/\*1. tworzy bibliotekę engine v9 wskazującą na katalog c:\sas\\*/

libname bib v9 'C:\sas\';

/\*2. tworzy bibliotekę engine v8 wskazującą na katalog c:\sas\\*/

libname bib v8 'C:\sas\';

/\*3. tworzy bibliotekę odbcwskazującą na sterownik odbc o nazwie CRM\*/

libname bib odbc dsn=CRM;

/\*4. czyta go instrukcją SET w jednym przebiegu datastepu\*/

**data** w;

do i=**1** to n\_obs;

set sashelp.class nobs=n\_obs;

przebieg=\_n\_;

output;

end;

run;

/\*5. czyta go i tworzy zbiór WYNIK z nową zmienną o nazwie DWA, która ma dwie pierwsze litery imienia\*/

**data** w;

set sashelp.class;

dwa=substr(Name,**1**,**2**);

run;

/\*a jak wydobyć dwie ostatnie litery imienia?\*/

**data** w;

set sashelp.class;

l=length(Name);

dwa=substr(Name,l-**1**,**2**);

run;

/\*6. wybiera tylko wiersze z wiekiem większym niż 12 lat\*/

**data** w;

set sashelp.class;

where age>**12**;

run;

/\*7. wybiera wiersze o imionach na literę A\*/

**data** w;

set sashelp.class; /\*(WHERE =(name=:'A'))\*/;

where substr(name,**1**,**1**)='A';

run;

/\*8. tworzy nową kolumnę DATA\_URODZENIA na podstawie wieku i dzisiejszej daty w systemie\*/

**data** w;

set sashelp.class;

data\_urodzenia=put(intnx('year',**"&sysdate"d**,-age),year4.);

/\*w dacie systemowej weź rok i odejmij od niego wartość z age\*/

run;

/\*lub\*/

**data** w;

format data\_urodzenia YEAR.;

set sashelp.class;

data\_urodzenia=(today()-age\***365**);

run;

/\*9. tworzy nową kolumnę i liczy liczbę liter w imieniu\*/

**data** w;

set sashelp.class;

liczba\_liter=length(name);

run;

/\*10. tworzy nową kolumnę: flagę zerojedynkową posiadania literki a w imieniu\*/

**data** w;

set sashelp.class;

if index(name,'a')=**0** and index(name,'A') =**0** then

/\*index zwraca numer pozycji w stringu gdzie występuje szukana liter, jak nie ma to 0 \*/

czy\_a=**0**;

else

czy\_a=**1**;

run;

/\*11. tworzy nową kolumnę konkatenując imię z wiekiem i płcią, np. Alfred-14-M\*/

**data** w;

set sashelp.class;

all=compress(name)||'-'||compress(age)||'-'||sex;

/\*albo\*/

**data** egzamin.zad11;

set sashelp.class;

laczenie=catx("-",name,age,sex);

run;

run;

/\*12. rozmnaża losowo każdą obserwację od 1 do 10 razy\*/

**data** w;

set sashelp.class;

/\*do i=1 to ceil(rand('Uniform')\*10); \*/

do i=**1** to int(ranuni(**0**) \* **10** + **1**);

output;

end;

drop i;

run;

/\*13. do zbioru WYNIK zapisuje tylko co drugą obserwację\*/

**data** wynik;

do i=**1** to obs;

set sashelp.class nobs=obs;

if mod(i,**2**)=**0** then output;

end;

run;

/\*liczba obserwacji w zbiorze- da się prosciej?\*/

**data** w;

put nobs=;

stop;

set sashelp.class nobs=nobs;

run;

/\*14. sortuje go po płci rosnąco i wieku malejąco\*/

**proc** **sort** data=sashelp.class out=w;

by sex descending age;

**run**;

/\*15. na podstawie mechanizmu przetwarzania w grupach wylicza liczbę kobiet i mężczyzn\*/

**proc** **sort** data=sashelp.class out=w;

by sex;

**run**;

**proc** **freq** data=w;

tables sex;

**run**;

/\*albo\*/

**proc** **means** data=sashelp.class noprint;

class sex;

output out=w n()=n;

**run**;

/\*16. na podstawie mechanizmu przetwarzania w grupach wylicza minimum i maksimum wieku dla płci\*/

