ABOUT INDEX CONTACT

LISTEN DATA
MAKE YOUR DATA TELL A STORY

HOME SAS R PYTHON DATA SCIENCE CREDIT RISK SQL EXCEL CALCULATORS OTHERS

SEARCH... GO

Home » SAS » SAS : PROC TRANSPOSE with Examples

Follow us on Facebook

SAS : PROC TRANSPOSE WITH EXAMPLES

Deepanshu Bhalla 55 Comments SAS

This tutorial explains how to use the PROC TRANSPOSE procedure in SAS, along with examples.

What does PROC TRANSPOSE do?

PROC TRANSPOSE in SAS is useful when you want to reshape your data. For example, if your data is in a vertical format but you want to convert it into a wide/horizontal format, PROC TRANSPOSE can do this task easily.

You can transpose data through data step technique but it would require writing complex code that can be time consuming to develop and test. Hence it is recommended to use the TRANSPOSE procedure to transpose your data in SAS.

Transpose Data with PROC TRANSPOSE

Sample Data Set

Let's create sample data which is used for explaining the TRANSPOSE procedure.

Suppose you have data for students with their marks in respective subjects. In the data set, you have three variables 'Name', 'Subject' and 'Marks'. See the table below showing this data.

Name	Subject	Marks
Samma	Maths	96
Sandy	English	76
Devesh	German	76
Rakesh	Maths	50
Priya	English	62
Kranti	Maths	92
William	German	87

Create data set in SAS

To see this data in SAS data set format, run the following code -

```
data transp;
input Name $ Subject $ Marks;
cards;
Samma Maths 96
Sandy English 76
Devesh German 76
Rakesh Maths 50
Priya English 62
Kranti Maths 92
William German 87
;
run;
```

It creates a data set named '**TRANSP**' which is stored in **WORK** library.

Simplest Form of PROC TRANSPOSE

```
proc transpose data = transp out= outdata;
run;
```

The above code creates a data set called **outdata** which contains values of variable 'Marks' stored in horizontal (wide) format. In other words, it transposes only variable i.e. Marks (which is numeric). It is because **by default, PROC TRANSPOSE transposes all numeric variables in the data set.**

Output Data Set

The output of the data set looks like below -

NAME	COL1	COL2	COL3	COL4	COL5	COL6	COL7
Marks	96	76	76	50	62	92	87

Options in PROC TRANSPOSE

1. The **NAME=** option allows you to change the name of the **_NAME_** variable. It is the name of the variable that is transposed.
2. The **PREFIX=** option allows you to change the prefix "**COL**". It is prefix to the transposed values.

```
proc transpose data = transp name=VarName prefix=Student out= outdata;
run;
```

Observe the above code with the previous section code - There are two changes in the code above that are : specifying name 'VarName' to the variable Name. The other change is adding a prefix 'Student' to the transposed marks.

Output Data Set

VarName	Student1	Student2	Student3	Student4	Student5	Student6	Student7
Marks	96	76	76	50	62	92	87

Statements in PROC TRANSPOSE

1. **ID -[Move to Column Name]** It allows you to include the values of a variable as variable names in the output data set. In other words, it tells SAS to give the variable names in the output file which were observations (rows) values in a variable in the input data set. If the variable in the ID statement is numeric, an underscore will be put by default at the beginning of the variable name. Instead of a default '_', you can use **PREFIX= option** to give a specific prefix which can be any

character value.For example, you want to add 'Height' as a prefix which would create variables like 'Height20' 'Height30'.

2. **BY -**It allows you to transpose data within the combination of the BY variables. The BY variables themselves aren’t transposed. The variables need to be sorted before running PROC TRANSPOSE. You can sort the variables with PROC SORT.
3. **VAR -[Transpose Column]** It lists the actual data that needs to be transposed. If you do not include a VAR statement, the procedure will transpose all numeric variables that are not included in a BY statement or a ID statement. If youwant to transpose a character variable, a VAR statement is required.

Example 2 : Give name to transposed columns

Suppose you want to have actual students' name instead of 'Student1 Student2 etc' in the variable names. You can use ID statement to accomplish this task. Check out the code below -

```
proc transpose data = transp name=VarName out= outdata;
id name;
run;
```

Output Data Set

VarName	Samma	Sandy	Devesh	Rakesh	Priya	Kranti	William
Marks	96	76	76	50	62	92	87

In this case, the variable **'name'** is used for naming variables.

