PBencoder

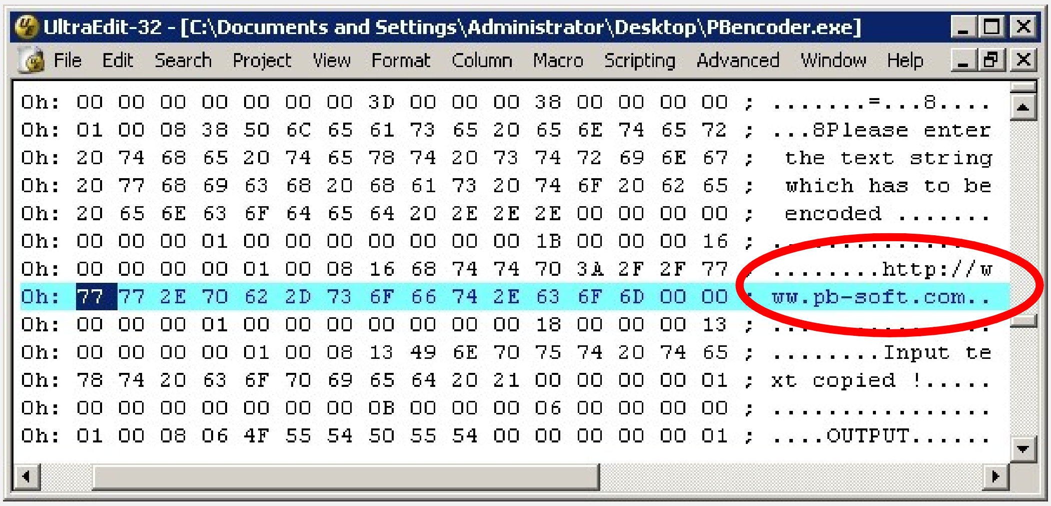
User Manual

PB
Soft

Contents

1 Introduction

PBencoder is a small application which obfuscates strings for the use in REALbasic applications. If there are important strings hardcoded into the source code, like a password for a database access, they should at least be obfuscated. If the strings are in plain text, everyone who can open an executable with a hex-editor can see them. On the following image you can see the website address www.pb-soft.com in the executable of this application. Of course that example data is not critical but you can see a plain text password the same way:


UltraEdit-32 - [C:\Documents and Settings\Administrator\Desktop\PBencoder.exe]
File
Edit
Search
Project
View
Format
Column
Macro
Scripting
Advanced
Window
Help

```
0h: 00 00 00 00 00 00 00 3D 00 00 00 38 00 00 00 00 ; .......=...8....
0h: 01 00 08 38 50 6C 65 61 73 65 20 65 6E 74 65 72 ; ...8Please enter
0h: 20 74 68 65 20 74 65 78 74 20 73 74 72 69 6E 67 ;  the text string
0h: 20 77 68 69 63 68 20 68 61 73 20 74 6F 20 62 65 ;  which has to be
0h: 20 65 6E 63 6F 64 65 64 20 2E 2E 2E 00 00 00 00 ;  encoded .......
0h: 00 00 00 01 00 00 00 00 00 00 00 1B 00 00 00 16 ; ................
0h: 00 00 00 00 01 00 08 16 68 74 74 70 3A 2F 2F 77 ; ........http://w
0h: 77 77 2E 70 62 2D 73 6F 66 74 2E 63 6F 6D 00 00 ; ww.pb-soft.com..
0h: 00 00 00 01 00 00 00 00 00 00 00 18 00 00 00 13 ; ................
0h: 00 00 00 00 01 00 08 13 49 6E 70 75 74 20 74 65 ; ........Input te
0h: 78 74 20 63 6F 70 69 65 64 20 21 00 00 00 00 01 ; xt copied !.....
0h: 00 00 00 00 00 00 00 0B 00 00 00 06 00 00 00 00 ; ................
0h: 01 00 08 06 4F 55 54 50 55 54 00 00 00 00 00 01 ; ....OUTPUT......
```

PBencoder helps to obfuscate the important strings used in REALbasic. The strings will be less readable for human beings with hex-editors. The internet address www.pb-soft.com would be obfuscated and the hex-editor would show the string QkJCdlNBdFooJ2l2Vyg/U31oaTQy.

