
ETL УПРАЖНЕНИЕ

В това упражнение ще създадем SSIS проект изпълняващ ETL обработката необходима за попълването на данни в data warehouse решението на Publications Industries.

ЗАДАЧА 1: СЪЗДАВАНЕ НА SSIS ПРОЕКТ

В тази задача ще създадем началния SSIS проект необходим за нашето BI решение.

1. Стартирайте Visual Studio и отворете решение PublicationsIndustries.sln от предходното упражнение.
2. Добавете нов проект към решението избирайки File → Add → New Project от основното меню. Това отваря диалогов прозорец Add New Project.
3. Изберете категория Business Intelligence отляво в диалоговия прозорец и след това Integration Services Project от центъра.
4. Кръстете проекта **DWPubsSalesETL** и кликнете OK, за да го добавите към решението.
5. Кликнете на View → Solution Explorer от менюто, за да изобразите Solution Explorer (ако вече не е видим).
6. Преименувайте пакет Package.dtsx (името по поразбиране) на ETLProcessForDWPubsSales.dtsx.
7. Запазете работата си до момента чрез File → Save All от основното меню.

До момента създадохме само SSIS проект и го добавихме към BI решението. В следващата задача ще конфигурираме SSIS пакета.

ЗАДАЧА 2: КОНФИГУРИРАНЕ НА SSIS ПАКЕТА

В тази задача ще конфигурираме създадения SSIS пакет с всичко необходимо за извършване на ETL обработката за нашето BI решение. Целта е да напълним data warehouse-а с данни от транзакционната база данни използвайки SSIS проекта.

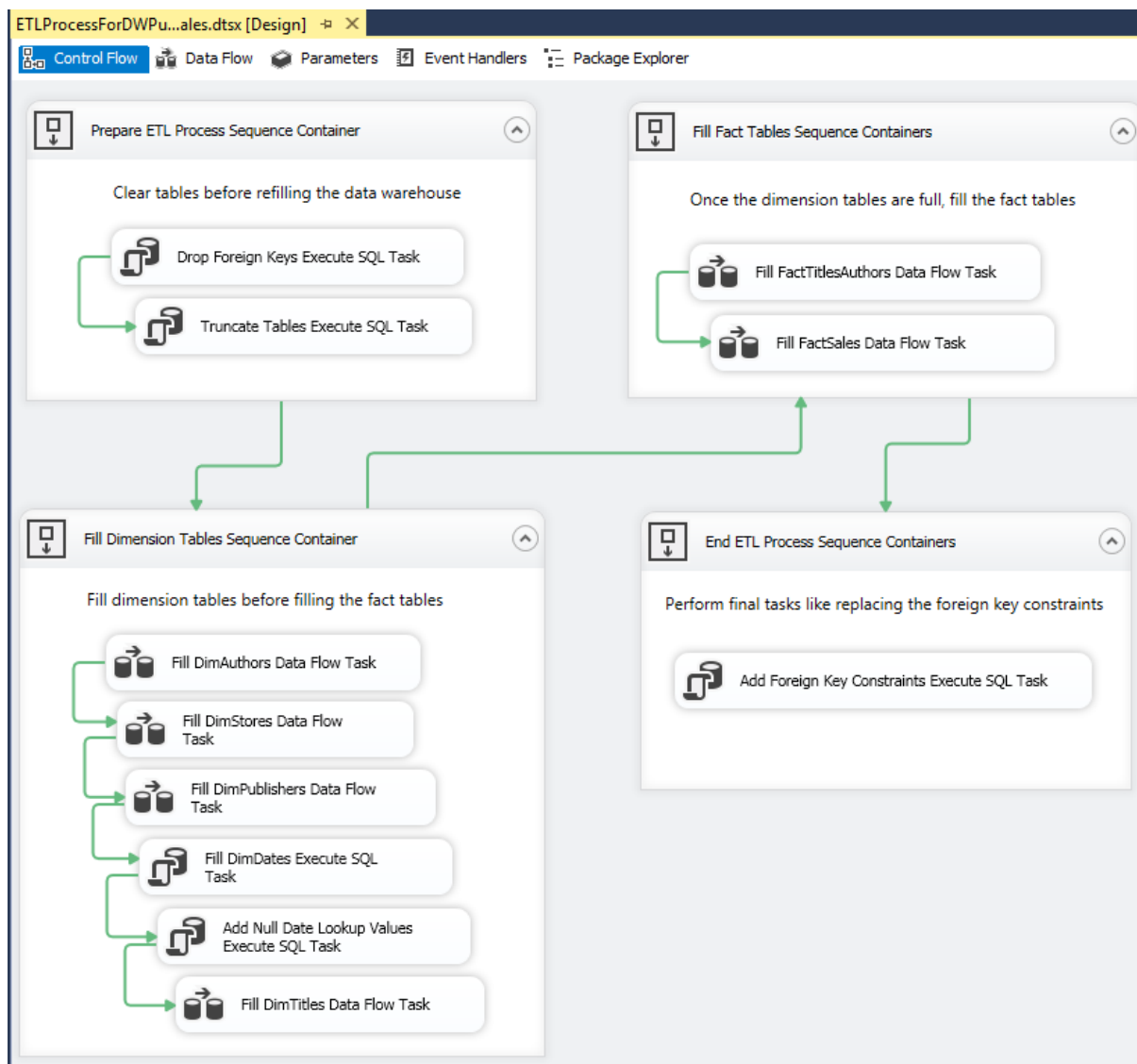
Тъй като тази задача е сложна и продължителна сме я разбили на няколко под-задачи за по-лесно изпълнение.

