Advanced Scripting   
Help Documentation

Last Updated: 5/27/2021 4:08 PM Version 1  
Document Prepared for: CIT361 Student

# Name Daniel Harris ID 235868292

# Instructions

Complete the steps then upload both this document and your completed PowerShell script.

# Overview

In this exercise your will add inline documentation to provide information about the function.

# Requirements

PowerShell

# Setup

Create a copy of the script you wrote for Exercise 7.2

# Task 1—Adding Help Documentation.

Help documentation is created by adding a multiline comment to the body of the function just after the function declaration. Start a help topic by using a . then the keyword on a single line. Everything you add below that will be in that help section.

## Steps

1. Using the information in the ‘about\_Comment\_Based\_Help’ help topic, create a help comment block in the Get-Soup function you created last exercise.
2. Include help for at least the following keywords
   1. SYNOPSIS
   2. DESCRIPTION
   3. EXAMPLE (at least two examples)
3. Test your help with the help command  
   get-help get-soup

# Wrap-up

Copy your script code here

<#  
.SYNOPSIS  
 Retrieves the specified quantity of soup.  
  
.DESCRIPTION  
 The Get-Soup function retrieves the specified quantity of soup based on the provided parameters.  
 It returns a string indicating the quantity, size, and type of soup requested.  
  
.PARAMETER Soup  
 Specifies the type of soup.  
  
.PARAMETER Size  
 Specifies the size of the soup.  
  
.PARAMETER Quantity  
 (Optional) Specifies the quantity of soup. The default value is 1.  
  
.PARAMETER Please  
 (Switch) When present, indicates a polite request for soup.  
  
.EXAMPLE  
 Get-Soup -Soup "Tomato" -Size "Large" -Quantity 2 -Please  
 Retrieves 2 Large bowls of Tomato soup for you.  
  
.EXAMPLE  
 Get-Soup -Soup "Chicken Noodle" -Size "Small" -Please  
 Retrieves 1 Small bowl of Chicken Noodle soup for you.  
  
#>  
function Get-Soup {  
 [Alias('Soup')]  
 param(  
 [Parameter(Mandatory=$true)]  
 [string]$soup,  
 [ValidateSet('Cup','Bowl','Pot')]  
 [string]$size,  
 [ValidateRange(1,5)]  
 [int]$Quantity = 1,  
 [switch]$Please   
 )  
 if ($please){  
 "$Quantity $size$(if($Quantity -gt 1){'s'}) of $soup soup for you."  
 }  
 else {  
 "No $soup soup for you!"  
 }  
}

# Deliverable

Upload this document with completed answers **and** your script to i-learn.