3.4 - Practice

If you restarted your SAS session, open and submit the **libname.sas** program in the course files.

Level 1

1. Sorting Data and Creating an Output Table

Create the **np_sort** table that contains data for national parks ordered by regional code and decreasing numbers of daily visitors.

a. Open **p103p08.sas** from the **practices** folder. Modify the PROC SORT step to read **pg1.np_summary** and create a temporary sorted table named **np_sort**.

b. Add a BY statement to order the data by **Reg** and descending **DayVisits**.

c. Add a WHERE statement to select **Type** equal to *NP*. Submit the program.

	Reg	Type	ParkName	DayVisits	OtherLodging	OtherCamping	TentCa
1	A	NP	Kenai Fjords National Park	346,534	0	0	
2	A	NP	Kobuk Valley National Park	15,500	0	0	
3	IM	NP	Grand Canyon National Park	5,969,811	600,307	24,257	2
4	IM	NP	Rocky Mountain National Park	4,517,585	0	0	1
5	IM	NP	Zion National Park	4,295,127	86,456	0	1
6	IM	NP	Yellowstone National Park	4,257,177	579,227	583,068	1
7	IM	NP	Grand Teton National Park	3,270,076	196,577	297,084	
8	IM	NP	Glacier National Park	2,946,681	99,550	0	1

Level 2

2. Sorting Data to Remove Duplicate Rows

The **pg1.np_largeparks** table contains gross acreage for large national parks. There are duplicate rows for some locations.

a. Open and review the **pg1.np_largeparks** table. Notice that there are exact duplicate rows for some parks.

b. Create a new program. Write a PROC SORT step that creates two tables (**park_clean** and **park_dups**), and removes the duplicate rows. Submit the program.

park_clean

	UnitCode	AreaName	State	Reg	GrossAcres
1	ACAD	ACADIA NP	ME	NE	49057.36
2	AMIS	AMISTAD NRA	TX	IM	58500.00
3	APIS	APOSTLE ISLANDS NL	WI	MW	69377.43
4	ARCH	ARCHES NP	UT	IM	76678.98
5	ASIS	ASSATEAGUE ISLAND NS	MD-VA 2/	NE	41346.50
6	BADL	BADLANDS NP	SD	MW	242755.94
7	BAND	BANDELIER NM	NM	IM	33676.67

park_dups

	UnitCode	AreaName	State	Reg	GrossAcres
1	ASIS	ASSATEAGUE ISLAND NS	MD-VA 2/	NE	41346.50
2	BLCA	BLACK CANYON OF GUNNISON NP	CO	IM	30749.75
3	BLRI	BLUE RIDGE PKWY	NC-VA	NT	98774.04
4	BRCA	BRYCE CANYON NP	UT	IM	35835.08
5	BUFF	BUFFALO N RVR	AR	MW	94293.31
6	CACO	CAPE COD NS	MA	NE	43608.43

Challenge

3. Creating a Lookup Table from a Detailed Table

The **pg1.eu_occ** table includes multiple rows from each country code and country name.
Create a lookup table that includes a single row for each country code and name.

a. Create a new program. Write a PROC SORT step to sort **pg1.eu_occ** and create an output table named **countrylist**. Remove duplicate key values. Sort by **Geo** and then **Country**.

b. To read only **Geo** and **Country** from the **pg1.eu_occ** table, you can use the KEEP= data set option. Add the KEEP= option immediately after the input table and list **Geo** and **Country**.

data-set **(KEEP=***varlist***)**

c. Run the program and verify that only one row per country is included.

	Geo	Country
1	AT	Austria
2	BE	Belgium
3	BG	Bulgaria
4	CY	Cyprus
5	CZ	Czech Republic
6	DE	Germany

End of Practices