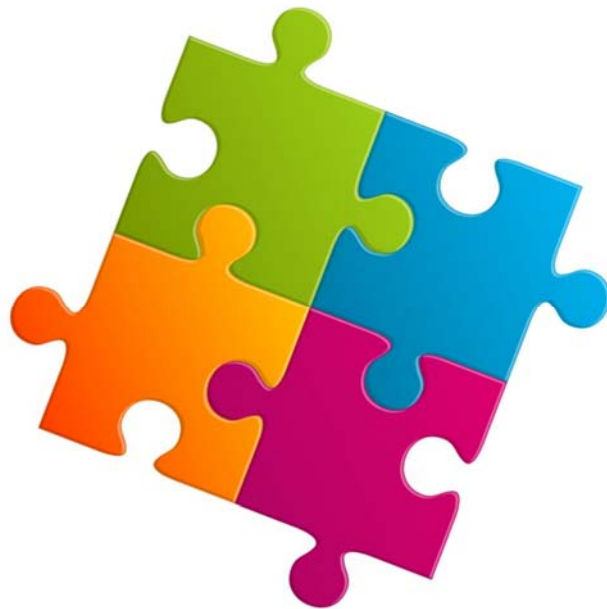


# SuperEDIT

## for Delphi VCL



**Caution**

- Without notice some or all of the contents of this document to the Company, be copied or reproduced in any form, it is strictly prohibited.
- Describing the contents of this document, but we strive to follow the revision of the software and hardware, and it causes may not be forced synchronization, please understand in advance.
- This software and documentation, can be used only under a license agreement for this product.
- MS, Microsoft, Windows is a registered trademark in the United States and other countries of United States Microsoft Corporation.  
Delphi is a trademark or registered trademarks of Embarcadero Technologies.  
Other company names are listed, product names are registered trademarks or trademarks of their respective owners.



# Contents

<b>1. With SuperEDIT .....</b>	<b>1</b>
<b>2. Feature .....</b>	<b>1</b>
<b>3. InpText / LabeledText component .....</b>	<b>2</b>
3.1 Unit .....	2
3.2 Description .....	2
3.3 Feature .....	2
3.4 Setting.....	3
3.5 Custom properties.....	4
3.6 Custom event.....	4
<b>4. InpNumber / LabeledNumber component .....</b>	<b>5</b>
4.1 Unit .....	5
4.2 Description .....	5
4.3 Feature .....	5
4.4 Setting.....	6
4.5 Custom Properties .....	8
4.6 Custom event.....	9
4.7 Custom methods.....	9
<b>5. InpDate / LabeledDate component.....</b>	<b>10</b>
5.1 Unit .....	10
5.2 Description .....	10
5.3 Feature .....	10
5.4 Setting.....	11
5.5 Custom Properties .....	12
5.6 Custom event.....	12
<b>6. InpComboBox / LabeledComboBox component .....</b>	<b>13</b>
6.1 Unit .....	13
6.2 Description .....	13
6.3 Feature .....	13
6.4 Setting.....	14
6.4 Custom Properties .....	15
6.5 Custom event.....	15
<b>7. To install the component .....</b>	<b>16</b>
7.1 Installation of components .....	16
7.2 File organization.....	18
7.3 Setting the library path .....	19
7.4 Installation of the package .....	21
7.5 For SuperEDIT dynamic link library .....	22
<b>8. To uninstall the component .....</b>	<b>23</b>
<b>9. About Message Display .....</b>	<b>24</b>

<b>10. Description of THintOut class .....</b>	<b>25</b>
10.1 Property .....	25
10.2 Property Description .....	26
HintBar .....	26
ShowHint.....	26
<b>11. Description of custom properties .....</b>	<b>27</b>
AllowCharKind.....	27
AllowEdit .....	28
AllowNullValue.....	28
AsCurrency .....	29
AsFloat .....	30
AsFurigana .....	31
AsInteger .....	32
AsPrice .....	33
AsString.....	34
AsString.....	35
AsTax .....	35
AsTotalPrice .....	36
AutoDropDown .....	36
AutoDropDown .....	37
AutoEnter .....	37
BeepOnError .....	38
ButtonStyle .....	38
ButtonType .....	39
ButtonType .....	40
CanNegative .....	41
ColorOnDisabled.....	41
ColorOnFocus .....	42
ColorOnNoFocus.....	42
ColumnCount .....	43
CursorEndField .....	43
DateFormat .....	44
Decimals .....	44
DelimiterChar .....	45
DisplayFormat .....	45
EditLabel .....	46
EllipsisHint.....	46
EllipsisShowHint.....	47
EraDate .....	47
Flat .....	48
FontOnFocus .....	49
FontOnNoFocus.....	49

FuriganaOut .....	50
FuriganaType .....	50
HintOut .....	51
InsertMode .....	52
LabelPosition .....	52
LabelSpacing .....	53
LeadingChar .....	53
ListValue .....	54
MaxLength .....	55
MaxValue .....	55
MaxValue .....	56
MinusColor .....	56
MinValue .....	57
MinValue .....	57
NextByArrowKey .....	58
PasswordChar .....	58
ReturnNext .....	59
SelectType .....	59
SelfAdjustWidth .....	60
ShowComma .....	60
Value .....	61
Value .....	61
Value .....	62
ValuePosition .....	63
Version .....	64
WidthOnFocus .....	65
ZeroAllowed .....	66
ZeroDisplay .....	66
<b>12. Description of custom event .....</b>	<b>67</b>
OnChangeValue .....	67
OnEllipsisClick .....	67
OnInsModeChanged .....	68
OnInvalidEntry .....	69
OnInvalidRange .....	70
OnPopup .....	71
<b>13. Description of custom methods / functions .....</b>	<b>72</b>
DoPopup .....	72
TaxCalc .....	73
DateSerial .....	74
GetDayCount .....	75
GetAge .....	76
GetLastDay .....	77

GetPassingDate .....	78
GetWeekLine.....	79
GetSeiza .....	80
<b>14. Use a calculator .....</b>	<b>81</b>
11.1 Start .....	81
11.2 End.....	81
11.3 Names of each part.....	81
11.4 Correspondence table of buttons and keyboard.....	82
11.5 Correction method.....	82
11.6 Error checking .....	82
<b>15. Use the drop-down calendar .....</b>	<b>83</b>
12.1 Start .....	83
12.2 End .....	83
12.3 Names of each part.....	83
12.4 Change Display month.....	83
<b>16. User support .....</b>	<b>84</b>

## 1. With SuperEDIT

Delphi is a native development environment corresponding to the development for Windows 10, Mac, mobile, IoT.

I can build the multi-device application that I can connect to a variety of systems and devices.

The high development efficiency by the object oriented using an execution speed in particular and refined various standard components,

I do not allow other tools to come near.

However, if even how superior tool is various when have been developing application for many years; dissatisfaction and the thing that want to improve

It occurs. Fortunately, Delphi can use the high component technology in Object Pascal.

For the purpose of Extensions largely performing improvement of a pro-input of the Delphi standard component, and SuperEDIT raising development efficiency more

It is a developed component set.

## 2. Feature

SuperEDIT is comprised of eight kinds of components realizing refined input environment.

Class name	Component name
TInpText	Text input component
TInpNumber	Numerical value input component
TInpDate	Date input component
TInpComboBox	Expansion ComboBox component
TLabeledText	Text input component with the label
TLabeledNumber	Numerical value input component with the label
TLabeledDate	Date input component with the label
TLabeledComboBox	Expansion ComboBox component with the label



## 3. InpText / LabeledText component

### 3.1 Unit

Vcl.ASCtrl

### 3.2 Description

There is "an automatic furigana function" that the InpText/LabeledText component (the following, text input component) only inputs "the letter class-limited input" that only a specific letter class accepts and kanji data and generates furigana automatically.

Here, we will describe the basic settings and features of the text input component.

### 3.3 Feature

- **Character type limited input**

Text input components, you can easily specify the type of character to be allowed to enter the property.

In this property, kanji, hiragana, katakana, alphabet, numbers, from the symbol, you can specify the type of character that allows input.

In addition, kanji, in all of the character types, except the Hiragana, it can also be distinguished from full-width and half-width.

- **Automatic check function of the input character**

Text input component checks the entered characters, and automatically eliminates a character other than the specified character type.

For this reason, you can greatly reduce the tedious coding for prevent typing errors in the end-user.

- **Automatic furigana function**

By the input such as names or addresses, I often attach furigana to kanji data for later data searches.

Because there is a function to generate furigana automatically just to input kanji data to the text input component, A burden of the input work of the end user is reduced drastically.

In addition, the letter class of furigana can appoint full size katakana, a hiragana letter other than half size katakana and can send furigana which I acquired to the different input component.

- **High formatting features degree of freedom**

Text input component can appoint the acquisition of the focus, a color and a font of the control at the time of the loss.

Therefore, because can judge an input position intuitively; at the time of the input of the end user can relieve stress.

- **Guidance display function**

Text input component, when the focus of the acquisition, will be able to display the Hint string in the status bar.

Therefore, since the coding for displaying the input guidance is not necessary, it can be reduced man-hours.

- **Extension of the movement keys**

Text input component can move the control with arrow key, the Return key other than the TAB key of the Windows standard.

- **Setting function of the input mode (insert / overwrite)**

Text input component, you can specify the input mode at the time of start-up in the property. Also, by pressing the Insert key at run time, you can also switch the input mode.

- **Setting function of focus at acquisition of control width**

Text input component receives the focus is extended in width is automatically of control, the contents of the long string that does not fit in the control be easily input, you can display.

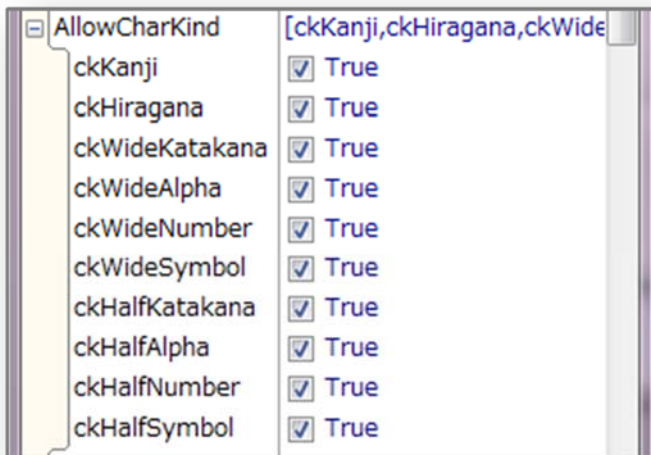
- **Input cancel function**

If it is before the movement of the controls, you can return the value of the previous input by pressing the [Esc] (Cancel key).

### 3.4 Setting

- **Setting of the input character type**

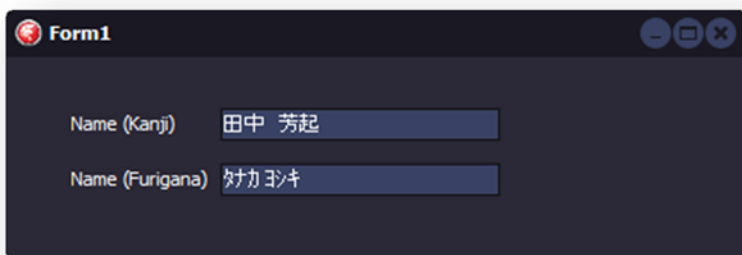
Input character type of text input component, it will set in **AllowCharKind** property of the object inspector.



- **Setting of the automatic furigana function**

Text input component by setting the **FuriganaOut** property, can read input using a kanji conversion system (IME) outputs the name to another control.

In addition, you can also get a phonetic using **AsFurigana** property.



### 3.5 Custom properties

Property name	Description	Page
AllowCharKind	It will specify the type of character that allows input	27
AsFurigana	Get the furigana <b>(read-only)</b>	31
AutoEnter	Specifies whether to perform a focus move in automatic	37
BeepOnError	Specifies whether sound an error warning sound	38
ButtonStyle	It will specify the display format of the button	38
ButtonType	It will specify the button of shape	39
ColorOnDisabled	Enabled property specifies the color at the time of the False	41
ColorOnFocus	Control you will specify the color of when it receives the input focus	42
ColorOnNoFocus	Control you will specify the color of the time that has lost the input focus	42
CursorEndField	Specify the position of the cursor when it receives the focus	43
EditLabel	It will specify the label information of the labeled control	46
EllipsisHint	It will specify the hint string of search button	46
EllipsisShowHint	Specifies whether to display the search button Tips	47
FontOnFocus	Control specify the font of when it receives the input focus	49
FontOnNoFocus	Control specify the font of the time that has lost the input focus	49
FuriganaOut	Specifies the control outputs the acquired furigana	50
FuriganaType	It will specify the character type of furigana output	50
HintOut	It will specify the status bar and display position to display a hint	51
InsertMode	It specifies the input mode when it receives the input focus	52
LabelPosition	It will specify the position of the label	52
LabelSpacing	It will specify the spacing of the input control and label	53
MaxLength	It will specify the maximum number of characters that can be entered	55
NextByArrowKey	Specifies whether to allow the control movement by the arrow keys	58
PasswordChar	Indicates the character, if any, to display in place of the actual characters typed in the control.	58
ReturnNext	Specifies whether to allow the control movement by the Enter key	59
Version	SuperEDIT version information is displayed <b>(read-only)</b>	64
WidthOnFocus	Specifies the width of the control when it receives the focus	65

### 3.6 Custom event

Event name	Description	Page
OnChangeValue	It will occur when the value is changed	67
OnEllipsisClick	Search button I happens is pushed	67
OnInsModeChanged	It occurs when it is input mode change	68
OnInvalidEntry	This error occurs when the wrong key is pushed	69

## 4. InpNumber / LabeledNumber component

### 4.1 Unit

Vcl.ASCtrls

### 4.2 Description

InpNumber / LabeledNumber component (the following, Numerical value input component) provides a positive or negative number and the amount of input interface.

Here, we'll explain the features and basic set of numeric input component.

### 4.3 Feature

- **High formatting features degree of freedom**

Numeric input component, currency symbol (¥, \$) and display the presence or absence of scale comma, display color of a negative value, can be used to specify the display format, focus control of colors and fonts at the time of acquisition and loss of time of zero you.

- **Automatic check function of the input character**

Numeric input component, single-byte numbers and - (minus) and the specified point symbol (".") The default value is), so has the ability to eliminate the character input other than to check the validity of the numerical input in the code There is no need.

In addition, you can also limit the numeric input that exceeds a preset upper limit and lower limit.

- **Calculation of consumption tax**

In the numeric input component, using **TaxCalc** methods and **AsTax**, **AsPrice**, the **AsTotalPrice** property, you can easily calculate the consumption tax of tax and outside tax.

- **Drop-down calculator / pop-up calculator**

Numeric input component, you can easily use the drop-down calculator with the specified property.

In addition, also provides pop-up calculator call methods that you can freely specify the call keys and display position.

Thus the user is able to perform using a calculator simple calculations in the input can also be passed to the input component results.

- **Guidance display function**

Numeric input component, when the focus of the acquisition, will be able to display the Hint string in the status bar.

Therefore, since the coding for displaying the input guidance is not necessary, it can be reduced man-hours.

- **Setting and taken out of value according to various data types**

Numeric input component, you can value setting and retrieving the following four types of data types.

- Double values(**Value** property)or(**AsFloat** property)
- Text values(**AsString** property)
- Integer values(**AsInteger** property)
- Currency values(**AsCurrency** property)

- **Extension of the movement keys**

Text input component can move the control with arrow key, the Return key other than the TAB key of the Windows standard.

