

Работа с элементом управления CheckBox

Рассмотрим особенности использования элемента управления CheckBox, который считается либо включенным при наличии флажка внутри элемента управления, либо выключенным при отсутствии флажка.

Элемент управления CheckBox имеет следующие свойства:

свойство IsChecked - указывает, включен или выключен элемент управления (есть ли флажок внутри элемента управления);

свойство Color - задает цвет элемента управления;

Элементу CheckBox соответствует событие CheckChanged, которое возникает при изменении значения свойства IsChecked.

При нахождении элемента управления CheckBox во включенном или выключенном состоянии, с помощью диспетчера визуальных состояний VisualStateManager и элемента Setter может производиться изменение визуального состояния элемента управления. При этом производится присвоение значения Value свойству элемента управления, указанному в Property для случая, когда у него наступает состояние, отмеченное с помощью атрибута x:Name.

В примере видно, что при включении CheckBox его свойство Color будет иметь значение Green.

```
<VisualStateManager.VisualStateGroups>
  <VisualStateGroup x:Name="States">
    <VisualState x:Name="IsChecked">
      <VisualState.Setters>
        <Setter Property="Color"
          Value="Green" />
      </VisualState.Setters>
    </VisualState>
  </VisualStateGroup>
</VisualStateManager.VisualStateGroups>
```

Для работы с элементом управления CheckBox создадим проект CheckBoxDemo для мультиплатформенного программного приложения Xamarin.Forms. Откроем файл MainPage.xaml и наберем программный код, приведенный ниже.

В программном коде внутри страницы содержится макет Grid, в котором три строки и четыре столбца.

В первом и третьем столбцах первой строки располагаются элементы управления Label, в которых отображается текст со значениями свойства FontAttributes, применяемого для выбора начертания текста.

Во втором и четвертом столбце первой строки размещаются элементы управления CheckBox, включение которых приводит к применению значений свойства FontAttributes к тексту в элементе управления Label, расположенном в третьей строке таблицы.

В первом и третьем столбцах второй строки располагаются элементы управления Label, в которых отображается текст со значениями свойства TextDecorations, применяемого для оформления текста.

Во втором и четвертом столбце второй строки размещаются элементы управления CheckBox, включение которых приводит к применению значений свойства TextDecorations к тексту в элементе управления Label, расположенном в третьей строке таблицы.

В третьей строке объединены все четыре столбца, и в ней размещается элемент управления Label, к тексту которого может применяться форматирование в случае включения одного или нескольких элементов управления CheckBox.

При изменении свойства IsChecked у элементов управления CheckBox возникает событие CheckedChanged.

Для реагирования на это событие в программном коде предусмотрено использование четырех обработчиков событий (по одному для каждого элемента управления CheckBox): OnItalic, OnBold, OnUnderline, OnStrikethrough.

При этом, элементу управления Label с помощью атрибута x:Name назначено имя «label» для обращения к нему из программного кода обработчиков события.

Программный код обработчиков событий приведен в файле MainPage.xaml.cs.

```
<ContentPage xmlns="http://xamarin.com/schemas/2014/forms"
  xmlns:x="http://schemas.microsoft.com/winfx/2009/xaml"
  x:Class="CheckBoxDemo.MainPage"
  BackgroundColor="Beige"
  Padding="0,100,0,0">
  <Grid Margin="20,35,20,20">
    <Grid.ColumnDefinitions>
      <ColumnDefinition Width="0.25*" />
      <ColumnDefinition Width="0.20*" />
      <ColumnDefinition Width="0.35*" />
      <ColumnDefinition Width="0.20*" />
    </Grid.ColumnDefinitions>
    <Grid.RowDefinitions>
      <RowDefinition Height="50" />
      <RowDefinition Height="50" />
      <RowDefinition Height="200" />
    </Grid.RowDefinitions>
    <Label Text="Italic:"
      TextColor="Black" FontAttributes="Bold"
      FontSize="16"
      VerticalOptions="Center"/>
    <CheckBox Grid.Column="1"
      CheckedChanged="OnItalic"
      VerticalOptions="Center" />
```

```

<Label Grid.Column="2"
    Text="Bold:"
    TextColor="Black" FontAttributes="Bold"
    FontSize="16"
    VerticalOptions="Center" />
<CheckBox Grid.Column="3"
    Color="Red"
    CheckedChanged="OnBold"
    VerticalOptions="Center" />
<Label Grid.Row="1"
    Text="Underline:"
    TextColor="Black" FontAttributes="Bold"
    FontSize="16"
    VerticalOptions="Center" />
<CheckBox Grid.Row="1"
    Grid.Column="1"
    Color="Green"
    CheckedChanged="OnUnderline"
    VerticalOptions="Center" />
<Label Grid.Row="1"
    Grid.Column="2"
    Text="Strikethrough:"
    TextColor="Black" FontAttributes="Bold"
    FontSize="16"
    VerticalOptions="Center" />
<CheckBox Grid.Row="1"
    Grid.Column="3"
    Color="Blue"
    CheckedChanged="OnStrikethrough"
    VerticalOptions="Center" />
<Label x:Name="label"
    Grid.Row="2"
    Grid.ColumnSpan="4"
    TextColor="Black" FontAttributes="Bold"
    FontSize="30"
    HorizontalTextAlignment="Center"
    VerticalTextAlignment="Center"
    Text="Форматирование текста в элементе управления Label"/>
</Grid>
</ContentPage>