Summarizing: The PBencoder application does only obfuscate strings ! The application does not create a kind of hash which cannot be reverted to the original string, because in the example the application needs to get the internet address out of the obfuscated string. The PBencoder also does not encrypt the strings because that would need other passwords if you want to decrypt them later !

Sure it is possible to decode the obfuscated strings (because that is really necessary !) but it is not as easy as just starting a hex-editor. With the PBencoder application you can also decode the obfuscated strings. So please don’t use this tool to hardcode your credit card information into the source code of your application !

We hope you will like this application. This manual will show you how to get started with PBencoder.

2 Requirements

PBencoder has very low hard- and software requirements. The following checklist will help you to determine if the hard- and software of your system meets the specific needs.

2.1 Hardware

- ✓ Personal computer (PC) with mouse
- ✓ 128 MB of memory (RAM)
- ✓ 6 MB of free harddisk space
- ✓ Screen resolution of 1024 x 768 pixel

2.2 Software

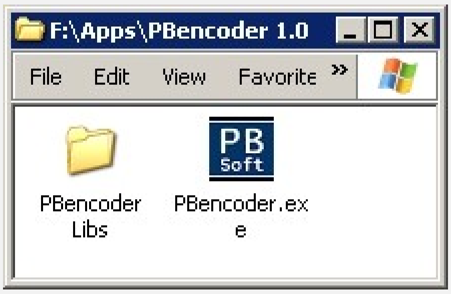
- ✓ Windows operating system (XP or higher)

3 Installation

If you have downloaded PBencoder, check if the MD5 hash from our website matches so you can be sure that the file was not modified.

If the check was successful, unpack the archive with your favorite decompression tool. If you don't have one you can use the free 7-Zip application from the website: http://www.7-zip.org

After decompressing the file you will get a folder called “**PBencoder 1.0**” with the following content:

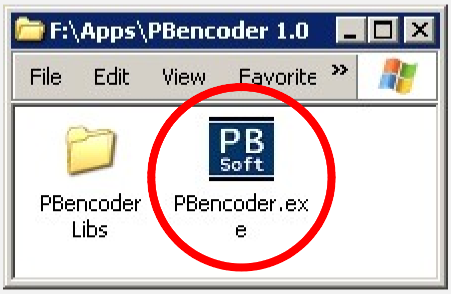


That's it ! The PBencoder application does not need further installation and is fully portable because it does not write anything into the Windows registry or to any other place.

