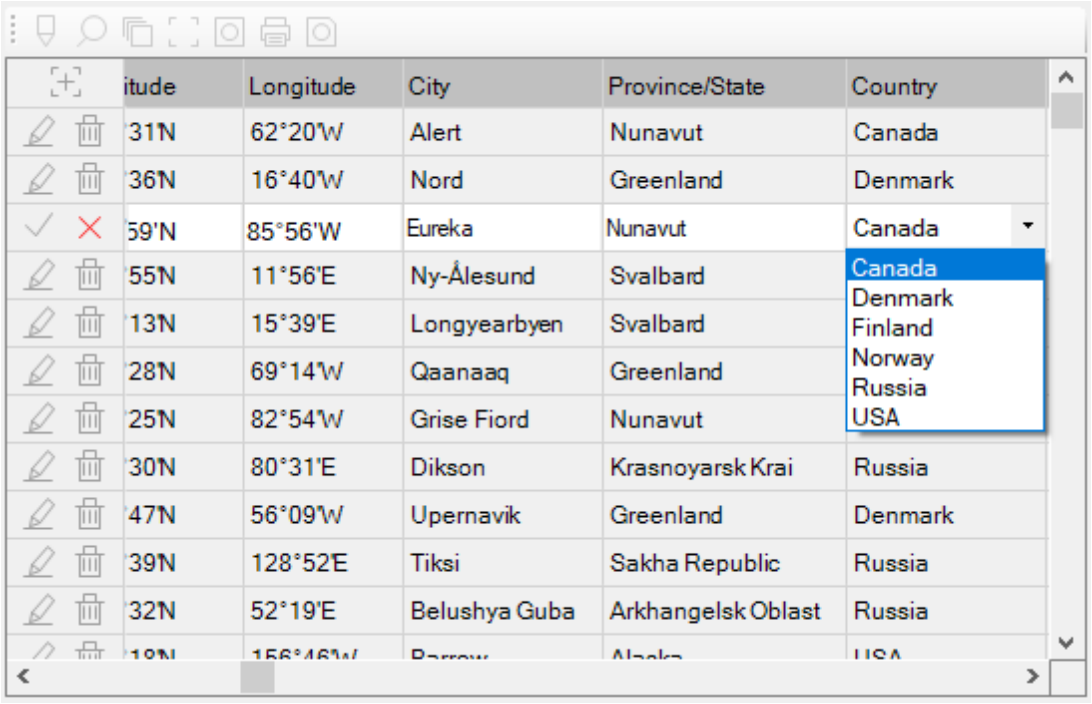
Documentation for MegaGrid .NET WinForms Control

Version 7.0

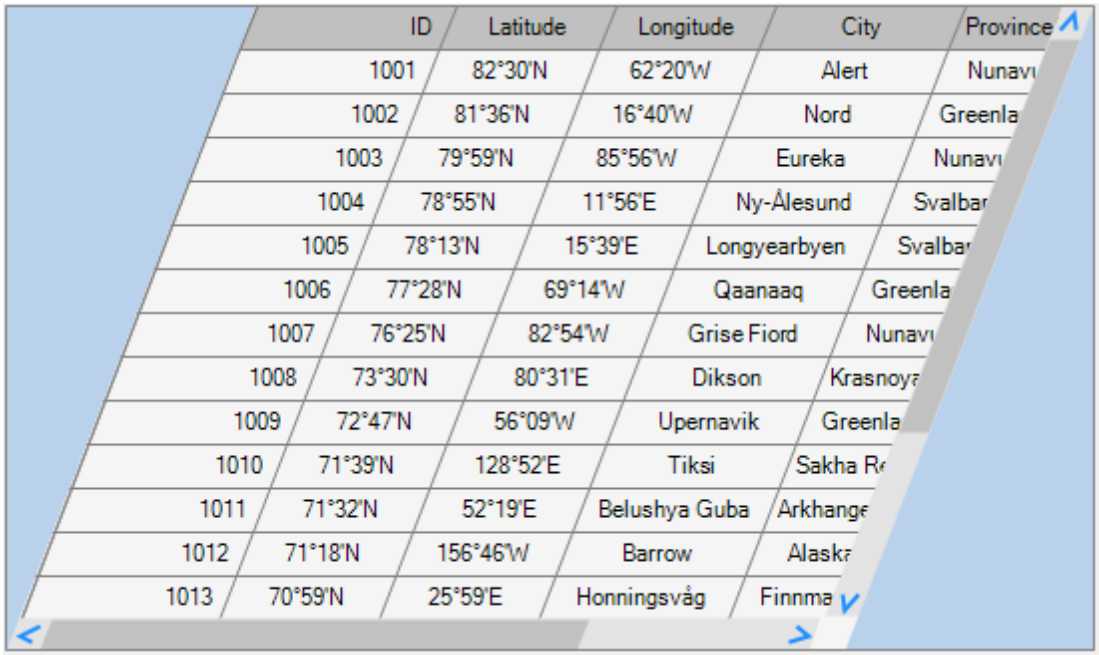
Table of Contents

1 Introduction

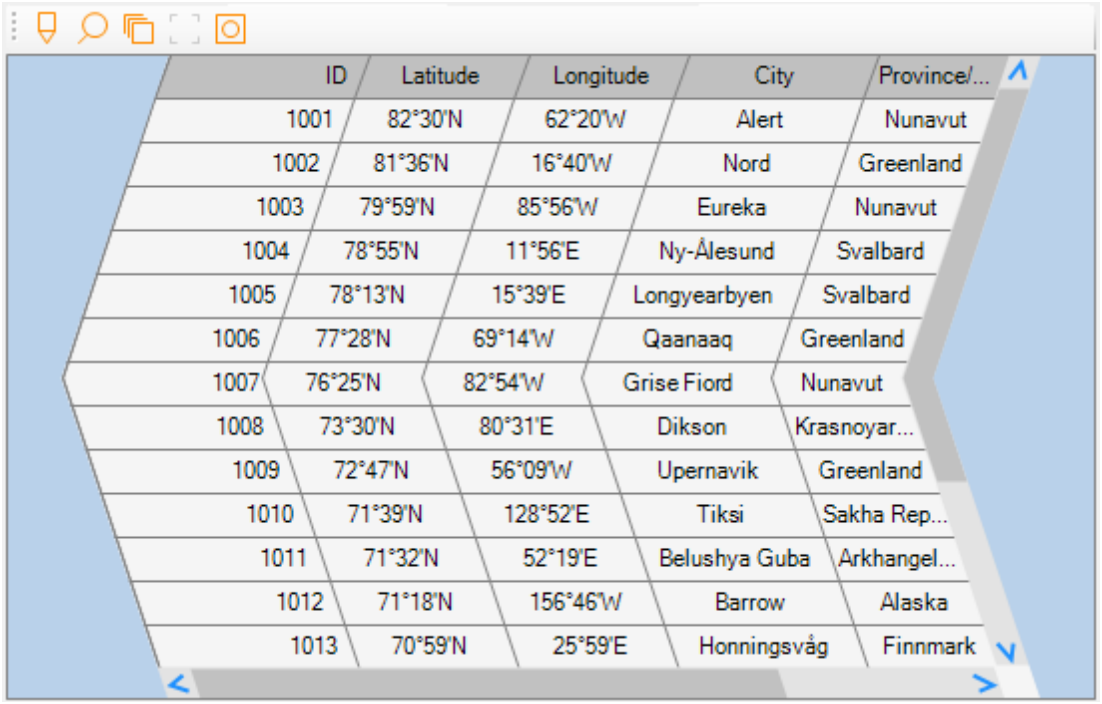
MegaGrid .NET WinForms Control is a versatile, feature-rich, and easy-to-use grid control for the .NET platform, and it allows developers to virtually control every aspect of populating data in the grid. It features 2 essential modes - readonly and edit mode, and you can leverage them to make the most of it and effortlessly meet your project requirements.



1: In Row Editing

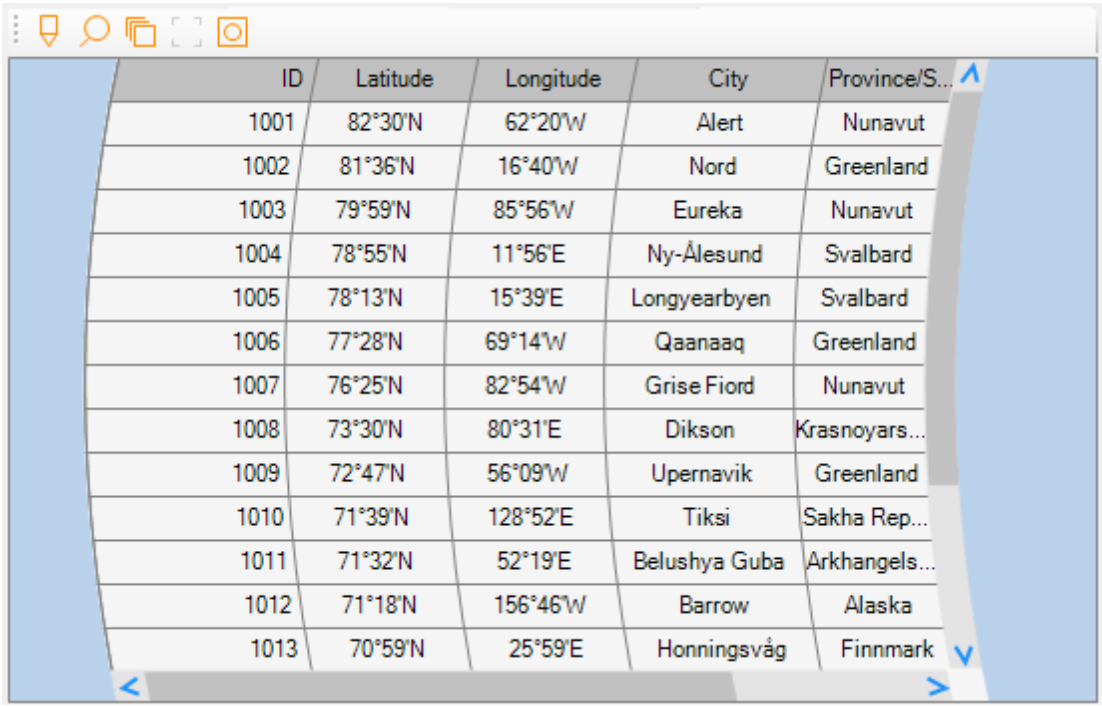


2: Quadrilateral Layout



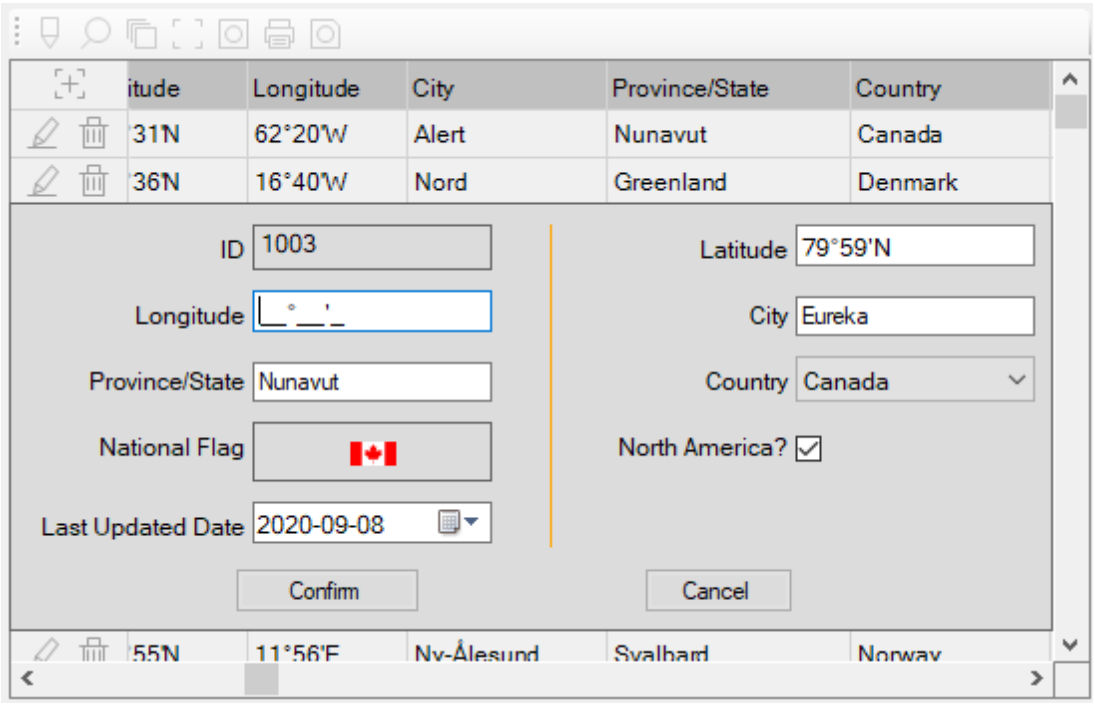
ID	Latitude	Longitude	City	Province/...
1001	82°30'N	62°20'W	Alert	Nunavut
1002	81°36'N	16°40'W	Nord	Greenland
1003	79°59'N	85°56'W	Eureka	Nunavut
1004	78°55'N	11°56'E	Ny-Ålesund	Svalbard
1005	78°13'N	15°39'E	Longyearbyen	Svalbard
1006	77°28'N	69°14'W	Qaanaaq	Greenland
1007	76°25'N	82°54'W	Grise Fiord	Nunavut
1008	73°30'N	80°31'E	Dikson	Krasnoyar...
1009	72°47'N	56°09'W	Upernavik	Greenland
1010	71°39'N	128°52'E	Tiksi	Sakha Rep...
1011	71°32'N	52°19'E	Belushya Guba	Arkhangel...
1012	71°18'N	156°46'W	Barrow	Alaska
1013	70°59'N	25°59'E	Honningsvåg	Finnmark

3: Arrow Layout

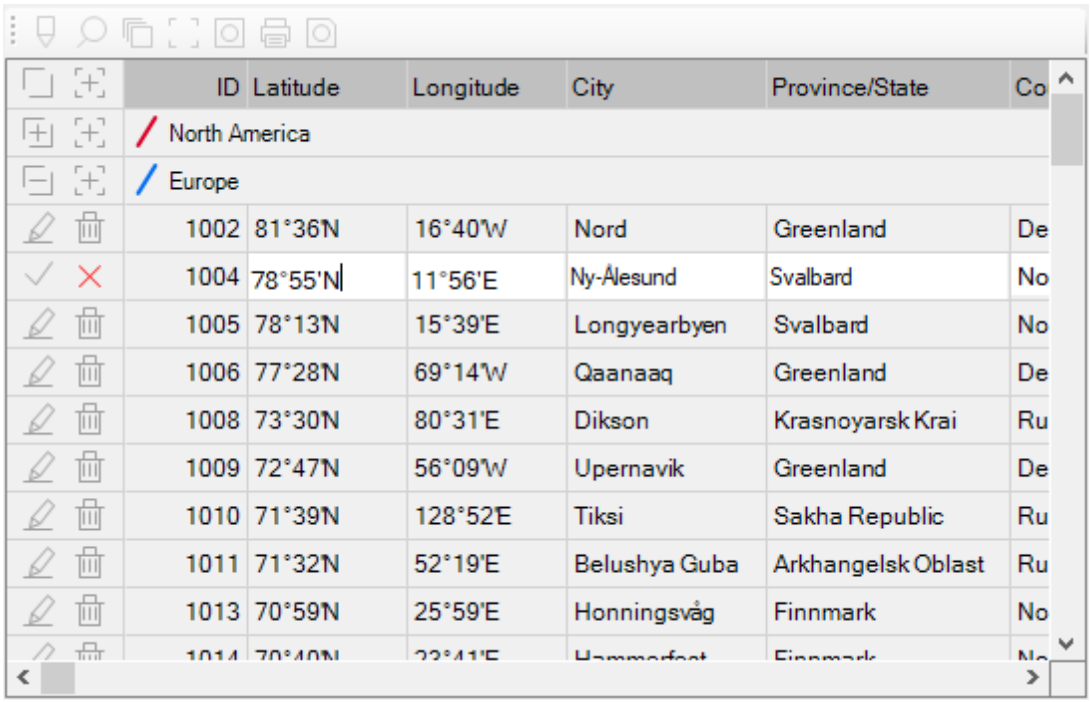


ID	Latitude	Longitude	City	Province/S...
1001	82°30'N	62°20'W	Alert	Nunavut
1002	81°36'N	16°40'W	Nord	Greenland
1003	79°59'N	85°56'W	Eureka	Nunavut
1004	78°55'N	11°56'E	Ny-Ålesund	Svalbard
1005	78°13'N	15°39'E	Longyearbyen	Svalbard
1006	77°28'N	69°14'W	Qaanaaq	Greenland
1007	76°25'N	82°54'W	Grise Fiord	Nunavut
1008	73°30'N	80°31'E	Dikson	Krasnoyars...
1009	72°47'N	56°09'W	Upernavik	Greenland
1010	71°39'N	128°52'E	Tiksi	Sakha Rep...
1011	71°32'N	52°19'E	Belushya Guba	Arkhangels...
1012	71°18'N	156°46'W	Barrow	Alaska
1013	70°59'N	25°59'E	Honningsvåg	Finnmark

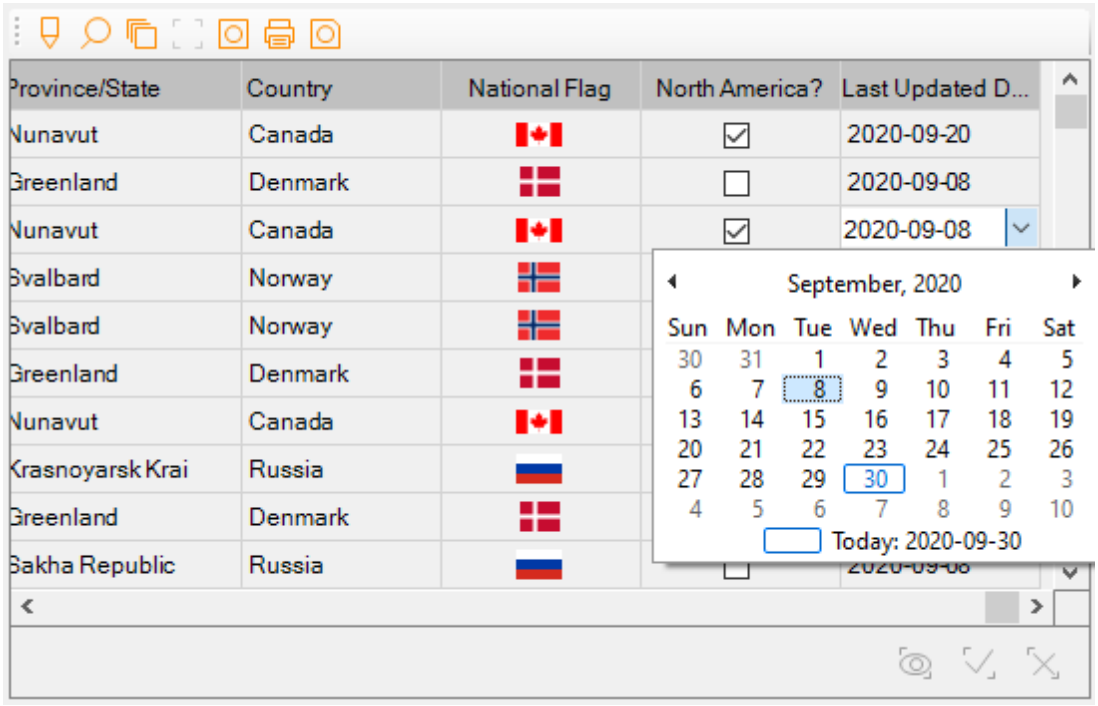
4: Curved Layout



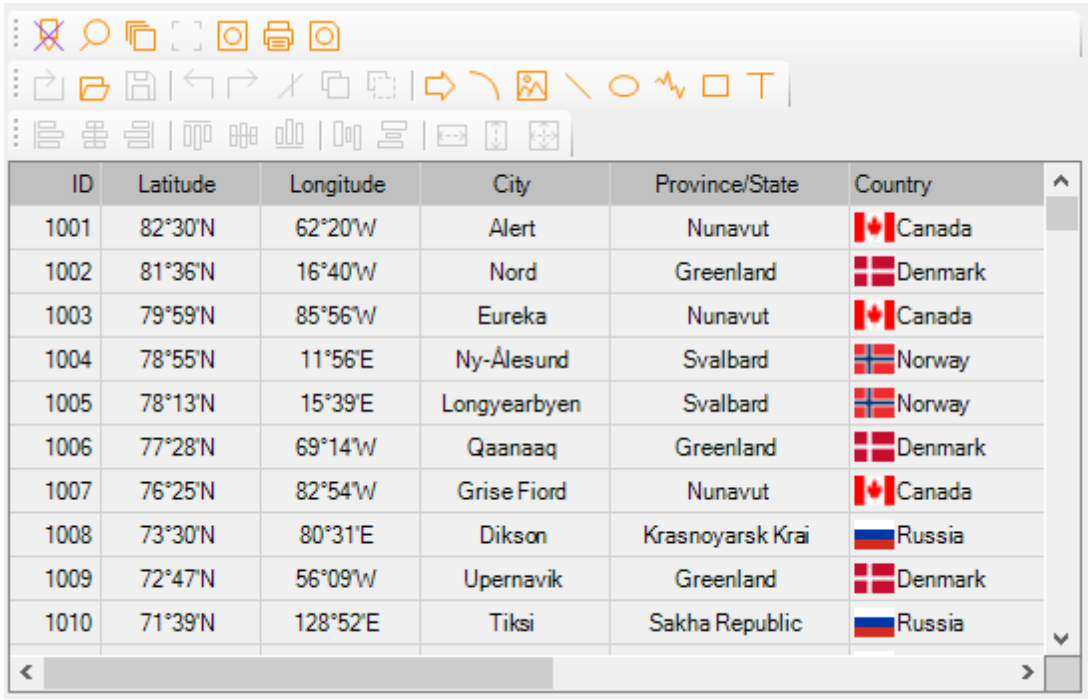
5: Editing in Edit Panel



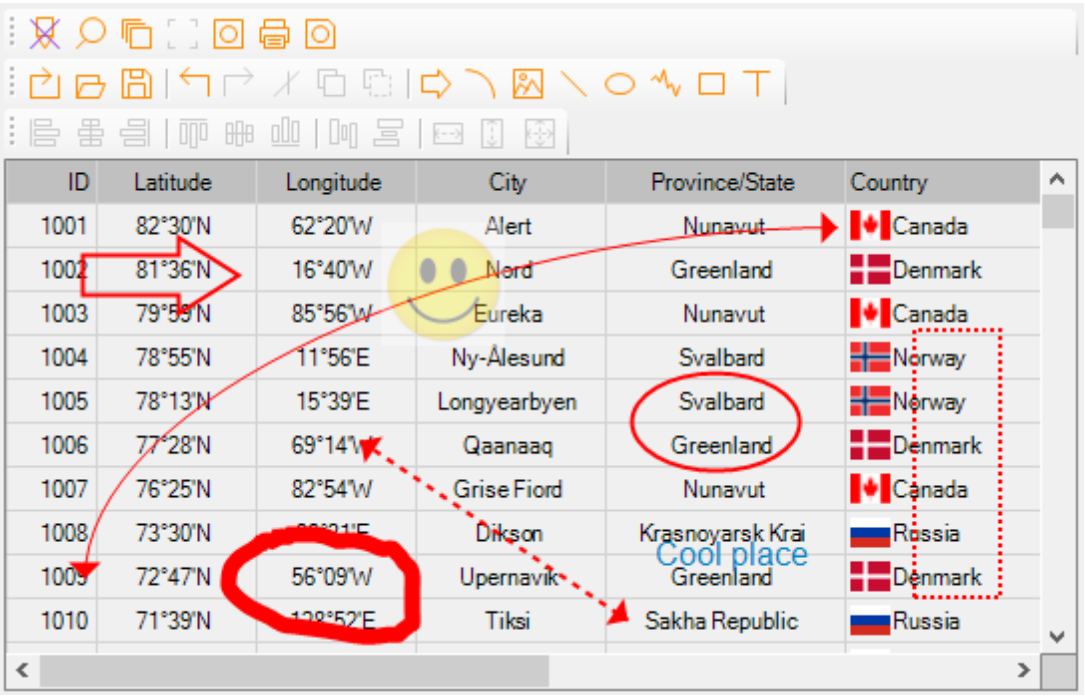
6: Edit Mode with Sections



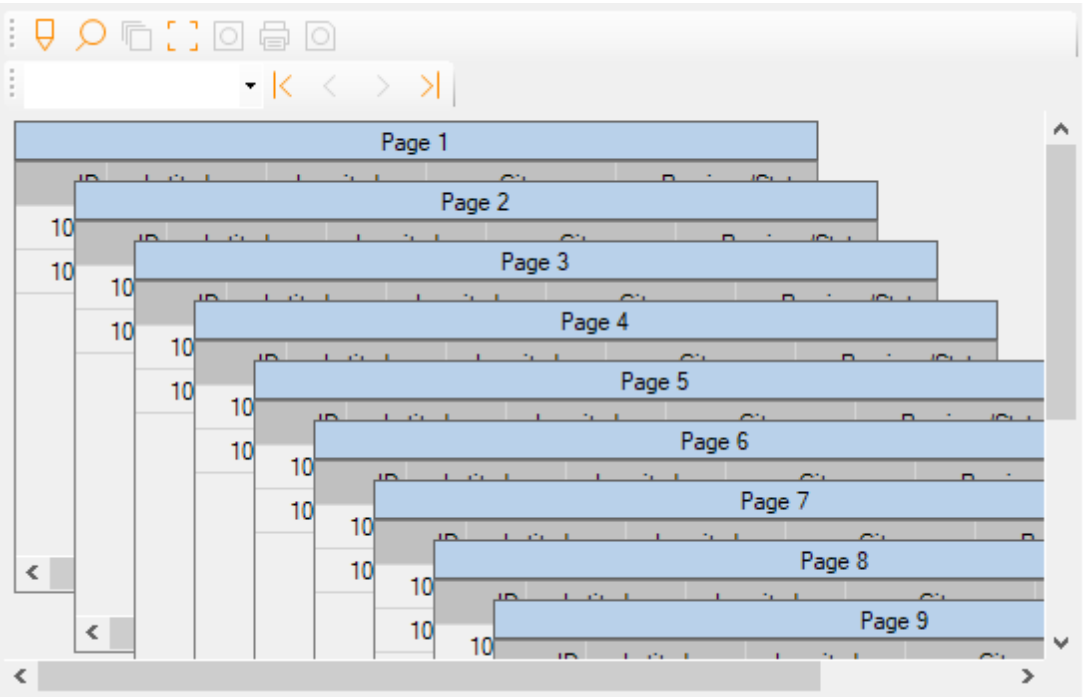
7: Batch Editing



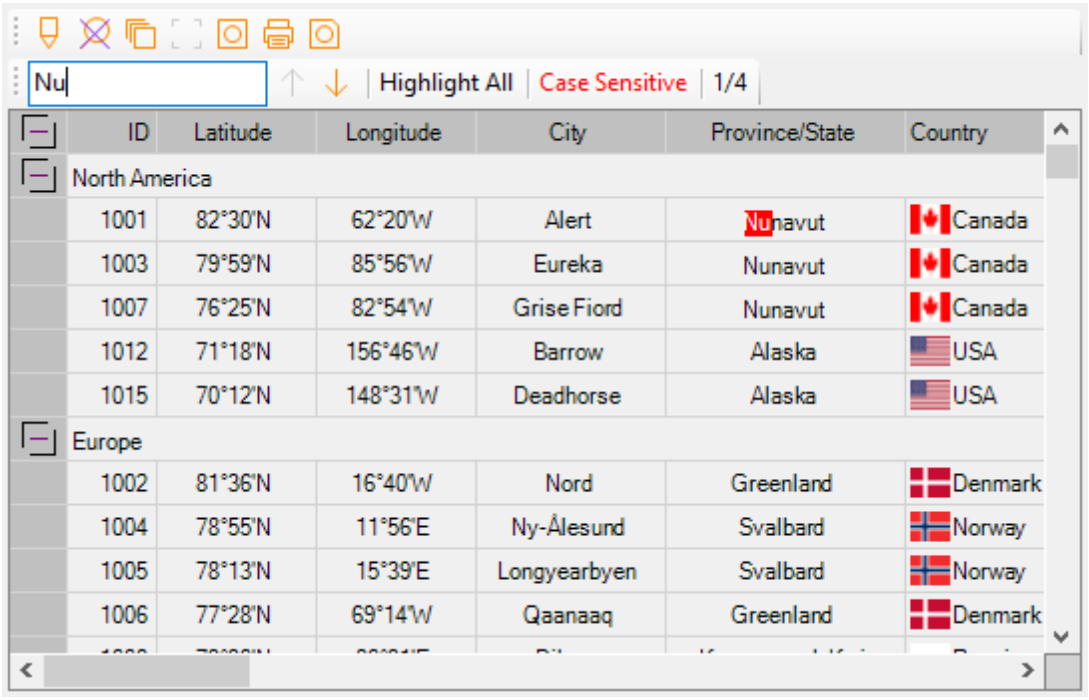
8: Readonly Mode



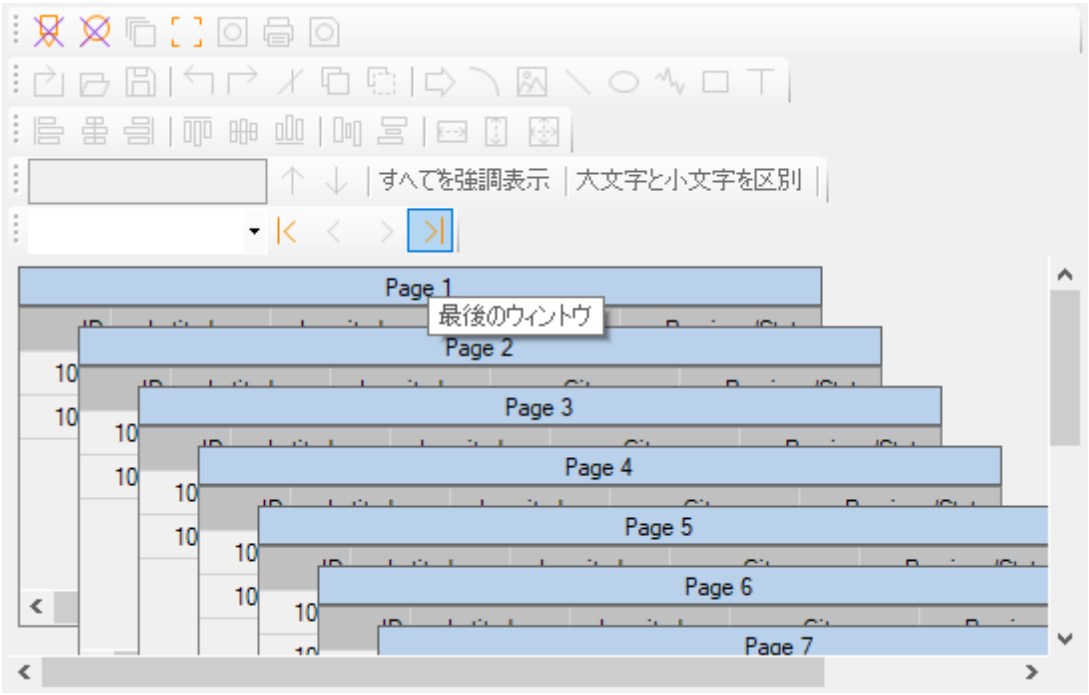
9: Annotation



10: Cascaded Sub Grids



11: Search Grid



12: Localization with 50+ Languages

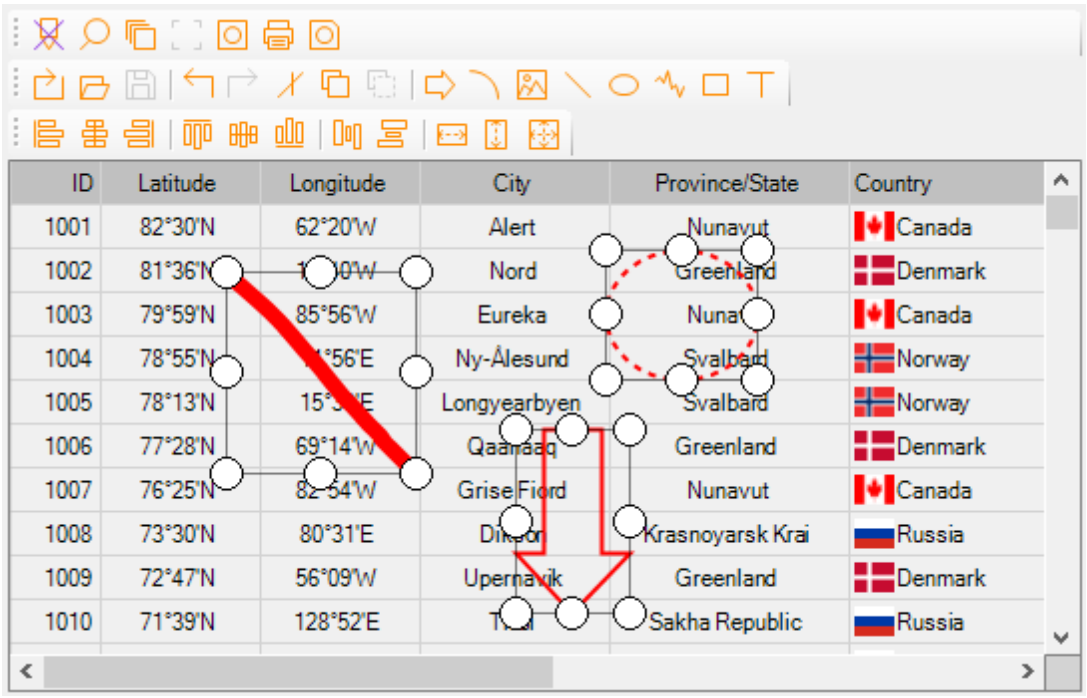
A few quick facts are listed below:

1. A non-databound grid that can be used to populate any structured, semi-structured, or unstructured data - either Small or Big.
2. Edit mode provides 5 options - in row, edit panel, edit panel and show row, popup window, and batch - which are powered by CheckBox, ComboBox, DateTimePicker, MaskedTextBox, NumericUpDown, PictureBox, RichTextBox, and TextBox.
3. Industry's 1st grid control with WYSIWYG annotating feature backed by 8 different tools - arrow, curve, image, line, oval, paint brush, rectangle, and text.
4. Industry's 1st grid control with cascaded sub grids.
5. Industry's 1st grid control with non-rectangle layouts - quadrilateral, curve, and arrow.
6. Non-rectangle scrollbars
7. Search the grid - highlight all and match case.
8. Unrivaled 50+ languages for localization.
9. Copy screenshot to clipboard.
10. Print and save screenshot.
11. Categorize the rows into different sections and group columns.
12. Display an arrow to indicate scroll bar's orientation - horizontal or vertical scroll bar.
13. Multiple trendy checkbox styles - 19 in total.
14. 8 different alignment options per cell for combined text(s) and image.
15. Up to 4 lines of texts per cell.
16. Configurable settings including background color, text color, font, section icon, row heights, column widths, sorting columns, resizing rows and columns, moving columns, shading alternate rows, linked texts, and etc.

During annotating process, multiple objects can be selected via SHIFT key and mouse clicking, and the following grid operations are disabled.

1. Select cell(s) or row(s).
2. Edit mode.
3. Move and resize columns.
4. Sort a column.
5. Click checkboxes if applicable.
6. Expand or collapse sections if applicable.

Since the bounds of a paint brush object are determined by its path, once it is done, it can't be resized. Also it can't get involved in being made same width, height, or size with other objects. However you can still drag and move it, change color, line width and type.



2 Installation

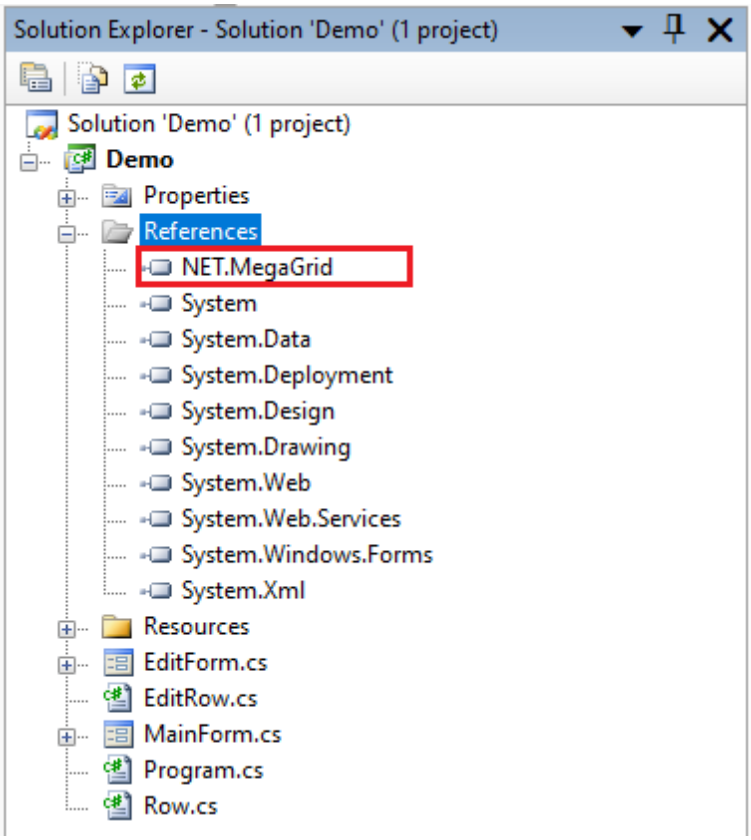
2.1 Trial Version

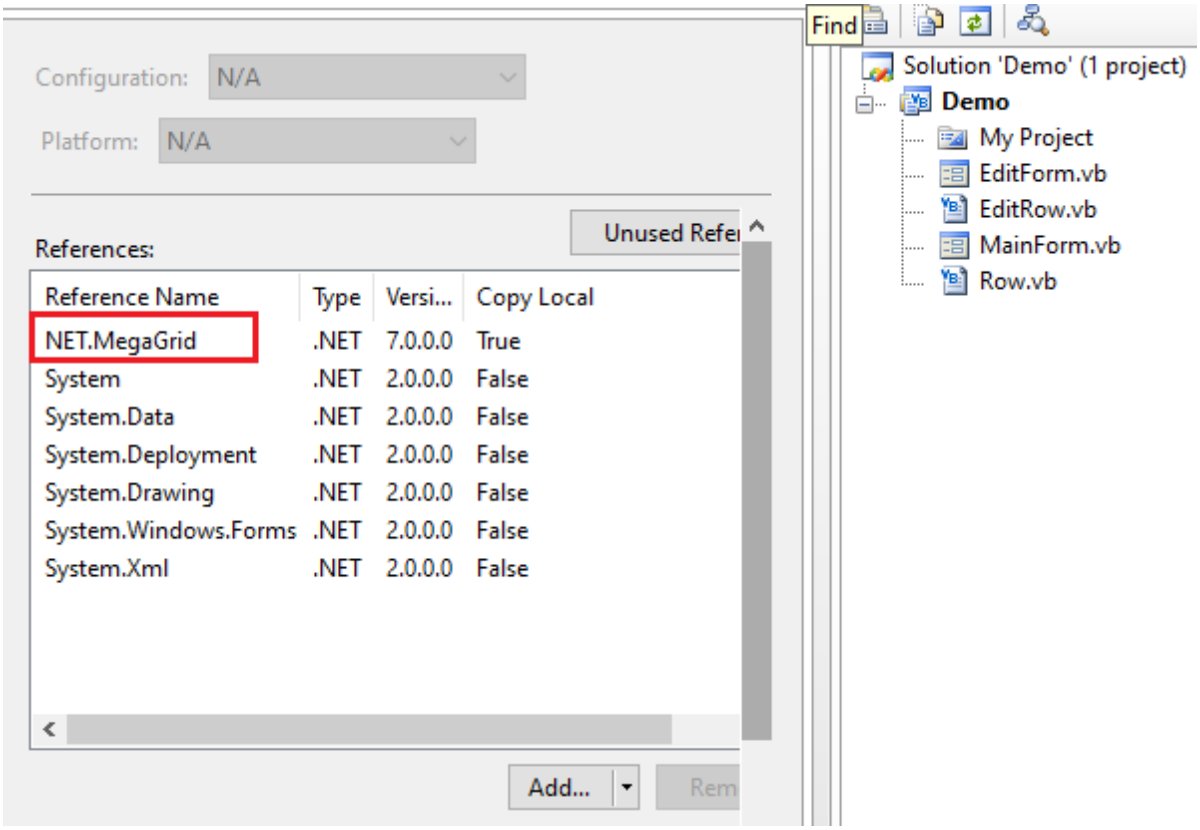
1. Unzip MegaGridNet.zip and extract all files to a local folder.

2. Unzip CSharp.zip or VB_NET.zip and extract all files to a local folder,

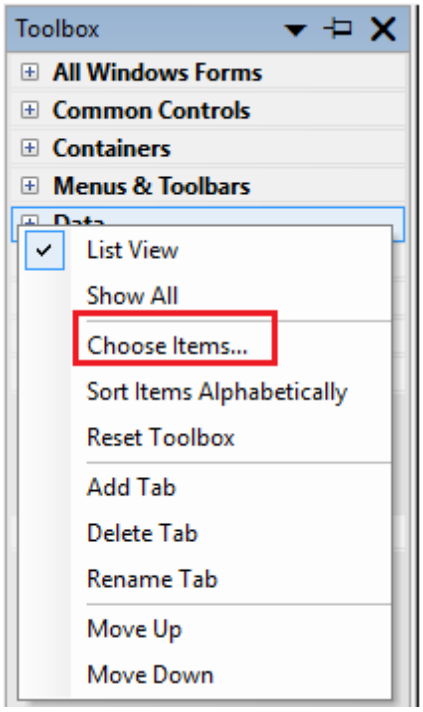
3. Locate and copy **NET.MegaGrid.dll** to your project folder.

4. Add reference to **NET.MegaGrid.dll** in your project - either C# or VB.NET.

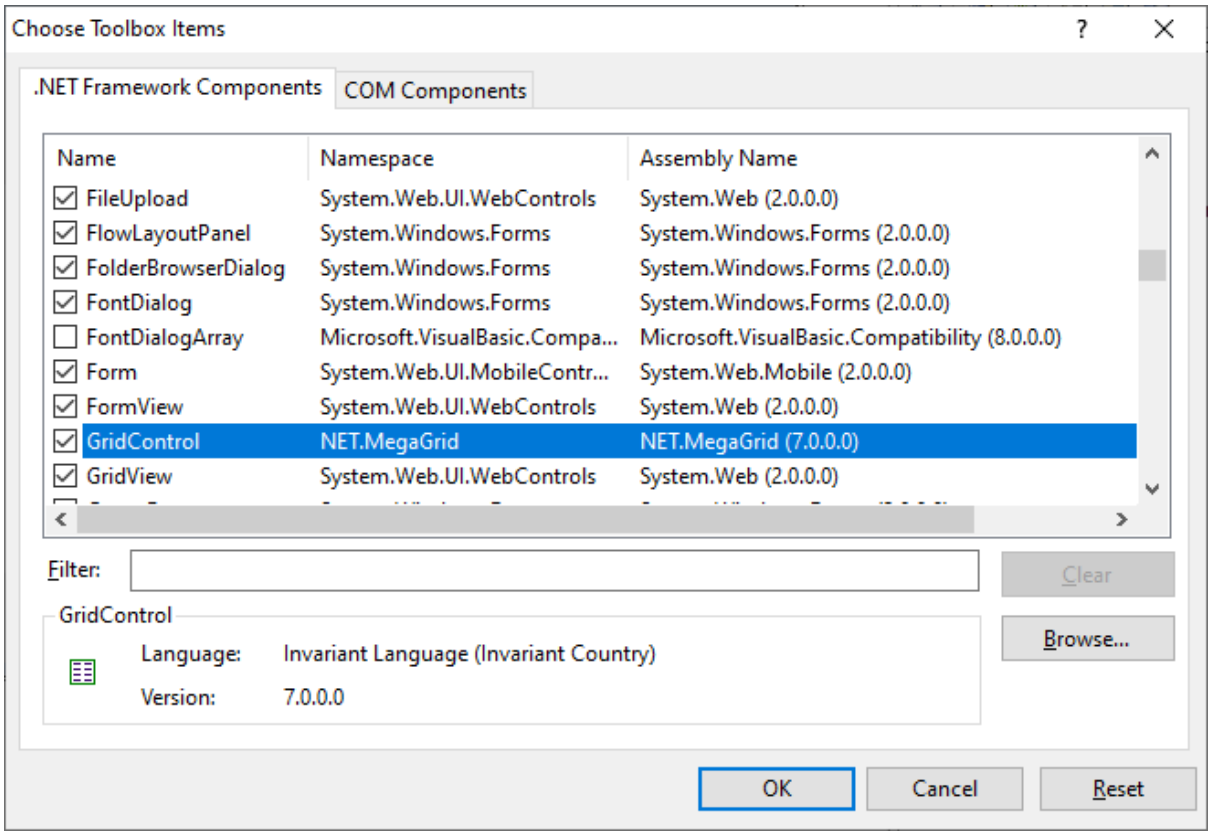




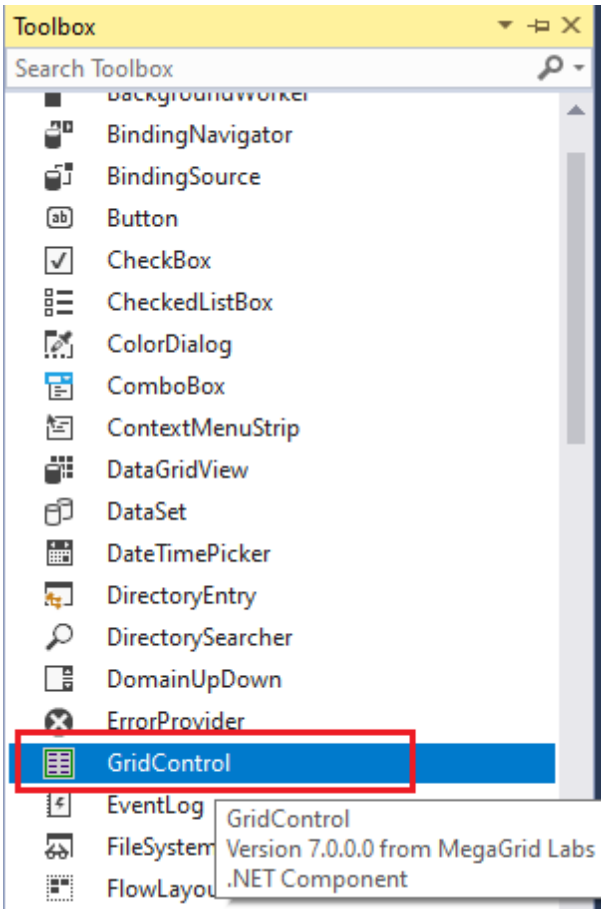
5. Right click anywhere on the Toolbox to select "**Choose Items...**".



6. In the "**Choose Toolbox Items**" dialog, select the "**.NET Framework Components**" tab, click on the "**Browse**" button and select **NET.MegaGrid.dll**.

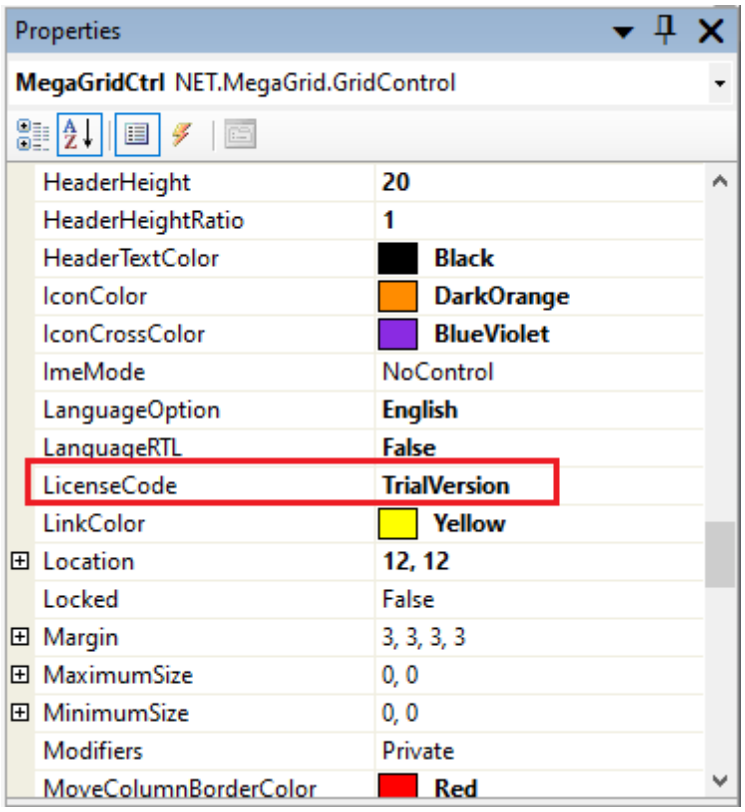


7. If the steps 5 & 6 fail and you see an error message "**There are no components in ....**", you can simply open Windows Explorer and navigate to **NET.MegaGrid.dll**, drag the DLL and drop it onto Visual Studio Toolbox area; MegaGrid control will appear instantly.



8. Drag and drop the control onto a Windows Form, and please keep in the mind that the trial version's control in edit mode randomly pops up a message box saying "You are using the trial version copy of MegaGrid .NET WinForms Control.".

9. A screenshot of the properties page is listed below:



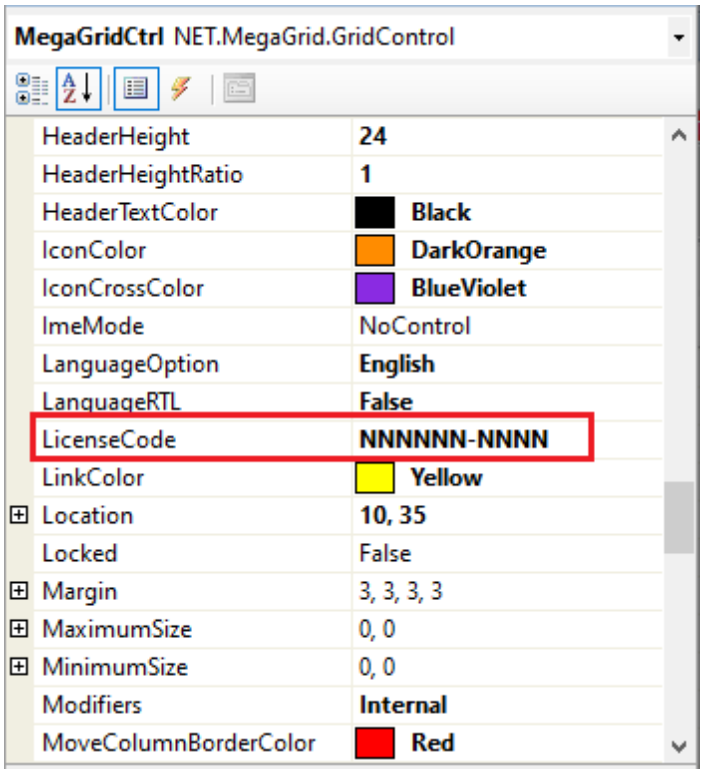
2.2 Full Version

1. Unzip MegaGridNet_Full.zip and extract all files to a local folder.

2. Unzip CSharp.zip or VB_NET.zip and extract all files to a local folder,

3. Locate and copy **NET.MegaGrid.dll** to your project folder to replace the trial version .dll file.

4. Enter a 10-digit license code with the format NNNNNN-NNNN into the **LicenseCode** property of the grid control to activate the control.