ЗАДАВАНЕ НА СТРУКТУРАТА НА ПАКЕТА

1. Изберете таб Control Flow в SSIS пакет ETLProcessForDWPubsSales.dtsx създаден в миналата задача.
2. Поставете четири Sequence контейнера в дизайнера от SSIS Toolbox-а и ги преименувайте чрез десен клик и избор на Rename от контекстното меню. Нека новите имена са:
 - a. Prepare ETL Process Sequence Container
 - b. Fill Dimension Tables Sequence Container
 - c. Fill Fact Tables Sequence Container
 - d. End ETL Process Sequence Container
3. Кликнете с десния бутон във всеки контейнер и изберете Add Annotation от контекстното меню.
4. Въведете следните анотации:
 - a. Clear tables before refilling the data warehouse
 - b. Fill dimension tables before filling the fact tables
 - c. Once the dimension tables are full, fill the fact tables
 - d. Perform the final tasks like replacing the foreign key constraints
5. Позиционирайте анотациите в горния ляв ъгъл на всеки контейнер.
6. Поставете две Execute SQL задачи в контейнер Prepare ETL Process Sequence Container и ги преименувайте на:

- a. Drop Foreign Keys Execute SQL Task
 - b. Truncate Tables Execute SQL Task
7. Поставете четири Data Flow задачи в контейнер Fill Dimension Tables Sequence Container и ги преименувайте на:
 - a. Fill DimAuthors Data Flow Task
 - b. Fill DimStores Data Flow Task
 - c. Fill DimPublishers Data Flow Task
 - d. Fill DimTitles Data Flow Task
8. Поставете две Execute SQL задачи в контейнер Fill Dimension Tables Sequence Container и ги преименувайте на:
 - a. Fill DimDates Execute SQL Task
 - b. Add Null Date Lookup Values Execute SQL Task
9. Тъй като дименсия DIMTITLES разчита на данни от таблица DIMDATES, трябва да я заредим с данни след като DIMDATES е заредена. Затова преместете задача Fill DimTitles Data Flow Task в края на контейнер Fill Dimension Tables Sequence Container.
10. Поставете две Data Flow задачи в контейнер Fill Fact Tables Sequence Container и ги преименувайте на:
 - a. Fill FactTitlesAuthors Data Flow Task
 - b. Fill FactSales Data Flow Task
11. Поставете една Execute SQL задача в контейнер End ETL Process Sequence Container и я преименувайте на Add Foreign Key Constraints Execute SQL Task.
12. Кликнете на задача Drop Foreign Keys Execute SQL Task и се появява стрелка, свържете я със задача Truncate Tables Execute SQL Task. Повторете същата процедура за следващите няколко стъпки.
13. Свържете със стрелка контейнер Prepare ETL Process Sequence Container с контейнер Fill Dimension Tables Sequence Container.
14. Свържете със стрелка задача Fill DimAuthors Data Flow Task със задача Fill DimStores Data Flow Task.
15. Свържете със стрелка задача Fill DimStores Data Flow Task със задача Fill DimPublishers Data Flow Task.
16. Свържете със стрелка задача Fill DimPublishers Data Flow Task със задача Fill DimDates Execute SQL Task.
17. Свържете със стрелка задача Fill DimDates Execute SQL Task със задача Add Null Date Lookup Values Execute SQL Task.
18. Свържете със стрелка задача Add Null Date Lookup Values Execute SQL Task със задача Fill DimTitles Data Flow Task.
19. Свържете със стрелка контейнер Fill Dimension Tables Sequence Container с контейнер Fill Fact Tables Sequence Container.
20. Свържете със стрелка задача Fill FactTitlesAuthors Data Flow Task със задача Fill FactSales Data Flow Task.
21. Свържете със стрелка контейнер Fill Fact Tables Sequence Container с контейнер End ETL Process Sequence Container.
22. Подредете стрелките като кликнете върху тях еднократно и ги влачите до друга позиция.
23. Запазете вашия SSIS пакет чрез File → Save от основното меню.

Когато сте готови, вашият SSIS пакет трябва да изглежда като този на фигура 1. Забележете, че някои задачи може да са отбелязани с червени точки. Това сигнализира че все още не са конфигурирани. В следващата подзадача ще се погрижим за това.



