

DENIECI MORENO AZOFEIFA

FUNDAMENTOS DE PROGRAMACIÓN WEB

Guía de estudio



UNED

UNIVERSIDAD ESTATAL A DISTANCIA

Institución Benemérita de la Educación y la Cultura



Universidad Estatal a Distancia

Vicerrectoría Académica

Escuela de Ciencias Naturales y Exactas

Esta guía de estudio ha sido confeccionada en la Uned, en el año 2011, para ser utilizada en la asignatura “Fundamentos de programación web”, código 3075, que se imparte en el programa de Ingeniería Informática.



Créditos

Edición académica:

Mario Marín Romero

Revisión filológica:

María Benavidez González

Encargado de cátedra:

Enrique Gómez Jiménez





PRESENTACIÓN

La expansión de la Internet y su uso generalizado en las organizaciones modernas han creado la necesidad de innovar cada día en el diseño y desarrollo de sitios web. Diseñadores gráficos, psicólogos, sociólogos, mercadólogos, entre otros profesionales, han tenido que intervenir muchas veces en la creación de soluciones para atender un nicho específico de usuarios, proveyéndoles de información, productos y servicios. Por tanto, el ingeniero informático es una pieza fundamental en el engranaje de todas las estructuras que conforman una aplicación web y, por ende, cada día se incrementa la necesidad de emplear mejores técnicas de diseño y desarrollo web.

Este curso pretende guiarle en el aprendizaje de las técnicas, métodos y estrategias existentes en el diseño y desarrollo de sitios web y la aplicación de soluciones optimizadas para su uso en el ambiente comercial, científico, académico o de cualquier otra área que requiera del procesamiento de información digital.

Para lograr los objetivos propuestos, se cuenta con el libro de texto del curso titulado *Aplicaciones con Visual Basic .NET. ¡Programe para Escritorio, Web y Dispositivos móviles!* del autor Enrique Gómez Jiménez. La presente guía le apoyará en el estudio de tal texto.



CONTENIDOS

Presentación.....	iii
Objetivos	vi
Estructura.....	viii
Tema 1. Introducción a Visual Basic .NET y su entorno de desarrollo	1
<i>Guía de lectura</i>	2
<i>Comentarios del tema</i>	3
Capítulo 1. Introducción a Visual Basic .NET	3
Capítulo 2. Entorno de desarrollo	5
<i>Ejercicios de autoevaluación</i>	6
Tema 2. Tipos de datos y modelos de programación.....	7
<i>Guía de lectura</i>	8
<i>Comentarios del tema</i>	9
Capítulo 3. Datos y programación en Visual Basic .NET	9
<i>Ejercicios de autoevaluación</i>	10

Tema 3. Programación orientada a objetos	11
<i>Guía de lectura</i>	12
<i>Comentarios del tema</i>	13
Capítulo 4. Programación orientada a objetos en Visual Basic .NET.....	13
<i>Ejercicios de autoevaluación</i>	14
Tema 4. Gestión de datos	15
<i>Guía de lectura</i>	17
<i>Comentarios del tema</i>	18
Capítulo 6. ADO .NET	18
Capítulo 9. Seguridad de aplicaciones ADO .NET	22
<i>Ejercicios de autoevaluación</i>	23
Tema 5. Tecnologías para el desarrollo web	25
<i>Guía de lectura</i>	27
<i>Comentarios del tema</i>	29
Capítulo 10. ASP .NET	29
Capítulo 11. Fundamentos de servicios web (<i>web services</i>) en ASP .NET	33
Capítulo 13. Implementando servicios web con ADO .NET	35
Capítulo 14. Implementación de servicios web en aplicaciones ASP .NET.....	36
Capítulo 15. Uso de datos XML en aplicaciones web ASP .NET.....	38
<i>Ejercicios de autoevaluación</i>	40
Respuestas a los ejercicios de autoevaluación.....	41



OBJETIVOS

Objetivo general

- ✓ Brindar al estudiante los conocimientos básicos en el desarrollo de aplicaciones web, utilizando herramientas integradas de diseño y desarrollo.

Objetivos específicos

- Explicar las funcionalidades básicas del entorno y de los componentes de una herramienta de diseño y desarrollo web.
- Comprender el modelo de programación y los tipos de datos que utiliza la herramienta de diseño y desarrollo de sitios web.
- Crear un sitio web mediante el uso de la programación orientada a objetos.
- Analizar las diferentes estrategias de gestión dinámica de datos de una herramienta de diseño y desarrollo de sitios web.
- Crear aplicaciones web mediante una herramienta de diseño y desarrollo que permita interfaces y múltiples funcionalidades.
- Analizar estrategias de seguridad de acceso al código y a los datos para el desarrollo de aplicaciones web.
- Analizar las funcionalidades y la arquitectura de una herramienta para desarrollar sitios web.
- Desarrollar aplicaciones web que utilicen la capa de lógica de negocios mediante el uso de servicios web.

- Crear modelos de base de datos que permitan la gestión dinámica de información en un sitio web.
- Desarrollar servicios web que implementen la gestión dinámica de datos en un sitio web.
- Aplicar el uso de servicios web en aplicaciones que gestionen datos.
- Crear aplicaciones web que utilicen archivos XML para la gestión de datos.



ESTRUCTURA

Esta guía se ha estructurado de tal manera que se facilite el estudio del libro de texto. Para cada tema se incluyen diversas secciones que debe tomar en consideración:

Guía de lectura

Le indica las páginas que usted debe leer y estudiar del libro de texto para cada tema y subtema.

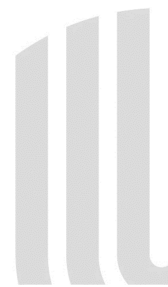
Comentarios generales

Brindan aspectos importantes del tema en cuestión, así como su ubicación dentro de cada capítulo del libro de texto; estos le servirán para sintetizar los conceptos presentados. De esta manera, podrá determinar si requiere repasar o aclarar alguno antes de desarrollar los ejercicios.

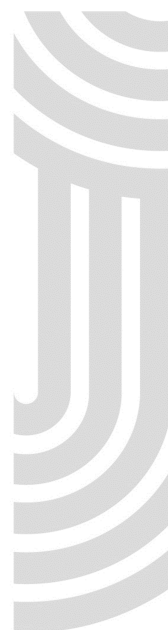
Preguntas de autoevaluación

Se presentan con el propósito de que usted autoevalúe su nivel de comprensión y aprendizaje del tema en estudio. Estos ejercicios consideran algunos de los conceptos tratados en cada capítulo del libro de texto. Se le invita además a formular sus propias preguntas y soluciones, resúmenes y otros elementos que le permitan autoevaluarse. Al final de la guía se presentan las respuestas a los ejercicios de autoevaluación.

INTRODUCCIÓN A VISUAL BASIC .NET
Y SU ENTORNO DE DESARROLLO



1





INTRODUCCIÓN

Este tema expone las características principales de Visual Studio .NET, versión 2008, considerando Visual Basic .NET como herramienta de desarrollo; así como el entorno integrado de desarrollo (IDE, por sus siglas en inglés), que permite el diseño de las interfaces. Se realiza un recorrido conceptual de las principales funcionalidades de la herramienta, de los datos y de la sintaxis por utilizar en el desarrollo de una aplicación informática, sea para Windows, Web o dispositivos móviles.

Al finalizar el estudio de este tema, usted estará en capacidad de comprender las funcionalidades básicas del entorno y de los componentes de una herramienta de diseño y desarrollo web.

GUÍA DE LECTURA

Para el estudio de este tema, lea detalladamente las páginas del libro que se indican a continuación:

Capítulo	Sección	Página(s)
1 Introducción a Visual Basic .NET	1.1. ¿Qué es Visual Studio .NET 2008?	1-2
	1.2. .NET Framework	2
	1.3. CLR	3-4
	1.4. Tecnologías Framework	4-13
	1.5. LINQ	13-14
	1.6. ADO .NET	13-16
	1.7. ASP .NET	16-17



Capítulo	Sección	Página(s)
2 Entorno de desarrollo	2.1. Integración del entorno	19-20
	2.2. Tipos de proyectos en Visual Basic .NET	20-22
	2.3. Menú principal de Visual Basic .NET	22-44

COMENTARIOS DEL TEMA

Capítulo 1. Introducción a Visual Basic .NET

1.1. ¿Qué es Visual Studio .NET 2008?

Visual Studio .NET es un conjunto de herramientas de *software* que permiten al programador desarrollar aplicaciones informáticas orientadas a Windows, Web o dispositivos móviles. Dentro de las características que incluye esta suite, podemos encontrar herramientas, estrategias de desarrollo, plantillas reutilizables, tecnologías de conectividad, entre otras funcionalidades. Como parte de los lenguajes que se incluyen en Visual Studio .NET se encuentran Visual Basic .NET, C# y Visual C++; mientras que existen estrategias para la gestión de datos (ADO .NET), la creación de sitios web (ASP .NET, AJAX, entre otras), el diseño de interfaces (WPF), de comunicaciones (WCF), entre otros.

