

Пользовательский интерфейс: Использование элемента управления **Stepper**

Приступим к изучению элемента управления **Stepper**, который состоит из двух кнопок с знаками «минус» и «плюс».

Пользователь может управлять этими кнопками для последовательного выбора значения из диапазона значений



Элементы управления **Stepper** имеет четыре свойства:

- свойство **Increment** - величина, на которую нужно изменить выбранное значение, равное 1 по умолчанию;

- свойство **Minimum** – минимальное значение диапазона, равное 0 по умолчанию;

- свойство **Maximum** - максимальное значение диапазона, равное 100 по умолчанию;

- свойство **Value** - значение, установленное с помощью элемента управления **Stepper**, которое может находиться в диапазоне от **Minimum** до **Maximum** и имеет значение по умолчанию, равное 0.

Значение свойства **Value**, установленное с помощью элемента управления **Stepper**, должно быть в пределах между значениями свойств **Minimum** и **Maximum** включительно.

Если свойству **Minimum** присвоено значение большее, чем значение свойства **Value**, то элемент управления **Stepper** установит для свойства **Minimum** значение свойства **Value**.

Аналогично, если для свойства **Maximum** задано значение, меньше значения свойства **Value**, то элемент управления **Stepper** установит для свойства **Maximum** значение свойства **Value**.

Элементу управления **Stepper** соответствует событие **ValueChanged**, возникающее при изменении значения свойства **Value**.

Событие **ValueChanged** сопровождается объектом **ValueChangedEventArgs**, который имеет два свойства: **OldValue** и **NewValue**.

Во время срабатывания события значение **NewValue** совпадает со значением **Value**.

Для работы с элементом управления **Stepper** создайте проект **StepperDemo** для мультиплатформенного приложения **Xamarin.Forms** с использованием шаблона «Пустой». Откройте файл **MainPage.xaml** и наберите программный код, приведенный далее.

В программном коде внутри страницы находится макет **StackLayout**, внутри которого расположены:

1. Элемент управления **Label**, который с помощью изменения свойства **Rotation** будет поворачиваться на угол, равный значению свойства **Value** элемента управления **Stepper**. Значение свойства **Rotation** данного элемента управления **Label** привязано с помощью **Binding** к значению свойства **Value** элемента управления **Stepper**.

2. Элемент управления `Stepper`, у которого значение свойства `Maximum` равно 360, а значение свойства `Increment` равно 30.

3. Элемент управления `Label`, в котором с помощью изменения свойства `Text` будет отображаться значение свойства `Value` элемента управления `Stepper`. Значение свойства `Text` данного элемента управления `Label` привязано с помощью `Binding` к значению свойства `Value` элемента управления `Stepper`.

Значение `Value` элемента управления `Stepper` может изменяться в пределах от минимального значения в соответствии с значением свойства `Minimum` до максимального значения в соответствии со значением свойства `Maximum`. Шаг изменения значения свойства `Value` равен значению свойства `Increment`.

Благодаря привязке `Binding` свойству `Rotation` элемента управления `Label`, объявленного в программном коде перед элементом управления `Stepper`, присваивается значение свойства `Value` элемента управления `Stepper`.

Также, благодаря привязке `Binding`, свойству `Text` элемента управления `Label`, объявленного в программном коде после элемента управления `Stepper`, присваивается значение свойства `Value` элемента управления `Stepper`.

При этом с использованием `StringFormat` производится перевода числового значения свойства `Value` элемента управления `Stepper` в текстовое значение для элемента управления `Label`.

```
<ContentPage xmlns="http://xamarin.com/schemas/2014/forms"
    xmlns:x="http://schemas.microsoft.com/winfx/2009/xaml"
    x:Class="StepperDemo.MainPage"
    BackgroundColor="Cyan">
<StackLayout Margin="20">
    <Label Text="Вращение текста"
        Rotation="{Binding Source={x:Reference step}, Path=Value}"
        FontSize="20"
        BackgroundColor="Red"
        TextColor="Yellow"
        WidthRequest="200"
        HeightRequest="50"
        VerticalTextAlignment="Center"
        HorizontalTextAlignment="Center"
        HorizontalOptions="Center"
        VerticalOptions="CenterAndExpand"
        FontAttributes="Bold"
    />
    <Stepper x:Name="step"
        Maximum="360"
        Increment="30"
        HorizontalOptions="Center"
        VerticalOptions="CenterAndExpand"/>
    <Label Text="{Binding Source={x:Reference step}, Path=Value,
        StringFormat='Значение свойства Value равно {0:F0}}'"
        FontSize="20"
        TextColor="Black"
```

```
FontAttributes="Bold"  
HorizontalOptions="Center"  
VerticalOptions="CenterAndExpand" />  
</StackLayout>  
</ContentPage>
```

Запустим программное приложение для демонстрации работы с элементом управления Stepper.

При запуске программного приложения значение Value элемента управления Stepper равно нулю.

При каждом нажатии на кнопку «плюс» элемента управления Stepper элемент управления Label поворачивается на 30 градусов.

Значение угла, на который осуществлен поворот, отображается в элементе управления Label, расположенном под элементом управления Stepper.

Если значение Value равно максимальному значению 360 градусов, то происходит блокировка кнопки со знаком «плюс».

Если значение Value достигает минимального значения 0 градусов, то происходит блокировка кнопки со знаком «минус».

