**Sales Metrics Code Snippets:**

**Refer Database:Adventureworks2019**

Dashboard Dataset:

There are five data set in dashboard Report :

Datasetname:parameter\_Dataset

1. Dropdown Parameter  in Dashboard :

Dataset Used:

**SELECT              territoryid,**

**NAME[Teritoryname],**

**CONCAT(TerritoryID ,+ '- '+Name)as Labeldata**

**FROM   adventureworks2019.sales.salesterritory**

**order by TerritoryID**

**Description:**

Label data gives the Territory Id name + territory id and territory id will be a value for the parameter.

It is used in Two table in Dashboard

    1.Sales Rep table

    2.Top 10 Stores

2.**Dataset name : DashboardGraphData**

**Dataset:**

**select sh.territoryid,sum(TotalDue)GrandTotal,name[Territoryname] from AdventureWorks2019.Sales.SalesOrderHeader sh**

**inner join AdventureWorks2019.Sales.SalesTerritory st on sh.TerritoryID=st.TerritoryID**

**group by sh.TerritoryID,Name**

**order by sh.TerritoryID**

**Decription:**

* this will be used to get territory wise sales Report and represented in graph.
* And by click the value in graph it will navigate to Territory wise customer table

**3.Dataset name: Top performing Customer data**

**Datset:**

SELECT a.Territoryid,

       NAME as Territoryname,

       b.Customerid,

    customername,

       tot as TotalValue

FROM   (SELECT Max(tot)tot,

               territoryid

        FROM   (SELECT territoryid,

                       customerid,

                       Sum(totaldue)AS Tot

                FROM   adventureworks2019.sales.salesorderheader

                GROUP  BY customerid,

                          territoryid)a

        GROUP  BY territoryid)a

       INNER JOIN (SELECT customerid,

                          Sum(totaldue)newtot

                   FROM   adventureworks2019.sales.salesorderheader

                   GROUP  BY customerid)b

               ON a.tot = b.newtot

       INNER JOIN (SELECT customerid,

                          personid,

                          Concat(firstname, + ' '+ lastname)Customername

                   FROM   adventureworks2019.sales.customer cus

                          JOIN adventureworks2019.person.person per

                            ON cus.personid = per.businessentityid)Customername

               ON b.customerid = Customername.customerid

       INNER JOIN (SELECT territoryid,

                          NAME

                   FROM   [AdventureWorks2019].sales.salesterritory)

                  territoryname

               ON a.territoryid = territoryname.territoryid

ORDER  BY a.territoryid

**Description:**

* This will be have  a Hyper link in customer id and it will navigate to **customer sales report**
* It will display Top customer in each territoty

**4.Datset name :Territory wise Top 10 stores**

**Dataset:**

**SELECT func.territoryid,**

**terrname.NAME,**

**func.storeid,**

**func.NAME[StoreName],**

**nofsales [countofsales],**

**salesamount,**

**maxsalesvalue,**

**ranking**

**FROM  (SELECT Max(storetotal)AS maxsalesvalue,**

**storeid**

**FROM   (SELECT territoryid,**

**storeid,**

**Sum(totaldue)storetotal,**

**NAME**

**--count(SalesOrderID)totalsales**

**FROM   adventureworks2019.sales.salesorderheader sh**

**INNER JOIN(SELECT storeid,**

**NAME,**

**customerid**

**FROM   adventureworks2019.sales.store st**

**JOIN adventureworks2019. [Sales].customer**

**ct**

**ON st.businessentityid = ct.storeid)b**

**ON sh.customerid = b.customerid**

**GROUP  BY storeid,**

**territoryid,**

**NAME)c**

**--where StoreID=600**

**GROUP  BY storeid)maxsalesvalue**

**INNER JOIN (SELECT \***

**FROM   (SELECT soh.territoryid,**

**storeid,**

**Sum(totaldue)                   Salesamount,**

**NAME,**

**Count(\*)                        AS Nofsales,**

**Row\_number()**

**OVER(**

**partition BY soh.territoryid**

**ORDER BY Sum(totaldue)DESC) AS Ranking**

**FROM   adventureworks2019.sales.salesorderheader soh**

**INNER JOIN adventureworks2019.sales.customer**

**cus**

**ON soh.customerid = cus.customerid**

**INNER JOIN adventureworks2019.sales.store st**

**ON cus.storeid = st.businessentityid**

**GROUP  BY soh.territoryid,**

**storeid,**

**NAME)A**

**WHERE  ranking <= 10)func**

**ON maxsalesvalue.storeid = func.storeid**

**INNER JOIN (SELECT \***

**FROM   adventureworks2019.sales.salesterritory)terrname**

**ON terrname.territoryid = func.territoryid**

**where func.territoryid in(@P\_Territoryid)**

**ORDER  BY func.territoryid ,ranking**

**Description**

* This query will select top  10 stores in each territory
* And it use territory  has a input parameter display selected terriory wise top 10 stores  rep only.
* And top 5 customer per territory will be highlighted in different colors .

