

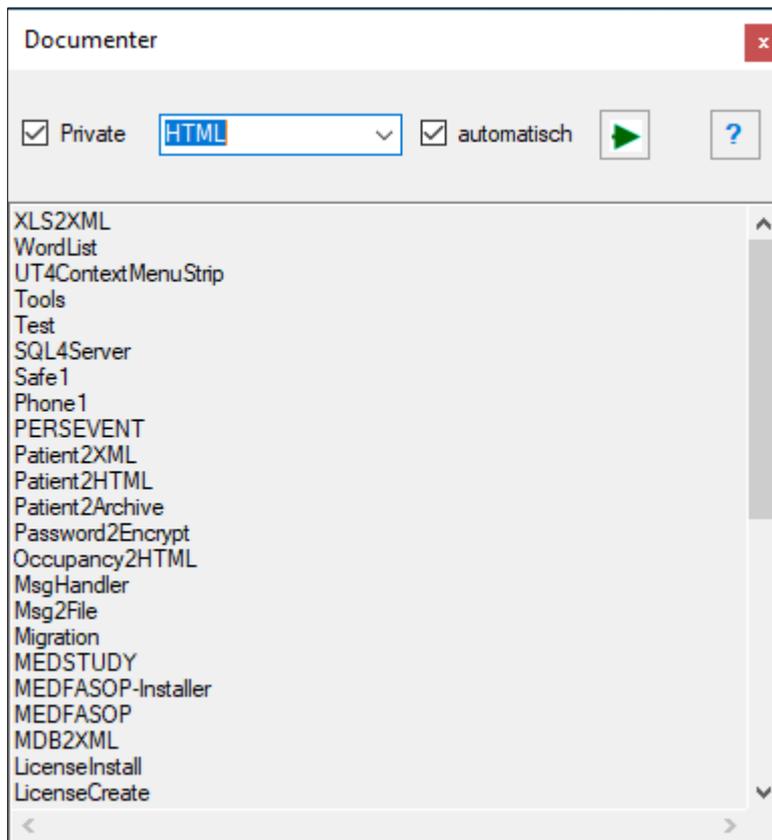


Die Anwendung extrahiert aus VB-Programmen Elemente (Module, Class, Function, Sub, Structure, Enum) und schreibt sie in den Ordner, welcher in der Datei "Documenter.txt" im Programmordner mit "output=" festgelegt ist. Ist [Private] aktiviert werden nebst «Public» auch «Private» deklarierte Elemente ausgewertet.

Ausgewertet werden zudem eingebundene Dynamic Link Librarys (DLL) ex «Verweise» im VS2017-Projekt.

Die Auswertung erfolgt wahlweise in HTML- oder Text-Dateien. HTML beinhaltet zudem den Link zum DOC-Ordner des Projekts<sup>1</sup> und den Programmdateien<sup>2</sup>.

Ist [automatisch] aktiviert, wird der Startordner, welcher in der Datei "Documenter.txt" im Programmordner mit "input=" festgelegt ist, ausgewertet. Andernfalls kann der Benutzer auswählen.



Mit  beginnt das Prozedere, welche alle Dateien mit Endung ".vb" im, und unterhalb des Start-Ordners, verarbeitet.

Identifizierte Projekte werden laufend und absteigend aufgelistet.

Mit  wird dieses Dokument im PDF-Format angezeigt.

Auf Seite 2 ist ein HTML-Ergebnis abgebildet.



Die Anwendung «WordList» erstellt eine Auflistung in den Programmen verwendeter Worte.

<sup>1</sup> PDF-Dokumente werden direkt angezeigt, andere nur über den Download.

<sup>2</sup> VB-Dateien werden direkt im Lesemodus angezeigt.

```

Adjustment4Patient x +
C:/documenter/Documenter/Adjustment4Patient.html
Apps ecofor hosttech hosttech CP Swisscom Google Kontakte Google Kalender Google Übersetzer

DOC=Adjustment4Patient

Eingebundene DLL=ADO2NET, BTree, LicenseCheck, Password2Encrypt, Tools

Form1.vb
Form1 :: Public Class Form1
| Ask4Password :: Private Function Ask4Password(tb As String, pwn As String) As String
| BReplace_Click :: Private Sub BReplace_Click(sender As Object, e As EventArgs) Handles BReplace.Click
| BStart_Click :: Private Sub BStart_Click(sender As System.Object, e As System.EventArgs) Handles BStart.Click
| ClearPassword :: Private Function ClearPassword(s As String) As String
| Extract :: Private Function Extract(ByRef ss As String, fs As String) As String
| Form1_Load :: Private Sub Form1_Load(sender As System.Object, e As System.EventArgs) Handles MyBase.Load
| GetDate :: Private Function GetDate(cs As String) As String
| GetUID :: Private Function GetUID() As String
| Kill4Password :: Private Sub Kill4Password(pwn As String)
| PDF_Click :: Private Sub PDF_Click(sender As System.Object, e As System.EventArgs) Handles PDF.Click
| Put4Password :: Private Function Put4Password(tb As String, pwn As String) As Boolean
| SetDate :: Private Sub SetDate(cs As String, cdt As String)
| Test4ActiveUser :: Private Function Test4ActiveUser(cs As String) As Boolean
| Test4Connect :: Private Function Test4Connect(cs As String, pwn As String) As Boolean
| Test4Path :: Private Function Test4Path(ps As String) As Boolean

Module1.vb
Module1 :: Module Module1
| Action4Row :: Private Function Action4Row(cs As String, sql As String, udt As Date, adt As Date) As myAction
| AllExOriginal2Copy :: Public Sub AllExOriginal2Copy()
| cKey [Inherits BTree2.Data] :: Public Class cKey
| | Compare :: Public Overrides Function Compare(ByRef d As BTree.BTree2.Data) As Short
| cProtocol [Inherits BTree2.Data] :: Public Class cProtocol
| | Compare :: Public Overrides Function Compare(ByRef d As BTree.BTree2.Data) As Short
| | New :: Public Sub New(prole As myRole, ptable As String)
| cRow [Inherits BTree2.Data] :: Public Class cRow
| | Compare :: Public Overrides Function Compare(ByRef d As BTree.BTree2.Data) As Short
| Documents :: Private Sub Documents(path1 As String, path2 As String, role As myRole, adt As Date)
| ExecuteSQL :: Private Sub ExecuteSQL(ByRef sql As String, ByRef cs As String)
| GetDTVAL :: Private Function GetDTVAL(ByRef dt As String, ByRef value As String) As String
| GetProtocol :: Public Sub GetProtocol()
| Integer :: Enum myRole As Integer
| IsKey :: Private Function IsKey(name As String) As Boolean
| LoadKeys :: Private Sub LoadKeys(keys As String)
| Msg2Form :: Sub Msg2Form(msg As String)
| NZ :: Public Function NZ(ByVal test As Object, ByVal substitute As Object) As Object
| Row2SQL :: Private Function Row2SQL(ByRef ado As adonet, ra As myAction, tab As String) As String
| Run4PID :: Private Sub Run4PID(ByVal oDir As IO.DirectoryInfo, p2 As String, role As myRole, adt As Date)
| SetProtocol :: Public Sub SetProtocol(role As myRole, table As String, action As myAction)

```