

SMIT CLOUD DATA ENGINEERING BATCH – 3

SQL SERVER QUIZ # 1

QUERIES AND OUTPUT

NAME: HAFIZ MUHAMMAD WAQAS

ROLL NO: 465476

Q1. List top 5 customers by total order amount.

Retrieve the top 5 customers who have spent the most across all sales orders. Show CustomerID, CustomerName, and TotalSpent.

SQL QUERY:

select

```
TOP 5  
c.CustomerID,  
c.Name,  
sum(s.TotalAmount) as TotalSales  
from [dbo].[Customer] c  
inner join SalesOrder s  
on s.CustomerID = c.CustomerID  
group by c.CustomerID, c.Name  
order by TotalSales desc;
```

OUTPUT:

Results		Messages	
	CustomerID	Name	TotalSales
1	98	Armando	232783.99
2	12	Dwight	175719.20
3	84	Courtney	156259.77
4	90	Chelsae	149290.57
5	27	Lawton	141343.95

Q2. Find the number of products supplied by each supplier.

Display SupplierID, SupplierName, and ProductCount. Only include suppliers that have more than 10 products.

SQL QUERY:

```
select s.SupplierID, s.Name, count(p.ProductID) as ProductsCount
from [dbo].[Supplier] s
inner join [dbo].[PurchaseOrder] po
on s.SupplierID = po.SupplierID
inner join [dbo].[PurchaseOrderDetail] pod
on po.OrderID = pod.OrderID
inner join [dbo].[Product] p
on pod.ProductID = p.ProductID
group by s.SupplierID, s.Name
having count(p.ProductID) > 10;
```

OUTPUT:

	SupplierID	Name	ProductsCount
1	1	Schamberger Inc	44
2	2	Bogisich-Block	51
3	3	Abshire Inc	49
4	4	Schmidt LLC	49
5	5	Mann, Will and Fritsch	45
6	6	Abernathy-Waelchi	46
7	7	Kassulke LLC	62
8	8	Lang-Rodriguez	59
9	9	Sawayn-Adams	43
10	10	Stokes, Funk and Herman	52

Q3. Identify products that have been ordered but never returned.

Show ProductID, ProductName, and total order quantity.

SQL QUERY:

```
select p.ProductID, p.Name, sum(sod.Quantity) as TotalOrderQuantity
```

```

from [dbo].[Product] p
left join [dbo].[ReturnDetail] r
on p.ProductID = r.ProductID
inner join [dbo].[SalesOrderDetail] sod
on p.ProductID = sod.ProductID
WHERE r.ProductID is null
group by p.ProductID, p.Name;

```

OUTPUT:

	ProductID	Name	TotalOrderQuantity
1	3	Sertraline Hydrochloride	20
2	15	Flecainide Acetate	55

Q4. For each category, find the most expensive product.

Display CategoryID, CategoryName, ProductName, and Price. Use a subquery to get the max price per category.

SQL QUERY:

```

select c.CategoryID, c.Name, p.Name, p.Price
from [dbo].[Category] c
inner join [dbo].[Product] p
on c.CategoryID = p.CategoryID
where p.Price = (select Max(p1.Price) from [dbo].[Product] p1
where p.CategoryID = p1.CategoryID
)order by p.Price desc;

```

OUTPUT:

Results | Messages

	CategoryID	Name	Name	Price
1	7	Fire Protection	Oral Analgesic	2980.75
2	16	Elevator	Sertraline Hydrochloride	2896.17
3	17	Termite Control	SohMed Cold Relief	2813.71
4	13	NULL	COMETRIQ	2778.36
5	14	Glass & Glazing	Losartan Potassium	2749.20
6	12	Asphalt Paving	Acetaminophen PM	2677.44
7	9	Termite Control	Secura Moisturizing Cleanser	2627.59
8	5	Soft Flooring and Base	Isopropyl Alcohol Surgical Scrub	2554.94
9	18	Wall Protection	Natural Cherry Honey Herb Throat Drops	2502.47
10	4	Painting & Vinyl Wall Covering	Focalin	2455.81
11	3	Framing (Wood)	ILLUMINATING DAILY MOISTURIZER	2437.99
12	10	Exterior Signage	Flecainide Acetate	2325.87
13	19	Epoxy Flooring	Entex T	2012.38
14	15	Hard Tile & Stone	Quetiapine Fumarate	1831.84
15	2	Landscaping & Irrigation	Betapace AF	1543.64
16	6	Electrical and Fire Alarm	Fluoxetine	726.23
17	20	Hard Tile & Stone	Butalbital, Acetaminophen and Caffeine	85.95

Q5. List all sales orders with customer name, product name, category, and supplier.