Example 3 : Restructure Data

Suppose you want to change the structure of data in the manner in which the row values of the variable 'Subjects' come at top i.e. heading / variable names and marks under the respective column in the output dataset.

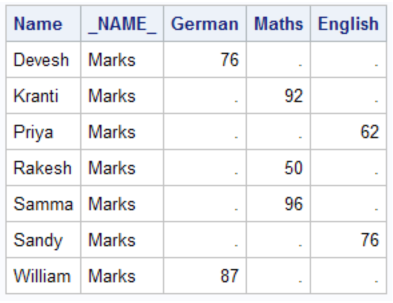
In this case, we need to sort the data as we are going to use BY processing in PROC TRANSPOSE.

```
proc sort data = transp;
by Name;
```

```
run;

proc transpose data = transp out= outdata;
by Name;
id Subject;
var Marks;
run;
```

In this example, we are specifying variable **Name** in the **BY** option which means we do not want to transpose this variable.. The variable **Marks** specified in the **VAR** option implies this variable is actually transposed and shape of the data format would be changed in the output data set.



Name	_NAME_	German	Maths	English
Devesh	Marks	76	.	.
Kranti	Marks	.	92	.
Priya	Marks	.	.	62
Rakesh	Marks	.	50	.
Samma	Marks	.	96	.
Sandy	Marks	.	.	76
William	Marks	87	.	.

Output : PROC TRANSPOSE

If you look at the output above, everything looks perfect except the variable '_NAME' which is not relevant. We can eliminate this variable with **DROP=** option.

```
proc transpose data = transp out= outdata (drop=_name_);
by Name;
id Subject;
var Marks;
run;
```

Is SORTING required when i use BY statement?

Answer is **No**. The **NOTSORTED** option tells SAS that data is not sorted and it is not required to sort it. If you don't specify **NOTSORTED** option, you need to sort the variable that is listed in **BY** statement.

```
proc transpose data = transp out= outdata (drop=_name_) ;
by Name NOTSORTED;
id Subject;
var Marks;
run;
```

Output Data Set

Name	Maths	English	German
Samma	96	.	.
Sandy	.	76	.
Devesh	.	.	76
Rakesh	50	.	.
Priya	.	62	.
Kranti	92	.	.
William	.	.	87

See the above output. You must have observed the names are not sorted in the output data set.

How to use Two Variables in ID Statement

We can use **DELIMITER= option** to separate values of two variables specified in the ID statements. In this example, we have used underscore (_) as a delimiter.

```
proc transpose data = transp delimiter=_ name=VarName out=
outdata;
id name subject;
run;
```

VarName	Samma_Maths	Sandy_English	Devesh_German	Rakesh_Maths
Marks	96	76	76	50

How to label the Output Variables with PROC TRANSPOSE

The ID statement tells SAS to provide variable names to the variables after the transpose. But if you want to label these variables, you can use **IDLABEL statement** which picks labels from a variable from the input file.

```
proc transpose data=temp out=outdata prefix=height;
by id;
var scores;
id height;
idlabel heightl;
run;
```

Practice Example

Suppose you have monthly financial data. You need to convert long formatted data to wide format.

ID	Months	Revenue	Balance
101	1	3	90
101	2	33	68
101	3	22	51
102	1	100	18
102	2	58	62
102	3	95	97

SAS Code

```
data example;
input ID Months Revenue Balance;
cards;
101 1 3 90
101 2 33 68
101 3 22 51
102 1 100 18
102 2 58 62
```

```
102 3 95 97
;
```

Output

The final output should like the following table.

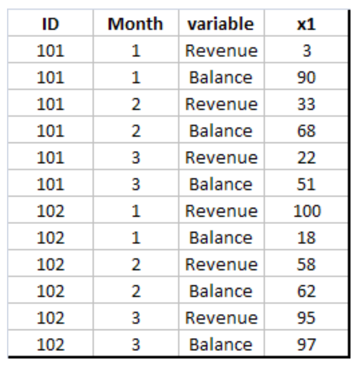
ID	balance_1	balance_2	balance_3
101	90	68	51
102	18	62	97

Solution :

```
proc transpose data=example out= Output1 (drop = _NAME_)
prefix=balance_;
id months;
var balance;
by ID;
run;
```

In this case, the variable 'Month' specified in ID statement is a numeric variable. Hence, we have added prefix 'balance_' to make it to the desired output.

