XAML文档定义了在WPF应用程序中组成窗口的面板、按钮以及各种控件的布局。

XAML是WPF技术中专门用于设计UI的语言。

程序设计要“高内聚-低耦合”。

XAML是一种由XML派生而来的语言。

XAML中使用标签声明一个元素，每个元素对应内存中的一个对象

非空标签：<Tag Attribute1 = Value1 Attribute2 = Value2>Content</Tag>

空标签：<Tag Attribute1 = Value1 Attribute2 = Value2 />

xmlns：名称空间

xmlns[:可选的映射前缀]=”名称空间”

如果没有前缀，就是默认名称空间，默认名称空间只能有一个。

<Window x:Class="xmal\_test1.MainWindow"

xmlns="http://schemas.microsoft.com/winfx/2006/xaml/presentation"

xmlns:x="http://schemas.microsoft.com/winfx/2006/xaml"

Title="MainWindow" Height="350" Width="525">

<Grid>

</Grid>

</Window>

上例中的XAML，<Window>和<Grid>都来自第二行的默认名称空间，而<Window>中的Class特征则来自第三行中的前缀为x的名称空间。

XAML文档的树形结构：

XAML中为对象属性赋值：

XAML中的一个标签的Attribute（特征）里有一部分与对象的Property互相对应。

例：程序xaml\_test2

<Window x:Class="xaml\_test2.MainWindow"

xmlns="http://schemas.microsoft.com/winfx/2006/xaml/presentation"

xmlns:x="http://schemas.microsoft.com/winfx/2006/xaml"

Title="MainWindow" Height="350" Width="525">

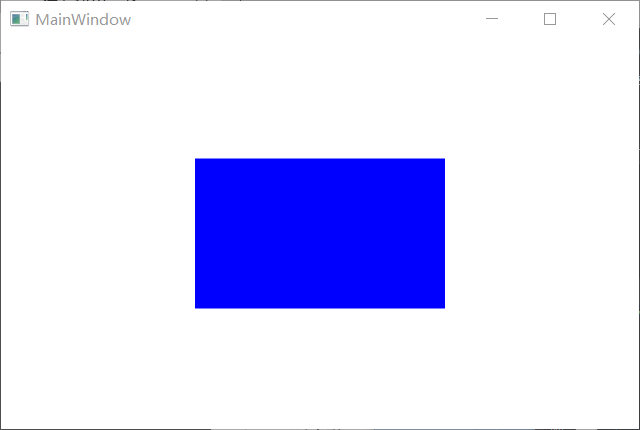
<Grid VerticalAlignment="Center" HorizontalAlignment="Center">

<Rectangle x:Name="rectangle" Width="200" Height="120" Fill="Blue"/>

</Grid>

</Window>

其中，<Rectangle>的标签的Fill特征初始值赋为Blue。效果如下：



通过Attribute=Value赋值时，Value只能是字符串值。这样会带来很大的不便。

使用属性元素为对象赋值

例：程序xaml\_test3

<Window x:Class="xaml\_test.MainWindow"

xmlns="http://schemas.microsoft.com/winfx/2006/xaml/presentation"

xmlns:x="http://schemas.microsoft.com/winfx/2006/xaml"

Title="MainWindow" Height="350" Width="525">

<Grid VerticalAlignment="Center" HorizontalAlignment="Center">

<Rectangle x:Name="rectangle" Width="200" Height="120">

<Rectangle.Fill>

<LinearGradientBrush>

<LinearGradientBrush.GradientStops>

<GradientStopCollection>

<GradientStop Offset="0.2" Color="LightBlue" />

<GradientStop Offset="0.7" Color="Blue" />

<GradientStop Offset="1.0" Color="DarkBlue" />

</GradientStopCollection>

</LinearGradientBrush.GradientStops>

</LinearGradientBrush>

</Rectangle.Fill>

</Rectangle>

</Grid>

</Window>

效果为：

