```sas
/* question of the exam: page 3 - problem 2 - 2.1 */
/************************************************/

libname shahram '/folders/myfolders/Exam/' inencoding=asciiany;
filename city '/folders/myfolders/exam/city.xlsx';
filename city2 '/folders/myfolders/exam/city.xlsx';
filename test21 '/folders/myfolders/exam/free-zipcode-database-Primary.csv';

proc import datafile=city out=WORK.city dbms=xlsx replace; Run;
proc import datafile=city2 out=WORK.city2 dbms=xlsx replace; Run;
```

```sas
proc sql;
  select a.city format=$10., a.state, a.lat 'Latitude', a.long 'Longitude',
         b.city format=$10., b.state, b.lat 'Latitude', b.long 'Longitude',
         sqrt((a.lat - b.lat)**2 + (a.long - b.long)**2) as distance format=6.1
    from city a, city b
   where a.city ne b.city
     and calculated distance in
       (select min(sqrt((c.lat - d.lat)**2 + (c.long - d.long)**2))
          from city c, city d
         where c.city = a.city
           and c.state = a.state
           and c.city ne d.city)
   order by a.city;
quit;
```

City	State	Latitude	Longitude	City	State	Latitude	Longitude	distance
AKIACHAK	AK	60.87	-161.42	AKIAK	AK	60.9	-161.29	0.1
AKIAK	AK	60.9	-161.29	AKIACHAK	AK	60.87	-161.42	0.1
AKUTAN	AK	54.13	-165.88	TOKSOOK BA	AK	60.54	-165.14	6.5
ALAKANUK	AK	62.44	-164.38	TOKSOOK BA	AK	60.54	-165.14	2.0
ALEKNAGIK	AK	59.24	-158.62	ANIAK	AK	61.2	-158.6	2.0
ALLAKAKET	AK	66.57	-152.95	ANAKTUVUK	AK	68.15	-151.71	2.0
AMBLER	AK	67.08	-157.9	ATQASUK	AK	70.48	-157.39	3.4
ANAKTUVUK	AK	68.15	-151.71	NUIQSUT	AK	69.83	-152.14	1.7
ANCHOR POI	AK	59.82	-151.59	ANCHORAGE	AK	61.45	-150.63	1.9
ANCHORAGE	AK	61.2	-149.82	JBER	AK	61.27	-149.79	0.1
ANCHORAGE	AK	61.2	-149.82	JBER	AK	61.27	-149.79	0.1
ANGOON	AK	57.44	-134.48	BEAVER	AK	66.32	-146.77	15.2
ANIAK	AK	61.2	-158.6	ALEKNAGIK	AK	59.24	-158.62	2.0
ANVIK	AK	62.63	-160.2	AKIAK	AK	60.9	-161.29	2.0
ARCTIC VIL	AK	68.09	-145.57	BEAVER	AK	66.32	-146.77	2.1
ATQASUK	AK	70.48	-157.39	BARROW	AK	71.28	-156.78	1.0
BARROW	AK	71.28	-156.78	ATQASUK	AK	70.48	-157.39	1.0
BEAVER	AK	66.32	-146.77	ARCTIC VIL	AK	68.09	-145.57	2.1
BETHEL	AK	60.93	-161.8	AKIACHAK	AK	60.87	-161.42	0.4
INDIAN	AK	61	-149.43	ANCHORAGE	AK	61.17	-149.63	0.3

```
/* solution */
data test30;
  set city(rename=(city=city1 state=state1 lat=lat1 long=long1) keep=city state lat long);
  do i = 1 to n;
    set city2(rename=(city=city2 state=state2 lat=lat2 long=long2) keep=city state lat long
    distance = sqrt((lat1 - lat2)**2 + (long1 - long2)**2);
    if (distance ne 0) then output;
  end;
run;

proc sort data=test30 out=test31;
  by city1 distance;
run;

data test32;
  set test31;
    by city1;
  if first.city1 then output;
run;

/************************************************/
```

```
data _null_ /nesting;
   do i = 1 to 10;
      do j = 1 to 5;
         put i= j=;
      end;
   end;
run;

data one;
  input id $ fruit $;
datalines;
a apple
a apple
b banana
c coconut
c coconut
c coconut
;
```

```
68 data two;
69   input id $ color $;
70 datalines;
71 a amber
72 b brown
73 b black
74 c cocoa
75 c cream
76 ;
77
78 data every_combination;
79
80   /* Set one of your data sets, usually the larger data set */
81   set two;
82   do i=1 to n;
83
84     /* For every observation in the first data set,     */
85     /* read in each observation in the second data set */
86     set one point=i nobs=n;
87     output;
88   end;
89 run;
```

```
91 proc print data=every_combination;
92 run;
93
94 data test33;
95   do obsnum = 1 to 5;
96     set work.test point=obsnum nobs=n; /* hatman bayad az nobs ham dar kenare point estefac
97     if _error_ then abort;
98     output;
99   end;
100  stop;
101 run;
102
103 data test34;
104   set work.test nobs=last; /* in dastur (nobs) hich kari nemikonad - banabarin dasture nobs
105 run;
```

Obs	id	color	fruit
1	a	amber	apple
2	a	amber	apple
3	b	amber	banana
4	c	amber	coconut
5	c	amber	coconut
6	c	amber	coconut
7	a	brown	apple
8	a	brown	apple
9	b	brown	banana
10	c	brown	coconut
11	c	brown	coconut
12	c	brown	coconut
13	a	black	apple
14	a	black	apple
15	b	black	banana
16	c	black	coconut
17	c	black	coconut
18	c	black	coconut
19	a	cocoa	apple
20	a	cocoa	apple

```
107 /* question of the exam: page 4 - problem 2 - 2.2 */
108 /*****************************************************/
109
110 data test35;
111   input a $ b;
112   datalines;
113 a 1
114 b 2
115 c .
