Subject: Re: exporting table data to a CSV file
Posted by K. Bowman on Wed, 17 Sep 2003 13:49:12 GMT
View Forum Message <> Reply to Message

In article <faf44c99.0309170518.45b76bff@posting.google.com>,
 timothy.williams@nvl.army.mil (Tim Williams) wrote:

> Hi.
>
> I want to output the contents of a table to a CSV file for import to
> Excel, or perhaps something else. I don't know in advance the data
> type of the cells, so I can't use the FORMAT= keyword. I've coded up
> something, but I know FOR loops are "bad" and was wondering if there
> is a better way.
>
> Here's what I have (coding from memory here)
>
> widget_control, table, get_value=values
> labels=tag_names(values)
>
> ;write header
> printf, lun, strjoin(labels, ', ')
> outarr=strarr(n_tags(values))
> ;write out each row
> for i=0, n_elements(values)-1 do begin
>   for j=0, n_tags(values)-1 do outarr[j]=values[i].(j)
>   printf, lun, strjoin(outarr, ', ')
> endfor
>
>
> Is there a better way to do this?
>
> I also saw David Fannings code about writing CSV files, and learned
> about the WIDTH keyword to OPEN. What's a good way to get the max
> width of the table so my output won't get truncated?
>
> Thanks.

It is probably easier to simply write the output file using free-form output. When you want to import into Excel, use the Excel Text-to-columns tool to split the input lines into columns. You can select, 'Space' as a delimiter when splitting the columns.

If you have strings in your output, it can cause problems. If the strings do not contain internal blanks, pad them with blanks at the beginning and end of each string. If they do have internal blanks, try adding quotes (") at the beginning and end.

Ken Bowman

Subject: Re: exporting table data to a CSV file
Posted by Isa Usman on Thu, 18 Sep 2003 10:19:54 GMT
View Forum Message <> Reply to Message

"Tim Williams" <timothy.williams@nvl.army.mil> wrote in message
news:faf44c99.0309170518.45b76bff@posting.google.com...
> Hi.
>
> I want to output the contents of a table to a CSV file for import to
> Excel, or perhaps something else. I don't know in advance the data
> type of the cells, so I can't use the FORMAT= keyword. I've coded up
> something, but I know FOR loops are "bad" and was wondering if there
> is a better way.
>
> Here's what I have (coding from memory here)
>
> widget_control, table, get_value=values
> labels=tag_names(values)
>
> ;write header
> printf, lun, strjoin(labels, ', ')
> outarr=strarr(n_tags(values))
> ;write out each row
> for i=0, n_elements(values)-1 do begin
> for j=0, n_tags(values)-1 do outarr[j]=values[i].(j)
> printf, lun, strjoin(outarr, ', ')
> endfor
>
>
> Is there a better way to do this?
>
> I also saw David Fannings code about writing CSV files, and learned
> about the WIDTH keyword to OPEN. What's a good way to get the max
> width of the table so my output won't get truncated?
>
> Thanks.

There is a routine called write_csv_data that does what you are looking for.
Just do a google search and you should find it.

Isa

Subject: Re: exporting table data to a CSV file

Posted by Rick Towler on Thu, 18 Sep 2003 15:15:38 GMT
View Forum Message <> Reply to Message

"Tim Williams" wrote...

> I want to output the contents of a table to a CSV file for import to
> Excel, or perhaps something else.

Have you considered writing sylk files? IDL ships with a WRITE_SYLK function which is a bit limited but works. I have modified it so that it allows for the addition of column headers to the output. We use it regularly to export data to Excel.

-Rick

Subject: Re: exporting table data to a CSV file
Posted by David Fanning on Thu, 18 Sep 2003 17:44:08 GMT
View Forum Message <> Reply to Message

Rick Towler writes:

> We use it regularly to export data to Excel.

Rick! Really! For the graphics, I presume. :-)

Cheers,

David
--
David W. Fanning, Ph.D.
Fanning Software Consulting, Inc.
Phone: 970-221-0438, E-mail: david@dfanning.com
Coyote's Guide to IDL Programming: http://www.dfanning.com/
Toll-Free IDL Book Orders: 1-888-461-0155

Subject: Re: exporting table data to a CSV file
Posted by Rick Towler on Thu, 18 Sep 2003 20:07:14 GMT
View Forum Message <> Reply to Message

"David Fanning" wrote in message...

