Unit 1-2 Exercises

I. Diagnosing and Correcting Syntax Errors

- Submit a SAS program with errors.
- Diagnose and correct the errors.
- Save the corrected program.

Submitting a SAS Program with Errors

```
daat work.NewSalesEmps;
   length First_Name $ 12
          Last_Name $ 18 Job_Title $ 25;
   infile 'newemps.csv' dlm=',';
   input First_Name $ Last_Name $
         Job_Title $ Salary;
run;

proc print data=work.NewSalesEmps
run;

proc means data=work.NewSalesEmps average max;
   class Job_Title;
   var Salary;
run;
```

The SAS log contains error messages and warnings.

```
36   daat work.NewSalesEmps;
     ----
     14
WARNING 14-169: Assuming the symbol DATA was misspelled as daat.

37      length First_Name $ 12
38             Last_Name $ 18 Job_Title $ 25;
39      infile 'newemps.csv' dlm=',';
40      input First_Name $ Last_Name $
41            Job_Title $ Salary;
42   run;

NOTE: The infile 'newemps.csv' is:
      Filename=S:\Workshop\newemps.csv,
      RECFM=V,LRECL=256,File Size (bytes)=2604,
      Last Modified=02Apr2008:09:10:12,
      Create Time=02Apr2008:09:10:12
```

(Continued on the next page.)

```
NOTE: 71 records were read from the infile 'newemps.csv'.
      The minimum record length was 28.
      The maximum record length was 47.
NOTE: The data set WORK.NEWSALESEMPS has 71 observations and 4 variables.


43
44   proc print data=work.NewSalesEmps
45   run;
     ---
     22
     202
ERROR 22-322: Syntax error, expecting one of the following: ;, (, BLANKLINE, DATA, DOUBLE,
              HEADING, LABEL, N, NOOBS, OBS, ROUND, ROWS, SPLIT, STYLE, SUMLABEL, UNIFORM,
              WIDTH.
ERROR 202-322: The option or parameter is not recognized and will be ignored.
46

NOTE: The SAS System stopped processing this step because of errors.


47   proc means data=work.NewSalesEmps average max;
                                       -------
                                       22
                                       202
ERROR 22-322: Syntax error, expecting one of the following: ;, (, ALPHA, CHARTYPE, CLASSDATA,
              CLM, COMPLETETYPES, CSS, CV, DATA, DESCEND, DESCENDING, DESCENDTYPES, EXCLNPWGT,
              EXCLNPWGTS, EXCLUSIVE, FW, IDMIN, KURTOSIS, LCLM, MAX, MAXDEC, MEAN, MEDIAN, MIN,
              MISSING, MODE, N, NDEC, NMISS, NOLABELS, NONOBS, NOPRINT, NOTHREADS, NOTRAP,
              NWAY, ORDER, P1, P10, P25, P5, P50, P75, P90, P95, P99, PCTLDEF, PRINT, PRINTALL,
              PRINTALLTYPES, PRINTIDS, PRINTIDVARS, PROBT, Q1, Q3, QMARKERS, QMETHOD, QNTLDEF,
              QRANGE, RANGE, SKEWNESS, STDDEV, STDERR, SUM, SUMSIZE, SUMWGT, T, THREADS, UCLM,
              USS, VAR, VARDEF.
ERROR 202-322: The option or parameter is not recognized and will be ignored.
48      class Job_Title;
49      var Salary;
50   run;

NOTE: The SAS System stopped processing this step because of errors.
```

Diagnosing and Correcting the Errors

The log indicates that SAS

- assumed that the keyword DATA was misspelled and executed the DATA step
- interpreted the word RUN as an option in the PROC PRINT statement (because there was a missing semicolon), so PROC PRINT was not executed
- did not recognize the word AVERAGE as a valid option in the PROC MEANS statement, so the PROC MEANS step was not executed.

1. If you are using the Enhanced Editor, the program remains in the editor.

   However, if you use the Program Editor, the code disappears with each submission. Use the RECALL command or select **Run** ⇨ **Recall Last Submit** to recall the program that you submitted. The original program is copied into the Program Editor.

2. Edit the program.
   a. Correct the spelling of DATA.
   b. Put a semicolon at the end of the PROC PRINT statement.
   c. Change the word AVERAGE to MEAN in the PROC MEANS statement.

```
data work.NewSalesEmps;
   length First_Name $ 12
          Last_Name $ 18 Job_Title $ 25;
   infile 'newemps.csv' dlm=',';
   input First_Name $ Last_Name $
         Job_Title $ Salary;
run;

proc print data=work.NewSalesEmps;
run;

proc means data=work.NewSalesEmps mean max;
   class Job_Title;
   var Salary;
run;
```

3. Submit the program. It runs successfully without errors and generates output.

Saving the Corrected Program

You can use the FILE command to save your program to a file. The program must be in the Enhanced Editor or Program Editor before you issue the FILE command. If the code is not in the Program Editor, recall your program before saving the program.

