Для базы данных по схеме "Звезда" со второго урока, настройте ETL-процессы между базой данных Adventure Works и вашей базой данных,

с использованием промежуточных баз данных stg и ods.

Пример: <https://youtu.be/enahePzYT6E>

Создаем базы данных **stg, ods, dwh**



База данных **stg**

USE [stg]

GO

/\*\*\*\*\*\* Object: Table [Person].[Person] Script Date: 24.09.2021 14:18:07 \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

CREATE TABLE [Product](

[ProductID] [int] IDENTITY(1,1) NOT NULL,

[Name] [nvarchar](50) NOT NULL,

[ProductNumber] [nvarchar](25) NOT NULL,

[MakeFlag] [bit] NOT NULL,

[FinishedGoodsFlag] [bit] NOT NULL,

[Color] [nvarchar](15) NULL,

[SafetyStockLevel] [smallint] NOT NULL,

[ReorderPoint] [smallint] NOT NULL,

[StandardCost] [money] NOT NULL,

[ListPrice] [money] NOT NULL,

[Size] [nvarchar](5) NULL,

[SizeUnitMeasureCode] [nchar](3) NULL,

[WeightUnitMeasureCode] [nchar](3) NULL,

[Weight] [decimal](8, 2) NULL,

[DaysToManufacture] [int] NOT NULL,

[ProductLine] [nchar](2) NULL,

[Class] [nchar](2) NULL,

[Style] [nchar](2) NULL,

[ProductSubcategoryID] [int] NULL,

[ProductModelID] [int] NULL,

[SellStartDate] [datetime] NOT NULL,

[SellEndDate] [datetime] NULL,

[DiscontinuedDate] [datetime] NULL,

[ModifiedDate] [datetime] NOT NULL)

USE [stg]

CREATE TABLE [SalesOrderDetail](

[SalesOrderID] [int] NOT NULL,

[SalesOrderDetailID] [int] NOT NULL,

[CarrierTrackingNumber] [nvarchar](25) NULL,

[OrderQty] [smallint] NOT NULL,

[ProductID] [int] NOT NULL,

[SpecialOfferID] [int] NOT NULL,

[UnitPrice] [money] NOT NULL,

[UnitPriceDiscount] [money] NOT NULL,

[LineTotal] AS (isnull(([UnitPrice]\*((1.0)-[UnitPriceDiscount]))\*[OrderQty],(0.0))),

[ModifiedDate] [datetime] NOT NULL)

USE [stg]

GO

/\*\*\*\*\*\* Object: Table [Sales].[SalesOrderHeader] Script Date: 24.09.2021 14:25:32 \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

CREATE TABLE [SalesOrderHeader](

[SalesOrderID] [int] NOT NULL,

[RevisionNumber] [tinyint] NOT NULL,

[OrderDate] [datetime] NOT NULL,

[DueDate] [datetime] NOT NULL,

[ShipDate] [datetime] NULL,

[Status] [tinyint] NOT NULL,

[OnlineOrderFlag] [bit] NOT NULL,

[SalesOrderNumber] AS (isnull(N'SO'+CONVERT([nvarchar](23),[SalesOrderID]),N'\*\*\* ERROR \*\*\*')),

[PurchaseOrderNumber] [varchar](25) NULL,

[AccountNumber] [varchar](25) NULL,

[CustomerID] [int] NOT NULL,

[SalesPersonID] [int] NULL,

[TerritoryID] [int] NULL,

[BillToAddressID] [int] NOT NULL,

[ShipToAddressID] [int] NOT NULL,

[ShipMethodID] [int] NOT NULL,

[CreditCardID] [int] NULL,

[CreditCardApprovalCode] [varchar](15) NULL,

[CurrencyRateID] [int] NULL,

[SubTotal] [money] NOT NULL,

[TaxAmt] [money] NOT NULL,

[Freight] [money] NOT NULL,

[TotalDue] AS (isnull(([SubTotal]+[TaxAmt])+[Freight],(0))),

[Comment] [nvarchar](128) NULL,

[ModifiedDate] [datetime] NOT NULL)

Вставляем данные в таблицы.

