```
Set dataManagerTables =
'','DataCoSupplyChainDataset','tokenized_access_logs';
//This block renames script tables from non generated section which conflict
with the names of managed tables

For each name in $(dataManagerTables)
    Let index = 0;
    Let currentName = name;
    Let tableNumber = TableNumber(name);
    Let matches = 0;
    Do while not IsNull(tableNumber) or (index > 0 and matches > 0)
        index = index + 1;
        currentName = name & '-' & index;
        tableNumber = TableNumber(currentName)
        matches = Match('$(currentName)', $(dataManagerTables));
    Loop
    If index > 0 then
            Rename Table '$(name)' to '$(currentName)';
    EndIf;
Next;
Set dataManagerTables = ;


Unqualify *;

__cityAliasesBase:
LOAD
        Alias AS [__City],
        geoKey AS [__geoKey],
        CountryCode AS [__CityCountryCode]
FROM [lib://DataFiles/cityAliases.qvd]
(qvd);

__cityGeoBase:
LOAD
        geoKey AS [__geoKey],
        geoPoint AS [__GeoPoint]
FROM [lib://DataFiles/cityGeo.qvd]
(qvd);

__countryAliasesBase:
LOAD
        Alias AS [__Country],
        ISO3Code AS [__ISO3Code]
```

```
FROM [lib://DataFiles/countryAliases.qvd]
(qvd);

__countryGeoBase:
LOAD
        ISO3Code AS [__ISO3Code],
        ISO2Code AS [__ISO2Code],
        Polygon AS [__Polygon]
FROM [lib://DataFiles/countryGeo.qvd]
(qvd);

__countryCodeAndCityName2Key:
MAPPING LOAD
        __CityCountryCode & __City,
        __geoKey
RESIDENT __cityAliasesBase;

__cityKey2GeoPoint:
MAPPING LOAD
        __geoKey,
        __GeoPoint
RESIDENT __cityGeoBase;

__countryName2IsoThree:
MAPPING LOAD
        __Country,
        __ISO3Code
RESIDENT __countryAliasesBase;

__countryCodeIsoThree2Polygon:
MAPPING LOAD
        __ISO3Code,
        __Polygon
RESIDENT __countryGeoBase;

[DataCoSupplyChainDataset]:
LOAD
        [Type],
        [Days for shipping (real)],
        [Days for shipment (scheduled)],
        [Benefit per order],
        [Sales per customer],
        [Delivery Status],
        [Late_delivery_risk],
        [Category Id],
        [Category Name],
        [Customer City],
        [Customer Country],
        [Customer Email],
        [Customer Fname],
        [Customer Id],
        [Customer Lname],
        [Customer Password],
        [Customer Segment],
        [Customer State],
        [Customer Street],
        [Customer Zipcode],
        [Department Id],
        [Department Name],
        [Latitude],
        [Longitude],
        [Market],
        [Order City],
        [Order Country],
        [Order Customer Id],
        Timestamp(Timestamp#([order date (DateOrders)], 'M/D/YYYY h:mm') )
AS [order date (DateOrders)],
        [Order Id],
        [Order Item Cardprod Id],
        [Order Item Discount],
        [Order Item Discount Rate],
        [Order Item Id],
        [Order Item Product Price],
        [Order Item Profit Ratio],
        [Order Item Quantity],
        [Sales],
        [Order Item Total],
        [Order Profit Per Order],
        [Order Region],
        [Order State],
        [Order Status],
        [Order Zipcode],
        [Product Card Id],
        [Product Category Id],
        [Product Description],
        [Product Image],
        [Product Name],
        [Product Price],
        [Product Status],
        Timestamp(Timestamp#([shipping date (DateOrders)], 'M/D/YYYY
h:mm') ) AS [shipping date (DateOrders)],
        [Shipping Mode],
        APPLYMAP( '__cityKey2GeoPoint',
APPLYMAP( '__countryCodeAndCityName2Key',
