client_Name	Order_Number
C102	Le Xuan	O20002
C101	Mai Xuan	O20005
C107	Nguyen Thanh	O20011

10. Display client name, client number of clients C101 and count the number of orders which were received

"successful".

```
87    -- 10.Display client name, client number of clients C101 and count the number of orders which were recei
88    SELECT c.Client_Number, c.Client_Name, COUNT(so.Order_Number) AS SuccessfulOrders
89    FROM Clients c
90    JOIN SalesOrder so ON c.Client_Number = so.Client_Number
91    WHERE c.Client_Number = 'C101'
92      AND so.Order_Status = 'Successful'
93    GROUP BY c.Client_Number, c.Client_Name;
94
95    -- 11.Count the number of clients orders placed for each product.
96    SELECT sod.Product_Number, COUNT(so.Order_Number) AS OrderCount
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

Client_Number	Client_Name	SuccessfulOrders
C101	Mai Xuan	2

11. Count the number of clients orders placed for each product.

```
95    -- 11.Count the number of clients orders placed for each product.
96    SELECT sod.Product_Number, COUNT(so.Order_Number) AS OrderCount
97    FROM SalesOrderDetails sod
98    JOIN SalesOrder so ON sod.Order_Number = so.Order_Number
99    GROUP BY sod.Product_Number;
100
101   -- 12.Find product numbers that were ordered by more than two clients then order in d
102   SELECT sod.Product_Number
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

Product_Number	OrderCount
P1001	4
P1002	4
P1008	3
P1007	4
P1004	2
P1003	2
P1005	2
P1006	2

12. Find product numbers that were ordered by more than two clients then order in descending by product number.

```
100
101     -- 12.Find product numbers that were ordered by more than two clients then order in descending by produc
102     SELECT sod.Product_Number
103     FROM SalesOrderDetails sod
104     JOIN SalesOrder so ON sod.Order_Number = so.Order_Number
105     GROUP BY sod.Product_Number
106     HAVING COUNT(DISTINCT so.Client_Number) > 2
107     ORDER BY sod.Product_Number DESC;
108
109
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

Product_Number
P1007
P1002
P1001

IV. Using nested query with operator (IN, EXISTS, ANY and ALL)

13. Find the salesman's names who is getting the second highest salary.

```
110     -- IV.Using nested query with operator (IN, EXISTS, ANY and ALL)
111     -- 13.Find the salesman's names who is getting the second highest salary.
112     SELECT Salesman_Name
113     FROM Salesman
114     WHERE Salary = (
115         SELECT MAX(Salary)
116         FROM Salesman
117         WHERE Salary < (SELECT MAX(Salary) FROM Salesman)
118     );
119
120     -- 14.Find the salesman's names who is getting second lowest salary.
121     SELECT Salesman_Name
122     FROM Salesman
123     WHERE Salary = (
124         SELECT MIN(Salary)
125         FROM Salesman
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

Salesman_Name
Tin

14. Find the salesman's names who is getting second lowest salary.

```
119
120     -- 14.Find the salesman's names who is getting second lowest salary.
121     SELECT Salesman_Name
122     FROM Salesman
123     WHERE Salary = (
124         SELECT MIN(Salary)
125         FROM Salesman
126         WHERE Salary > (SELECT MIN(Salary) FROM Salesman)
127     );
128
129     -- 15.Write a query to find the name and the salary of the salesman who have
130     SELECT Salesman_Name, Salary
131     FROM Salesman
132     WHERE Salary > (SELECT Salary FROM Salesman WHERE Salesman_Number = 'S001');
133
134     -- 16.Write a query to find the name of all salesman who sold the product ha
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

Salesman_Name
Huu

15. Write a query to find the name and the salary of the salesman who have a higher salary than the salesman whose salesman number is S001.

