



عنوان مقاله: مبانی Event Handler در پکیج SSIS Microsoft

نویسنده مقاله: تیم فنی نیک آموز

تاریخ انتشار: فروردین ماه ۱۴۰۱

منبع: <https://nikamooz.com/event-handlers-ssis>

مقدمه

کنترل‌کننده‌های رویداد SSIS ساده‌ترین وسیله برای تبدیل یک اسکریپت SSIS به یک سیستم قابل اعتماد است که قابل بازرسی است، به شرایط خطا واکنش مناسب نشان می‌دهد، پیشرفت را گزارش می‌دهد و امکان ابزار دقیق و نظارت بر بسته‌های SSIS شما را فراهم می‌کند. پیاده‌سازی آنها آسان است و انعطاف پذیری زیادی را ارائه می‌دهند. راب شلدون یک بار دیگر مقدمه آسان و واضح را ارائه می‌دهد.

از زمان انتشار SQL Server ۲۰۰۵، SQL Server Integration Services (SSIS) ثابت کرده است که ابزاری موثر برای مدیریت عملیات استخراج، بارگذاری و تبدیل (ETL) است. با این حال، بیشتر مطالبی که در مورد توسعه بسته‌های SSIS پیدا می‌کنید، بر روی data flow و control flow تمرکز می‌کنند، همانطور که در استودیوی توسعه هوش تجاری (BIDS) ارائه می‌شوند. اما یکی دیگر از ویژگی‌های مهم و اغلب نادیده گرفته شده در بسته SSIS، کنترل‌کننده رویداد است. کنترل‌کننده‌های رویداد به شما این امکان را می‌دهند که اجزای SSIS را بر اساس هر اجرا و بر اساس هر رویداد اجرا کنید. به عنوان مثال، فرض کنید بسته شما شامل یک محفظه حلقه Foreach است. شما می‌توانید یک یا چند جزء SSIS را با هر رویدادی که توسط آن container ایجاد می‌شود در هنگام اجرا مرتبط کنید. این شامل رویدادهایی مانند OneError، OnInformation و OnPostExecute می‌شود. اجزایی که با رویدادهای container مرتبط می‌کنید از جریان control flow جدا هستند. در نتیجه، می‌توانید آن‌ها را به‌طور خاص بر اساس نیازهای کنترل‌کننده رویداد مرتبط پی‌کربندی کنید. با نشان دادن نحوه پی‌کربندی یک کنترل‌کننده رویداد، همه اینها واضح تر می‌شود.

نکته مهم:

در SSIS، یک فایل اجرایی هر مؤلفه ای است که به جریان control flow اضافه می‌کنید، به علاوه خود بسته. component به عنوان فایل‌های اجرایی package در child عمل می‌کنند. اگر Component را به یک Container اضافه کنید، container، اجرایی والد (اما Child فایل اجرایی package)، و Component درون Container، اجرایی child است. در این مقاله نحوه اضافه کردن Event handler به بسته SSIS را توضیح می‌دهم. بسته در این مورد، داده‌ها را در دو جدولی که من در پایگاه داده نمونه AdventureWorks ۲۰۰۸ در یک نمونه محلی از SQL Server ۲۰۰۸ ایجاد کردم، وارد می‌کند. اولین جدول، People، پس از استخراج داده‌ها از جدول Person.Person، هدف بسته خواهد بود. در پایگاه داده

AdventureWorks۲۰۰۸. جدول دوم، RunInfo، داده‌هایی را ذخیره می‌کند که توسط کنترل‌کننده‌های رویداد که من به بسته اضافه می‌کنم تولید می‌شود. اسکریپت Transact-SQL زیر شامل دستورات CREATE TABLE است که برای افزودن هر دو جدول ضروری است:

```
IF OBJECT_ID('People', 'U') IS NOT NULL
```

```
DROP TABLE dbo.People;
```

```
CREATE TABLE dbo.People
```

```
(
```

```
PersonID INT NOT NULL,
```

```
FirstName NVARCHAR(۵۰) NOT NULL,
```

```
LastName NVARCHAR(۵۰) NOT NULL,
```

```
CONSTRAINT PK_People PRIMARY KEY CLUSTERED (PersonID ASC)
```

```
);
```

```
IF OBJECT_ID('RunInfo', 'U') IS NOT NULL
```

```
DROP TABLE dbo.RunInfo;
```

```
CREATE TABLE dbo.RunInfo
```

```
(
```

```
RunID INT NOT NULL IDENTITY,
```

```
TaskID NVARCHAR(۵۰) NOT NULL,
```

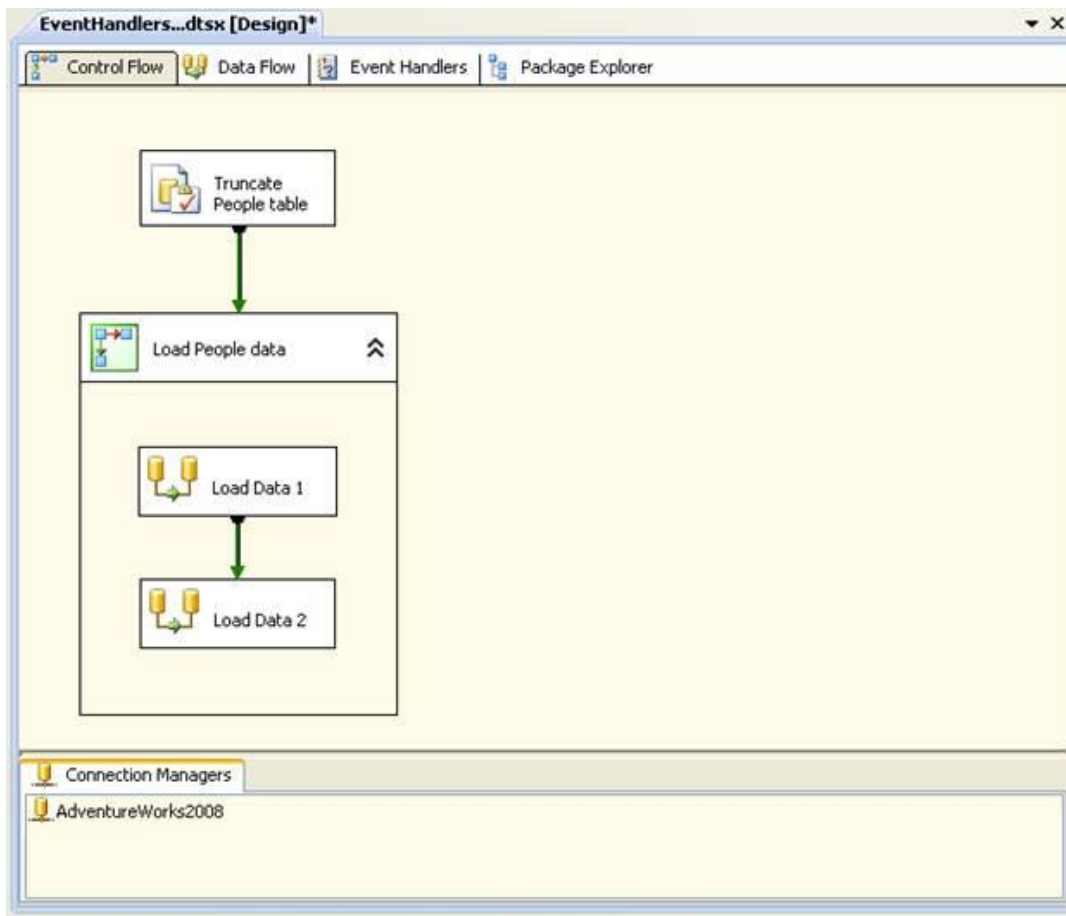
```
TaskName NVARCHAR(۵۰) NOT NULL,
```

```
TaskTime DATETIME NOT NULL DEFAULT(GETDATE())
```

```
CONSTRAINT PK_RunInfo PRIMARY KEY CLUSTERED (RunID ASC)
```