- **Input cancel function**

If it is before the movement of the controls, you can return the value of the previous input by pressing the [Esc] (Cancel key).

## 4.4 Setting

- **Setting of the input format**

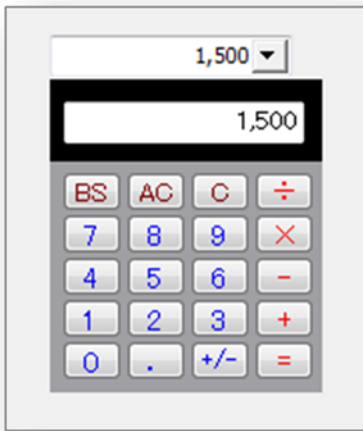
The input format of the numeric input component, set the following properties in the Object Inspector.

- **CanNegative** property
- **ColorOnFocus** property
- **ColorOnNoFocus** property
- **Decimals** property
- **FontOnFocus** property
- **FontOnNoFocus** property
- **LeadingChar** property
- **MinusColor** property
- **ShowComma** property
- **ZeroDisplay** property

- **Setting of the drop-down calculator**

If you click the drop blanking down button at run time, numeric input component included calculator appears at the bottom of the control.

The **ButtonStyle** property to **bsOnFocus** or **bsAlways** To display a drop-down button, and set the **ButtonType** property to **btDropdown**.



- **Calculation of consumption tax**

Numeric input component, using **TaxCalc** methods and **AsTax**, **AsPrice**, three properties of **AsTotalPrice**, you can easily calculate the consumption tax of tax and outside tax. Below, price ¥ 1000, example shows how to calculate the tax and outside tax at the time of the consumption tax rate to 5%.

**[In the case of tax included]**

- TaxCalc methods

```
InpNumber1.TaxCalc(5, 0, 0); // Consumption tax rate 5%, tax included, rounding
```

- Calculation result

```
AsTax      : 48    // consumption tax
AsPrice    : 952   // No tax price
AsTotalPrice : 1000 // Tax-inclusive price
```

**[In the case of Outside tax]**

- TaxCalc methods

```
InpNumber1.TaxCalc(5, 1, 0); // Consumption tax rate 5%, Outside tax, rounding
```

- Calculation result

```
AsTax      : 50    // consumption tax
AsPrice    : 1000  // No tax price
AsTotalPrice : 1050 // Tax-inclusive price
```

## 4.5 Custom Properties

Property name	Description	Page
AsCurrency	Currency values by setting the value, it will retrieve	29
AsFloat	Double values by setting the value, it will retrieve	30
AsInteger	Integer values by setting the value, it will retrieve	32
AsPrice	Use the TaxCalc method Retrieves the "price" <b>(read only)</b>	33
AsString	String values by setting the value, it will retrieve	34
AsTax	Use the TaxCalc method Retrieves the "consumption tax" <b>(read only)</b>	35
AsTotalPrice	Use the TaxCalc method Retrieves the "tax price" <b>(read only)</b>	36
BeepOnError	Specifies whether sound an error warning sound	38
ButtonStyle	It will specify the display format of the button	38
ButtonType	It will specify the button of shape	40
CanNegative	It limits negative number input	41
ColorOnDisabled	Enabled property specifies the color at the time of the False	41
ColorOnFocus	Control you will specify the color of when it receives the input focus	42
ColorOnNoFocus	Control you will specify the color of the time that has lost the input focus	42
Decimals	It will specify the number of digits in the fractional part of the input numerical value	44
EditLabel	It will specify the label information of the labeled control	46
FontOnFocus	Control specify the font of when it receives the input focus	49
FontOnNoFocus	Control specify the font of the time that has lost the input focus	49
HintOut	It will specify the status bar and display position to display a hint	51
LabelPosition	It will specify the position of the label	52
LabelSpacing	It will specify the spacing of the input control and label	53
LeadingChar	It will specify the currency symbol	53
MaxValue	It will specify the maximum value of a number	55
MinusColor	The number is I will specify the text color at the time of the negative	56
MinValue	It will specify the minimum value of a number	57
NextByArrowKey	Specifies whether to allow the control movement by the arrow keys	58
ReturnNext	Specifies whether to allow the control movement by the Enter key	59
ShowComma	It will specify the display format of the scale comma	60
Value	You will a numeric value	61
Version	SuperEDIT version information is displayed <b>(read-only)</b>	64
ZeroAllowed	Specifies or not to allow a zero input	66
ZeroDisplay	Specify which one to the display of the 0	66

## 4.6 Custom event

Event name	Description	Page
OnChangeValue	It will occur when the value is changed	67
OnEllipsisClick	Search button I happens is pushed	67
OnInvalidEntry	This error occurs when the wrong key is pushed	69
OnInvalidRange	It occurs when the out-of-range value input, or you are trying to set	70
OnPopup	When you press the drop button, or it occurs when to display the pop-up calculator	71

## 4.7 Custom methods

Method name	Description	Page
DoPopup	It will display a pop-up calculator to the specified position	72
TaxCalc	It will calculate the amount of tax	73



## 5. InpDate / LabeledDate component

### 5.1 Unit

Vcl.ASCtrlis

### 5.2 Description

InpDate / LabeledDate component (the following, date input component) provides a date input interface that corresponds to the year.

Here, we will describe the basic settings and features of the date input component.

### 5.3 Feature

- **High formatting features degree of freedom**

The date input component can appoint the acquisition of the focus, a color and a font of the control at the time of the loss.

Therefore, because can judge an input position intuitively; at the time of the input of the end user can relieve stress.

Moreover, by combining the DisplayFormat property type is divided display format, it can be easily implemented by specifying the properties of the display of complex date.

- **Automatic check function of the input character**

Date input component, single-byte numerical value as the specified date delimiter (the default value is "/") because there is a function to eliminate the character input other than, you do not need to check in the code the validity of the date input.

In addition, you can also limit the input of a date that exceeds the preset upper limit and lower limit.

- **The drop-down calendar**

Date input component, you can easily use the drop-down calendar in the specified property.

- **Guidance display function**

Date input component, when the focus of the acquisition, will be able to display the Hint string in the status bar.

Therefore, since the coding for displaying the input guidance is not necessary, it can be reduced man-hours.

- **Selection range specification function of AutoSelect**

Date input component, you can specify the selection of text when the focus of the acquisition.

- **Extension of the movement keys**

Date input component can move the control with arrow key, the Return key other than the TAB key of the Windows standard.

- **Input cancel function**

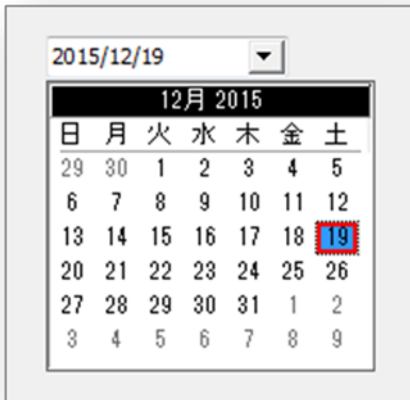
If it is before the movement of the controls, you can return the value of the previous input by pressing the [Esc] (Cancel key).

## 5.4 Setting

- **Setting of the drop-down calendar**

If you click at the time of the drop blanking down button execution, calendar accessory date input component is displayed in the bottom of the control.

To display a drop-down button, set the **ButtonStyle** property to **bsOnFocus** or **bsAlways**.



- **Input format and display format**

Date input component, but we have the basic AD input, it can be used to divide the display format as the input using the **DisplayFormat** property. You can also display the date of the input year in the Japanese calendar.

Also, if you set the **DateFormat** property, you can set the input format that ignores the Windows date setting.



## 5.5 Custom Properties

Property name	Description	Page
AllowNullValue	Sets whether to allow blank input	28
AsString	String values by setting the value, it will retrieve	35
AutoDropDown	Automatically Specify whether you want to display a drop-down calendar	36
BeepOnError	Specifies whether sound an error warning sound	38
ButtonStyle	It will specify the display format of the button	38
ColorOnDisabled	Enabled property specifies the color at the time of the False	41
ColorOnFocus	Control you will specify the color of when it receives the input focus	42
ColorOnNoFocus	Control you will specify the color of the time that has lost the input focus	42
DateFormat	It will specify the input format of date	44
DisplayFormat	It will specify the display format of date	45
EditLabel	It will specify the label information of the labeled control	46
EraDate	Retrieves the date of the Japanese calendar <b>(read only)</b>	47
FontOnFocus	Control specify the font of when it receives the input focus	49
FontOnNoFocus	Control specify the font of the time that has lost the input focus	49
HintOut	It will specify the status bar and display position to display a hint	51
LabelPosition	It will specify the position of the label	52
LabelSpacing	It will specify the spacing of the input control and label	53
MaxValue	You set the maximum value of the date that can be entered	56
MinValue	It sets the minimum value of the date that can be entered	57
NextByArrowKey	Specifies whether to allow the control movement by the arrow keys	58
ReturnNext	Specifies whether to allow the control movement by the Enter key	59
SelectType	It will specify the selection range of text at the time of the focus acquisition	59
Value	You set the date	61
Version	SuperEDIT version information is displayed <b>(read-only)</b>	64

## 5.6 Custom event

Event name	Description	Page
OnChangeValue	It will occur when the value is changed	67
OnInvalidEntry	This error occurs when the wrong key is pushed	69
OnInvalidRange	It occurs when a date out of range input, or you are trying to set	70

## 6. InpComboBox / LabeledComboBox component

### 6.1 Unit

Vcl.ASCtrls

### 6.2 Description

InpComboBox / LabeledComboBox component (hereinafter, extended ComboBox component), and extends the capabilities of Delphi standard ComboBox component, it has been easier to use.

Here, we will describe the basic settings and features of the extended combo box component.

### 6.3 Feature

- **Multi-column display of the drop-down list**

Expanded ComboBox component, you can display the data of a plurality of columns in the drop-down list.

Also, you can omit the troublesome coding because the column to be displayed in the edit box can also be specified in the properties.

- **Acquisition function of any string of drop-down list**

Expanded ComboBox component, you can retrieve any string from the items that have been selected in the drop-down list.

- **Incremental search function**

Expanded ComboBox component, against the data of the input data and the drop-down list, incremental search function is equipped to automatically display the candidate.

- **Guidance display function**

Expanded ComboBox component, when the focus of the acquisition, will be able to display the Hint string in the status bar. Therefore, since the coding for displaying the input guidance is not necessary, it can be reduced man-hours.

- **High formatting features degree of freedom**

Expanded ComboBox component can appoint the acquisition of the focus, a color and a font of the control at the time of the loss.

Therefore, because can judge an input position intuitively; at the time of the input of the end user can relieve stress.

- **Extension of the movement keys**

Expanded ComboBox component can move the control with arrow key, the Return key other than the TAB key of the Windows standard.

- **It automatically adjusts the width of the drop-down list**

Expanded combo box component, you can automatically adjust the width of the drop-down list on the maximum length of the field.

- **Setting function of focus at acquisition of control width**

Expanded ComboBox component, receives the focus is extended to the width is automatically of control, the contents of the long string that does not fit in the control be easily input, you can display.

- **Input cancel function**

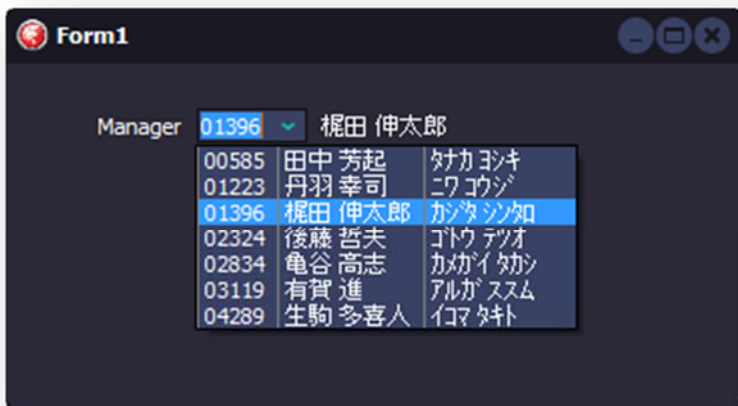
If it is before the movement of the controls, you can return the value of the previous input by pressing the [Esc] (Cancel key).

## 6.4 Setting

- **Multi-column display of the drop-down list**

Items to be displayed in the drop-down list to set **ListValue** property, **ColumnCount** property the number of columns, the delimiter of columns in **DelimiterChar** property.

Set the column, which will be displayed in the edit box to **ValuePosition** property. The value of the selected item in the column, you can remove it using the **Value** property.



## 6.4 Custom Properties

Property name	Description	Page
AllowEdit	Sets whether to allow input	28
AutoDropDown	Automatically Specify whether you want to display a drop-down list	37
ColorOnDisabled	Enabled property specifies the color at the time of the False	41
ColorOnFocus	Control you will specify the color of when it receives the input focus	42
ColorOnNoFocus	Control you will specify the color of the time that has lost the input focus	42
ColumnCount	Specify the number of columns to be displayed in the drop-down list	43
DelimiterChar	You will specify a drop-down list item delimiter	45
EditLabel	It will specify the label information of the labeled control	46
Flat	It will specify the appearance of control	48
HintOut	It will specify the status bar and display position to display a hint	51
LabelPosition	It will specify the position of the label	52
LabelSpacing	It will specify the spacing of the input control and label	53
ListValue	It will specify the item in the drop-down list	54
MaxLength	It will specify the maximum number of characters that can be entered	55
NextByArrowKey	Specifies whether to allow the control movement by the arrow keys	58
ReturnNext	Specifies whether to allow the control movement by the Enter key	59
SelfAdjustWidth	Specifies whether to automatically adjust the width of the drop-down list	60
Value	Retrieves the value of a column of the selected drop-down list ( <b>read only</b> )	62
ValuePosition	When the drop-down list selection, you will specify the column number to be displayed in the edit box	63
Version	SuperEDIT version information is displayed ( <b>read-only</b> )	64
WidthOnFocus	Specifies the width of the control when it receives the focus	65

## 6.5 Custom event

Event name	Description	Page
OnChangeValue	It will occur when the value is changed	67

## 7. To install the component

SuperEDIT component (hereinafter referred to as components) in order to use, please install the components in Delphi development environment in accordance with the following procedure.

- Installation of components
- Setting the library path
- Installation of the package

### 7.1 Installation of components

To use a component, you will need to copy the component to the hard disk.

This section describes how to install the components.

#### ● To end the Delphi

All Delphi it has started to exit.

#### ● To start the setup program

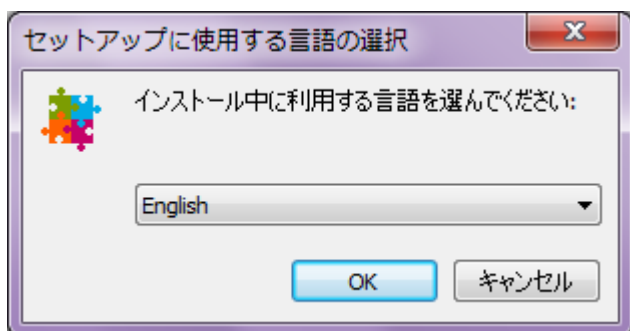
Start the downloaded "setup.exe".

If, when you see the screen of the [User Account Control], and then click [Yes].

#### ● Select the language you want to use to set up

Please select a language to use during the installation.

