

Level 2 Practice: Processing Statements Conditionally with DO Groups

Use conditional processing to split **pg1.np_summary** into two tables: **parks** and **monuments**.

Reminder: If you restarted your SAS session, you must recreate the **PG1** library so you can access your practice files. In SAS Studio, open and submit the **libname.sas** program in the **EPG194** folder. In Enterprise Guide, run the **Autoexec** process flow.

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*.)

```
data parks monuments;
  set pg1.np_summary;
  where type in ('NM', 'NP');
run;
```

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.

```
data parks monuments;
  set pg1.np_summary;
  where type in ('NM', 'NP');
  Campers=sum(OtherCamping, TentCampers, RVCampers,
              BackcountryCampers);
  format Campers comma17.;
run;
```

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.

```
data parks monuments;
  set pg1.np_summary;
  where type in ('NM', 'NP');
  Campers=sum(OtherCamping, TentCampers, RVCampers,
              BackcountryCampers);
  format Campers comma17.;
  length ParkType $ 8;
  if type='NP' then do;
    ParkType='Park';
    output parks;
  end;
  else do;
    ParkType='Monument';
    output monuments;
  end;
run;
```

4. Keep **Reg**, **ParkName**, **DayVisits**, **OtherLodging**, **Campers**, and **ParkType** in both output tables. Submit the program and view the output data.

```

data parks monuments;
  set pgl.np_summary;
  where type in ('NM', 'NP');
  Campers=sum(OtherCamping, TentCampers, RVCampers,
              BackcountryCampers);
  format Campers commal7.;
  length ParkType $ 8;
  if type='NP' then do;
    ParkType='Park';
    output parks;
  end;
  else do;
    ParkType='Monument';
    output monuments;
  end;
  keep Reg ParkName DayVisits OtherLodging Campers ParkType;
run;

```

5. How many rows are in each table?

Work.parks contains 51 rows, and **work.monuments** contains 63 rows.