1.2. .NET Framework

Es un componente que acompaña a Visual Studio .NET y sirve de marco de trabajo para el desarrollo de aplicaciones Windows, Web y para dispositivos móviles. Es un conjunto de componentes y librerías que hacen posible el ensamblaje de elementos o la programación de aplicaciones. Entre los componentes del .NET Framework se encuentran el CLR (Common Language Runtime), el BCL (Base Class Library), entre otros.



1.3. CLR

El CLR (Common Language Runtime) es un entorno administrado que permite la ejecución de aplicaciones creadas con Visual Studio .NET o aquellas que soporten el Framework de Visual Studio .NET. Actúa como un pequeño sistema operativo o motor encargado de administrar los recursos de una aplicación .NET, proveyéndola de espacio de ejecución, manejo de excepciones, servicios, entre otros.

1.4. Tecnologías Framework

El Framework es un marco de trabajo que permite la creación, administración y ejecución de aplicaciones basados en .NET. El Framework de Microsoft, para Visual Studio .NET, incorpora funcionalidades necesarias para implementar aplicaciones ricas en interfaces como WPF (Windows Presentation Foundation), sistemas conectados mediante WCF (Windows Communication Presentation), ADO .NET, Windows Forms, entre otras.

Vea las tecnologías Framework en la página 4 del libro de texto y sus detalles en las páginas de la 7 a la 13.
--

1.5. LINQ

LINQ constituye un modelo de programación que se utiliza para consultar y actualizar datos mediante Visual Basic .NET o C#. Las fuentes de datos pueden ser de SQL Server, XML u otras que soporten el modelo. Existen implementaciones para utilizar sobre conjuntos de datos desconectados (Dataset), con XML o con objetos (Objects).

1.6. ADO .NET

ADO .NET constituye la estrategia de Microsoft® para la gestión de datos, tanto para fuentes nativas de la compañía como para aquellas tecnologías de manejo de datos que soportan el Framework de Visual Studio .NET, tal como Oracle. También incluye el manejo de datos mediante ADO .NET Entity o ADO .NET Data Services.



1.7. ASP .NET

ASP .NET es la tecnología que permite el diseño y desarrollo de aplicaciones orientadas a la Web. Provee las funcionalidades necesarias para poner en marcha un sitio web estático (que muestre información) o dinámico (que facilite el manejo de datos). Existen tecnologías alternativas que permiten crear productos de *software* más optimizados, como por ejemplo AJAX, que implementa manejo de lógica funcional del lado del cliente.

En resumen, se puede mostrar gráficamente el paradigma .NET en la siguiente tabla:

Tecnología	Descripción funcional
ADO .NET	Tecnología para el acceso a datos de diferentes fuentes.
ASP .NET	Tecnología para el diseño y desarrollo de páginas web.
LINQ	Tecnología que permite la gestión de consultas utilizando un lenguaje creado por Microsoft.
CLR	Motor que permite la ejecución de aplicaciones y la administración eficiente de los recursos.
NET Framework	Corazón de Visual Studio .NET. En este radican todas las funcionalidades de la herramienta.

Capítulo 2. Entorno de desarrollo

2.1. Integración del entorno

Visual Studio .NET posee un entorno de desarrollo integrado, lo cual permite a los desarrolladores crear soluciones informáticas utilizando cualquiera de las herramientas que provee. Ya sea que el programador desee desarrollar con Visual Basic .NET o con C#, siempre dispondrá de un IDE similar para cualquiera de los dos lenguajes.



2.2. Tipos de proyectos en Visual Basic .NET

Visual Studio .NET permite desarrollar varios tipos de proyectos diferentes. Por ejemplo, se pueden crear para Windows, Web o dispositivos móviles; también es factible crear servicios Windows o Web que permiten agrupar la lógica funcional de una aplicación informática.

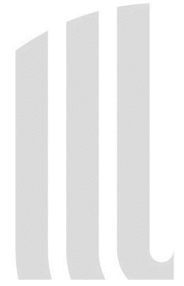
Vea los tipos de proyectos que puede crear con Visual Studio .NET en las páginas 21 y 22 del el libro de texto.

2.3. Menú principal de Visual Basic .NET

El menú principal de Visual Basic .NET constituye el IDE integrado de Visual Studio .NET y agrupa las funcionalidades necesarias para la creación de soluciones informáticas orientadas a Windows, Web o dispositivos móviles. Las opciones de menú también se encuentran implementadas en la barra de iconos de ejecución rápida, lo que facilita enormemente las tareas de edición y uso de herramientas de depuración, entre otros.

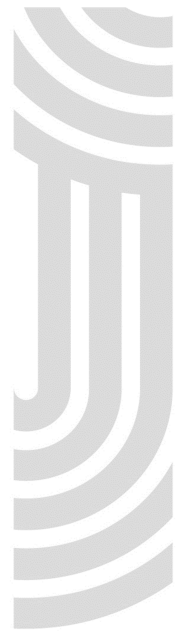
EJERCICIOS DE AUTOEVALUACIÓN DEL TEMA 1

- 1) Mencione las características que incluye Visual Studio .NET 2008
- 2) Describa la funcionalidad de ASP .NET.
- 3) Describa la funcionalidad de ADO .NET.
- 4) Explique la funcionalidad de LINQ.
- 5) Explique el funcionamiento del Framework de Visual Studio .NET 2008.
- 6) Describa la funcionalidad de CLR.
- 7) Explique brevemente la principal ventaja del entorno de Visual Studio .NET.
- 8) Cite y explique los principales proyectos que se pueden desarrollar en Visual Studio .NET.



TIPOS DE DATOS
Y MODELOS DE PROGRAMACIÓN

2





INTRODUCCIÓN

En este tema se estudian los tipos de datos que soporta Visual Studio 2008 y el modelo de desarrollo utilizado; también se explica su persistencia según el tipo de variable declarada y se aborda el tema de las funciones predefinidas en la herramienta y su utilización.

Al finalizar el estudio de este contenido, usted estará en capacidad de comprender el modelo de programación y los tipos de datos que utiliza la herramienta de diseño y desarrollo de sitios web.

GUÍA DE LECTURA

Para el estudio de este tema, lea detalladamente las páginas que se indican a continuación:

Capítulo	Sección	Página(s)
3 Datos y programación en Visual Basic .NET	3.1. Introducción a la programación en Visual Basic .NET	47-48
	3.2. Tipos de datos en Visual Basic .NET	48-50
	3.3. Ámbito y declaración de variables en Visual Basic .NET	51-52
	3.4. Control de flujos en Visual Basic .NET	52-57
	3.5. Funciones predefinidas en Visual Basic .NET	57-68



COMENTARIOS DEL TEMA

Capítulo 3. Datos y programación en Visual Basic .NET

3.1. Introducción a la programación en Visual Basic .NET

La forma de programar en Visual Studio .NET ha presentado pocos cambios en cuanto a la sintaxis utilizada en su construcción funcional. Muchos componentes reutilizables aún se mantienen, otros han desaparecido o mejorado, pero la familiaridad con el entorno no se ha perdido y mantiene vigencia en las nuevas versiones de la herramienta.

3.2. Tipos de datos en Visual Basic .NET

Visual Basic .NET permite el empleo de varios tipos de estructuras de datos, lo cual es generalizado en los lenguajes modernos de programación. Estas representan datos simples, como son los enteros, o estructuras complejas como un Arraylist. Son muchos los tipos de datos en la herramienta, y se debe tener cautela cuando desarrollamos una solución y utilizamos datos de diferente tipo, dado que eventualmente podríamos desmejorar su desempeño.

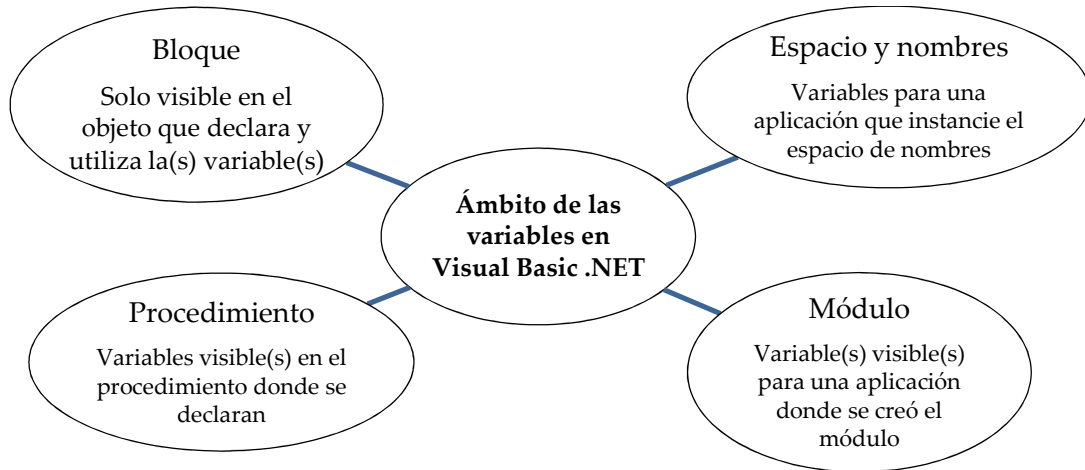
Vea los tipos de datos que se pueden utilizar con Visual Studio .NET en las páginas 49 y 50 del libro de texto.

