SP104-105 Getting Started with SAS Programming

104 Reading and Filtering Data, Computing New Columns, Conditional Processing

105 Enhancing Reports, Creating Frequency Reports and Graphs, Creating Summary Reports and Data

```
************************************************************;
*  p104a03.sas Activity 4.03                               *;
*    1) Change the name of the output table to             *;
*       STORM_CAT5.                                        *;
*    2) Include only Category 5 storms (MaxWindMPH         *;
*       greater than or equal to 156) with StartDate on    *;
*       or after 01JAN2000.                                *;
*    3) Add a statement to include the following columns    *;
*       in the output data: Season, Basin, Name, Type,     *;
*       and MaxWindMPH. How many Category 5 storms         *;
*       occurred since January 1, 2000?                    *;
************************************************************;

* if necessary, change the path to your output folder *;
libname out "s:/workshop/output";

data out.storm_new;
   set pg1.storm_summary;
run;

libname out "/home/u58304328/EPG1V2/output";
data out.storm_cat5;
          set pg1.storm_summary;
```

```
        where MaxWindMPH >= 156 and StartDate >= "01JAN2000"d;
        keep Season Basin Name Type MaxWindMPH;
run;
```

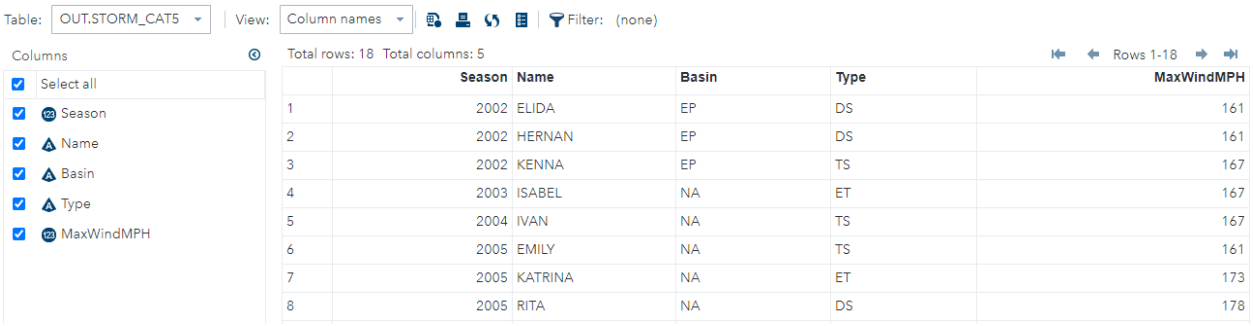


Table: OUT.STORM_CAT5 | View: Column names | Filter: (none)

Columns: Select all, Season, Name, Basin, Type, MaxWindMPH

Total rows: 18 Total columns: 5 — Rows 1-18

	Season	Name	Basin	Type	MaxWindMPH
1	2002	ELIDA	EP	DS	161
2	2002	HERNAN	EP	DS	161
3	2002	KENNA	EP	TS	167
4	2003	ISABEL	NA	ET	167
5	2004	IVAN	NA	TS	167
6	2005	EMILY	NA	TS	161
7	2005	KATRINA	NA	ET	173
8	2005	RITA	NA	DS	178

```
*************************************************************;
*  p104a04.sas Activity 4.04                                *;
*    1) Add an assignment statement to create StormLength *;
*       that represents the number of days between        *;
*       StartDate and EndDate.                            *;
*    2) Run the program. In 1980, how long did the storm  *;
*       named Agatha last?                                *;
*************************************************************;

data storm_length;
        set pg1.storm_summary;
        drop Hem_EW Hem_NS Lat Lon;
        *Add assignment statement;
        StormLength = EndDate - StartDate;
run;
```

Table: WORK.STORM_LENGTH | View: Column names | Filter: (none)

Columns

- Select all
- Season
- Name
- Basin
- Type
- MaxWindMPH
- MinPressure
- StartDate
- EndDate
- StormLength

Total rows: 3118 Total columns: 9

	Season	Name	Basin	Type	MaxWindMPH	MinPressure	StartDate	EndDate
1	1980		na	TS	35	.	17JUL1980	18NOV1980
2	1980		SP	NR	.	998	27MAR1980	30MAR1980
3	1980	AGATHA	EP	TS	115	.	09JUN1980	15JUN1980
4	1980	ALBINE	SI	ET	.	.	27NOV1979	06DEC1979
5	1980	ALEX	WP	TS	40	998	09OCT1980	14OCT1980
6	1980	ALLEN	NA	TS	190	899	31JUL1980	11AUG1980
7	1980	AMY	SI	NR	132	915	04JAN1980	12JAN1980
8	1980	BERENICE	SI	TS	.	.	15DEC1979	21DEC1979
9	1980	BETTY	WP	ET	115	925	28OCT1980	08NOV1980
10	1980	BLAS	EP	TS	58	.	16JUN1980	19JUN1980

```
************************************************************;
*  p104a05.sas Activity 4.05                               *;
*    1) Open the PG1.STORM_RANGE table and examine the     *;
*       columns. Notice that each storm has four wind      *;
*       speed measurements.                                *;
*    2) Create a new column named WindAvg that is the      *;
*       mean of Wind1, Wind2, Wind3, and Wind4.            *;
*    3) Create a new column WindRange that is the range    *;
*       of Wind1, Wind2, Wind3, and Wind4.                 *;
************************************************************;


data storm_wingavg;
        set pg1.storm_range;
        *Add assignment statements;
        WindAvg = mean(Wind1, Wind2, Wind3, Wind4);
        WindRange = range(Wind1, Wind2, Wind3, Wind4);
run;
```

Table: WORK.STORM_WINGAVG | View: Column names | Filter: (none)

Columns

- Select all
- Season
- Basin
- Name
- Wind1
- Wind2
- Wind3
- Wind4
- WindAvg
- WindRange

Total rows: 2959 Total columns: 9

	Season	Basin	Name	Wind1	Wind2	Wind3	Wind4	WindAvg	WindRange
1	1980	EP	AGATHA	100	95	90	85	92.5	15
2	1980	EP	BLAS	50	50	50	45	48.75	5
3	1980	EP	CELIA	65	65	65	65	65	0
4	1980	EP	DARBY	45	45	35	30	38.75	15
5	1980	EP	ESTELLE	40	35	35	25	33.75	15
6	1980	EP	FRANK	45	40	35	35	38.75	10
7	1980	EP	GEORGETTE	65	55	50	45	53.75	20
8	1980	EP	HOWARD	90	85	80	80	83.75	10
9	1980	EP	ISIS	85	80	80	75	80	10
10	1980	EP	JAVIER	100	100	100	95	98.75	5

```
************************************************************;
*  p104a06.sas Activity 4.06                                *;
*    1) Add a WHERE statement that uses the SUBSTR          *;
*       function to include rows where the second letter    *;
*       of Basin is P (Pacific ocean storms).               *;
*    2) Run the program and view the log and data. How      *;
*       many storms were in the Pacific basin?              *;
************************************************************;
*  Syntax                                                   *;
*    SUBSTR (char, position, <length>)                      *;
************************************************************;


data pacific;
        set pg1.storm_summary;
        drop Type Hem_EW Hem_NS MinPressure Lat Lon;
        *Add a WHERE statement that uses the SUBSTR function;
        where substr(Basin, 2, 1) = "P";
run;
```

Table: WORK.PACIFIC | View: Column names | Filter: (none)

Columns

- Select all
- Season
- Name
- Basin
- MaxWindMPH
- StartDate
- EndDate

Total rows: 1958 Total columns: 6

	Season	Name	Basin	MaxWindMPH	StartDate	EndDate
1	1980		SP	.	27MAR1980	30MAR1980
2	1980	AGATHA	EP	115	09JUN1980	15JUN1980
3	1980	ALEX	WP	40	09OCT1980	14OCT1980
4	1980	BETTY	WP	115	28OCT1980	08NOV1980
5	1980	BLAS	EP	58	16JUN1980	19JUN1980
6	1980	CARMEN	WP	69	05APR1980	07APR1980
7	1980	CARY	WP	52	28OCT1980	02NOV1980
8	1980	CELIA	EP	75	25JUN1980	29JUN1980

```
************************************************************;
*  p104a07.sas Activity 4.07                                *;
*    1) Add the ELSE keyword to test conditions             *;
*       sequentially until a true condition is met.         *;
*    2) Change the final IF-THEN statement to an ELSE       *;
*       statement.                                          *;
*    3) How many storms are in PressureGroup 1?             *;
************************************************************;


data storm_cat;
        set pg1.storm_summary;
        keep Name Basin MinPressure StartDate PressureGroup;
        *add ELSE keyword and remove final condition;
        if MinPressure=. then PressureGroup=.;
        else if MinPressure<=920 then PressureGroup=1;
        else PressureGroup=0;
run;


proc freq data=storm_cat;
        tables PressureGroup;
run;
```

The FREQ Procedure

PressureGroup	Frequency	Percent	Cumulative Frequency	Cumulative Percent
0	2733	93.53	2733	93.53
1	189	6.47	2922	100.00
Frequency Missing = 196				

```
************************************************************;
*  p104a08.sas Activity 4.08                                *;
*    1) Run the program and examine the results. Why is     *;
*       Ocean truncated? What value is assigned when        *;
*       Basin='na'?                                         *;
*    2) Modify the program to add a LENGTH statement to     *;
*       declare the name, type, and length of Ocean         *;
*       before the column is created.                       *;
*    3) Add an assignment statement after the KEEP          *;
*       statement to convert Basin to uppercase. Run the    *;
*       program.                                            *;
*    4) Move the LENGTH statement to the end of the DATA    *;
*       step. Run the program. Does it matter where the     *;
*       LENGTH statement is in the DATA step?               *;
************************************************************;
*  Syntax                                                   *;
*       LENGTH char-column $ length;                        *;
************************************************************;


data storm_summary2;
        set pg1.storm_summary;
        *Add a LENGTH statement;
        length Ocean $ 8;
```

```
        keep Basin Season Name MaxWindMPH Ocean;
        *Add assignment statement;
        Basin=upcase(Basin);
        OceanCode=substr(Basin,2,1);
        if OceanCode="I" then Ocean="Indian";
        else if OceanCode="A" then Ocean="Atlantic";
        else Ocean="Pacific";
run;
```

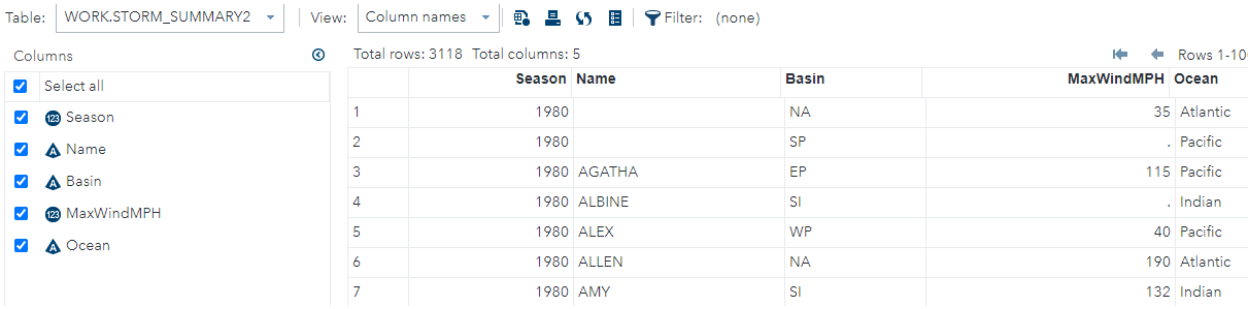


Table: WORK.STORM_SUMMARY2 | View: Column names | Filter: (none)

Columns: Select all, Season, Name, Basin, MaxWindMPH, Ocean

Total rows: 3118 Total columns: 5 | Rows 1-10

	Season	Name	Basin	MaxWindMPH	Ocean
1	1980		NA	35	Atlantic
2	1980		SP	.	Pacific
3	1980	AGATHA	EP	115	Pacific
4	1980	ALBINE	SI	.	Indian
5	1980	ALEX	WP	40	Pacific
6	1980	ALLEN	NA	190	Atlantic
7	1980	AMY	SI	132	Indian

```
**********************************************************;
*  p104a09.sas Activity 4.09                             *;
*    Run the program. Why does the program fail?         *;
**********************************************************;


data front rear;
   set sashelp.cars;
   if DriveTrain="Front" then do;
      DriveTrain="FWD";
      output front;
   end;
   else if DriveTrain='Rear' then do;
      DriveTrain="RWD";
      output rear;
   end;
```

run;

Table: WORK.FRONT | View: Column names | Filter: (none)

Columns: Select all, Make, Model, Type, Origin, DriveTrain, MSRP, Invoice, EngineSize, Cylinders, Horsepower, MPG_City, MPG_Highway, Weight

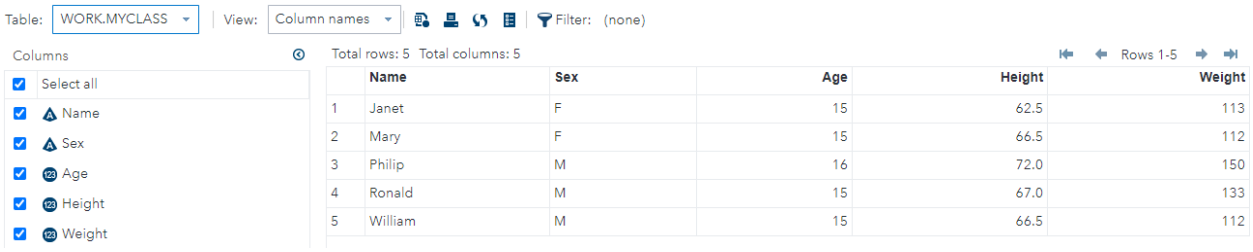
Total rows: 226 Total columns: 15 Rows 1-100

	Make	Model	Type	Origin	DriveTrain	MSRP	Invoice	EngineSize
1	Acura	RSX Type S 2dr	Sedan	Asia	FWD	$23,820	$21,761	2
2	Acura	TSX 4dr	Sedan	Asia	FWD	$26,990	$24,647	2.4
3	Acura	TL 4dr	Sedan	Asia	FWD	$33,195	$30,299	3.2
4	Acura	3.5 RL 4dr	Sedan	Asia	FWD	$43,755	$39,014	3.5
5	Acura	3.5 RL w/Navigation 4dr	Sedan	Asia	FWD	$46,100	$41,100	3.5
6	Audi	A4 1.8T 4dr	Sedan	Europe	FWD	$25,940	$23,508	1.8
7	Audi	A41.8T convertible 2dr	Sedan	Europe	FWD	$35,940	$32,506	1.8
8	Audi	A4 3.0 4dr	Sedan	Europe	FWD	$31,840	$28,846	3
9	Audi	A6 3.0 4dr	Sedan	Europe	FWD	$36,640	$33,129	3
10	Audi	A4 3.0 convertible 2dr	Sedan	Europe	FWD	$42,490	$38,325	3
11	Audi	RS 6 4dr	Sports	Europe	FWD	$84,600	$76,417	4.2
12	Audi	TT 1.8 convertible 2dr (coupe)	Sports	Europe	FWD	$35,940	$32,512	1.8
13	Buick	Rendezvous CX	SUV	USA	FWD	$26,545	$24,085	3.4
14	Buick	Century Custom 4dr	Sedan	USA	FWD	$22,180	$20,351	3.1

```
************************************************;
*  p104d01.sas Read a SAS table and create a subset as new   *;
*  SAS table                                    *;
************************************************;
*  Syntax and Example                           *;
*                                               *;
*    DATA output-table;                         *;
*       SET input-table;                        *;
*                   WHERE expression;           *;
*    RUN;                                       *;
************************************************;


data myclass;
   set sashelp.class;
   where age >= 15;
run;


************************************************;
*  Subset columns using the DROP or KEEP        *;
```

```
*  statements                          *;
************************************************;
*  Syntax and Example                  *;
*                                      *;
*   DATA output-table;                 *;
*     SET input-table;                 *;
*                 DROP variable-list;  *;
*     /*OR*/                          *;
*                 KEEP variable-list;  *;
*   RUN;                               *;
************************************************;

data myclass;
   set sashelp.class;
   where age >= 15;
   *keep name age height;
   drop sex weight;
run;

************************************************;
*  Apply permanent formats using the   *;
*  FORMAT statement                    *;
************************************************;
*  Syntax and Example                  *;
*                                      *;
*   DATA output-table;                 *;
*     SET input-table;                 *;
*     FORMAT variable(s) format;       *;
*   RUN;                               *;
```

```
*************************************************;

data myclass;
   set sashelp.class;
   where age >= 15;
   format height 4.1 weight 3.;
run;
```



