```
1   /*************************************************************************
2    Program Name: jblubau1_hw11_script
3    Date Created: 10/30/2016
4    Author: Joseph Blubaugh
5    Purpose: Homework Assignment 11
6   ************************************************************************* */
7
8   libname datadb 'C:\Users\Joseph\Projects\learning\Statistics\STAT_604\Materials'
8  ! access=readonly;
NOTE: Libref DATADB was successfully assigned as follows:
      Engine:        V9
      Physical Name: C:\Users\Joseph\Projects\learning\Statistics\STAT_604\Materials
9   libname output 'C:\Users\Joseph\Projects\learning\Statistics\STAT_604\Data';
NOTE: Libref OUTPUT was successfully assigned as follows:
      Engine:        V9
      Physical Name: C:\Users\Joseph\Projects\learning\Statistics\STAT_604\Data
10
11   filename outpdf
11 ! 'C:\Users\Joseph\Projects\learning\Statistics\STAT_604\Homework\jblubau1_hw11_output.pdf';
12
13   * 2) Read in an clean up zip codes data;
14   data output.zips;
15      * change the length of the county field;
16      length county $ 31;
17      set datadb.zip_codes (rename = (estimated_population = EstPopChar));
18      * Remove decommissioned zips;
19      where decommissioned = '0';
20      * Move redundent name info;
21      if scan(county, -1) = 'County' then do
22         county = tranwrd(county, 'County', '');
23      end;
24      if scan(county, -1) = 'Parish' then do
25         county = tranwrd(county, 'Parish', '');
26      end;
27      if scan(county, -1) = 'Borough' then do
28         county = tranwrd(county, 'Borough', '');
29      end;
30      * Convert population to numeric);
31      estimated_population = input(EstPopChar, 10.);
32      * Replace underscore with blank;
33      if find(timezone, '_') > 0 then do;
34         substr(timezone, find(timezone, '_'), 1) = ' ';
35      end;
36      * Keep needed variables only;
37      keep county estimated_population primary_city state timezone zip;
38      * Relabel variables;
39      label county = 'County'
40            estimated_population = 'Est. Population'
41            primary_city = 'City'
42            state = 'State'
43            timezone = 'Time Zone'
44            zip = 'Zip Code';
45   run;

WARNING: Multiple lengths were specified for the variable county by input data set(s). This can
         cause truncation of data.
NOTE: There were 41859 observations read from the data set DATADB.ZIP_CODES.
```

```
    WHERE decommissioned='0';
NOTE: The data set OUTPUT.ZIPS has 41859 observations and 6 variables.
NOTE: DATA statement used (Total process time):
    real time          0.06 seconds
    cpu time           0.06 seconds


46
47   * 3) Summarise the data by State and City;
48   * a) Sort the clean data set by State and City;
49   proc sort data=output.zips;
50       by state primary_city;
51   run;

NOTE: There were 41859 observations read from the data set OUTPUT.ZIPS.
NOTE: The data set OUTPUT.ZIPS has 41859 observations and 6 variables.
NOTE: PROCEDURE SORT used (Total process time):
    real time          0.01 seconds
    cpu time           0.01 seconds


52
53   * c) Aggregate population to city;
54   data zipsagg;
55       * Make sure zips length is long enough;
56       length zips $ 1700;
57       set output.zips;
58       * Reset count every time a new city is encountered;
59       by state primary_city;
60          if first.primary_city then do;
61              total = 0;
62              zips = '';
63              retain total zips;
64          end;
65       * Create a running sum of the population;
66       total = sum(total, estimated_population);
67       * Create a list of all of the zip codes in a city;
68       zips = catx(',', zips, zip);
69       if last.primary_city;
70       label zips = 'Zip Codes'
71             total = 'Est. City Population';
72       format total comma10.0;
73       keep primary_city state county zips total;
74       if total > 0;
75   run;

NOTE: There were 41859 observations read from the data set OUTPUT.ZIPS.
NOTE: The data set WORK.ZIPSAGG has 21404 observations and 5 variables.
NOTE: DATA statement used (Total process time):
    real time          0.03 seconds
    cpu time           0.03 seconds


76
77   * 4) Open pdf, turn off bookmarks;
78   ods pdf file=outpdf bookmarkgen=no;
NOTE: Writing ODS PDF output to DISK destination "OUTPDF", printer "PDF".
```

```
79
80   * 5) Print descriptor portion and subset of both data sets;
81   proc contents data=output.zips;
NOTE: Writing HTML Body file: sashtml.htm
82   run;

NOTE: PROCEDURE CONTENTS used (Total process time):
      real time           0.26 seconds
      cpu time            0.18 seconds


83
84   proc print data=output.zips label;
85        where primary_city in ('Albany', 'Center', 'Reno', 'Rome', 'Paris', 'San Juan', 'Juneau',
85 ! 'Washington');
86        var zip primary_city state timezone county estimated_population;
87   run;

NOTE: There were 478 observations read from the data set OUTPUT.ZIPS.
      WHERE primary_city in ('Albany', 'Center', 'Juneau', 'Paris', 'Reno', 'Rome', 'San Juan',
      'Washington');
NOTE: PROCEDURE PRINT used (Total process time):
      real time           0.31 seconds
      cpu time            0.23 seconds


88
89   proc contents data=zipsagg;
90   run;

NOTE: PROCEDURE CONTENTS used (Total process time):
      real time           0.07 seconds
      cpu time            0.03 seconds


91
92   proc print data=zipsagg label;
93        where primary_city in ('Albany', 'Center', 'Reno', 'Rome', 'Paris', 'San Juan', 'Juneau',
93 ! 'Washington');
94        var primary_city state county zips total;
95   run;

NOTE: There were 58 observations read from the data set WORK.ZIPSAGG.
      WHERE primary_city in ('Albany', 'Center', 'Juneau', 'Paris', 'Reno', 'Rome', 'San Juan',
      'Washington');
NOTE: PROCEDURE PRINT used (Total process time):
      real time           0.28 seconds
      cpu time            0.09 seconds


96
97   ods pdf close;
NOTE: ODS PDF printed 18 pages to
      C:\Users\Joseph\Projects\learning\Statistics\STAT_604\Homework\jblubau1_hw11_output.pdf.
```