3.3. Ámbito y declaración de variables en Visual Basic .NET

El ámbito de ejecución de una variable es una condición importante de considerar cuando se desarrolla una solución informática, sea cual sea el lenguaje de programación utilizado. Tenemos ámbitos de persistencia de estas variables a nivel local o global. Para ello, debemos considerar cuáles valores queremos mantener siempre en memoria y cuáles serán volátiles una vez terminado un proceso de ejecución.

Vea los ámbitos de ejecución de variables que se pueden utilizar con Visual Studio .NET en la página 51 del libro de texto.

A continuación, se muestra un resumen de los alcances de variables en Visual Studio .NET.



3.4. Control de flujos en Visual Basic .NET

El control de flujo de un programa informático permite ejecutar una u otra condición. También nos permite la ejecución de una misma instrucción un determinado número de veces, o mediante el cumplimiento o incumplimiento de una determinada condición. Visual Studio .NET 2008 permite, al igual que la mayoría de lenguajes de programación modernos, utilizar una amplia gama de controladores de flujo, que van desde un sencillo IF hasta un For Each que es más elaborado.

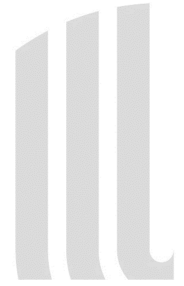
3.5. Funciones predefinidas en Visual Basic .NET

En Visual Basic .NET existe gran cantidad funciones predefinidas, las cuales pueden utilizarse para implementar una lógica que emplee cadenas, números o formatos de fecha. Dependiendo de la lógica a implementar, podemos utilizar una u otra librería de funciones.

Vea las funciones predefinidas que se pueden utilizar con Visual Studio .NET en las páginas de la 57 a la 68 del libro de texto.

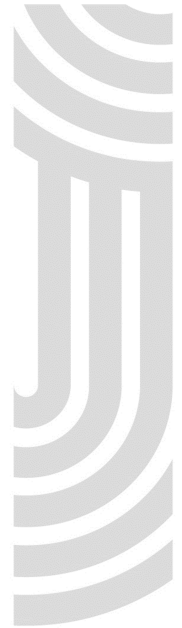
EJERCICIOS DE AUTOEVALUACIÓN DEL TEMA 2

- 1) Explique dos entornos de variables utilizados en Visual Studio .NET.
- 2) Cite al menos dos funciones de cadena, dos funciones de fecha y dos de aplicación matemática que se utilizan en Visual Studio .NET.



PROGRAMACIÓN
ORIENTADA A OBJETOS

3





INTRODUCCIÓN

Este tema trata la orientación a objetos como refuerzo al modelo programático que debe existir en una arquitectura de desarrollo dirigido a la Web. No es específico a la creación de sitios web; sin embargo, la lógica funcional es la misma, pues se trata de una conceptualización, no solo aplicada a desarrollo Windows, sino que a cualquier tipo de desarrollo que se presuma sea orientado a objetos.

Al finalizar el estudio de este tema, usted será capaz de crear un sitio Web mediante la teoría de orientación a objetos.

GUÍA DE LECTURA

Para el estudio de este tema, lea detalladamente las páginas del libro de texto que se indican a continuación:

Capítulo	Sección	Página(s)
4 Programación orientada a objetos en Visual Basic .NET	4.1. Introducción	71
	4.2. Fundamentos de la programación orientada a objetos	72 – 78
	4.3. Métodos y propiedades	78

COMENTARIOS DEL TEMA

Capítulo 4. Programación orientada a objetos en Visual Basic .NET

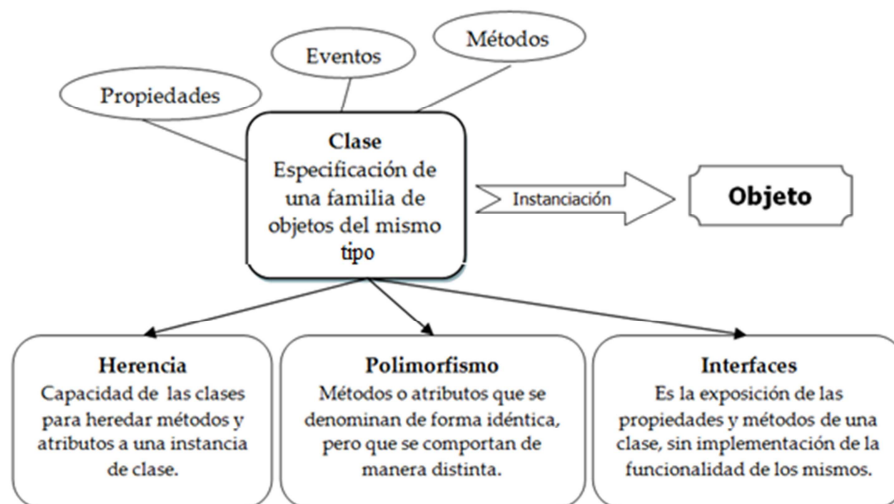
4.1 Introducción

La programación orientada a objetos no es un paradigma nuevo de desarrollo. Desde hace unos años se ha generalizado su utilización, determinándose el verdadero beneficio de su uso: la reutilización.

Los objetos son la parte fundamental de este paradigma; por lo tanto, son muchos los lenguajes existentes que le dan soporte, y sobre los cuales se desarrollan aplicaciones de negocios, científicas y de cualquier índole.

4.2 Fundamentos de la programación orientada a objetos.

La programación orientada a objetos se compone de una serie de conceptos que es necesario conocer para entender su funcionamiento. En el siguiente cuadro se resumen estos conceptos.





En la figura anterior se muestran algunos de los conceptos de la programación orientada a objetos; conociendo su funcionalidad y uso, se pueden desarrollar aplicaciones para Web realmente reutilizables.

Vea los fundamentos de la programación orientada a objetos en las páginas de la 72 a la 78 del libro de texto.

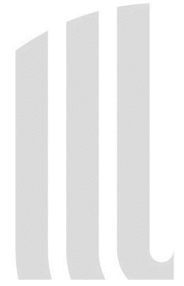
4.3 Métodos y propiedades

Los métodos de una clase implementan su funcionalidad. Es decir, establecen la operabilidad sobre los atributos de la clase. Así, un método `agregarCliente()` permite agregar atributos como cédula, nombre, ciudad o teléfono a una clase *Clientes*.

Las propiedades, por su parte, establecen el retorno o la actualización de los atributos de una clase. Generalmente, mediante un `Get`, podríamos obtener el valor de un atributo de la clase y, por medio de un `Set`, actualizar su valor.

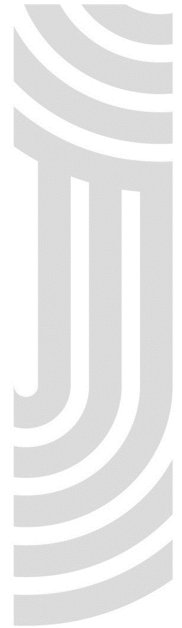
EJERCICIOS DE AUTOEVALUACIÓN DEL TEMA 3

- 1) ¿Qué es un objeto?
- 2) ¿Qué es una clase?
- 3) ¿Qué es instanciación de una clase?
- 4) ¿Cuál es la funcionalidad de un evento?
- 5) ¿Cuál es la importancia de los espacios de nombres?



GESTIÓN
DE DATOS

4





INTRODUCCIÓN

En este tema se desarrollan los conceptos básicos de la arquitectura que utiliza ADO .NET para la gestión de datos. Esta arquitectura permite crear objetos de conectividad y manejo de datos, ya sea para brindar mantenimiento o bien para permitir consultas. Estos objetos pueden trabajar de forma conectada o desconectada a la fuente de datos. También es posible construir aplicaciones con datos de diversas fuentes y formatos como los archivos planos, XML o bases de datos Oracle.

La seguridad también es un subtema incluido en este apartado, considerando que en aplicaciones web la información se ve expuesta a tal grado que puede ser víctima de vandalismo o uso inadecuado. Además, se incorporan algunas recomendaciones provenientes del propio Microsoft para el uso de ADO .NET en aplicaciones con Visual Studio .NET.

Al finalizar el estudio de este tema, usted estará en capacidad de describir las principales características de ADO .NET, su lógica funcional y la arquitectura de gestión que implementa, además de identificar los principales problemas que se pueden presentar en el desarrollo de una aplicación web e implementarlas recomendaciones adecuadas propuestas por Microsoft para evitar estos inconvenientes.