фигура 1

КОНФИГУРИРАНЕ И ТЕСТВАНЕ НА CONTROL FLOW ЗАДАЧИТЕ

1. В долната част на дизайнера се намира таб Connection Managers. Кликнете с десен бутон в него и изберете New OLE DB Connection от контекстното меню. Това отваря диалогов прозорец OLE DB Connection manager.
2. В диалоговия прозорец кликнете бутон New, за да създадете нова конекция. Това отваря диалогов прозорец Connection Manager.
3. В новия диалогов прозорец въведете името на сървъра си. Обикновено е достатъчно да въведете **(local)** заедно със скобите, освен ако инстанцията на SQL Server не е именувана (т.е. named instance).
4. В същият диалогов прозорец изберете или въведете **pubs** в полето Select or enter database name. Накрая, тествайте връзката като натиснете Test Connection, за да се уверите, че всичко е наред.
5. Повторете същия процес за създаване на OLE DB конекция до база данни DW PubsSales .
6. Преименувайте мениджърите на:
 - a. Pubs
 - b. DW PubsSales
7. Добавете ETL кода на решението от външен файл. За целта кликнете с десен бутон върху проект DW PubsSales ETL и изберете Add → Existing Item от контекстното меню,

наavigирайте до директорията съдържаща файл PublicationsIndustriesETLCode.sql и го изберете. Този файл съдържа SQL заявките нужни за нашето решение. За да не се налага ръчно да ги въвеждате, може директно да ги копирате от новодобавения файл.

8. Отворете файла използвайки Transact-SQL редактора (при първо отваряне IDE-то може да попита какъв редактор да използва за дадения тип файл). При проблем опитайте с друг редактор.
9. Намерете етикет **1a)** и от коментара потвърдете, че целта му е да премахне проверките на външни ключове. Кодът трябва да изглежда така:

```
-- 1a) Drop Foreign Keys
Alter Table [dbo].[DimTitles] Drop Constraint [FK_DimTitles_DimPublishers]
Alter Table [dbo].[FactTitlesAuthors] Drop Constraint
[FK_FactTitlesAuthors_DimAuthors]
Alter Table [dbo].[FactTitlesAuthors] Drop Constraint
[FK_FactTitlesAuthors_DimTitles]
Alter Table [dbo].[FactSales] Drop Constraint [FK_FactSales_DimStores]
Alter Table [dbo].[FactSales] Drop Constraint [FK_FactSales_DimTitles]
Alter Table [dbo].[FactSales] Drop Constraint [FK_FactSales_DimDates]
Alter Table [dbo].[DimTitles] Drop Constraint [FK_DimTitles_DimDates]
```

10. Копирайте SQL кода, за да може да го поставите в Execute SQL задачата на име Drop Foreign Keys Execute SQL Task.
11. За целта отворете редактора на задача Drop Foreign Keys Execute SQL Task чрез десен клик и избор на Edit от контекстното меню.
12. Конфигурирайте конекцията като изберете опция DWPubSales за поле Connection.
13. Конфигурирайте SQL командата като кликнете на бутона с многоточие на полето SQL Statement и поставите в него SQL кода копиран в стъпка 10. Уверете се, че не сте копирали GO команди, тъй като те могат да доведат до грешки при изпълнение. Затворете Enter SQL Query диалоговия прозорец от бутон OK.
14. Кликнете OK, за да затворите редактора на Execute SQL задачата.
15. Намерете етикет **1b)** и от коментара потвърдете, че целта му е да изчисти съдържанието на таблиците в data warehouse-a. Кодът трябва да изглежда така:

```
-- 1b) Clear all tables data warehouse tables and reset their Identity
Truncate Table dbo.FactSales
Truncate Table dbo.FactTitlesAuthors
Truncate Table dbo.DimTitles
Truncate Table dbo.DimPublishers
Truncate Table dbo.DimStores
Truncate Table dbo.DimAuthors
```

16. Копирайте SQL кода, за да може да го поставите в Execute SQL задачата на име Truncate Tables Execute SQL Task.
17. За целта отворете редактора на задача Truncate Tables Execute SQL Task чрез десен клик и избор на Edit от контекстното меню.
18. Конфигурирайте конекцията като изберете опция DWPubSales за поле Connection.
19. Конфигурирайте SQL командата като кликнете на бутона с многоточие на полето SQL Statement и поставите в него SQL кода копиран в стъпка 16. Затворете Enter SQL Query диалоговия прозорец от бутон OK.
20. Кликнете OK, за да затворите редактора на Execute SQL задачата.
21. Намерете етикет **Step 3)** и от коментара потвърдете, че целта му е да възстанови проверките на външни ключове, които преди малко премахнахме. Кодът трябва да изглежда така:

```
-- Step 3) Add Foreign Keys back
Alter Table [dbo].[DimTitles] With Check Add Constraint
[FK_DimTitles_DimPublishers] Foreign Key ([PublisherKey]) References
[dbo].[DimPublishers] ([PublisherKey])
Alter Table [dbo].[FactTitlesAuthors] With Check Add Constraint
```

```
[FK_FactTitlesAuthors_DimAuthors] Foreign Key ([AuthorKey]) References
[dbo].[DimAuthors] ([AuthorKey])
Alter Table [dbo].[FactTitlesAuthors] With Check Add Constraint
[FK_FactTitlesAuthors_DimTitles] Foreign Key ([TitleKey]) References
[dbo].[DimTitles] ([TitleKey])
Alter Table [dbo].[FactSales] With Check Add Constraint
[FK_FactSales_DimStores] Foreign Key ([StoreKey]) References
[dbo].[DimStores] ([StoreKey])
Alter Table [dbo].[FactSales] With Check Add Constraint
[FK_FactSales_DimTitles] Foreign Key ([TitleKey]) References
[dbo].[DimTitles] ([TitleKey])
Alter Table [dbo].[FactSales] With Check Add Constraint
[FK_FactSales_DimDates] Foreign Key ([OrderDateKey]) References
[dbo].[DimDates] ([DateKey])
Alter Table [dbo].[DimTitles] With Check Add Constraint
[FK_DimTitles_DimDates] Foreign Key ([PublishedDateKey]) References
[dbo].[DimDates] ([DateKey])
```