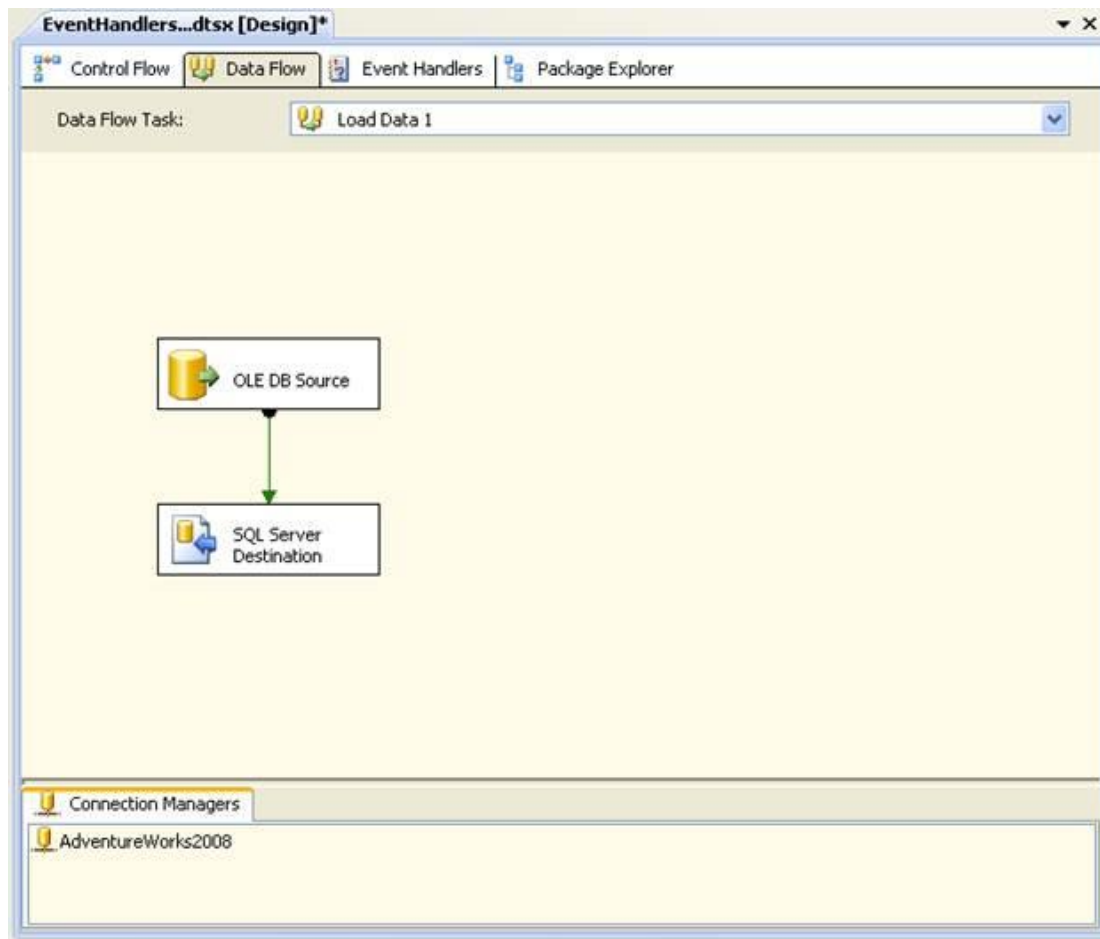
```
);
```

بعد از اینکه دو جدول را به پایگاه داده اضافه کردم، بسته SSIS را ایجاد کردم. شکل ۱ control flow بسته را پس از اضافه کردن اجزای لازم نشان می‌دهد. همانطور که شکل نشان می‌دهد، جریان کنترل شامل یک کار SQL Execute برای کوتاه کردن جدول People و شامل یک ظرف Sequence است که شامل دو data flow tasks است.



شکل ۱: تنظیمات control flow را نمایش می دهد

توجه داشته باشید که شکل ۱ - Connection Manager از دیتابیس Adventure Works ۲۰۰۸ را نشان می دهد، که یک OLEDB connection manager است که به پایگاه داده AdventureWorks۲۰۰۸ در نمونه محلی SQL Server ۲۰۰۸ متصل می شود. من از این connection manager برای همه اتصالات خود استفاده می کنم. بعد من دو Data Flow را تنظیم کردم. جریان داده ۱ Load Data که در شکل ۲ نشان داده شده است، از یک منبع OLE DB برای بازیابی داده ها از جدول Person از اسکیمای Person جدول و یک مقصد SQL Server برای درج داده ها در جدول People استفاده می کند.



تنظیمات load data از Data Flow

زمانی که source OLEDB را تنظیم کردیم از دستور DQL از نوع Select فیلدهای کد پرسنلی و نام و نام خانوادگی از اسکیمای Person جدول Person استخراج کرده به شرطی که رکوردهایی را می خواهیم که کد پرسنلی آن ها کمتر از ۱۰۰۰ باشد برای این منظور کوئری زیر را می نویسیم:

SELECT

BusinessEntityID,

FirstName,

LastName

FROM

Person.Person

WHERE

BusinessEntityID < ۱۰۰۰۰;

توجه داشته باشید که من فقط ردیف هایی را بازیابی می کنم که مقدار BusinessEntityID آنها کمتر از ۱۰۰۰۰ باشد. جریان داده ۲ Load Data با ۱ Load Data یکسان است، با این تفاوت که از عبارت SELECT زیر استفاده کردم:

برای این منظور کوئری زیر را می نویسیم:

