

# 5. PROC TRANSPOSE and PROC IMPORT





Daniel Alconchel Vázquez



## Exercise 1. Transpose variable Surname from table

```
data surnames;
  input stid Surname$;
  datalines;
100 Kowalski
101 Cicha
102 Cwir
103 Wokulski
104 Andersen
110 Lubicz
111 Samosia
112 Eris
113 Kowalska
114 Losski
116 Rajska
;
run;
```

```
proc transpose data=surnames;
  var surname;
run;
```





CÓDIGO LOG RESULTADOS DATOS DE SALIDA

Tabla: WORK.DATA2 Ver: Nombres de columna     Filtrar: (ninguno)

Columnas  N° total de filas: 1 N° total de columnas: 12 

|   | _NAME_  | COL1     | COL2  | COL3 | COL4     | COL5     | COL6   | COL7    | COL8 | COL9     |
|---|---------|----------|-------|------|----------|----------|--------|---------|------|----------|
| 1 | Surname | Kowalski | Cicha | Cwir | Wokulski | Andersen | Lubicz | Samosia | Eris | Kowalska |

Columnas

- ☒ Seleccionar todo
- ☒  \_NAME\_
- ☒  COL1
- ☒  COL2
- ☒  COL3

**Exercise 2.** Merge the following tables and surnames by variable stid and then transpose the output set by shoesize. Make variable names begin with 'person'.

```

data names;
  input stid name$;
  datalines;
100 Anna
101 Jan
102 Karol
103 Leszek
104 Jakub
110 Gosia
111 Donata
112 Justyna
113 Genowefa
114 Lucyna
116 Maja
115 Konstancja
105 Remek
  ;
run;

data shoes;
  input stid shoesize$;
  datalines;
101 34
101 65
102 32
103 21
100 43
110 43
111 35
112 44
113 32
114 34
116 25
115 34
105 34
  ;
run;

```

We do merge first:

```
proc sort data=shoes;
    by stid;
run;
proc sort data=surnames;
    by stid;
run;
proc sort data=names;
    by stid;
run;
data exercise2;
    merge names surnames shoes;
    by stid;
run;
```

Now we can:

```
proc sort data=exercise2;
    by shoesize;
run;
proc transpose data=exercise2 out=exercise2transposed(rename=(_name_=person))
prefix=person;
    var stid name surname shoesize;
    by shoesize;
run;
```

Tabla: WORK.EXERCISE2TRANPOSED Ver: Nombres de columna

Columnas

Nº total de filas: 36 Nº total de columnas: 6

☒ Seleccionar todo

☒ shoesize

☒ person

☒ person1

☒ person2

☒ person3

☒ person4

|    | shoesize | person   | person1  | person2  | person3 | person4 |
|----|----------|----------|----------|----------|---------|---------|
| 1  |          | stid     | 104      |          |         |         |
| 2  |          | name     | Jakub    |          |         |         |
| 3  |          | Surname  | Andersen |          |         |         |
| 4  |          | shoesize |          |          |         |         |
| 5  | 21       | stid     | 103      |          |         |         |
| 6  | 21       | name     | Leszek   |          |         |         |
| 7  | 21       | Surname  | Wokulski |          |         |         |
| 8  | 21       | shoesize | 21       |          |         |         |
| 9  | 25       | stid     | 116      |          |         |         |
| 10 | 25       | name     | Maja     |          |         |         |
| 11 | 25       | Surname  | Rajska   |          |         |         |
| 12 | 25       | shoesize | 25       |          |         |         |
| 13 | 32       | stid     | 102      | 113      |         |         |
| 14 | 32       | name     | Karol    | Genowefa |         |         |

**Exercise 3.** Create your own PROC TRANSPOSE example using ID.

```

data sales;
  input product_id region_id price name$;
  cards;
1011 1 12 t-shirt
1012 1 5 socks
1013 2 15 helmet
1014 3 20 shoes
  ;
run;
proc transpose data=sales out=sales_tr prefix=product_
id product_id;
run;

```

abla: WORK.SALES\_TR | Ver: Nombres de columna | | Filtrar: (ninguno)

Columnas ⓘ N° total de filas: 2 N° total de columnas: 5 ⏪

|   | _NAME_    | product_1011 | product_1012 | product_1013 |
|---|-----------|--------------|--------------|--------------|
| 1 | region_id | 1            | 1            | 2            |
| 2 | price     | 12           | 5            | 15           |

☒ Seleccionar todo  
☒ ▲ \_NAME\_  
☒ 🗄 product\_1011  
☒ 🗄 product\_1012  
☒ 🗄 product\_1013

**Exercise 4.** Find some data to import it to sas.

Let's create in our sas\_home directory a csv file called mydata.csv with the following data:

```

A , B , C
1 , 4 , 76
2 , 3 , 49
2 , 3 , 85
4 , 5 , 88
2 , 2 , 90

```

Now:

```

proc import out=csvfiledata
  datafile="/home/u63324691/soulutions/mydata.csv"
  dbms=csv
  replace;
  getnames=yes;
run;
proc print data=csvfiledata;

```

## Archivos y carpetas del servidor



## odaws01-euw1

## Accesos directos a carpetas

## Archivos (Inicio)

## resources

przedmioty.txt

sasuser.v94

## soulutions

bmitable.sas7bdat

by\_players\_asc.sas7bdat

by\_players\_dsc.sas7bdat

carsvat.sas7bdat

concatenated\_students2

erasmus\_students.sas7bdat

exercise2.sas7bdat

jugadores.sas7bdat

Laboratory 4.sas

Laboratory2.sas

Laboratory3.sas

Laboratory5.sas

Laboratory5.sas~

mydata.csv

national\_players.sas7bdat

people.sas7bdat

players\_salaries.sas7bdat

players\_teams.sas7bdat

players.sas7bdat

\*Laboratory5.sas x

CÓDIGO

LOG

RESULTADOS

DATOS DE SALIDA



## Tabla de contenido

| Obs | A | B | C  |
|-----|---|---|----|
| 1   | 1 | 4 | 76 |
| 2   | 2 | 3 | 49 |
| 3   | 2 | 3 | 85 |
| 4   | 4 | 5 | 88 |
| 5   | 2 | 2 | 90 |