

¿Qué ofrece Autentia Real Business Solutions S.L?

Somos su empresa de **Soporte a Desarrollo Informático**.
 Ese apoyo que siempre quiso tener...

1. Desarrollo de componentes y proyectos a medida



2. Auditoría de código y recomendaciones de mejora

3. Arranque de proyectos basados en nuevas tecnologías

1. Definición de frameworks corporativos.
2. Transferencia de conocimiento de nuevas arquitecturas.
3. Soporte al arranque de proyectos.
4. Auditoría preventiva periódica de calidad.
5. Revisión previa a la certificación de proyectos.
6. Extensión de capacidad de equipos de calidad.
7. Identificación de problemas en producción.



4. Cursos de formación (impartidos por desarrolladores en activo)

Spring MVC, JSF-PrimeFaces /RichFaces,
 HTML5, CSS3, JavaScript-jQuery

Gestor portales (Liferay)
 Gestor de contenidos (Alfresco)
 Aplicaciones híbridas

Tareas programadas (Quartz)
 Gestor documental (Alfresco)
 Inversión de control (Spring)

Control de autenticación y
 acceso (Spring Security)
 UDDI
 Web Services
 Rest Services
 Social SSO
 SSO (Cas)

JPA-Hibernate, MyBatis
 Motor de búsqueda empresarial (Solr)
 ETL (Talend)

Dirección de Proyectos Informáticos.
 Metodologías ágiles
 Patrones de diseño
 TDD

BPM (jBPM o Bonita)
 Generación de informes (JasperReport)
 ESB (Open ESB)

	Hosting Patrocinado por enREDados.com 
---	--

[Home](#) | [Quienes Somos](#) | [Empleo](#) | [Foros](#) | [Tutoriales](#) | [Servicios Gratuitos](#) | [Contacte](#)

	<p>Tutorial desarrollado por: @ Roberto Canales Mora 2003-2005.</p> <p>Si crees que te he ayudado con este tutorial tambien tu me puedes ayudar a mi..</p> <p>Es muy fácil, mándame un correo con los datos de contacto de tu responsable de Formación y/o Recursos Humanos y es posible que podamos ofrecer un curso interesante a tu organización (y por tanto a tí). Mas detalles...</p> <p>Contacta: rcanales@autentia.com.</p>	
---	---	---

Descargar este documento en formato PDF [devesb.pdf](#)

[Curso de Programación](#)

Curso Presencial en Madrid
 Solicita más información.

[Desarrollo](#)

Estamos presentes en 132 países
 para atender a niños sin hogar

[SourceStyler C++](#)

An advanced C/C++ code
 formatter Download a free trial
 today!

[Visual Studio .NET](#)

Learn online at your own pace.
 Certifications. Job Placements.

Anuncios Goooooogle

Primer vistazo a C# en Visual Studio 2005 Beta2

Ya me ha llegado el CD con la Beta 2 de Visual Studio 2005 y me he puesto a probarlo... y como siempre ... aprovecho para compartirlo con vosotros (por cierto ... que también os podéis animar)

Aunque antes de empezar voy a hacer unas reflexiones en voz alta (que podéis no compartir y criticar):

Los que vivimos de la informática de modo profesional creo que tenemos que ser objetivos. Si hacemos la pregunta **¿que es mejor C# o JAVA?** yo respondería "**y a quién le importa ...**". Si son nuestros clientes y el mercado los que deciden las tecnologías. Nosotros tenemos que adaptarnos a las dinámicas del mercado (aunque siempre hay una pequeña capacidad de aconsejar y marcarlas).. Lo mejor y peor normalmente es subjetivo... la prueba está que en España hay clientes pro Microsoft y pro software libre y Java ... es igual que cuando preguntas si es mejor una marca de coches que otra (del mismo segmento) si las dos sobreviven... es que gustarán a suficientes personas ambas.