22. Копирайте SQL кода, за да може да го поставите в Execute SQL задачата на име Add Foreign Key Execute SQL Task.
23. За целта отворете редактора на задача Add Foreign Key Execute SQL Task чрез десен клик и избор на Edit от контекстното меню.
24. Конфигурирайте конекцията като изберете опция DW Pubs Sales за поле Connection.
25. Конфигурирайте SQL командата като кликнете на бутона с многоточие на полето SQL Statement и поставите в него SQL кода копиран в стъпка 22. Затворете Enter SQL Query диалоговия прозорец от бутон OK.
26. Кликнете OK, за да затворите редактора на Execute SQL задачата.
27. Тествайте така създадените задачи, като кликнете с десен бутон и изберете Execute Task от контекстното меню. Потвърдете, че задачата работи нормално или открийте и поправете появилите се грешки. Изпълнете Execute SQL задачите в следния ред:
 - a. Drop Foreign Keys Execute SQL Task
 - b. Truncate Tables Execute SQL Task
 - c. Add Foreign Keys Execute SQL Task

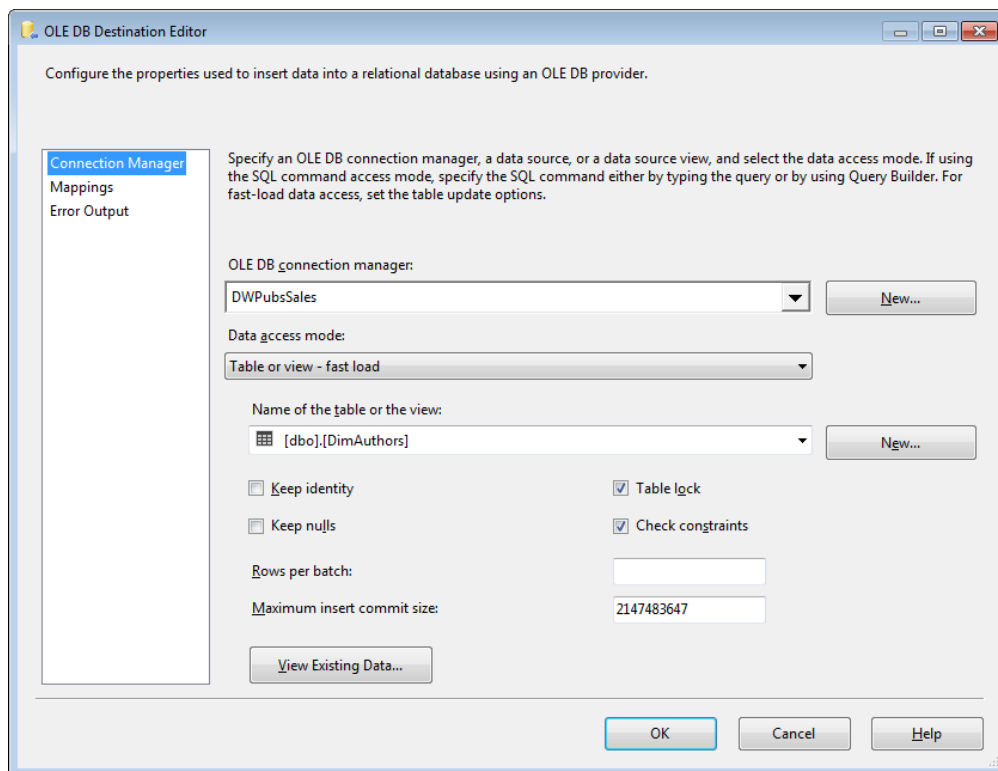
ЗАВЪРШВАНЕ НА ПАКЕТА

1. Отворете SQL скрипт PublicationsIndustriesETLCode.sql и открийте кода отбелязан с коментар:
--Step 2 Code used to fill tables
2. За да **попълните дименсия DIMAUTHORS**, отворете таб Data Flow и изберете Fill DimAuthors Data Flow Task от падащото меню в горната част на екрана.
3. Добавете OLE DB Source в дизайнера и го преименувайте на Get Authors Data OLE DB Source.
4. Отворете редактора на източника с двоен клик. Изберете мениджър Pubs в полето OLE DB connection manager и опция SQL command в полето Data access mode.
5. Намерете коментар --Step 2a) Get source data from pubs.dbo.authors в SQL файла и го копирайте. Трябва да изглежда така:

```
Select
  [AuthorId] = Cast( au_id as nChar(11) ),
  [AuthorName] = Cast((au_fname + ' ' + au_lname) as
nVarChar(100)),
  [AuthorState] = Cast(state as nChar(2))
From pubs.dbo.authors
```