**proc** **means** data=sashelp.class ;

class sex;

var age;

output out=w min()=min max()=max;

**run**;

/\*albo\*/

**proc** **means** data=sashelp.class min max ;

class sex;

var age;

**run**;

/\*17. na podstawie mechanizmu przetwarzania w grupach wylicza średnią wieku dla płci\*/

**proc** **means** data=sashelp.class ;

class sex;

var age;

output out=w mean()=mean;

**run**;

/\*18. na podstawie zbioru SASHELP.AIR wylicza skumulowaną liczbę tysięcy pasażerów, data po dacie - rosnąco, na podstawie kolumny AIR\*/

**data** w;

set sashelp.air;

cum=sum(cum,air);

retain cum;

run;

/\*19. na podstawie funkcji lag wyrzuca do WYNIK tylko wiersze, dla których poprzedni wiersz był kobietą\*/

**data** w;

set sashelp.class;

lagged=lag(sex);

run;

**data** w;

set w;

where lagged='F';

drop lagged;

run;

/\*20. tworzy nową kolumnę jako sumę lat ostatnich dwóch wierszy\*/

**data** w;

set sashelp.class;

suma=sum(age,lag(age));

run;

/\*21. który do zbioru WYNIK wyrzuca tylko obserwacje o numerach od 5 do 7\*/

**data** wynik;

set sashelp.class (firstobs=**5** obs=**7**);

run;

/\*22. który do zbioru WYNIK wyrzuca tylko pierwszą obserwację\*/

**data** wynik;

set sashelp.class (obs=**1**);

run;

/\*23. który do zbioru WYNIK wyrzuca tylko ostatnią obserwację\*/

**data** w;

set sashelp.class end=a;

if a=**1** then output;

run;

/\*24. który tworzy nowe zmienne wiek1 do wiek20 i tak je uzupełnia, by np. dla wieku równego 10 kolumna wiek10 miała wartość 1 a inne 0\*/

**data** w;

array t(**20**) w1-w20;

set sashelp.class;

do i=**1** to **20**;

if age=i then t(i)=**1**;

else t(i)=**0**;

end;

run;

/\*25. ze zbioru SASHELP.ZHC wybrać tylko te obserwacje, których kod PCMS jest w zbiorze SASHELP.ZTC\*/

**proc** **sql**;

create table w as

select \* from sashelp.zhc where pcms in (select pcms from sashelp.ztc);

**quit**;

/\*26. do zbioru SASHELP.ZHC dokleić kolumnę EUC ze zbioru SASHELP.ZTC łącząc po PCMS\*/

**proc** **sql**;

create table w as

select a.\*, b.euc from sashelp.zhc as a left join sashelp.ztc as b

on a.pcms=b.pcms;

**quit**;

/\*27. zrobić test łączenia tabel SASHELP.ZHC i SASHELP.ZTC łącząc po PCMS, ile się połączyło, ile zostało obserwacji w pierwszym a ile w drugim\*/

**proc** **sort** data=sashelp.ztc out=w;

by pcms;

**run**;

**proc** **sort** data=sashelp.zhc out=s;

by pcms;

**run**;

**data** pol lewy prawy;

merge work.s(in=zhc) work.w(in=ztc);

by pcms;

if zhc and ztc then output pol;

if zhc and not ztc then output lewy;

if not zhc and ztc then output prawy;

run;

/\*28. stworzyć zbiór WYNIK, który ma kolumny od age11 do age16 i name.

Dla przykładu gdy uczeń ma 11 lat to zmienna age11 jest wypełniona i posiada wartość zmiennej płeć, a inne zmienne są puste\*/

**data** wynik;

array t(**6**) $ age11-age16;

set sashelp.class;

do i=**1** to **6**;

if age=i+**10** then t(i)=sex;

end;

keep name a1:;

run;

/\*29. na podstawie zbioru SASHELP.CARS stworzyć zbiór WYNIK, który posiada informacje o modelach samochodów

w wielu wierszach. Dla danego modelu w wierszach istnieją dwie dodatkowe kolumny o nazwach: PROPERTY i VALUE,

w pierwszej jest nazwa kolumny zbioru oryginalnego a w drugiej jej wartość\*/

**data** zad29;

length property value $**200**;

do i=**1** to obs;

set sashelp.cars nobs=obs;

array zmienne(\*) \_NUMERIC\_;

array zmiennechar(\*) \_CHARACTER\_;

do j=**2** to **11**;

property=vname(zmienne(j));

value=zmienne{j};

output;

if (j >= **3** AND j<=**7**) then do;

property=vname(zmiennechar{j});

value=zmiennechar{j};

output;

end;

end;

end;

keep model property value;

run;

