```
CL:CHGLIBL LIBL(QTEMP COMSQLQRY SYSTOOLS QGPL);
CL:CHGJOB LANGID(DEU) CNTRYID(DE) CCSID(1141);

--+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
-+-+-+-
-- Group By, Having and Multi-dimensional Grouping
--+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
-+-+-+-

--
***********************************************************************************************
*****
-- Basic SELECT-Statement
--
***********************************************************************************************
*****
Select * from AddressX;

Select CustNo, CustName1, Street, ZipCode, City
  From AddressX;

Select *
   From OrderHdrx
   Where     DelDate   >= Current_Date - 8 years
         and OrderType  = 'DO';

Select *
  From AddressX
  Where City = 'Frankfurt'
  Order By CustName1, CustName2, CustNo;

--
***********************************************************************************************
*****
-- GROUP BY / HAVING
--
***********************************************************************************************
******
Select * from Library_List_Info;

Select * from Sales;

Select CustNo, Sum(Amount), Avg(Amount)
  From Sales
  Where SalesDate between '2009-01-01' and '2009-12-31'
  Group By CustNo
--  Order By CustNo
  ;

Select Year(SalesDate) as SalesYear , CustNo,
       Sum(Amount)     as Total,      Count(*) as Positions
   From Sales
   Group By Year(SalesDate), CustNo
   Order By SalesYear, CustNo;

-- Select Year(SalesDate) as SalesYear , CustNo,
--        Sum(Amount)     as Total,      Count(*) as Positions
--    From Sales
--    Group By SalesYear, CustNo
--    Order By SalesYear, CustNo;

-- With Where Conditions
Select Year(SalesDate) as SalesYear , CustNo,
       Sum(Amount)     as Total,      Count(*) as Positions
   From Sales
   Where CustNo between '10002' and '10004'
```

```
   Group By Year(SalesDate), CustNo
   Order By SalesYear, CustNo;


-- Using multiple Aggregate Functions
Select CustNo, Count(*)    "NbrRows", Sum(Amount) "Total" ,
               Cast(Avg(Amount) as Dec(11, 2))    "Average",
               Min(Amount) "Minimum", Max(Amount) "Maximum"
   From Sales
   Group By CustNo
   Order By CustNO;

-- Multiple Aggregate Functions / Grouping Expression / Where condition / Different sort
sequence
Select Year(SalesDate) SalesYear, CustNo,
       Count(*)    "NbrRows", Sum(Amount) "Total" ,
       Cast(Avg(Amount) as Dec(11, 2))    "Average",
       Min(Amount) "Minimum", Max(Amount) "Maximum"
   From Sales
   Where ItemNo Between '5100' and '5300'
   Group By Year(SalesDate), CustNo
   Order By CustNO, SalesYear;

--
================================================================================================
=====
-- Aggregate Functions and DISTINCT
--
================================================================================================
=====
-- Different Customers
---------------------------
Select * from Sales;

Select CustNo
  From Sales
  Group By CustNo;

Select Distinct CustNo
  From Sales;

Select Count(CustNo)
From Sales;

-- Number of different customers
Select Count(Distinct CustNo) "Nbr Customers"
  From Sales;

-- Number of different compositions CustNo, ItemNo
-----------------------------------------------------
Select Distinct CustNo, ItemNo
  From Sales
  Order By CustNo, ItemNo;


Select Count(Distinct CustNo, ItemNo)
  FRom Sales;

Select Count(Distinct CustNo concat ItemNo) "NbrCustItem"
  From Sales;


-- Distinct Numbers per year
--------------------------------
Select Year(SalesDate)                      "SalesYear",
       Count(*)                             "Nbr Positions",
```

```
       Count(Distinct CustNo)               "Nbr Customer",
       Count(Distinct ItemNo)               "Nbr Items",
       Count(Distinct CustNo concat ItemNo) "Nbr Customer/Items"
  From Sales
  Group By Year(SalesDate)
  Order By "SalesYear"
  ;
--
================================================================================================
=====
-- NULL Values and Aggredate Functions
--
================================================================================================
=====
Select *
  From NULLFile;

Select MyChar concat MyGraph2 "ConcatCols", a.*
  from NULLFile a;

Select MyChar, MyGraph2
  from NULLFile
  Where MyChar = MyGraph2;

Select MyChar, MyGraph2
  from NULLFile
  Where     MyChar = MyGraph2
        or (MyChar is NULL and MyGraph2 is NULL);

Select Count(*)      "All",
       Count(MyInt)  "Count MyInt",  Count(MyDec)    "Count MyDec",
       Count(MyChar) "Count MyChar", Count(MyGraph2) "Count MyGraph2"
  From NULLFile;

-- Determine Average Delivery Quantity
Select * --ItemNo, DelQty
From OrderDetx
Order By ItemNo;

Select Company, ItemNo, Avg(DelQty)
from OrderDetX
Where DelQty > 0
Group By Company, ItemNo;

-- Attention: DelQty defined as Integer!
