



Investigación

Instituto tecnológico del sur de Nayarit

Nombre del alumno: Sergio verdín flores

Nombre del docente: Cinthia Anahí mata bravo

Materia: Programación orientada a objetos

Fecha: 29/01/2020

Semestre: 2



Índice

Aplicación consola c#: -----

pag.3

Consola Writline y Readline: -----

pag.4

Windows forms: -----

pag.5

Implementar aplicaciones en equipos: -----

Pag.6

Conclusión: -----

Pag.7

Aplicaciones de Consola en C#

C# es un lenguaje de programación desarrollado por Microsoft, orientado a objetos, que ha sido diseñado para compilar diversas aplicaciones que se ejecutan en .NET Framework. Se trata de un lenguaje simple, eficaz y con seguridad de tipos. Las numerosas innovaciones de C# permiten desarrollar aplicaciones rápidamente y mantener la expresividad y elegancia de los lenguajes de estilo de C.

En Visual Studio .Net siempre se han desarrollado aplicaciones de consola con la finalidad de poder iniciarnos y familiarizarnos con las sentencias y estructuras de la programación .Net

El principal motivo de este post es que la mayor parte de los que leen este blog no saben cómo interactuar con el usuario a través de la consola (lo cual es comprensible), y no tienen una forma dinámica de probar sus códigos.

Generalmente cuando nos enseñan a programar, parten de la parte de cómo crear métodos, la sintaxis del lenguaje, las estructuras de control (for, foreach, etc) y se pasan por alto la parte de cómo crear una aplicación, donde el usuario vea resultados a través de la consola. Bueno, empecemos desde 0:

Crear una aplicación de Consola:

Para esto, deberemos ir a File->New->Project si usas Visual Studio, pero con cualquier otro IDE debe ser algo parecido. Entonces se mostrará una nueva ventana como la de la figura, seleccionamos C# en el menu izquierdo y luego como tipo de proyecto marcamos "Console Application", le ponemos un nombre al proyecto y le damos Ok. Ya estaremos listos para empezar con el código.

El principal motivo de este post es que la mayor parte de los que leen este blog no saben cómo interactuar con el usuario a través de la consola (lo cual es comprensible), y no tienen una forma dinámica de probar sus códigos.

Generalmente cuando nos enseñan a programar, parten de la parte de cómo crear métodos, la sintaxis del lenguaje, las estructuras de control (for, foreach, etc) y se pasan por alto la parte de cómo crear una aplicación, donde el usuario vea resultados a través de la consola. Bueno, empecemos desde 0.

Para interactuar con el usuario es que existen algunos métodos básicos cuando trabajamos con la Consola. Veamos los más usados:

1- Console.WriteLine();

Escribe una nueva línea en la consola, es como presionar Enter y escribir una línea. Por ejemplo, podemos hacer:

```
Console.WriteLine("Lo que quiera q salga en la consola");
```

También podremos imprimir las variables de nuestro programa, para esto hay dos formas, pero veamos mejor dos ejemplos que son lo mismo:

Como ven la segunda vía es mucho más simple, solo debes recordar que el número que está entre corchetes es el índice del parámetro que se pasa al final del método.

2- Console.Write()

