



SSIS Introduction

Sample manual - first two chapters



Wise Owl
Training

TABLE OF CONTENTS (1 of 6)

1	INTRODUCING SSIS	Page
1.1	Overview of SSIS	8
	<i>SSIS Packages and Solution Explorer</i>	8
	<i>Control Flow</i>	8
	<i>Data Flow</i>	9

2	USING VISUAL STUDIO	Page
2.1	Starting Visual Studio	10
	<i>Visual Studio and SQL Server Data Tools</i>	10
	<i>Running Visual Studio</i>	10
2.2	Creating a Project	11
2.3	Visual Studio Windows	12
	<i>The Solution Explorer, Properties and SSIS Toolbox Windows</i>	12
	<i>Floating Windows</i>	13
	<i>Auto-hiding Windows</i>	13

3	PACKAGES	Page
3.1	Working with Packages	14
	<i>Creating a Package</i>	14
	<i>Renaming Packages</i>	14
	<i>Opening or Editing Packages</i>	15
	<i>Viewing a Package's XML</i>	15
	<i>Saving a Package</i>	16
	<i>Closing Individual Packages</i>	16
	<i>Closing Multiple Packages</i>	16
	<i>Copying Packages</i>	16
3.2	The Package Tabs	17
3.3	Running a Package	18
	<i>Executing a Single Package</i>	18
	<i>Making a Package the Start-up Object and Debugging a Project</i>	18
	<i>Monitoring Package Execution</i>	19
	<i>Using the Progress Window</i>	19
3.4	Stopping Running a Package	20

4	TASKS	Page
4.1	Working with Tasks	21
	<i>Adding a Task</i>	21
	<i>Renaming a Task</i>	21
	<i>Editing a Task</i>	22
	<i>Executing an Individual Task</i>	22
	<i>Deleting and Suspending (Disabling) Tasks</i>	23
4.2	Working with Task Diagrams	24
	<i>Selecting Tasks</i>	24
	<i>Moving Tasks</i>	24
	<i>Auto-layout of Diagrams</i>	25
	<i>Zooming In and Out</i>	25
4.3	Annotations	26
4.4	Grouping Tasks	27
	<i>Grouping Tasks</i>	27
	<i>Grouping using Sequence Containers</i>	28

5	DATA FLOW	Page
5.1	Introducing Data Flow	29
	<i>Sources, Transforms and Destinations This and Subsequent Chapters</i>	29
5.2	Data Flow Tasks	30
	<i>Creating Data Flow Tasks</i>	30
	<i>Switching to Data Flow Tasks</i>	30
5.3	Connections	31
	<i>Types of Connection</i>	31
	<i>Project-Level and Package-Level Connections</i>	31
	<i>Creating Connections in Advance</i>	32
	<i>Creating Connections as Part of Sources/Destinations</i>	32
5.4	Creating Our Package	33
5.5	The Advanced Editor	34

TABLE OF CONTENTS (2 of 6)

6	WORKING WITH SQL SERVER	Page
6.1	Creating a SQL Server Connection	35
6.2	Creating a SQL Server Source/Destination	36
	<i>Step 1 – Creating the SQL Server Source/Destination</i>	36
	<i>Step 2 – Choosing the Table or Query</i>	37
	<i>Step 3 – Mapping Columns</i>	37
6.3	Creating SQL Server Tables	38
6.4	Execute SQL Tasks	39
	<i>Step 1 - Adding an Execute SQL Task</i>	39
	<i>Step 2 – Assign a Connection</i>	39
	<i>Step 3 – Enter a SQL Statement</i>	40
	<i>Step 4 – Optionally, Parse your SQL Command</i>	40

7	WORKING WITH FLAT FILES	Page
7.1	Examples for this Chapter	41
7.2	Existing Flat File - Source	42
	<i>Connecting to a Flat File</i>	42
	<i>Configuring Rows and Columns</i>	43
	<i>Creating the Flat File Source</i>	43
7.3	Existing Flat File – Destination	44
7.4	Creating New Flat Files	44
	<i>Step 1 – Creating a Destination</i>	45
	<i>Step 2 – Starting a New Connection</i>	45
	<i>Step 3 – Choose a File Format</i>	46
	<i>Step 4 – Browsing to a Folder</i>	46
	<i>Step 5 – Choosing a File Name</i>	47
	<i>Step 6 – Configuring the New File (Rows)</i>	47
	<i>Step 7 – Configure the New File (Columns)</i>	48
	<i>Step 8 – Configuring Mappings</i>	48
	<i>Step 9 – Running your Package</i>	48
7.5	Using Multiple Flat Files	49
	<i>Our Example</i>	49
	<i>Adding the MultiFlatFile Connector</i>	50
	<i>Using a MultiFlatFile Connector</i>	51