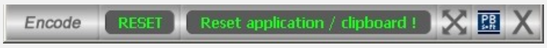
4 Application Usage

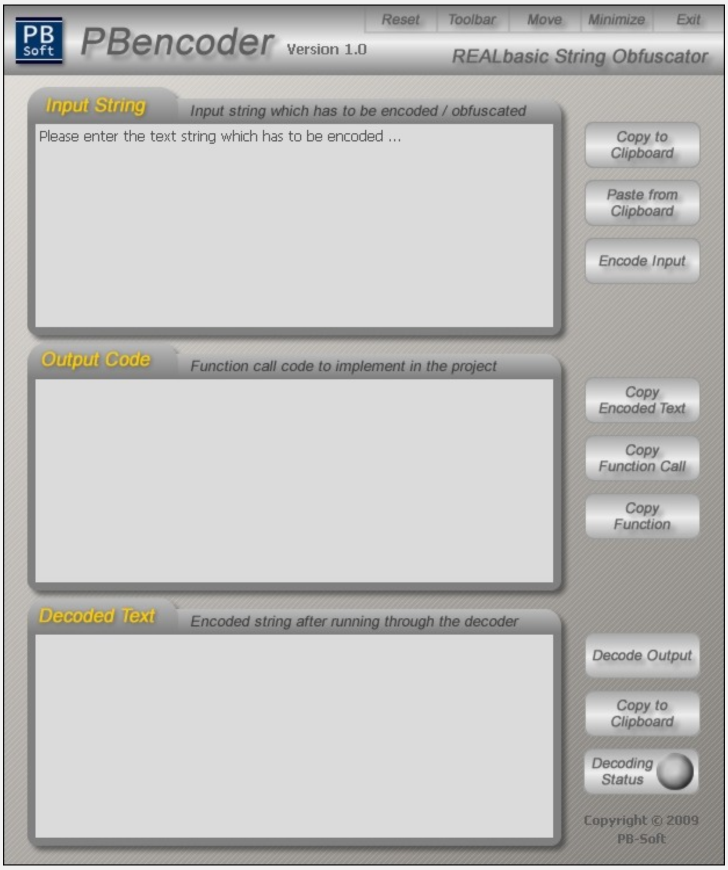
4.1 Start PBencoder

To start PBcode just double-click on the file “**PBencoder.exe**” inside the folder “**PBencoder 1.0**”.



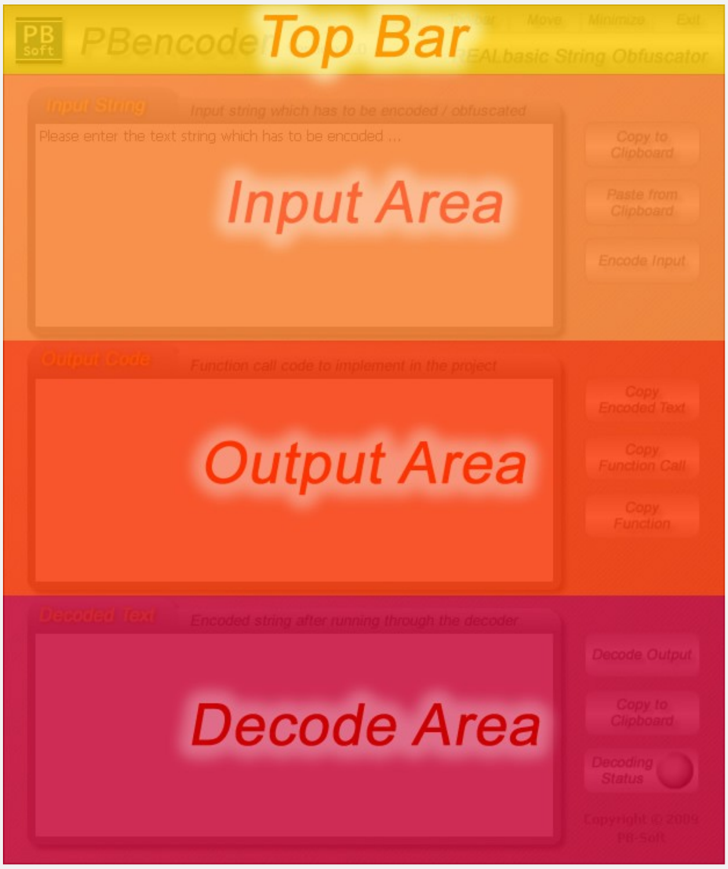
After the PBencoder application has started the main window and the toolbar will be displayed:





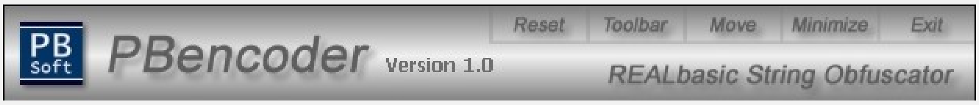
The main window has four different areas. On the following image this four areas are colored for better understanding:

1. Top Bar (Yellow)
2. Input Area (Orange)
3. Output Area (Dark Orange)
4. Decode Area (Red)



4.2 The Top Bar

The top bar of the main window includes the application title, and some navigation buttons which will be explained on this page:



4.2.1 Reset the Application

To reset the application, click on the button “**Reset**” from the top bar of the window. All text areas will be cleared and also the clipboard.

4.2.2 Show or hide the Toolbar

To show or hide the toolbar, click on the button “**Toolbar**” from the top bar of the window.

4.2.3 Move the main Window

To move the window, click on the area “**Move**” from the top bar of the window and drag the window.

