

# **Zeus ValueConverters Ver. 1.0**

## **Manual**

**Copyright © 2019 Χρήστος Μουρατίδης**

---

## Πίνακας περιεχομένων

<b>ZEUS VALUECONVERTERS VERSION 1.0</b> .....	<b>2</b>
ΟΙ ΟΡΙΣΜΟΙ ΤΩΝ ΚΛΑΣΕΩΝ .....	5
ΔΙΑΝΟΜΗ .....	12
ΕΠΙΚΟΙΝΩΝΙΑ.....	13
<b>BOOLEANREVERSECONVERTER</b> .....	<b>14</b>
<b>BOOLEANTOBITCONVERTER</b> .....	<b>16</b>
ΙΔΙΟΤΗΤΕΣ.....	18
<i>BitWhenTrue</i> .....	19
<i>BitWhenFalse</i> .....	21
<b>BOOLEANTOBRUSHCONVERTER</b> .....	<b>23</b>
ΙΔΙΟΤΗΤΕΣ.....	26
<i>BrushWhenTrue</i> .....	27
<i>BrushWhenFalse</i> .....	29
<b>BOOLEANTOFONTSIZECONVERTER</b> .....	<b>31</b>
ΙΔΙΟΤΗΤΕΣ.....	34
<i>FontSizeWhenTrue</i> .....	35
<i>FontSizeWhenFalse</i> .....	37
<b>BOOLEANTOFONTSTYLECONVERTER</b> .....	<b>39</b>
ΙΔΙΟΤΗΤΕΣ.....	42
<i>FontStyleWhenTrue</i> .....	43
<i>FontStyleWhenFalse</i> .....	45
<b>BOOLEANTOFONTWEIGHTCONVERTER</b> .....	<b>47</b>
ΙΔΙΟΤΗΤΕΣ.....	50
<i>FontWeightWhenTrue</i> .....	51
<i>FontWeightWhenFalse</i> .....	53
<b>BOOLEANTOIMAGESOURCECONVERTER</b> .....	<b>55</b>
ΙΔΙΟΤΗΤΕΣ.....	58
<i>ImageSourceWhenTrue</i> .....	59
<i>ImageSourceWhenFalse</i> .....	61
<b>BOOLEANTOSTRINGCONVERTER</b> .....	<b>63</b>
ΙΔΙΟΤΗΤΕΣ.....	66
<i>StringWhenTrue</i> .....	67
<i>StringWhenFalse</i> .....	69
<b>BOOLEANTOVISIBILITYCONVERTER</b> .....	<b>71</b>
ΙΔΙΟΤΗΤΕΣ.....	74
<i>VisibilityWhenTrue</i> .....	75
<i>VisibilityWhenFalse</i> .....	77

## Zeus ValueConverters version 1.0



**Κλάσεις:** BooleanReverseConverter, BooleanToBitConverter, BooleanToBrushConverter, BooleanToFontSizeConverter, BooleanToFontStyleConverter, BooleanToFontWeightConverter, BooleanToImageSourceConverter, BooleanToStringConverter, BooleanToVisibilityConverter.

**Inherits:** System.Windows.Data.IValueConverter

**Namespace:** Zeus.WPF.Classes.ValueConverters

**Assembly:** ZeusValueConverters (in ZeusValueConverters.dll)

**Dependencies:** -

### Περιγραφή

Η βιβλιοθήκη **Zeus ValueConverters** περιλαμβάνει (προς το παρόν στην έκδοση 1.0) **εννέα κλάσεις ValueConverters**. Τα ValueConverters αντικείμενα μπορούν να χρησιμοποιηθούν σε binding εκφράσεις, όταν απαιτείται μετατροπή τιμής από έναν τύπο σε άλλον.

Ως γνωστόν, μία binding έκφραση συνδέει δύο ιδιότητες μεταξύ τους, μία source και μία target. Αν οι ιδιότητες αυτές είναι διαφορετικού τύπου τότε χρειάζεται να χρησιμοποιήσουμε στην binding έκφραση ένα αντικείμενο ValueConverter. Το ποιοί γνωστό παράδειγμα, είναι η σύνδεση της ιδιότητας Visibility ενός γραφικού element (π.χ. TextBlock) , με μία Boolean ιδιότητα ενός source αντικειμένου. Αν η boolean ιδιότητα είναι True, τότε η ιδιότητα Visibility να έχει την τιμή Visible.

Στην περίπτωση που το γραφικό αντικείμενο είναι ένα input element (π.χ. TextBox), αυτό σημαίνει ότι δέχεται τιμή από τον χρήστη. Έτσι, η έκφραση binding είναι two-way. Δηλαδή, μία αλλαγή τιμής στο target element (TextBox) αλλάζει την τιμή στην συνδεδεμένη ιδιότητα του source αντικειμένου. Η κλάση μετατροπής (ValueConverter), που χρησιμοποιείται στην binding έκφραση, πρέπει να υποστηρίζει το two-way mode. Τεχνικά μιλώντας, στην υλοποίησή της, πρέπει να έχει κώδικα μετατροπής στην μέθοδο Convert (one-way) αλλά και στην μέθοδο ConvertBack (two-way).

Παρακάτω, παρατίθενται οι **κλάσεις ValueConverter**.

VC	BooleanReverseConverter	<p>Μετατρέπει μία <b>boolean</b> τιμή στο αντίθετό της. Υποστηρίζει <b>two-way</b> binding mode.</p> <p>Για παράδειγμα, αν η source τιμή είναι True, τότε επιστρέφει False.</p>
VC	BooleanToBitConverter	<p>Μετατρέπει μία <b>boolean</b> τιμή σε <b>0</b> ή <b>1</b>. Υποστηρίζει <b>two-way</b> binding mode.</p> <p>Για παράδειγμα, αν η source τιμή είναι True, τότε να επιστρέφει 1.</p>
VC	BooleanToBrushConverter	<p>Μετατρέπει μία <b>boolean</b> τιμή σε <b>Brush</b>. Υποστηρίζει <b>one-way</b> binding mode.</p> <p>Για παράδειγμα, αν η source τιμή είναι True, τότε να επιστρέφει Brushes.Orange.</p>
VC	BooleanToFontSizeConverter	<p>Μετατρέπει μία <b>boolean</b> τιμή σε <b>Double</b>, που δηλώνει το font size. Υποστηρίζει <b>one-way</b> binding mode.</p> <p>Για παράδειγμα, αν η source τιμή είναι True, τότε να επιστρέφει 16.0.</p>
VC	BooleanToFontStyleConverter	<p>Μετατρέπει μία <b>boolean</b> τιμή σε <b>FontStyle</b>. Υποστηρίζει <b>one-way</b> binding mode.</p> <p>Για παράδειγμα, αν η source τιμή είναι True, τότε να επιστρέφει FontStyles.Italic.</p>
VC	BooleanToFontWeightConverter	<p>Μετατρέπει μία <b>boolean</b> τιμή σε <b>FontWeight</b>. Υποστηρίζει <b>one-way</b> binding mode.</p> <p>Για παράδειγμα, αν η source τιμή είναι True, τότε να επιστρέφει FontWeights,Bold.</p>
VC	BooleanToImageSourceConverter	<p>Μετατρέπει μία <b>boolean</b> τιμή σε <b>ImageSource</b>. Υποστηρίζει <b>one-way</b> binding mode.</p> <p>Για παράδειγμα, αν η source τιμή είναι True, τότε να επιστρέφει μία συγκεκριμένη εικόνα.</p>

<b>VC</b>	BooleanToStringConverter	Μετατρέπει μία <b>boolean</b> τιμή σε <b>String</b> . Υποστηρίζει <b>two-way</b> binding mode.  Για παράδειγμα, αν η source τιμή είναι True, τότε να επιστρέφει "Ναι".
<b>VC</b>	BooleanToVisibilityConverter	Μετατρέπει μία <b>boolean</b> τιμή σε <b>Visibility</b> . Υποστηρίζει <b>one-way</b> binding mode.  Για παράδειγμα, αν η source τιμή είναι True, τότε να επιστρέφει <code>Visibility.Visible</code> .

## Παρατηρήσεις

Περισσότερα ValueConverters θα προστίθενται στη βιβλιοθήκη σε μελλοντικές εκδόσεις.

## Οι ορισμοί των κλάσεων

Οι κλάσεις έχουν οριστεί ως εξής:

---

- Για την κλάση **BooleanReverseConverter**:

**Σύνταξη:**

**VB:**

```
Imports System.Windows.Data
```

```
<ValueConversion(GetType(Nullable(Of Boolean)), GetType(Nullable(Of Boolean)))>  
Public Class BooleanReverseConverter  
    Implements IValueConverter
```

**XAML Object Element Usage:**

Εισαγωγή namespace:

```
xmlns:zeus="clr-namespace:Zeus.WPF.Classes.ValueConverters;assembly=ZeusValueConverters"
```

Χρήση:

```
<container.Resources >  
    <zeus:BooleanReverseConverter x:Key="booleanReverseConverter"/>  
</container.Resources>  
  
<Element property="{Binding sourcePropertyBooleanType,  
    Mode=OneOrTwoWay,  
    Converter={StaticResource booleanReverseConverter}}" />
```

- 
- Για την κλάση **BooleanToBitConverter**:

**Σύνταξη:**

**VB:**

```
Imports System.Windows.Data
```

```
<ValueConversion(GetType(Nullable(Of Boolean)), GetType(Byte))>  
Public Class BooleanToBitConverter  
    Implements IValueConverter
```

## XAML Object Element Usage:

Εισαγωγή namespace:

```
xmlns:zeus="clr-namespace:Zeus.WPF.Classes.ValueConverters;assembly=ZeusValueConverters"
```

Χρήση:

```
<container.Resources >
    <zeus:BooleanToBitConverter x:Key="booleanToBitConverter"/>
</container.Resources>

<Element property="{Binding sourcePropertyBooleanType,
    Mode=OneOrTwoWay,
    Converter={StaticResource booleanToBitConverter}}" />
```

---

- **ια την κλάση BooleanToBrushConverter:**

Σύνταξη:

VB:

```
Imports System.Windows.Data
```

```
<ValueConversion(GetType(Nullable(Of Boolean)), GetType(Brush))>
Public Class BooleanToBrushConverter
    Implements IValueConverter
```

## XAML Object Element Usage:

Εισαγωγή namespace:

```
xmlns:zeus="clr-namespace:Zeus.WPF.Classes.ValueConverters;assembly=ZeusValueConverters"
```

Χρήση:

```
<container.Resources >
    <zeus:BooleanToBrushConverter x:Key="booleanToBrushConverter"/>
</container.Resources>

<Element property="{Binding sourcePropertyBooleanType,
    Converter={StaticResource booleanToBrushConverter}}" />
```

---

- Για την κλάση **BooleanToFontSizeConverter**:

**Σύνταξη:**

**VB:**

```
Imports System.Windows.Data
```

```
<ValueConversion(GetType(Nullable(Of Boolean)), GetType(Double))>  
Public Class BooleanToFontSizeConverter  
    Implements IValueConverter
```

**XAML Object Element Usage:**

Εισαγωγή namespace:

```
xmlns:zeus="clr-namespace:Zeus.WPF.Classes.ValueConverters;assembly=ZeusValueConverters"
```

Χρήση:

```
<container.Resources >  
    <zeus:BooleanToFontSizeConverter x:Key="booleanToFontSizeConverter"/>  
</container.Resources>  
  
<Element property="{Binding sourcePropertyBooleanType,  
                           Converter={StaticResource booleanToFontSizeConverter}}" />
```

- 
- Για την κλάση **BooleanToFontStyleConverter**:

**Σύνταξη:**

**VB:**

```
Imports System.Windows.Data
```

```
<ValueConversion(GetType(Nullable(Of Boolean)), GetType(FontStyle))>  
Public Class BooleanToFontStyleConverter  
    Implements IValueConverter
```

**XAML Object Element Usage:**



Εισαγωγή namespace:

```
xmlns:zeus="clr-namespace:Zeus.WPF.Classes.ValueConverters;assembly=ZeusValueConverters"
```

Χρήση:

```
<container.Resources >  
    <zeus:BooleanToFontStyleConverter x:Key="booleanToFontStyleConverter"/>  
</container.Resources>  
  
<Element property="{Binding sourcePropertyBooleanType,  
    Converter={StaticResource booleanToFontStyleConverter}}" />
```

---

- Για την κλάση **BooleanToFontWeightConverter**:

Σύνταξη:

VB:

```
Imports System.Windows.Data
```

```
<ValueConversion(GetType(Nullable(Of Boolean)), GetType(FontWeight))>  
Public Class BooleanToFontWeightConverter  
    Implements IValueConverter
```

**XAML Object Element Usage:**

Εισαγωγή namespace:

```
xmlns:zeus="clr-namespace:Zeus.WPF.Classes.ValueConverters;assembly=ZeusValueConverters"
```

Χρήση:

```
<container.Resources >  
    <zeus:BooleanToFontWeightConverter x:Key="booleanToFontWeightConverter"/>  
</container.Resources>  
  
<Element property="{Binding sourcePropertyBooleanType,  
    Converter={StaticResource booleanToFontWeightConverter}}" />
```

---

- Για την κλάση **BooleanToImageSourceConverter**:

**Σύνταξη:**

**VB:**

```
Imports System.Windows.Data
Imports System.Windows.Media

<ValueConversion(GetType(Nullable(Of Boolean)), GetType(ImageSource))>
Public Class BooleanToImageSourceConverter
    Implements IValueConverter
```

**XAML Object Element Usage:**

Εισαγωγή namespace:

```
xmlns:zeus="clr-namespace:Zeus.WPF.Classes.ValueConverters;assembly=ZeusValueConverters"
```