```
128
129     -- 15.Write a query to find the name and the salary of the salesman who have a higher salary than the sa
130     SELECT Salesman_Name, Salary
131     FROM Salesman
132     WHERE Salary > (SELECT Salary FROM Salesman WHERE Salesman_Number = 'S001');
133
134     -- 16.Write a query to find the name of all salesman who sold the product has number: P1002.
135     SELECT DISTINCT s.Salesman_Name
136     FROM Salesman s
137     JOIN SalesOrder so ON s.Salesman_Number = so.Salesman_Number
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

Salesman_Name	Salary
Phat	25000.0000
Khoa	17500.0000
Tien	16500.0000
Tin	20000.0000
Quang	25000.0000

16. Write a query to find the name of all salesman who sold the product has number: P1002.

```
134     -- 16.Write a query to find the name of all salesman who sold the product has numbe
135     SELECT DISTINCT s.Salesman_Name
136     FROM Salesman s
137     JOIN SalesOrder so ON s.Salesman_Number = so.Salesman_Number
138     JOIN SalesOrderDetails sod ON so.Order_Number = sod.Order_Number
139     WHERE sod.Product_Number = 'P1002';
140
141     -- 17.Find the name of the salesman who sold the product to client C108 with delive
142     SELECT DISTINCT s.Salesman_Name
143     FROM Salesman s
144     JOIN SalesOrder so ON s.Salesman_Number = so.Salesman_Number
145     WHERE so.Client_Number = 'C108'
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

Salesman_Name
Khoa
Deb
Huu

17. Find the name of the salesman who sold the product to client C108 with delivery status is "delivered".

```
140
141     -- 17.Find the name of the salesman who sold the product to client C108 with delivery status is "deliver
142     SELECT DISTINCT s.Salesman_Name
143     FROM Salesman s
144     JOIN SalesOrder so ON s.Salesman_Number = so.Salesman_Number
145     WHERE so.Client_Number = 'C108'
146     AND so.Delivery_Status = 'Delivered';
147
148     -- 18.Display lists the ProductName in ANY records in the sale Order Details table has Order Quantity eq
149     SELECT DISTINCT p.Product_Name
150     FROM Product p
151     JOIN SalesOrderDetails sod ON p.Product_Number = sod.Product_Number
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

Salesman_Name

Result Grid

Form

18. Display lists the ProductName in ANY records in the sale Order Details table has Order Quantity equal to

5.

```
146     AND so.Delivery_Status = 'Delivered';
147
148     -- 18.Display lists the ProductName in ANY records in the sale Order Details table has Order Quantity ec
149     SELECT DISTINCT p.Product_Name
150     FROM Product p
151     JOIN SalesOrderDetails sod ON p.Product_Number = sod.Product_Number
152     WHERE sod.Order_Quantity = 5;
153
154     -- 19.Write a query to find the name and number of the salesman who sold pen or TV or laptop.
155     SELECT DISTINCT s.Salesman_Number, s.Salesman_Name
156     FROM Salesman s
157     JOIN SalesOrder so ON s.Salesman_Number = so.Salesman_Number
158     JOIN SalesOrderDetails sod ON so.Order_Number = sod.Order_Number
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

Product_Name
TV
Laptop
AC

Result Grid

Form Editor

19. Write a query to find the name and number of the salesman who sold pen or TV or laptop.

```
153
154     -- 19.Write a query to find the name and number of the salesman who sold pen or TV or laptop.
155     SELECT DISTINCT s.Salesman_Number, s.Salesman_Name
156     FROM Salesman s
157     JOIN SalesOrder so ON s.Salesman_Number = so.Salesman_Number
158     JOIN SalesOrderDetails sod ON so.Order_Number = sod.Order_Number
159     JOIN Product p ON sod.Product_Number = p.Product_Number
160     WHERE p.Product_Name IN ('Pen', 'TV', 'Laptop');
161
162     -- 20.Lists the salesman's name sold product with a product price less than 800 and Quantity_On_Har
163     SELECT DISTINCT s.Salesman_Name
164     FROM Salesman s
165     JOIN SalesOrder so ON s.Salesman_Number = so.Salesman_Number
166     JOIN SalesOrderDetails sod ON so.Order_Number = sod.Order_Number
167     JOIN Product p ON sod.Product_Number = p.Product_Number
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