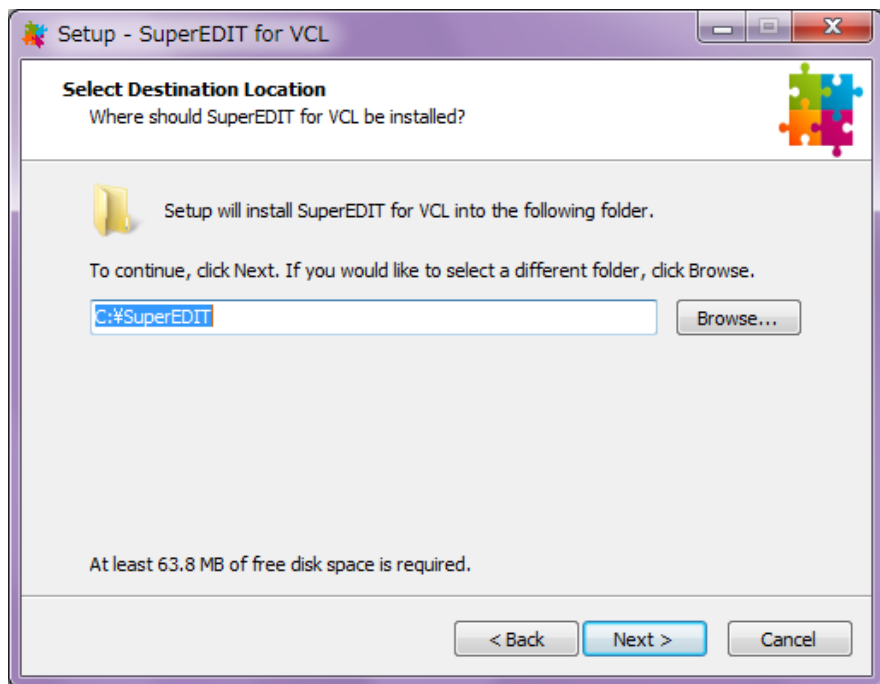
Here, select the "English" and click the [OK] button.



#### ● To verify the license agreement

For this software, be asked to accept the "License Agreement" has become a condition of your use. In advance the contents well on the check, please use only if you agree.

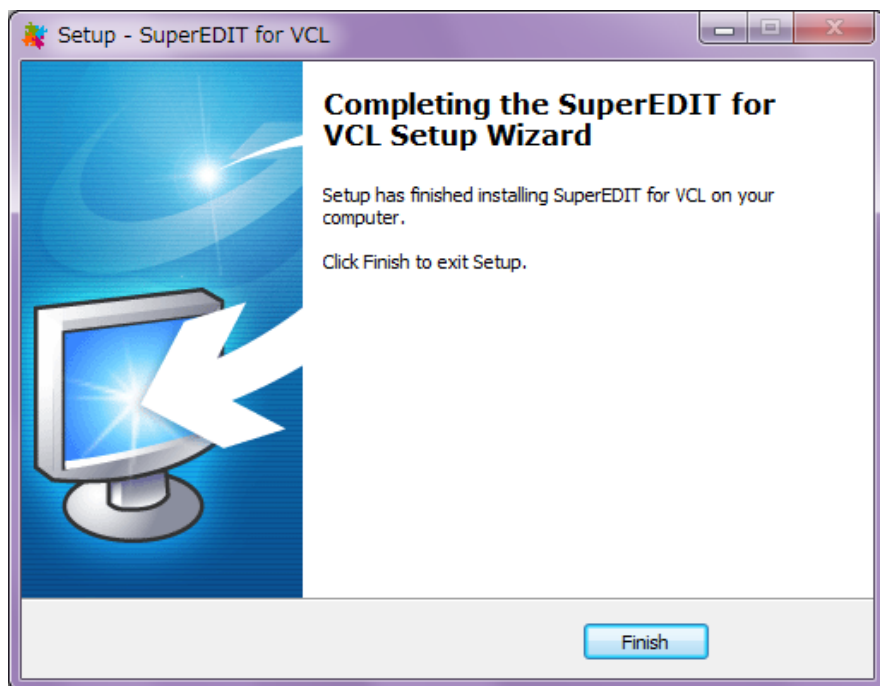
In [Setup - SuperEDIT for VCL] screen, specify the installation folder of the component (component of the destination), and then click [Next].



- **To exit the setup program**

When all of the files are copied to the hard disk, you will see the following screen.

Click [Finish] to complete the setup.





## 7.2 File organization

When the installation is done correctly, the next folder will be created on the hard disk.

This section describes the copy is the file in the folder.

Folder / subfolder name		Content
D2007 *1		Delphi 2007 32-bit environment *.bpl / *.dcp: Compiled package files *.dcu: Component body *.res: Resource file
D2010 *1		The contents are the same as above
DXE7	Win32	Delphi XE7 For for 32-bit environment (the contents are the same as above)
	Win64	Delphi XE7 For for 64-bit environment (the contents are the same as above)
DXE8	Win32	Delphi XE8 For for 32-bit environment (the contents are the same as above)
	Win64	Delphi XE8 For for 64-bit environment (the contents are the same as above)
DXE10	Win32	Delphi 10 Seattle For for 32-bit environment (the contents are the same as above)
	Win64	Delphi 10 Seattle For for 64-bit environment (the contents are the same as above)
DXE101	Win32	Delphi 10.1 Berlin For for 32-bit environment (the contents are the same as above)
	Win64	Delphi 10.1 Berlin For for 64-bit environment (the contents are the same as above)
DLL		SE32.DLL: Dynamic link library for 32-bit environment SE64.DLL: Dynamic link library for 64-bit environment
DOC		LicenseAgreement.pdf: License Agreement SuperEDIT_Manual_JP.pdf: Japanese manual (SuperEDIT) SuperEDIT_Manual_EN.pdf: English manual (SuperEDIT) SuperGRID_Manual_JP.pdf: Japanese manual (SuperGRID) SuperGRID_Manual_EN.pdf: English manual (SuperGRID)

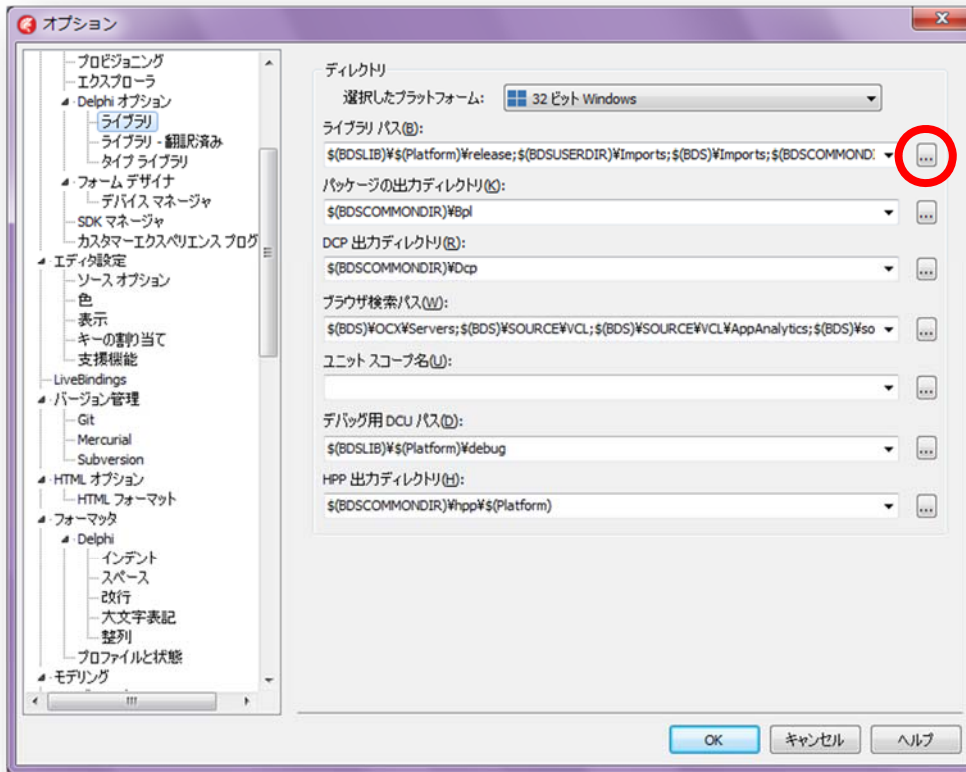
\* 1: The 32-bit only and will.

### 7.3 Setting the library path

Components are not street compilation and not in the folder in your path.

To do this, you need to set the library path of Delphi. This section describes the "Delphi XE8" as an example.

[Tools], click [Options], and then display the [Options] screen.



- Selection of Delphi options

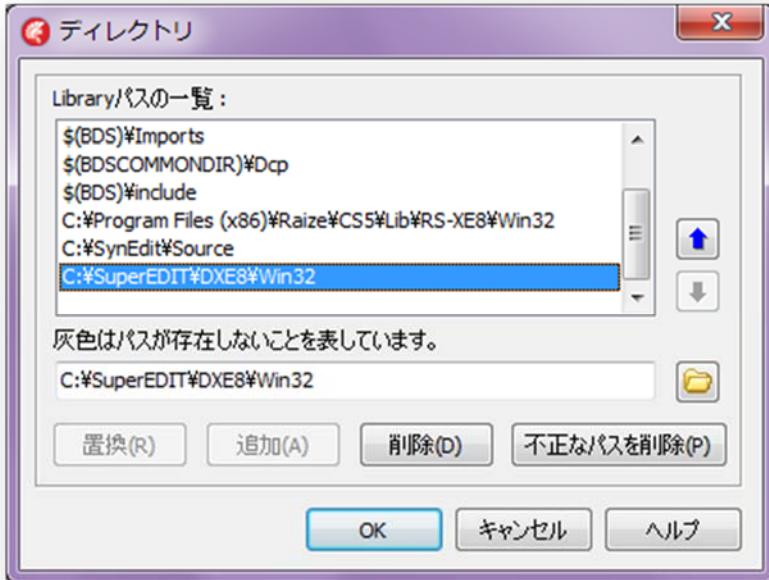
From the left side of the pane, [Delphi Options], and then select the [Library].

- **Setting the library path**

Select the "32-bit Windows" in the [Selected platform] in the right pane.

Next, click the Browse button in the [Library Path] (**circles**), of the components that you have installed, set the folder that matches the Delphi version you are using, and then click the [Add] button.

In this example, it specifies a 32-bit environment of Delphi XE8. (**C:\SuperEDIT\DXE8\Win32**)



Finally is complete, click the [OK] button.

Continue to the setting of the 64-bit environment.

Select the "64-bit Windows" in the [Selected platform], you can make settings of the library path of a 32-bit same 64-bit environment.

For example, it will be "**C:\SuperEDIT\DXE8\Win64**".

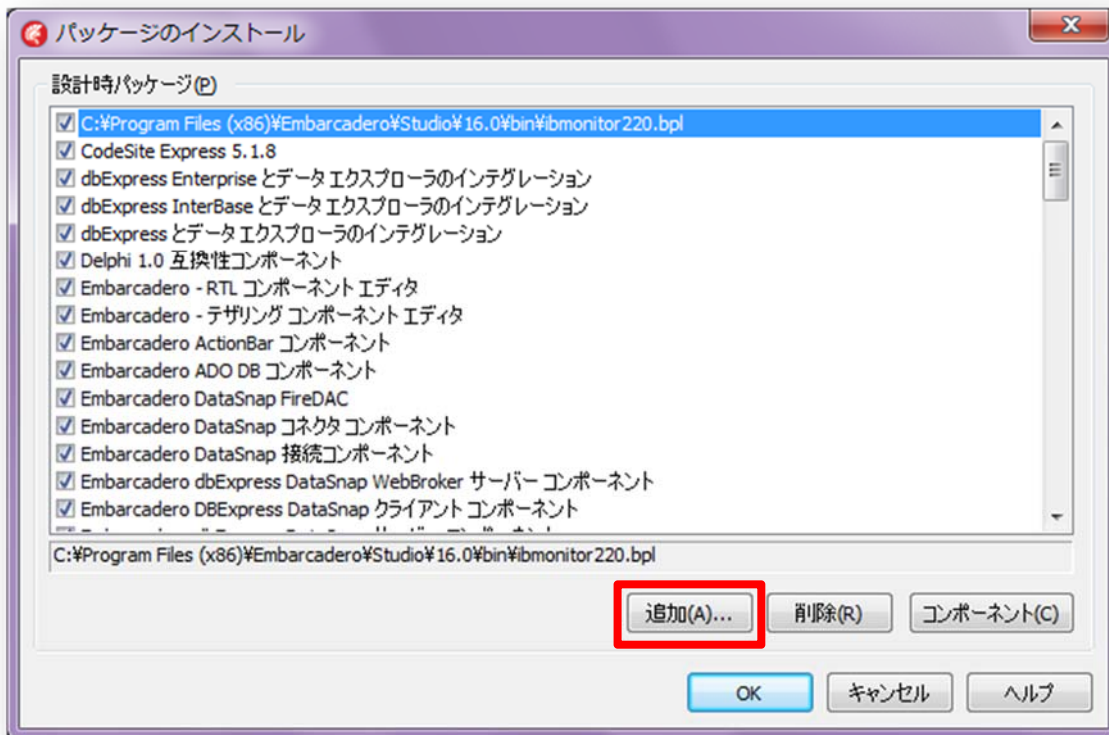
## 7.4 Installation of the package

Finally, install the "design-time package" in Delphi.

When the installation is done correctly, an icon will appear in the component palette.

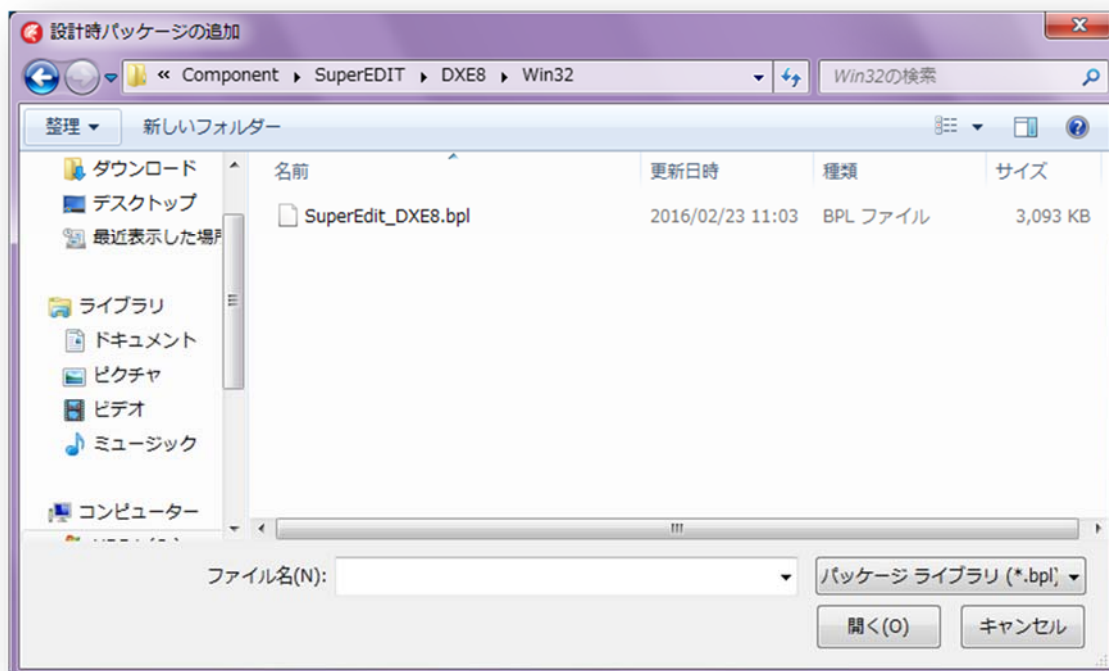
[Components], then click the [Install package], and then display the package installation of the screen.