GUÍA DE LECTURA

Para el estudio de este tema, lea detalladamente las páginas que se indican a continuación:

Capítulo	Sección	Página(s)
6 ADO .NET	6.1. Información general de ADO .NET	137-139
	6.2. Arquitectura de ADO .NET	139
	6.3. Proveedores de ADO .NET	140-142
	6.4. Dataset en ADO .NET	142-146
	6.5. Datareader en ADO .NET	146-149
	6.6. Novedades de ADO .NET	149-152
	6.7. LINQ	152-172
9 Seguridad de aplicaciones ADO .NET	9.1. Seguridad de acceso al código	265
	9.2. Protección de aplicaciones de ADO .NET	266
	9.3. Recomendaciones para el diseño seguro de aplicaciones ADO .NET	266-267
	9.4. Diseño de la seguridad	267-268
	9.5. Modelo de amenazas	268
	9.6. Seguridad de la base de datos	268-270



COMENTARIOS DEL TEMA

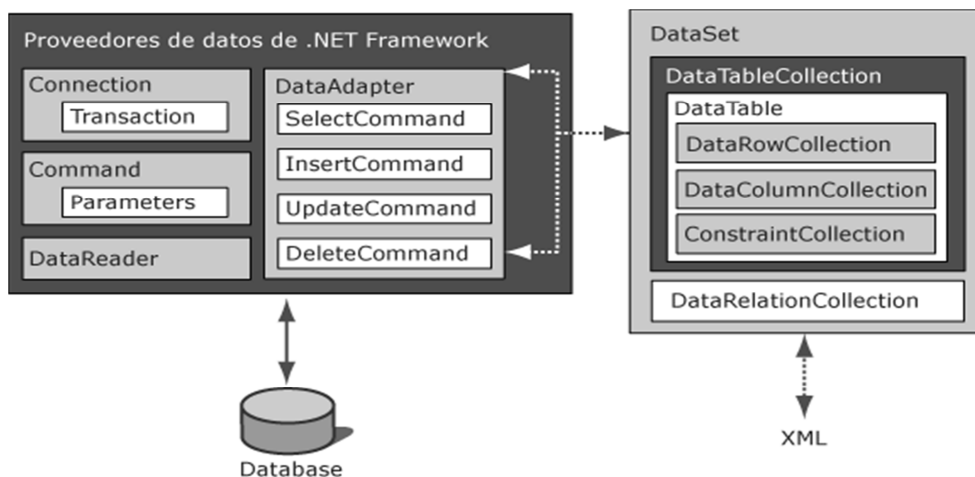
Capítulo 6. ADO .NET

6.1 Información general de ADO .NET

ADO .NET está compuesto por un conjunto de clases que implementan funcionalidades para la gestión de datos. Estas se refieren al mantenimiento y consultas de datos que radican en fuentes nativas de Microsoft tal como SQL Server. También existen clases Ole DB; las cuales permiten la conexión con otros orígenes de datos.

6.2 Arquitectura de ADO .NET

La arquitectura de ADO .NET se basa en una serie de clases que hacen funcionar la gestión de datos en un repositorio como SQL Server. Estas poseen propiedades y métodos que pueden ser instanciados y utilizados en una aplicación informática orientada a datos. En la siguiente imagen se muestra esta arquitectura.



Fuente: <<http://msdn.microsoft.com/es-es/library/27y4ybxw%28VS.80%29.aspx>>, tomada el 22 de setiembre de 2010.

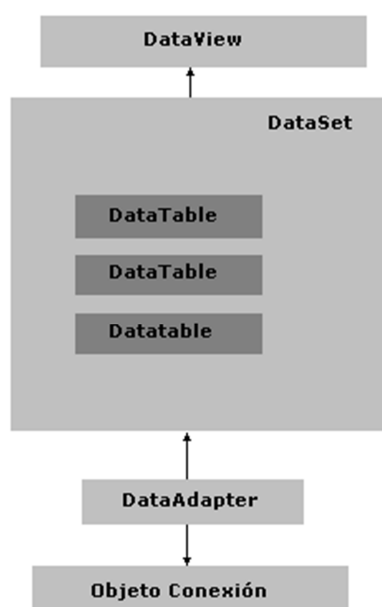


6.3 Proveedores de ADO .NET

Los proveedores de ADO .NET permiten la conexión a una base de datos determinada, ejecutando comandos y recuperando información mediante los objetos que implementa. La acumulación de datos se puede dar en memoria y en medios permanentes como un disco duro. Existen proveedores de datos para SQL Server, Oracle, entre otros.

6.4 Dataset en ADO .NET

Un dataset permite la representación de datos en memoria, puede ordenarlos y restringirlos; esos están almacenados en la caché de la memoria interna del PC y se compone de un conjunto de objetos datatable, los cuales pueden relacionarse entre sí mediante objetos datarelation. Permite, asimismo, gestionar la integridad de los datos. Puede leer y escribir datos y esquemas, sean de una fuente de datos como SQL Server, o documentos XML. En la siguiente imagen se muestra la arquitectura de un dataset en ADO .NET:



Fuente:

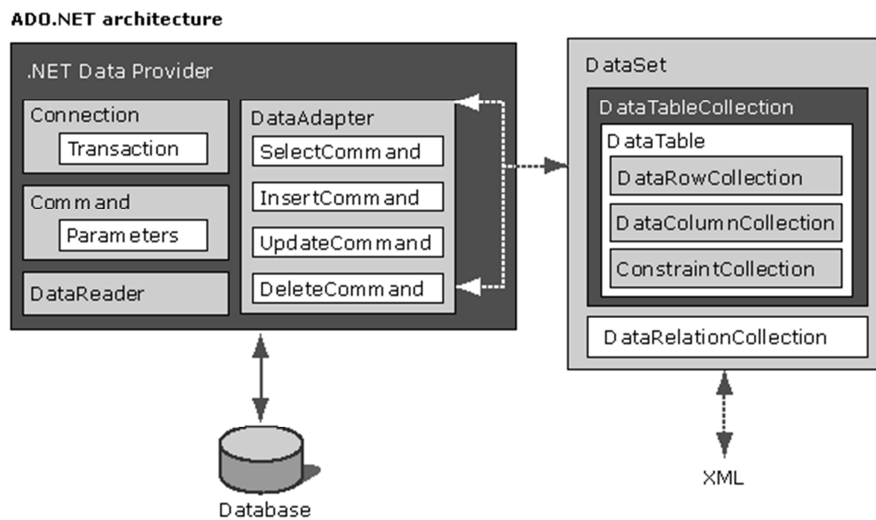
<<http://www.adrformacion.com/cursos/puntonetav/leccion3/tutorial5.html>>, tomada el 22 de setiembre de 2010.



6.5 Datareader en ADO .NET

Un datareader se utiliza para recuperar datos de solo lectura de una fuente. Estos se devuelven hacia el objeto a medida que se realiza la consulta, almacenándose en el búfer de red del usuario y presentándose hasta que se ejecute el método Read del datareader. El uso de datareader puede aumentar el rendimiento de una aplicación, tanto en la recuperación de datos como en el almacenamiento.

En la siguiente figura se muestra la arquitectura del datareader de ADO .NET:



Fuente: <http://www.akadia.com/img/dotnet_ado_architecture.gif>, tomada el 22 de setiembre, 2010.

6.6 Novedades de ADO .NET

ADO .NET del framework 3.5 de Visual Studio 2008, aporta muchas novedades en comparación con su antecesor, la versión 3.0. En la del 2008 existe soporte para Windows Workflow foundation (WWF), Windows Communication Foundation (WCF), Windows Presentation Foundation (WPF) y Windows CardSpace. Posee soporte para LINQ (Language Integrated Query), AJAX, JSON, REST, POX, RSS, ATOM y diversos estándares WS.

Véanse las diferentes novedades de ADO .NET en la página 149 del libro de texto.

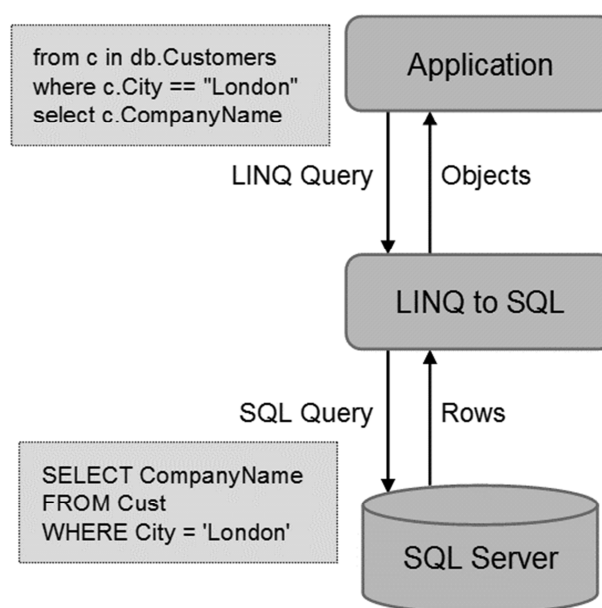


Con el .NET Framework se tienen otras novedades que puede ver ingresando al sitio <http://msdn.microsoft.com/es-es/library/ex6y04yf.aspx>, cuyo enlace se verificó el 22 de setiembre de 2010.