If you want to see your output looks like the data shown in the image below -



ID	Month	variable	x1
101	1	Revenue	3
101	1	Balance	90
101	2	Revenue	33
101	2	Balance	68
101	3	Revenue	22
101	3	Balance	51
102	1	Revenue	100
102	1	Balance	18
102	2	Revenue	58
102	2	Balance	62
102	3	Revenue	95
102	3	Balance	97

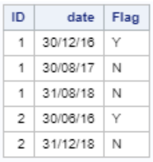
PROC TRANSPOSE : Reshape Data

```
proc transpose data=example out=out1 name=variable prefix=x;
by id months;
run;
```

In this case, the information of the 'Revenue' and 'Balance' variables are stacked to one variable. And the variable 'x1' refers to the values corresponding to it.

Exercise : Try yourself!

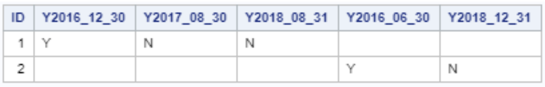
Suppose you have three columns - ID, Date and Flag. ID refers to unique value assigned to customers. Flag refers to status of customer as on date whether it is active or not. Input Dataset looks like the image shown below.



ID	date	Flag
1	30/12/16	Y
1	30/08/17	N
1	31/08/18	N
2	30/06/16	Y
2	31/12/18	N

```
data readin;
   input ID date ddmmyy8. Flag$;
   format date ddmmyy8.;
   cards;
1 30-12-16 Y
1 30-08-17  N
1 31-08-18  N
2 30-06-16 Y
2 31-12-18 N
;
run;
```

Desired Output should be appeared like the table below. **Post your solution in the comment box below.**



ID	Y2016_12_30	Y2017_08_30	Y2018_08_31	Y2016_06_30	Y2018_12_31
1	Y	N	N		
2				Y	N

Related Tutorial : Transpose Multiple Variables

Related Posts

- How to Calculate Difference Between Dates in SAS
- SAS : How to Add Days to Dates
- How to Integrate Google Bard into SAS
- SAS PROC PRINT: Learn with Examples
- How to Raise a Number to a Power in SAS

SAS Tutorials : Top 100 SAS Tutorials

Spread the Word!

Share in Share Tweet

Deepanshu founded ListenData with a simple objective - Make analytics easy to understand and follow. He has over 10 years of experience in data science. During his tenure, he worked with global clients in various domains like Banking, Insurance, Private Equity, Telecom and HR.

While I love having friends who agree, I only learn from those who don't

Let's Get Connected

Email in LinkedIn

55 Responses to "SAS : PROC TRANSPOSE with Examples"

achin September 29, 2016 at 10:57 AM

Thanks for the post

Reply Delete

Anonymous January 9, 2017 at 4:01 AM

what is use of let option in transpose process

Reply Delete

Replies

chandu September 14, 2021 at 11:16 AM

LET will keep duplicate values of an ID Variable

Delete

Reply

Unknown February 6, 2017 at 7:36 PM

proc transpose data = transp delimiter=_ name=VarName out= outdata;
id name subject;
run;

its not working

Reply Delete

Replies

Unknown February 11, 2017 at 8:47 PM

proc transpose data=transp delimiter=_ name=varname out=outdata;
id name subject;
run;

Delete

Reply

Unknown October 27, 2017 at 11:18 PM

in proc transpose we are using id var by in that what purpose using ID please explain me

Reply Delete

Anonymous November 5, 2017 at 8:39 AM

```
proc transpose data=example1 out=out1 name=variable prefix=x;
by id month;
run;
```

Hi please chk the above mentioned code it is not providing the desired output.

Reply Delete

Replies

Deepanshu Bhalla November 5, 2017 at 9:08 AM

There were some typos. I have corrected it. Thanks for highlighting!

Delete

Reply

Anonymous November 11, 2017 at 12:12 AM

Hi There, As the name here in the dataset does not contain repetitive names then it is working fine but what if we have below scenario?

```
data one;
input cust$ account$ balance dollar9.;
datalines;
smith checking $1,000.00
smith savings $4,000.00
smith mortgage $150,000
smith credit_card $500
Jones checking $973.78
Jones savings $2613.44
Jones mortgage .
Jones credit_card $140.48
;
run;

proc transpose data = one name=cust out = two;
id cust;
run;

proc print data = two;
run;
```

It is saying
ERROR: The ID value "smith" occurs twice in the input data set

What is to be done in this case?