Windows	`file 'myprog.sas'`

You can also select **File** ⇨ **Save As**.

A note appears that indicates that the statements are saved to the file.

II. Diagnosing and Correcting Syntax Errors

- Submit a SAS program that contains unbalanced quotation marks.
- Diagnose and correct the error.
- Resubmit the program.

Submitting a SAS Program that Contains Unbalanced Quotation Marks

The closing quotation mark for the DLM= option in the INFILE statement is missing.

```
data work.NewSalesEmps;
   length First_Name $ 12 Last_Name $ 18
          Job_Title $ 25;
   infile 'newemps.csv' dlm=',;
   input First_Name $ Last_Name $
         Job_Title $ Salary;
run;

proc print data=work.NewSalesEmps;
run;

proc means data=work.NewSalesEmps;
   class Job_Title;
   var Salary;
run;
```

SAS Log

```
51   data work.NewSalesEmps;
52      length First_Name $ 12 Last_Name $ 18
53             Job_Title $ 25;
54      infile 'newemps.csv' dlm=',;
55      input First_Name $ Last_Name $
56            Job_Title $ Salary;
57   run;
58
59   proc print data=work.NewSalesEmps;
60   run;
61
62   proc means data=work.NewSalesEmps;
63      class Job_Title;
64      var Salary;
65   run;
```

Diagnosing and Correcting the Errors

There are no notes in the SAS log because all of the SAS statements after the DLM= option became part of the quoted delimiter.

The banner in the window indicates that the DATA step is still running, and it is still running because the RUN statement was not recognized.

You can correct the unbalanced quotation marks programmatically by adding the following code before your previous statements:

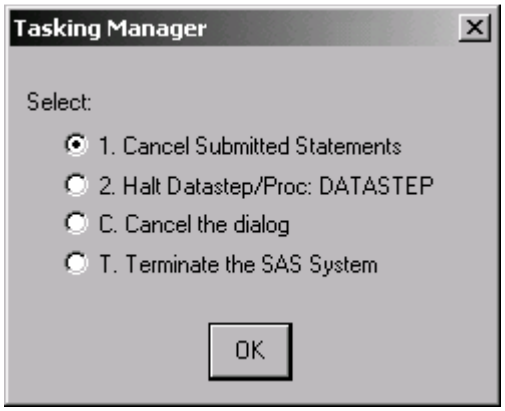
```
*';*";run;
```

If the quotation mark counter within SAS has an uneven number of quotation marks, as seen in the above program, SAS reads the quotation mark in the code above as the matching quotation mark in the quotation mark counter. SAS then has an even number of quotation marks in the quotation mark counter and runs successfully, assuming no other errors occur. Both single quotation marks and double quotation marks are used in case you submitted double quotation marks instead of single quotation marks.

Point-and-Click Approaches to Balancing Quotation Marks

Windows

1. To correct the problem in the Windows environment, click the break icon  or press the **CTRL** and **Break** keys.
2. Select **1. Cancel Submitted Statements** in the Tasking Manager window and select **OK**.



3. Select **Y to cancel submitted statements,** ⇨ **OK**.

Resubmitting the Program

1. In the appropriate Editor window, add a closing quotation mark to the DLM= option in the INFILE statement.

```
data work.NewSalesEmps;
   length First_Name $ 12 Last_Name $ 18
          Job_Title $ 25;
   infile 'newemps.csv' dlm=',';
   input First_Name $ Last_Name $
         Job_Title $ Salary;
run;

proc print data=work.NewSalesEmps;
run;

proc means data=work.NewSalesEmps;
   class Job_Title;
   var Salary;
run;
```

2. Resubmit the program.

When you make changes to the program in the Enhanced Editor and did not save the new version of the program, the window bar and the top border of the window reflect that you changed the program without saving it by putting an asterisk (*) beside the window name.
When you save the program, the * disappears.

1. **Diagnosing and Correcting a Misspelled Word**
   - **a.** With the appropriate Editor window active, include the SAS program **p103e01**.
   - **b.** Submit the program.
   - **c.** Use the notes in the SAS log to identify the error.
   - **d.** Correct the error and resubmit the program.

2. **Diagnosing and Correcting a Missing Statement**
   - **a.** With the appropriate Editor window active, include the SAS program **p103e02**.
   - **b.** Submit the program.
   - **c.** Are there any errors in the SAS log? _____
   - **d.** Notice the message in the title bar of the Editor window.
   - **e.** Why is PROC PRINT running? _____
   - **f.** Add the missing statement to execute the PROC PRINT step.
   - **g.** Submit the added statement.
   - **h.** Confirm that the output was created for the program by viewing the Log and Output windows.

3. **Using the Help Facility to Determine the Types of Errors in SAS**
   - **a.** In the Help facility, type **`syntax errors`** on the Index tab.
   - **b.** Double-click **syntax errors** in the results box.
   - **c.** In the Topics Found pop-up box, select **Error Processing and Debugging: Types of Errors in SAS**.
   - **d.** Name the five types of errors.

     _____