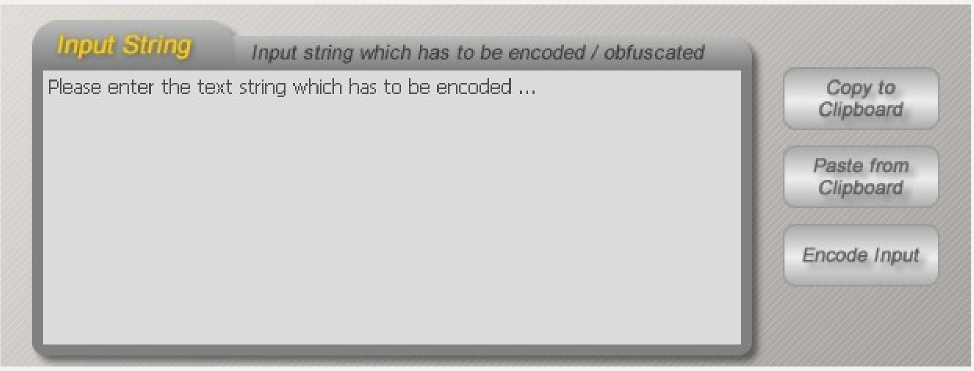
4.2.4 Minimize the main Window

To minimize the window, click on the button “**Minimize**” from the top bar of the window.

4.2.5 Exit the Application

To exit the application, click on the button “**Exit**” from the top bar of the window.

4.3 The Input Area

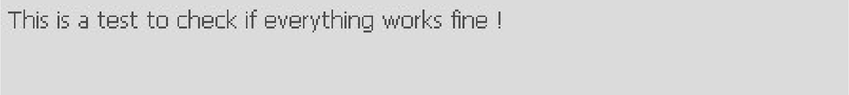


Input string which has to be encoded / obfuscated

Please enter the text string which has to be encoded ...

4.3.1 Enter and encode a plain Text String

The input area contains all elements to encode strings. If you want to obfuscate information you have to copy the string into the input area like in the following example:



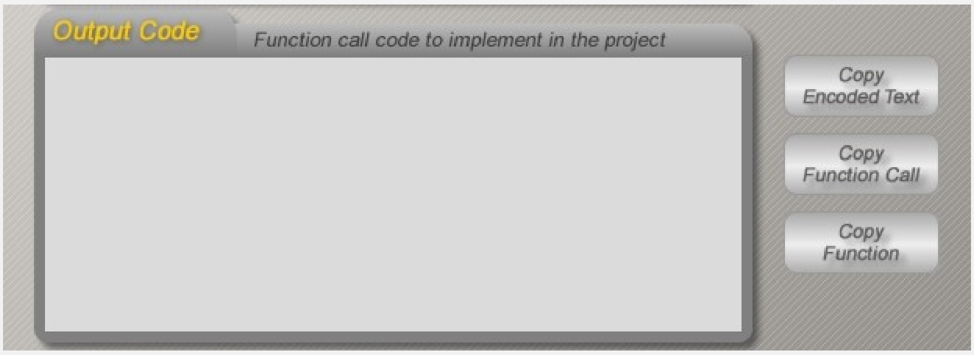
This is a test to check if everything works fine !

You can use the button “**Paste to Clipboard**” or you can mark the text of the input area and press “**CTRL+C**” to copy the content of the input area to the clipboard.

You can use the button “**Paste from Clipboard**” or you can set the cursor into the input area and press “**CTRL+V**” to copy the content of the clipboard into the input area.

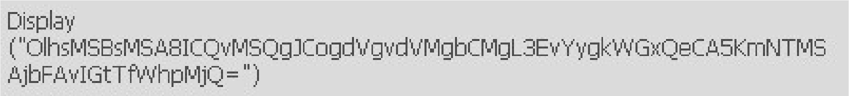
Then you can use the button “**Encode Input**” or you can press “**CTRL+WIN**” to encode the string. If you press the button again you will get another obfuscated string for the same input.