	Name	Sex	Age	Height	Weight
1	Janet	F	15	62.5	113
2	Mary	F	15	66.5	112
3	Philip	M	16	72.0	150
4	Ronald	M	15	67.0	133
5	William	M	15	66.5	112

```
************************************************************;
*  p104d02.sas Using Expressions to Create New Columns          *;
************************************************************;
*  Syntax and Example                          *;
*                                              *;
*    DATA output-table;                        *;
*       SET input-table;                       *;
*       new-column = expression;               *;
*    RUN;                                      *;
************************************************************;


data cars_new;
            set sashelp.cars;
            where Origin ne "USA";
            Profit=MSRP-invoice;
            Source="Non-US Cars";
            format Profit dollar10.;
```

```
        keep Make Model MSRP Invoice Profit Source;
run;

*************************************************************;
*  Demo                                  *;
*    1) Add an assignment statement to create a numeric   *;
*       column named MaxWindKM by multiplying MaxWindMPH  *;
*       by 1.60934.                        *;
*    2) Add a FORMAT statement to round MaxWindKM to the  *;
*       nearest whole number.                  *;
*    3) Add an assignment statement to create a new       *;
*       character column named StormType that is equal to *;
*       Tropical Storm. Highlight the DATA step and run   *;
*       the selected code.                     *;
*************************************************************;

data tropical_storm;
        set pg1.storm_summary;
        drop Hem_EW Hem_NS Lat Lon;
        where Type="TS";
        *Add assignment and FORMAT statements;
        MaxWindKM= MaxWindMPH * 1.60934;
        Format MaxWindKM 3.;
        StormType= "Tropical Storm";
run;
```

Table: WORK.CARS_NEW | View: Column names | Filter: (none)

Columns

- Select all
- Make
- Model
- MSRP
- Invoice
- Profit
- Source

Total rows: 281 Total columns: 6 Rows 1-100

	Make	Model	MSRP	Invoice	Profit	Source
1	Acura	MDX	$36,945	$33,337	$3,608	Non-US Cars
2	Acura	RSX Type S 2dr	$23,820	$21,761	$2,059	Non-US Cars
3	Acura	TSX 4dr	$26,990	$24,647	$2,343	Non-US Cars
4	Acura	TL 4dr	$33,195	$30,299	$2,896	Non-US Cars
5	Acura	3.5 RL 4dr	$43,755	$39,014	$4,741	Non-US Cars
6	Acura	3.5 RL w/Navigation 4dr	$46,100	$41,100	$5,000	Non-US Cars
7	Acura	NSX coupe 2dr manual S	$89,765	$79,978	$9,787	Non-US Cars

```
************************************************************;
* p104d03.sas Using Character Functions                    *;
************************************************************;
*  Syntax and Example                                      *;
*   DATA output-table;                                     *;
*      SET input-table;                                    *;
*      new-column=function(arguments);                     *;
*   RUN;                                                   *;
*                                                          *;
* Numeric Functions:                                       *;
*  SUM(num1, num2, ...)                                    *;
*  MEAN(num1, num2, ...)                                   *;
*  MEDIAN(num1, num2, ...)                                 *;
*  RANGE(num1, num2, ...)                                  *;
*                                                          *;
* Character Functions:                                     *;
*  UPCASE(char)                                            *;
*  PROPCASE(char, <delimiters>)                            *;
*  CATS(char1, char2, ...)                                 *;
*  SUBSTR(char, position, <length>)                        *;
************************************************************;

data cars_new;
```

```
        set sashelp.cars;
        MPG_Mean=mean(MPG_City, MPG_Highway);
        Type=upcase(Type);
        format MPG_Mean 4.1;
        keep Make Model MSRP Invoice MPG_Mean Type;
run;

*************************************************************;
*  Demo                                  *;
*    1) Add an assignment statement to convert Basin to   *;
*       all uppercase letters using the UPCASE function.  *;
*    2) Add an assignment statement to convert Name to    *;
*       proper case using the PROPCASE function.         *;
*    3) Add an assignment statement to create Hemisphere, *;
*       which concatenates Hem_NS and Hem_EW using the    *;
*       CATS function.                                    *;
*    4) Add an assignment statement to create Ocean,      *;
*       which extracts the second letter of Basin using   *;
*       the SUBSTR function. Highlight the DATA step and  *;
*       run the selected code.                            *;
*************************************************************;

data storm_new;
        set pg1.storm_summary;
        drop Type Hem_EW Hem_NS MinPressure Lat Lon;
        *Add assignment statements;
        Basin=upcase(basin);
        Name=propcase(Name);
        Hemisphere=cats(Hem_NS, Hem_EW);
```

```
        Ocean=substr(Basin, 2, 1);
run;
```

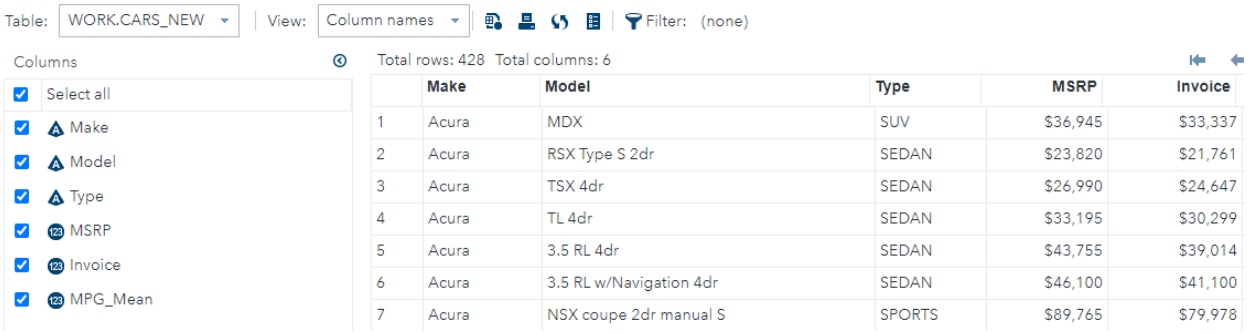


Table: WORK.CARS_NEW | View: Column names | Filter: (none)

Columns

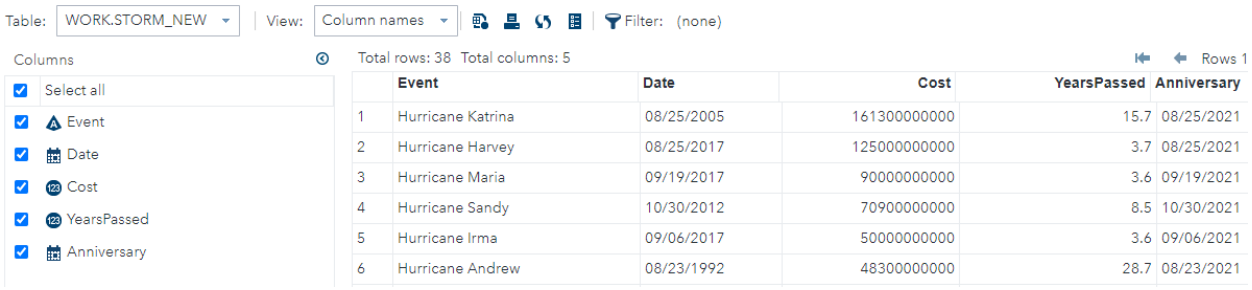
- Select all
- Make
- Model
- Type
- MSRP
- Invoice
- MPG_Mean

Total rows: 428 Total columns: 6

	Make	Model	Type	MSRP	Invoice
1	Acura	MDX	SUV	$36,945	$33,337
2	Acura	RSX Type S 2dr	SEDAN	$23,820	$21,761
3	Acura	TSX 4dr	SEDAN	$26,990	$24,647
4	Acura	TL 4dr	SEDAN	$33,195	$30,299
5	Acura	3.5 RL 4dr	SEDAN	$43,755	$39,014
6	Acura	3.5 RL w/Navigation 4dr	SEDAN	$46,100	$41,100
7	Acura	NSX coupe 2dr manual S	SPORTS	$89,765	$79,978

```
************************************************************;
*  p104d04.sas Using Date Functions                        *;
************************************************************;
*  Syntax                                                  *;
*                                                          *;
*  Date function examples:                                 *;
*    YEAR (SAS-date)                                       *;
*    MONTH (SAS-date)                                      *;
*    DAY (SAS-date)                                        *;
*    WEEKDAY (SAS-date)                                    *;
*    TODAY ()                                              *;
*    MDY (month, day, year)                                *;
*    YRDIF (startdate, enddate, 'AGE')                     *;
************************************************************;


************************************************************;
*  Demo                                                    *;
*    1) Create the column YearsPassed and use the YRDIF    *;
*       function. The difference in years should be based  *;
*       on each Date value and today's date.               *;
*    2) Create Anniversary as the day and month of each    *;
```

```
*      storm in the current year.                    *;
*    3) Format YearsPassed to round the value to one      *;
*      decimal place, and Date and Anniversary as        *;
*      MM/DD/YYYY. Highlight the DATA step and run the   *;
*      selected code.                                *;
************************************************************;


data storm_new;
            set pg1.storm_damage;
            drop Summary;
            *Add assignment and FORMAT statements;
            YearsPassed=yrdif(Date, today(), "age");
            Anniversary=mdy(month(Date), day(Date), year(today()));
            format YearsPassed 4.1 Date Anniversary mmddyy10.;
run;
```



	Event	Date	Cost	YearsPassed	Anniversary
1	Hurricane Katrina	08/25/2005	161300000000	15.7	08/25/2021
2	Hurricane Harvey	08/25/2017	125000000000	3.7	08/25/2021
3	Hurricane Maria	09/19/2017	90000000000	3.6	09/19/2021
4	Hurricane Sandy	10/30/2012	70900000000	8.5	10/30/2021
5	Hurricane Irma	09/06/2017	50000000000	3.6	09/06/2021
6	Hurricane Andrew	08/23/1992	48300000000	28.7	08/23/2021

```
************************************************************;
*  p104d05.sas Conditional Processing with IF-THEN              *;
************************************************************;
*  Syntax and Example                          *;
*                                              *;
*    IF expression THEN statement;                  *;
************************************************************;


data cars2;
```

```
   set sashelp.cars;
   if MSRP<30000 then Cost_Group=1;
   if MSRP>=30000 then Cost_Group=2;
         keep Make Model Type MSRP Cost_Group;
run;

*************************************************************;
*  Demo                                  *;
*    1) Create a column named PressureGroup that is based *;
*      on the following assignments:                *;
*      MinPressure<=920 => 1                    *;
*      MinPressure>920 => 0                     *;
*    2) Highlight the DATA step, run the selected code,   *;
*      and examine the data. What value is assigned      *;
*      to PressureGroup when MinPressure is missing?     *;
*    3) Add a new IF-THEN statement before the existing   *;
*      IF-THEN statements to assign PressureGroup=. if   *;
*      MinPressure is missing.                  *;
*    4) Highlight the DATA step and run the selected      *;
*      code. What value is assigned to PressureGroup?    *;
*************************************************************;

data storm_new;
         set pg1.storm_summary;
         keep Season Name Basin MinPressure PressureGroup;
         *Add IF-THEN statements;
         if MinPressure <= 920 then PressureGroup=1;
         if MinPressure > 920 then PressureGroup=0;
         if MinPressure=. then PressureGroup=.;
```

run;

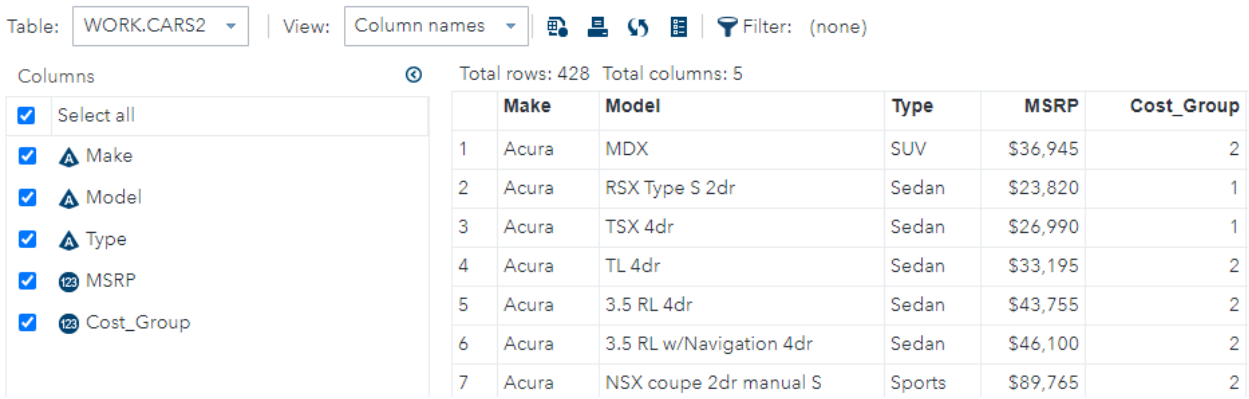


Table: WORK.CARS2 | View: Column names | Filter: (none)

Columns

- Select all
- Make
- Model
- Type
- MSRP
- Cost_Group

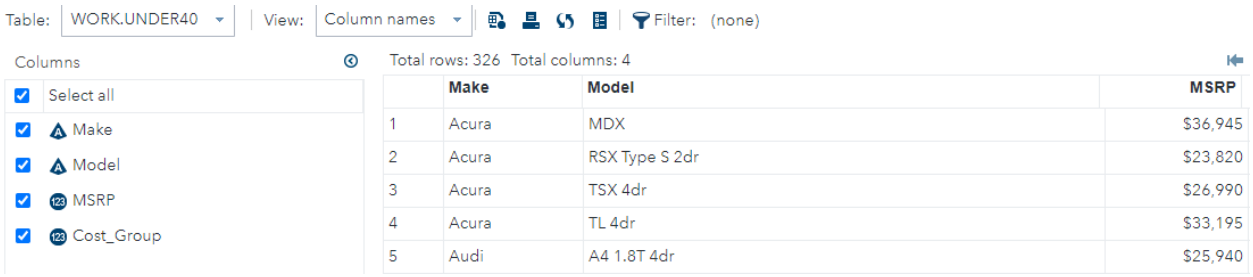
Total rows: 428 Total columns: 5

	Make	Model	Type	MSRP	Cost_Group
1	Acura	MDX	SUV	$36,945	2
2	Acura	RSX Type S 2dr	Sedan	$23,820	1
3	Acura	TSX 4dr	Sedan	$26,990	1
4	Acura	TL 4dr	Sedan	$33,195	2
5	Acura	3.5 RL 4dr	Sedan	$43,755	2
6	Acura	3.5 RL w/Navigation 4dr	Sedan	$46,100	2
7	Acura	NSX coupe 2dr manual S	Sports	$89,765	2

```
************************************************************;
* p104d07.sas Processing Multiple Statements with IF-THEN/DO      *;
************************************************************;
* Syntax and Example                          *;
*                                          *;
*   IF expression THEN DO;                    *;
*      <executable statements>                *;
*   END;                                  *;
*   ELSE IF expression THEN DO;                *;
*      <executable statements>                *;
*   END;                                  *;
*   ELSE DO;                                *;
*      <executable statements>                *;
*   END;                                  *;
************************************************************;


data under40 over40;
   set sashelp.cars;
         keep Make Model MSRP Cost_Group;
   if MSRP<20000 then do;
     Cost_Group=1;
```

```
    output under40;
  end;
  else if MSRP<40000 then do;
    Cost_Group=2;
    output under40;
  end;
  else do;
    Cost_Group=3;
    output over40;
  end;
run;

************************************************************;
*  Demo                                    *;
*     Modify the IF-THEN statements to use IF-THEN/DO   *;
*     syntax to write rows to either the indian,       *;
*     atlantic, or pacific table based on the value of  *;
*     Ocean. Highlight the DATA step and run the       *;
*     selected code.                          *;
************************************************************;

data indian atlantic pacific;
        set pg1.storm_summary;
        length Ocean $ 8;
        keep Basin Season Name MaxWindMPH Ocean;
        Basin=upcase(Basin);
        OceanCode=substr(Basin,2,1);
        *Modify the program to use IF-THEN-DO syntax;
        if OceanCode="I" then Do;
```