Χρήση:

```
<container.Resources >

    <zeus:BooleanToImageSourceConverter x:Key="booleanToImageSourceConverter"/>

</container.Resources>

<Element property="{Binding sourcePropertyBooleanType,
    Converter={StaticResource booleanToImageSourceConverter}}" />
```

- 
- Για την κλάση **BooleanToStringConverter**:

**Σύνταξη:**

**VB:**

```
Imports System.Windows.Data

<ValueConversion(GetType(Nullable(Of Boolean)), GetType(String))>
Public Class BooleanToStringConverter
    Implements IValueConverter
```

**XAML Object Element Usage:**

Εισαγωγή namespace:

```
xmlns:zeus="clr-namespace:Zeus.WPF.Classes.ValueConverters;assembly=ZeusValueConverters"
```

Χρήση:

```
<container.Resources >  
    <zeus:BooleanToStringConverter x:Key="booleanToStringConverter"/>  
</container.Resources>  
  
<Element property="{Binding sourcePropertyBooleanType,  
    Mode=OneOrTwoWay,  
    Converter={StaticResource booleanToStringConverter}}" />
```

---

- Για την κλάση **BooleanToVisibilityConverter**:

Σύνταξη:

VB:

```
Imports System.Windows.Data
```

```
<ValueConversion(GetType(Nullable(Of Boolean)), GetType(Visibility))>  
Public Class BooleanToVisibilityConverter  
    Implements IValueConverter
```

**XAML Object Element Usage:**

Εισαγωγή namespace:

```
xmlns:zeus="clr-namespace:Zeus.WPF.Classes.ValueConverters;assembly=ZeusValueConverters"
```

Χρήση:

```
<container.Resources >  
    <zeus:BooleanToVisibilityConverter x:Key="booleanToVisibilityConverter"/>  
</container.Resources>  
  
<Element property="{Binding sourcePropertyBooleanType,  
    Converter={StaticResource booleanToVisibilityConverter}}" />
```

---



## Διανομή

Κατά τη διανομή, στο φάκελο της εφαρμογής σας πρέπει να αντιγράψετε το **assembly αρχείο ZeusValueConverters.dll**.

## Επικοινωνία

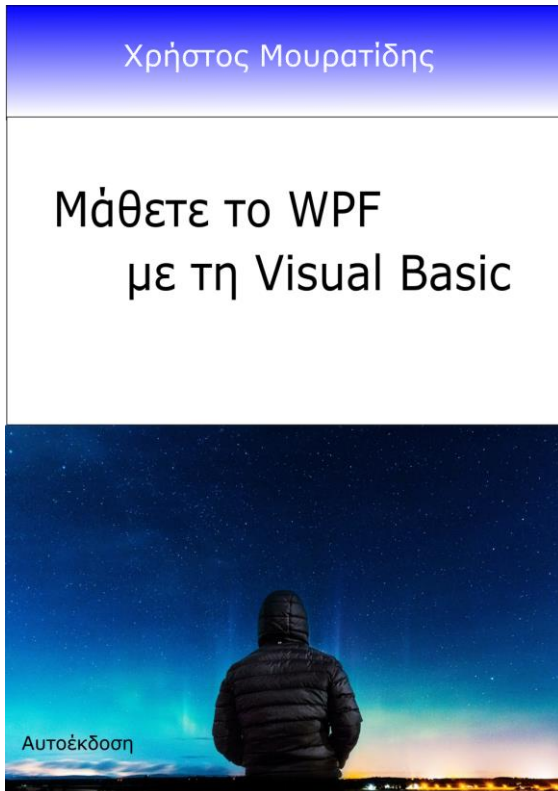
Για οποιαδήποτε πληροφορία ή διευκρίνιση παρακαλώ επικοινωνήστε στο :

[mouratx@yahoo.com](mailto:mouratx@yahoo.com) ή [mouratx@hotmail.com](mailto:mouratx@hotmail.com)



Χρήστος Μουρατίδης,  
Πειραιάς, Αύγουστος 2019

Υ.Γ. Μπορείτε να επικοινωνήσετε μαζί μου για να προμηθευτείτε το **βιβλίο** μου  
"Μάθετε το WPF με τη Visual Basic" (1.333 σελίδες, Αυτοέκδοση 2018).





## BooleanReverseConverter

Μετατρέπει μία **boolean** τιμή στο αντίθετό της. Υποστηρίζει **two-way** binding mode.

### Σύνταξη:

#### VB:

```
Imports System.Windows.Data

<ValueConversion(GetType(Nullable(Of Boolean)), GetType(Nullable(Of Boolean)))>
Public Class BooleanReverseConverter
    Implements IValueConverter
```

#### XAML Object Element Usage:

Εισαγωγή namespace:

```
xmlns:zeus="clr-namespace:Zeus.WPF.Classes.ValueConverters;assembly=ZeusValueConverters"
```

Χρήση:

```
<container.Resources >

    <zeus:BooleanReverseConverter x:Key="booleanReverseConverter"/>

</container.Resources>

<Element property="{Binding sourcePropertyBooleanType,
    Mode=OneOrTwoWay,
    Converter={StaticResource booleanReverseConverter}}" />
```

### Παράδειγμα:

Στο παράδειγμα που ακολουθεί, έχουμε εισάγει, ως resource, ένα BooleanReverseConverter αντικείμενο. Παρακάτω, υπάρχει μία φόρμα μαθητή, όπου μεταξύ άλλων ιδιοτήτων του source αντικειμένου Student είναι η ιδιότητα IsActive, τύπου Boolean. Αυτή η ιδιότητα υποδηλώνει ότι ο μαθητής είναι ενεργός (δεν έχει διακόψει τη φοίτησή του). Θέλουμε όταν η **ιδιότητα IsActive** είναι **True**, τότε η **ιδιότητα IsChecked** του **CheckBox** να γίνεται **False**. Στην σύνδεση με την binding έκφραση, βλέπουμε την αντιστροφή που κάνει το **BooleanReverseConverter** αντικείμενο.

```
<Window x:Class="MainWindow"
  xmlns="http://schemas.microsoft.com/winfx/2006/xaml/presentation"
  xmlns:x="http://schemas.microsoft.com/winfx/2006/xaml"
  xmlns:d="http://schemas.microsoft.com/expression/blend/2008"
  xmlns:mc="http://schemas.openxmlformats.org/markup-compatibility/2006"
  xmlns:zeus="clr-namespace:Zeus.WPF.Classes.ValueConverters;assembly=ZeusValueCo
nverters"
  mc:Ignorable="d"
  Title="ValueConverters Sample Project" Height="467.775" Width="799.064"
  WindowStartupLocation="CenterScreen" >

  <Window.Resources >

    <!-- ValueConverter -->
    <zeus:BooleanReverseConverter x:Key="booleanReverseConverter"/>

  </Window.Resources>

  <Grid Margin="10" Name="grdStudent">

    <!-- Title-->
    <TextBlock Text="Μαθητής" />

    ...

    <!-- Ενεργός μαθητής. Source ιδιότητα IsActive. Αν ο μαθητής είναι ενεργός
να μην διαγραφεί. -->
    <StackPanel Orientation="Horizontal" Grid.Row="7" Grid.Column="1" >

      <TextBlock Text="Να διαγραφεί;" />
      <CheckBox IsChecked="{Binding IsActive, Mode=OneWay,
        Converter={StaticResource booleanReverseConverter }}"
        Margin="15,0,0,0"/>

    </StackPanel>
  </Grid>

</Window>
```





## BooleanToBitConverter

Μετατρέπει μία **boolean** τιμή σε **0** ή **1**. Υποστηρίζει **two-way** binding mode.

### Σύνταξη:

#### VB:

```
Imports System.Windows.Data

<ValueConversion(GetType(Nullable(Of Boolean)), GetType(Byte))>
Public Class BooleanToBitConverter
    Implements IValueConverter
```

#### XAML Object Element Usage:

Εισαγωγή namespace:

```
xmlns:zeus="clr-namespace:Zeus.WPF.Classes.ValueConverters;assembly=ZeusValueConverters"
```

Χρήση:

```
<container.Resources >

    <zeus:BooleanToBitConverter x:Key="booleanToBitConverter"/>

</container.Resources>

<Element property="{Binding sourcePropertyBooleanType,
                           Mode=OneOrTwoWay,
                           Converter={StaticResource booleanToBitConverter}}" />
```

### Παρατηρήσεις:

Η κλάση BooleanToBitConverter, εξ' ορισμού μετατρέπει την τιμή True σε 1 και την τιμή False σε 0. Αν θέλουμε να συμβαίνει το αντίθετο, τότε θέτουμε, την **ιδιότητα BitWhenTrue** σε 0 και την **ιδιότητα BitWhenFalse** σε 1.

```
<container.Resources >

    <zeus:BooleanToBitConverter x:Key="booleanToBitConverter"
```

```
BitWhenTrue="0" BitWhenFalse="1" />
```

```
</container.Resources>
```

## Παράδειγμα:

Στο παράδειγμα που ακολουθεί, έχουμε εισάγει, ως resource, ένα BooleanToBitConverter αντικείμενο. Παρακάτω, υπάρχει μία φόρμα μαθητή, όπου μεταξύ άλλων ιδιοτήτων του source αντικειμένου Student είναι η ιδιότητα IsActive, τύπου Boolean. Αυτή η ιδιότητα υποδηλώνει ότι ο μαθητής είναι ενεργός (δεν έχει διακόψει τη φοίτησή του). Θέλουμε όταν η **ιδιότητα IsActive** είναι **True**, τότε η **ιδιότητα Text** του **TextBlock** να γίνεται **1**, αλλιώς **0**. Στην σύνδεση με την binding έκφραση, την δουλειά αυτή κάνει το **BooleanToBitConverter** αντικείμενο.

```
<Window x:Class="MainWindow"
  xmlns="http://schemas.microsoft.com/winfx/2006/xaml/presentation"
  xmlns:x="http://schemas.microsoft.com/winfx/2006/xaml"
  xmlns:d="http://schemas.microsoft.com/expression/blend/2008"
  xmlns:mc="http://schemas.openxmlformats.org/markup-compatibility/2006"
  xmlns:zeus="clr-namespace:Zeus.WPF.Classes.ValueConverters;assembly=ZeusValueCo
nverters"
  mc:Ignorable="d"
  Title="ValueConverters Sample Project" Height="467.775" Width="799.064"
  WindowStartupLocation="CenterScreen" >

  <Window.Resources >

    <!-- ValueConverter -->
    <zeus:BooleanToBitConverter x:Key="booleanToBitConverter"/>

  </Window.Resources>

  <Grid Margin="10" Name="grdStudent">

    <!-- Title-->
    <TextBlock Text="Μαθητής" />

    ...

    <!-- Ενεργός μαθητής. Source ιδιότητα IsActive. Αν ο μαθητής είναι ενεργός
να εμφανίζει την τιμή 1. -->
    <StackPanel Orientation="Horizontal" Grid.Row="7" Grid.Column="1" >

      <TextBlock Text="1 ή 0; " />
      <TextBlock Text="{Binding IsActive, Mode=OneWay,
        Converter={StaticResource booleanToBitConverter }}"
        Margin="15,0,0,0" />

    </StackPanel>

  </Grid>

</Window>
```

## Ιδιότητες

Όνομα	Περιγραφή
BitWhenTrue	Καθορίζει την τιμή Bit (1 ή 0) όταν η source boolean τιμή είναι <b>True</b> .
BitWhenFalse	Καθορίζει την τιμή Bit (1 ή 0) όταν η source boolean τιμή είναι <b>False</b> .

## BitWhenTrue

Καθορίζει την τιμή Bit (1 ή 0) όταν η source boolean τιμή είναι True.

### Σύνταξη:

#### VB:

```
Public Property BitWhenTrue As Byte = 1
```

#### Τύπος: System.Byte

Προσδιορίζουμε μία τιμή Byte (1 ή 0) που θα αποτελεί την μετατρεπόμενη τιμή όταν η source boolean τιμή είναι True. Η default τιμή είναι 1.

### Παράδειγμα:

Στο παράδειγμα που ακολουθεί, έχουμε εισάγει, ως resource, ένα BooleanToBitConverter αντικείμενο. Παρακάτω, υπάρχει μία φόρμα μαθητή, όπου μεταξύ άλλων ιδιοτήτων του source αντικειμένου Student είναι η ιδιότητα IsActive, τύπου Boolean. Αυτή η ιδιότητα υποδηλώνει ότι ο μαθητής είναι ενεργός (δεν έχει διακόψει τη φοίτησή του). Θέλουμε όταν η **ιδιότητα IsActive** είναι **True**, τότε η **ιδιότητα Text** του **TextBlock** να γίνεται **0**, αλλιώς **1**. Στην σύνδεση με την binding έκφραση, την δουλειά αυτή κάνει το **BooleanToBitConverter** αντικείμενο έχοντας προσδιορίσει τις μετατρεπόμενες τιμές στις **ιδιότητες BitWhenTrue** και **BitWhenFalse**.

```
<Window x:Class="MainWindow"
    xmlns="http://schemas.microsoft.com/winfx/2006/xaml/presentation"
    xmlns:x="http://schemas.microsoft.com/winfx/2006/xaml"
    xmlns:d="http://schemas.microsoft.com/expression/blend/2008"
    xmlns:mc="http://schemas.openxmlformats.org/markup-compatibility/2006"
    xmlns:zeus="clr-namespace:Zeus.WPF.Classes.ValueConverters;assembly=ZeusValueCo
nverters"
    mc:Ignorable="d"
    Title="ValueConverters Sample Project" Height="467.775" Width="799.064"
    WindowStartupLocation="CenterScreen" >

    <Window.Resources >

        <!-- ValueConverter -->
        <zeus:BooleanToBitConverter x:Key="booleanToBitConverter"
            BitWhenTrue="0" BitWhenFalse="1"/>

    </Window.Resources >

    <Grid Margin="10" Name="grdStudent">

        <!-- Title-->
        <TextBlock Text="Μαθητής" />
```

```
...

<!-- Ενεργός μαθητής. Source ιδιότητα IsActive. Αν ο μαθητής είναι ενεργός
να εμφανίζει την τιμή 0 αλλιώς την 1. -->
<StackPanel Orientation="Horizontal" Grid.Row="7" Grid.Column="1" >

    <TextBlock Text="1 ή 0; " />
    <TextBlock Text="{Binding IsActive, Mode=OneWay,
        Converter={StaticResource booleanToBitConverter }}"
        Margin="15,0,0,0" />

</StackPanel>

</Grid>

</Window>
```

## BitWhenFalse

Καθορίζει την τιμή Bit (1 ή 0) όταν η source boolean τιμή είναι True.