4.4 The Output Area



4.4.1 Get an encoded Text

The result of the obfuscation will be displayed in the output area. It includes the function call like on the following image:



```
Display
("OlhsMSBsMSA8ICQvMSQgJCogdVgvdVMgbCMgL3EvYygkWGxQeCA5KmNTMS
AjbFAvIGtTfWhpMjQ=")
```

If you use the same string repeatedly in your code you can obfuscate it in different ways. Just press button "**Encode Input**" again and you will get another output. The following example shows different output strings for the phrase "**This is a test !**":

```
Display("OlhsMSBsMSA8ICQvMSQga1N9aGkyNA==")
Display("Uk4yNyAyNyBTIHg9N3ggXVN9aGk2OQ==")
Display("OFs6OSA6OSAyIH5OOX4gX1N9aGk3MQ==")
Display("bWg9WCA9WCAwID97WD8gZ1N9aGk4Ng==")
Display("bzQ5SSA5SSA3IDZwSTYgclN9aGkyOA==")
```

4.4.2 Copy an encoded Text to the Clipboard

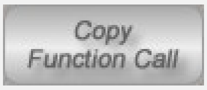
Now if you only need the encoded text you can press the button "**Copy Encoded Text**" and you will get only the text string without the function call.

The output string from the previous example would look like:

```
Pj0gYVBSeiBWVyB7cnNhU31oaTQ0
```

4.4.3 Copy a Function Call to the Clipboard

If you want to copy the function call, including the encoded text, to the clipboard, use the button "**Copy Function Call**".



This is the code with the obfuscated string which you should enter in your application in place of the plain text string. The output from the previous example would look like:

```
Display("Pj0gYVBSeiBWVyB7cnNhU31oaTQ0")
```

4.4.4 Copy the REALbasic Function to the Clipboard

If you want to copy the REALbasic function to the clipboard, press the button "**Copy Function**" on the main window.

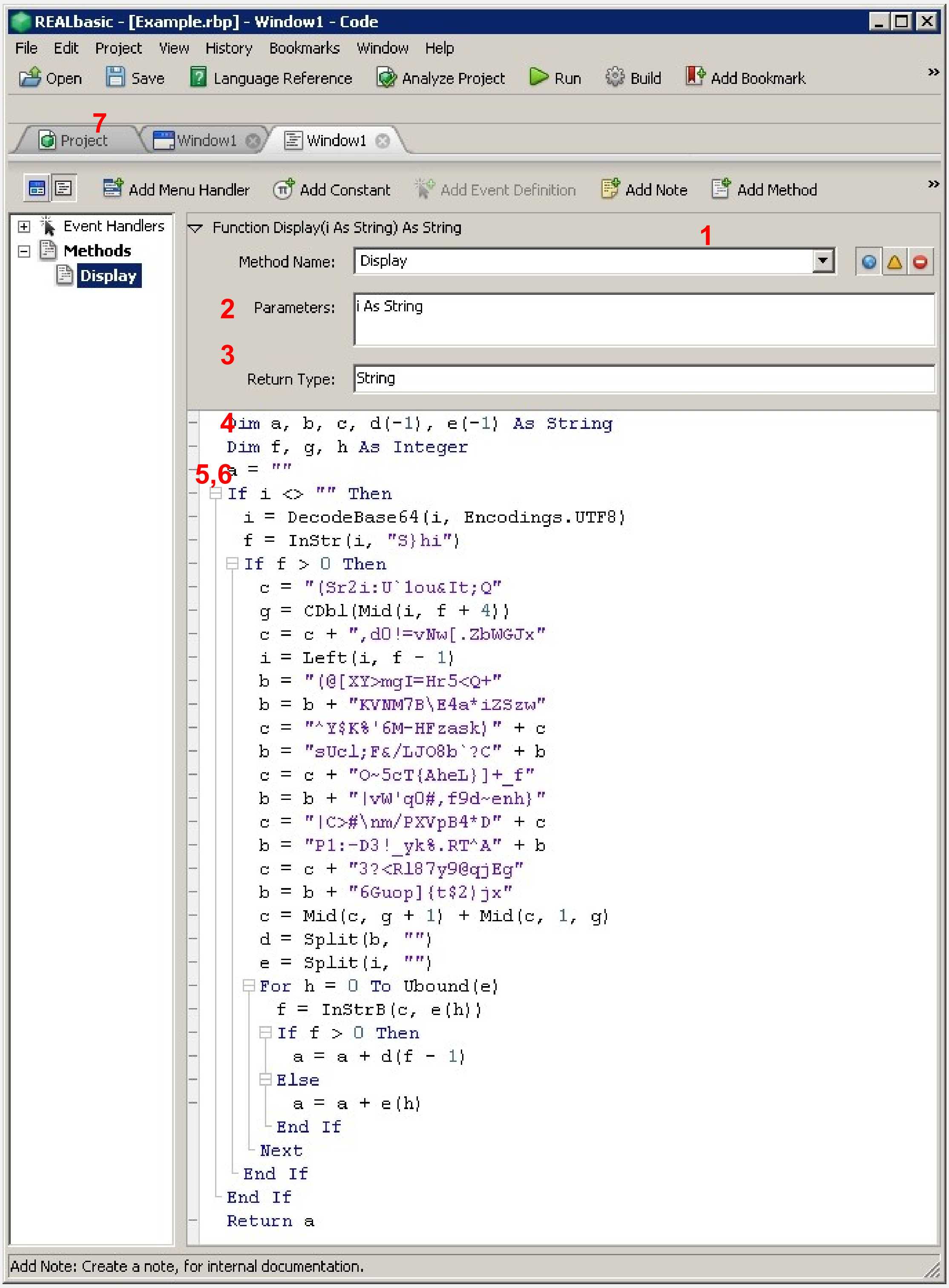
The whole REALbasic function will be copied to the clipboard and you can insert it into your application. This function will be used to decode the obfuscated strings in your application and is called "**Display**".