3 How to Distribute It

If you want to redistribute the MegaGrid .NET WinForms Control as part of your application, simply put NET.MegaGrid.dll into the application local folder on the target machine.

4 Reference Guide

Here we only list the properties, methods, events, and enumerations that are specific to MegaGrid .NET control.

For all other ones that inherit from .NET UserControl class, please check out Microsoft MSDN website for more information.

4.1 Properties

4.1.1 ArchHeight

Gets or sets the height in pixels of arch in the curved layout.

[Visual Basic .NET]

```
Public Property ArchHeight As Integer
```

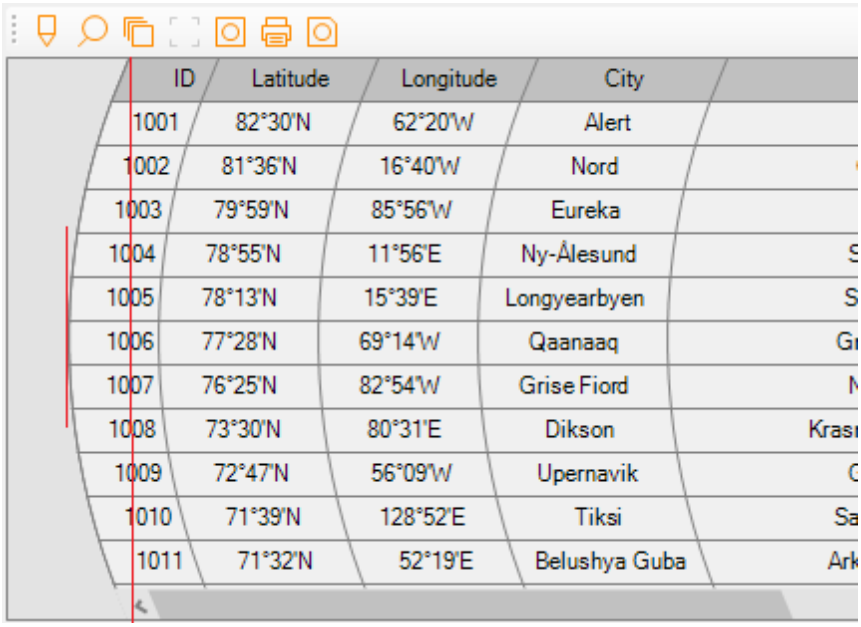
[C#]

```
public int ArchHeight {get; set;}
```

Remarks

The default value is 10 and the minimal value is 5, and this property becomes irrelevant if the GridLaytout property is neither Lake_Geneva nor Lake_Saimaa.

In the screenshot below, this value determines the distance between 2 red lines.



ID	Latitude	Longitude	City	
1001	82°30'N	62°20'W	Alert	
1002	81°36'N	16°40'W	Nord	
1003	79°59'N	85°56'W	Eureka	
1004	78°55'N	11°56'E	Ny-Ålesund	S
1005	78°13'N	15°39'E	Longyearbyen	S
1006	77°28'N	69°14'W	Qaanaaq	Gr
1007	76°25'N	82°54'W	Grise Fiord	N
1008	73°30'N	80°31'E	Dikson	Krasn
1009	72°47'N	56°09'W	Upernavik	C
1010	71°39'N	128°52'E	Tiksi	Sa
1011	71°32'N	52°19'E	Belushya Guba	Ark

4.1.2 AllowCheckBox

Gets or sets a boolean flag indicating whether the checkboxes will be displayed or not for the grid in readonly mode.

[Visual Basic .NET]

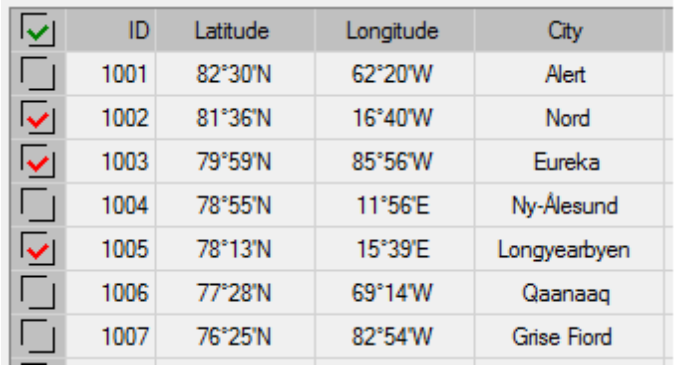
```
Public Property AllowCheckBox As Boolean
```

[C#]

```
public bool AllowCheckBox {get; set;}
```

Remarks

The default value is TRUE, and a screenshot is listed below.



	ID	Latitude	Longitude	City
☐	1001	82°30'N	62°20'W	Alert
☑	1002	81°36'N	16°40'W	Nord
☑	1003	79°59'N	85°56'W	Eureka
☐	1004	78°55'N	11°56'E	Ny-Ålesund
☑	1005	78°13'N	15°39'E	Longyearbyen
☐	1006	77°28'N	69°14'W	Qaanaaq
☐	1007	76°25'N	82°54'W	Grise Fiord

4.1.3 AllowHeaderGrouping

Gets or sets a boolean flag indicating whether it is allowed to group the column headers or not.

[Visual Basic .NET]

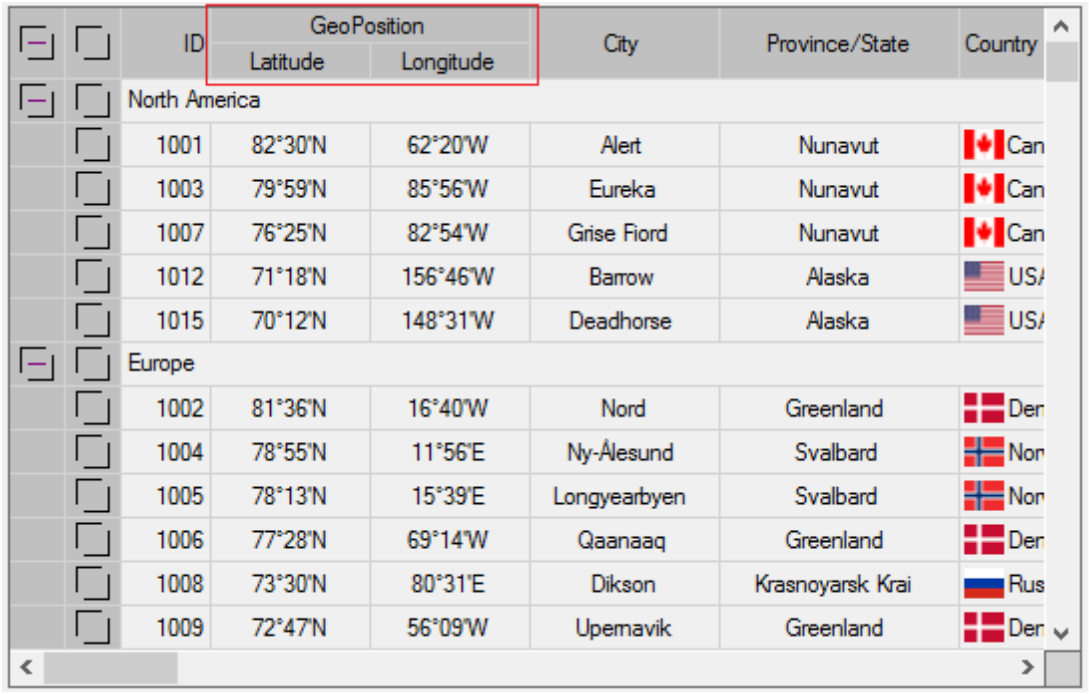
```
Public Property AllowHeaderGrouping As Boolean
```

[C#]

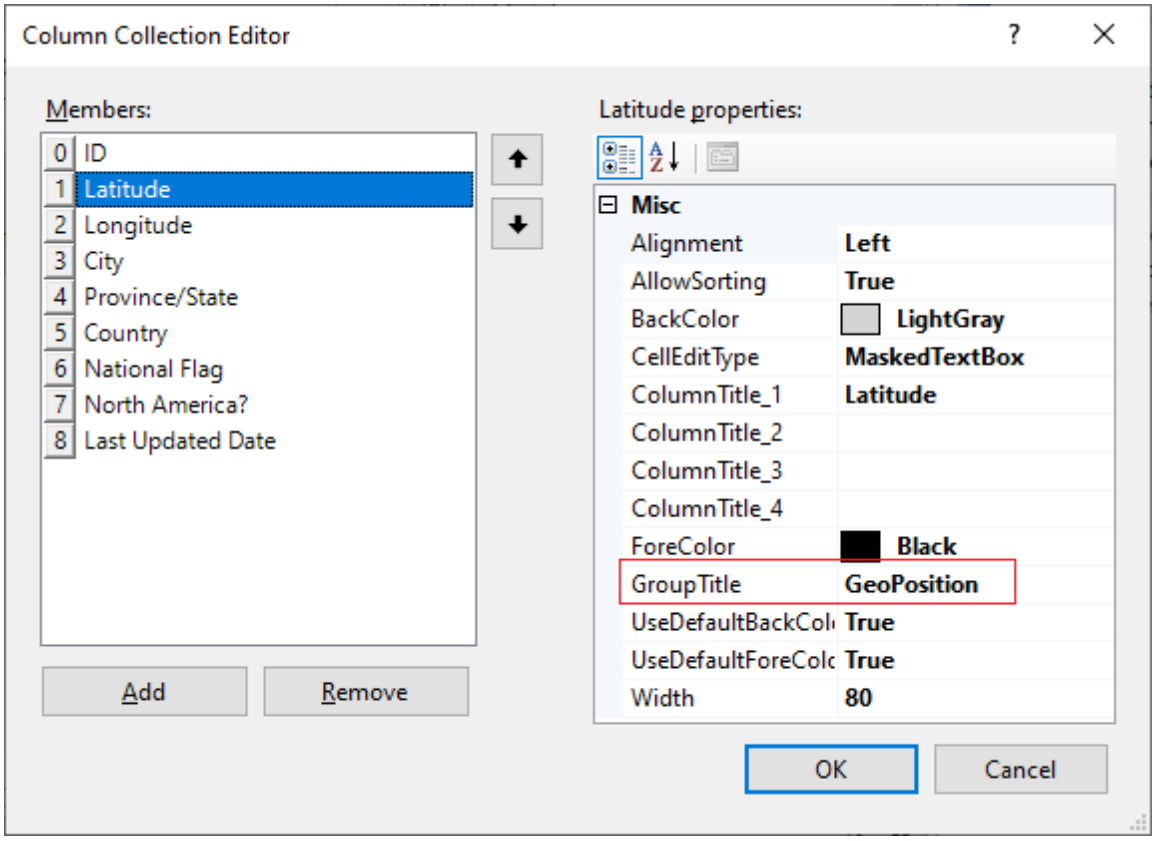
```
public bool AllowHeaderGrouping {get; set;}
```

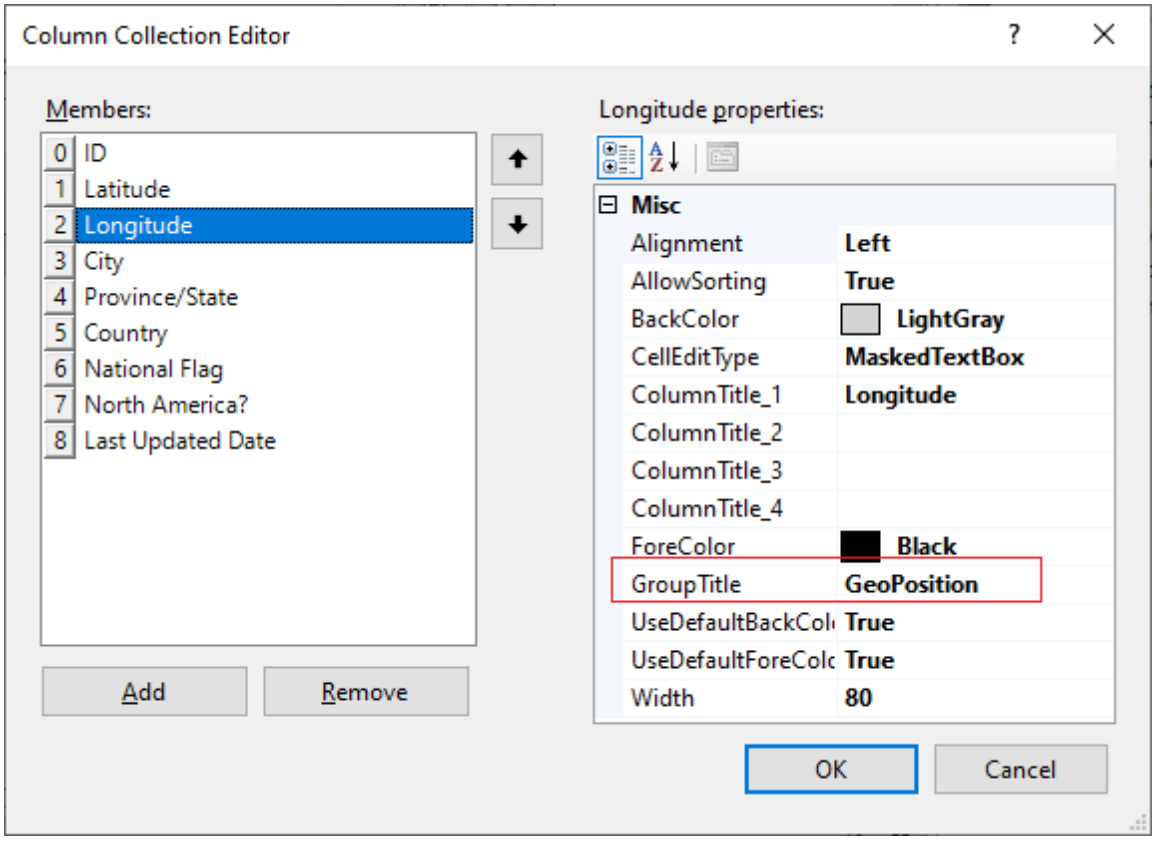
Remarks

The default value is TRUE, and 3 screenshots are listed below; in this example, the group titles for both Latitude and Longitude are GeoPosition.



ID	GeoPosition		City	Province/State	Country
	Latitude	Longitude			
North America					
1001	82°30'N	62°20'W	Alert	Nunavut	Can
1003	79°59'N	85°56'W	Eureka	Nunavut	Can
1007	76°25'N	82°54'W	Grise Fiord	Nunavut	Can
1012	71°18'N	156°46'W	Barrow	Alaska	USA
1015	70°12'N	148°31'W	Deadhorse	Alaska	USA
Europe					
1002	81°36'N	16°40'W	Nord	Greenland	Den
1004	78°55'N	11°56'E	Ny-Ålesund	Svalbard	Nor
1005	78°13'N	15°39'E	Longyearbyen	Svalbard	Nor
1006	77°28'N	69°14'W	Qaanaaq	Greenland	Den
1008	73°30'N	80°31'E	Dikson	Krasnoyarsk Krai	Rus
1009	72°47'N	56°09'W	Upernavik	Greenland	Den





4.1.4 AllowSectionIcon

Gets or sets a boolean flag indicating whether the icon(s) for the section(s) are allowed to be displayed or not for the grid.

[Visual Basic .NET]

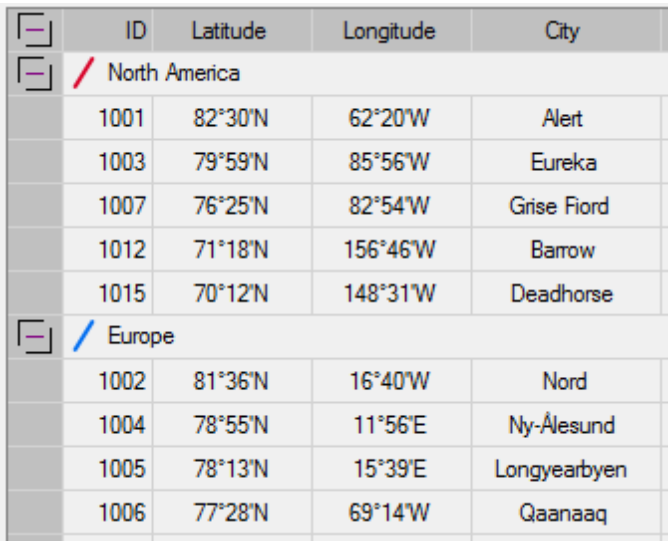
```
Public Property AllowSectionIcon As Boolean
```

[C#]

```
public bool AllowSectionIcon {get; set;}
```

Remarks

The default value is TRUE, and a screenshot is listed below.



ID	Latitude	Longitude	City
North America			
1001	82°30'N	62°20'W	Alert
1003	79°59'N	85°56'W	Eureka
1007	76°25'N	82°54'W	Grise Fiord
1012	71°18'N	156°46'W	Barrow
1015	70°12'N	148°31'W	Deadhorse
Europe			
1002	81°36'N	16°40'W	Nord
1004	78°55'N	11°56'E	Ny-Ålesund
1005	78°13'N	15°39'E	Longyearbyen
1006	77°28'N	69°14'W	Qaanaaq

4.1.5 AllowSections

Gets or sets a boolean flag indicating whether the section(s) are allowed or not for the grid.

[Visual Basic .NET]

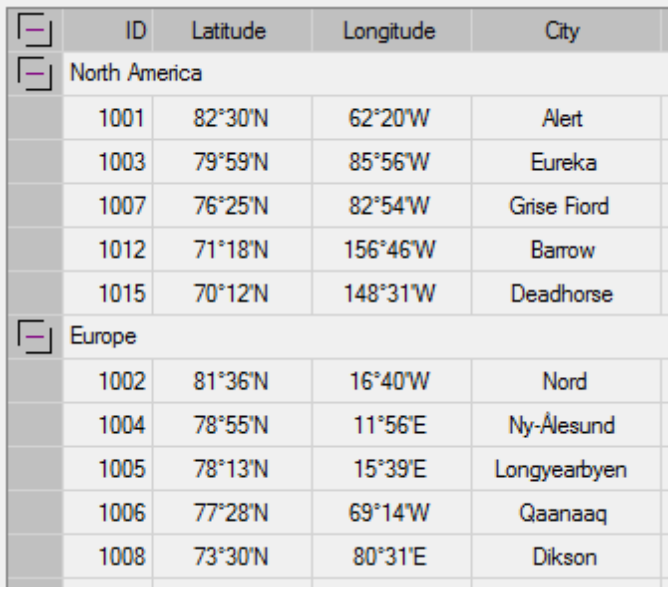
```
Public Property AllowSections As Boolean
```

[C#]

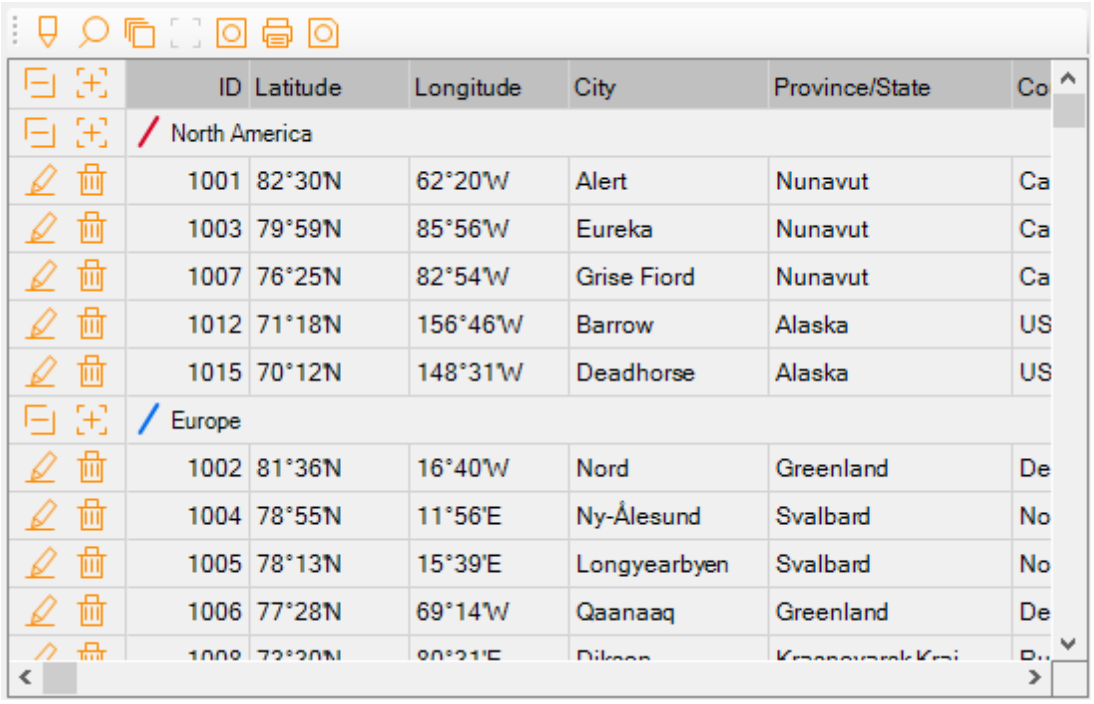
```
public bool AllowSections {get; set;}
```

Remarks

The default value is TRUE, and 2 screenshots are listed below. This property allows you to group the rows into different sections.



ID	Latitude	Longitude	City
North America			
1001	82°30'N	62°20'W	Alert
1003	79°59'N	85°56'W	Eureka
1007	76°25'N	82°54'W	Grise Fiord
1012	71°18'N	156°46'W	Barrow
1015	70°12'N	148°31'W	Deadhorse
Europe			
1002	81°36'N	16°40'W	Nord
1004	78°55'N	11°56'E	Ny-Ålesund
1005	78°13'N	15°39'E	Longyearbyen
1006	77°28'N	69°14'W	Qaanaaq
1008	73°30'N	80°31'E	Dikson



ID	Latitude	Longitude	City	Province/State	Co
North America					
1001	82°30'N	62°20'W	Alert	Nunavut	Ca
1003	79°59'N	85°56'W	Eureka	Nunavut	Ca
1007	76°25'N	82°54'W	Grise Fiord	Nunavut	Ca
1012	71°18'N	156°46'W	Barrow	Alaska	US
1015	70°12'N	148°31'W	Deadhorse	Alaska	US
Europe					
1002	81°36'N	16°40'W	Nord	Greenland	De
1004	78°55'N	11°56'E	Ny-Ålesund	Svalbard	No
1005	78°13'N	15°39'E	Longyearbyen	Svalbard	No
1006	77°28'N	69°14'W	Qaanaaq	Greenland	De

4.1.6 AllowWheelSection

Gets or sets a boolean flag indicating whether a new cell or row - one row up or down - is allowed to be selected via scrolling the mouse wheel for the grid in readonly mode.

[Visual Basic .NET]

```
Public Property AllowWheelSelection As Boolean
```

[C#]

```
public bool AllowWheelSelection {get; set;}
```

Remarks

The default value is TRUE. If neither a cell nor a row is currently selected or the property SelectMultiCells is set to TRUE, a new cell or row will not be selected via scrolling the mouse wheel even if it is set to TRUE.

4.1.7 AllowZebras

Gets or sets a boolean flag indicating whether it is allowed to shade alternate rows or not for the grid in readonly mode.

[Visual Basic .NET]

```
Public Property AllowZebras As Boolean
```

[C#]

```
public bool AllowZebras {get; set;}
```

Remarks

The default value is FALSE, and a screenshot for TRUE value is listed below.

ID	Latitude	Longitude	City	Province/State	Country
1001	82°30'N	62°20'W	Alert	Nunavut	Canada
1002	81°36'N	16°40'W	Nord	Greenland	Denmark
1003	79°59'N	85°56'W	Eureka	Nunavut	Canada
1004	78°55'N	11°56'E	Ny-Ålesund	Svalbard	Norway
1005	78°13'N	15°39'E	Longyearbyen	Svalbard	Norway
1006	77°28'N	69°14'W	Qaanaaq	Greenland	Denmark
1007	76°25'N	82°54'W	Grise Fiord	Nunavut	Canada

4.1.8 AnnotateBorderColor

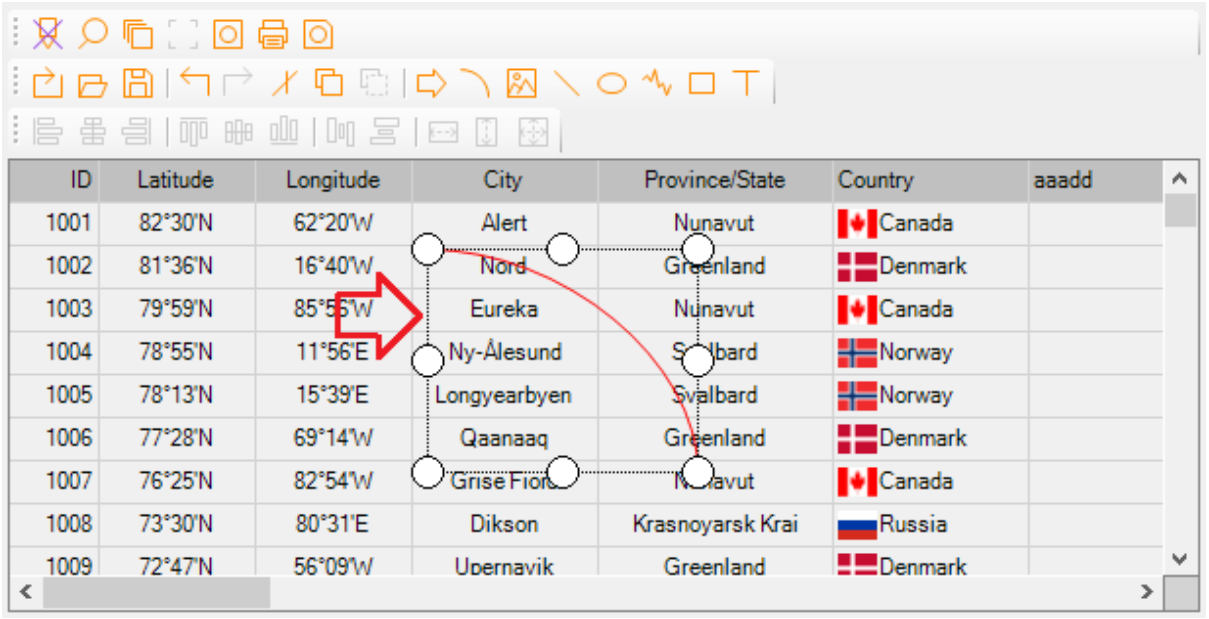
Gets or sets the color of border for annotation tools.

```
[Visual Basic .NET]
Public Property AnnotateBorderColor As Color
```

```
[C#]
public Color AnnotateBorderColor {get; set;}
```

Remarks

The default value is black, and a screenshot is listed below.



4.1.9 AnnotateBorderMarkerFillColor

Gets or sets the fill color of border markers for annotation tools.

[Visual Basic .NET]

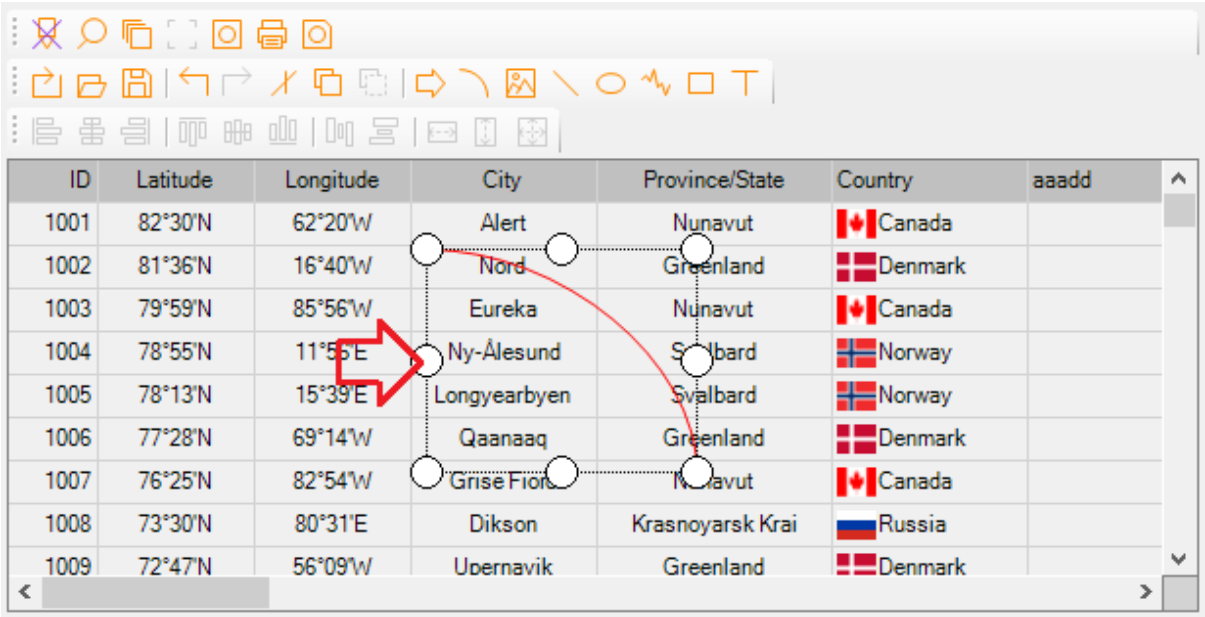
Public Property AnnotateBorderMarkerFillColor As Color

[C#]

public Color AnnotateBorderMarkerFillColor {get; set;}

Remarks

The default value is white, and a screenshot is listed below.



4.1.10 AnnotateBorderMarkerImage

Gets or sets the marker image of border for annotation tools.

[Visual Basic .NET]

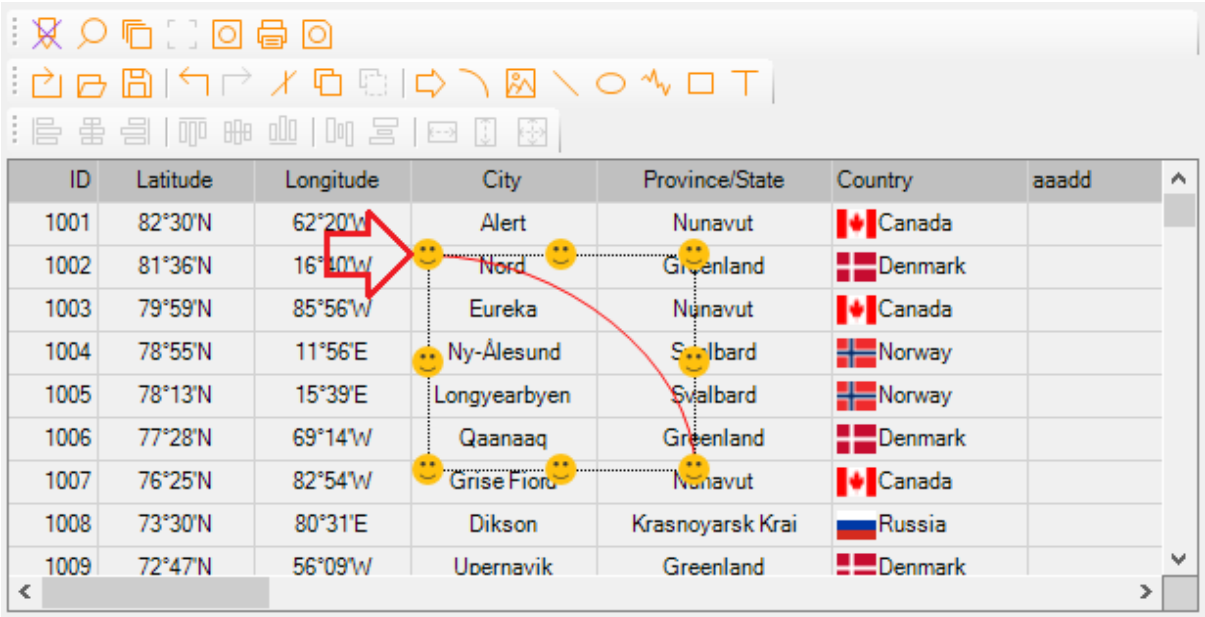
Public Property AnnotateBorderMarkerImage As Image

[C#]

public Image AnnotateBorderMarkerImage {get; set;}

Remarks

A screenshot for happy face image is listed below. Please set AnnotateBorderMarkerOption to Icon_File before this property can kick in.



4.1.11 AnnotateBorderMarkerOption

Gets or sets the marker option of border for annotation tools.

[Visual Basic .NET]

Public Property AnnotateBorderMarkerOption As enumAnnotateBorderMarker

[C#]

public enumAnnotateBorderMarker AnnotateBorderMarkerOption {get; set;}

Remarks

The default value is Circle_Shape with the value of 1.

4.1.12 AnnotateWinFormsIcon

Gets or sets the icon image of all windows forms related to annotating activities.

[Visual Basic .NET]

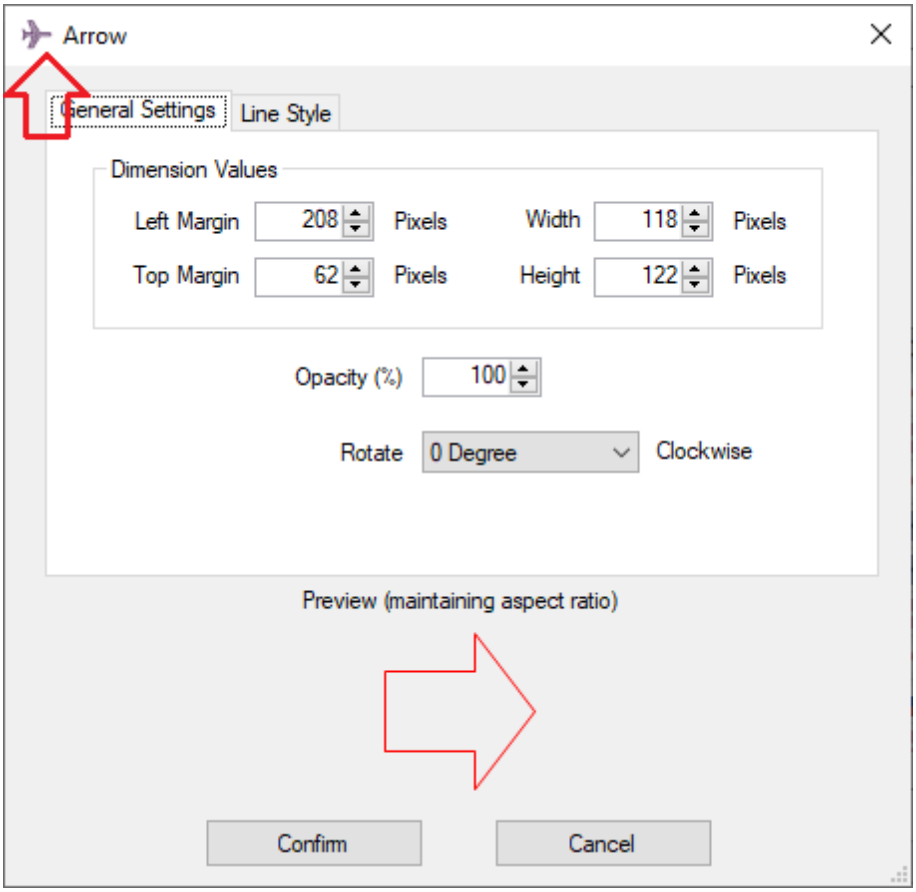
Public Property AnnotateWinFormsIcon As Icon

[C#]

public Icon AnnotateWinFormsIcon {get; set;}

Remarks

You can set this property to your organization's logo or something else, and a screenshot is listed below.



4.1.13 ArrowColor

Gets or sets the color of the arrow indicating scroll bar's orientation, which can be either upward / downward for the vertical scroll bar or leftward / rightward for the horizontal scroll bar.

[Visual Basic .NET]

```
Public Property ArrowColor As Color
```

[C#]

```
public Color ArrowColor {get; set;}
```

Remarks

The default value is red, and a screenshot is listed below.

ID	Latitude	Longitude	City	Province/State	Country
1002	81°36'N	16°40'W	Nord	Greenland	Denmark
1003	79°59'N	85°56'W	Eureka	Nunavut	Canada
1004	78°55'N	11°56'E	Ny-Ålesund	Svalbard	Norway
1005	78°13'N	15°39'E	Longyearbyen	Svalbard	Norway
1006	77°28'N	69°14'W	Qaanaaq	Greenland	Denmark
1007	76°25'N	82°54'W	Grise Fiord	Nunavut	Canada
1008	73°30'N	80°31'E	Dikson	Krasnoyarsk Krai	Russia
1009	72°47'N	56°09'W	Upernavik	Greenland	Denmark
1010	71°39'N	128°52'E	Tiksi	Sakha Republic	Russia
1011	71°32'N	52°19'E	Belushya Guba	Arkhangelsk Oblast	Russia
1012	71°18'N	156°46'W	Barrow	Alaska	USA
1013	70°59'N	25°59'E	Honningsvåg	Finnmark	Norway
1014	70°40'N	23°41'E	Hammerfest	Finnmark	Norway
1015	70°12'N	148°31'W	Deadhorse	Alaska	USA

4.1.14 ArrowLineWidth

Gets or sets the line width of the arrow indicating scroll bar's orientation.

[Visual Basic .NET]

Public Property ArrowLineWidth As enumArrowLineWidth

[C#]

public enumArrowLineWidth ArrowLineWidth {get; set;}

Remarks

The default value is enumArrowLineWidth.Medium.

4.1.15 ArrowOpacity

Gets or sets the opacity level of the arrow indicating scroll bar's orientation.

[Visual Basic .NET]

Public Property ArrowOpacity As Double

[C#]

public double ArrowOpacity {get; set;}

Remarks

The default value is 0.5, and a valid value must be between 0 and 1.

- If it is 0, the arrow will be fully transparent and invisible.
- If it is 1, the arrow will be fully opaque.

4.1.16 ArrowSize

Gets or sets the size of the arrow indicating scroll bar's orientation.

```
[Visual Basic .NET]
Public Property ArrowSize As enumArrowSize
```

```
[C#]
public enumArrowSize ArrowSize {get; set;}
```

Remarks

The default value is enumArrowSize.Medium.

4.1.17 BorderColor

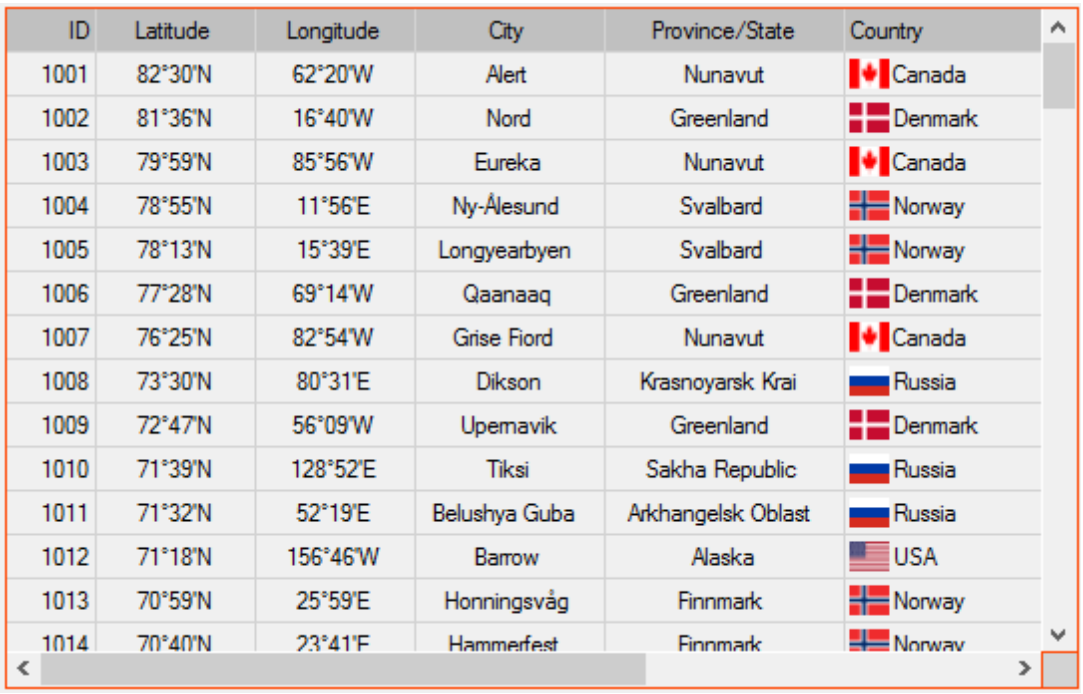
Gets or sets the border color of the grid control.

```
[Visual Basic .NET]
Public Property BorderColor As Color
```

```
[C#]
public Color BorderColor {get; set;}
```

Remarks

The default value is gray, and a screenshot for the orange color is listed below.



ID	Latitude	Longitude	City	Province/State	Country
1001	82°30'N	62°20'W	Alert	Nunavut	Canada
1002	81°36'N	16°40'W	Nord	Greenland	Denmark
1003	79°59'N	85°56'W	Eureka	Nunavut	Canada
1004	78°55'N	11°56'E	Ny-Ålesund	Svalbard	Norway
1005	78°13'N	15°39'E	Longyearbyen	Svalbard	Norway
1006	77°28'N	69°14'W	Qaanaaq	Greenland	Denmark
1007	76°25'N	82°54'W	Grise Fiord	Nunavut	Canada
1008	73°30'N	80°31'E	Dikson	Krasnoyarsk Krai	Russia
1009	72°47'N	56°09'W	Upernavik	Greenland	Denmark
1010	71°39'N	128°52'E	Tiksi	Sakha Republic	Russia
1011	71°32'N	52°19'E	Belushya Guba	Arkhangelsk Oblast	Russia
1012	71°18'N	156°46'W	Barrow	Alaska	USA
1013	70°59'N	25°59'E	Honningsvåg	Finnmark	Norway
1014	70°40'N	23°41'E	Hammerfest	Finnmark	Norway

4.1.18 BoxColor

Gets or sets the box color for checkbox(es) and/or section node(s) for the grid in readonly mode.

[Visual Basic .NET]

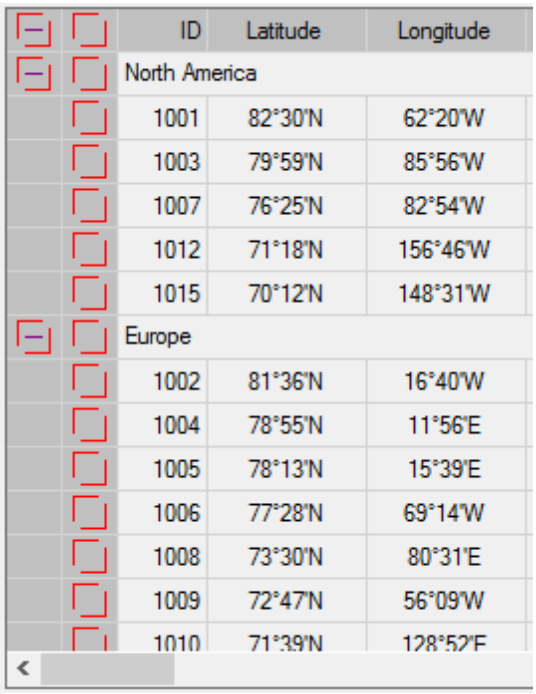
```
Public Property BoxColor As Color
```

[C#]

```
public Color BoxColor {get; set;}
```

Remarks

The default value is black, and a screenshot for the red color is listed below.



ID	Latitude	Longitude
North America		
1001	82°30'N	62°20'W
1003	79°59'N	85°56'W
1007	76°25'N	82°54'W
1012	71°18'N	156°46'W
1015	70°12'N	148°31'W
Europe		
1002	81°36'N	16°40'W
1004	78°55'N	11°56'E
1005	78°13'N	15°39'E
1006	77°28'N	69°14'W
1008	73°30'N	80°31'E
1009	72°47'N	56°09'W
1010	71°39'N	128°52'E

4.1.19 BoxSize

Gets or sets the width / height in pixels of the square box for the checkbox(es) and/or section node(s) for the grid in readonly mode.

[Visual Basic .NET]

```
Public Property BoxSize As Integer
```

[C#]

```
public int BoxSize {get; set;}
```

Remarks

The default value is 16.

4.1.20 BoxStyle

Gets or sets the box style for checkbox(es) and/or section node(s) for the grid.

[Visual Basic .NET]

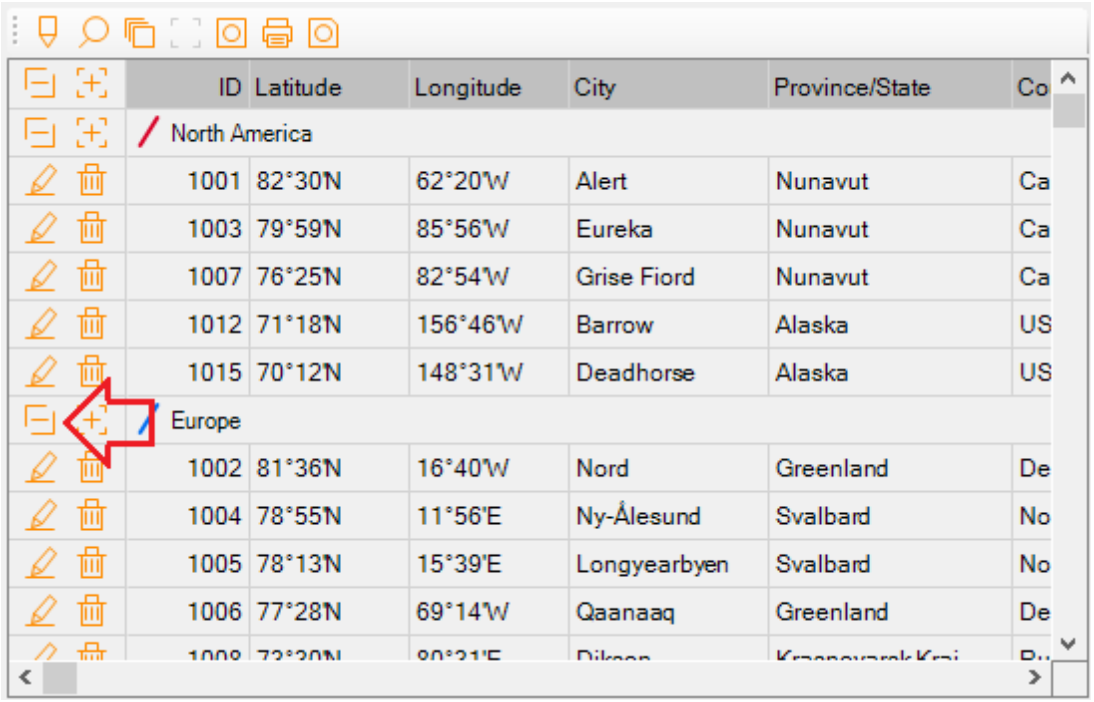
```
Public Property BoxStyle As enumBoxStyle
```

[C#]

```
public enumBoxStyle BoxStyle {get; set;}
```

Remarks

The default value is Calgary with the value of 4, and a screenshot for edit mode is listed.



4.1.21 BulgePointLocationRatio

Gets or sets the bulge point's vertical position, in percentage relative to the grid's height, in the arrow layout.

[Visual Basic .NET]

```
Public Property BulgePointLocationRatio As Float
```

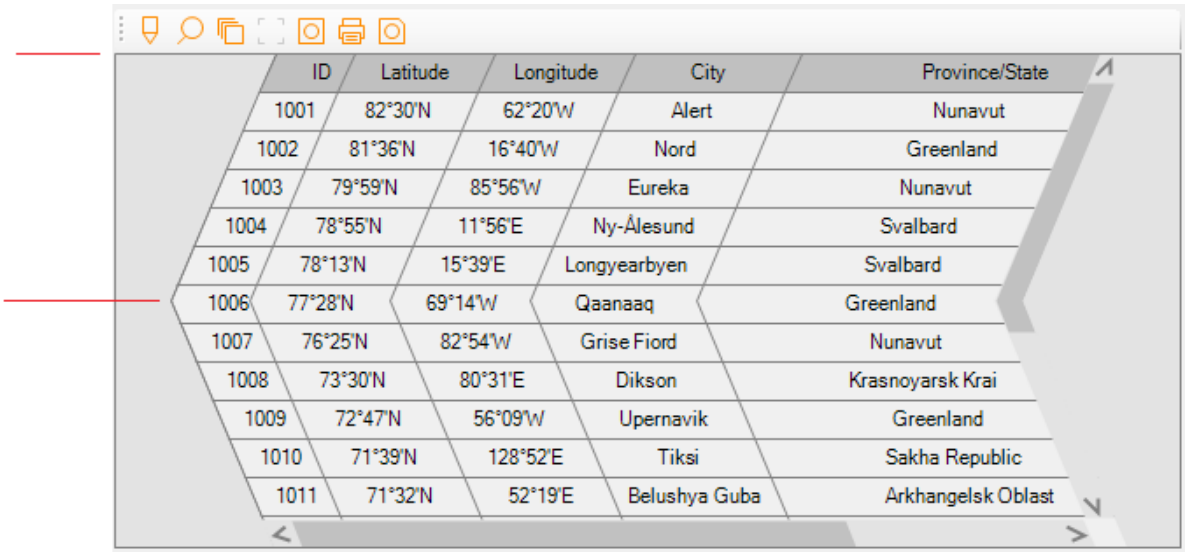
[C#]

```
public float BulgePointLocationRatio {get; set;}
```

Remarks

The default value is 50 and a valid value must be between 10 and 90, and this property becomes irrelevant if the GridLaytout property is neither Moraine_Lake nor Plitvice_Lakes.

In the screenshot below, this value determines the distance between 2 red lines.



ID	Latitude	Longitude	City	Province/State
1001	82°30'N	62°20'W	Alert	Nunavut
1002	81°36'N	16°40'W	Nord	Greenland
1003	79°59'N	85°56'W	Eureka	Nunavut
1004	78°55'N	11°56'E	Ny-Ålesund	Svalbard
1005	78°13'N	15°39'E	Longyearbyen	Svalbard
1006	77°28'N	69°14'W	Qaanaaq	Greenland
1007	76°25'N	82°54'W	Grise Fiord	Nunavut
1008	73°30'N	80°31'E	Dikson	Krasnoyarsk Krai
1009	72°47'N	56°09'W	Upernavik	Greenland
1010	71°39'N	128°52'E	Tiksi	Sakha Republic
1011	71°32'N	52°19'E	Belushya Guba	Arkhangelsk Oblast

4.1.22 BulgeSizeRatio

Gets or sets the bulge point's horizontal position in the arrow layout.