>> We use it regularly to export data to Excel.
>
> Rick! Really! For the graphics, I presume. :-)

Of course! I love those Excel plots.

:o)

Subject: Re: exporting table data to a CSV file
Posted by lefsky on Mon, 22 Sep 2003 01:48:11 GMT
View Forum Message <> Reply to Message

The following program will export an array of structures (including structures that includes arrays and nested structures) to a csv file, with a header that include the tag-names.

```
function mk_struct_data_format,struct1,iter
struct=struct1
tn=tag_names(struct)
ntn=n_elements(tn)
if n_elements(iter) eq 0 then iter = 0

format2=''
format3=strarr(ntn)
for i=0l,ntn-1 do begin
   t=size(struct(0).(i),/type)
   nel=''
   nel_int=fix(size(struct(0).(i),/n_elements))
   if nel_int ne 1 then nel=string(nel_int)
   prefix=','
   if i eq 0 and iter eq 0 then prefix=''
   if t eq 7 then format3(i)=nel + '(A20,"|")'
   if t eq 8 then format3(i)=mk_struct_data_format(struct(0).(i),0)
   if t ne 7 and t ne 8 then format3(i)=nel + '(F20.8,"|")'

endfor

count=0
format4=''


prefix=''
for i=0l,ntn-1 do begin

   if i eq 0 then begin
      curr_format=format3(i)
      count=1
   endif

   if i ne 0 then begin

      if format3(i) eq curr_format then count=count+1
```

```
        if format3(i) ne curr_format then begin
           format4=format4+prefix+string(count)+'('+curr_format+')'
           count = 1
           curr_format=format3(i)
           if prefix eq '' then prefix=','
        endif


     endif


;print,i,curr_format,count
endfor

format4=format4+prefix+string(count)+'('+curr_format+')'

format4=strcompress(format4,/remove_all)

return,format4

end



function get_all_tagnames,struct,prefix=prefix
FORWARD_FUNCTION get_all_tagnames
if n_elements(prefix) eq 0 then prefix=''
tn=tag_names(struct)
ntn=n_elements(tn)

if n_elements(iter) eq 0 then iter = 0
out=['']
for i=0l,ntn-1 do begin
    t=size(struct(0).(i),/type)
    nel=size(struct(0).(i),/n_elements)
    if nel eq 1 then begin
        if t ne 8 then out=[out,strcompress(prefix+tn(i),/remove_all)]
        if t eq 8 then
 out=[out,get_all_tagnames(struct(0).(i),prefix=strcompress(t n(i)+'_',/remove_all))]
    endif
    if (nel ne 1 and t ne 8) then
 out=[out,strcompress(prefix+tn(i)+string(indgen(nel)),/remov e_all)]
endfor

out=out(1:*)
return,out
end
```

```
function remove_big_arr,in,maxarr
out=in
tmp=0
for i=0,n_tags(in)-1 do begin
 if n_elements(in(0).(i)) gt maxarr then tmp=[tmp,i]
endfor
if n_elements(tmp) gt 1 then begin
 out=delete_tags(in,tmp(1:*))
 return,out
endif else return,in
end


pro write_struct_psv,in1,fname,maxarr=maxarr
in=in1
close,7
openw,7,fname
if keyword_set(maxarr) then in=remove_big_arr(in,maxarr)
tn=get_all_tagnames(in)
for i=0l,n_elements(tn)-1 do tn(i)=strcompress(tn(i),/remove_all)
ntn=n_elements(tn)
nin=n_elements(in)
format1=strcompress('('+ string(ntn) +'(A20,"|"))',/remove_all)
;print,format1
printf,7,format=format1,tn
format2=mk_struct_data_format(in)
format2=strcompress('('+format2+')',/remove_all)
for i=0l,nin-1 do begin
printf,7,format=format2,in(i)
endfor
close,7
end
```