Select Company, ItemNo,
       Avg(DelQty)            "Incl.*Zero DelQty",
       Avg(NullIf(DelQty, 0)) "*Zero DelQty = NULL",
       Avg(OrderQty)          "Order Quantity"
  From OrderDetX
  Group By Company, ItemNo
--  Having Avg(DelQty) <> Avg(NullIf(DelQty, 0))
  ;

-- Syntax Alternatives
Select Company, ItemNo,
       Avg(DelQty)                        "Int.*Zero DelQty",
       Avg(NullIf(DelQty, 0))             "Int. DelQty = NULL",
       Avg(Dec(DelQty, 11, 0))            "Incl.*Zero DelQty",
       Avg(NullIf(Dec(DelQty, 11, 0), 0)) "*Zero DelQty = NULL"
  From OrderDetX
  Group By Company, ItemNo
  Having Avg(DelQty) <> Avg(NullIf(DelQty, 0))
  ;

With x as (Select Company, Trim(ItemNo) ItemNo,
```

```
                Cast(DelQty            as Dec(11, 0)) DelQty,
                Cast(NulliF(DelQty, 0) as Dec(11, 0)) DelQtyNULL
           From OrderDetX)
Select Company, ItemNo,
       Sum(DelQty)                        "Sum DelQty",
       Sum(DelQtyNull)                    "Sum DelQtyNULL",
       Cast(Avg(DelQty) as Dec(11, 2))    "Avg DelQty",
       Cast(Avg(DelQtyNUll) as Dec(11, 2)) "Avg DelQtyNULL"
  From x
  Group By Company, ItemNo
  Having Avg(DelQty) <> Avg(DelQtyNULL)
  ;

 Select Company, OrderNo,
        Count(*)    "TotalPos",  Count(NULLIf(DelQty, 0)) "PosWithDelQty",
        Avg(DelQty) "AvgDelQty", Avg(NULLIf(DelQty, 0))   "AvgDelQty > 0"
   From OrderDetX
   Group By Company, OrderNo
   ;

--
================================================================================================
=====
-- HAVING conditions
--
================================================================================================
=====
-- Determine all orders with more than 4 Positions
 Select Company, OrderNo, Count(*) Positions
   from OrderDetx
   Group By Company, OrderNo
   Having Count(*) > 4;

-- Determine all customers with an annual amout > 3000 and more than 10 positions
Select Year(SalesDate) SalesYear, CustNo, Sum(Amount) Total, Count(*)
  From Sales
  Group By Year(SalesDate), CustNo
  Having     Sum(Amount) > 3000
         and Count(*)    > 10;

--
**********************************************************************************************
*****
-- Multidimensional Grouping 6.1
--
**********************************************************************************************
*****
-- RollUp
--
================================================================================================
=====
Select * from Sales;

-- Sales / Year and Customer
Select Year(SalesDate) SalesYear, CustNo, Sum(Amount) Total
   From Sales
   Where CustNo In ('10001', '10003')
   Group By Year(SalesDate), CustNo
   Order By SalesYear, CustNo
;

Select Year(SalesDate) SalesYear, CustNo, Sum(Amount) Total
   From Sales
   Where CustNo In ('10001', '10003')
   Group By RollUp(Year(SalesDate), CustNo)
   Order By SalesYear, CustNo
```

```
;

-- Performance Rollup versus Group By
--    Select Year(SalesDate) SalesYear, CustNo, Sum(Amount) Total
--      From Sales
--      Group By Year(SalesDate), CustNo
--  Union All
--    Select Year(SalesDate), Cast(NULL as Char(15)), Sum(Amount)
--      From Sales
--      Group By Year(SalesDate)
--  Union All
--    Select Cast(NULL as Dec(4, 0)), Cast(Null as Char(15)), Sum(Amount)
--      From Sales
--  Order By 1, 2;

-- Sales / Year/Month/Day
---------------------------------
select Year(SalesDate) SalesYear, Month(SalesDate) SalesMonth,
       Day(SalesDate) SalesDay,   Sum(Amount) Total
From Sales
Where salesDate between '2008-10-01' and '2009-03-31'
Group By Year(SalesDate), Month(SalesDate), Day(SalesDate)
Order By SalesYear, SalesMonth, SalesDay
;

select Year(SalesDate) SalesYear, Month(SalesDate) SalesMonth,
       Day(SalesDate) SalesDay,   Sum(Amount) Total
From Sales
Where salesDate between '2008-10-01' and '2009-03-31'
Group By RollUp(Year(SalesDate), Month(SalesDate), Day(SalesDate))
Order By SalesYear, SalesMonth, SalesDay
;

--
================================================================================================
=====
-- Cube
--
================================================================================================
=====

select Year(SalesDate) SalesYear, CustNo, Item, Sum(Amount) Total
From Sales
where     SalesDate between '2008-10-01' and '2009-03-31'
      and CustNo in ('10001', '10003')
Group By Cube(Year(SalesDate), CustNo, Item)
--Order By 1, 2, 3
Order By SalesYear, CustNo, Item
;

--
================================================================================================
=====
-- Grouping Sets
--
================================================================================================
=====
 select Year(SalesDate) SalesYear, CustNo, ItemNo, Sum(Amount) Total
    From Sales
    Where CustNo between 10001 and 10003
    Group By Grouping Sets((Year(SalesDate), CustNo),
                           (Year(SalesDate), ItemNo),
                           (Year(SalesDate)), ())
 Order By SalesYear, CustNo, ItemNo
;
```

```
--
================================================================================================
=====
-- Aggregate Function Grouping
--
================================================================================================
=====
Select * from SalesV01
Order By SalesYear, custNo, ItemNo;

Select SalesYear, CustNo, Sum(Total) Total,
       Grouping(SalesYear), Grouping(CustNo)
  from SalesV01
  Group By RollUp(SalesYear, CustNo)
  Order By SalesYear, custNo;

Select -- Case When Grouping(SalesYear) = 1 Then 'Grand Total'
       --     When Grouping(CustNo)    = 1 Then 'Total Year' concat SalesYear