Yo personalmente pienso que los extremos son siempre malos ... y hay que tener criterio:

- Si miráis la aplicación demo del CD de la Beta 2 y comprobáis lo que cuesta hacer una automatización de extracción de datos dinámicos de Excel te quedas flipao.... yo usaría el Wizard de Microsoft para hacer esto
- Lo mismo si quisiera hacer un interfaz intuitivo para usuarios finales tradicionales ... pues usaría entornos Microsoft donde la productividad, rendimiento y aspecto son buenísimos.
- Pero si quieres hacer una aplicación Web que corra distribuida entre una máquinas pues casi que ha haría en Java. O si quisiera que mis aplicaciones tuvieran alta escalabilidad vertical ...
- ahora.... también te digo que a mis clientes (de grandes empresas) les sigo proponiendo que no tiren su código COBOL/NATURAL simplemente porque no está de moda y un iluminado les ha dicho que hay que rehacerlo todo en Java Me da que hacer lo mismo en otro lenguaje carece un pelín de sentido (siempre y cuando tengas en el mercado gente que lo soporte)

También me pregunto ¿hace cuantos años que tu editor de textos hacía mas cosas de las que tu usas?... y cada n meses sale otra versión (con más opciones a desconocer ...). Me da que con esto de los lenguajes nos está pasando lo mismo se da más importancia al lenguaje que a la técnica de construcción de software que se sigue realizando, en la mayoría de los sitios (al menos que yo conozca), de un modo artesanal. Os invito a haceros una preguntas:

- ¿Pintas (diagramas) lo que vas a hacer antes de hacerlo?
- ¿Hablas de patrones de software?
- ¿Preguntas a tus compañeros si han resuelto ese problema antes?
- ¿Tratas de resolver un problema ya conocido con otra técnica por la satisfacción del conocimiento?
- ¿Se comparte código en la organización? ¿Y conocimientos?
- ¿Has construido algún componente como tal y que se use después de algunos años?

- ¿Has introducido en tu ciclo de trabajo las pruebas unitarias? ¿Y el repositorio de código?
- ¿Sabes para que valen todas las opciones de los menús de tu entorno de trabajo? ¿Incluidas las de depuración?
- ¿Te has planteado construir extensiones a tus herramientas de desarrollo?
- ¿Cuántos desarrollos que has realizado todavía están en uso?

Si ya hacéis estas cosas, no se os olvide que tengo una empresa y busco gente www.autentia.com (no es broma y no os cortéis)

Bueno dejando el plano metafísico, lo mejor, cuando se va a probar una herramienta es plantearse hacer una aplicación real y ver lo que tardas ... pero ante de nada, como usaré C#, me he parado a repasar un poco las diferencias respecto a otros lenguajes que tengo más calientes como Java y C++.

Un consejo para navegantes en vez de tirarte n horas navegando por Internet buscando manuales suele ser mas barato (el tiempo vale dinero) comprarte un libro (y más barato si tu jefe es el que lo paga)

Un enlace:

<http://msdn.microsoft.com/library/spa/default.asp?url=/library/SPA/dncscol/html/csharp01182001.asp>

Para programadores Java (muy resumido)

Básicamente, las diferencias que me he apuntado:

- package por **namespace**, poniendo la clase entre llaves
- import por **using**
- Campos de solo lectura con **readonly**
- Paso de parámetros por referencia con palabra **ref** (inicializados) y **out**
- palabra **base** para hace referencia al constructor de la clase base
- final en clases por **sealed** (sellada)
- Número de parámetros variables utilizando la palabra reservada **params** y pasando un array. Aunque la notación de un array es como en C++, se llama a la clase System.Array
- El polimorfismo es a estilo C++ y requiere de la declaración del método en la clase base como **virtual** y en la sobrecargada **override (ojo con esto los programadores Java)**
- Cast de objetos como en Java y palabra clave **as** (no falla nunca porque devolvería null)
- Determinación del tipo original de un objeto con **is**
- Modificadores de miembros: public, protected, private e **internal** (algo como protected y private para la unidad de compilación actual)
- Existencia de métodos de acceso para acceder a propiedades sin conocer la existencia de getters y setters, con bloques **get** y **set** (con palabra clave **value** para parámetro pasado). Además se pueden crear atributos de solo lectura solamente ignorando el set.
- Existencia de enumeraciones
- La declaración de arrays multidimensionales se realiza utilizando comas: Ejemplo de dos dimensiones: array[,]. Tambien se pueden crear arrays irregulares como en Java.
- Para mostrar mensajes con parámetros tipo printf podemos usar **System.WriteLine("{0}", valor);**
- Para imprimir el número de dimensiones de un array usar la propiedad **Rank**
- Posibilidad de crear arrays inteligentes, o mejor dicho, acceder a un objeto a través de un índice como si fuera un array

```
public object this(int indice) {  
  
    get{ // operar con indice  
  
    }  
  
    set{ // operar con indice y value  
  
    }  
  
}
```