/\*30. liczy odległość Levenshteina pomiędzy imieniem a słowem Alfred\*/

**data** zad30;

imie1='alfred';

imie2='patrycja';

odleglosc=complev(imie1,imie2);

run;

/\*31. tworzy 19 zbiorów: od name1 do name19, każdy z jedną obserwacją, pierwszy z pierwszą obserwacją zbioru wejściowego, a ostatni z ostatnią\*/

**%MACRO** ***tworz***;

%DO I = **1** %TO **19**;

data name&I;

do i=**1** to obs;

set sashelp.class nobs=obs;

if i=&I then output;

end;

drop i;

run;

%END;

**%MEND** tworz;

%***tworz***

/\*32. tworzy zbiór WYNIK z powtórzonymi obserwacjami tyle razy ile wynosi wartość zmiennej wiek\*/

**data** wynik;

set sashelp.class;

do i =**1** to age;

output;

end;

run;

/\*33. tworzy tablicę dwuwymiarową w IML biorąc dwie pierwsze obserwacje i kolumny wiek i wagę oraz oblicza jej wyznacznik \*/

**PROC** **IML**;

USE sashelp.class;

/\*list point {1 2} var {age weight};\*/

read point {**1** **2**} var {age weight} into x;

print x;

c=det(x);

print c;

**quit**;

/\*część druga\*/

/\* 1. tworząc format grupy wiekowej: do 13 lat włącznie i powyżej, wyznaczyć procedurą

MEANS liczby i uczniów i średnie wartości wieku \*/

**proc** **format**;

value wiek

low-**13**='Mały'

**13**-high='Duży'

;

**run**;

**proc** **means** data=sashelp.class mean;

class age;

format age wiek.;

var age;

**run**;

/\*2. wyznaczyć dwuwymiarową tablicę liczebności wieku i płci\*/

**proc** **freq** data=sashelp.class ;

table age\*sex /out=freq nocum nopercent;

**run**;

/\*3. wyznaczyć tablicę liczebności wieku z jej kumulatywnymi statystykami liczebności i procentu\*/

**proc** **freq** data=sashelp.class;

table age /out=freq outcum;

**run**;

/\*4. tworzy raport HTML z tabelą z boczkiem imieniem a nagłówkiem płcią i wypełnionymi wartościami wiekiem\*/

ods listing close;

ods html

path='C:\Users\Pati\Desktop' (url=none) ;

**proc** **tabulate** data=sashelp.class;

class name sex;

var age;

table name,sex='płec'\*age;

**run**;

ods html close;

ods listing;

/\*5. tworzy raport w PDF na podstawie SASHELP.PRDSALE

z informacjami o aktualnej sumie sprzedaży i procentach po latach i regionie z podsumowaniami\*/

ods listing close;

ods pdf

file='C:\Users\Pati\Desktop\rap.pdf'

;

**proc** **tabulate** data=sashelp.prdsale;

class year region;

var actual;

table year='' all,actual='Sprzedaż'\*region ;

**run**;

ods pdf close;

ods listing;

/\*6. tworzy seryjny raport dla każdego regionu, czyli tworzy tyle HTMLi ile regionów i w każdym raportuje sumę

sprzedaży w rozbiciu po latach i produktach z podsumowaniami\*/

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

select distinct region into :nazwy separated by '#'/\*wszystkie wartości do 1 makrozmiennej\*/

from sashelp.prdsale order by region;

**quit**;

%let n\_name=&sqlobs;

%put &n\_name\*\*\*&nazwy;

**%MACRO** ***DO\_MEAN***;

%DO I = **1** %TO &n\_name;

ods listing close;

ods html body="C:\Users\Pati\Desktop\zad6&I..htm";

proc tabulate data=sashelp.prdsale (where=(region="%scan(&nazwy, &I, '#')"));

class product year;

var actual;

table product='' all, mean=''\*actual=''

\*(year='' all)\*f=nlnum12.2

/ box='Actual mean';

run;

ods html close;

ods listing;