116 d 4
117 ;
118 run;
119
120 data test36;
121   input a $ b;
122   datalines;
123 a 1
124 b 2
125 c .
126 d 4
127 e .
128 ;
```

```
proc sql;
  select test35.a 'One', test35.b, test36.a 'Two', test36.b
    from test35, test36
   where test35.b = test36.b and test35.b is not missing;
quit;

/* solution */

data test37;
  set test35;
  where b is not missing;
  rename a=One;
run;

data test38;
  set test36;
  rename a=Two;
run;

proc sort data=test37; by b; run;
proc sort data=test38; by b; run;
```

```
data test39;
  merge test37(in=one1) test38(in=two1); by b;
  if (one1 = 1) and (two1 = 1);
run;
/*************************************************/
```

```
/* question of the exam: page 4 - problem 2 - 2.3 */
/*************************************************/

data xsales;
  input id $ p_no $ quantity price;
  datalines;
1 1 10 20
1 2 15 25
2 1 17 23
2 3 21 35
3 3 21 32
3 4 52 12
3 5 14 25
;
run;
```

```
data xparts;
  input p_no $ name $;
  datalines;
1 kabab
2 table
3 chair
4 pen
5 board
6 pencil
;
run;
data empl;
  input id $ name $ dob mmddyy10.;
  datalines;
1 John 01/02/1975
2 Morgan 01/03/1976
3 Jim 05/02/1977
4 Anna 01/04/1978
;
run;
```

```
/* solution */
title 'sales total';
proc sql;
  select id, count(total) 'N', mean(total) 'Mean', min(total) 'Min', max(total) 'Max', std
    from (
          select xsales.id, price * quantity as total
            from xsales left join xparts on xsales.p_no = xparts.p_no
                        left join empl on xsales.id = empl.id
           group by xsales.id
          )
    group by id;
quit;

/*************************************************/
```

```
/* question of the exam: page 5 - problem 2 - 2.4 */
/*************************************************/
data empl1;
  input idnum $ name $ dob mmddyy10. sex $ salary;
  datalines;
1 John 01/02/1975 M 250000
2 Morgan 01/03/1976 M 300000
3 Jim 05/02/1977 M 350000
4 Anna 01/04/1978 F 150000
;
run;
data empl2;
  input idnum $ name $ dob mmddyy10. salary;
  datalines;
1 John 01/02/1975 250000
2 Morgan 01/03/1976 300000
3 Jim 05/02/1977 350000
4 Anna 01/04/1978 150000
;
run;
```

```
data empl3;
  input id $ name $ dob mmddyy10. salary;
  datalines;
1 John 01/02/1975 250000
2 Morgan 01/03/1976 300000
3 Jim 05/02/1977 350000
4 Anna 01/04/1978 150000
;
run;

data all_of_them;
  set empl1 (drop=sex)
      empl2 (keep=idnum dob salary rename=(idnum=id))
      empl3 (keep=id dob salary);
  where dob le '01JAN1990'd and dob is not missing and salary ge 250000;
  format dob MMDDYY8.;
run;
```

```
proc sql;
  select idnum, name, dob format=MMDDYY8., salary from empl1
    where dob le '01JAN1990'd and dob is not missing and salary ge 250000
  union all
  select '' as idnum, '' as name, dob, salary, idnum as id from empl2
    where dob le '01JAN1990'd and dob is not missing and salary ge 250000
  union all
  select '' as idnum, '' as name, dob, salary, idnum as id from empl2
    where dob le '01JAN1990'd and dob is not missing and salary ge 250000
;
quit;

/*************************************************/
```

```
/* question of the exam: page 5 - problem 2 - 2.5 */
/*************************************************/

data score;
  input id string $ ;
  datalines;
1 852512
2 525411
3 856255
4 854741
;

data newX(drop=u);
  set score;
  array x{5} a1-a5;
  do u = 1 to 5;
    x{u} = input(substr(string, u, 1), 1.);
  end;
run;
```

```
283 /* solution */
284 data newX(drop=u);
285   set score;
286   a1  = input(substr(string, 1, 1), 1.);
287   a2  = input(substr(string, 2, 1), 1.);
288   a3  = input(substr(string, 3, 1), 1.);
289   a4  = input(substr(string, 4, 1), 1.);
290   a5  = input(substr(string, 5, 1), 1.);
291 run;
292
293 /*****************************************************/
```

```
295 /* question of the exam: page 5 - problem 3 - 3.1 */
296 /*****************************************************/
297
298 libname exam '/folders/myfolders/exam/' inencoding=asciiany;
299
300 proc sort data=exam.new01 nodup out=test40; by idnumber01; run;
301 proc sort data=exam.new02 nodup out=test41; by idnumber01; run;