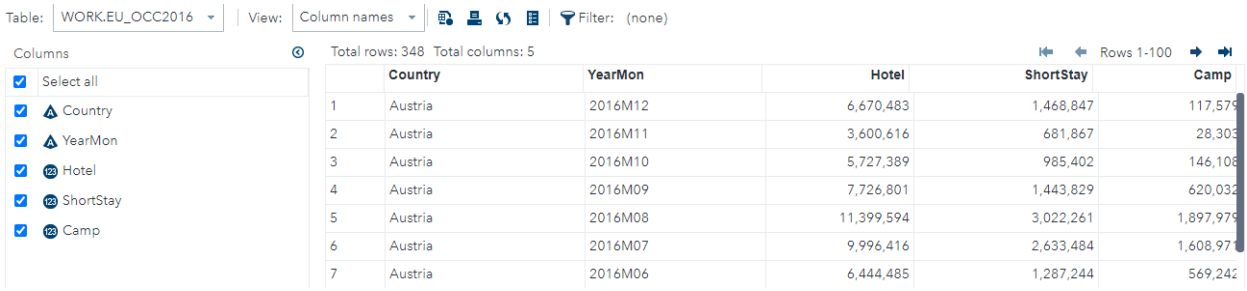
```
            Ocean="Indian";
            output indian;
        End;
        else if OceanCode="A" then Do;
            Ocean="Atlantic";
            output atlantic;
        End;
        else Do;
            Ocean="Pacific";
            output pacific;
        end;
run;
```



	Make	Model	MSRP
1	Acura	MDX	$36,945
2	Acura	RSX Type S 2dr	$23,820
3	Acura	TSX 4dr	$26,990
4	Acura	TL 4dr	$33,195
5	Audi	A4 1.8T 4dr	$25,940

```
**********************************************************;
*  p104p01.sas LESSON 4, PRACTICE 1                      *;
*    a) Open the PG1.EU_OCC table and examine the column  *;
*       names and values.                                 *;
*    b) Modify the code to create a temporary table named *;
*       EU_OCC2016 and read PG1.EU_OCC.                   *;
*    c) Complete the WHERE statement to select only the   *;
*       stays that were reported in 2016. Notice that     *;
*       YearMon is a character column and the first four  *;
*       positions represent the year.                     *;
*    d) Complete the FORMAT statement in the DATA step to *;
*       apply the COMMA17. format to the Hotel,           *;
```

```
*      ShortStay, and Camp columns.                    *;
*    e) Complete the DROP statement to exclude Geo from   *;
*      the output table.                               *;
************************************************************;

data  EU_OCC2016;
        set pg1.eu_occ;
        where YearMon like "2016%";
        format Hotel ShortStay Camp comma17.;
        drop Geo;
run;
```



	Country	YearMon	Hotel	ShortStay	Camp
1	Austria	2016M12	6,670,483	1,468,847	117,579
2	Austria	2016M11	3,600,616	681,867	28,303
3	Austria	2016M10	5,727,389	985,402	146,108
4	Austria	2016M09	7,726,801	1,443,829	620,032
5	Austria	2016M08	11,399,594	3,022,261	1,897,979
6	Austria	2016M07	9,996,416	2,633,484	1,608,971
7	Austria	2016M06	6,444,485	1,287,244	569,242

*p104p02.sas Open a new program window and write a DATA step to read the pg1.np_species table and;

*create a new permanent table named out.fox in the EPG1V2/output (or EPG194/output) folder.;

*Include only the rows where Category is Mammal and Common_Names includes Fox.;

*Exclude the Category, Record_Status, Occurrence, and Nativeness columns from the output data.;

*Submit the program.;

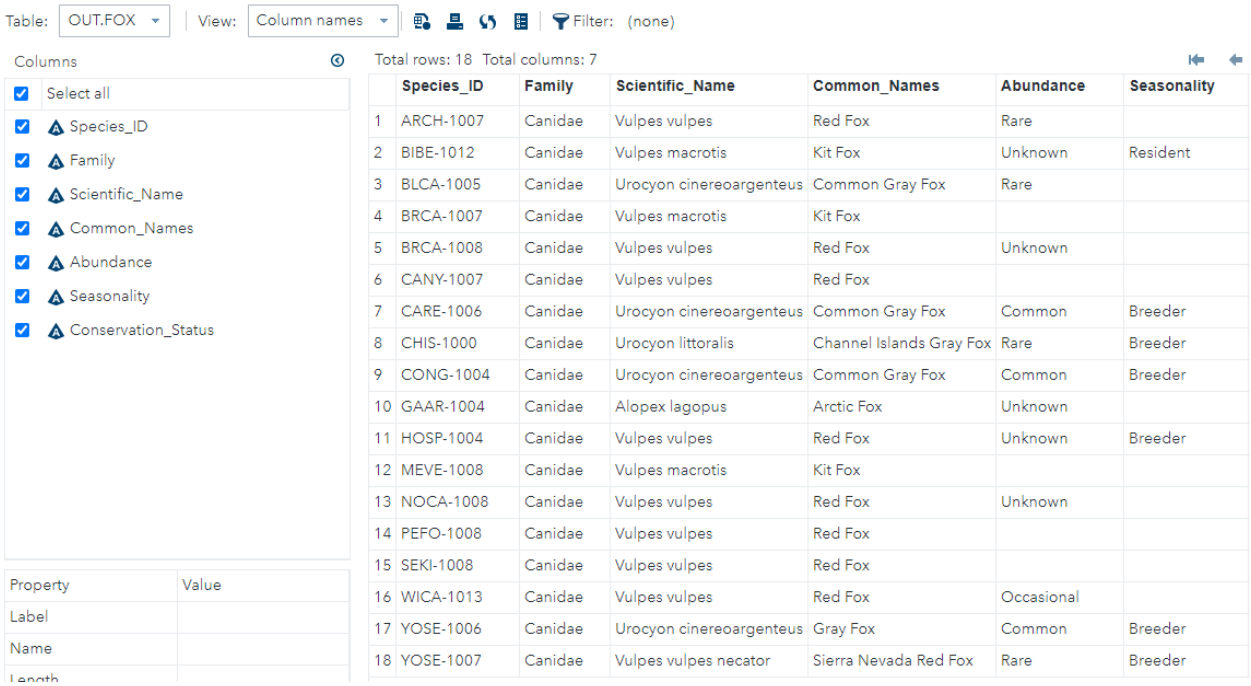
*Notice that Fox Squirrels are included in the output table. Add a condition in the WHERE statement to exclude rows that include Squirrel. ;

*Submit the program and verify the results.;

*Sort the fox table by Common_Names.;

```
libname out "/home/u58304328/EPG1V2/output";
data out.fox;
```

```
        set pg1.np_species;
        where Category="Mammal" and Common_Names like "%Fox%" and Common_Names NOT like
"%Squirrel%";
        drop Category Record_Status Occurrence Nativeness;
run;


proc sort data=out.fox out=np_sort;
        by Common_Names;
run;
```



	Species_ID	Family	Scientific_Name	Common_Names	Abundance	Seasonality
1	ARCH-1007	Canidae	Vulpes vulpes	Red Fox	Rare	
2	BIBE-1012	Canidae	Vulpes macrotis	Kit Fox	Unknown	Resident
3	BLCA-1005	Canidae	Urocyon cinereoargenteus	Common Gray Fox	Rare	
4	BRCA-1007	Canidae	Vulpes macrotis	Kit Fox		
5	BRCA-1008	Canidae	Vulpes vulpes	Red Fox	Unknown	
6	CANY-1007	Canidae	Vulpes vulpes	Red Fox		
7	CARE-1006	Canidae	Urocyon cinereoargenteus	Common Gray Fox	Common	Breeder
8	CHIS-1000	Canidae	Urocyon littoralis	Channel Islands Gray Fox	Rare	Breeder
9	CONG-1004	Canidae	Urocyon cinereoargenteus	Common Gray Fox	Common	Breeder
10	GAAR-1004	Canidae	Alopex lagopus	Arctic Fox	Unknown	
11	HOSP-1004	Canidae	Vulpes vulpes	Red Fox	Unknown	Breeder
12	MEVE-1008	Canidae	Vulpes macrotis	Kit Fox		
13	NOCA-1008	Canidae	Vulpes vulpes	Red Fox	Unknown	
14	PEFO-1008	Canidae	Vulpes vulpes	Red Fox		
15	SEKI-1008	Canidae	Vulpes vulpes	Red Fox		
16	WICA-1013	Canidae	Vulpes vulpes	Red Fox	Occasional	
17	YOSE-1006	Canidae	Urocyon cinereoargenteus	Gray Fox	Common	Breeder
18	YOSE-1007	Canidae	Vulpes vulpes necator	Sierra Nevada Red Fox	Rare	Breeder

```
**********************************************************;
*  p104p04.sas LESSON 4, PRACTICE 4                      *;
*    a) Create a new column named SqMiles by multiplying  *;
*       Acres by .0015625.                                *;
*    b) Create a new column named Camping as the sum of   *;
*       OtherCamping, TentCampers, RVCampers, and         *;
*       BackcountryCampers.                               *;
*    c) Format SqMiles and Camping to include commas and  *;
```

```
*      zero decimal places.                    *;
*    d) Modify the KEEP statement to include the new     *;
*      columns. Run the program.                *;
*************************************************************;

data np_summary_update;
        set pg1.np_summary;
        keep Reg ParkName DayVisits OtherLodging Acres SqMiles Camping;
        *Add assignment statements;
        SqMiles = Acres * 0.0015625;
        Camping = Sum(OtherCamping, TentCampers, RVCampers, BackcountryCampers);
        format sqMiles Camping comma10.;
run;

*The pg1.eu_occ table contains individual columns for nights spent at hotels, short-stay
accommodations, or camps for each year and month.;
* The YearMon column is character. If necessary, start SAS Studio before you begin. ;

*Reminder: If you restarted your SAS session,you must run the libname.sas program to access your
course data. ;

*Write a DATA step to create a temporary table named eu_occ_total that is based on the pg1.eu_occ
table. Create the following new columns:;
*Column          Tips for calculating values;
*Year    The four-digit year extracted from YearMon.;
*MonthThe two-digit month extracted from YearMon.;
*ReportDate     The first day of the reporting month. Note: Use the MDY function and the new Year and
Month columns ;
*Total   The total nights spent at any establishment.;
```

*Format Hotel, ShortStay, Camp, and Total with commas. Format ReportDate to display the values in the form JAN2018.;

*Keep Country, Hotel, ShortStay, Camp, ReportDate, and Total in the new table.;

*Submit the program and view the output data.;

*What is the value of ReportDate in row one?;

```
data eu_occ_total;
        set pg1.eu_occ;
        Year=substr(YearMon,1,4);
        Month=substr(YearMon,6,2);
        ReportDate=mdy(Month, 1, Year);
        Total=sum(Hotel, ShortStay, Camp);
        format Hotel ShortStay Camp Total comma10. ReportDate MONYY7.;
        keep Country Hotel ShortStay Camp ReportDate Total;
run;
```

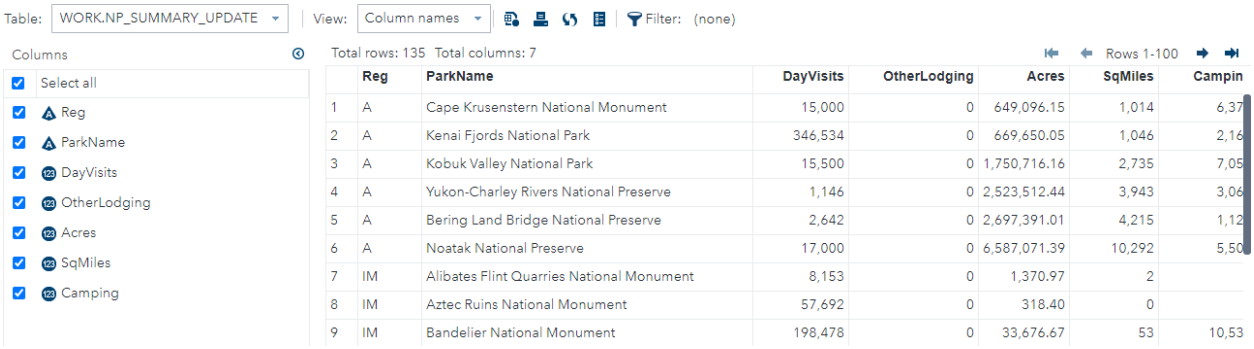


Table: WORK.NP_SUMMARY_UPDATE | View: Column names | Filter: (none)

Columns: Select all, Reg, ParkName, DayVisits, OtherLodging, Acres, SqMiles, Camping

Total rows: 135 Total columns: 7 — Rows 1-100

	Reg	ParkName	DayVisits	OtherLodging	Acres	SqMiles	Campin
1	A	Cape Krusenstern National Monument	15,000	0	649,096.15	1,014	6,37
2	A	Kenai Fjords National Park	346,534	0	669,650.05	1,046	2,16
3	A	Kobuk Valley National Park	15,500	0	1,750,716.16	2,735	7,05
4	A	Yukon-Charley Rivers National Preserve	1,146	0	2,523,512.44	3,943	3,06
5	A	Bering Land Bridge National Preserve	2,642	0	2,697,391.01	4,215	1,12
6	A	Noatak National Preserve	17,000	0	6,587,071.39	10,292	5,50
7	IM	Alibates Flint Quarries National Monument	8,153	0	1,370.97	2	
8	IM	Aztec Ruins National Monument	57,692	0	318.40	0	
9	IM	Bandelier National Monument	198,478	0	33,676.67	53	10,53

```
**********************************************************;
*  p104p07.sas LESSON 4, PRACTICE 7                      *;
*    a) Submit the program and view the generated output. *;
*    b) In the DATA step, use IF-THEN/ELSE statements to  *;
*       create a new column, ParkType, based on the value *;
*       of Type.                                          *;
*       NM -> Monument                                    *;
*       NP -> Park                                        *;
```

```
*      NPRE, PRE, or PRESERVE -> Preserve              *;
*      NS -> Seashore                          *;
*      RVR or RIVERWAYS -> River                   *;
*    c) Modify the PROC FREQ step to generate a frequency *;
*      report for ParkType.                        *;
************************************************************;


data park_type;
        set pg1.np_summary;
        *Add IF-THEN-ELSE statements;
        If Type = "NM" Then ParkType = "Monument";
        else if Type = "NP" Then ParkType = "Park";
        else if Type = "NS" Then ParkType = "Seashore";
        else if Type = "RVR" or Type = "RIVERWAYS" Then ParkType = "River";
        else ParkType = "Preserve";
run;


proc freq data=park_type;
        tables ParkType;
run;



/*Level 2 Practice: Processing Statements Conditionally with DO Groups
TOTAL POINTS 1
Question 1
Use conditional processing to split pg1.np_summary into two tables: parks and monuments. If
necessary, start SAS Studio before you begin.
Reminder: If you restarted your SAS session, you must run the libname.sas program to access your
course files.
```

1. Write a DATA step to create two temporary tables, named parks and monuments, that are based on the pg1.np_summary table.