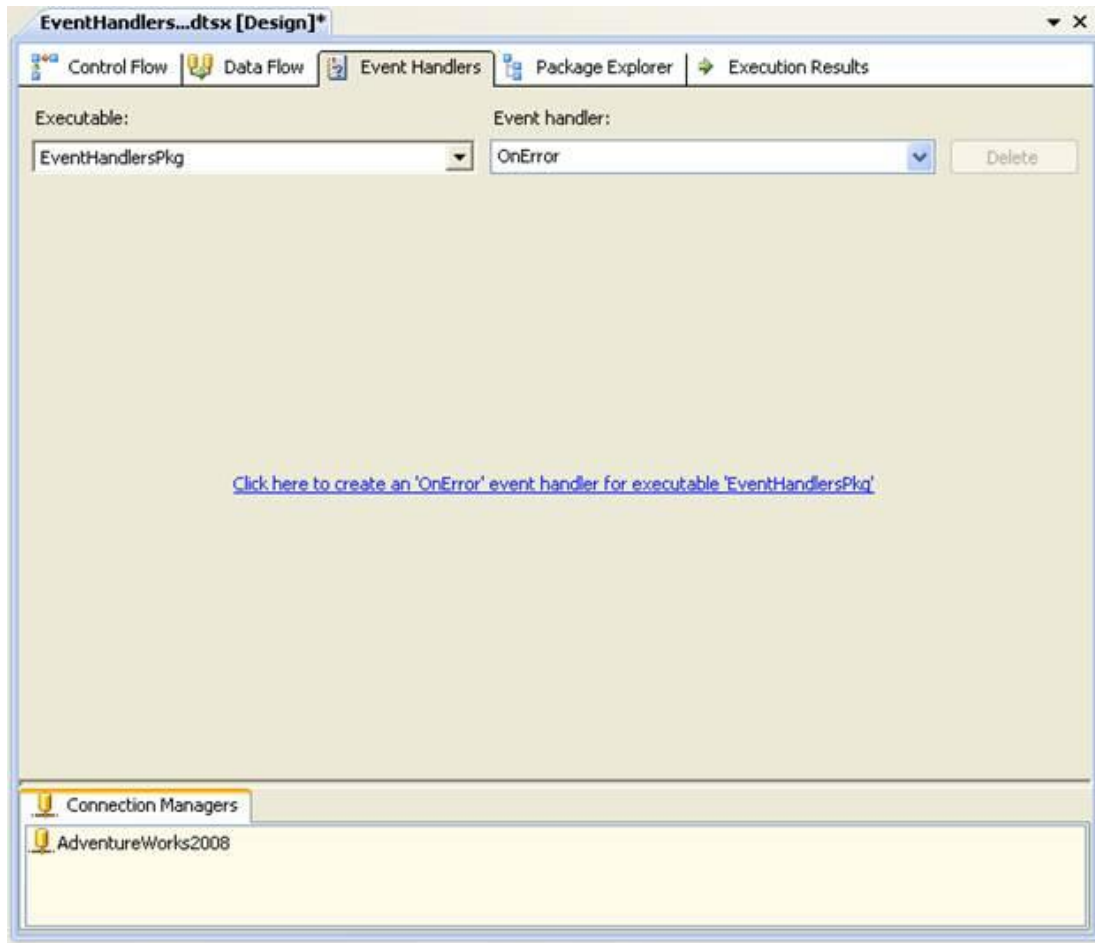
فیلدهای کد پرسنلی و نام و نام خانوادگی از اسکیمای Person جدول Person استخراج کرده به شرطی که رکوردهایی را می خواهیم که کد پرسنلی آن ها بیشتر از ۱۰۰۰ باشد برای این منظور کوئری زیر را می نویسیم.

```
SELECT
    BusinessEntityID,
    FirstName,
    LastName
FROM
    Person.Person
WHERE
    BusinessEntityID >= ۱۰۰۰۰;
```

همانطور که می بینید، این بار من فقط ردیف هایی را بازبایی می کنم که مقدار BusinessEntityID آنها بزرگتر یا مساوی ۱۰۰۰۰ است. من دو جریان داده را به این ترتیب تنظیم کردم تا ارتباط بین اجرایی و event handlers در package SSIS بهتر نشان دهم. می توانید بسته تکمیل شده را از حباب گفتار در بالای این مقاله دانلود کنید یا به سادگی می توانید package را خودتان ایجاد کنید. اگر مطمئن نیستید که چگونه یک Package SSIS کنید یا هر یک از این مؤلفه ها را پیگیربندی کنید، حتماً SQL Server Books Online را بررسی کنید. هنگامی که Package خود را تنظیم کردید، آماده هستید تا event handlers را اضافه کنید.