Click the [Add] button (red frame).

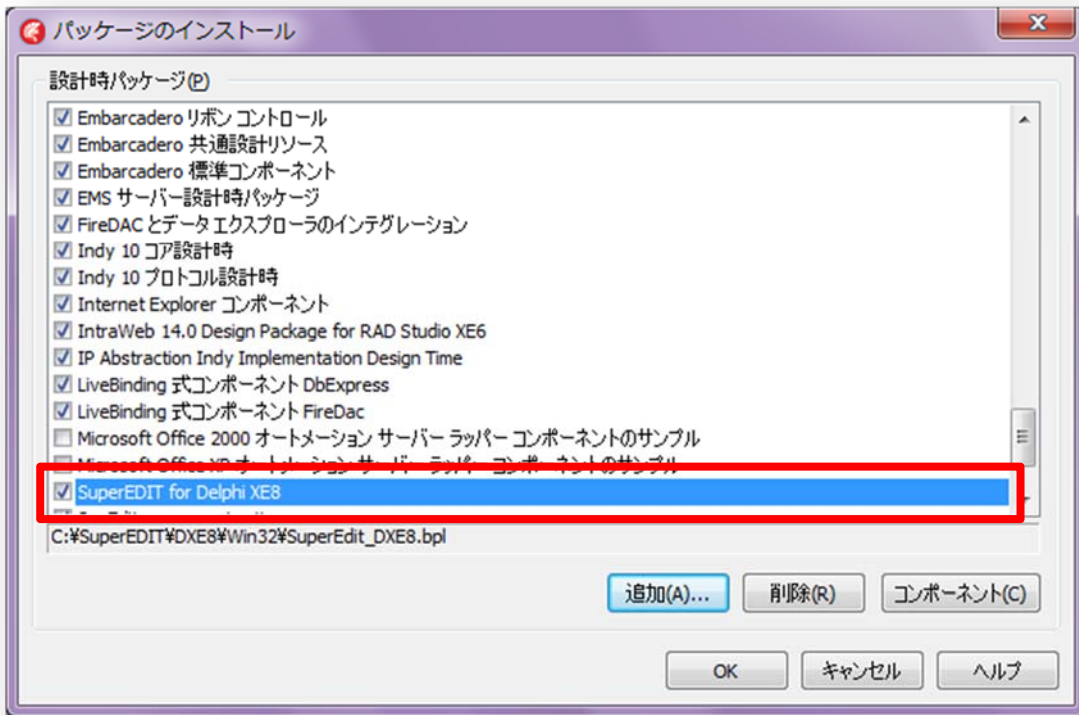


### ● Adding Components

Package file of the same components as the Delphi version to use (extension, bpl), and then click the [Open] button.



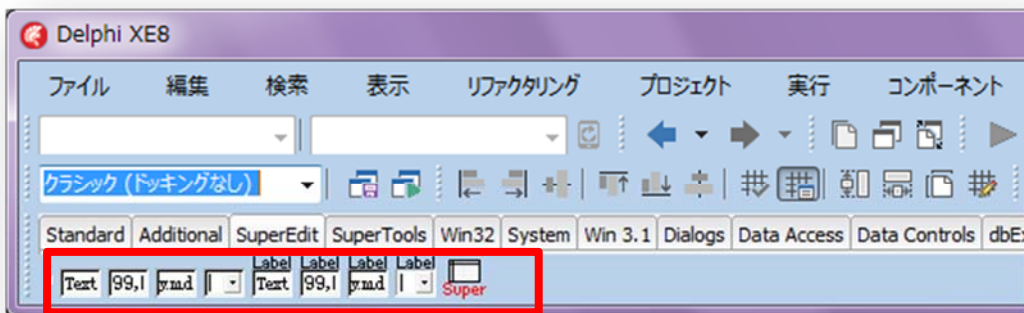
When properly package is installed, you will like the following screen.



Finally is complete, click the [OK] button.

When the installation is done correctly, an icon will appear in the component palette.

**Notice: If the display of classic**



## 7.5 For SuperEDIT dynamic link library

Applications that have been developed using SuperEDIT, it must have run during the next dynamic link library (DLL).

- SE32.DLL: 32-bit dynamic link library for the environment.
- SE64.DLL: dynamic link library for the 64-bit environment.

Please use to copy the suits DLL to the environment to **"the same folder as the execution module."**

## 8. To uninstall the component

Uninstall of SuperEDIT, perform the following steps.

- **To end the Delphi**

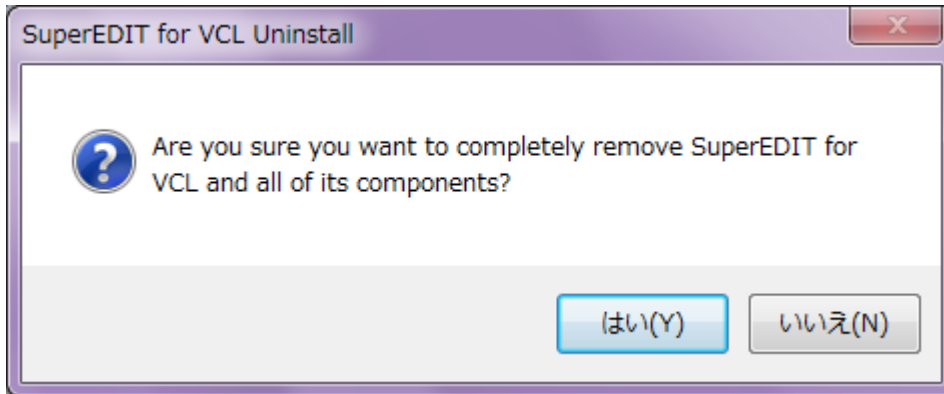
All Delphi it has started to exit.

- **To start the uninstall program**

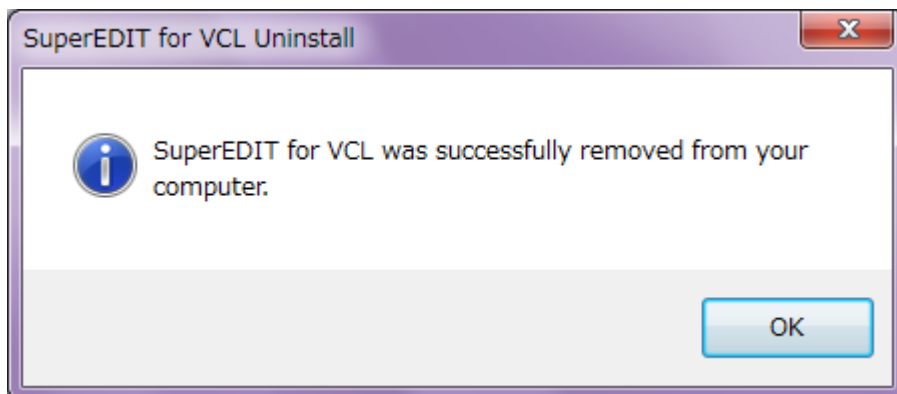
Select [Programs and Features] in Control Panel, please go deletion of application (uninstall).

When the screen of the [User Account Control] appears, click [Yes].

Confirmation screen of the uninstallation appear. Click [Yes] to run the uninstall.



When properly uninstall is executed, so you will see the following screen and click the [OK] button, and exit.



- **To delete a SuperEDIT dynamic link library**

DLL that you copied to the run folder (**SE32.DLL** / **SE64.DLL**) Please also deleted.

Uninstall or more, it was all completed.

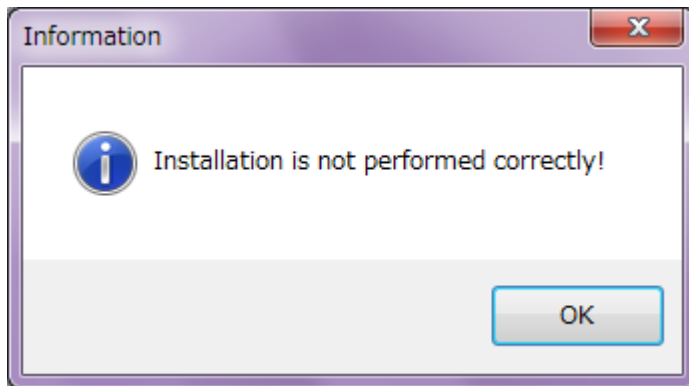
## 9. About Message Display

This section describes the messages that are displayed during the SuperEDIT run.

- **"Installation has not been done correctly!"**

This message, SuperEDIT dynamic link library will be displayed when there is no to run the application folder.

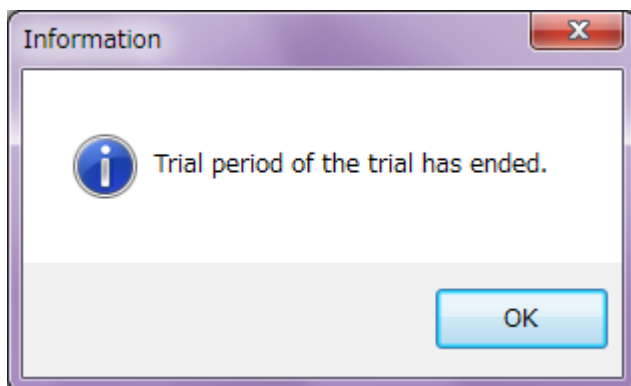
In the folder "**SE32.DLL** (or, **SE64.DLL**)" to copy, please re-start the application.



- **"Trial period of the trial has ended."**

This message is displayed when the trial period of SuperEDIT (30 days) has passed.

If you want to use these products, thank you to purchase and registration of license.



## 10. Description of THintOut class

ThintOut class, status bar that displays the input guidance, the position of the panel to be displayed, and stores the display existence.

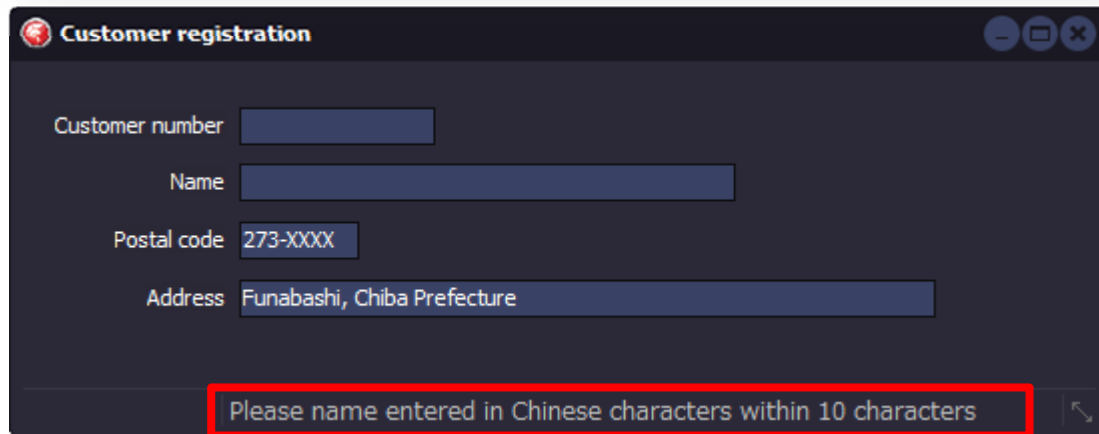
### Description

By setting the HintOut property, you will be able to automatically display the text string that is set in the Hint property at the time of the focus of acquisition in the status bar.

By using this property, you can reduce the number of steps since the coding is not required to display the input guidance.

Also, you can set the position of the panel which displays the presence and display of input guidance in the property.

### Setting example of HintOut property



### 10.1 Property

Property Name	Description
HintBar	It will specify the status bar that displays the input guidance
PanelId	You specify the index of the stator panel to display the input guidance
ShowHint	Specifies whether to display the input guidance



## 10.2 Property Description

**P** **HintBar**

InpText InpNumber InpDate **InpComboBox** LabeledText LabeledNumber LabeledDate **LabeledComboBox**

HintBar property specifies the status bar that displays the input guidance.

### Declaration

```
property HintBar: TStatusBar;
```

### Description

HintBar property specifies the status bar that displays the input guidance.

**P** **PanelId**

InpText InpNumber InpDate **InpComboBox** LabeledText LabeledNumber LabeledDate **LabeledComboBox**

PanelId property specifies the index of the stator panel to display the input guidance.

### Declaration

```
property PanelId: Integer;
```

### Description

PanelId property specifies the index of the stator panel to display the input guidance.

**P** **ShowHint**

InpText InpNumber InpDate **InpComboBox** LabeledText LabeledNumber LabeledDate **LabeledComboBox**

ShowHint property specifies whether to display the input guidance.

### Declaration

```
property ShowHint: Boolean;
```

### Description

ShowHint property, you can decide whether to display the input guidance. It will show the value of ShowHint below.

Value	Description
True	Display
False	Do not display (default)

## 11. Description of custom properties

### **P** AllowCharKind

InpText

InpNumber

InpDate

InpComboBox

LabeledText

LabeledNumber

LabeledDate

LabeledComboBox

AllowCharKind property specifies the type of character to be allowed to enter. Automatic check of input characters is done by the designation of this property.

#### Declaration

```
type TCharKind =  
    (ckKanji, ckHiragana, ckWideKatakana, ckWideAlpha, ckWideNumber, ckWideSymbol,  
     ckHalfKatakana, ckHalfAlpha, ckHalfNumber, ckHalfSymbol);  
TCharKinds = Set of TCharKind;  
property AllowCharKind: TCharKinds;
```

#### Description

AllowCharKind property specifies the type of character to be allowed to enter. If you enter anything other than the specified character type, OnInvalidEntry event occurs. It will show the value of AllowCharKind below.

Value	Description
ckKanji	"Kanji" allowed to enter.
ckHiragana	"Hiragana" allowed to enter.
ckWideKatakana	"Katakana" allowed to enter.
ckWideAlpha	It will allow the "full-width alphabet" input.
ckWideNumber	It will allow the "full-size number" input.
ckWideSymbol	It will allow "double-byte symbol" input.
ckHalfKatakana	It will allow the "half-width katakana" input.
ckHalfAlpha	It will allow the "half-size letters" input.
ckHalfNumber	It will allow the "byte numbers" input.
ckHalfSymbol	It will allow "single-byte symbol" input.

#### Setting example of AllowCharKind property

The following code will only be input Kanji, Hiragana.

```
procedure TForm1.FormCreate(Sender: TObject);  
begin  
    InpText1.AllowCharKind := [ckKanji, ckHiragana];  
end;
```

## P AllowEdit

InpText

InpNumber

InpDate

InpComboBox

LabeledText

LabeledNumber

LabeledDate

LabeledComboBox

AllowEdit property, you can set whether to allow the input of the edit box.

### Declaration

```
property AllowEdit: Boolean;
```

### Description

AllowEdit property, you can set whether to allow the input of the edit box.

If AllowEdit property is False, will not be able to input text in the edit box, it will be to select an item from the drop-down list. It will show the value of AllowEdit below.

Value	Description
True	Input is possible (default)
False	Input disabled

## P AllowNullValue

InpText

InpNumber

InpDate

InpComboBox

LabeledText

LabeledNumber

LabeledDate

LabeledComboBox

AllowNullValue property specifies whether to allow the input of Null (blank).

### Declaration

```
property AllowNullValue: Boolean;
```

### Description

AllowNullValue property specifies whether to allow the input of Null (blank).

Even if AllowNullValue property is set to True, if it is something input in control, date check is performed.