### Σύνταξη:

#### VB:

```
Public Property BitWhenFalse As Byte = 0
```

#### Τύπος: System.Byte

Προσδιορίζουμε μία τιμή Byte (1 ή 0) που θα αποτελεί την μετατρεπόμενη τιμή όταν η source boolean τιμή είναι False. Η default τιμή είναι 0.

### Παράδειγμα:

Στο παράδειγμα που ακολουθεί, έχουμε εισάγει, ως resource, ένα BooleanToBitConverter αντικείμενο. Παρακάτω, υπάρχει μία φόρμα μαθητή, όπου μεταξύ άλλων ιδιοτήτων του source αντικειμένου Student είναι η ιδιότητα IsActive, τύπου Boolean. Αυτή η ιδιότητα υποδηλώνει ότι ο μαθητής είναι ενεργός (δεν έχει διακόψει τη φοίτησή του). Θέλουμε όταν η **ιδιότητα IsActive** είναι **True**, τότε η **ιδιότητα Text** του **TextBlock** να γίνεται **0**, αλλιώς **1**. Στην σύνδεση με την binding έκφραση, την δουλειά αυτή κάνει το **BooleanToBitConverter** αντικείμενο έχοντας προσδιορίσει τις μετατρεπόμενες τιμές στις **ιδιότητες BitWhenTrue** και **BitWhenFalse**.

```
<Window x:Class="MainWindow"
    xmlns="http://schemas.microsoft.com/winfx/2006/xaml/presentation"
    xmlns:x="http://schemas.microsoft.com/winfx/2006/xaml"
    xmlns:d="http://schemas.microsoft.com/expression/blend/2008"
    xmlns:mc="http://schemas.openxmlformats.org/markup-compatibility/2006"
    xmlns:zeus="clr-namespace:Zeus.WPF.Classes.ValueConverters;assembly=ZeusValueCo
nverters"
    mc:Ignorable="d"
    Title="ValueConverters Sample Project" Height="467.775" Width="799.064"
    WindowStartupLocation="CenterScreen" >

    <Window.Resources >

        <!-- ValueConverter -->
        <zeus:BooleanToBitConverter x:Key="booleanToBitConverter"
            BitWhenTrue="0" BitWhenFalse="1"/>

    </Window.Resources >

    <Grid Margin="10" Name="grdStudent">

        <!-- Title-->
        <TextBlock Text="Μαθητής" />
```

```
...

<!-- Ενεργός μαθητής. Source ιδιότητα IsActive. Αν ο μαθητής είναι ενεργός
να εμφανίζει την τιμή 0 αλλιώς την 1. -->
<StackPanel Orientation="Horizontal" Grid.Row="7" Grid.Column="1" >

    <TextBlock Text="1 ή 0; " />
    <TextBlock Text="{Binding IsActive, Mode=OneWay,
        Converter={StaticResource booleanToBitConverter }}"
        Margin="15,0,0,0" />

</StackPanel>

</Grid>

</Window>
```



## BooleanToBrushConverter

Μετατρέπει μία **boolean** τιμή σε **Brush**. Υποστηρίζει **one-way** binding mode.

### Σύνταξη:

#### VB:

```
Imports System.Windows.Data

<ValueConversion(GetType(Nullable(Of Boolean)), GetType(Brush))>
Public Class BooleanToBrushConverter
    Implements IValueConverter
```

#### XAML Object Element Usage:

Εισαγωγή namespace:

```
xmlns:zeus="clr-namespace:Zeus.WPF.Classes.ValueConverters;assembly=ZeusValueConverters"
```

Χρήση:

```
<container.Resources >

    <zeus:BooleanToBrushConverter x:Key="booleanToBrushConverter"/>

</container.Resources>

<Element property="{Binding sourcePropertyBooleanType,
                          Converter={StaticResource booleanToBrushConverter}}" />
```

### Παρατηρήσεις:

Η κλάση BooleanToBrushConverter, εξ' ορισμού μετατρέπει την τιμή True σε Brushes.Transparent και την τιμή False επίσης σε Brushes.Transparent. Για να προσδιορίσουμε ποιά Brush θα επιστραφεί για την τιμή True θα θέσουμε κατάλληλα την **ιδιότητα BrushWhenTrue**. Παρόμοια, για να προσδιορίσουμε ποιά Brush θα επιστραφεί για την τιμή False θα θέσουμε κατάλληλα την **ιδιότητα BrushWhenFalse**.

```
<container.Resources >
```



```
<zeus:BooleanToBrushConverter x:Key="booleanToBrushConverter"
    BrushWhenTrue="Green" BrushWhenFalse="Red" />
```

```
</container.Resources>
```

## Παράδειγμα:

Στο παράδειγμα που ακολουθεί, έχουμε εισάγει, ως resource, ένα BooleanToBrushConverter αντικείμενο. Παρακάτω, υπάρχει μία φόρμα μαθητή, όπου μεταξύ άλλων ιδιοτήτων του source αντικειμένου Student είναι η ιδιότητα IsActive, τύπου Boolean. Αυτή η ιδιότητα υποδηλώνει ότι ο μαθητής είναι ενεργός (δεν έχει διακόψει τη φοίτησή του). Θέλουμε όταν η **ιδιότητα IsActive** είναι **True**, τότε η **ιδιότητα Foreground** του **TextBlock** να γίνεται **"Green"**, αλλιώς **"Red"**. Στην σύνδεση με την binding έκφραση, την δουλειά αυτή κάνει το **BooleanToBrushConverter** αντικείμενο.

```
<Window x:Class="MainWindow"
    xmlns="http://schemas.microsoft.com/winfx/2006/xaml/presentation"
    xmlns:x="http://schemas.microsoft.com/winfx/2006/xaml"
    xmlns:d="http://schemas.microsoft.com/expression/blend/2008"
    xmlns:mc="http://schemas.openxmlformats.org/markup-compatibility/2006"
    xmlns:zeus="clr-namespace:Zeus.WPF.Classes.ValueConverters;assembly=ZeusValueCo
nverters"
    mc:Ignorable="d"
    Title="ValueConverters Sample Project" Height="467.775" Width="799.064"
    WindowStartupLocation="CenterScreen" >

    <Window.Resources >

        <!-- ValueConverter -->
        <zeus:BooleanToBrushConverter x:Key="booleanToBrushForegroundConverter"
            BrushWhenTrue="Green" BrushWhenFalse="Red"/>

    </Window.Resources >

    <Grid Margin="10" Name="grdStudent">

        <!-- Title-->
        <TextBlock Text="Μαθητής" />

        ...

        <!-- Ενεργός μαθητής. Source ιδιότητα IsActive. Αν ο μαθητής είναι ενεργός
        να έχει πράσινο χρώμα αλλιώς κόκκινο. -->
        <StackPanel Orientation="Horizontal" Grid.Row="7" Grid.Column="1" >

            <TextBlock Text="Σήμανση: " />
            <TextBlock Foreground="{Binding IsActive,
                Converter={StaticResource booleanToBrushForegroundConverter}}"
                Margin="15,0,0,0" />

        </StackPanel>

    </Grid>
```

</Window>

Φυσικά, αν επιθυμούσαμε μία ανάλογη χρωματική σήμανση και για το background θα δημιουργούσαμε άλλο ένα resource BooleanToBrushConverter, για παράδειγμα:

```
<zeus:BooleanToBrushConverter x:Key="booleanToBrushBackgroundConverter"
    BrushWhenTrue="Yellow"
    BrushWhenFalse="Orange"/>
```

και θα θέταμε :

```
<TextBlock Foreground="{Binding IsActive,
    Converter={StaticResource booleanToBrushForegroundConverter}}"

    Background="{Binding IsActive,
    Converter={StaticResource booleanToBrushBackgroundConverter}}"

    Margin="15,0,0,0" />
```

## Ιδιότητες

Όνομα	Περιγραφή
BrushWhenTrue	Καθορίζει το αντικείμενο <b>Brush</b> όταν η <b>source boolean</b> τιμή είναι <b>True</b> .
BrushWhenFalse	Καθορίζει το αντικείμενο <b>Brush</b> όταν η <b>source boolean</b> τιμή είναι <b>False</b> .

## BrushWhenTrue

Καθορίζει το αντικείμενο **Brush** όταν η **source boolean** τιμή είναι **True**.

### Σύνταξη:

#### VB:

```
Public Property BrushWhenTrue As Brush = Brushes.Transparent
```

#### Τύπος: **System.Media.Brush**

Προσδιορίζουμε ένα αντικείμενο **Brush** που θα αποτελεί την μετατρεπόμενη τιμή όταν η **source boolean** τιμή είναι **True**. Η **default** τιμή είναι **Brushes.Transparent**.

### Παράδειγμα:

Στο παράδειγμα που ακολουθεί, έχουμε εισάγει, ως **resource**, ένα **BooleanToBrushConverter** αντικείμενο. Παρακάτω, υπάρχει μία φόρμα μαθητή, όπου μεταξύ άλλων ιδιοτήτων του **source** αντικειμένου **Student** είναι η ιδιότητα **IsActive**, τύπου **Boolean**. Αυτή η ιδιότητα υποδηλώνει ότι ο μαθητής είναι ενεργός (δεν έχει διακόψει τη φοίτησή του). Θέλουμε όταν η **ιδιότητα IsActive** είναι **True**, τότε η **ιδιότητα Foreground** του **TextBlock** να γίνεται **"Green"**, αλλιώς **"Red"**. Στην σύνδεση με την **binding** έκφραση, την δουλειά αυτή κάνει το **BooleanToBrushConverter** αντικείμενο.

```
<Window x:Class="MainWindow"
  xmlns="http://schemas.microsoft.com/winfx/2006/xaml/presentation"
  xmlns:x="http://schemas.microsoft.com/winfx/2006/xaml"
  xmlns:d="http://schemas.microsoft.com/expression/blend/2008"
  xmlns:mc="http://schemas.openxmlformats.org/markup-compatibility/2006"
  xmlns:zeus="clr-namespace:Zeus.WPF.Classes.ValueConverters;assembly=ZeusValueCo
nverters"
  mc:Ignorable="d"
  Title="ValueConverters Sample Project" Height="467.775" Width="799.064"
  WindowStartupLocation="CenterScreen" >

  <Window.Resources >

    <!-- ValueConverter -->
    <zeus:BooleanToBrushConverter x:Key="booleanToBrushForegroundConverter"
      BrushWhenTrue="Green"
      BrushWhenFalse="Red"/>

  </Window.Resources>

  <Grid Margin="10" Name="grdStudent">

    <!-- Title-->
    <TextBlock Text="Μαθητής" />
```

```
...

<!-- Ενεργός μαθητής. Source ιδιότητα IsActive. Αν ο μαθητής είναι ενεργός
να έχει πράσινο χρώμα αλλιώς κόκκινο. -->
<StackPanel Orientation="Horizontal" Grid.Row="7" Grid.Column="1" >

    <TextBlock Text="Σήμανση: " />
    <TextBlock Foreground="{Binding IsActive,
        Converter={StaticResource booleanToBrushForegroundConverter}}"
        Margin="15,0,0,0" />

</StackPanel>

</Grid>

</Window>
```

## BrushWhenFalse

Καθορίζει το αντικείμενο **Brush** όταν η **source boolean** τιμή είναι **False**.

### Σύνταξη:

#### VB:

```
Public Property BrushWhenFalse As Brush = Brushes.Transparent
```

#### Τύπος: **System.Media.Brush**

Προσδιορίζουμε ένα αντικείμενο **Brush** που θα αποτελεί την μετατρεπόμενη τιμή όταν η **source boolean** τιμή είναι **False**. Η **default** τιμή είναι **Brushes.Transparent**.