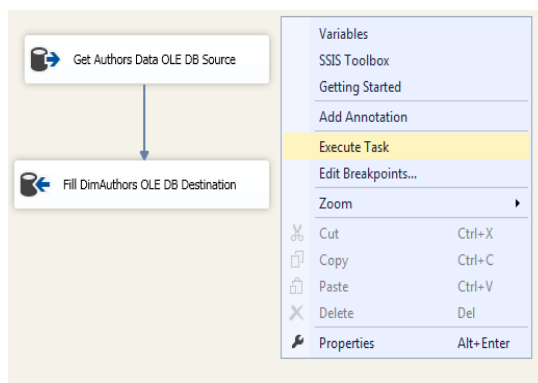
6. Поставете този код в SQL command text полето на OLE DB източника и натиснете бутон Preview, за да потвърдите, че заявката е правилна.

7. Разгледайте страница Columns показваща избраните колони. Това кара Visual Studio да запише съответния XML коректно в .dtsx файла, което може да ви предпази от грешки по-късно. **Продължете с тази допълнителна стъпка и при останалите конфигурации.**
8. Затворете редактора на OLE DB източника от бутона OK.
9. Добавете OLE DB Destination в дизайнера и го преименувайте на Fill DimAuthors OLE DB Destination.
10. Свържете с пътека източника и дестинацията на данните, използвайки синята стрелка.
11. Конфигурирайте OLE DB дестинацията както е показано на фигура 2.



фигура 2

12. Разгледайте страница Mappings показваща мапването на данните. Това кара Visual Studio да запише съответния XML коректно в .dtsx файла, което може да ви предпази от грешки по-късно. **Продължете с тази допълнителна стъпка и при останалите конфигурации.**
13. Уверете се, че data flow-а изглежда като показания на **Error! Reference source found.**
14. Както е показано на фигурата кликнете десен бутон в дизайнера и изберете Execute Task от контекстното меню, за изпълните задачата.
15. След като изпълнението завърши, спрете дебъгера и верифицирайте, че в DIMAUTHORS има данни, използвайки SQL Management Studio.



фигура 3

Инструкциите за попълване на следващите дименсии са почти идентични, така че може да се опитате да ги направите сами. Но за всеки случай ще ги опишем по-долу за референция или ако имате проблем при реализацията.

16. За да **попълните** **дименсия DIMSTORES**, отворете таб Data Flow и изберете Fill DimStores Data Flow Task от падащото меню в горната част на екрана.
17. Добавете OLE DB Source в дизайнера и го преименувайте на Get Stores Data OLE DB Source.
18. Отворете редактора на източника с двоен клик. Изберете мениджър Pubs в полето OLE DB connection manager и опция SQL command в полето Data access mode.
19. Намерете коментар --2b) Get source data from pubs.dbo.stores в SQL файла и го копирайте. Трябва да изглежда така:

```
Select
    [StoreId] = Cast( stor_id as nChar(4) ),
    [StoreName] = Cast( stor_name as nVarchar(50) )
From pubs.dbo.stores
```

20. Поставете този код в SQL command text полето на OLE DB източника и натиснете бутон Preview, за да потвърдите, че заявката е правилна.
 21. Разгледайте страница Columns показваща избраните колони и затворете редактора на OLE DB източника от бутона OK.
 22. Добавете OLE DB Destination в дизайнера и я преименувайте на Fill DimStores OLE DB Destination.
 23. Свържете с пътека източника и дестинацията на данните.
 24. Конфигурирайте OLE DB дестинацията, така че да вкарва данни в таблица DIMSTORES в DWPubsSales.
 25. Разгледайте страница Mappings и затворете редактора от бутона OK.
 26. Кликнете с десен бутон в дизайнера и изберете опция Execute Task от контекстното меню, за да изпълните задачата.
 27. След като изпълнението завърши, спрете дебъгера и верифицирайте, че в DIMSTORES има данни, използвайки SQL Management Studio.
-
28. За да **попълните** **дименсия DIMPUBLISHERS**, отворете таб Data Flow и изберете Fill DimPublishers Data Flow Task от падащото меню в горната част на екрана.
 29. Добавете OLE DB Source в дизайнера и го преименувайте на Get Publishers Data OLE DB Source.
 30. Отворете редактора на източника с двоен клик. Изберете мениджър Pubs в полето OLE DB connection manager и опция SQL command в полето Data access mode.
 31. Намерете коментар --2c) Get source data from pubs.dbo.publishers в SQL файла и го копирайте. Трябва да изглежда така:

```
Select
    [PublisherId] = Cast( pub_id as nChar(4) ),
    [PublisherName] = Cast( pub_name as nVarchar(50) )
From pubs.dbo.publishers
```

32. Поставете този код в SQL command text полето на OLE DB източника и натиснете бутон Preview, за да потвърдите, че заявката е правилна.
33. Разгледайте страница Columns показваща избраните колони и затворете редактора на OLE DB източника от бутона OK.
34. Добавете OLE DB Destination в дизайнера и я преименувайте на Fill DimPublishers OLE DB Destination.
35. Свържете с пътека източника и дестинацията на данните.
36. Конфигурирайте OLE DB дестинацията, така че да вкарва данни в таблица DIMPUBLISHERS в DWPubsSales..
37. Разгледайте страница Mappings и затворете редактора от бутона OK.