       --     Else '' End SummaryText,
       SalesYear, CustNo, Sum(Total) Total,
       Grouping(SalesYear), Grouping(CustNo)
  from SalesV01
  Group By RollUp(SalesYear, CustNo)
  Order By SalesYear, custNo;

Select Case When Grouping(SalesYear) = 1
            Then 'Grand Total'
            When Grouping(CustNo)    = 1
            Then 'Total Year ' concat SalesYear
            Else '' End,
      SalesYear, CustNo, Sum(Total) as Total
   from SalesV01
   Group By Rollup(SalesYear, CustNo)
--   Having Grouping(CustNo) = 1
   Order By SalesYear, CustNo
;

Select Case When     Grouping(SalesYear) = 1
                 and Grouping(CustNo)    = 1
            Then 'Grand Total'
            When Grouping(CustNo)        = 1
            Then 'Total Year ' concat SalesYear
            When Grouping(SalesYear)     = 1
            Then 'Total Customer ' concat CustNo
            Else '' End,
      SalesYear, CustNo, Sum(Total) as Total
   from SalesV01
   Group By Cube(SalesYear, CustNo)
--  Having     Grouping(CustNo)    >= 0
--         and Grouping(SalesYear) >= 0
   Order By SalesYear, CustNo;


--
================================================================================================
=====
-- Case Clauses, Group By and Aggregate Functions
--
================================================================================================
=====
Select * from OrderDetX;

-- Display different values depending on a column value
Select Status, Case Status When 'CP' Then 'Completed'
                           When 'CL' Then 'Cancelled'
                           When 'EN' Then 'Entered'
```

```
                            When 'PD' Then 'Partly Delivered'
                            Else 'Unknown Status' End,
       a.*
    From OrderDetX a;

-- Merge Values from different columns
Select OrderQty, DelQty, Status,
       Case Status When 'CP' Then DelQty
                   When 'CL' Then OrderQty
                   When 'EN' Then OrderQty
                   When 'PD' Then DelQty
                   Else 0 End as Quantity,
        a.*
   From OrderDetX a;

Select OrderQty, DelQty, Status,
       Case When Status in ('CP', 'PD') Then DelQty
            When Status in ('CL', 'EN') Then OrderQty
            Else 0 End as Quantity,
       a.*
   From OrderDetX a;
-------------------------------------------------------------------------------------------------------
-------
-- Pivot Table
-------------------------------------------------------------------------------------------------------
-------
Select * from Sales;

-- Pivot-Table
Drop view HScommon10/MonSum;

-- Create or Replace view Hscommon10.Monsum as

Select Trim(CustNo) CustNo, Year(SalesDate) SalesYear,
    sum(case when Month(SalesDate)=  1 then Amount else 0 end) as Jan,
    sum(case when Month(SalesDate)=  2 then Amount else 0 end) as Feb,
    sum(case when Month(SalesDate)=  3 then Amount else 0 end) as Mar,
    sum(case when Month(SalesDate)=  4 then Amount else 0 end) as Apr,
    sum(case when Month(SalesDate)=  5 then Amount else 0 end) as May,
    sum(case when Month(SalesDate)=  6 then Amount else 0 end) as Jun,
    sum(case when Month(SalesDate)=  7 then Amount else 0 end) as Jul,
    sum(case when Month(SalesDate)=  8 then Amount else 0 end) as Aug,
    sum(case when Month(SalesDate)=  9 then Amount else 0 end) as Sep,
    sum(case when Month(SalesDate)= 10 then Amount else 0 end) as Oct,
    sum(case when Month(SalesDate)= 11 then Amount else 0 end) as Nov,
    sum(case when Month(SalesDate)= 12 then Amount else 0 end) as Dec,
    sum(Amount) as Total
   from  Sales
   group by CustNo, Year(SalesDate)
  order by CustNo, SalesYear
;

Commit;

Select * from monsum
where SalesYear = 2010;
-------------------------------------------------------------------------------------------------------
-------
-- ABC Customer
-------------------------------------------------------------------------------------------------------
-------
Select CustNo, Case When Sum(Amount) > 10000 Then 'A'
                    When Sum(Amount) >  4000 Then 'B'
                    Else 'C' End "ABC",

               Case When Sum(Amount) > 10000
```

```
                    then Sum(Amount) Else 0 End  "A Customer",
               Case When Sum(Amount) between 4000 and 10000
                    Then Sum(Amount) Else 0 End  "B Customer",
               Case When Sum(Amount) < 4000
                    Then Sum(Amount) Else 0 End  "C Customer"
  From Sales
  Group by CustNo
  Order By Sum(Amount) Desc;

--+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
-+-+-+-
-- OLAP - Specifications
--+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
-+-+-+-
-- Windows Order Clause
-----------------------------------------------------------------------------------------------
-------
Select a.* from SalesCustY a
Where SalesYear = 2009
;

Select Row_Number() Over() RowNbr,
       CustNo, Amount
  From SalesCusty
  Where SalesYear = 2009
--  Order By CustNo
--  Order By Amount Desc
;

Select Row_Number() over(Order By Amount Desc) RowNbr,
       CustNo, Amount
  From SalesCustY
  Where SalesYear = 2009
  Order By Amount Desc
--  Order By CustNo
  ;

Select Row_Number() over(Order By CustNo) RowNbr,
       CustNo, Amount
  From SalesCustY
  Where SalesYear = 2009
--  Order By Amount Desc
  Order By CustNo
  ;


-- Select * from Salesx;
--
-- Create View COMSQLQRY.SalesCustY
-- as (Select Year(SalesDate) SalesYear, CustNo, Sum(Amount) Amount
--       from SalesX
--       Group By Year(SalesDate), custNo);
--
-- Commit;

Select * From SalesEmp