6.7 LINQ

LINQ es un proyecto Microsoft que ejecuta consultas semejantes a SQL en Visual Basic .NET y C#. Este proyecto incluye operadores de consultas estándar que permiten el filtrado y mantenimiento de datos, tal y como se usaría en SQL estándar. Se pretende establecer que el código desarrollado en estos lenguajes siga un modelo orientado a objetos.

En la siguiente imagen se muestra la funcionalidad de LINQ:



Fuente:

http://blogs.msdn.com/blogfiles/wriju/WindowsLiveWriter/LINQtoSQLExecutionArchitecture_B06A/clip_image001_2.gif, tomada el 22 de setiembre de 2010.



Capítulo 9. Seguridad de aplicaciones ADO .NET

9.1. Protección de aplicaciones de ADO .NET

Considerar la seguridad en el desarrollo de aplicaciones ADO .NET es un asunto de vital importancia. Existen muchas estratagemas que se utilizan en el medio informático para atacar o tratar de encontrar alguna vulnerabilidad en los sistemas. Por ello, es importante tener en cuenta el aspecto de la seguridad en el desarrollo de aplicaciones informáticas.

En este caso, .NET framework ofrece muchas clases, servicios y herramientas que benefician al desarrollador de aplicaciones, tales como CLR, CAS y protección en la gestión de datos.

9.2. Recomendaciones para el diseño seguro de aplicaciones ADO .NET

Para desarrollar aplicaciones seguras se deben considerar algunas recomendaciones, por ejemplo, contrarrestar amenazas como inyección de SQL, suplantación de privilegios, autenticación y contraseñas.

Véanse las recomendaciones para el diseño seguro de aplicaciones de ADO .NET en la página 266 del libro de texto.

9.3. Diseño de la seguridad

ClickOnce es una tecnología que permite establecer la seguridad en aplicaciones .NET. En resumen, la seguridad de aplicaciones .NET se basa en autenticación basada en credenciales, autorización según roles y administración de perfiles. En la página 268 del libro de texto se explican estas categorías de seguridad.

9.4. Modelo de amenazas

En el libro de texto se argumenta que para determinar un modelo de amenazas se debe considerar la visión del adversario (qué nos podría ocasionar), caracterizar la seguridad del sistema (cuán preparado se está para enfrentarlo) y determinar las posibles amenazas (cuáles perjuicios puede presentar un ataque a nuestros sistemas).



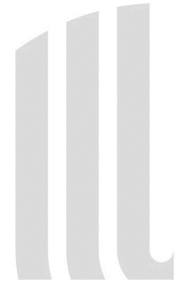
9.5. Seguridad de la base de datos

El modelo de seguridad de la base de datos que utiliza tecnología ADO .NET, establece una serie de recomendaciones surgidas, principalmente, por el propio Microsoft; como crear cuentas con los privilegios mínimos posibles, no permitir que los usuarios obtengan acceso a cuentas administrativas, no devolver mensajes de error de servidor a las aplicaciones cliente, validar todas las entradas tanto en el cliente como en el servidor, entre otras.

Véanse las recomendaciones para la seguridad de una base de datos que utilice tecnología ADO .NET en la página 269 del libro de texto.

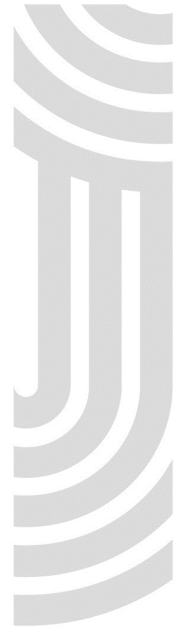
EJERCICIOS DE AUTOEVALUACIÓN DEL TEMA 4

- 1) Defina qué es la tecnología ADO .NET.
- 2) Cite al menos dos características de ADO .NET.
- 3) Cite al menos dos casos en los cuáles se utilizan dataset o datareader.
- 4) Mencione los dos nuevos marcos de ADO .NET incluidas con el framework 3.5 de Visual Studio .NET 2008.
- 5) Cite dos clases que se utilizan en la gestión de datos por parte de ADO .NET.
- 6) Defina qué es un dataset.
- 7) Defina qué es un datareader.
- 8) Cite al menos dos novedades de ADO .NET.
- 9) Explique en qué consiste una inyección a SQL.
- 10) Explique en qué consiste el esquema de seguridad basada en autenticación por credenciales.
- 11) Cite las consideraciones para determinar un modelo de amenazas.
- 12) Cite dos estrategias de seguridad de base de datos en la cual se utilice la tecnología ADO .NET.



TECNOLOGÍAS
PARA EL DESARROLLO WEB

5





INTRODUCCIÓN

En este tema se tratan los conceptos básicos de una de las estrategias existentes en el mercado para el desarrollo de sitios web: *ASP .NET*. Esta tecnología desarrollada por *Microsoft* se ha extendido a nivel mundial y, actualmente, se utiliza de forma generalizada por muchos desarrolladores individuales y empresariales.

Entre los temas que se desarrollan se encuentra el tratamiento de los servicios web como intermediario entre las capas de datos, lógica de negocios y la interfaz del usuario. Con ello, se espera que el estudiante asimile esta forma de trabajo y pueda familiarizarse con el concepto; asimismo, se desarrolla el tema de XML como formato de información utilizado en el intercambio de datos entre aplicaciones web.

Al finalizar el estudio de este tema, usted estará en capacidad de describir el marco de trabajo que utiliza *ASP .NET* en el desarrollo de sitios web, de aplicar el uso de servicios web para el manejo de datos y de aplicar el uso de archivos XML en el intercambio de información entre aplicaciones web.



GUÍA DE LECTURA

Para el estudio de este tema, lea detalladamente; del libro de texto, las páginas que se indican a continuación.

Capítulo	Sección	Página(s)
10 ASP .NET	10.1. ¿Qué es ASP .NET?	271-272
	10.2. Ventajas que ofrece ASP .NET	272-273
	10.3. Arquitectura de una aplicación ASP .NET	273
	10.4. Formularios web ASP .NET	273
	10.5. IDE de ASP .NET	274
	10.6. Directorio virtual y físico en aplicaciones ASP .NET	274
	10.7. Ubicación de proyectos ASP .NET	275-276
	10.8. IIS local y de servidor	277-278
	10.9. Creación de páginas maestras en ASP .NET	278-279
	10.10. Páginas de contenido	279-287
	10.11. Navegación entre formularios	287
	10.12. Código subyacente en formularios web	287-289
	10.13. Definición de entradas de usuario	289-290
	10.14. Uso de eventos de servidor o de cliente	290-291
	10.15. Controles de validación de ASP .NET	291-297
	10.16. Variables de aplicación y sesión en ASP .NET	297-299
	10.17. Cookies en ASP .NET	299-304
	10.18. Validación en controles ASP .NET	304-308

Capítulo	Sección	Página(s)
11 Fundamentos de servicios web	11.1. Servicios web (Web Services)	311-313
	11.2. Estándares de servicios web	313-314
	11.3. Ventajas de utilizar servicios web	314
	11.4. Inconveniencias de utilizar servicios web	314-315
	11.5. Tres razones para crear servicios web	315
	11.6. Ejemplo de aplicación: servicios web para una calculadora	315-319
13 Implementando servicios web con ADO .NET	13.1. ADO .NET en ASP .NET	347-348
	13.2. Web.config	348-350
	13.3. Servicios web con ADO .NET	350-355
	13.4. Prueba de servicios web en un explorador de Internet	355-357
14 Implementación de servicios web en aplicaciones web ASP .NET	14.1. Implementación de servicios web en aplicaciones ASP .NET	359-369
	14.2. Configuración y codificación del formulario webConsultaCuenta.aspx	369-373
	14.3. Configuración y codificación del formulario webPagoPrestamo.aspx	373-380
	14.4. Configuración y codificación del formulario webPagoTransferencia.aspx	380-386
	14.5. Configuración y codificación del formulario webPagos.aspx	386-391
	14.6. Mostrar datos de maestro-detalle en la misma página	391-396
	14.7. Configurando el menú principal	396-398
	14.8. Utilizando full text search de SQL Server en ASP .NET	398-410
15 Uso de datos XML en aplicaciones ASP .NET	15.1. Introducción	413
	15.2. Definición de XML	413-414
	15.3. Componentes de un documento XML	414-415
	15.4. Tecnologías XML	415-417
	15.5. Ejemplo de aplicación	417-422



COMENTARIOS DEL TEMA

Capítulo 10. ASP .NET

10.1 ¿Qué es ASP .NET?