For each sales order, display:

OrderID, CustomerName, ProductName, CategoryName, SupplierName, and Quantity.

SQL QUERY:

```

select so.OrderID, cus.Name CustomerName, p.Name ProductName, c.Name
CategoryName, s.Name SupplierName, sod.Quantity

from [dbo].[Supplier] s

inner join [dbo].[PurchaseOrder] po
on s.SupplierID = po.SupplierID

inner join [dbo].[PurchaseOrderDetail] pod
on po.OrderID = pod.OrderID

inner join [dbo].[Product] p
on pod.ProductID = p.ProductID

inner join [dbo].[Category] c
on p.CategoryID = c.CategoryID

inner join [dbo].[SalesOrderDetail] sod
on p.ProductID = sod.ProductID

```

```
inner join [dbo].[SalesOrder] so  
on sod.OrderID = so.OrderID  
inner join [dbo].[Customer] cus  
on so.CustomerID = cus.CustomerID;
```

--- OR ---

```
SELECT  
    so.OrderID,  
    cus.Name AS CustomerName,  
    p.Name AS ProductName,  
    c.Name AS CategoryName,  
    m.Name AS SupplierName,  
    sod.Quantity  
FROM dbo.SalesOrder so  
INNER JOIN dbo.Customer cus  
    ON so.CustomerID = cus.CustomerID  
INNER JOIN dbo.SalesOrderDetail sod  
    ON so.OrderID = sod.OrderID  
INNER JOIN dbo.Product p  
    ON sod.ProductID = p.ProductID  
INNER JOIN dbo.Category c  
    ON p.CategoryID = c.CategoryID  
INNER JOIN dbo.Manufacturer m  
    ON p.ManufacturerID = m.ManufacturerID;
```

OUTPUT:

Results Messages

OrderID	CustomerName	ProductName	CategoryName	SupplierName	Quantity
1	Ann Arbor	Foodim	Flooring & Vinyl Wall Covering	Fender	12
2	Armenia	Acetylene	HULL	Cogitex	10
3	Bustos	Mefenim Hydrochloride	Termite Control	Misa	21
4	Savoy	Model Complete	Termite Control	Mennin	29
5	Diegert	KODIS-CHOCO	Soft Flooring and Base	Tender	30
6	Dreiss	Another From Weeks	Fire Protection	Fogeks	29
7	Dreiss	SSD Cream	Exterior Siding	Misa	26
8	Gesar	Melatonin Hydrochloride	Termite Control	Misa	10
9	Euse	Desmethylaspirin	Landscape & Irrigation	Cogitex	25
10	Euse	Severe Moisturizing Ointment	Termite Control	Blodgett	22
11	S.Ayer	Granisetron Hydrochloride	Fire Protection	Cogitex	22
12	Tremont	Emol T	Exterior Painting	Moren	31
13	Kat	Polo One	Exterior Siding	Taco	15
14	Wells	Aztreminopan PM	Rough Paving	Topdrive	21
15	Tecza	Permethrin Hydrochloride	Framing (Wood)	Misa	13
16	Morris	Resiproc AF	Landscape & Irrigation	Taco	24
17	Tecza	Maximum Strength	Framing (Wood)	Fenocle	21
18	Lewton	Isosopropyl Alcohol Surgical E.	Soft Flooring and Base	Fistorm	32
19	Loulka	PASIPALUM-NOTATUM PO	Glass & Glazing	Topdrive	14
20	Ocasa	COMETRO	HULL	Cogitex	18
21	Mannie	Fluoxetine	Electrical and Fire Alarm	Cogitex	31
22	Lewton	Nature Cherry Honey Herb	Wall Protection	Topdrive	29
23	Francesca	Ondansetron	Hard Tile & Stone	Taco	14
24	Moxosa	PASIPALUM-NOTATUM PO	Glass & Glazing	Topdrive	13
25	Reek	KODIS-CHOCO	Soft Flooring and Base	Tender	30
26	Truda	Extra Strength Naproxen	Glass & Glazing	Tender	18
27	Stevie	Fluoxetine	Electrical and Fire Alarm	Cogitex	12
28	Truda	Severe Moisturizing Ointment	Termite Control	Blodgett	18

Query executed successfully.