Where SalesYear = 2008
order By Amount Desc;

 Select  Employee, Amount, Row_Number() Over(Order By Amount Desc)
   From SalesEmp
   Where SalesYear = 2008
   Order By Amount Desc;

 Select Employee, Amount,
        Rank()       Over(Order By Amount Desc) Rank,
```

```
        Dense_Rank() Over(Order By Amount Desc) DenseRank
    from SalesEmp
    Where SalesYear = 2008
    Order By Amount Desc, Employee
    ;

With Year2008 as (Select  Employee, Amount Amount2008,
                          Rank() Over(Order By Amount Desc) as Rank2008
                  From  SalesEmp
                  Where SalesYear = 2008),
     Year2009 as (Select  Employee, Amount Amount2009,
                          Rank() Over(Order By Amount Desc) as Rank2009
                  From  SalesEmp
                  Where SalesYear = 2009)

Select Coalesce(a.Employee, b.Employee) AllEmployees,
       Amount2008, Rank2008, Amount2009, Rank2009,
       case When Rank2009 = Rank2008 then ' = '
            When Rank2009 > Rank2008 then ' - '
            When Rank2009 < Rank2008 Then ' + '
            When Rank2008 is NULL     and Rank2009 is Not NULL Then ' ++'
            When Rank2008 is Not NULL and Rank2009 is NULL Then ' --'
       End as Progress
from year2008 a full join year2009 b on a.employee = b.employee
Order By AllEmployees
;

-------------------------------------------------------------------------------------------------
-------
-- Partition By Clause
-------------------------------------------------------------------------------------------------
-------
Select * FRom SalesCusty
Order By SalesYear, CustNo;

Select SalesYear, Dense_Rank() Over(Order By Amount Desc) Rank, a.*
  from SalesCusty a
--  Order By SalesYear, CustNo
  Order By SalesYear, Amount Desc
  ;

Select SalesYear, Dense_Rank() Over(Partition By SalesYear Order By Amount Desc) Rank, a.*
  from SalesCusty a
--  Order By SalesYear, CustNo
  Order By SalesYear, Amount Desc
  ;

 Select SalesYear,
        Dense_Rank() Over(Partition By SalesYear Order By Amount Desc) Rank,
        Employee, Amount
   From SalesEmp
   Order BY SalesYear, Amount Desc;

 Select SalesYear,
        Dense_Rank() Over(Partition By SalesYear
                          Order By Amount Desc) Rank,
        Employee, Amount
   From SalesEmp
   Order By SalesYear, Amount Desc;
--   Order BY Employee, SalesYear, Amount Desc
   ;

-------------------------------------------------------------------------------------------------
-------
-- Rank 2-4 with Sub-Select or CTE
-------------------------------------------------------------------------------------------------
```

```
-------
Select *
   from (Select a.*, Dense_Rank() Over(Order By Amount Desc) as DenseRank
            from SalesEmp a
            Where SalesYear = 2009) as x
Where DenseRank between 2 and 4
;


With x as (Select a.*,
                  Dense_Rank() Over(Order By Amount Desc) as Rang
              From SalesEmp a
              Where SalesYear = 2009)
Select * from x
  where
Rang between 2 and 4
;

-------------------------------------------------------------------------------------------------------
-------
-- Create Views if not yet exist
-------------------------------------------------------------------------------------------------------
-------
-- Create Or Replace View COMSQLQRY/SalesVWYearMonth as
-- Select  Year(SalesDate) SalesYear, Month(SalesDate) SalesMonth, CustNo, ItemNo, --Item,
--         Sum(Amount) Total, Min(Amount) Minimum, Max(Amount) Maximum, Cast(Avg(Amount) as
Dec(11, 4)) Average, Count(*) NbrPos
--   From Salesx
--   Group By Year(SalesDate), Month(SalesDate), CustNo, ItemNo --Item
--   ;
--
-- Create Or Replace View COMSQLQRY/SalesVWYear as
-- Select  Year(SalesDate) SalesYear, CustNo, ItemNo, --- Item,
--         Sum(Amount) Total, Min(Amount) Minimum, Max(Amount) Maximum, Cast(Avg(Amount) as
Dec(11, 4)) Average, Count(*) NbrPos
--   From SAlesx
--   Group By Year(SalesDate), CustNo, ItemNo--, Item
--   ;
--
-- Commit;
;
--
=============================================================================================
=====
-- Ordered OLAP Specifications
--
=============================================================================================
=====
Select * from SalesVWYear;

Select      SalesYear, Amount,
            Rank()         Over(Partition by SalesYear Order By Amount Desc)
as "Rank",
            Dense_Rank()   Over(Partition By SalesYear Order By Amount Desc)
as "Dense_Rank",
            Ntile(3)       Over(Partition By SalesYear Order By Amount Desc)
as "NTile",
       Cast(Cume_Dist()    Over(Partition By SalesYear Order By Amount Desc) as Dec(7, 5))
as "Cume_Dist",
       Cast(Percent_Rank() Over(Partition By SalesYear Order By Amount Desc) as Dec(7, 5))
as "Percent_Rank"
 --      , a.*
 from SalesEmp a
 Order By SalesYear, Amount Desc;

-------------------------------------------------------------------------------------------------
```

```
-------
 -- LAG / LEAD
-------------------------------------------------------------------------------------------------------
-------
-- Determine previous and following row values
  Select * from SalesVWYear;

  Select SalesYear, CustNo, ItemNo, Item, Total,
         Lag(Total)  Over(Order By Total Desc)  as "Previous",
         Lead(Total) Over(Order By Total Desc)  as "Next"
     from SalesVWYear a
     Where SalesYear = 2009 and ItemNo = '5200'
     Order By SalesYear, Total Desc
 ;

   Select SalesYear, CustNo, ItemNo, Item, Total,
         Lag(Total,  3)             Over(Order By Total Desc)  as "Previous 3",