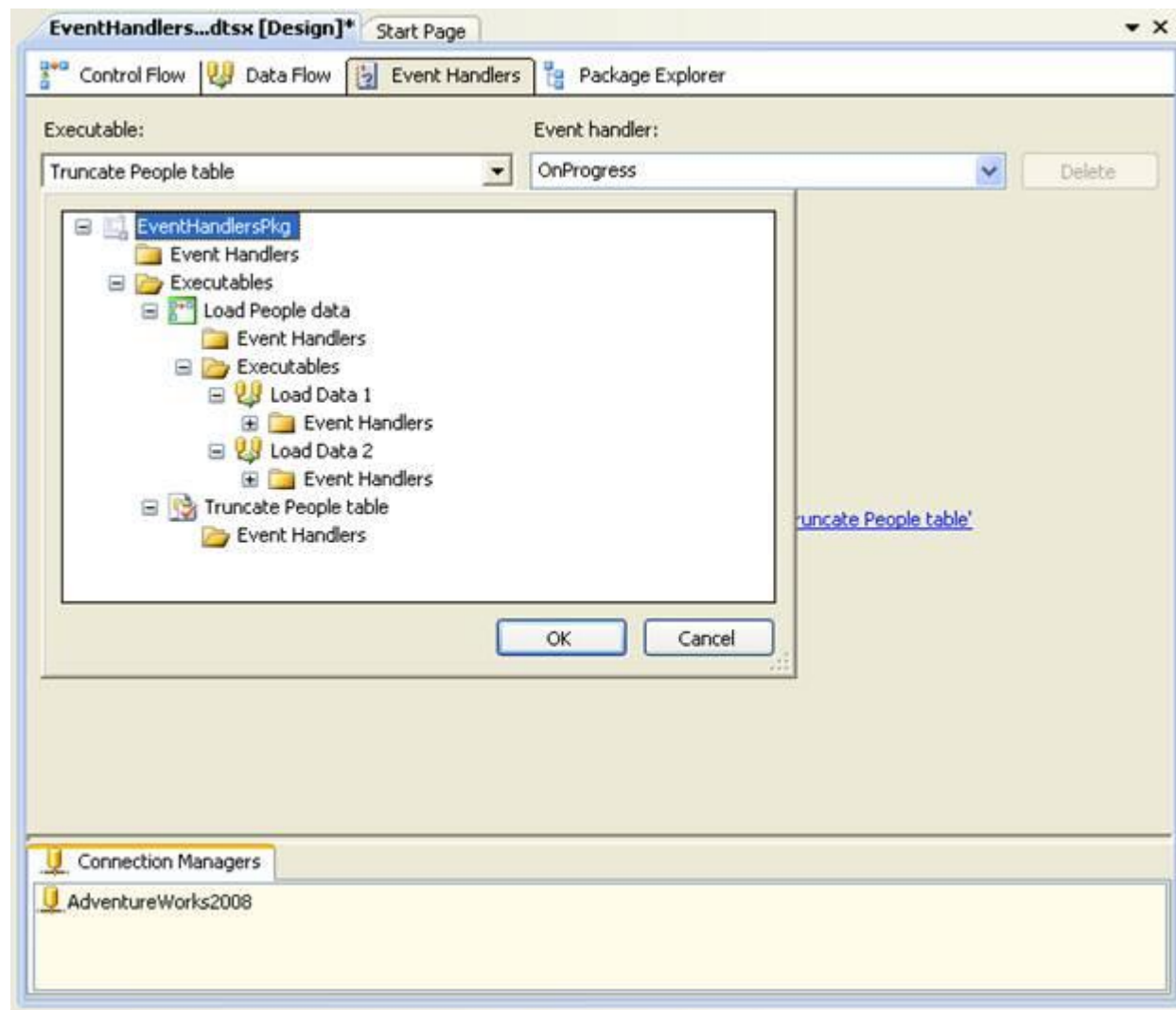
انتخاب یک اجرا و رویداد

کنترل کننده های رویداد خود را در تب Event Handlers در SSIS Designer پیگیربندی می کنید. برگه نشان داده شده در شکل ۳، دسترسی به فایل های اجرایی بسته و رویدادهای مرتبط با هر فایل اجرایی را فراهم می کند. این برگه همچنین سطح طراحی لازم برای افزودن مؤلفه ها به کنترل کننده رویداد را فراهم می کند، درست همانطور که اجزایی را به جریان کنترل اضافه می کنید.



شکل ۳: دسترسی به تنظیمات Event Handler

هنگامی که برای اولین بار به تب Event Handlers دسترسی پیدا می کنید، فایل اجرایی انتخاب شده خود بسته است که در این مورد، من آن را EventHandlersPkg نامیده ام. علاوه بر این، رویداد انتخاب شده OneError است. در نتیجه، هر مؤلفه ای که در این مرحله به سطح طراحی اضافه می کنید، مختص این ترکیب از این جفت اجرایی و رویداد خواهد بود. برای مشاهده تمامی فایل های اجرایی، روی فلش رو به پایین در کادر متن Executable کلیک کنید و سپس لیست فایل های اجرایی را مانند شکل ۴ گسترش دهید.



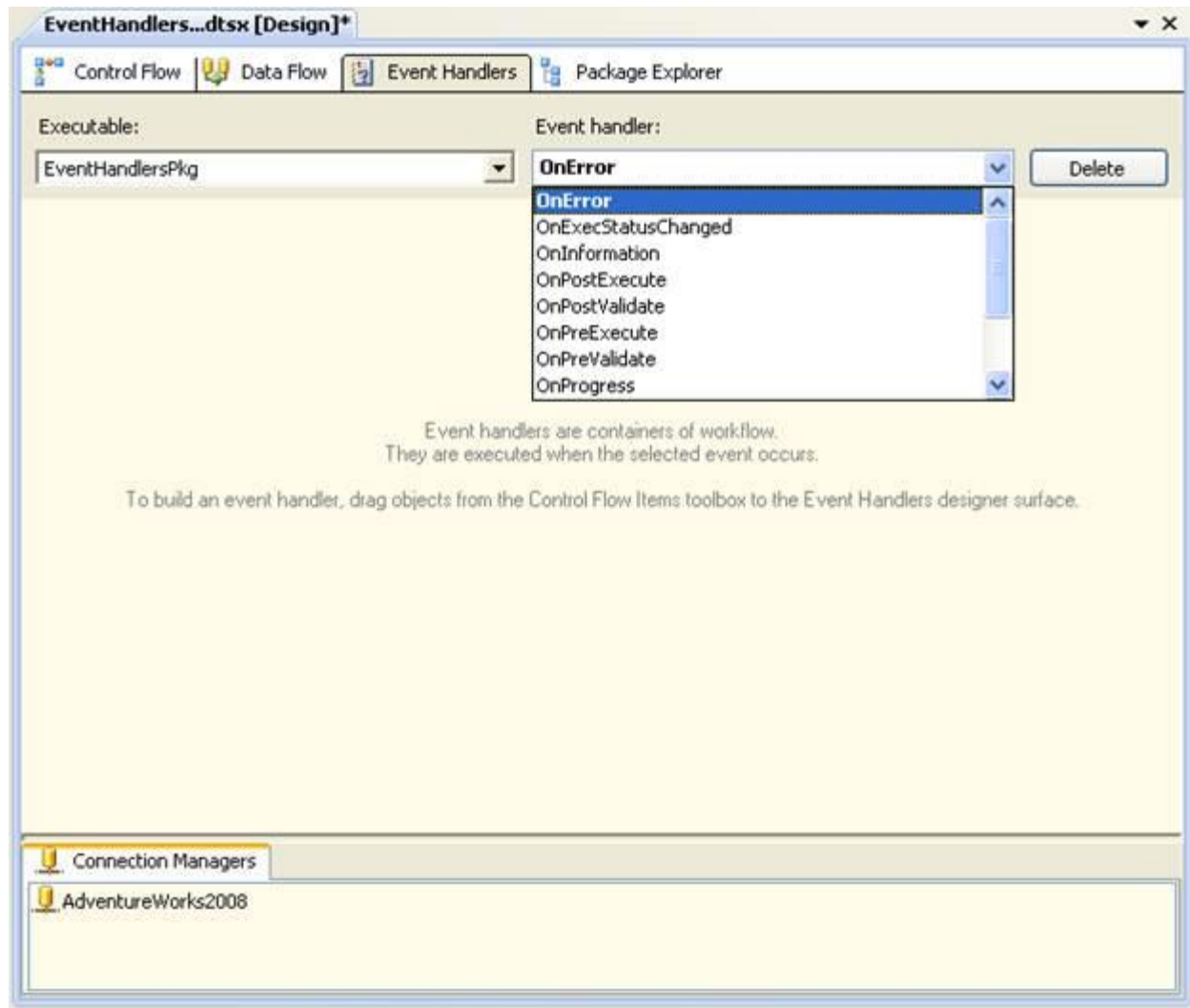
شکل ۴: مشاهده فایل های اجرایی package

توجه داشته باشید که فایل های اجرایی به صورت سلسله مراتبی فهرست شده اند، با EventHandlersPkg در بالای سلسله مراتب و اجرای وظیفه SQL (جدول Truncate People) و ظرف (Load People data) Sequence در سطح دوم سلسله مراتب. در سطح سوم، زیر ظرف Sequence، دو وظیفه data flow (Load Data ۱ و Load Data ۲) قرار دارند. برای هر فایل اجرایی، پوشه ای به نام Event Handlers فهرست شده است. هر event handler که برای یک فایل اجرایی پیکربندی می‌کنید در آن پوشه فهرست می‌شود و کنترل‌کننده‌های رویداد بر اساس رویداد مرتب شده‌اند.