8	WORKING WITH EXCEL	Page
8.1	Excel Workbooks – Overview	52
	<i>The Example Used in this Chapter</i>	52
	<i>What Can Go Wrong</i>	52
8.2	Importing/Exporting using Excel	53
	<i>Step 1 – Creating an Excel Connection</i>	53
	<i>Step 2 – Creating a Source or Destination</i>	54
	<i>Step 3 – Mapping Columns</i>	54
8.3	The 32-Bit Problem	55
	<i>The Problem</i>	55
	<i>The Solution</i>	56

9	DATA VIEWERS	Page
9.1	Using Data Viewers	57
9.2	Creating Data Viewers	58
	<i>The Data Viewer Buffer</i>	58

10	TRANSFORMS	Page
10.1	Adding Transforms	59
10.2	Sorting / Getting Unique Rows	60
10.3	Sampling	61
	<i>The Sampling Transforms</i>	61
	<i>Incorporating Sampling Transforms into Data Flow</i>	62
10.4	Combining and Splitting Data	63
	<i>Introduction to the Transforms</i>	63
	<i>An Example using Multicast and Union All Transforms</i>	64
	<i>Configuring Union All Transforms</i>	64
10.5	Aggregating Data	65

11	DATA TYPES	Page
11.1	The Data Types in SSIS	66
	<i>The Three Data Type Families in SSIS</i>	66
	<i>Data Types Listed in this Chapter</i>	66
11.2	SSIS Data Types	67
	<i>String Data Types</i>	67
	<i>Integer Data Types</i>	67
	<i>Boolean or Logical Data Types</i>	67
	<i>Non-Integer Data Types</i>	68
	<i>Date/time Data Types</i>	68
	<i>Time Data Types</i>	68

TABLE OF CONTENTS (3 of 6)

12	DATA CONVERSION TRANSFORMS	Page
12.1	Overview of Data Conversion	69
12.2	Creating Data Conversion Transforms	70
	<i>Our Example: Importing Excel Unicode Data to Varchar Columns</i>	70
	<i>Adding a Data Conversion Transform</i>	70
	<i>Configuring a Data Conversion Transform</i>	71
	<i>Mapping Data Conversion Transform Columns</i>	72
	<i>The Complete Package</i>	72

13	VARIABLES	Page
13.1	Overview of Variables	73
	<i>bles</i>	73
13.2	Working with Variables	74
	<i>The Variables Window</i>	74
	<i>Creating a Variable</i>	74
	<i>Deleting Variables</i>	74
	<i>Changing the Scope of Variables</i>	75
	<i>Choosing What to See</i>	75
13.3	User and System Variables	76

14	USING VARIABLES	Page
14.1	Example for this Chapter	77
	<i>Creating the Variables Needed</i>	77
14.2	Creating Row Count Transforms	78
	<i>Adding a Row Count Transform</i>	78
14.3	Creating Expression Tasks	79
14.4	Debugging	80
	<i>Setting and Removing Breakpoints</i>	80
	<i>Debugging using Breakpoints</i>	81

15	DERIVED COLUMN TRANSFORMS	Page
15.1	Overview of Derived Column Transforms	82
	<i>Our Example – Calculating Film Length Statistics</i>	82
	<i>Adding a Derived Column Transform</i>	82
15.2	Simple Column Expressions	83
15.3	Concatenating, Casting and Conditions	83
	<i>Simple Concatenation Won't Work</i>	84
	<i>Casting</i>	85
	<i>The Ternary or Conditional Operator</i>	85
	<i>The Double = Sign When Testing Conditions</i>	87
	<i>The Relational Operators Allowed</i>	87
15.4	More Examples and Syntax	88
	<i>Example of a Mathematical Function</i>	88
	<i>Dealing with Nulls</i>	89
	<i>Example of a String Function</i>	89

16	SSIS FUNCTIONS	Page
16.1	Working with Strings of Text	90
	<i>Functions to Find and Replace Text</i>	90
	<i>Getting Extracts from a String of Text</i>	90
	<i>Other Text Functions</i>	91
	<i>New Line and Other Special Characters</i>	91
16.2	Date Functions	92
	<i>The DATEPART Function</i>	92
	<i>Using DATEADD to Add Dates</i>	93
	<i>Using DATEDIFF to Take the Difference between Two Dates</i>	93
16.3	Casting Data	94