         Lead(Total, 2, -999999,99) Over(Order By Total Desc)  as "Next 2"
     from SalesVWYear a
     Where SalesYear = 2009 and ItemNo = '5200'
     Order By SalesYear, Total Desc
--     Order By CustNo
     ;

  Select SalesYear, CustNo, ItemNo, Item, Total,
         Lag(Total)  Over(Partition By SalesYear Order By Total Desc)  as "Previous",
         Lead(Total) Over(Partition By SalesYear Order By Total Desc)  as "Next"
     from SalesVWYear a
     Where ItemNo = '5200'
     Order By SalesYear, Total Desc
--     Order By SalesYear, CustNo
;

 -- Caluclate differences (incl. Partition By)
 ----------------------------------------
  Select * from Comsqlqry.Salescustyear;

  -- 1. For a single year
    Select  SalesYear, Sales,
          Lag(Sales)   Over(Order By Sales Desc)                  as "Previous",
          Lead(Sales)  Over(Order By Sales Desc)                  as "Next",
          Lag(Sales)   Over(Order By Sales Desc) - Sales          as "DiffPrv",
          Sales - Lead(Sales) Over(Order By Sales Desc)           as "DiffNext",
          Lag(Sales, 1, 0)    Over(Order By Sales Desc) - Sales   as "DiffPrvDft",
          Sales - Lead(Sales, 1, 0) Over(Order By Sales Desc)     as "DiffNextDft"
    --      , a.*
 from SalesCustYear a
 Where SalesYear = 2009
 Order By SalesYear, Sales Desc;

   -- 2. For a multiple years
 Select  SalesYear, Sales,
          Lag(Sales)   Over(Partition By SalesYear Order By Sales Desc)                  as
"Previous",
          Lead(Sales)  Over(Partition By SalesYear Order By Sales Desc)                  as
"Next",
          Lag(Sales)   Over(Partition By SalesYear Order By Sales Desc) - Sales          as
"DiffPrv",
          Sales - Lead(Sales) Over(Partition By SalesYear Order By Sales Desc)           as
"DiffNext",
          Lag(Sales, 1, 0)    Over(Partition By SalesYear Order By Sales Desc) - Sales   as
"DiffPrvDft",
          Sales - Lead(Sales, 1, 0) Over(Partition By SalesYear Order By Sales Desc)     as
"DiffNextDft"
    --      , a.*
 from SalesCustYear a
```

```
 Order By SalesYear, Sales Desc;

 -- 3. Including Total
 With x as ( Select  SalesYear, Sales,
                     Lag(Sales)   Over(Partition By SalesYear Order By Sales Desc)
as "Previous",
                     Lead(Sales)  Over(Partition By SalesYear Order By Sales Desc)
as "Next",
                     Lag(Sales)   Over(Partition By SalesYear Order By Sales Desc) - Sales
as "DiffPrv",
                     Sales - Lead(Sales) Over(Partition By SalesYear Order By Sales Desc)
as "DiffNext",
                     Lag(Sales, 1, 0)    Over(Partition By SalesYear Order By Sales Desc) -
Sales   as "DiffPrvDft",
                     Sales - Lead(Sales, 1, 0) Over(Partition By SalesYear Order By Sales Desc)
as "DiffNextDft"
                     --      , a.*
                from SalesCustYear a)
 Select SalesYear,
        Case When Grouping(Sales) = 1
             then Sum(Sales)
             Else Sales
             End                        as Sales,
        "Previous", "Next", "DiffPrv", "DiffNext", "DiffPrvDft", "DiffNextDft"
   from x
   Group By Grouping Sets((SalesYear, Sales, "Previous", "Next", "DiffPrv", "DiffNext",
"DiffPrvDft", "DiffNextDft"),
                          (SalesYear));

 -- LAG/LEAD with SALESEMP
-- Select * from SalesEmp;

 -- 1. For a single year
--    Select  SalesYear, Amount,
--            Lag(Amount)  Over(Order By Amount Desc)                 as "Previous",
--            Lead(Amount) Over(Order By Amount Desc)                 as "Next",
--            Lag(Amount)  Over(Order By Amount Desc) - Amount        as "DiffPrv",
--            Amount - Lead(Amount) Over(Order By Amount Desc)        as "DiffNext",
--            Lag(Amount, 1, 0)  Over(Order By Amount Desc) - Amount  as "DiffPrvDft",
--            Amount - Lead(Amount, 1, 0) Over(Order By Amount Desc)  as "DiffNextDft"
--            , a.*
--  from SalesEmp a
--  Where SalesYear = 2009
--  Order By SalesYear, Amount Desc;

 -- 2. For Multiple Years: Over(Partition By AND Order By)
--    Select  SalesYear, Employee, Amount,
--           Dense_Rank() Over(Partition By SalesYear Order By Amount Desc)  as "DenseRank",
--           Lag(Amount)  Over(Partition by SalesYear Order By Amount Desc)  as "Previous",
--           Lead(Amount) Over(Partition By SalesYear Order By Amount Desc)  as "Next",
--           Lag(Amount)  Over(Partition by SalesYear Order By Amount Desc) - Amount  as
"DiffPrv",
--           Amount - Lead(Amount) Over(Partition By SalesYear Order By Amount Desc)  as
"DiffNext"
--           , a.*
--  from SalesEmp a
--  Order By SalesYear, Amount Desc;

 -- Calculate sales differences between 3 diffent years
   With x as (Select Year(SalesDate) SalesYear, CustNo, Sum(Amount) Total
                From Sales
                Group By Year(SalesDate), CustNo),
        y as (Select SalesYear, CustNo,
                     Lag(Total, 2, 0) Over(Partition By CustNo Order By SalesYear)  as
PrvPrvYear,
                     Lag(Total, 1, 0) Over(Partition By CustNo Order By SalesYear)  as PrvYear,
```

```
                    Total
                from x)

       Select SalesYear, CustNo,
              PrvPrvYear "2008", PrvYear "2009", Total   "2010",
              Total - PrvPrvYear   "2010 - 2008",
              Total - PrvYear      "2010 - 2009",
              PrvYear - PrvPrvYear "2009 - 2008"
         From y
         Where SalesYear = 2010
         Order By CustNo, Salesyear
 ;

-- Multiple Aggregate Functions: Attention Order By Total for Average! to get the same sequence
as for Total!!!