Q6. Find all shipments with details of warehouse, manager, and products shipped.

Display:

ShipmentID, WarehouseName, ManagerName, ProductName, QuantityShipped, and TrackingNumber.

SQL QUERY:

select

```
s.ShipmentID,
l.Name Warehouse_Name,
e.Name Manager_Name,
p.Name Product_Name,
sd.Quantity,
s.TrackingNumber
from [dbo].[Shipment] s
inner join [dbo].[Warehouse] w
on s.WarehouseID = w.WarehouseID
inner join [dbo].[Location] l
on w.LocationID = l.LocationID
inner join [dbo].[Employee] e
on w.ManagerID = e.ManagerID
inner join [dbo].[ShipmentDetail] sd
on s.ShipmentID = sd.ShipmentID
```

inner join [dbo].[Product] p
on sd.ProductID = p.ProductID

OUTPUT:

ProductID	Warehouse_Name	Manager_Name	Product_Name	Quantity	TrackingNumber
1	M1	Raisio	INSPALUM ROTATIUM POLLEN	8	NULL
2	M1	Dalene	SHISEIDO ADVANCED HYDRO-LIQUID COMPACT (REFILL)	1	NULL
3	M1	Aurora	SHISEIDO ADVANCED HYDRO-LIQUID COMPACT (REFILL)	1	NULL
4	M1	Kapooe	SHISEIDO ADVANCED HYDRO-LIQUID COMPACT (REFILL)	1	NULL
5	M1	Baskin	Icy Hot Back and Large Areas	2	NULL
6	M1	Dalene	Natural Cherry Honey Hots Throat Drops	20	NULL
7	M1	Aurora	Natural Cherry Honey Hots Throat Drops	20	NULL
8	M1	Kapooe	Natural Cherry Honey Hots Throat Drops	20	NULL
9	M1	Dalene	Maximum Strength	16	NULL
10	M1	Aurora	Maximum Strength	16	NULL
11	M1	Kapooe	Maximum Strength	16	NULL
12	M1	Baskin	Quinine Fumurate	21	NULL
13	M1	Nelke	Antibacterial Feminine Hand Sanitizer	21	NULL
14	M1	Baskin	INSPALUM ROTATIUM POLLEN	1	NULL
15	M1	Dalene	Salicylic Acid Hydrochloride	30	NULL
16	M1	Aurora	Salicylic Acid Hydrochloride	30	NULL
17	M1	Kapooe	Salicylic Acid Hydrochloride	30	NULL
18	M1	Baskin	SALICYLIC ACID POLLEN	8	NULL
19	M1	Nelke	Methionine Hydrochloride	26	NULL
20	M1	Dalene	Mitol Complete	26	NULL
21	M1	Aurora	Mitol Complete	26	NULL
22	M1	Kapooe	Mitol Complete	26	NULL
23	M1	Dalene	Anti-Eos Foaming Wash	27	NULL

Q7. Find the top 3 highest-value orders per customer using RANK(). Display CustomerID, CustomerName, OrderID, and TotalAmount.

SQL QUERY:

with cte as (

select

c.CustomerID,

c.Name,

so.OrderID,

so.TotalAmount,

RANK() OVER (PARTITION BY so.CustomerID ORDER BY so.TotalAmount DESC) rnk

from

[dbo].[Customer] c

inner join [dbo].[SalesOrder] so

on c.CustomerID = so.CustomerID

)

select *

from

cte

where rnk <= 3

OUTPUT:

	CustomerID	Name	OrderID	TotalAmount	rnk
1	2	Davne	87	20549.27	1
2	3	Neulis	68	53375.84	1
3	3	Neulis	93	40001.82	2
4	4	Euvince	9	28705.68	1
5	6	Munep	16	12849.15	1
6	7	Lam	91	73162.12	1
7	11	Dorv	43	51943.81	1
8	12	Dwight	5	91621.22	1
9	12	Dwight	52	99504.03	2
10	12	Dwight	9	24503.96	3
11	13	Liliane	48	84724.08	1
12	16	Savven	27	23096.23	1
13	19	Xesia	64	70385.28	1
14	19	Xesia	74	61123.58	2
15	21	Luzene	36	90207.48	1
16	21	Luzene	87	84102.24	2
17	24	Edify	82	85551.84	1
18	25	Lief	41	98152.77	1
19	25	Lief	47	32960.68	2
20	26	Turbol	96	62386.56	1
21	28	Turbol	80	20782.61	2
22	27	Lawton	71	83378.19	1
23	27	Lawton	22	31801.84	2

Q8. For each product, show its sales history with the previous and next sales quantities (based on order date). Display ProductID, ProductName, OrderID, OrderDate, Quantity, PrevQuantity, and NextQuantity.