17	ERROR-HANDLING	Page
17.1	Our Example	95
17.2	Configuring Error-Handling	96
17.3	Tracking Error Rows	97

18	CONDITIONAL SPLITS	Page
18.1	Our Example	98
18.2	Creating a Conditional Split	99
	<i>Beginning a Conditional Split</i>	99
	<i>The Operators Allowed</i>	100
	<i>Typing Conditions</i>	100
	<i>Spotting Errors (Red Text)</i>	100
18.3	Directing Conditional Split Output	101

TABLE OF CONTENTS (4 of 6)

19	LOOKUP TRANSFORMS	Page
19.1	Our Example	102
19.2	Creating a Lookup Transform	103
	<i>Step 1 – Creating the Transform</i>	103
	<i>Step 2 – Choose the Lookup Table</i>	103
	<i>Step 3 – Redirect Non-Matching Rows</i>	104
	<i>Step 4 – Choosing a Cache Mode</i>	105
	<i>Step 5 – Choosing Columns to Output</i>	106
	<i>Step 6 – Choosing Columns to Capture</i>	106
19.3	Dealing with Matched Rows	107
19.4	Dealing with Unmatched Rows	108
	<i>Strategy 1 – Store Unmatched Rows for Inspection</i>	108
	<i>Strategy 2 – Add Unmatched Shop Names as Unknown</i>	109
	<i>Strategy 3 – Add Unmatched Rows to the Lookup Table</i>	110
19.5	Caching Lookup Tables	111
	<i>Benefits of using Cache Transforms</i>	111
	<i>Our Example – Looking Up Property Types</i>	111
	<i>Filling the Cache</i>	112
	<i>Using the Cache</i>	113

20	FILE SYSTEM TASKS	Page
20.1	Adding File System Tasks	114
20.2	Working with Files	115
	<i>Creating a File Connection</i>	115
20.3	Working with Folders	116
	<i>Creating a Folder Connection</i>	116
	<i>Case Study: Creating a Folder</i>	117

21	PRECEDENCE CONSTRAINTS	Page
21.1	Simple Constraints	118
21.2	Expression Constraints	119
	<i>Automatically Annotating Expression Constraints</i>	120
21.3	Combining Constraints	121

22	LOOPING OVER FILES	Page
22.1	Our Example	122
22.2	Setting up the Package	123
	<i>Emptying the SQL Server Table</i>	123
	<i>Creating a String Variable to Hold Each File</i>	123
22.3	Looping Over Files	124
	<i>Step 1 – Adding the Foreach Loop Container Task</i>	124
	<i>Step 2 – Configuring the Loop Task</i>	125
	<i>Step 3 – Capturing the File Path in a Variable</i>	126
	<i>Step 4 – Checking the Package Works</i>	126
22.4	Omitting Certain Files	127
22.5	The Data Import Task	128
	<i>The Steps to Follow</i>	128
	<i>Step 1 – Create an Excel Connection</i>	129
	<i>Step 2 – Make this Excel Connection Dynamic</i>	130
	<i>Step 3 – Initialise the File Path Variable Value</i>	131
	<i>Step 4 – Configure and Finish the Data Flow Task</i>	131
22.6	Running the Package	132

TABLE OF CONTENTS (5 of 6)

23	LOOPING OVER ADO ROWS	Page
23.1	How ADO Enumerators Work	133
23.2	Generating Multiple Files from a Single Table	134
	<i>The Details of this Example</i>	134
	<i>Summary of the Steps to Follow</i>	134
	<i>Step 1 – Creating the Variables</i>	135
	<i>Step 2 – Create an Execute SQL Task to get Unique Shop Names</i>	135
	<i>Step 3 – Store the Results Set in the Object Variable</i>	136
	<i>Step 4 – Create the ADO Enumerator Loop</i>	136
	<i>Step 5 – Configure the ADO Enumerator Loop</i>	137
	<i>Step 6 – Add a Fixed Data Flow Task</i>	138
	<i>Step 7 – Make the Flat File Connection Dynamic</i>	139
	<i>Step 8 – Run your Package</i>	139
23.3	Importing a Workbook's Worksheets	140
	<i>The Underlying Principle</i>	140
	<i>Creating the Variable</i>	141
	<i>Create the Outline Package</i>	141
	<i>The Foreach Loop Type – Creating a Connection</i>	142
	<i>Choosing to Return the Workbook's Worksheet Names</i>	143
	<i>Creating the Data Flow Task (Source)</i>	143
	<i>Adding a Data Flow Destination</i>	144
	<i>Running the Package</i>	144

24	SQL PARAMETERS	Page
24.1	Passing Parameters by Position	145
	<i>Our Example</i>	145
	<i>Creating the Variables</i>	145
	<i>Creating the Loop over Files</i>	146
	<i>Creating the Execute SQL Task using Parameters</i>	147
	<i>The Parameter Mapping Choices</i>	148
	<i>Running the Package</i>	148
24.2	Passing Parameters by Name	149
	<i>Creating the Stored Procedure</i>	149
	<i>Changing the Execute SQL Task</i>	149