%END;

**%MEND** DO\_MEAN;

%***DO\_MEAN***;

/\*7. tworzy listę makrozmiennych name1 do name19 wstawiając do nich nazwy uczniów w kolejności alfabetycznej\*/

**proc** **sql** noprint;/\*tworzenie makrozmiennych z poziomu SQL\*/

select name into :name1-:name99999 /\*możemy wpisać dużo ale i tak stworzy tyle ile jest zmiennych - gorzej jak damy mniejszą liczbe - nie zostanie stworzonych np 19 makrozmiennych, tylko 9\*/

from sashelp.class order by name;

**quit**;

%let n\_name=&sqlobs;/\*zmienna ta przechowuje numer ostatniej wykonywanej obserwacji w zapytaniu sql-owym\*/

%put &name1;

%put \_user\_;

/\*/8. tworzy listę makrozmiennych o nazwach uczniów wstawiając do nich ich płcie\*/

**data** \_null\_;

set sashelp.class;

call symput(name,trim(sex));

run;

%put &Alfred;

/\*9. tworzy jedną długą makrozmienną wstawiając do niej listę imion uczniów w kolejności alfabetycznej oddzielanych znakiem #\*/

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

select name into :nazwy separated by '#'/\*wszystkie wartości do 1 makrozmiennej\*/

from sashelp.class order by name;

**quit**;

%let n\_name=&sqlobs;

%put &n\_name\*\*\*&nazwy;

/\*10. tworzy makroprogram z parametrem płeć, który listuje (PROC PRINT) obserwacje dla wybranej płci\*/

**%macro** drukuj(plec);

proc print data=sashelp.class (where=(sex="&plec"));

run;

**%mend**;

%***drukuj***(M);

/\*11. tworzy wykres słupkowy z średnim wiekiem dla płci\*/

**proc** **gchart** data=sashelp.class;

vbar3d sex / type=mean

sumvar=age;

**run**;

/\* 12. tworzy wykres liniowy szeregu czasowego liczby pasażerów AIR ze zbioru SASHELP.AIR\*/

symbol v=star i=join;

title "wykres liniowy";

**proc** **gplot** data=sashelp.air;

plot air\*date;

**run**;

**quit**;

/\*13. liczy percentyle zmiennej wiek: 5, 25, 50, 75 i 95\*/

**proc** **univariate** data=sashelp.class;

var age;

output out=percentile pctlpre=P\_ pctlpts= **5** **25** **50** **75** **85**;

**run**;

**proc** **print** data=percentile;

**run**;

/\*14. liczy odchylenie standardowe, średnią i rozstęp między-kwartylowy \*/

**proc** **univariate** data=sashelp.class ;

var age;

output out=stat mean=mean qrange=qrange std=std ;

**run**;

**proc** **print** data=stat;

**run**;

/\*15. wrzuca do EXCELA do komórek A1,A2i A3 liczby 3,4,5.\*/

/\*16. tworzy zbiór danych w którym są zapisane nazwy plików z katalogu c:\\*/

filename f pipe 'dir c:\ /w /a:d /b';/\*wskazujemy na plik zewnętrzny - ale przez komendę dosową dir\*/

**data** w;/\*używamy to np do plikow które trzeba czesto czytac - robimy liste i je potem czytamy\*/

infile f;

input;

a=\_infile\_;

run;

/\*17. tworzy pliki tekstowe o nazwach takich jak wartości zmiennej PRODUCT zbioru SASHELP.PRDSALE i

wrzuca do nich informacje o aktualnej sprzedaży i roku\*/

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

select distinct product into :nazwy separated by '#'/\*wszystkie wartości do 1 makrozmiennej\*/

from sashelp.prdsale order by product;

**quit**;

%let n\_name=&sqlobs;

%put &n\_name\*\*\*&nazwy;

%let dir=C:\Users\Pati\Desktop\egz;

**%macro** ***rob***;

%do i=**1** %to **20**;

data \_null\_;

file "&dir.%scan(&nazwy, &I, '#').txt";

do l=**1** to obs;

set sashelp.prdsale (where=(product="%scan(&nazwy, &I, '#')") keep=actual year PRODUCT) nobs=obs;

put actual year;

end;

run;

%end;

**%mend**;

%***rob***;

