WPF中的，模板Template。

Template分为两大类：

1. ControlTemplate:算法内容的表现形式，控件如何组织起内部结构，并让程序员在原有的内部逻辑基础上扩展自己的逻辑
2. DataTemplate:数据内容的表现形式。一条数据显示成什么样子。

ControlTemplate是控件的外衣，DataTemplate是数据的外衣。

DataTemplate：

未使用DataTemplate的情形：事件驱动

MainWindow.xaml

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

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

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

xmlns:local="clr-namespace:template\_test1"

Title="MainWindow" Height="350" Width="623">

<StackPanel Orientation="Horizontal" Margin="5">

<local:CarDetailView x:Name="detailView" />

<ListBox x:Name="listBoxCars" Width="180" Margin="5,0" SelectionChanged="listBoxCars\_SelectionChanged" />

</StackPanel>

</Window>

MainWindow.xaml.cs

using *System*;

using *System*.*Collections*.*Generic*;

using *System*.*Linq*;

using *System*.*Text*;

using *System*.*Threading*.*Tasks*;

using *System*.*Windows*;

using *System*.*Windows*.*Controls*;

using *System*.*Windows*.*Data*;

using *System*.*Windows*.*Documents*;

using *System*.*Windows*.*Input*;

using *System*.*Windows*.*Media*;

using *System*.*Windows*.*Media*.*Imaging*;

using *System*.*Windows*.*Navigation*;

using *System*.*Windows*.*Shapes*;

namespace template\_test1

{

/// <summary>

/// MainWindow.xaml 的交互逻辑

/// </summary>

public partial class MainWindow : *Window*

{

public MainWindow()

{

*InitializeComponent*();

InitialCarList();

}

// 初始化ListBox

private void InitialCarList()

{

*List*<Car> carList = new *List*<Car>()

{

new Car() {Automaker = "Lamborghini", Name = "benz", Year="1990", TopSpeed = "340"},

new Car() {Automaker = "Lamborghini", Name = "dasatuo", Year="2001", TopSpeed = "353"}

};

foreach (Car car in carList)

{

CarListItemView view = new CarListItemView();

view.Car = car;

listBoxCars.*Items*.*Add*(view);

}

}

private void listBoxCars\_SelectionChanged(object sender, *SelectionChangedEventArgs* e)

{

CarListItemView view = e.*AddedItems*[0] as CarListItemView;

if (view != null)

{

detailView.Car = view.Car;

}

}

}

}

Car.cs

using *System*;

using *System*.*Collections*.*Generic*;

using *System*.*Linq*;

using *System*.*Text*;

using *System*.*Threading*.*Tasks*;

namespace template\_test1

{

public class Car

{

public string Automaker { get; set; }

public string Name { get; set; }

public string Year { get; set; }

public string TopSpeed { get; set; }

}

}

CarDetailView.xaml

<UserControl x:Class="template\_test1.CarDetailView"

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

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

xmlns:mc="http://schemas.openxmlformats.org/markup-compatibility/2006"

xmlns:d="http://schemas.microsoft.com/expression/blend/2008"

mc:Ignorable="d"

d:DesignHeight="300" d:DesignWidth="300">

<Border BorderBrush="Black" BorderThickness="1" CornerRadius="6">

<StackPanel Margin="5">

<Image x:Name="imagePhoto" Width="400" Height="250" />

<StackPanel Orientation="Horizontal" Margin="5,0">

<TextBlock Text="Name:" FontWeight="Bold" FontSize="20" />

<TextBlock x:Name="textBlockName" FontSize="20" Margin="5,0" />

</StackPanel>

<StackPanel Orientation="Horizontal" Margin="5,0">

<TextBlock Text="Automaker:" FontWeight="Bold" />

<TextBlock x:Name="textBlockAutomaker" Margin="5,0" />

<TextBlock Text="Year:" FontWeight="Bold" />

<TextBlock x:Name="textBlockYear" Margin="5,0" />

<TextBlock Text="Top Speed:" FontWeight="Bold" />

<TextBlock x:Name="textBlockTopSpeed" Margin="5,0" />

</StackPanel>

</StackPanel>

</Border>

</UserControl>

CarDetailView.xaml.cs

using *System*;

using *System*.*Collections*.*Generic*;

using *System*.*IO*;

using *System*.*Linq*;

using *System*.*Text*;

using *System*.*Threading*.*Tasks*;

using *System*.*Windows*;

using *System*.*Windows*.*Controls*;

using *System*.*Windows*.*Data*;

using *System*.*Windows*.*Documents*;

using *System*.*Windows*.*Input*;

using *System*.*Windows*.*Media*;

using *System*.*Windows*.*Media*.*Imaging*;

using *System*.*Windows*.*Navigation*;

using *System*.*Windows*.*Shapes*;

namespace template\_test1

{

/// <summary>

/// CarDetailView.xaml 的交互逻辑

/// </summary>

public partial class CarDetailView : *UserControl*

{

public CarDetailView()