302
303 data test42;
304   set test40; where idnumber01 is not missing;
305 run;
306
307 data test43;
308   set test41; where idnumber01 is not missing;
309 run;
```

```
311 /* put: Returns a value using a specified format. */
312 /* 'put' (here) converts number data to char data */
313 data test44;
314   set test43;
315   idnumber01char = put(idnumber01, 8.);
316   drop idnumber01;
317   rename idnumber01char = idnumber01;
318 run;
319 data exam1; merge test42 test44; by idnumber01; drop keytype; run;
320
321 /*****************************************************/
```

```
/* question of the exam: page 6 - problem 3 - 3.2 */
/*************************************************/
proc sort data=exam.key02 nodup out=test45; by idnumber01; run;
proc sort data=exam.key03 nodup out=test46; by idnumber01; run;

data exam2; merge test45 test46; by idnumber01; run;

proc sort data=exam.employee out=test48; by idnumber01; run;
proc sort data=exam.expenditures out=test49; by idnumber01; run;

data test50;
  merge exam2 test48 test49;
  by idnumber01;
run;

proc sort data=test50 out=exam2_1;
  by idnumber01;
run;

/*************************************************/
```

```
/* question of the exam: page 6 - problem 3 - 3.3 */
/*************************************************/
data test47; set exam.exam; drop deposit; run;
proc sort data=test47 nodup out=exam3; by keytype;
  where substr(position, 8, 1) ne '8';
run;

/* mikhastam 3 ta table exam1 va exam2 va exam3 ro ba ham join konam */
/* ama exam1 filed keytype ro nadasht banabarin anra dobare az aval neveshtam */
data exam1; merge test42 test44; by idnumber01; run;

/* idnumber01 dar exam2 va exam3 number ast */
data exam2;
  set exam2;
  idnumber01char = put(idnumber01, 8.);
  drop idnumber01;
  rename idnumber01char = idnumber01;
run;
```

```
362 data exam3;
363   set exam3;
364   idnumber01char = put(idnumber01, 8.);
365   drop idnumber01;
366   rename idnumber01char = idnumber01;
367 run;
368
369 proc sort data=exam1; by keytype; run;
370 proc sort data=exam2; by keytype; run;
371
372 data exam3_small exam3_big exam3_missing;
373   merge exam1(in=key1) exam2(in=key2) exam3(in=key3);
374   by keytype;
375   select;
376     when (keytype le 3) output exam3_small;
377     when ((keytype gt 3) && (keytype le 10)) output exam3_big;
378     otherwise output exam3_missing;
379   end;
380 run;
381
382 /*************************************************/
```

```
384 /* question of the exam: page 6 - problem 3 - 3.4 */
385 /*************************************************/
386
387 proc sql;
388   create table exam4 as
389   select sum(sumlillebuffet) as lille, weekday
390     from exam.x2012_1_6cleaned10
391    where sumlillebuffet > (1.9 * sumstorbuffet)
392    group by weekday;
393 quit;
394
395 /*************************************************/
```

```
/* question of the exam: page 6 - problem 3 - 3.5 */
/*************************************************/

%let afdeling=AgroTech A/S;

proc sql outobs=500;
  create table exam5 as
  select distinct * from exam.x201204
   where afdeling ne "&afdeling"
     and initialer not in (select initials from exam.sample1
                            union
                           select initials from exam.sample2
                            union
                           select initials from exam.sample3);
quit;


/*************************************************/
```

```
/* question of the exam: page 6 - problem 3 - 3.6 */
/*************************************************/

%macro chk_dir(dir=) ;
   %local rc fileref ;
   %let rc = %sysfunc(filename(fileref,&dir)) ;
   %if %sysfunc(fexist(&fileref))  %then
      %put NOTE: The directory "&dir" exists ;
   %else
     %do ;
         %sysexec md   &dir ;
         %put %sysfunc(sysmsg()) The directory has been created. ;
   %end ;
   %let rc=%sysfunc(filename(fileref)) ;
%mend chk_dir ;

%chk_dir(dir='/folders/myfolders/temp') ;    /*   <==  your directory specification goes he
%chk_dir(dir=c:\temp\sascode)
```