

---

Subject: Re: deleting structure element  
Posted by [davidf](#) on Thu, 27 Jul 2000 07:00:00 GMT  
[View Forum Message](#) <> [Reply to Message](#)

---

David Fanning ([davidf@dfanning.com](mailto:davidf@dfanning.com)) writes:

> Bernard Puc ([bpuc@va.aetc.com](mailto:bpuc@va.aetc.com)) writes:  
>  
>> Is there a neat, clever way to "delete" an element from a structure?  
>  
> Set its pointer to a NULL object.  
>  
> Cheers,  
>  
> David  
>  
> P.S. Alright, maybe it's not "neat", but I think it  
> qualifies as "clever". :-)

On further (and more sober) reflection I would probably answer like this:

No way, no how, if you have a named structure. If you have an anonymous structure you *could* (under extreme duress) simply redefine the structure without the `&%%$^` field:

```
struct = { a:0, b:0, c:0}  
struct = { a:struct.a, b:struct.b }
```

Cheers,

David

--

David Fanning, Ph.D.  
Fanning Software Consulting  
Phone: 970-221-0438 E-Mail: [davidf@dfanning.com](mailto:davidf@dfanning.com)  
Coyote's Guide to IDL Programming: <http://www.dfanning.com/>  
Toll-Free IDL Book Orders: 1-888-461-0155

---

---

Subject: Re: deleting structure element  
Posted by [davidf](#) on Thu, 27 Jul 2000 07:00:00 GMT  
[View Forum Message](#) <> [Reply to Message](#)

---

Bernard Puc ([bpuc@va.aetc.com](mailto:bpuc@va.aetc.com)) writes:

> Is there a neat, clever way to "delete" an element from a structure?

Set its pointer to a NULL object.

Cheers,

David

P.S. Alright, maybe it's not "neat", but I think it qualifies as "clever". :-)

--

David Fanning, Ph.D.

Fanning Software Consulting

Phone: 970-221-0438 E-Mail: davidf@dfanning.com

Coyote's Guide to IDL Programming: <http://www.dfanning.com/>

Toll-Free IDL Book Orders: 1-888-461-0155

---

Subject: Re: deleting structure element

Posted by [Alex Schuster](#) on Fri, 28 Jul 2000 07:00:00 GMT

[View Forum Message](#) <> [Reply to Message](#)

---

David Fanning wrote:

>> Bernard Puc (bpuc@va.aetc.com) writes:

>>

>>> Is there a neat, clever way to "delete" an element from a structure?

> On further (and more sober) reflection I would probably

> answer like this:

>

> No way, no how, if you have a named structure. If you

> have an anonymous structure you *could* (under extreme

> duress) simply redefine the structure without the

> `*&%%$^` field:

>

> `struct = { a:0, b:0, c:0}`

> `struct = { a:struct.a, b:struct.b }`

I once wrote a routine for that purpose (attached). It replaces an element (tag) of a structure by another element or deletes it. Which doesn't work for the first element, well, if anyone wants to fix this...

```
struct = { a:0, b:0, c:0
replace_tag, struct, 'c', '' ; deletes tag c
```

or:

replace\_tag, struct, 'c', 'd', 100 ; replaces tag c with d (an int with a value of 100)

Alex

```
--
Alex Schuster   Wonko@weird.cologne.de   PGP Key available
                alex@pet.mpin-koeln.mpg.de
```

```
;;+
; NAME:
;   REPLACE_TAG
;
; PURPOSE:
;   Replaces a tag in a structure by another
;
; CALLING:
;   replace_tag, struct, old_tag, new_tag, value
;
; INPUTS:
; struct - the structure to be changed
; old_tag - name of the tag to be changed
; new_tag - name of the new tag
;   If set to "", old_tag is deleted from the structure
; value - value of the new tag
;
; RESTRICTIONS:
; Does not work with named structures
;
; MODIFICATION HISTORY:
;   Alex Schuster, MPIfnF, 8/97 - Written
;-
```

```
pro replace_tag, struct, old_tag, new_tag, value
```

```
tags = tag_names( struct )
```

```
pos = (where( tags eq strupcase( old_tag ) ))(0)
if ( pos eq -1L ) then begin
  print, 'Error: Tag ', old_tag, ' not in struct.'
  return
endif
```

```
if ( ( pos eq 0 ) and ( new_tag ne " ) ) then begin
  new_struct = create_struct( new_tag, value )
endif else begin
  new_struct = create_struct( tags(0), struct.(0) )
  for i = 1, pos-1 do $
    new_struct = create_struct( new_struct, tags(i), struct.(i) )
  if ( new_tag ne " ) then $
```

```
    new_struct = create_struct( new_struct, new_tag, value )
endelse
for i = pos+1, n_elements( tags )-1 do $
    new_struct = create_struct( new_struct, tags(i), struct.(i) )

struct = new_struct

end
```

## File Attachments

---

1) [replace\\_tag.pro](#), downloaded 168 times

---