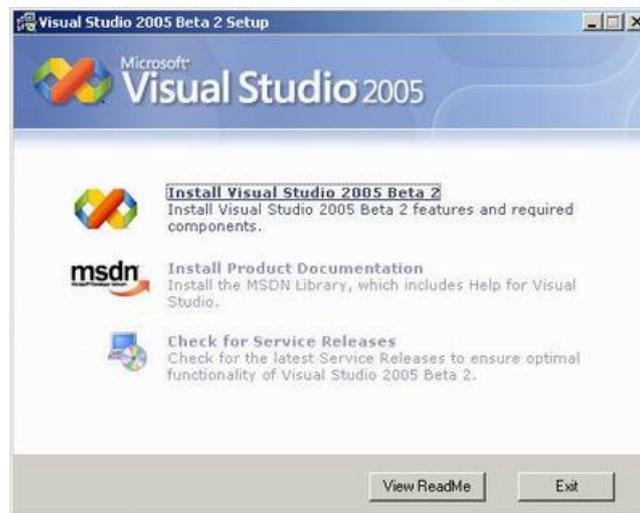
- Mecanismo de anotaciones (a atributos y métodos) a través de clases heredadas de **System.Attribute** y uso de corchetes.
- Para reflexión utilizaremos el operador **typeof** y la clase **System.Type**
- Se pueden construir conversiones definidas por usuario (cast) con **static implicit/explicit operator**
- Determinar el espacio que ocupa una variable con **sizeof** (al estilo c)
- Modificador **unsafe** para código tipo heredado (aritmética de punteros)
- Sentencia **foreach** para iterar entre colecciones
- Se incluye la sobrecarga de operadores, tan echados de menos en Java (al menos para mi). Aunque no se puede sobrecargar el operador **new** ni **=**.
- Existencia de delegados (similares a punteros a funciones)

Enlace de Interés

http://www.monohispano.org/varios/comp_csharp/index.html

Instalación de Visual Studio 2005 Beta 2

Buen, una vez solicitado (de modo gratuito) el paquete a Microsoft (4 DVDs), nos ponemos a instalar el entorno

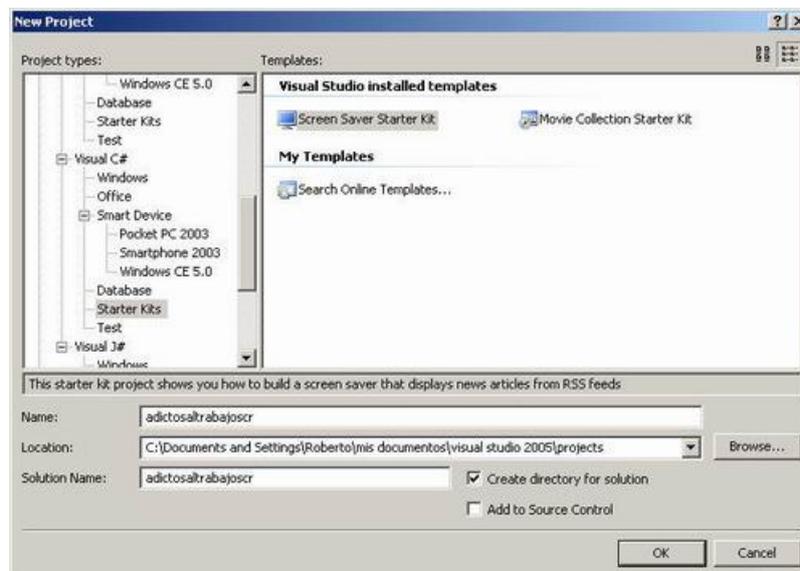


La verdad es que hasta el instalador esta muy cuidado



No me ha fallado nada y después de un rato ... ya podemos arrancar la aplicación.