In addition, if you set the Null to control in the program as the initial value, or if you want to determine the Null, use the AsString property.

Value	Description
True	Null (blank) can be input
False	Null (blank) No input. (Default)

AsCurrency property, set of values by the Currency type, will retrieve.

## Declaration

```
property AsCurrency: Currency;
```

## Description

AsCurrency property, set of values by the Currency type, will retrieve.

If you attempt to set a wrong value, at the same time OnInvalidEntry events or OnInvalidRange event when setting the value becomes invalid occurs.

## Example of the use of AsCurrency property

The following code sets the value in the Currency type.

```
procedure TForm1.Button1Click(Sender: TObject);  
begin  
  InpNumber1.AsCurrency := 15000.1234;  
end;
```

The following code retrieves the value of the Currency type, you can display and edit the Label.

```
procedure TForm1.Button1Click(Sender: TObject);  
begin  
  Label1.Caption := FormatCurr('#,###.0000', InpNumber1.AsCurrency);  
end;
```

# P AsFloat

\_inpText

**\_inpNumber**

\_inpDate

\_inpComboBox

\_labeledText

**\_labeledNumber**

\_labeledDate

\_labeledComboBox

AsFloat property, set of values by the Double type, will retrieve.

## Declaration

```
property AsFloat: Double;
```

## Description

AsFloat property, set of values by the Double type, will retrieve. If you attempt to set a wrong value, invalid and made at the same time OnInvalidEntry event is set value or OnInvalidRange event occurs.

## Example of the use of AsFloat property

The following code sets the value to the Double type.

```
procedure TForm1.Button1Click(Sender: TObject);
begin
  InputNumber1.AsFloat := 2500.123456789;
end;
```

The following code retrieves the value of the Double type, you can display and edit the Label.

```
procedure TForm1.Button1Click(Sender: TObject);
begin
  Label1.Caption := FormatFloat('#,###.#####', InputNumber1.AsFloat);
end;
```

AsFurigana property gets the phonetic entered using the kana-kanji conversion system (IME). **(Read-only)**

## Declaration

```
property AsFurigana: string;
```

## Description

AsFurigana property gets the phonetic. **(Read-only)**

Normally, you specify the control to output the phonetic in FuriganaOut property, if you want to output to the control that does not have a Text property, you can use the AsFurigana property.

## Example of the use of phonetic property

The following code, and then set the Caption property of the Label component to get the phonetic in AsFurigana property.

```
procedure TForm1.InpText1Change (Sender: TObject) ;  
begin  
    Label1.Caption := InpText1.AsFurigana;  
end;
```

# P AsInteger

InpText

**InpNumber**

InpDate

InpComboBox

LabeledText

**LabeledNumber**

LabeledDate

LabeledComboBox

AsInteger property, set of values by type Integer, will retrieve.

## Declaration

```
property AsInteger: Integer;
```

## Description

AsInteger property, set of values by type Integer, will retrieve. If you attempt to set a wrong value, at the same time OnInvalidEntry events or OnInvalidRange event when setting the value becomes invalid occurs.

## Example of the use of AsInteger property

The following code sets the value to Integer.

```
procedure TForm1.Button1Click(Sender: TObject);  
begin  
  InpNumber1.AsInteger := 2500;  
end;
```

The following code retrieves the value of type Integer, and then output to the Edit.

```
procedure TForm1.Button1Click(Sender: TObject);  
begin  
  Edit1.Text := IntToStr(InpNumber1.AsInteger);  
end;
```

# P AsPrice

InpText

InpNumber

InpDate

InpComboBox

LabeledText

LabeledNumber

LabeledDate

LabeledComboBox

AsPrice property, take out the "price" using the TaxCalc method. **(Read-only)**

## Declaration

```
property AsPrice: Currency;
```

## Description

The calculation results of TaxCalc method, AsPrice property returns the "commodity price". **(Read-only)**

## Example of the use of TaxCalc method

The following code, call the TaxCalc method commodity price (Label1), tax (Label2), to get the tax-included price (Label3).

```
procedure TForm1.Button1Click(Sender: TObject);
begin
  // Consumption tax 8 percent, tax, rounding
  InputNumber1.TaxCalc(8, 0, 0);
  // Commodity prices (without tax price)
  Label1.Caption := FormatCurr('#,###.###', InputNumber1.AsPrice);
  // consumption tax
  Label2.Caption := FormatCurr('#,###.###', InputNumber1.AsTax);
  // Tax-inclusive price
  Label3.Caption := FormatCurr('#,###.###', InputNumber1.AsTotalPrice);
end;
```



AsString property, set of values by type String, will retrieve.

## Declaration

```
property AsString: string;
```

## Description

AsString property, set of values by type String, will retrieve. If you attempt to set a wrong value, at the same time OnInvalidEntry events or OnInvalidRange event when setting the value becomes invalid occurs.

## Example of the use of AsString property

The following code sets the value to a String type.

```
procedure TForm1.Button1Click(Sender: TObject);  
begin  
  InpNumber1.AsString := '3500.12';  
end;
```

The following code retrieves the value of AsString type, and then output to the Edit.

```
procedure TForm1.Button1Click(Sender: TObject);  
begin  
  Edit1.Text := InpNumber1.AsString;  
end;
```

## P AsString

InpText

InpNumber

**InpDate**

InpComboBox

LabeledText

LabeledNumber

**LabeledDate**

LabeledComboBox

AsString property, set of values by type String, will retrieve.

### Declaration

```
property AsString: string;
```

### Description

AsString property, set of values by type String, will retrieve.

If DisplayFormat property is set, you can get to formalize the date.

### Example of the use of AsString

The following code, edit the value entered in the date input component, it has displayed in the Label component.

```
procedure TForm1.Col orButton1Cl ick (Sender: TObj ect) ;  
begin  
  InpDate1.Di spl ayFormat := 'gggee"年"mm"月"dd"日" (aaa) ' ;  
  Label 1. Capti on := InpDate1.AsStri ng;  
end;
```

## P AsTax

InpText

**InpNumber**

InpDate

InpComboBox

LabeledText

**LabeledNumber**

LabeledDate

LabeledComboBox

AsTax property, take out the "consumption tax" using the TaxCalc method. **(Read-only)**

### Declaration

```
property AsTax: Currency;
```

### Description

The calculation results of TaxCalc method, AsTax property returns "consumption tax". **(Read-only)**

# P AsTotalPrice

InpText

InpNumber

InpDate

InpComboBox

LabeledText

LabeledNumber

LabeledDate

LabeledComboBox

AsTotalPrice property, take out the "tax price" using the TaxCalc method. **(Read-only)**

## Declaration

```
property AsTotalPrice: Currency;
```

## Description

The calculation results of TaxCalc method, AsTotalPrice property returns the "tax price". **(Read-only)**

# P AutoDropDown

InpText

InpNumber

InpDate

InpComboBox

LabeledText

LabeledNumber

LabeledDate

LabeledComboBox

AutoDropDown property, control and specify whether to automatically display the drop-down calendar when it receives the input focus.

## Declaration

```
property AutoDropDown: Boolean;
```

## Description

AutoDropDown property, control and specify whether to automatically display the drop-down calendar when it receives the input focus. OnDrop event occurs each time a drop-down calendar is called. It will show the value of AutoDropDown below.

Value	Description
True	We want to display a drop-down calendar.
False	Do not show the drop-down calendar. (Default)

# P AutoDropDown

InpText

InpNumber

InpDate

InpComboBox

LabeledText

LabeledNumber

LabeledDate

LabeledComboBox

AutoDropDown property, control and specify whether to automatically display the drop-down list when it receives the input focus.

## Declaration

```
property AutoDropDown: Boolean;
```

## Description

AutoDropDown property, control and specify whether to automatically display the drop-down list when it receives the input focus. OnDropDown event occurs every time the drop-down list is displayed. It will show the value of AutoDropDown below.

Value	Description
True	You want to display a drop-down list.
False	Do not show the drop-down list. (Default)

# P AutoEnter

InpText

InpNumber

InpDate

InpComboBox

LabeledText

LabeledNumber

LabeledDate

LabeledComboBox

AutoEnter property, specify whether to automatically move the focus to the next control when there is an input of the number of digits that are specified by the MaxLength property.

## Declaration

```
property AutoEnter: Boolean;
```

## Description

AutoEnter property is a valid only if the MaxLength property is specified, when there is an input for a specified number of digits in the MaxLength property, specifies automatically whether to move the focus to the next control. It will show the value of AutoEnter below.

Value	Description
True	You want to move a control.
False	It does not move the controls. (Default)

# P BeepOnError

InpText

InpNumber

InpDate

InpComboBox

LabeledText

LabeledNumber

LabeledDate

LabeledComboBox

BeepOnError property, you can specify whether to sound the time of input error warning sound.

## Declaration

```
property BeepOnError: Boolean;
```

## Description

BeepOnError property, you can specify whether to sound the time of input error warning sound.

It will show the value of BeepOnError below.

Value	Description
True	It sounds a warning sound. (Default)
False	No warning tone will sound.

# P ButtonStyle

InpText

InpNumber

InpDate

InpComboBox

LabeledText

LabeledNumber

LabeledDate

LabeledComboBox

ButtonStyle property specifies the display format of the button.

## Declaration

```
type TButtonStyle = (bsNormal, bsOnFocus, bsAlways);  
property ButtonStyle: TButtonStyle;
```

## Description

ButtonStyle property is valid only if the Alignment property is taLeftJustify, to specify the display format of the search button. It will show the value of ButtonStyle below.

Value	Description
bsNormal	Do not show the button. (Default)
bsOnFocus	Show button when it receives the focus.
bsAlways	Always you want to display a button.

**Notice: If ButtonStyle property is specified other than bsNormal In text input component, PasswordChar property is disabled.**

ButtonType property specifies the button of shape.

## Declaration

```
type TButtonType = (btDropdown, btEllipsis) ;
property ButtonType: TButtonType;
```

## Description

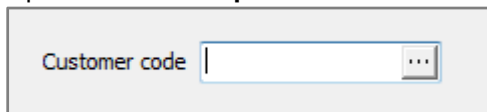
ButtonType property, ButtonStyle property bsOnFocus, the case of bsAlways only valid, specify the button of shape.

OnEllipsisClick event occurs each time the button is pressed. It will show the value of ButtonType below.

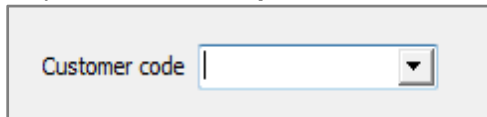
Value	Description
btEllipsis	You want to display the search button. (Default)
btDropdown	You want to display a drop-down button.

## Setting example of ButtonType property

Specifies the **btEllipsis**.



It specifies the **btDropdown**.



ButtonType property specifies the button of shape.

### Declaration

```
type TButtonType = (btDropdown, btEllipsis) ;
property ButtonType: TButtonType;
```

### Description

ButtonType property, ButtonStyle property bsOnFocus, the case of bsAlways only valid, specify the button of shape.

#### btDropdown

Drop-down calculator is displayed.

OnPopup event occurs each time the button is pressed.

#### btEllipsis

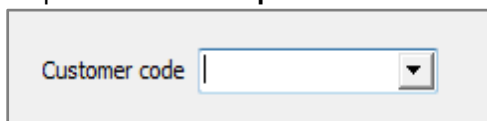
OnEllipsisClick event occurs each time the button is pressed.

It will show the value of ButtonType below.

Value	Description
btDropdown	You want to display a drop-down button. (Default)
btEllipsis	You want to display the search button.

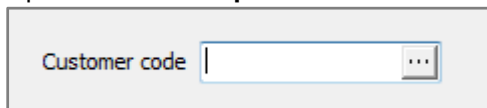
### Setting example of ButtonType property

It specifies the **btDropdown**.



Customer code

Specifies the **btEllipsis**.



Customer code

# CanNegative

InpText

InpNumber

InpDate

InpComboBox

LabeledText

LabeledNumber

LabeledDate

LabeledComboBox

CanNegative property is used if you want to limit the negative input.

## Declaration

```
property CanNegative: Boolean;
```

## Description

CanNegative property, you can either allow the negative input, and specify whether or not.

Value	Description
True	Negative number of possible input. (Default)
False	Negative disabled input.

# ColorOnDisabled

InpText

InpNumber

InpDate

InpComboBox

LabeledText

LabeledNumber

LabeledDate

LabeledComboBox

ColorOnDisabled property, Enabled property to specify the color at the time of the False.

## Declaration

```
property ColorOnDisabled: TColor;
```

## Description

ColorOnDisabled property, Enabled property to specify the color at the time of the False.

It is used when you want to visually represent an item that can not be input. It will show the value of ColorOnDisabled below.

Value	Description
clBtnFace	The button face color (default)

**Notice:** Others, please refer to the help TColor type of Delphi.



# P ColorOnFocus

InpText

InpNumber

InpDate

InpComboBox

LabeledText

LabeledNumber

LabeledDate

LabeledComboBox

ColorOnFocus property specifies the color when the control receives the input focus.

## Declaration

```
property ColorOnFocus: TColor;
```

## Description

ColorOnFocus property specifies the color when the control receives the input focus.

It will show the value of ColorOnFocus below.

Value	Description
clWindow	Window background color (default)
Notice: Others, please refer to the help TColor type of Delphi.	

# P ColorOnNoFocus

InpText

InpNumber

InpDate

InpComboBox

LabeledText

LabeledNumber

LabeledDate

LabeledComboBox

ColorOnNoFocus property specifies the color when the control loses the input focus.

## Declaration

```
property ColorOnNoFocus: TColor;
```

## Description

ColorOnNoFocus property specifies the color when the control loses the input focus.

If you set the value in the Object Inspector at design time the changes are reflected immediately in control.

It will show the value of ColorOnNoFocus below.

Value	Description
clWindow	Window background color (default)
Notice: Others, please refer to the help TColor type of Delphi.	

## P ColumnCount

InpText

InpNumber

InpDate

InpComboBox

LabeledText

LabeledNumber

LabeledDate

LabeledComboBox

ColumnCount property specifies the number of columns to be displayed in the drop-down list.

### Declaration

```
property ColumnCount: Word;
```

### Description

ColumnCount property specifies the number of columns to be displayed in the drop-down list.

**Notice:** If you specify a value smaller than the column that is set in the drop-down list, behind the column does not appear.

## P CursorEndField

InpText

InpNumber

InpDate

InpComboBox

LabeledText

LabeledNumber

LabeledDate

LabeledComboBox

CursorEndField property specifies the position of the cursor when it receives the focus.

### Declaration

```
property CursorEndField: Boolean;
```

### Description

CursorEndField property is valid only when AutoSelect property is False, the control is to specify whether to automatically move the cursor when it receives the input focus to the end of the text.

Value	Description
True	We want to move the cursor to the end of the text. (Default)
False	It does not move the cursor to the end of the text.

# P DateFormat

InpText

InpNumber

**InpDate**

InpComboBox

LabeledText

LabeledNumber

**LabeledDate**

LabeledComboBox

DateFormat property specifies the input format of date.

## Declaration

```
type TDateFormat = (dfShort, dfLong) ;  
property DateFormat: TDateFormat;
```

Value	Description
dfShort	Year (Anno Domini) it will be displayed in the two-digit.
dfLong	Year (Anno Domini) it will be a 4-digit display. (Default)

## Description

If you set the DateFormat to dfShort, the date appears as 15/11/11, it will be displayed as of 2015/11/11 When set to dfLong.

By setting this property, you can edit the date that ignores the settings in the Control Panel.

