Creating and Redefining Variables using Assignment Statements

The basic form of the assignment statements

variable = expression;

On the left side of the equal sign is a variable name, either a new or an existing variable in the current dataset.

On the right side of the equal sign may appear a constant, another variable, or a mathematical expression.

Here are examples of some basic types of assignment statements:

Type of expression	Assignment statement
numeric constant	Var-name = 10;
character constant	Var-name = 'ten';
a variable	Var-name = Existing-Var;
addition	Var-name = Existing-Var + 10;
subtraction	Var-name = Existing-Var - 10;
multiplication	Var-name = Existing-Var * 10;
division	Var-name = Existing-Var / 10;
exponentiation	Var-name = Existing-Var ** 10;

Whether the variable Var (on the left side of the equal signs) is numeric or character depends on the expression that defines it. When the expression is numeric, Var will be numeric; when it is character, Var will be character.

When deciding how to interpret your expression, SAS follows the standard mathematical rules to precede: SAS performs exponentiation first, then multiplication and division, followed by addition and subtraction. You can use parentheses to override that order. If you use parentheses, your programs will be a lot easier to read.

Set Statement

Besides raw data files, SAS data sets can also be used as input to a DATA step. For example, you might want to use the information in an existing SAS data set to compute new variables. We can use Set statement to read in observations from an existing SAS dataset.

```
DATA scoredata1;      ---- new SAS dataset is going to created
  set scoredata0;     ---- existing SAS dataset to be read in
```