Tiene un montón de asistentes que nos permiten hacer cosas más que interesantes desde el primer momento (os recomiendo que miréis el salva-pantallas RSS y las pocas líneas de código que tiene)



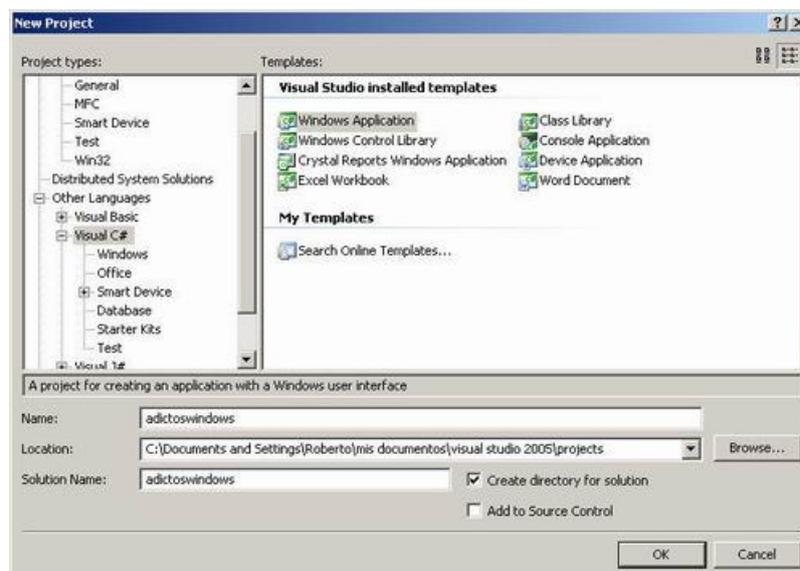
Nuestro proyecto

En la mayoría de los sistemas, la gente reniega de la pruebas unitarias porque dicen que lo que importa es el comportamiento de la aplicación cuando ésta está desplegada.

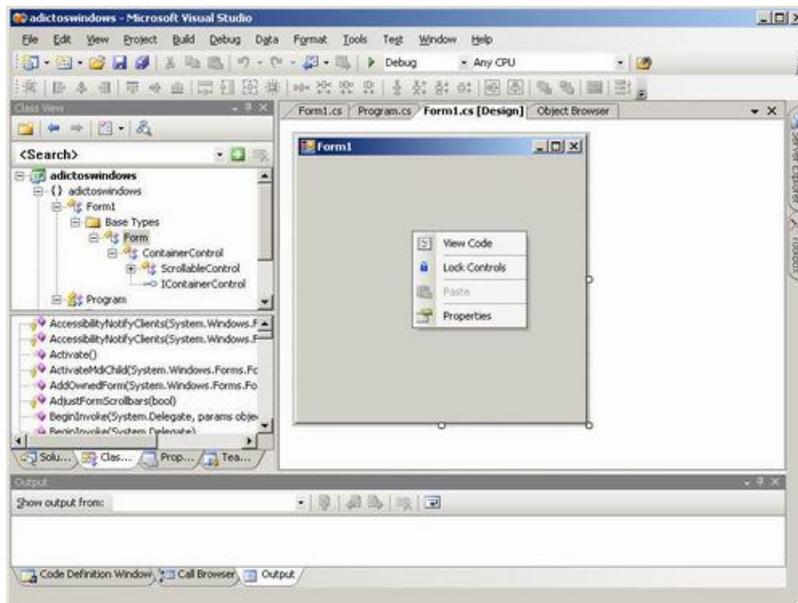
Bueno, pues vamos a construir una aplicación para automatizar las secuencias de navegación simulando el comportamiento real del usuario.