نکته مهم :

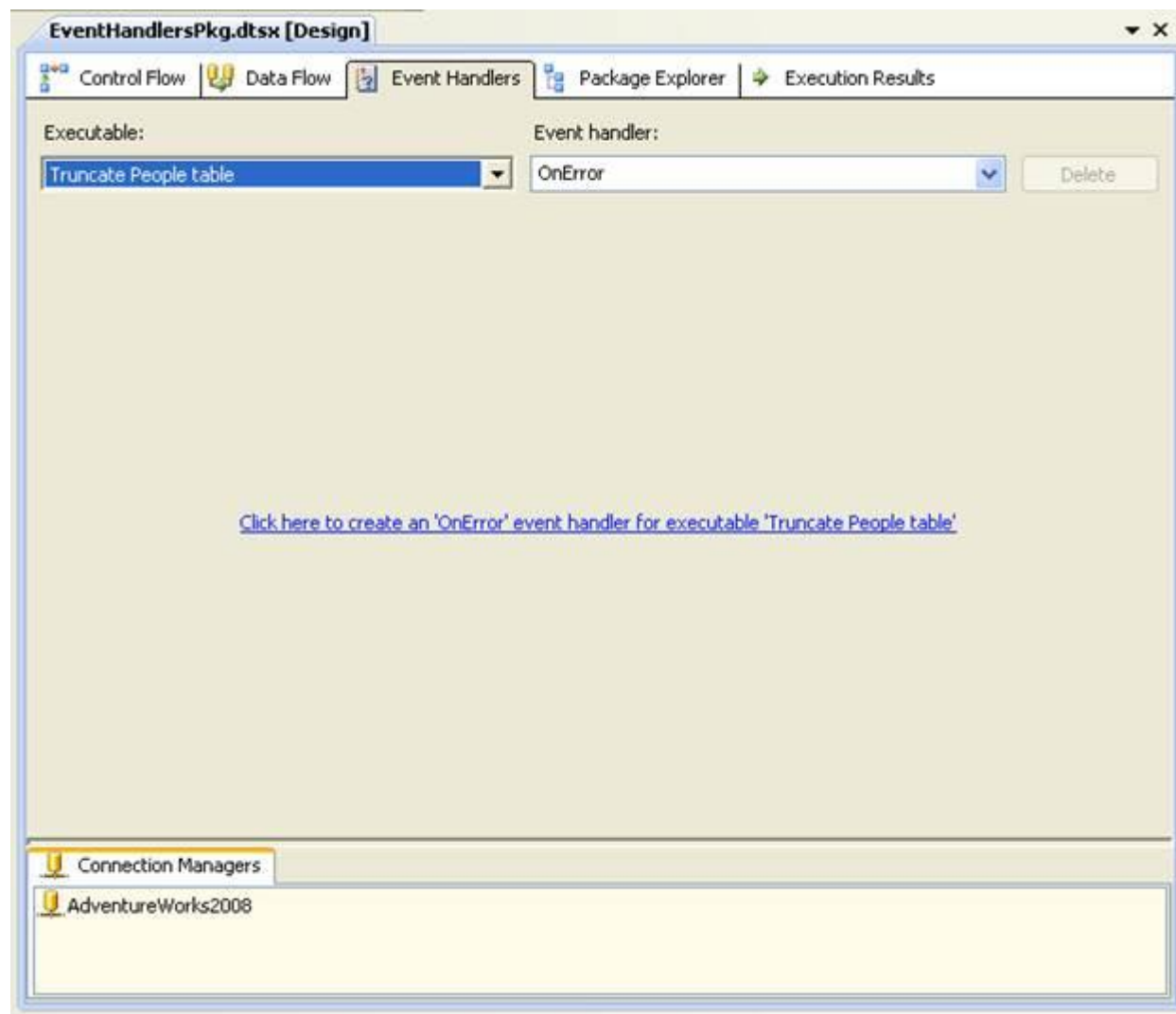
منابع، تبدیل‌ها و مقصدهایی که به یک جریان داده اضافه می‌کنید، قابل اجرا نیستند. همه آنها بخشی از فایل اجرایی Data Flow هستند، به همین دلیل است که این مؤلفه‌ها در لیست فایل های اجرایی در تب Event Handlers قرار داده نشده اند

برای هر فایل اجرایی، لیستی از رویدادها را در لیست مدیریت رویداد پیدا خواهید کرد. لیست شامل تمام رویدادهای مرتبط با فایل اجرایی انتخاب شده است. برای انتخاب یک رویداد برای یک فایل اجرایی، روی فلش رو به پایین در لیست کنترل رویداد که در شکل ۵ نشان داده شده است کلیک کنید و سپس رویداد را انتخاب کنید.



شکل ۵: نمایی از Event های Package

هنگام کار بر روی سطح طراحی تب Event Handlers، همیشه با یک جفت رویداد اجرایی خاص کار می کنید. این بدان معناست که برای پیگیری یک کنترلر رویداد، ابتدا باید یک فایل اجرایی و سپس رویداد را انتخاب کنید. به عنوان مثال، اگر به شکل ۶ مراجعه کنید، خواهید دید که من جدول اجرایی Truncate People را انتخاب کرده ام و سپس رویداد OnError را انتخاب کرده ام. در نتیجه، هر کامپوننتی که به این جفت رویداد اجرایی اضافه کنم، هر زمان که جدول اجرایی Truncate People یک رویداد OnError ایجاد کند، اجرا خواهد شد.