[Visual Basic .NET]

```
Public Property BulgeSizeRatio As Float
```

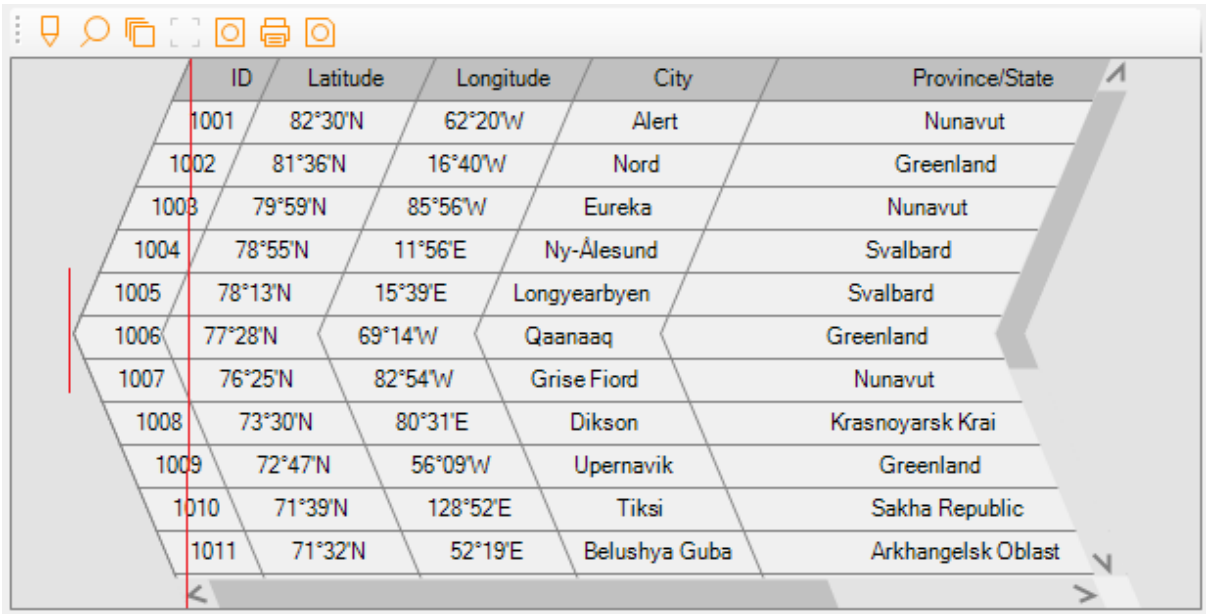
[C#]

```
public float BulgeSizeRatio {get; set;}
```

Remarks

The default value is 10 and a valid value must be between 1 and 30, and this property becomes irrelevant if the GridLaytout property is neither Moraine_Lake nor Plitvice_Lakes.

In the screenshot below, this value - in percentage relative to the grid's width - determines the distance between 2 red lines



4.1.23 CascadeXOffset

Gets or sets the horizontal gap in pixels between 2 adjacent sub grids in cascade mode.

[Visual Basic .NET]

```
Public Property CascadeXOffset As Integer
```

[C#]

```
public int CascadeXOffset {get; set;}
```

Remarks

The default value is 30 and a valid value should be between 1 and 500.

4.1.24 CascadeYOffset

Gets or sets the vertical gap in pixels between 2 adjacent sub grids in cascade mode.

[Visual Basic .NET]

```
Public Property CascadeYOffset As Integer
```

[C#]

```
public int CascadeYOffset {get; set;}
```

Remarks

The default value is 30 and a valid value should be between 1 and 500.

4.1.25 CellSelBKColor

Gets or sets the background color for cell(s) that are currently selected for the grid in readonly mode.

[Visual Basic .NET]

Public Property CellSelBKColor As Color

[C#]

public Color CellSelBKColor {get; set;}

Remarks

The default value is blue, and a screenshot is listed below.

ID	Latitude	Longitude	City	Province/State	Country
1001	82°30'N	62°20'W	Alert	Nunavut	Canada
1002	81°36'N	16°40'W	Nord	Greenland	Denmark
1003	79°59'N	85°56'W	Eureka	Nunavut	Canada
1004	78°55'N	11°56'E	Ny-Ålesund	Svalbard	Norway
1005	78°13'N	15°39'E	Longyearbyen	Svalbard	Norway
1006	77°28'N	69°14'W	Qaanaaq	Greenland	Denmark
1007	76°25'N	82°54'W	Grise Fiord	Nunavut	Canada

4.1.26 CellSelTextColor

Gets or sets the text color for cell(s) that are currently selected for the grid in readonly mode.

[Visual Basic .NET]

Public Property CellSelTextColor As Color

[C#]

public Color CellSelTextColor {get; set;}

Remarks

The default value is white.

4.1.27 CheckColor

Gets or sets the check color for the checkbox(es) for the grid in readonly mode.

[Visual Basic .NET]

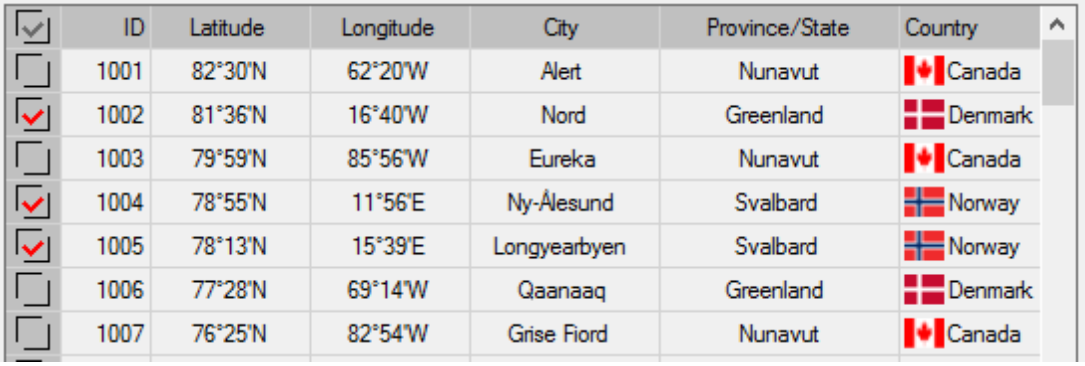
Public Property CheckColor As Color

[C#]

public Color CheckColor {get; set;}

Remarks

The default value is red, and a screenshot is listed below.



ID	Latitude	Longitude	City	Province/State	Country
1001	82°30'N	62°20'W	Alert	Nunavut	Canada
1002	81°36'N	16°40'W	Nord	Greenland	Denmark
1003	79°59'N	85°56'W	Eureka	Nunavut	Canada
1004	78°55'N	11°56'E	Ny-Ålesund	Svalbard	Norway
1005	78°13'N	15°39'E	Longyearbyen	Svalbard	Norway
1006	77°28'N	69°14'W	Qaanaaq	Greenland	Denmark
1007	76°25'N	82°54'W	Grise Fiord	Nunavut	Canada

4.1.28 CheckHalfColor

Gets or sets the check color for the checkbox(es) of the header and section(s) (if applicable) when some child rows are checked and some are not for the grid in readonly mode.

[Visual Basic .NET]

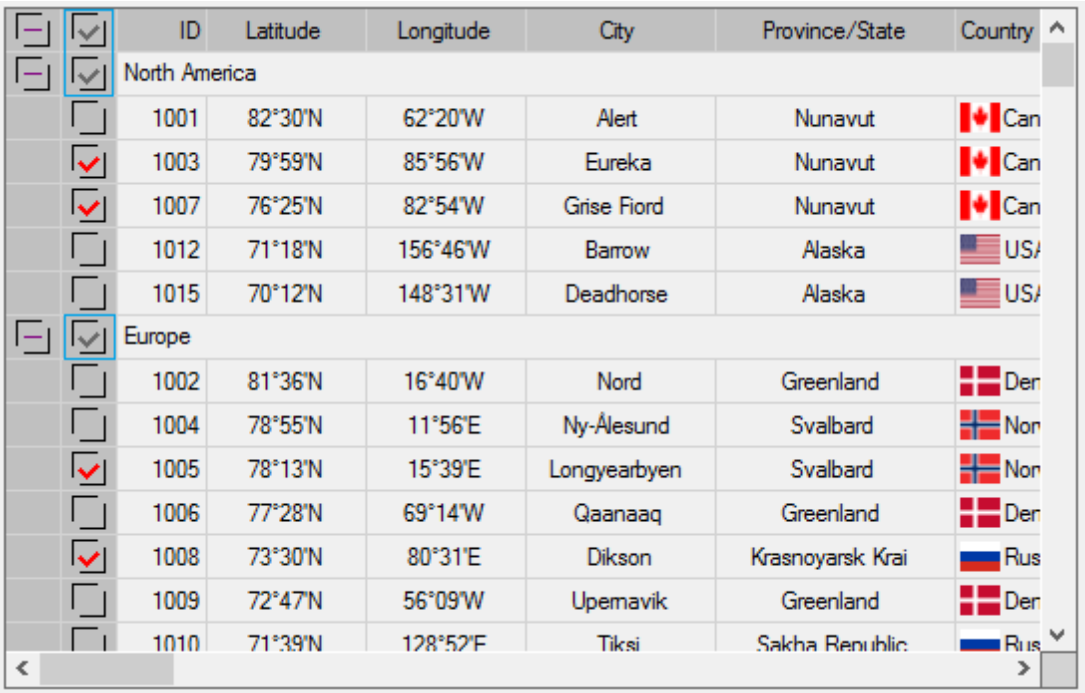
```
Public Property CheckHalfColor As Color
```

[C#]

```
public Color CheckHalfColor {get; set;}
```

Remarks

The default value is green, and a screenshot is listed below.



ID	Latitude	Longitude	City	Province/State	Country
North America					
1001	82°30'N	62°20'W	Alert	Nunavut	Can
1003	79°59'N	85°56'W	Eureka	Nunavut	Can
1007	76°25'N	82°54'W	Grise Fiord	Nunavut	Can
1012	71°18'N	156°46'W	Barrow	Alaska	USA
1015	70°12'N	148°31'W	Deadhorse	Alaska	USA
Europe					
1002	81°36'N	16°40'W	Nord	Greenland	Den
1004	78°55'N	11°56'E	Ny-Ålesund	Svalbard	Nor
1005	78°13'N	15°39'E	Longyearbyen	Svalbard	Nor
1006	77°28'N	69°14'W	Qaanaaq	Greenland	Den
1008	73°30'N	80°31'E	Dikson	Krasnoyarsk Krai	Rus
1009	72°47'N	56°09'W	Upernavik	Greenland	Den
1010	71°39'N	128°52'E	Tiksi	Sakha Republic	Rus

4.1.29 CheckLineWidth

Gets or sets the check line's width in pixels for the checkbox(es) for the grid in readonly mode.

[Visual Basic .NET]

```
Public Property CheckLineWidth As Integer
```

[C#]

```
public int CheckLineWidth {get; set;}
```

Remarks

The default value is 1, and it can only be either 1 or 2.

4.1.30 Columns

Gets or sets the settings of the columns for the grid control.

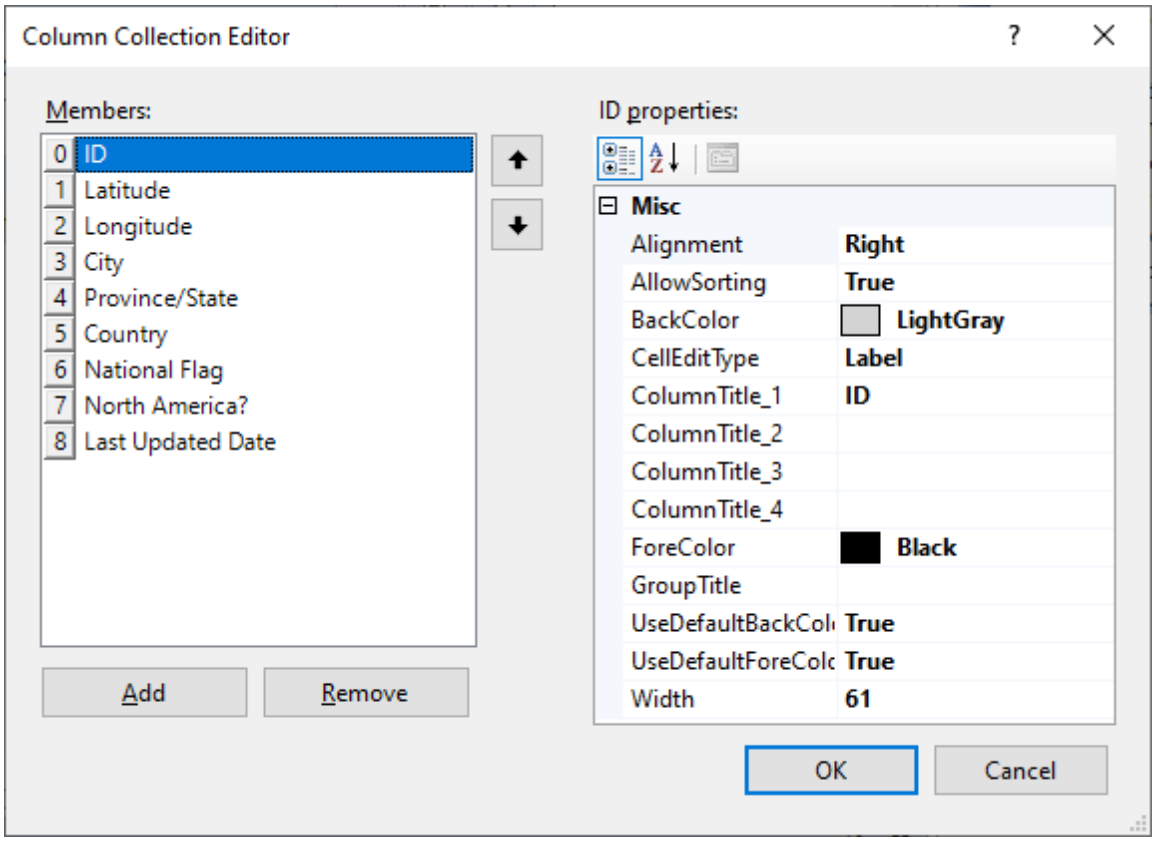
[Visual Basic .NET]

```
Public Property Columns As System.Collections.CollectionBase
```

[C#]

```
public System.Collections.CollectionBase Columns {get; set;}
```

Remarks



From the screenshot above, you can see that each column has the following fields:

1. **Alignment** specifies the column text alignment, which can be Left, Center, or Right.
2. **AllowSorting** indicates whether it is allowed to sort the rows when the column header is mouse clicked. If the section(s) are present, the sorting will be applied against the child rows of each section.
3. **CellEditType** specifies the cell edit type for the grid in edit mode. It becomes irrelevant for the grid in readonly mode.
4. **BackColor** specifies the background color of the column header, and it becomes irrelevant if **UseDefaultBackColor** is TRUE.
5. **ColumnTitle_1** specifies the 1st line of the column header.
6. **ColumnTitle_2** specifies the 2nd line of the column header.
7. **ColumnTitle_3** specifies the 3rd line of the column header.
8. **ColumnTitle_4** specifies the 4th line of the column header.
9. **ForeColor** specifies the text color of the column header, and it becomes irrelevant if **UseDefaultForeColor** is TRUE.
10. **GroupTitle** specifies the group title of the column header.
11. **UseDefaultBackColor** indicates whether the background color of the column header is dictated by the GridBackColor property or not.
12. **UseDefaultForeColor** indicates whether the text color of the column header is dictated by the HeaderTextColor property or not.
13. **Width** specifies the width in pixels of the column.

4.1.31 EditBatchDirtyCellBkColor

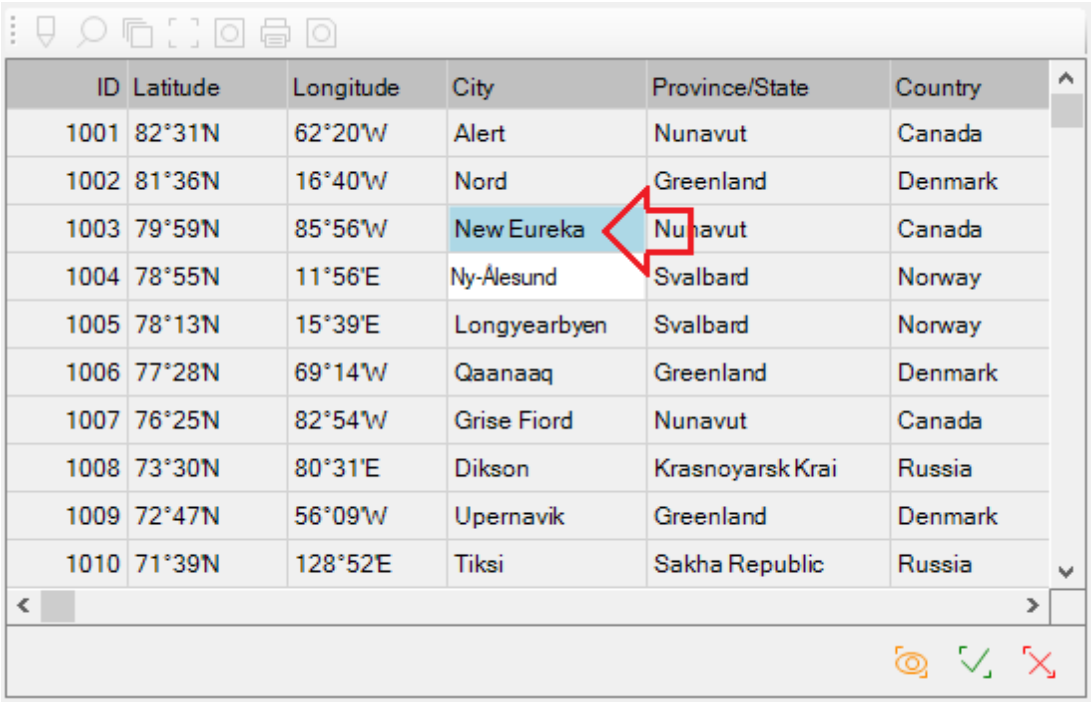
Gets or sets the background color for dirty cell(s) for the grid in batch edit mode.

```
[Visual Basic .NET]
Public Property EditBatchDirtyCellBkColor As Color
```

```
[C#]
public Color EditBatchDirtyCellBkColor {get; set;}
```

Remarks

The default value is light blue, and a screenshot is listed below.



ID	Latitude	Longitude	City	Province/State	Country
1001	82°31'N	62°20'W	Alert	Nunavut	Canada
1002	81°36'N	16°40'W	Nord	Greenland	Denmark
1003	79°59'N	85°56'W	New Eureka	Nunavut	Canada
1004	78°55'N	11°56'E	Ny-Ålesund	Svalbard	Norway
1005	78°13'N	15°39'E	Longyearbyen	Svalbard	Norway
1006	77°28'N	69°14'W	Qaanaaq	Greenland	Denmark
1007	76°25'N	82°54'W	Grise Fiord	Nunavut	Canada
1008	73°30'N	80°31'E	Dikson	Krasnoyarsk Krai	Russia
1009	72°47'N	56°09'W	Upernavik	Greenland	Denmark
1010	71°39'N	128°52'E	Tiksi	Sakha Republic	Russia

4.1.32 EditIconCancelColor

Gets or sets the color for Cancel Icon to cancel editing row(s) for the grid in edit mode - either in row editing or batch editing.

[Visual Basic .NET]

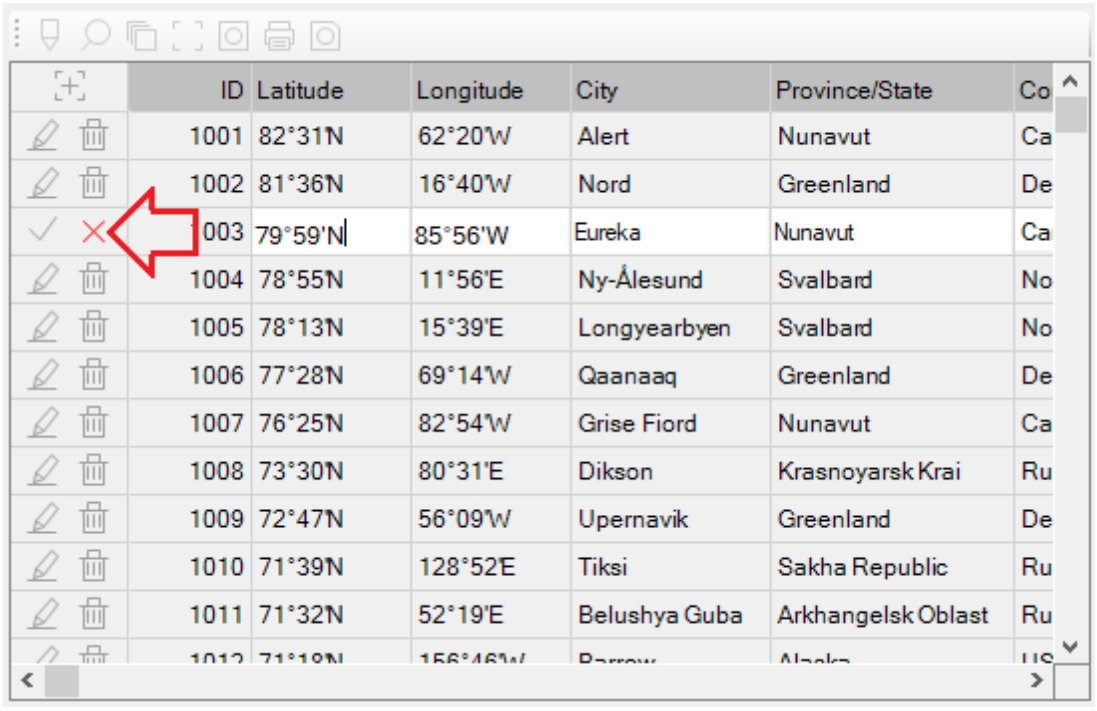
```
Public Property EditIconCancelColor As Color
```

[C#]

```
public Color EditIconCancelColor {get; set;}
```

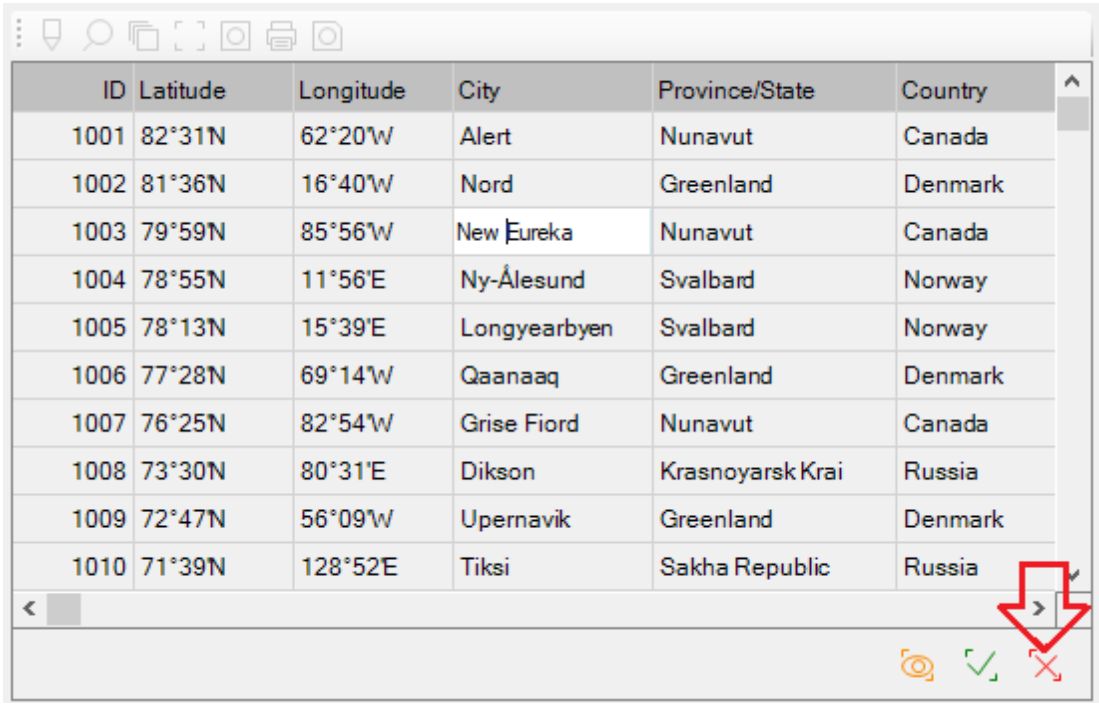
Remarks

The default value is red, and 2 screenshots are listed below.



ID	Latitude	Longitude	City	Province/State	Co
1001	82°31'N	62°20'W	Alert	Nunavut	Ca
1002	81°36'N	16°40'W	Nord	Greenland	De
1003	79°59'N	85°56'W	Eureka	Nunavut	Ca
1004	78°55'N	11°56'E	Ny-Ålesund	Svalbard	No
1005	78°13'N	15°39'E	Longyearbyen	Svalbard	No
1006	77°28'N	69°14'W	Qaanaaq	Greenland	De
1007	76°25'N	82°54'W	Grise Fiord	Nunavut	Ca
1008	73°30'N	80°31'E	Dikson	Krasnoyarsk Krai	Ru
1009	72°47'N	56°09'W	Upernavik	Greenland	De
1010	71°39'N	128°52'E	Tiksi	Sakha Republic	Ru
1011	71°32'N	52°19'E	Belushya Guba	Arkhangelsk Oblast	Ru

1: In Row Editing



ID	Latitude	Longitude	City	Province/State	Country
1001	82°31'N	62°20'W	Alert	Nunavut	Canada
1002	81°36'N	16°40'W	Nord	Greenland	Denmark
1003	79°59'N	85°56'W	New Eureka	Nunavut	Canada
1004	78°55'N	11°56'E	Ny-Ålesund	Svalbard	Norway
1005	78°13'N	15°39'E	Longyearbyen	Svalbard	Norway
1006	77°28'N	69°14'W	Qaanaaq	Greenland	Denmark
1007	76°25'N	82°54'W	Grise Fiord	Nunavut	Canada
1008	73°30'N	80°31'E	Dikson	Krasnoyarsk Krai	Russia
1009	72°47'N	56°09'W	Upernavik	Greenland	Denmark
1010	71°39'N	128°52'E	Tiksi	Sakha Republic	Russia

2: Batch Editing

4.1.33 EditIconGrayedoutColor

Gets or sets the color for grayed out icons for the grid in edit mode.

[Visual Basic .NET]

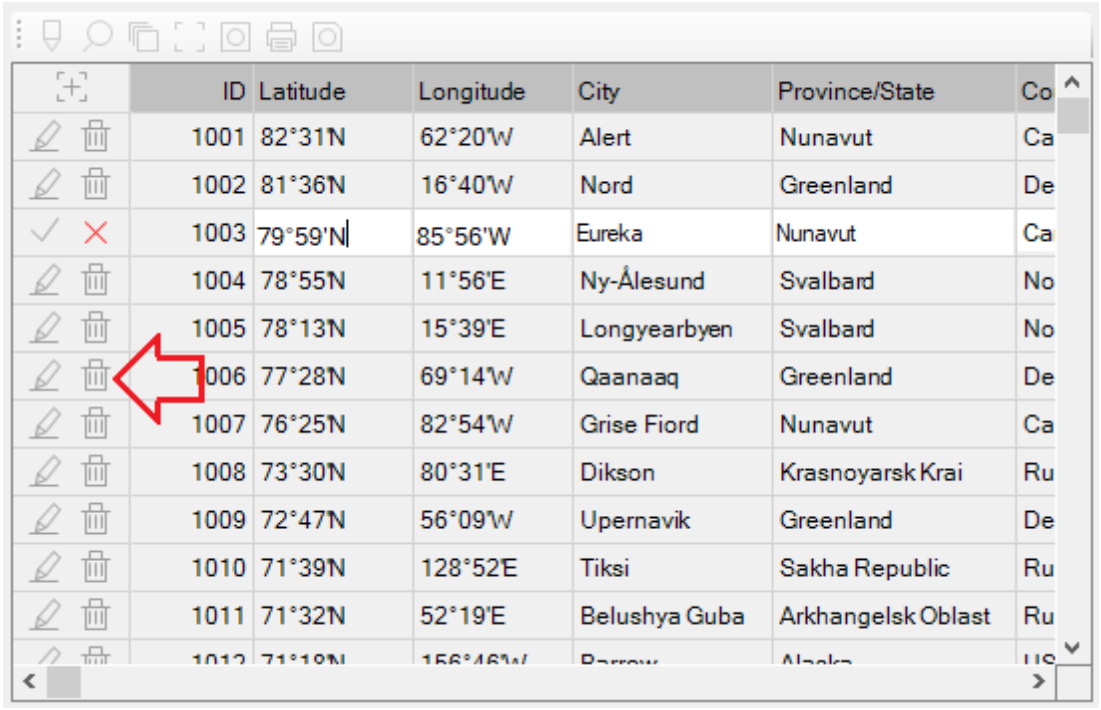
```
Public Property EditIconGrayedoutColor As Color
```

[C#]

```
public Color EditIconGrayedoutColor {get; set;}
```

Remarks

The default value is ControlDark, and a screenshot is listed below.



4.1.34 EditIconUpdateColor

Gets or sets the color for Update Icon to confirm editing row(s) for the grid in edit mode - either in row editing or batch editing.

[Visual Basic .NET]

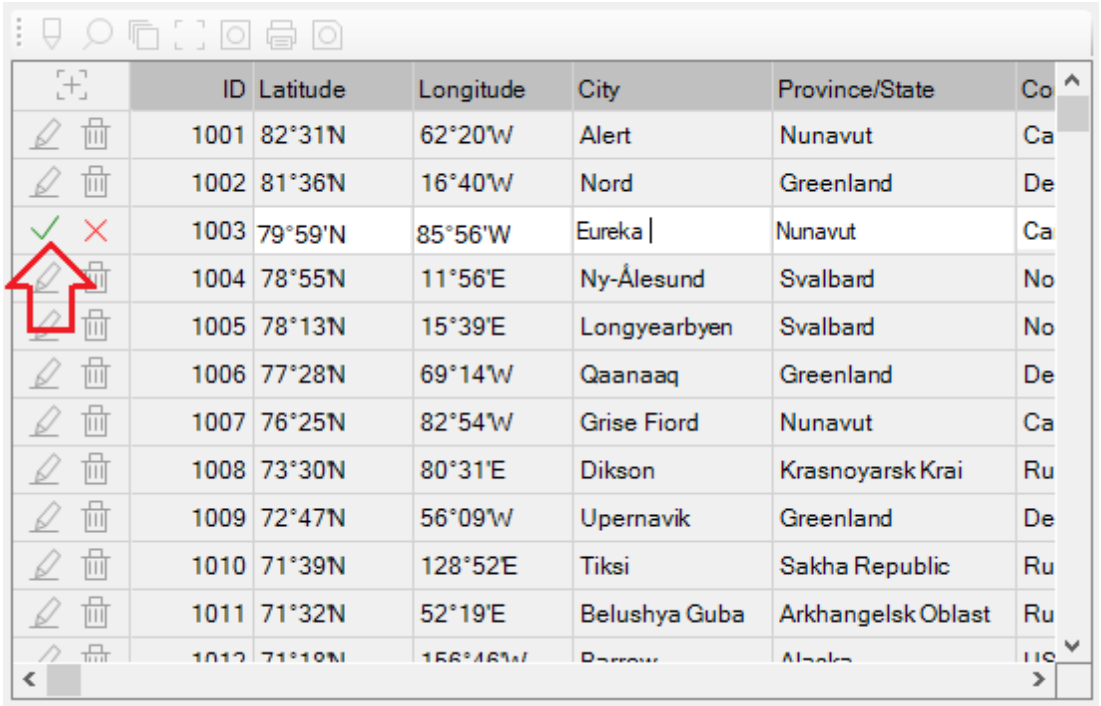
```
Public Property EditIconUpdateColor As Color
```

[C#]

```
public Color EditIconUpdateColor {get; set;}
```

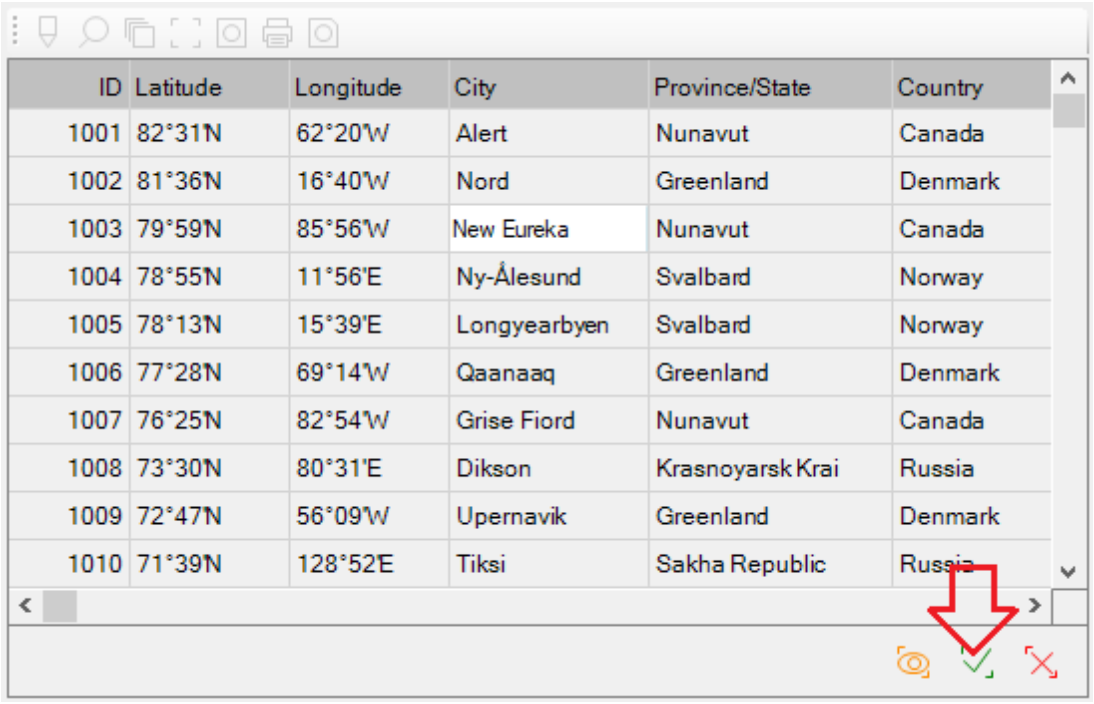
Remarks

The default value is green, and 2 screenshots are listed below.



ID	Latitude	Longitude	City	Province/State	Co
1001	82°31'N	62°20'W	Alert	Nunavut	Ca
1002	81°36'N	16°40'W	Nord	Greenland	De
1003	79°59'N	85°56'W	Eureka	Nunavut	Ca
1004	78°55'N	11°56'E	Ny-Ålesund	Svalbard	No
1005	78°13'N	15°39'E	Longyearbyen	Svalbard	No
1006	77°28'N	69°14'W	Qaanaaq	Greenland	De
1007	76°25'N	82°54'W	Grise Fiord	Nunavut	Ca
1008	73°30'N	80°31'E	Dikson	Krasnoyarsk Krai	Ru
1009	72°47'N	56°09'W	Upernavik	Greenland	De
1010	71°39'N	128°52'E	Tiksi	Sakha Republic	Ru
1011	71°32'N	52°19'E	Belushya Guba	Arkhangelsk Oblast	Ru

1: In Row Editing



ID	Latitude	Longitude	City	Province/State	Country
1001	82°31'N	62°20'W	Alert	Nunavut	Canada
1002	81°36'N	16°40'W	Nord	Greenland	Denmark
1003	79°59'N	85°56'W	New Eureka	Nunavut	Canada
1004	78°55'N	11°56'E	Ny-Ålesund	Svalbard	Norway
1005	78°13'N	15°39'E	Longyearbyen	Svalbard	Norway
1006	77°28'N	69°14'W	Qaanaaq	Greenland	Denmark
1007	76°25'N	82°54'W	Grise Fiord	Nunavut	Canada
1008	73°30'N	80°31'E	Dikson	Krasnoyarsk Krai	Russia
1009	72°47'N	56°09'W	Upernavik	Greenland	Denmark
1010	71°39'N	128°52'E	Tiksi	Sakha Republic	Russia

2: Batch Editing

4.1.35 EditImageCellKeepAspectRatio

Gets or sets a boolean flag indicating whether the aspect ratio is kept to display cell images for the grid in edit mode.

[Visual Basic .NET]

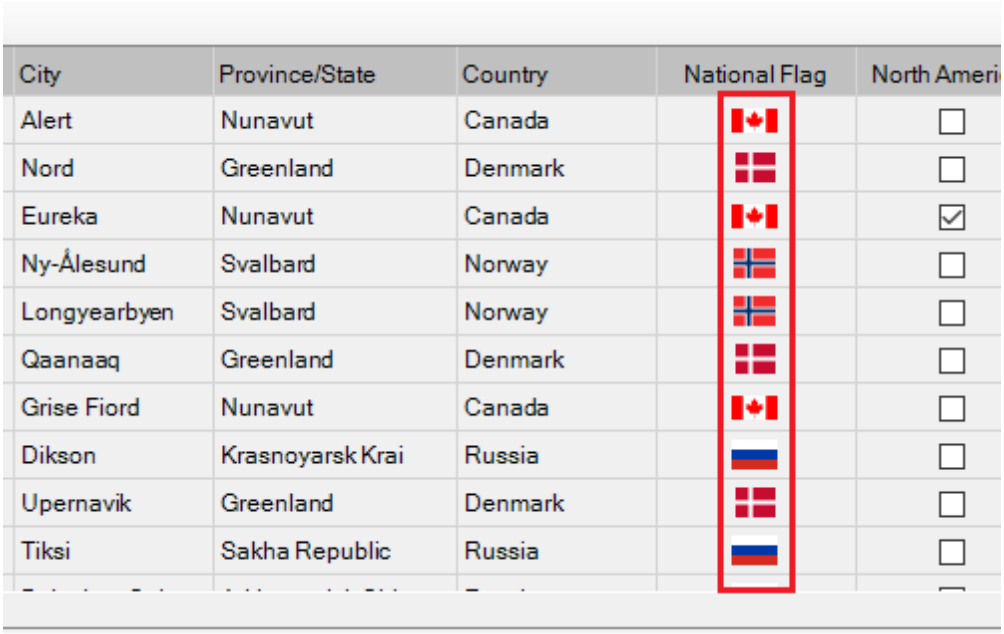
```
Public Property EditImageCellKeepAspectRatio As Boolean
```

[C#]

```
public bool EditImageCellKeepAspectRatio {get; set;}
```

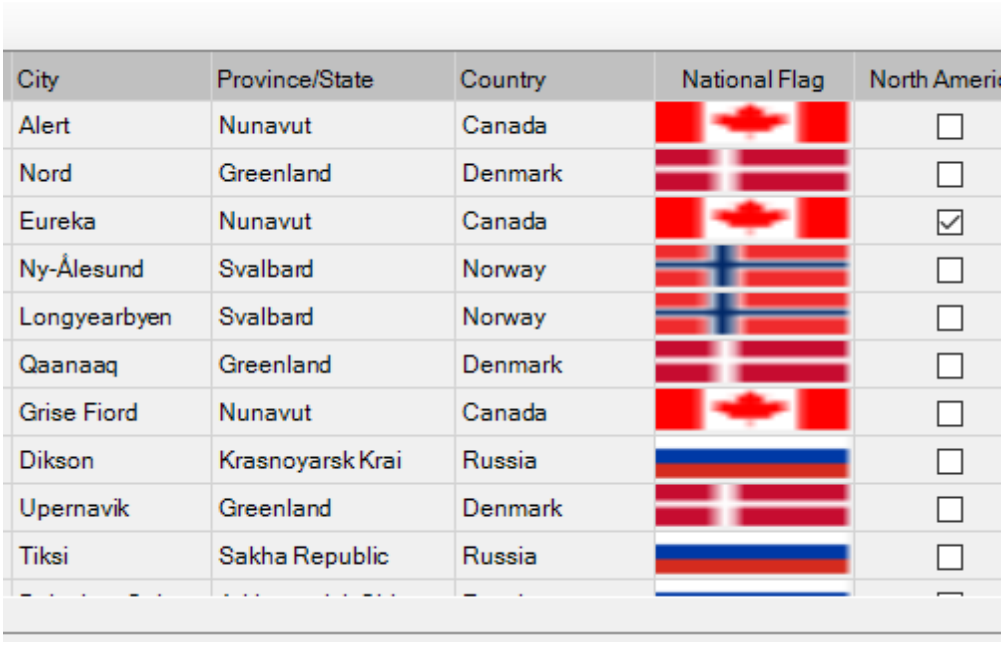
Remarks

The default value is TRUE, and a screenshot is listed below.



City	Province/State	Country	National Flag	North Ameri
Alert	Nunavut	Canada		☐
Nord	Greenland	Denmark		☐
Eureka	Nunavut	Canada		☑
Ny-Ålesund	Svalbard	Norway		☐
Longyearbyen	Svalbard	Norway		☐
Qaanaaq	Greenland	Denmark		☐
Grise Fiord	Nunavut	Canada		☐
Dikson	Krasnoyarsk Krai	Russia		☐
Upernavik	Greenland	Denmark		☐
Tiksi	Sakha Republic	Russia		☐

If it is set to FALSE, you will see something like this below - images are rendered in entire cell area.



City	Province/State	Country	National Flag	North Ameri
Alert	Nunavut	Canada		☐
Nord	Greenland	Denmark		☐
Eureka	Nunavut	Canada		☑
Ny-Ålesund	Svalbard	Norway		☐
Longyearbyen	Svalbard	Norway		☐
Qaanaaq	Greenland	Denmark		☐
Grise Fiord	Nunavut	Canada		☐
Dikson	Krasnoyarsk Krai	Russia		☐
Upernavik	Greenland	Denmark		☐
Tiksi	Sakha Republic	Russia		☐

4.1.36 EditMode

Gets or sets the edit mode option for the grid in edit mode.

```
[Visual Basic .NET]
Public Property EditMode As enumEditMode
```

```
[C#]
public enumEditMode EditMode {get; set;}
```

Remarks

The default value is InRow with the value of 1. Currently, the grid doesn't support EditPanel or EditPanelAndShowRow for non-rectangle layout.

4.1.37 EditPanelBackColor

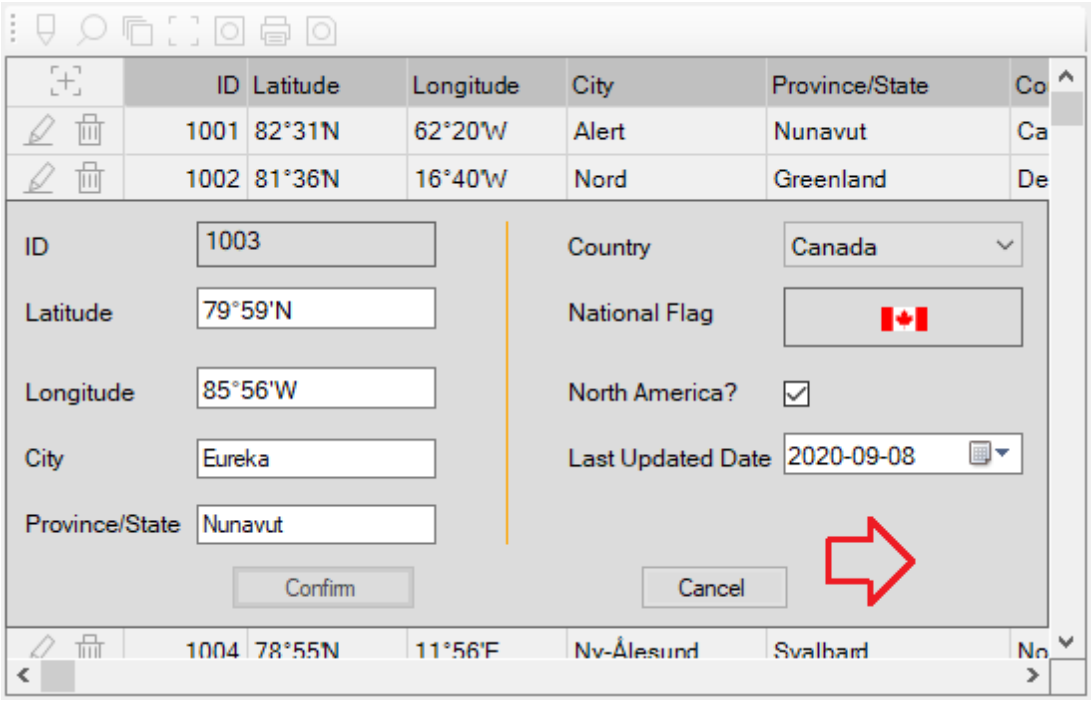
Gets or sets the background color for edit panel if EditMode property is EditPanel, EditPanelAndShowRow, or PopupEditWindow for the grid in edit mode.

```
[Visual Basic .NET]
Public Property EditPanelBackColor As Color
```

```
[C#]
public Color EditPanelBackColor {get; set;}
```

Remarks

The default value is ControlLight, and a screenshot is listed below.



4.1.38 EditPanelControlWidth

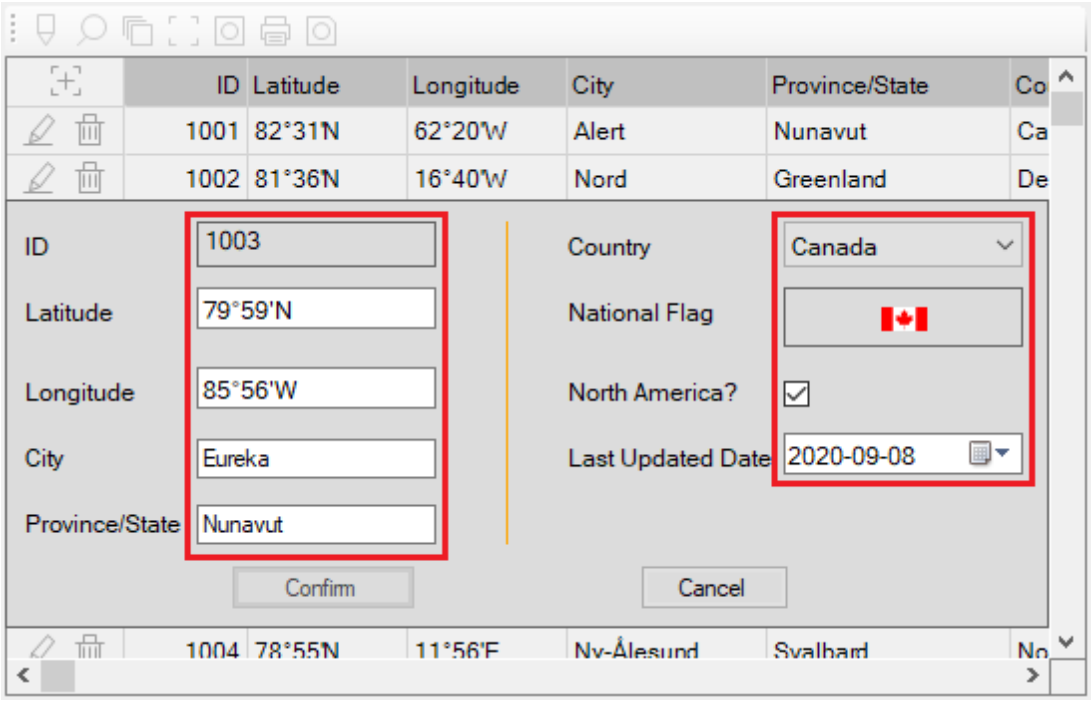
Gets or sets the width in pixels of edit controls in edit panel if EditMode property is EditPanel, EditPanelAndShowRow, or PopupEditWindow for the grid in edit mode.

```
[Visual Basic .NET]
Public Property EditPanelControlWidth As Integer
```

```
[C#]
public int EditPanelControlWidth {get; set;}
```

Remarks

The default value is 120 and a valid value should be between 50 and 500. A screenshot is listed below.



4.1.39 EditPanelDividerColor

Gets or sets the divider color for edit panel if EditMode property is EditPanel, EditPanelAndShowRow, or PopupEditWindow for the grid in edit mode.

[Visual Basic .NET]

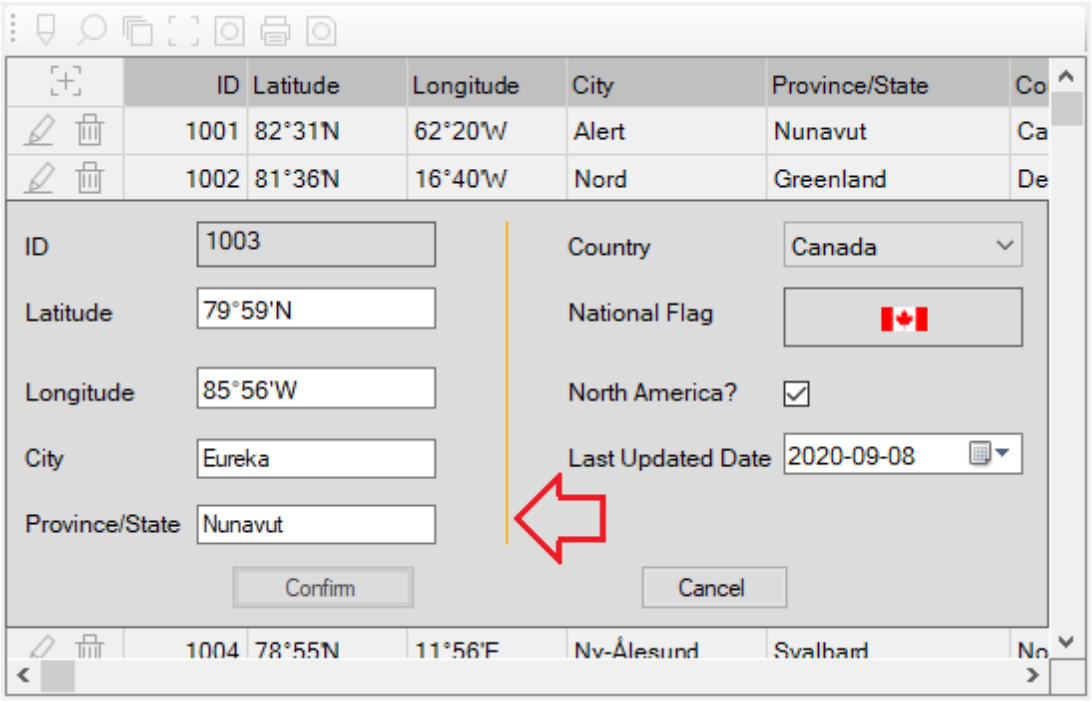
```
Public Property EditPanelDividerColor As Color
```

[C#]

```
public Color EditPanelDividerColor {get; set;}
```

Remarks

The default value is orange, and a screenshot is listed below.



4.1.40 EditPanelImageHeight

Gets or sets the height in pixels of picture box in edit panel if EditMode property is EditPanel, EditPanelAndShowRow, or PopupEditWindow for the grid in edit mode.

[Visual Basic .NET]

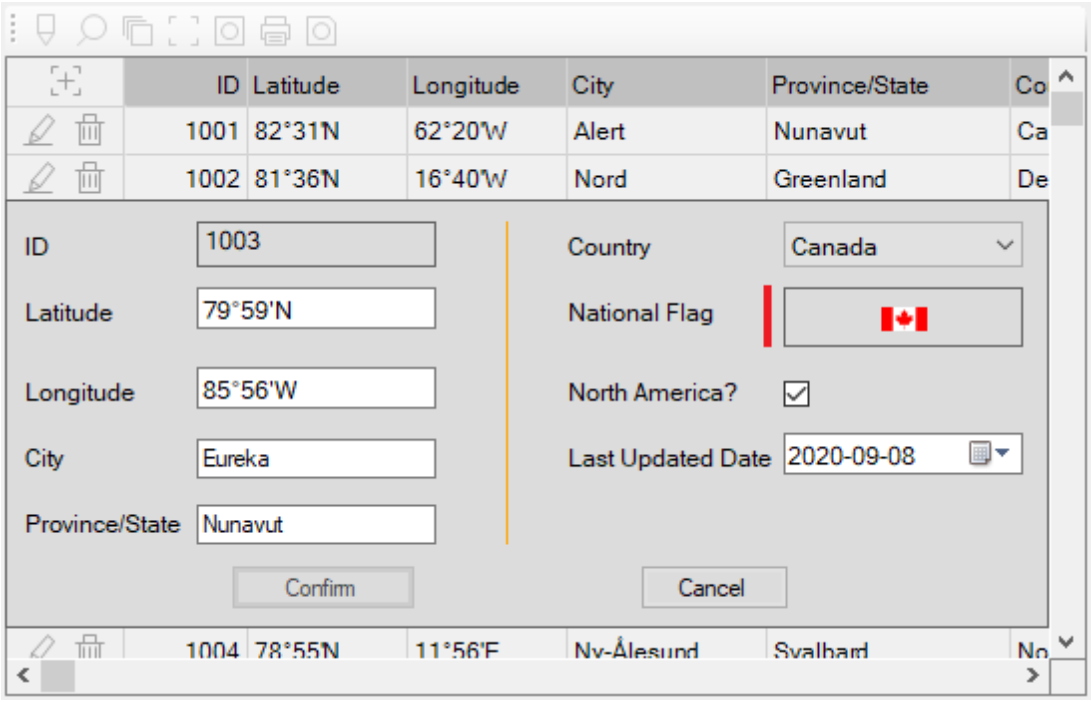
```
Public Property EditPanelImageHeight As Integer
```

[C#]

```
public int EditPanelImageHeight {get; set;}
```

Remarks

The default value is 80 and a valid value should be between 30 and 500. A screenshot is listed below.