# P Decimals

InpText

**InpNumber**

InpDate

InpComboBox

LabeledText

**LabeledNumber**

LabeledDate

LabeledComboBox

Decimals property specifies the number of digits in the fractional part of the input value.

## Declaration

```
property Decimals: SmallInt;
```

## Description

Decimals property specifies the number of digits in the fractional part of the input value. (Default is 2)

Number of decimal places, it can be up to 9 digits.

## P DelimiterChar

InpText

InpNumber

InpDate

InpComboBox

LabeledText

LabeledNumber

LabeledDate

LabeledComboBox

DelimiterChar property specifies the delimiter of the drop-down list item.

### Declaration

```
property Del i mi terChar: stri ng;
```

### Description

Decimals property specifies the delimiter of the drop-down list item. (Default value is "," (comma))  
Item list to be displayed in the drop-down list, you specified in the ListView property.

## P DisplayFormat

InpText

InpNumber

InpDate

InpComboBox

LabeledText

LabeledNumber

LabeledDate

LabeledComboBox

DisplayFormat property specifies the display format of date.

### Declaration

```
property Di spl ayFormat: stri ng;
```

### Description

DisplayFormat property specifies the display format of date. When the control loses the input focus, it is the date editing in the format of DisplayFormat property.

**Notice:** Format specifier to be used in format, see "FormatDateTime function" of Delphi Help.

### Example of the use of DisplayFormat

The following code, edit the value entered in the date input component, it has displayed in the Label component.

```
procedure TForm1.FormCreate (Sender: TObj ect) ;  
begin  
  InpDate1. Di spl ayFormat := ' gggee"年"mm"月"dd"日" (aaa) ' ;  
end;
```

平成27年11月13日 (金)

## P EditLabel

InpText

InpNumber

InpDate

InpComboBox

LabeledText

LabeledNumber

LabeledDate

LabeledComboBox

EditLabel property holds the label information of the labeled control.

### Declaration

```
type
  TBoundLabel = class(TCustomLabel)
  property EditLabel : TBoundLabel;
```

### Description

Labeled control is added to the label on the input control.

EditLabel property specifies the label information of the labeled control.

### Example of the use of EditLabel property

The following code sets the Caption and Color of the label.

```
procedure TForm1.FormCreate(Sender: TObject);
begin
  LabeledText1.EditLabel.Caption := 'Name';
  LabeledText1.EditLabel.Font.Color := clRed;
end;
```

## P EllipsisHint

InpText

InpNumber

InpDate

InpComboBox

LabeledText

LabeledNumber

LabeledDate

LabeledComboBox

EllipsisHint property holds the text string that the mouse is displayed when moving on the search button.

### Declaration

```
property EllipsisHint: string;
```

### Description

EllipsisHint property specifies the text string that is displayed when the user moves the mouse pointer to the control.

# P EllipsisShowHint

InpText

InpNumber

InpDate

InpComboBox

LabeledText

LabeledNumber

LabeledDate

LabeledComboBox

EllipsisShowHint property, when the mouse cursor is moved on the search button, it indicates whether to display the Help tips.

## Declaration

```
property EllipsisShowHint: Boolean;
```

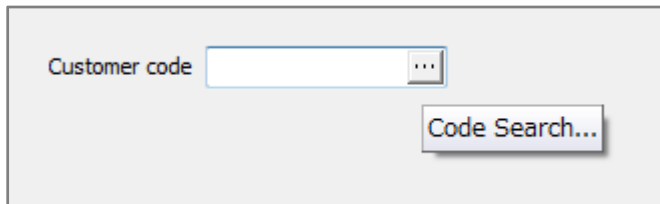
## Description

Help Hint is the value of EllipsisHint property, will be displayed below the search button.

By specifying the EllipsisShowHint property, it will be decided whether to display the Search button in Help tips.

値	意味
True	We want to display the Help Hint.
False	We do not want to display the Help Hint. (Default)

## EllipsisHint, setting example of EllipsisShowHint property



# P EraDate

InpText

InpNumber

InpDate

InpComboBox

LabeledText

LabeledNumber

LabeledDate

LabeledComboBox

EraDate property, take out the date of the Japanese calendar. **(Read-only)**

## Declaration

```
property EraDate: string;
```

## Description

EraDate property, take out the date of the Japanese calendar.

# P Flat

InpText

InpNumber

InpDate

InpComboBox

LabeledText

LabeledNumber

LabeledDate

LabeledComboBox

Flat property specifies the appearance of the control.

## Declaration

```
property Flat: Boolean;
```

## Description

Flat property specifies the appearance of the control.

Value	Description
True	Control appears in the flat.
False	Control does not appear in the flat. (Default)

## Setting example of Flat property

Specify **True**

Specify **False**



# FontOnFocus

InpText

InpNumber

InpDate

InpComboBox

LabeledText

LabeledNumber

LabeledDate

LabeledComboBox

FontOnFocus property specifies the text of the attribute when the control receives the input focus.

## Declaration

```
property FontOnFocus: TFont;
```

## Description

FontOnFocus property, the height of the text when the control receives the input focus, font name, a name such as attributes (bold, italics).

**Notice:** TFont, please refer to the help of Delphi.



# FontOnNoFocus

InpText

InpNumber

InpDate

InpComboBox

LabeledText

LabeledNumber

LabeledDate

LabeledComboBox

FontOnNoFocus property specifies the text of the attribute when the control loses the input focus.

## Declaration

```
property FontOnNoFocus: TFont;
```

## Description

FontOnNoFocus property, the height of the text when the control loses the input focus, font name, a name such as attributes (bold, italics).

**Notice:** TFont, please refer to the help of Delphi.



FuriganaOut property specifies the control to output a phonetic obtained when using the automatic phonetic function.

### Declaration

```
property FuriganaOut: TCustomEdit;
```

### Description

FuriganaOut property specifies the controls to output the acquired phonetic.

Component that can be specified as the destination of phonetic is limited to control derived from TCustomEdit object (TInpText, TEdit, TMemo, etc.). Phonetic is set to the destination of the Text property.

In addition, you can also get the phonetic in AsFurigana property.

FuriganaType property specifies the character type of phonetic output. Character type, you can select single-byte katakana, hiragana, from three full-width Katakana.

### Declaration

```
type TFuriganaType = (ftHalfKatakana, ftWideKatakana, ftHiragana);
property FuriganaType: TFuriganaType;
```

### Description

FuriganaType property specifies the character type of phonetic output.

It will show the value of the phonetic Type below.

Value	Description
ftHalfKatakana	It outputs a phonetic in the "half-width katakana". (Default)
ftHiragana	It outputs a phonetic in "hiragana".
ftWideKatakana	It outputs a phonetic in "Katakana".

# P HintOut

InpText

InpNumber

InpDate

InpComboBox

LabeledText

LabeledNumber

LabeledDate

LabeledComboBox

HintOut property, if you want to display a text string that is specified in the Hint property in the status bar, specify the status bar and the display position and the like to be displayed.

## Declaration

```
type
  THintOut = class(TPersistent)
  property HintOut: THintOut;
```

## Description

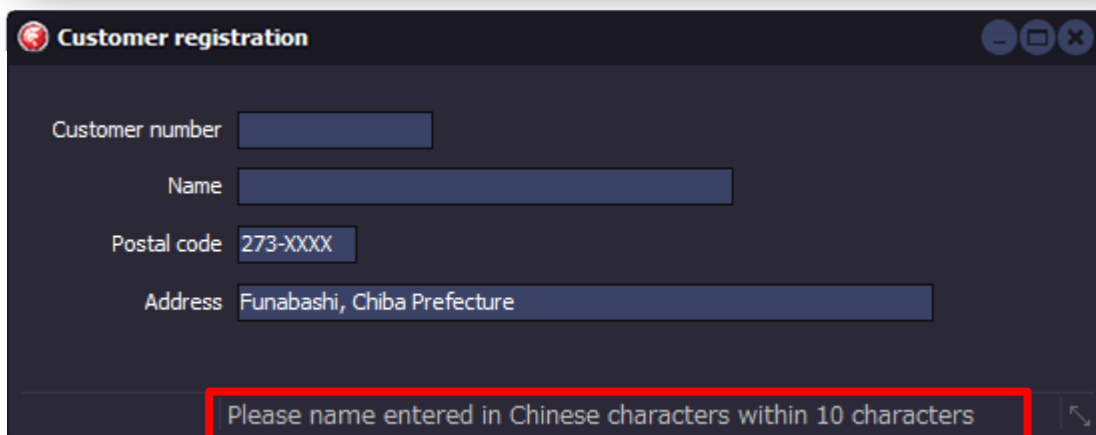
By specifying the HintOut property, you will be able to automatically display the text string in the status bar that is specified in the Hint property at the time of the focus of the acquisition.

By using the HintOut property, you can reduce the number of steps for coding to display the input guidance is unnecessary.

Also, you can specify the location of the panel which displays the presence and display of input guidance in the property.

**Notice:** For more information, see the "7. Description of THintOut class".

## Setting example of HintOut property



# P InsertMode

InpText

InpNumber

InpDate

InpComboBox

LabeledText

LabeledNumber

LabeledDate

LabeledComboBox

InsertMode property specifies the input mode (insert / overwrite) when it receives the input focus.

## Declaration

```
property InsertMode: Boolean;
```

## Description

Text input component can specify the input mode when it receives the input focus.

Also, by pressing the at the time of execution [Insert] (insert key), you can also switch the input mode.

OnInsModeChanged event occurs each time the input mode changes.

Value	Description
True	Insert mode (default)
False	Overwrite mode

# P LabelPosition

InpText

InpNumber

InpDate

InpComboBox

LabeledText

LabeledNumber

LabeledDate

LabeledComboBox

LabelPosition property specifies the location of the label.

## Declaration

```
type  
  TLabelPosition = (lpAbove, lpBelow, lpLeft, lpRight);  
property LabelPosition: TLabelPosition;
```

## Description

Labeled control is added to the label on the input control.

LabelPosition property specifies the location of the label.

Value	Description
lpAbove	Place on top of the input controls the label. (the default)
lpBelow	Place it in the bottom of the input control the label.
lpLeft	It will place the label to the left of the input control.
lpRight	It will place the label to the right of the input control.

## P LabelSpacing

InpText

InpNumber

InpDate

InpComboBox

LabeledText

LabeledNumber

LabeledDate

LabeledComboBox

LabelSpacing property specifies the interval of the input control and label.

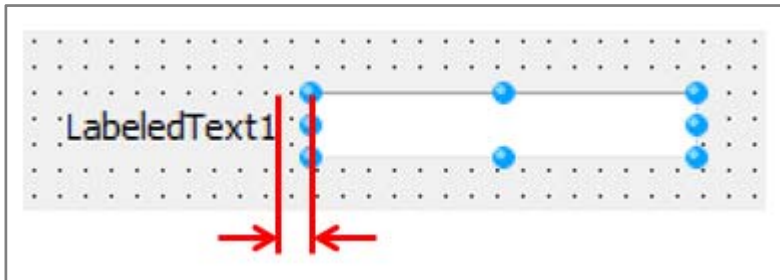
### Declaration

```
property LabelSpacing: Integer;
```

### Description

Labeled control is added to the label on the input control.

LabelSpacing property specifies the interval of the input control and label. (The default value, 3)



## P LeadingChar

InpText

InpNumber

InpDate

InpComboBox

LabeledText

LabeledNumber

LabeledDate

LabeledComboBox

LeadingChar property specifies the currency symbol.

### Declaration

```
property LeadingChar: Char;
```

### Description

LeadingChar property is valid only when ZeroDisplay property is True, when you add a currency symbol to the beginning of the input numerical value, you specify its character. The default value, space (#32) has been set.

# P ListValue

InpText

InpNumber

InpDate

**InpComboBox**

LabeledText

LabeledNumber

LabeledDate

LabeledComboBox

ListValue property specifies the item in the drop-down list.

## Declaration

```
property ListValue: TStrings;
```

## Description

In InpComboBox / LabeledComboBox control, Items property in the ComboBox of Delphi standard we do not have. Items to be displayed in the drop-down list, set to ListValue property.

## Example of the use of ListValue property

The following code sets the value to ListValue.

```
procedure TForm1.FormCreate(Sender: TObject);
begin
  // Clear of the drop-down list
  InpComboBox1.ListValue.Clear;
  // Set of list items
  InpComboBox1.ListValue.Add(' 00585, 田中 芳起, ㊦㊰㊱');
  InpComboBox1.ListValue.Add(' 01223, 丹羽 幸司, ㊦㊰㊱');
  InpComboBox1.ListValue.Add(' 01396, 梶田 伸太郎, ㊦㊰㊱');
  InpComboBox1.ListValue.Add(' 02324, 後藤 哲夫, ㊦㊰㊱');
  InpComboBox1.ListValue.Add(' 02834, 亀谷 高志, ㊦㊰㊱');
  InpComboBox1.ListValue.Add(' 03119, 有賀 進, ㊦㊰㊱');
  InpComboBox1.ListValue.Add(' 04289, 生駒 多喜人, ㊦㊰㊱');
  // Setting Properties
  InpComboBox1.ColumnCount := 3;
  InpComboBox1.ValuePosition := 2;
end;
```

## P MaxLength

InpText

InpNumber

InpDate

InpComboBox

LabeledText

LabeledNumber

LabeledDate

LabeledComboBox

MaxLength property specifies the maximum number of characters that can be entered in the control.

### Declaration

```
property MaxLength: Integer;
```

### Description

MaxLength property specifies the maximum number of characters that can be entered in the control. (The default value is 0)

## P MaxValue

InpText

InpNumber

InpDate

InpComboBox

LabeledText

LabeledNumber

LabeledDate

LabeledComboBox

MaxValue property specifies the maximum value of a number that can be entered into the control.

### Declaration

```
property MaxValue: Double;
```

### Description

Set value of the MaxValue property specifies a value greater than the set value of MinValue property.

If you attempt to set the MinValue property value less than, it will be set to automatically value of MinValue property.

In addition, the effective setting is the maximum 13-digit -999999999999 ~ 999999999999. The number of digits that can be input integer part, are influenced by the number of digits in the fractional part.

If the entered value exceeds the range, at the same time OnInvalidRange events subsequent input becomes impossible occurs.

However, MaxValue property, if the zero to MinValue property both are set, range check is not performed.

### Example of the use of MaxValue property

The following code, the number of non-0-1000 will be treated as an error.

```
procedure TForm1.FormCreate (Sender: TObject) ;  
begin  
  InpNumber1.MaxValue := 1000;  
  InpNumber1.MinValue := 0;  
end;
```

## P MaxValue

InpText

InpNumber

**InpDate**

InpComboBox

LabeledText

LabeledNumber

**LabeledDate**

LabeledComboBox

MaxValue property sets the maximum value of the date that can be entered into the control.

### Declaration

```
property MaxValue: TDateTime;
```

### Description

Set value of the MaxValue property sets a value greater than the set value of MinValue property. If you attempt to set the MinValue property value less than, it will be set to automatically value of MinValue property.

If the entered value exceeds the range, at the same time OnInvalidRange events subsequent input becomes impossible occurs. However, MaxValue property, if the zero to MinValue property both are set, range check is not performed.

### Example of the use of MaxValue property

The following code, date other than December 2015 (2015/12/1-2015/12/31) will be treated as an error.

```
procedure TForm1.FormCreate (Sender: TObject) ;
begin
  InpDate1.MinValue := StrToDateTime(' 2015/12/1' );
  InpDate1.MaxValue := StrToDateTime(' 2015/12/31' );
end;
```

## P MinusColor

InpText

**InpNumber**

InpDate

InpComboBox

LabeledText

**LabeledNumber**

LabeledDate

LabeledComboBox

MinusColor property, numeric value specifies the text color at the time of the negative number.

