

ZylVscEditor

All Units

| Name | Description |
|----------------------------|-----------------------------------|
| ZylVscEditor | Copyright by Zyl Soft 2003 - 2025 |
| Copyright by Zyl Soft 2024 | |

ZylVscEditor

Generated by *PasDoc 0.16.0*.

Unit ZylVscEditor

Description

Uses

Classes, Interfaces, Objects and Records

Functions and Procedures

Types

Constants

Variables

Description

Copyright by Zyl Soft 2003 - 2025

Overview

Classes, Interfaces, Objects and Records

| Name | Description |
|----------------------------|---|
| Class TZylVscEditor | TZylVscEditor is a component to integrate Visual Studio Code IDE in your application. |

Functions and Procedures

```
procedure Register;
```

Types

```
TEditorMode = (...);  
TEditorContentChangeEvent = procedure(Sender: TObject; content: string) of object;  
TEditorLanguageChangeEvent = procedure(Sender: TObject; oldLanguage: string) of object;  
TEditorMouseEvent = procedure(Sender: TObject; Button: TMouseButton; Shift: TShiftState; X,  
Y: Integer) of object;  
TEditorKeyEvent = procedure(Sender: TObject; Key: Word; Shift: TShiftState) of object;
```

Description

Functions and Procedures

```
procedure Register;
```

Types

```
TEditorMode = (...);  
Editor mode enum. It can be simple editor or compare editor.
```

Values

- emEditor
- emCompare

```
TEditorContentChangeEvent = procedure(Sender: TObject; content: string) of object;  
Content change event type.
```

| |
|---|
| <div><div>Parameters</div><div><div>content</div><div>New content of the editor.</div></div></div> |
| <div><div><div><div>TEditorLanguageChangeEvent = procedure(Sender: TObject; oldLanguage: string) of object;</div><div>Language change event type.</div></div><div><div>Parameters</div><div><div>oldLanguage</div><div>Previous language.</div></div></div></div></div> |
| <div><div><div><div>TEditorMouseEvent = procedure(Sender: TObject; Button: TMouseButton; Shift: TShiftState; X, Y: Integer) of object;</div><div>Mouse event type.</div></div><div><div>Parameters</div><div><div>Button</div><div>Pressed mouse button.</div><div>Shift</div><div>Control keys state.</div><div>X</div><div>Horizontal position of the cursor.</div><div>Y</div><div>Vertical position of the cursor.</div></div></div></div></div> |
| <div><div><div><div>TEditorKeyEvent = procedure(Sender: TObject; Key: Word; Shift: TShiftState) of object;</div><div>Key event type.</div></div><div><div>Parameters</div><div><div>Key</div><div>Pressed key code.</div><div>Shift</div><div>Control keys state.</div></div></div></div></div> |

Copyright by Zyl Soft 2024

Class TZylVscEditor

- Description
- Hierarchy
- Fields
- Methods
- Properties

Unit

ZylVscEditor

Declaration

```
type TZylVscEditor = class(TEdgeBrowser)
```

Description

TZylVscEditor is a component to integrate Visual Studio Code IDE in your application.











Hierarchy

- TEdgeBrowser
- TZylVscEditor

Overview

Methods







```

 P constructor Create(aOwner: TComponent); override;
 P procedure AfterConstruction(); override;
 P destructor Destroy(); override;
 P function OpenFile(strFileName: string): boolean;
 P procedure CloseFile();
 P procedure NewFile(strLanguage: string = '');
 P function SaveFile(strFileName: string): boolean; overload;
 P function SaveFile(): boolean; overload;
 P function CompareFile(strOriginalFileName: string): boolean;
 P procedure SetReadOnly(readOnly: Boolean = true);

