



FEED YOUR BRAIN[®]

C# 3.0 y LINQ

**Aprende a sacar todo el partido
a la última versión de .NET**

Octavio Hernández Leal

Prólogo de José Manuel Alarcón, Microsoft MVP.



Contenido

PRÓLOGO DEL AUTOR	xv
AGRADECIMIENTOS	xvii
INTRODUCCIÓN	xix

Parte I **CONCEPTOS PRELIMINARES**

MÉTODOS ANÓNIMOS	3
1.1. Breve repaso a los delegados y eventos en C# 1	3
1.2. Los métodos anónimos de C# 2.0.....	7
1.3. Paso de delegados anónimos como parámetros.....	8
1.4. Acceso a las variables de ámbitos externos	9
GENÉRICOS	13
2.1. Cada oveja con su pareja	13
2.2. La expresión default(T).....	21
2.3. La implementación de la genericidad en .NET.....	21
2.4. Colecciones genéricas.....	22
2.5. Más de un parámetro de tipo.....	24
2.6. Restricciones.....	26
2.7. Métodos genéricos.....	27
2.8. Delegados genéricos.....	29
2.9. Interfaces genéricas	32
2.10. La interfaz IEnumerable<T>	32
2.11. La semántica de foreach para IEnumerable<T>.....	34
2.12. Ejemplo de implementación de IEnumerable<T>	35
2.13. La interfaz ICollection<T>.....	37
ITERADORES	39
3.1. Bloques de iteración	39
3.2. La secuencia del 1 al 1000, <i>revisited</i>	40

3.3.	Detalles internos	40
3.4.	La generación bajo demanda durante la iteración	41
3.5.	<i>Don't go breaking my iteration</i>	42
3.6.	Mejorando la iteración	43
3.7.	Primos y enteros grandes.....	44
3.8.	Un ejemplo más práctico	46
TIPOS VALOR ANULABLES		49
4.1.	Fundamentos.....	49
4.2.	Implementación de los tipos valor anulables.....	51
4.3.	Un detalle a tener en cuenta.....	53

Parte 2

NOVEDADES EN C# 3.0

NOVEDADES “BÁSICAS” EN C# 3.0.....		57
5.1.	Declaración implícita del tipo de variables locales	57
5.1.1.	Sobre la conveniencia de utilizar var	59
5.2.	Propiedades implementadas automáticamente.....	59
5.3.	Inicializadores de objetos y colecciones	60
5.3.1.	Inicializadores de objetos.....	60
5.3.2.	Inicializadores de colecciones	62
5.4.	Tipos anónimos inmutables.....	63
5.5.	Arrays de tipo definido implícitamente.....	65
5.6.	Métodos parciales	65
5.6.1.	Utilidad de los métodos parciales.....	68
MÉTODOS EXTENSORES.....		69
6.1.	Introducción.....	69
6.2.	La sintaxis.....	71
6.3.	Más ejemplos.....	72
6.4.	El acceso a los métodos extensores.....	75
6.5.	Recomendaciones de utilización	76
6.6.	La razón de ser de los métodos extensores.....	77
EXPRESIONES LAMBDA.....		79
7.1.	Introducción.....	79
7.2.	Las expresiones lambda como objetos de código.....	80
7.3.	La sintaxis.....	81
7.4.	El tipo delegado Func	82
7.5.	Más ejemplos.....	82

7.6.	Uno más complejo.....	83
7.7.	Parámetros por referencia.....	84
7.8.	Recursividad en expresiones lambda.....	85
7.9.	Las expresiones lambda como objetos de datos.....	86
ÁRBOLES DE EXPRESIONES.....		87
8.1.	De expresiones lambda a árboles de expresiones.....	87
8.1.1.	Los árboles como representación de expresiones.....	88
8.2.	La jerarquía de clases de expresiones.....	91
8.3.	Más ejemplos.....	96
8.4.	Manipulación programática de árboles.....	98
8.5.	Cálculo de derivadas.....	107
8.6.	Y ahora, los deberes.....	112

Parte 3

CONSULTAS INTEGRADAS EN C#

FUNDAMENTOS DE LINQ.....		115
9.1.	Presentación de LINQ.....	115
9.2.	Las expresiones de consulta.....	117
9.3.	Reescritura de las expresiones de consulta.....	119
9.4.	La (no) semántica de los operadores de consulta.....	121
9.5.	Resolución de llamadas a operadores.....	121
9.6.	Los operadores de consulta estándar.....	123
9.7.	El patrón de expresiones de consulta.....	124
9.8.	La utilidad ObjectDumper.....	125
9.9.	Ejemplos básicos.....	126
9.10.	De nuevo la ejecución diferida.....	128
9.11.	Sintaxis de las expresiones de consulta.....	130
9.12.	Productos cartesianos.....	131
9.12.1.	Restricción de productos y optimización de consultas.....	133
9.13.	Encuentros.....	133
9.13.1.	Particularidades sintácticas.....	134
9.13.2.	Diferencia con el producto cartesiano restringido.....	135
9.14.	Grupos.....	135
9.15.	La cláusula into.....	137
9.15.1.	Continuaciones.....	137
9.15.2.	Encuentros agrupados.....	139
9.15.3.	Emulando encuentros externos con encuentros agrupados.....	140
9.16.	La cláusula let.....	142
9.17.	Algunos ejemplos prácticos.....	143
9.18.	¿Un nuevo modelo de escritura de bucles?.....	145

