SOLUTIONS(Adventureworks 2005 Database)

Q.1

```
SELECT
DISTINCT customer.CustomerID,
contact.Firstname,
contact.LastName,
CONCAT(contact.Firstname, ' ' , contact.LastName) AS fullname,
CASE WHEN contact.title IS NULL THEN 'Dear' ELSE contact.title END AS Addressing_title,
contact.Emailaddress,
contact.phone,
customer.AccountNumber,
customer.CustomerType,
address.city,
(stateprovince.name) AS State,
(countryregion.name) AS Country,
address.AddressLine1 AS AddressLine1,
address.AddressLine2 AS AddressLine2,
COUNT(sales.SalesOrderID) AS number_of_orders,
SUM(sales.totalDue) AS total_amount_of_purchases,
MAX(DATE_TRUNC(sales.OrderDate,DAY)) AS last_order_date

FROM `adwentureworks_db.customer` customer
LEFT JOIN `adwentureworks_db.individual` ind -- to obtain FK for the contact table,LJ to keep all in customer table--
ON customer.CustomerID = ind.CustomerID
LEFT JOIN `adwentureworks_db.contact` contact
ON ind.ContactID = contact.ContactID
LEFT JOIN `adwentureworks_db.customeraddress` ca-- to obtain FK for address table--
ON customer.CustomerID = ca.CustomerID
LEFT JOIN `adwentureworks_db.address` address
ON ca.AddressID = address.AddressID
LEFT JOIN `adwentureworks_db.stateprovince` stateprovince
ON stateprovince.StateProvinceID = address.StateProvinceID
LEFT JOIN `adwentureworks_db.countryregion` countryregion
ON stateprovince.CountryRegionCode = countryregion.CountryRegionCode
LEFT JOIN `adwentureworks_db.salesorderheader` sales
ON Customer.CustomerId = sales.CustomerID
INNER JOIN `adwentureworks_db.salesterritory` salesT
ON customer.TerritoryID = salesT.TerritoryID

WHERE CustomerType = 'I'
GROUP BY 1,2,3,4,5,6,7,8,9,10,11,12,13,14
ORDER BY total_amount_of_purchases DESC
LIMIT 200;
```

Q1.2

```
WITH
table1 AS
(SELECT
DISTINCT customer.CustomerID,
contact.Firstname,
contact.LastName,
CONCAT(contact.Firstname, ' ' , contact.LastName) AS fullname,
CASE WHEN contact.title IS NULL THEN 'Dear' ELSE contact.title END AS Addressing_title,
contact.Emailaddress,
contact.phone,
customer.AccountNumber,
customer.CustomerType,
address.city,
(stateprovince.name) AS State,
(countryregion.name) AS Country,
address.AddressLine1 AS AddressLine1,
address.AddressLine2 AS AddressLine2,
COUNT(sales.SalesOrderID) AS number_of_orders,
SUM(sales.totalDue) AS total_amount_of_purchases,
MAX(DATE_TRUNC(sales.OrderDate,DAY)) AS last_order_date

FROM `adwentureworks_db.customer` customer
LEFT JOIN `adwentureworks_db.individual` ind -- to obtain FK for the contact table,LJ to keep all in customer table--
ON customer.CustomerID = ind.CustomerID
LEFT JOIN `adwentureworks_db.contact` contact
ON ind.ContactID = contact.ContactID
LEFT JOIN `adwentureworks_db.customeraddress` ca-- to obtain FK for address table--
ON customer.CustomerID = ca.CustomerID
LEFT JOIN `adwentureworks_db.address` address
ON ca.AddressID = address.AddressID
LEFT JOIN `adwentureworks_db.stateprovince` stateprovince
ON stateprovince.StateProvinceID = address.StateProvinceID
LEFT JOIN `adwentureworks_db.countryregion` countryregion
ON stateprovince.CountryRegionCode = countryregion.CountryRegionCode
LEFT JOIN `adwentureworks_db.salesorderheader` sales
ON Customer.CustomerId = sales.CustomerID
INNER JOIN `adwentureworks_db.salesterritory` salesT
ON customer.TerritoryID = salesT.TerritoryID

WHERE CustomerType = 'I'
GROUP BY 1,2,3,4,5,6,7,8,9,10,11,12,13,14
ORDER BY total_amount_of_purchases DESC
)

SELECT *
FROM table1
WHERE last_order_date < (SELECT DATE_SUB(MAX(last_order_date), INTERVAL 365 DAY)-- DATE_SUB subtracts 365days from the la
date ---
FROM table1
)
ORDER BY total_amount_of_purchases DESC
LIMIT 200;
```