/\*18. tworzy wykres pudełkowy dla zmiennej wiek po płci\*/

**proc** **sort** data=sashelp.class out=klasa;

by sex;

**run**;

**proc** **boxplot** data=klasa;

plot age\*sex /

boxstyle = schematic ;

**run**;

**quit**;

/\*19. tworzy wykres rozproszenia wieku i wagi\*/

**proc** **gplot** data=sashelp.class;

plot weight\*age;

**run**;

**quit**;

/\*20. liczy średnie odporne, np. obcięte lub windsorowane dla wieku\*/

**proc** **univariate** data=sashelp.class trimmed=**1** **0.01** **0.1** winsorized=**1** **0.01** **0.1** ;

var age;

**run**;

/\*21. tworzy jego listing do LaTeXa\*/

ods tagsets.tablesonlylatex

file="C:\Users\Pati\Desktop\egz\egzamin.tex" (notop nobot);

**proc** **univariate** data=sashelp.class trimmed=**1** **0.01** **0.1** winsorized=**1** **0.01** **0.1** ;

var age;

**run**;

ods tagsets.tablesonlylatex close;

ods listing;

/\*22. przerzuca statystyki dopasowania do rozkładu normalnego zmiennej wiek z procedury UNIVARIATE

do zbioru danych ZGODNOSC\*/

**proc** **univariate** data=sashelp.class normal;

var age;

output out=zgodnosc normaltest=normal ;

**run**;

/\*część 3\*/

/\*1. rysuje histogram zmiennej wiek z naniesioną krzywą Gaussa\*/

**proc** **univariate** data=sashelp.class;

var age;

HISTOGRAM / NORMAL (COLOR=RED W=**5**) ;

**run**;

/\*2. wyznacza test chi-kwadrat dla płci i wieku, traktując obie jako nominalne, policzyć p-value\*/

**proc** **freq** data=sashelp.class;

tables sex\*age /

chisq ;

**run**;

/\*3. wyznacza statystykę V-Cramera dla płci i wieku\*/

**proc** **freq** data=sashelp.class;

tables sex\*age /chisq ;

**run**;

/\*4. wyznacza statystę Giniego dla płci i wieku\*/

**proc** **univariate** data=sashelp.class;

var age ;

class sex;

output out=zad4 gini=gini;

**run**;

/\*5. i 6 liczy współczynnik korelacji Spearmana i pearsona dla wieku i wagi\*/

**proc** **corr** data=sashelp.class outs=spearman outp=pearson ;

var age weight;

**run**;

/\*7. liczy statystykę Kołmogorowa—Smirnowa testu zgodności empirycznego rozkładu wieku z teoretycznym normalnym\*/

**proc** **univariate** data=sashelp.class normal;

var age;

**run**;

/\*8. liczy statystykę Shapiro—Wilka testu zgodności empirycznego rozkładu wieku z teoretycznym normalnym\*/

**proc** **univariate** data=sashelp.class normal;

var age;

/\*output out=asd normal=normalny; \*/

**run**;

/\*9. liczy korelację cząstkową Pearsona dla wieku i wagi z wyrugowanym wpływem wzrostu\*/

**proc** **corr** data=sashelp.class rank pearson;

var weight age ;

partial height;

**run**;

/\*10. liczy współczynniki regresji liniowej dla wieku i wagi\*/

**proc** **reg** data=sashelp.class outest=e;

model age=height weight;

**run**;

**quit**;

/\*11. liczy współczynnik determinacji regresji liniowej dla wieku i wagi\*/

**proc** **reg** data=sashelp.class outest=e;

model age=height weight/rsquare;

**run**;

**quit**;

/\*12. przeprowadza diagnostykę modelu regresji liniowej wieku i wagi- czyli co?\*/

**proc** **reg** data=sashelp.class;

model age=weight / spec;

/\*spec to test jednorodnosci wariancji\*/

output out=reszty r=r;

plot r. \* (p. age);\*dwa rodzaje wykresu reszt;

plot student. \* obs.;

plot nqq. \* student.;\*normalnosc reszt;

**run**;

**quit**;

**proc** **univariate** data=reszty mu0=**0** alpha=**0.01**;

var r;

