

INSTITUTO TECNOLÓGICO DEL SUR DE NAYARIT



Interfaces C#

Materia:

Programación Orientada a Objetos

Docente

Cinthia Anahi Mata Bravo

Alumno:

Jesús Ernesto Benítez González

No. de control:

191140015

Índice

Interfaces.....	3
Resumen.....	4

Interfaces

La interfaz de una clase es todo lo que podemos hacer con ella. A efectos prácticos: todos los métodos, propiedades y variables públicas (aunque no debería haber nunca variables públicas, debemos usar propiedades en su lugar) de la clase conforman su interfaz.

Una interfaz es como un contrato. En el mundo humano, el contrato entre los dos o más humanos los obliga a actuar según el contrato. Del mismo modo, la interfaz incluye la declaración de una o más funcionalidades. Las entidades que implementan la interfaz deben definir funcionalidades declaradas en la interfaz. En C #, una clase o una estructura pueden implementar una o más interfaces.

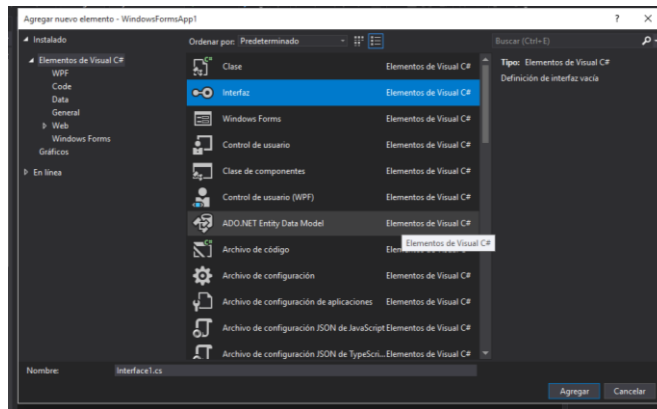
En C #, se puede definir una interfaz utilizando la palabra clave de interfaz. Las interfaces pueden contener métodos, propiedades, indexadores y eventos como miembros.

Una interfaz contiene definiciones para un grupo de funcionalidades relacionadas que una clase no abstracta o una estructura deben implementar. Una interfaz puede definir staticmétodos, que deben tener una implementación. Una interfaz puede proporcionar una implementación predeterminada para cualquiera o todos sus miembros de instancia declarados. Una interfaz no puede declarar datos de instancia como campos, propiedades implementadas automáticamente o eventos similares a propiedades.

Mediante el uso de interfaces, puede, por ejemplo, incluir el comportamiento de múltiples fuentes en una clase. Esa capacidad es importante en C # porque el lenguaje no admite la herencia múltiple de clases. Además, debe usar una interfaz si desea simular la herencia de estructuras, porque en realidad no pueden heredar de otra estructura o clase.

Resumen

La interfase es un contrato el cual contiene un grupo de funcionalidades que están relacionadas a una clase no abstracta o una estructura. La interface no puede declarar datos de instancia como campos, propiedades implementadas automáticamente o eventos similares a propiedades.



Son mecanismos que pueden interactuar varios objetos que no están relacionados entre sí. Contiene las declaraciones de los métodos, pero no su implementación.

Al igual que las clases abstractas, son plantillas de comportamiento que deben ser implementados por otras clases.

Los interfaces a diferencia de las clases abstractas, estas cambian en los modificadores de acceso ya que en la interface automáticamente todos sus miembros son públicos.

Las clases que hereden de una interfase debe de implementar todas las definiciones que están contenidas en la interfase. Todos los elementos declarados dentro de una interface se consideran públicos y abstractos.

Descripción: Volúmen

Realizar un programa que calcule el volumen de un cubo, un cilindro y una pirámide cuadrangular.

El programa debe contener un interfase de padre llamada Volumen el cual tenga un solo método de CalcularVolumen.

Cada una de las figuras debe tener su propia clase y debe ser heredada de la clase padre.

La clase Cubo además del método heredado se debe añadir las variables que necesiten(double).

La clase Cilindro aparte del método debemos añadir las variables que necesitemos para llegar al resultado final(double).

Y por último una clase de Pirámide además del método que es heredado, añadir las variables que sean necesarias(double)

Elaborar el proyecto WindowsForms que permita recibir la información y calcular el volumen de cada figura.