شکل ۶: انتخاب جداول اجرایی و Event Handler

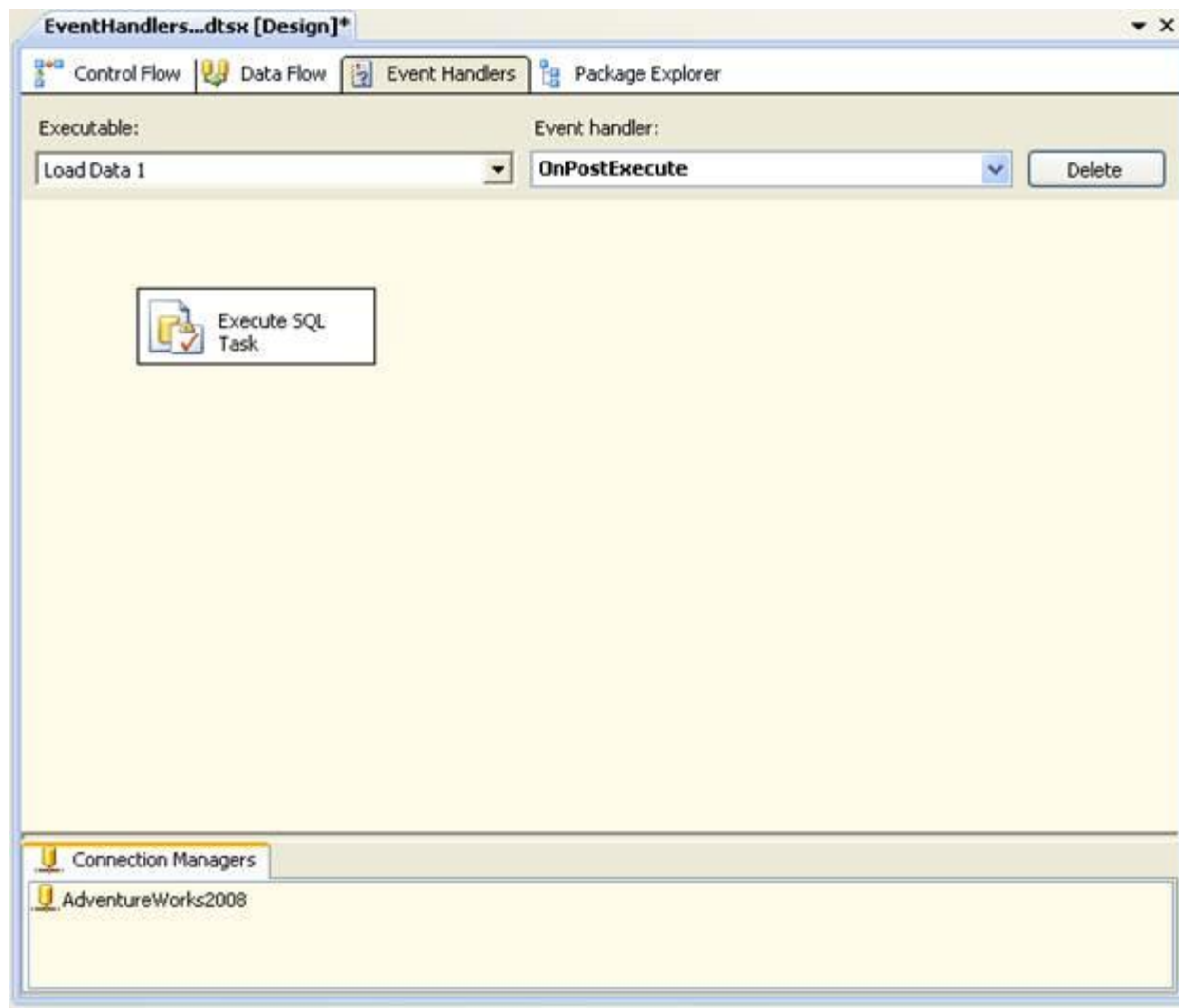
تنظیمات Event Handler

اگر به شکل ۶ مراجعه کنید، متوجه خواهید شد که سطح طراحی شامل پیوندی است که به شما دستور می دهد روی آن کلیک کنید تا یک کنترل کننده رویداد برای آن رویداد خاص و قابل اجرا ایجاد کنید. برای هر جفت رویداد اجرایی که کنترل کننده رویداد برای آن پیگیری نشده است، این پیوند به شما نشان داده می شود. برای ایجاد مدیریت رویداد و افزودن هر مؤلفه، باید روی این پیوند کلیک کنید.

نکته مهم:

هنگامی که روی پیوند روی سطح طراحی برای یک جفت رویداد اجرایی خاص کلیک کردید، حتی اگر مؤلفه ای اضافه نکنید، یک کنترل کننده رویداد ایجاد می شود. برای حذف یک کنترل کننده رویداد برای یک جفت اجرایی-رویداد خاص، روی دکمه حذف در سمت راست کادر نوشتاری کنترل کننده رویداد کلیک کنید.

اولین کنترل کننده رویدادی که ایجاد می کنیم برای فایل اجرایی ۱ Load Data و رویداد OnPostExecute است. پس از انتخاب جفت رویداد اجرایی و کلیک روی پیوند، آماده حرکت هستید. همانطور که در شکل ۷ نشان داده شده است، برای بسته EventHandlersPkg، یک Execute SQL را اضافه کردم.



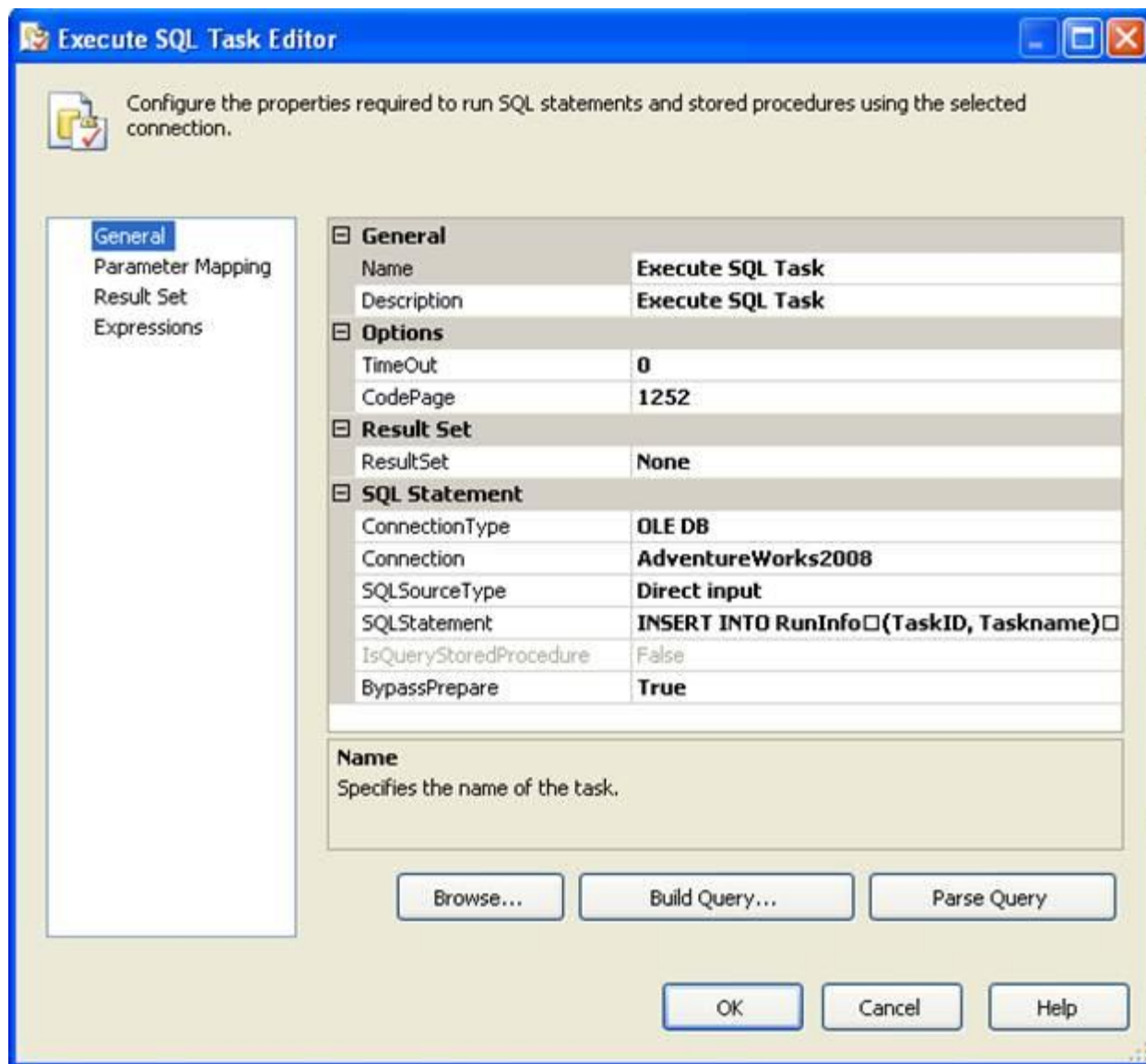
شکل ۷: اضافه کردن task اجرایی SQL در Event handler