Read only national parks or monuments from the input table. (Type is either NP or NM.)

2. Create a new column named Campers that is the sum of all columns that contain counts of campers. Format the column to include commas.

3. When Type is NP, create a new column named ParkType that is equal to Park, and write the row to the parks table.

When Type is NM, assign ParkType as Monument and write the row to the monuments table.

4. Keep Reg, ParkName, DayVisits, OtherLodging, Campers, and ParkType in both output tables.

5. Submit the program and view the output data.

Which table has the most rows?

*/

```
data parks monuments;
	set pg1.np_summary;
	where Type in ("NP" "NM");
	keep Reg ParkName DayVisits OtherLodging Campers ParkType;
	Campers=sum(OtherCamping, TentCampers, RVCampers, BackCountryCampers);
	format Campers comma17.;
	length ParkType $ 8;
	*Add IF-THEN-DO-ELSE statements;
	If Type = "NP" Then Do;
		ParkType = "Park";
		output parks;
	end;
	else do;
		ParkType = "Monument";
		output monuments;
	end;
run;
```

The FREQ Procedure

ParkType	Frequency	Percent	Cumulative Frequency	Cumulative Percent
Monument	63	46.67	63	46.67
Park	51	37.78	114	84.44
Preserve	8	5.93	122	90.37
River	3	2.22	125	92.59
Seashore	10	7.41	135	100.00

```
************************************************************;
*  p105a01.sas Activity 5.01                                *;
*    1) In the program, notice that there is a TITLE        *;
*       statement followed by two procedures. Run the       *;
*       program. Where does the title appear in the         *;
*       output?                                             *;
*    2) Add a TITLE2 statement above PROC MEANS to print    *;
*       a second line:                                      *;
*       Summary Statistics for MaxWind and MinPressure      *;
*    3) Add another TITLE2 statement above PROC FREQ with   *;
*       this title: Frequency Report for Basin              *;
*    4) Run the program. Which titles appear above each     *;
*       report?                                             *;
************************************************************;

title "Storm Analysis";

title2 "Summary Statistics for MaxWind and MinPressure";
proc means data=pg1.storm_final;
            var MaxWindMPH MinPressure;
run;
```

```
title2 "Frequency Report for Basin";
proc freq data=pg1.storm_final;
        tables BasinName;
run;
```

Storm Analysis
Summary Statistics for MaxWind and MinPressure

The MEANS Procedure

Variable	N	Mean	Std Dev	Minimum	Maximum
MaxWindMPH	3071	79.9101270	31.6029479	6.0000000	213.0000000
MinPressure	2913	969.5719190	27.3397126	872.0000000	1012.00

Storm Analysis
Frequency Report for Basin

The FREQ Procedure

BasinName	Frequency	Percent	Cumulative Frequency	Cumulative Percent
East Pacific	675	21.83	675	21.83
North Atlantic	478	15.46	1153	37.29
North Indian	60	1.94	1213	39.23
South Indian	594	19.21	1807	58.44
South Pacific	359	11.61	2166	70.05
West Pacific	926	29.95	3092	100.00

```
************************************************************;
*  p105a03.sas Activity 5.03                                *;
*    1) Modify the LABEL statement in the DATA step to      *;
*       label the Invoice column as Invoice Price.          *;
*    2) Run the program. Why do the labels appear in the    *;
*       PROC MEANS report but not in the PROC PRINT         *;
*       report? Fix the program and run it again.           *;
************************************************************;

data cars_update;
   set sashelp.cars;
        keep Make Model MSRP Invoice AvgMPG;
```

```
        AvgMPG=mean(MPG_Highway, MPG_City);
        label MSRP="Manufacturer Suggested Retail Price"
      AvgMPG="Average Miles per Gallon"
         Invoice="Invoice Price";
run;

proc means data=cars_update min mean max;
  var MSRP Invoice;
run;

proc print data=cars_update label;
  var Make Model MSRP Invoice AvgMPG;
run;
```

The MEANS Procedure

Variable	Label	Minimum	Mean	Maximum
MSRP	Manufacturer Suggested Retail Price	10280.00	32774.86	192465.00
Invoice	Invoice Price	9875.00	30014.70	173560.00

Obs	Make	Model	Manufacturer Suggested Retail Price	Invoice Price	Average Miles per Gallon
1	Acura	MDX	$36,945	$33,337	20.0
2	Acura	RSX Type S 2dr	$23,820	$21,761	27.5
3	Acura	TSX 4dr	$26,990	$24,647	25.5
4	Acura	TL 4dr	$33,195	$30,299	24.0
5	Acura	3.5 RL 4dr	$43,755	$39,014	21.0
6	Acura	3.5 RL w/Navigation 4dr	$46,100	$41,100	21.0
7	Acura	NSX coupe 2dr manual S	$89,765	$79,978	20.5
8	Audi	A4 1.8T 4dr	$25,940	$23,508	26.5
9	Audi	A41.8T convertible 2dr	$35,940	$32,506	26.5
10	Audi	A4 3.0 4dr	$31,840	$28,846	24.0
11	Audi	A4 3.0 Quattro 4dr manual	$33,430	$30,366	21.5
12	Audi	A4 3.0 Quattro 4dr auto	$34,480	$31,388	21.5
13	Audi	A6 3.0 4dr	$36,640	$33,129	23.5
14	Audi	A6 3.0 Quattro 4dr	$39,640	$35,992	21.5
15	Audi	A4 3.0 convertible 2dr	$42,490	$38,325	23.5
16	Audi	A4 3.0 Quattro convertible 2dr	$44,240	$40,075	21.5
17	Audi	A6 2.7 Turbo Quattro 4dr	$42,840	$38,840	21.5
18	Audi	A6 4.2 Quattro 4dr	$49,690	$44,936	20.5
19	Audi	A8 L Quattro 4dr	$69,190	$64,740	20.5
20	Audi	S4 Quattro 4dr	$48,040	$43,556	17.0
21	Audi	RS 6 4dr	$84,600	$76,417	18.5
22	Audi	TT 1.8 convertible 2dr (coupe)	$35,940	$32,512	24.0
23	Audi	TT 1.8 Quattro 2dr (convertible)	$37,390	$33,891	24.0
24	Audi	TT 3.2 coupe 2dr (convertible)	$40,590	$36,739	25.0
25	Audi	A6 3.0 Avant Quattro	$40,840	$37,060	21.5
26	Audi	S4 Avant Quattro	$49,090	$44,446	18.0
27	BMW	X3 3.0i	$37,000	$33,873	19.5
28	BMW	X5 4.4i	$52,195	$47,720	19.0
29	BMW	325i 4dr	$28,495	$26,155	24.5

```
**********************************************************;
*  p105a04.sas Activity 5.04                              *;
*    1) Create a temporary output table named storm_count *;
*       by completing the OUT= option in the TABLES       *;
*       statement.                                        *;
*    2) Add the NOPRINT option on the PROC FREQ statement *;
*       to suppress the printed report.                   *;
*    3) Run the program. Which statistics are included in *;
```

```
*      the output table? Which month has the highest     *;
*      number of storms?                                 *;
************************************************************;

title "Frequency Report for Basin and Storm Month";

proc freq data=pg1.storm_final order=freq noprint;
            tables StartDate / out=storm_count ;
            format StartDate monname.;
run;
```

Table: WORK.STORM_COUNT | View: Column names | Filter: (none)

Columns: Select all, StartDate, COUNT, PERCENT

Total rows: 12 Total columns: 3 — Rows 1-12

	StartDate	COUNT	PERCENT
1	September	486	15.717981889
2	August	485	15.685640362
3	July	346	11.190168176
4	October	326	10.543337646
5	January	246	7.9560155239
6	February	224	7.2445019405
7	November	189	6.1125485123
8	June	186	6.0155239327
9	March	182	5.8861578266
10	December	180	5.8214747736
11	April	125	4.042690815
12	May	117	3.7839586028

```
************************************************************;
*  p105a05.sas Activity 5.05                                *;
*    1) Add options to include N (count), MEAN, and MIN     *;
*       statistics. Round each statistic to the nearest     *;
*       integer.                                            *;
*    2) Add a CLASS statement to group the data by Season   *;
*       and Ocean. Run the program.                         *;
*    3) Modify the program to add the WAYS statement so     *;
*       that separate reports are created for Season and    *;
*       Ocean statistics. Run the program.                  *;
*       Which ocean had the lowest mean for minimum         *;
```

```
*     pressure?                          *;
*     Which season had the lowest mean for minimum     *;
*     pressure?                          *;
************************************************************;

proc means data=pg1.storm_final;
          var MinPressure;
          where Season >=2010;
run;

proc means data=pg1.storm_final n mean min maxdec=0;
          var MinPressure;
          where Season >=2010;
          class Season Ocean;
run;

proc means data=pg1.storm_final n mean min maxdec=0;
          var MinPressure;
          where Season >=2010;
          class Season Ocean;
          ways 1;
run;
```

The MEANS Procedure

Analysis Variable : MinPressure				
N	Mean	Std Dev	Minimum	Maximum
654	972.4266055	27.9928884	872.0000000	1012.00

The MEANS Procedure

Analysis Variable : MinPressure					
Season	Ocean	N Obs	N	Mean	Minimum
2010	Atlantic	21	21	978	924
	Indian	19	19	968	910
	Pacific	38	38	973	885
2011	Atlantic	19	19	984	940
	Indian	18	18	983	949
	Pacific	43	43	967	920
2012	Atlantic	19	19	982	940
	Indian	19	19	975	925
	Pacific	45	45	969	900
2013	Atlantic	15	15	998	979
	Indian	22	22	970	930
	Pacific	58	58	975	895
2014	Atlantic	9	9	980	940
	Indian	20	20	963	915
	Pacific	56	56	968	900
2015	Atlantic	12	12	989	931
	Indian	19	19	966	910
	Pacific	62	62	960	872
2016	Atlantic	16	16	984	934
	Indian	15	15	975	910
	Pacific	58	55	968	884
2017	Atlantic	17	17	971	908
	Indian	12	12	982	922
	Pacific	25	25	982	935

The MEANS Procedure

Analysis Variable : MinPressure				
Ocean	N Obs	N	Mean	Minimum
Atlantic	128	128	983	908
Indian	144	144	972	910
Pacific	385	382	969	872

Analysis Variable : MinPressure				
Season	N Obs	N	Mean	Minimum
2010	78	78	973	885
2011	80	80	975	920
2012	83	83	973	900
2013	95	95	977	895
2014	85	85	968	900
2015	93	93	965	872
2016	89	86	972	884
2017	54	54	979	908

```
************************************************************;
*  p105a06.sas Activity 5.06                                *;
*    1) Run the PROC MEANS step and compare the report      *;
*       and the wind_stats table. Are the same statistics   *;
*       in the report and table? What do the first five     *;
*       rows in the table represent?                        *;
*    2) Uncomment the WAYS statement. Delete the            *;
```

```
*      statistics listed in the PROC MEANS statement and *;
*      add the NOPRINT option. Run the program. Notice   *;
*      that a report is not generated and the first five *;
*      rows from the previous table are excluded.        *;
*    3) Add the following options in the OUTPUT statement *;
*      and run the program again. How many rows are in   *;
*      the output table?                                 *;
*       output out=wind_stats mean=AvgWind max=MaxWind; *;
************************************************************;

proc means data=pg1.storm_final noprint;
          var MaxWindMPH;
          class BasinName;
          ways 1;
          output out=wind_stats mean=AvgWind max=MaxWind;
run;

/*
Level 1 Practice: Producing a Descriptive Statistics Report
TOTAL POINTS 1
1.
Question 1
The pg1.np_westweather table contains weather-related information for four national parks:
Death Valley National Park, Grand Canyon National Park, Yellowstone National Park, and Zion National
Park.
Use the MEANS procedure to analyze the data in this table.

Reminder: If you restarted your SAS session, you must run the libname.sas program to access your
course files.
```

Create a new program. Write a PROC MEANS step to analyze rows from pg1.np_westweather with the following specifications:

Generate the mean, minimum, and maximum statistics for the Precip, Snow, TempMin, and TempMax columns.

Use the MAXDEC= option to display the values with a maximum of two decimal positions.

Use the CLASS statement to group the data by Year and Name.

Use Weather Statistics by Year and Park as the report title.

Submit the program and review the results.

What is the mean TempMin in DEATH VALLEY, CA US in 2016?

*/

```
title "Weather Statistics by Year and Park";
proc means data=pg1.np_westweather mean min max maxdec=2;
        var Precip Snow TempMin TempMax;
        class Year Name;
run;
title;
```

/*Level 2 Practice: Creating an Output Table with Custom Columns

TOTAL POINTS 2

1.

Question 1

The pg1.np_westweather table contains weather-related information for four national parks: Death Valley National Park, Grand Canyon National Park, Yellowstone National Park, and Zion National Park. Use the MEANS procedure to analyze the data in this table.

Reminder: If you restarted your SAS session, you must run libname.sas.

Open a new program window and write a PROC MEANS step to analyze rows from pg1.np_westweather.

Exclude rows where values for Precip are equal to 0.

Analyze precipitation amounts grouped by Name and Year.

Create only an output table, named rainstats, with columns for the N and SUM statistics.

Name the columns RainDays and TotalRain, respectively.

Keep only those rows that are the combination of Year and Name.

Submit the program and view the output data.

How many rows are in work.rainstats?

*/

```
title "Rain Statistics by Park and Year";
proc means data=pg1.np_westweather mean min max maxdec=2;
        where Precip ne 0;
        var Precip Snow TempMin TempMax;
        class Name Year;
        output out=rainstats n=RainDays sum=TotalRain;
        ways 2;
run;
title;
```

/*Write a PROC PRINT step to print the rainstats table.

Suppress the printing of observation numbers, and display column labels.

Display the columns in the following order: Name, Year, RainDays, and TotalRain.

Label Name as Park Name, RainDays as Number of Days Raining, and TotalRain as Total Rain Amount (inches).

Use Rain Statistics by Year and Park as the report title.

