

QUESTION 1:

```
select MONTH(HireDate) As month ,count(BusinessEntityID) As Number_of_employee  
  
from HumanResources.Employee  
  
group by MONTH(HireDate)
```

```
4  select * from HumanResources.Employee
5
6
7  |
8  select MONTH(HireDate) As month ,count(BusinessEntityID) As Number_of_employee
9  from HumanResources.Employee
10 group by MONTH(HireDate)
11
12
13
14
```

Results		Messages
	month	Number_of_employee
1	1	85
2	2	78
3	3	26
4	4	1
5	5	14
6	6	1
7	9	1
8	11	1
9	12	83

```

7  -----1
8  select MONTH(HireDate) As month ,count(BusinessEntityID) As Number_of_employee
9  from HumanResources.Employee
10 group by MONTH(HireDate)
11
12 -----2
13
14 select DepartmentID ,count(BusinessEntityID) As Number_of_employee

```

Results Messages

	month	Number_of_employee
1	1	85
2	2	78
3	3	26
4	4	1
5	5	14
6	6	1
7	9	1
8	11	1
9	12	83

QUESTION:2

```

select DepartmentID ,count(BusinessEntityID) As Number_of_employee
from HumanResources.EmployeeDepartmentHistory
group by DepartmentID

```

```

9  from HumanResources.Employee
10 group by MONTH(HireDate)
11
12 -----2
13
14 select DepartmentID, count(BusinessEntityID) As Number_of_employee
15 from HumanResources.EmployeeDepartmentHistory
16 group by DepartmentID
17 -----
18 3. List the person and their country region name who is not an employee and their

```

Results Messages

	DepartmentID	Number_of_employee
1	1	7
2	2	4
3	3	18
4	4	10
5	5	13
6	6	4
7	7	180
8	8	6
9	9	6
10	10	11
11	11	10
12	12	5
13	13	7
14	14	7
15	15	6
16	16	2

The screenshot shows the SQL Server Enterprise Manager interface. The query editor displays the following SQL code:

```

5
6
7 -----1
8 select MONTH(HireDate) As month ,count(BusinessEntityID) As Number_of_employee
9 from HumanResources.Employee
10 group by MONTH(HireDate)
11
12 -----2
13
14 select DepartmentID ,count(BusinessEntityID) As Number_of_employee
15 from HumanResources.EmployeeDepartmentHistory
16 group by DepartmentID
17
18 3. List the person and their country region name who is not an employee and their

```

The Results tab shows the following data:

DepartmentID	Number_of_employee
1	7
2	4
3	18
4	10
5	13
6	4
7	100
8	6
9	6
10	11
11	10
12	5
13	7
14	7
15	6
16	2

### QUESTION 3:

select CONCAT(FirstName,MiddleName,LastName) AS NAME from Person.Person

where BusinessEntityID NOT IN(select BusinessEntityID

from HumanResources.Employee) and FirstName like 'S%'

The screenshot shows the SQL Server Enterprise Manager interface. The query editor displays the following SQL code:

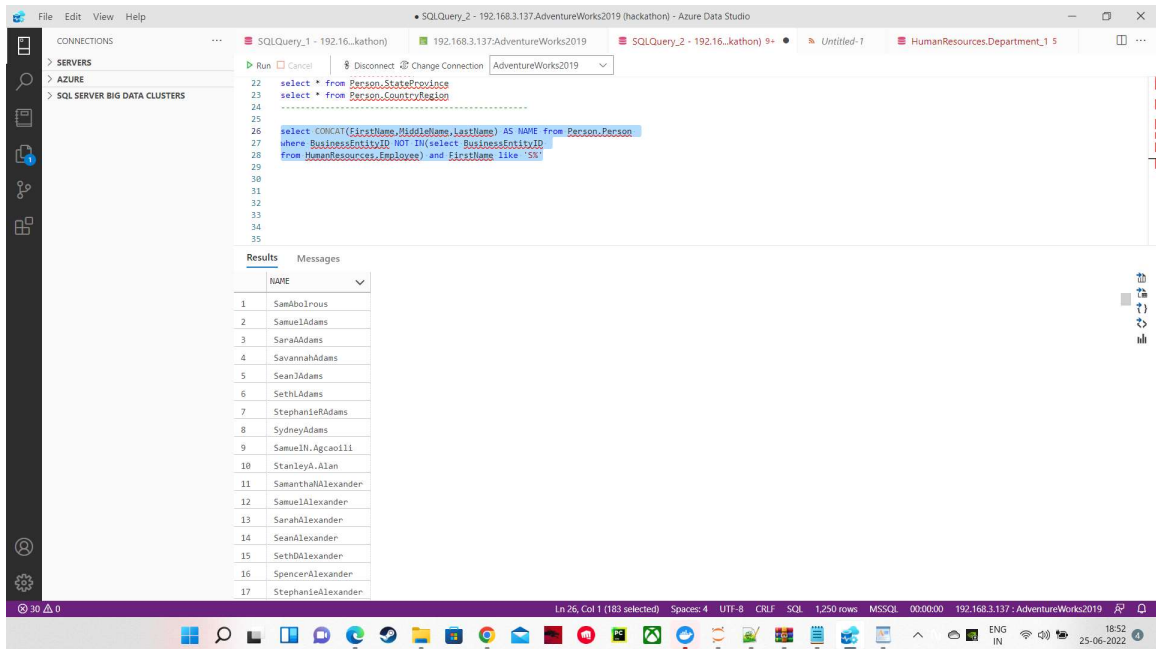
```

24
25
26 select FirstName from Person.Person where BusinessEntityID NOT IN(select BusinessEntityID from HumanResources.Employee) and FirstName like 'S%'
27
28
29
30

```

The Results tab shows the following data:

FirstName	
1	Sam
2	Samuel
3	Sara
4	Savannah
5	Sean
6	Seth
7	Stephanie
8	Sydney
9	Samuel
10	Stanley
11	Samantha
12	Samuel
13	Sarah
14	Sean
15	Seth
16	Spencer
17	Stephanie



Question 4:

select CONCAT(A.FirstName,A.MiddleName,A.LastName),B.JobTitle,B.OrganizationLevel,  
COUNT(B.OrganizationLevel) OVER(PARTITION BY B.OrganizationLevel ) as  
NumberOfEmployeeReporting

from Person.Person A

JOIN HumanResources.Employee B ON

A.BusinessEntityID=B.BusinessEntityID

```

34
35
36
37 select CONCAT(A.FirstName,A.MiddleName,A.LastName),B.JobTitle,B.OrganizationLevel,
38 COUNT(B.OrganizationLevel) OVER(PARTITION BY B.OrganizationLevel ) as NumberOfEmployeeReporting
39 from Person.Person A
40 JOIN HumanResources.Employee B ON
41 A.BusinessEntityID=B.BusinessEntityID
42
43

```

## Results Messages

	(No column name)	JobTitle	OrganizationLevel	NumberOfEmployeeReporting
1	KenJSánchez	Chief Executive Officer	NULL	0
2	TerrileeDuffy	Vice President of Engineering	1	6
3	DavidMBradley	Marketing Manager	1	6
4	JamesRHamilton	Vice President of Production	1	6
5	LauraFNorman	Chief Financial Officer	1	6
6	JeanETrenary	Information Services Manager	1	6
7	BrianSWelcker	Vice President of Sales	1	6
8	StephenYJiang	North American Sales Manager	2	27
9	DavidMBarber	Assistant to the Chief Finan...	2	27
10	SyedEAbbas	Pacific Sales Manager	2	27
11	AmyEAlberts	European Sales Manager	2	27
12	StephanieAConroy	Network Manager	2	27
13	KarenABerg	Application Specialist	2	27
14	RameshVMeyyappan	Application Specialist	2	27
15	DanKBacon	Application Specialist	2	27
16	FrançoisPAjenstat	Database Administrator	2	27
17	DanBWilson	Database Administrator	2	27

File Edit View Help • SQLQuery\_2 - 192.168.3.137:AdventureWorks2019 (hackerthon) - Azure Data Studio

