



Clase presencial



Sesion4: Trabajo con cadenas (String)

Instructor: David Paúl Porras Córdova

@iscodem



Objetivo General

- Identificar las características del tipo de dato String.
- Trabajar con funciones de cadenas en Visual Studio.
- Identificar el ciclo de vida de un String.



Variables de cadena

A una instancia de una cadena se le puede asignar un valor literal que represente una serie de caracteres.

```
Dim MyString As String  
MyString = "This is an example of the String data type"
```



String - cadenas

Una variable String también puede aceptar cualquier expresión que se evalúe como una cadena.

```
Dim OneString As String
Dim TwoString As String
OneString = "one, two, three, four, five"
' Evaluates to "two".
TwoString = OneString.Substring(5, 3)
OneString = "1"
' Evaluates to "11".
TwoString = OneString & "1"
```



String - cadenas

Cualquier literal que se asigna a una variable String debe ir entre comillas ("""). Esto significa que una comilla contenida en una cadena no se puede representar mediante una comilla. Por ejemplo, el código siguiente causa un error del compilador:

```
Dim myString As String
```

```
' This line would cause an error.
```

```
' myString = "He said, "Look at this example!"""
```

```
' The value of myString is: He said, "Look at this example!"
```

```
myString = "He said, ""Look at this example!"""
```



String

Los literales de cadena pueden contener varias líneas:

```
Dim x = "hello  
world"
```

La cadena resultante contiene secuencias de nueva línea que usó en el literal de cadena (vbcr, vbcrLf, etc.). La solución anterior ya no es necesaria:

```
Dim x = <xml><![CDATA[Hello  
World]]></xml>.Value
```



Caracteres en cadenas

Una cadena puede considerarse como una serie de valores **Char** y el tipo **String** tiene funciones integradas que permiten realizar muchas manipulaciones en una cadena similares a las permitidas por las matrices. Como toda la matriz en .NET Framework, son matrices de base cero.

Puede hacer referencia a un carácter específico de una cadena con la propiedad **Chars**, que proporciona una forma de acceso a un carácter mediante la posición en la que este aparece en la cadena. Por ejemplo:

```
Dim myString As String = "ABCDE"  
Dim myChar As Char  
' The value of myChar is "D".  
myChar = myString.Chars(3)
```



Tipos de datos

También puede obtener la longitud de una cadena concreta mediante la propiedad Length.

```
Dim myString As String = "abcdefghijklmnop"  
Dim myArray As Char() = myString.ToCharArray
```




Inmutabilidad de cadenas

Una cadena es inmutable, lo que significa que su valor no puede modificarse una vez que se ha creado. Sin embargo, esto no impide asignar más de un valor a una variable de cadena. Considere el ejemplo siguiente:

```
Dim myString As String = "This string is immutable"  
myString = "Or is it?"
```

Aquí se crea una variable de cadena con un valor y, a continuación, se cambia su valor.



Manipulación de cadenas

Más concretamente, en la primera línea, se crea una instancia de tipo String y se le asigna el valor ***This string is immutable***. En la segunda línea del ejemplo se crea una nueva instancia y se le asigna el valor ***Or is it?***, y la variable de cadena descarta su referencia a la primera instancia y almacena una referencia a la nueva instancia.



Tipos de métodos de manipulación de cadenas en Visual Basic

Hay varias maneras de analizar y manipular las cadenas. Algunos de los métodos forman parte de la Visual Basic idioma, mientras que otras son inherentes a la String clase.

```
Dim aString As String = "SomeString"  
Dim bString As String  
' Assign "meS" to bString.  
bString = Mid(aString, 3, 3)
```

En este ejemplo, el **Mid** función realiza una operación directa en “*aString*” y asigna el valor a “*bString*”.



Resumen de manipulaciones de cadenas (Visual Basic)

Acción	Elemento del lenguaje
Comparar dos cadenas.	StrComp
Convertir cadenas.	StrConv
Invertir una cadena.	InStrRev, StrReverse
Convertir en mayúsculas o minúsculas.	Format, LCase, UCase
Crear una cadena de caracteres repetidos.	Space, StrDup



Resumen de manipulaciones de cadenas (Visual Basic)

Acción	Elemento del lenguaje
Comparar dos cadenas.	StrComp
Trabajar con valores ASCII y ANSI.	Asc , AscW , Chr , ChrW
Reemplazar una subcadena especificada.	Replace
Devolver una matriz de cadenas basada en un filtro.	Filter
Devolver un número de subcadenas especificado.	Split , Join



Métodos compartidos

Un método compartido es un método que proviene de la String propia clase y no requiere una instancia de dicha clase para funcionar. Estos métodos se pueden calificar con el nombre de la clase (String) en lugar de con una instancia de la String clase. Por ejemplo:

```
Dim aString As String = String.Copy("A literal string")
```



Métodos de instancia

Métodos de instancia, por el contrario, se derivan de una instancia determinada de String y deben estar calificados con el nombre de instancia. Por ejemplo:

```
Dim aString As String = "A String"  
Dim bString As String  
' Assign "String" to bString.  
bString = aString.Substring(2, 6)
```



Nothing y cadenas en Visual Basic

El Visual Basic en tiempo de ejecución y el .NET Framework evaluar Nothing forma diferente cuando se trata de cadenas.

```
Dim MyString As String = "This is my string"
Dim stringLength As Integer
' Explicitly set the string to Nothing.
MyString = Nothing
' stringLength = 0
stringLength = Len(MyString)
' This line, however, causes an exception to be thrown.
stringLength = MyString.Length
```

El Visual Basic normalmente se evalúa como el tiempo de ejecución Nothing como una cadena vacía ("").