Submit the program and review the results.

```
What is the Total Rain Amount (inches) value in row one?
*/
title "Rain Statistics by Year and Park";
proc print data=rainstats noobs label;
        var Name Year RainDays TotalRain;
        label Name="Park Name"
                RainDays="Number of Days Raining"
                TotalRain="Total Rain Amount (inches)";
run;
title;
```

Weather Statistics by Year and Park

The MEANS Procedure

Year	NAME	N Obs	Variable	Mean	Minimum	Maximum
2015	DEATH VALLEY, CA US	365	PRECIP	0.01	0.00	0.55
			SNOW	0.00	0.00	0.00
			TEMPMIN	64.44	26.00	97.00
			TEMPMAX	93.29	53.00	125.00
	GRAND CANYON VISITOR CENTER, AZ US	365	PRECIP	0.07	0.00	2.20
			SNOW	0.13	0.00	5.70
			TEMPMIN	40.99	8.00	69.00
			TEMPMAX	61.60	17.00	93.00
	YELLOWSTONE NATIONAL PARK EAST ENTRANCE, WY US	363	PRECIP	0.06	0.00	0.85
			SNOW	0.33	0.00	10.00
			TEMPMIN	23.05	-22.00	48.00
			TEMPMAX	53.11	9.00	89.00
	ZION NATIONAL PARK, UT US	362	PRECIP	0.05	0.00	1.23
			SNOW	0.01	0.00	4.00
			TEMPMIN	49.63	7.00	78.00
			TEMPMAX	75.60	35.00	110.00
2016	DEATH VALLEY, CA US	366	PRECIP	0.00	0.00	0.46
			SNOW	0.00	0.00	0.00
			TEMPMIN	65.61	29.00	101.00
			TEMPMAX	92.77	52.00	127.00
	GRAND CANYON VISITOR CENTER, AZ US	365	PRECIP	0.06	0.00	2.20
			SNOW	0.13	0.00	7.00
			TEMPMIN	41.51	4.00	69.00
			TEMPMAX	61.71	21.00	94.00
	YELLOWSTONE NATIONAL PARK EAST ENTRANCE, WY US	357	PRECIP	0.07	0.00	1.06
			SNOW	0.48	0.00	13.00
			TEMPMIN	22.79	-32.00	46.00
			TEMPMAX	52.07	4.00	86.00
	ZION NATIONAL PARK, UT US	366	PRECIP	0.06	0.00	2.29
			SNOW	0.01	0.00	3.00
			TEMPMIN	49.28	15.00	77.00
			TEMPMAX	75.28	37.00	110.00
2017	DEATH VALLEY, CA US	365	PRECIP	0.00	0.00	0.65
			SNOW	0.00	0.00	0.00
			TEMPMIN	66.66	33.00	103.00
			TEMPMAX	93.47	56.00	127.00
	GRAND CANYON VISITOR CENTER, AZ US	365	PRECIP	0.03	0.00	0.80
			SNOW	0.15	0.00	11.00
			TEMPMIN	42.93	7.00	74.00
			TEMPMAX	63.00	22.00	94.00
	YELLOWSTONE NATIONAL PARK EAST ENTRANCE, WY US	351	PRECIP	0.08	0.00	1.33
			SNOW	0.69	0.00	9.00
			TEMPMIN	22.38	-31.00	45.00
			TEMPMAX	51.07	4.00	87.00
	ZION NATIONAL PARK, UT US	365	PRECIP	0.04	0.00	1.16
			SNOW	0.01	0.00	2.00
			TEMPMIN	49.16	12.00	78.00
			TEMPMAX	77.01	37.00	113.00

Rain Statistics by Park and Year

The MEANS Procedure

NAME	Year	N Obs	Variable	Mean	Minimum	Maximum
DEATH VALLEY, CA US	2015	15	PRECIP	0.16	0.01	0.55
			SNOW	0.00	0.00	0.00
			TEMPMIN	62.60	49.00	86.00
			TEMPMAX	90.20	70.00	125.00
	2016	16	PRECIP	0.09	0.01	0.46
			SNOW	0.00	0.00	0.00
			TEMPMIN	56.75	39.00	81.00
			TEMPMAX	79.13	56.00	120.00
	2017	11	PRECIP	0.13	0.01	0.65
			SNOW	0.00	0.00	0.00
			TEMPMIN	66.45	43.00	89.00
			TEMPMAX	87.45	58.00	120.00
GRAND CANYON VISITOR CENTER, AZ US	2015	97	PRECIP	0.27	0.01	2.20
			SNOW	0.50	0.00	5.70
			TEMPMIN	39.39	10.00	62.00
			TEMPMAX	57.95	17.00	91.00
	2016	84	PRECIP	0.26	0.01	2.20
			SNOW	0.57	0.00	7.00
			TEMPMIN	41.32	9.00	64.00
			TEMPMAX	59.80	21.00	92.00
	2017	65	PRECIP	0.17	0.01	0.80
			SNOW	0.82	0.00	11.00
			TEMPMIN	43.03	15.00	66.00
			TEMPMAX	60.89	28.00	92.00
YELLOWSTONE NATIONAL PARK EAST ENTRANCE, WY US	2015	155	PRECIP	0.15	0.01	0.85
			SNOW	0.79	0.00	10.00
			TEMPMIN	25.72	-13.00	46.00
			TEMPMAX	50.13	17.00	88.00
	2016	154	PRECIP	0.16	0.01	1.06
			SNOW	1.12	0.00	13.00
			TEMPMIN	21.99	-32.00	46.00
			TEMPMAX	45.23	4.00	84.00
	2017	172	PRECIP	0.18	0.01	1.33
			SNOW	1.50	0.00	9.00
			TEMPMIN	20.80	-30.00	45.00
			TEMPMAX	43.44	4.00	86.00
ZION NATIONAL PARK, UT US	2015	77	PRECIP	0.22	0.01	1.23
			SNOW	0.06	0.00	4.00
			TEMPMIN	46.38	13.00	69.00
			TEMPMAX	66.81	35.00	100.00
	2016	68	PRECIP	0.32	0.01	2.29
			SNOW	0.06	0.00	3.00
			TEMPMIN	45.47	24.00	70.00
			TEMPMAX	66.97	38.00	100.00

Rain Statistics by Year and Park

Park Name	Year	Number of Days Raining	Total Rain Amount (inches)
DEATH VALLEY, CA US	2015	15	2.45
DEATH VALLEY, CA US	2016	16	1.42
DEATH VALLEY, CA US	2017	11	1.46
GRAND CANYON VISITOR CENTER, AZ US	2015	97	25.9
GRAND CANYON VISITOR CENTER, AZ US	2016	82	21.1
GRAND CANYON VISITOR CENTER, AZ US	2017	65	11
YELLOWSTONE NATIONAL PARK EAST ENTRANCE, WY US	2015	150	22.2
YELLOWSTONE NATIONAL PARK EAST ENTRANCE, WY US	2016	149	23.4
YELLOWSTONE NATIONAL PARK EAST ENTRANCE, WY US	2017	143	25.7
ZION NATIONAL PARK, UT US	2015	77	16.9
ZION NATIONAL PARK, UT US	2016	68	21.7
ZION NATIONAL PARK, UT US	2017	56	14.5

```
************************************************************;
*  p105d01.sas Enhancing Reports                           *;
************************************************************;
*  Syntax and Example                                      *;
*                                                          *;
*   TITLEn "title-text";                                   *;
*   FOOTNOTEn "footnote-text";                             *;
*                                                          *;
*   LABEL col-name="label-text"                            *;
*         col-name="label-text";                           *;
*                                                          *;
*   Grouped Reports (sort first):                          *;
*   PROC procedure-name;                                   *;
*     BY col-name;                                         *;
*   RUN;                                                   *;
************************************************************;


*Titles and Footnotes;
title1 "Class Report";
```

```
title2 "All Students";
footnote1 "Report Generated on 01SEP2018";

proc print data=pg1.class_birthdate;
run;

*Using macro variables;
%let age=13;

title1 "Class Report";
title2 "Age=&age";
footnote1 "Report Generated on %sysfunc(today(),date9.)";

proc print data=pg1.class_birthdate;
        where age=&age;
run;

*Labels;
proc means data=sashelp.cars;
        where type="Sedan";
        var MSRP MPG_Highway;
        label MSRP="Manufacturer Suggested Retail Price"
      MPG_Highway="Highway Miles per Gallon";
run;

*Grouped Report;
proc sort data=sashelp.cars out=cars_sort;
        by Origin;
run;
```

```
proc freq data=cars_sort;
        by Origin;
        tables Type;
run;

*************************************************************;
*  Demo                                  *;
*    1) Add a PROC SORT step before PROC PRINT to sort    *;
*       PG1.STORM_FINAL by BasinName and descending       *;
*       MaxWindMPH. Create a temporary table named        *;
*       STORM_SORT. Filter the rows to include only       *;
*       MaxWindMPH>156.                                   *;
*    2) Modify the PROC PRINT step to read the STORM_SORT *;
*       table and group the report by BasinName.          *;
*    3) Add the following title: Category 5 Storms. Clear *;
*       the title for future results.                     *;
*    4) Add labels for the following columns and ensure   *;
*       that PROC PRINT displays the labels:              *;
*       MaxWindMPH => Max Wind (MPH)                      *;
*       MinPressure => Min Pressure                       *;
*       StartDate => Start Date                           *;
*       StormLength => Length of Storm (days)             *;
*    5) Add the NOOBS option in the PROC PRINT statement  *;
*       to suppress the OBS column. Highlight the demo    *;
*       program and run the selected code.                *;
*************************************************************;

proc sort data=pg1.storm_final out=storm_sort;
```

```
        by BasinName descending MaxWindMPH;
        where MaxWindMPH > 156;
run;

title "Category 5 Storms";
proc print data=storm_sort label noobs;
        var Season Name MaxWindMPH MinPressure StartDate StormLength;
        by BasinName;
        label MaxWindMPH="Max Wind (MPH)"
                MinPressure="Min Pressure"
                StartDate="Start Date"
                StormLength="Length of Storm (Days)";
run;

title;
```

Class Report
All Students

Obs	Name	Sex	Age	Height	Weight	Birthdate
1	Alfred	M	14	69.0	112.5	16370
2	Alice	F	13	56.5	84.0	16756
3	Barbara	F	13	65.3	98.0	16451
4	Carol	F	14	62.8	102.5	16256
5	Henry	M	14	63.5	102.5	16406
6	James	M	12	57.3	83.0	16967
7	Jane	F	12	59.8	84.5	16873
8	Janet	F	15	62.5	112.5	15797
9	Jeffrey	M	13	62.5	84.0	16552
10	John	M	12	59.0	99.5	17036
11	Joyce	F	11	51.3	50.5	17169
12	Judy	F	14	64.3	90.0	16410
13	Louise	F	12	56.3	77.0	17021
14	Mary	F	15	66.5	112.0	15790
15	Philip	M	16	72.0	150.0	15665
16	Robert	M	12	64.8	128.0	16958
17	Ronald	M	15	67.0	133.0	15992
18	Thomas	M	11	57.5	85.0	17243
19	William	M	15	66.5	112.0	16067

Report Generated on 01SEP2018

Class Report
Age=13

Obs	Name	Sex	Age	Height	Weight	Birthdate
2	Alice	F	13	56.5	84	16756
3	Barbara	F	13	65.3	98	16451
9	Jeffrey	M	13	62.5	84	16552

Report Generated on 26APR2021

Class Report
Age=13

The MEANS Procedure

Variable	Label	N	Mean	Std Dev	Minimum	Maximum
MSRP	Manufacturer Suggested Retail Price	262	29773.62	15584.59	10280.00	128420.00
MPG_Highway	Highway Miles per Gallon	262	28.6297710	4.4674591	17.0000000	46.0000000

Report Generated on 26APR2021

Class Report
Age=13

The FREQ Procedure

Origin=Asia

Type	Frequency	Percent	Cumulative Frequency	Cumulative Percent
Hybrid	3	1.90	3	1.90
SUV	25	15.82	28	17.72
Sedan	94	59.49	122	77.22
Sports	17	10.76	139	87.97
Truck	8	5.06	147	93.04
Wagon	11	6.96	158	100.00

Report Generated on 26APR2021

Class Report
Age=13

The FREQ Procedure

Origin=Europe

Type	Frequency	Percent	Cumulative Frequency	Cumulative Percent
SUV	10	8.13	10	8.13
Sedan	78	63.41	88	71.54
Sports	23	18.70	111	90.24
Wagon	12	9.76	123	100.00

Report Generated on 26APR2021

Class Report
Age=13

The FREQ Procedure

Origin=USA

Type	Frequency	Percent	Cumulative Frequency	Cumulative Percent
SUV	25	17.01	25	17.01
Sedan	90	61.22	115	78.23
Sports	9	6.12	124	84.35
Truck	16	10.88	140	95.24
Wagon	7	4.76	147	100.00

Report Generated on 26APR2021

Category 5 Storms

BasinName=East Pacific

Season	Name	Max Wind (MPH)	Min Pressure	Start Date	Length of Storm (Days)
2015	PATRICIA	213	872	20OCT2015	4
1997	LINDA	184	902	09SEP1997	55
2009	RICK	178	906	15OCT2009	6
1994	JOHN	173	929	11AUG1994	30
2002	KENNA	167	913	22OCT2002	4
2014	MARIE	161	918	22AUG2014	11
2010	CELIA	161	921	18JUN2010	12
2006	IOKE	161	900	16AUG2006	22
2002	ELIDA	161	921	23JUL2002	8
2002	HERNAN	161	921	30AUG2002	7
1997	GUILLERMO	161	919	30JUL1997	25
1994	EMILIA	161	926	16JUL1994	9
1994	GILMA	161	920	21JUL1994	10

BasinName=North Atlantic

Season	Name	Max Wind (MPH)	Min Pressure	Start Date	Length of Storm (Days)
1980	ALLEN	190	899	31JUL1980	11
2017	IRMA	185	914	30AUG2017	13
2005	WILMA	184	882	15OCT2005	11
1988	GILBERT	184	888	08SEP1988	12
2005	RITA	178	895	18SEP2005	8
1998	MITCH	178	905	22OCT1998	18
2017	MARIA	175	908	16SEP2017	14
2007	DEAN	173	905	13AUG2007	10
2007	FELIX	173	929	31AUG2007	6
2005	KATRINA	173	902	23AUG2005	8
1992	ANDREW	173	922	16AUG1992	12
2016	MATTHEW	167	934	28SEP2016	12
2004	IVAN	167	910	02SEP2004	22
2003	ISABEL	167	915	06SEP2003	14
2005	EMILY	161	929	11JUL2005	10
1989	HUGO	161	918	10SEP1989	15

BasinName=South Pacific

Season	Name	Max Wind (MPH)	Min Pressure	Start Date	Length of Storm (Days)
2016	WINSTON	173	884	10FEB2016	16

Report Generated on 26APR2021

```
************************************************************;
*  p105d02.sas Creating Frequency Reports and Graphs              *;
************************************************************;
*  Syntax and Example                          *;
*                                          *;
*   ODS GRAPHICS ON;                           *;
*   PROC FREQ DATA=input-table <proc-options>;         *;
*      TABLES col-name(s) / options;               *;
*   RUN;                                    *;
*                                          *;
*   PROC FREQ statement options:                   *;
*      ORDER=FREQ|FORMATTED|DATA                   *;
*      NLEVELS                                 *;
*   TABLES statement options:                      *;
*      NOCUM                                   *;
*      NOPERCENT                                 *;
*      PLOTS=FREQPLOT                             *;
*        (must turn on ODS Graphics)               *;
*      OUT=output-table                           *;
************************************************************;

ods graphics on;

proc freq data=sashelp.heart order=freq nlevels;
          tables Chol_Status / nocum plots=freqplot(orient=horizontal scale=freq);
run;

************************************************************;
*  Demo                                    *;
```

```
*   1) Highlight the PROC FREQ step and run the selected *;
*      code. Examine the default results.              *;
*   2) In the PROC FREQ statement, add the ORDER=FREQ    *;
*      option to sort results by descending frequency.   *;
*      Add the NLEVELS option to include a table with    *;
*      the number of distinct values.                    *;
*   3) Add the NOCUM option in the TABLES statement to   *;
*      suppress the cumulative columns.                  *;
*   4) Change Season to StartDate in the TABLES          *;
*      statement. Add a FORMAT statement to display      *;
*      StartDate as the month name (MONNAME.).           *;
*   5) Add the ODS GRAPHICS ON statement before PROC     *;
*      FREQ. Use the PLOTS=FREQPLOT option in the TABLES *;
*      statement to create a bar chart. Add the chart    *;
*      options ORIENT=HORIZONTAL and SCALE=PERCENT.      *;
*   6) Add the title Frequency Report for Basin and      *;
*      Storm Month. Turn off the procedure title with    *;
*      the ODS NOPROCTITLE statement. Add a LABEL        *;
*      statement to display BasinName as Basin and       *;
*      StartDate as Storm Month. Clear the titles and    *;
*      turn the procedure titles back on.                *;
************************************************************;

ods graphics on;
ods noproctitle;
title "Frequency Report for Basin and Storm Month";
proc freq data=pg1.storm_final order=freq nlevels;
        tables BasinName StartDate / nocum plots=freqplot(orient=horizontal scale=percent);
        format StartDate monname.;
```

run;