**run**;

/\*13. bada homoskedastyczność modelu regresji liniowej wieku i wagi\*/

**proc** **reg** data=sashelp.class;

model age=weight / spec;

output out=reszty r=r;

plot r.\*p. /\*reszty vs predicted- tu widac heteroskedastycznosc ale statystyka spec wazniejsza bo tu malo obs\*/;

**run**;

**quit**;

/\*14. bada normalność reszt modelu regresji liniowej wieku i wagi\*/

**proc** **reg** data=sashelp.class;

model age=weight / spec;

output out=reszty r=r;

plot nqq. \* student.;\*normalnosc reszt;

**run**;

**quit**;

/\*a proc univariate pokazuje też testy na normalnosc\*/

goptions reset=all;

**proc** **univariate** data=reszty normal;

var r;

qqplot r / normal(mu=est sigma=est);

**run**;

/\*15. przeprowadza selekcję zmiennych metodami krokowymi dla modelu regresji liniowej wielorakiej

wieku zależnego od wagi i wzrostu\*/

**proc** **reg** data=sashelp.class outest=e;

model age=height weight/ selection=FORWARD slentry=**0.1**;

**run**;

**quit**;

**proc** **reg** data=sashelp.class outest=e;

model age=height weight/ selection=BACKWARD slstay=**0.1**;

**run**;

**quit**;

**proc** **reg** data=sashelp.class outest=e;

model age=height weight/ selection=STEPWISE slentry=**0.1** slstay=**0.1**;

**run**;

**quit**;

/\*16. liczy mierniki współliniowości dla modelu regresji liniowej wielorakiej wieku zależnego od wagi i wzrostu\*/

**proc** **reg** data=sashelp.class outest=e;

model age=height weight/ TOL VIF COLLIN;

**run**;

**quit**;

/\*17. liczy współczynniki determinacji dla wszystkich modeli regresji liniowej wielorakiej wieku zależnego od wagi i wzrostu\*/

**proc** **reg** data=sashelp.class outest=e;

model age=height weight/ selection=rsquare;

/\*chodzi o tą tabelkę z trzema modelami i r2 policzone ?\*/

**run**;

**quit**;

/\*18. wyznacza wartości odstające i/lub wpływające w modelu regresji liniowej wieku i wzrostu\*/

%let p=2;

%let n=19;

ods listing close;

ods output OutputStatistics=statystyki;

**proc** **reg** data=sashelp.class outest=e;

model age=height/ influence vif collin spec collinoint r cli clm p;

**run**;

**quit**;

ods output close;

ods listing;

**data** nietypowe;

set statystyki;

if

abs(RStudent)>**3**

or HatDiagonal>(**2**\*&p/&n)

or CooksD > (**4**/&n)

or abs(DFFITS)> **2**\*sqrt(&p/&n)

or min(of DFB:) > **2**/sqrt(&n)

or max(of DFB:) < -**2**/sqrt(&n)

or abs(CovRatio-**1**) > **3**\*&p/&n

;

run;

/\*19. liczy procent wyjaśnienia wariancji przez główne składowe dla zmiennych wiek, waga i wzrost\*/

**proc** **princomp** data=sashelp.class out=AGS n=**3**;

var age weight height;

**run**;

/\*dla jednej glownej skladowej-87%, dla dwoch(optymalne wg kryt osypiska)- 96% \*/

/\*20. liczy prognozę wieku zależnego od wagi wynoszącej 80 w modelu regresji liniowej wieku i wagi\*/

**data** klasa;

weight=**80**;

output;

do i=**1** to obs;

set sashelp.class nobs=obs;

output;

end;

stop;

run;

**proc** **reg** data=klasa;

model age=weight ;

output out=predyk p=predykcja;

**run**;

**quit**;

/\*21. wyznacza klastry (skupienia) zmiennych (PROC VARCLUS) dla wszystkich zmiennych numerycznych zbioru SASHELP.CARS i

dla wyjaśnionej wariancji (proporcji) 80%\*/

**proc** **corr** data=sashelp.cars outp=cars\_matrix noprint; /\*tylko dla sprawdzenia korelacji, nie jest niezbedna w poleceniu\*/

var \_NUMERIC\_;

**run**;

ods listing close;

ods output;

**proc** **varclus** data=sashelp.cars PROPORTION=**0.8**;

var \_NUMERIC\_;

**run**;

ods output close;

ods listing;