SALSA Quick Reference:
Object Creation, Modification, Input, and Output Commands

(***Note***: The SALSA Language is **not** case sensitive. Use double hyphens ('--') to start a comment, which continues to the end of the line.)

LANGUAGE COMMANDS

Create

Description: Use the **create** command to create new variables, arrays, and array indexes (variables that reference array cells).

Examples:

```
create array a1 with 5 cells  --creates a new array 'a1' with 5 cells
create array a2 with 2 rows and 4 columns --creates a new 2 row by 4 column array 'a2'
create variable v1 --creates a new variable 'v1'; value defaults to 0
create variable i1 as index of a1 --creates new array index variable 'i1' at a1[0]
create variable i2 as index 2 of a1 --creates new array index variable 'i2' at a1[2]
```

Set

Description: Use the **set** command to create new variables and array indexes (variables that reference array cells), as well as to change the values and properties of existing variables, arrays, and array indexes.

Examples:

```
set v1 to 5 –-variable 'v1' is created if it doesn't already exist
set i1 to index 0 of a1 --array index 'i1' now references cell 0 of a1; i1 is created
                        --if it doesn't already exist
set a1[1] to random int between 1 and 5 --sets the variable at array location a1[1] to
                                        --a random integer value between 1 and 5
```

Populate

Description: Use the **populate** command to fill the empty cells of an existing array with variables.

Example

```
populate a1 with random ints between 1 and 100 --fills array 'a1' with random integers
populate a2 with random floats between 1.0 and 5.0 --fills array 'a2' w/random floats
populate a3 with random strings of length 2 --fills array 'a3' with random strings
populate a4 with random booleans --fills array 'a4' with random true/false values
```

Math

Description: Use the **math** commands (**add**, **subtract**, **multiply**, **divide**) to change an integer or float variable's value by applying a mathematical operation involving another integer or float value.

Tip: Use `add 1 to i1` to move array index `i1` to the next cell. Use `subtract 1 from i1` to move index `i1` to the previous cell.

Examples:

```
add 20 to v1 --adds 20 to the variable 'v1'
subtract v1 from v2 --subtracts the value of variable 'v1' from value of variable 'v2'
multiply value of v3 by 2.4 --multiplies the value of v3 by 2.4
divide a1[3] by 2 --The value of the variable at a1[3] is divided by 2
```

Print

Description: Use the **print** command to output a text string to the user. Use the & character to concatenate strings together, as illustrated below.

Examples:

```
print "The value of v1 is" & v1 --Assuming v1 = 3, prints out "The value of v1 is 3"
print "Array a1 has " & cells of a1 & " cells" --Assuming 'a1' has 5 cells, prints out
                                               --"Array a1 has 5 cells"
```

See next page over for more commands ☞

SALSA Quick Reference:
Conditional and iterative Execution Commands

LANGUAGE COMMANDS

If-Elseif-Else

Description: Use the **if…elseif…else** construct to specify blocks of code that execute conditionally based on the results of true-false tests.

Example:

```
if a1[left] = a1[right] --equality test
  print a1[left] & " equals " & a1[right]
elseif a1[left] < a1[right] --less than test
  print a1[left] & " is less than " & a1[right]
else
  print a1[left] & " is greater than " & a1[right]
endif
```

Tip: Below is a table of legal true-false (boolean) operators:

Operator	Meaning
=	is equal to
>	is greater than
<	less than
<=	is less than or equal to
>=	is greater than or equal to
<>	is not equal to

Tip: To enhance readability, you may enclose a true-false test in parentheses.

While

Description: Use the **while…endwhile** construct to specify a loop that executes as long as its true-false condition evaluates to true.

Example:

```
while i1 < cells of a1 --continue looping while
                       --array index 'i1' refers
                       --to a valid cell in 'a1'
  if a1[i1] > currentMax
    set currentMax to a1[i1]
  endif
  add 1 to i1
endwhile
```

Tip: True-false tests for **while** loops are the same as those in **if…elseif…else** statements; consult the table above for legal operators.

SALSA Quick Reference:
Data Movement Commands

LANGUAGE COMMANDS

Move

Description: Use the **move** command to move a variable, array, or array index to a new location.

Tip: It is illegal to move a variable to an array cell that is occupied. Before doing so, test whether the cell contains a variable by using the **is-occupied** operator, e.g., **if** a1[x] **is-occupied**…

Examples:

```
move v2 to a1[3] --moves variable 'v2' to cell 3 of a1, which must not be occupied.
move a1[x] to a1[x+1]  --moves variable at a1[x] one cell to the right
move i1 left --moves array index 'i1' left 1 cell
move i1 right 2 cells --moves array index 'i1' right 2 cells
```

Swap

Description: Use the **swap** command to cause two variables or array indexes to swap positions.

Examples:

```
swap a1[i] with a2[j]  --exchanges two variables in different arrays
swap a1[i] with a1[i+1] --exchanges two adjacent variables in same array exchange
```