Exploring S-Plus 5.1 on UNIX (SOL) - Part II

In the August 1999 issue of RSS matters we provided an introduction to using S-Plus on SOL (http://www.unt.edu/benchmarks/archives/1999/august99/rss.htm). In this issue we continue our exploration of S-Plus on SOL.

Constructing Expressions in S-Plus

To begin our session we must invoke S-Plus after logging onto SOL. Type "Splus5" at the UNIX prompt to start the S-Plus session. You should see the following screen:

```
S-PLUS : Copyright (c) 1988, 1999 MathSoft, Inc.
S : Copyright Lucent Technologies, Inc.
Version 5.1 Release 1 for Sun SPARC, SunOS 5.5 : 1999
Working data will be in .Data
>
```

S-Plus expressions are typed in at the ">=" prompt. S-Plus will print out the results of the evaluation once the "Enter" key is pressed:

```
> 2+2
[1] 4
> sin(pi)
[1] 1.224647e-16
> sqrt(1000)
[1] 31.62278
>
```

An incomplete expression will lead to a second prompt, "+". You can continue with your expression at the second prompt:

```
> sqrt(100)
[1] 10
>
```

If the "+" prompt continues after pressing "Enter", then enter many "(" to get the "=" prompt back again. Then start your expression once again:

```
> sqrt( +
+
)))))
Problem: Syntax error: No opening parenthesis before unbalanced ("=") on input line 3:
>
```

 Scalars and Assignments

The assignment operator is the sequence of characters, "<" (less than) and "-" (hyphen). Assigning the variable “weight” the value of 190 we use the following:
Character values are inserted in quotes. If the quotes are omitted, S-Plus will look for a possibly non-existent data object called "Jim" to assign to the variable "person". The result is not printed until you enter the object name:

```r
> person
[1] "Jim"
> 
```

Vectors

The function "rnorm()", returns a vector of random deviates from the normal distribution The "[n]" on the left shows where the row starts:

```r
> rnorm(10)
[1] -0.63147304 1.25447805 -0.84064508 -0.36729337 0.09650417 -0.76198708
[7] 0.96427683 -2.32446837 0.10866023 0.73403819
> 
```

A single number is a vector of length 1. We can make vectors using the concatenation function, "c()". Then we can assign the integers 1,2,3 to the vector x:

```r
> mean(rnorm(10))
[1] -0.240037
> x<-c(1,2,3)
> x
[1] 1 2 3
> 
```

We can create a vector of names. Also we can create a vector of sequential integers using the function, "a:b", where a is the starting integer and b is the ending integer:

```r
> people<-c("Jim", "Sue", "Dave")
> people
[1] "Jim" "Sue" "Dave"
> 
```

```r
> seqvar<-5:10
> seqvar
[1] 5 6 7 8 9 10
> 
```

Object Names

Object names may contain letters, "abcDEF", or numbers, "0123456789", or a dot, ".". Examples of valid names: height, weight, x.var, .yvar, x.y.var, or x110. Objects names cannot use an underscore, a hyphen, begin with a number, or use reserved symbols. Examples of invalid object names: _xvar, y_var, x-yvar, 120xvar, T, F, or NA.

Handling Objects

We can list out all of the objects in our workspace:
Objects remain until removed, even if one quits S-Plus:

```r
> rm(x)
> x
Problem: Object "x" not found
```

### Objects as Variables

Objects can be used in expressions:

```r
> x<-1:10
> x
[1]  1  2  3  4  5  6  7  8  9 10
> mean(x)
[1] 5.5
> y<-c(x, 10)
> y
[1]  1  2  3  4  5  6  7  8  9 10 10
> length(y)
[1] 11
> 2*y
[1]  2  4  6  8 10 12 14 16 18 20 20
```

### Vector Arithmetic

Scalar Functions work on an element-wise basis. It is also possible to perform scalar and vector arithmetic:

```r
> x<-1:5
> x^2
[1]  1  4  9 16 25
> x
> 2*x
[1]  2  4  6  8 10
```

### Logical Vectors

Expressions with relational operators return logical vectors, "T" is True, "F" is False:
Missing Values

A missing value is represented by "NA". Operations on NA return NA. The function is.na() checks for missing values:

```r
> x<-c(1, NA, 3)
> x
[1] 1 NA 3
> x+1
[1] 2 NA 4
> sum(x)
[1] NA
> is.na(x)
[1] F T F
```

Vector Indexing

S-Plus uses brackets, [], to select elements of a vector. Negative indices remove elements:

```r
> x<-c(2,4,5,8,10)
> x
[1] 2 4 5 8 10
> x[1]
[1] 2
> x[3:5]
[1] 5 8 10
> x[c(1,2,3)]
[1] 2 4 6
> x[-c(1:3)]
[1] 8 10
```

Logical Indices

A logical index selects elements. Symbols for the logical operators are: "<" (less than), ">" (greater than), "<=" (less than or equal to), ">=" (greater than or equal to), "==" (equal to), "!" (negation operator), "!='" (not equal to).

```r
> x<-rnorm(5)
> x
[1] -0.6932126 -0.2386601 1.0713995 0.1983262 1.0289510
> x[x<=0]
[1] -0.6932126 -0.2386601
```
Replacement

You can use [ ] on the left hand side of an assignment, "<-" :

```r
> x <- sample(1:8)
> x
[1] 6 4 7 1 2 3 5 8
> x[2] <- NA
> x
[1] 6 NA 7 1 2 3 5 8
```

Next Time

Next time we will cover matrices, arrays, and lists, among other topics. Good luck with S-Plus!