
Subject: Re: Reading in text data

Posted by [Vince Hradil](#) on Fri, 11 Aug 2000 07:00:00 GMT

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

What about the READ_ASCII function supplied by RSI?

<reardonb@my-deja.com> wrote in message news:8mps7k\$gi2\$1@nnrp1.deja.com...

> Hi. I am reading in text data (columns and rows of numbers) and I would
> like to know if there is a more elegant way of doing it. Currently, the
> user must specify how many columns there are. In my case the number of
> columns is manually inserted into the first line of the file like this:

>
> 3
> 0 1 2
> 1 2 3
> 2 3 4
> 3 4 5
> 4 5 6
> 5 6 7
> 6 7 8
> 7 8 9
> 8 9 10
> 9 10 11

>
> The attached procedure reads in the data. Is there a way to read in the
> data such that the user does not have to a priori know how many columns
> there are and such that IDL does not have to reserve a large amount of
> memory for the number of rows?

> Thanks.

> -Brian

>
> ;-----
>

> pro BG_File, event
> compile_opt IDL2
> @catch_procedure
>
> widget_control, event.top, Get_Uvalue = pstate
> IF XREGISTERED((*pstate).filename) GT 0 THEN RETURN
>
> result = dialog_pickfile()
> (*pstate).filename = result
>
> columns = 0
> max_rows = 30000
> if (result) then begin
> OpenR, lun, result, /Get_lun
> Point_lun, lun, 0

```

> ; Read in the number of data columns which should be
> ; listed on the first line
> readf, lun, Columns
> temporary_data = fltarr(columns,/nozero)
> temporary_array = fltarr(columns,max_rows,/nozero)
> counter = 0
> While (Not(EOF(lun)) and (counter lt max_rows)) Do Begin
>   Readf, lun, temporary_data
>   temporary_array[* ,counter] = temporary_data
>   counter = counter+1
>   if counter eq max_rows then print, $
>   'Sorry, too many rows in your data.'
> Endwhile
>   Free_lun, lun
> (*pstate).data =ptr_new(temporary_array[0:columns-1, $
> 0:counter-1])
>   endif
>
>   tlb   = widget_base (title=(*pstate).filename,/column, $
> /base_align_center,group_leader=event.top)
>   status = widget_label(tlb, value=",/dynamic_resize)
>   tdata   = widget_table(tlb,$
>   group_leader=event.top, $
> /all_events,$
> /editable,$
> value=*( (*pstate).data),$
>   event_pro='BG_table_interactive',$
>   x_scroll_size = columns,$
>   y_scroll_size = 10)
>
>   table_struct = {status:status}
>   ptable_struct = ptr_new(table_struct, /no_copy)
>
>   widget_control, tlb, set_uvalue = ptable_struct
>   widget_control, tlb, /realize
>
>   xmanager, (*pstate).filename,tlb,cleanup='table_clean'
>
> end
> ;-----
>
>
>
>
> Sent via Deja.com http://www.deja.com/
> Before you buy.

```