Q1.3

```
WITH customer_info AS (SELECT
DISTINCT customer.CustomerID,
contact.Firstname,
contact.LastName,
CONCAT(contact.Firstname, ' ' , contact.LastName) AS fullname,
CASE WHEN contact.title IS NULL THEN 'Dear' ELSE contact.title END AS Addressing_title,
contact.Emailaddress,
contact.phone,
customer.AccountNumber,
customer.CustomerType,
address.city,
(stateprovince.name) AS State,
(countryregion.name) AS Country,
address.AddressLine1 AS AddressLine1,
address.AddressLine2 AS AddressLine2,
COUNT(sales.SalesOrderID) AS number_of_orders,
SUM(sales.totalDue) AS total_amount_of_purchases,
MAX(DATE_TRUNC(sales.OrderDate,DAY)) AS last_order_date

FROM `adventureworks_db.customer` customer
LEFT JOIN `adventureworks_db.individual` ind -- to obtain FK for the contact table,LJ to keep all in customer table--
ON customer.CustomerID = ind.CustomerID
LEFT JOIN `adventureworks_db.contact` contact
ON ind.ContactID = contact.ContactID
LEFT JOIN `adventureworks_db.customeraddress` ca-- to obtain FK for address table--
ON customer.CustomerID = ca.CustomerID
LEFT JOIN `adventureworks_db.address` address
ON ca.AddressID = address.AddressID
LEFT JOIN `adventureworks_db.stateprovince` stateprovince
ON stateprovince.StateProvinceID = address.StateProvinceID
LEFT JOIN `adventureworks_db.countryregion` countryregion
ON stateprovince.CountryRegionCode = countryregion.CountryRegionCode
LEFT JOIN `adventureworks_db.salesorderheader` sales
ON Customer.CustomerId = sales.CustomerID
INNER JOIN `adventureworks_db.salesterritory` salesT
ON customer.TerritoryID = salesT.TerritoryID

WHERE CustomerType = 'I'
GROUP BY 1,2,3,4,5,6,7,8,9,10,11,12,13,14
ORDER BY total_amount_of_purchases DESC)


SELECT *,
CASE WHEN DATE(last_order_date) >= DATE_SUB((SELECT MAX(DATE(last_order_date))
FROM customer_info), INTERVAL 365 DAY) AND DATE(last_order_date) < (SELECT
MAX(DATE(last_order_date))
FROM customer_info) THEN 'active'
ELSE 'inactive'
END as Customer_status
FROM customer_info
order by CustomerId DESC
LIMIT 500
```

Q1.4

```
WITH customer_info AS (SELECT
DISTINCT customer.CustomerID,
contact.Firstname,
contact.LastName,
CONCAT(contact.Firstname, ' ' , contact.LastName) AS fullname,
CASE WHEN contact.title IS NULL THEN 'Dear' ELSE contact.title END AS Addressing_title,
contact.Emailaddress,
contact.phone,
customer.AccountNumber,
customer.CustomerType,
address.city,
(stateprovince.name) AS State,
(countryregion.name) AS Country,
address.AddressLine1 AS AddressLine1,
address.AddressLine2 AS AddressLine2,
salesT.Group AS Group_,
COUNT(sales.SalesOrderID) AS number_of_orders,
SUM(sales.totalDue) AS total_amount_of_purchases,
MAX(DATE_TRUNC(sales.OrderDate,DAY)) AS last_order_date


FROM `adwentureworks_db.customer` customer
LEFT JOIN `adwentureworks_db.individual` ind -- to obtain FK for the contact table,LJ to keep all in customer
table--
ON customer.CustomerID = ind.CustomerID
LEFT JOIN `adwentureworks_db.contact` contact
ON ind.ContactID = contact.ContactID
LEFT JOIN `adwentureworks_db.customeraddress` ca-- to obtain FK for address table--
ON customer.CustomerID = ca.CustomerID
LEFT JOIN `adwentureworks_db.address` address
ON ca.AddressID = address.AddressID
LEFT JOIN `adventureworks_db.stateprovince` stateprovince
ON stateprovince.StateProvinceID = address.StateProvinceID
LEFT JOIN `adventureworks_db.countryregion` countryregion
ON stateprovince.CountryRegionCode = countryregion.CountryRegionCode
LEFT JOIN `adventureworks_db.salesorderheader` sales
ON Customer.CustomerId = sales.CustomerID
INNER JOIN `adventureworks_db.salesterritory` salesT
ON customer.TerritoryID = salesT.TerritoryID

WHERE CustomerType = 'I'
GROUP BY 1,2,3,4,5,6,7,8,9,10,11,12,13,14,15
ORDER BY total_amount_of_purchases DESC),

customer_status_help AS(
SELECT *,
CASE WHEN DATE(last_order_date) >= DATE_SUB((SELECT MAX(DATE(last_order_date))
FROM customer_info), INTERVAL 365 DAY) AND DATE(last_order_date) < (SELECT
MAX(DATE(last_order_date))
FROM customer_info) THEN 'active'
ELSE 'inactive'
END as Customer_status
FROM customer_info
order by CustomerId DESC)

Select *,
split(AddressLine1, ' ')[safe_offset(0)] as address_no_new,
CONCAT(split(AddressLine1, ' ')[safe_offset(1)], " ", split(AddressLine1, ' ')[safe_offset(2)]) as address_st
FROM customer_status_help
WHERE Group_= 'North America' AND total_amount_of_purchases= 2500 OR number_of_orders>= 5
```