ASP .NET es un marco de trabajo que se utiliza para desarrollar sitios web. Se marcan adecuadamente la capa de presentación y la de contenido para tener un entorno que separa ambos escenarios, aunque mantiene la integración del proyecto de desarrollo. Se puede utilizar Visual Basic o C# para desarrollar páginas web ASP .NET.

10.2 Ventajas que ofrece ASP .NET

ASP .NET tiene varias ventajas, pueden citarse: un mejor rendimiento que las versiones antecesoras, compatibilidad con la lógica de trabajo de un desarrollador de aplicaciones Windows que utiliza Visual Studio .NET, eficacia y flexibilidad en su marco de trabajo, simplicidad, facilidad de uso, entre otras ventajas.

Vea las ventajas que ofrece ASP .NET en la página 272 del libro de texto.

10.3 Arquitectura de una aplicación ASP .NET

La arquitectura de una aplicación web posibilita determinar la forma como trabaja la estrategia. El funcionamiento de todos los componentes de una aplicación web desarrollada mediante ASP .NET permite estructurarla de una forma lógica y orientada a objetos. La funcionalidad de los objetos de enlace a datos también permite una visión dinámica de la estructuración de una aplicación web desarrollada con ASP .NET.



10.4 Formularios web ASP .NET

Los formularios web constituyen la interfaz de un sitio que permite la interacción con el usuario; también a que internamente se implemente lógica de negocios mediante procesos programados. Existe una importante cantidad de componentes que pueden ser implementados en formularios web, con lo cual se enriquece la experiencia del usuario en el uso de esta interfaz.

10.5 IDE de ASP .NET

El IDE o entorno de Visual Studio es totalmente integrado, por lo que será el mismo si se desarrolla una aplicación para Windows, Web o dispositivo móvil. Por ende, una familiarización con este entorno permitirá que el desarrollador se encuentre cómodo creando cualquiera de este tipo de aplicaciones.

10.6 Directorio virtual y físico en aplicaciones web ASP .NET

Cuando se desarrolla un sitio web, este podrá ejecutarse en un directorio virtual que, en el caso de Windows, por lo general, físicamente se ubicaría en la dirección `c:\inetpub\wwwroot` y virtualmente se representaría en `http://localhost`

10.7 Ubicación de proyectos ASP .NET

De acuerdo con su ubicación, ASP .NET permite crear tres tipos de proyectos: sistema de archivos, sistemas HTTP y sistemas FTP.

- Sistema de archivos

Un sistema de archivos permite mantener los archivos que la conforman en una carpeta de disco local, no constituyéndose en una aplicación de IIS (Internet Information Server).

- HTTP

Se crean aplicaciones web ASP .NET de tipo HTTP cuando se utilice IIS como servidor web; la solución estaría en un servidor de Internet. En este caso, el sitio sería accesible externamente por muchos usuarios.



- FTP

Una solución FTP establece comunicaciones a través de internet pero no ejecuta páginas web. Se utiliza para crear un sitio web para descarga de archivos.

10.8 IIS local y servidor

Un servidor IIS local es aquel instalado en la máquina del desarrollador para que pueda recrear un ambiente real de ejecución de un sitio web en desarrollo. Mientras que un IIS que se instala en un servidor constituirá el repositorio general del sitio por donde accederán los usuarios externos de este.

10.9 Creación de páginas maestras en ASP .NET

Las páginas maestras constituyen un estándar de diseño que permite que un sitio web cuente con una plantilla general para desarrollar el mismo. Con una página maestra podemos elaborar una plantilla que reutilizaremos en todo el sitio, permitiendo un diseño equilibrado y estandarizado.

10.10 Páginas de contenido

Se utilizan para mostrar la información dinámica de una página. Una página maestra es una plantilla que estandariza el diseño y presentación del sitio, mientras que las páginas de contenido sirven para mostrar la información del sitio web.

10.11 Navegación entre formularios

En ASP .NET, la navegación entre formularios se realiza mediante un `Response.Redirect` (nombre página), lo cual resulta muy fácil de asimilar.

10.12 Código subyacente en formularios web

El código subyacente referencia el código de una página web ASP .NET que se encuentra en un archivo de clases independiente. Con ello, se separa la lógica de presentación y el HTML de la página. El siguiente fragmento es un ejemplo de código subyacente:



```
<%@ Language="C#" Inherits="MiCodigoAux.MiClase" %>
<HTML>
  <body>
    <form id="MiFormulario" runat="server">
      <asp:textbox id="MiCuadroDeTexto" text="Hola Mundo!"
runat="server"></asp:textbox>
      <asp:button id="MiBoton" text="Echo Input" OnClick="MiBoton_OnClick"
runat="server"></asp:button>
      <asp:label id="MiEtiqueta" runat="server" />
    </form>
  </body>
</HTML>
```

10.13 Uso de eventos de servidor o de cliente

Los eventos son mensajes enviados para anunciar que ha sucedido algo. Estos eventos se implementan mediante delegados, lo que permite invocar, indirectamente, una función mediante una referencia a esta.

En ASP .NET se pueden utilizar tanto eventos del lado del cliente como del lado del servidor. En el primero, es importante cuando se requiere algún procesamiento local antes de ser enviados al servidor; por ejemplo, la longitud de una contraseña o la validez en el formato de una fecha. Del segundo, es cuando los procedimientos son ejecutados directamente en este.

10.14 Controles de validación de ASP .NET

Existen muchos controles en ASP .NET que facilitan las labores de validación en una página web; con ellos se automatizan muchos de los procesos requeridos para validar alguna funcionalidad. Por ejemplo, podemos validar el requerimiento de entrada de un dato, una expresión o la comparación entre dos valores para determinar si son iguales.



10.15 Variables de aplicación y sesión en ASP .NET

Una variable de aplicación se utiliza para mantener un valor que se mantiene durante toda una sesión del sitio web. Por ejemplo, si se desea mantener el nombre del usuario permanentemente mientras usa el sistema, podría almacenarlo en una variable de aplicación.

Una variable de sesión debe ser redefinida entre cada ejecución de formulario. Como se necesita mantener un valor al pasar de un formulario a otro, debe crearse una variable de sesión para este.

10.16 Cookies en ASP .NET

Constituyen un trozo de código que acompaña a las solicitudes de usuario y a las páginas web mientras viajan hacia un servidor web y viceversa. Se asocian a un sitio web y no a una página particular.

Existen dos tipos de cookies: temporales y persistentes. Las primeras solo existen en la memoria del navegador que se utilice. Las persistentes se limitan al tiempo de expiración que se les determine.

10.17 Validación en controles ASP .NET

La validación en un sitio web es un asunto sumamente trascendental, pues reducirá los errores que puedan presentarse al ejecutar la solución web. Para ello, Visual Studio proporciona varios controles que pueden utilizarse fácilmente.

Capítulo 11. Fundamentos de servicios web (Web Services) en ASP .NET

11.1 Servicios web (Web Services)

Un servicio web se define como un conjunto de protocolos y estándares que se utilizan para el intercambio de datos entre distintas aplicaciones. Estas pueden haber sido desarrolladas utilizando diferentes lenguajes de programación.



Mediante servicios web, podemos programar lógica de negocios que realice validaciones, comparaciones, cálculos, procesos, entre otras actividades; también desarrollar en ellos lógica de acceso a datos, por ejemplo funciones de selección o mantenimiento de datos.

11.2 Estándares de servicios web

Según el libro de texto, un estándar permite establecer reglas sobre las cuales, tanto desarrolladores como usuarios, se basan para la creación o utilización de software.

Existen varios estándares que se utilizan para la creación y uso de servicios web, entre ellos están SOAP, WSDL, UDDI, etc.

Véanse los detalles de los estándares para servicios web en las páginas 313 y 314 del libro de texto.

11.3 Ventajas de utilizar servicios web

Algunas ventajas de utilizar servicios web son: interoperabilidad, fomento de estándares, aprovechamiento de los sistemas de seguridad e integración.

Véanse los detalles de la importancia de utilizar servicios web en la página 314 del libro de texto.

11.4 Inconveniencias de utilizar servicios web

Se presentan inconveniencias en el momento de utilizar servicios web, entre los cuales están: transaccionalmente poco desarrollados, inferiores en rendimiento en comparación con RMI, Corba o DCOM. Como se apoyan en HTTP, pueden infringir la seguridad de los firewall de una red.

Véanse los detalles de la inconveniencia de utilizar servicios web en las páginas 314 y 315 del libro de texto.



11.5 Tres razones para crear servicios web

Las tres razones para utilizar servicios web, en aplicaciones orientadas a Internet, son:

- Como se apoya en HTTP, se evita lidiar con los problemas de acceso a la red.
- SOAP aporta interfaces adecuadas para apoyarse entre equipos de distintas arquitecturas.
- Independencia entre la aplicación web y los servicios web.