  Select SalesYear, CustNo, ItemNo, Item, Total,
         Lag(Total)    Over(Order By Total Desc)  as "Previous",
         Lead(Total)   Over(Order By Total Desc)  as "Next",
         Average,
         Lag(Average)  Over(Order By Total Desc)  as "Prev.Avg",
         Lead(Average) Over(Order By Total Desc)  as "Next Avg"
     from SalesVWYear a
     Where SalesYear = 2009 and CustNo = '10001'
--     Order By SalesYear, Total Desc
     Order By CustNo, SalesYear, Total Desc
 ;

-------------------------------------------------------------------------------------------------
-------
-- Quantile NTILE(X) - Example
-------------------------------------------------------------------------------------------------
-------
 Select SalesYear, Amount,
        Ntile(2) Over(Order By Amount Desc) Quantile2,
        Ntile(5) Over(Order By Amount Desc) Quantile5
 From SalesEmp
 Where SalesYear = 2009;

  Select SalesYear, Amount,
        Ntile(2) Over(Partition By SalesYear Order By Amount Desc) Quantile2,
        Ntile(5) Over(Partition By SalesYear Order By Amount Desc) Quantile5
 From SalesEmp
 Order By SalesYear, Amount Desc;


-------------------------------------------------------------------------------------------------
-------
 -- Cume_Dist() / Percent_Rank()

-------------------------------------------------------------------------------------------------
-------
 -- / Number of Rows        --> 2 / 5 = 0,6
 -- Percent Rank: (Current Dense Rank - 1) / (Number of rows - 1)  --> 2-1 / (5-1) = 0,25

  With x as (Select Year(SalesDate) SalesYear, CustNo, Sum(Amount) Total
              From Sales
              Group By Year(SalesDate), CustNo)
  Select SalesYear, Total,
        Cast(Cume_Dist()    Over(Order By Total ) as Dec(7, 5))    as "Cume_Dist",
        Cast(Percent_Rank() Over(Order By Total ) as Dec(7, 5))    as "Percent_Rank"
    from  x
    Where SalesYear = 2009
    Order By SalesYear, Total ;

--
=================================================================================================
```

```
=====
-- OLAP Aggregate Functions
--
=============================================================================================================
=====
Select * from SalesVWYear;

Select * from SalesCustY
Order By SalesYear, CustNo;

Refresh Table SalesCustY;

Commit;

-------------------------------------------------------------------------------------------------------------
-------
-- Without windows-Aggregation-Group Clause
-------------------------------------------------------------------------------------------------------------
-------
-- Without Window-Aggregation-Group and without Windows-Order
--    --> All rows are accumulated
-- Accumulated Aggregate Information for a specific year
 Select  SalesYear, CustNo, CustName1, Amount,
         Sum(Amount) Over() "Sum Part",
         Avg(Amount) Over() "Avg Part",
         Max(Amount) Over() "Maximum Part",
         Min(Amount) Over() "Minmum Part"
   From SalesCustY
   Where SalesYear = 2009
   Order By SalesYear, CustNo;


-- Accumulated Aggregate Information for a specific year - Incl. Calculating % and Grand Total
 Select  SalesYear, CustNo, CustName1,
         Case When Grouping(CustNo) = 0
              Then Amount
              else Sum(Amount) Over()
              End           "Amount",
         Sum(Amount) Over() "Sum Part",
         Avg(Amount) Over() "Avg Part",
         Max(Amount) Over() "Maximum Part",
         Min(Amount) Over() "Minmum Part",
         Sum(Amount) * 100,0000 / Sum(Amount) Over() "% Year"
   From SalesCustY
   Where SalesYear = 2009
   Group By Grouping Sets((SalesYear, CustNo, CustName1, Amount), ())
   Order By SalesYear, CustNo;

-------------------------------------------------------------------------------------------------------------
-------
-- With Partition By
-------------------------------------------------------------------------------------------------------------
-------
-- Accumulated Aggregate Information for all years
 Select  SalesYear, CustNo, CustName1, Amount,
         Sum(Amount) Over(Partition By SalesYear) "Sum Part",
         Avg(Amount) Over(Partition By SalesYear) "Avg Part",
         Max(Amount) Over(Partition By SalesYear) "Maximum Part",
         Min(Amount) Over(Partition By SalesYear) "Minmum Part"
   From SalesCustY
   Order By SalesYear, CustNo
--Where SalesYear = 2009
--Order BY SalesYear, Amount Desc
;

-- Accumulated Aggreagate Information for all years (Level break SALESYEAR)
```

```
-- Including % per year, Total per Sales Year and Grand Total
 Select  SalesYear, CustNo, CustName1,
         Case When Grouping(SalesYear) = 1
              Then Sum(Amount) Over()
              When Grouping(CustNo)    = 1
              Then Sum(Amount) Over(Partition By SalesYear)
              Else Amount End                      "Amount",
         Sum(Amount) Over(Partition By SalesYear) "Sum Part",
         Avg(Amount) Over(Partition By SalesYear) "Avg Part",
         Max(Amount) Over(Partition By SalesYear) "Maximum Part",
         Min(Amount) Over(Partition By SalesYear) "Minmum Part",
         Sum(Amount) Over()                       "Sum Total",
         Case When Grouping(CustNo) = 1
              Then Sum(Amount)
              Else Amount End * 100,0000 / Sum(Amount) Over(Partition By SalesYear) "% Year"
   From SalesCustY
   Group By Grouping Sets((SalesYear, CustNo, CustName1, Amount), (SalesYear), ())
   Order By SalesYear, CustNo ;

-------------------------------------------------------------------------------------------------------
-------
-- with OVER(Order By)
-------------------------------------------------------------------------------------------------------
-------
-- 2. Without window-aggregation-group-Clause but with window order clause
--    --> Rolling results
Select  SalesYear, CustNo, CustName1, Amount,
        Sum(Amount) Over(Order By Amount desc) "Rolling Sum",
        Avg(Amount) Over(Order By Amount Desc) "Rolling Avg",
        Max(Amount) Over(Order By Amount Desc) "Rolling Maximum",
        Min(Amount) Over(Order By Amount Desc) "Rolling Minmum"
From SalesCustY
Where SalesYear = 2009
Order BY SalesYear, Amount Desc
--Order By SalesYear, CustNo
;

-------------------------------------------------------------------------------------------------------
-------
-- with OVER(Partition By ... Order By ...)
