Creating and Using Custom Formats

1. Add a FORMAT statement in the DATA step to format the following values:
   a) **Date**: display three-letter month and four-digit year values
   b) **Volume**: add commas
   c) **CloseOpenDiff** and **HighLowDiff**: add dollar signs and include two decimal places.

```
format Date monyy7. Volume comma12.
       CloseOpenDiff HighLowDiff dollar8.2;
```

2. Add a FORMAT statement in the PROC MEANS step to format the values of **Date** to show only a four-digit year. Run the PROC MEANS step again.

```
format Date year4.;
```

3. What is the advantage of adding a FORMAT statement to the DATA step versus the PROC step?

Formats that you use in the DATA step are permanent attributes that are stored in the descriptor portion of the table. Formats that you use in a PROC step are temporary attributes.

4. In the PROC FORMAT step, modify the second VALUE statement to create a format named HRANGE that has the following criteria:
   1) A range of 50 – 57 has a formatted value of Below Average.
   2) A range of 58 – 60 has a formatted value of Average.
   3) A range of 61 – 70 has a formatted value of Above Average.

```
value hrange 50-57='Below Average'
             58-60='Average'
             61-70='Above Average';
```

5. In the PROC PRINT step, modify the FORMAT statement to format **Height** with the HRANGE format.

```
format Height hrange.;
```

6. In the PROC FORMAT statement, add the LIBRARY= option to save the formats to the **pg2.formats** catalog.

```
proc format library=pg2.formats;
```

7. Add an OPTIONS statement so that SAS can find the two permanent formats.

```
options fmtsearch=(pg2.formats);
```