La gracia es, que el usuario navegue e ir grabando lo que hace (registrando información) y luego comparar las secuencias antiguas con las modernas (pruebas de regresión funcional, velocidad e incluso capacidad). *Nota: Ya hay un montón de software que hace esto pero me hace ilusión construir el mío propio.*

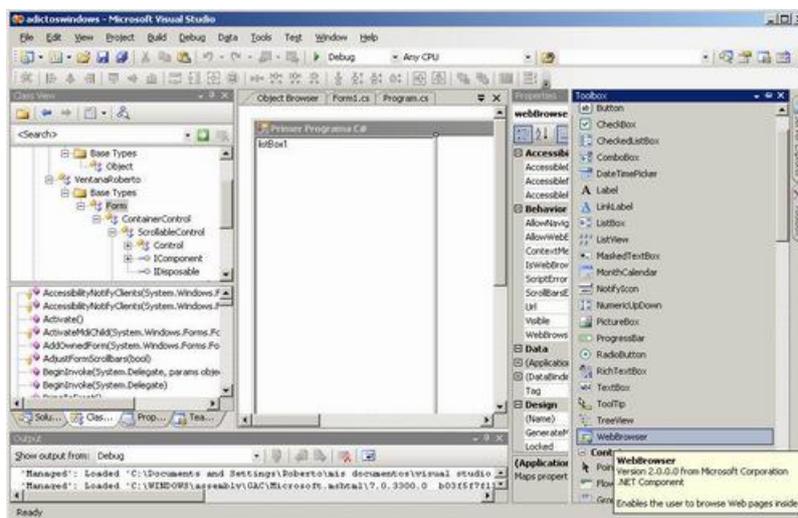
Voy a crear un proyecto con aplicación Windows típica



Nos aparece el editor de formularios

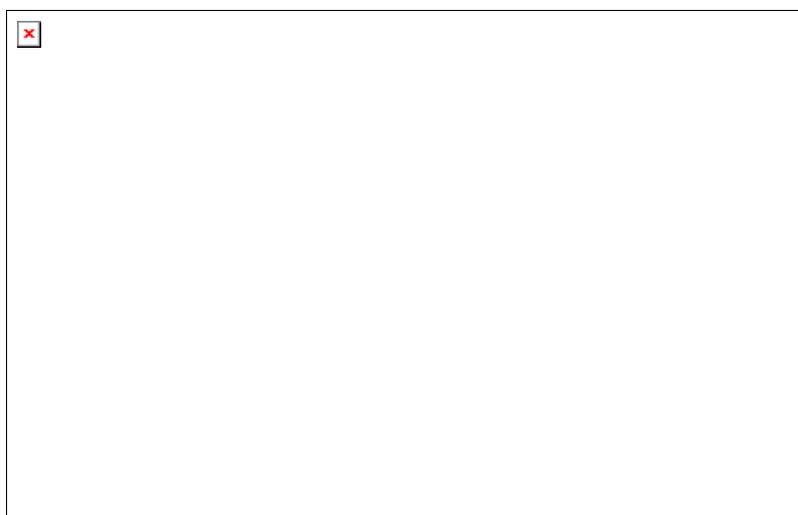


Y a la izquierda podemos ver los controles que podemos incorporar



Voy a contaros solo el primer paso (el tiempo es limitado y hay que dejar cosas para los cursos)

Elegimos un panel divisible y situamos un ListBox a la izquierda y un WebBrowser a la derecha, estableciendo como home www.adictosaltrabajo.com (uno 10 clicks)



Hacemos doble click en el navegador y en el ListBox y añadimos el código deseado:

- Cuando se cargue una página, guardamos su URL
- Cuando se pinche el ListBox, cogemos la URL y la cargamos en el navegador (anda, si hemos construido un navegador limitado con un history)

```
using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Drawing;
using System.Text;
using System.Windows.Forms;
using System.Collections;

// declaración del espacio de nombres
namespace adictoswindows
{
    // declaración parcial de la clase (concepto interesante)
    public partial class VentanaRoberto : Form
    {
        public VentanaRoberto()
        {
            InitializeComponent();
        }

        // funcion asistente para no meter depuraciones descontroladas
        void depura(String cadena)
        {
            System.Console.WriteLine(cadena);
        }

        // respondemos al evento de carga completa de documento
        private void webBrowser1_DocumentCompleted(object sender, WebBrowserDocumentCompletedEventArgs e)
        {
            // recogemos el objeto que envia el evento
            WebBrowser x = (WebBrowser)sender;

            // actualizamos la lista de la izquierda
            listBox1.BeginUpdate();
            listBox1.Items.Add(x.Url);
            listBox1.EndUpdate();

            // depuramos para asegurarnos que la cosa va bien
            depura("Escribimos: " + x.Url);
        }

        // cuando el usuario selecciona una URL de la lista, la mostramos en el navegador
        private void listBox1_SelectedIndexChanged(object sender, EventArgs e)
        {
            string cadena = listBox1.SelectedItem.ToString();
            webBrowser1.Url = new Uri(cadena);
        }
    }
}
```



Conclusiones

La verdad es que es impresionante ... rápido, elegante y potente

La única lástima es que nos tenemos que aprender otra inmensa librería de clases para ser productivos en este entorno aunque si tienes la dinámica Java y has programado un poco en Windows las cosas aparecen delante de tus ojos.