If we would choose a descriptive name, something like "**Decode_my_secret_string**" it would be obvious for what the function will be used.

4.4.5 Integrate the Function into your Project

To integrate the function "**Display**" into your REALbasic project please start the REALbasic developer environment and follow the steps below:

1 - Create a new method
2 - Enter '**Display**' into the 'Method Name' field
3 - Enter '**i As String**' into the 'Parameters' field
4 - Enter '**String**' into the 'Return Type' field
5 - Copy this function code into the code area
6 - Delete the help text from the code
7 - Save the method


REALbasic - [Example.rbp] - Window1 - Code
File Edit Project View History Bookmarks Window Help
Open Save Language Reference Analyze Project Run Build Add Bookmark
Project Window1 Window1
7
Add Menu Handler Add Constant Add Event Definition Add Note Add Method
Event Handlers
Methods
Display
1
2
3
4
5,6

Function Display(i As String) As String

Method Name: Display

Parameters: i As String

Return Type: String

```
Dim a, b, c, d(-1), e(-1) As String
Dim f, g, h As Integer
a = ""
If i <> "" Then
  i = DecodeBase64(i, Encodings.UTF8)
  f = InStr(i, "S}hi")
  If f > 0 Then
    c = "(Sr2i:U`lou&It;Q"
    g = CDbl(Mid(i, f + 4))
    c = c + ",d0!=vNw[.ZbWGJx"
    i = Left(i, f - 1)
    b = "(@[XY>mgI=Hr5<Q+"
    b = b + "KVNM7B\E4a*iZSzw"
    c = "^Y$K%'6M-HFzask}" + c
    b = "sUcl;F&/LJO8b`?C" + b
    c = c + "O~5cT{AheL}]+_f"
    b = b + "|vW'q0#,f9d~enh}"
    c = "|C>#\nm/PXVpB4*D" + c
    b = "P1:-D3!_yk%.RT^A" + b
    c = c + "3?<R187y9@qjEg"
    b = b + "6Guop]{t$2}jx"
    c = Mid(c, g + 1) + Mid(c, 1, g)
    d = Split(b, "")
    e = Split(i, "")
    For h = 0 To Ubound(e)
      f = InStrB(c, e(h))
      If f > 0 Then
        a = a + d(f - 1)
      Else
        a = a + e(h)
      End If
    Next
  End If
End If
Return a
```

Add Note: Create a note, for internal documentation.