```

Откроем файл MainPage.xaml.cs и наберем программный код, приведенный далее.

В качестве значения аргумента sender в обработчики событий (OnItalic, OnBold, OnUnderline, OnStrikethrough) передаются те элементы управления CheckBox, за которыми закреплены обработчики событий в соответствии с программным кодом в файле MainPage.xaml.

Также в обработчиках событию `CheckedChanged` соответствует объект `CheckedChangedEventArgs`, который имеет свойство с именем `Value`.

При возникновении события `CheckedChanged` в качестве значения свойства `Value` устанавливается новое значение свойства `IsChecked`.

При включении элементов управления `CheckBox` происходит изменение значения свойств `FontAttributes` и `TextDecorations` для текста, отображаемого в объекте `label`, который связан с элементом управления `Label`, сформированном в третьей строке таблицы в файле `MainPage.xaml`.

При отключении элементов управления `CheckBox` происходит отмена возврат значений свойств `FontAttributes` и `TextDecorations` элемента управления `Label` в состояние, которое было до включения элемента управления.

`using Xamarin.Forms;`

```
namespace CheckBoxDemo
{
    public partial class MainPage: ContentPage
    {
        public MainPage()
        {
            InitializeComponent();
        }
        void OnItalic(object sender, CheckedChangedEventArgs e)
        {
            if (e.Value)
            {
                label.FontAttributes |= FontAttributes.Italic;
            }
            else
            {
                label.FontAttributes &= ~FontAttributes.Italic;
            }
        }
        void OnBold(object sender, CheckedChangedEventArgs e)
        {
            if (e.Value)
            {
                label.FontAttributes |= FontAttributes.Bold;
            }
            else
            {
                label.FontAttributes &= ~FontAttributes.Bold;
            }
        }
        void OnUnderline(object sender, CheckedChangedEventArgs e)
        {
            if (e.Value)
```

```

{
    label.TextDecorations |= TextDecorations.Underline;
}
else
{
    label.TextDecorations &= ~TextDecorations.Underline;
}
}
void OnStrikethrough(object sender, CheckedChangedEventArgs e)
{
    if (e.Value)
    {
        label.TextDecorations |= TextDecorations.Strikethrough;
    }
    else
    {
        label.TextDecorations &= ~TextDecorations.Strikethrough;
    }
}
}
}

```

Запустим программное приложение для работы с элементом управления CheckBox.

На пользовательском интерфейсе приведены четыре элемента управления CheckBox.

Включение двух элементов управления CheckBox, расположенных в первой строке, приводит к изменению начертания текста, расположенного под элементами управления CheckBox, с помощью свойства FontAttributes.

Включение двух элементов управления CheckBox, расположенных во второй строке, приводит к изменению оформления текста, расположенного под элементами управления CheckBox, с помощью свойства TextDecorations. Выключение элементов управления CheckBox приводит к возврату текста в элементе управления Label к состоянию, которое было до включения элементов управления CheckBox.