25	MERGE JOINS	Page
25.1	What Merge Joins Do	150
25.2	How Merge Joins Work	151
25.3	Creating the Package	152
	<i>Step 1 – Loading the Data</i>	152
	<i>Step 2 – Sorting the Data</i>	152
	<i>Step 3 – Merging the Data</i>	153
	<i>Step 4 – Picking out the Differences</i>	154
	<i>Step 5 – Finishing the Package</i>	154

26	DEPLOYMENT	Page
26.1	Overview of Deployment	155
26.2	Preparing to Deploy	156
	<i>Creating a Catalog</i>	156
	<i>Creating Folders</i>	156
26.3	Deploying Projects and Packages	157
	<i>Starting Project Deployment</i>	157
	<i>Finishing Deployment</i>	158
	<i>Deploying Individual Packages</i>	158

27	EXECUTING DEPLOYED PACKAGES	Page
27.1	Executing Reports from the SSMS Menu	159
27.2	Viewing Reports	160
	<i>Viewing Package Reports</i>	160
	<i>Viewing Catalog Reports</i>	160
27.3	Executing Packages from SQL	161
27.4	Scheduling Packages	162
	<i>Step 1 - Run SQL Server Agent</i>	162
	<i>Step 2 – Add a Job</i>	163
	<i>Step 3 – Adding Steps to the Job</i>	163
	<i>Step 4 – Choosing a Schedule</i>	164
	<i>Step 5 – Test your Job</i>	164

TABLE OF CONTENTS (6 of 6)

28	PARAMETERS	Page
28.1	Overview of Parameters	165
	<i>Our Example</i>	165
	<i>Variables and Parameters Needed</i>	165
28.2	Creating Parameters	166
	<i>Creating Project-Level Parameters</i>	166
	<i>Creating Package-Level Parameters</i>	166
28.3	Setting up the Example	167
	<i>The Basic Control Flow Tasks</i>	167
	<i>Starting to Configure the Loop over Files</i>	167
	<i>Making the Loop Folder Dynamic</i>	168
	<i>The Data Flow Task Excel Source</i>	168
	<i>Finishing the Data Flow Task</i>	169
	<i>Running the Package</i>	169
28.4	Deploying Packages using Parameters	170
	<i>Deploying Package-Level Parameters</i>	170
	<i>Deploying Project-Level Parameters</i>	170
28.5	Running Deployed Packages with Parameters	171

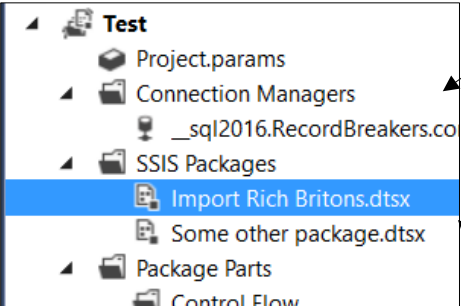
CHAPTER 1 - INTRODUCING SSIS

1.1 Overview of SSIS

SQL Server Integration Services (SSIS) is a program which allows you to build packages to Extract, Transform and Load data (it's often called an ETL application for this reason).

SSIS Packages and Solution Explorer

A package in SSIS is a file with the extension **.dtsx**:



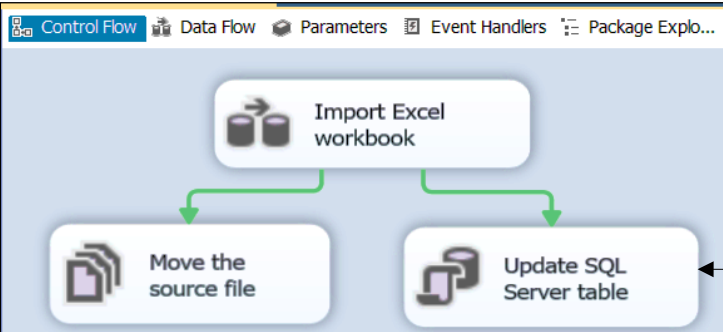
This window is called Solution Explorer (more on this shortly) – it allows you to navigate between the packages that you've created in SSIS.

This is what an SSIS package looks like in Solution Explorer before you open it. The **DTS** stands for **Data Transformation Services**, which is what SSIS used to be called about ten years ago!

A package consists of two main parts: *control flow* and *data flow*, as shown under separate headings below and overleaf.