### Declaration

```
property MinusColor: TColor;
```

### Description

MinusColor property, numeric value specifies the text color at the time of the negative number.

値	意味
clRed	Red (default)

**Notice: Others, please refer to the help TColor type of Delphi.**

## P MinValue

InpText

InpNumber

InpDate

InpComboBox

LabeledText

LabeledNumber

LabeledDate

LabeledComboBox

MinValue property sets the minimum value of a number that can be entered into the control.

### Declaration

```
property MinValue: TDateTime;
```

### Description

Set value of MinValue property sets the following settings MaxValue property. If you attempt to set the MaxValue property value greater than, it will be set to automatically value of MaxValue property.

In addition, the effective setting is the maximum 13-digit -9999999999999 ~ 9999999999999. The number of digits that can be input integer part, are influenced by the number of digits in the fractional part.

If the entered value exceeds the range, at the same time OnInvalidRange events subsequent input becomes impossible occurs.

However, MaxValue property, if the zero to MinValue property both are set, range check is not performed.

## P MinValue

InpText

InpNumber

InpDate

InpComboBox

LabeledText

LabeledNumber

LabeledDate

LabeledComboBox

MinValue property sets the minimum value of the date that can be entered into the control.

### Declaration

```
property MinValue: TDateTime;
```

### Description

Set value of MinValue property sets the following settings MaxValue property. If you attempt to set the MaxValue property value greater than, it will be set to automatically value of MaxValue property.

If the entered value exceeds the range, at the same time OnInvalidRange events subsequent input becomes impossible occurs.

However, MaxValue property, if the property MinValue both space (blank) is set, the range check is not performed.



## P NextByArrowKey

InpText

InpNumber

InpDate

InpComboBox

LabeledText

LabeledNumber

LabeledDate

LabeledComboBox

NextByArrowKey property, you can specify whether to allow the input focus movement of the control by the arrow keys.

### Declaration

```
property NextByArrowKey: Boolean;
```

### Description

NextByArrowKey property, [↑][↓] by (arrow keys), you can either perform the movement of the control, and specify whether or not.

值	意味
True	Moving it controls. (Default)
False	It is not carried out the transfer of control.

## P PasswordChar

InpText

InpNumber

InpDate

InpComboBox

LabeledText

LabeledNumber

LabeledDate

LabeledComboBox

PasswordChar property specifies the character to be displayed instead of the actual characters that are typed.

### Declaration

```
property PasswordChar: Char;
```

### Description

PasswordChar property, will create an edit control that displays a different character instead of the input character.

If you specify a PasswordChar the null character, edit control will directly display the text.

If PasswordChar is non-null character, edit control to display the character of PasswordChar instead of the character that has been input.

PasswordChar only affects the display of the edit control. The value of the Text property indicates the actual characters entered.

**Notice:** If ButtonStyle property is specified other than **bsNormal**, PasswordChar property is disabled.

## P ReturnNext

InpText

InpNumber

InpDate

InpComboBox

LabeledText

LabeledNumber

LabeledDate

LabeledComboBox

ReturnNext property, you can specify whether to allow the input focus movement of the control by the arrow keys.

### Declaration

```
property ReturnNext: Boolean;
```

### Description

ReturnNext property, by the [Enter] key, you can either perform the movement of the control, and specify whether or not.

Value	Description
True	Moving it controls. (Default)
False	Is not carried out the Moving of control.

## P SelectType

InpText

InpNumber

InpDate

InpComboBox

LabeledText

LabeledNumber

LabeledDate

LabeledComboBox

SelectType property specifies the selected range of text at the time of the focus acquisition.

### Declaration

```
type TSelectType = (stNon, stAll, stYear, stMonth, stDay);  
property SelectType: TSelectType;
```

### Description

SelectType property specifies the selected range of text at the time of the focus acquisition.

When Always enter the position constant, you can reduce the input load of the end-user by setting this property.

Value	Description
stAll	Select all. (Default)
stDay	select the day.
stMonth	select the month.
stYear	select the year.
stNon	Not selected.

## P SelfAdjustWidth

InpText

InpNumber

InpDate

InpComboBox

LabeledText

LabeledNumber

LabeledDate

LabeledComboBox

SelfAdjustWidth property, you specify whether to automatically adjust the width of the drop-down list.

### Declaration

```
property SelfAdjustWidth: Boolean;
```

### Description

ReturnNext property, by the [Enter] key, you can either perform the movement of the control, and specify whether or not.

Value	Description
True	Automatic adjust the control width. (Default)
False	It does not change the control width.

## P ShowComma

InpText

InpNumber

InpDate

InpComboBox

LabeledText

LabeledNumber

LabeledDate

LabeledComboBox

ShowComma property specifies the display format of the scale comma.

### Declaration

```
property ShowComma: Boolean;
```

### Description

ShowComma property, you can either display the scale comma every three digits, and specify whether or not.

Value	Description
True	We want to display a comma. (Default)
False	Do not show the comma.

## P Value

InpText

InpNumber

InpDate

InpComboBox

LabeledText

LabeledNumber

LabeledDate

LabeledComboBox

Value property to set the value in the control.

### Declaration

```
property Value: Double;
```

### Description

Possible settings in Value property is up to 13 digits of -9999999999999 ~ 9999999999999.

The number of digits that can be input integer part, are influenced by the number of digits in the fractional part.

If the entered value exceeds the 13 digits, at the same time OnInvalidRange events subsequent input becomes impossible occurs.

## P Value

InpText

InpNumber

InpDate

InpComboBox

LabeledText

LabeledNumber

LabeledDate

LabeledComboBox

Value property, and set the date to control.

### Declaration

```
property Value: TDateTime;
```

### Description

Value property, and set the date to control.

### Example of the use of the Value property

The following code sets the current date.

```
procedure TForm1.SpeedButton1Click(Sender: TObject);
begin
  InpDate1.Value := Date; // Now we want to set the date
end;
```

# P Value

InpText

InpNumber

InpDate

InpComboBox

LabeledText

LabeledNumber

LabeledDate

LabeledComboBox

Value property retrieves the value of the selected row of a drop-down list.

## Declaration

```
property Value[Index: Integer]: string;
```

## Description

Value property retrieves the value of the selected row of a drop-down list. **(Read-only)**

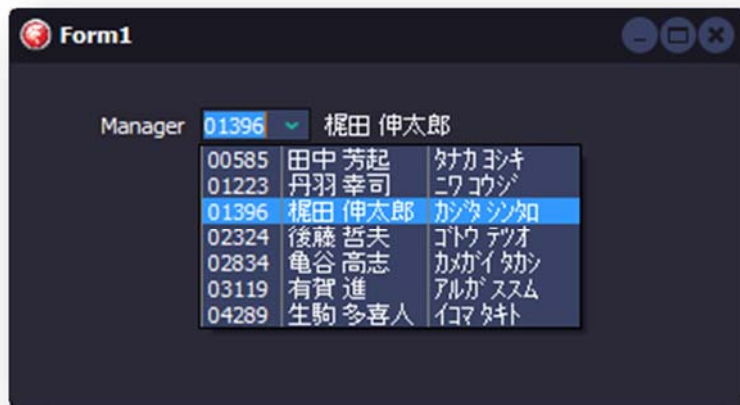
**Index:** Integer

It will specify the column number to retrieve. The value of the Index, the index value of the first of column index value of 0, the second column 1, also is assigned sequentially index thereafter.

## Example of the use of the Value property

The following code gets the value of the second column, in the drop-down list that is selected, and then displayed on the TLabel.

```
procedure TForm1.InpComboBox1Click(Sender: TObject);
begin
    Label1.Caption := InpComboBox1.Value[1];
end;
```



ValuePosition property specifies the column number at the time of drop-down list selection.

## Declaration

```
property ValuePosition: Integer;
```

## Description

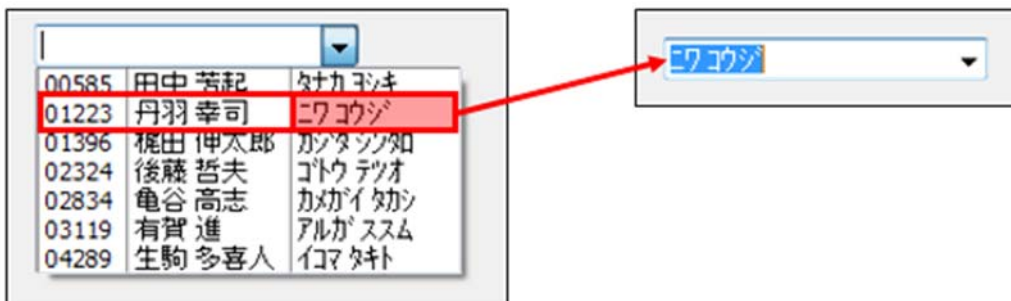
ValuePosition property specifies the column number at the time of drop-down list selection.

The value of ValuePosition, the index value of the first of column index value of 0, the second column 1, also is assigned sequentially index thereafter. The maximum value of ValuePosition is the "ColumnCount-1".

## Example of the use of ValuePosition property

The following code sets the item in the drop-down list.

```
procedure TForm1.FormCreate(Sender: TObject);
begin
  // Clear of the drop-down list
  InpComboBox1.ListValue.Clear;
  // Set of list items
  InpComboBox1.ListValue.Add('00585, 田中 芳起, タカヨシ');
  InpComboBox1.ListValue.Add('01223, 丹羽 幸司, ニワコウジ');
  :
  InpComboBox1.ListValue.Add('04289, 生駒 多喜人, イマタキ');
  // Setting Properties
  InpComboBox1.ColumnCount := 3;
  InpComboBox1.ValuePosition := 2;
end;
```



P

# Version

InpText

InpNumber

InpDate

InpComboBox

LabeledText

LabeledNumber

LabeledDate

LabeledComboBox

Version property, version information of SuperEDIT is displayed. **(Read-only)**

## Declaration

```
property Version: string;
```

## Description

Version property, version information of SuperEDIT is displayed. **(Read-only)**

# P WidthOnFocus

InpText

InpNumber

InpDate

InpComboBox

LabeledText

LabeledNumber

LabeledDate

LabeledComboBox

WidthOnFocus property specifies the width of the control when the control receives focus in pixels.

## Declaration

```
property WidthOnFocus: Integer;
```

## Description

With WidthOnFocus property, you can change the width of when the control receives focus.

The contents of the long string that does not fit in the control by specifying this property is also easily input, you can display.

**Notice:** The default value, it is 0 (do not change the size).

When it loses focus



When there is a focus





## P ZeroAllowed

InpText

InpNumber

InpDate

InpComboBox

LabeledText

LabeledNumber

LabeledDate

LabeledComboBox

ZeroAllowed property specifies whether to allow the input of zero.

### Declaration

```
property ZeroAllowed: Boolean;
```

### Description

ZeroAllowed property, if the value in the control is 0, and specify whether to forcibly enter a numeric value other than 0.

If ZeroAllowed property is set to False, you can not input of 0. This way, you can be prompted to enter a numeric value other than 0 to the user.

Value	Description
True	It can be zero input. (Default)
False	Zero input disabled.

## P ZeroDisplay

InpText

InpNumber

InpDate

InpComboBox

LabeledText

LabeledNumber

LabeledDate

LabeledComboBox

ZeroDisplay property specifies whether the display of the 0.

### Declaration

```
property ZeroDisplay: Boolean;
```

### Description

ZeroDisplay property, if the value in the control is 0, and specify whether to display the 0.

Value	Description
True	0 displayed.
False	0 Do not show. (Default)

## 12. Description of custom event

### E OnChangeValue

InpText

InpNumber

InpDate

InpComboBox

LabeledText

LabeledNumber

LabeledDate

LabeledComboBox

OnChangeValue event occurs when the value of the control has been changed.

The presence or absence of change are checked at the time the control loses the input focus.

#### Declaration

```
type TNotifyEvent = procedure (Sender: TObject) of object;  
property OnChangeValue: TNotifyEvent;
```

#### Description

OnChangeValue event occurs when the value of the control has been changed.

This by using the event procedure, you can grasp the timing of the re-set of values from the check processing and table of the entered value.

### E OnEllipsisClick

InpText

InpNumber

InpDate

InpComboBox

LabeledText

LabeledNumber

LabeledDate

LabeledComboBox

OnEllipsisClick event occurs when the search button is pressed.

#### Declaration

```
type TNotifyEvent = procedure (Sender: TObject) of object;  
property OnEllipsisClick: TNotifyEvent;
```

#### Description

OnEllipsisClick event occurs when the search button is pressed.

#### OnEllipsisClick example of the use of events

The following code, When the search button is pressed, to display the search screen.

```
procedure TForm1.InpText1EllipsisClick (Sender: TObject) ;  
begin  
  Form2 := TForm2.Create(Application);  
  Form2.ShowModal ;  
  Form2.Release;  
end;
```

# E OnInsModeChanged

InpText

InpNumber

InpDate

InpComboBox

LabeledText

LabeledNumber

LabeledDate

LabeledComboBox

OnInsModeChanged event occurs when you switch the input mode (insert / overwrite).

## Declaration

```
type TInsModeChangeEvent = procedure (Sender: TObject; State: Boolean) of object;  
property OnInsModeChanged: TInsModeChangeEvent;
```

## Description

OnInsModeChanged event, when you press the [Insert] key on the keyboard, or occurs when you change the value of InsertMode property.

State parameter indicates the state of the input mode.

Value	Description
True	Insert mode
False	Overwrite mode

## OnInsModeChanged example of the use of events

The following code, when the input mode switches, and displays the current input mode on the status bar.

```
procedure TForm1.InpText1InsModeChanged (Sender: TObject; State: Boolean) ;  
begin  
  if InpText1.InsertMode then  
    StatusBar1.Panels[1].Text := 'Insert'  
  else  
    StatusBar1.Panels[1].Text := 'Overwrite'  
end;
```

# E OnInvalidEntry

InpText

InpNumber

InpDate

InpComboBox

LabeledText

LabeledNumber

LabeledDate

LabeledComboBox

OnInvalidEntry event occurs when you press the key the user is wrong.

## Declaration

```
type TNotifyEvent = procedure (Sender: TObject) of object;  
property OnInvalidEntry: TNotifyEvent;
```

## Description

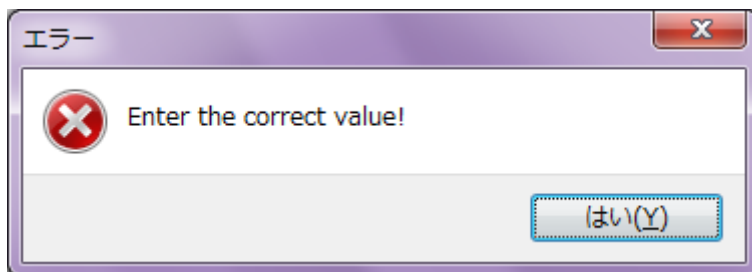
OnInvalidEntry event occurs when an invalid key is entered.

When you use this event procedure to display a message box or dialog box, you will be able to inform the mistake to the user.

## OnInvalidEntry example of the use of events

The following code displays a message box when the wrong key is entered.

```
procedure TForm1.InpText1InvalidEntry (Sender: TObject) ;  
begin  
  MessageDlg('Enter the correct value!', mtError, [mbYes], 0);  
end;
```



# E OnInvalidRange

InpText

InpNumber

InpDate

InpComboBox

LabeledText

LabeledNumber

LabeledDate

LabeledComboBox

OnInvalidRange event occurs when an attempt is made to input or set, out-of-range values.

