Chapter 14 TRANSFORMING DATA WITH SAS FUNCTIONS

SAS FUNCTIONS THAT COMPUTE DESCRIPTIVE STATISTICS

|  |  |  |
| --- | --- | --- |
| Function | Syntax | Computes for non-missing values |
| SUM  MEAN  MIN  MAX  VAR  STD | sum(*argument\_1,…,argument\_n*)  mean(*argument\_1,…,argument\_n*)  min(*argument\_1,…,argument\_n*)  max(*argument\_1,…,argument\_n*)  var(*argument\_1,…,argument\_n*)  std(*argument\_1,…,argument\_n*) | sum  average  minimum value  maximum value  variance  standard deviation |

* Note that arguments must be separated by commas: sum(x, y, z).
* If arguments differ only in ending numbers, then they may be separated by a hyphen. The word OF must precede the list: sum(of x1-x3). If OF is omitted, then sum(x1-x3) computes x1 minus x3.

Example. The following data set contains student id number and scores on quizzes 1 through 5. For each student, the average, minimum, maximum, and standard deviation of scores are computed.

data quizzes;

input id$ quiz1-quiz5;

avg\_quiz=mean(quiz1, quiz2, quiz3, quiz4, quiz5);

min\_quiz=min(of quiz1-quiz5);

max\_quiz=max(of quiz1-quiz5);

std\_quiz=std (of quiz1-quiz5);

cards;

0495 94 97 95 94 100

8612 89 92 82 97 100

6236 94 89 85 91 94

;

proc print noobs;

run;

The output is

id quiz1 quiz2 quiz3 quiz4 quiz5 avg\_quiz min\_quiz max\_quiz std\_quiz

495 94 97 95 94 100 96.0 94 100 2.54951

8612 89 92 82 97 100 92.0 82 100 7.03562

6236 94 89 85 91 94 90.6 85 94 3.78153

MANIPULATING SAS DATE VALUES WITH FUNCTIONS

* The following functions are used to extract days, months, quarters, years, and days of the week from SAS date values.

|  |  |  |
| --- | --- | --- |
| Function | Syntax | Result |
| DAY | day(date) | day of month (1-31) |
| MONTH | month(date) | month(1-12) |
| QTR | qtr(date) | quarter (1-4) |
| YEAR | year(date) | year (4 digits) |
| WEEKDAY | weekday(date) | day of week (1=Sunday, …, 7=Saturday) |

Example. The following program outputs the day, month, quarter, year, and day of the week for January 24th, 2004.

data birthday;

input bday mmddyy10.;

day=day(bday);

month=month(bday);

qtr=qtr(bday);

yr=year(bday);

wkday=weekday(bday);

cards;

01/24/2004

;

proc format; value wkdayfmt 1='Sunday' 2='Monday'

3='Tuesday' 4='Wednesday' 5='Thursday' 6='Friday' 7='Saturday';

run;

proc print noobs;

format bday date9. wkday wkdayfmt.;

run;

The output is

bday day month qtr yr wkday

24JAN2004 24 1 1 2004 Saturday

* The functions **DATDIF** and **YRDIF** are used to compute the difference between two SAS date values expressed in days and years, respectively. The syntax is

datdif(*start\_date*,*end\_date*, *basis)*

yrdif(*start\_date*,*end\_date*,*basis*)

The basis specifies how SAS calculates the date difference. The options are

|  |  |  |  |
| --- | --- | --- | --- |
| Basis | Meaning | Valid in DATDIF | Valid in YRDIF |
| '30/360' | Uses a 30-day month and a 360-day year | yes | yes |
| 'ACT/ACT' | Uses the actual number of days or years between dates | yes | yes |
| 'ACT/360' | Uses the actual number of days and calculates the number of years as the number of days divided by 360 | no | yes |
| 'ACT/365' | Uses the actual number of days and calculates the number of years as the number of days divided by 365 | no | yes |

Example. The following program calculates duration of a project that started on March 23, 2017 and ended on April 23, 2017. The duration is expressed in days and years, using all the available options for basis.

data project;

input started date9. @11 ended date9.;

datdif1=datdif(started, ended, '30/360');

datdif2=datdif(started,ended, 'act/act');

yrdif1=yrdif(started,ended,'30/360');

yrdif2=yrdif(started,ended, 'act/act');

yrdif3=yrdif(started,ended, 'act/360');

yrdif4=yrdif(started,ended, 'act/365');

cards;

23MAR2017 23APR2017

;

proc print noobs;

format started ended date9.;

run;

The output is

started ended datdif1 datdif2 yrdif1 yrdif2 yrdif3 yrdif4

23MAR2017 23APR2017 30 31 0.083333 0.084932 0.086111 0.084932

* The function **MDY** creates a SAS date value from numeric values that represent the month, day, and year. The syntax is mdy(month, day, year)

Example. The following program creates a single date from variables month, day, and year.

data birthday;

input day month year;

mdy=mdy(month,day,year);

cards;

16 10 2005

;

proc print noobs;

format mdy date9.;

run;

The output is

day month year mdy

16 10 2005 16OCT2005

* The function **TODAY()** (equivalently, **DATE()**) returns the current date from the system clock as a SAS date value. The function **TIME()** returns the current time as a SAS time.

Example. The code below outputs the date and time when the code was run.

data current;

currentdate=today();

currenttime=time();

run;

proc print noobs;

format currentdate date9. currenttime time11.;

run;

The output is

currentdate currenttime

30JAN2018 11:07:49

CONVERTING DATA WITH FUNCTIONS

* The **INPUT** function coverts **character** values to **numeric** values. The syntax is

input(*variable\_name*, *numeric format*)

* The **PUT** function coverts **numeric** values to **character** values. The syntax is

put(*variable\_name*, *character* *format*)

Example. The following program coverts a character variable weeklyrate to a numeric variable rate, and a numeric variable weeks to a character variable nweeks.

data salary;

input weeklyrate$ weeks;

rate=input(weeklyrate,3.0);

nweeks=put(weeks, 1.);

cards;

350 7

;

proc contents;

run;

A partial output is

Alphabetic List of Variables and Attributes

# Variable Type Len

4 nweeks Char 1

3 rate Num 8

1 weeklyrate Char 8

2 weeks Num 8

THE CONCATINATION OPERATOR

The **concatenation operator (**||**)** is used to join character variables, or user specified characters enclosed between double quotation marks (e.g., " / " or " "

or " , "). The syntax is

*new\_variable*=*variable1*||"/" ||*variable2*||…

Example. The following program appends a title “M.D.” after a doctor’s last name.

data doctor;

input lastname$ 7.;

name=lastname||", M.D.";

cards;

Johnson

;

proc print noobs;

run;

The output is

lastname name

Johnson Johnson, M.D.