INSERT INTO [stg].dbo.SalesOrderDetail

([SalesOrderID]

,[SalesOrderDetailID]

,[CarrierTrackingNumber]

,[OrderQty]

,[ProductID]

,[SpecialOfferID]

,[UnitPrice]

,[UnitPriceDiscount]

,[ModifiedDate])

SELECT [SalesOrderID]

,[SalesOrderDetailID]

,[CarrierTrackingNumber]

,[OrderQty]

,[ProductID]

,[SpecialOfferID]

,[UnitPrice]

,[UnitPriceDiscount]

,[ModifiedDate] FROM [AdventureWorks2019].[Sales].[SalesOrderDetail]

WHERE ModifiedDate = (SELECT max(ModifiedDate) FROM [AdventureWorks2019].[Sales].[SalesOrderDetail])

INSERT INTO [stg].dbo.[SalesOrderHeader]

([SalesOrderID]

,[RevisionNumber]

,[OrderDate]

,[DueDate]

,[ShipDate]

,[Status]

,[OnlineOrderFlag]

,[PurchaseOrderNumber]

,[AccountNumber]

,[CustomerID]

,[SalesPersonID]

,[TerritoryID]

,[BillToAddressID]

,[ShipToAddressID]

,[ShipMethodID]

,[CreditCardID]

,[CreditCardApprovalCode]

,[CurrencyRateID]

,[SubTotal]

,[TaxAmt]

,[Freight]

,[Comment]

,[ModifiedDate])

SELECT [SalesOrderID]

,[RevisionNumber]

,[OrderDate]

,[DueDate]

,[ShipDate]

,[Status]

,[OnlineOrderFlag]

,[PurchaseOrderNumber]

,[AccountNumber]

,[CustomerID]

,[SalesPersonID]

,[TerritoryID]

,[BillToAddressID]

,[ShipToAddressID]

,[ShipMethodID]

,[CreditCardID]

,[CreditCardApprovalCode]

,[CurrencyRateID]

,[SubTotal]

,[TaxAmt]

,[Freight]

,[Comment]

,[ModifiedDate] FROM [AdventureWorks2019].[Sales].[SalesOrderHeader]

WHERE ModifiedDate = (SELECT max(ModifiedDate) FROM [AdventureWorks2019].[Sales].[SalesOrderHeader])

INSERT INTO [stg].dbo.[Product]

([Name]

,[ProductNumber]

,[MakeFlag]

,[FinishedGoodsFlag]

,[Color]

,[SafetyStockLevel]

,[ReorderPoint]

,[StandardCost]

,[ListPrice]

,[Size]

,[SizeUnitMeasureCode]

,[WeightUnitMeasureCode]

,[Weight]

,[DaysToManufacture]

,[ProductLine]

,[Class]

,[Style]

,[ProductSubcategoryID]

,[ProductModelID]

,[SellStartDate]

,[SellEndDate]

,[DiscontinuedDate]

,[ModifiedDate])

SELECT [Name]

,[ProductNumber]

,[MakeFlag]

,[FinishedGoodsFlag]

,[Color]

,[SafetyStockLevel]

,[ReorderPoint]

,[StandardCost]

,[ListPrice]

,[Size]

,[SizeUnitMeasureCode]

,[WeightUnitMeasureCode]

,[Weight]

,[DaysToManufacture]

,[ProductLine]

,[Class]

,[Style]

,[ProductSubcategoryID]

,[ProductModelID]

,[SellStartDate]