## Declaration

```
type TNotifyEvent = procedure (Sender: TObject) of object;  
property OnInvalidRange: TNotifyEvent;
```

## Description

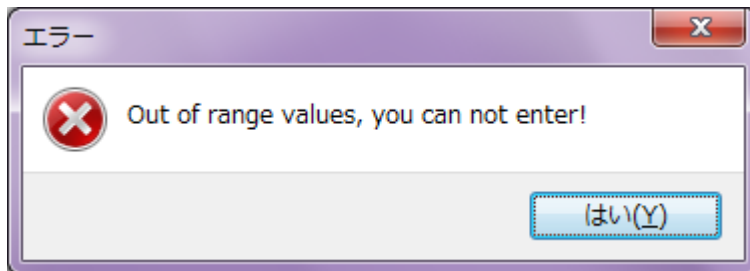
OnInvalidRange event occurs when an attempt is made to input or set, out-of-range values.

By using this event procedure, displays a message box or dialog box, you can be notified of the mistake to the user.

## OnInvalidRange example of the use of events

The following code displays a message box when you try to set the wrong value.

```
procedure TForm1.InpNumber1InvalidRange (Sender: TObject) ;  
begin  
  MessageDlg('Out of range values, you can not enter!', mtError, [mbYes], 0);  
end;
```



OnPopup event, the drop button (or search button) when you press, or occurs when you have to display the pop-up calculator.

## Declaration

```
type TNotifyEvent = procedure (Sender: TObject) of object;  
property OnPopup: TNotifyEvent;
```

## Description

OnPopup event occurs when you have to display the pop-up calculator.

## 13. Description of custom methods / functions

M

# DoPopup

InpText

InpNumber

InpDate

InpComboBox

LabeledText

LabeledNumber

LabeledDate

LabeledComboBox

DoPopup method displays a pop-up calculator to the specified position.

### Declaration

```
procedure DoPopup(Top, Left: Integer);
```

**Top:** Integer

Specifies the Y coordinate of the screen to display the calculator in pixels.

**Left:** Integer

The X coordinate of the screen to display the calculator, specified in pixels.

### Description

DoPopup method is used when you want to freely specify the call keys and display position.

OnPopup event occurs every time the pop-up calculator is called.

### Example of the use of DoPopup method

The following code, When is pressed [Pause] key, to display the calculator under the control.

```
procedure TForm1.InpNumber1KeyDown(Sender: TObject; var Key: Word;
  Shift: TShiftState);
var
  CX, CY: Integer;
begin
  if Key = VK_PAUSE then
  begin
    CX := TInpNumber(Sender).ClientToScreen(Point(0, 0)).X;
    CY := TInpNumber(Sender).ClientToScreen(Point(0, InpNumber(Sender).Height)).Y;
    InpNumber1.DoPopup(CY, CX);
  end;
end;
```

TaxCalc method to calculate the tax.

## Declaration

```
procedure TaxCalc (Percentage, CalcType, RoundType: Integer) ;
```

### Percentage: Integer

It will specify the tax rate. (0-500)

### CalcType: Integer

It will specify the Tax included and Tax-exclusive price.

0 : Tax included

1 : Tax-exclusive price

### RoundType: Integer

It will specify the rounding method.

0 : Rounding

1 : Rounded up

2 : Truncation

## Description

When you run the TaxCalc method to calculate the consumption tax by the value of the Value property of InpNumber control, the AsPrice property price, AsTax property tax, AsTotalPrice property returns the tax-included price respectively.

## Example of the use of TaxCalc method

The following code, call the TaxCalc method commodity price (Label1), tax (Label2), to get the tax-included price (Label3).

```
procedure TForm1.Button1Click (Sender: TObject) ;
begin
  // Consumption tax 8 percent, tax, rounding
  InpNumber1.TaxCalc (8, 0, 0) ;
  // Commodity prices (without tax price)
  Label1.Caption := FormatCurr ('#,###.####', InpNumber1.AsPrice) ;
  // consumption tax
  Label2.Caption := FormatCurr ('#,###.####', InpNumber1.AsTax) ;
  // Tax-inclusive price
  Label3.Caption := FormatCurr ('#,###.####', InpNumber1.AsTotalPrice) ;
end;
```



It returns the specified year, month, date type that represents the date that corresponds to the day the (TDateTime).

## Declaration

```
function DateSerial (y, m, d: Integer): TDateTime;
```

**y:** Integer

It specifies the seek year. (1 to 9999)

**m:** Integer

Specify the month to seek. (0 to 12)

**d:** Integer

Specify the day you want. (0 to 31)

## Description

It returns the specified year, month, date type that represents the date that corresponds to the day the (TDateTime).

It provides the same function as DateSerial function of Visual Basic.

## Example of the use of the DateSerial function

The following code, the year that has been input (LabeledNumber1), month (LabeledNumber2), and displays it in LabeledDate1 determine the value of the date type (TDateTime) from day (LabeledNumber3).

```
procedure TForm1.Button1Click (Sender: TObject) ;
var
  yy, mm, dd: Integer;
  dt: TDateTime;
  str: String;
begin
  yy := LabeledNumber1.AsInteger;
  mm := LabeledNumber2.AsInteger;
  dd := LabeledNumber3.AsInteger;
  dt := DateSerial (yy, mm, dd);
  LabeledDate1.Value := dt;
end;
```

It returns the number of days in the specified month.

## Declaration

```
function GetDayCount(Date: TDateTime): Integer;
```

**Date:** TDateTime

Specify the date you want.

## Description

It returns the number of days in the specified month.

## Example of the use of GetDayCount function

The following code, from the input date (LabeledDate1), to display the number of days in the month.

```
procedure TForm1.Button1Click(Sender: TObject);  
var  
    dc: Integer;  
begin  
    dc := GetDayCount(LabeledDate1.Value);  
    MessageDlg(IntToStr(dc), mtInformation, [mbYes], 0);  
end;
```

## F GetAge

Returns age.

### Declaration

```
function GetAge(DateOfBi rth, DateOfObj ect: TDateTime): Integer;
```

**DateOfBirth:** TDateTime

Specify the date of birth in the AD.

**DateOfObject:** TDateTime

Specify the AD the target day to find the age.

### Description

Specify the date of birth and date of retirement, returns the age.

### Example of the use of GetAge function

The following code, from the date of birth (LabeledDate1) and the target day (LabeledDate2), to display asked the age.

```
procedure TForm1.Button1Click (Sender: TObject) ;  
var  
    age: Integer;  
begin  
    age := GetAge(LabeledDate1.Value, LabeledDate2.Value);  
    MessageDlg(IntToStr(age), mtInformation, [mbYes], 0);  
end;
```

It returns the last day of the month.

## Declaration

```
function GetLastDay(Date: TDateTime): TDateTime;
```

**Date:** TDateTime

Specify the date type (TDateTime) the target day.

## Description

Specify a target date, returns the last day of the month of the month.

## Example of the use of GetLastDay function

The following code, specify the target date (LabeledDate1), to display asked the last day of the month of the month.

```
procedure TForm1.Button1Click(Sender: TObject);  
var  
    dt: TDateTime;  
    str: String;  
begin  
    dt := GetLastDay(LabeledDate1.Value);  
    DateTimeToString(str, 'yyyy-MM-dd', dt);  
    MessageDlg(str, mtInformation, [mbYes], 0);  
end;
```

It returns the number of days elapsed.

## Declaration

```
function GetPassingDate(Source, Target: TDateTime): Integer;
```

**Source:** TDateTime

The target date (start) Specify the date type (TDateTime).

**Target:** TDateTime

The target date (the end) specified by the date type (TDateTime).

## Description

The target date (start, end) from, and returns the number of days elapsed.

## Example of the use of GetPassingDate function

The following code, from the start date (LabeledDate1) and end date (LabeledDate2), to display asked the number of days elapsed.

```
procedure TForm1.Button1Click(Sender: TObject);  
var  
    pd: Integer;  
begin  
    pd := GetPassingDate(LabeledDate1.Value, LabeledDate2.Value);  
    MessageDlg(IntToStr(pd), mtInformation, [mbYes], 0);  
end;
```

The relevant date, returns the week.

## Declaration

```
function GetWeekLine(Date: TDateTime): Integer;
```

**Date:** TDateTime

Specify the date type (TDateTime) the target day.

## Description

The relevant date, returns the week.

## Example of the use of GetWeekLine function

The following code, specify the target date (LabeledDate1), to display asked the week.

```
procedure TForm1.Button1Click(Sender: TObject);  
var  
    wl: Integer;  
begin  
    wl := GetWeekLine(LabeledDate1.Value);  
    MessageDlg(IntToStr(wl), mtInformation, [mbYes], 0);  
end;
```

## F GetSeiza

It returns the constellation of the target date.

### Declaration

```
function GetSeiza(Target: TDateTime): String;
```

**Target:** TDateTime

Specify the date type (TDateTime) the target day.

### Description

It returns the constellation of the target date.

### Example of the use of GetSize function

The following code, specify the target date (LabeledDate1), to display the constellation.

```
procedure TForm1.Button1Click(Sender: TObject);  
var  
    Seiza: String;  
begin  
    Seiza := GetSeiza(LabeledDate1.Value);  
    MessageDlg(Seiza, mtInformation, [mbYes], 0);  
end;
```

## 14. Use a calculator

InpNumber / LabeledNumber components, you can use the calculator on input.

Numeric value that has been input is passed through to the calculator.

Also, results of the calculation are set to call control.

### 11.1 Start

The call method of the calculator, there are two ways of next.

·[ButtonStyle] the property, [bsAlways] or, is set to [bsOnFocus], then click the drop-down button at run time.

·[F4] key or pressing the [Alt] + [↓] (down arrow).

### 11.2 End

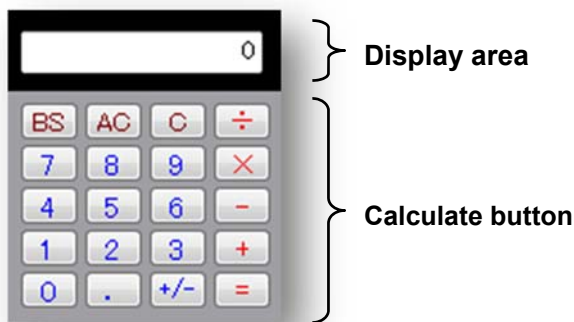
Calculator is finished by pressing the [End] key, control is passed to the called control (InpNumber / LabeledNumber).

At this time, the calculation results will be passed at the same time results if other than zero.

Also, press the [Esc] key, you will be able to terminate the calculator when you click on the non-calculator, the result is not passed. (Cancel)

### 11.3 Names of each part

Calculator displays the contents and calculation results input "display area" and the (top), to specify the input and calculation method of the number, farewell to the "calculation button" (bottom).





## 11.4 Correspondence table of buttons and keyboard

It is also possible to input directly from the keyboard in addition to the button.

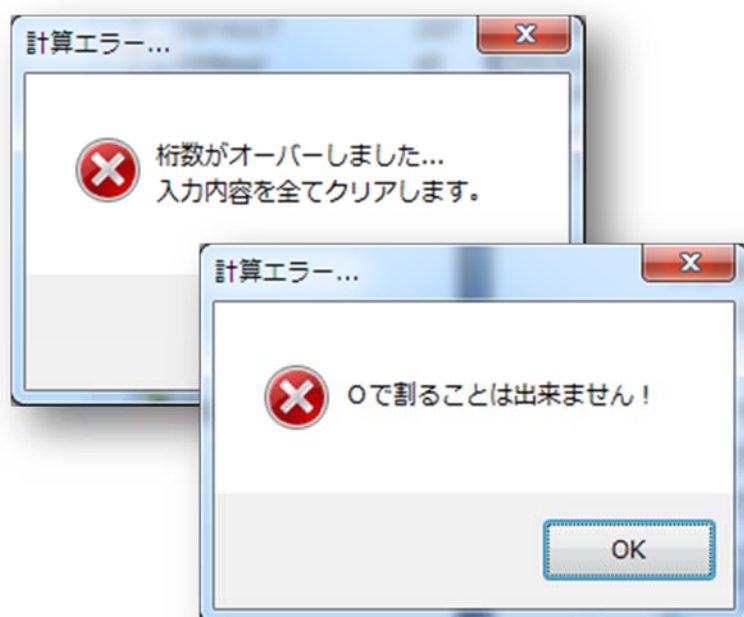
Button name	Corresponding key
Numerical buttons (0-9)	0~9
Sign switching button (+/-)	F9
Point button	. (Depending on the setting of the control panel)
Calculation instruction button ( $\div$ )	/
(X)	*
(-)	-
(+)	+
(=)	Enter
Control button (B S)	Backspace
(A C)	A
(C)	Delete

## 11.5 Correction method

- Press mistake of the number buttons, press the [C (Clear)] button. If able to re correctly numeric and continue, it will be able to continue as it is calculated.
- Press mistake of calculation instruction button, press the correct calculation instruction button to continue. It will be corrected in the calculation instruction button pressed later.

## 11.6 Error checking

When the integer part exceeds the 13-digit, when the division of the divisor 0, a message box is displayed, you will not be able to the subsequent calculation. (Calculated up to it, all will be cleared)



## 15. Use the drop-down calendar

InpDate / LabeledDate components, will be able to use the calendar.

The date that has been input is passed through to the calendar.

Also, it is set to control that call the selected date.

### 12.1 Start

The call method of the calendar, there are two ways of next.

·[ButtonStyle] the property, [bsAlways] or, is set to [bsOnFocus], then click the drop-down button at run time.

·[F4] key or pressing the [Alt] + [↓] (down arrow).

### 12.2 End

[Enter] or press the [End] key, or calendar will close when you select a date with the mouse.

### 12.3 Names of each part

The calendar, you can see the 1-month calendar of the selected date.

Movement of the display month, [PageUp] or [PageDown] key, move the date, is carried out with the arrow keys, the selection done in the [Enter] (or [End]) key or mouse.

It should be noted that the value of the control that was called the date changes (InpDate / LabeledDate) also changes at the same time.

11月 2015						
日	月	火	水	木	金	土
1	2	3	4	5	6	7
8	9	10	11	12	13	14
15	16	17	18	19	20	21
22	23	24	25	26	27	28
29	30	1	2	3	4	5
6	7	8	9	10	11	12

### 12.4 Change Display month

·[PageUp] or [PageDown]: to move the display month.

·Click on the calendar title: I can move to any of the years. (See figure below)

2015/11						
日	月	火	水	木	金	土
1	2	3	4	5	6	7
8	9	10	11	12	13	14
15	16	17	18	19	20	21
22	23	24	25	26	27	28
29	30	1	2	3	4	5
6	7	8	9	10	11	12

8月 2015  
9月 2015  
10月 2015  
11月 2015  
12月 2015  
1月 2016  
2月 2016

## 16. User support

Bug reports, requests, inquiries, etc., I hope in e-mail.

E-mail address: yoshiki.tanaka-avsoft@nifty.com

If the failure occurs in SuperEDIT it is,

- usage environment
- Usage state
- Key operation
- Mouse operation
- The presence or absence of reproducibility
- Defect Details

If you can report, it will help solve.

### **[Please]**

Support for the bug will proactively deal, but sometimes it is not possible to respond quickly due to circumstances of the environment or the like for possession.

Also, version-up by function addition, we do not assume the Company its performance obligation.

---

# SuperEDIT

## **Reference Manual**

Version 1.0 (Jun 2016)

Distributor : On And On Corp.

<http://on-and-on.biz/en>

Developer : Adventure software

<http://www.avsoft.jp/>