### Παράδειγμα:

Στο παράδειγμα που ακολουθεί, έχουμε εισάγει, ως **resource**, ένα **BooleanToBrushConverter** αντικείμενο. Παρακάτω, υπάρχει μία φόρμα μαθητή, όπου μεταξύ άλλων ιδιοτήτων του **source** αντικειμένου **Student** είναι η ιδιότητα **IsActive**, τύπου **Boolean**. Αυτή η ιδιότητα υποδηλώνει ότι ο μαθητής είναι ενεργός (δεν έχει διακόψει τη φοίτησή του). Θέλουμε όταν η **ιδιότητα IsActive** είναι **True**, τότε η **ιδιότητα Foreground** του **TextBlock** να γίνεται **"Green"**, αλλιώς **"Red"**. Στην σύνδεση με την **binding** έκφραση, την δουλειά αυτή κάνει το **BooleanToBrushConverter** αντικείμενο.

```
<Window x:Class="MainWindow"
    xmlns="http://schemas.microsoft.com/winfx/2006/xaml/presentation"
    xmlns:x="http://schemas.microsoft.com/winfx/2006/xaml"
    xmlns:d="http://schemas.microsoft.com/expression/blend/2008"
    xmlns:mc="http://schemas.openxmlformats.org/markup-compatibility/2006"
    xmlns:zeus="clr-namespace:Zeus.WPF.Classes.ValueConverters;assembly=ZeusValueCo
nverters"
    mc:Ignorable="d"
    Title="ValueConverters Sample Project" Height="467.775" Width="799.064"
    WindowStartupLocation="CenterScreen" >

    <Window.Resources >

        <!-- ValueConverter -->
        <zeus:BooleanToBrushConverter x:Key="booleanToBrushForegroundConverter"
            BrushWhenTrue="Green"
            BrushWhenFalse="Red"/>

    </Window.Resources>

    <Grid Margin="10" Name="grdStudent">

        <!-- Title-->
        <TextBlock Text="Μαθητής" />
```

```
...

<!-- Ενεργός μαθητής. Source ιδιότητα IsActive. Αν ο μαθητής είναι ενεργός
να έχει πράσινο χρώμα αλλιώς κόκκινο. -->
<StackPanel Orientation="Horizontal" Grid.Row="7" Grid.Column="1" >

    <TextBlock Text="Σήμανση: " />
    <TextBlock Foreground="{Binding IsActive,
        Converter={StaticResource booleanToBrushForegroundConverter}}"
        Margin="15,0,0,0" />

</StackPanel>

</Grid>

</Window>
```



## BooleanToFontSizeConverter

Μετατρέπει μία **boolean** τιμή σε **Double**, που δηλώνει το **font size**. Υποστηρίζει **one-way** binding mode.

### Σύνταξη:

#### VB:

```
Imports System.Windows.Data

<ValueConversion(GetType(Nullable(Of Boolean)), GetType(Double))>
Public Class BooleanToFontSizeConverter
    Implements IValueConverter
```

### XAML Object Element Usage:

Εισαγωγή namespace:

```
xmlns:zeus="clr-namespace:Zeus.WPF.Classes.ValueConverters;assembly=ZeusValueConverters"
```

Χρήση:

```
<container.Resources >

    <zeus:BooleanToFontSizeConverter x:Key="booleanToFontSizeConverter"/>

</container.Resources>

<Element property="{Binding sourcePropertyBooleanType,
                          Converter={StaticResource booleanToFontSizeConverter}}" />
```

### Παρατηρήσεις:

Η κλάση BooleanToFontSizeConverter, εξ' ορισμού μετατρέπει την τιμή True σε 14.0 και την τιμή False σε 12.0. Για να προσδιορίσουμε ποιά FontSize θα επιστραφεί για την τιμή True θα θέσουμε κατάλληλα την **ιδιότητα FontSizeWhenTrue**. Παρόμοια, για να προσδιορίσουμε ποιά FontSize θα επιστραφεί για την τιμή False θα θέσουμε κατάλληλα την **ιδιότητα FontSizeWhenFalse**.



```

<container.Resources >

    <zeus:BooleanToFontSizeConverter x:Key="booleanToFontSizeConverter"
        FontSizeWhenTrue="16.0"
        FontSizeFalse="14.0" />

</container.Resources>

```

## Παράδειγμα:

Στο παράδειγμα που ακολουθεί, έχουμε εισάγει, ως resource, ένα BooleanToFontSizeConverter αντικείμενο. Παρακάτω, υπάρχει μία φόρμα μαθητή, όπου μεταξύ άλλων ιδιοτήτων του source αντικειμένου Student είναι η ιδιότητα IsActive, τύπου Boolean. Αυτή η ιδιότητα υποδηλώνει ότι ο μαθητής είναι ενεργός (δεν έχει διακόψει τη φοίτησή του). Θέλουμε όταν η **ιδιότητα IsActive** είναι **True**, τότε η **ιδιότητα FontSize** του **TextBlock** να γίνεται **16.0**, αλλιώς **14.0**. Στην σύνδεση με την binding έκφραση, την δουλειά αυτή κάνει το **BooleanToFontSizeConverter** αντικείμενο.

```

<Window x:Class="MainWindow"
    xmlns="http://schemas.microsoft.com/winfx/2006/xaml/presentation"
    xmlns:x="http://schemas.microsoft.com/winfx/2006/xaml"
    xmlns:d="http://schemas.microsoft.com/expression/blend/2008"
    xmlns:mc="http://schemas.openxmlformats.org/markup-compatibility/2006"
    xmlns:zeus="clr-namespace:Zeus.WPF.Classes.ValueConverters;assembly=ZeusValueCo
nverters"
    mc:Ignorable="d"
    Title="ValueConverters Sample Project" Height="467.775" Width="799.064"
    WindowStartupLocation="CenterScreen" >

    <Window.Resources >

        <!-- ValueConverter -->
        <zeus:BooleanToFontSizeConverter x:Key="booleanToFontSizeConverter"
            FontSizeWhenTrue="16.0"
            FontSizeWhenFalse="14.0"/>

    </Window.Resources>

    <Grid Margin="10" Name="grdStudent">

        <!-- Title-->
        <TextBlock Text="Μαθητής" />

        ...

        <!-- Ενεργός μαθητής. Source ιδιότητα IsActive. Αν ο μαθητής είναι ενεργός
να έχει μεγαλύτερο font size. -->
        <StackPanel Orientation="Horizontal" Grid.Row="7" Grid.Column="1" >

            <TextBlock Text="Μέγεθος γραμματοσειράς: " />
            <TextBlock FontSize="{Binding IsActive,
                Converter={StaticResource booleanToFontSizeConverter}}"
                Margin="15,0,0,0" />

        </StackPanel>

```

</Grid>

</Window>

## Ιδιότητες

Όνομα	Περιγραφή
FontSizeWhenTrue	Καθορίζει το μέγεθος γραμματοσειράς, τύπου <b>Double</b> , όταν η <b>source boolean</b> τιμή είναι <b>True</b> .
FontSizeWhenFalse	Καθορίζει το μέγεθος γραμματοσειράς, τύπου <b>Double</b> , όταν η <b>source boolean</b> τιμή είναι <b>False</b> .

## FontSizeWhenTrue

Καθορίζει το μέγεθος γραμματοσειράς, τύπου `Double`, όταν η `source boolean` τιμή είναι `True`.

### Σύνταξη:

#### VB:

```
Public Property FontSizeWhenTrue As Double = 14.0
```

#### Τύπος: `System.Double`

Προσδιορίζουμε ένα μέγεθος γραμματοσειράς (`FontSize`) που θα αποτελεί την μετατρεπόμενη τιμή όταν η `source boolean` τιμή είναι `True`. Η `default` τιμή είναι `14.0`.

### Παράδειγμα:

Στο παράδειγμα που ακολουθεί, έχουμε εισάγει, ως `resource`, ένα `BooleanToFontSizeConverter` αντικείμενο. Παρακάτω, υπάρχει μία φόρμα μαθητή, όπου μεταξύ άλλων ιδιοτήτων του `source` αντικειμένου `Student` είναι η ιδιότητα `IsActive`, τύπου `Boolean`. Αυτή η ιδιότητα υποδηλώνει ότι ο μαθητής είναι ενεργός (δεν έχει διακόψει τη φοίτησή του). Θέλουμε όταν η ιδιότητα `IsActive` είναι `True`, τότε η ιδιότητα `FontSize` του `TextBlock` να γίνεται **16.0**, αλλιώς **14.0**. Στην σύνδεση με την `binding` έκφραση, την δουλειά αυτή κάνει το `BooleanToFontSizeConverter` αντικείμενο.

```
<Window x:Class="MainWindow"
    xmlns="http://schemas.microsoft.com/winfx/2006/xaml/presentation"
    xmlns:x="http://schemas.microsoft.com/winfx/2006/xaml"
    xmlns:d="http://schemas.microsoft.com/expression/blend/2008"
    xmlns:mc="http://schemas.openxmlformats.org/markup-compatibility/2006"
    xmlns:zeus="clr-namespace:Zeus.WPF.Classes.ValueConverters;assembly=ZeusValueCo
nverters"
    mc:Ignorable="d"
    Title="ValueConverters Sample Project" Height="467.775" Width="799.064"
    WindowStartupLocation="CenterScreen" >

    <Window.Resources >

        <!-- ValueConverter -->
        <zeus:BooleanToFontSizeConverter x:Key="booleanToFontSizeConverter"
            FontSizeWhenTrue="16.0"
            FontSizeWhenFalse="14.0"/>

    </Window.Resources>

    <Grid Margin="10" Name="grdStudent">

        <!-- Title-->
        <TextBlock Text="Μαθητής" />
```

```
...

<!-- Ενεργός μαθητής. Source ιδιότητα IsActive. Αν ο μαθητής είναι ενεργός
να έχει μεγαλύτερο font size. -->
<StackPanel Orientation="Horizontal" Grid.Row="7" Grid.Column="1" >

    <TextBlock Text="Μέγεθος γραμματοσειράς: " />
    <TextBlock FontSize="{Binding IsActive,
        Converter={StaticResource booleanToFontSizeConverter}}"
        Margin="15,0,0,0" />

</StackPanel>

</Grid>

</Window>
```

## FontSizeWhenFalse

Καθορίζει το μέγεθος γραμματοσειράς, τύπου `Double`, όταν η `source boolean` τιμή είναι `False`.

### Σύνταξη:

#### VB:

```
Public Property FontSizeWhenFalse As Double = 12.0
```

#### Τύπος: `System.Double`

Προσδιορίζουμε ένα μέγεθος γραμματοσειράς (`FontSize`) που θα αποτελεί την μετατρεπόμενη τιμή όταν η `source boolean` τιμή είναι `True`. Η `default` τιμή είναι `12.0`.

### Παράδειγμα:

Στο παράδειγμα που ακολουθεί, έχουμε εισάγει, ως `resource`, ένα `BooleanToFontSizeConverter` αντικείμενο. Παρακάτω, υπάρχει μία φόρμα μαθητή, όπου μεταξύ άλλων ιδιοτήτων του `source` αντικειμένου `Student` είναι η ιδιότητα `IsActive`, τύπου `Boolean`. Αυτή η ιδιότητα υποδηλώνει ότι ο μαθητής είναι ενεργός (δεν έχει διακόψει τη φοίτησή του). Θέλουμε όταν η ιδιότητα `IsActive` είναι `True`, τότε η ιδιότητα `FontSize` του `TextBlock` να γίνεται **16.0**, αλλιώς **14.0**. Στην σύνδεση με την `binding` έκφραση, την δουλειά αυτή κάνει το `BooleanToFontSizeConverter` αντικείμενο.

```
<Window x:Class="MainWindow"
    xmlns="http://schemas.microsoft.com/winfx/2006/xaml/presentation"
    xmlns:x="http://schemas.microsoft.com/winfx/2006/xaml"
    xmlns:d="http://schemas.microsoft.com/expression/blend/2008"
    xmlns:mc="http://schemas.openxmlformats.org/markup-compatibility/2006"
    xmlns:zeus="clr-namespace:Zeus.WPF.Classes.ValueConverters;assembly=ZeusValueCo
nverters"
    mc:Ignorable="d"
    Title="ValueConverters Sample Project" Height="467.775" Width="799.064"
    WindowStartupLocation="CenterScreen" >

    <Window.Resources >

        <!-- ValueConverter -->
        <zeus:BooleanToFontSizeConverter x:Key="booleanToFontSizeConverter"
            FontSizeWhenTrue="16.0"
            FontSizeWhenFalse="14.0"/>

    </Window.Resources>

    <Grid Margin="10" Name="grdStudent">

        <!-- Title-->
        <TextBlock Text="Μαθητής" />
```

```
...

<!-- Ενεργός μαθητής. Source ιδιότητα IsActive. Αν ο μαθητής είναι ενεργός
να έχει μεγαλύτερο font size. -->
<StackPanel Orientation="Horizontal" Grid.Row="7" Grid.Column="1" >

    <TextBlock Text="Μέγεθος γραμματοσειράς: " />
    <TextBlock FontSize="{Binding IsActive,
        Converter={StaticResource booleanToFontSizeConverter}}"
        Margin="15,0,0,0" />

</StackPanel>

</Grid>

</Window>
```



## BooleanToFontStyleConverter

Μετατρέπει μία **boolean** τιμή σε **FontStyle**. Υποστηρίζει **one-way** binding mode.

### Σύνταξη:

#### VB:

```
Imports System.Windows.Data

<ValueConversion(GetType(Nullable(Of Boolean)), GetType(FontStyle))>
Public Class BooleanToFontStyleConverter
    Implements IValueConverter
```

#### XAML Object Element Usage:

Εισαγωγή namespace:

```
xmlns:zeus="clr-namespace:Zeus.WPF.Classes.ValueConverters;assembly=ZeusValueConverters"
```

Χρήση:

```
<container.Resources >

    <zeus:BooleanToFontStyleConverter x:Key="booleanToFontStyleConverter"/>

</container.Resources>

<Element property="{Binding sourcePropertyBooleanType,
    Converter={StaticResource booleanToFontStyleConverter}}"/>
```

### Παρατηρήσεις:

Η κλάση BooleanToFontStyleConverter, εξ' ορισμού μετατρέπει την τιμή True σε FontStyles.Italic και την τιμή False σε FontStyles.Normal. Για να προσδιορίσουμε ποιά FontStyle θα επιστραφεί για την τιμή True θα θέσουμε κατάλληλα την **ιδιότητα FontStyleWhenTrue**. Παρόμοια, για να προσδιορίσουμε ποιά FontStyle θα επιστραφεί για την τιμή False θα θέσουμε κατάλληλα την **ιδιότητα FontStyleWhenFalse**.

```
<container.Resources >
```



```

<zeus:BooleanToFontStyleConverter x:Key="booleanToFontStyleConverter"
    FontStyleWhenTrue="Oblique"
    FontStyleFalse="Normal" />

</container.Resources>

```

Σε γενικές γραμμές, δεν χρειάζεται να προσδιορίσουμε αυτές τις ιδιότητες αφού τα font styles είναι περιορισμένα: Italic, Oblique και Normal. Συνεπώς, οι default τιμές τους είναι αρκετές.