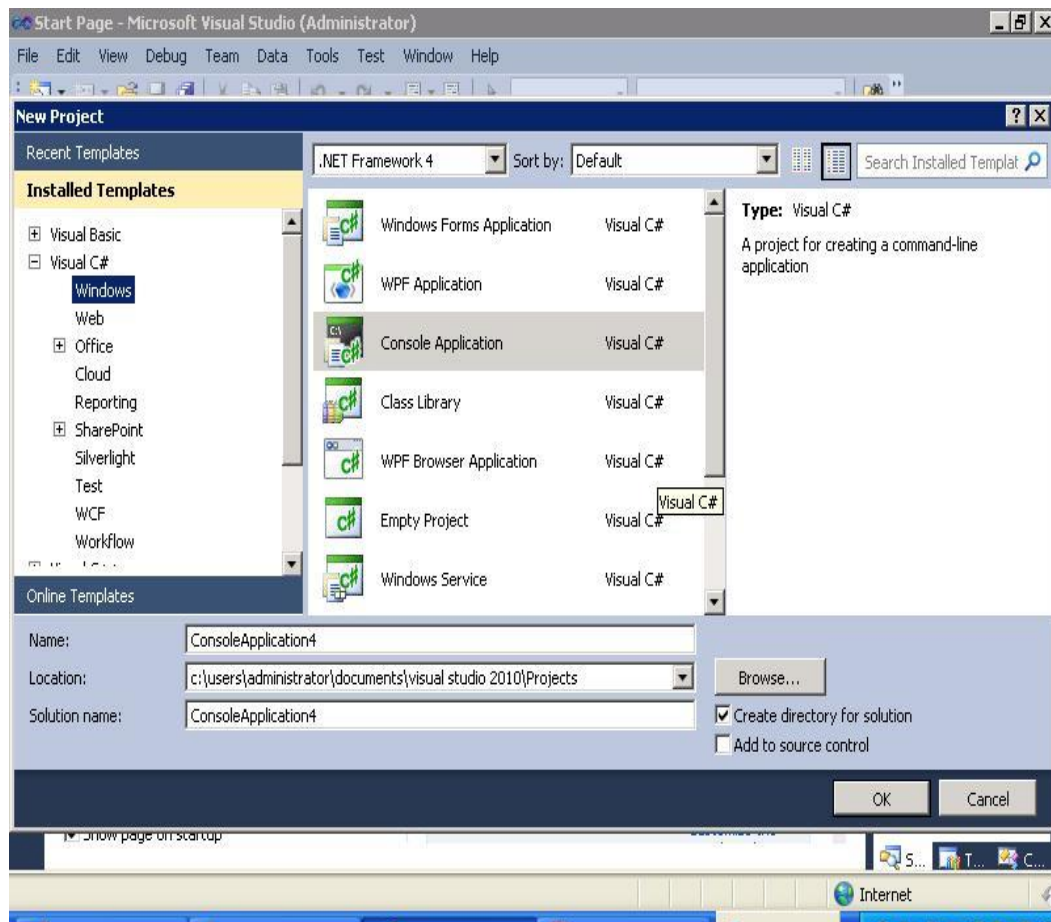
Hace casi lo mismo que Console.WriteLine() pero sin cambiar de línea, o sea... concatena el texto que le pasemos al método con lo que se haya escrito hasta ese momento en la consola.

3- Console.ReadLine()

Se usa para leer lo que el usuario escribió en la consola, y podemos guardarlo en una variable para después usarlo en nuestro programa. No hay que pasarle ningún parámetro y el programa solo continuará cuando el usuario teclee alguna línea y presione Enter.

Lo más que contiene es lo mas relevante al proyecto, pero en código cuando lo que se programó.

En si más que nada se definir una aplicación de consola como aquella que se ejecuta en una ventana ms- dos es decir en linea de comandos.



Windows forms

Aplicaciones de Windows Forms y Smart Client

Como los formularios son la unidad básica de la aplicación, es esencial realizar algunas consideraciones sobre su función y diseño. En última instancia, un formulario es una pizarra en blanco que usted, como desarrollador, mejora con controles para crear una interfaz de usuario y con código para manipular los datos. Para ello, Visual Studio proporciona un entorno de desarrollo integrado (IDE) que ayuda a escribir código, así como un conjunto de controles enriquecidos escrito con el .NET Framework. Complementando la funcionalidad de estos controles con el código, puede desarrollar las soluciones que necesita de forma fácil y rápida.