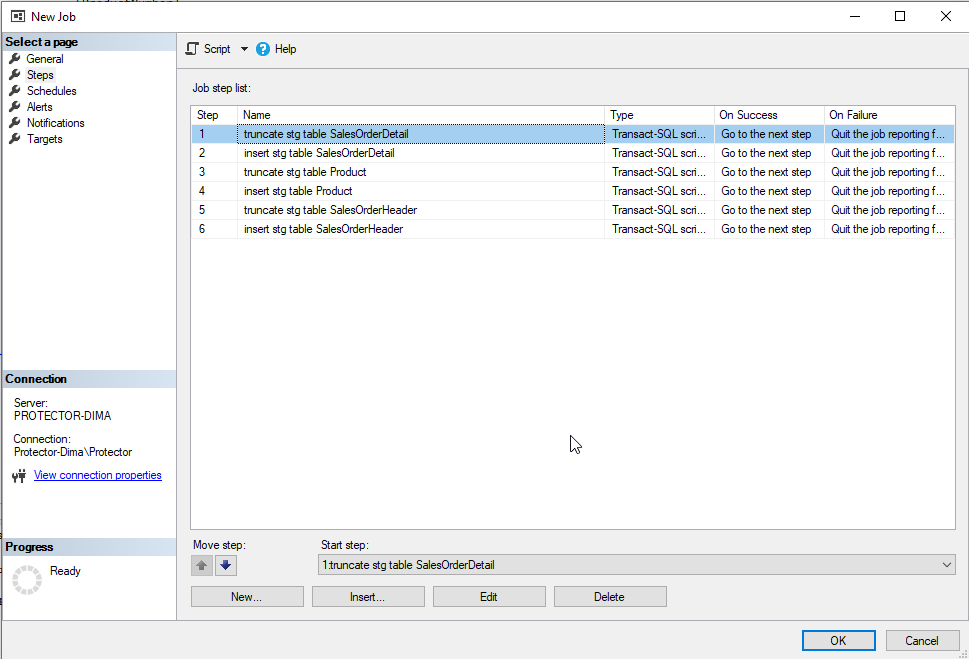
,[SellEndDate]

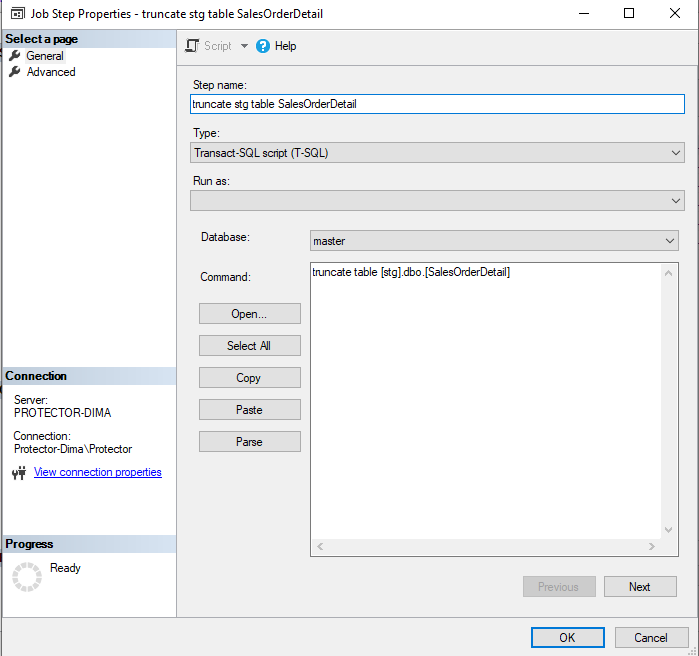
,[DiscontinuedDate]

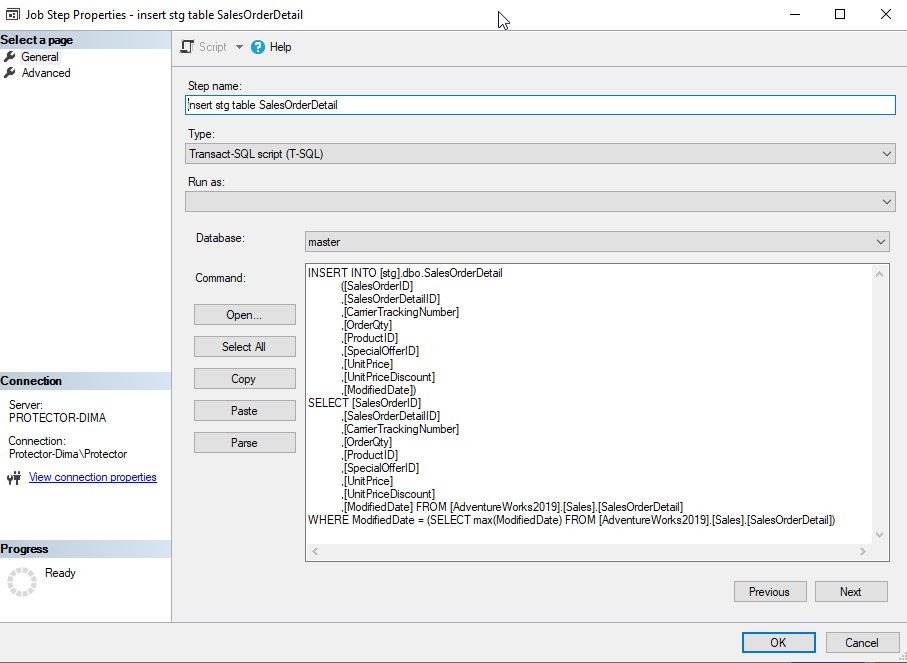
,[ModifiedDate] FROM [AdventureWorks2019].[Production].[Product]