## Παράδειγμα:

Στο παράδειγμα που ακολουθεί, έχουμε εισάγει, ως resource, ένα BooleanToFontStyleConverter αντικείμενο. Παρακάτω, υπάρχει μία φόρμα μαθητή, όπου μεταξύ άλλων ιδιοτήτων του source αντικειμένου Student είναι η ιδιότητα IsActive, τύπου Boolean. Αυτή η ιδιότητα υποδηλώνει ότι ο μαθητής είναι ενεργός (δεν έχει διακόψει τη φοίτησή του). Θέλουμε όταν η **ιδιότητα IsActive** είναι **True**, τότε η **ιδιότητα FontStyle** του **TextBlock** να γίνεται **"Oblique"**, αλλιώς **"Normal"**. Στην σύνδεση με την binding έκφραση, την δουλειά αυτή κάνει το **BooleanToFontStyleConverter** αντικείμενο. Να σημειώσουμε ότι την ιδιότητα FontStyleWhenFalse θα μπορούσαμε να την παραλείψουμε αφού ήδη η default τιμή είναι "Normal".

```

<Window x:Class="MainWindow"
    xmlns="http://schemas.microsoft.com/winfx/2006/xaml/presentation"
    xmlns:x="http://schemas.microsoft.com/winfx/2006/xaml"
    xmlns:d="http://schemas.microsoft.com/expression/blend/2008"
    xmlns:mc="http://schemas.openxmlformats.org/markup-compatibility/2006"
    xmlns:zeus="clr-namespace:Zeus.WPF.Classes.ValueConverters;assembly=ZeusValueCo
nverters"
    mc:Ignorable="d"
    Title="ValueConverters Sample
Project" Height="467.775" Width="799.064" WindowStartupLocation="CenterScreen" >

    <Window.Resources >

        <!-- ValueConverter -->
        <zeus:BooleanToFontStyleConverter x:Key="booleanToFontStyleConverter"
            FontStyleWhenTrue="Oblique"
            FontStyleWhenFalse="Normal"/>

    </Window.Resources >

    <Grid Margin="10" Name="grdStudent">

        <!-- Title-->
        <TextBlock Text="Μαθητής" />

        ...

        <!-- Ενεργός μαθητής. Source ιδιότητα IsActive. Αν ο μαθητής είναι ενεργός
να έχει font style "Oblique". -->
        <StackPanel Orientation="Horizontal" Grid.Row="7" Grid.Column="1" >

```

```
<TextBlock Text="Στυλ κειμένου: " />
<TextBlock FontStyle="{Binding IsActive,
Converter={StaticResource booleanToFontStyleConverter}}"
Margin="15,0,0,0" />

</StackPanel>

</Grid>

</Window>
```

## Ιδιότητες

Όνομα	Περιγραφή
FontStyleWhenTrue	Καθορίζει το <b>FontStyle</b> όταν η <b>source boolean</b> τιμή είναι <b>True</b> .
FontStyleWhenFalse	Καθορίζει το <b>FontStyle</b> όταν η <b>source boolean</b> τιμή είναι <b>False</b> .

## FontStyleWhenTrue

Καθορίζει το **FontStyle** όταν η **source boolean** τιμή είναι **True**.

### Σύνταξη:

#### VB:

```
Public Property FontStyleWhenTrue As FontStyle = FontStyles.Italic
```

#### Τύπος: **System.Windows.FontStyle**

Προσδιορίζουμε ένα **FontStyle** που θα αποτελεί την μετατρεπόμενη τιμή όταν η **source boolean** τιμή είναι **True**. Η **default** τιμή είναι **FontStyles.Italic**.

### Παράδειγμα:

Στο παράδειγμα που ακολουθεί, έχουμε εισάγει, ως **resource**, ένα **BooleanToFontStyleConverter** αντικείμενο. Παρακάτω, υπάρχει μία φόρμα μαθητή, όπου μεταξύ άλλων ιδιοτήτων του **source** αντικειμένου **Student** είναι η ιδιότητα **IsActive**, τύπου **Boolean**. Αυτή η ιδιότητα υποδηλώνει ότι ο μαθητής είναι ενεργός (δεν έχει διακόψει τη φοίτησή του). Θέλουμε όταν η **ιδιότητα IsActive** είναι **True**, τότε η **ιδιότητα FontStyle** του **TextBlock** να γίνεται **"Oblique"**, αλλιώς **"Normal"**. Στην σύνδεση με την **binding** έκφραση, την δουλειά αυτή κάνει το **BooleanToFontStyleConverter** αντικείμενο. Να σημειώσουμε ότι την ιδιότητα **FontStyleWhenFalse** θα μπορούσαμε να την παραλείψουμε αφού ήδη η **default** τιμή είναι **"Normal"**.

```
<Window x:Class="MainWindow"
    xmlns="http://schemas.microsoft.com/winfx/2006/xaml/presentation"
    xmlns:x="http://schemas.microsoft.com/winfx/2006/xaml"
    xmlns:d="http://schemas.microsoft.com/expression/blend/2008"
    xmlns:mc="http://schemas.openxmlformats.org/markup-compatibility/2006"
    xmlns:zeus="clr-namespace:Zeus.WPF.Classes.ValueConverters;assembly=ZeusValueCo
nverters"
    mc:Ignorable="d"
    Title="ValueConverters Sample Project" Height="467.775" Width="799.064"
    WindowStartupLocation="CenterScreen" >

    <Window.Resources >

        <!-- ValueConverter -->
        <zeus:BooleanToFontStyleConverter x:Key="booleanToFontStyleConverter"
            FontStyleWhenTrue="Oblique"
            FontStyleWhenFalse="Normal"/>

    </Window.Resources >

    <Grid Margin="10" Name="grdStudent">
```

```
<!-- Title-->
<TextBlock Text="Μαθητής" />

...

<!-- Ενεργός μαθητής. Source ιδιότητα IsActive. Αν ο μαθητής είναι ενεργός
να έχει font style "Oblique". -->
<StackPanel Orientation="Horizontal" Grid.Row="7" Grid.Column="1" >

    <TextBlock Text="Στυλ κειμένου: " />
    <TextBlock FontStyle="{Binding IsActive,
        Converter={StaticResource booleanToFontStyleConverter}}"
        Margin="15,0,0,0" />

</StackPanel>

</Grid>

</Window>
```

## FontStyleWhenFalse

Καθορίζει το **FontStyle** όταν η **source boolean** τιμή είναι **False**.

### Σύνταξη:

#### VB:

```
Public Property FontStyleWhenFalse As FontStyle = FontStyles.Normal
```

#### Τύπος: **System.Windows.FontStyle**

Προσδιορίζουμε ένα **FontStyle** που θα αποτελεί την μετατρεπόμενη τιμή όταν η **source boolean** τιμή είναι **False**. Η **default** τιμή είναι **FontStyles.Normal**.

### Παράδειγμα:

Στο παράδειγμα που ακολουθεί, έχουμε εισάγει, ως **resource**, ένα **BooleanToFontStyleConverter** αντικείμενο. Παρακάτω, υπάρχει μία φόρμα μαθητή, όπου μεταξύ άλλων ιδιοτήτων του **source** αντικειμένου **Student** είναι η ιδιότητα **IsActive**, τύπου **Boolean**. Αυτή η ιδιότητα υποδηλώνει ότι ο μαθητής είναι ενεργός (δεν έχει διακόψει τη φοίτησή του). Θέλουμε όταν η **ιδιότητα IsActive** είναι **True**, τότε η **ιδιότητα FontStyle** του **TextBlock** να γίνεται **"Oblique"**, αλλιώς **"Normal"**. Στην σύνδεση με την **binding** έκφραση, την δουλειά αυτή κάνει το **BooleanToFontStyleConverter** αντικείμενο. Να σημειώσουμε ότι την ιδιότητα **FontStyleWhenFalse** θα μπορούσαμε να την παραλείψουμε αφού ήδη η **default** τιμή είναι **"Normal"**.

```
<Window x:Class="MainWindow"
    xmlns="http://schemas.microsoft.com/winfx/2006/xaml/presentation"
    xmlns:x="http://schemas.microsoft.com/winfx/2006/xaml"
    xmlns:d="http://schemas.microsoft.com/expression/blend/2008"
    xmlns:mc="http://schemas.openxmlformats.org/markup-compatibility/2006"
    xmlns:zeus="clr-namespace:Zeus.WPF.Classes.ValueConverters;assembly=ZeusValueCo
nverters"
    mc:Ignorable="d"
    Title="ValueConverters Sample Project" Height="467.775" Width="799.064"
    WindowStartupLocation="CenterScreen" >

    <Window.Resources >

        <!-- ValueConverter -->
        <zeus:BooleanToFontStyleConverter x:Key="booleanToFontStyleConverter"
            FontStyleWhenTrue="Oblique"
            FontStyleWhenFalse="Normal"/>

    </Window.Resources >

    <Grid Margin="10" Name="grdStudent">
```

```
<!-- Title-->
<TextBlock Text="Μαθητής" />

...

<!-- Ενεργός μαθητής. Source ιδιότητα IsActive. Αν ο μαθητής είναι ενεργός
να έχει font style "Oblique". -->
<StackPanel Orientation="Horizontal" Grid.Row="7" Grid.Column="1" >

    <TextBlock Text="Στυλ κειμένου: " />
    <TextBlock FontStyle="{Binding IsActive,
        Converter={StaticResource booleanToFontStyleConverter}}"
        Margin="15,0,0,0" />

</StackPanel>

</Grid>

</Window>
```



## BooleanToFontWeightConverter

Μετατρέπει μία **boolean** τιμή σε **FontWeight**. Υποστηρίζει **one-way** binding mode.

### Σύνταξη:

#### VB:

```
Imports System.Windows.Data

<ValueConversion(GetType(Nullable(Of Boolean)), GetType(FontWeight))>
Public Class BooleanToFontWeightConverter
    Implements IValueConverter
```

#### XAML Object Element Usage:

Εισαγωγή namespace:

```
xmlns:zeus="clr-namespace:Zeus.WPF.Classes.ValueConverters;assembly=ZeusValueConverters"
```

Χρήση:

```
<container.Resources >

    <zeus:BooleanToFontWeightConverter x:Key="booleanToFontWeightConverter"/>

</container.Resources>

<Element property="{Binding sourcePropertyBooleanType,
                          Converter={StaticResource booleanToFontWeightConverter}}" />
```

### Παρατηρήσεις:

Η κλάση BooleanToFontWeightConverter, εξ' ορισμού μετατρέπει την τιμή True σε FontWeights.Bold και την τιμή False σε FontWeights.Normal. Για να προσδιορίσουμε ποιά FontWeight θα επιστραφεί για την τιμή True θα θέσουμε κατάλληλα την **ιδιότητα FontWeightWhenTrue**. Παρόμοια, για να προσδιορίσουμε ποιά FontWeight θα επιστραφεί για την τιμή False θα θέσουμε κατάλληλα την **ιδιότητα FontWeightWhenFalse**.

```
<container.Resources >
```



```

<zeus:BooleanToFontWeightConverter x:Key="booleanToFontWeightConverter"
    FontWeightWhenTrue="ExtraBold"
    FontWeightFalse="ExtraLight" />

</container.Resources>

```

## Παράδειγμα:

Στο παράδειγμα που ακολουθεί, έχουμε εισάγει, ως resource, ένα BooleanToFontWeightConverter αντικείμενο. Παρακάτω, υπάρχει μία φόρμα μαθητή, όπου μεταξύ άλλων ιδιοτήτων του source αντικειμένου Student είναι η ιδιότητα IsActive, τύπου Boolean. Αυτή η ιδιότητα υποδηλώνει ότι ο μαθητής είναι ενεργός (δεν έχει διακόψει τη φοίτησή του). Θέλουμε όταν η **ιδιότητα IsActive** είναι **True**, τότε η **ιδιότητα FontWeight** του **TextBlock** να γίνεται **"ExtraBold"**, αλλιώς **"ExtraLight"**. Στην σύνδεση με την binding έκφραση, την δουλειά αυτή κάνει το **BooleanToFontWeightConverter** αντικείμενο.

```

<Window x:Class="MainWindow"
    xmlns="http://schemas.microsoft.com/winfx/2006/xaml/presentation"
    xmlns:x="http://schemas.microsoft.com/winfx/2006/xaml"
    xmlns:d="http://schemas.microsoft.com/expression/blend/2008"
    xmlns:mc="http://schemas.openxmlformats.org/markup-compatibility/2006"
    xmlns:zeus="clr-namespace:Zeus.WPF.Classes.ValueConverters;assembly=ZeusValueCo
nverters"
    mc:Ignorable="d"
    Title="ValueConverters Sample Project" Height="467.775" Width="799.064"
    WindowStartupLocation="CenterScreen" >

    <Window.Resources >

        <!-- ValueConverter -->
        <zeus:BooleanToFontWeightConverter x:Key="booleanToFontWeightConverter"
            FontWeightWhenTrue="ExtraBold"
            FontWeightWhenFalse="ExtraLight"/>

    </Window.Resources>

    <Grid Margin="10" Name="grdStudent">

        <!-- Title-->
        <TextBlock Text="Μαθητής" />

        ...

        <!-- Ενεργός μαθητής. Source ιδιότητα IsActive. Αν ο μαθητής είναι ενεργός
να έχει πολύ έντονη γραφή αλλιώς πολύ λεπτή γραφή. -->
        <StackPanel Orientation="Horizontal" Grid.Row="7" Grid.Column="1" >

            <TextBlock Text="Ένταση γραφής κειμένου: " />
            <TextBlock FontWeight="{ Binding IsActive,
                Converter={StaticResource booleanToFontWeightConverter}}"
                Margin="15,0,0,0" />

        </StackPanel>

```

</Grid>

</Window>

## Ιδιότητες

Όνομα	Περιγραφή
FontWeightWhenTrue	Καθορίζει το <b>FontWeight</b> όταν η <b>source boolean</b> τιμή είναι <b>True</b> .
FontWeightWhenFalse	Καθορίζει το <b>FontWeight</b> όταν η <b>source boolean</b> τιμή είναι <b>False</b> .