Salesman_Number	Salesman_Name
S003	Khoa
S005	Deb
S001	Huu
S004	Tien
S006	Tin

20. Lists the salesman's name sold product with a product price less than 800 and Quantity_On_Hand more than

50.

```
161
162     -- 20.Lists the salesman's name sold product with a product price less than 800 and Quantity_On_Hand mor
163     SELECT DISTINCT s.Salesman_Name
164     FROM Salesman s
165     JOIN SalesOrder so ON s.Salesman_Number = so.Salesman_Number
166     JOIN SalesOrderDetails sod ON so.Order_Number = sod.Order_Number
167     JOIN Product p ON sod.Product_Number = p.Product_Number
168     WHERE p.Sell_Price < 800
169     AND p.Quantity_On_Hand > 50;
170
171     -- 21.Write a query to find the name and salary of the salesman whose salary is greater than the average
172     SELECT Salesman_Name, Salary
173     FROM Salesman
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

Salesman_Name
Khoa
Tien
Quang

21. Write a query to find the name and salary of the salesman whose salary is greater than the average salary.

```
170
171     -- 21.Write a query to find the name and salary of the salesman whose salary is greater than the average
172     SELECT Salesman_Name, Salary
173     FROM Salesman
174     WHERE Salary > (SELECT AVG(Salary) FROM Salesman);
175
176     -- 22.Write a query to find the name and Amount Paid of the clients whose amount paid is greater than th
177     SELECT Client_Name, Amount_Paid
178     FROM Clients
179     WHERE Amount_Paid > (SELECT AVG(Amount_Paid) FROM Clients);
180
181
182     -- V.Additional excersices:
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

Salesman_Name	Salary
Phat	25000.0000
Tin	20000.0000
Quang	25000.0000

22. Write a query to find the name and Amount Paid of the clients whose amount paid is greater than the average amount

paid.

```
176    -- 22.Write a query to find the name and Amount Paid of the clients whose amount paid is greater than
177    SELECT Client_Name, Amount_Paid
178    FROM Clients
179    WHERE Amount_Paid > (SELECT AVG(Amount_Paid) FROM Clients);
180
181
182    -- V.Additional excersices:
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

Client_Name	Amount_Paid
Le Xuan	18000.0000
Nguyen Sy	15000.0000
Duong Thanh	12000.0000

V. Additional excersices:

23. Find the product price that was sold to Le Xuan.

```
181
182    -- V.Additional excersices:
183    -- 23.Find the product price that was sold to Le Xuan.
184    SELECT DISTINCT p.Sell_Price
185    FROM Product p
186    JOIN SalesOrderDetails sod ON p.Product_Number = sod.Product_Number
187    JOIN SalesOrder so ON sod.Order_Number = so.Order_Number
188    JOIN Clients c ON so.Client_Number = c.Client_Number
189    WHERE c.Client_Name = 'Le Xuan';
190
191    -- 24.Determine the product name, client name and amount due that was delivered.
192    SELECT DISTINCT p.Product_Name, c.Client_Name, c.Amount_Due
193    FROM SalesOrder so
194    JOIN SalesOrderDetails sod ON so.Order_Number = sod.Order_Number
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

Sell_Price
120.0000
250.0000

24. Determine the product name, client name and amount due that was delivered.

```
190
191     -- 24.Determine the product name, client name and amount due that was delivered.
192     SELECT DISTINCT p.Product_Name, c.Client_Name, c.Amount_Due
193     FROM SalesOrder so
194     JOIN SalesOrderDetails sod ON so.Order_Number = sod.Order_Number
195     JOIN Product p ON sod.Product_Number = p.Product_Number
196     JOIN Clients c ON so.Client_Number = c.Client_Number
197     WHERE so.Delivery_Status = 'Delivered';
198
199     -- 25.Find the salesman's name and their product name which is cancelled.
200     SELECT DISTINCT s.Salesman_Name, p.Product_Name
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

Product_Name	Client_Name	Amount_Due
TV	Mai Xuan	5000.0000
Laptop	Mai Xuan	5000.0000
Keyboard	Le Xuan	3000.0000
AC	Trinh Huu	3200.0000
Modem	Tran Tuan	0.0000
Pen	Tran Hai	1300.0000
Mouse	Tran Hai	1300.0000
TV	Tran Hai	1300.0000

25. Find the salesman's name and their product name which is cancelled.