{

*InitializeComponent*();

}

private Car car;

public Car Car

{

get { return car; }

set

{

car = value;

textBlockName.*Text* = car.Name;

textBlockYear.*Text* = car.Year;

textBlockTopSpeed.*Text* = car.TopSpeed;

textBlockAutomaker.*Text* = car.Automaker;

string uri = string.*Format*(@"/Resources/Image/{0}.jpg", car.Name);

if (*File*.*Exists*(uri))

{

}

imagePhoto.*Source* = new *BitmapImage*(new *Uri*(uri, *UriKind*.*RelativeOrAbsolute*));

}

}

}

}

CarListItemView.xaml

<UserControl x:Class="template\_test1.CarListItemView"

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

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

xmlns:mc="http://schemas.openxmlformats.org/markup-compatibility/2006"

xmlns:d="http://schemas.microsoft.com/expression/blend/2008"

mc:Ignorable="d"

d:DesignHeight="300" d:DesignWidth="300">

<Grid Margin="2">

<StackPanel Orientation="Horizontal">

<Image x:Name="imageLogo" Grid.RowSpan="3" Width="64" Height="64" />

<StackPanel Margin="5,10">

<TextBlock x:Name="textBlockName" FontSize="16" FontWeight="Bold" />

<TextBlock x:Name="textBlockYear" FontSize="14" />

</StackPanel>

</StackPanel>

</Grid>

</UserControl>

CarListItemView.xaml.cs

using *System*;

using *System*.*Collections*.*Generic*;

using *System*.*IO*;

using *System*.*Linq*;

using *System*.*Text*;

using *System*.*Threading*.*Tasks*;

using *System*.*Windows*;

using *System*.*Windows*.*Controls*;

using *System*.*Windows*.*Data*;

using *System*.*Windows*.*Documents*;

using *System*.*Windows*.*Input*;

using *System*.*Windows*.*Media*;

using *System*.*Windows*.*Media*.*Imaging*;

using *System*.*Windows*.*Navigation*;

using *System*.*Windows*.*Shapes*;

namespace template\_test1

{

/// <summary>

/// CarListItemView.xaml 的交互逻辑

/// </summary>

public partial class CarListItemView : *UserControl*

{

public CarListItemView()

{

*InitializeComponent*();

}

private Car car;

public Car Car

{

get { return car; }

set

{

car = value;

textBlockName.*Text* = car.Name;

textBlockYear.*Text* = car.Year;

// 转义字符@

string uriStr = string.*Format*(@"/Resources/logo/{0}.png", car.Automaker);

imageLogo.*Source* = new *BitmapImage*(new *Uri*(uriStr, *UriKind*.*Relative*));

}

}

}

}

使用DataTemplate：数据驱动

Car.cs:

using *System*;

using *System*.*Collections*.*Generic*;

using *System*.*Linq*;

using *System*.*Text*;

using *System*.*Threading*.*Tasks*;

namespace template\_test2

{

public class Car

{

public string Automaker { get; set; }

public string Name { get; set; }

public string Year { get; set; }

public string TopSpeed { get; set; }

}

}

AutomakerToLogoPathConvert.cs

using *System*;

using *System*.*Collections*.*Generic*;

using *System*.*Globalization*;

using *System*.*Linq*;

using *System*.*Text*;

using *System*.*Threading*.*Tasks*;

using *System*.*Windows*.*Data*;

using *System*.*Windows*.*Media*;

using *System*.*Windows*.*Media*.*Imaging*;

namespace template\_test2

{

// 厂商名称转换为Logo图片路径

public class AutomakerToLogoPathConverter:*IValueConverter*

{

// 正向转换

public object Convert(object value, *Type* targetType, object parameter, *CultureInfo* culture)

{

string uriStr = string.*Format*(@"/Resources/logo/{0}.png", (string)value);

return new *BitmapImage*(new *Uri*(uriStr, *UriKind*.*Relative*));