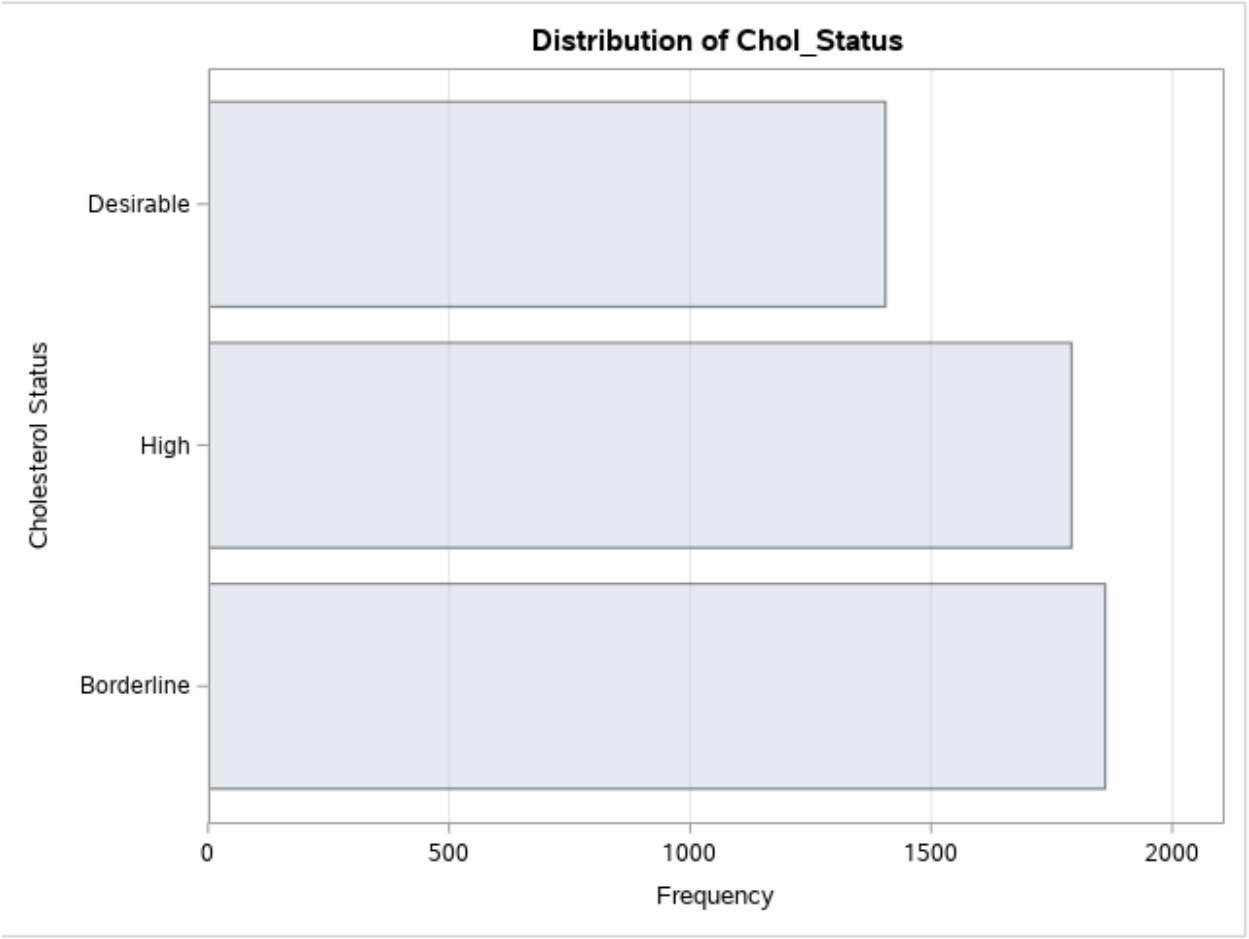
title;

ods proctitle;

The FREQ Procedure

Number of Variable Levels				
Variable	Label	Levels	Missing Levels	Nonmissing Levels
Chol_Status	Cholesterol Status	4	1	3

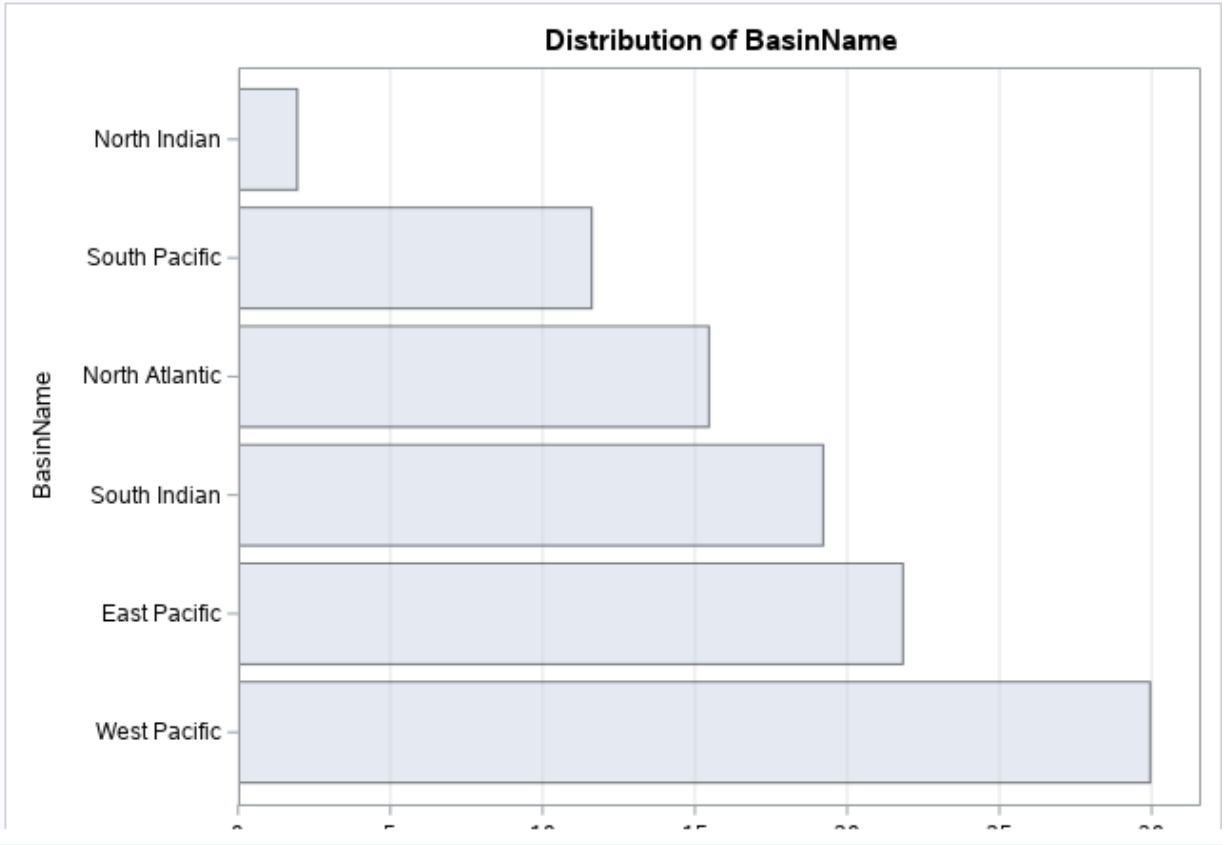
Cholesterol Status		
Chol_Status	Frequency	Percent
Borderline	1861	36.80
High	1791	35.42
Desirable	1405	27.78
Frequency Missing = 152		



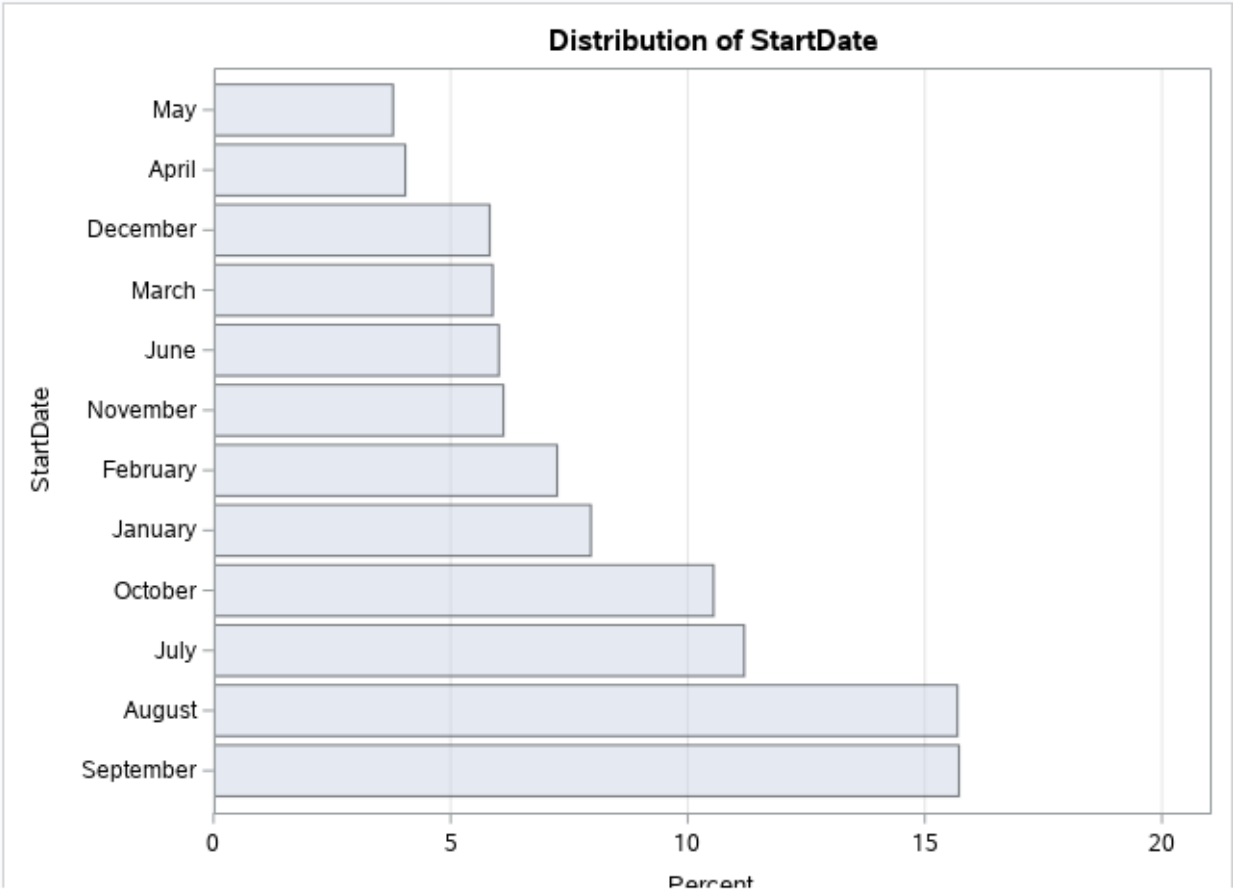
Frequency Report for Basin and Storm Month

Number of Variable Levels	
Variable	Levels
BasinName	6
StartDate	12

BasinName	Frequency	Percent
West Pacific	926	29.95
East Pacific	675	21.83
South Indian	594	19.21
North Atlantic	478	15.46
South Pacific	359	11.61
North Indian	60	1.94



StartDate	Frequency	Percent
September	486	15.72
August	485	15.69
July	346	11.19
October	326	10.54
January	246	7.96
February	224	7.24
November	189	6.11
June	186	6.02
March	182	5.89
December	180	5.82
April	125	4.04
May	117	3.78



```
*************************************************************;
*  p105d04.sas Creating Two-Way Frequency Reports              *;
*************************************************************;
*  Syntax and Example                          *;
*                                              *;
*   PROC FREQ DATA=input-table;                 *;
*     TABLES col-name*col-name </ options>;     *;
*   RUN;                                       *;
*                                              *;
*   PROC FREQ statement options:                *;
*     NOPRINT                                  *;
*   TABLES statement options:                   *;
*     NOROW, NOCOL, NOPERCENT                  *;
*     CROSSLIST, LIST                          *;
*     OUT=output-table                         *;
*************************************************************;

title "Blood Pressure by Cholesterol Status";
proc freq data=sashelp.heart;
          tables BP_Status*Chol_Status;
run;
title;

*************************************************************;
*  Demo (Highlight the PROC FREQ step and run                *;
*     the selected code after each step.)                   *;
*   1) Highlight the PROC FREQ step, run the selected       *;
*     code, and examine the default results.                *;
*   2) Add the NOPERCENT, NOROW, and NOCOL options in       *;
```

```
*     the TABLES statement.                    *;
*   3) Delete the options in the TABLES statement and   *;
*     add the CROSSLIST option.                *;
*   4) Change the CROSSLIST option to the LIST option in *;
*     the TABLES statement.                    *;
*   5) Delete the previous options and add          *;
*     OUT=STORMCOUNTS. Add NOPRINT to the PROC FREQ   *;
*     statement to suppress the report.          *;
************************************************************;



proc freq data=pg1.storm_final;
        tables BasinName*StartDate;
        format StartDate monname.;
        label BasinName="Basin"
                StartDate="Storm Month";
run;

proc freq data=pg1.storm_final noprint;
        tables BasinName*StartDate / norow nocol nopercent;
        format StartDate monname.;
        label BasinName="Basin"
                StartDate="Storm Month";
run;

proc freq data=pg1.storm_final noprint;
        tables BasinName*StartDate / crosslist;
        format StartDate monname.;
        label BasinName="Basin"
```

```
                StartDate="Storm Month";
run;

proc freq data=pg1.storm_final noprint;
        tables BasinName*StartDate / list;
        format StartDate monname.;
        label BasinName="Basin"
                StartDate="Storm Month";
run;

proc freq data=pg1.storm_final noprint;
        tables BasinName*StartDate / out=stormcounts;
        format StartDate monname.;
        label BasinName="Basin"
                StartDate="Storm Month";
run;

/*Level 1 Practice: Creating One-Way Frequency Reports
TOTAL POINTS 2
1.
Question 1
The pg1.np_species table provides a detailed species list for selected national parks.
Use this table to analyze the categories of reported species. If necessary, start SAS Studio before you begin.

Reminder: If you restarted your SAS session, you must run libname.sas.

Write a PROC FREQ step to analyze rows from pg1.np_species as follows:
```

Use the TABLES statement to generate a frequency table for Category.

Use the NOCUM option to suppress the cumulative columns.

Use the ORDER=FREQ option in the PROC FREQ statement to sort the results by descending frequency.

Use Categories of Reported Species as the report title.

Submit the program and review the results.

What percent of the species are Fungi?

Note: For your answer, type exactly what you see for the column value.

*/

```
title "Categories of Reported Species";
proc freq data=pg1.np_species order=freq;
        tables Category / nocum;
run;
```

/*Modify the PROC FREQ step to make the following changes:

Include only the rows where Species_ID starts with EVER and Category is not Vascular Plant.

Note: EVER represents Everglades National Park.

Turn on ODS Graphics before the PROC FREQ step and turn off the procedure title.

Add the PLOTS=FREQPLOT option to display frequency plots.

Add in the Everglades as a second title.

Submit the program and review the results.

Which Category value has the smallest frequency?