Control Flow

The *control flow* part of a package consists of a series of instructions you ask SSIS to execute:



In this example, these are two file system tasks – SSIS will choose which one to run depending on the outcome of the previous task.



When you tell your child to tidy their room, do their homework and then come and help make dinner, this is a series of control flow tasks (as well as being a tad optimistic).

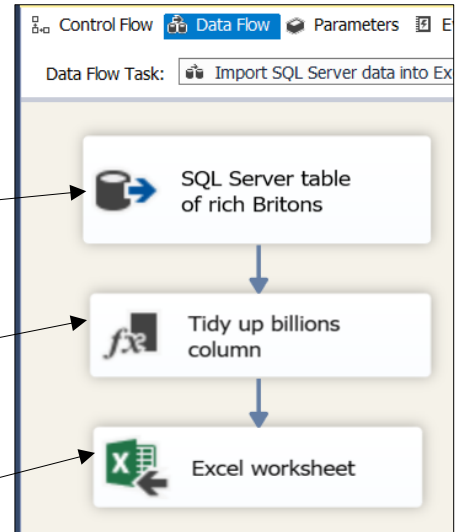
Data Flow

You can open any single data flow task to reveal instructions on how to load, transform and store data:

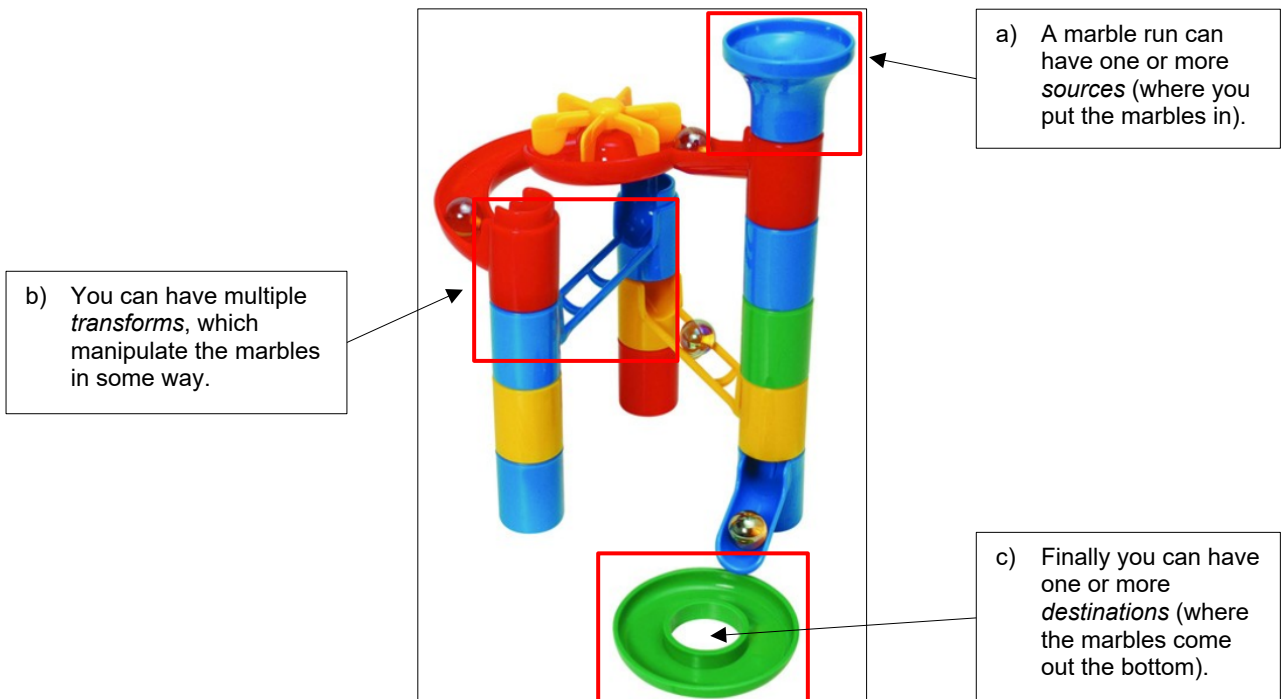
This is a *source* (where the data comes from – in this case a SQL Server table).

This is a *transform* (in this case, it's tidying up a numerical column).

This is a *destination* (where the data ends up – in this case in an Excel workbook).



A good analogy for data flow is a marble run:



At a rough guess, you typically will spend about 70% of your time in Integration Services inside data flow tasks, and 30% in control flow.

CHAPTER 2 - USING VISUAL STUDIO

2.1 Starting Visual Studio

Visual Studio and SQL Server Data Tools

Visual Studio is Microsoft's flagship development application – you can use it to create websites, mobile phone apps, SSIS packages and Windows applications, among other things:

The part of Visual Studio which you use to create BI tools (in SSIS, Reporting Services and Analysis Services) also goes under the name *SQL Server Data Tools*, or *SSDT*.