## FontWeightWhenTrue

Καθορίζει το **FontWeight** όταν η **source boolean** τιμή είναι **True**.

### Σύνταξη:

#### VB:

```
Public Property FontWeightWhenTrue As FontWeight = FontWeights.Bold
```

#### Τύπος: **System.Windows.FontWeight**

Προσδιορίζουμε ένα **FontWeight** που θα αποτελεί την μετατρεπόμενη τιμή όταν η **source boolean** τιμή είναι **True**. Η **default** τιμή είναι **FontWeights.Bold**.

### Παράδειγμα:

Στο παράδειγμα που ακολουθεί, έχουμε εισάγει, ως **resource**, ένα **BooleanToFontWeightConverter** αντικείμενο. Παρακάτω, υπάρχει μία φόρμα μαθητή, όπου μεταξύ άλλων ιδιοτήτων του **source** αντικειμένου **Student** είναι η ιδιότητα **IsActive**, τύπου **Boolean**. Αυτή η ιδιότητα υποδηλώνει ότι ο μαθητής είναι ενεργός (δεν έχει διακόψει τη φοίτησή του). Θέλουμε όταν η **ιδιότητα IsActive** είναι **True**, τότε η **ιδιότητα FontWeight** του **TextBlock** να γίνεται **ExtraBold**, αλλιώς **ExtraLight**. Στην σύνδεση με την **binding** έκφραση, την δουλειά αυτή κάνει το **BooleanToFontWeightConverter** αντικείμενο.

```
<Window x:Class="MainWindow"
    xmlns="http://schemas.microsoft.com/winfx/2006/xaml/presentation"
    xmlns:x="http://schemas.microsoft.com/winfx/2006/xaml"
    xmlns:d="http://schemas.microsoft.com/expression/blend/2008"
    xmlns:mc="http://schemas.openxmlformats.org/markup-compatibility/2006"
    xmlns:zeus="clr-namespace:Zeus.WPF.Classes.ValueConverters;assembly=ZeusValueCo
nverters"
    mc:Ignorable="d"
    Title="ValueConverters Sample Project" Height="467.775" Width="799.064"
    WindowStartupLocation="CenterScreen" >

    <Window.Resources >

        <!-- ValueConverter -->
        <zeus:BooleanToFontWeightConverter x:Key="booleanToFontWeightConverter"
            FontWeightWhenTrue="ExtraBold"
            FontWeightWhenFalse="ExtraLight"/>

    </Window.Resources >

    <Grid Margin="10" Name="grdStudent">

        <!-- Title-->
        <TextBlock Text="Μαθητής" />
```

```
...

<!-- Ενεργός μαθητής. Source ιδιότητα IsActive. Αν ο μαθητής είναι ενεργός
να έχει πολύ έντονη γραφή αλλιώς πολύ λεπτή γραφή. -->
<StackPanel Orientation="Horizontal" Grid.Row="7" Grid.Column="1" >

    <TextBlock Text="Ένταση γραφής κειμένου: " />
    <TextBlock FontWeight="{ Binding IsActive,
        Converter={StaticResource booleanToFontWeightConverter}}"
        Margin="15,0,0,0" />

</StackPanel>

</Grid>

</Window>
```

## FontWeightWhenFalse

Καθορίζει το **FontWeight** όταν η source boolean τιμή είναι **False**.

**Σύνταξη:**

**VB:**

```
Public Property FontWeightWhenFalse As FontWeight = FontWeights.Normal
```

**Τύπος:** **System.Windows.FontWeight**

Προσδιορίζουμε ένα **FontWeight** που θα αποτελεί την μετατρεπόμενη τιμή όταν η source boolean τιμή είναι **True**. Η default τιμή είναι **FontWeights.Normal**.

**Παράδειγμα:**

Στο παράδειγμα που ακολουθεί, έχουμε εισάγει, ως resource, ένα **BooleanToFontWeightConverter** αντικείμενο. Παρακάτω, υπάρχει μία φόρμα μαθητή, όπου μεταξύ άλλων ιδιοτήτων του source αντικειμένου **Student** είναι η ιδιότητα **IsActive**, τύπου **Boolean**. Αυτή η ιδιότητα υποδηλώνει ότι ο μαθητής είναι ενεργός (δεν έχει διακόψει τη φοίτησή του). Θέλουμε όταν η **ιδιότητα IsActive** είναι **True**, τότε η **ιδιότητα FontWeight** του **TextBlock** να γίνεται **"ExtraBold"**, αλλιώς **"ExtraLight"**. Στην σύνδεση με την binding έκφραση, την δουλειά αυτή κάνει το **BooleanToFontWeightConverter** αντικείμενο.

```
<Window x:Class="MainWindow"
  xmlns="http://schemas.microsoft.com/winfx/2006/xaml/presentation"
  xmlns:x="http://schemas.microsoft.com/winfx/2006/xaml"
  xmlns:d="http://schemas.microsoft.com/expression/blend/2008"
  xmlns:mc="http://schemas.openxmlformats.org/markup-compatibility/2006"
  xmlns:zeus="clr-namespace:Zeus.WPF.Classes.ValueConverters;assembly=ZeusValueCo
nverters"
  mc:Ignorable="d"
  Title="ValueConverters Sample Project" Height="467.775" Width="799.064"
  WindowStartupLocation="CenterScreen" >

  <Window.Resources >

    <!-- ValueConverter -->
    <zeus:BooleanToFontWeightConverter x:Key="booleanToFontWeightConverter"
      FontWeightWhenTrue="ExtraBold"
      FontWeightWhenFalse="ExtraLight"/>

  </Window.Resources>

  <Grid Margin="10" Name="grdStudent">

    <!-- Title-->
```

```
<TextBlock Text="Μαθητής" />

...

<!-- Ενεργός μαθητής. Source ιδιότητα IsActive. Αν ο μαθητής είναι ενεργός
να έχει πολύ έντονη γραφή αλλιώς πολύ λεπτή γραφή. -->
<StackPanel Orientation="Horizontal" Grid.Row="7" Grid.Column="1" >

    <TextBlock Text="Ένταση γραφής κειμένου: " />
    <TextBlock FontWeight="{ Binding IsActive,
        Converter={StaticResource booleanToFontWeightConverter}}"
        Margin="15,0,0,0" />

</StackPanel>

</Grid>

</Window>
```



## BooleanToImageSourceConverter

Μετατρέπει μία **boolean** τιμή σε **ImageSource**. Υποστηρίζει **one-way** binding mode.

### Σύνταξη:

#### VB:

```
Imports System.Windows.Data
Imports System.Windows.Media

<ValueConversion(GetType(Nullable(Of Boolean)), GetType(ImageSource))>
Public Class BooleanToImageSourceConverter
    Implements IValueConverter
```

#### XAML Object Element Usage:

Εισαγωγή namespace:

```
xmlns:zeus="clr-namespace:Zeus.WPF.Classes.ValueConverters;assembly=ZeusValueConverters"
```

Χρήση:

```
<container.Resources >

    <zeus:BooleanToImageSourceConverter x:Key="booleanToImageSourceConverter"/>

</container.Resources>

<Element property="{Binding sourcePropertyBooleanType,
    Converter={StaticResource booleanToImageSourceConverter}}" />
```

### Παρατηρήσεις:

Η κλάση `BooleanToImageSourceConverter`, εξ' ορισμού μετατρέπει την τιμή `True` σε `Nothing`, όπως και για την τιμή `False`. Για να προσδιορίσουμε ποιά `ImageSource` επιστραφεί για την τιμή `True` θα θέσουμε κατάλληλα την ιδιότητα **`ImageSourceWhenTrue`**. Παρόμοια, για να προσδιορίσουμε ποιά `ImageSource` θα επιστραφεί για την τιμή `False` θα θέσουμε κατάλληλα την ιδιότητα **`ImageSourceWhenFalse`**.



```
<container.Resources >
```

```
<zeus:BooleanToImageSourceConverter x:Key="booleanToImageSourceConverter"
    ImageSourceWhenTrue="Images/Happy.png"
    ImageSourceWhenFalse="Images/Unhappy.png" />
```

```
</container.Resources>
```

Υποθέτουμε ότι στην εφαρμογή έχουμε προσθέσει έναν φάκελο με όνομα *Images* κι έχουμε τοποθετήσει δύο εικόνες: την *Happy.png* και την *Unhappy.png*. Έτσι, οι παραπάνω τιμές στις ιδιότητες είναι το σχετικό *Uri* προς αυτά τα αρχεία.

## Παράδειγμα:

Στο παράδειγμα που ακολουθεί, έχουμε εισάγει, ως resource, ένα `BooleanToImageSourceConverter` αντικείμενο. Παρακάτω, υπάρχει μία φόρμα μαθητή, όπου μεταξύ άλλων ιδιοτήτων του source αντικειμένου `Student` είναι η ιδιότητα `IsActive`, τύπου `Boolean`. Αυτή η ιδιότητα υποδηλώνει ότι ο μαθητής είναι ενεργός (δεν έχει διακόψει τη φοίτησή του). Θέλουμε όταν η **ιδιότητα `IsActive`** είναι **`True`**, τότε η **ιδιότητα `Source`** του **`Image`** να γίνεται **`"Images/Happy.png"`** (χαρούμενο πρόσωπο), αλλιώς **`"Images/Unhappy.png"`** (λυπημένο πρόσωπο). Στην σύνδεση με την binding έκφραση, την δουλειά αυτή κάνει το **`BooleanToImageSourceConverter`** αντικείμενο.

```
<Window x:Class="MainWindow"
    xmlns="http://schemas.microsoft.com/winfx/2006/xaml/presentation"
    xmlns:x="http://schemas.microsoft.com/winfx/2006/xaml"
    xmlns:d="http://schemas.microsoft.com/expression/blend/2008"
    xmlns:mc="http://schemas.openxmlformats.org/markup-compatibility/2006"
    xmlns:zeus="clr-namespace:Zeus.WPF.Classes.ValueConverters;assembly=ZeusValueConverters"
    mc:Ignorable="d"
    Title="ValueConverters Sample Project" Height="467.775" Width="799.064"
    WindowStartupLocation="CenterScreen" >

    <Window.Resources >

        <!-- ValueConverter -->
        <zeus:BooleanToImageSourceConverter x:Key="booleanToImageSourceConverter"
            ImageSourceWhenTrue="Images/Happy.png"
            ImageSourceWhenFalse="Images/Unhappy.png" />

    </Window.Resources >

    <Grid Margin="10" Name="grdStudent">

        <!-- Title-->
        <TextBlock Text="Μαθητής" />

        ...

        <!-- Ενεργός μαθητής. Source ιδιότητα IsActive. Αν ο μαθητής είναι ενεργός
        τότε να εμφανίζει ένα χαρούμενο προσωπάκι αλλιώς ένα λυπημένο. -->
```

```
<StackPanel Orientation="Horizontal" Grid.Row="7" Grid.Column="1" >
    <TextBlock Text="Γραφικό: " />
    <Viewbox Width="50" Height="50" Stretch="Fill" >
        <Image Source="{Binding IsActive,
            Converter={StaticResource booleanToImageSourceConverter}}" />
    </Viewbox>
</StackPanel>
</Grid>
</Window>
```

- Υποθέτουμε ότι στην εφαρμογή έχουμε προσθέσει έναν φάκελο με όνομα Images κι έχουμε τοποθετήσει δύο εικόνες: την Happy.png και την Unhappy.png. Έτσι, οι παραπάνω τιμές στις ιδιότητες είναι το σχετικό Uri προς αυτά τα αρχεία.

## Ιδιότητες

Όνομα	Περιγραφή
ImageSourceWhenTrue	Καθορίζει το <b>ImageSource</b> όταν η <b>source boolean</b> τιμή είναι <b>True</b> .
ImageSourceWhenFalse	Καθορίζει το <b>ImageSource</b> όταν η <b>source boolean</b> τιμή είναι <b>False</b> .

## ImageSourceWhenTrue

Καθορίζει το `ImageSource` όταν η `source` boolean τιμή είναι `True`.

**Σύνταξη:**

**VB:**

```
Public Property ImageSourceWhenTrue As ImageSource = Nothing
```

**Τύπος:** `System.Windows.Media.ImageSource`

Προσδιορίζουμε ένα `ImageSource` που θα αποτελεί την μετατρεπόμενη τιμή όταν η `source` boolean τιμή είναι `True`. Η default τιμή είναι `Nothing`.