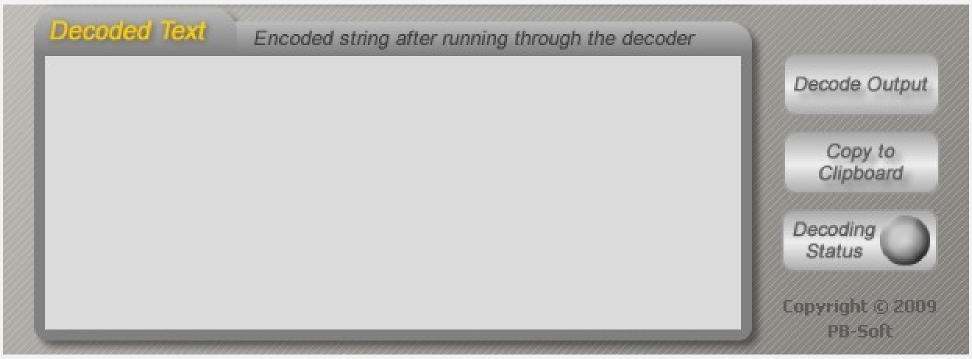
You can call this function with a function call of the type:

```
Display("Pj0gYVBSeiBWVyB7cnNhU31oaTQ0")
```

Or you can get the decoded text into a variable:

```
Result = Display("Pj0gYVBSeiBWVyB7cnNhU31oaTQ0")
```

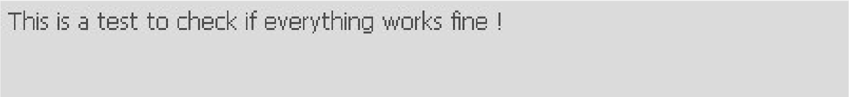
4.5 The Decode Area



4.5.1 Decode an encoded String

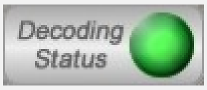
When you want to check if your input string was obfuscated successfully and if it can be decoded again, press the button "**Decode Output**".

The decoded text from the output area will be displayed in the decode area:

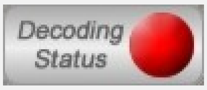


This is a test to check if everything works fine !

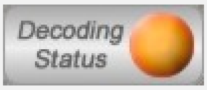
This text should be the same like the text in the input area. If the status light on the right side is green, the decoding process was successful and the two texts are equal.



If the status light is red, the decoding process failed. That can happen if you, after encoding the text, have changed the input text without encoding it after the change.



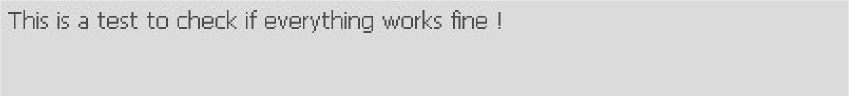
If the status light on the right side is orange, the decoding process could not be verified because there is no text or only the help text “**Please enter the text string which has to be encoded ...**” in the input area.



The PBencoder application uses the same type of function to decode the obfuscated string like you will use in your application.

If you only want to decode a string you can also paste the encoded / obfuscated string into the output area and press the button “**Decode Output**”.

The decoded text from the output area will be displayed in the decode area:



This is a test to check if everything works fine !

Attention: In this case the status light normally will be orange because the input area does not contain the input string. If the input area contains another text the status light will be red.

You can enter a whole function call and decode it or you can just enter the obfuscated text string. The PBencoder application will choose the right form to decode it. As an example, you can enter:

```
Display("Pj0gYVBSeiBWVyB7cnNhU31oaTQ0")
```

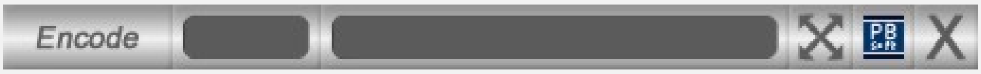
or:

```
Pj0gYVBSeiBWVyB7cnNhU31oaTQ0
```

Both ways will work !