Cómo: Crear cadenas mediante un objeto **StringBuilder** en Visual Basic

Este ejemplo crea una cadena larga de muchas cadenas más pequeñas mediante la **StringBuilder** clase. El **StringBuilder** clase es más eficaz que la **&=** operador para concatenar muchas cadenas.

Ejemplo:

En el ejemplo siguiente se crea una instancia de la **StringBuilder** (clase), anexan 1.000 cadenas a esa instancia y, a continuación, devuelve la representación de cadena.

```
Private Function StringBuilderTest() As String
    Dim builder As New System.Text.StringBuilder
    For i As Integer = 1 To 1000
        builder.Append("Step " & i & vbCrLf)
    Next
    Return builder.ToString
End Function
```



Cómo: Buscar en una cadena (Visual Basic)

Este ejemplo llama a la **IndexOf** método en un **String** objeto para informar del índice de la primera aparición de una subcadena.

```
Dim SearchWithinThis As String = "ABCDEFGHijklmnop"  
Dim SearchForThis As String = "DEF"  
Dim FirstCharacter As Integer = SearchWithinThis.IndexOf(SearchForThis)
```

El [IndexOf](#) método notifica la ubicación del primer carácter de la primera aparición de la subcadena. El índice está basado en 0, lo que significa que el primer carácter de una cadena tiene un índice de 0.

Si [IndexOf](#) no encuentra la subcadena, devuelve -1.

El [IndexOf](#) método distingue entre mayúsculas y minúsculas y utiliza la referencia cultural actual.



Funciones propias de Visual Basic

Función Lcase y Ucase

Lcase (cadena que queremos convertir a minúsculas)

Ejemplo:

```
Dim cadena As String  
  
cadena = "HOLA MUNDO"  
  
'Convertimos  
cadena = LCase(cadena)  
  
'La variable cadena ahora vale "hola mundo"
```



Funciones propias de Visual Basic

Función Trim, LTrim y Rtrim

Elimina los espacios en blanco de un texto.

Función Len

Len (Aquí va la cadena que queremos averiguar su tamaño)

Función Asc y Chr

La función Asc nos permite obtener el código ASCII de un determinado carácter

```
Dim num As Integer  
num = Asc("a")
```

```
Dim letra As String  
letra = Chr(97)
```



Funciones propias de Visual Basic

Función InStr

```
InStr (comienzo, Cadena donde buscar , La Cadena a buscar)
```

```
Dim posicion As Integer
```

```
posicion = InStr("Estoy tomando mate", "mate")
```

```
Public Class Form1
    Dim Cadena As String
    Private Sub Form1_Load(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles MyBase.Load
        Cadena = "visual studio 2010"
        MsgBox(Funciones_vb.InStr(Cadena, "1"))
        MsgBox(InStr(Cadena, "1"))
    End Sub
End Class
```



Funciones propias de Visual Basic

Función Str y Val

```
cadena = "123456"  
cadena = Val(cadena)  
  
'Ahora cadena vale 123456  
  
cadena2 = 123456  
cadena2 = Str(cadena2)  
  
'Ahora cadena2 vale "123456"
```



Funciones propias de Visual Basic

Función LCase, Ucase

Convierte de Mayúscula a minúscula y viceversa.

Función strComp

Compara si 2 cadenas son iguales



Funciones propias de Visual Basic

Función Left, Right y Mid

```
Dim Cadena As String  
  
Cadena = Left("Ya es la madrugada", 9)  
  
'Cadena, que es un string, sería igual a: "Ya es la"
```

Función Replace

```
Public Class Form1  
    Dim Cadena As String  
    Private Sub Form1_Load(ByVal sender As System.Object, ByVal e As  
System.EventArgs) Handles MyBase.Load  
        Cadena = "Visual Studio 6.0"  
        MsgBox(Funciones_vb.Replace(Cadena, "6.0", "2010"))  
        MsgBox(Replace(Cadena, "6.0", "2010"))  
    End Sub  
End Class
```




Funciones propias de Visual Basic

Función Asc, AscW

Convierte un carácter en código ascii

```
Public Class Form1
    Dim Cadena As String
    Private Sub Form1_Load(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles MyBase.Load
        Cadena = "@"
        MsgBox (Funciones_vb.Asc (Cadena) )
        MsgBox (Asc (Cadena) )
    End Sub
End Class
```



Funciones propias de Visual Basic

Función Chr, ChrW

```
Public Class Form1
    Dim Cadena As Integer
    Private Sub Form1_Load(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles MyBase.Load
        Cadena = 64
        MsgBox(Funciones_vb.Chr(Cadena))
        MsgBox(Chr(Cadena))
    End Sub
End Class
```



Ejercicios de aplicación

- Ejercicio 1:
 - ▣ Tiempo: 20 minutos
 - ▣ Diseñar un programa utilizando un método que determine cuantas cifras posee un número entero positivo introducido por teclado.
 - ▣ Leer una cadena de caracteres, digitar el carácter que se quiera eliminar e imprimir la cadena resultante.
 - ▣ Crear un programa para imprimir la suma de los números impares menores o iguales que n .
 - ▣ Escribir un programa que reciba una cadena de caracteres e imprima cuantas vocales tiene dicha cadena.
 - ▣ Identificar si la cadena ingresada por teclado es palindrómica.