OPERADORES DE CONSULTA ESTÁNDAR	149
10.1. Tabla de operadores de consulta estándar.....	149
10.2. Operadores básicos.....	153
10.2.1. El operador Where()	153
10.2.2. El operador Select()	154
10.2.3. Caso trivial de Select()	155
10.2.4. El operador SelectMany()	156
10.2.5. Operadores de ordenación	158
10.2.6. El operador GroupBy().....	160
10.2.7. El operador Join().....	162
10.2.8. El operador GroupJoin()	164
10.3. Operadores de partición.....	166
10.3.1. El operador Take()	167
10.3.2. El operador Skip().....	167
10.3.3. El operador TakeWhile().....	168
10.3.4. El operador SkipWhile().....	168
10.4. Operadores conjuntuales.....	169
10.4.1. El operador Distinct().....	169
10.4.2. El operador Union()	170
10.4.3. El operador Intersect().....	171
10.4.4. El operador Except().....	171
10.5. Operadores de conversión	172
10.5.1. El operador ToArray().....	172
10.5.2. El operador ToList()	172
10.5.3. El operador ToDictionary().....	173
10.5.4. El operador ToLookup().....	174
10.5.5. El operador AsEnumerable().....	175
10.5.6. El operador Cast<T>().....	176
10.5.7. El operador OfType<T>()	177
10.6. Operadores de generación de secuencias.....	177
10.6.1. El operador Range()	178
10.6.2. El operador Repeat<T>().....	178
10.6.3. El operador Empty<T>().....	178
10.7. Otros operadores de transformación de secuencias	179
10.7.1. El operador Concat().....	179
10.7.2. El operador Reverse().....	179
10.8. Cuantificadores	180
10.8.1. El operador Any()	180
10.8.2. El operador All().....	180
10.8.3. El operador Contains().....	181
10.8.4. El operador SequenceEqual().....	181
10.9. Operadores de elementos	182
10.9.1. El operador First()	182
10.9.2. El operador FirstOrDefault().....	182
10.9.3. El operador Last().....	183

10.9.4. El operador LastOrDefault()	183
10.9.5. El operador Single()	184
10.9.6. El operador SingleOrDefault()	184
10.9.7. El operador ElementAt()	185
10.9.8. El operador ElementAtOrDefault()	185
10.9.9. El operador DefaultIfEmpty()	186
10.10. Agregados	186
10.10.1. Los operadores Count() y LongCount()	186
10.10.2. Los operadores Max() y Min()	187
10.10.3. El operador Sum()	188
10.10.4. El operador Average()	189
10.10.5. El operador Aggregate()	190
EL PATRÓN LINQ	193
11.1. Acercando LINQ a nuevos tipos de datos	193
11.1.1. LINQ to Pipes	195
11.2. El patrón de expresiones de consulta	198
11.3. Una implementación alternativa a LINQ to Objects	207
11.4. La interfaz IQueryable<T>	212
11.4.1. Definiciones	213
11.4.2. Ejemplo básico	214
11.4.3. Implementación de IQueryable<T>	217
11.5. Qué hace el proveedor de consultas	218
11.5.1. El método CreateQuery()	219
11.5.2. El método Execute()	221
11.6. Un ejemplo real: LINQ to TFS	222
11.6.1. Presentación de las API de TFS	222
11.6.2. Ejemplo básico de consulta	224
11.6.3. Un proveedor básico	224
11.6.4. La puerta de entrada a LINQ	225
11.6.5. La implementación de IQueryable<WorkItem>	226
11.6.6. El proveedor de consultas	228
11.6.7. El mecanismo de enumeración	229

Parte 4 EXTENSIONES DE LINQ

LINQ TO XML	235
12.1. Presentación	235
12.2. Expresiones de consulta sobre documentos XML	241
12.3. Operadores de consulta específicos de LINQ to XML	244
12.4. Búsquedas en documentos XML	245
12.4.1. Búsquedas XPath	246

12.5. Inserción, modificación y borrado de nodos.....	246
12.6. Transformación de documentos	247
12.6.1. Transformaciones XSLT	249
12.7. Conclusión.....	249
LINQ TO DATASET.....	251
13.1. Presentación.....	251
13.2. Consultas contra DataSet tipados.....	253
13.3. Consultas contra DataSet no tipados	254
13.4. De vuelta a un DataTable.....	256
13.5. Actualizaciones.....	256
13.6. El método AsDataView().....	257
13.7. Conclusión.....	258
LINQ TO SQL.....	259
14.1. Presentación.....	259
14.2. Contextos de datos y clases de entidad.....	261
14.2.1. El código generado por el diseñador.....	263
14.3. Ejecución de consultas integradas.....	266
14.4. El mapeado objeto/relacional.....	274
14.4.1. Gestión de la identidad	274
14.5. Propiedades de navegación.....	275
14.5.1. Gestión de la carga de las propiedades de navegación.....	276
14.6. Consultas dinámicas	276
14.6.1. El método ExecuteQuery<T>().....	277
14.6.2. La clase IQueryable.....	278
14.7. Actualización de datos.....	279
14.7.1. Utilización de transacciones.....	281
14.7.2. Gestión de la concurrencia	282
14.7.3. Personalización de las actualizaciones.....	284
14.8. Ejecución de procedimientos y funciones	285
14.9. Combinando tecnologías.....	286
14.10. La clase LinqDataSource.....	286
14.11. Conclusión.....	288
ÍNDICE ANALÍTICO.....	289