**Παράδειγμα:**

Στο παράδειγμα που ακολουθεί, έχουμε εισάγει, ως resource, ένα `BooleanToImageSourceConverter` αντικείμενο. Παρακάτω, υπάρχει μία φόρμα μαθητή, όπου μεταξύ άλλων ιδιοτήτων του `source` αντικειμένου `Student` είναι η ιδιότητα `IsActive`, τύπου `Boolean`. Αυτή η ιδιότητα υποδηλώνει ότι ο μαθητής είναι ενεργός (δεν έχει διακόψει τη φοίτησή του). Θέλουμε όταν η ιδιότητα `IsActive` είναι `True`, τότε η ιδιότητα `Source` του `Image` να γίνεται `"Images/Happy.png"` (χαρούμενο πρόσωπο), αλλιώς `"Images/Unhappy.png"` (λυπημένο πρόσωπο). Στην σύνδεση με την binding έκφραση, την δουλειά αυτή κάνει το `BooleanToImageSourceConverter` αντικείμενο.

```
<Window x:Class="MainWindow"
    xmlns="http://schemas.microsoft.com/winfx/2006/xaml/presentation"
    xmlns:x="http://schemas.microsoft.com/winfx/2006/xaml"
    xmlns:d="http://schemas.microsoft.com/expression/blend/2008"
    xmlns:mc="http://schemas.openxmlformats.org/markup-compatibility/2006"
    xmlns:zeus="clr-namespace:Zeus.WPF.Classes.ValueConverters;assembly=ZeusValueCo
nverters"
    mc:Ignorable="d"
    Title="ValueConverters Sample Project" Height="467.775" Width="799.064"
    WindowStartupLocation="CenterScreen" >

    <Window.Resources >

        <!-- ValueConverter -->
        <zeus:BooleanToImageSourceConverter x:Key="booleanToImageSourceConverter"
            ImageSourceWhenTrue="Images/Happy.png"
            ImageSourceWhenFalse="Images/Unhappy.png"/>

    </Window.Resources >

    <Grid Margin="10" Name="grdStudent">

        <!-- Title-->
        <TextBlock Text="Μαθητής" />
```

...

```
<!-- Ενεργός μαθητής. Source ιδιότητα IsActive. Αν ο μαθητής είναι ενεργός
τότε να εμφανίζει ένα χαρούμενο προσωπάκι αλλιώς ένα λυπημένο. -->
<StackPanel Orientation="Horizontal" Grid.Row="7" Grid.Column="1" >

    <TextBlock Text="Γραφικό: " />
    <Viewbox Width="50" Height="50" Stretch="Fill" >
        <Image Source="{Binding IsActive,
            Converter={StaticResource booleanToImageSourceConverter}}" />
    </Viewbox>

</StackPanel>

</Grid>

</Window>
```

- Υποθέτουμε ότι στην εφαρμογή έχουμε προσθέσει έναν φάκελο με όνομα Images κι έχουμε τοποθετήσει δύο εικόνες: την Happy.png και την Unhappy.png. Έτσι, οι παραπάνω τιμές στις ιδιότητες είναι το σχετικό Uri προς αυτά τα αρχεία.

## ImageSourceWhenFalse

Καθορίζει το `ImageSource` όταν η `source` boolean τιμή είναι `False`.

**Σύνταξη:**

**VB:**

```
Public Property ImageSourceWhenFalse As ImageSource = Nothing
```

**Τύπος:** `System.Windows.Media.ImageSource`

Προσδιορίζουμε ένα `ImageSource` που θα αποτελεί την μετατρεπόμενη τιμή όταν η `source` boolean τιμή είναι `False`. Η default τιμή είναι `Nothing`.

**Παράδειγμα:**

Στο παράδειγμα που ακολουθεί, έχουμε εισάγει, ως resource, ένα `BooleanToImageSourceConverter` αντικείμενο. Παρακάτω, υπάρχει μία φόρμα μαθητή, όπου μεταξύ άλλων ιδιοτήτων του `source` αντικειμένου `Student` είναι η ιδιότητα `IsActive`, τύπου `Boolean`. Αυτή η ιδιότητα υποδηλώνει ότι ο μαθητής είναι ενεργός (δεν έχει διακόψει τη φοίτησή του). Θέλουμε όταν η ιδιότητα `IsActive` είναι `True`, τότε η ιδιότητα `Source` του `Image` να γίνεται `"Images/Happy.png"` (χαρούμενο πρόσωπο), αλλιώς `"Images/Unhappy.png"` (λυπημένο πρόσωπο). Στην σύνδεση με την binding έκφραση, την δουλειά αυτή κάνει το `BooleanToImageSourceConverter` αντικείμενο.

```
<Window x:Class="MainWindow"
    xmlns="http://schemas.microsoft.com/winfx/2006/xaml/presentation"
    xmlns:x="http://schemas.microsoft.com/winfx/2006/xaml"
    xmlns:d="http://schemas.microsoft.com/expression/blend/2008"
    xmlns:mc="http://schemas.openxmlformats.org/markup-compatibility/2006"
    xmlns:zeus="clr-namespace:Zeus.WPF.Classes.ValueConverters;assembly=ZeusValueCo
nverters"
    mc:Ignorable="d"
    Title="ValueConverters Sample Project" Height="467.775" Width="799.064"
    WindowStartupLocation="CenterScreen" >

    <Window.Resources >

        <!-- ValueConverter -->
        <zeus:BooleanToImageSourceConverter x:Key="booleanToImageSourceConverter"
            ImageSourceWhenTrue="Images/Happy.png"
            ImageSourceWhenFalse="Images/Unhappy.png"/>

    </Window.Resources >

    <Grid Margin="10" Name="grdStudent">

        <!-- Title-->
        <TextBlock Text="Μαθητής" />
```

...

```
<!-- Ενεργός μαθητής. Source ιδιότητα IsActive. Αν ο μαθητής είναι ενεργός
τότε να εμφανίζει ένα χαρούμενο προσωπάκι αλλιώς ένα λυπημένο. -->
<StackPanel Orientation="Horizontal" Grid.Row="7" Grid.Column="1" >

    <TextBlock Text="Γραφικό: " />
    <Viewbox Width="50" Height="50" Stretch="Fill" >
        <Image Source="{Binding IsActive,
            Converter={StaticResource booleanToImageSourceConverter}}" />
    </Viewbox>

</StackPanel>

</Grid>

</Window>
```

- Υποθέτουμε ότι στην εφαρμογή έχουμε προσθέσει έναν φάκελο με όνομα Images κι έχουμε τοποθετήσει δύο εικόνες: την Happy.png και την Unhappy.png. Έτσι, οι παραπάνω τιμές στις ιδιότητες είναι το σχετικό Uri προς αυτά τα αρχεία.



## BooleanToStringConverter

Μετατρέπει μία **boolean** τιμή σε **String**. Υποστηρίζει **two-way** binding mode.

### Σύνταξη:

#### VB:

```
Imports System.Windows.Data

<ValueConversion(GetType(Nullable(Of Boolean)), GetType(String))>
Public Class BooleanToStringConverter
    Implements IValueConverter
```

#### XAML Object Element Usage:

Εισαγωγή namespace:

```
xmlns:zeus="clr-namespace:Zeus.WPF.Classes.ValueConverters;assembly=ZeusValueConverters"
```

Χρήση:

```
<container.Resources >

    <zeus:BooleanToStringConverter x:Key="booleanToStringConverter"/>

</container.Resources>

<Element property="{Binding sourcePropertyBooleanType,
    Mode=OneOrTwoWay,
    Converter={StaticResource booleanToStringConverter}}" />
```

### Παρατηρήσεις:

Η κλάση BooleanToStringConverter, εξ' ορισμού μετατρέπει την τιμή True σε "Ναι" και την τιμή False σε "Όχι". Για να προσδιορίσουμε ποιό String θα επιστραφεί για την τιμή True θα θέσουμε κατάλληλα την **ιδιότητα StringWhenTrue**. Παρόμοια, για να προσδιορίσουμε ποιό String θα επιστραφεί για την τιμή False θα θέσουμε κατάλληλα την **ιδιότητα StringWhenFalse**.

```
<container.Resources >
```



```
<zeus:BooleanToStringConverter x:Key="booleanToStringConverter"
    StringWhenTrue="Ισχύει"
    StringFalse="Δεν ισχύει" />
```

```
</container.Resources>
```

## Παράδειγμα:

Στο παράδειγμα που ακολουθεί, έχουμε εισάγει, ως resource, ένα BooleanToStringConverter αντικείμενο. Παρακάτω, υπάρχει μία φόρμα μαθητή, όπου μεταξύ άλλων ιδιοτήτων του source αντικειμένου Student είναι η ιδιότητα IsActive, τύπου Boolean. Αυτή η ιδιότητα υποδηλώνει ότι ο μαθητής είναι ενεργός (δεν έχει διακόψει τη φοίτησή του). Θέλουμε όταν η **ιδιότητα IsActive** είναι **True**, τότε η **ιδιότητα Text** του **TextBlock** να γίνεται **"Είναι ενεργός"**, αλλιώς **"Έχει διακόψει"**. Στην σύνδεση με την binding έκφραση, την δουλειά αυτή κάνει το **BooleanToStringConverter** αντικείμενο.

```
<Window x:Class="MainWindow"
    xmlns="http://schemas.microsoft.com/winfx/2006/xaml/presentation"
    xmlns:x="http://schemas.microsoft.com/winfx/2006/xaml"
    xmlns:d="http://schemas.microsoft.com/expression/blend/2008"
    xmlns:mc="http://schemas.openxmlformats.org/markup-compatibility/2006"
    xmlns:zeus="clr-namespace:Zeus.WPF.Classes.ValueConverters;assembly=ZeusValueCo
nverters"
    mc:Ignorable="d"
    Title="ValueConverters Sample Project" Height="467.775" Width="799.064"
    WindowStartupLocation="CenterScreen" >

    <Window.Resources >

        <!-- ValueConverter -->
        <zeus:BooleanToStringConverter x:Key="booleanToStringConverter"
            StringWhenTrue="Είναι ενεργός"
            StringWhenFalse="Έχει διακόψει"/>

    </Window.Resources>

    <Grid Margin="10" Name="grdStudent">

        <!-- Title-->
        <TextBlock Text="Μαθητής" />

        ...

        <!-- Ενεργός μαθητής. Source ιδιότητα IsActive. Αν ο μαθητής είναι ενεργός
να επιστρέφει το "Είναι ενεργός" αλλιώς "Έχει διακόψει". -->
        <StackPanel Orientation="Horizontal" Grid.Row="7" Grid.Column="1" >

            <TextBlock Text="Κατάσταση μαθητή: " />
            <TextBlock Text="{Binding IsActive,
                Converter={StaticResource booleanToStringConverter}}"
                Margin="15,0,0,0" />

        </StackPanel>
```

</Grid>

</Window>

## Ιδιότητες

Όνομα	Περιγραφή
StringWhenTrue	Καθορίζει το <b>String</b> όταν η <b>source boolean</b> τιμή είναι <b>True</b> .
StringWhenFalse	Καθορίζει το <b>String</b> όταν η <b>source boolean</b> τιμή είναι <b>False</b> .

## StringWhenTrue

Καθορίζει το **String** όταν η **source boolean** τιμή είναι **True**.

### Σύνταξη:

#### VB:

```
Public Property StringWhenTrue As String = "Ναι"
```

#### Τύπος: System.String

Προσδιορίζουμε ένα **String** που θα αποτελεί την μετατρεπόμενη τιμή όταν η **source boolean** τιμή είναι **True**. Η **default** τιμή είναι **"Ναι"**.

### Παράδειγμα:

Στο παράδειγμα που ακολουθεί, έχουμε εισάγει, ως **resource**, ένα **BooleanToStringConverter** αντικείμενο. Παρακάτω, υπάρχει μία φόρμα μαθητή, όπου μεταξύ άλλων ιδιοτήτων του **source** αντικειμένου **Student** είναι η ιδιότητα **IsActive**, τύπου **Boolean**. Αυτή η ιδιότητα υποδηλώνει ότι ο μαθητής είναι ενεργός (δεν έχει διακόψει τη φοίτησή του). Θέλουμε όταν η **ιδιότητα IsActive** είναι **True**, τότε η **ιδιότητα Text** του **TextBlock** να γίνεται **"v"**, αλλιώς **"x"**. Στην σύνδεση με την **binding** έκφραση, την δουλειά αυτή κάνει το **BooleanToStringConverter** αντικείμενο.

```
<Window x:Class="MainWindow"
    xmlns="http://schemas.microsoft.com/winfx/2006/xaml/presentation"
    xmlns:x="http://schemas.microsoft.com/winfx/2006/xaml"
    xmlns:d="http://schemas.microsoft.com/expression/blend/2008"
    xmlns:mc="http://schemas.openxmlformats.org/markup-compatibility/2006"
    xmlns:zeus="clr-namespace:Zeus.WPF.Classes.ValueConverters;assembly=ZeusValueCo
nverters"
    mc:Ignorable="d"
    Title="ValueConverters Sample Project" Height="467.775" Width="799.064"
    WindowStartupLocation="CenterScreen" >

    <Window.Resources >

        <!-- ValueConverter -->
        <zeus:BooleanToStringConverter x:Key="booleanToStringConverter"
            StringWhenTrue="v"
            StringWhenFalse="x"/>

    </Window.Resources>

    <Grid Margin="10" Name="grdStudent">

        <!-- Title-->
        <TextBlock Text="Μαθητής" />
```

```
...

<!-- Ενεργός μαθητής. Source ιδιότητα IsActive. Αν ο μαθητής είναι ενεργός
να επιστρέφει το "v" αλλιώς "x". -->
<StackPanel Orientation="Horizontal" Grid.Row="7" Grid.Column="1" >

    <TextBlock Text="Είναι ενεργός; " />
    <TextBlock Text="{Binding IsActive,
        Converter={StaticResource booleanToStringConverter}}"
        Margin="15,0,0,0" />

</StackPanel>

</Grid>

</Window>
```

## StringWhenFalse

Καθορίζει το **String** όταν η **source boolean** τιμή είναι **False**.

### Σύνταξη:

#### VB:

```
Public Property StringWhenFalse As String = "Όχι"
```

#### Τύπος: **System.String**

Προσδιορίζουμε ένα **String** που θα αποτελεί την μετατρεπόμενη τιμή όταν η **source boolean** τιμή είναι **False**. Η **default** τιμή είναι **"Όχι"**.