Reply Delete

Replies

Anonymous December 17, 2017 at 3:26 AM

PLEASE USE BELOW CODE:

```
PROC TRANSPOSE DATA = ONE NAME=CUST OUT = TWO;
BY CUST ACCOUNT NOTSORTED;
ID CUST;
RUN;

PROC PRINT DATA = TWO;
RUN;
```

Delete

Unknown October 13, 2018 at 1:25 AM

LET me correct you :)

```
proc sort data=one
by cust;
run;

proc transpose data=one out=two;
id account;
by cust;
run;
```

or

if you not want to sort dataset one then used below code

```
proc transpose data=one out=two;
id account;
by cust notsorted;
run;
```

Delete

Unknown July 20, 2020 at 11:22 PM

```
proc transpose data = one name=cust out = two;
id cust;
by account notsorted;
run;

proc print data = two;
run;
```

Delete

Hemant Negi October 5, 2020 at 3:47 AM

```
/*Use let option*/
proc transpose data = one name=cust out = two let;
```

```
id cust;
run;
```

Delete

Yogendra Negi January 22, 2021 at 2:59 AM

```
proc transpose data = one out = two(drop=_name_);
id cust account;
run;
```

Delete

zubair April 26, 2021 at 8:41 AM

Proc transpose doesnt work on duplicate values.

Delete

Reply

ELLAMMA December 6, 2018 at 9:02 AM

thanks for your post can u post something good about match merging and interleaving of sas datasets. thanks

Reply Delete

Anonymous December 20, 2018 at 2:48 AM

Awesome post..Thanks for this...

Reply Delete

Anonymous February 16, 2019 at 12:06 AM

You are good bro.

Reply Delete

Anonymous February 24, 2019 at 3:49 AM

Congratulations!
Very helpful

Reply Delete

Anonymous March 8, 2019 at 12:16 AM

I Have transaction dataset in which I have a column of expenses I want to keep all transactions side by side using comma based on the account id wise.Below i have mentioned small scenario of the one.

Acct_Id gender expenses
101 M 20000
102 F 20000
103 F 50000
101 M 10000
103 F 18000
102 F 21000
102 F 11000
103 F 49000
101 M 20000
I want all expenses in one column side by side using deimeter as comaa, I want it as below in SAS, Can anyone please assist me in doing this will be a great help for me.

101 M 20000,10000,20000
102 F 20000,21000,11000
103 F 50000,18000,49000
Thanks and regards,
Swarupa

Reply Delete

Replies

Richa March 24, 2019 at 6:37 AM

This comment has been removed by the author.

Delete

Richa March 24, 2019 at 6:41 AM

Try this and let me know if it works:

```
data expenses;
input Acct_Id gender $ expenses;

datalines;

101 M 20000
102 F 20000
103 F 50000
101 M 10000
103 F 18000
102 F 21000
102 F 11000
103 F 49000
101 M 20000
;

proc sort data = expenses;
by Acct_id;
run;

data final(drop= expenses);
set expenses;
by Acct_id;
length Tot_exp $20;
retain tot_exp;
if first.acct_id then Tot_exp = put(expenses, 6.);
else Tot_exp = cats(Tot_exp,',',put(expenses, 6.));
if last.acct_id then output;
run;
```

Delete

Pate Shital Subhash April 5, 2022 at 3:17 AM

it works..

Delete

Reply

Richa March 24, 2019 at 6:36 AM

This comment has been removed by the author.

Reply Delete

Unknown July 22, 2019 at 2:54 AM

Sir, Please help me.. I'm working in SASA last few months. I am not getting how to write a code. Here Market names are missing, for those above market name is considered for bellow blanked rows up to next market name (above same market name continues to the next market and next market name continues to the another market name). Like this each crop has 65000 rows in single crop and totally I have 46 crops. Please suggest the code with examples..