*/

```
ods graphics;
ods noproctitle;

title "Categories of Reported Species";
title2 "in the Everglades";
```

```
proc freq data=pg1.np_species order=freq;
        where Species_ID like "EVER%" and Category ne "Vascular Plant";
        tables Category / nocum plots=freqplot;
run;
title;
ods proctitle;
```

```
/* Level 2 Practice: Creating Two-Way Frequency Reports
TOTAL POINTS 2
1.
Question 1
The pg1.np_codelookup table is primarily used to look up a park name or park code. However, the table also includes columns for the park type and park region. Use this table to analyze the frequency of park types by the various regions. If necessary, start SAS Studio before you begin.

Reminder: If you restarted your SAS session, you must run the libname.sas program to access your course files.

Open a new program window and write a PROC FREQ step to analyze rows from pg1.np_codelookup.
Generate a two-way frequency table for Type by Region.
Exclude any park type that contains the word Other.
The levels with the most rows should come first in the order.
Suppress the display of column percentages.
Use Park Types by Region as the report title.
Submit the program and review the results.
What are the top three park types based on total frequency count?
*/
title "Park Types by Region";
```

```
proc freq data=pg1.np_codelookup order=freq;
        where Type NOT like "%Other%";
        table Type*Region / nocol;
run;
title;

/*Question 2
Modify the PROC FREQ step to make the following changes:

Limit the park types to the three that were determined in the previous step.
In addition to suppressing the display of column percentages, use the CROSSLIST option to display the table.
Add a frequency plot that groups the bars by the row variable, displays row percentages, and has a horizontal orientation.
Note: Use SAS documentation to learn how the GROUPBY=, SCALE=, and ORIENT= options can be used to control the appearance of the plot.
Use Selected Park Types by Region as the report title.
Submit the program and review the results.
Which Region has the highest Row Percent value?
*/
title "Selected Park Types by Region";
proc freq data=pg1.np_codelookup order=freq;
        where Type in ("National Historic Site", "National Monument", "National Park");
        table Type*Region / nocol crosslist plots=freqplot(groupby=row orient=horizontal
scale=percent);
run;
title;
```

Blood Pressure by Cholesterol Status

The FREQ Procedure

Frequency
Percent
Row Pct
Col Pct

Table of BP_Status by Chol_Status				
	Chol_Status(Cholesterol Status)			
BP_Status(Blood Pressure Status)	Borderline	Desirable	High	Total
High	798 15.78 36.21 42.88	456 9.02 20.69 32.46	950 18.79 43.10 53.04	2204 43.58
Normal	793 15.68 38.09 42.61	634 12.54 30.45 45.12	655 12.95 31.46 36.57	2082 41.17
Optimal	270 5.34 35.02 14.51	315 6.23 40.86 22.42	186 3.68 24.12 10.39	771 15.25
Total	1861 36.80	1405 27.78	1791 35.42	5057 100.00
Frequency Missing = 152				

The FREQ Procedure

Frequency
Percent
Row Pct
Col Pct

Table of BasinName by StartDate													
	StartDate(Storm Month)												
BasinName(Basin)	November	December	January	February	March	April	May	June	July	August	September	October	Total
East Pacific	15 0.49 2.22 7.94	3 0.10 0.44 1.67	2 0.06 0.30 0.81	0 0.00 0.00 0.00	1 0.03 0.15 0.55	0 0.00 0.00 0.00	29 0.94 4.30 24.79	76 2.46 11.26 40.86	156 5.05 23.11 45.09	163 5.27 24.15 33.61	142 4.59 21.04 29.22	88 2.85 13.04 26.99	675 21.83
North Atlantic	26 0.84 5.44 13.76	5 0.16 1.05 2.78	1 0.03 0.21 0.41	0 0.00 0.00 0.00	0 0.00 0.00 0.00	2 0.06 0.42 1.60	10 0.32 2.09 8.55	30 0.97 6.28 16.13	45 1.46 9.41 13.01	132 4.27 27.62 27.22	152 4.92 31.80 31.28	75 2.43 15.69 23.01	478 15.46
North Indian	19 0.61 31.67 10.05	7 0.23 11.67 3.89	1 0.03 1.67 0.41	0 0.00 0.00 0.00	0 0.00 0.00 0.00	4 0.13 6.67 3.20	8 0.26 13.33 6.84	5 0.16 8.33 2.69	1 0.03 1.67 0.29	0 0.00 0.00 0.00	2 0.06 3.33 0.41	13 0.42 21.67 3.99	60 1.94
South Indian	41 1.33 6.90 21.69	89 2.88 14.98 49.44	139 4.50 23.40 56.50	126 4.08 21.21 56.25	93 3.01 15.66 51.10	66 2.13 11.11 52.80	14 0.45 2.36 11.97	4 0.13 0.67 2.15	3 0.10 0.51 0.87	1 0.03 0.17 0.21	4 0.13 0.67 0.82	14 0.45 2.36 4.29	594 19.21
South Pacific	14 0.45 3.90 7.41	37 1.20 10.31 20.56	88 2.85 24.51 35.77	93 3.01 25.91 41.52	74 2.39 20.61 40.66	32 1.03 8.91 25.60	13 0.42 3.62 11.11	4 0.13 1.11 2.15	0 0.00 0.00 0.00	0 0.00 0.00 0.00	0 0.00 0.00 0.00	4 0.13 1.11 1.23	359 11.61
West Pacific	74 2.39 7.99 39.15	39 1.26 4.21 21.67	15 0.49 1.62 6.10	5 0.16 0.54 2.23	14 0.45 1.51 7.69	21 0.68 2.27 16.80	43 1.39 4.64 36.75	67 2.17 7.24 36.02	141 4.56 15.23 40.75	189 6.11 20.41 38.97	186 6.02 20.09 38.27	132 4.27 14.25 40.49	926 29.95
Total	189 6.11	180 5.82	246 7.96	224 7.24	182 5.89	125 4.04	117 3.78	186 6.02	346 11.19	485 15.69	486 15.72	326 10.54	3092 100.00

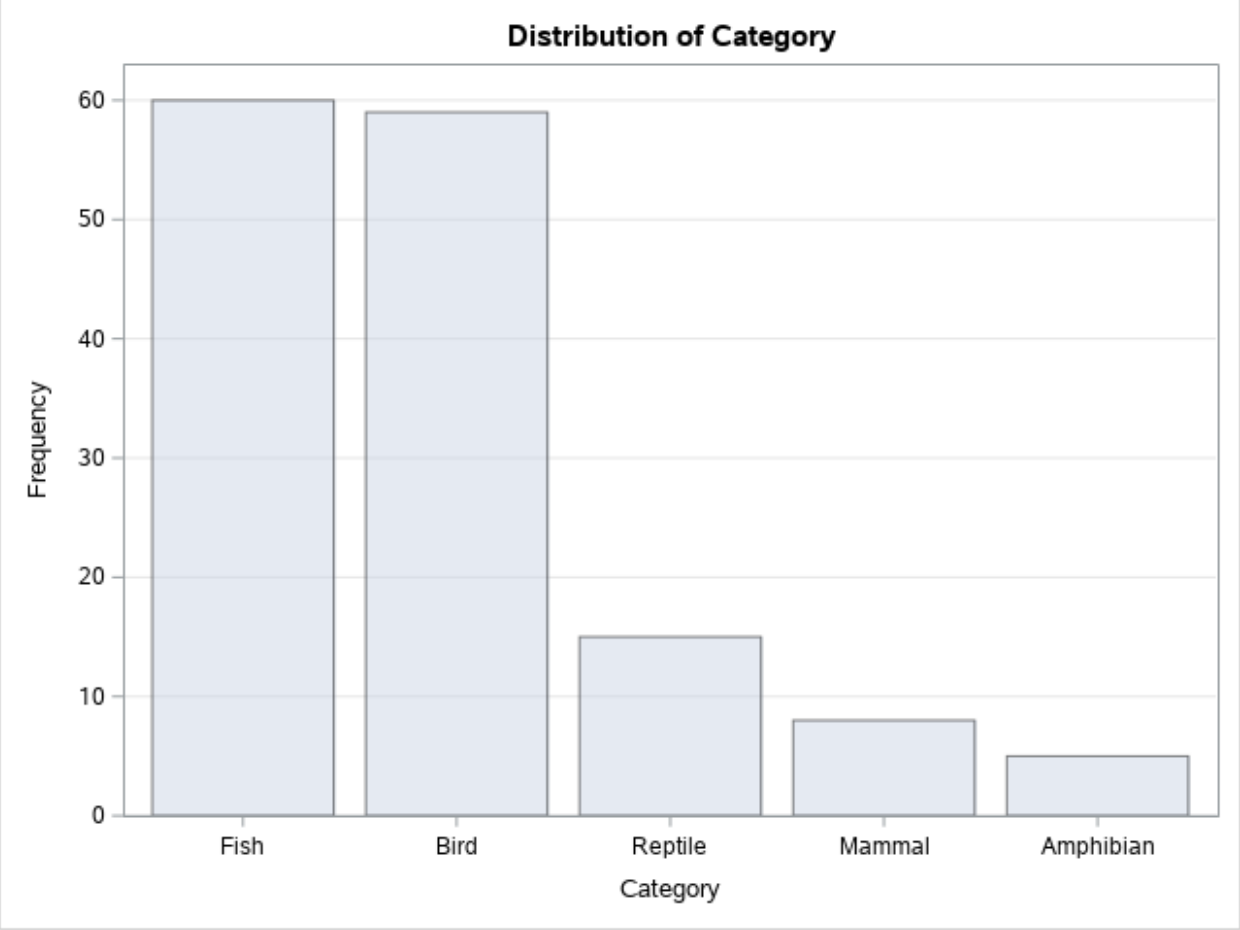
Categories of Reported Species

The FREQ Procedure

Category	Frequency	Percent
Vascular Plant	9614	54.30
Bird	2141	12.09
Insect	2104	11.88
Fungi	939	5.30
Nonvascular Plant	672	3.80
Mammal	619	3.50
Fish	576	3.25
Invertebrate	224	1.27
Reptile	216	1.22
Algae	167	0.94
Spider/Scorpion	121	0.68
Amphibian	114	0.64
Slug/Snail	110	0.62
Crab/Lobster/Shrimp	89	0.50

Categories of Reported Species in the Everglades

Category	Frequency	Percent
Fish	60	40.82
Bird	59	40.14
Reptile	15	10.20
Mammal	8	5.44
Amphibian	5	3.40



Park Types by Region

The FREQ Procedure

Frequency
Percent
Row Pct

Table of Type by Region

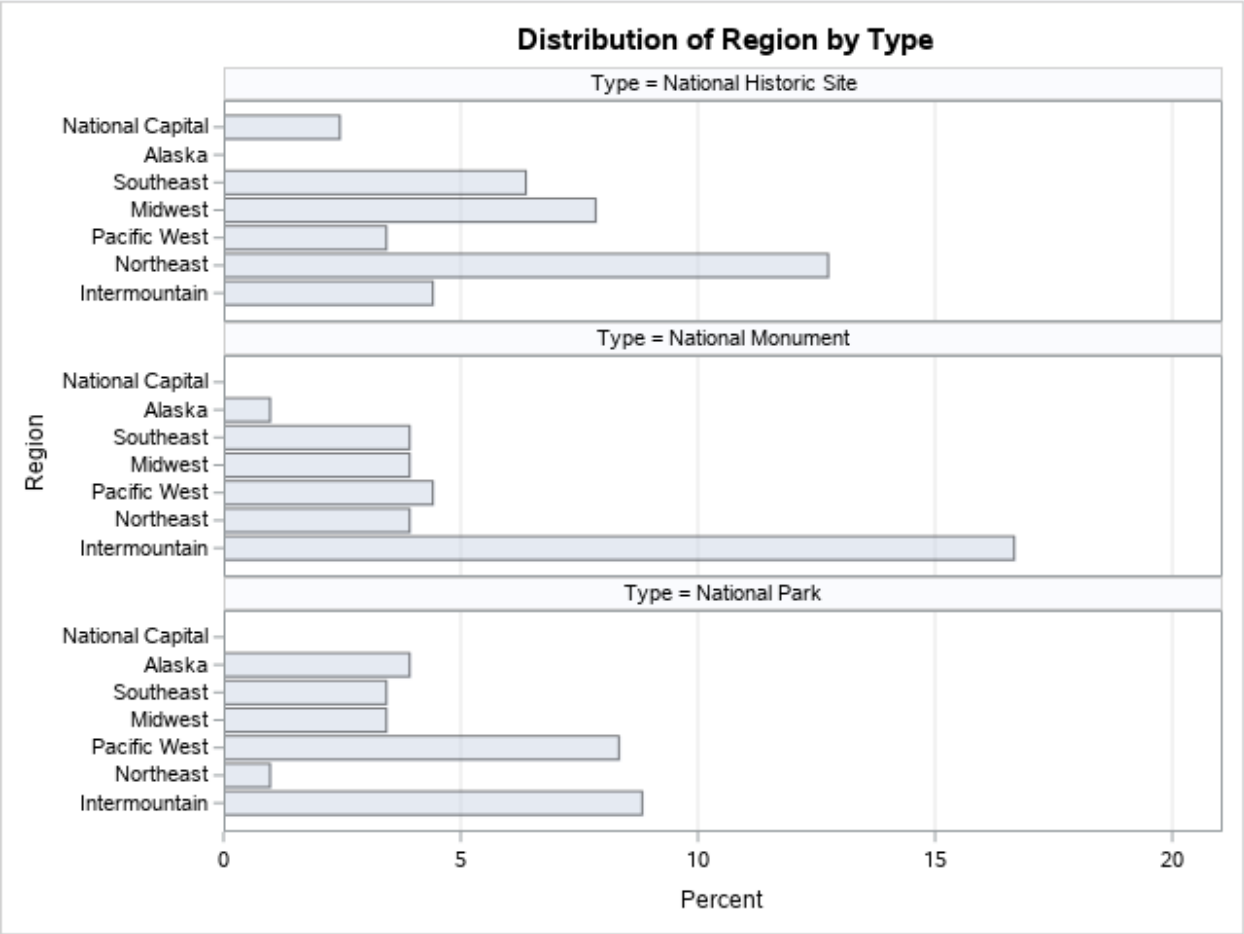
Type	Region							
	Intermountain	Northeast	Southeast	Pacific West	Midwest	National Capital	Alaska	Total
National Historic Site	9 2.52 11.84	26 7.28 34.21	13 3.64 17.11	7 1.96 9.21	16 4.48 21.05	5 1.40 6.58	0 0.00 0.00	76 21.29
National Monument	34 9.52 49.28	8 2.24 11.59	8 2.24 11.59	9 2.52 13.04	8 2.24 11.59	0 0.00 0.00	2 0.56 2.90	69 19.33
National Park	18 5.04 30.51	2 0.56 3.39	7 1.96 11.86	17 4.76 28.81	7 1.96 11.86	0 0.00 0.00	8 2.24 13.56	59 16.53
National Historical Park	6 1.68 13.95	14 3.92 32.56	7 1.96 16.28	9 2.52 20.93	3 0.84 6.98	2 0.56 4.65	2 0.56 4.65	43 12.04
National Memorial	2 0.56 6.90	7 1.96 24.14	3 0.84 10.34	1 0.28 3.45	5 1.40 17.24	11 3.08 37.93	0 0.00 0.00	29 8.12
National Recreation Area	6 1.68 35.29	3 0.84 17.65	1 0.28 5.88	7 1.96 41.18	0 0.00 0.00	0 0.00 0.00	0 0.00 0.00	17 4.76
National Battlefield	0 0.00 0.00	2 0.56 20.00	4 1.12 40.00	1 0.28 10.00	1 0.28 10.00	2 0.56 20.00	0 0.00 0.00	10 2.80
National Seashore	1 0.28 10.00	3 0.84 30.00	5 1.40 50.00	1 0.28 10.00	0 0.00 0.00	0 0.00 0.00	0 0.00 0.00	10 2.80
National Military Park	0 0.00 0.00	2 0.56 22.22	6 1.68 66.67	0 0.00 0.00	1 0.28 11.11	0 0.00 0.00	0 0.00 0.00	9 2.52
National Preserve	1 0.28 11.11	0 0.00 0.00	3 0.84 33.33	1 0.28 11.11	1 0.28 11.11	0 0.00 0.00	3 0.84 33.33	9 2.52
National Wild & Scenic River	1 0.28 14.29	2 0.56 28.57	1 0.28 14.29	0 0.00 0.00	3 0.84 42.86	0 0.00 0.00	0 0.00 0.00	7 1.96
National River	0 0.00 0.00	1 0.28 20.00	1 0.28 20.00	0 0.00 0.00	3 0.84 60.00	0 0.00 0.00	0 0.00 0.00	5 1.40
National Battlefield Park	0 0.00 0.00	1 0.28 25.00	1 0.28 25.00	0 0.00 0.00	1 0.28 25.00	1 0.28 25.00	0 0.00 0.00	4 1.12