4.1.41 EditPanelLeftLabelAlignment

Gets or sets a boolean flag indicating whether the labels should be left aligned or not if EditMode property is EditPanel, EditPanelAndShowRow, or PopupEditWindow for the grid in edit mode.

[Visual Basic .NET]

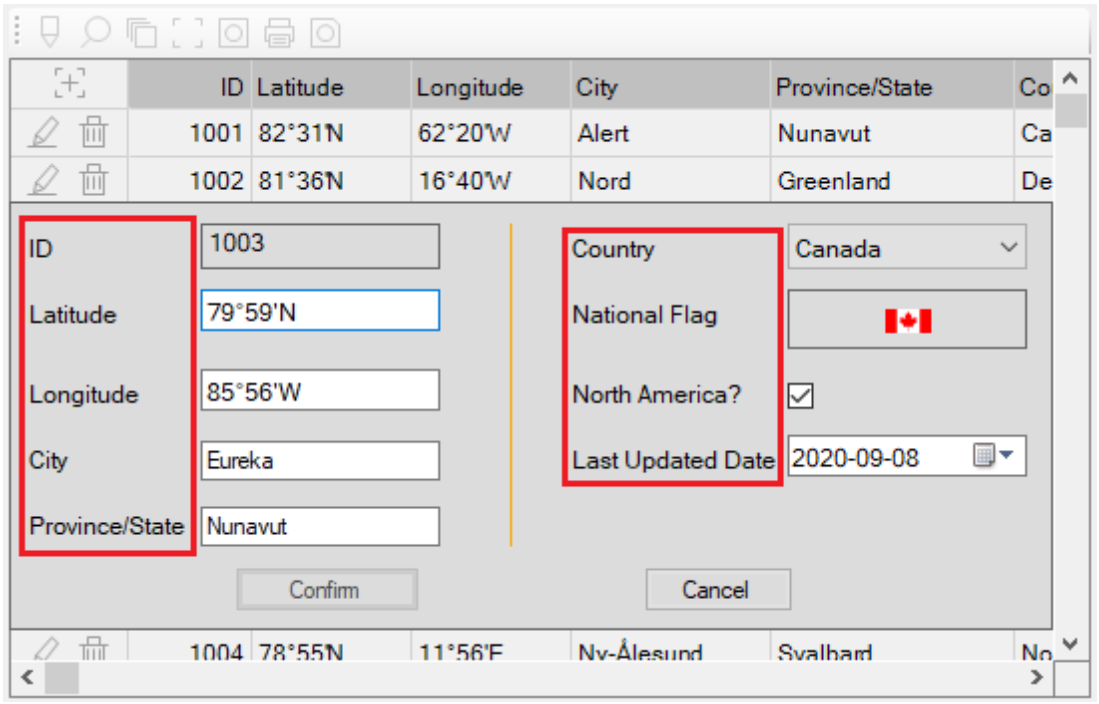
```
Public Property EditPanelLeftLabelAlignment As Boolean
```

[C#]

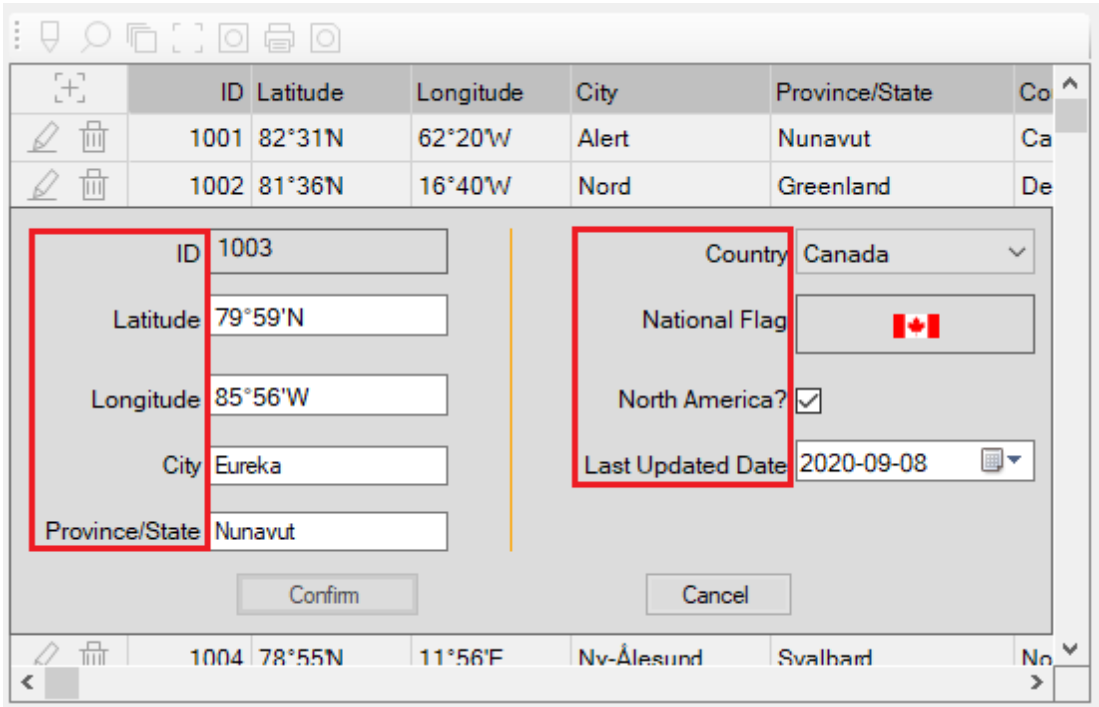
```
public bool EditPanelLeftLabelAlignment {get; set;}
```

Remarks

The default value is TRUE, and a screenshot is listed below.



If it is FALSE, all labels are right aligned.



4.1.42 EditPanelRichTextHeight

Gets or sets the height in pixels of rich text controls in edit panel if EditMode property is EditPanel, EditPanelAndShowRow, or PopupEditWindow for the grid in edit mode.

[Visual Basic .NET]

```
Public Property EditPanelRichTextHeight As Integer
```

[C#]

```
public int EditPanelRichTextHeight {get; set;}
```

Remarks

The default value is 40 and a valid value should be between 20 and 200.

4.1.43 EditPanelRowMajorOrder

Gets or sets a boolean flag indicating whether all controls in edit panel are populated in row- or column-major order if EditMode property is EditPanel, EditPanelAndShowRow, or PopupEditWindow for the grid in edit mode.

[Visual Basic .NET]

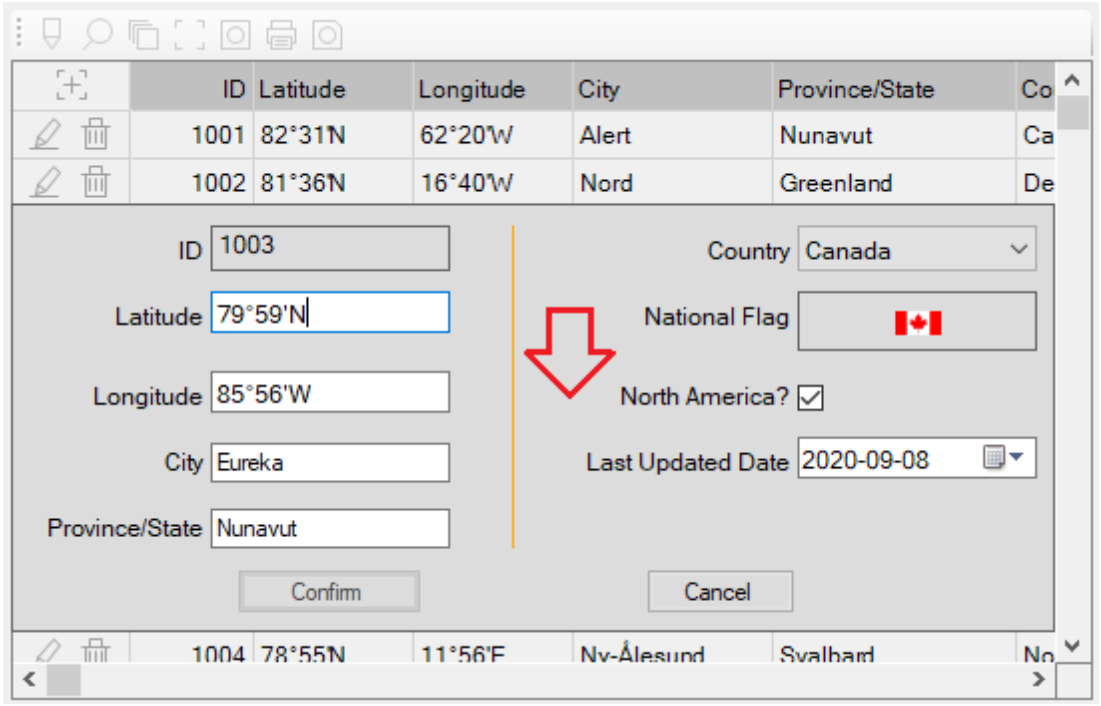
```
Public Property EditPanelRowMajorOrder As Boolean
```

[C#]

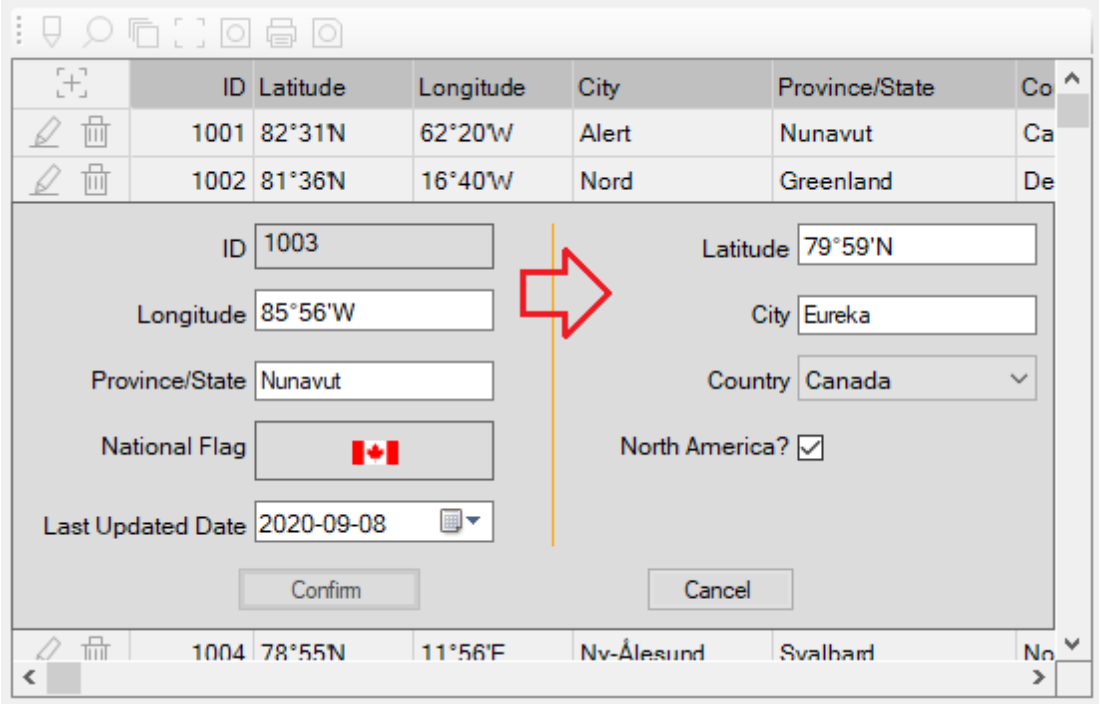
```
public bool EditPanelRowMajorOrder {get; set;}
```

Remarks

The default value is FALSE, and a screenshot is listed below.



It it is TRUE, all controls are populated in row-major order.



4.1.44 EditRowEventTimeout

Gets or sets the timeout in milliseconds when adding, updating, deleting row(s) for the grid in edit mode.

[Visual Basic .NET]

```
Public Property EditRowEventTimeout As Integer
```

[C#]

```
public int EditRowEventTimeout {get; set;}
```

Remarks

The default value is 5000 (5 seconds) and the minimum value is 1000 (1 second).

4.1.45 ElasticCordColor

Gets or sets the color of the elastic cord when resizing columns.

[Visual Basic .NET]

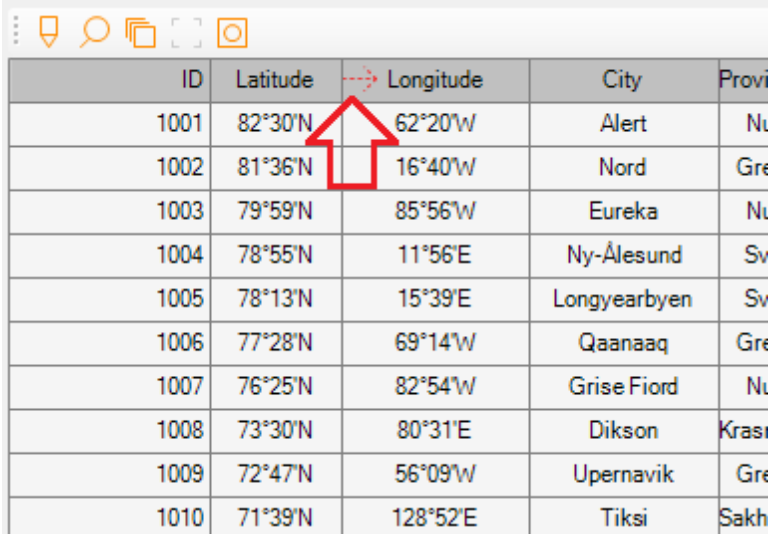
```
Public Property ElasticCordColor As Color
```

[C#]

```
public Color ElasticCordColor {get; set;}
```

Remarks

The default value is red.



ID	Latitude	Longitude	City	Provi
1001	82°30'N	62°20'W	Alert	Nu
1002	81°36'N	16°40'W	Nord	Gre
1003	79°59'N	85°56'W	Eureka	Nu
1004	78°55'N	11°56'E	Ny-Ålesund	Sv
1005	78°13'N	15°39'E	Longyearbyen	Sv
1006	77°28'N	69°14'W	Qaanaaq	Gre
1007	76°25'N	82°54'W	Grise Fiord	Nu
1008	73°30'N	80°31'E	Dikson	Krasi
1009	72°47'N	56°09'W	Upernavik	Gre
1010	71°39'N	128°52'E	Tiksi	Sakh

4.1.46 ElasticCordType

Gets or sets the dash pattern of the elastic cord when resizing columns.

[Visual Basic .NET]

```
Public Property ElasticCordType As enumElasticCordType
```

```
[C#]
public enumElasticCordType ElasticCordType {get; set;}
```

Remarks

The default value is DashedLine_2_2.

4.1.47 ElasticCordWidth

Gets or sets the line width in pixels for the elastic cord when resizing columns.

```
[Visual Basic .NET]
Public Property ElasticCordWidth As Integer
```

```
[C#]
public int ElasticCordWidth {get; set;}
```

Remarks

The default value is 1, and it can only be either 1 or 2.

4.1.48 FixedColor

Gets or sets the color of the fixed cell(s) including the header(s) for the grid in readonly mode.

```
[Visual Basic .NET]
Public Property FixedColor As Color
```

```
[C#]
public Color FixedColor {get; set;}
```

Remarks

The default value is Silver, and a screenshot is listed below.

ID	Latitude	Longitude	City	Province/State	Country
1001	82°30'N	62°20'W	Alert	Nunavut	Canada
1002	81°36'N	16°40'W	Nord	Greenland	Denmark
1003	79°59'N	85°56'W	Eureka	Nunavut	Canada
1004	78°55'N	11°56'E	Ny-Ålesund	Svalbard	Norway
1005	78°13'N	15°39'E	Longyearbyen	Svalbard	Norway
1006	77°28'N	69°14'W	Qaanaaq	Greenland	Denmark
1007	76°25'N	82°54'W	Grise Fiord	Nunavut	Canada
1008	73°30'N	80°31'E	Dikson	Krasnoyarsk Krai	Russia
1009	72°47'N	56°09'W	Upernavik	Greenland	Denmark
1010	71°39'N	128°52'E	Tiksi	Sakha Republic	Russia
1011	71°32'N	52°19'E	Belushya Guba	Arkhangelsk Oblast	Russia
1012	71°18'N	156°46'W	Barrow	Alaska	USA
1013	70°59'N	25°59'E	Honningsvåg	Finnmark	Norway
1014	70°40'N	23°41'E	Hammerfest	Finnmark	Norway

4.1.49 FixedColumns

Gets or sets the number of fixed column(s) that are not horizontally scrolled for the grid in readonly mode.

[Visual Basic .NET]

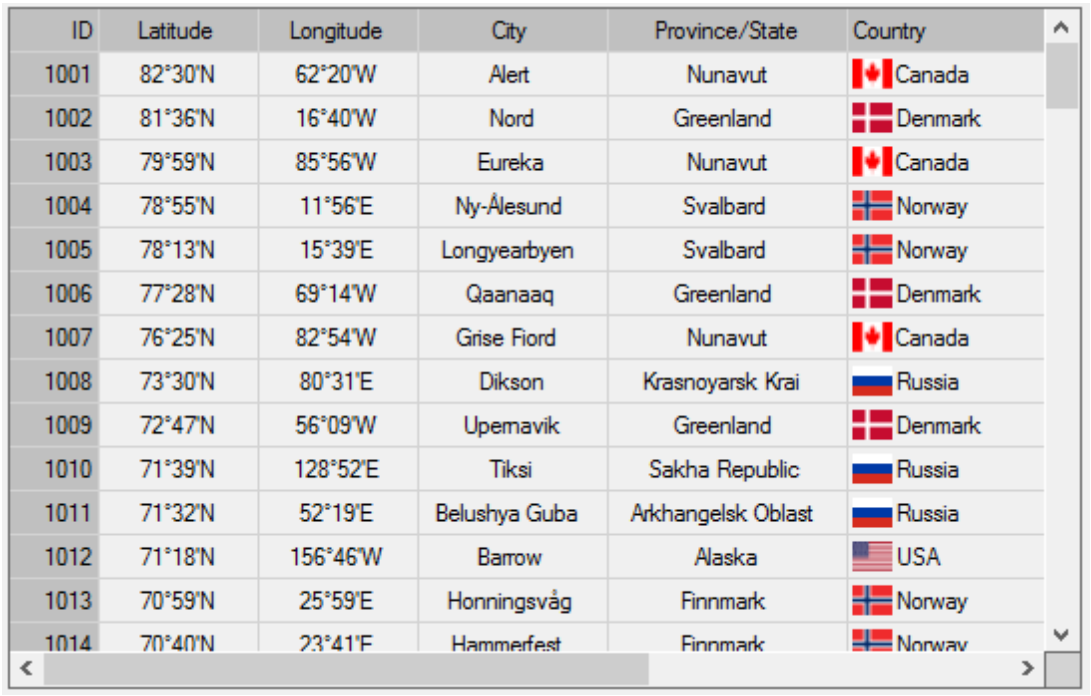
```
Public Property FixedColumns As Integer
```

[C#]

```
public int FixedColumns {get; set;}
```

Remarks

The default value is 0, and a valid value must between 0 and the number of columns. A screenshot for 1 fixed column is listed below.



ID	Latitude	Longitude	City	Province/State	Country
1001	82°30'N	62°20'W	Alert	Nunavut	Canada
1002	81°36'N	16°40'W	Nord	Greenland	Denmark
1003	79°59'N	85°56'W	Eureka	Nunavut	Canada
1004	78°55'N	11°56'E	Ny-Ålesund	Svalbard	Norway
1005	78°13'N	15°39'E	Longyearbyen	Svalbard	Norway
1006	77°28'N	69°14'W	Qaanaaq	Greenland	Denmark
1007	76°25'N	82°54'W	Grise Fiord	Nunavut	Canada
1008	73°30'N	80°31'E	Dikson	Krasnoyarsk Krai	Russia
1009	72°47'N	56°09'W	Upernavik	Greenland	Denmark
1010	71°39'N	128°52'E	Tiksi	Sakha Republic	Russia
1011	71°32'N	52°19'E	Belushya Guba	Arkhangelsk Oblast	Russia
1012	71°18'N	156°46'W	Barrow	Alaska	USA
1013	70°59'N	25°59'E	Honningsvåg	Finnmark	Norway
1014	70°40'N	23°41'E	Hammerfest	Finnmark	Norway

4.1.50 FixedLeadingPart

Gets or sets a boolean flag indicating whether the leading part of header, section(s) if applicable, or data row(s) that holds a checkbox and/or section node is horizontally scrolled or not for the grid in readonly mode.

[Visual Basic .NET]

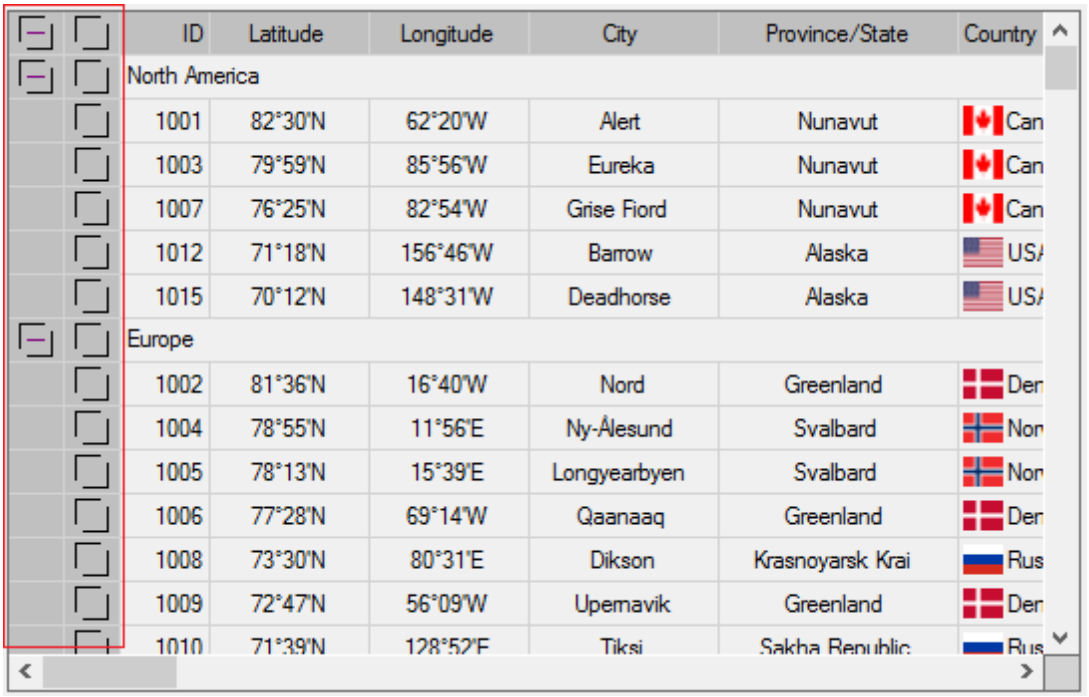
```
Public Property FixedLeadingPart As Boolean
```

[C#]

```
public bool FixedLeadingPart {get; set;}
```

Remarks

The default value is TRUE. If the property FixedColumns is nonzero, this property will be automatically set to TRUE; a screenshot is listed below.



ID	Latitude	Longitude	City	Province/State	Country
North America					
1001	82°30'N	62°20'W	Alert	Nunavut	Can
1003	79°59'N	85°56'W	Eureka	Nunavut	Can
1007	76°25'N	82°54'W	Grise Fiord	Nunavut	Can
1012	71°18'N	156°46'W	Barrow	Alaska	USA
1015	70°12'N	148°31'W	Deadhorse	Alaska	USA
Europe					
1002	81°36'N	16°40'W	Nord	Greenland	Den
1004	78°55'N	11°56'E	Ny-Ålesund	Svalbard	Nor
1005	78°13'N	15°39'E	Longyearbyen	Svalbard	Nor
1006	77°28'N	69°14'W	Qaanaaq	Greenland	Den
1008	73°30'N	80°31'E	Dikson	Krasnoyarsk Krai	Rus
1009	72°47'N	56°09'W	Upernavik	Greenland	Den
1010	71°39'N	128°52'E	Tiksi	Sakha Republic	Rus

4.1.51 FixedRows

Gets or sets the number of fixed row(s) that are not vertically scrolled for the grid in readonly mode.

[Visual Basic .NET]

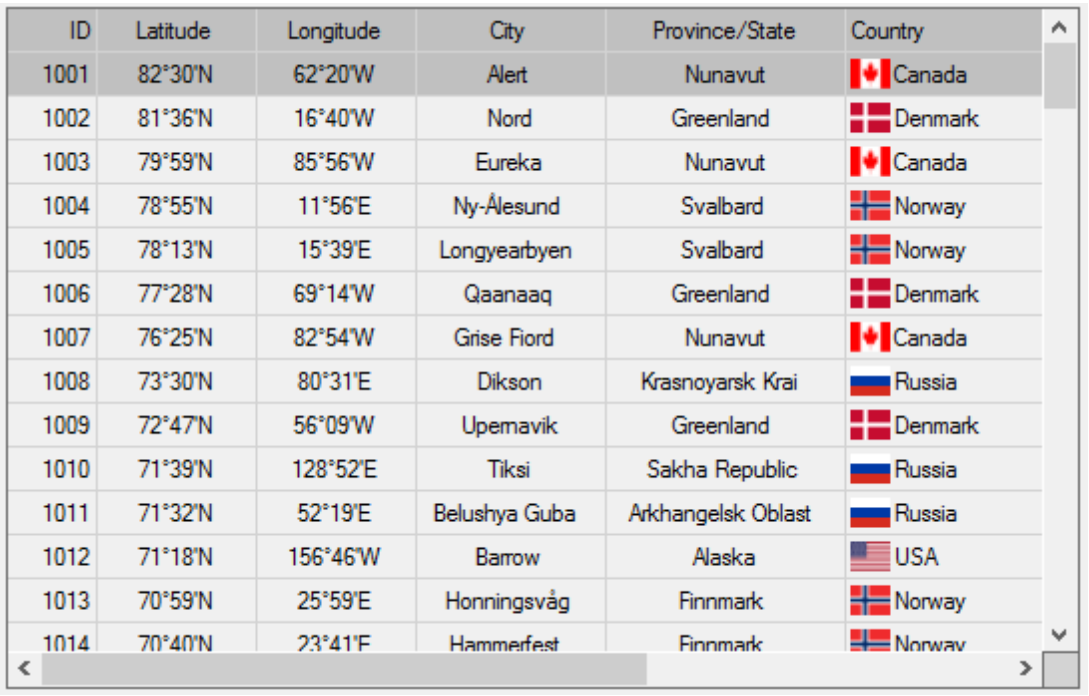
```
Public Property FixedRows As Integer
```

[C#]

```
public int FixedRows {get; set;}
```

Remarks

If the AllowSections property is set to TRUE, this property becomes irrelevant. Otherwise, the default value is 0, and a valid value must between 0 and the number of rows. A screenshot for 1 fixed row is listed below.



ID	Latitude	Longitude	City	Province/State	Country
1001	82°30'N	62°20'W	Alert	Nunavut	Canada
1002	81°36'N	16°40'W	Nord	Greenland	Denmark
1003	79°59'N	85°56'W	Eureka	Nunavut	Canada
1004	78°55'N	11°56'E	Ny-Ålesund	Svalbard	Norway
1005	78°13'N	15°39'E	Longyearbyen	Svalbard	Norway
1006	77°28'N	69°14'W	Qaanaaq	Greenland	Denmark
1007	76°25'N	82°54'W	Grise Fiord	Nunavut	Canada
1008	73°30'N	80°31'E	Dikson	Krasnoyarsk Krai	Russia
1009	72°47'N	56°09'W	Upernavik	Greenland	Denmark
1010	71°39'N	128°52'E	Tiksi	Sakha Republic	Russia
1011	71°32'N	52°19'E	Belushya Guba	Arkhangelsk Oblast	Russia
1012	71°18'N	156°46'W	Barrow	Alaska	USA
1013	70°59'N	25°59'E	Honningsvåg	Finnmark	Norway
1014	70°40'N	23°41'E	Hammerfest	Finnmark	Norway

4.1.52 GridBackColor

Gets or sets the background color of the grid.

```
[Visual Basic .NET]
Public Property GridBackColor As Color
```

```
[C#]
public Color GridBackColor {get; set;}
```

Remarks

The default value is Control.

4.1.53 GridLineColor

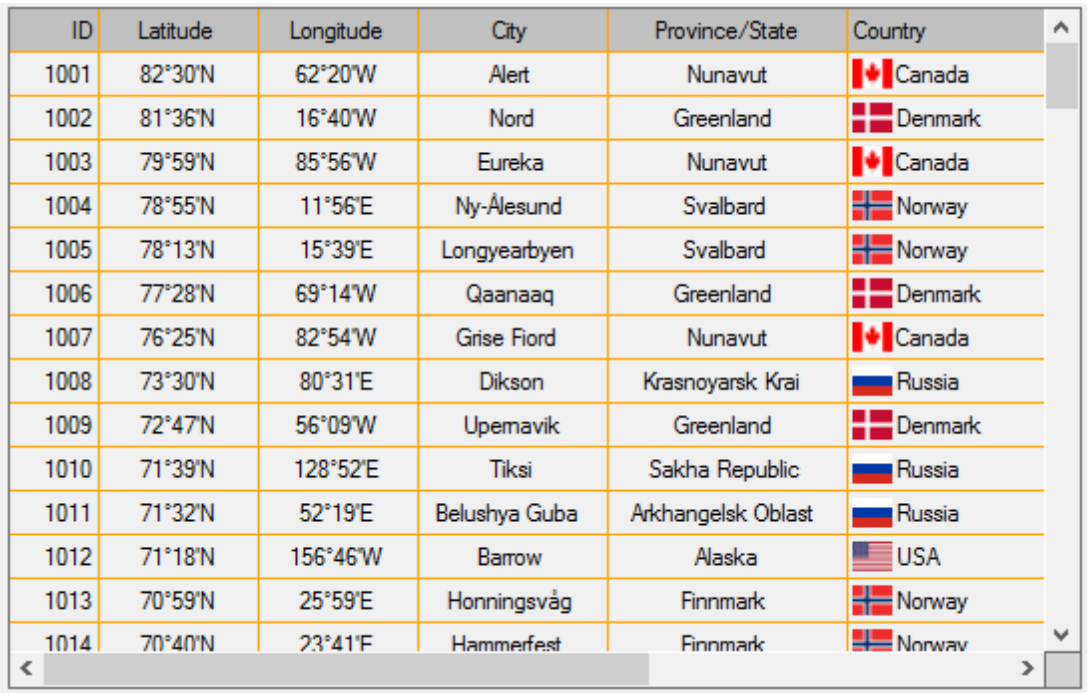
Gets or sets the color of the grid line.

```
[Visual Basic .NET]
Public Property GridLineColor As Color
```

```
[C#]
public Color GridLineColor {get; set;}
```

Remarks

The default value is light gray, and a screenshot for the orange color is listed below.



ID	Latitude	Longitude	City	Province/State	Country
1001	82°30'N	62°20'W	Alert	Nunavut	Canada
1002	81°36'N	16°40'W	Nord	Greenland	Denmark
1003	79°59'N	85°56'W	Eureka	Nunavut	Canada
1004	78°55'N	11°56'E	Ny-Ålesund	Svalbard	Norway
1005	78°13'N	15°39'E	Longyearbyen	Svalbard	Norway
1006	77°28'N	69°14'W	Qaanaaq	Greenland	Denmark
1007	76°25'N	82°54'W	Grise Fiord	Nunavut	Canada
1008	73°30'N	80°31'E	Dikson	Krasnoyarsk Krai	Russia
1009	72°47'N	56°09'W	Upernavik	Greenland	Denmark
1010	71°39'N	128°52'E	Tiksi	Sakha Republic	Russia
1011	71°32'N	52°19'E	Belushya Guba	Arkhangelsk Oblast	Russia
1012	71°18'N	156°46'W	Barrow	Alaska	USA
1013	70°59'N	25°59'E	Honningsvåg	Finnmark	Norway
1014	70°40'N	23°41'E	Hammerfest	Finnmark	Norway

4.1.54 GridLayout

Gets or sets the grid layout.

[Visual Basic .NET]

Public Property GridLayout As enumGridLayout

[C#]

public enumGridLayout GridLayout {get; set;}

Remarks

The default value is Lake_Baikal with the value of 1.

4.1.55 HeaderFont

Gets or sets the font characteristics of the header.

[Visual Basic .NET]

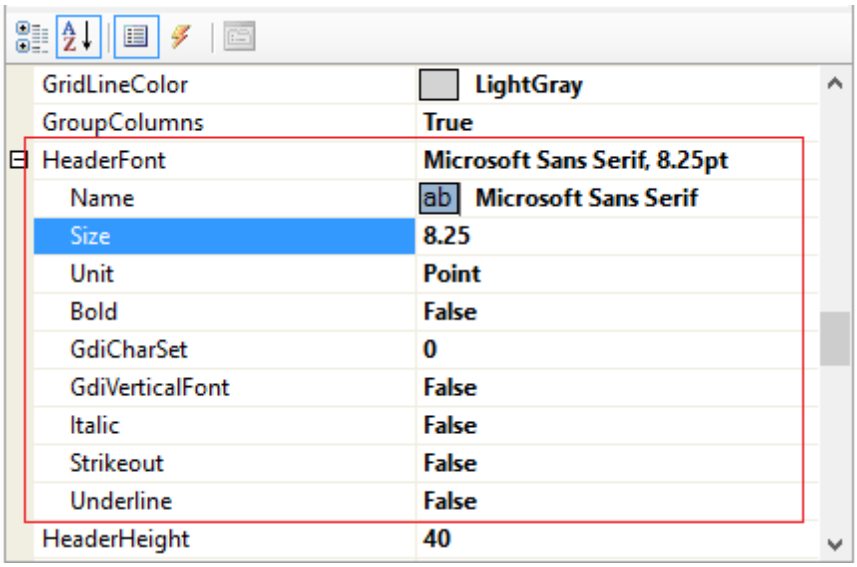
Public Property HeaderFont As Font

[C#]

public Font HeaderFont {get; set;}

Remarks

The default value is illustrated by the screenshot below:



4.1.56 HeaderHeight

Gets or sets the header height in pixels for the grid.

[Visual Basic .NET]

```
Public Property HeaderHeight As Integer
```

[C#]

```
public int HeaderHeight {get; set;}
```

Remarks

The default value is 20.

4.1.57 HeaderHeightRatio

Gets or sets the ratio between the height of the rectangle containing column group title and the height of the rectangle containing up to 4 lines of column titles.

[Visual Basic .NET]

```
Public Property HeaderHeightRatio As Double
```

[C#]

```
public double HeaderHeightRatio {get; set;}
```

Remarks

The default value is 1, and a valid value must be between 0.1 an 10.

4.1.58 HeaderTextColor

Gets or sets the text color of the header.

[Visual Basic .NET]

```
Public Property HeaderTextColor As Color
```

[C#]

```
public Color HeaderTextColor {get; set;}
```

Remarks

The default value is black.

4.1.59 IconColor

Gets or sets the color for all icons.

[Visual Basic .NET]

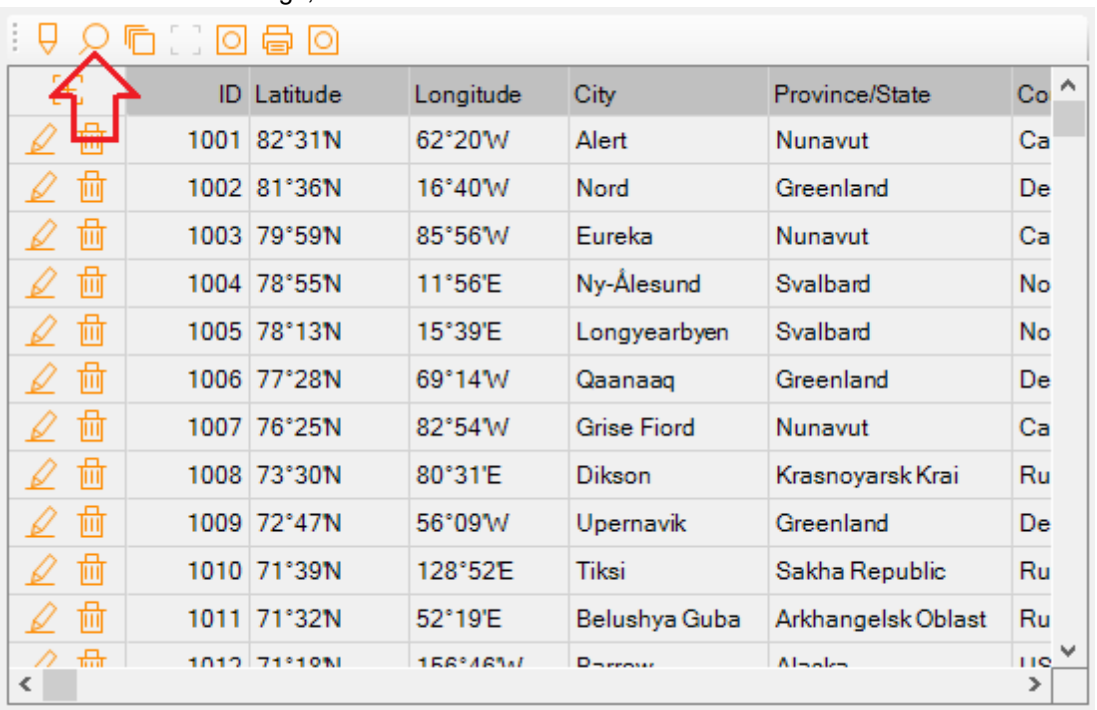
```
Public Property IconColor As Color
```

[C#]

```
public Color IconColor {get; set;}
```

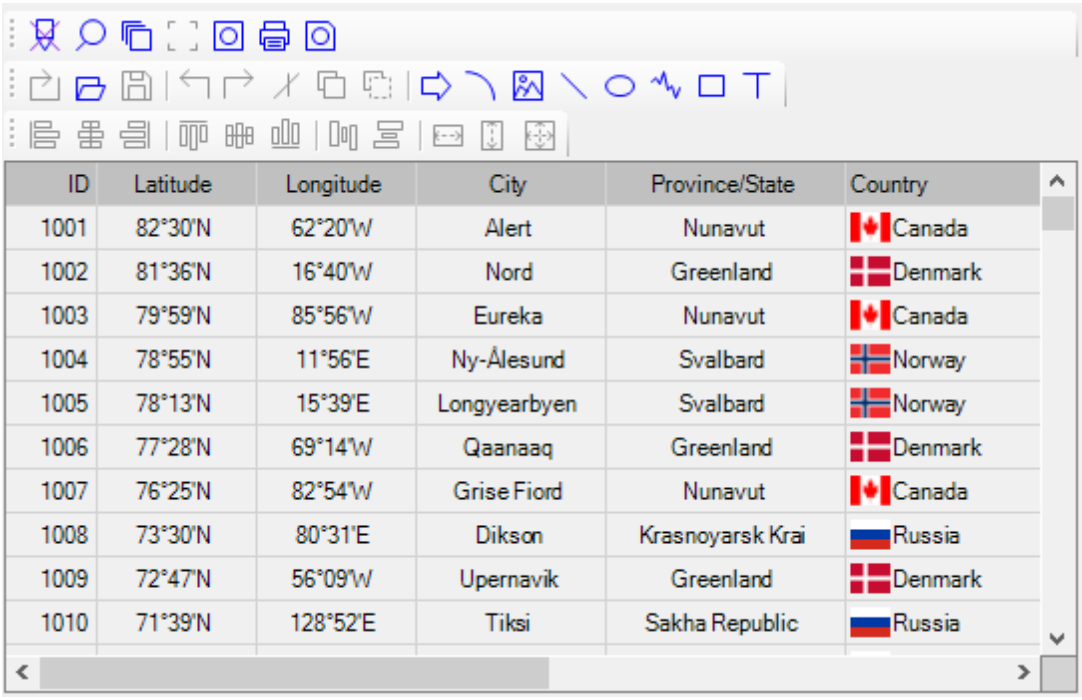
Remarks

The default value is dark orange, and a screenshot is listed below:



ID	Latitude	Longitude	City	Province/State	Co
1001	82°31'N	62°20'W	Alert	Nunavut	Ca
1002	81°36'N	16°40'W	Nord	Greenland	De
1003	79°59'N	85°56'W	Eureka	Nunavut	Ca
1004	78°55'N	11°56'E	Ny-Ålesund	Svalbard	No
1005	78°13'N	15°39'E	Longyearbyen	Svalbard	No
1006	77°28'N	69°14'W	Qaanaaq	Greenland	De
1007	76°25'N	82°54'W	Grise Fiord	Nunavut	Ca
1008	73°30'N	80°31'E	Dikson	Krasnoyarsk Krai	Ru
1009	72°47'N	56°09'W	Upernavik	Greenland	De
1010	71°39'N	128°52'E	Tiksi	Sakha Republic	Ru
1011	71°32'N	52°19'E	Belushya Guba	Arkhangelsk Oblast	Ru

A screenshot for blue color is listed below:



ID	Latitude	Longitude	City	Province/State	Country
1001	82°30'N	62°20'W	Alert	Nunavut	Canada
1002	81°36'N	16°40'W	Nord	Greenland	Denmark
1003	79°59'N	85°56'W	Eureka	Nunavut	Canada
1004	78°55'N	11°56'E	Ny-Ålesund	Svalbard	Norway
1005	78°13'N	15°39'E	Longyearbyen	Svalbard	Norway
1006	77°28'N	69°14'W	Qaanaaq	Greenland	Denmark
1007	76°25'N	82°54'W	Grise Fiord	Nunavut	Canada
1008	73°30'N	80°31'E	Dikson	Krasnoyarsk Krai	Russia
1009	72°47'N	56°09'W	Upernavik	Greenland	Denmark
1010	71°39'N	128°52'E	Tiksi	Sakha Republic	Russia

4.1.60 IconCrossColor

Gets or sets the color for the cross in icons.

[Visual Basic .NET]

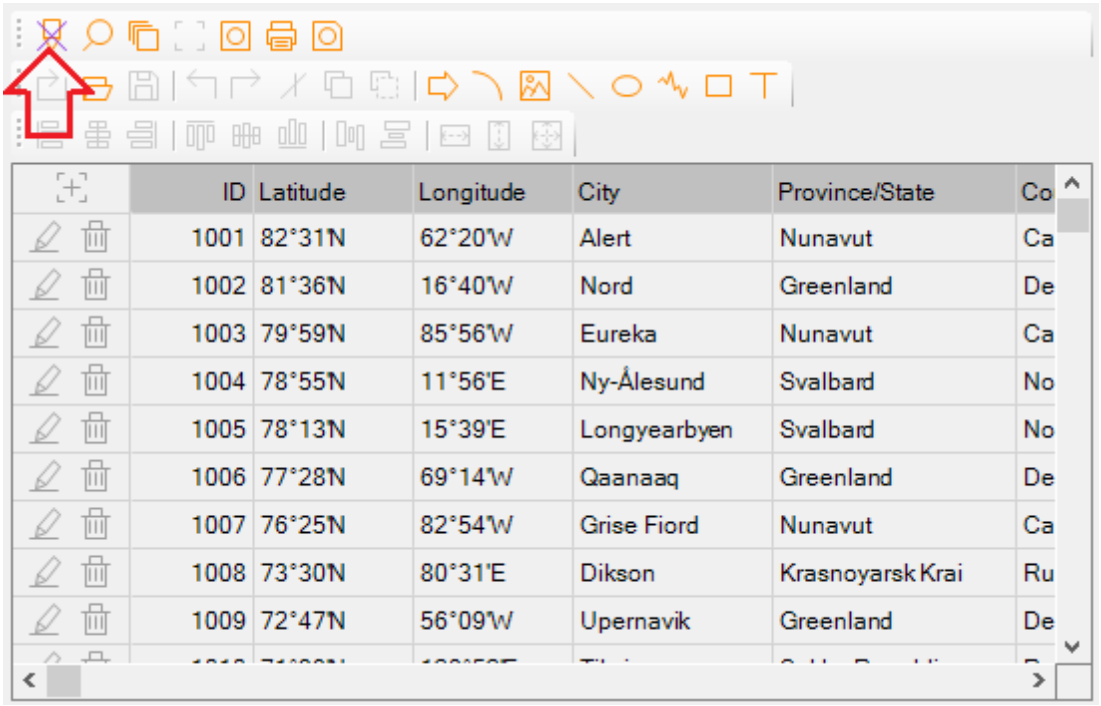
```
Public Property IconCrossColor As Color
```

[C#]

```
public Color IconCrossColor {get; set;}
```

Remarks

The default value is blue violet, and a screenshot is listed below:



	ID	Latitude	Longitude	City	Province/State	Co
	1001	82°31'N	62°20'W	Alert	Nunavut	Ca
	1002	81°36'N	16°40'W	Nord	Greenland	De
	1003	79°59'N	85°56'W	Eureka	Nunavut	Ca
	1004	78°55'N	11°56'E	Ny-Ålesund	Svalbard	No
	1005	78°13'N	15°39'E	Longyearbyen	Svalbard	No
	1006	77°28'N	69°14'W	Qaanaaq	Greenland	De
	1007	76°25'N	82°54'W	Grise Fiord	Nunavut	Ca
	1008	73°30'N	80°31'E	Dikson	Krasnoyarsk Krai	Ru
	1009	72°47'N	56°09'W	Upernavik	Greenland	De

4.1.61 LanguageOption

Gets or sets the language option.

[Visual Basic .NET]

Public Property LanguageOption As enumLanguage

[C#]

public enumLanguage LanguageOption {get; set;}

Remarks

The default value is English with the value of 16.

4.1.62 LanguageRTL

Gets or sets a boolean flag indicating whether the language indicated by the property LanguageOption is written from right to left.

[Visual Basic .NET]

Public Property LanguageRTL As Boolean

[C#]

public bool LanguageRTL {get; set;}

Remarks

The default value is FALSE.

4.1.63 LicenseCode

Gets or sets the license code for the grid control.

[Visual Basic .NET]

```
Public Property LicenseCode As String
```

[C#]

```
public string LicenseCode {get; set;}
```

Remarks:

The value is "TrialVersion" for the trial version's control, and please enter a 10-digit code - issued by us - into this property for the full version's control.

4.1.64 LinkColor

Gets or sets the color of the hyperlink text(s) for the grid in readonly mode.

[Visual Basic .NET]

```
Public Property LinkColor As Color
```

[C#]

```
public Color LinkColor {get; set;}
```

Remarks
The default value is yellow.

4.1.65 MarginBottomLeft

Gets or sets the bottom left margin, in percentage relative to the grid's width, of layout.

[Visual Basic .NET]

```
Public Property MarginBottomLeft As Float
```

[C#]

```
public float MarginBottomLeft {get; set;}
```

Remarks
A valid value must be between 0 and 50. Please refer to Margins Notes for more information.

4.1.66 MarginBottomRight

Gets or sets the bottom right margin, in percentage relative to the grid's width, of layout.

[Visual Basic .NET]

```
Public Property MarginBottomRight As Float
```

[C#]

```
public float MarginBottomRight {get; set;}
```

Remarks
A valid value must be between 0 and 50. Please refer to Margins Notes for more information

4.1.67 MarginTopLeft

Gets or sets the top left margin, in percentage relative to the grid's width, of layout.

```
[Visual Basic .NET]
Public Property MarginTopLeft As Float
```

```
[C#]
public float MarginTopLeft {get; set;}
```

Remarks
A valid value must be between 0 and 50. Please refer to Margins Notes for more information

4.1.68 MarginTopRight

Gets or sets the top right margin, in percentage relative to the grid's width, of layout.

```
[Visual Basic .NET]
Public Property MarginTopRight As Float
```

```
[C#]
public float MarginTopRight {get; set;}
```

Remarks
A valid value must be between 0 and 50. Please refer to Margins Notes for more information

4.1.69 MoveColumnBorderColor

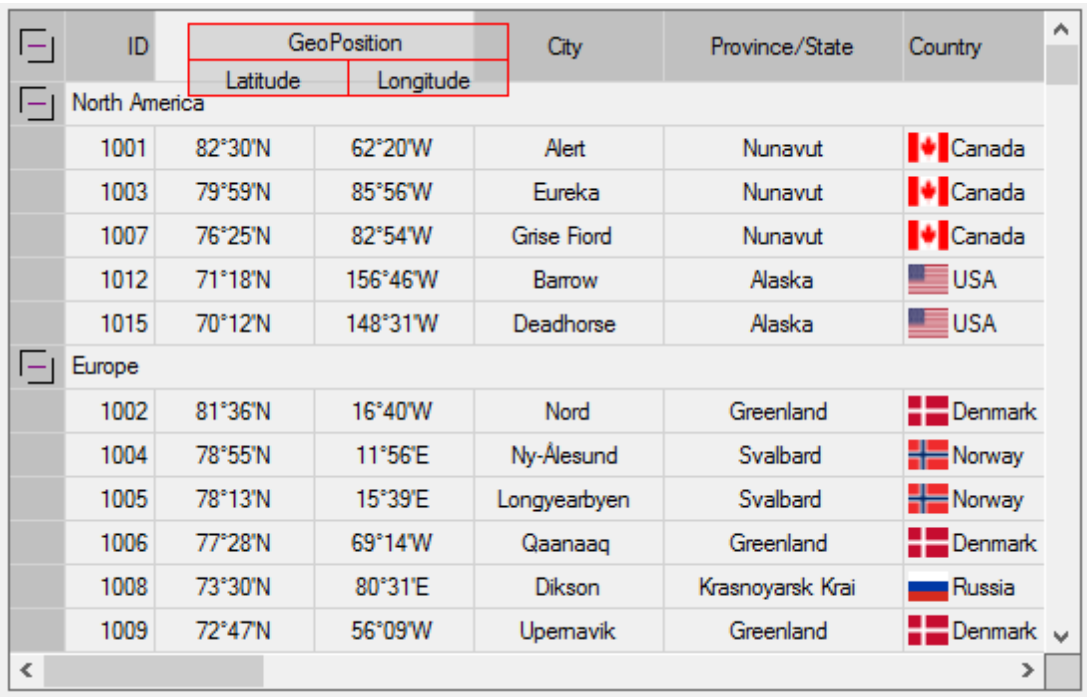
Gets or sets the border color of the moving column(s) for the grid.

```
[Visual Basic .NET]
Public Property MoveColumnBorderColor As Color
```

```
[C#]
public Color MoveColumnBorderColor {get; set;}
```

Remarks

The default value is red, and a screenshot is listed below.



ID	GeoPosition		City	Province/State	Country
	Latitude	Longitude			
North America					
1001	82°30'N	62°20'W	Alert	Nunavut	Canada
1003	79°59'N	85°56'W	Eureka	Nunavut	Canada
1007	76°25'N	82°54'W	Grise Fiord	Nunavut	Canada
1012	71°18'N	156°46'W	Barrow	Alaska	USA
1015	70°12'N	148°31'W	Deadhorse	Alaska	USA
Europe					
1002	81°36'N	16°40'W	Nord	Greenland	Denmark
1004	78°55'N	11°56'E	Ny-Ålesund	Svalbard	Norway
1005	78°13'N	15°39'E	Longyearbyen	Svalbard	Norway
1006	77°28'N	69°14'W	Qaanaaq	Greenland	Denmark
1008	73°30'N	80°31'E	Dikson	Krasnoyarsk Krai	Russia
1009	72°47'N	56°09'W	Upernavik	Greenland	Denmark

4.1.70 MoveColumnOpacity

Gets or sets the opacity level of the moving column(s).