*Business Intelligence tools
(including SSIS)*

*ASP.NET
(websites)*

*Windows applications
(Visual Basic and C#)*

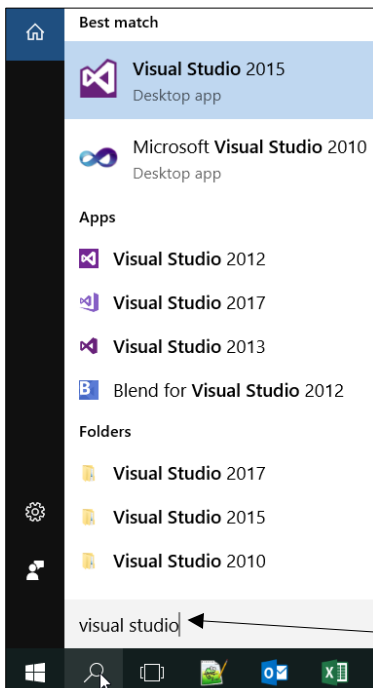
*Mobile phone apps
(Apple and Android)*



What the above shows is that it doesn't matter whether you run Visual Studio or SQL Server Data Tools – the second is just a subset of the first.

Running Visual Studio

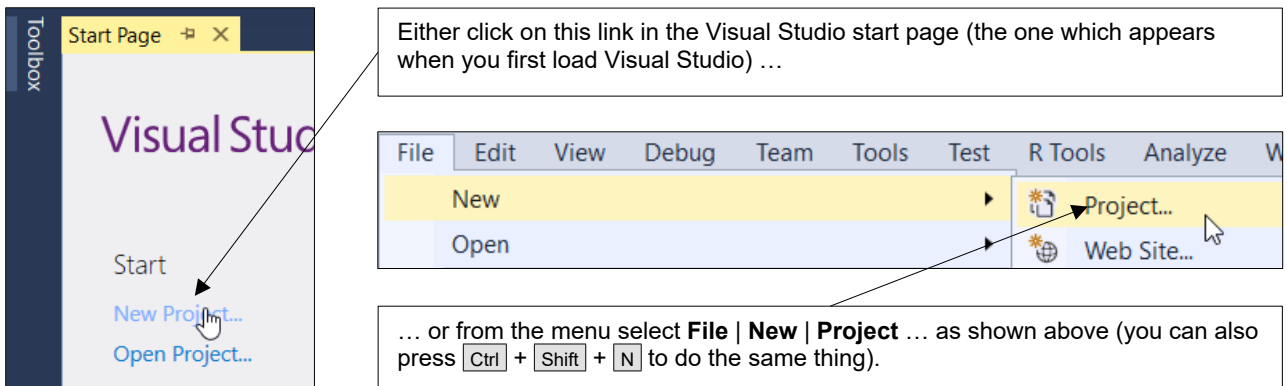
You can start Visual Studio in many ways – one method for Windows 10 is shown here:



Click on the magnifying glass icon, then type in the program you want to run. This Wise Owl has got four different versions of Visual Studio on his computer! The one used in writing this manual was Visual Studio 2015.