APPLYMAP( '__countryName2IsoThree', LOWER([Order Country])) &
LOWER([Customer City])), '-') AS [DataCoSupplyChainDataset.Customer
City_GeoInfo],
	GeoMakePoint([Latitude], [Longitude]) AS [Longitude_Latitude],
	APPLYMAP( '__cityKey2GeoPoint',
APPLYMAP( '__countryCodeAndCityName2Key',
APPLYMAP( '__countryName2IsoThree', LOWER([Order Country])) &
LOWER([Order City])), '-') AS [DataCoSupplyChainDataset.Order City_GeoInfo],
	APPLYMAP( '__countryCodeIsoThree2Polygon',
APPLYMAP( '__countryName2IsoThree', LOWER([Order Country])), '-') AS
[DataCoSupplyChainDataset.Order Country_GeoInfo]
 FROM [lib://DataFiles/DataCoSupplyChainDataset.csv]
(txt, codepage is 28591, embedded labels, delimiter is ',', msq);

[tokenized_access_logs]:
LOAD
	[Product],
	[Category],
	Timestamp(Timestamp#([Date], 'M/D/YYYY h:mm') ) AS [Date],
	[Month],
	[Hour],
	[Department],
	[ip],
	[url]
 FROM [lib://DataFiles/tokenized_access_logs.csv]
(txt, codepage is 28591, embedded labels, delimiter is ',', msq);



TAG FIELD [Customer City] WITH '$geoname',
'$relates_DataCoSupplyChainDataset.Customer City_GeoInfo';
TAG FIELD [DataCoSupplyChainDataset.Customer City_GeoInfo] WITH
'$geopoint', '$hidden', '$relates_Customer City';
TAG FIELD [Order City] WITH '$geoname',
'$relates_DataCoSupplyChainDataset.Order City_GeoInfo';
TAG FIELD [DataCoSupplyChainDataset.Order City_GeoInfo] WITH '$geopoint',
'$hidden', '$relates_Order City';
TAG FIELD [Order Country] WITH '$geoname',
'$relates_DataCoSupplyChainDataset.Order Country_GeoInfo';
TAG FIELD [DataCoSupplyChainDataset.Order Country_GeoInfo] WITH
'$geopolygon', '$hidden', '$relates_Order Country';

DROP TABLES __cityAliasesBase, __cityGeoBase, __countryAliasesBase,
__countryGeoBase;
[autoCalendar]:
  DECLARE FIELD DEFINITION Tagged ('$date')
FIELDS
```

```
  Dual(Year($1), YearStart($1)) AS [Year] Tagged ('$axis', '$year'),
  Dual('Q'&Num(Ceil(Num(Month($1))/3)),Num(Ceil(NUM(Month($1))/3),00)) AS
[Quarter] Tagged ('$quarter', '$cyclic'),
  Dual(Year($1)&'-Q'&Num(Ceil(Num(Month($1))/3)),QuarterStart($1)) AS
[YearQuarter] Tagged ('$yearquarter', '$qualified'),
  Dual('Q'&Num(Ceil(Num(Month($1))/3)),QuarterStart($1)) AS [_YearQuarter]
Tagged ('$yearquarter', '$hidden', '$simplified'),
  Month($1) AS [Month] Tagged ('$month', '$cyclic'),
  Dual(Year($1)&'-'&Month($1), monthstart($1)) AS [YearMonth] Tagged
('$axis', '$yearmonth', '$qualified'),
  Dual(Month($1), monthstart($1)) AS [_YearMonth] Tagged ('$axis',
'$yearmonth', '$simplified', '$hidden'),
  Dual('W'&Num(Week($1),00), Num(Week($1),00)) AS [Week] Tagged
('$weeknumber', '$cyclic'),
  Date(Floor($1)) AS [Date] Tagged ('$axis', '$date', '$qualified'),
  Date(Floor($1), 'D') AS [_Date] Tagged ('$axis', '$date', '$hidden',
'$simplified'),
  If (DayNumberOfYear($1) <= DayNumberOfYear(Today()), 1, 0) AS [InYTD] ,
  Year(Today())-Year($1) AS [YearsAgo] ,
  If (DayNumberOfQuarter($1) <= DayNumberOfQuarter(Today()),1,0) AS
[InQTD] ,
  4*Year(Today())+Ceil(Month(Today())/3)-4*Year($1)-Ceil(Month($1)/3) AS
[QuartersAgo] ,
  Ceil(Month(Today())/3)-Ceil(Month($1)/3) AS [QuarterRelNo] ,
  If(Day($1)<=Day(Today()),1,0) AS [InMTD] ,
  12*Year(Today())+Month(Today())-12*Year($1)-Month($1) AS [MonthsAgo] ,
  Month(Today())-Month($1) AS [MonthRelNo] ,
  If(WeekDay($1)<=WeekDay(Today()),1,0) AS [InWTD] ,
  (WeekStart(Today())-WeekStart($1))/7 AS [WeeksAgo] ,
  Week(Today())-Week($1) AS [WeekRelNo] ;

DERIVE FIELDS FROM FIELDS [order date (DateOrders)], [shipping date
(DateOrders)], [Date] USING [autoCalendar] ;
```