[Visual Basic .NET]

```
Public Property MoveColumnOpacity As Double
```

[C#]

```
public double MoveColumnOpacity {get; set;}
```

Remarks

The default value is 0.5, and a valid value must be between 0 and 1.

- If it is 0, the moving column(s) will be fully transparent and invisible.
- If it is 1, the moving column(s) will be fully opaque.

4.1.71 MoveColumns

Gets or sets a boolean flag indicating whether it is allowed to move column(s) or not.

[Visual Basic .NET]

```
Public Property MoveColumns As Boolean
```

[C#]

```
public bool MoveColumns {get; set;}
```

Remarks

The default value is TRUE.

4.1.72 NodeColor

Gets or sets the color of the section nodes, either expanded or collapsed, for the grid in readonly mode.

```
[Visual Basic .NET]
Public Property NodeColor As Color
```

```
[C#]
public Color NodeColor {get; set;}
```

Remarks

The default value is purple.

4.1.73 PaddingBKColor

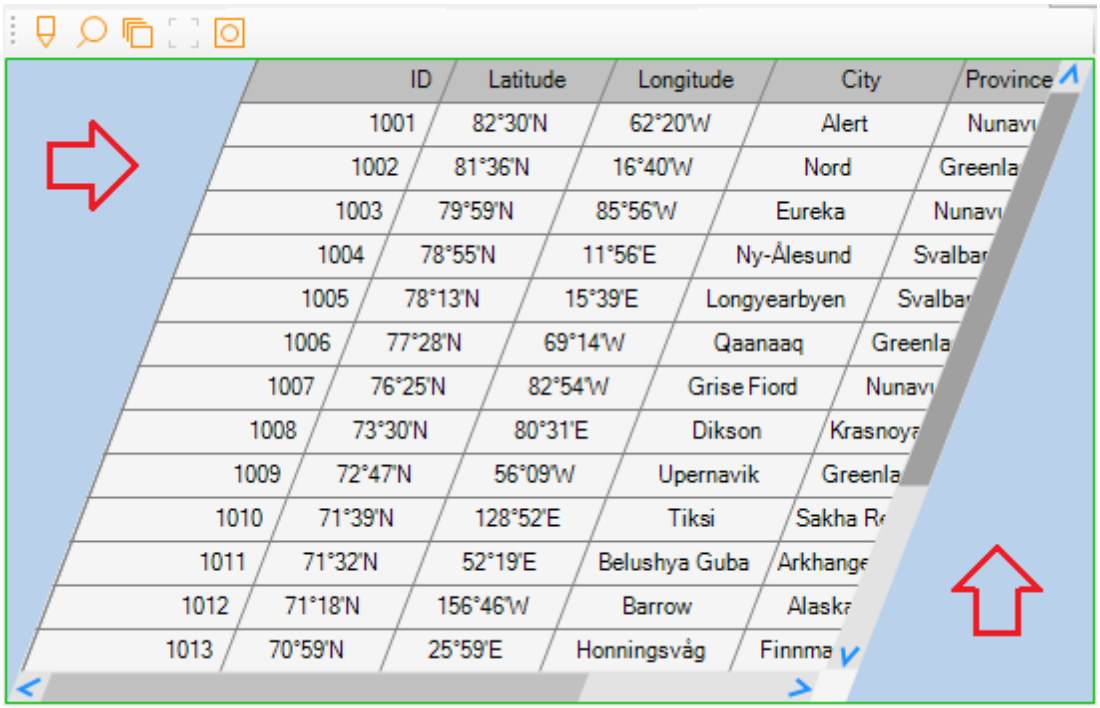
Gets or sets the padding color of non-rectangle layout.

```
[Visual Basic .NET]
Public Property PaddingBKColor As Color
```

```
[C#]
public Color PaddingBKColor {get; set;}
```

Remarks

The default value is SystemColors.GradientActiveCaption and a screehshot is listed below.



4.1.74 PopupMenu

Gets or sets the popup menu for the grid in readonly mode.

[Visual Basic .NET]

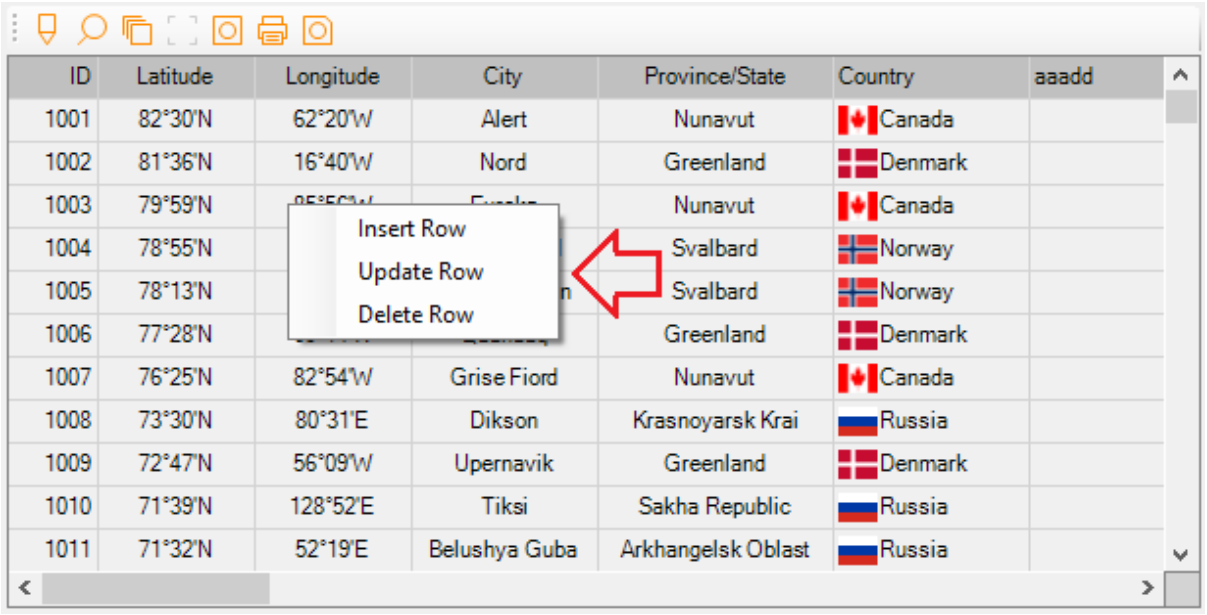
```
Public Property PopupMenu As ContextMenuStrip
```

[C#]

```
public ContextMenuStrip PopupMenu {get; set;}
```

Remarks

You can leverage this property to pop up a menu containing a few items via right-clicking a row. If the row(s) are sorted by clicking one column header, the menu is invisible.



ID	Latitude	Longitude	City	Province/State	Country	aaadd
1001	82°30'N	62°20'W	Alert	Nunavut	Canada	
1002	81°36'N	16°40'W	Nord	Greenland	Denmark	
1003	79°59'N			Nunavut	Canada	
1004	78°55'N			Svalbard	Norway	
1005	78°13'N			Svalbard	Norway	
1006	77°28'N			Greenland	Denmark	
1007	76°25'N	82°54'W	Grise Fiord	Nunavut	Canada	
1008	73°30'N	80°31'E	Dikson	Krasnoyarsk Krai	Russia	
1009	72°47'N	56°09'W	Upernavik	Greenland	Denmark	
1010	71°39'N	128°52'E	Tiksi	Sakha Republic	Russia	
1011	71°32'N	52°19'E	Belushya Guba	Arkhangelsk Oblast	Russia	

4.1.75 ReadOnly

Gets or sets a boolean flag indicating whether the grid is in read-only or edit mode.

[Visual Basic .NET]

```
Public Property ReadOnly As Boolean
```

[C#]

```
public bool ReadOnly {get; set;}
```

Remarks

The default value is TRUE.

4.1.76 ResizeColumns

Gets or sets a boolean flag indicating whether it is allowed to resize column(s) or not.

[Visual Basic .NET]

```
Public Property ResizeColumns As Boolean
```

[C#]

```
public bool ResizeColumns {get; set;}
```

Remarks

The default value is TRUE.

4.1.77 RowFont

Gets or sets the font characteristics of the rows.

[Visual Basic .NET]

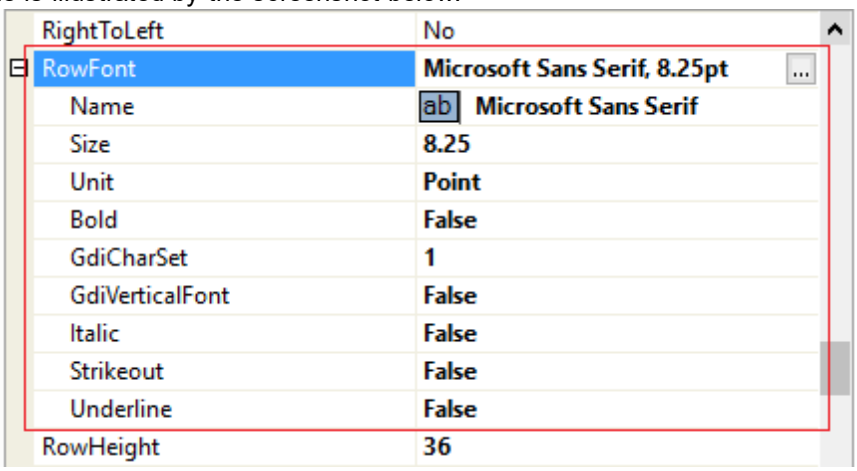
```
Public Property RowFont As Font
```

[C#]

```
public Font RowFont {get; set;}
```

Remarks

The default value is illustrated by the screenshot below.



4.1.78 RowHeight

Gets or sets the row height in pixels for the grid control.

[Visual Basic .NET]

```
Public Property RowHeight As Integer
```

[C#]

```
public int RowHeight {get; set;}
```

Remarks

The default value is 20.

4.1.79 RowsCount

Gets or sets the number of rows.

[Visual Basic .NET]

```
Public Property RowsCount As Integer
```

[C#]

```
public int RowsCount {get; set;}
```

Remarks

The default value is 10, and this property becomes irrelevant if the AllowSections property is set to TRUE.

4.1.80 RowTextColor

Gets or sets the text color of rows in the grid.

```
[Visual Basic .NET]
Public Property RowTextColor As Color
```

```
[C#]
public Color RowTextColor {get; set;}
```

Remarks

The default value is black.

4.1.81 ScrollBarArrowColor

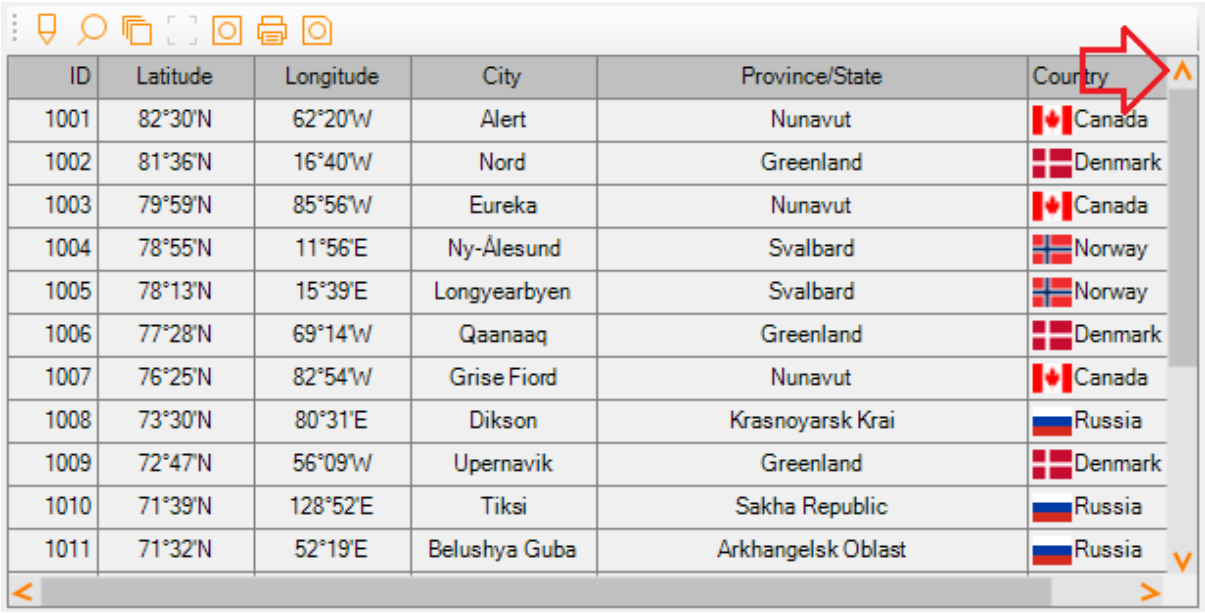
Gets or sets the arrow color of scrollbar(s) in the grid.

```
[Visual Basic .NET]
Public Property ScrollBarArrowColor As Color
```

```
[C#]
public Color ScrollBarArrowColor {get; set;}
```

Remarks

The default value is Gray, and a screenshot for orange color is listed.



ID	Latitude	Longitude	City	Province/State	Country
1001	82°30'N	62°20'W	Alert	Nunavut	Canada
1002	81°36'N	16°40'W	Nord	Greenland	Denmark
1003	79°59'N	85°56'W	Eureka	Nunavut	Canada
1004	78°55'N	11°56'E	Ny-Ålesund	Svalbard	Norway
1005	78°13'N	15°39'E	Longyearbyen	Svalbard	Norway
1006	77°28'N	69°14'W	Qaanaaq	Greenland	Denmark
1007	76°25'N	82°54'W	Grise Fiord	Nunavut	Canada
1008	73°30'N	80°31'E	Dikson	Krasnoyarsk Krai	Russia
1009	72°47'N	56°09'W	Upernavik	Greenland	Denmark
1010	71°39'N	128°52'E	Tiksi	Sakha Republic	Russia
1011	71°32'N	52°19'E	Belushya Guba	Arkhangelsk Oblast	Russia

4.1.82 ScrollBarBigArrow

Gets or sets a boolean flag indicating whether the arrows of scrollbar(s) should be big or not.

[Visual Basic .NET]

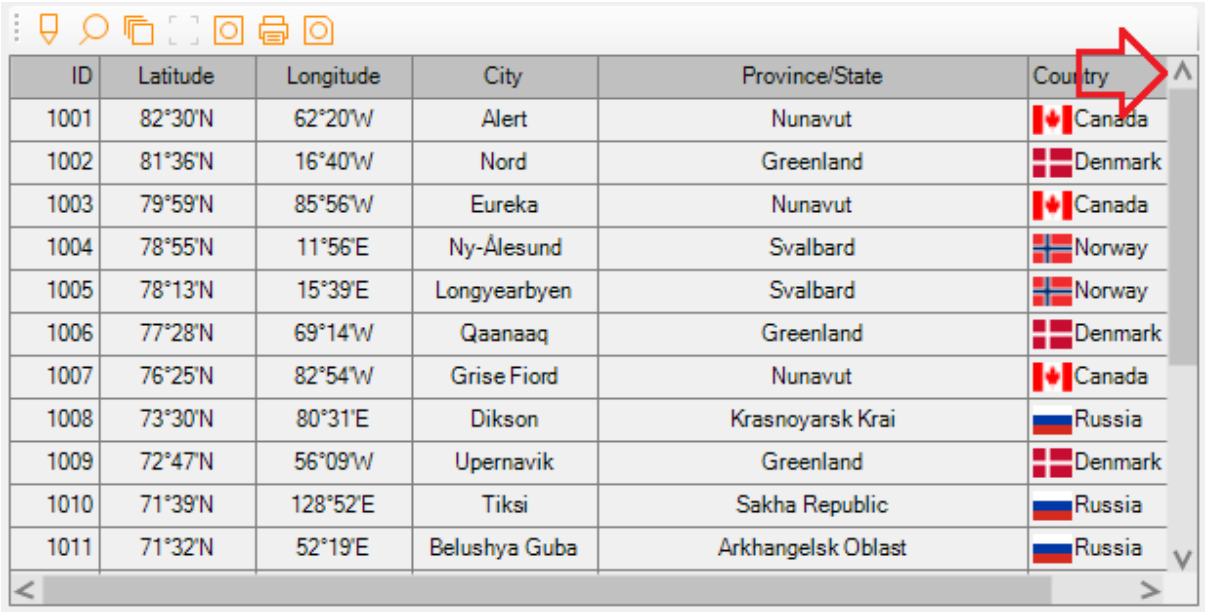
```
Public Property ScrollBarBigArrow As Boolean
```

[C#]

```
public bool ScrollBarBigArrow {get; set;}
```

Remarks

The default value is FALSE, and a screenshot for TRUE is listed below.



ID	Latitude	Longitude	City	Province/State	Country
1001	82°30'N	62°20'W	Alert	Nunavut	Canada
1002	81°36'N	16°40'W	Nord	Greenland	Denmark
1003	79°59'N	85°56'W	Eureka	Nunavut	Canada
1004	78°55'N	11°56'E	Ny-Ålesund	Svalbard	Norway
1005	78°13'N	15°39'E	Longyearbyen	Svalbard	Norway
1006	77°28'N	69°14'W	Qaanaaq	Greenland	Denmark
1007	76°25'N	82°54'W	Grise Fiord	Nunavut	Canada
1008	73°30'N	80°31'E	Dikson	Krasnoyarsk Krai	Russia
1009	72°47'N	56°09'W	Upernavik	Greenland	Denmark
1010	71°39'N	128°52'E	Tiksi	Sakha Republic	Russia
1011	71°32'N	52°19'E	Belushya Guba	Arkhangelsk Oblast	Russia

.

4.1.83 SearchHighlightBackColor

Gets or sets the background color of matched text(s) in search mode.

[Visual Basic .NET]

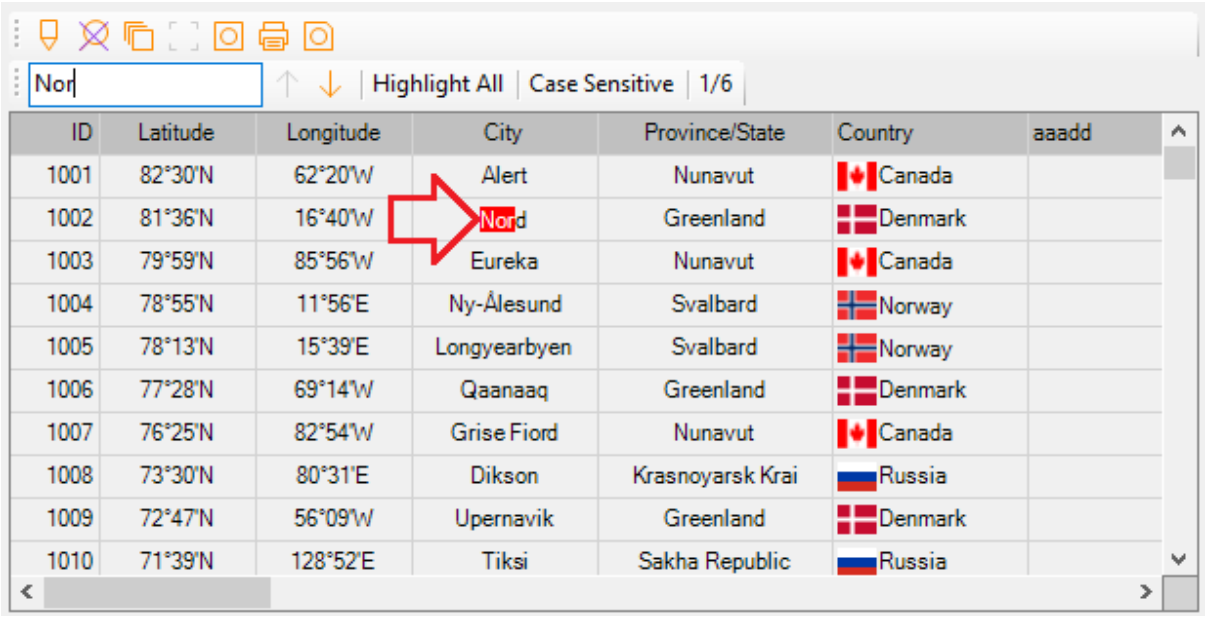
```
Public Property SearchHighlightBackColor As Color
```

[C#]

```
public Color SearchHighlightBackColor {get; set;}
```

Remarks

The default value is red, and a screenshot is listed below.



4.1.84 SearchHighlightTextColor

Gets or sets the text color of matched text(s) in search mode.

[Visual Basic .NET]

```
Public Property SearchHighlightTextColor As Color
```

[C#]

```
public Color SearchHighlightTextColor {get; set;}
```

Remarks

The default value is white.

4.1.85 SearchLabelToggleColor

Gets or sets the text color of 2 labels, Highlight All and Case Sensitive, in search mode when the option indicated by them is toggled on.

[Visual Basic .NET]

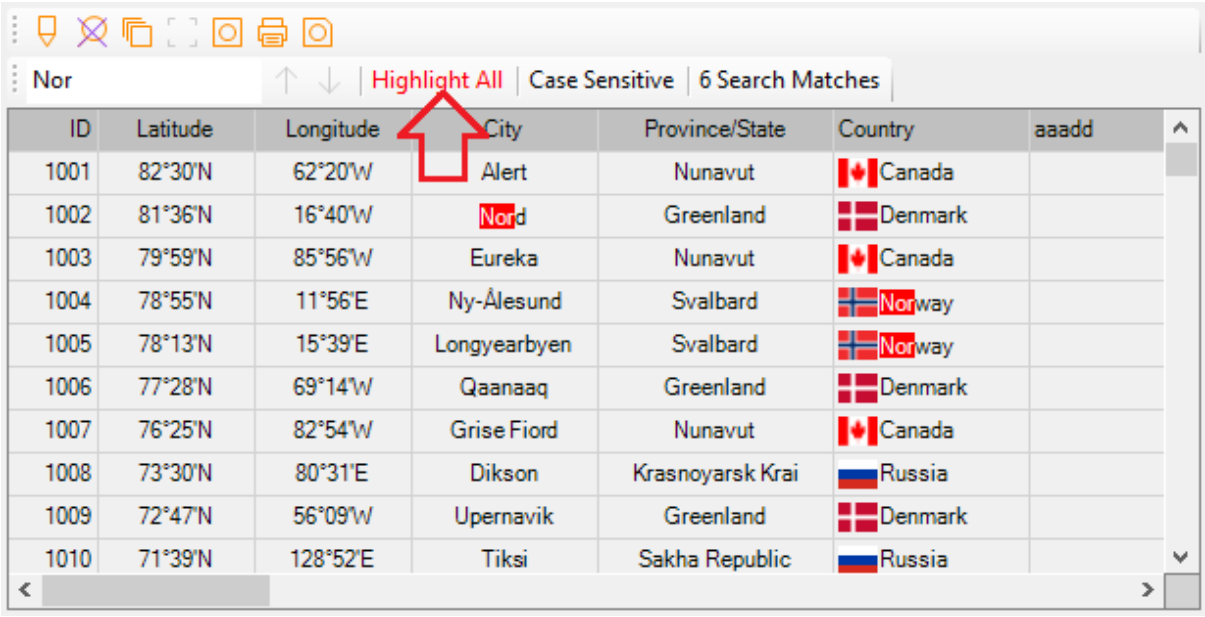
```
Public Property SearchLabelToggleColor As Color
```

[C#]

```
public Color SearchLabelToggleColor {get; set;}
```

Remarks

The default value is red.



4.1.86 SectionIconHeight

Gets or sets the height in pixels of the rectangle area where to render the section icon(s) for the grid.

[Visual Basic .NET]

```
Public Property SectionIconHeight As Integer
```

[C#]

```
public int SectionIconHeight {get; set;}
```

Remarks

The default value is 16.

4.1.87 SectionIconIsLeft

Gets or sets a boolean flag indicating whether the icon is rendered to the left of the section title or not for the grid.

[Visual Basic .NET]

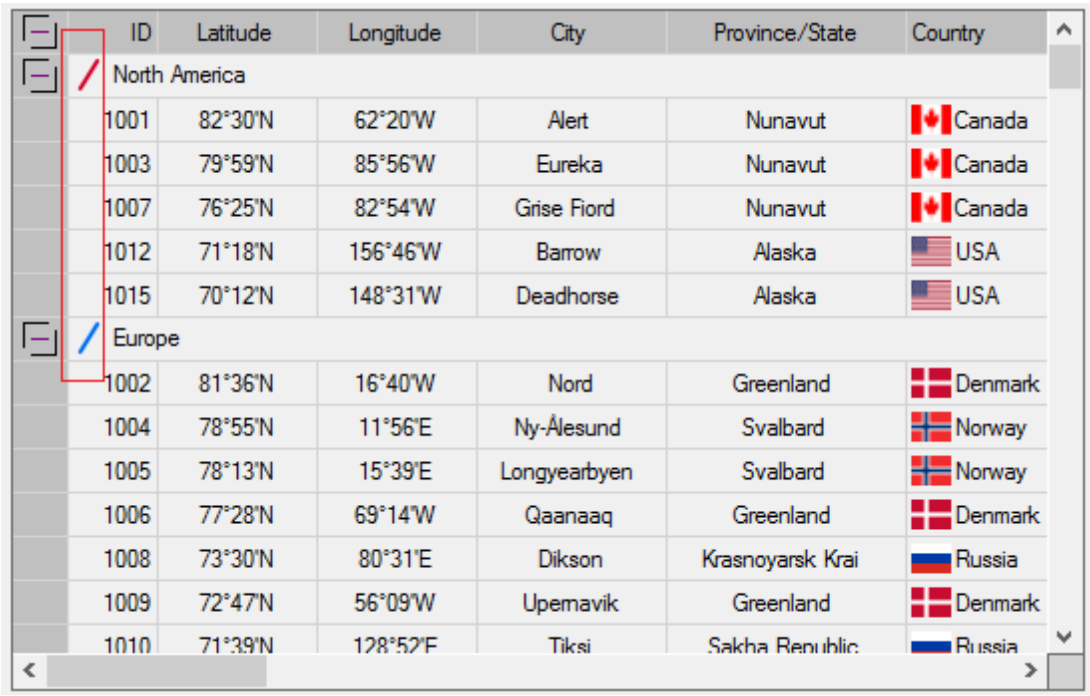
```
Public Property SectionIconIsLeft As Boolean
```

[C#]

```
public bool SectionIconIsLeft {get; set;}
```

Remarks

The default value is TRUE, and a screenshot is listed below.



ID	Latitude	Longitude	City	Province/State	Country
North America					
1001	82°30'N	62°20'W	Alert	Nunavut	Canada
1003	79°59'N	85°56'W	Eureka	Nunavut	Canada
1007	76°25'N	82°54'W	Grise Fiord	Nunavut	Canada
1012	71°18'N	156°46'W	Barrow	Alaska	USA
1015	70°12'N	148°31'W	Deadhorse	Alaska	USA
Europe					
1002	81°36'N	16°40'W	Nord	Greenland	Denmark
1004	78°55'N	11°56'E	Ny-Ålesund	Svalbard	Norway
1005	78°13'N	15°39'E	Longyearbyen	Svalbard	Norway
1006	77°28'N	69°14'W	Qaanaaq	Greenland	Denmark
1008	73°30'N	80°31'E	Dikson	Krasnoyarsk Krai	Russia
1009	72°47'N	56°09'W	Upernavik	Greenland	Denmark
1010	71°39'N	128°52'E	Tiksi	Sakha Republic	Russia

4.1.88 SectionIconWidth

Gets or sets the width in pixels of the rectangle area where to render the section icon(s) for the grid.

```
[Visual Basic .NET]
Public Property SectionIconWidth As Integer
```

```
[C#]
public int SectionIconWidth {get; set;}
```

Remarks

The default value is 16.

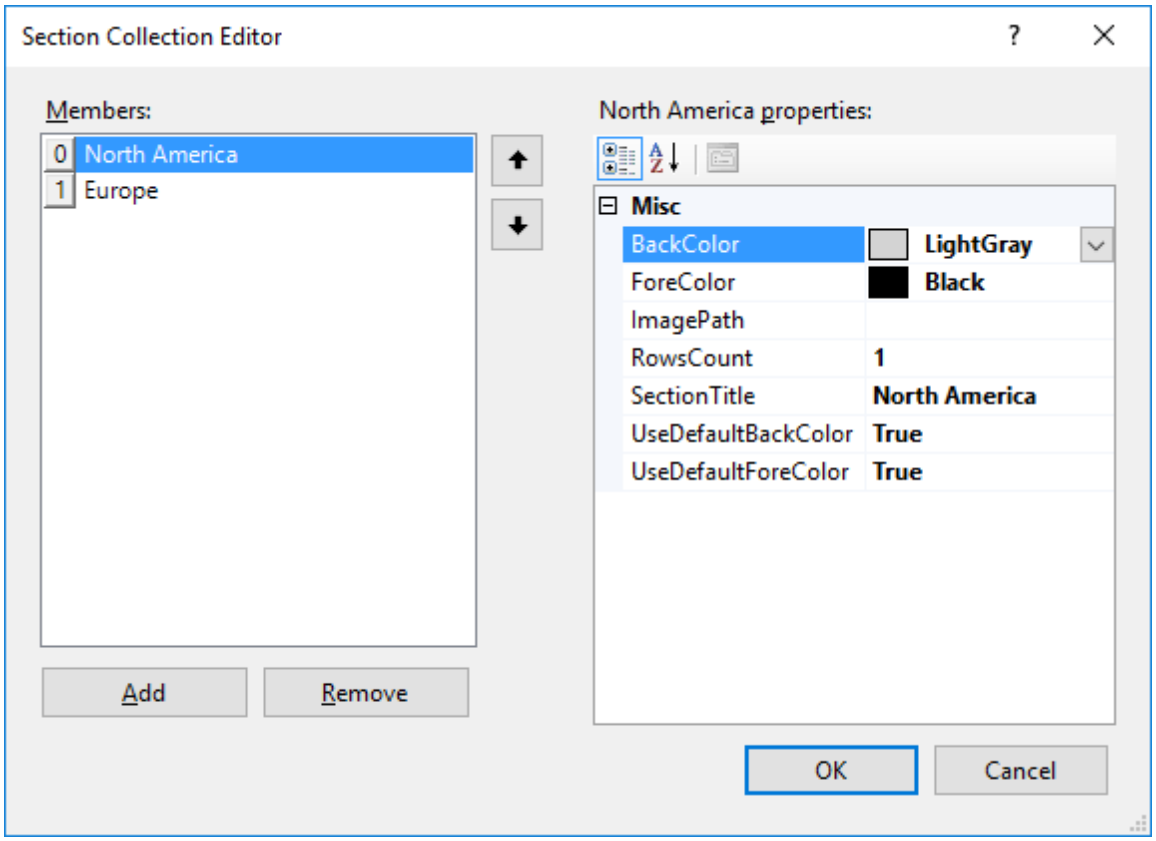
4.1.89 Sections

Gets or sets the settings of the sections for the grid.

```
[Visual Basic .NET]
Public Property Sections As System.Collections.CollectionBase
```

```
[C#]
public System.Collections.CollectionBase Sections {get; set;}
```

Remarks



From the screenshot above, you can see that each section has the following fields:

1. **BackColor** specifies the background color of the section title, and it becomes irrelevant if **UseDefaultBackColor** is TRUE.
2. **ForeColor** specifies the text color of the section title, and it becomes irrelevant if **UseDefaultForeColor** is TRUE.
3. **ImagePath** specifies the full path of the section icon image.
4. **RowsCount** specifies the number of the child rows belonging to this section.
5. **SectionTitle** specifies the section title.
6. **UseDefaultBackColor** indicates whether the background color of the section title is dictated by the GridBackColor property or not.
7. **UseDefaultForeColor** indicates whether the text color of the section title is dictated by the RowTextColor property or not.

4.1.90 SectionTitleFont

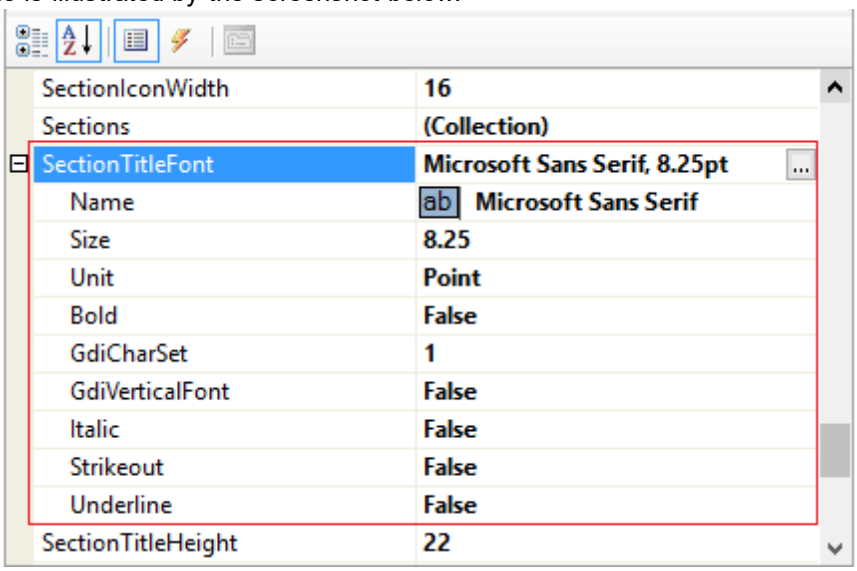
Gets or sets the font characteristics of the section title(s) for the grid.

```
[Visual Basic .NET]
Public Property SectionTitleFont As Font
```

```
[C#]
public Font SectionTitleFont {get; set;}
```

Remarks

The default value is illustrated by the screenshot below.



4.1.91 SectionTitleHeight

Gets or sets the height in pixels of the section title's row for the grid.

[Visual Basic .NET]

```
Public Property SectionTitleHeight As Integer
```

[C#]

```
public int SectionTitleHeight {get; set;}
```

Remarks

The default value is 20.

4.1.92 SelectCellsInSameColumn

Gets or sets a boolean flag indicating whether the multiple cell(s) via CTRL key and mouse clicking should be selected in single column or not for the grid in readonly mode.

[Visual Basic .NET]

```
Public Property SelectCellsInSameColumn As Boolean
```

[C#]

```
public bool SelectCellsInSameColumn {get; set;}
```

Remarks

The default value is TRUE.

4.1.93 SelectMultiCells

Gets or sets a boolean flag indicating whether the multiple cell(s) are allowed to be selected or not for the grid in readonly mode.

[Visual Basic .NET]

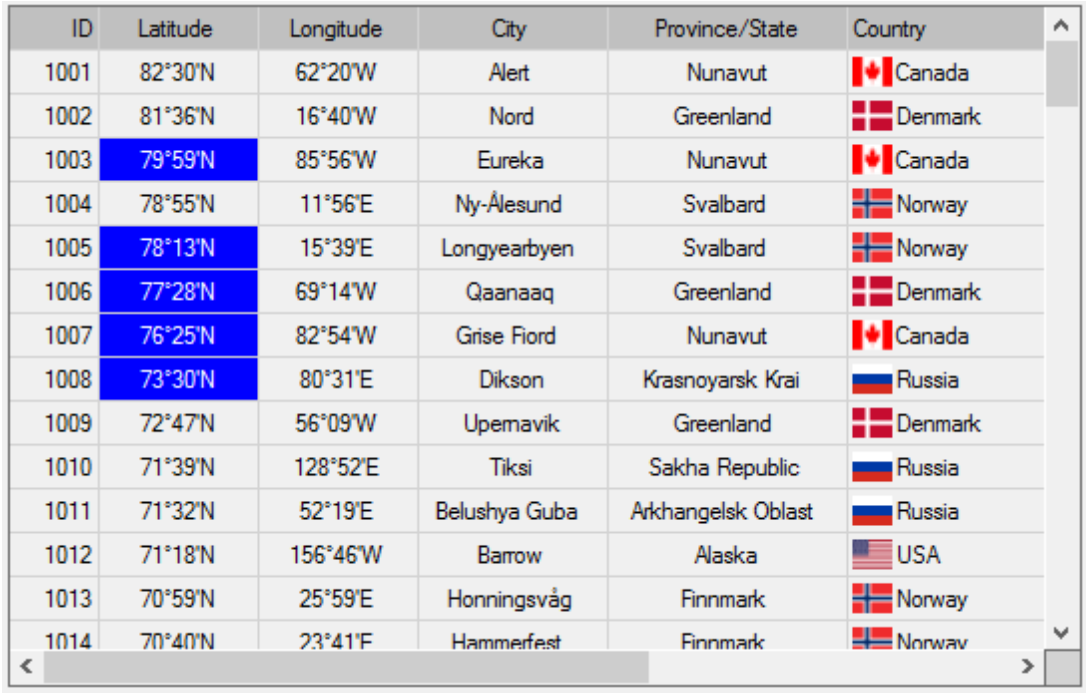
```
Public Property SelectMultiCells As Boolean
```

[C#]

```
public bool SelectMultiCells {get; set;}
```

Remarks

The default value is FALSE, and a screenshot for TRUE value is listed below.



ID	Latitude	Longitude	City	Province/State	Country
1001	82°30'N	62°20'W	Alert	Nunavut	Canada
1002	81°36'N	16°40'W	Nord	Greenland	Denmark
1003	79°59'N	85°56'W	Eureka	Nunavut	Canada
1004	78°55'N	11°56'E	Ny-Ålesund	Svalbard	Norway
1005	78°13'N	15°39'E	Longyearbyen	Svalbard	Norway
1006	77°28'N	69°14'W	Qaanaaq	Greenland	Denmark
1007	76°25'N	82°54'W	Grise Fiord	Nunavut	Canada
1008	73°30'N	80°31'E	Dikson	Krasnoyarsk Krai	Russia
1009	72°47'N	56°09'W	Upernavik	Greenland	Denmark
1010	71°39'N	128°52'E	Tiksi	Sakha Republic	Russia
1011	71°32'N	52°19'E	Belushya Guba	Arkhangelsk Oblast	Russia
1012	71°18'N	156°46'W	Barrow	Alaska	USA
1013	70°59'N	25°59'E	Honningsvåg	Finnmark	Norway
1014	70°40'N	23°41'E	Hammerfest	Finnmark	Norway

4.1.94 SelectRow

Gets or sets a boolean flag indicating whether the entire row should be selected or not once one cell of it is moused-clicked for the grid in readonly mode.

[Visual Basic .NET]

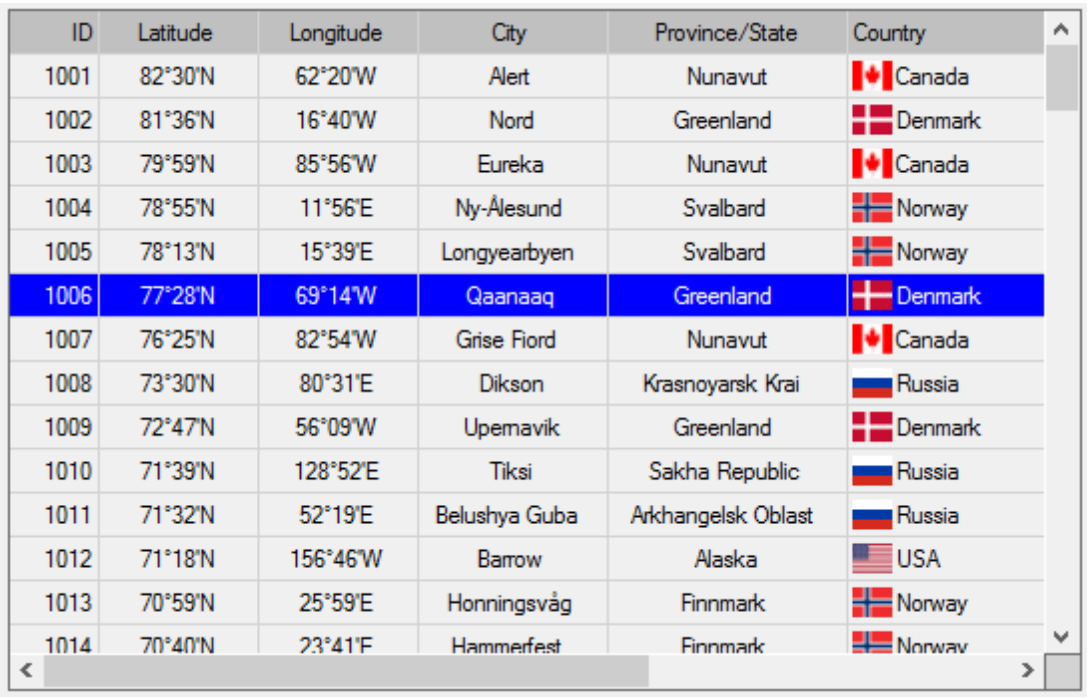
```
Public Property SelectRow As Boolean
```

[C#]

```
public bool SelectRow {get; set;}
```

Remarks

The default value is FALSE, and a screenshot for TRUE value is listed below.



ID	Latitude	Longitude	City	Province/State	Country
1001	82°30'N	62°20'W	Alert	Nunavut	Canada
1002	81°36'N	16°40'W	Nord	Greenland	Denmark
1003	79°59'N	85°56'W	Eureka	Nunavut	Canada
1004	78°55'N	11°56'E	Ny-Ålesund	Svalbard	Norway
1005	78°13'N	15°39'E	Longyearbyen	Svalbard	Norway
1006	77°28'N	69°14'W	Qaanaaq	Greenland	Denmark
1007	76°25'N	82°54'W	Grise Fiord	Nunavut	Canada
1008	73°30'N	80°31'E	Dikson	Krasnoyarsk Krai	Russia
1009	72°47'N	56°09'W	Upernavik	Greenland	Denmark
1010	71°39'N	128°52'E	Tiksi	Sakha Republic	Russia
1011	71°32'N	52°19'E	Belushya Guba	Arkhangelsk Oblast	Russia
1012	71°18'N	156°46'W	Barrow	Alaska	USA
1013	70°59'N	25°59'E	Honningsvåg	Finnmark	Norway
1014	70°40'N	23°41'E	Hammerfest	Finnmark	Norway

4.1.95 ShowArrow

Gets or sets a boolean flag indicating whether it is allowed to show an arrow indicating scroll bar's orientation.

[Visual Basic .NET]

```
Public Property ShowArrow As Boolean
```

[C#]

```
public bool ShowArrow {get; set;}
```

Remarks

The default value is TRUE.

4.1.96 ShowPrintIcon

Gets or sets a boolean flag indicating whether Print-Icon is visible or not.

[Visual Basic .NET]

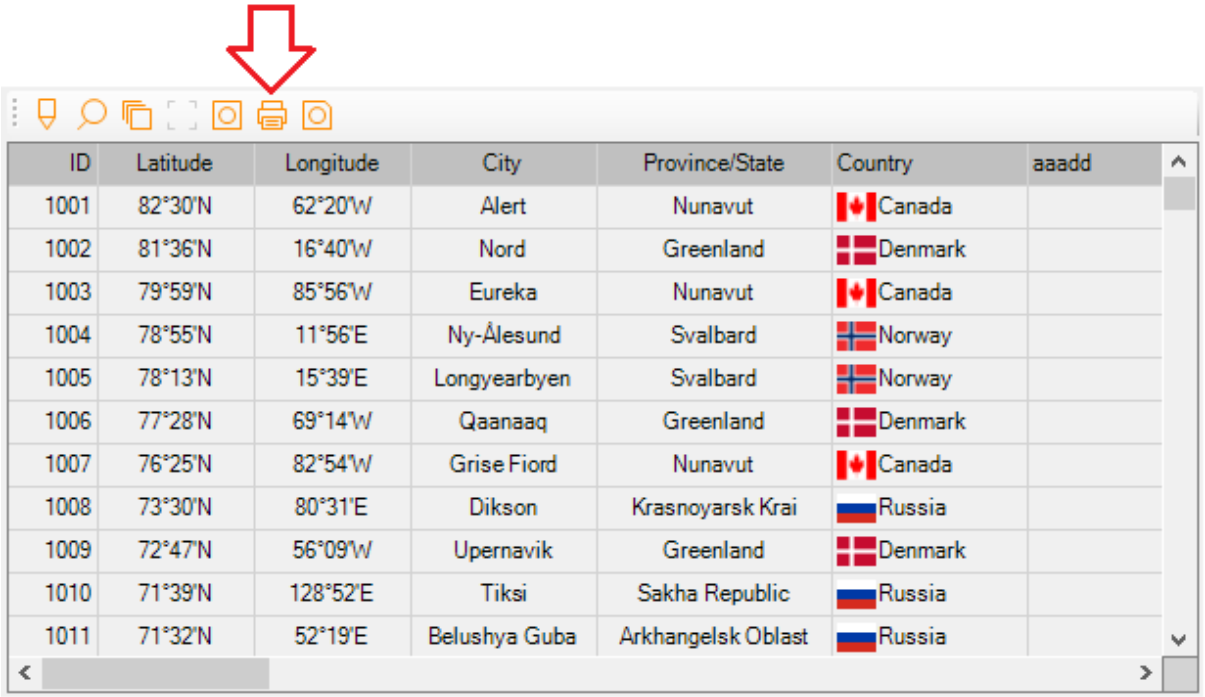
```
Public Property ShowPrintIcon As Boolean
```

[C#]

```
public bool ShowPrintIcon {get; set;}
```

Remarks

The default value is TRUE.



ID	Latitude	Longitude	City	Province/State	Country	aaadd
1001	82°30'N	62°20'W	Alert	Nunavut	Canada	
1002	81°36'N	16°40'W	Nord	Greenland	Denmark	
1003	79°59'N	85°56'W	Eureka	Nunavut	Canada	
1004	78°55'N	11°56'E	Ny-Ålesund	Svalbard	Norway	
1005	78°13'N	15°39'E	Longyearbyen	Svalbard	Norway	
1006	77°28'N	69°14'W	Qaanaaq	Greenland	Denmark	
1007	76°25'N	82°54'W	Grise Fiord	Nunavut	Canada	
1008	73°30'N	80°31'E	Dikson	Krasnoyarsk Krai	Russia	
1009	72°47'N	56°09'W	Upernavik	Greenland	Denmark	
1010	71°39'N	128°52'E	Tiksi	Sakha Republic	Russia	
1011	71°32'N	52°19'E	Belushya Guba	Arkhangelsk Oblast	Russia	

4.1.97 ShowSaveScreenshotIcon

Gets or sets a boolean flag indicating whether Save-Screenshot-Icon is visible or not.

[Visual Basic .NET]

```
Public Property ShowSaveScreenshotIcon As Boolean
```

[C#]

```
public bool ShowSaveScreenshotIcon {get; set;}
```

Remarks

The default value is TRUE.

ID	Latitude	Longitude	City	Province/State	Country	aaadd
1001	82°30'N	62°20'W	Alert	Nunavut	Canada	
1002	81°36'N	16°40'W	Nord	Greenland	Denmark	
1003	79°59'N	85°56'W	Eureka	Nunavut	Canada	
1004	78°55'N	11°56'E	Ny-Ålesund	Svalbard	Norway	
1005	78°13'N	15°39'E	Longyearbyen	Svalbard	Norway	
1006	77°28'N	69°14'W	Qaanaaq	Greenland	Denmark	
1007	76°25'N	82°54'W	Grise Fiord	Nunavut	Canada	
1008	73°30'N	80°31'E	Dikson	Krasnoyarsk Krai	Russia	
1009	72°47'N	56°09'W	Upernavik	Greenland	Denmark	
1010	71°39'N	128°52'E	Tiksi	Sakha Republic	Russia	
1011	71°32'N	52°19'E	Belushya Guba	Arkhangelsk Oblast	Russia	

4.1.98 SortingIconColor

Gets or sets the color of the sorting icon in a triangle shape when the sorting process is applied against one column.

[Visual Basic .NET]

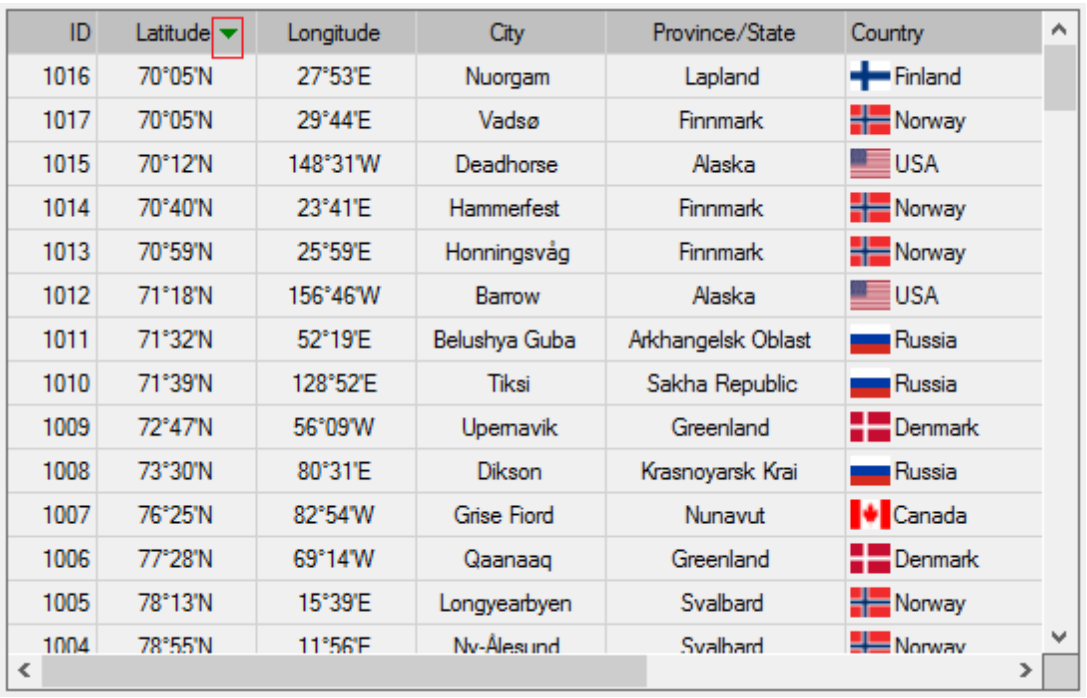
```
Public Property SortingIconColor As Color
```

[C#]

```
public Color SortingIconColor {get; set;}
```

Remarks

The default value is green, and a screenshot is listed below.



ID	Latitude	Longitude	City	Province/State	Country
1016	70°05'N	27°53'E	Nuorgam	Lapland	Finland
1017	70°05'N	29°44'E	Vadsø	Finnmark	Norway
1015	70°12'N	148°31'W	Deadhorse	Alaska	USA
1014	70°40'N	23°41'E	Hammerfest	Finnmark	Norway
1013	70°59'N	25°59'E	Honningsvåg	Finnmark	Norway
1012	71°18'N	156°46'W	Barrow	Alaska	USA
1011	71°32'N	52°19'E	Belushya Guba	Arkhangelsk Oblast	Russia
1010	71°39'N	128°52'E	Tiksi	Sakha Republic	Russia
1009	72°47'N	56°09'W	Upernavik	Greenland	Denmark
1008	73°30'N	80°31'E	Dikson	Krasnoyarsk Krai	Russia
1007	76°25'N	82°54'W	Grise Fiord	Nunavut	Canada
1006	77°28'N	69°14'W	Qaanaaq	Greenland	Denmark
1005	78°13'N	15°39'E	Longyearbyen	Svalbard	Norway
1004	78°55'N	11°56'E	Ny-Ålesund	Svalbard	Norway

4.1.99 SubGridHeaderBackColor

Gets or sets the background color of sub grid's header.