CONNECTIONS ... SQLQuery\_1 - 192.16...kathon) 192.168.3.137:AdventureWorks2019 SQLQuery\_2 - 192.16...kathon) 9+ • Untitled-1 HumanResources.Department\_1 5

SERVERS

AZURE

SQL SERVER BIG DATA CLUSTERS

Run Cancel Disconnect Change Connection AdventureWorks2019

```

32
33
34
35
36
37 select CONCAT(A.FirstName,A.MiddleName,A.LastName),B.JobTitle,B.OrganizationLevel,
38 COUNT(B.OrganizationLevel) OVER(PARTITION BY B.OrganizationLevel ) as NumberOfEmployeeReporting
39 from Person.Person A
40 JOIN HumanResources.Employee B ON
41 A.BusinessEntityID=B.BusinessEntityID
42
43

```

Results Messages

	(No column name)	JobTitle	OrganizationLevel	NumberOfEmployeeReporting
1	KenJSánchez	Chief Executive Officer	NULL	0
2	TerrileeDuffy	Vice President of Engineering	1	6
3	DavidMBradley	Marketing Manager	1	6
4	JamesRHamilton	Vice President of Production	1	6
5	LauraFNorman	Chief Financial Officer	1	6
6	JeanETrenary	Information Services Manager	1	6
7	BrianSWelcker	Vice President of Sales	1	6
8	StephenYJiang	North American Sales Manager	2	27
9	DavidMBarber	Assistant to the Chief Finan...	2	27
10	SyedEAbbas	Pacific Sales Manager	2	27
11	AmyEAlberts	European Sales Manager	2	27
12	StephanieAConroy	Network Manager	2	27
13	KarenABerg	Application Specialist	2	27
14	RameshVMeyyappan	Application Specialist	2	27
15	DanKBacon	Application Specialist	2	27
16	FrançoisPAjenstat	Database Administrator	2	27
17	DanBWilson	Database Administrator	2	27

Ln 37, Col 1 (277 selected) Spaces: 4 UTF-8 CRLF SQL 290 rows MSSQL 00:00:00 192.168.3.137: AdventureWorks2019

40 19:17 25-06-2022

---

QUESTION 5:

select A.productid,A.Name,SUM(B.ActualCost)

OVER(PARTITION BY B.transactionDate) AS CUMMULATIVE\_SUM ,SUM(B.quantity)

OVER(PARTITION BY B.transactionDate) AS COMMULATIVE\_QUANTITY

from Production.Product A

join Production.TransactionHistory B

on A.Productid=B.Productid

```
51 select A.productid,A.Name,SUM(B.ActualCost)
52 OVER(PARTITION BY B.transactionDate) AS CUMMULATIVE_SUM ,SUM(B.quantity) OVER(PARTITION BY B.transactionDate) AS COMMULATIVE_QUANTITY
53 from Production.Product A
54 join Production.TransactionHistory B
55 on A.Productid=B.Productid
```

Results Messages

	productid	Name	CUMMULATIVE_SUM	COMMULATIVE_QUANTITY
14	826	LL Road Rear Wheel	22578.823	7639
15	890	HL Touring Frame - Blue, 46	22578.823	7639
16	876	Hitch Rack - 4-Bike	22578.823	7639
17	875	Racing Socks, L	22578.823	7639
18	975	Road-350-W Yellow, 44	22578.823	7639
19	876	Hitch Rack - 4-Bike	22578.823	7639
20	318	ML Crankarm	22578.823	7639
21	997	Road-750 Black, 44	22578.823	7639
22	747	HL Mountain Frame - Black, 38	22578.823	7639
23	811	LL Road Handlebars	22578.823	7639
24	342	Flat Washer 6	23362.4635	12925
25	959	Touring-3000 Blue, 58	23362.4635	12925
26	977	Road-750 Black, 58	23362.4635	12925
27	742	HL Mountain Frame - Silver, ...	23362.4635	12925
28	810	HL Mountain Handlebars	23362.4635	12925
29	824	ML Mountain Rear Wheel	23362.4635	12925
30	827	ML Road Rear Wheel	23362.4635	12925