```
198
199     -- 25.Find the salesman's name and their product name which is cancelled.
200     SELECT DISTINCT s.Salesman_Name, p.Product_Name
201     FROM SalesOrder so
202     JOIN SalesOrderDetails sod ON so.Order_Number = sod.Order_Number
203     JOIN Salesman s ON so.Salesman_Number = s.Salesman_Number
204     JOIN Product p ON sod.Product_Number = p.Product_Number
205     WHERE so.Order_Status = 'Cancelled';
206
207     -- 26.Find product names, prices and delivery status for those products purchased by
208     SELECT DISTINCT p.Product_Name, p.Sell_Price, so.Delivery_Status
209     FROM Clients c
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

Salesman_Name	Product_Name
Khoa	Keyboard
Khoa	TV
Khoa	Headset
Khoa	Laptop
Deb	Keyboard

26. Find product names, prices and delivery status for those products purchased by Nguyen Thanh.

```
206
207     -- 26.Find product names, prices and delivery status for those products purchased by Nguyen Thanh.
208     SELECT DISTINCT p.Product_Name, p.Sell_Price, so.Delivery_Status
209     FROM Clients c
210     JOIN SalesOrder so ON c.Client_Number = so.Client_Number
211     JOIN SalesOrderDetails sod ON so.Order_Number = sod.Order_Number
212     JOIN Product p ON sod.Product_Number = p.Product_Number
213     WHERE c.Client_Name = 'Nguyen Thanh';
214
215     -- 27.Display the product name, sell price, salesperson name, delivery status, and order quantity infor
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

Product_Name	Sell_Price	Delivery_Status

27. Display the product name, sell price, salesperson name, delivery status, and order quantity information for each

customer.

```
216 •  SELECT c.Client_Name, p.Product_Name, p.Sell_Price, s.Salesman_Name, so.Delivery_Status, sod.Order_Quant
217    FROM Clients c
218    JOIN SalesOrder so ON c.Client_Number = so.Client_Number
219    JOIN SalesOrderDetails sod ON so.Order_Number = sod.Order_Number
220    JOIN Product p ON sod.Product_Number = p.Product_Number
221    JOIN Salesman s ON so.Salesman_Number = s.Salesman_Number;
222
223    -- 28.Find the names, product names, and order dates of all sales staff whose product order status has b
224 •  SELECT DISTINCT s.Salesman_Name, p.Product_Name, so.Order_Date
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

Client_Name	Product_Name	Sell_Price	Salesman_Name	Delivery_Status	Order_Quantity
Mai Xuan	TV	1000.0000	Khoa	Delivered	5
Mai Xuan	TV	1000.0000	Khoa	On Way	8
Tran Hai	TV	1000.0000	Tin	Delivered	9
Nguyen Sy	TV	1000.0000	Tien	On Way	2
Mai Xuan	Laptop	1500.0000	Khoa	Delivered	4
Mai Xuan	Laptop	1500.0000	Khoa	On Way	14
Ho Nhu	Laptop	1500.0000	Deb	Ready to Ship	5
Tran Minh	Laptop	1500.0000	Huu	On Way	20
Trinh Huu	AC	400.0000	Phat	Delivered	12
Duong Thanh	AC	400.0000	Hoa	Ready to Ship	5
Tran Tuan	Modem	250.0000	Khoa	Delivered	3
Le Xuan	Modem	250.0000	Tin	Ready to Ship	8
Tran Hai	Pen	12.0000	Huu	Delivered	6
Nguyen Sy	Pen	12.0000	Tien	On Way	3
Tran Hai	Mouse	100.0000	Tin	Delivered	11
Duong Thanh	Mouse	100.0000	Huu	Ready to Ship	10
Le Xuan	Keyboard	120.0000	Khoa	Delivered	10
Nguyen Thanh	Keyboard	120.0000	Deb	Ready to Ship	6
Nguyen Sy	Keyboard	120.0000	Quang	On Way	10
Duong Thanh	Keyboard	120.0000	Hoa	Ready to Ship	20
Mai Xuan	Headset	50.0000	Khoa	On Way	15
Mai Xuan	Headset	50.0000	Tien	On Way	2
Nguyen Sy	Headset	50.0000	Quang	On Way	15