2.2 Creating a Project

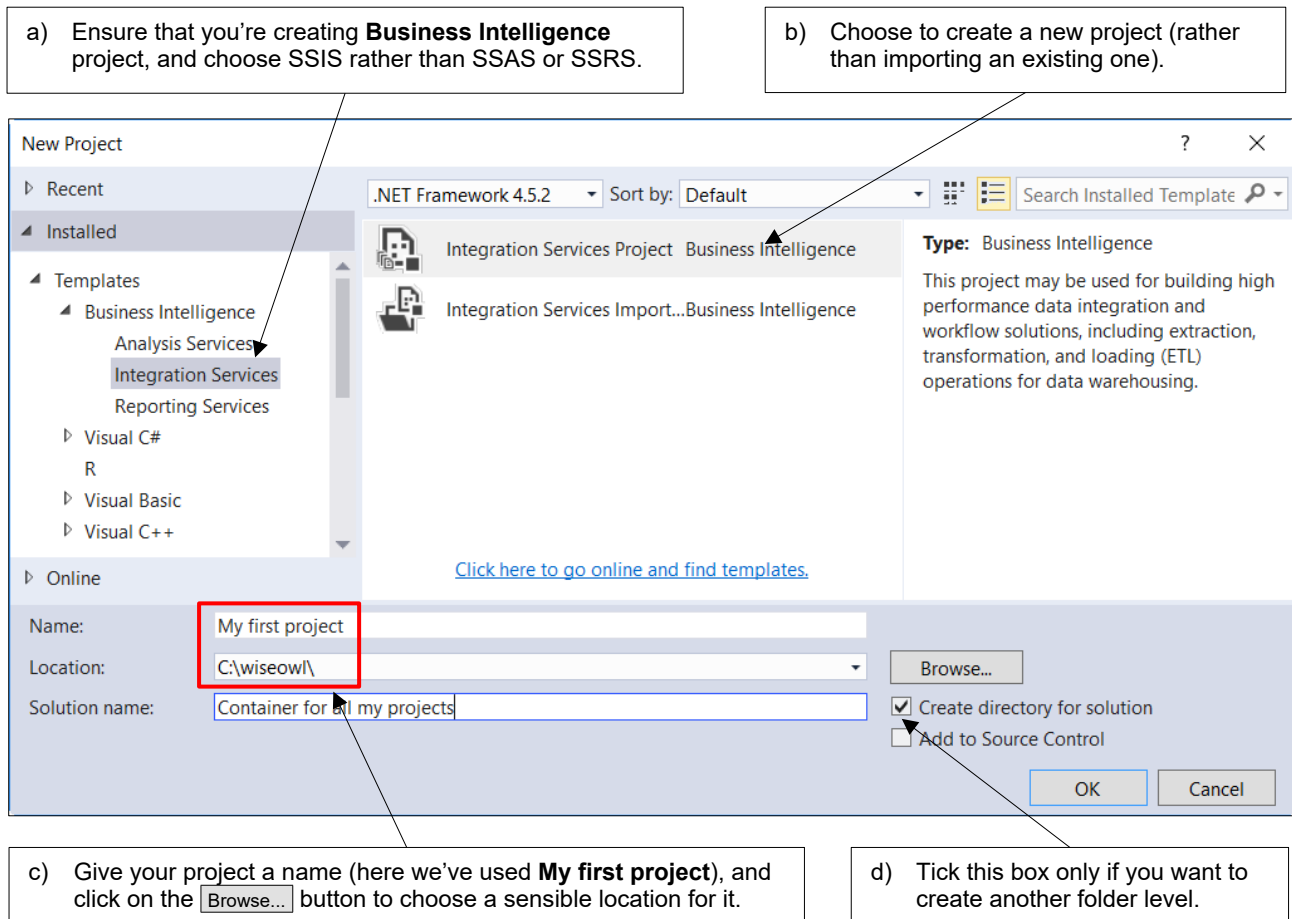
You store SSIS packages in a container called a *project*. Here's how to create one:



Either click on this link in the Visual Studio start page (the one which appears when you first load Visual Studio) ...

... or from the menu select **File | New | Project ...** as shown above (you can also press **Ctrl + Shift + N** to do the same thing).

You can now give your project a name, and choose where to put it:



a) Ensure that you're creating **Business Intelligence** project, and choose SSIS rather than SSAS or SSRS.

b) Choose to create a new project (rather than importing an existing one).

c) Give your project a name (here we've used **My first project**), and click on the **Browse...** button to choose a sensible location for it.

d) Tick this box only if you want to create another folder level.

If you choose to create a directory for your solution as above, you'll end up with a long path!



This PC > OS (C:) > wiseowl >
 Container for all my projects >
 My first project

Base location
Folder for solution
Project folder

2.3 Visual Studio Windows

The Solution Explorer, Properties and SSIS Toolbox Windows


When using SSIS, there are three main windows that you will use:

Use *Solution Explorer* to add, view and edit packages and other files.

Use the *SSIS Toolbox* to add control or data flow tasks.

Use the *Properties* window to set selected items' behaviour.

Here's how to display these three windows:

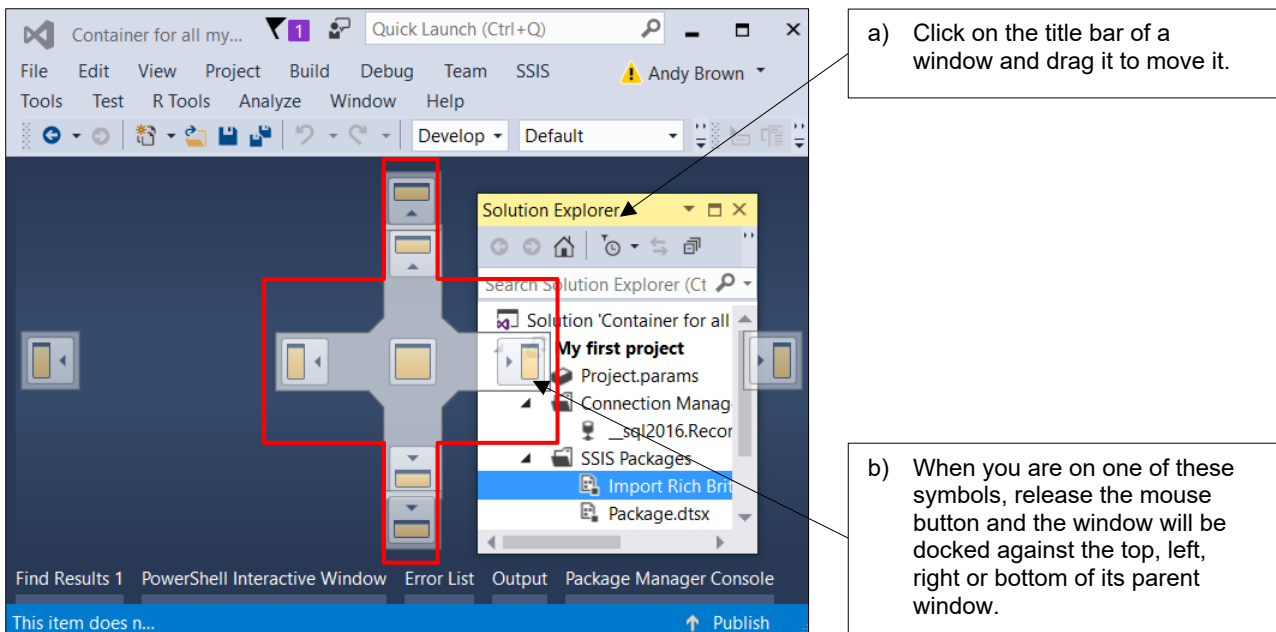
Window	Top menu	Keyboard	Other method
<i>Solution Explorer</i>	View → Solution Explorer	Ctrl + Alt + L	
<i>SSIS Toolbox</i>	SSIS → SSIS Toolbox		
<i>Properties</i>	View → Properties Windows	F4	



To get the SSIS toolbox menu to appear you often have to open a package and click on it first. Don't confuse the (invaluable) SSIS toolbox with the (useless, in this context) standard Visual Studio toolbox.

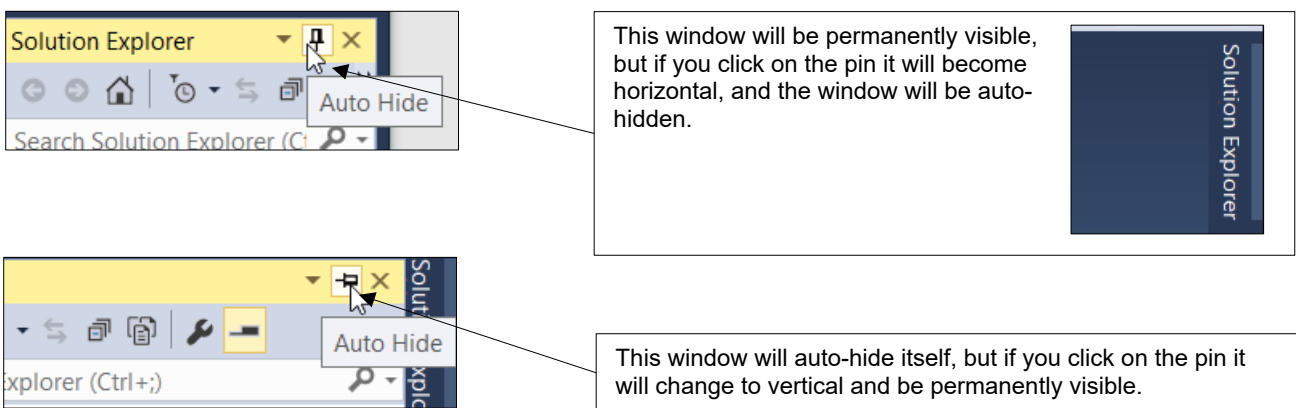
Floating Windows

You can click and drag on the title bar of any window to reposition it:



































Auto-hiding Windows

You can click on the pin at the top right of any docked window to make it shrink when you're not using it:



What we do!

		Basic training	Advanced training	Systems / consultancy
Office	Microsoft Excel			
	VBA macros			
	Office Scripts			
	Microsoft Access			
Power BI, etc	Power BI and DAX			
	Power Apps			
	Power Automate (both)			
SQL Server	SQL			
	Reporting Services			
	Report Builder			
	Integration Services			
	Analysis Services			
Coding	Visual C#			
	VB programming			
	MySQL			
	Python	