**CÓMO: Definir y utilizar propiedades en Visual Basic .NET**

Id. de artículo: 308230 - [Ver los productos a los que se aplica este artículo](https://support.microsoft.com/es-es/kb/308230/es#appliesto)

Este artículo se publicó anteriormente con el número E308230

Expandir todo | Contraer todo

## En esta página

* Resumen
  + Requisitos
  + Crear una aplicación que contiene una clase de Visual Basic .NET
  + Definir campos, constructores y métodos
  + Definir propiedades en la clase
  + Crear un objeto y utilizar las propiedades
  + Comprobar que funciona
* Propiedades
* Enviar comentarios

## Resumen

Un objetivo importante de muchos lenguajes de programación orientados a objetos es permitirle encapsular los detalles internos en una clase. Visual Basic .NET le permite definir propiedades para proporcionar acceso controlado a los detalles internos de una clase. En este artículo se describe cómo definir y utilizar propiedades en una aplicación de Visual Basic .NET.   
  
Una propiedad puede ser de sólo lectura, de sólo escritura o de lectura y escritura. Los descriptores de acceso **Get** y **Set** proporcionan acceso de lectura y de escritura, respectivamente. Puede agregar código a estos descriptores de acceso para proporcionar la funcionalidad adecuada para obtener o configurar la propiedad.   
  
En este artículo creará una clase llamada **Employee**. Dentro de esta clase definirá una propiedad de lectura y escritura para el nombre del empleado, una propiedad para leer el sueldo del empleado y otra propiedad para calcular cuánto tiempo lleva trabajando el empleado en la compañía. Después creará un objeto **Employee** en la aplicación y utilizará sus propiedades.

### Requisitos

En el presente artículo se da por supuesto que está familiarizado con los temas siguientes:

* Sintaxis de Visual Basic .NET
* Conceptos orientados a objetos
* Microsoft Visual Studio .NET

### Crear una aplicación que contiene una clase de Visual Basic .NET

1. Inicie Visual Studio .NET y cree un nuevo proyecto de aplicación de consola de Visual Basic .NET llamado EmployeeProperties.
2. En el menú **Proyecto**, haga clic en **Agregar clase**. En el cuadro de diálogo **Agregar nuevo elemento**, escriba Employee.vb en el cuadro **Nombre** y, a continuación, haga clic en **Abrir**.
3. Examine el código de la nueva clase en la ventana Ver código.

### Definir campos, constructores y métodos

1. Agregue tres campos **Private** a la clase **Employee** de la manera siguiente:

Private Name As String

Private Salary As Decimal

Private HireDate As DateTime

1. Defina un constructor para la clase con el fin de inicializar los campos de la clase **Employee**:

Public Sub New(ByVal Name As String, \_

ByVal Salary As Decimal, \_

ByVal HireDate As DateTime)

Me.Name = Name

Me.Salary = Salary

Me.HireDate = HireDate

End Sub

1. Agregue un método **Public** a la clase:

Public Sub PayRise(ByVal Amount As Decimal)

Me.Salary += Amount

End Sub

### Definir propiedades en la clase

1. Agregue el código siguiente para agregar una propiedad de lectura y escritura llamada **FullName**. El descriptor de acceso **Get** devuelve el nombre del empleado. El descriptor de acceso **Set** utiliza el parámetro **Value** que se pasa en el descriptor de acceso **Set** para establecer el nombre del empleado.

Public Property FullName() As String

Get

Return Me.Name

End Get

Set(ByVal Value As String)

Me.Name = Value

End Set

End Property

1. Agregue el código siguiente para agregar una propiedad **ReadOnly** llamada **Remuneration**. El descriptor de acceso **Get** devuelve el sueldo del empleado. No hay ningún descriptor de acceso **Set** porque el sueldo no se puede establecer directamente; la única forma de cambiar el sueldo es llamar al método **PayRise**.

Public ReadOnly Property Remuneration() As Decimal

Get

Return Me.Salary

End Get

End Property

1. Agregue el código siguiente para agregar otra propiedad **ReadOnly** llamada **DaysSinceJoined**. El descriptor de acceso **Get** calcula cuántos días ha trabajado el empleado para la compañía.

Public ReadOnly Property DaysSinceJoined() As Integer

Get

Dim timespan As TimeSpan = DateTime.Now.Subtract(Me.HireDate)

return timespan.Days

End Get

End Property

### Crear un objeto y utilizar las propiedades

1. Muestre el código de Module1.vb en la ventana Ver código.
2. En el método **Main**, utilice el operador **New** para crear un objeto **Employee**. Asigne la referencia de objeto a una variable local **Employee** del siguiente modo:
3. Dim worker As New Employee( "Jayne Doe", \_
4. 125000, \_

New DateTime(1998, 7, 31) )

1. Agregue el código siguiente para cambiar la propiedad **FullName** del empleado. Esto invocará implícitamente el descriptor de acceso **Set** para la propiedad **FullName**.

worker.FullName = "Jayne Lafayette"

1. Agregue el código siguiente para mostrar los detalles del empleado mediante los descriptores de acceso **Get** para las propiedades **FullName**, **Remuneration** y **DaysSinceJoined**:
2. Console.WriteLine("Nombre completo: " & worker.FullName)
3. Console.WriteLine("Remuneración: " & worker.Remuneration)

Console.WriteLine("Días trabajados: " & worker.DaysSinceJoined)

### Comprobar que funciona

1. Genere y ejecute la aplicación.
2. La aplicación mostrará la siguiente información en la consola (el valor "Días trabajados" depende de la fecha actual):
3. Nombre completo: Jayne Lafayette
4. Remuneración: 125000

Días trabajados: 1139

1. Vuelva al archivo Module1.vb en la ventana Ver código y agregue el código siguiente al final del método **Main**:

worker.Remuneration = 1000000

1. Pruebe a generar la aplicación. Aparecerá un error de compilación porque **Remuneration** es una propiedad de sólo lectura.
2. Cambie la instrucción de la manera siguiente:

worker.Salary = 1000000

1. Pruebe a generar de nuevo la aplicación. Aparecerá otro error de compilación porque el campo **Salary** está marcado como **Private**. La única forma de modificar el sueldo del empleado es utilizar el método público **PayRise**.

[[Volver arriba](https://support.microsoft.com/es-es/kb/308230/es#top)Volver arriba](https://support.microsoft.com/es-es/kb/308230/es#top) | [Enviar comentarios](https://support.microsoft.com/es-es/kb/308230/es#survey)

## Collapse image[Propiedades](javascript:void(0);)

Id. de artículo: 308230 - Última revisión: martes, 15 de enero de 2002 - Versión: 1.0

##### La información de este artículo se refiere a:

* Microsoft Visual Basic .NET 2002 Standard Edition

|  |  |
| --- | --- |
| Palabras clave: | kbhowto kbhowtomaster KB308230 |

[[Volver arriba](https://support.microsoft.com/es-es/kb/308230/es#top)Volver arriba](https://support.microsoft.com/es-es/kb/308230/es#top) | [Enviar comentarios](https://support.microsoft.com/es-es/kb/308230/es#survey)

## Collapse image[Enviar comentarios](javascript:void(0);)

¿Fue útil esta información?