```

Properties

```

 P property Content: string read fContent write SetContent;
 P property OriginalContent: string read fOriginalContent write fOriginalContent;
 P property ModifiedContent: string read fContent write fContent;
 P property SelectedContent: string read fSelectedContent;
 P property UnlockKey: AnsiString read FUnlockKey write FUnlockKey;
 P property Version: Double read FVersion;


property FileName: string read fFileName write fFileName;
property OriginalFileName: string read fOriginalFileName write fOriginalFileName;
property Mode: TEditorMode read fMode write SetMode;
property InlineCompare: boolean read fInlineCompare write SetInlineCompare default false;
property Language: string read fLanguage write fLanguage;
property Theme: string read fTheme write SetTheme;
property OnEditorContentChange: TEditorContentChangeEvent read FOnEditorContentChange write FOnEditorContentChange;
property OnEditorLanguageChange: TEditorLanguageChangeEvent read FOnEditorLanguageChange write FOnEditorLanguageChange;
property OnEditorMouseUp: TEditorMouseEvent read FOnEditorMouseUp write FOnEditorMouseUp;
property OnEditorMouseMove: TEditorMouseEvent read FOnEditorMouseMove write FOnEditorMouseMove;
property OnEditorMouseDown: TEditorMouseEvent read FOnEditorMouseDown write FOnEditorMouseDown;
property OnEditorKeyUp: TEditorKeyEvent read FOnEditorKeyUp write FOnEditorKeyUp;
property OnEditorKeyDown: TEditorKeyEvent read FOnEditorKeyDown write FOnEditorKeyDown;

```

Description

Methods


```

 P constructor Create(aOwner: TComponent); override;

```

Constructor.

```

 P procedure AfterConstruction(); override;

```


After construction event.

```

 P destructor Destroy(); override;

```

Destructor.

```
 P function OpenFile(strFileName: string): boolean;
```


Open file in the editor.

Parameters


strFileName
Name of the file.

Returns

true, if succeeded.

```
 P procedure CloseFile();
```


Close the file from the editor.

```
 P procedure NewFile(strLanguage: string = '');
```

Close the file from the editor to edit a new one.

Parameters

strLanguage
Programming language of the new file.

```
 P function SaveFile(strFileName: string): boolean; overload;
```


Save the file.

Parameters

strFileName
Name of the file.

Returns


true, if succeeded.

```
 P function SaveFile(): boolean; overload;
```

Save the file from the editor.

Returns

true, if succeeded.

```
 P function CompareFile(strOriginalFileName: string): boolean;
```


Compare the current file with another file.

Parameters

strOriginalFileName
Another file path.

Returns

true, if succeeded.


```
 P procedure SetReadOnly(readOnly: Boolean = true);
```

Set editor read-only.


Parameters

readOnly
Read-only if true.


Properties

```
 P property Content: string read fContent write SetContent;
```


Content of the editor.

```
 P property OriginalContent: string read fOriginalContent write fOriginalContent;
```


Original content of compare editor.

```
 P property ModifiedContent: string read fContent write fContent;
```


Modified content of compare editor. Same as Content.

```
 P property SelectedContent: string read fSelectedContent;
```

Selected content.

```
 P property UnlockKey: AnsiString read FUnlockKey write FUnlockKey;
```

Unlock key.

```
 P property Version: Double read FVersion;
```

Component version.

```
property FileName: string read fFileName write fFileName;
```

Path of the file to edit.

```
property OriginalFileName: string read fOriginalFileName write fOriginalFileName;
```

Path of the original file to compare with.

```
property Mode: TEditorMode read fMode write SetMode;
```

Editor mode. It can be simple editor or compare editor.

```
property InlineCompare: boolean read fInlineCompare write SetInlineCompare default false;
```

Inline compare.

```
property Language: string read fLanguage write fLanguage;
```

Programming language name.

```
property Theme: string read fTheme write SetTheme;
```

Name of the theme.

<example>vs-dark</example>

```
property OnEditorContentChange: TEditorContentChangeEvent read FOnEditorContentChange  
write FOnEditorContentChange;
```

Content change event.

```
property OnEditorLanguageChange: TEditorLanguageChangeEvent read FOnEditorLanguageChange  
write FOnEditorLanguageChange;
```

Language change event.

```
property OnEditorMouseUp: TEditorMouseEvent read FOnEditorMouseUp write FOnEditorMouseUp;
```

Mouse up event.

```
property OnEditorMouseMove: TEditorMouseEvent read FOnEditorMouseMove write  
FOnEditorMouseMove;
```

Mouse move event.

```
property OnEditorMouseDown: TEditorMouseEvent read FOnEditorMouseDown write  
FOnEditorMouseDown;
```

Mouse down event.

```
property OnEditorKeyUp: TEditorKeyEvent read FOnEditorKeyUp write FOnEditorKeyUp;
```

Key up event.

```
property OnEditorKeyDown: TEditorKeyEvent read FOnEditorKeyDown write FOnEditorKeyDown;
```

Key down event.

Copyright by Zyl Soft 2024

Generated by *PasDoc 0.16.0*.