Con Windows Forms, puede desarrollar aplicaciones smart client. Las aplicaciones de *cliente inteligente* son aplicaciones gráficamente enriquecidas, fáciles de implementar y actualizar, que pueden funcionar con o sin conexión a Internet y que pueden acceder a los recursos del equipo local de un modo más seguro que las aplicaciones tradicionales basadas en Windows.

Crea interfaces de usuario completas e interactivas

Windows Forms es una tecnología de Smart Client para el .NET Framework, un conjunto de bibliotecas administradas que simplifican las tareas comunes de la aplicación, como la lectura y la escritura en el sistema de archivos. Cuando se usa un entorno de desarrollo como Visual Studio, puede crear Windows Forms aplicaciones de cliente inteligente que muestran información, solicitan a los usuarios la entrada de datos y se comunican con equipos remotos a través de una red.

En Windows Forms, un *formulario* es una superficie visual en la que se muestra información al usuario. Normalmente, las aplicaciones de Windows Forms se compilan mediante la adición de controles a los formularios y el desarrollo de respuestas a las acciones del usuario, como clics del mouse o presiones de teclas. Un *control* es un elemento de interfaz de usuario (UI) discreto que muestra datos o acepta la entrada de datos.

Windows Forms contiene diversos controles que puede agregar a los formularios: controles que muestran cuadros de texto, botones, cuadros desplegables, botones de radio e incluso páginas web. Para obtener una lista de todos los controles que puede usar en un formulario, consulte el artículo sobre. se utiliza en formularios Windows forms. Si un control existente no satisface sus necesidades, Windows Forms también permite crear controles personalizados mediante la clase Use control.

Windows Forms tiene controles de interfaz de usuario enriquecidos que emulan las características de las aplicaciones de tecnología avanzada como Microsoft Office. Los controles toolstrip y menustrip le permiten crear barras de herramientas

y menús que contienen texto e imágenes, muestran submenús y hospedan otros controles como cuadros de texto y cuadros combinados.

Con la **Diseñador de Windows Forms** de arrastrar y colocar en Visual Studio, puede crear fácilmente aplicaciones de Windows Forms. Simplemente seleccione los controles con el cursor y agréguelos donde desee en el formulario. El diseñador proporciona herramientas como líneas de cuadrícula y líneas de ajuste para minimizar la molestia de alinear los controles.

Mostrar y manipular datos

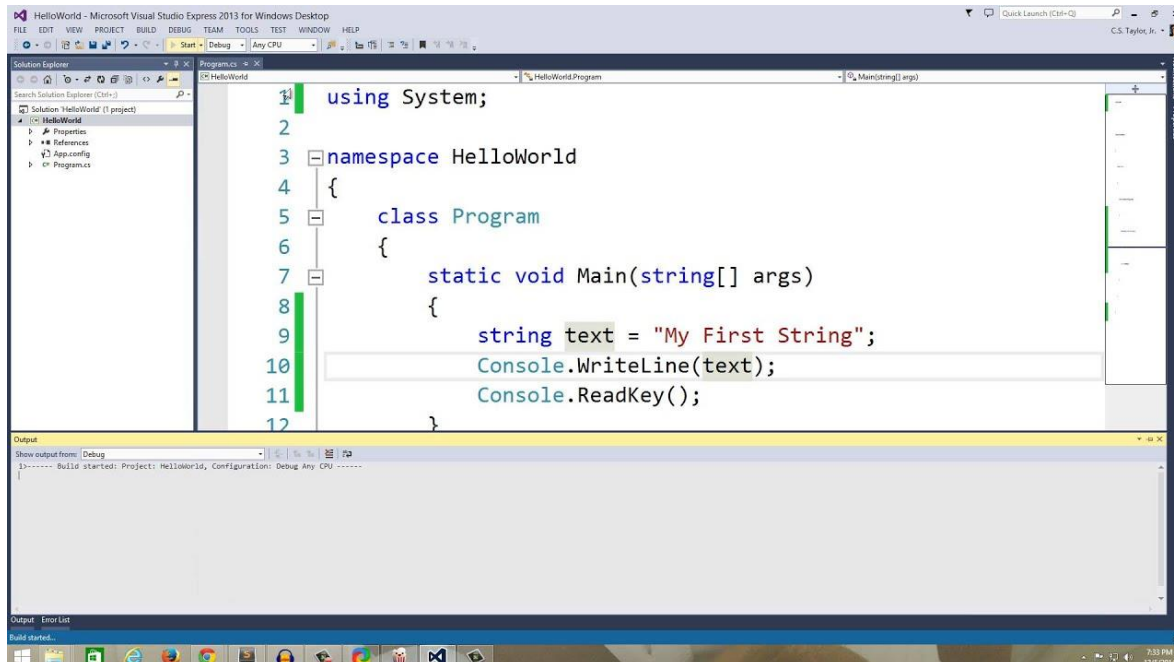
Muchas aplicaciones deben mostrar datos procedentes de una base de datos, archivo XML, servicio web XML u otro origen de datos. Windows Forms proporciona un control flexible denominado control para mostrar esa información tabulada en un formato tradicional de filas y columnas, de modo que cada dato ocupe su propia celda. Al usar, puede personalizar la apariencia de celdas individuales, bloquear en su posición filas y columnas arbitrarias y mostrar controles complejos dentro de las celdas, entre otras características. Puede crear fácilmente controles enlazados a datos en la ventana Orígenes de datos. La ventana muestra los orígenes de datos como bases de datos, servicios web y objetos en el proyecto. Para crear controles enlazados a datos, arrastre los elementos desde esta ventana hasta los formularios de su proyecto. También puede enlazar controles existentes a datos si arrastra los objetos desde la ventana Orígenes de datos a los controles existentes.