[Visual Basic .NET]

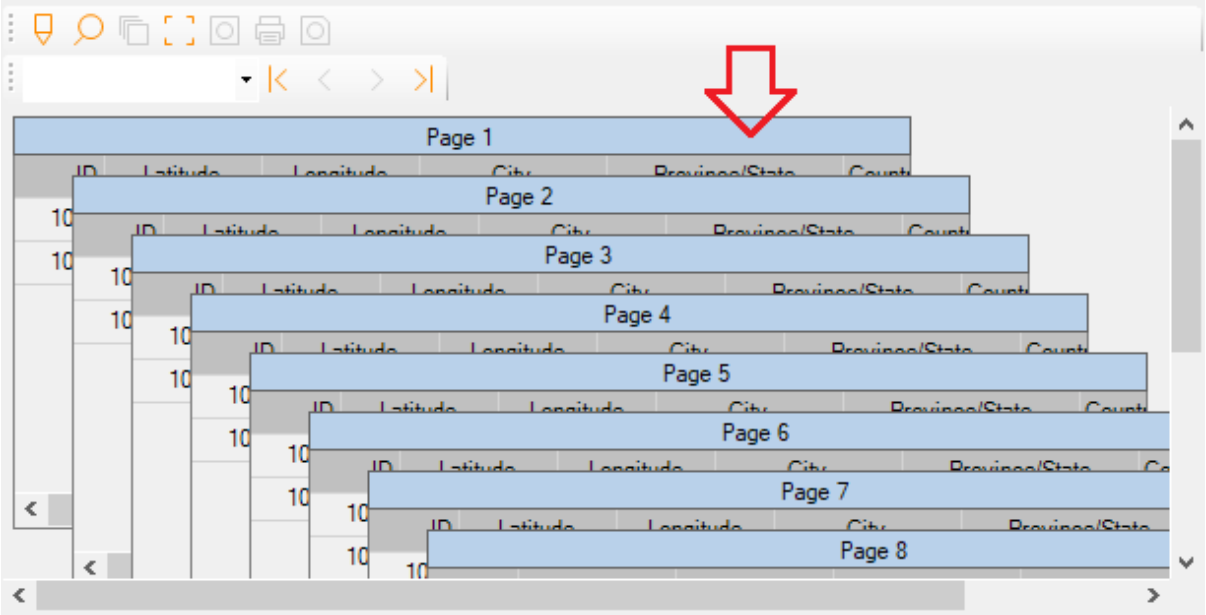
```
Public Property SubGridHeaderBackColor As Color
```

[C#]

```
public Color SubGridHeaderBackColor {get; set;}
```

Remarks

The default value is orange red and a screenshot for light blue is listed below.



4.1.100 SubGridHeaderFont

Gets or sets the font characteristics of sub grid's header.

[Visual Basic .NET]

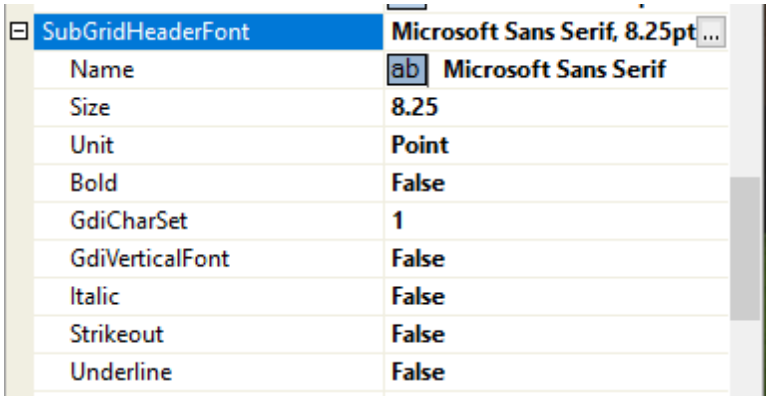
```
Public Property SubGridHeaderFont As Font
```

[C#]

```
public Font SubGridHeaderFont {get; set;}
```

Remarks

The default value is illustrated by the screenshot below.



4.1.101 SubGridHeaderHeight

Gets or sets the height in pixels for sub grid's header.

[Visual Basic .NET]

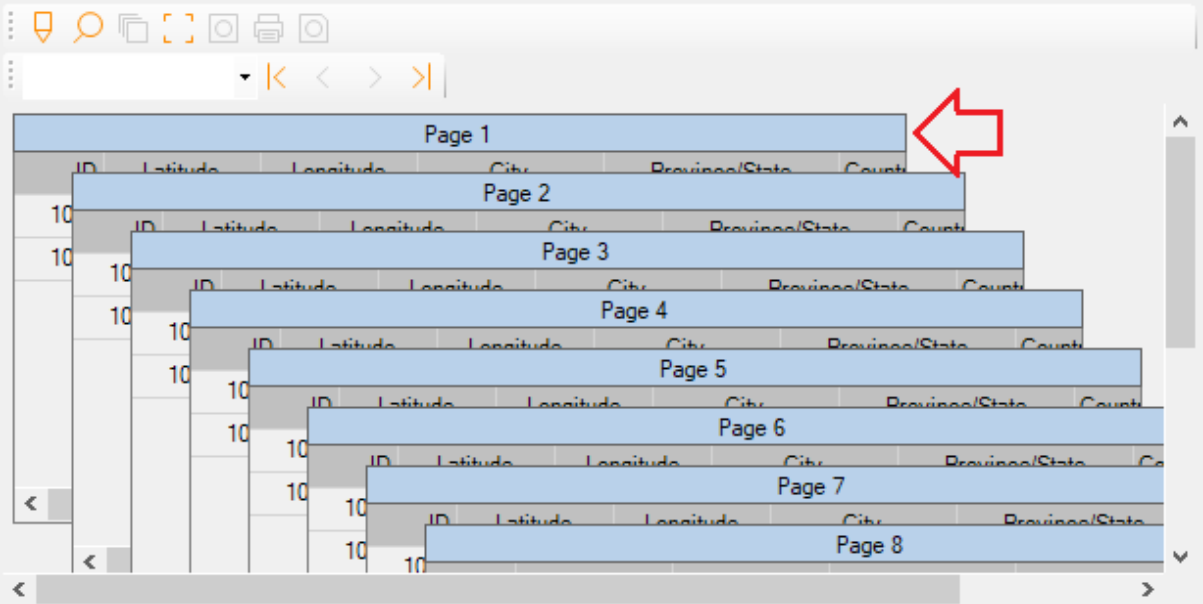
Public Property SubGridHeaderHeight As Integer

[C#]

public int SubGridHeaderHeight {get; set;}

Remarks

The default value is 20.



4.1.102 SubGridHeaderTextColor

Gets or sets the text color of sub grid's header.

[Visual Basic .NET]

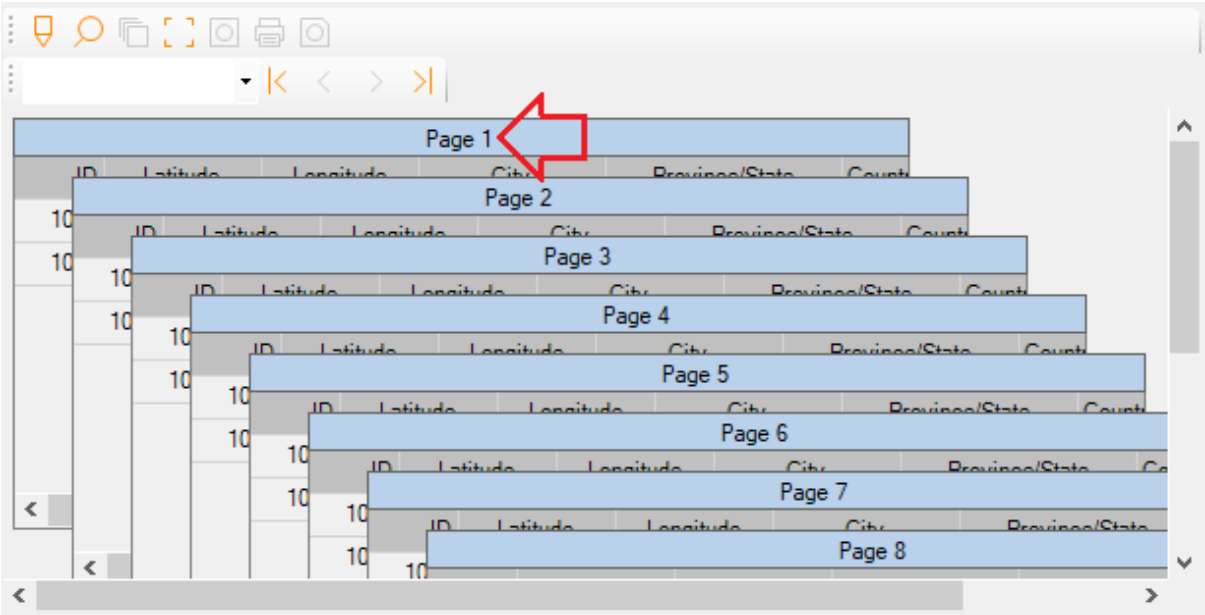
Public Property SubGridHeaderTextColor As Color

[C#]

public Color SubGridHeaderTextColor {get; set;}

Remarks

The default value is black, and a screenshot is listed below.



4.1.103 SubGridSize

Gets or sets the number of rows in each sub grid.

[Visual Basic .NET]

```
Public Property SubGridSize As Integer
```

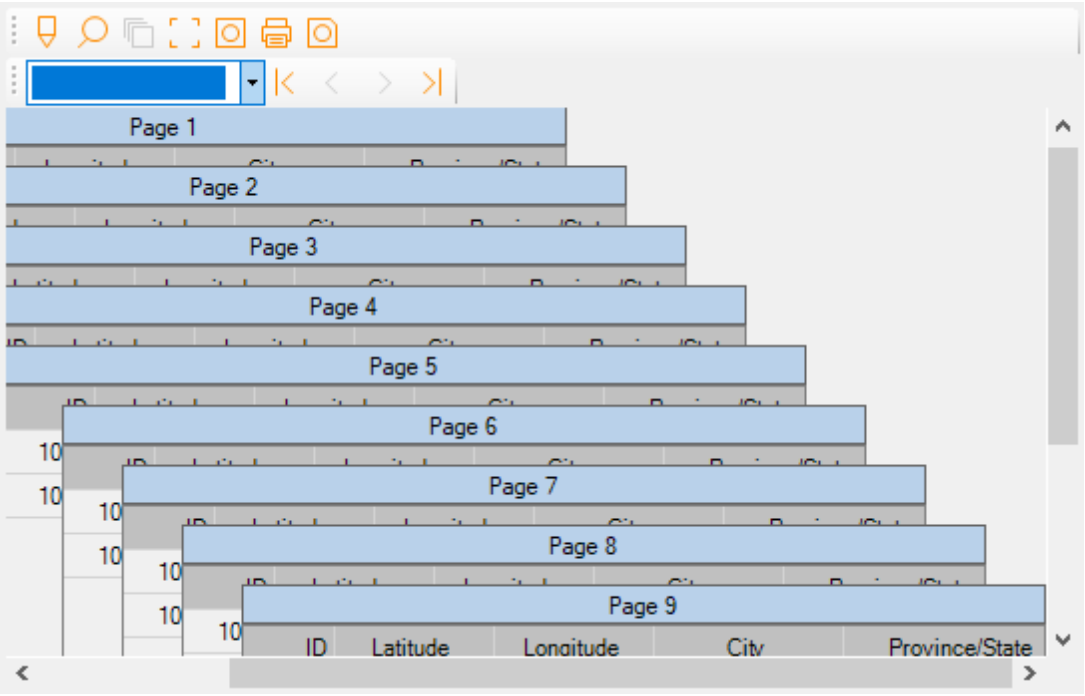
[C#]

```
public int SubGridSize {get; set;}
```

Remarks

If the property AllowSections is TRUE or the property SubGridSizesPattern is valid, this property becomes irrelevant. Otherwise the property RowsCount and this property determine how many sub grids will be generated and how many rows are in each sub grid. If RowsCount is not divisible by SubGridSize, the number of rows in last sub grid will be less than this property.

For example, if RowsCount is 17 and SubGridSize is 2, there are 9 sub grids that are generated, and the number of rows in the last sub grid is 1 while all other sub grids contain 2 rows.



4.1.104 SubGridSizesPattern

Gets or sets the initial sizes of all sub grids.

[Visual Basic .NET]

```
Public Property SubGridSizesPattern As String
```

[C#]

```
public string SubGridSizesPattern {get; set;}
```

Remarks:

It contains comma-separated values for initial sizes of all sub grids. For example, if the property RowsCount is 17, you want to create 4 sub grids with the sizes of 3, 5 2, and 7, so you can pass "3, 5, 2, 7" to this property since the sum up of them is equal to 17.

If the sum up of values in this property is not equal to the RowsCount property, this property is discarded and the property SubGridSize will kick in to determine how to generate all sub grids.

If the property AllowSections is TRUE, this property becomes irrelevant.

4.1.105 UnderlinedLink

Gets or sets a boolean flag indicating whether the hyper link text(s) should be underlined or not for the grid in readonly mode.

[Visual Basic .NET]

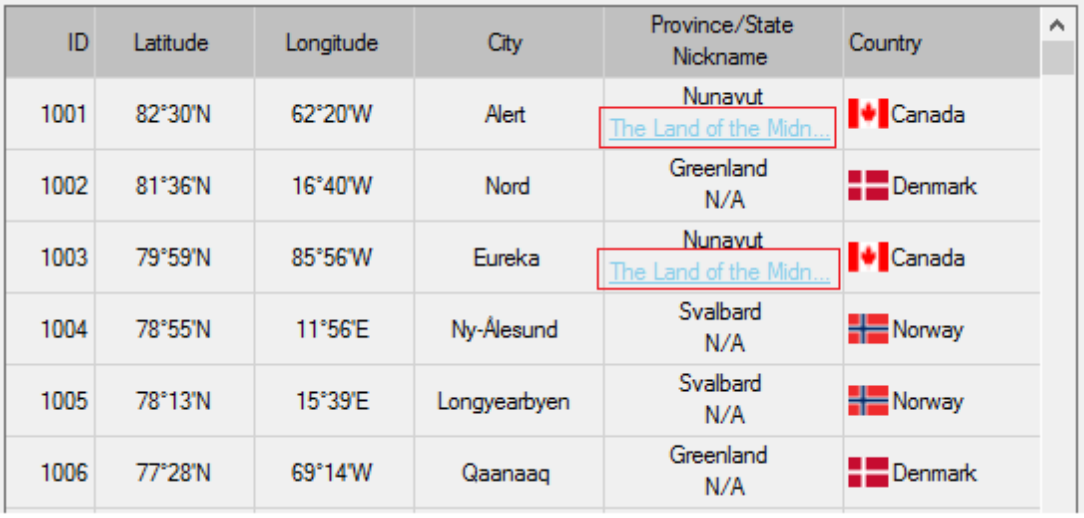
```
Public Property UnderlinedLink As Boolean
```

[C#]

```
public bool UnderlinedLink {get; set;}
```

Remarks

The default value is TRUE, and a screenshot is listed below.



ID	Latitude	Longitude	City	Province/State Nickname	Country
1001	82°30'N	62°20'W	Alert	Nunavut The Land of the Midn...	Canada
1002	81°36'N	16°40'W	Nord	Greenland N/A	Denmark
1003	79°59'N	85°56'W	Eureka	Nunavut The Land of the Midn...	Canada
1004	78°55'N	11°56'E	Ny-Ålesund	Svalbard N/A	Norway
1005	78°13'N	15°39'E	Longyearbyen	Svalbard N/A	Norway
1006	77°28'N	69°14'W	Qaanaaq	Greenland N/A	Denmark

4.1.106 ZebraColor

Gets or sets the color of the zebras being used to shade alternate rows for the grid in readonly mode.

[Visual Basic .NET]

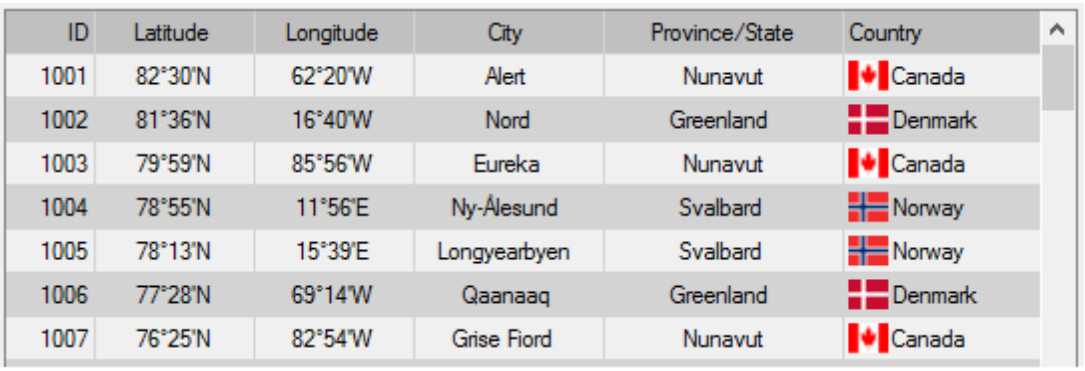
```
Public Property ZebraColor As Color
```

[C#]

```
public Color ZebraColor {get; set;}
```

Remarks

The default value is light gray, and a screenshot is listed below.



ID	Latitude	Longitude	City	Province/State	Country
1001	82°30'N	62°20'W	Alert	Nunavut	Canada
1002	81°36'N	16°40'W	Nord	Greenland	Denmark
1003	79°59'N	85°56'W	Eureka	Nunavut	Canada
1004	78°55'N	11°56'E	Ny-Ålesund	Svalbard	Norway
1005	78°13'N	15°39'E	Longyearbyen	Svalbard	Norway
1006	77°28'N	69°14'W	Qaanaaq	Greenland	Denmark
1007	76°25'N	82°54'W	Grise Fiord	Nunavut	Canada

4.2 Methods

4.2.1 ActivateIcon

Programmingly clicks an icon in the first tool strip in the control.

[Visual Basic .NET]

```
Public Function ActivateIcon(ByVal IconFlag As Integer) As Boolean
```

[C#]

```
public bool ActivateIcon(int IconFlag);
```

Parameters

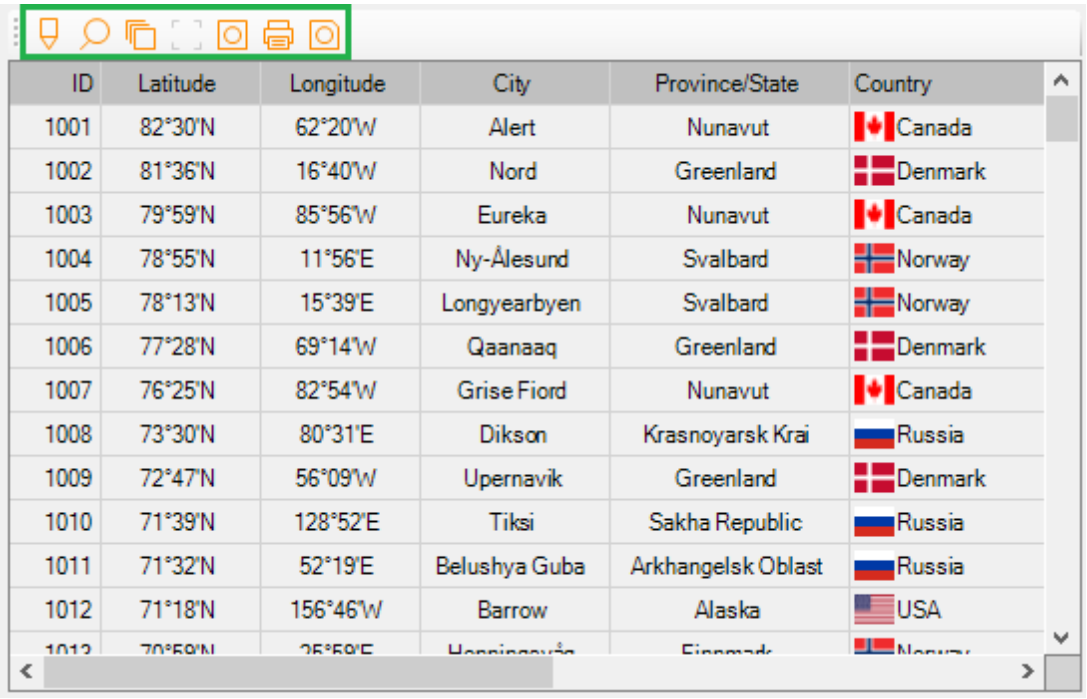
IconFlag

Specifies which icon gets clicked, and all possible values are listed below.

Value	Comment
1	Open annotation
2	Close annotation
3	Open search
4	Close search
5	Open cascade
6	Full grid
7	Copy screenshot to clipboard
8	Print screenshot
9	Save screenshot to a file

Return Value:

The return value is TRUE if the action succeeds; otherwise it is FALSE.



ID	Latitude	Longitude	City	Province/State	Country
1001	82°30'N	62°20'W	Alert	Nunavut	Canada
1002	81°36'N	16°40'W	Nord	Greenland	Denmark
1003	79°59'N	85°56'W	Eureka	Nunavut	Canada
1004	78°55'N	11°56'E	Ny-Ålesund	Svalbard	Norway
1005	78°13'N	15°39'E	Longyearbyen	Svalbard	Norway
1006	77°28'N	69°14'W	Qaanaaq	Greenland	Denmark
1007	76°25'N	82°54'W	Grise Fiord	Nunavut	Canada
1008	73°30'N	80°31'E	Dikson	Krasnoyarsk Krai	Russia
1009	72°47'N	56°09'W	Upernavik	Greenland	Denmark
1010	71°39'N	128°52'E	Tiksi	Sakha Republic	Russia
1011	71°32'N	52°19'E	Belushya Guba	Arkhangelsk Oblast	Russia
1012	71°18'N	156°46'W	Barrow	Alaska	USA

4.2.2 ConfigureAllGrids

Creates and configures the full grid and all sub grid(s).

```
[Visual Basic .NET]
Public Function ConfigureAllGrids() As Boolean
```

```
[C#]
public bool ConfigureAllGrids();
```

Parameters

None.

Return Value:

If the operation is successful, it returns TRUE; otherwise it returns FALSE.

Remarks

This API is very critical and needs to be called before setting up cells for both readonly and edit mode .

4.2.3 ConfigureCellImage

Specifies the attributes of an image that is rendered onto an individual cell for the grid in readonly mode - either the full grid or a sub grid.

```
[Visual Basic .NET]
Public Function ConfigureCellImage(ByVal GridIndex As Integer,
                                   ByVal SectionIndex As Integer,
                                   ByVal RowIndex As Integer,
                                   ByVal ColumnIndex As Integer,
                                   ByVal ImagePath As String,
                                   ByVal PaintWidth As Integer,
                                   ByVal PaintHeight As Integer,
                                   ByVal Alignment As Integer) As Boolean
```

```
[C#]
public bool ConfigureCellImage(int GridIndex,
                               int SectionIndex,
                               int RowIndex,
                               int ColumnIndex,
                               string ImagePath,
                               int PaintWidth,
                               int PaintHeight,
                               int Alignment);
```

Parameters

GridIndex
Specifies the grid index - either the full gird or a sub grid in cascade mode. ***-1*** indicates the full grid; for a sub grid, the parameter's value must be between 0 and the return value of GetTotalSubGrids() minus 1.

SectionIndex
Specifies the index (0-based) of the section in which the cell resides. If the AllowSections is set to FALSE, this parameter becomes irrelevant.

RowIndex
Specifies the index (0-based) of the row in which the cell resides.

ColumnIndex
Specifies the index (0-based) of the column in which the cell resides.

ImagePath
Specifies the absolute path of the image that will be rendered onto the cell.

PaintWidth

Specifies the width (in pixels) of the rectangle onto which the image is rendered.

PaintHeight

Specifies the height (in pixels) of the rectangle onto which the image is rendered.

Alignment

Specifies the alignment's type, and all valid values are listed below:

Value	Comment
3	If the multiple lines (up to 4) of texts are specified via ConfigureCellText, all texts are left-aligned and the image is rendered to the right of them. Otherwise the image is left-aligned.
4	The image is left-aligned and all texts are drawn to the right of it.
5	The image is center-aligned and no text is drawn.
6	If the multiple lines (up to 4) of texts are specified via ConfigureCellText, all texts are right-aligned and the image is rendered to the left of them. Otherwise the image is right-aligned.
7	The image is right-aligned and all texts of cell are drawn to the left of it.

Return Value:

If the operation is successful, it returns TRUE; otherwise it returns FALSE. For example, the image is not in a valid format.

Remarks

If GridIndex *is -1, the sync* cell in the associated sub grid is automatically configured. Otherwise, the sync cell is the full grid is automatically configured.

4.2.4 ConfigureCellComboBox

Specifies the attributes of a ComboBox column for the grid in edit mode.

```
[Visual Basic .NET]
Public Function ConfigureCellComboBox(ByVal SectionIndex As Integer,
                                      ByVal ColumnIndex As Integer,
                                      ByVal Items As ArrayList) As Boolean
```

```
[C#]
public bool ConfigureCellComboBox(int SectionIndex,
                                  int ColumnIndex,
                                  ArrayList Items);
```

Parameters

SectionIndex
Specifies the index (0-based) of the section in which configuring occurs. If the AllowSections is set to FALSE, this parameter becomes irrelevant. If it is set to TRUE, your code can set this parameter to -1 to let configuring take effect for all sections.

ColumnIndex
Specifies the index (0-based) of the column.

Items
Contains the items list in the ComboBox.

Return Value:

If the operation is successful, it returns TRUE; otherwise it returns FALSE.

4.2.5 ConfigureCellDatePicker

Specifies the attributes of a DatePicker column for the grid in edit mode.

```
[Visual Basic .NET]
Public Function ConfigureCellDatePicker(ByVal SectionIndex As Integer,
                                        ByVal ColumnIndex As Integer,
                                        ByVal MaxDate As DateTime,
                                        ByVal MinDate As DateTime,
                                        ByVal Format As DateTimePickerFormat,
                                        ByVal CustomFormat As String) As Boolean
```

```
[C#]
public bool ConfigureCellDatePicker(int SectionIndex,
                                    int ColumnIndex,
                                    DateTime MaxDate,
                                    DateTime MinDate,
                                    DateTimePickerFormat Format,
                                    String CustomFormat);
```

Parameters

SectionIndex
Specifies the index (0-based) of the section in which configuring occurs. If the AllowSections is set to FALSE, this parameter becomes irrelevant. If it is set to TRUE, your code can set this parameter to -1 to let configuring take effect for all sections.

ColumnIndex

Specifies the index (0-based) of the column.

MaxDate
Specifies the maximum date of DatePicker.

MinDate
Specifies the minimum date of DatePicker.

Format
Specifies the format of DataPicker.

CustomFormat
Specifies the custom format (e.g. "yyyy-MM-dd") of DatePicker if ***Format*** is set to DateTimePickerFormat.Custom.

Return Value:

If the operation is successful, it returns TRUE; otherwise it returns FALSE.

4.2.6 ConfigureCellMaskedTextBox

Specifies the attributes of a MaskedTextBox column for the grid in edit mode.

```
[Visual Basic .NET]
Public Function ConfigureCellMaskedTextBox(ByVal SectionIndex As Integer,
                                           ByVal ColumnIndex As Integer,
                                           ByVal AllowPromptAsInput As Boolean,
                                           ByVal CutCopyMaskFormat As MaskFormat,
                                           ByVal HidePromptOnLeave As Boolean,
                                           ByVal HideSelection As Boolean,
                                           ByVal Mask As String,
                                           ByVal PasswordChar As Char,
                                           ByVal PromptChar As Char,
                                           ByVal RejectInputOnFirstFailure As Boolean,
                                           ByVal ResetOnPrompt As Boolean,
                                           ByVal ResetOnSpace As Boolean,
                                           ByVal RTL As RightToLeft,
                                           ByVal TextAlign As HorizontalAlignment,
                                           ByVal TextMaskFormat As MaskFormat,
                                           ByVal UseSystemPasswordChar As Boolean) As
Boolean
```

```
[C#]
public bool ConfigureCellMaskedTextBox(int SectionIndex,
```

```
                                int ColumnIndex,
                                bool AllowPromptAsInput,
                                MaskFormat CutCopyMaskFormat,
                                bool HidePromptOnLeave,
                                bool HideSelection,
                                string Mask,
                                char PasswordChar,
                                char PromptChar,
                                bool RejectInputOnFirstFailure,
                                bool ResetOnPrompt,
                                bool ResetOnSpace,
                                RightToLeft RTL,
                                HorizontalAlignment TextAlign,
                                MaskFormat TextMaskFormat,
                                bool UseSystemPasswordChar);
```

Parameters

SectionIndex
Specifies the index (0-based) of the section in which configuring occurs. If the AllowSections is set to FALSE, this parameter becomes irrelevant. If it is set to TRUE, your code can set this parameter to -1 to let configuring take effect for all sections.

ColumnIndex
Specifies the index (0-based) of the column.

AllowPromptAsInput
Indicates whether PromptChar can be entered as valid data by the user, and the recommended value is TRUE.

CutCopyMaskFormat
Indicates whether the literals and prompt characters are copied to the clipboard, and the recommended value is IncludeLiterals.

HidePromptOnLeave
Indicates whether the prompt characters in the input mask are invisible when the control loses the focus, and the recommended value is FALSE.

HideSelection
Indicates whether the selected text becomes hidden when the control loses the focus, and the recommended value is TRUE.

Mask
Specifies the input mask.

PasswordChar
Specifies the character to be displayed in substitute when keying in a password.

PromptChar
Indicates the character used to prompt the user for input, and the recommended value underscore character "_".

RejectInputOnFirstFailure
Indicates whether the input should be terminated after the first invalid character is detected, and the recommended value is FALSE.

ResetOnPrompt
Indicates whether the prompt character entered as input causes the current editable position in the mask to be reset or not, and the recommended value is TRUE

ResetOnSpace
Indicates whether the space entered as input causes the current editable position in the mask to be reset, and the recommended value is TRUE.

RTL
Indicates whether control's elements are aligned to support locales using right-to-left fonts, and the recommended value is Inherit.

TextAlign
Indicates how text is aligned in the control.

TextMaskFormat
Indicates whether the literals and prompt characters are included in the formatted string, and the recommended value is IncludeLiterals.

UseSystemPasswordChar
Indicates whether the system password should be used as the prompt character, and the recommended value is FALSE.

Return Value:

If the operation is successful, it returns TRUE; otherwise it returns FALSE.

4.2.7 ConfigureCellNumericUpDown

Specifies the attributes of a NumericUpDown column for the grid in edit mode.

```
[Visual Basic .NET]
Public Function ConfigureCellNumericUpDown(ByVal SectionIndex As Integer,
                                           ByVal ColumnIndex As Integer,
                                           ByVal MaxValue As Double,
                                           ByVal MinValue As Double,
                                           ByVal DecimalPlaces As Integer,
                                           ByVal Increment As Double,
                                           ByVal ThousandsSeparator As Boolean,
                                           ByVal UpDownAlign As LeftRightAlignment) As Boolean
```

```
[C#]
public bool ConfigureCellNumericUpDown(int SectionIndex,
                                       int ColumnIndex,
                                       double MaxValue,
                                       double MinValue,
                                       int DecimalPlaces,
                                       double Increment,
                                       bool ThousandsSeparator,
                                       LeftRightAlignment UpDownAlign);
```

Parameters

SectionIndex
Specifies the index (0-based) of the section in which configuring occurs. If the AllowSections is set to FALSE, this parameter becomes irrelevant. If it is set to TRUE, your code can set this parameter to -1 to let configuring take effect for all sections.

ColumnIndex
Specifies the index (0-based) of the column.

MaxValue
Specifies the maximum value for the spin box.

MinValue
Specifies the minimum value for the spin box.

DecimalPlaces
Specifies the number of decimal places to display in the spin box.

Increment
Specifies the value to increase or decrease the Value property when the up or down buttons are clicked on the spin box.

ThousandsSeparator
Indicates whether a thousands separator is displayed in the spin box if applicable.

UpDownAlign
Indicates whether the up or down buttons are left or right aligned.

Return Value:

If the operation is successful, it returns TRUE; otherwise it returns FALSE.

4.2.8 ConfigureCellRichTextBox

Specifies the attributes of a RichTextBox column for the grid in edit mode.

```
[Visual Basic .NET]
```

```
Public Function ConfigureCellRichTextBox(ByVal SectionIndex As Integer,
                                         ByVal ColumnIndex As Integer,
                                         ByVal RTL As RightToLeft,
                                         ByVal WordWrap As Boolean,
                                         ByVal ZoomFactor As Double) As Boolean
```

```
[C#]
public bool ConfigureCellRichTextBox(int SectionIndex,
                                     int ColumnIndex,
                                     RightToLeft RTL,
                                     bool WordWrap,
                                     double ZoomFactor);
```

Parameters

SectionIndex
Specifies the index (0-based) of the section in which configuring occurs. If the AllowSections is set to FALSE, this parameter becomes irrelevant. If it is set to TRUE, your code can set this parameter to -1 to let configuring take effect for all sections.

ColumnIndex
Specifies the index (0-based) of the column.

RTL
Indicates whether control's elements are aligned to support locales using right-to-left fonts, and the recommended value is Inherit.

WordWrap
Indicates whether to break words when the content exceeds the boundaries of its container.

ZoomFactor
Specifies the current zoom level.

Return Value:

If the operation is successful, it returns TRUE; otherwise it returns FALSE.

4.2.9 ConfigureCellText

Specifies the multiple lines (up to 4) of texts that are drawn onto an individual cell for the grid in readonly mode - either the full grid or a sub grid.

```
[Visual Basic .NET]
Public Function ConfigureCellText(ByVal GridIndex As Integer,
                                  ByVal SectionIndex As Integer,
```

```
                                    ByVal RowIndex As Integer,
                                    ByVal ColumnIndex As Integer,
                                    ByVal Line_1 As String,
                                    ByVal IsLink_1 As Boolean,
                                    ByVal Line_2 As String,
                                    ByVal IsLink_2 As Boolean,
                                    ByVal Line_3 As String,
                                    ByVal IsLink_3 As Boolean,
                                    ByVal Line_4 As String,
                                    ByVal IsLink_4 As Boolean,
                                    ByVal BackColor As Integer,
                                    ByVal ForeColor As Integer,
                                    ByVal Alignment As Integer) As Boolean
```

```
[C#]
public bool ConfigureCellText(int GridIndex,
                              int SectionIndex,
                              int RowIndex,
                              int ColumnIndex,
                              string Line_1,
                              bool IsLink_1,
                              string Line_2,
                              bool IsLink_2,
                              string Line_3,
                              bool IsLink_3,
                              string Line_4,
                              bool IsLink_4,
                              long BackColor,
                              long ForeColor,
                              int Alignment);
```

Parameters

GridIndex

Specifies the grid index - either the full gird or a sub grid. ***-1*** indicates the full grid; for a sub grid, the parameter's value must be between 0 and the return value of GetTotalSubGrids() minus 1.

SectionIndex

Specifies the index (0-based) of the section in which the cell resides. If the AllowSections is set to FALSE, this parameter becomes irrelevant.

RowIndex
Specifies the index (0-based) of the row in which the cell resides.

ColumnIndex
Specifies the index (0-based) of the column in which the cell resides.

Line_1
Specifies the 1st line of Unicode text that is drawn onto the cell, and an empty string can be assigned here.

IsLink_1
Specifies whether the 1st line of Unicode text is a linked text or not.

Line_2
Specifies the 2nd line of Unicode text that is drawn onto the cell, and an empty string can be assigned here.

IsLink_2
Specifies whether the 2nd line of Unicode text is a linked text or not.

Line_3
Specifies the 3rd line of Unicode text that is drawn onto the cell, and an empty string can be assigned here.

IsLink_3
Specifies whether the 3rd line of Unicode text is a linked text or not.

Line_4
Specifies the 4th line of Unicode text that is drawn onto the cell, and an empty string can be assigned here.

IsLink_4
Specifies whether the 4th line of Unicode text is a linked text or not.

BackColor
Specifies the cell background's color value in 32-bit ARGB format. If it is 1, the background color will be determined by the GridBackColor property of the control. For example, -65536 can be passed to indicate the red color, as Color.Red.ToArgb() returns -65536.

ForeColor
Specifies the cell text's color value in 32-bit ARGB format. If it is 1, the text color will be determined by the RowTextColor property of the control.

Alignment
Specifies the alignment's type, and all valid values are listed below:

Value	Comment
0	If ConfigureCellImage() is not called, all texts are left-aligned. Otherwise the alignment will be determined by that API.
1	If ConfigureCellImage() is not called, all texts are center-aligned. Otherwise the alignment will be determined by that API.
2	If ConfigureCellImage() is not called, all texts are right-aligned. Otherwise

	the alignment will be determined by that API.

Return Value:

If the operation is successful, it returns TRUE; otherwise it returns FALSE. For example, one or more index values are out of bound.

Remarks

The sync cell in the associated grid is automatically configured.

4.2.10 ConfigureCellTextEdit

Specifies the attributes of a TextEdit column for the grid in edit mode.

```
[Visual Basic .NET]
Public Function ConfigureCellTextEdit(ByVal SectionIndex As Integer,
                                      ByVal ColumnIndex As Integer,
                                      ByVal CharCasing As CharacterCasing,
                                      ByVal RTL As RightToLeft,
                                      ByVal PasswordChar As Char,
                                      ByVal TextAlign As HorizontalAlignment,
                                      ByVal UseSystemPasswordChar As Boolean,
                                      ByVal Multiline As Boolean) As Boolean
```

```
[C#]
public bool ConfigureCellTextEdit(int SectionIndex,
                                  int ColumnIndex,
                                  CharacterCasing CharCasing,
                                  RightToLeft RTL,
                                  char PasswordChar,
                                  HorizontalAlignment TextAlign,
                                  bool UseSystemPasswordChar,
                                  bool Multiline);
```

Parameters

SectionIndex

Specifies the index (0-based) of the section in which configuring occurs. If the AllowSections is set to FALSE, this parameter becomes irrelevant. If it is set to TRUE, your code can set this parameter to -1 to let configuring take effect for all sections.

ColumnIndex

Specifies the index (0-based) of the column.

CharCasing
Indicates how characters are cased when they are manually keyed in.

RTL
Indicates whether control's elements are aligned to support locales using right-to-left fonts, and the recommended value is Inherit.

PasswordChar
Specifies the character to be displayed in substitute when keying in a password.

TextAlign
Indicates how text is aligned in the control.

UseSystemPasswordChar
Indicates whether the system password should be used as the prompt character, and the recommended value is FALSE.

Multiline
Indicates whether the multiline feature should be turned on or not.

Return Value:

If the operation is successful, it returns TRUE; otherwise it returns FALSE.

4.2.11 ConfigureCellTimePicker

Specifies the attributes of a TimePicker column for the grid in edit mode.

```
[Visual Basic .NET]
Public Function ConfigureCellTimePicker(ByVal SectionIndex As Integer,
                                        ByVal ColumnIndex As Integer,
                                        ByVal CustomFormat As String) As Boolean
```

```
[C#]
public bool ConfigureCellTimePicker(int SectionIndex,
                                    int ColumnIndex,
                                    string CustomFormat);
```

Parameters

SectionIndex
Specifies the index (0-based) of the section in which configuring occurs. If the AllowSections is set to FALSE, this parameter becomes irrelevant. If it is set to TRUE, your code can set this parameter to -1 to let configuring take effect for all sections.

ColumnIndex

Specifies the index (0-based) of the column.

CustomFormat
Specifies the custom format of TimePicker (e.g. "hh:mm").

Return Value:

If the operation is successful, it returns TRUE; otherwise it returns FALSE.

4.2.12 ConfigureEditCell

Specifies the content of an individual cell for the full grid in edit mode.

```
[Visual Basic .NET]
Public Function ConfigureEditCell(ByVal RowIndex As Integer,
                                  ByVal ColumnIndex As Integer,
                                  ByVal CellVal As String) As Boolean
```

```
[C#]
public bool ConfigureEditCell(int RowIndex,
                              int ColumnIndex,
                              string CellVal);
```

Parameters

RowIndex
Specifies the index (0-based) of the row in which the cell resides.

ColumnIndex
Specifies the index (0-based) of the column in which the cell resides.

CellVal
Specifies the cell value.

1. If the edit type of current column is CheckBox, set "YES" or "NO" to indicate whether it is checked or not.
2. If the edit type of current column is ComboBox, convert SelectedIndex to a string and pass it to this parameter.
3. If the edit type of current column is DatePicker, pass the Date string to this parameter.
4. If the edit type of current column is Image, pass the full path of image to this parameter.
5. If the edit type of current column is MaskedTextBox, NumericUpDown, Rich TextBox, or plain TextBox, simply pass the text value to this parameter. For the content with a few lines, insert "\r\n" between them and pass to this parameter.
6. If the edit type of current column is TimePicker, pass the Time string to this parameter.

Return Value:

If the operation is successful, it returns TRUE; otherwise it returns FALSE. For example, one or more index values are out of bound.

Remarks

The sync cell in the associated sub grid is automatically configured.

4.2.13 ConfigureEditRow

Specifies the attributes of a row for the full grid in edit mode.

```
[Visual Basic .NET]
Public Function ConfigureEditRow(ByVal RowIndex As Integer,
                                 ByVal Tag As Object) As Boolean
```

```
[C#]
public bool ConfigureEditRow(int RowIndex,
                             object Tag);
```

Parameters

RowIndex
Specifies the index (0-based) of the row.

Tag
Specifies the unique tag for current row.

Return Value:

If the operation is successful, it returns TRUE; otherwise it returns FALSE. For example, Tag is equal to another row's one.

Remarks

The sync row in the associated sub grid is automatically configured.

4.2.14 ConfigureRow

Specifies the attributes of a row for the full grid in readonly mode.

```
[Visual Basic .NET]
Public Function ConfigureRow(ByVal SectionIndex As Integer,
                             ByVal RowIndex As Integer,
                             ByVal Tag As Object,
                             ByVal IsChecked As Boolean) As Boolean
```

```
[C#]
public bool ConfigureRow(int SectionIndex,
                         int RowIndex,
                         object Tag,
                         int IsChecked);
```

Parameters

SectionIndex
Specifies the index (0-based) of the section in which the row resides. If the AllowSections is set to FALSE, this parameter becomes irrelevant.

RowIndex
Specifies the index (0-based) of the row.

Tag
Specifies the tag value of the row, which is used to uniquely identify the current row. Since after sorting a column, all rows are probably repositioned, so the tag value is used to identify which is which.

IsChecked
Specifies whether the current row is checked or not - either TRUE (checked) or FALSE (unchecked).

Return Value:

If the operation is successful, it returns TRUE; otherwise it returns FALSE. For example, one or more index values are out of bound.

If the row(s) are sorted by clicking one column header, this method returns FALSE.

Remarks

The sync row in the associated sub grid is automatically configured.

4.2.15 DeleteRow

Deletes a row in a readonly grid - either full grid or a sub grid.

```
[Visual Basic .NET]
Public Function DeleteRow(ByVal GridIndex As Integer,
                          ByVal SectionIndex As Integer,
                          ByVal RowIndex As Integer) As Boolean
```

```
[C#]
public bool DeleteRow(int GridIndex,
                      int SectionIndex,
```

```
                                int RowIndex);
```

Parameters

GridIndex
Specifies the grid index - either the full gird or a sub grid. ***-1*** indicates the full grid; for a sub grid, the parameter's value must be between 0 and the return value of GetTotalSubGrids() minus 1.

SectionIndex
Specifies the index (0-based) of the section in which the cell resides. If the AllowSections is set to FALSE, this parameter becomes irrelevant.

RowIndex
Specifies the index (0-based) of the row.

Return Value:

If the operation is successful, it returns TRUE; otherwise it returns FALSE. For example, one or more index values are out of bound.

If the row(s) are sorted by clicking one column header, this method returns FALSE.

Remarks

The sync row in the associated grid is automatically deleted.

4.2.16 GetCheckedRowAt

Gets the attributes of a checked row for the grid in readonly mode - either the full gird or a sub grid.

```
[Visual Basic .NET]
Public Function GetCheckedRowAt(ByVal GridIndex As Integer,
                                ByVal Index As Integer,
                                ByRef SectionIndex As Integer,
                                ByRef RowIndex As Integer,
                                ByRef OrigRowIndex As Integer) As Boolean
```

```
[C#]
public bool GetCheckedRowAt(int GridIndex,
                            int Index,
                            ref int SectionIndex,
                            ref int RowIndex,
                            ref int OrigRowIndex);
```

Parameters

GridIndex

Specifies the grid index - either the full gird or a sub grid. ***-1*** indicates the full grid; for a sub grid, the parameter's value must be between 0 and the return value of GetTotalSubGrids() minus 1.

Index

Specifies the index (0-based) of a checked row, and this parameter's value must be between 0 and the return value of GetCheckedRowsCount() minus 1.

SectionIndex

Points to a variable holding the section's index value (0-based) of a checked row. If the AllowSections is set to FALSE, this parameter becomes irrelevant.

RowIndex

Points to a variable holding the row's index value (0-based) of a checked row.

OrigRowIndex

Points to a variable holding the row's original index value (0-based) of a checked row.

Return Value:

If the operation is successful, it returns TRUE; otherwise it returns FALSE.

4.2.17 GetCheckedRowsCount

Gets the total number of checked row(s) for the grid in readonly mode - either the full gird or a sub grid.

[Visual Basic .NET]

```
Public Function GetCheckedRowsCount(ByVal GridIndex As Integer) As Integer
```

[C#]

```
public int GetCheckedRowsCount(int GridIndex);
```

Parameters

GridIndex

Specifies the grid index - either the full gird or a sub grid. ***-1*** indicates the full grid; for a sub grid, the parameter's value must be between 0 and the return value of GetTotalSubGrids() minus 1.

Return Value:

It returns the total number of checked row(s) in a grid.

4.2.18 GetColumnWidth

Gets the width in pixels of a column of current active grid - either the full grid or a sub grid.

[Visual Basic .NET]

```
Public Function GetColumnWidth(ByVal GridIndex As Integer, ByVal ColumnIndex As Integer) As
```

Integer

[C#]

public int GetColumnWidth(int GridIndex, int ColumnIndex);

Parameters

GridIndex

Specifies the grid index - either the full gird or a sub grid. ***-1*** indicates the full grid; for a sub grid, the parameter's value must be between 0 and the return value of GetTotalSubGrids() minus 1.

ColumnIndex

Specifies the index (0-based) of a column, and this parameter value must be between 0 and total number of the columns minus 1.

Return Value:

The return value is the width of the column.

4.2.19 GetCurrGridIndex

Gets the index of current active grid - either the full gird or a sub grid.

[Visual Basic .NET]

Public Function GetCurrGridIndex() As Integer

[C#]

public int GetCurrGridIndex();

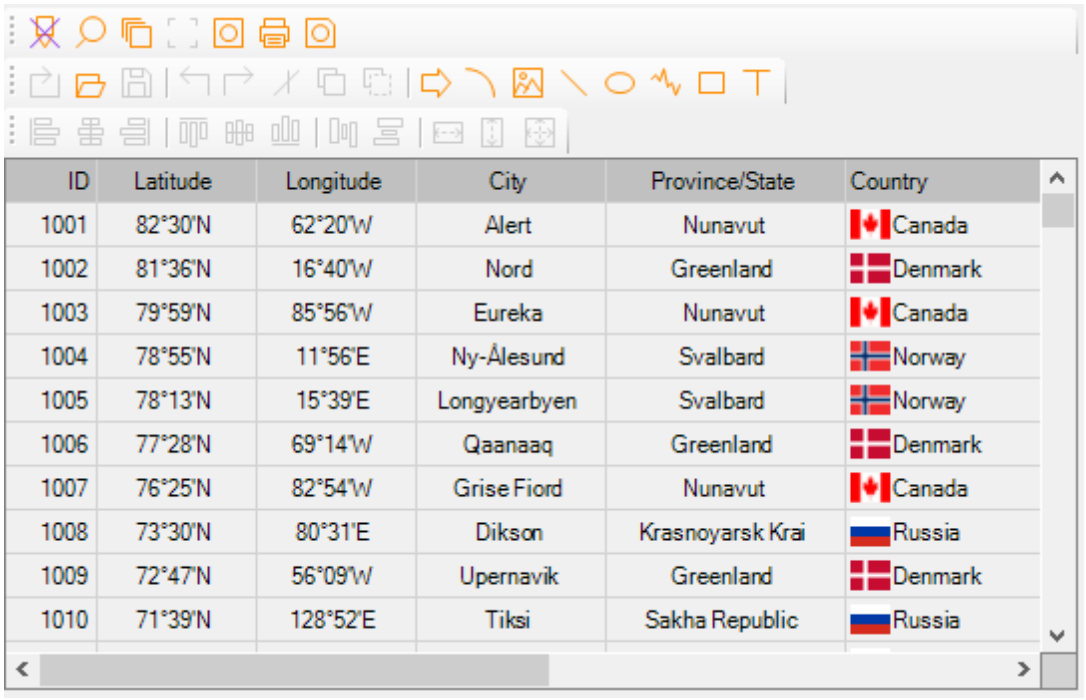
Parameters

None.

Return Value:

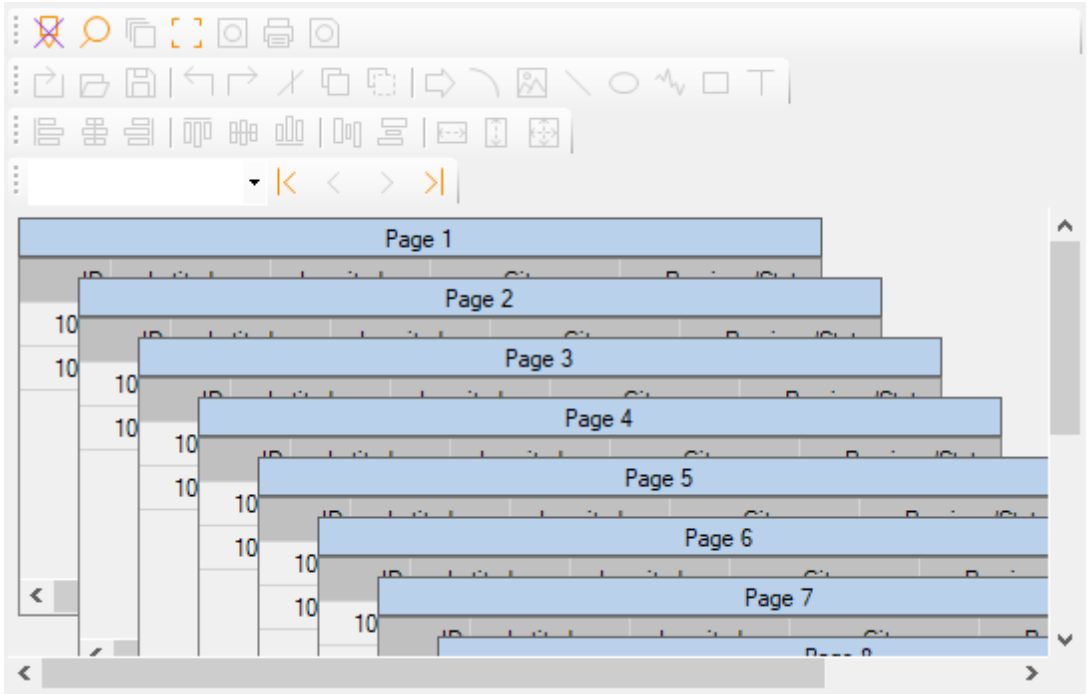
The return value is the index of current active grid.

(1) If the full grid is active, the return value is -1.