Q2.1

```
SELECT
(LAST_DAY(DATE_TRUNC(DATE (soh.OrderDate),Month))) AS OrderDate,--lastday returns last day of each month--
sal.CountryRegionCode,
sal.Name AS Region,
COUNT(DISTINCT soh.SalesOrderID) AS TotalOrders,
COUNT(DISTINCT soh.CustomerID) AS Number_Customers,
COUNT(DISTINCT soh.SalesPersonID) AS Number_Sales_Persons,
ROUND (SUM (soh.TotalDue)) AS Total_w_tax -- Add round to remove extra decimal places --
FROM `adventureworks_db.salesorderheader` soh
LEFT JOIN `adventureworks_db.salesterritory` sal
ON soh.TerritoryID = sal.TerritoryID

GROUP BY 1,2,3 -- Always remember to add columns in select to GROUP BY, if they are not aggregated--
```

Q2.2

```
WITH t1 AS
(SELECT
(LAST_DAY(DATE_TRUNC(DATE (soh.OrderDate),Month))) AS OrderDate,--lastday returns last day of each month--
sal.CountryRegionCode,
sal.Name AS Region,
COUNT(DISTINCT soh.SalesOrderID) AS TotalOrders,
COUNT(DISTINCT soh.CustomerID) AS Number_Customers,
COUNT(DISTINCT soh.SalesPersonID) AS Number_Sales_Persons,
ROUND (SUM (soh.TotalDue)) AS Total_w_tax -- Add round to remove extra decimal places --

FROM `adventureworks_db.salesorderheader` soh
LEFT JOIN `adventureworks_db.salesterritory` sal
ON soh.TerritoryID = sal.TerritoryID

GROUP BY 1,2,3)

SELECT *,
SUM(t1.Total_w_tax) OVER (PARTITION BY CountryRegionCode, t1.Region
ORDER BY Total_w_tax) AS Cum_Sum
FROM t1
```

Q 2.3

```
With T1 AS(SELECT
LAST_DAY(DATE(OrderDate), MONTH) AS month_order,
T.CountryRegionCode AS countryregcode,
T.Name AS Region,
COUNT(SalesOrderID) AS num_orders,
COUNT (DISTINCT (CustomerID)) AS num_customers,
COUNT (DISTINCT (SalesPersonID)) AS num_sales_rep,
SUM(TotalDue) AS Total_W_tax

FROM `tc-da-1.adventureworks_db.salesorderheader` S
JOIN `tc-da-1.adventureworks_db.salesterritory` T
ON T.TerritoryID = S.TerritoryID
GROUP BY 1, 2, 3)


Select
month_order, countryregcode, Region,num_orders,num_customers,num_sales_rep,Total_W_tax,
DENSE_RANK() OVER (PARTITION BY countryregcode ORDER BY Total_W_tax DESC) as
sales_rank,
SUM(Total_W_tax) OVER (PARTITION BY Region ORDER BY month_order) AS cum_sum
FROM T1
```

Q 2.4

```
With T1 AS(SELECT
LAST_DAY(DATE(OrderDate), MONTH) AS month_order,
T.CountryRegionCode AS CountryRegionCode,
T.Name AS Region,
COUNT(SalesOrderID) AS num_orders,
COUNT (DISTINCT (CustomerID)) AS num_customers,
COUNT (DISTINCT (SalesPersonID)) AS num_sales_rep,
SUM(TotalDue) AS Total_W_tax

FROM `tc-da-1.adventureworks_db.salesorderheader` S
JOIN `tc-da-1.adventureworks_db.salesterritory` T
ON T.TerritoryID = S.TerritoryID
GROUP BY 1, 2, 3),
T2 AS (SELECT province.CountryRegionCode, AVG(tax.TaxRate) as avg_tax_rate,
ROUND(COUNT(tax.StateProvinceID)/COUNT(province.StateProvinceID), 2) as perc_W_tax
FROM `tc-da-1.adventureworks_db.stateprovince` province
LEFT JOIN `tc-da-1.adventureworks_db.salestaxrate` tax
ON tax.StateProvinceID = province.StateProvinceID
GROUP BY 1)

Select
T1.*,
DENSE_RANK() OVER (PARTITION BY T1.CountryRegionCode ORDER BY Total_W_tax DESC) as
sales_rank,
SUM(Total_W_tax) OVER (PARTITION BY Region ORDER BY month_order) AS cum_sum,
T2.avg_tax_rate, T2.perc_W_tax

FROM T1
JOIN T2
ON T1.CountryRegionCode =T2.CountryRegionCode
```