- ¿Os parece C# un Java++-++-++ ? Pues entonces solo tenéis que mirar el J# ... (en otro tutorial)

De todos modos, supongo que pasará como siempre.... nadie se comerá a nadie y coexistirán durante unos añitos, cada uno con su cuota de mercado.

Si desea contratar formación, consultoría o desarrollo de piezas a medida puede contactar con

Gestión de contenidos

[Autentia S.L.](#) Somos expertos en:
J2EE, C++, OOP, UML, Vignette, Creatividad ..
 y muchas otras cosas

Nuevo servicio de notificaciones

Si deseas que te enviemos un correo electrónico cuando introduzcamos nuevos tutoriales, inserta tu dirección de correo en el siguiente formulario.

Subscribirse a Novedades	
e-mail	<input type="text"/>
	<input type="text"/>

Otros Tutoriales Recomendados ([También ver todos](#))

Nombre Corto

[Invocación a un componente COM desde Java](#)

[Introducción a Dev-C++](#)

[Web Services en tu IPAQ](#)

[Almacenamiento en Windows Pocket 2003](#)

[Patrones de diseño J2EE](#)

[Novedades en Java 1.5](#)

[Gráficas en Java con JFreeChart](#)

[Ado y Net](#)

[Test con JUnit](#)

[Construir un Servidor Web en Java](#)

Descripción

Os mostramos como invocar a un servidor de automatización OLE (COM) desde una aplicación Java, a través del producto gratuito JACOB

Os mostramos el entorno de desarrollo Dev-C++, una buena opción a la hora de desarrollar en C/C++ utilizando un entorno gratuito, siempre mejor que hacerlo a mano.

Cesar Crespo nos enseña como programar accesos Web Services desde tu IPAQ en Visual C++ con PocketSOAP, Apache SOAP y Axis

Cesar Crespo nos enseña como utilizar ObjectStore en nuestros programas Visual C++. El "Object Store" (ObS) en Windows Pocket 2003 cumple en muchos casos la misma función que el disco duro en un equipo de escritorio.

Os mostramos una interpretación particular de los patrones de diseño J2EE

Ya está disponible la versión Beta del J2SDK 1.5. Os mostramos algunas de las nuevas características introducidas en el lenguaje Java: Clases genéricas, enumeraciones, bucles simplificados, etc.

Os mostramos como generar gráficas profesionales, en aplicaciones y servlets, en Java con la librería gratuita JFreeChart

En este tutorial podeis descubrir como funciona ADO y las nuevas diferencias en arquitectura .Net .

Cuando se hacen desarrollo profesionales, no basta con hacer los programas, hay que asegurarse de que van a funcionar. Una de las técnicas más seguras es crear aplicaciones que incluyan el código para autoprobarse. Os mostramos como usar JUnit

En este tutorial os enseñamos los principios de las aplicaciones multi-hilo a través de la creación de un servidor web básico en Java. Podremos ver en un ejemplo real el uso de sockets, threads, excepciones, etc.

Nota: Los tutoriales mostrados en este Web tienen como objetivo la difusión del conocimiento.

Los contenidos y comentarios de los tutoriales son responsabilidad de sus respectivos autores.

En algún caso se puede hacer referencia a marcas o nombres cuya propiedad y derechos es de sus respectivos dueños. Si algún afectado desea que incorporemos alguna reseña específica, no tiene más que solicitarlo.

Si alguien encuentra algún problema con la información publicada en este Web, rogamos que informe al administrador rcanales@adictosaltrabajo.com para su resolución.

[Patrocinados por enredados.com Hosting en Castellano con soporte Java/J2EE](#)