38. Кликнете с десен бутон в дизайнера и изберете опция Execute Task от контекстното меню, за да изпълните задачата.
39. След като изпълнението завърши, спрете дебъгера и верифицирайте, че в DIMPUBLISHERS има данни, използвайки SQL Management Studio.
40. За да **попълните** **дименсия DIMDATES**, изберете таб Control Flow и отворете редактора на задача Fill DimDates Execute SQL Task чрез двоен клик.
41. Конфигурирайте полето Connection да използва DWPubsSales.
42. Намерете коментар --2d) Create values for DimDates as needed в SQL файла и го копирайте. Трябва да изглежда така:

```
-- Create variables to hold the start and end date
Declare @StartDate datetime = '01/01/1990'
Declare @EndDate datetime = '01/01/1995'

-- Use a while loop to add dates to the table
Declare @DateInProgress datetime
Set @DateInProgress = @StartDate
While @DateInProgress <= @EndDate
Begin
    -- Add a row into the date dimension table for this
    date
    Insert Into DimDates ([Date], [DateName], [Month],
    [MonthName], [Quarter], [QuarterName], [Year],
    [YearName])
    Values (
        @DateInProgress -- [Date]
        , DateName( weekday, @DateInProgress ) -- [DateName]
        , Month( @DateInProgress ) -- [Month]
        , DateName( month, @DateInProgress ) -- [MonthName]
        , DateName( quarter, @DateInProgress ) -- [Quarter]
        , 'Q' + DateName( quarter, @DateInProgress ) + ' - '
+ Cast( Year(@DateInProgress) as nVarchar(50) ) --
[QuarterName]
        , Year( @DateInProgress )
        , Cast( Year(@DateInProgress ) as nVarchar(50) ) --
[YearName]
    )
    -- Add a day and loop again
    Set @DateInProgress = DateAdd(d, 1, @DateInProgress)
End
```

43. Поставете този код в полето SQLStatement, премахвайки всички GO извиквания, за да не се счупи задачата при изпълнение.
44. Затворете редактора на Execute SQL задачата от бутона OK.
45. Кликнете с десен бутон върху задачата и изберете опция Execute Task от контекстното меню, за да я изпълните.
46. След като изпълнението завърши, спрете дебъгера и верифицирайте, че в DIMDATES има данни, използвайки SQL Management Studio.
47. За да **конфигурирате** **задача ADD NULL DATE LOOKUP VALUES EXECUTE SQL TASK**, изберете таб Control Flow и отворете редактора на задачата чрез двоен клик.
48. Конфигурирайте полето Connection да използва DWPubsSales.

49. Намерете коментар --2e) Add additional lookup values to DimDates в SQL файла и го копирайте. Трябва да изглежда така:

```
Set Identity_Insert [DWPubsSales].[dbo].[DimDates] On

Insert Into [DWPubsSales].[dbo].[DimDates]
( [DateKey]
, [Date]
, [DateName]
, [Month]
, [MonthName]
, [Quarter]
, [QuarterName]
, [Year], [YearName] )
Select
    [DateKey] = -1
, [Date] = Cast('01/01/1900' as nVarchar(50) )
, [DateName] = Cast('Unknown Day' as nVarchar(50) )
, [Month] = -1
, [MonthName] = Cast('Unknown Month' as nVarchar(50) )
, [Quarter] = -1
, [QuarterName] = Cast('Unknown Quarter' as nVarchar(50)
)
, [Year] = -1
, [YearName] = Cast('Unknown Year' as nVarchar(50) )
Union
Select
    [DateKey] = -2
, [Date] = Cast('01/01/1900' as nVarchar(50) )
, [DateName] = Cast('Corrupt Day' as nVarchar(50) )
, [Month] = -2
, [MonthName] = Cast('Corrupt Month' as nVarchar(50) )
, [Quarter] = -2
, [QuarterName] = Cast('Corrupt Quarter' as nVarchar(50)
)
, [Year] = -2
, [YearName] = Cast('Corrupt Year' as nVarchar(50) )

Set Identity_Insert [DWPubsSales].[dbo].[DimDates] Off
```

50. Поставете този код в полето SQLStatement, премахвайки всички GO извиквания, за да не се счупи задачата при изпълнение.
51. Затворете редактора на Execute SQL задачата от бутона OK.
52. Кликнете с десен бутон върху задачата и изберете опция Execute Task от контекстното меню, за да я изпълните.
53. След като изпълнението завърши, спрете дебъгера и верифицирайте, че в DIMDATES има данни за неопределени дати, използвайки SQL Management Studio. Ще ги разпознаете по отрицателните ключове.
54. За да **попълните** **дименсия** **DIMTITLES**, отворете таб Data Flow и изберете Fill DimTitles Data Flow Task от падащото меню в горната част на екрана.
55. Добавете OLE DB Source в дизайнера и го преименувайте на Get Titles Data OLE DB Source.