Implementar aplicaciones en equipos

Una vez escrita la aplicación, hay que enviarla a los usuarios para que puedan instalarla y ejecutarla en sus equipos cliente. Al usar la tecnología ClickOnce, puede implementar sus aplicaciones desde Visual Studio con tan solo unos clics y proporcionar a los usuarios una dirección URL que apunte a la aplicación en la Web. ClickOnce administra todos los elementos y dependencias de la aplicación y garantiza que la aplicación se instala correctamente en el equipo cliente.

Las aplicaciones ClickOnce se pueden configurar para que se ejecuten solo cuando el usuario está conectado a la red o para ejecutarse tanto en línea como sin conexión. Cuando se especifica que una aplicación debe admitir el funcionamiento sin conexión, ClickOnce agrega un vínculo a la aplicación en el menú **Inicio** del usuario. El usuario puede entonces abrir la aplicación sin usar la dirección URL.

Hay muchas otras características en Windows Forms que simplifican y agilizan las tareas comunes de implementación, como la posibilidad de crear cuadros de diálogo, imprimir, agregar ayuda y documentación, y localizar la aplicación a varios idiomas. Además, Windows Forms se basa en el sólido sistema de seguridad de la .NET Framework. Con este sistema, puede publicar aplicaciones más seguras para sus clientes.



conclusión:

en este trabajo me sirvió porque te da información sobre que mas adelante vamos a hacer en programar en los programas nuevos en si es una ayuda en nos podemos enfocar para programar en futuras materias relacionadas con la carrera que estoy cursando. Me acuerdo de que la primera vez que empecé en la programación (Visual Basic) empezábamos a programar aplicaciones sencillas en consola tipo ventanas de dos. Era la manera de iniciar al principiante como alumno a poder desarrollar aplicaciones más robustas.

Siempre vi esta manera de aprender como la mejor forma de iniciarnos en este mundo de la programación sin embargo nunca me puse a pensar detalladamente lo importante y conveniente que era desarrollar estas pequeñas aplicaciones de consola. Siempre vi esta manera de aprender como la mejor forma de iniciarnos en este mundo de la programación me puse a pensar detalladamente lo importante y conveniente que era desarrollar estas pequeñas aplicaciones de consola.