4.6 The Toolbar

The toolbar can be used when the main window is visible or not visible. If you hide the main window and only work with the toolbar, you don’t have to switch between different windows if you want to encode a string. The toolbar has different elements which are explained in the following paragraphs:

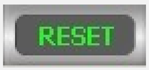


4.6.1 Encode a plain Text String

The button “**Encode**” is used to obfuscate the actual text stored in the clipboard. First you have to copy some text into the clipboard and then press the button “**Encode**”. You will get the function call with the encoded text copied to the clipboard. This is exactly the same as when you press “**CTRL+WIN**”.

4.6.2 The Status Display

The small display to the right of the “**Encode**” button displays the status of the PBencoder application.



The following modes are available:

RESET: The application and clipboard was reset.

INPUT: The clipboard contains input data.

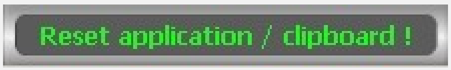
OUTPUT: The clipboard contains output data.

EMPTY: The clipboard contains no data.

ERROR: There was an error.

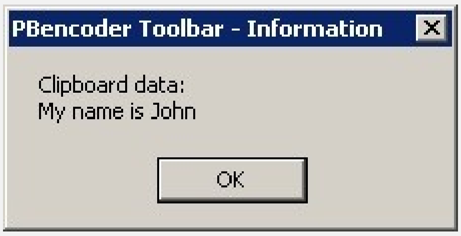
4.6.3 The Information Display

This display can show information about the application, the content of the clipboard or an information / error message:



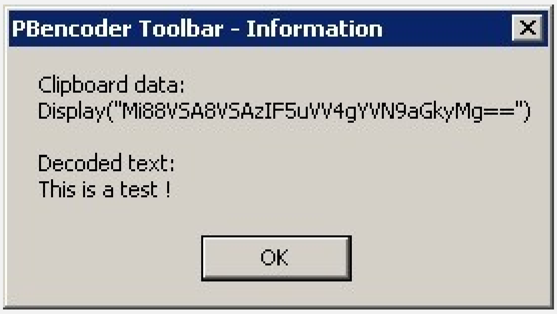
Reset application / clipboard !

If the clipboard contains plain text and you click on the information display, the actual content of the clipboard will be displayed like on the following image:



Clipboard data:
My name is John

If the clipboard contains a lot of text, only the text begin of the text will be displayed. If the clipboard data is encoded the encoded and decoded text will be shown:



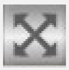
Clipboard data:
Display("Mi88VSA8VSAzIF5uVV4gYVN9aGkyMg==")

Decoded text:
This is a test !

So if you are not sure what content the clipboard holds, just click on the information display to see it.

4.6.4 Move the main Window

There is a symbol with four arrows which you can use to drag the toolbar to another location. Just click on it, hold the mouse button pressed and drag the window.



4.6.5 Hide the main Window

The symbol with the company logo can be clicked to hide the main window. Then only the toolbar stays visible. If you have the toolbar you can encode text strings without the need of the main window.

4.6.6 Exit the Toolbar

If you want to exit the toolbar you have to press the button with the "**X**". The toolbar will be closed and the main window will be displayed.

4.7 Encode, copy and check a String – All in one

If you want to the whole thing in one step you have first to copy your input string into the clipboard and press "**CTRL+WIN**". The application will copy the information from the clipboard to the input field, encodes the string, checks the output and returns it to the clipboard. The following example will show that:

- Select the phrase "**My name is John**".
- Press "**CTRL+C**" to copy the text to the clipboard.
- Press "**CTRL+WIN**" to process and encode the text.
- Press "**CTRL+V**" to insert the output (function call).

The output will look similar like:

```
Display("Pj0gYVBSeiBWVyB7cnNhU31oaTQ0")
```

The PBencoder window can be invisible and you can work from your favorite application without switching between one or more windows to encode some text.

5 Feedback

We always like feedback from you so that we can improve / enhance our products. We don’t want only positive feedback; we also like constructive criticism so we can change for the better.

If you think there is something missing in this manual, please contact our support so that we can add it in the next release !

Our website: http://www.pb-soft.com

Thank you for your attention !

P. Biegel

Patrick Biegel