56. Отворете редактора на източника с двоен клик. Изберете мениджър **DWPubsSales** в полето OLE DB connection manager и опция SQL command в полето Data access mode. Обърнете внимание, че за разлика от останалите дименсии, тук мениджър DWPubsSales е източник на данните, но и база данни Pubs се използва в заявката.
57. Намерете коментар --2f) Get source data from pubs.dbo.titles в SQL файла и го копирайте. Трябва да изглежда така:

```
Select
    [TitleId] = Cast( isNull( [title_id], -1 ) as
nvarchar(6) )
, [TitleName] = Cast( isNull( [title], 'Unknown' ) as
nvarchar(50) )
, [TitleType] = Cast( isNull( [type], 'Unknown' ) as
nvarchar(50) )
, [PublisherKey] =
[DWPubsSales].[dbo].[DimPublishers].[PublisherKey]
, [TitlePrice] = Cast( isNull( [price], -1 ) as
decimal(18, 4) )
, [PublishedDateKey] =
isNull([DWPubsSales].[dbo].[DimDates].[DateKey], -1)

From [Pubs].[dbo].[Titles]
Join [DWPubsSales].[dbo].[DimPublishers]
On [Pubs].[dbo].[Titles].[pub_id] =
[DWPubsSales].[dbo].[DimPublishers].[PublisherId]
Left Join [DWPubsSales].[dbo].[DimDates]
On [Pubs].[dbo].[Titles].[pubdate] =
[DWPubsSales].[dbo].[DimDates].[Date]
```

58. Поставете този код в SQL command text полето на OLE DB източника и натиснете бутон Preview, за да потвърдите, че заявката е правилна.
59. Разгледайте страница Columns показваща избраните колони и затворете редактора на OLE DB източника от бутона OK.
60. Добавете OLE DB Destination в дизайнера и я преименувайте на Fill DimTitles OLE DB Destination.
61. Свържете с пътека източника и дестинацията на данните.
62. Конфигурирайте OLE DB дестинацията, така че да вкарва данни в таблица DIMTITLES в DWPubsSales.
63. Разгледайте страница Mappings и затворете редактора от бутона OK.
64. Кликнете с десен бутон в дизайнера и изберете опция Execute Task от контекстното меню, за да изпълните задачата.
65. След като изпълнението завърши, спрете дебъгера и верифицирайте, че в DIMTITLES има данни, използвайки SQL Management Studio.
66. За да **попълните факт FACTTITLESAUTHORS**, отворете таб Data Flow и изберете Fill FactTitlesAuthors Data Flow Task от падащото меню в горната част на екрана.
67. Добавете OLE DB Source в дизайнера и го преименувайте на Get TitlesAuthors Data OLE DB Source.
68. Отворете редактора на източника с двоен клик. Изберете мениджър DWPubsSales в полето OLE DB connection manager и опция SQL command в полето Data access mode.
69. Намерете коментар --2g) Get source data from pubs.dbo.titleauthor в SQL файла и го копирайте. Трябва да изглежда така:

```

Select
  [TitleKey] = DimTitles.TitleKey
, [AuthorKey] = DimAuthors.AuthorKey
, [AuthorOrder] = au_ord
From pubs.dbo.titleauthor
JOIN DWPubsSales.dbo.DimTitles
On pubs.dbo.titleauthor.Title_id =
DWPubsSales.dbo.DimTitles.TitleId
JOIN DWPubsSales.dbo.DimAuthors
On pubs.dbo.titleauthor.Au_id =
DWPubsSales.dbo.DimAuthors.AuthorId

