

Introducción WPF

Tópicos Selectos de Computación I

Octubre de 2013

WPF

- Windows Presentation Foundation (WPF) es un marco de trabajo de interfaz de usuario que crea aplicaciones cliente enriquecidas e interactivas.
- La plataforma de desarrollo de WPF admite un amplio conjunto de características de desarrollo de aplicaciones, que incluye un modelo de aplicaciones, recursos, controles, gráficos, diseño, enlace de datos, documentos y seguridad.

WPF

- Es un subconjunto de .NET Framework, de modo que si ha compilado previamente aplicaciones con .NET Framework mediante ASP.NET o Windows Forms, la experiencia de programación debe resultarle familiar.
- WPF utiliza el lenguaje XAML para proporcionar un modelo declarativo para la programación de aplicaciones.

Lenguaje XAML

Lenguaje XAML

- XAML es un lenguaje declarativo de marcado tal y como se aplica en el modelo de programación .NET Framework.
- XAML simplifica la creación de la UI para una aplicación .NET Framework.
- Se pueden crear elementos visibles de la UI en el marcado XAML declarativo y, a continuación, separar la definición de la UI de la lógica en tiempo de ejecución mediante archivos de código subyacente, que se unen al marcado mediante definiciones de clases parciales.
- XAML representa directamente la creación de instancias de objetos en un conjunto concreto de tipos de respaldo definidos en ensamblados. Esto no es lo que sucede con la mayoría de lenguajes de marcado, que normalmente se interpretan sin esa relación directa con un sistema de tipos de respaldo.

Lenguaje XAML

- El código XAML habilita un flujo de trabajo en el que las partes independientes pueden funcionar en la UI y la lógica de una aplicación, a través de herramientas potencialmente diferentes.
- Cuando se representan como texto, los archivos XAML son archivos XML que generalmente tienen la extensión .xaml. Los archivos se pueden codificar con cualquier codificación XML, pero lo habitual es la codificación UTF-8.

XAML

- Un elemento de objeto declara una instancia de un tipo. Ese tipo se define en los ensamblados que proporcionan los tipos de respaldo para una tecnología que utiliza XAML como un lenguaje.
- La sintaxis del elemento de objeto siempre se inicia con un corchete angular de apertura (<), que va seguido del nombre del tipo donde se desea crear una instancia.
- Después de esto, puede declarar opcionalmente los atributos en el elemento de objeto. Para completar la etiqueta del elemento de objeto, finalice con un corchete angular de cierre (>).
- Opcionalmente, puede usar un formulario de autocierre que no tenga contenido y completar la etiqueta con una barra diagonal y un corchete angular de cierre (/>) seguidos. Por ejemplo, examine de nuevo el fragmento de código de marcado mostrado anteriormente:

```
<StackPanel>  
  <Button Content="Click Me"/>  
</StackPanel>
```

[Detalles de la sintaxis](#)

XAML

```
<StackPanel>  
  <Button Content="Click Me"/>  
</StackPanel>
```

- Esto especifica dos elementos de objeto:
 <StackPanel> (con contenido y una etiqueta de cierre después) y <Button .../> (el formulario de autocierre, con varios atributos). Los elementos de objeto StackPanel y Button se asignan al nombre de una clase definida por WPF y que forma parte de los ensamblados de WPF. Al especificar una etiqueta de elemento de objeto, crea una instrucción para que el procesamiento XAML cree una nueva instancia. Cada instancia se crea llamando al constructor predeterminado del tipo subyacente al analizar y cargar el código XAML.

Sintaxis de atributos (propiedades)

- Las propiedades de un objeto se pueden expresar a menudo como atributos del elemento de objeto. Una sintaxis de atributo denomina la propiedad que está estableciendo en la sintaxis de atributo, seguida del operador de asignación (=).

Sintaxis de atributos (propiedades)

El valor de un atributo siempre se especifica como una cadena incluida entre comillas.

```
<Button Background="Blue" Foreground="Red" Content="This is a button"/>
```

Sintaxis de elementos de propiedad

- En algunas propiedades de un elemento de objeto, no es posible usar la sintaxis de atributo, ya que el objeto o la información necesaria para proporcionar el valor de la propiedad no se puede expresar correctamente dentro de las comillas y restricciones de cadena de la sintaxis de atributo.

Sintaxis de elementos de propiedad

```
<Button>
  <Button.Background>
    <SolidColorBrush Color="Blue"/>
  </Button.Background>
  <Button.Foreground>
    <SolidColorBrush Color="Red"/>
  </Button.Foreground>
  <Button.Content>
    This is a button
  </Button.Content>
</Button>
```