Véanse los detalles de estas tres razones para utilizar servicios web en la página 315 del libro de texto.

11.5 Ejemplo de aplicación: servicios web para una calculadora

Lo más práctico para probar el uso de un servicio web es crear una aplicación. En este caso, se desarrolla un sitio web que implemente la funcionalidad de una calculadora web. En el libro de texto se explica detalladamente su funcionalidad, así como la implementación de su código fuente.

Véanse los detalles de la implementación de un sitio web que utiliza Web Service de la página 315 a la 326 del libro de texto.

Capítulo 13. Implementando servicios web con ADO .NET

13.1 ADO .NET en ASP .NET

ADO .NET permite gestionar datos en forma desconectada para luego cargarlos desde una fuente de datos. Esto posibilita la independencia de la gestión de datos al trabajarlo directamente en la memoria de la máquina cliente y no en el servidor. Funciona de forma similar que en aplicaciones Windows.



13.2 Web.config

El Web.config es un archivo XML que permite configurar muchas opciones de una aplicación web. Entre las opciones se encuentran: conexiones a bases de datos, seguridad, estados de sesión, compilación y lenguaje.

13.3 Servicios web con ADO .NET

Los servicios web trabajan de forma similar que una DLL creada para una aplicación Windows; en este caso, constituyen código separado de una aplicación web que incorpora código funcional. Se puede implementar código para la lógica de negocios o para la de datos en un servicio web. En cuanto a esta última, se puede implementar el acceso o gestión de los datos, utilizando ADO .NET.

13.4 Prueba de servicios web en un explorador de Internet

Se puede realizar la funcionalidad de un servicio web, ejecutándolo directamente en un explorador web. Con ello, es posible determinar si los resultados de las funciones implementadas, en dicho servicio web, son los esperados.

Capítulo 14. Implementación de servicios web en aplicaciones ASP .NET

14.1 Implementación de servicios web en aplicaciones ASP .NET

La implementación de un servicio web en una aplicación ASP .NET consiste en desarrollar el servicio primero, luego probarlo y compilarlo. Posteriormente, en la aplicación web, se crea una referencia a dicho servicio y, mediante instanciación, se usan todas las funcionalidades que tiene implementadas. Se deben tener algunas consideraciones para implementar servicios web en aplicaciones ASP .NET.

Véanse las consideraciones requeridas a la hora de referenciar un servicio web en una aplicación ASP .NET en las páginas 359 y 360 del libro de texto.



14.2 Configuración y codificación del formulario webConsultaCuenta.aspx

Este formulario web tiene como propósito implementar la funcionalidad de consultar las cuentas de un cliente, una vez que este se haya validado en el sistema.

14.3 Configuración y codificación del formulario webPagoPrestamo.aspx

Este formulario web tiene como fin implementar la funcionalidad de consultar los préstamos de un cliente una vez que éste se haya validado en el sistema y, así, proceder a cancelarlos o abonarlos.

14.4 Configuración y codificación del formulario webPagoTransferencia.aspx

Este formulario web tiene como objeto implementar la funcionalidad de realizar transacciones de las cuentas de un cliente hacia la cuenta o cuentas de otro cliente (transferencias).

14.5 Configuración y codificación del formulario webPagos.aspx

Este formulario web tiene como propósito implementar la funcionalidad de realizar pagos (servicios públicos, por ejemplo) a través de la agencia virtual bancaria. Lógicamente, las deducciones se realizan de las cuentas del cliente que posean suficientes fondos.

14.6 Mostrar datos de maestro-detalle en la misma página

Este formulario web tiene como fin implementar la funcionalidad de consultar las transferencias de un cliente de acuerdo con una determinada cuenta. El archivo tbCuentas será la tabla maestra; mientras que la tabla tbTransferencias detalle.

14.7 Configurando el menú principal

Este formulario web tiene como objeto implementar la funcionalidad del menú principal de la aplicación web.



14.8 Utilizando full text search de SQL Server en ASP .NET

Muchas veces se requieren sitios web que realicen consultas complejas en grandes bases de datos documentales. SQL Server lo permite a bases de datos documentales creando catálogos indexados para consultas más eficientemente, utilizando para ello dos comandos especializados: Contains y FreeText.

- Contains

Mediante Contains, se puede realizar una búsqueda de un texto específico en una tabla. Este comando puede buscar en grandes campos textuales siempre y cuando se encuentre catalogado. No hace diferenciación entre mayúsculas o minúsculas.

- Freetext

Con FreeText se puede obtener una frase completa o parte de ella en la búsqueda en un campo de texto. A diferencia de Contains, puede buscar coincidencias exactas en toda una columna de texto.

Capítulo 15. Uso de datos XML en aplicaciones web ASP .NET

15.1. Introducción

XML significa Xtended Markup Language o lenguaje de marcas extendido. Es un lenguaje que permite la separación de la capa de datos del proceso que transforma o utiliza esos datos. Tal como se usaba en HTML, en XML también se utilizan etiquetas no predefinidas, por lo que puede crear estructuras anidadas.

15.2. Definición de XML

XML se define como un metalenguaje que permite formatear datos, facilitando la declarativa de estos, así como su almacenamiento y recuperación. En el libro de texto se define como un formato universal que se utiliza para describir e intercambiar documentos y datos de forma estructurada en Internet.

Véanse las características de XML en la página 414 del libro de texto.
--



15.3. Componentes de un documento XML

Los componentes de un documento XML son elementos, atributos y comentarios. Los elementos se refieren a una colección de datos XML tales como clientes, proveedores, empleados, entre otros. Estos proporcionan las estructuras que servirán para almacenar los datos. Los atributos conforman la estructuración de cada campo en el grupo de datos (elemento) y los comentarios se utilizan para escribir anotaciones que describan el detalle de elementos y/o atributos.

15.4. Tecnologías XML

Las tecnologías XML son XSD, XSLT, XPath, DOM, XQuery. Cada una tiene características particulares que las definen y las diferencian a la vez. En el caso de XSD, constituye un estándar para la definición de esquemas para la estructura de un documento; XSLT permite la transformación del contenido de un documento XML a otro formato, XPath direcciona un documento XML; DOM es un modelo de objetos para trabajar en memoria y XQuery se utiliza para consultas XML.

15.5. Ejemplo de aplicación

En este apartado se crea una pequeña aplicación para demostrar el uso de XML. Es importante analizar su contexto, así como el código implementado.



EJERCICIOS DE AUTOEVALUACIÓN DEL TEMA 5

- 1) ¿Qué es ASP .NET?
- 2) Describa el funcionamiento de los componentes de una aplicación desarrollada con ASP .NET.
- 3) ¿Cuál es la funcionamiento de los objetos de enlace a datos en una aplicación ASP .NET?
- 4) Describa la funcionalidad de un directorio virtual y uno físico en aplicaciones web ASP .NET.
- 5) Explique la funcionalidad de las páginas maestras y de contenido en una aplicación web ASP .NET.
- 6) ¿A qué se le llama código subyacente en una aplicación web ASP .NET?
- 7) ¿Cómo funcionan los eventos del lado del servidor y del cliente?
- 8) Defina qué son variables de aplicación.
- 9) Defina qué es una cookie. ¿Qué tipos existen y cuándo se usan? ¿Qué estándares se utilizan para crearlos?
- 10) ¿Cómo se define un servicio web ASP .NET?
- 11) ¿Qué ventajas existen cuando se utilizan servicios web?
- 12) ¿Para qué se utiliza un archivo web.config?
- 13) Si utiliza Full Text Search en SQL Server ¿para qué se utilizan contains y freetext?
- 14) Defina XML.
- 15) ¿Cuáles son los componentes de un documento XML?



RESPUESTAS A LOS EJERCICIOS DE AUTOEVALUACIÓN

Respuestas a los ejercicios de autoevaluación del tema 1

- 1) Las características que incluye Visual Studio .NET son los lenguajes de desarrollo Visual Basic .NET y C#, principalmente, y muchas funciones incluidas en el framework que le permiten desarrollar rápidamente aplicaciones para Windows, Web o para dispositivos móviles.
- 2) ASP .NET permite la creación de sitios y servicios web, ya sea mediante páginas estáticas (que manejan solo información) o dinámicas (que manejan datos de una base de datos o de cualquier otra fuente soportada).
- 3) ADO .NET es la metodología propuesta por Microsoft® para la gestión de datos provenientes de varias fuentes: nativas como SQL Server o externas tales como Oracle.
- 4) LINQ permite ejecutar consultas y brindar mantenimiento de datos en una aplicación desarrollada con Visual Studio .NET. Es un lenguaje similar a SQL con la excepción que implementa su propia sintaxis.
- 5) El framework de Visual Studio .NET funciona como un entorno de trabajo y de ejecución de aplicaciones desarrolladas con la herramienta o con aquellas que son compatibles con este motor. Incluye varios componentes y librerías reutilizables que permiten el desarrollo de aplicaciones informáticas orientadas a Windows, Web o para dispositivos móviles.
- 6) El CLR (Common Language Runtime) es un entorno administrado el cual permite la ejecución de aplicaciones que se creen con Visual Studio .NET o aquellas que soporten el Framework de Visual Studio .NET.