}

public object ConvertBack(object value, *Type* targetType, object parameter, *CultureInfo* culture)

{

throw new *NotImplementedException*();

}

}

public class NameToLogoPathConverter : *IValueConverter*

{

// 正向转换

public object Convert(object value, *Type* targetType, object parameter, *CultureInfo* culture)

{

string uriStr = string.*Format*(@"/Resources/Image/{0}.jpg", (string)value);

return new *BitmapImage*(new *Uri*(uriStr, *UriKind*.*Relative*));

}

public object ConvertBack(object value, *Type* targetType, object parameter, *CultureInfo* culture)

{

throw new *NotImplementedException*();

}

}

}

MainWindow.xaml

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

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

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

xmlns:local="clr-namespace:template\_test2"

Title="MainWindow" Height="350" Width="623">

<Window.Resources>

<!--Converters-->

<local:AutomakerToLogoPathConverter x:Key="a2l" />

<local:NameToLogoPathConverter x:Key="n2p" />

<!--Data Template for Detail View-->

<DataTemplate x:Key="carDetailViewTemplate">

<Border BorderBrush="Black" BorderThickness="1" CornerRadius="6">

<StackPanel Margin="5">

<Image Width="400" Height="250" Source="{Binding Name, Converter={StaticResource n2p}}" />

<StackPanel Orientation="Horizontal" Margin="5,0">

<TextBlock Text="Name:" FontWeight="Bold" FontSize="20" />

<TextBlock Text="{Binding Name}" FontSize="20" Margin="5,0" />

</StackPanel>

<StackPanel Orientation="Horizontal" Margin="5,0">

<TextBlock Text="Automaker:" FontWeight="Bold" />

<TextBlock Text="{Binding Automaker}" Margin="5,0" />

<TextBlock Text="Year:" FontWeight="Bold" />

<TextBlock Text="{Binding Year}" Margin="5,0" />

<TextBlock Text="Top Speed:" FontWeight="Bold" />

<TextBlock Text="{Binding TopSpeed}" Margin="5,0" />

</StackPanel>

</StackPanel>

</Border>

</DataTemplate>

<!--Data Template for Item View-->

<DataTemplate x:Key="carListItemViewTemplate">

<Grid Margin="2">

<StackPanel Orientation="Horizontal">

<Image Source="{Binding Automaker, Converter={StaticResource a2l}}" Grid.RowSpan="3" Width="64" Height="64" />

<StackPanel Margin="5,10">

<TextBlock Text="{Binding Name}" FontSize="16" FontWeight="Bold" />

<TextBlock Text="{Binding Year}" FontSize="14" />

</StackPanel>

</StackPanel>

</Grid>

</DataTemplate>

</Window.Resources>

<!--窗体的内容-->

<StackPanel Orientation="Horizontal" Margin="5">

<UserControl ContentTemplate="{StaticResource carDetailViewTemplate}"

Content="{Binding SelectedItem, ElementName=listBoxCars}"/>

<ListBox x:Name="listBoxCars" Width="180" Margin="5,0"

ItemTemplate="{StaticResource carListItemViewTemplate}"/>

</StackPanel>

</Window>

MainWindow.xaml.cs

using *System*;

using *System*.*Collections*.*Generic*;

using *System*.*Linq*;

using *System*.*Text*;

using *System*.*Threading*.*Tasks*;

using *System*.*Windows*;

using *System*.*Windows*.*Controls*;

using *System*.*Windows*.*Data*;

using *System*.*Windows*.*Documents*;

using *System*.*Windows*.*Input*;

using *System*.*Windows*.*Media*;

using *System*.*Windows*.*Media*.*Imaging*;

using *System*.*Windows*.*Navigation*;

using *System*.*Windows*.*Shapes*;

namespace template\_test2

{

/// <summary>

/// MainWindow.xaml 的交互逻辑

/// </summary>

public partial class MainWindow : *Window*

{

public MainWindow()

{

*InitializeComponent*();

InitialCarList();

}

// 初始化ListBox

private void InitialCarList()

{

*List*<Car> carList = new *List*<Car>()

{

new Car() {Automaker = "Lamborghini", Name = "benz", Year="1990", TopSpeed = "340"},

new Car() {Automaker = "Lamborghini", Name = "dasatuo", Year="2001", TopSpeed = "353"}

};

// 填充数据源

listBoxCars.*ItemsSource* = carList;

}

}

}

ControlTemplate：控件的外衣

可以看看开放云书院和橙立方是如何换肤的。