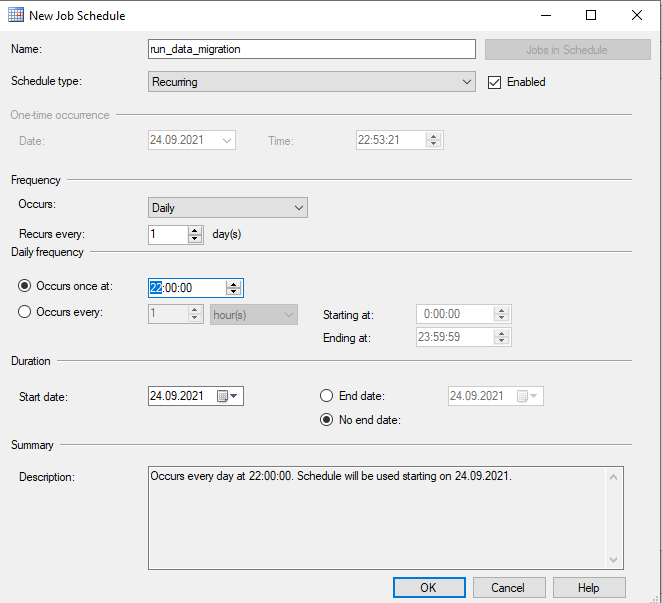
WHERE ModifiedDate = (SELECT max(ModifiedDate) FROM [AdventureWorks2019].[Production].[Product])

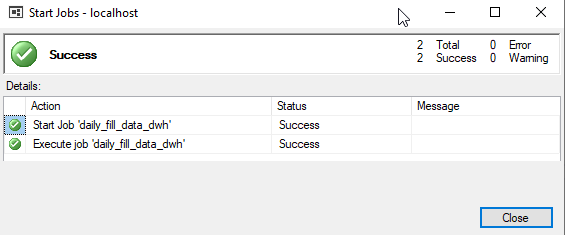
Настройка jobs и расписания их запуска











База данных **ods**

SELECT d.ProductID,

d.SalesOrderID,

d.OrderQty,

d.UnitPrice,

d.ModifiedDate

into ods.[dbo].[FctSales]

from [stg].[dbo].[SalesOrderDetail] d

INSERT INTO ods.[dbo].[FctSales]

([ProductID],

[SalesOrderID],

[OrderQty],

[UnitPrice],

[ModifiedDate])

SELECT d.ProductID,

d.SalesOrderID,

d.OrderQty,

d.UnitPrice,

d.ModifiedDate

from [stg].[dbo].[SalesOrderDetail] d

CREATE TABLE [ods].[dbo].[DmnOrder](

[SalesOrderID] [int] NOT NULL,

[SalesOrderNumber] [nvarchar](25) NOT NULL,

[AccountNumber] [nvarchar](15) NULL,

[CreditCardID] [int] NULL,

[OrderDate] [datetime] NOT NULL,

[ModifiedDate] [datetime] NOT NULL

)

INSERT INTO [ods].[dbo].[DmnOrder]

([SalesOrderID]