هنگامی که یک رویداد کنترل کننده را پیکربندی می کنید، می توانید از هر یک از متغیرهای سیستم یا کاربر موجود برای فایل اجرایی استفاده کنید، بنابراین بیاید به متغیرهای موجود برای فایل اجرایی ۱ Load Data نگاه کنیم. برای مشاهده این متغیرها، با کلیک بر روی متغیرها در منوی SSIS، پنجره Variables را در SSIS Designer باز کنید. سپس، در پنجره Variables، روی Show System Variables کلیک کنید تا تمام متغیرهای موجود برای اجرای رویداد handler شما فهرست شود، همانطور که در شکل ۸ نشان داده شده است.

Name	Scope	Data Type	Value
EventHandlerStartTime	OnPostExecute	DateTime	4/17/2011 7:09 PM
LocaleID	OnPostExecute	Int32	English (United States)
Propagate	OnPostExecute	Boolean	True
SourceDescription	OnPostExecute	String	
SourceID	OnPostExecute	String	
SourceName	OnPostExecute	String	
SourceParentGUID	OnPostExecute	String	
ContainerStartTime	Load Data 1	DateTime	4/17/2011 7:09 PM
CreationName	Load Data 1	String	SSIS.Pipeline.2
LocaleID	Load Data 1	Int32	English (United States)
ParentContainerGUID	Load Data 1	String	{45868B32-A8C2-468F-8B88-61B6FF0D5EA8}
TaskID	Load Data 1	String	{85B4ED54-D20D-4E90-B60C-E0151D7B1348}
TaskName	Load Data 1	String	Load Data 1
TaskTransactionOption	Load Data 1	Int32	1
LocaleID	Load People data	Int32	English (United States)
ParentContainerGUID	Load People data	String	{036484F0-969C-42D9-BA78-53508780EC70}
CancelEvent	EventHandlersPkg	Int32	0
ContainerStartTime	EventHandlersPkg	DateTime	4/17/2011 7:09 PM
CreationDate	EventHandlersPkg	DateTime	4/14/2011 3:53 PM
CreatorComputerName	EventHandlersPkg	String	BOBE024
CreatorName	EventHandlersPkg	String	BOBE024\Administrator
ExecutionInstanceGUID	EventHandlersPkg	String	{8C8E0D55-C907-4A20-A2BC-2D4B78C1479D}
FailedConfigurations	EventHandlersPkg	String	
InteractiveMode	EventHandlersPkg	Boolean	False
LastModifiedProductV...	EventHandlersPkg	String	10.0.2531.0

شکل ۸: مشاهده متغیرها در variable pane

در این مورد، می‌خواهیم از متغیرهای سیستم SourceName و SourceID برای شناسایی مؤلفه‌هایی که رویدادهای OnPostExecute را ایجاد می‌کنند، استفاده کنیم، این همان چیزی است که به جدول RunInfo وارد می‌کنیم. ما متغیرها را در کار Execute SQL اضافه می‌کنیم، بنابراین بیا باید به نحوه پیکربندی آن کار نگاه کنیم. شکل ۹ صفحه عمومی ویرایشگر Execute SQL Task را نشان می‌دهد.