Market Date Variety Arriv Min Max Modal District Year
BALLARI 02/03/2004 NH-44 COTTON 163 1770 2870 2670 Ballari 2004
13/02/2004 NH-44 COTTON 324 1750 2850 2280 Ballari 2004
20/02/2004 NH-44 COTTON 99 1867 2713 2450 Ballari 2004
27/02/2004 NH-44 COTTON 126 1726 2740 2330 Ballari 2004
Total 712 2432 2004
BELUR 25/02/2004 COTTON GINNED 32 2400 2550 2500 Hassan 2004
Total 32 2500 2004
BYADGI 02/03/2004 D.C.H. 6 2429 2669 2429 Haveri 2004
02/05/2004 D.C.H. 8 2069 2689 2269 Haveri 2004
02/09/2004 D.C.H. 7 2009 2669 2419 Haveri 2004
02/12/2004 D.C.H. 4 2059 2669 2459 Haveri 2004
16/02/2004 D.C.H. 12 2349 2869 2409 Haveri 2004
19/02/2004 D.C.H. 5 2329 2569 2529 Haveri 2004
23/02/2004 D.C.H. 5 2019 2369 2329 Haveri 2004
26/02/2004 D.C.H. 4 869 2219 879 Haveri 2004
Total 51 2215 2004
CRNAGAR 15/02/2004 OTHER 94 2814 2814 2814 CRNagara 2004
Total 94 2814 2004
C.DURGA 02/03/2004 D.C.H. 94 1680 3294 2650 C.durga 2004
02/05/2004 D.C.H. 121 1580 2930 2525 C.durga 2004
02/07/2004 D.C.H. 59 1600 3189 2600 C.durga 2004
02/10/2004 D.C.H. 112 1450 2890 2540 C.durga 2004
02/12/2004 D.C.H. 117 1680 2782 2440 C.durga 2004
17/02/2004 D.C.H. 249 1330 2790 2600 C.durga 2004
21/02/2004 D.C.H. 173 1580 3139 2550 C.durga 2004
24/02/2004 D.C.H. 227 1400 2933 2590 C.durga 2004
25/02/2004 D.C.H. 0 0 0 0 C.durga 2004
26/02/2004 D.C.H. 148 1590 2959 2590 C.durga 2004
28/02/2004 D.C.H. 197 1750 2806 2450 C.durga 2004
Total 1497 2321 2004
D.GERE 02/04/2004 D.C.H. 7 929 2329 2029 D.gere 2004
02/07/2004 D.C.H. 1 1969 1969 1969 D.gere 2004
02/09/2004 D.C.H. 2 2029 2029 2029 D.gere 2004
13/02/2004 D.C.H. 15 1002 2619 2120 D.gere 2004
16/02/2004 D.C.H. 1 1929 1929 1929 D.gere 2004
19/02/2004 D.C.H. 3 2169 2339 2309 D.gere 2004
20/02/2004 D.C.H. 5 2009 2059 2025 D.gere 2004
23/02/2004 D.C.H. 12 1839 2459 2169 D.gere 2004
26/02/2004 D.C.H. 6 2079 2279 2159 D.gere 2004
27/02/2004 D.C.H. 2 2159 2369 2259 D.gere 2004
28/02/2004 D.C.H. 2 2109 2309 2209 D.gere 2004
Total 56 2109 2004

Reply Delete

Anonymous August 14, 2019 at 12:05 AM

```
proc transpose data = transpsorted out= outdata3;
by Name;
id Subject;
var Marks;
run;
```

Reply Delete

Ravi Sachdeva August 14, 2019 at 3:54 AM

This comment has been removed by the author.

Reply Delete

Replies

Deepanshu Bhalla August 16, 2019 at 9:47 AM

Great. Data step code can be shortened. Thanks!

Delete

Ravi Sachdeva August 19, 2019 at 5:42 AM

yeah , it can be .. i just tried , you can shorten it too

Delete

Reply

Ravi Sachdeva August 19, 2019 at 5:40 AM

```
data readin;
input ID date ddmmyy8. Flag$;
format date ddmmyy8.;
cards;
1 30-12-16 Y
1 30-08-17 N
1 31-08-18 N
2 30-06-16 Y
2 31-12-18 N
;
run;


data new(rename=(date5=date));
set readin;
date1=put(date,ddmmyy8.);
date2=scan(date1,1,"/");
date3=scan(date1,2,"/");
date4=scan(date1,3,"/");
date5=catx("-",date4,date3,date2);
drop date date1 date2 date3 date4;
run;


proc transpose data=new out=readin_t(drop=_name_) prefix=Y20;
var flag;
by id;
```

```
id date;
run;
```