,[SalesOrderNumber]

,[AccountNumber]

,[CreditCardID]

,[OrderDate]

,[ModifiedDate])

SELECT SalesOrderID

,SalesOrderNumber

,AccountNumber

,CreditCardID

,OrderDate

,ModifiedDate

FROM [stg].[dbo].[SalesOrderHeader]

CREATE TABLE [ods].[dbo].[DmnProduct](

[ProductID] [int] NOT NULL,

[Name] [nvarchar](50) NOT NULL,

[ProductNumber] [nvarchar](25) NULL,

[ModifiedDate] [datetime] NOT NULL

)

INSERT INTO [ods].[dbo].[DmnProduct]

([ProductID]

,[Name]

,[ProductNumber]

,[ModifiedDate])

SELECT ProductID

,Name

,ProductNumber

,ModifiedDate

FROM [stg].[dbo].[Product]

CREATE TABLE [ods].[dbo].[DmnDate](

[OrderDate] [datetime] NOT NULL,

[DueDate] [datetime] NOT NULL,

[ShipDate] [datetime] NULL,

[Status] [tinyint] NOT NULL,

[SalesOrderNumber] [nvarchar](50) NOT NULL,

[TerritoryID] [int] NULL,

[ModifiedDate] [datetime] NOT NULL

)

INSERT INTO [ods].[dbo].[DmnDate]

([OrderDate]

,[DueDate]

,[ShipDate]

,[Status]

,[SalesOrderNumber]

,[TerritoryID]

,[ModifiedDate])

SELECT [OrderDate]

,[DueDate]

,[ShipDate]

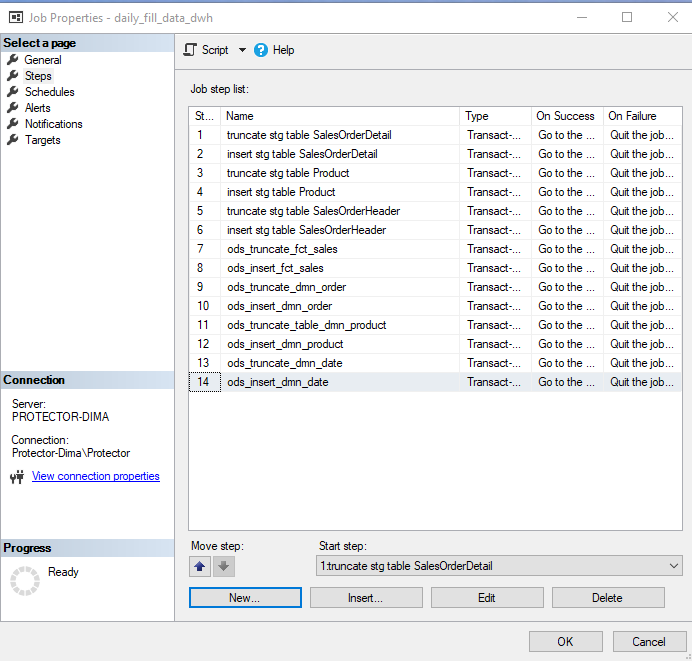
,[Status]

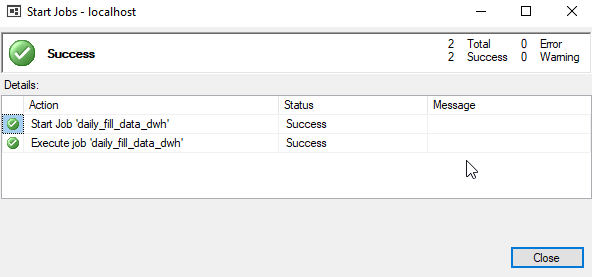
,[SalesOrderNumber]

,[TerritoryID]

,[ModifiedDate]

FROM [stg].[dbo].[SalesOrderHeader]





