Summary of Lesson 4: Creating Custom Formats

Creating and Using Custom Formats

```
PROC FORMAT;
        VALUE format-name value-or-range-1 = 'formatted-value'
                                   value-or-range-2 = 'formatted-value'
. . . ;
RUN;
```

- The FORMAT procedure is used to create custom formats.
- A VALUE statement specifies the criteria for creating one custom format.
- Multiple VALUE statements can be used in the PROC FORMAT step.
- The format name can be up to 32 characters in length, must begin with a $ followed by a letter or underscore for character formats, and must begin with a letter or underscore for numeric formats.
- On the left side of the equal sign, you specify individual values or a range of values that you want to convert to formatted values. Character values must be in quotation marks; numeric values are not quoted.
- On the right side of the equal sign, you specify the formatted values that you want the values on the left side to become. Formatted values need to be in quotation marks.
- The keywords LOW, HIGH, and OTHER can be used in the VALUE statement.
- You do not include a period in the format name when you create the format, but you do include the period in the name when you use the format.
- Custom formats can be used in the FORMAT statement and the PUT function.

Creating Custom Formats from Tables

```
PROC FORMAT CNTLIN=input-table FMTLIB;
        SELECT format-names;
RUN;
```

- The CNTLIN= option specifies a table from which PROC FORMAT builds formats.
- The input table must contain at a minimum three character columns:
    - **Start**, which represents the raw data values to be formatted.

- **Label**, which represents the formatted labels.

- **FmtName**, which contains the name of the format that you are creating. Character formats start with a dollar sign.