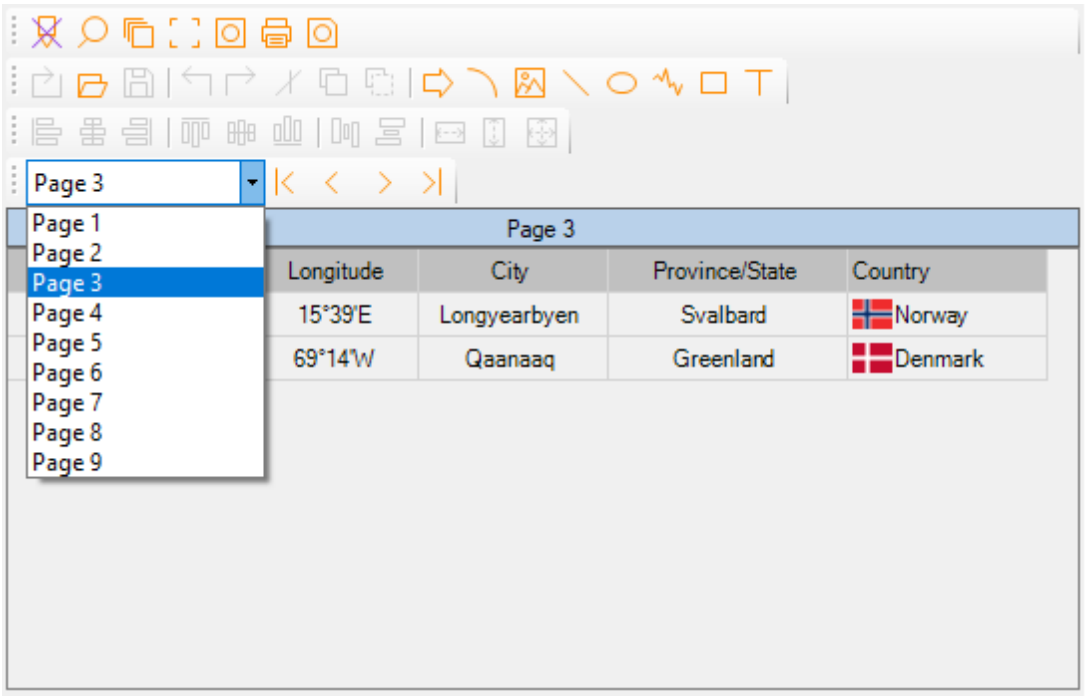


ID	Latitude	Longitude	City	Province/State	Country
1001	82°30'N	62°20'W	Alert	Nunavut	Canada
1002	81°36'N	16°40'W	Nord	Greenland	Denmark
1003	79°59'N	85°56'W	Eureka	Nunavut	Canada
1004	78°55'N	11°56'E	Ny-Ålesund	Svalbard	Norway
1005	78°13'N	15°39'E	Longyearbyen	Svalbard	Norway
1006	77°28'N	69°14'W	Qaanaaq	Greenland	Denmark
1007	76°25'N	82°54'W	Grise Fiord	Nunavut	Canada
1008	73°30'N	80°31'E	Dikson	Krasnoyarsk Krai	Russia
1009	72°47'N	56°09'W	Upernavik	Greenland	Denmark
1010	71°39'N	128°52'E	Tiksi	Sakha Republic	Russia

(2) For cascade mode, the return value is -2.



(3) For a active sub grid in cascade mode, the return value is between 0 and the return value of GetTotalSubGrids() minus 1. For example, the return value is 2 for the following case.



4.2.20 GetSelectedCellAt

Gets the attributes of a selected cell for the grid in readonly mode - either the full gird or a sub grid.

```
[Visual Basic .NET]
Public Function GetSelectedCellAt(ByVal GridIndex As Integer,
                                  ByVal Index As Integer,
                                  ByRef SectionIndex As Integer,
                                  ByRef RowIndex As Integer,
                                  ByRef OrigRowIndex As Integer,
                                  ByRef RowTag As Object,
                                  ByRef ColumnIndex As Integer,
                                  ByRef OrigColumnIndex As Integer) As Boolean
```

```
[C#]
public bool GetSelectedCellAt(int GridIndex,
                              int Index,
                              ref int SectionIndex,
                              ref int RowIndex,
                              ref int OrigRowIndex,
                              ref object RowTag,
                              ref int ColumnIndex,
```

```
                                        ref int OrigColumnIndex);
```

Parameters

GridIndex
Specifies the grid index - either the full gird or a sub grid. **-1** indicates the full grid; for a sub grid, the parameter's value must be between 0 and the return value of GetTotalSubGrids() minus 1.

Index
Specifies the index (0-based) of a selected cell, and this parameter's value must be between 0 and the return value of GetSelectedCellsCount() minus 1.

SectionIndex
Points to a variable holding the section's index value (0-based) of a selected cell. If the AllowSections is set to FALSE, this parameter becomes irrelevant.

RowIndex
Points to a variable holding the row's index value (0-based) of a selected cell.

OrigRowIndex
Points to a variable holding the row's original index value (0-based) of a selected cell.

RowTag
Points to a variable holding the row's tag value of a selected cell.

ColumnIndex
Points to a variable holding the column's index value (0-based) of a selected cell.

OrigColumnIndex
Points to a variable holding the column's original index value (0-based) of a selected cell.

Return Value:

If the operation is successful, it returns TRUE; otherwise it returns FALSE.

4.2.21 GetSelectedCellsCount

Gets the total number of selected cell(s) for the grid in readonly mode - either the full gird or a sub grid.

```
[Visual Basic .NET]
Public Function GetSelectedCellsCount(ByVal GridIndex As Integer) As Integer
```

```
[C#]
public int GetSelectedCellsCount(int GridIndex);
```

Parameters

GridIndex
Specifies the grid index - either the full gird or a sub grid. ***-1*** indicates the full grid; for a sub grid, the parameter's value must be between 0 and the return value of GetTotalSubGrids() minus 1.

Return Value:

It returns the total number of selected cell(s).

4.2.22 GetSubGridSizesList

Gets the list containing the numbers of rows for all sub grids.

```
[Visual Basic .NET]
Public Function GetSubGridSizesList() As ArrayList
```

```
[C#]
public ArrayList GetSubGridSizesList();
```

Parameters

None.

Return Value:

An array list containing the sizes of all sub grids from 1st sub grid to the last one.

4.2.23 GetTotalSubGrids

Gets the number of sub grids.

```
[Visual Basic .NET]
Public Function GetTotalSubGrids() As Integer
```

```
[C#]
public int GetTotalSubGrids();
```

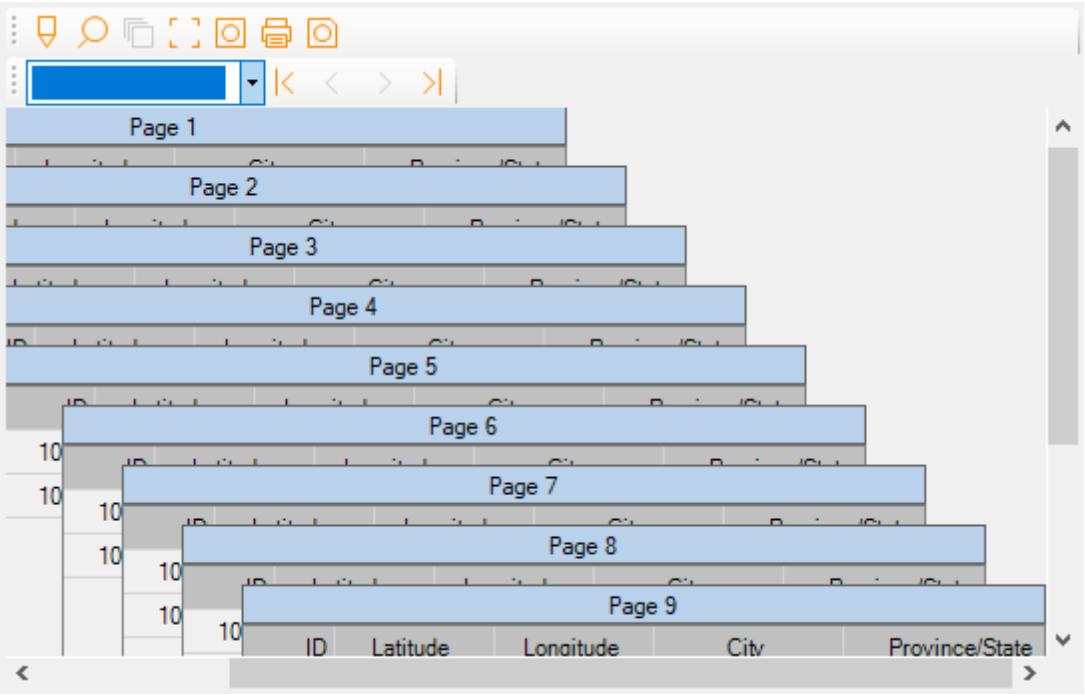
Parameters

None.

Return Value:

The return value is the number of sub grids.

For example, If AllowSections is FALSE, RowsCount is 17, SubGridSizesPattern is empty and SubGridSize is 2, and therefore the return value is 9.



4.2.24 GetVisibleIndexes

Gets the attributes of the visible area for the grid in readonly - either the full gird or a sub grid.

```
[Visual Basic .NET]
Public Function GetVisibleIndexes(ByVal GridIndex As Integer,
                                  ByRef SectionStart As Integer,
                                  ByRef RowStart As Integer,
                                  ByRef ColumnStart As Integer
                                  ByRef SectionEnd As Integer,
                                  ByRef RowEnd As Integer,
                                  ByRef ColumnEnd As Integer) As Boolean
```

```
[C#]
public bool GetVisibleIndexes(int GridIndex,
                              ref int SectionStart,
                              ref int RowStart,
                              ref int ColumnStart,
                              ref int SectionEnd,
                              ref int RowEnd,
                              ref int ColumnEnd);
```

Parameters

GridIndex

Specifies the grid index - either the full gird or a sub grid. ***-1*** indicates the full grid; for a sub grid, the parameter's value must be between 0 and the return value of GetTotalSubGrids() minus 1.

SectionStart
Points to a variable holding the section index (0-based) of the first visible row. If the AllowSections is set to FALSE, this parameter becomes irrelevant.

RowStart
Points to a variable holding the index (0-based) of the first visible row.

ColumnStart
Points to a variable holding the index (0-based) of the first visible column.

SectionEnd
Points to a variable holding the section index (0-based) of the last visible row. If the AllowSections is set to FALSE, this parameter becomes irrelevant.

RowEnd
Points to a variable holding the index (0-based) of the last visible row.

ColumnEnd
Points to a variable holding the index (0-based) of the last visible column.

Return Value:

If the operation is successful, it returns TRUE; otherwise it returns FALSE.

4.2.25 InsertRow

Inserts a row for the grid in readonly mode - either the full grid or a sub grid.

```
[Visual Basic .NET]
Public Function InsertRow(ByVal GridIndex As Integer,
                          ByVal SectionIndex As Integer,
                          ByVal RowIndex As Integer,
                          ByVal Tag As Object,
                          ByVal UseNextGrid As Boolean,
                          ByRef CopyGridIndex As Integer,
                          ByRef CopyRowIndex As Integer) As Boolean
```

```
[C#]
public bool InsertRow(int GridIndex,
                      int SectionIndex,
                      int RowIndex,
                      object Tag,
                      bool UseNextGrid,
```

```
                    ref int CopyGridIndex,
                    ref int CopyRowIndex);
```

Parameters

GridIndex
Specifies the working grid index - either the full gird or a sub grid. ***-1*** indicates the full grid; for a sub grid, the parameter's value must be between 0 and the return value of GetTotalSubGrids() minus 1.

SectionIndex
Specifies the index (0-based) of the section in which the cell resides. If the AllowSections is set to FALSE, this parameter becomes irrelevant.

RowIndex
Specifies the index (0-based) of the row in which the cell resides.

Tag
Specifies the Tag property of working row. If this row is a new one, it is null.

UseNextCopyGrid
Specifies whether to use next sub grid to insert a new crow. It becomes irrelevant if the new row is not on the border or the work grid is a sub grid**.**

CopyGridIndex
Points to a variable holding the index of sync grid, either the full grid or a sub grid, where to insert a sync row. If the working grid is the full grid, the sync grid is a sub grid; otherwise the sync grid is the full grid.

CopyRowIndex
Points to a variable holding the position (0-based) of a sync row in the sync grid indicated by *CopyGridIndex*.

Return Value:

If the operation is successful, it returns TRUE; otherwise it returns FALSE. For example, one or more index values are out of bound.

If the row(s) are sorted by clicking one column header, this method returns FALSE.

Remarks

The sync row in the associated grid is automatically inserted.

4.2.26 SaveScreenShot

Saves the screenshot of grid - - either the full gird or a sub grid - to an image file.

```
[Visual Basic .NET]
Public Function SaveScreenShot(ByVal  ImgFileName As String,
```

```
                                        ByVal  ImgFormat As ImageFormat) As Boolean
```

```
[C#]
public bool SaveScreenShot(string ImgFileName,
                                        ImageFormat ImgFormat);
```

Parameters

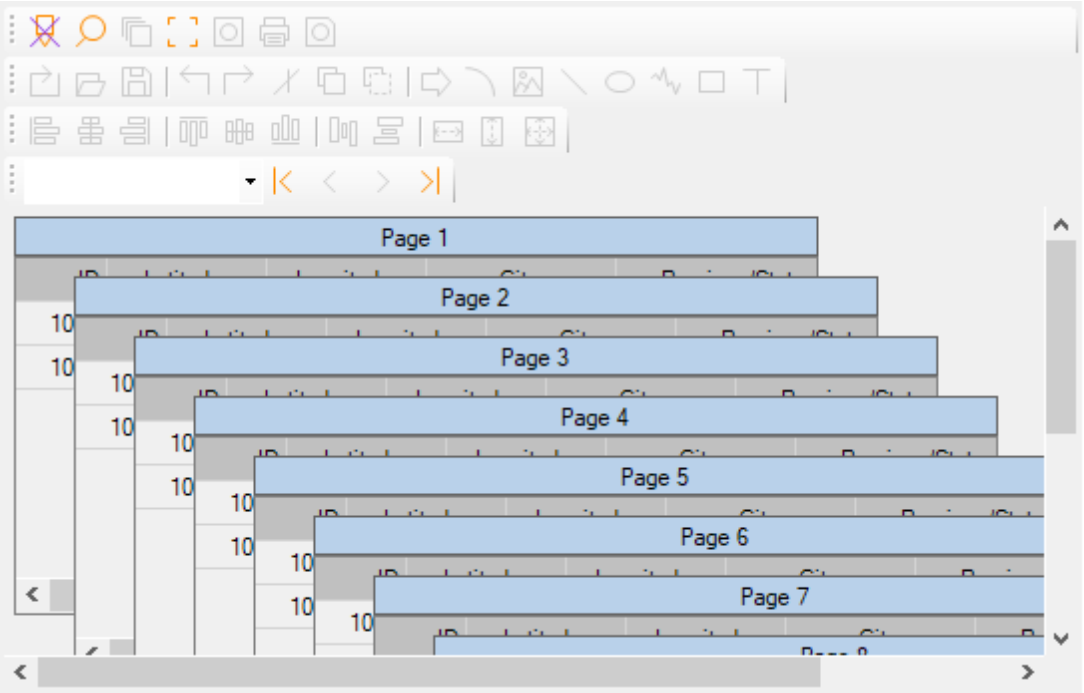
ImgFileName
Specifies the physical file name that contains the screenshot of grid.

ImgFormat
Specifies image format (e.g. BMP, GIF, PNG, etc.).

Return Value:

The return value is TRUE if the action succeeds; otherwise it is FALSE.

For cascade mode, the return value is FALSE and no screenshot can be saved until a sub grid is clicked and expanded.



4.2.27 SelectCell

Sets the status for a specific cell for the grid in readonly mode - either selected or not selected.

```
[Visual Basic .NET]
Public Function SelectCell(ByVal GridIndex As Integer,
```

```
                                ByVal SectionIndex As Integer,
                                ByVal RowIndex As Integer,
                                ByVal ColumnIndex As Integer,
                                ByVal IsSelected As Boolean) As Boolean
```

```
[C#]
public bool SelectCell(int GridIndex,
                       int SectionIndex,
                       int RowIndex,
                       int ColumnIndex,
                       bool IsSelected);
```

Parameters

GridIndex
Specifies the grid index - either the full gird or a sub grid. ***-1*** indicates the full grid; for a sub grid, the parameter's value must be between 0 and the return value of GetTotalSubGrids() minus 1.

SectionIndex
Specifies the index (0-based) of the section in which the cell resides. If the AllowSections is set to FALSE, this parameter becomes irrelevant.

RowIndex
Specifies the 0-based index of the row in which the cell resides.

ColumnIndex
Specifies the 0-based index of the column in which the cell resides. If the SelectRow is set to TRUE, this parameter becomes irrelevant.

IsSelected
Indicates the status of the cell - either TRUE (selected) or FALSE (not selected).

Return Value:

The return value is TRUE if the action succeeds; otherwise it is FALSE.

Remarks

The sync cell in the associated grid is automatically affected.

4.2.28 SetCheckedRow

Sets the status for the checkbox of a specific row for the grid in readonly mode - either the full gird or a sub grid.

```
[Visual Basic .NET]
```

```
Public Function SetCheckedRow(ByVal GridIndex As Integer,
                              ByVal SectionIndex As Integer,
                              ByVal RowIndex As Integer,
                              ByVal IsChecked As Boolean) As Boolean
```

```
[C#]
public bool SetCheckedRow(int GridIndex,
                          int SectionIndex,
                          int RowIndex,
                          bool IsChecked);
```

Parameters

GridIndex
Specifies the grid index - either the full gird or a sub grid. ***-1*** indicates the full grid; for a sub grid, the parameter's value must be between 0 and the return value of GetTotalSubGrids() minus 1.

SectionIndex
Specifies the index (0-based) of the section. If the AllowSections is set to FALSE, this parameter becomes irrelevant.

RowIndex
Specifies the index (0-based) of the row.

IsChecked
Indicates the status of row's checkbox - either TRUE (checked) or FALSE (unchecked).

Return Value:

The return value is TRUE if the action succeeds; otherwise it is FALSE.

Remarks

The sync row in the associated grid is automatically affected.

4.2.29 SetCheckedRows

Sets the status for all rows' checkbox(es) under a specific section for the full grid in readonly mode.

```
[Visual Basic .NET]
Public Function SetCheckedRows(ByVal SectionIndex As Integer, ByVal IsChecked As Boolean) As Boolean
```

```
[C#]
public bool SetCheckedRows(int SectionIndex, bool IsChecked);
```

Parameters

SectionIndex
Specifies the index (0-based) of the section. If it is -1, the action will be applied against all sections; otherwise the action will be applied against the specific section identified by this parameter.

IsChecked
Indicates the status of checkboxes - either TRUE (checked) or FALSE (unchecked).

Return Value:

The return value is TRUE if the action succeeds; otherwise it is FALSE.

Remarks

The sync row(s) in the associated sub grid are automatically affected.

4.2.30 SetSectionStatus

Sets the status - either expanded or collapsed - for a specific section - for the full grid.

[Visual Basic .NET]

```
Public Function SetSectionStatus(ByVal SectionIndex As Integer, ByVal IsExpanded As Boolean) As Boolean
```

[C#]

```
public bool SetSectionStatus(int SectionIndex, bool IsExpanded);
```

Parameters

SectionIndex
Specifies the index (0-based) of the section. If it is -1, the action will be applied against all sections; otherwise the action will be applied against the specific section identified by this parameter.

IsExpanded
Indicates the node status of the section(s) - either TRUE (expanded) or FALSE (collapsed).

Return Value:

The return value is TRUE if the action succeeds; otherwise it is FALSE.

4.2.31 SetSubGridTitle

Sets the title of a sub grid in cascade mode.

```
[Visual Basic .NET]
Public Function SetSubGridTitle(ByVal GridIndex As Integer, ByVal GridTitle As String) As Boolean
```

```
[C#]
public bool SetSubGridTitle(int GridIndex, String GridTitle);
```

Parameters

GridIndex
Specifies the grid index - either the full gird or a sub grid. ***-1*** indicates the full grid; for a sub grid, the parameter's value must be between 0 and the return value of GetTotalSubGrids() minus 1.

GridTitle
Specifies the title of sub grid.

Return Value:

The return value is TRUE if the action succeeds; otherwise it is FALSE.

4.2.32 SetVisibleCell

Makes a specific cell visible for the grid in readonly mode - either the full gird or a sub grid.

```
[Visual Basic .NET]
Public Function SetVisibleCell(ByVal GridIndex As Integer,
                               ByVal SectionIndex As Integer,
                               ByVal RowIndex As Integer,
                               ByVal ColumnIndex As Integer) As Boolean
```

```
[C#]
public bool SetVisibleCell(int GridIndex,
                           int SectionIndex,
                           int RowIndex,
                           int ColumnIndex);
```

Parameters

GridIndex
Specifies the working grid index - either the full gird or a sub grid. ***-1*** indicates the full grid; for a sub grid, the parameter's value must be between 0 and the return value of GetTotalSubGrids() minus 1.

SectionIndex

Specifies the index (0-based) of the section in which the cell resides. If the AllowSections is set to FALSE, this parameter becomes irrelevant.

RowIndex
Specifies the index (0-based) of the row in which the cell resides.

ColumnIndex
Specifies the index (0-based) of the column in which the cell resides. If the SelectRow is set to TRUE, this parameter becomes irrelevant.

Return Value:

The return value is TRUE if the action succeeds; otherwise it is FALSE.

4.2.33 UpdateEditColumns

Updates the attributes of columns for the full grid and all sub grids in edit mode.

```
[Visual Basic .NET]
Public Function UpdateEditColumns() As Boolean
```

```
[C#]
public bool UpdateEditColumns();
```

Parameters

None.

Return Value:

If the operation is successful, it returns TRUE; otherwise it returns FALSE.

Remarks

This API is very critical and needs to be called before setting up cells for both readonly and edit mode. Immediately call this API after calling ConfigureAllGrids()

4.3 Events

4.3.1 MGChangeGrid

This event is fired when a new sub grid is displayed in cascade mode.

```
[Visual Basic .NET]
Public Event MGChangeGrid(ByVal s As Object, ByVal e As ChangeGridEventArgs)
```

```
[C#]
```

```
public event MGChangeGrid(object s, ChangeGridEventArgs e);
```

Parameters

s

Specifies the sender object.

e

is the object of ***ChangeGridEventArgs*** Class with the following members:

1. **PrevGridIndex** specifies the index of previously displayed sub grid, this member's value must be between 0 and the return value of GetTotalSubGrids() minus 1.
2. **CurrGridIndex** specifies the index of currently displayed sub grid, this member's value must be between 0 and the return value of GetTotalSubGrids() minus 1.
3. **NewRowsCount** specifies the number of rows in currently displayed sub grid.
4. **NewGridTitle** specifies the grid title of currently displayed sub grid.

4.3.2 MGChangeSearchMatch

This event is fired when a new matched text is highlighted in search mode.

```
[Visual Basic .NET]
Public Event MGChangeSearchMatch(ByVal s As Object, ByVal e As ChangeSearchMatchEventArgs)
```

```
[C#]
public event MGChangeSearchMatch(object s, ChangeSearchMatchEventArgs e);
```

Parameters

s

Specifies the sender object.

e

is the object of ***ChangeSearchMatchEventArgs*** Class with the following members:

1. **CurrMatchIndex** specifies the index of currently highlighted text, this member's value must be between 0 and the number of total matches minus 1.
2. **GridIndex** specifies the index of current active grid where searching occurs. ***-1*** indicates the full grid; for a sub grid in cascade mode, this member's value must be between 0 and the return value of GetTotalSubGrids() minus 1.

4.3.3 MGClickCheckBox

This event is fired when the checkbox of a row is checked or unchecked.

```
[Visual Basic .NET]
```

```
Public Event MGClickCheckBox(ByVal s As Object, ByVal e As CheckBoxEventArgs)
```

```
[C#]
public event MGClickCheckBox(object s, CheckBoxEventArgs e);
```

Parameters

s

Specifies the sender object.

e

is the object of ***CheckBoxEventArgs*** Class with the following members:

1. **GridIndex** specifies the index of current active grid where clicking occurs. ***-1*** indicates the full grid; for a sub grid, this member's value must be between 0 and the return value of GetTotalSubGrids() minus 1.
2. **SectionIndex** specifies the section index (0-based) of the row containing the checkbox. If the AllowSections is set to FALSE or the active grid is a sub grid not the full grid, this parameter becomes irrelevant.
3. **CurrRowIndex** specifies the current index (0-based) of the row containing the checkbox.
4. **OrigRowIndex** specifies the original index (0-based) of the row containing the checkbox.
5. **NewValue** specifies the checked status of the checkbox - either TRUE (checked) or FALSE (unchecked).

4.3.4 MGClickIcon

This event is fired when an icon - in the first tool strip in the control - gets clicked.

```
[Visual Basic .NET]
Public Event MGClickIcon(ByVal s As Object, ByVal e As ClickIconEventArgs)
```

```
[C#]
public event MGClickIcon(object s, ClickIconEventArgs e);
```

Parameters

s

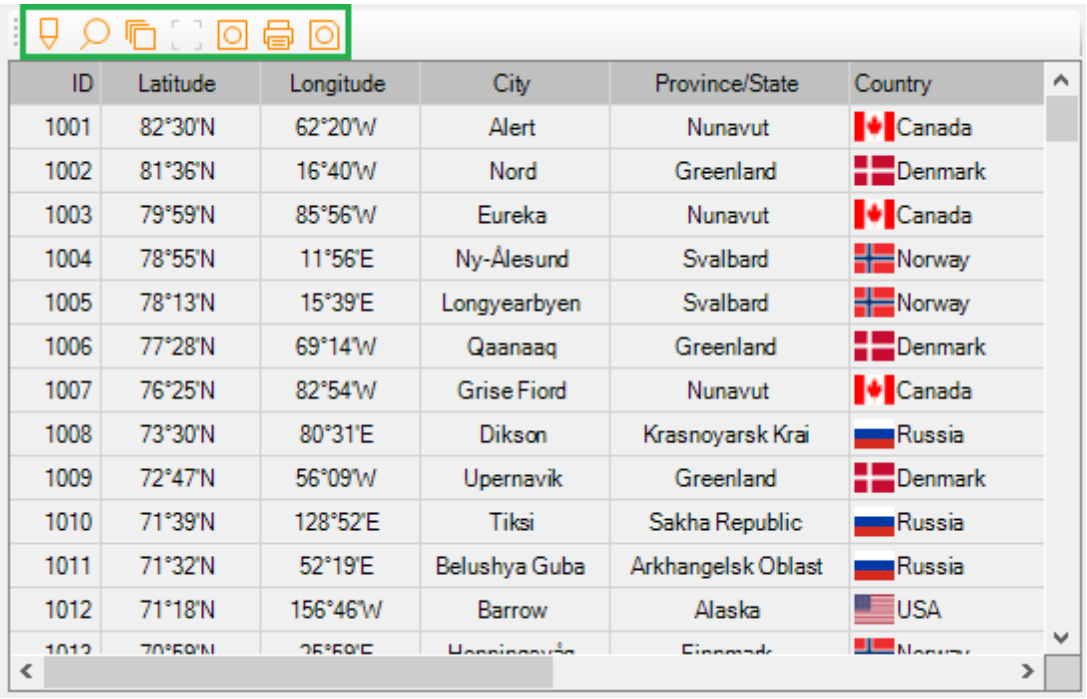
Specifies the sender object.

e

is the object of ***ClickIconEventArgs*** Class with the following member:

1. **IconFlag** specifies which icon gets clicked, and all possible values are listed below.

Value	Comment
1	Open annotation
2	Close annotation
3	Open search
4	Close search
5	Open cascading
6	Open full grid
7	Capture screen shot



ID	Latitude	Longitude	City	Province/State	Country
1001	82°30'N	62°20'W	Alert	Nunavut	Canada
1002	81°36'N	16°40'W	Nord	Greenland	Denmark
1003	79°59'N	85°56'W	Eureka	Nunavut	Canada
1004	78°55'N	11°56'E	Ny-Ålesund	Svalbard	Norway
1005	78°13'N	15°39'E	Longyearbyen	Svalbard	Norway
1006	77°28'N	69°14'W	Qaanaaq	Greenland	Denmark
1007	76°25'N	82°54'W	Grise Fiord	Nunavut	Canada
1008	73°30'N	80°31'E	Dikson	Krasnoyarsk Krai	Russia
1009	72°47'N	56°09'W	Upernavik	Greenland	Denmark
1010	71°39'N	128°52'E	Tiksi	Sakha Republic	Russia
1011	71°32'N	52°19'E	Belushya Guba	Arkhangelsk Oblast	Russia
1012	71°18'N	156°46'W	Barrow	Alaska	USA

.

4.3.5 MGClickLink

This event is fired when one hyperlink text gets clicked.

```
[Visual Basic .NET]
Public Event MGClickLink(ByVal s As Object, ByVal e As ClickLinkEventArgs)
```

```
[C#]
public event MGClickLink(object s, ClickLinkEventArgs e);
```

Parameters

s

Specifies the sender object.

e

is the object of ***ClickLinkEventArgs*** Class with the following members:

1. **GridIndex** specifies the index of current active grid where clicking occurs. ***-1*** indicates the full grid; for a sub grid, this member's value must be between 0 and the return value of GetTotalSubGrids() minus 1.
2. **SectionIndex** specifies the section index (0-based) of the cell containing the clicked hyperlink text. If the AllowSections is set to FALSE or the active grid is a sub grid not the full grid, this parameter becomes irrelevant.
3. **CurrRowIndex** specifies the current row index (0-based) of the cell containing the clicked hyperlink text.
4. **OrigRowIndex** specifies the original row index (0-based) of the cell containing the clicked hyperlink text.
5. **CurrColumnIndex** specifies the current column index (0-based) of the cell containing the clicked hyperlink text.
6. **OrigColumnIndex** specifies the original column index (0-based) of the cell containing the clicked hyperlink text.
7. **LinkIndex** specifies the index of the clicked text inside the cell, which can be 0, 1, 2, and 3, as a cell can hold up to 4 lines of texts.
8. **LinkStr** specifies the clicked text value.

4.3.6 MGClickPopupMenuItem

This event is fired when a menu item in the popup menu gets clicked in the grid control.

```
[Visual Basic .NET]
Public Event MGClickPopupMenuItem(ByVal s As Object, ByVal e As
PopupMenuItemClickedEventArgs)
```

```
[C#]
public event MGClickPopupMenuItem(object s, PopupMenuItemClickedEventArgs e);
```

Parameters

s

Specifies the sender object.

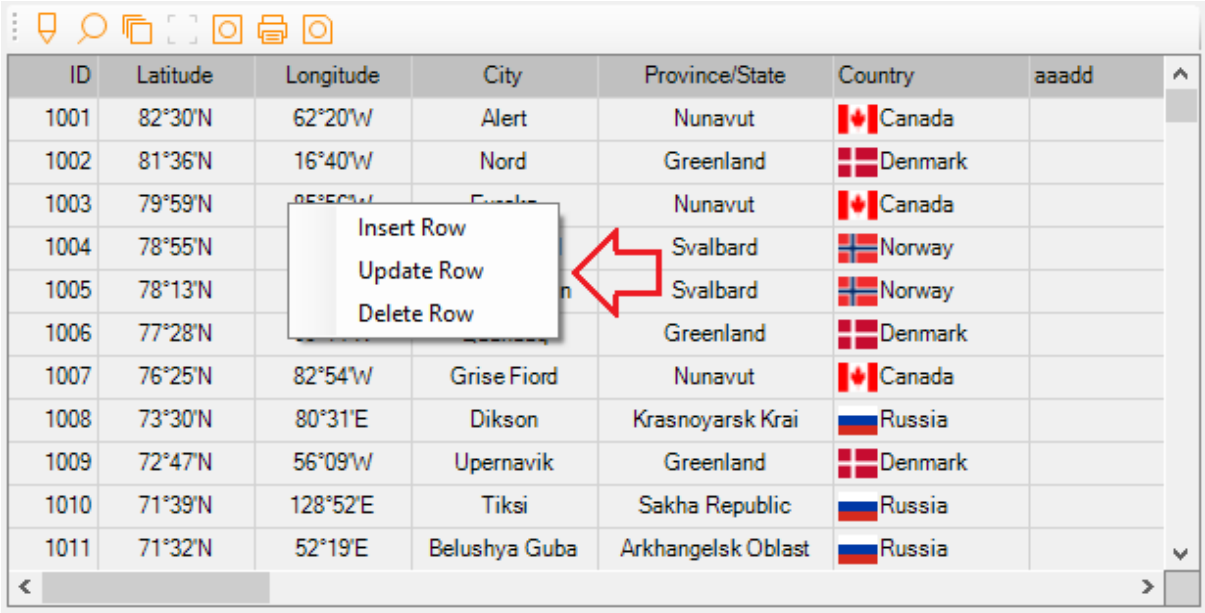
e

is the object of ***PopupMenuItemClickedEventArgs*** Class with the following members:

1. **GridIndex** specifies the index of current active grid. ***-1*** indicates the full grid; for a sub grid, this member's value must be between 0 and the return value of GetTotalSubGrids() minus 1.
2. **SectionIndex** specifies the section index (0-based). If the AllowSections is set to FALSE or

the active grid is a sub grid, this parameter becomes irrelevant.

3. **RowIndex** specifies the current index (0-based) of the row where popup menu shows via right-clicking.
4. **RowTag** specifies the tag of the row where popup menu shows via right-clicking.
5. **Tag** specifies the tag of the popup menu item.
6. **Text** specifies the title of the popup menu item.



ID	Latitude	Longitude	City	Province/State	Country	aaadd
1001	82°30'N	62°20'W	Alert	Nunavut	Canada	
1002	81°36'N	16°40'W	Nord	Greenland	Denmark	
1003	79°59'N			Nunavut	Canada	
1004	78°55'N			Svalbard	Norway	
1005	78°13'N			Svalbard	Norway	
1006	77°28'N			Greenland	Denmark	
1007	76°25'N	82°54'W	Grise Fiord	Nunavut	Canada	
1008	73°30'N	80°31'E	Dikson	Krasnoyarsk Krai	Russia	
1009	72°47'N	56°09'W	Upernavik	Greenland	Denmark	
1010	71°39'N	128°52'E	Tiksi	Sakha Republic	Russia	
1011	71°32'N	52°19'E	Belushya Guba	Arkhangelsk Oblast	Russia	

4.3.7 MGClickSectionNode

This event is fired when the node of a section is expanded or collapsed.

```
[Visual Basic .NET]
Public Event MGClickSectionNode(ByVal s As Object, ByVal e As NodeEventArgs)
```

```
[C#]
public event MGClickSectionNode(object s, NodeEventArgs e);
```

Parameters

s

Specifies the sender object.

e

is the object of ***NodeEventArgs*** Class with the following members:

1. **SectionIndex** specifies the section index (0-based) of the row containing the checkbox. If the AllowSections is set to FALSE, this parameter becomes irrelevant and this event will not

be fired.

2. **NewValue** specifies the node status of the section - either TRUE (expanded) or FALSE (collapsed).

4.3.8 MGCompareRows

This event may be fired many times when sorting a column - triggered by clicking the column header - is processed.

```
[Visual Basic .NET]
Public Event MGCompareRows(ByVal s As Object, ByVal e As CompareRowsEventArgs)
```

```
[C#]
public event MGCompareRows(object s, CompareRowsEventArgs e);
```

Parameters

s

Specifies the sender object.

e

is the object of ***CompareRowsEventArgs*** Class with the following members:

1. **GridIndex** specifies the index of current active grid where sorting occurs. ***-1*** indicates the full grid; for a sub grid, this member's value must be between 0 and the return value of GetTotalSubGrids() minus 1.
2. **SectionIndex** specifies the section index (0-based) of the sorted cells. If the AllowSections is set to FALSE or the active grid is a sub grid not the full grid, this parameter becomes irrelevant.
3. **CurrColumnIndex** specifies the current column index (0-based) of the sorted cell(s).
4. **OrigColumnIndex** specifies the original column index (0-based) of the sorted cell(s).
5. **LeftCurrRowIndex** specifies the current row index (0-based) of the left cell involved in the comparison operation.
6. **LeftOrigRowIndex** specifies the original row index (0-based) of the left cell involved in the comparison operation.
7. **LeftString** specifies the text value of the left cell involved in the comparison operation.
8. **RightCurrRowIndex** specifies the current row index (0-based) of the right cell involved in the comparison operation.
9. **RightOrigRowIndex** specifies the original row index (0-based) of the right cell involved in the comparison operation.
10. **RightString** specifies the text value of the right cell involved in the comparison operation.
11. **Result** points to a variable holding the comparison result, which can be -1, 0, or 1.

4.3.9 MGEditChangeRows

This event is fired when an action - adding, updating, or deleting row(s) - occurs for the grid in edit mode.

```
[Visual Basic .NET]
Public Event MGEditChangeRows(ByVal s As Object, ByVal e As EditChangeRowsEventArgs)

[C#]
public event MGEditChangeRows(object s, EditChangeRowsEventArgs e);
```

Parameters

s

Specifies the sender object.

e

is the object of ***EditChangeRowsEventArgs*** Class with the following members:

1. **Done** indicates whether the event subscriber has finished processing rows or not. The initial value is FALSE, the subscriber should set it to TRUE after everything is done.
2. **GridIndex** specifies the index of working grid, ***-1*** for the full grid or the grid index for the working sub grid between 0 and the return value of GetTotalSubGrids() minus 1.
3. **RowsList** is the object of ArrayList containing all related rows - 1 row for adding and deleting, potentially more than 1 row for updating. For more information, please refer to **Remarks**.

Remarks

Each item of **RowsList** is the object of ***EventEditGridRow*** Class containing the members below:

1. **Flag** indicates the action type - 1 for adding, 2 for updating, and 3 for deleting.
2. **SectionIndex** specifies the index (0-based) of the section where the operation occurs for the full grid. If the AllowSections is set to FALSE, this parameter becomes irrelevant. Otherwise the members in red become irrelevant, as there is no ambiguity on which sub grid is used to insert a new sync row when a new row is inserted into the full grid with the AllowSections set to TRUE, and the sub grid is uniquely determined by this member.
3. **OnBorder** indicates whether the newly added row lies on the border of 2 adjacent sub grids or not. If a new row is inserted into the full grid and a sync row should be inserted to the associated sub grid, MegaGrid needs to find out which sub grid to add it. For example, if the property RowsCount is 17, there are 4 sub grids containing 3, 5, 2, and 7 rows respectively. If a new row is inserted into position 7 (0-based) of the full grid, which is the border position between 2nd (0-based index 1) and 3rd (0-based index 2) sub grid, this field is set to TRUE. Otherwise it is set to FALSE. If a new row is inserted to a sub grid, this field is always set to FALSE.
4. **UseNextCopyGrid** specifies whether to use next sub grid to insert a new crow. It becomes irrelevant if **OnBorder** is FALSE.

5. **CopyGridIndex** specifies the index (0-based) of sub grid where to insert a sync row if a new row is inserted into the full grid.
6. **CopyRowIndex** specifies the position (0-based) of a sync row in the sub grid indicated by **CopyGridIndex**.
7. **NextCopyGridIndex** specifies the index (0-based) of next sub grid where to insert a sync row if a new row is inserted into the full grid. If **OnBorder** is FALSE**,** this field value is equal to **CopyGridIndex.**
8. **NextCopyRowIndex** specifies the position (0-based) of a sync row in the next sub grid indicated by **NextCopyGridIndex**. If **OnBorder** is FALSE**,** this field value is equal to **CopyRowIndex.**
9. **Tag** specifies the Tag property of working row. If the row is a new one, it is null.
10. **RowIndex** specifies the position (0-based) of working row in the working grid - either the full grid or a sub grid**.**
11. **MainRowIndex** specifies the position (0-based) of sync row for the working row in the working grid. If the working grid is the full grid, this field value is equal to **RowIndex**.
12. **Success** indicates whether the subscriber has successfully finished processing the working row.
13. **Message** contains the error information sent back to MegaGrid if something is wrong and **Success** is set to FALSE.
14. **CellsList** is the object of ArrayList containing all related working cells.

Each item of **CellsList** is the object of ***EventEditGridCell*** Class containing the members below:

1. **ColumnName** specifies the column name of working cell.
2. **OrigColumnIndex** specifies the original column index of working cell.
3. **CurrColumnIndex** specifies the current column index of work cell. This field is probably different from **OrigColumnIndex** after swapping columns.
4. **OldValue** specifies the old value of working cell.
5. **NewValue** specifies the new value of working cell.

4.3.10 MGMoveColumn

This event is fired when moving column(s) is finished.

```
[Visual Basic .NET]
Public Event MGMoveColumn(ByVal s As Object, ByVal e As MoveColumnEventArgs)
```

```
[C#]
public event MGMoveColumn(object s, MoveColumnEventArgs e);
```

Parameters

s

Specifies the sender object.

e

is the object of ***MoveColumnEventArgs*** Class with the following members:

1. **GridIndex** specifies the index of current active grid where moving occurs. ***-1*** indicates the full grid; for a sub grid in cascade mode, this member's value must be between 0 and the return value of GetTotalSubGrids() minus 1.
2. **SrcStartColumnIndex** specifies the index (0-based) of the first moving column.
3. **SrcEndColumnIndex** specifies the index (0-based) of the last moving column.
4. **DstColumnIndex** specifies the index (0-based) of the target column.

4.3.11 MGResizeColumn

This event is fired when resizing a column is finished.

```
[Visual Basic .NET]
Public Event MGResizeColumn(ByVal s As Object, ByVal e As ResizeColumnEventArgs)
```

```
[C#]
public event MGResizeColumn(object s, ResizeColumnEventArgs e);
```

Parameters

s

Specifies the sender object.

e

is the object of ***ResizeColumnEventArgs*** Class with the following members:

1. **GridIndex** specifies the index of current active grid where resizing occurs. ***-1*** indicates the full grid; for a sub grid in cascade mode, this member's value must be between 0 and the return value of GetTotalSubGrids() minus 1.
2. **LeftColumnIndex** specifies the index (0-based) of the left resizing column. As when it comes to resizing a column except for the last column, it will automatically resize an adjacent column. This member indicates the index of the left column being involved in resizing.
3. **RightColumnIndex** specifies the index (0-based) of the right resizing column. If **LeftColumnIndex** indicates the last column, this member becomes -1**.**

4.3.12 MGSearchSummary

This event is fired when a search is being conducted.

```
[Visual Basic .NET]
Public Event MGSearchSummary(ByVal s As Object, ByVal e As SearchSummaryEventArgs)
```

```
[C#]
```

```
public event MGSearchSummary(object s, SearchSummaryEventArgs e);
```

Parameters

s

Specifies the sender object.

e

is the object of ***SearchSummaryEventArgs*** Class with the following members:

1. **GridIndex** specifies the index of current active grid where searching occurs. ***-1*** indicates the full grid; for a sub grid in cascade mode, this member's value must be between 0 and the return value of GetTotalSubGrids() minus 1.
2. **SearchText** specifies the search text.
3. **TotalMatches** specifies the number of total matches.

4.3.13 MGSelectCell

This event is fired when a cell is selected for the grid in readonly mode.

```
[Visual Basic .NET]
Public Event MGSelectCell(ByVal s As Object, ByVal e As SelectCellEventArgs)
```

```
[C#]
public event MGSelectCell(object s, SelectCellEventArgs e);
```

Parameters

s

Specifies the sender object.

e

is the object of ***SelectCellEventArgs*** Class with the following members:

1. **GridIndex** specifies the index of current active grid where selecting occurs. ***-1*** indicates the full grid; for a sub grid in cascade mode, this member's value must be between 0 and the return value of GetTotalSubGrids() minus 1.
2. **SectionIndex** specifies the section index (0-based) of the selected cell. If the AllowSections is set to FALSE or the active grid is a sub grid not the full grid, this parameter becomes irrelevant.
3. **RowTag** specifies the tag value of current row.
4. **CurrRowIndex** specifies the current row index (0-based) of the selected cell.
5. **CurrColumnIndex** specifies the current column index (0-based) of the selected cell.

6. **OrigColumnIndex** specifies the original column index (0-based) of the selected cell.

4.3.14 MGShowPopup

This event is fired when the popup menu is displayed for the grid in readonly mode.

```
[Visual Basic .NET]
Public Event MGShowPopup(ByVal s As Object, ByVal e As ShowPopupEventArgs)
```

```
[C#]
public event MGShowPopup(object s, ShowPopupEventArgs e);
```

Parameters

s

Specifies the sender object.

e

is the object of ***ShowPopupEventArgs*** Class with the following members:

1. **GridIndex** specifies the index of current active grid. ***-1*** indicates the full grid; for a sub grid, this member's value must be between 0 and the return value of GetTotalSubGrids() minus 1.
2. **SectionIndex** specifies the section index (0-based). If the AllowSections is set to FALSE or the active grid is a sub grid, this parameter becomes irrelevant.
3. **RowIndex** specifies the current index (0-based) of the row where popup menu shows via right-clicking.
4. **RowTag** specifies the tag of the row where popup menu shows via right-clicking.
5. **X** specifies X-coordinate of right-clicking.
6. **Y** specifies Y-coordinate of right-clicking.

4.4 Enumerations

4.4.1 enumAnnotateBorderMarker

An enumeration type for all possible values of border marker for annotation tools.

Members

Value	Comment
0 - Icon_File	Image file as a marker
1 - Circle_Shape	Circle markers
2 - Triangle_Shape	Triangle markers
3 - Rhombus	Rhombus markers
4 - Square_Shape	Square markers

5 - Square_Shape_Diagonal_Lines	
6 - Square_Shape_Cross_Lines	

4.4.2 enumArrowLineWidth

An enumeration type for all possible values of the arrow line width.

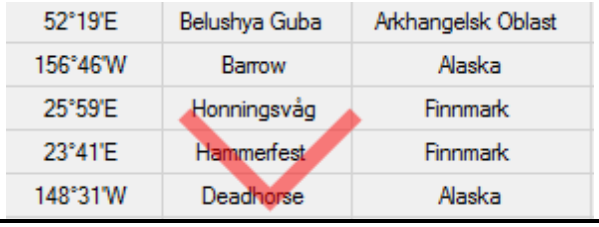
Members

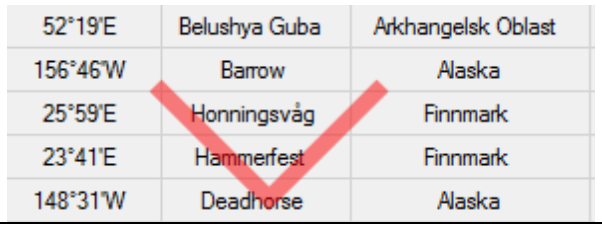
Name	Screen shot
Small	
Medium	
Large	

4.4.3 enumArrowSize

An enumeration type for all possible values of the arrow size.

Members

Name	Screen shot
Small	

Medium	
Large	

4.4.4 enumBoxStyle

An enumeration type for all possible values of the box style - checkbox(es) and/or section node(s) - for the grid.

Members

Value	Screenshot for Checkbox
0 - Athens	
1 - Auckland	
2 - Barcelona	
3 - Budapest	
4 - Calgary	
5 - Cape_Town	
6 - Frankfurt	
7 - Honolulu	
8 - Lisbon	
9 - London	
10 - Moscow	
11 - NYC	
12 - Paris	
13 - Prague	

14 - Rio	
15 - Rome	
16 - Shanghai	
17 - Sydney	
18 - Tokyo	

4.4.5 enumCellEditType

An enumeration type for all available control types for edit mode.

Members

Value	Comment
1 - CheckBox	
2 - ComboBox	
3 - DatePicker	
4 - Image	
5 - Label	Can be used to show non-editable field (e.g. Auto Increment) for a database table.
6 - MaskedTextBox	
7 - NumericUpDown	
8 - RichTextBox	
9 - TextEdit	
10 - TimePicker	

4.4.6 enumEditMode

An enumeration type for all edit modes.

Members

Value	Comment
1 - InRow	In row editing
2 - EditPanel	Edit a row in a panel
3 - EditPanelAndShowRow	Edit a row in a panel and show it
4 - PopupEditWindow	Edit a row in a WinForms window
5 - Batch	Edit multiple rows

4.4.7 enumElasticCordType

An enumeration type for all possible dash patterns of the elastic cord when resizing columns.

Members

Value	Comment
1 - DashedLine_1_1	dash length: 1 and space length: 1
2 - DashedLine_2_2	dash length: 2 and space length: 2
3 - DashedLine_3_3	dash length: 3 and space length: 3
4 - DashedLine_4_4	dash length: 4 and space length: 4

4.4.8 enumGridLayout

An enumeration type for all possible values of grid layout.

Members

Value	Comment
1 - Lake_Baikal	Regular rectangle layout
2 - Lake_Como	Quadrilateral layout
3 - Lake_Geneva	Curved layout with the arch pointing to the left side
4 - Lake_Saimaa	Curved layout with the arch pointing to the right side
5 - Moraine_Lake	Left arrow layout
6 - Plitvice_Lakes	Right arrow layout

4.4.9 enumLanguage

An enumeration type for all possible language options.

Members

Value
0 - Afrikaans
1 - Albanian
2 - Arabic
3 - Armenian
4 - Azerbaijani
5 - Basque
6 - Belarusian
7 - Bengali
8 - Bulgarian

9 - Catalan
10 - Chinese_Simplified
11 - Chinese_Traditional
12 - Croatian
13 - Czech
14 - Danish
15 - Dutch
16 - English
17 - Estonian
18 - Filipino
19 - Finnish
20 - French
21 - Galician
22 - German
23 - Greek
24 - Hebrew
25 - Hindi
26 - Hungarian
27 - Icelandic
28 - Indonesian
29 - Irish
30 - Italian
31 - Japanese
32 - Korean
33 - Latvian
34 - Lithuanian
35 - Macedonian
36 - Malay
37 - Maltese
38 - Norwegian
39 - Persian
40 - Polish
41 - Portuguese
42 - Romanian
43 - Russian

44 - Serbian
45 - Slovak
46 - Slovenian
47 - Spanish
48 - Swahili
49 - Swedish
50 - Thai
51 - Turkish
52 - Ukrainian
53 - Vietnamese
54 - Welsh
55 - Yiddish

5 Annotation Notes

5.1 Icons

5.1.1 General Icons

The following table contains the icons in the first tool strip in the control, and corresponding descriptions.

Icons	Descriptions
	Open annotation
	Close annotation
	Open search
	Close search
	Open cascade
	Open full grid
	Copy screenshot to clipboard
	Print screenshot
	Save screenshot to a file

5.1.2 Annotation

The following table contains the icons related to annotations, and corresponding descriptions.

Icons	Descriptions
	Remove existing annotations
	Open an annotation file
	Save annotations to a file
	Undo an operation
	Redo an operation
	Cut an annotation object
	Copy an annotation object
	Paste an annotation object
	Draw an arrow
	Draw a curve
	Display an image
	Draw a line
	Draw an oval
	Apply paint brush
	Draw a rectangle
	Display a text
	Left-align objects
	Center-align objects
	Right-align objects
	Top-align objects
	Middle-align objects
	Bottom-align objects
	Horizontally distribute objects (>=3)
	Vertically distribute objects (>= 3)
	Make objects same width
	Make objects same height
	Make objects same size

5.1.3 Search

The following table contains the icons related to search, and corresponding descriptions.

