

## Infraestructura Informática - Práctica 4 – Sesión 3

### Programación de scripts de PowerShell

#### Alumno 1: Francisco Puga Lojo UO272109

**(1) EJERCICIO.** El objetivo de este ejercicio es construir un script que cree un fichero con información relativa a un proceso, cuya identificación es introducida por el usuario a través de la consola. El script ha de realizar las siguientes tareas: ○ Enviar a la consola un mensaje explicativo de su cometido. El mensaje enviado será el

siguiente: "Script que genera un fichero informativo de un proceso indicado mediante su ID".

○ Solicitar al usuario el ID del proceso. Para ello, se ha de enviar a la consola el siguiente mensaje: "Introduzca el ID del proceso".

○ Solicitar al usuario la ruta del directorio en el que se creará el fichero informativo. Para ello, se ha de enviar a la consola el siguiente mensaje: "Introduzca la ruta de la carpeta en la que se creará el fichero informativo del proceso".

○ Con el ID del proceso y la ruta, el script debe generar un fichero cuyo nombre sea el mismo que el del proceso + la extensión *.txt*. El script escribirá en este fichero en líneas consecutivas el nombre, el ID y la hora de inicio del proceso. A continuación se proporciona un ejemplo del contenido almacenado en este fichero:

Nombre: explorer

ID: 1248

Hora de inicio: 03/26/2013 14:08:21

```
Write-Host "Script que genera un fichero informativo de un proceso  
indicado mediante su id"
```

```
$id = read-host "Introduce el ID del proceso: "  
$ruta = read-host "Introduce la ruta de la carpeta en la que se  
creará el fichero informativo del proceso: "  
$id = get-process -id $id  
$nombre = $id.name  
$hora = $id.starttime  
$ruta = $ruta + "\" + $nombre + ".txt"  
new-item -path $ruta -Type File add-  
content $ruta ("Nombre: " + $nombre) add-  
content $ruta ("ID: " + $id) add-content  
$ruta ("Hora :" + $hora)
```

**(2) EJERCICIO.** Crear un script que defina un *array* conteniendo los números del 0 al 9. El script debe ejecutar un bucle que recorra todos los elementos del *array* e imprima en la consola solamente los pares.

Este script se realizará con la herramienta PowerShell ISE y se almacenará en la carpeta *\Scripts* con el nombre *Prueba04.ps1*.

```
$array = "0 1 2 3 4 5 6 7 8 9"
$i =
0; do
{
    if($array[$i] %2 -eq 0){
write-host $array[$i]
    }
    $i++;
} while ($i -lt $array.length )
```

**(3) EJERCICIO.** Con objeto de probar más detenidamente el formateo extendido de cadenas, debes realizar un script que imprima en la consola una tabla con todos los procesos en ejecución en el sistema. La tabla tendrá dos columnas. En la columna de la izquierda se imprimirá el nombre de cada proceso, alineado a la izquierda y con un ancho de 25. En la columna de la derecha se escribirá el tamaño de la memoria virtual utilizada por el proceso (propiedad VM), alineado a la derecha (ya que es un valor numérico), y con un ancho de 25.

Este script se realizará con la herramienta Powershell ISE y se almacenará en la carpeta *\Scripts* con el nombre *Prueba05.ps1*.

```
$procesos = get-process $nombre =
"Nombre"; $mv = "Memoria Virtual";

write-host ('{0, -25} {1, 25}' -f $nombre, $mv)
for($i = 0; $i -lt $procesos.length ; $i++){
    $nombre = $procesos[$i].name
    $vm = $procesos[$i].vm
    write-host ('{0, -25} {1, 25}' -f $nombre, $vm)
}
```