Advanced Scripting   
Arithmetic and Assignment Operators

Last Updated: 10/11/2021 4:59 PM Version 1  
Document Prepared for: CIT361 Student

# Name Zach Lucas ID 895649438

# Instructions

Save a copy of this document. Answer all questions directly in this document. You will save and upload this completed document as your homework submission.

# Overview

Explore Arithmetic and Assignment Operators

# Requirements

PowerShell

# Task 1—Arithmetic Operators

## Steps

1. The + (addition/concatenation) operator:
   1. Defined for numbers, strings, and arrays. For both numbers and strings of differing data types are used, the right-hand operand will be converted to the left-hand type. Widening of the data type is performed if necessary. Try the following expressions, then record the returned value, type, and error if one occurs.

|  |  |  |  |
| --- | --- | --- | --- |
| **Expression** | **Value** | **Type** | **Error if any** |
| 1+2 | 3 | Int32 | N/A |
| 1 + '2' | 3 | Int32 | N/A |
| 1 + 'two' | Error | Error | Can’t convert the string “two” to an int32 |
| 1+ '2.5' | 3.5 | Double | N/A |
| '1'+ 2 | 12 | String | N/A |
| 'red' + 'blue' | redblue | String | N/A |
| 1,2,3 + 4 | 1 2 3 4 | Object[] | N/A |
| 1,2 + 2,3 | 1 2 2 3 | Object[] | N/A |

1. The \* operator:
   1. Defined for numbers, strings, and arrays. For both numbers and strings, if differing data types are used, the right-hand operand will be converted to the left-hand type. Widening of the data type is performed if necessary. Try the following expressions, then record the returned value, type, and error if one occurs.

|  |  |  |  |
| --- | --- | --- | --- |
| **Expression** | **Value** | **Type** | **Error if any** |
| 1\*2 | 2 | Int32 | N/A |
| 1 \* '2' | 2 | Int32 | N/A |
| 1\* '2.5' | 2.5 | Double | N/A |
| '1' \* 2 | 11 | String | N/A |
| 1 \* 'two' | Error | Error | Cannot convert “two” to an Int32 |
| 'red' \* 2 | redred | String | N/A |
| 1,2,3 \* 2 | 1 2 3 1 2 3 | Object[] | N/A |
| 1,2 \* 2,3 | Error | Error | Cannot convert an object to an Int32 |

1. – (subtraction) /(division) and %(modulus).
   1. Defined only for numbers. If no number data types are used, PowerShell will attempt to convert to numbers. Widening of the data type is performed if necessary. Try the following expressions, then record the returned value, type, and error if one occurs.

|  |  |  |  |
| --- | --- | --- | --- |
| **Expression** | **Value** | **Type** | **Error if any** |
| 1-2 | -1 | Int32 | N/A |
| 1 / '2' | 0.5 | Double | N/A |
| '10'-'5' | 5 | Int32 | N/A |
| '5' % 2 | 1 | Int32 | N/A |

# Task 2—Assignment Operators

The = (assignment) operator is used to assign a value or the results of an expression or pipeline to a variable. If a variable does not exist, the assignment operator creates it.

## Steps

1. Basic assignment  
   $a=10
   1. What is in $a? 10
2. Assignment as an expression result  
   $b=1gb/1mb
   1. What is in $b? 1024
3. Assignment as the result of an expression  
   $c=gps|measure|select count
   1. What is in $c? 528
4. Multiple assignment  
   $d=$e=$f=100
   1. What is in $d, $e, $f? 100
   2. And this one  
      $g=($h = ($i = 10) + 4) \* 2
   3. What are the values of $g, $h, and $i? $g = 28, $h = 14, $i = 10
5. You can combine the math operators to the assignment operator to perform a reassignment of the variable to its current contents with the operator and operand applied. Eg. $v+=1 is the same as

|  |  |
| --- | --- |
| **Enter** | **What is the value of $j** |
| $j=10 | 10 |
| $j\*=2 | 20 |
| $j-=20 | 0 |
| $j+=1000 | 1000 |
| $j/=10 | 100 |
| $j%=10 | 0 |

1. You can also assign multiple variables to an array of values.  
   $k,$l,$m=10,'red',(get-date)
   1. What is the value in $k? 10
   2. What is the value in $l? red
   3. What is the value in $m? Monday, October 11, 2021 6:10:43 PM
2. Easily swap values  
   $n='n'  
   $o='o'  
   $n,$o=$o,$n
   1. What is the value of $n? o
   2. What is the value of $o? n

# Deliverable

Upload this document with completed answers to i-learn.