Reply Delete

Replies

Unknown July 21, 2020 at 1:02 AM

```
data akshay;
set readin;
format date yymmdd8.;
run;


proc transpose data=akshay out=aww prefix=Y20_;
id date;
var flag;
by id;
run;
```

Delete

Unknown January 29, 2021 at 11:18 PM

```
data akshay;
set readin;
format date yymmdd10.;
run;
```

It will give you date format 2016-12-30

Then you run this code:

```
proc transpose data= readin1 out= Output1 delimiter= _ prefix=Y
Name= date ;
id Date;
var flag;
by ID;
run;
```

Delete

Reply

Unknown September 2, 2019 at 8:12 AM

```
data test;
set readin;
format date yymmdd10.;
run;

proc transpose data=test prefix=Y out=new1 (drop=_name_);
var flag;
id date;
by id;
run;
```

Reply Delete

Hockleberry Clover September 13, 2019 at 2:27 PM

Thanks so much!! Glad to have found this site.

```
proc transpose data=readin out=out(drop=_name_) prefix=Y20 delimiter=_;
by id;
var flag;
id date;
format date yymmdd8.;
run;
```

Reply Delete

Replies

Sankalp October 8, 2020 at 10:47 AM

How does the delimiter will come into play as it is used to separate 2 vars in ID statement, I can see ur code working. Could you please help to explain

Delete

Reply

Unknown December 20, 2019 at 6:48 AM

Clear, Simple and Perfect!! Thanks much!

Reply Delete

1 January 14, 2020 at 9:48 PM

sir how to use the label option in proc transpose because when i am useing that optio i am getting the error that the variable written using label statment is not found

Reply Delete

Unknown April 17, 2020 at 3:06 PM

Solution:

```
data readin;
input ID date ddmmyy8. Flag$;
format date ddmmyy8.;
cards;
1 30-12-16 Y
1 30-08-17 N
1 31-08-18 N
2 30-06-16 Y
2 31-12-18 N
;
run;

data readin;
set readin;
format date1 yymmdd10.;
date1=date;
run;

proc transpose data=readin prefix=Y out=reading1(drop=_name_);
by id notsorted;
```

```
id date1;
var flag;
run;
```

Reply Delete

Anonymous April 23, 2020 at 1:58 AM

```
proc transpose data=readin out=prac (drop=_name_) prefix=Y_;
format date yymmdd10.;
by id;
id date;
var flag;
run;
```

Reply Delete

Anonymous May 13, 2020 at 9:17 PM

Ans:

```
proc transpose data=readin out=trans_readin(drop=_name_) prefix=Y ;
format date yymmdd10.;
by id;
var flag;
id date;
run;
```

Reply Delete

Anonymous November 4, 2020 at 7:22 AM

Here is right answer:

```
data readin2;
set readin;
format date yymmdd10.;
run;


data readin3;
set readin2;
date1 = put(date, yymmdd10.);
date2 = tranwrd(date1 , '-' , '_');
run;

proc transpose data=readin3 out=new1 (drop=_name_) prefix=Y;
by ID;
id date2;
var flag;
run;
```

Reply Delete

Unknown November 16, 2020 at 1:10 AM

```
data readin;
input ID date ddmmyy8. Flag$;
format date ddmmyy8.;
cards;
1 30-12-16 Y
1 30-08-17 N
1 31-08-18 N
```

```
2 30-06-16 Y
2 31-12-18 N
;
run;

data read;
set readin;
year=year(date);
month=month(date);
day=day(date);
run;

proc transpose data=read out=read_transposed (drop=_name_) prefix=Y
delimiter=_;
id year month day;
var flag;
by id;
run;
```

Reply Delete

kate January 14, 2021 at 1:17 PM

```
data new;
set readin;
year=year(date);
month=month(date);
day=day(date);
run;

proc transpose data=new out=readin_transp(drop=_name_) prefix=Y_
delimiter=_;
by id;
var flag;
id year month day;
run;

proc print data=readin_transp;
run;
```

Reply Delete

kate January 14, 2021 at 1:19 PM

thanks a lot!!