SQL QUERY:

select

```
p.ProductID,  
p.Name,  
sod.OrderID,  
so.OrderDate,  
sod.Quantity,  
LAG(sod.Quantity) OVER (PARTITION BY p.ProductID ORDER BY so.OrderDate)  
PrevQuantity,  
LEAD(sod.Quantity) OVER (PARTITION BY p.ProductID ORDER BY so.OrderDate)  
NextQuantity
```

from

```
[dbo].[SalesOrderDetail] sod  
inner join [dbo].[Product] p  
on sod.ProductID = p.ProductID  
inner join [dbo].[SalesOrder] so  
on sod.OrderID = so.OrderID;
```

OUTPUT:

Results Messages

	ProductID	Name	OrderID	OrderDate	Quantity	Freight	ShipQuantity
1	1	BuzzTonic: Aspartamefree and Caffeine	79	2003-04-21 04:11:36	19	NULL	17
2	1	BuzzTonic: Aspartamefree and Caffeine	81	2003-05-16 0:0:37:46	17	79	13
3	1	BuzzTonic: Aspartamefree and Caffeine	88	2003-07-11 14:0:27	13	17	NULL
4	2	JH-ECO-DO ADVANCED HYDRO LIQUID COMPACT (REFILL)	90	2000-12-26 15:50:28	23	NULL	NULL
5	3	Serratiopeptidase	98	2006-10-20 02:52:28	20	NULL	NULL
6	4	SofMed Cold Relief	12	2003-11-14 11:55:28	24	NULL	14
7	4	SofMed Cold Relief	44	2004-04-04 09:26:01	14	24	21
8	4	SofMed Cold Relief	58	2012-04-24 07:48:19	11	14	24
9	4	SofMed Cold Relief	70	2021-04-30 09:18:44	34	11	NULL
10	5	Pravastatin Hydrochloride	16	2000-05-20 09:31:55	13	NULL	23
11	5	Pravastatin Hydrochloride	80	2017-06-06 16:48:16	25	11	NULL
12	6	Ondansetron	21	2002-02-08 08:48:16	14	NULL	24
13	6	Ondansetron	88	2011-08-18 11:07:42	24	14	37
14	6	Ondansetron	93	2013-01-29 02:38:28	27	24	38
15	6	Ondansetron	99	2020-10-08 08:33:28	35	37	NULL
16	7	TopCare Antacid Calcium	43	2007-12-19 07:06:49	15	NULL	29
17	7	TopCare Antacid Calcium	94	2008-01-01 05:58:21	29	15	NULL
18	8	Maximum Strength	17	2019-07-20 20:13:49	21	NULL	NULL
19	8	Secure Manufacturing Cleanse	28	2003-01-18 13:11:31	34	NULL	18
20	8	Secure Manufacturing Cleanse	28	2014-10-07 17:30:02	18	34	22
21	8	Secure Manufacturing Cleanse	30	2005-05-22 22:58:14	22	18	NULL
22	10	SM-GOLKA KALI POLLEN	29	2008-01-01 11:47:48	37	NULL	NULL
23	11	Easa Length Mopan	36	2004-10-16 18:46:47	16	NULL	19

Query executed successfully.

Q9. Create a view named vw_CustomerOrderSummary that shows for each customer:

CustomerID, CustomerName, TotalOrders, TotalAmountSpent, and LastOrderDate.

SQL QUERY:

```
create view vw_CustomerOrderSummary
```

```
as
```

```
select
```

```
    c.CustomerID,
```

```
    c.Name,
```

```
    COUNT(so.OrderID) TotalOrders,
```

```
    SUM(so.TotalAmount) TotalAmount,
```

```
    MAX(so.OrderDate) LastOrderDate
```

```
from
```

```
    [dbo].[Customer] c
```

```
inner join [dbo].[SalesOrder] so
```

```
on c.CustomerID = so.CustomerID
```

```
group by
```

```
    c.CustomerID,
```

```
    c.Name;
```

```
select * from vw_CustomerOrderSummary;
```

OUTPUT:

Results | Messages

	CustomerID	Name	TotalOrder	TotalAmount	LastOrderDate
1	2	Dunne	1	20546.27	2008-12-11 16:36:26
2	3	Neville	3	89452.79	2010-11-30 13:33:06
3	4	Fusco	1	29706.58	2010-12-22 16:57:15
4	6	Whaley	1	12848.15	2008-07-09 01:10:56
5	7	Lam	3	73192.12	2012-03-18 16:36:10
6	11	Dixie	3	51043.81	2007-12-19 07:00:49
7	12	Dwight	3	77279.29	2009-07-31 09:01:11
8	13	Liliane	1	40724.08	2010-12-21 02:25:15
9	15	Sapna	1	22686.23	2003-11-16 00:14:29
10	19	Kerri	2	127718.85	2007-06-10 06:50:46
11	21	Lorraine	2	94310.23	2009-08-12 14:14:23
12	24	Eddy	1	80551.64	2003-03-20 21:39:58
13	29	Lief	2	90704.27	2008-03-11 04:02:48
14	26	Tobin	4	81112.37	2005-08-08 01:37:52
15	27	Leontine	3	113143.95	2010-02-12 21:50:09
16	28	Innkeeper	3	160188.68	2010-06-24 11:15:00
17	29	Ashley	1	92044.84	2009-10-08 09:33:28
18	31	Marcia	2	111324.89	2011-06-05 06:52:50
19	33	Gaines	1	37648.58	2006-03-18 00:00:43
20	36	Walt	1	15561.00	2017-05-05 11:38:48
21	37	Francoise	3	94618.31	2005-11-09 18:07:14
22	38	Jasmine	1	38029.61	2012-02-27 21:22:15
23	39	Benny	1	81216.61	2008-03-01 03:00:21

Query executed successfully.

Q10. Write a stored procedure sp_GetSupplierSales that takes a SupplierID as input and returns the total sales amount for all products supplied by that supplier.

SQL QUERY:

```
alter procedure sp_total_sales
    @SupplierID int
as
begin
select
    ISNULL(SUM(pod.TotalAmount), 0) TotalSalesAmount
from
    [dbo].[Supplier] sp
inner join [dbo].[PurchaseOrder] po
on sp.SupplierID = po.SupplierID
inner join [dbo].[PurchaseOrderDetail] pod
on po.OrderID = pod.OrderID
where sp.SupplierID = @SupplierID
end;

exec sp_total_sales 2;
```

OUTPUT:

	CustomerID	Name	TotalOrders	TotalAmount	LastOrderDate
1	2	Davne	1	29549.27	2008-12-11 11:36:29
2	3	Heire	2	89432.77	2010-11-30 13:23:08
3	4	Eduara	1	28955.93	2019-12-22 19:57:18
4	6	Munip	1	12345.15	2008-07-06 01:16:54
5	7	Lero	1	73162.12	2012-03-16 16:38:18
6	11	Doyv	1	51043.81	2007-12-16 07:18:49
7	12	Bright	3	19679.20	2020-01-21 08:01:11
8	13	Liliane	1	44724.08	2013-12-21 02:36:19
9	15	Seynre	1	23995.23	2022-11-16 00:14:29
10	18	Kenia	2	13718.09	2007-04-18 06:35:46
11	21	Lucas	2	94310.23	2005-08-12 14:14:23
12	24	Eddy	1	82551.88	2002-03-20 23:06:58
13	26	Liel	2	80764.27	2008-02-17 04:02:45
14	28	Talbot	2	81142.37	2006-09-06 07:37:33
15	27	Leontine	3	141943.09	2019-02-12 21:59:03
16	29	Imenne	3	143199.68	2019-06-24 11:19:58
17	29	Abrey	1	82943.34	2020-10-06 05:33:28
18	31	Malakia	2	111324.65	2017-08-05 06:52:53
19	35	Gurak	1	53843.58	2006-03-16 06:25:42
20	36	Wali	1	65551.93	2017-05-05 11:38:48
21	37	Fremmese	2	84416.31	2006-11-09 16:07:14
22	38	Jocelle	1	78029.91	2010-02-27 21:22:16
23	39	Burne	1	41218.81	2008-03-01 03:16:21

Query executed successfully.

DESKTOP-H96K00L (16.0 RTM) | DESKTOP-H96K00LU\USER (62) | NEXT_COA_OUTP | 00:00:00 | 63 rows