База данных **dwh**

select ProductID, f.SalesOrderID, h.OrderDate, sum(f.OrderQty \* f.UnitPrice) as Sum

into dwh.dbo.FctSales

from [ods].[dbo].[FctSales] f

inner join [ods].[dbo].DmnOrder h on h.SalesOrderID = f.SalesOrderID

where 1=2

group by ProductID, f.SalesOrderID, OrderDate

MERGE dwh.dbo.FctSales AS TARGET

USING (

select ProductID, f.SalesOrderID, h.OrderDate, sum(f.OrderQty \* f.UnitPrice) as Sum

from [ods].[dbo].[FctSales] f

inner join [ods].[dbo].DmnOrder h on h.SalesOrderID = f.SalesOrderID

group by ProductID, f.SalesOrderID, OrderDate) AS SOURCE

ON (TARGET.ProductID = SOURCE.ProductID and

TARGET.SalesOrderID = SOURCE.SalesOrderID and

TARGET.OrderDate = SOURCE.OrderDate)

WHEN MATCHED THEN

UPDATE SET TARGET.Sum = SOURCE.Sum

WHEN NOT MATCHED THEN

INSERT (ProductID, SalesOrderID, OrderDate, Sum)

VALUES ( SOURCE.ProductID, SOURCE.SalesOrderID, SOURCE.OrderDate, SOURCE.Sum);

select SalesOrderID,

SalesOrderNumber,

AccountNumber,

CreditCardID

into dwh.dbo.DmnOrder

from ods.dbo.DmnOrder

where 1=2

MERGE dwh.dbo.DmnOrder AS TARGET

USING (

select SalesOrderID,

SalesOrderNumber,

AccountNumber,

CreditCardID

from ods.dbo.DmnOrder) AS SOURCE

ON (TARGET.SalesOrderID = SOURCE.SalesOrderID)

WHEN MATCHED THEN

UPDATE SET TARGET.SalesOrderNumber = SOURCE.SalesOrderNumber,

TARGET.AccountNumber = SOURCE.AccountNumber,

TARGET.CreditCardID = SOURCE.CreditCardID

WHEN NOT MATCHED THEN

INSERT (SalesOrderID, SalesOrderNumber, AccountNumber, CreditCardID)

VALUES ( SOURCE.SalesOrderID, SOURCE.SalesOrderNumber, SOURCE.AccountNumber, SOURCE.CreditCardID);

select ProductID, Name, ProductNumber

into dwh.dbo.[DmnProduct]

from [ods].[dbo].[DmnProduct]

where 1=2

MERGE dwh.dbo.[DmnProduct] AS TARGET

USING (

select ProductID, Name, ProductNumber

from [ods].[dbo].[DmnProduct]) AS SOURCE

ON (TARGET.ProductID = SOURCE.ProductID)

WHEN MATCHED THEN

UPDATE SET TARGET.Name = SOURCE.Name,

TARGET.ProductNumber = SOURCE.ProductNumber

WHEN NOT MATCHED THEN

INSERT (ProductID, Name, ProductNumber)

VALUES ( SOURCE.ProductID, SOURCE.Name, SOURCE.ProductNumber);

select OrderDate

,DueDate

,ShipDate

,Status

,SalesOrderNumber

,TerritoryID

into dwh.dbo.[DmnDate]

from [ods].[dbo].[ DmnDate]

where 1=2

MERGE dwh.dbo.[DmnDate] AS TARGET

USING (

select OrderDate

,DueDate

,ShipDate

,Status

,SalesOrderNumber

,TerritoryID

from [ods].[dbo].[DmnDate]) AS SOURCE

ON (TARGET.OrderDate = SOURCE.OrderDate)

WHEN MATCHED THEN

UPDATE SET TARGET.DueDate = SOURCE.DueDate,

TARGET.ShipDate = SOURCE.ShipDate,

TARGET.Status = SOURCE.Status,

TARGET.SalesOrderNumber = SOURCE.SalesOrderNumber,

TARGET.TerritoryID = SOURCE.TerritoryID

WHEN NOT MATCHED THEN

INSERT (OrderDate, DueDate, ShipDate, Status, SalesOrderNumber, TerritoryID)

VALUES ( SOURCE.OrderDate, SOURCE.DueDate, SOURCE.ShipDate, SOURCE.Status, SOURCE.SalesOrderNumber, SOURCE.TerritoryID);