Reply Delete

Soumyadeep March 10, 2021 at 1:51 PM

```
DATA readin2(drop=year month month1 day date);
set readin;
format date ddmmyy10.;
year=year(date);
month=month(date);
IF month <10 then month1=('0'||strip(month));
else month1=month;
day =day(date);
date1 =(STRIP(YEAR)||"_"||STRIP(MONTH1)||"_"||STRIP(DAY));
run;
```

```
proc transpose data=readin2 prefix=Y out=outdata(drop=_Name_);
id date1;
BY ID;
var Flag;
run;
```

Reply Delete

Sas Scoop May 31, 2021 at 6:49 PM

```
proc transpose data=readin out=trans_readin (drop=_name_) prefix=y;
id date;
var Flag;
by ID;
run;
proc print;
run;
```

Reply Delete

makey June 21, 2021 at 7:57 AM

```
proc transpose
data = readin
prefix = Y20
out = out(drop = _name_);
by id;
var flag;
id date
;run;


proc sql ;
select cats("'", name, "'n", '=', tranwrd(name, '/', '_'))
into :new_names separated by " "
from dictionary.columns
where libname eq 'WORK'
and memname eq 'OUT'
and name like 'Y%'
;quit;


data work.out;
set work.out
(rename = (&new_names.))
;run;
```

Reply Delete

Anonymous August 18, 2021 at 8:47 PM

```
data readin;
input ID date ddmmyy8. Flag$;
format date ddmmyy8.;
cards;
1 30-12-16 Y
1 30-08-17 N
1 31-08-18 N
2 30-06-16 Y
2 31-12-18 N
```

```
;


data new;
set readin;
day=day(date);
month=month(date);
yr=year(date);
date1=catx('_',day,month,yr);
drop day month yr date;
run;


proc transpose data=new out=readin_t(drop=_name_) prefix=Y20;
var flag;
by id;
id date1;
run;
```

Reply Delete

chandu September 14, 2021 at 11:07 AM

```
data readin;
input ID date ddmmyy8. Flag$;
format date ddmmyy8.;
cards;
1 30-12-16 Y
1 30-08-17 N
1 31-08-18 N
2 30-06-16 Y
2 31-12-18 N
;
run;

proc transpose data=readin out=read(drop=_name_) prefix=Y20;
format date yymmdd8.;
var flag;
by id;
id date;
run;
```

Reply Delete

Unknown October 7, 2021 at 3:35 AM

```
data readin;
input ID date ddmmyy8. Flag$;
format date ddmmyy8.;
cards;
1 30-12-16 Y
1 30-08-17 N
1 31-08-18 N
2 30-06-16 Y
2 31-12-18 N
;
run;
proc sort data=readin;
by ID; run;
proc transpose data=readin out=readin1(drop=_name_) prefix=y;
```

```
id date;
var Flag;
by ID;
run;
```

Reply Delete

santosh October 8, 2021 at 10:09 AM

```
data readin;
input ID date ddmmyy8. Flag$;
format date yymmdd10.;
cards;
1 30-12-16 Y
1 30-08-17 N
1 31-08-18 N
2 30-06-16 Y
2 31-12-18 N
;
run;
proc print;
run;
proc sort data=example;
by id;
run;
proc transpose data=readin out=dad (drop=_name_) prefix=y ;
id date;
var flag;
by id;
run;
```

Reply Delete

Anonymous October 20, 2021 at 12:02 AM

```
proc transpose data=readin name=id prefix=Y out=out2 (DROP= id);
format date yymmddb10.;
by id;
id date;
var flag;
run;
```

Reply Delete

Karimulla November 30, 2021 at 12:39 PM

```
proc transpose data = readin out= readin2 (DROP=_NAME_) prefix=Y;
format date yymmdd10.;
by ID;
ID date;
VAR FLAG;
run;
```

Reply Delete

Shakila Radhakrishnan December 4, 2021 at 1:35 AM

```
proc transpose data=readin out=outdata(drop=_NAME_) prefix=Y;
id date;
format date yymmdd10.;
var flag;
```

```
by id;
run;
```

Reply Delete

Sasi February 8, 2023 at 10:44 AM

```
proc transpose data=readin2 out=readin3(rename=(_30_12_16=y2016_12_30
_30_08_17=y2017_08_30
_31_08_18=y2018_08_31 _30_06_16=y2016_06_30
_31_12_18=y2018_12_31)drop=_name_);
by id;
id date;
var flag;
run;
```

Reply Delete

To leave a comment, click the button below to sign in with Google.

SIGN IN WITH GOOGLE

← PREV NEXT →