In fact, each column in the data flow has been implicitly converted to an SSIS type. Table 1 shows the data type of the source data and the SSIS type in which it has been converted.

|  |  |  |
| --- | --- | --- |
| **Column** | **SQL Server data type** | **SSIS data type** |
| **ProductID** | **INT** | **DT\_I4** (four-byte signed integer) |
| **Name** | **NVARCHAR(50)** | **DT\_WSTR** (Unicode string) |
| **MakeFlag** | **BIT** | **DT\_BOOL** (Boolean) |
| **ReorderPoint** | **SMALLINT** | **DT\_I2** (two-byte signed integer) |
| **ListPrice** | **MONEY** | **DT\_CY** (currency) |
| **Weight** | **DECIMAL(8,2)** | **DT\_NUMERIC** (numeric) |
| **ProductLine** | **NCHAR(2)** | **DT\_WSTR** (Unicode string) |
| **SellStartDate** | **DATETIME** | **DT\_DBTIMESTAMP** (database timestamp) |
| **EndDate** | **DATE** | **DT\_DBDATE** (database date) |
| **rowguid** | **UNIQUEIDENTIFIER** | **DT\_GUID** (unique identifier) |

Table 1: Implicit conversions from SQL Server to SSIS

**Convert**

@[User::DateProc] = (DT\_WSTR, 4)YEAR( @[$Package::Date1] ) +

(DT\_WSTR, 2)MONTH( @[$Package::Date1] ) +

(DT\_WSTR, 2)DAY( @[$Package::Date1] )

**FOR-LOOP DATES**

Init value @[User::StartValue]

Condition DATEDIFF("Day", @[User::StartValue] , @[User::EndValue] ) >=0

Another @[User::StartValue] = DATEADD("Day",1, @[User::StartValue])

(DT\_STR, 4, 1252)YEAR(GETDATE()) +

RIGHT("0" + (DT\_STR, 2, 1252)MONTH(GETDATE()), 2) +

RIGHT("0" + (DT\_STR, 2, 1252)DAY(GETDATE()), 2)

or

(DT\_WSTR, 4)YEAR(GETDATE()) +

RIGHT("0" + (DT\_WSTR, 2)MONTH(GETDATE()),2) +

RIGHT("0" + (DT\_WSTR, 2)DAY(GETDATE()), 2)

(DT\_DBTIMESTAMP)(SUBSTRING(dateCreated,24,5) + "-" + SUBSTRING(dateCreated,4,4) + "-" +

"insert into DatesDemo values ( '" +   
(DT\_STR, 4, 1252) DATEPART("yyyy", @[System::StartTime]) + "-" +  
(DT\_STR, 4, 1252) DATEPART("mm", @[System::StartTime]) + "-" +  
(DT\_STR, 4, 1252) DATEPART("dd", @[System::StartTime]) + " " +  
(DT\_STR, 4, 1252) DATEPART("hh", @[System::StartTime]) + ":" +  
(DT\_STR, 4, 1252) DATEPART("mi", @[System::StartTime]) + ":" +  
(DT\_STR, 4, 1252) DATEPART("ss", @[System::StartTime]) +  
"')"

"Range of dates where from " + SUBSTRING( (DT\_WSTR,50) (@[$Package::sDate] ) ,1,10)