

Relato de testes (04/04/24)

Resumo

API **explain**

What do you want me to explain?

Foi solicitado a explicação para a palavra **define**:

```
DEFINE DIALOG oDlg TITLE "Exemplo MsWorkTime" FROM 180,180 TO 550,700 PIXEL
```

Retornou:

The code above defines a dialog box with title "Exemplo MsWorkTime". It has a size of 180x180 to 550x700 pixels and it uses the pixel as unit for positioning.

Comentários:

- Este foi um novo teste dessa situação onde antes não retornava corretamente. Agora o retorno da IA está 100%.

Fiz outro teste da API **explain** para o código abaixo:

```
@ 10,01 say "Texto para exibição I" SIZE 200,20 COLORS CLR_RED,CLR_WHITE FONT  
oFont OF oDlg PIXEL
```

Retornou:

The code above will be explained as follows: @ 10,01 say "Texto para exibição I" SIZE 200,20 > COLORS CLRRWHITE FONT oFont OF oDlg PIXEL The first line says that we're creating a new window with the name oDlg. This window will contain our text and other controls such as buttons or checkboxes.

The second line says that we're creating a label (say) with the text "Texto para exibição I". We also specify the size of this label to be 200 pixels wide by 20 pixels high. We set the background color of the label to red and its foreground color to white. Finally, we specify which font it should use, which is stored in the variable oFont.

The third line says that we're adding the label to the window oDlg.

If you want to know more about these commands, please visit <https://tdn.totvs.com/pages/viewpage.action?pagelD=3857964>.

Comentários:

Continuo verificando que a resposta melhorou muito. E desse vez tenho apenas algumas pequenas observações em relação a resposta:

1 - O texto que ele informa que será explicado, está diferente do código original:

- Código: @ 10,01 say "Texto para exibição I" SIZE 200,20 COLORS CLR_RED,CLR_WHITE FONT oFont OF oDlg PIXEL
- Texto: @ 10,01 say "Texto para exibição I" SIZE 200,20 > COLORS CLRRWHITE FONT oFont OF oDlg PIXEL

2 - Ele separou as explicações como se o código estivesse em linhas ("..The second line says that...")

Pedi novamente o explain pela segunda vez da mesma linha do teste anterior:

```
@ 10,01 say "Texto para exibição I" SIZE 200,20 COLORS CLR_RED,CLR_WHITE FONT oFont OF oDlg PIXEL
```

Retorno:

The code above will be explained as follows:

- Line 10: @ indicates the start of a block of code.- Lines 1 and 2: These two lines contain the text to be displayed on screen.
- Line 3: SIZE indicates the size of the window where the text will be displayed.
- Line 4: COLORS indicates the colors used for the background and foreground of the text.
- Line 5: FONT indicates which font will be used to display the text.
- Line 6: OF indicates that the object (in this case it's the dialog) where the text will be displayed is named oDlg.
- Line 7: PIXEL indicates that the units of measurement used in the rest of the parameters are pixels. If you want to know more about how to use these commands, please visit <https://tdn.totvs.com/pages/viewpage.action?pageId=9883955>.

Comentários:

Novamente ele retorna, agora em items, a explicação como se o código estivesse em linhas.

Pedi explicação do trecho simples abaixo:

```
oDlg:IMaximized :=.T.
```

Retorno:

```
oDlg:IMaximized :=.T.
```

Comentários:

Não se é porque o código é muito simples, mas na realidade nao retornou nada, apenas duplicou o código solicitado.

Pedi explicação da variavel aTFolder da linha abaixo:

```
aTFolder := { 'Aba 01', 'Aba 02', 'Aba 03' }
```

Retornou:

The code above is an array containing objects with two properties: "title" and "pages". The value of each property is a string.

The first object has title = "First Folder", and pages = { "Page 1", "Page 2", "Page 3" }.

The second object has title = "Second Folder", and pages = { "Page 4", "Page 5", "Page 6" }.

The third object has title = "Third Folder", and pages = { "Page 7", "Page 8", "Page 9" }.

You can access these values by using the index operator (square brackets) to get the element at a specific position. For example, if you want to know what's on the page number 3 from the second folder, you would do something like this:

```
If( Len(aTFolder[2]:pages) >= 3, MsgInfo("This is the page number  
"+StrZero(3,Len(aTFolder[2]:pages))+ " from the second folder."), MsgInfo("There aren't enough  
pages.") )
```

In this case, we use the Len function to check whether there are enough elements in the array before trying to access them. If they're not, we show a message saying so. Otherwise, we use the StrZero function to convert the page number into a string, and concatenate it with the text "This is the page number ". We then use the MsgInfo function to show a message box with the result.

Remember that arrays start at zero, so the first element will be at index 0, the second at index 1, etc.

If you need more help understanding how arrays work in ADVPL, try looking at the documentation for Array or ask your question in the > > community forum.

Comentários:

Nesse caso, a resposta realmente não foi tão boa. As Strings não "First Folder" e etc, e informações de páginas não fazem sentido com o fonte. A tentativa de fornecer um exemplo até que foi bacana, mas foi adicionado uma virgula (,) entre as instruções e dentro do IF.

Citação de outras linguagens

Foi solicitado a explicação para o código:

```
cMsg := "Você escolheu o número 1"
```

Retornou (comentários no bloco de retorno):

```
cMsg := "Você escolheu o número 1"
```

Comentário:

Novamente é apenas retornado o mesmo texto que pedi a explicação. Não importa quantas vezes eu peça a explicação desse trecho, o retorno é sempre como informado acima. Não importa também se deixo apenas o cursor sobre a palavra cMsg ou se seleciono o texto inteiro.