Result Grid | Form Editor | Field Types | Query Stats | Execution Plan

28. Find the names, product names, and order dates of all sales staff whose product order status has been successful but the items have not yet been delivered to the

client.

```
222
223     -- 28.Find the names, product names, and order dates of all sales staff whose product order status has b
224     SELECT DISTINCT s.Salesman_Name, p.Product_Name, so.Order_Date
225     FROM SalesOrder so
226     JOIN SalesOrderDetails sod ON so.Order_Number = sod.Order_Number
227     JOIN Product p ON sod.Product_Number = p.Product_Number
228     JOIN Salesman s ON so.Salesman_Number = s.Salesman_Number
229     WHERE so.Order_Status = 'Successful'
230     AND so.Delivery_Status <> 'Delivered';
231
232     -- 29.Find each clients' product which in on the way.
233     SELECT c.Client_Name, p.Product_Name
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

Salesman_Name	Product_Name	Order_Date
Tien	Headset	2022-04-24
Tien	Pen	2022-05-12
Tien	TV	2022-05-12
Huu	Laptop	2022-05-16
Quang	Headset	2022-05-12
Quang	Keyboard	2022-05-12

Result Grid | Form Editor

29. Find each clients' product which in on the

```
231
232     -- 29.Find each clients' product which in on the way.
233     SELECT c.Client_Name, p.Product_Name
234     FROM Clients c
235     JOIN SalesOrder so ON c.Client_Number = so.Client_Number
236     JOIN SalesOrderDetails sod ON so.Order_Number = sod.Order_Number
237     JOIN Product p ON sod.Product_Number = p.Product_Number
238     WHERE so.Delivery_Status = 'On Way';
239
240     -- 30.Find salary and the salesman's names who is getting the highest salary
241     SELECT Salesman_Name, Salary
242     FROM Salesman
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

Client_Name	Product_Name
Mai Xuan	TV
Mai Xuan	Headset
Mai Xuan	Laptop
Mai Xuan	Headset
Nguyen Sy	Pen
Nguyen Sy	TV
Tran Minh	Laptop
Nguyen Sy	Headset
Nguyen Sy	Keyboard

way.

30. Find salary and the salesman's names who is getting the highest salary.

```
239
240     -- 30.Find salary and the salesman's names who is getting the highest salary
241     SELECT Salesman_Name, Salary
242     FROM Salesman
243     WHERE Salary = (SELECT MAX(Salary) FROM Salesman);
244
245     -- 31.Find salary and the salesman's names who is getting second lowest salary.
246     SELECT Salesman_Name, Salary
247     FROM Salesman
248     WHERE Salary = (
249         SELECT MIN(Salary)
250         FROM Salesman
251         WHERE Salary > (SELECT MIN(Salary) FROM Salesman)
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

Salesman_Name	Salary
Phat	25000.0000
Quang	25000.0000

31. Find salary and the salesman's names who is getting second lowest salary.

```
244
245     -- 31.Find salary and the salesman's names who is getting second lowest salary.
246     SELECT Salesman_Name, Salary
247     FROM Salesman
248     WHERE Salary = (
249         SELECT MIN(Salary)
250         FROM Salesman
251         WHERE Salary > (SELECT MIN(Salary) FROM Salesman)
252     );
253
254     -- 32.Display lists the ProductName in ANY records in the sale Order Details table has Order Qua
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

Salesman_Name	Salary
Huu	15000.0000

32. Display lists the ProductName in ANY records in the sale Order Details table has Order Quantity more than

9.

```
253
254     -- 32.Display lists the ProductName in ANY records in the sale Order Details table has Order Quantity mo
255     SELECT DISTINCT p.Product_Name
256     FROM Product p
257     JOIN SalesOrderDetails sod ON p.Product_Number = sod.Product_Number
258     WHERE sod.Order_Quantity > 9;
259
260     -- 33.Find the name of the customer who ordered the same item multiple times.
261     SELECT DISTINCT c.Client_Name
262     FROM (
263         SELECT so.Client_Number, sod.Product_Number, COUNT(*) AS cnt
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

Product_Name
AC
Headset
Keyboard
Laptop
Mouse

Result Grid | Form Editor

33. Find the name of the customer who ordered the same item multiple times.