Selected Park Types by Region

The FREQ Procedure

Table of Type by Region

Type	Region	Frequency	Percent	Row Percent
National Historic Site	Intermountain	9	4.41	11.84
	Northeast	26	12.75	34.21
	Pacific West	7	3.43	9.21
	Midwest	16	7.84	21.05
	Southeast	13	6.37	17.11
	Alaska	0	0.00	0.00
	National Capital	5	2.45	6.58
	Total	76	37.25	100.00
National Monument	Intermountain	34	16.67	49.28
	Northeast	8	3.92	11.59
	Pacific West	9	4.41	13.04
	Midwest	8	3.92	11.59
	Southeast	8	3.92	11.59
	Alaska	2	0.98	2.90
	National Capital	0	0.00	0.00
	Total	69	33.82	100.00
National Park	Intermountain	18	8.82	30.51
	Northeast	2	0.98	3.39
	Pacific West	17	8.33	28.81
	Midwest	7	3.43	11.86
	Southeast	7	3.43	11.86
	Alaska	8	3.92	13.56
	National Capital	0	0.00	0.00
	Total	59	28.92	100.00
Total	Intermountain	61	29.90	
	Northeast	36	17.65	
	Pacific West	33	16.18	
	Midwest	31	15.20	
	Southeast	28	13.73	
	Alaska	10	4.90	
	National Capital	5	2.45	
	Total	204	100.00	


Distribution of Region by Type
Type = National Historic Site
Type = National Monument
Type = National Park
Region
National Capital
Alaska
Southeast
Midwest
Pacific West
Northeast
Intermountain
0
5
10
15
20
Percent

```
*************************************************************;
*  p105d04.sas Creating Summary Statistics Reports          *;
*************************************************************;
*  Syntax and Example                                       *;
*                                                           *;
*   PROC MEANS DATA=input-table stat-list;                  *;
*     VAR col-name (s);                                     *;
*     CLASS col-name (s);                                   *;
*     WAYS n;                                               *;
*   RUN;                                                    *;
*************************************************************;

proc means data=sashelp.heart mean median std maxdec=1;
          var Height Weight Cholesterol;
          class Chol_Status BP_Status;
          ways 1;
run;

*************************************************************;
*  Demo (Highlight the PROC MEANS step and run the          *;
*      selected code after each step.)                      *;
*   1) Run the step and examine the starting report.        *;
*   2) List the following statistics in the PROC MEANS      *;
*     statement: MEAN, MEDIAN, MIN, and MAX. Add the        *;
*     MAXDEC=0 option to round statistics to the            *;
*     nearest integer.                                      *;
*   3) The CLASS statement can be used to calculate         *;
*     statistics for groups. Add a CLASS statement and      *;
*     list the BasinName column.                            *;
```

```
*   4) Add StormType as an additional column in the      *;
*      CLASS statement. Run the program and notice that  *;
*      one report is created with statistics that are    *;
*      calculated for the combination of BasinName and   *;
*      StormType values.                                 *;
*   5) The WAYS statement can be used to indicate the    *;
*      combinations of class columns to use for creating *;
*      the report. Add the WAYS statement and provide a  *;
*      value of 1.                                       *;
*   6) Change the WAYS statement to list 0, 1, and 2.    *;
************************************************************;

proc means data=pg1.storm_final mean median min max maxdec=0;
          var MaxWindMPH;
          class BasinName StormType;
          ways 0 1 2;
run;
```

The MEANS Procedure

Blood Pressure Status	N Obs	Variable	Mean	Median	Std Dev
High	2204	Height	64.7	64.5	3.6
		Weight	161.7	159.0	29.6
		Cholesterol	236.5	232.0	46.7
Normal	2082	Height	65.0	64.8	3.6
		Weight	149.2	147.0	26.8
		Cholesterol	223.0	219.0	41.8
Optimal	771	Height	64.6	64.3	3.5
		Weight	138.9	136.0	24.1
		Cholesterol	213.5	209.0	42.5

Cholesterol Status	N Obs	Variable	Mean	Median	Std Dev
Borderline	1861	Height	65.0	65.0	3.6
		Weight	154.3	152.0	28.6
		Cholesterol	219.1	220.0	11.5
Desirable	1405	Height	65.0	64.8	3.5
		Weight	148.4	144.0	29.6
		Cholesterol	177.8	180.0	16.1
High	1791	Height	64.5	64.3	3.6
		Weight	155.4	153.0	28.2
		Cholesterol	275.0	267.0	33.0

The MEANS Procedure

Analysis Variable : MaxWindMPH				
N Obs	Mean	Median	Minimum	Maximum
3038	80	75	6	213

Analysis Variable : MaxWindMPH					
StormType	N Obs	Mean	Median	Minimum	Maximum
Disturbance	289	75	63	17	178
Extratropical	758	90	86	35	184
Not Reported	647	78	69	6	173
Subtropical	4	60	52	43	92
Tropical	1340	76	69	23	213

Analysis Variable : MaxWindMPH					
BasinName	N Obs	Mean	Median	Minimum	Maximum
East Pacific	655	83	75	17	213
North Atlantic	461	82	75	23	190
North Indian	57	63	52	6	146
South Indian	585	78	69	23	155
South Pacific	354	78	69	35	173
West Pacific	926	80	81	40	144

Analysis Variable : MaxWindMPH						
BasinName	StormType	N Obs	Mean	Median	Minimum	Maximum
East Pacific	Disturbance	197	78	69	17	161
	Extratropical	19	117	115	40	173
	Not Reported	8	46	38	35	98
	Tropical	431	84	75	23	213
North Atlantic	Disturbance	92	69	58	23	178
	Extratropical	191	96	86	40	184
	Subtropical	3	65	52	52	92
	Tropical	175	75	63	35	190
North Indian	Not Reported	57	63	52	6	146
South Indian	Extratropical	97	83	78	35	144
	Not Reported	228	83	81	31	150
	Subtropical	1	43	43	43	43
	Tropical	259	71	63	23	155
South Pacific	Not Reported	354	78	69	35	173
West Pacific	Extratropical	451	87	86	40	144
	Tropical	475	73	69	40	144

```
*Week 5 Quizzes;
proc freq data=sashelp.cars;
   where Cylinders in (4,6) and Type in ('Sedan','SUV');
   tables Type*Cylinders / nocol norow crosslist;
run;

data baseball2;
   set sashelp.baseball;
   BatAvg=CrHits/CrAtBat;
   label BatAvg="Batting Average";
run;

proc print data=baseball2;
   var Name Team BatAvg;
run;

proc means data=baseball2;
   var BatAvg;
   class Team;
run;

title1 'The First Line';
title2 'The Second Line';
proc print data=pg1.np_final;
run;
title2 'The Next Line';
proc print data=pg1.np_final;
run;
title 'The Top Line';
```

```
proc print data=pg1.np_final;
run;
```

The FREQ Procedure

Table of Type by Cylinders			
Type	Cylinders	Frequency	Percent
SUV	4	7	2.77
	6	30	11.86
	Total	37	14.62
Sedan	4	96	37.94
	6	120	47.43
	Total	216	85.38
Total	4	103	40.71
	6	150	59.29
	Total	253	100.00

Obs	Name	Team	BatAvg
1	Allanson, Andy	Cleveland	0.22526
2	Ashby, Alan	Houston	0.24210
3	Davis, Alan	Seattle	0.28140
4	Dawson, Andre	Montreal	0.27985
5	Galarraga, Andres	Montreal	0.25505
6	Griffin, Alfredo	Oakland	0.25703
7	Newman, Al	Montreal	0.19626
8	Salazar, Argenis	Kansas City	0.21218
9	Thomas, Andres	Atlanta	0.25220
10	Thornton, Andre	Cleveland	0.25586
11	Trammell, Alan	Detroit	0.28072
12	Trevino, Alex	Los Angeles	0.24893

The MEANS Procedure

Analysis Variable : BatAvg Batting Average						
Team at the End of 1986	N Obs	N	Mean	Std Dev	Minimum	Maximum
Atlanta	11	11	0.2635799	0.0164637	0.2404555	0.2860886
Baltimore	15	15	0.2650890	0.0196035	0.2377817	0.2985420
Boston	10	10	0.2751029	0.0361943	0.2179487	0.3520518
California	13	13	0.2583657	0.0183789	0.2207407	0.2900506
Chicago	24	24	0.2600042	0.0185187	0.2305085	0.2880035
Cincinnati	12	12	0.2715615	0.0268060	0.2293907	0.3204420
Cleveland	12	12	0.2662801	0.0199865	0.2252560	0.2880741
Detroit	12	12	0.2679524	0.0125517	0.2471783	0.2807169
Houston	11	11	0.2678076	0.0134476	0.2420992	0.2873394
Kansas City	14	14	0.2610102	0.0308178	0.2121807	0.3138577
Los Angeles	14	14	0.2579128	0.0256615	0.2151703	0.3070727
Milwaukee	14	14	0.2570255	0.0269623	0.2155172	0.3000423
Minneapolis	13	13	0.2526106	0.0272962	0.2089552	0.3044606
Montreal	14	14	0.2503642	0.0314113	0.1962617	0.3048636
New York	24	24	0.2736371	0.0221246	0.2317189	0.3315340
Oakland	12	12	0.2604103	0.0184234	0.2329317	0.2918714
Philadelphia	12	12	0.2563488	0.0256152	0.2081575	0.2977208
Pittsburgh	11	11	0.2617873	0.0201612	0.2227603	0.2882411
San Diego	13	13	0.2696728	0.0260152	0.2475928	0.3257191
San Francisco	14	14	0.2582010	0.0222495	0.2228571	0.2929185
Seattle	12	12	0.2534791	0.0272142	0.2087379	0.3020566
St Louis	11	11	0.2588610	0.0198392	0.2209302	0.2981872
Texas	13	13	0.2619301	0.0157218	0.2352941	0.2886466
Toronto	11	11	0.2695877	0.0145457	0.2479375	0.2951252

The First Line
The Second Line

Obs	Region	Type	ParkName	DayVisits	Campers	OtherLodging	Acres
1	Alaska	MONUMENT	Cape Krusenstern National Monument	15,000	6375	0	649,096.15
2	Alaska	PARK	Kenai Fjords National Park	346,534	2162	0	669,650.05
3	Alaska	PARK	Kobuk Valley National Park	15,500	7050	0	1,750,716.16
4	Alaska	PRESERVE	Yukon-Charley Rivers National Preserve	1,146	3063	0	2,523,512.44
5	Alaska	PRESERVE	Bering Land Bridge National Preserve	2,642	1123	0	2,697,391.01
6	Alaska	PRESERVE	Noatak National Preserve	17,000	5500	0	6,587,071.39
7	Intermountain	MONUMENT	Alibates Flint Quarries National Monument	8,153	0	0	1,370.97
8	Intermountain	MONUMENT	Aztec Ruins National Monument	57,692	0	0	318.40
9	Intermountain	MONUMENT	Bandelier National Monument	198,478	10533	0	33,676.67
10	Intermountain	MONUMENT	Canyon De Chelly National Monument	821,406	11918	23,259	83,840.00
11	Intermountain	MONUMENT	Capulin Volcano National Monument	60,132	0	0	792.84
12	Intermountain	MONUMENT	Casa Grande Ruins National Monument	75,752	0	0	472.50
13	Intermountain	MONUMENT	Cedar Breaks National Monument	899,676	4277	0	6,154.60
14	Intermountain	MONUMENT	Chiricahua National Monument	51,277	11674	0	12,024.73
15	Intermountain	MONUMENT	Colorado National Monument	391,075	31161	0	20,536.39
16	Intermountain	MONUMENT	Devils Tower National Monument	496,210	17383	0	1,347.21
17	Intermountain	MONUMENT	Dinosaur National Monument	304,312	65402	0	210,281.92
18	Intermountain	MONUMENT	El Malpais National Monument	157,440	0	0	114,347.11

The First Line
The Next Line

Obs	Region	Type	ParkName	DayVisits	Campers	OtherLodging	Acres
1	Alaska	MONUMENT	Cape Krusenstern National Monument	15,000	6375	0	649,096.15
2	Alaska	PARK	Kenai Fjords National Park	346,534	2162	0	669,650.05
3	Alaska	PARK	Kobuk Valley National Park	15,500	7050	0	1,750,716.16
4	Alaska	PRESERVE	Yukon-Charley Rivers National Preserve	1,146	3063	0	2,523,512.44
5	Alaska	PRESERVE	Bering Land Bridge National Preserve	2,642	1123	0	2,697,391.01
6	Alaska	PRESERVE	Noatak National Preserve	17,000	5500	0	6,587,071.39
7	Intermountain	MONUMENT	Alibates Flint Quarries National Monument	8,153	0	0	1,370.97
8	Intermountain	MONUMENT	Aztec Ruins National Monument	57,692	0	0	318.40
9	Intermountain	MONUMENT	Bandelier National Monument	198,478	10533	0	33,676.67
10	Intermountain	MONUMENT	Canyon De Chelly National Monument	821,406	11918	23,259	83,840.00
11	Intermountain	MONUMENT	Capulin Volcano National Monument	60,132	0	0	792.84
12	Intermountain	MONUMENT	Casa Grande Ruins National Monument	75,752	0	0	472.50
13	Intermountain	MONUMENT	Cedar Breaks National Monument	899,676	4277	0	6,154.60
14	Intermountain	MONUMENT	Chiricahua National Monument	51,277	11674	0	12,024.73
15	Intermountain	MONUMENT	Colorado National Monument	391,075	31161	0	20,536.39
16	Intermountain	MONUMENT	Devils Tower National Monument	496,210	17383	0	1,347.21
17	Intermountain	MONUMENT	Dinosaur National Monument	304,312	65402	0	210,281.92
18	Intermountain	MONUMENT	El Malpais National Monument	157,440	0	0	114,347.11

The Top Line

Obs	Region	Type	ParkName	DayVisits	Campers	OtherLodging	Acres
1	Alaska	MONUMENT	Cape Krusenstern National Monument	15,000	6375	0	649,096.15
2	Alaska	PARK	Kenai Fjords National Park	346,534	2162	0	669,650.05
3	Alaska	PARK	Kobuk Valley National Park	15,500	7050	0	1,750,716.16
4	Alaska	PRESERVE	Yukon-Charley Rivers National Preserve	1,146	3063	0	2,523,512.44
5	Alaska	PRESERVE	Bering Land Bridge National Preserve	2,642	1123	0	2,697,391.01
6	Alaska	PRESERVE	Noatak National Preserve	17,000	5500	0	6,587,071.39
7	Intermountain	MONUMENT	Alibates Flint Quarries National Monument	8,153	0	0	1,370.97
8	Intermountain	MONUMENT	Aztec Ruins National Monument	57,692	0	0	318.40
9	Intermountain	MONUMENT	Bandelier National Monument	198,478	10533	0	33,676.67
10	Intermountain	MONUMENT	Canyon De Chelly National Monument	821,406	11918	23,259	83,840.00
11	Intermountain	MONUMENT	Capulin Volcano National Monument	60,132	0	0	792.84
12	Intermountain	MONUMENT	Casa Grande Ruins National Monument	75,752	0	0	472.50
13	Intermountain	MONUMENT	Cedar Breaks National Monument	899,676	4277	0	6,154.60
14	Intermountain	MONUMENT	Chiricahua National Monument	51,277	11674	0	12,024.73
15	Intermountain	MONUMENT	Colorado National Monument	391,075	31161	0	20,536.39
16	Intermountain	MONUMENT	Devils Tower National Monument	496,210	17383	0	1,347.21
17	Intermountain	MONUMENT	Dinosaur National Monument	304,312	65402	0	210,281.92
18	Intermountain	MONUMENT	El Malpais National Monument	157,440	0	0	114,347.11