

The Lazy Rabbit Catches The List Box

by William Baker

For a rabbit, Randolph can be astonishingly slow sometimes. It took him five or six months to get around to trying out the graphical preferences screen I had developed, and then he wasn't too interested in my brilliant programming techniques; he preferred to criticize GUIBuilder™, which I'd used to write the application.

"I'm astounded that GUIBuilder can't handle highlighting a line in a list box," he grouched at me. "Something that simple, a decent development tool should handle it."

"If it's so simple, why would you mind writing your own code to highlight a line in a list box?" I asked.

"I said it was a simple operation. I didn't say the code was simple. Far from it. Looking at those GUI mnemonics makes my ears spin."

**Randolph learns
that GUI boxes
just got easier
with Visual
PRO/5 2.20...**

"Well, you'll be glad to know good things sometimes come to those who do nothing. I just found out that Visual PRO/5® 2.20, the latest revision of BASIS' Microsoft Windows-based product, has a new GUIBuilder feature to deal with list boxes better. With your astounding memory, I'm certain you recall the GUIBuilder functions that pass data to and from a form with a string template. Each control has one field in this string template. The field for a list box is made up of all the items in the list, separated by line feeds. So the field for a list box might look like this:

```
form.list_box$="Item 1"+$0a$+"Item 2"+$0a$+"Item 3"+$0a$
```

"When you pass this field with the `fngb__put_screen` function, which prints data on a form, the list box will be initialized with these values. The enhancement is that you can specify one or more items to be highlighted by adding one or more string numerics, separated by `ff` characters. So this field will cause the list box to highlight the first and third items because the item numbering is zero-based:

```
form.list_box$="Item 1"+$0a$+"Item 2"+$0a$+"Item 3"+$0a$ +$ff$+"0"+$ff$+"2"
```

"I can see how that would be useful for displaying the list box, but that's only half the problem," Randolph replied. "It's just as confusing to query the list box after you're done with the form."

"That's covered, of course. If you use `fngb__get_screen` to get data from a form into the string, the list box field will include the list box selection or selections that the user made. So you just have to parse the field to get those selections. This command will return the selection in a list box where only one selection is allowed:

```
selection=num(form.list_box$(pos($ff$=form.list_box$)+1))
```

"If you want to query a list box that allows multiple selections, then you need to loop until all the selections in the list are accounted for."





Randolph snickered at me, and I demanded to know what was so funny.

"I just started to wonder how you found out about this new feature, and I imagined you ran file compares on every BBx program in the distribution to see what changed."



"That's very amusing, though I suppose it wouldn't be a bad practice. But if you must know, I found it in the Knowledge Base on the BASIS Web site. I suggest to everybody except you that they take a look at it, but there are hundreds of articles in t here, and you can be so slow that you might never finish if you started reading them."

William is a Senior Software Test Engineer in Quality Assurance and has been developing software for 14 years. He has been chronicling the programming adventures of Randolph for nine years.