- 7) La principal ventaja del entorno de Visual Studio .NET es que integra en un sólo IDE toda la funcionalidad de la herramienta; sin importar el lenguaje con el cual se desarrolle la solución informática, siempre tendrá la misma interfaz de desarrollo, por lo que el programador se familiarizará rápidamente con ella.
- 8) Existen varios tipos de proyectos que se pueden desarrollar con Visual Studio .NET. Entre estos tipos de proyectos se encuentran aplicaciones para Windows, para Web, para dispositivos móviles, servicios Windows y Web, aplicaciones WPF, entre otros.

Respuestas a los ejercicios de autoevaluación del tema 2

- 1) Dos entornos de variables que se utilizan en Visual Studio .NET son:
 - a) ámbito de procedimiento: bloque de código en que se ha declarado.
 - b) ámbito de espacio de nombre: disponible para todo el código dentro del espacio de nombres en que se ha declarado.
- 2) Dos funciones de cadena:
 - a) Mid: obtiene una subcadena a partir de una cadena.
 - b) Format: da formato a una cadena de texto.Dos funciones de fecha:
 - a) Now: obtiene la fecha y hora actual del sistema.
 - b) Dateadd: agrega una variable de tipo fecha.Dos funciones matemáticas:
 - a) Round: efectúa un cálculo de redondeo.
 - b) Pow: permite elevar un número a una potencia específica.

Respuestas a los ejercicios de autoevaluación del tema 3

- 1) Un objeto se puede definir como una agrupación de código que se compone de propiedades, métodos, atributos, eventos, entre otros. Proviene de implementar una clase mediante a una instanciación de dicha agrupación.
- 2) Una clase se define como una especificación de objetos de un mismo tipo, que comparten las mismas propiedades, métodos, atributos y demás componentes.
- 3) Instanciar una clase es crear un objeto derivado de ella. Es crear una instancia de la clase para ser utilizada dentro de un programa computacional, dado que la clase por sí misma no puede ser utilizada.

- 4) Un evento permite que los objetos que lo contienen puedan comunicarse con el objeto que lo está manipulando.
- 5) Los espacios de nombres se utilizan para agrupar clases con una lógica funcional determinada. Por ejemplo, pueden crearse clases que implementen la seguridad, el acceso a bases de datos, utilitarios, entre otras funcionalidades en un solo espacio de nombres.

Respuestas a los ejercicios de autoevaluación del tema 4

- 1) ADO .NET es la tecnología propuesta por Microsoft® para la conectividad y gestión de datos.
- 2) Dos características de ADO .NET son:
 - a) Soporte nativo para XML y la posibilidad de serializar objetos a XML.
 - b) Se cuenta con el comando For Each que permite recorrer los elementos de un dataset sin necesidad de utilizar el comando MoveNext.

Puede ver más características en la página 138 del libro de texto.

- 3) Dos casos en los cuales utilizar dataset o datareader:
 - a) Si se requiere escritura o lectura hacia y desde una fuente de datos (los datareader son solo de lectura).
 - b) Si requiere que un formulario web se enlace a datos a través de un control. Los datareader lo hacen a un único control.

Puede ver más casos en la página 139 del libro de texto.

- 4) Los dos nuevos marcos de trabajo que se ofrecen en el framework 3.5 de Visual Studio .NET 2008 son ADO .NET Entity Framework y ADO .NET Data Services.
- 5) Dos clases son:
 - a) DbTransaction. Esta clase incluye los comandos necesarios para realizar transacciones en un origen de datos.
 - b) DbError. Permite obtener la información relacionada con una advertencia o un error que se producen en un origen de datos.

Puede ver más clases en la página 141 del libro de texto.

- 6) Un dataset es una caché de memoria interna con datos recuperados de una fuente previamente conectada. Es un objeto desconectado de la fuente de datos.
- 7) Un datareader es un objeto que genera un flujo de solo lectura (read only) y sólo hacia adelante (forward only) Por lo tanto, únicamente puede almacenar un registro en memoria cada vez que lo extrae del servidor.

- 8) Dos novedades de ADO .NET son:
 - a) Entity Data Model (EDM)
 - b) Linq To Entities.
- 9) Una inyección a SQL es una amenaza que se caracteriza por transacciones SQL que se pueden empotrar en entradas válidas, pasar al servidor sin ningún tipo de validación y daña los datos que se encuentran en la base de datos.
- 10) Una credencial es generalmente una contraseña que se guarda en un almacén de identidades (identity storage). Puede establecerse en un servidor de directorios (como Active Directory).
- 11) Se considera:
 - a) Comprender la visión del adversario.
 - b) Caracterizar la seguridad del sistema.
 - c) Determinar las posibles amenazas.
- 12) Dos estrategias son:
 - a) Crear cuentas con privilegios mínimos.
 - b) No devolver mensajes de error de servidor a las aplicaciones cliente.

Puede ver más recomendaciones en la página 269 del libro de texto.

Respuestas a los ejercicios de autoevaluación del tema 5

- 1) ASP .NET es un marco de trabajo que se utiliza para desarrollar sitios web. Se marca adecuadamente la capa de presentación y la de contenido para tener un entorno que separa ambos escenarios, aunque mantiene la integración del proyecto de desarrollo.
- 2) El funcionamiento de todos los componentes de una aplicación web, desarrollada mediante ASP .NET, permite estructurarla de una forma lógica y orientada a objetos.
- 3) La funcionalidad de los objetos de enlace a datos también permite una visión dinámica de cómo se estructura una aplicación web desarrollada con ASP .NET.
- 4) Un sitio web se ejecuta en un directorio virtual, que en el caso de Windows, se ubica en la dirección <c:\inetpub\wwwroot>, y virtualmente se representaría en <http://localhost>.
- 5) Las páginas maestras constituyen un estándar de diseño el cual permite a un sitio web contar con una plantilla general para desarrollarlo. Con una página maestra, podemos diseñar una plantilla que reutilizaremos en todo el sitio, permitiendo un diseño equilibrado y estandarizado. Las páginas de contenido se utilizan para mostrar la información dinámica de una página. Una página maestra es una plantilla que

estandariza el diseño y presentación del sitio, mientras que en las de contenido se mostrará la información del sitio web.

- 6) El código subyacente referencia el código de una página web ASP .NET que se encuentra en un archivo independiente de clases. Con ello, se separa la lógica de presentación y el HTML de la página.
- 7) Un evento del lado del cliente es importante cuando se requiere algún procesamiento local antes de ser enviados al servidor. Del lado del servidor es cuando los procedimientos son ejecutados directamente en el servidor.
- 8) Una variable de aplicación se utiliza para mantener un valor que permanece durante toda una sesión del sitio web.
- 9) Constituye un trozo de código que acompaña a las solicitudes de usuario y a las páginas web, mientras estas viajan hacia un servidor web y viceversa. Se asocian a un sitio web y no a una página particular. Existen dos tipos de cookies: temporales y persistentes. Las primeras solo existen en la memoria del navegador que se utilice. Las persistentes se limitan al tiempo de expiración que se les determine.
- 10) Un servicio web se define como un conjunto de protocolos y estándares que se utilizan para el intercambio de datos entre distintas aplicaciones. Estas pueden haber sido desarrolladas utilizando diferentes lenguajes de programación. Existen varios estándares que se utilizan para la creación y uso de servicios web, entre los cuales están SOAP, WSDL, UDDI, entre otros.
- 11) Algunas ventajas importantes de utilizar servicios web son: interoperabilidad, fomento de estándares, aprovechamiento de los sistemas de seguridad e integración.
- 12) El Web.config es un archivo XML que permite configurar muchas opciones de una aplicación web. Entre otras opciones que se pueden configurar se encuentran: conexiones a bases de datos, seguridad, estados de sesión, compilación, lenguaje, entre otros.
- 13) Mediante Constains se puede realizar una búsqueda de un texto específico en una tabla. Este comando es capaz de hacerlo en grandes campos textuales, siempre y cuando se encuentre catalogado; no hace diferenciación entre mayúsculas o minúsculas. Con FreeText se puede obtener una frase completa o parte de ella en la búsqueda en un campo de texto. A diferencia de Constains, puede buscar coincidencias exactas en toda una columna de texto.
- 14) XML se define como un metalenguaje que permite formatear datos, facilitando tanto su declarativa como su almacenamiento y recuperación. En el libro de texto se define como un formato universal utilizado para describir e intercambiar documentos y datos de forma estructurada en Internet.
- 15) Los componentes de un documento XML son elementos, atributos y comentarios.