**data** a;

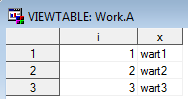
do i=**1** to **3**;

x='wart'||put(i,**1.**);

output;

end;

**run**;



**proc** **sql** noprint;

select i

into :mz

from a

;

%put \*\*\*&mz\*\*;



select i

into :mz separated by ' '

from a

;

%put \*\*\*&mz\*\*;



select i

into :mz separated by ' sdfsdf '

from a

;

%put \*\*\*&mz\*\*;



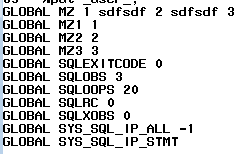
select i

into :mz1-:mz3

from a

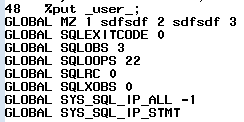
;

%put \_user\_;



%symdel mz1 mz2 mz3;

%put \_user\_;



select i

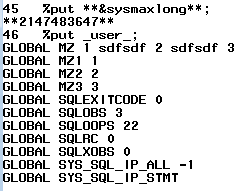
into :mz1-:mz&sysmaxlong

from a

;

%put \*\*&sysmaxlong\*\*;

%put \_user\_;



select i,x

into :mzi,:mzx

from a

;

%put \*\*\*&mzi\*\*\*&mzx\*\*\*;



\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*;

**data** zbior;

input x$ y;

cards;

A 1

A 2

A 3

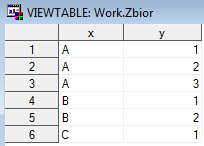
B 1

B 2

C 1

;

**run**;



**%macro** t(zbior,zmienna);

%local wartosci1 wartosci2;

proc sql noprint;

select distinct &zmienna

into :wartosci1 separated by ' '

from &zbior;

proc sql noprint;

select distinct 'when ("'

|| compress(&zmienna)

|| '") output '

|| &zmienna

into :wartosci2 separated by ';'

from &zbior;

data &wartosci1;

set &zbior;

select(&zmienna);

&wartosci2;

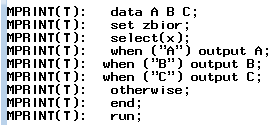
otherwise;

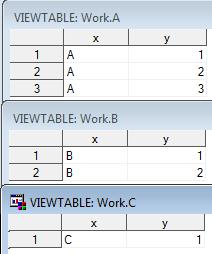
end;

run;

**%mend**;

%***t***(zbior,x)

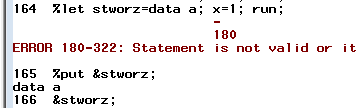




%let stworz=data a; x=**1**; **run**;

%put &stworz;

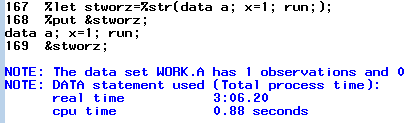
&stworz;



%let stworz=%str(data a; x=**1**; **run**;);

%put &stworz;

&stworz;





libname bib 'C:\Users\sommerm\Dropbox\sas';

**%macro** stworz(biblioteka,zbior);

data &biblioteka.&zbior;

x=**1**;

run;

**%mend**;

%***stworz***(bib,a)





**%macro** stworz(biblioteka,zbior);

data &biblioteka.**.**&zbior;

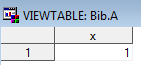
x=**1**;

run;

**%mend**;

%***stworz***(bib,a)





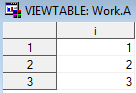
**data** a;

do i=**1** to **3**;

output;

end;

**run**;



\* chcemy znaleźć liczbę obserwacji;

\* zbioru a;

**data** \_null\_;

set a nobs=ile;

call symput('ile',put(ile,**1.**));

**run**;

%put \*\*&ile\*\*;



\* tak trochę lepiej:;

**data** \_null\_;

set a nobs=ile;

if \_n\_=**1** then call symput('ile',put(ile,**1.**));

stop;

**run**;

%put \*\*&ile\*\*;



\* a tak jeszcze lepiej:;

**data** \_null\_;

call symput('ile',put(ile,**1.**));

stop;

set a nobs=ile;

**run**;

%put \*\*&ile\*\*;



**data** b;

array tablica(**3**) x1-x3 (**3**\***100**);

y=vname(tablica(**2**));

**run**;



\* ale można to wszystko zrobić jeszcze \* sprytniej!;

\* dictionary tables;

**proc** **sql**;

select memname,

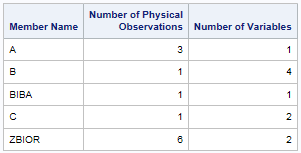
nobs,

nvar

from dictionary.tables

where libname='WORK'

;



select memname

from dictionary.columns

where libname='WORK' and name='x2'

;



\* columns,

\* extfiles,

\* indexes,

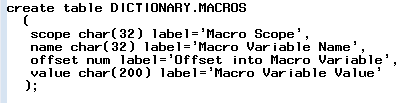
\* macros,

\* members,

\* tables,

\* catalogs;

describe table dictionary.macros;



\* nie trzeba więc pamiętać nazw, tylko

\* pisać w ten sposób i sobie odczytać;

**%macro** histogram(zbior);

%let bib=%scan(&zbior,1,'.');

%let zb=%scan(&zbior,2,'.');

proc sql noprint;

select name

into :zmienne separated by ' '

from dictionary.columns

where libname=%upcase("&bib") and memname=%upcase("&zb")

and type='num'

;

proc univariate data=&zbior;

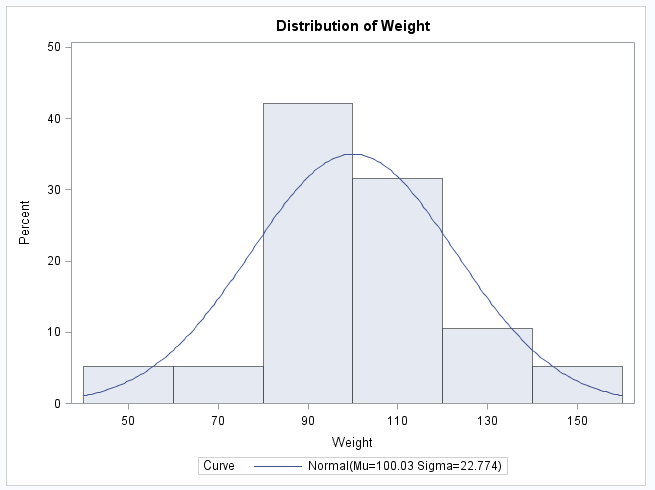
var &zmienne;

histogram / normal (mu=est sigma=est color=red);

run;

**%mend**;

%***histogram***(sashelp.class)



**data** c;

set sashelp.vmember;

where libname='WORK';

**run**;