Icons	Descriptions
↑	Move to previous search match
↓	Move to next search match

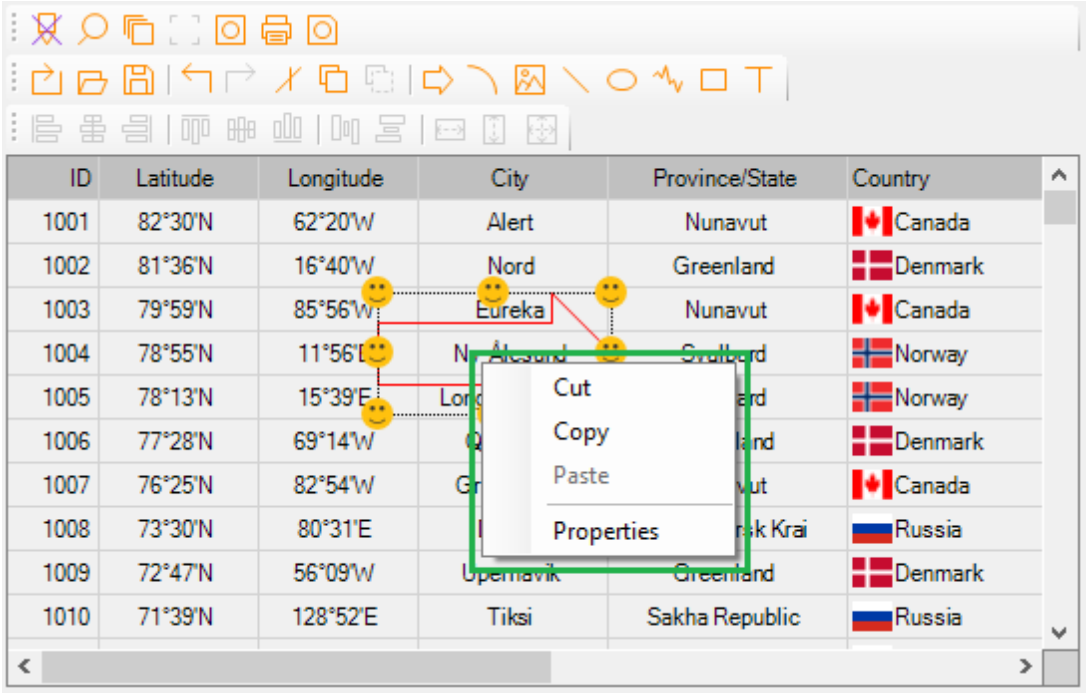
5.1.4 Cascade

The following table contains the icons related to cascade mode, and corresponding descriptions.

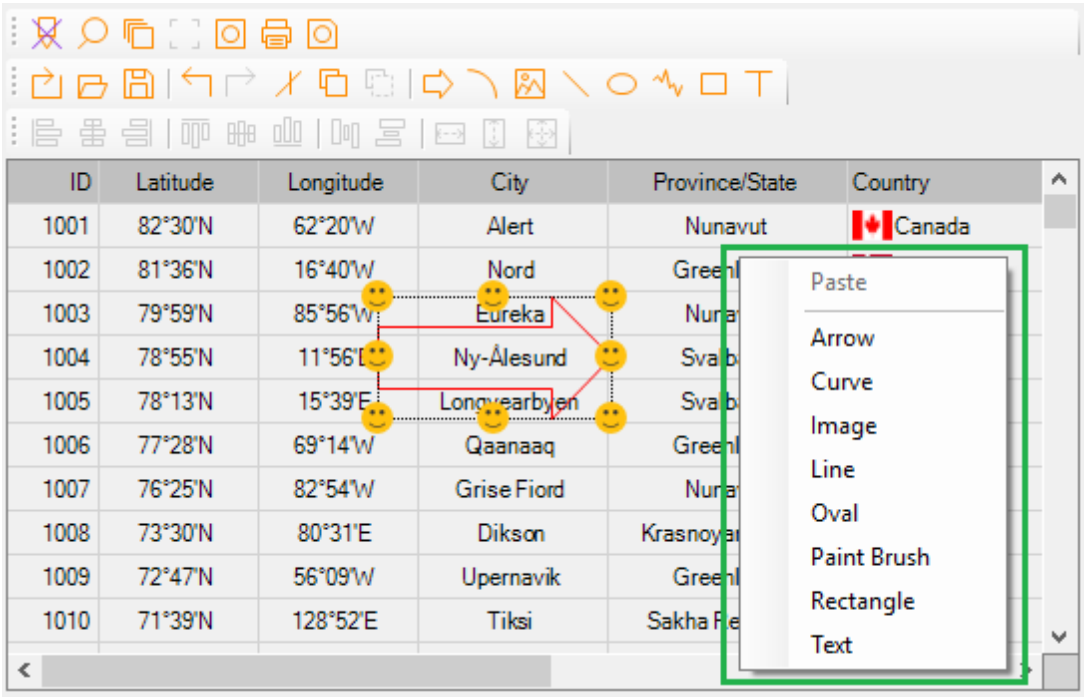
Icons	Descriptions
|<	Display 1st sub grid
<	Display previous sub grid
>	Display next sub grid
>|	Display last sub grid

5.2 Popup Menus

Right-click inside an annotation object to bring up a popup menu below, and then click Properties to bring up a Dialog to edit the properties for selected annotation object.



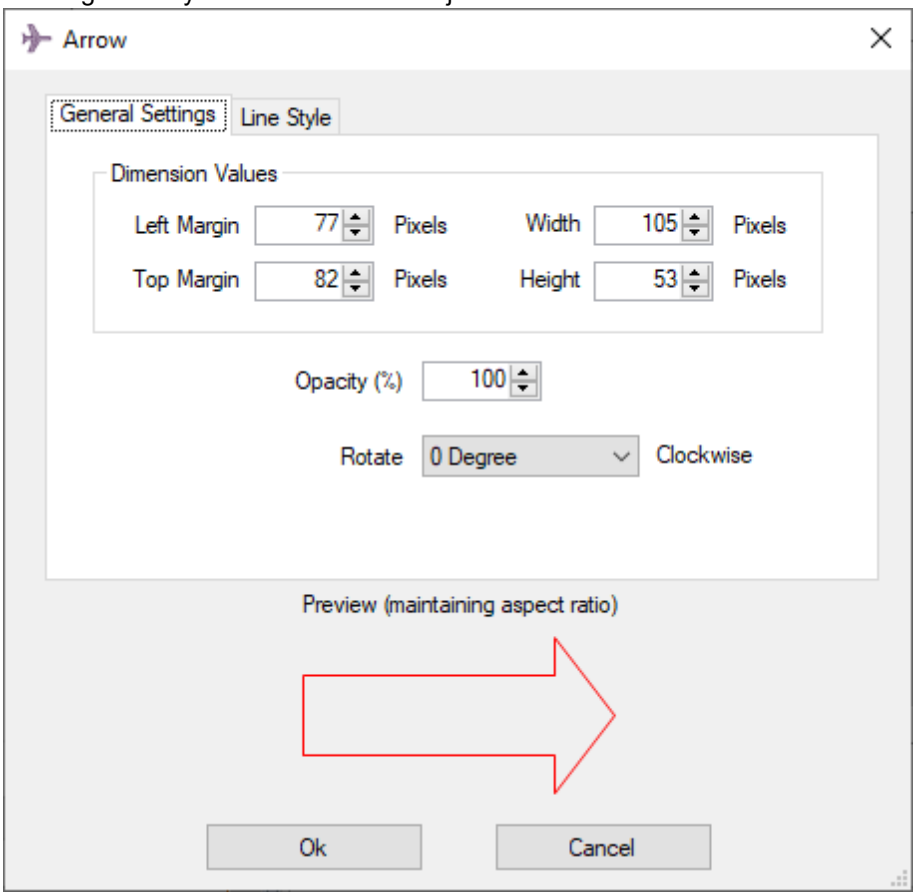
Right-click outside annotation object(s) to bring up a popup menu below and you can do a few other things here.

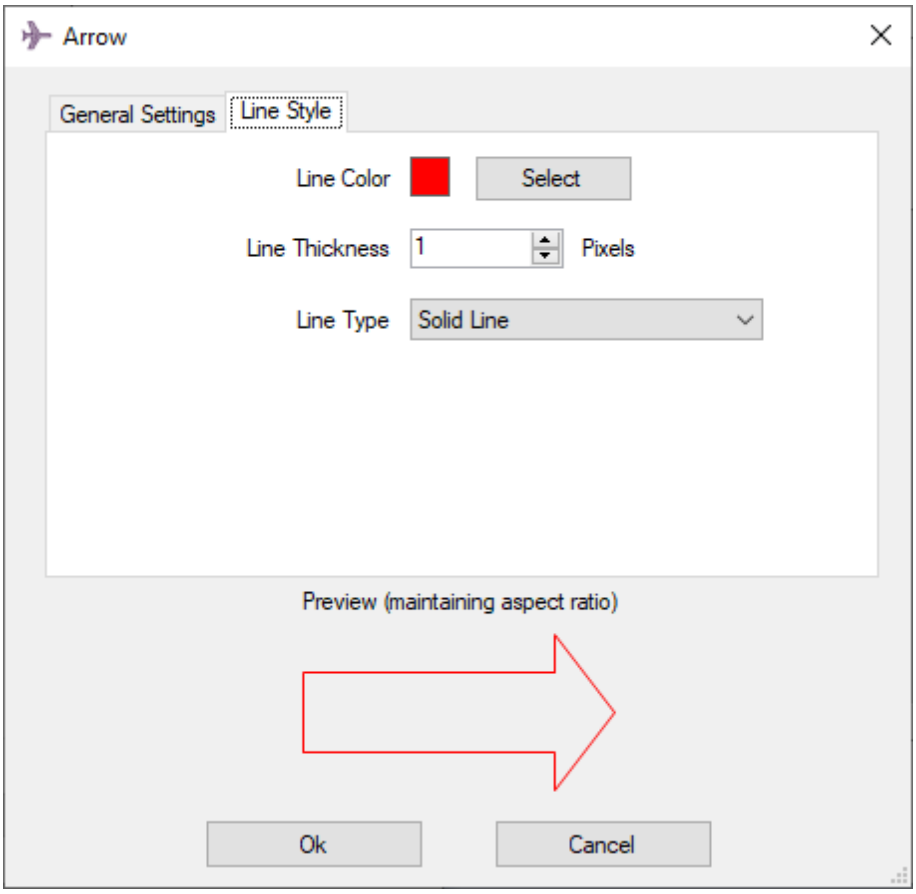


5.3 Dialogs

5.3.1 Arrow

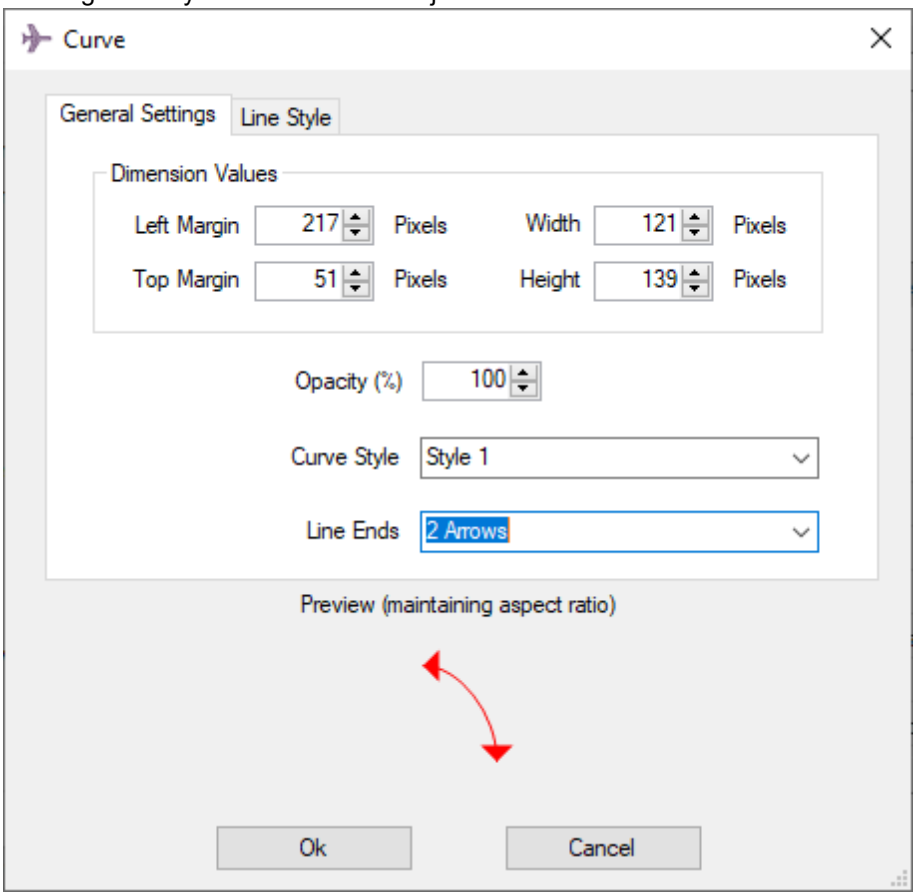
The properties dialog allows you to edit an arrow object.

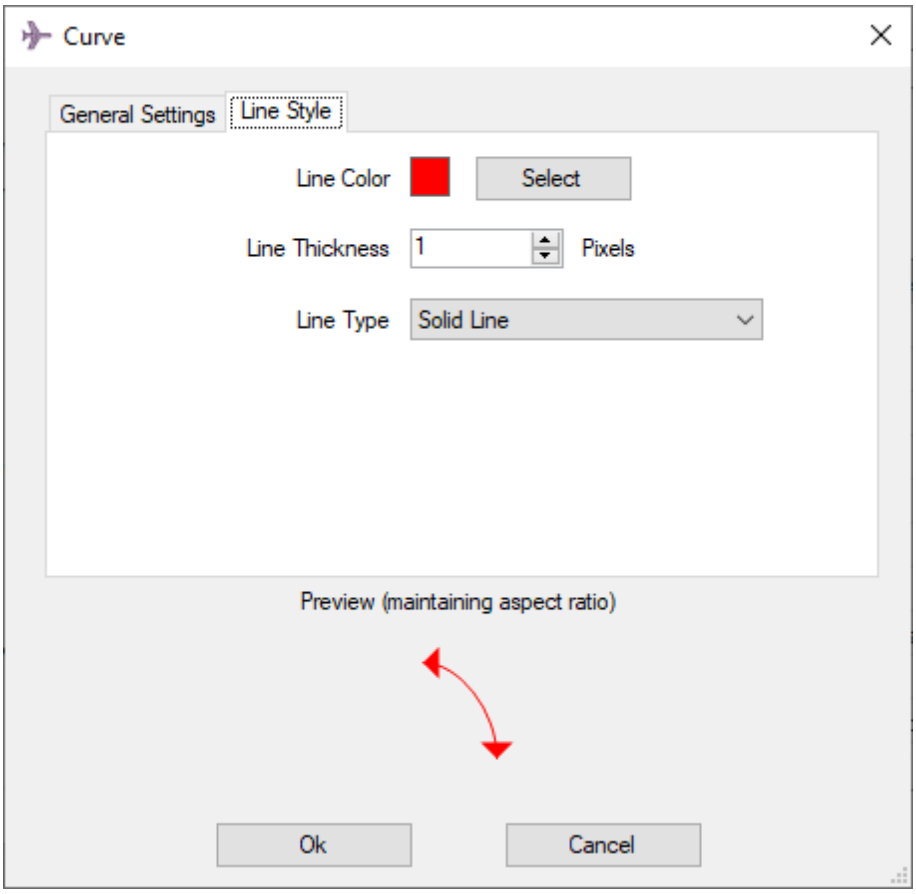




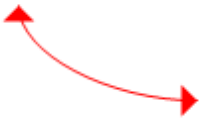
5.3.2 Curve

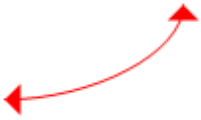
The properties dialog allows you to edit a curve object.





The following table contains all curve styles and screenshots.

Descriptions	Screenshots
Style 1	
Style 2	
Style 3	

Style 4	
Style 5	
Style 6	
Style 7	
Style 8	

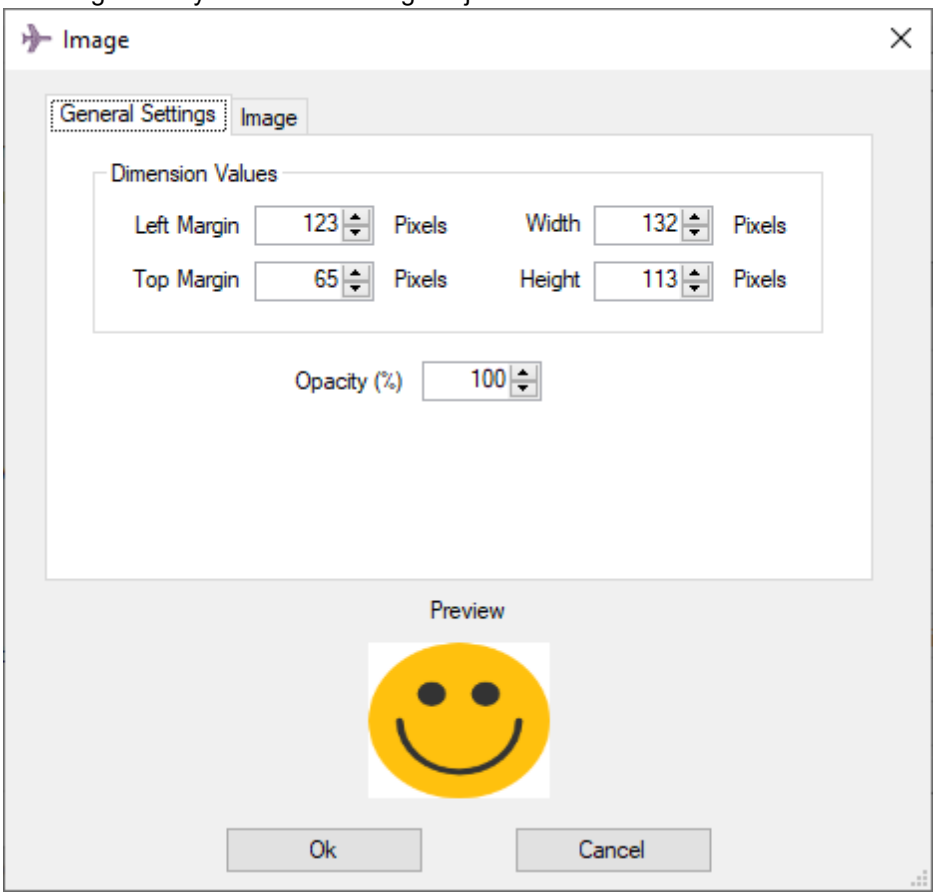
The following table contains all line ends and screenshots.

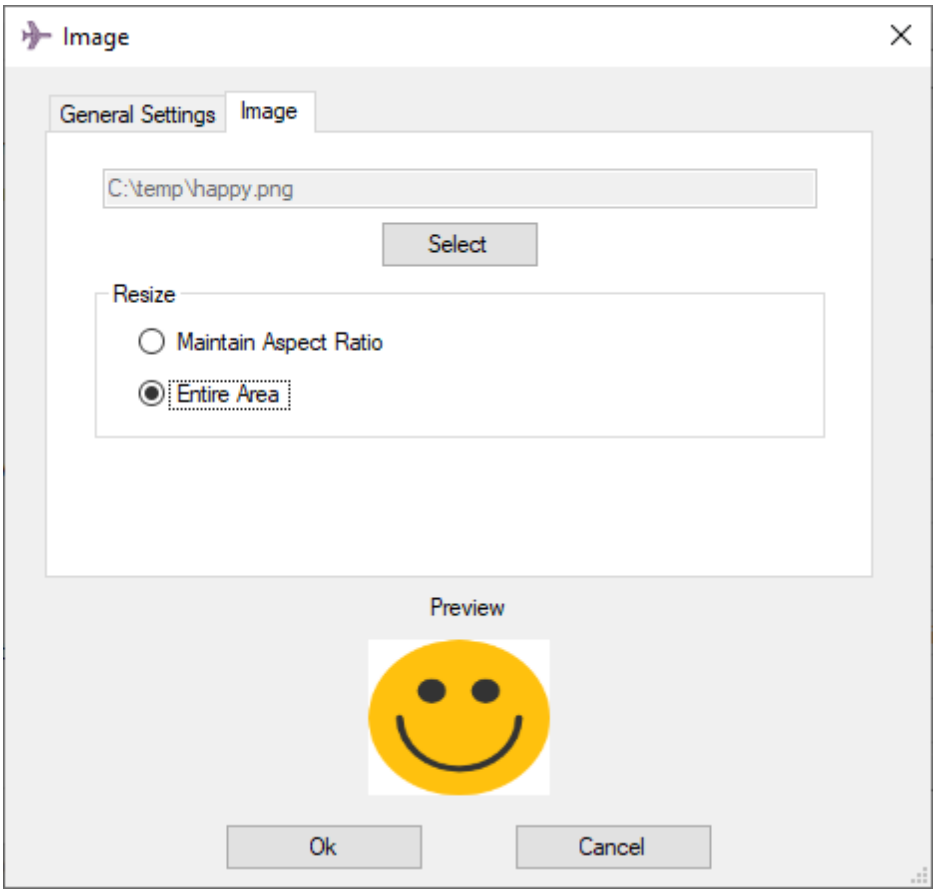
Descriptions	Screenshots
No Arrow	
1 Arrow (A)	
1 Arrow (B)	

2 Arrows	

5.3.3 Image

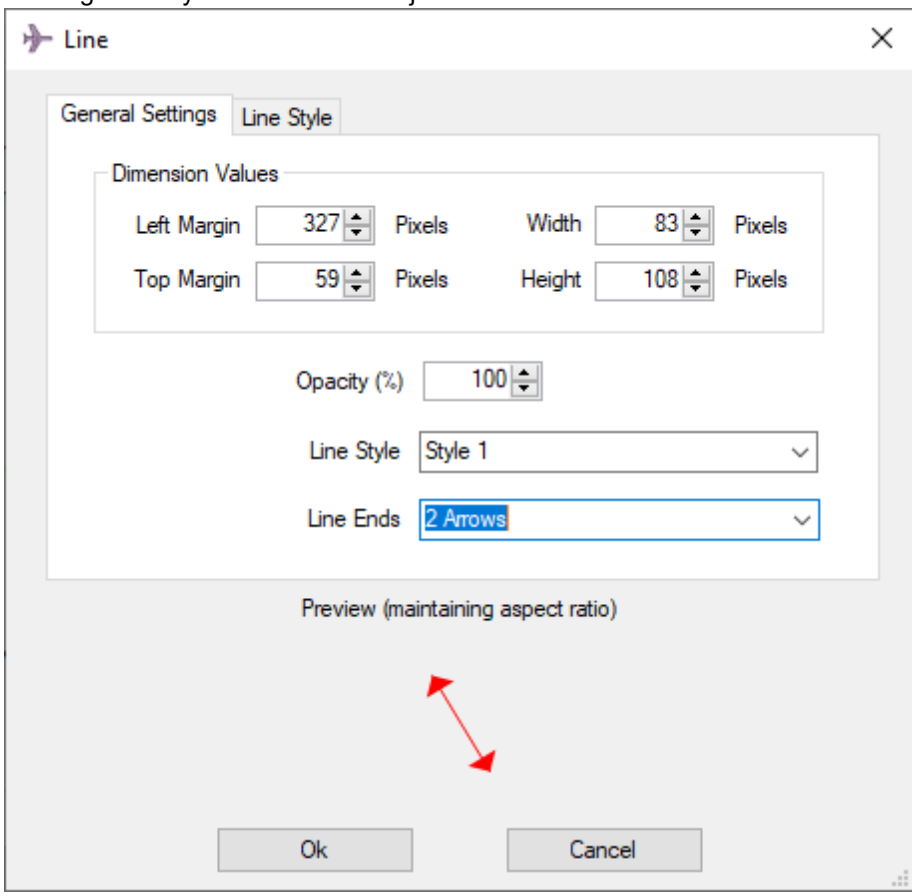
The properties dialog allows you to edit an image object.

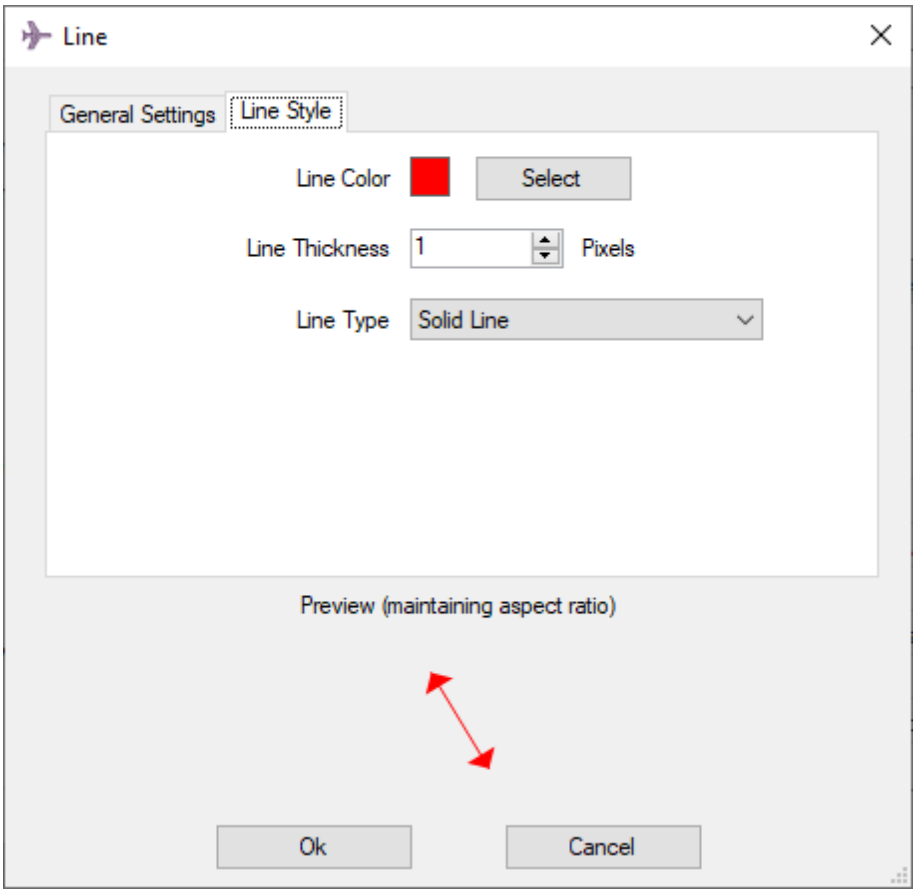




5.3.4 Line

The properties dialog allows you to edit a line object.





The following table contains all line styles and screenshots.

Descriptions	Screenshots
Style 1	
Style 2	
Style 3	

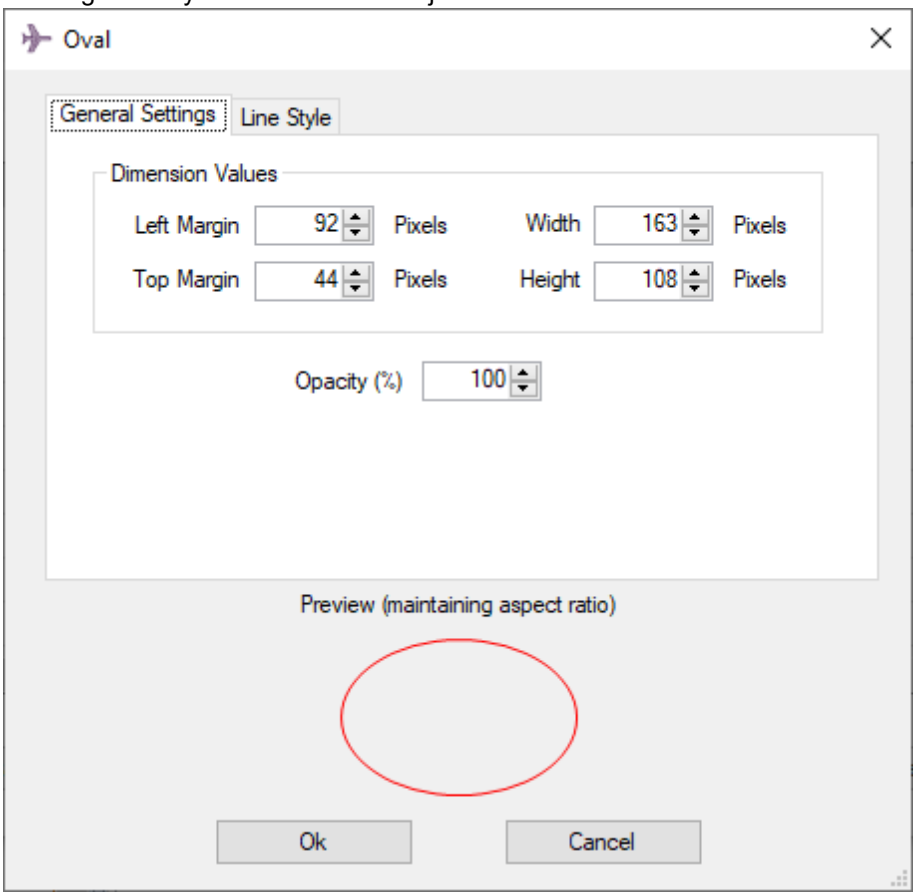
Style 4	

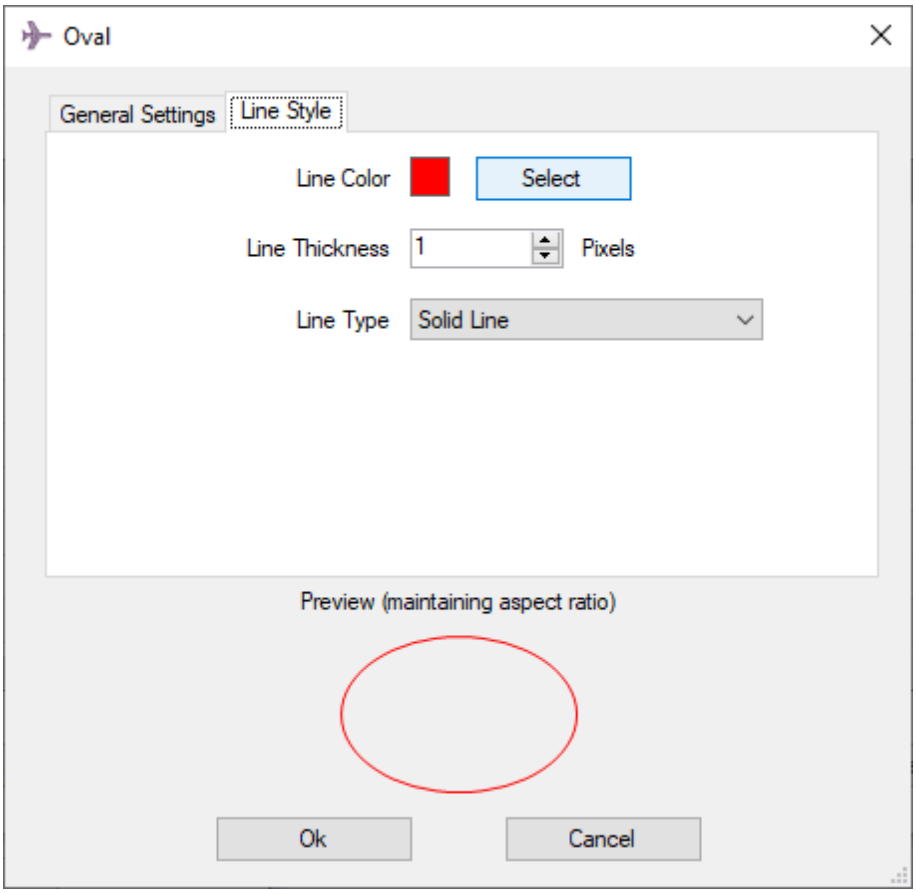
The following table contains all line ends and screenshots.

Descriptions	Screenshots
No Arrow	
1 Arrow (A)	
1 Arrow (B)	
2 Arrows	

5.3.5 Oval

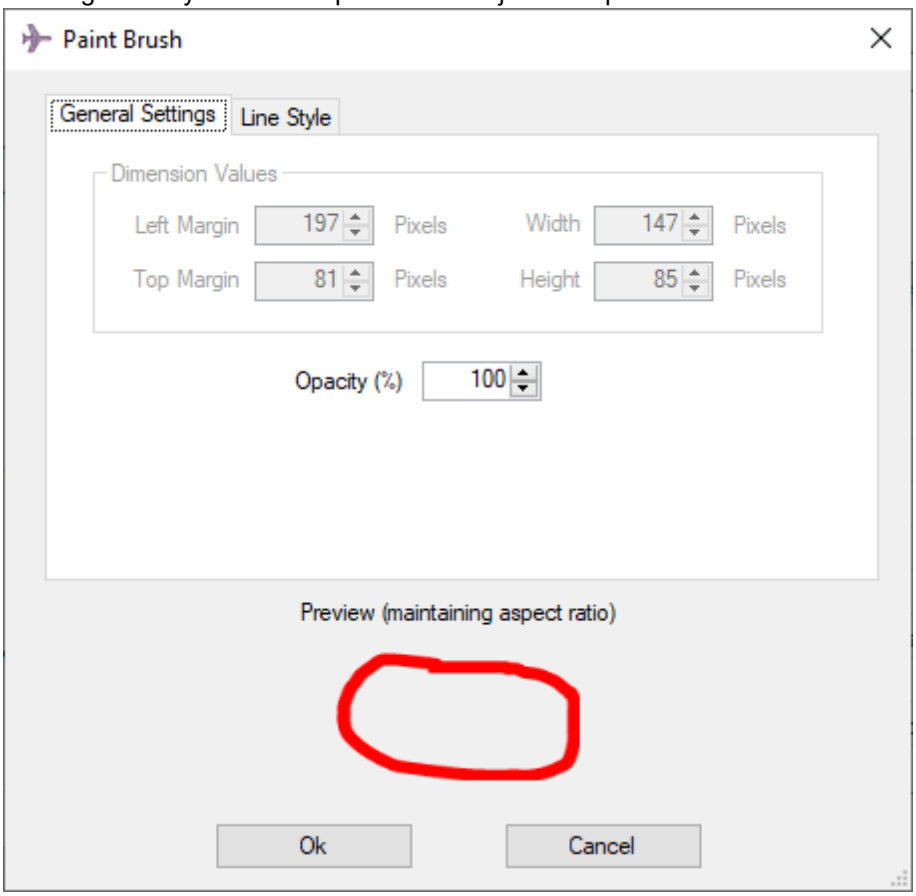
The properties dialog allows you to edit an oval object.

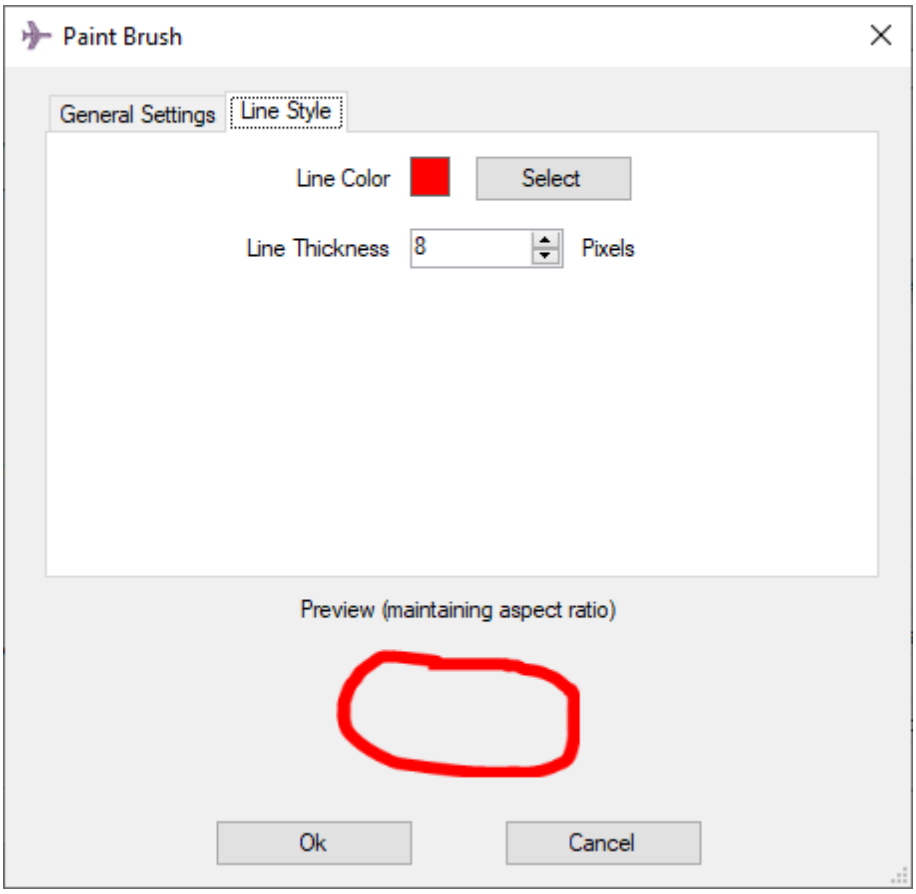




5.3.6 Paint Brush

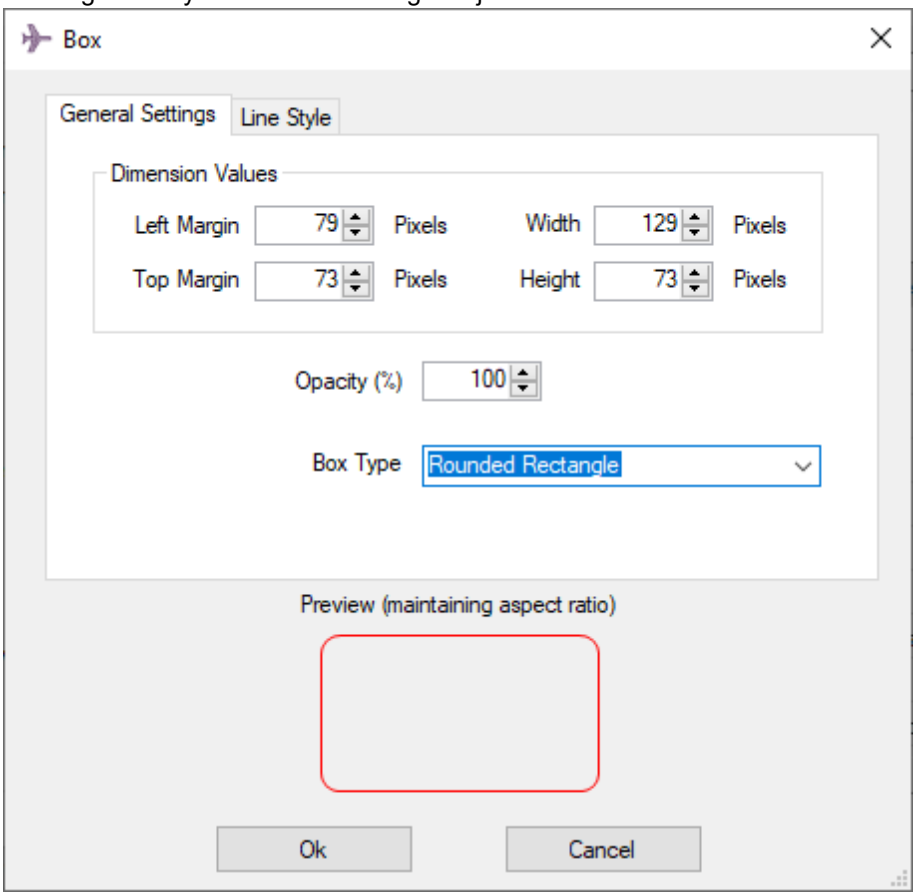
The properties dialog allows you to edit a paint brush object except dimension values.

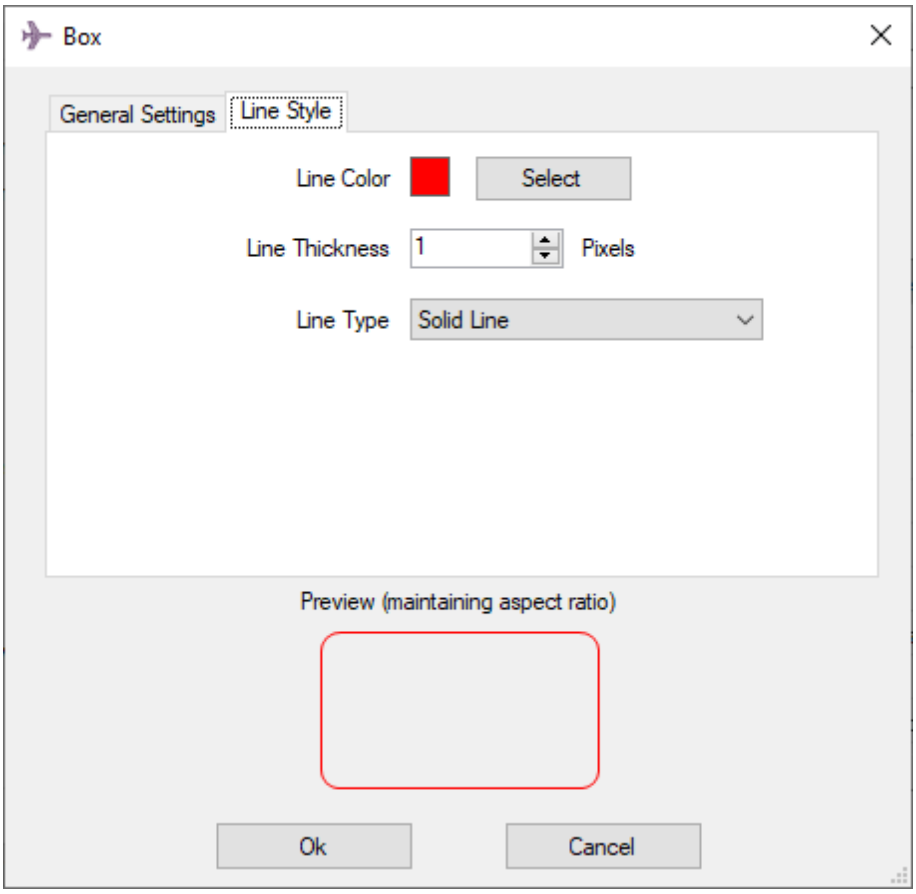




5.3.7 Rectangle

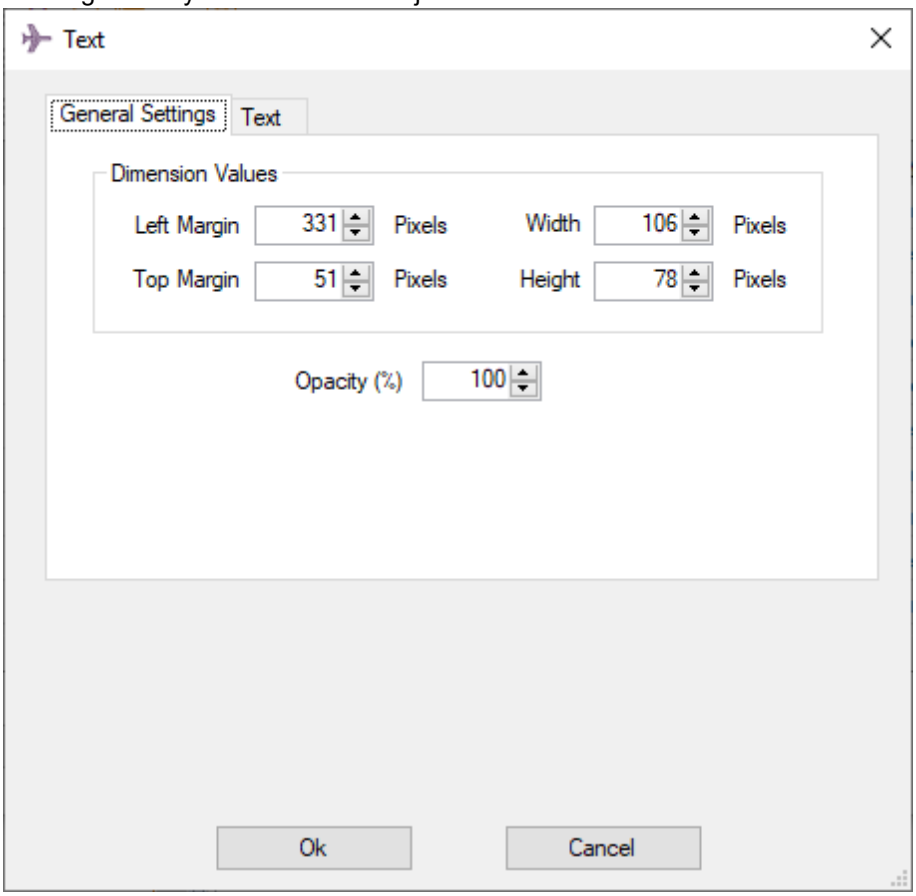
The properties dialog allows you to edit a rectangle object.

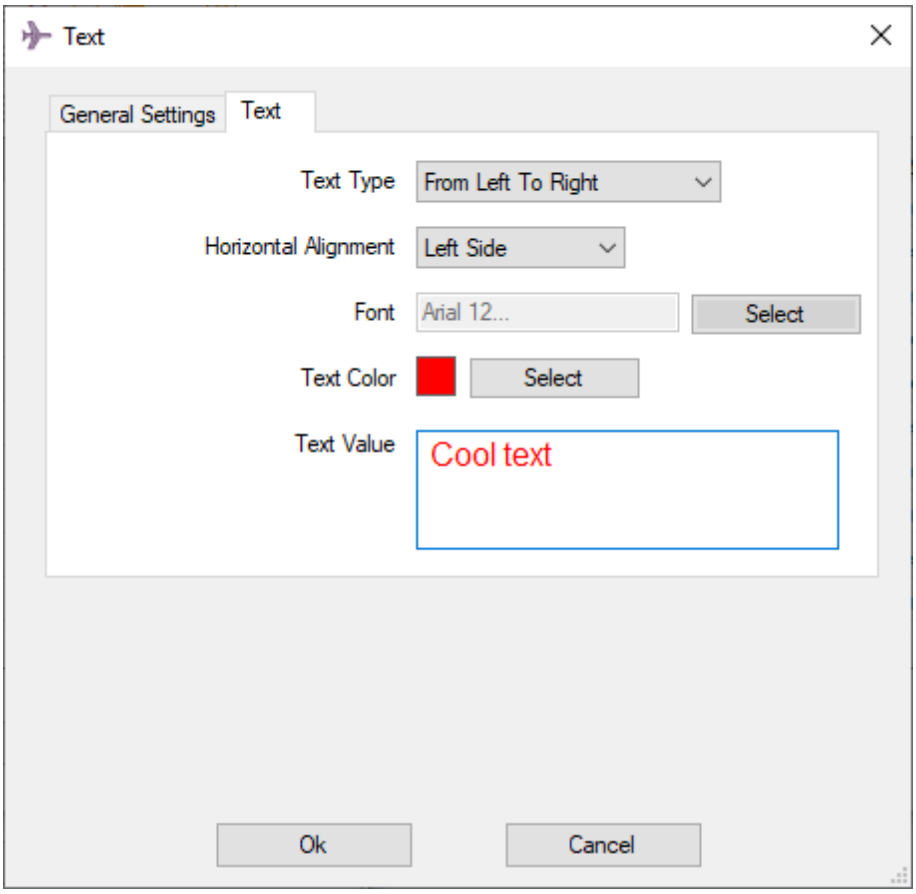




5.3.8 Text

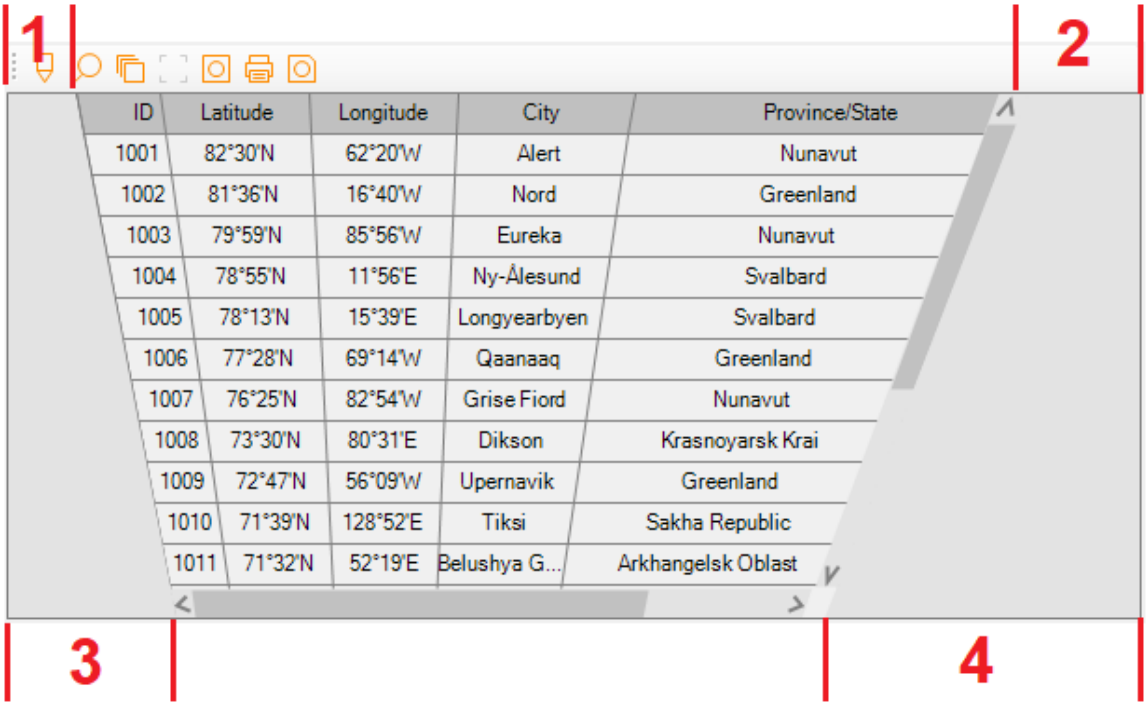
The properties dialog allows you to edit a text object.





6 Margins Notes

Four margin properties - MarginTopLeft (1), MarginTopRight (2), MarginBottomLeft (3), and MarginBottomRight (4) - determine how the layout is implemented.



For the sake of simplicity, we define a few variables below:

- Top-Sum = MarginTopLeft (1) + MarginTopRight (2)
- Bottom-Sum = MarginBottomLeft (3) + MarginBottomRight (4)
- Left-Diff = MarginTopLeft (1) - MarginBottomLeft (3)
- Right-Diff = MarginBottomRight (4) - MarginTopRight (2)

If the GridLaytout property is Lake_Baikal, all 4 margins become irrelevant and are internally set to 0.

If the GridLaytout property is Lake_Como, the following conditions must hold true.

- Both Top-Sum and Bottom-Sum can't be greater than 50.
- At least one of Top-Sum and Bottom-Sum can't be less than 2.

Otherwise, margins will be internally set to the defaults below:

- MarginTopLeft (1) = 5
- MarginTopRight (2) = 0
- MarginBottomLeft (3) = 0
- MarginBottomRight (4) = 5

If the GridLaytout property is either Lake_Geneva or Lake_Saimaa, the following conditions must hold true.

- Both Top-Sum and Bottom-Sum can't be greater than 50.
- Both Top-Sum and Bottom-Sum can't be less than 2.
- Left-Diff must be equal to Right-Diff.

Otherwise, margins will be internally set to the defaults below:

- MarginTopLeft (1) = 5
- MarginTopRight (2) = 0
- MarginBottomLeft (3) = 0
- MarginBottomRight (4) = 5

If the GridLaytout property is either Moraine_Lake or Plitvice_Lakes, the following conditions must hold true.

- Both Top-Sum and Bottom-Sum can't be greater than 50.
- Both Top-Sum and Bottom-Sum can't be less than 2.
- Both Left-Diff and Right-Diff must be equal to 0.

Otherwise, margins will be internally set to the defaults below:

- MarginTopLeft (1) = 5
- MarginTopRight (2) = 5
- MarginBottomLeft (3) = 5
- MarginBottomRight (4) = 5