```
259
260     -- 33.Find the name of the customer who ordered the same item multiple times.
261     SELECT DISTINCT c.Client_Name
262     FROM (
263         SELECT so.Client_Number, sod.Product_Number, COUNT(*) AS cnt
264         FROM SalesOrder so
265         JOIN SalesOrderDetails sod ON so.Order_Number = sod.Order_Number
266         GROUP BY so.Client_Number, sod.Product_Number
267         HAVING COUNT(*) > 1
268         ) t
269     JOIN Clients c ON t.Client_Number = c.Client_Number;
270
271     -- 34.Write a query to find the name, number and salary of the salemans who earns less than t
272     SELECT Salesman_Number, Salesman_Name, Salary
273     FROM Salesman
274     WHERE Salary < (SELECT AVG(Salary) FROM Salesman)
275     AND City = 'Thu Dau Mot';
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

Client_Name
Mai Xuan

34. Write a query to find the name, number and salary of the salemans who earns less than the average salary and works in any of Thu Dau Mot city.

```
269     JOIN Clients c ON t.Client_Number = c.Client_Number;
270
271     -- 34.Write a query to find the name, number and salary of the salemans who earns less than the average
272     SELECT Salesman_Number, Salesman_Name, Salary
273     FROM Salesman
274     WHERE Salary < (SELECT AVG(Salary) FROM Salesman)
275     AND City = 'Thu Dau Mot';
276
277     -- 35.Write a query to find the name, number and salary of the salemans who earn a salary that is higher
```

Result Grid | Filter Rows: | Edit: | Export/Import: | Wrap Cell Content:

Salesman_Number	Salesman_Name	Salary
S003	Khoa	17500.0000
S005	Deb	13500.0000
S008	Hoa	13500.0000
NULL	NULL	NULL

35. Write a query to find the name, number and salary of the salemans who earn a salary that is higher than the salary of all the salesman have (Order_status = ‘Cancelled’). Sort the results of the salary of the lowest to highest.

```
276
277     -- 35.Write a query to find the name, number and salary of the salemans who earn a salary that is higher
278     SELECT s.Salesman_Number, s.Salesman_Name, s.Salary
279     FROM Salesman s
280     WHERE s.Salary > ALL (
281           SELECT s2.Salary
282           FROM Salesman s2
283           JOIN SalesOrder so ON s2.Salesman_Number = so.Salesman_Number
284           WHERE so.Order_Status = 'Cancelled'
285         )
286     ORDER BY s.Salary ASC;
287
288     -- 36.Write a query to find the 4th maximum salary on the salesman's table.
```

Result Grid | Filter Rows: | Edit: | Export/Import: | Wrap Cell Content:

Salesman_Number	Salesman_Name	Salary
S006	Tin	20000.0000
S002	Phat	25000.0000
S007	Quang	25000.0000
NULL	NULL	NULL

36. Write a query to find the 4th maximum salary on the salesman's table.

```
287
288     -- 36.Write a query to find the 4th maximum salary on the salesman's table.
289 •   SELECT DISTINCT Salary
290     FROM Salesman
291     ORDER BY Salary DESC
292     LIMIT 1 OFFSET 3;
293
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content: | Fetch rows:

Salary
16500.0000

37. Write a query to find the 3th minimum salary in the salesman's table.

```
293
294     -- 37.Write a query to find the 3th minimum salary in the salesman's table.
295 •   SELECT DISTINCT Salary
296     FROM Salesman
297     ORDER BY Salary ASC
298     LIMIT 1 OFFSET 2;
299
300
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content: | Fetch rows:

Salary
16500.0000