

WinFormsX - Button

Properties

| Name | Description |
|-----------|--------------------------------------------------------------------------------------------------------------------------------------------------|
| CtrlType | Gets or sets the control type value. Always ControlType.Button and used when adding control to its form's controls collection. |
| TextAlign | Gets or sets a value indicating the alignment of the text on a control. Refer to ButtonAlignment enum. |
| Enabled | Gets or sets a value (true/false) indicating whether the control can respond to user interaction. |
| Focused | Gets a value (true/false) indicating whether the control has input focus. |
| Height | Gets or sets the height of the control. |
| hWindow | Gets the Windows handle (hwnd) of the control. |
| Left | Gets or sets the distance, in pixels, between the left edge of the control and the left edge of its container's client area (normally the form). |
| Name | Gets or sets the name of the control. |
| Parent | Gets or sets the parent container of the control. |
| Position | Gets or sets the top and left position of the control relative to the form. Get: returns wfxPoint object Set: (left, top) |
| Size | Gets or sets the size of the control. Get: returns wfxSize object. Set: (width, height) |
| TabStop | Gets or sets a value (true/false) indicating whether the user can give the focus to this control using the TAB key |
| Text | Gets or sets the text (caption) associated with this control. |
| Top | Gets or sets the distance, in pixels, between the top edge of the control and the top edge of its container's client area (normally the form). |
| Visible | Gets or sets a value (true/false) indicating whether the control is displayed. |
| Width | Gets or sets the width of the control. |

Methods

| Name | Description |
|-----------|-----------------------------------------------------------------------------------------------|
| Hide | Conceals the control from the user. |
| Refresh | Forces the control to invalidate its client area and immediately redraw itself |
| SetBounds | Sets the bounds of the control to the specified location and size. (left, top, width, height) |
| Show | Makes the control visible to the user (and creates the control should it not already exist) |

Events

| Name | Description |
|--------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| OnAllEvents | Special handler where all events are routed through. Use this handler if you prefer to use the Win32 api style messages and wParam and lParam parameters. Set the <i>Handled</i> element of <i>EventArgs</i> to true if you handle a message and do not want Windows to perform any further processing on the message. |
| OnClick | Occurs when the client area of the control is clicked. |
| OnDestroy | Occurs immediately before the control is about to be destroyed and all resources associated with it released. |
| OnGotFocus | Occurs when the control gains input focus. |
| OnLostFocus | Occurs when the control loses input focus. |
| OnKeyDown | Occurs when a key is pressed while the control has focus. |
| OnKeyPress | Occurs when a character, space or backspace key is pressed while the control has focus. |
| OnKeyUp | Occurs when a key is released while the control has focus. |
| OnMouseDoubleClick | Occurs when the control is double clicked by the mouse. |
| OnMouseDown | Occurs when the mouse pointer is over the control and a mouse button is pressed. |
| OnMouseEnter | Occurs when the mouse pointer enters the control. |
| OnMouseHover | Occurs when the mouse pointer rests on the control. |
| OnMouseLeave | Occurs when the mouse pointer leaves the control. |
| OnMouseMove | Occurs when the mouse pointer is moved over the control. |
| OnMouseUp | Occurs when the mouse pointer is over the control and a mouse button is released. |

Button Alignment Enum

```
enum ButtonAlignment
    BottomCenter = BS_CENTER OR BS_BOTTOM
    BottomLeft   = BS_LEFT   or BS_BOTTOM
    BottomRight  = BS_RIGHT  or BS_BOTTOM
    MiddleCenter = BS_CENTER OR BS_VCENTER
    MiddleLeft   = BS_LEFT   or BS_VCENTER
    MiddleRight  = BS_RIGHT  or BS_VCENTER
    TopCenter    = BS_CENTER or BS_TOP
    TopLeft      = BS_LEFT   or BS_TOP
    TopRight     = BS_RIGHT  or BS_TOP
end enum
```