-------------------------------------------------------------------------------------------------------
-------
Select  SalesYear, CustNo, CustName1, Amount,
        Sum(Amount) Over(Partition By SalesYear Order By Amount desc) "Rolling Sum",
        Avg(Amount) Over(Partition By SalesYear Order By Amount Desc) "Rolling Avg",
        Max(Amount) Over(Partition By SalesYear Order By Amount Desc) "Rolling Maximum",
        Min(Amount) Over(Partition By SalesYear Order By Amount Desc) "Rolling Minmum"
From SalesCustY
-- Where SalesYear = 2009
Order BY SalesYear, Amount Desc
-- Order By SalesYear, CustNo
;

-- Rolling Totals over multiple years
Select  SalesYear, CustNo, CustName1, Amount,
        Sum(Amount) Over(Partition By SalesYear Order By Amount desc) "Rolling Sum",
        Sum(Amount) Over(Partition By SalesYear)                      "Sum/Partition",
        Avg(Amount) Over(Partition By SalesYear Order By Amount Desc) "Rolling Avg",
        Avg(Amount) Over(Partition By SalesYear)                      "Avg/Partition"
From SalesCustY
Order BY SalesYear, Amount Desc ;

-- Rolling Totals over multiple years including Totals per customer and year
Select  SalesYear, CustNo, CustName1,
        Case When Grouping(SalesYear) = 1
             Then Sum(Amount) Over()
```

```
             When Grouping(CustNo) = 1
             Then Sum(Amount) Over(Partition By SalesYear)
             Else Amount End                                          "Amount",
        Case When Grouping(CustNo) = 1
             Then NULL
             Else Sum(Amount) Over(Partition By SalesYear
                                   Order By Amount Asc) end           "Rolling Sum",
        Sum(Amount) Over(Partition By SalesYear)                      "Sum/Partition",
        Case When Grouping(CustNo) = 1
             then NULL
             Else Avg(Amount) Over(Partition By SalesYear Order By Amount Asc)
             End                                                      "Rolling Avg",
        Avg(Amount) Over(Partition By SalesYear)                      "Avg/Partition"
From SalesCustY
Group By Grouping Sets((SalesYear, CustNo, CustName1, Amount), (SalesYear), ())
Order BY SalesYear, Amount Asc
;

 with SalesPlan as (Select *
                      from (Values(2008, 2500,00 ),
                                  (2009, 15000,00),
                                  (2010, 20000,00)) x (SalesYear, Plan)),
      SalesCust as (Select SalesYear, CustNo, Sum(Total) TotalCustYear
                      From SalesVWYear
                      Group By SalesYear, CustNo),
      Totals    as (Select s.SalesYear, CustNo, TotalCustYear, Plan,
                            Sum(TotalCustYear) Over(Partition By s.SalesYear)
"Total Year",
                            Sum(TotalCustYear) Over(Partition By s.SalesYear Order By CustNo)
"Rolling Sum",
                            Sum(TotalCustYear) Over(Partition By s.SalesYear Order By CustNo) -
Plan "Rolling Diff Plan"
                     From SalesCust s join SalesPlan p on s.SalesYear = p.SalesYear)
 Select SalesYear, CustNo,   "Total Year",
        TotalCustYear,       TotalCustYear * 100,0000 / "Total Year" "% Year",
        "Rolling Sum",       "Rolling Sum" * 100,0000 / "Total Year" "% Rolling",
        Plan,                TotalCustYear * 100,0000 / Plan         "% Plan",
                             "Rolling Sum" * 100,0000 / Plan         "% Rolling Plan",
        "Rolling Diff Plan", "Rolling Diff Plan" * 100,0000 / Plan   "% Rolling Diff Plan"
      from Totals
      Order By SalesYear, CustNo;

--
================================================================================================
=====
-- OLAP Aggregate Functions
--
================================================================================================
=====
Select  SalesYear, CustNo, CustName1, Amount,
        First_Value(CustNo)  Over(Partition By SalesYear Order By Amount desc) "First Value",
        Last_Value(CustNo)   Over(Partition By SalesYear)                      "Last Value" ,
        Nth_Value(CustNo, 4) Over(Partition By SalesYear)                      "4th Value",
        Nth_Value(CustNo, 4) From Last Over(Partition By SalesYear)            "4th From Last"
From SalesCustY
-- Where SalesYear = 2009
Order BY SalesYear, Amount Desc  ;

------------------------------------------------------------------------------------------------
-------
-- Ratio to Report (Percent Caluclation)
------------------------------------------------------------------------------------------------
-------
  Select CustNo, SalesYear, Amount,
         Cast(Ratio_To_Report(Amount) Over(Partition By CustNo) * 100,0000
            as Dec(11, 2))                                             "Ratio Amount",
```

```
       Amount * 100,000 / Sum(Amount) Over(Partition By CustNo)            "% Amount"
  From SalesCustY
  Order By CustNo, SalesYear;

-- Incl. Totals
With x as (Select CustNo, SalesYear, Amount,
                  Ratio_To_Report(Amount) Over(Partition By CustNo) * 100,0000       "Ratio
Amount",
                  Amount * 100,0000 / Sum(Double(Amount)) Over(Partition By CustNo)  "% Amount"
              From SalesCustY)
Select CustNo, SalesYear, Sum(Amount)                               "Total Amount",
                          Cast(Sum("Ratio Amount") as Dec(11, 2)) "Ratio Amount",
                          Cast(Sum("% Amount")     as Dec(11, 2)) "% Amount"
