

/* Question 1

Write a SAS programme to address the following questions. You may not alter the csv files prior to reading them into SAS.

(a) Read the three datasets into the work library in SAS.*/

```
PROC IMPORT DATAFILE='/home/u45187342/SAS_EXAM/Medical1.csv'  
  DBMS=CSV replace  
  OUT=WORK.medical1;  
  GETNAMES=YES;  
RUN;
```

```
PROC IMPORT DATAFILE='/home/u45187342/SAS_EXAM/Medical2.csv'  
  DBMS=CSV replace  
  OUT=WORK.medical2;  
  GETNAMES=YES;  
RUN;
```

```
PROC IMPORT DATAFILE='/home/u45187342/SAS_EXAM/Medical3.csv'  
  DBMS=CSV replace  
  OUT=WORK.medical3;  
  GETNAMES=YES;  
RUN;
```

```
proc print data=work.medical1;  
run;  
proc print data=work.medical2;  
run;  
proc print data=work.medical3;  
run;
```

/* (b) Combine the datasets into one SAS data file called medical. The columns of medical should be Patient, Visit, Outcome, Age, Day. */

```
PROC TRANSPOSE DATA=work.medical3 OUT=work.medical4(drop=_NAME_);  
  BY visitnum;  
  copy visitnum;  
RUN;
```

```
data work.medical4;  
  set work.medical4;  
  Day = COL1;  
  Visit = _n_;  
run;
```

```
proc sql;  
create table work.medical as  
select medical1.patient,medical1.visit,outcome,medical2.age,medical4.Day  
  from medical1,medical2,medical4 where medical1.patient=medical2.patient and  
  medical4.visit=medical1.visit;  
Quit;
```

```
proc print data=work.medical;  
run;
```

/* (c) Screen the data. Change all error values to either missing, or to another value if appropriate. List any problems encountered and the errors found and state how they were resolved. E.g. Obs XX, out of range, replaced by missing value. */

```
proc freq data=work.medical;
tables patient visit outcome day age /nocum nopercent;
run;
```

```
data work.medical;
set work.medical;
if (outcome<0) or (outcome>27) then outcome=.;
else outcome=outcome;
if age>=50 and age<=59 then age=age;
else age=.;
run;
```

```
proc print data=work.medical;
run;
```

```
/* list of problems encountered and resolution :
Identified all the entries based on the given column constrains.
1. outcome column of the dataset have 5 vlaues which are out of range.
   Resolution :These values are numeric in nature so we replaced with ".".
2.age column have few values marked as ERROR the count is 36
   and few values are out of range values are 12 in count
   Resolution :these values were marked as missing while cleaning the dataset. */
```

```
/*(d) What is the average age of participants in the study? */
```

```
data work.medical;
set work.medical;
n_age=age*1; /*converting the age in character to numeric */
run;
proc means data=work.medical mean;
var n_age;
run;
```

```
/* The average age of participants in the study is 54.3965863 */
```

```
/*(e) What is the average (across all patients) outcome at visits 1, 6 and 12? */
```

```
proc means data=work.medical mean;
where visit in (1,6,12);
var outcome;
run;
```

```
/* The average outcome across all the patient is 16.8456304 */
```

```
/* Question 2:
```

```
2. The daily sales for a rm are recorded each month in a le named SalesMon-
thYear.csv (where month is the text of the month and year is the year in numbers).
The les contain the variables day (day of the month) and sales (daily sales in
thousands of euro). The sales les for January 2018 (SalesJanuary2018.csv) and
February 2019 (SalesFebruary2019.csv) are posted on Moodle.
(a) Write a SAS macro that can be applied to either dataset to do the following
activities: */
```

```
%macro sales(month=,year=,monthnum=);
proc import out= sales&month&year
datafile="/home/u45187342/SAS_EXAM/Sales&month&year..csv"
```

```

dbms=csv replace;
getnames=yes;
run;
/* □ Generate a line plot of sales versus day, including a title that is specific
to the month and year. */

title' line plot for' &month&year;
proc Sgplot data=sales&month&year;
series x=day y=sales;
run;
title;

/*□ Print out the highest daily sales from the month, including a title that is
specific to the month and year. */

proc sort data=Sales&month&year;
by Sales;
run;
title' highest daily sales for' &month&year;
proc means data=Sales&month&year max;
var Sales;
run;
title;

/* □ Print out the highest daily sales from the month for weekdays and week-
ends separately, including a title that is specific to the month and year.
Hint: recall that when the weekday function is applied to a date, it returns
values 1 to 7, where 1 = Sunday, 2 = Monday, etc. */

Data sales&month&year;
set sales&month&year;
newdate =mdy(&monthnum,&day,&year);
format newdate ddmmyy10.;
dow = weekday(newdate);
run;
title "Maximum Sales for Weekday" &Month &Year;
proc sort data = sales&Month&Year;
by Sales descending Sales;
run;
proc MEANS data = sales&Month&Year max;
where dow in (2,3,4,5,6);
var Sales;
run;
title;
title "Maximum Sales for Weekend" &Month &Year;
proc sort data = sales&Month&Year;
by Sales descending Sales;
run;
proc MEANS data = sales&Month&Year max;
where dow in (1,7);
var Sales;
run;
title;

%mend sales;

%sales(month=January,year=2018,monthnum=1);
%sales(month=February,year=2019,monthnum=2);

```