The screenshot shows the SQL Server Enterprise Manager interface. The query editor displays the following SQL query:

```

select * from Production.Product
select * from Production.TransactionHistory
--
select A.productid, A.Name, SUM(B.ActualCost)
OVER(PARTITION BY B.transactionDate) AS CUMULATIVE_SUM, SUM(B.quantity) OVER(PARTITION BY B.transactionDate) AS CUMULATIVE_QUANTITY
from Production.Product A
join Production.TransactionHistory B
on A.Productid=B.Productid
--
select

```

The Results pane shows the following data:

productid	Name	CUMULATIVE_SUM	CUMULATIVE_QUANTITY
94	930	HL Mountain Tire	39824.63
95	979	Touring-3000 Blue, 50	39824.63
96	522	HL Touring Seat Assembly	39824.63
97	722	LL Road Frame - Black, 58	39824.63
98	829	Touring Rear Wheel	39824.63
99	836	ML Road Frame-W - Yellow, -	39824.63
100	329	Road End Caps	39824.63
101	529	Stem	39824.63
102	974	Road-350-W Yellow, 42	22578.823
103	324	Chain Stays	22578.823
104	792	Road-250 Red, 58	22578.823
105	877	Bike Wash - Dissolver	22578.823
106	859	Half-Finger Gloves, M	22578.823
107	934	Touring Tire	22578.823
108	317	LL Crankarm	22578.823
109	364	Thin-Jam Hex Nut 16	22578.823
110	968	Touring-1000 Blue, 54	22578.823

Question 9:

QUESTION:9

query:

select CurrencyRateID AS CurrencyCode,

RANK() OVER( ORDER BY EndofDayRate) as current\_rate

from Sales.Currencyrate



Run
Cancel
Disconnect
Change Connection
AdventureWorks2019

```

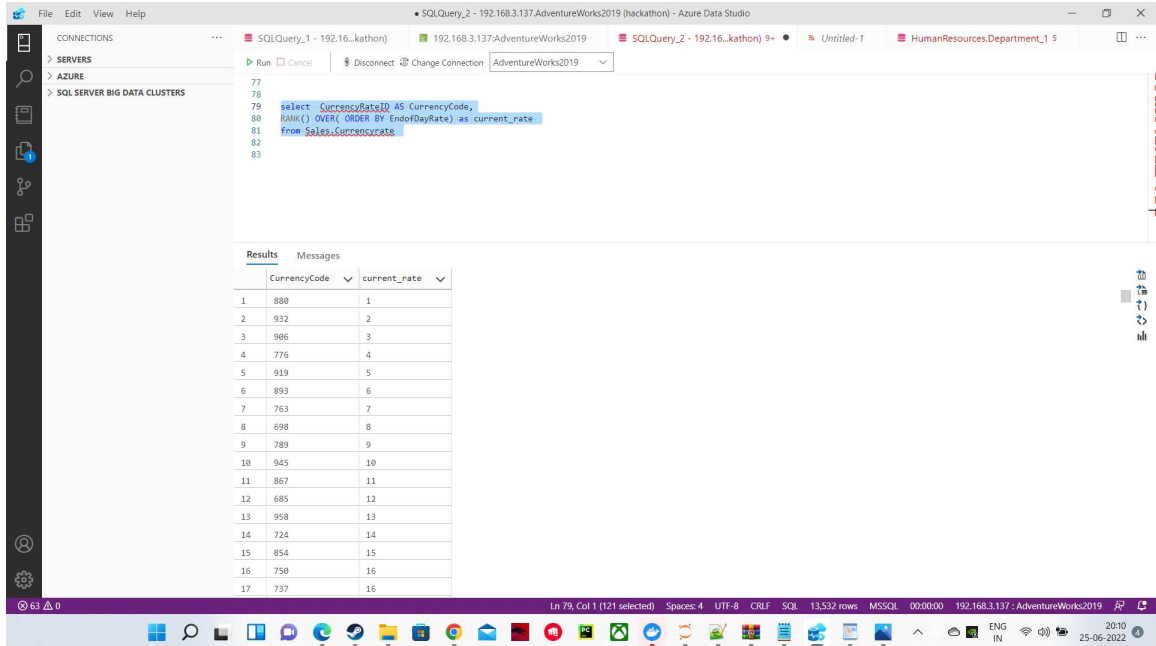
77
78
79 select CurrencyRateID AS CurrencyCode,
80 RANK() OVER( ORDER BY EndOfDayRate) as current_rate
81 from Sales.Currencyrate
82
83