  From x
  Group By Grouping Sets((CustNo, SalesYear), (CustNo), ())
  Order By CustNo, SalesYear;

-------------------------------------------------------------------------------------------------
-------
-- Running Ratio / %
-------------------------------------------------------------------------------------------------
-------
Select SalesYear, CustNo, CustName1, Amount,
       Cast(Ratio_To_Report(Amount) Over(Partition By SalesYear
                                         Order By Amount Desc) * 100,00
            as Dec(7, 2))                                                          "Ratio
Rolling",
       Cast(Dec(Amount, 11, 2) * 100,00 / Sum(Amount) Over(Partition By SalesYear
                                                           Order By Amount Desc)
            as Dec(7, 2))                                                          "% Rolling",
       Cast(Ratio_To_Report(Amount) Over(Partition By SalesYear) * 100,00
            as Dec(7, 2))                                                          "Ratio
Amount",
       Cast(Dec(Amount, 11, 2) *100,00 / Sum(Amount) Over(Partition By SalesYear)
            as Dec(7, 2))                                                          "% Amount"
  From SalesCustY
  Order By SalesYear, Amount Desc;

  Values('1350,00 / 1350,00',                                        1350,00 / 1350,00),
        (' 535,00 / (1350,00 + 535,00)',                              535,00 / (1350,00 +
535,00)),
        (' 470,00 / (1350,00 + 535,00 + 470,00)',                     470,00 / (1350,00 + 535,00
+ 470,00)),
        (' 310,00 / (1350,00 + 535,00 + 470,00 + 310,00)',            310,00 / (1350,00 + 535,00
+ 470,00 + 310,00)),
        (' 115,00 / (1350,00 + 535,00 + 470,00 + 310,00 + 115,00)',   115,00 / (1350,00 + 535,00
+ 470,00 + 310,00 + 115,00));

  Values('1350,00 / (1350,00 + 535,00 + 470,00 + 310,00 + 115,00)', 1350,00 / (1350,00 + 535,00
+ 470,00 + 310,00 + 115,00)),
        (' 535,00 / (1350,00 + 535,00 + 470,00 + 310,00 + 115,00)', 535,00 / (1350,00 + 535,00
+ 470,00 + 310,00 + 115,00)),
        (' 470,00 / (1350,00 + 535,00 + 470,00 + 310,00 + 115,00)', 470,00 / (1350,00 + 535,00
+ 470,00 + 310,00 + 115,00)),
        (' 310,00 / (1350,00 + 535,00 + 470,00 + 310,00 + 115,00)', 310,00 / (1350,00 + 535,00
+ 470,00 + 310,00 + 115,00)),
        (' 115,00 / (1350,00 + 535,00 + 470,00 + 310,00 + 115,00)', 115,00 / (1350,00 + 535,00
+ 470,00 + 310,00 + 115,00));

--
=============================================================================================
=====
-- Rows and Ranges
--
=============================================================================================
=====
```

```
-- Rows
-------------------------------------------------------------------------------------------------------
-------
 Select  SalesYear, CustNo, CustName1, Amount,
         Sum(Amount) Over(Partition by SalesYear
                          Order By Amount desc
                          Rows between 1 preceding and 1 following) "Sum Rows +/- 1",
           Lag(Amount, 1, 0) Over(Partition By SalesYear
                                  Order By Amount Desc)
         + Amount
         + Lead(Amount, 1, 0) Over(Partition By SalesYear
                                  Order By Amount Desc)            "Lag/Lead"
   From SalesCustY
   Where SalesYear = 2009
   Order BY SalesYear, Amount Desc;

-- Other Aggregate Functions
-------------------------------*
Select  SalesYear, CustNo, CustName1, Amount,
        Sum(Amount) Over(Partition by SalesYear
                         Order By Amount desc
                         Rows between 1 preceding and 1 following) "Sum Rows +/- 1",
        Avg(Amount) Over(Order By Amount Desc
                         Rows between 1 preceding and 1 following) "Avg Rows +/- 1",
        Max(Amount) Over(Order By Amount Desc
                         Rows between 1 preceding and 1 following) "Maximum Rows +/- 1",
        Min(Amount) Over(Order By Amount Desc
                         Rows between 1 preceding and 1 following) "Minmum Rows +/- 1"
From SalesCustY
Where SalesYear = 2009
Order BY SalesYear, Amount Desc;

-------------------------------------------------------------------------------------------------------
-------
-- 2. Range
-------------------------------------------------------------------------------------------------------
-------

Select * from SalesVWYearMonth;
Select * from SalesVwYear;

Select SalesYear, Sum(Total)
From SalesVwYear
Group By SalesYear;

-- Range
-------------------------------------
 Select  SalesYear, Trim(CustNo) CustNo, Trim(CustName1) CustName1, Amount,
         Sum(Amount) Over(Partition by SalesYear
                          Order By Amount desc
                          Range between 1000 preceding and 1000 following) "Sum +/- 1000",
         Avg(Amount) Over(Order By Amount Desc
                          Range between 1000 preceding and 1000 following) "Avg +/- 1000",
         Max(Amount) Over(Order By Amount Desc
                          Range between 1000 preceding and 1000 following) "Maximum +/- 1000",
         Min(Amount) Over(Order By Amount Desc
                          Range between 1000 preceding and 1000 following) "Minmum +/- 1000"
   From SalesCustY
   Where SalesYear = 2009
   Order BY SalesYear, Amount Desc;

Values(4589,86 + 3741,95 ,
       2673,95 + 2634,30 ,
       2673,95 + 2634,20 + 1636,25 ,
       2634,20 + 1636,25);
```