### Παράδειγμα:

Στο παράδειγμα που ακολουθεί, έχουμε εισάγει, ως **resource**, ένα **BooleanToStringConverter** αντικείμενο. Παρακάτω, υπάρχει μία φόρμα μαθητή, όπου μεταξύ άλλων ιδιοτήτων του **source** αντικειμένου **Student** είναι η ιδιότητα **IsActive**, τύπου **Boolean**. Αυτή η ιδιότητα υποδηλώνει ότι ο μαθητής είναι ενεργός (δεν έχει διακόψει τη φοίτησή του). Θέλουμε όταν η **ιδιότητα IsActive** είναι **True**, τότε η **ιδιότητα Text** του **TextBlock** να γίνεται **"v"**, αλλιώς **"x"**. Στην σύνδεση με την **binding** έκφραση, την δουλειά αυτή κάνει το **BooleanToStringConverter** αντικείμενο.

```
<Window x:Class="MainWindow"
    xmlns="http://schemas.microsoft.com/winfx/2006/xaml/presentation"
    xmlns:x="http://schemas.microsoft.com/winfx/2006/xaml"
    xmlns:d="http://schemas.microsoft.com/expression/blend/2008"
    xmlns:mc="http://schemas.openxmlformats.org/markup-compatibility/2006"
    xmlns:zeus="clr-namespace:Zeus.WPF.Classes.ValueConverters;assembly=ZeusValueCo
nverters"
    mc:Ignorable="d"
    Title="ValueConverters Sample Project" Height="467.775" Width="799.064"
    WindowStartupLocation="CenterScreen" >

    <Window.Resources >

        <!-- ValueConverter -->
        <zeus:BooleanToStringConverter x:Key="booleanToStringConverter"
            StringWhenTrue="v"
            StringWhenFalse="x"/>

    </Window.Resources>

    <Grid Margin="10" Name="grdStudent">

        <!-- Title-->
        <TextBlock Text="Μαθητής" />
```

```
...

<!-- Ενεργός μαθητής. Source ιδιότητα IsActive. Αν ο μαθητής είναι ενεργός
να επιστρέφει το "v" αλλιώς "x". -->
<StackPanel Orientation="Horizontal" Grid.Row="7" Grid.Column="1" >

    <TextBlock Text="Είναι ενεργός; " />
    <TextBlock Text="{Binding IsActive,
        Converter={StaticResource booleanToStringConverter}}"
        Margin="15,0,0,0" />

</StackPanel>

</Grid>

</Window>
```



## BooleanToVisibilityConverter

Μετατρέπει μία **boolean** τιμή σε **Visibility**. Υποστηρίζει **one-way** binding mode.

### Σύνταξη:

#### VB:

```
Imports System.Windows.Data

<ValueConversion(GetType(Nullable(Of Boolean)), GetType(Visibility))>
Public Class BooleanToVisibilityConverter
    Implements IValueConverter
```

#### XAML Object Element Usage:

Εισαγωγή namespace:

```
xmlns:zeus="clr-namespace:Zeus.WPF.Classes.ValueConverters;assembly=ZeusValueConverters"
```

Χρήση:

```
<container.Resources >

    <zeus:BooleanToVisibilityConverter x:Key="booleanToVisibilityConverter"/>

</container.Resources>

<Element property="{Binding sourcePropertyBooleanType,
    Converter={StaticResource booleanToVisibilityConverter}}" />
```

### Παρατηρήσεις:

Η κλάση BooleanToVisibilityConverter, εξ' ορισμού μετατρέπει την τιμή True σε Visibility.Visible και την τιμή False σε Visibility.Collapsed. Για να προσδιορίσουμε ποιά String θα επιστραφεί για την τιμή True θα θέσουμε κατάλληλα την **ιδιότητα VisibilityWhenTrue**. Παρόμοια, για να προσδιορίσουμε ποιά String θα επιστραφεί για την τιμή False θα θέσουμε κατάλληλα την **ιδιότητα VisibilityWhenFalse**.



```

<container.Resources >

    <zeus:BooleanToVisibilityConverter x:Key="booleanToVisibilityConverter"
        VisibilityWhenTrue="Visible"
        VisibilityWhenFalse="Hidden" />

</container.Resources>

```

## Παράδειγμα:

Στο παράδειγμα που ακολουθεί, έχουμε εισάγει, ως resource, ένα BooleanToVisibilityConverter αντικείμενο. Παρακάτω, υπάρχει μία φόρμα μαθητή, όπου μεταξύ άλλων ιδιοτήτων του source αντικειμένου Student είναι η ιδιότητα IsActive, τύπου Boolean. Αυτή η ιδιότητα υποδηλώνει ότι ο μαθητής είναι ενεργός (δεν έχει διακόψει τη φοίτησή του). Θέλουμε όταν η **ιδιότητα IsActive** είναι **True**, τότε η **ιδιότητα Visibility** του **TextBlock** να γίνεται **"Visible"**, αλλιώς **"Hidden"**. Στην σύνδεση με την binding έκφραση, την δουλειά αυτή κάνει το **BooleanToVisibilityConverter** αντικείμενο.

```

<Window x:Class="MainWindow"
    xmlns="http://schemas.microsoft.com/winfx/2006/xaml/presentation"
    xmlns:x="http://schemas.microsoft.com/winfx/2006/xaml"
    xmlns:d="http://schemas.microsoft.com/expression/blend/2008"
    xmlns:mc="http://schemas.openxmlformats.org/markup-compatibility/2006"
    xmlns:zeus="clr-namespace:Zeus.WPF.Classes.ValueConverters;assembly=ZeusValueCo
nverters"
    mc:Ignorable="d"
    Title="ValueConverters Sample Project" Height="467.775" Width="799.064"
    WindowStartupLocation="CenterScreen" >

    <Window.Resources >

        <!-- ValueConverter -->
        <zeus:BooleanToVisibilityConverter x:Key="booleanToVisibilityConverter"
            VisibilityWhenTrue="Visible"
            VisibilityWhenFalse="Hidden"/>

    </Window.Resources>

    <Grid Margin="10" Name="grdStudent">

        <!-- Title-->
        <TextBlock Text="Μαθητής" />

        ...

        <!-- Ενεργός μαθητής. Source ιδιότητα IsActive. Αν ο μαθητής είναι ενεργός
να εμφανίζεται αλλιώς να αποκρύπτεται. -->
        <StackPanel Orientation="Horizontal" Grid.Row="7" Grid.Column="1" >

            <TextBlock Text="Εμφάνιση στοιχείων μαθητή: " />
            <TextBlock Visibility="{ Binding IsActive,
                Converter={StaticResource booleanToVisibilityConverter}}"
                Margin="15,0,0,0" />

```

```
        </StackPanel>  
    </Grid>  
</Window>
```

## Ιδιότητες

Όνομα	Περιγραφή
VisibilityWhenTrue	Καθορίζει την <b>Visibility</b> όταν η <b>source boolean</b> τιμή είναι <b>True</b> .
VisibilityWhenFalse	Καθορίζει την <b>Visibility</b> όταν η <b>source boolean</b> τιμή είναι <b>False</b> .

## VisibilityWhenTrue

Καθορίζει την **Visibility** όταν η **source boolean** τιμή είναι **True**.

### Σύνταξη:

#### VB:

```
Public Property VisibilityWhenTrue As Visibility = Visibility.Visible
```

#### Τύπος: **System.Windows.Visibility**

Προσδιορίζουμε την **Visibility** που θα αποτελεί την μετατρεπόμενη τιμή όταν η **source boolean** τιμή είναι **True**. Η **default** τιμή είναι **Visibility.Visible**.

### Παράδειγμα:

Στο παράδειγμα που ακολουθεί, έχουμε εισάγει, ως **resource**, ένα **BooleanToVisibilityConverter** αντικείμενο. Παρακάτω, υπάρχει μία φόρμα μαθητή, όπου μεταξύ άλλων ιδιοτήτων του **source** αντικειμένου **Student** είναι η ιδιότητα **IsActive**, τύπου **Boolean**. Αυτή η ιδιότητα υποδηλώνει ότι ο μαθητής είναι ενεργός (δεν έχει διακόψει τη φοίτησή του). Θέλουμε όταν η **ιδιότητα IsActive** είναι **True**, τότε η **ιδιότητα Visibility** του **TextBlock** να γίνεται **"Visible"**, αλλιώς **"Hidden"**. Στην σύνδεση με την **binding** έκφραση, την δουλειά αυτή κάνει το **BooleanToVisibilityConverter** αντικείμενο.

```
<Window x:Class="MainWindow"
    xmlns="http://schemas.microsoft.com/winfx/2006/xaml/presentation"
    xmlns:x="http://schemas.microsoft.com/winfx/2006/xaml"
    xmlns:d="http://schemas.microsoft.com/expression/blend/2008"
    xmlns:mc="http://schemas.openxmlformats.org/markup-compatibility/2006"
    xmlns:zeus="clr-namespace:Zeus.WPF.Classes.ValueConverters;assembly=ZeusValueCo
nverters"
    mc:Ignorable="d"
    Title="ValueConverters Sample Project" Height="467.775" Width="799.064"
    WindowStartupLocation="CenterScreen" >

    <Window.Resources >

        <!-- ValueConverter -->
        <zeus:BooleanToVisibilityConverter x:Key="booleanToVisibilityConverter"
            VisibilityWhenTrue="Visible"
            VisibilityWhenFalse="Hidden"/>

    </Window.Resources>

    <Grid Margin="10" Name="grdStudent">

        <!-- Title-->
        <TextBlock Text="Μαθητής" />
```

```
...

<!-- Ενεργός μαθητής. Source ιδιότητα IsActive. Αν ο μαθητής είναι ενεργός
να εμφανίζεται αλλιώς να αποκρύπτεται. -->
<StackPanel Orientation="Horizontal" Grid.Row="7" Grid.Column="1" >

    <TextBlock Text="Εμφάνιση στοιχείων μαθητή: " />
    <TextBlock Visibility="{ Binding IsActive,
        Converter={StaticResource booleanToVisibilityConverter}}"
        Margin="15,0,0,0" />

</StackPanel>

</Grid>

</Window>
```

- Στο resource BooleanToVisibilityConverter αντικείμενο, θα μπορούσαμε να παραλείψουμε την ιδιότητα VisibilityWhenTrue αφού η default τιμή είναι Visibility.Visible.

## VisibilityWhenFalse

Καθορίζει την **Visibility** όταν η **source boolean** τιμή είναι **False**.

### Σύνταξη:

#### VB:

```
Public Property VisibilityWhenFalse As Visibility = Visibility.Collapsed
```

**Τύπος:** **System.Windows.Visibility**

Προσδιορίζουμε την **Visibility** που θα αποτελεί την μετατρεπόμενη τιμή όταν η **source boolean** τιμή είναι **True**. Η **default** τιμή είναι **Visibility.Collapsed**.

### Παράδειγμα:

Στο παράδειγμα που ακολουθεί, έχουμε εισάγει, ως **resource**, ένα **BooleanToVisibilityConverter** αντικείμενο. Παρακάτω, υπάρχει μία φόρμα μαθητή, όπου μεταξύ άλλων ιδιοτήτων του **source** αντικειμένου **Student** είναι η ιδιότητα **IsActive**, τύπου **Boolean**. Αυτή η ιδιότητα υποδηλώνει ότι ο μαθητής είναι ενεργός (δεν έχει διακόψει τη φοίτησή του). Θέλουμε όταν η **ιδιότητα IsActive** είναι **True**, τότε η **ιδιότητα Visibility** του **TextBlock** να γίνεται **"Visible"**, αλλιώς **"Hidden"**. Στην σύνδεση με την **binding** έκφραση, την δουλειά αυτή κάνει το **BooleanToVisibilityConverter** αντικείμενο.

```
<Window x:Class="MainWindow"
    xmlns="http://schemas.microsoft.com/winfx/2006/xaml/presentation"
    xmlns:x="http://schemas.microsoft.com/winfx/2006/xaml"
    xmlns:d="http://schemas.microsoft.com/expression/blend/2008"
    xmlns:mc="http://schemas.openxmlformats.org/markup-compatibility/2006"
    xmlns:zeus="clr-namespace:Zeus.WPF.Classes.ValueConverters;assembly=ZeusValueCo
nverters"
    mc:Ignorable="d"
    Title="ValueConverters Sample Project" Height="467.775" Width="799.064"
    WindowStartupLocation="CenterScreen" >

    <Window.Resources >

        <!-- ValueConverter -->
        <zeus:BooleanToVisibilityConverter x:Key="booleanToVisibilityConverter"
            VisibilityWhenTrue="Visible"
            VisibilityWhenFalse="Hidden"/>

    </Window.Resources>

    <Grid Margin="10" Name="grdStudent">

        <!-- Title-->
        <TextBlock Text="Μαθητής" />
```

```
...

<!-- Ενεργός μαθητής. Source ιδιότητα IsActive. Αν ο μαθητής είναι ενεργός
να εμφανίζεται αλλιώς να αποκρύπτεται. -->
<StackPanel Orientation="Horizontal" Grid.Row="7" Grid.Column="1" >

    <TextBlock Text="Εμφάνιση στοιχείων μαθητή: " />
    <TextBlock Visibility="{ Binding IsActive,
        Converter={StaticResource booleanToVisibilityConverter}}"
        Margin="15,0,0,0" />

</StackPanel>

</Grid>

</Window>
```

- Στο resource BooleanToVisibilityConverter αντικείμενο, θα μπορούσαμε να παραλείψουμε την ιδιότητα VisibilityWhenTrue αφού η default τιμή είναι Visibility.Visible.

Τέλος Manual