```

70. Поставете този код в SQL command text полето на OLE DB източника и натиснете бутон Preview, за да потвърдите, че заявката е правилна.
71. Разгледайте страница Columns показваща избраните колони и затворете редактора на OLE DB източника от бутона OK.
72. Добавете OLE DB Destination в дизайнера и я преименувайте на Fill FactTitlesAuthors OLE DB Destination.
73. Свържете с пътека източника и дестинацията на данните.
74. Конфигурирайте OLE DB дестинацията, така че да вкарва данни в таблица FACTTITLESAUTHORS в DWPubsSales.
75. Разгледайте страница Mappings и затворете редактора от бутона OK.
76. Кликнете с десен бутон в дизайнера и изберете опция Execute Task от контекстното меню, за да изпълните задачата.
77. След като изпълнението завърши, спрете дебъгера и верифицирайте, че в FACTTITLESAUTHORS има данни, използвайки SQL Management Studio.
78. За да **попълните факт FACTSALES**, отворете таб Data Flow и изберете Fill FactSales Data Flow Task от падащото меню в горната част на екрана.
79. Добавете OLE DB Source в дизайнера и го преименувайте на Get Sales Data OLE DB Source.
80. Отворете редактора на източника с двоен клик. Изберете мениджър DWPubsSales в полето OLE DB connection manager и опция SQL command в полето Data access mode.
81. Намерете коментар --2h) Get source data from pubs.dbo.sales в SQL файла и го копирайте. Трябва да изглежда така:

```

Select
  [OrderNumber] = Cast(ord_num as nVarchar(50))
, [OrderDateKey] = DateKey
, [TitleKey] = DimTitles.TitleKey
, [StoreKey] = DimStores.StoreKey
, [SalesQuantity] = qty
From pubs.dbo.sales
JOIN DWPubsSales.dbo.DimDates
On pubs.dbo.sales.ord_date = DWPubsSales.dbo.DimDates.date
JOIN DWPubsSales.dbo.DimTitles
On pubs.dbo.sales.Title_id =
DWPubsSales.dbo.DimTitles.TitleId
JOIN DWPubsSales.dbo.DimStores
On pubs.dbo.sales.Stor_id =
DWPubsSales.dbo.DimStores.StoreId

```

82. Поставете този код в SQL command text полето на OLE DB източника и натиснете бутон Preview, за да потвърдите, че заявката е правилна.
83. Разгледайте страница Columns показваща избраните колони и затворете редактора на OLE DB източника от бутона OK.
84. Добавете OLE DB Destination в дизайнера и я преименувайте на Fill FactSales OLE DB Destination.
85. Свържете с пътека източника и дестинацията на данните.
86. Конфигурирайте OLE DB дестинацията, така че да вкарва данни в таблица FACTSALES в DWPubsSales.
87. Разгледайте страница Mappings и затворете редактора от бутона OK.
88. Кликнете с десен бутон в дизайнера и изберете опция Execute Task от контекстното меню, за да изпълните задачата.
89. След като изпълнението завърши, спрете дебъгера и верифицирайте, че в FACTSALES има данни, използвайки SQL Management Studio.

В тази задача конфигурирахте всички SSIS задачи използвани за извличането на данни от pubs транзакционната база данни, трансформирането им и зареждането им в data warehouse DWPubsSales. Също така тествахте всяка задача поотделно, за да се уверите, че работи правилно. Сега трябва да верифицираме, че целия SSIS пакет се изпълнява успешно като едно цяло.

ЗАДАЧА 3: ТЕСТВАНЕ НА SSIS ПАКЕТА

В тази задача ще тестваме че целия пакет се изпълнява успешно. За целта трябва да изчистим данните от нашия data warehouse и след това да изпълни пакета.

1. Отворете файл Create DWPubsSales Script.sql във Visual Studio и го изпълнете в SQL Server Management Studio (трябва да се намира в папка DWPubsSales на решението).
2. След няколко секунди трябва изпълнението да завърши и всички данни да са зачистени.
3. В Solution Explorer на решението кликнете с десен бутон върху файла на ETL пакета на име ETLProcessForDWPubsSales.dtsx и изберете Execute Package от контекстното меню, както е показано на **Error! Reference source not found..** Visual Studio ще стартира пакета в дебъг режим.
4. След като изпълнението приключи, спрете дебъгера и верифицирайте, че пакета е завършил успешно, сравнявайки го с показаното на **Error! Reference source not found..** В случай на проблем, потърсете причината, коригирайте конфигурацията и изпълнете пакета отново.
5. Верифицирайте, че всички таблици са заредени с данни чрез SQL Server Management Studio, използвайки следните заявки:

```
Select Top 100 * from dbo.DimAuthors
Select Top 100 * from dbo.DimStores
Select Top 100 * from dbo.DimPublishers
Select Top 100 * from dbo.DimDates
Select Top 100 * from dbo.DimTitles
Select Top 100 * from dbo.FactSales
Select Top 100 * from dbo.FactTitlesAuthors
```

6. Накрая по желание може да създадете бекъп на базата данни, за да сте сигурни, че ще разполагате с нея за следващото упражнение. В SQL Management Studio отворете New Query (Ctrl + N) прозорец и там изпълнете командата (уверете се, че пътя съществува):

```
Backup Database DWPubsSales
```

```
To Disk =  
'C:\prj\PublicationsIndustries\DWPubSales\DWPubSales_AfterETL.bak'  
With Init
```

При имплементиране на реални ETL процеси за клиенти е добре този процес да се повтори поне два пъти, за да сме сигурни, че повторните изпълнения също минават успешно. Пакетите на SSIS обикновено се изпълняват чрез автоматизирани задачи планирани през нощта (когато натоварването на инфраструктурата е по-малко), така че не бихме искали да ви извикат заради недостатъчно тестван пакет. За целите на упражнението обаче, това не е нужно.

С това завършихте създаването на ETL процеса за BI решението на Publications Industries. В следващите упражнения ще добавим SSAS и SSRS проекти към solution-a, с които да реализираме анализ на данните чрез куб и представянето им в доклади.