**5.DatSet name :Territory Wise Top 5 sales Rep**

**Data set name :Sales Person Data**

Data Set:

SELECT topsalesperson.Territoryid,

       Teritoryname,

       Topsalesperson.salespersonid,

   TotalSales,

       salespersonname,

       totalddue,

       due[Max\_salesvalue],

   Ranking

FROM   (SELECT \*

        FROM   (SELECT soh.salespersonid,

                       Sum(totaldue)                   Totalddue,

                       Count(salesorderid)             TotalSales,

                       soh.territoryid,

                       Row\_number()

                         OVER(

                           partition BY soh.territoryid

                           ORDER BY Sum(totaldue)DESC) AS Ranking

                FROM   adventureworks2019.sales.salesperson peer

                       INNER JOIN adventureworks2019.sales.salesorderheader soh

                               ON peer.businessentityid = soh.salespersonid

                GROUP  BY soh. salespersonid,

                          soh.territoryid)A

        WHERE  ranking <= 5)topsalesperson

       INNER JOIN (SELECT salespersonid,

                          Max(totalddue)AS due

                   FROM   (SELECT soh.salespersonid,

                                  Sum(totaldue)      Totalddue,

                                  Count(salesorderid)TotalSales1,

                                  soh.territoryid

                           FROM   adventureworks2019.sales.salesperson peer

                                  INNER JOIN

                                  adventureworks2019.sales.salesorderheader

                                  soh

                                          ON

                                  peer.businessentityid = soh.salespersonid

                           GROUP  BY soh. salespersonid,

                                     soh.territoryid)B

                   GROUP  BY salespersonid) Maxvalue

               ON Maxvalue.salespersonid = topsalesperson.salespersonid

       INNER JOIN (SELECT sp.businessentityid,

                          Concat(firstname, ' ' + lastname)AS Salespersonname

                   FROM   adventureworks2019.sales.salesperson sp

                          INNER JOIN adventureworks2019.person.person pers

                                  ON sp.businessentityid = pers.businessentityid

                  )

              salespersonname

               ON topsalesperson.salespersonid =

                  salespersonname.businessentityid

       INNER JOIN (SELECT territoryid,

                          NAME[Teritoryname],

  CONCAT(TerritoryID ,+ '- '+Name)as Labeldata

                   FROM   adventureworks2019.sales.salesterritory)T\_name

               ON topsalesperson.territoryid = T\_name.territoryid

where topsalesperson.TerritoryID in(@P\_Territoryid)

ORDER  BY topsalesperson.territoryid,Totalddue desc

**Description :**

* This query will give   top 5 sales rep in each territory
* And it use territory  has a input parameter display selected territory wise sales rep only.

**Navigation Reports :**

Dataset name: Territory wise customer :

Datset:

SELECT Customer.customerid,

       customername,

       postalcode,

       city,

       countryname,

       totalvalue

FROM   (SELECT customerid,

               Sum(totaldue) AS Totalvalue

        FROM   adventureworks2019.sales.salesorderheader

        WHERE  territoryid = @P\_Territoryid

        GROUP  BY customerid)Customer

       Left JOIN (SELECT customername,

                          postalcode,

                          businessentityid,

                          countryname,

                          city

                   FROM   (SELECT businessentityid,

                                  Concat(firstname, + '  ' + lastname)AS

                                  customername,

                                  postalcode,

                                  city,

                                  stateprovinceid

                           FROM   adventureworks2019.person.person ps

                                  INNER JOIN adventureworks2019.person.address

                                             ad

                                          ON ps.businessentityid = ad.addressid)

                          b

                          INNER JOIN (SELECT sp.countryregioncode,

                                             stateprovinceid,

                                             cr.NAME AS countryname

                                      FROM

                          adventureworks2019.person.stateprovince sp

                          INNER JOIN

       adventureworks2019.person.countryregion cr

                                                     ON

       sp.countryregioncode = cr.countryregioncode)

       countryname

       ON countryname.stateprovinceid = b.stateprovinceid)country

               ON country.businessentityid = Customer.customerid

**Description:**

* This report displays all the customer in the selected territory
* It use link from previous graph report
* This report has hyper link ,which will navigate to next report i.e Customer sales report

Navigation Report Level 2:

**Data set name :Customer sales data**

Data set : select customerid,SalesOrderID,OrderDate,ShipDate,TaxAmt,Freight,TotalDue from AdventureWorks2019.Sales.SalesOrderHeader

where CustomerID=@P\_customerid

order by OrderDate

**Description:**

* This query will give the sales order in a selected customer id
* This use customer id has input and it is hidden parameter
* And use Running total on total due.