```

Results

Messages

	CurrencyCode	current_rate
1	880	1
2	932	2
3	906	3
4	776	4
5	919	5
6	893	6
7	763	7
8	698	8
9	789	9
10	945	10
11	867	11
12	685	12
13	958	13
14	724	14
15	854	15
16	750	16



#### QUESTION 6:

Select SalesOrderHeader.TerritoryID,Store.Name As  
Store\_Name,MONTH(SalesOrderHeader.OrderDate) as Month,SalesOrderHeader.CustomerID,  
MAX(SalesOrderHeader.TotalDue)

OVER(PARTITION BY SalesOrderHeader.TerritoryID) AS MAX\_Transaction

FROM Sales.SalesOrderHeader As SalesOrderHeader

INNER JOIN Sales.SalesPerson As SalesPerson

On SalesPerson.TerritoryID=SalesOrderHeader.TerritoryID

INNER JOIN Sales.Customer As Customer

On Customer.CustomerID=SalesOrderHeader.CustomerID

INNER JOIN Sales.Store as Store

On Store.BusinessEntityID=Customer.StoreID

INNER JOIN Sales.SalesTerritory As SalesTerritory

ON SalesTerritory.TerritoryID=SalesOrderHeader.TerritoryID

Run Cancel Disconnect Change Connection AdventureWorks2019

```

105 Select SalesOrderHeader.TerritoryID,Store_Name As Store_Name,MONTH(SalesOrderHeader.OrderDate) as Month,SalesOrderHeader.CustomerID,
106 MAX(SalesOrderHeader.TotalDue)
107
108 OVER(PARTITION BY SalesOrderHeader.TerritoryID) AS MAX_Transaction
109 FROM Sales.SalesOrderHeader As SalesOrderHeader
110 INNER JOIN Sales.SalesPerson As SalesPerson
111 On SalesPerson.TerritoryID=SalesOrderHeader.TerritoryID
112 INNER JOIN Sales.Customer As Customer
113 On Customer.CustomerID=SalesOrderHeader.CustomerID
114 INNER JOIN Sales.Store as Store
115 On Store.BusinessEntityID=Customer.StoreID
116 INNER JOIN Sales.SalesTerritory As SalesTerritory
117 ON SalesTerritory.TerritoryID=SalesOrderHeader.TerritoryID

```

Results		Messages			
	TerritoryID	Store_Name	Month	CustomerID	MAX_Transaction
1	1	Capable Sales and Service	5	29898	126852.1615
2	1	Capable Sales and Service	5	29898	126852.1615
3	1	Capable Sales and Service	5	29898	126852.1615
4	1	Latest Sports Equipment	5	29580	126852.1615
5	1	Latest Sports Equipment	5	29580	126852.1615
6	1	Latest Sports Equipment	5	29580	126852.1615
7	1	The Bike Shop	5	29747	126852.1615
8	1	The Bike Shop	5	29747	126852.1615
9	1	The Bike Shop	5	29747	126852.1615
10	1	Basic Bike Company	5	29890	126852.1615
11	1	Basic Bike Company	5	29890	126852.1615
12	1	Basic Bike Company	5	29890	126852.1615
13	1	Great Bikes	5	29497	126852.1615
14	1	Great Bikes	5	29497	126852.1615
15	1	Great Bikes	5	29497	126852.1615
16	1	Fifth Bike Store	5	29606	126852.1615
17	1	Fifth Bike Store	5	29606	126852.1615