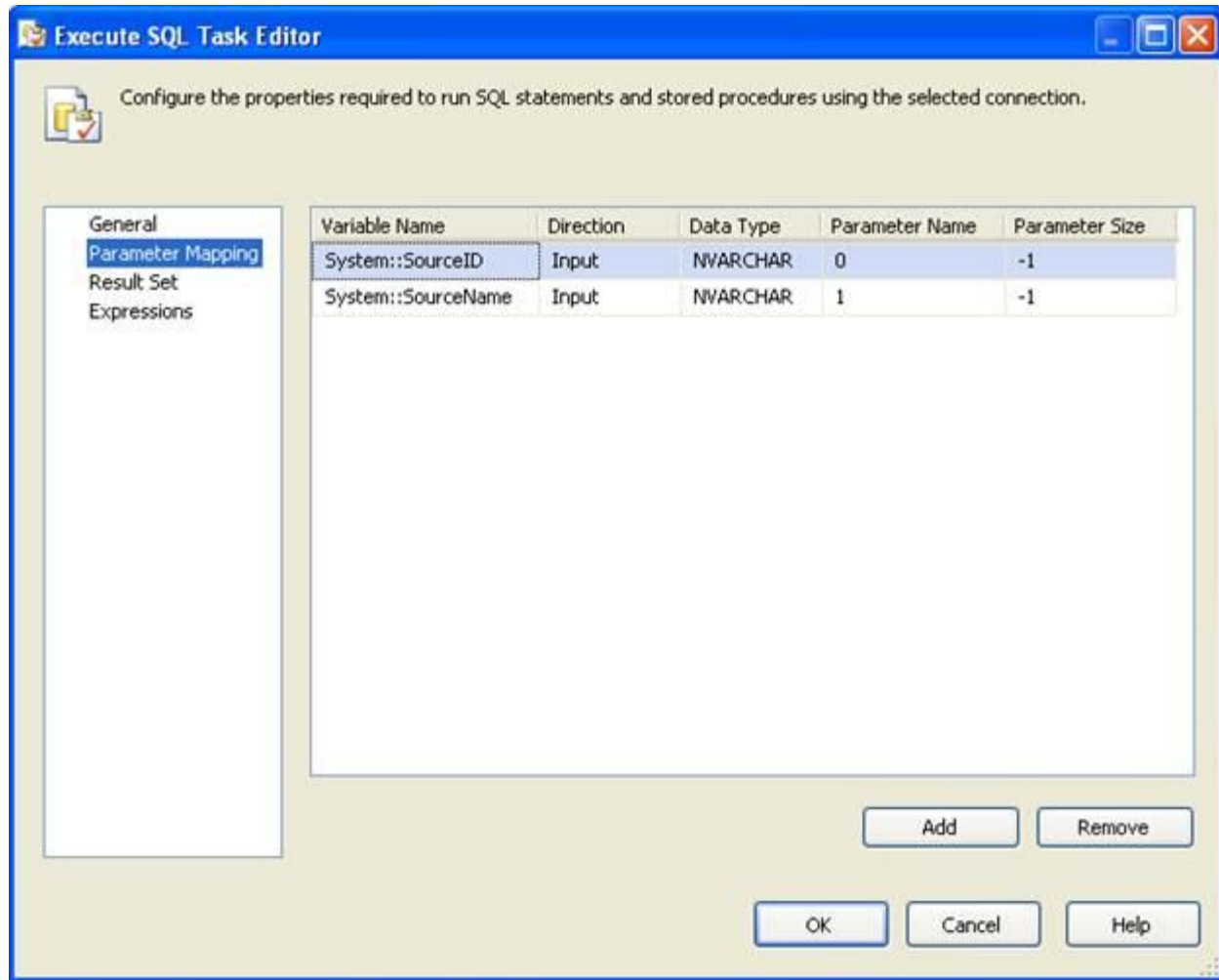
شکل ۹: پنجره Execute SQL Task Editor

نکته مهم: توجه داشته باشید که در Connection Manager در دیتابیس AdventureWorks2008 را در ویژگی Connection مشخص کرده ام. سپس، در ویژگی SQLStatement، عبارت INSERT زیر را اضافه کردم:
حال در ادامه با استفاده از دستور insert into-values به جدول Run info به فیلدهای task id, task name مقادیر ؟ تخصیص می دهیم

```
INSERT INTO RunInfo
(TaskID, Taskname)
VALUES (?, ?);
```

همانطور که می بینید، من دو علامت سوال را در بند VALUES قرار داده ام. نگهدارنده‌ها به شما اجازه می‌دهند که مقادیر متغیرهای سیستم SourceID و SourceName را در جدول RunInfo وارد کنید. با این حال، برای انجام این کار، باید متغیرها

را نیز به عبارت نگاشت کنید. شکل ۱۰ صفحه Parameter Mapping ویرایشگر Execute SQL Task را نشان می دهد که شامل فهرستی برای هر متغیری است که توسط عبارت INSERT استفاده می شود.



شکل ۱۰: پنجره parameter mapping برای اجرای SQL Task Editor

همانطور که شکل ۱۰ نشان می دهد، هر دو متغیر متغیرهای ورودی هستند که با نوع داده NVARCHAR پیگیربندی شده اند. علاوه بر این، نام متغیر اول (SourceID) و نام متغیر دوم (SourceName) است. این از قراردادهای نامگذاری لازم برای ارسال مقادیر پارامتر به عبارت INSERT پیروی می کند. این تمام چیزی است که برای پیگیربندی Execute SQL وجود دارد. و این تنها مؤلفه ای است که به فایل اجرایی ۱ Load Data اضافه کردم. با این حال، من همان رویداد را برای فایل اجرایی ۲ Load Data پیگیربندی کردم و سپس یک کار Execute SQL را به کنترل کننده رویداد اضافه کردم، درست مانند وظیفه در ۱ Load Data تنظیم شده است. این بدان معناست که وقتی بسته SSIS اجرا می شود، دو رویداد را اجرا می کند. کنترل کننده ها، یکی برای هر جریان داده.

اجرای Package ssis

هنگامی که کنترل کننده رویداد خود را پیکربندی کردید، آماده اجرای بسته SSIS هستید. اجرای بسته ای که شامل کنترل کننده رویداد است با اجرای هر نوع بسته دیگری تفاوتی ندارد. تا زمانی که قابل اجرا رویدادی را که اجزای SSIS برای آن پیکربندی شده اند صادر کند، کنترل کننده رویداد اجرا می شود. این بدان معناست که در این مورد، تا زمانی که وظایف جریان داده با موفقیت اجرا شوند، رویدادهای OnPostExecute صادر می شوند و وظایف SQL اجرا می شوند. سپس عبارات INSERT در آن وظایف، اطلاعات متغیر را به جدول RunInfo اضافه می کند.

پس از اینکه اولین بار بسته EventHandlersPkg را اجرا کردم، جدول RunInfo را جستجو کردم و نتایج زیر را دریافت کردم:

After I ran the **EventHandlersPkg** package the first time, I queried the RunInfo table and received the following results:

RunID	TaskID	TaskName	TaskTime
1	{85B4ED54-D20D-4E90-B60C-E0151D7B1348}	Load Data 1	2011-04-17 19:25:13.370
2	{057C0C52-99B1-4B96-BDC8-923A6A85CCBE}	Load Data 2	2011-04-17 19:25:13.670

As you can see, the task ID and task names have been added to the table, and as expected, there is one row for each task. If you run the package multiple times, you will see additional rows.

Although the example I've demonstrated in this article is very basic, it does show you the power of event handlers to capture a variety of information under specific circumstances. And you can perform other actions as well. For example, in addition to being able to set up a system to audit your packages, you can take such steps as sending an email if an executable issues an **OnError** event. SSIS event handlers are flexible and provide many options for auditing and monitoring your SSIS packages. And given how easy they are to implement, it is well worth the time and effort to take full advantage of all that event handlers have to offer.

Rate this article Subscribe to our fortnightly newsletter

همانطور که مشاهده می کنید task id و task name به جدول اضافه شده است و همانطور که انتظار می رود برای هر کار یک ردیف وجود دارد. اگر بسته را چندین بار اجرا کنید، ردیف های اضافی را مشاهده خواهید کرد.

اگرچه مثالی که در این مقاله نشان دادم بسیار ابتدایی است، اما به شما نشان می دهد که کنترل کننده های رویداد می توانند اطلاعات مختلفی را تحت شرایط خاص ثبت کنند. و می توانید اقدامات دیگری را نیز انجام دهید. به عنوان مثال، علاوه بر اینکه می توانید سیستمی برای بازرسی بسته های خود راه اندازی کنید، می توانید اقداماتی مانند ارسال ایمیل در صورتی که یک فایل اجرایی رویداد OnError را صادر کند، انجام دهید. کنترل کننده های رویداد SSIS انعطاف پذیر هستند و گزینه های زیادی را برای ممیزی و نظارت بر پکیج SSIS شما ارائه می دهند. و با توجه به اینکه پیاده سازی آنها چقدر آسان است، ارزش زمان و تلاش برای استفاده کامل از تمام مواردی که گردانندگان رویداد ارائه می دهند را دارد.

<https://www.red-gate.com/simple-talk/databases/sql-server/bi-sql-server/ssis-event-handlers-basics/#:~:text=SSIS%20event%20handlers%20are%20the,a%20great%20deal%20of%20flexibility>