

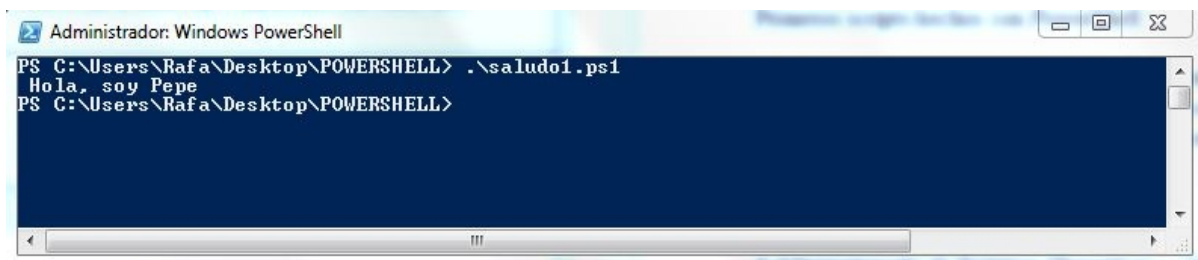
Ejercicios básicos de programación PowerShell

1. Crea una nueva carpeta en el escritorio llamada powershell, para ejecutar los cmdlets abre la consola de powershell y haz `cd .\Desktop\powershell`, desde el powershell ISE crea un cmdlet llamado `saludo1.ps1` que defina dos variables: un `$nombre` y un `$saludo`. Luego muestra por consola un mensaje en el que se muestre ese saludo y ese nombre.

```
# saludo1.ps1
# Administración de Sistemas Operativos – 2ASIR
```

```
$nombre = 'Pepe'
$saludo = 'Hola'
```

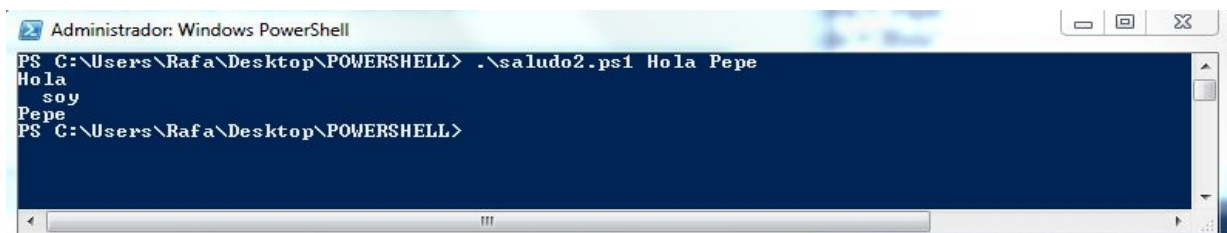
```
Write-Output " $saludo, soy $nombre "
```



2. Crea un cmdlet de PowerShell como el anterior llamado `saludo2.ps1`, pero en lugar de definir dos variables las debe recoger como argumentos de la consola. (`$args[0]` sería `$1` en bash y `$args[1]` sería `$2`)

```
# saludo2.ps1
# Administración de Sistemas Operativos – 2ASIR
```

```
Write-Output $args[0] " soy " $args[1]
```

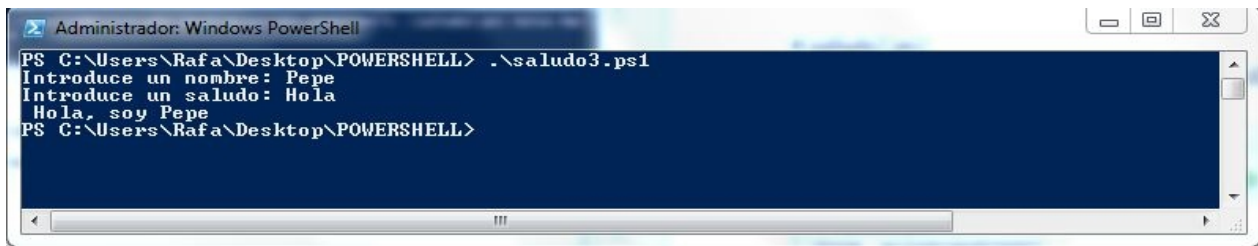


3. Crea un cmdlet de PowerShell como el anterior llamado `saludo3.ps1` pero en vez de definir o recoger variables como argumentos, debe solicitárselas al usuario con `Read-Host` (equivalente a `read` en bash o `set /p` en cmd)

```
# saludo3.ps1
# Administración de Sistemas Operativos – 2ASIR
```

```
$nombre=Read-Host "Introduce un nombre"
$saludo=Read-Host "Introduce un saludo"
```

```
Write-Output " $saludo, soy $nombre "
```



```
Administrador: Windows PowerShell
PS C:\Users\Rafa\Desktop\POWERSHELL> .\saludo3.ps1
Introduce un nombre: Pepe
Introduce un saludo: Hola
Hola, soy Pepe
PS C:\Users\Rafa\Desktop\POWERSHELL>
```

4. Crea un cmdlet de PowerShell que recoja dos variables por consola con Read y lleve a cabo todas las operaciones aritméticas entre ellas y las muestre por consola.

```
# operaciones.ps1
```

```
# Administración de Sistemas Operativos – 2ASIR
```

```
Write-Output "Vamos a realizar cuatro operaciones con dos numeros"
```

```
$numero1=Read-Host "Introduce el primero"
```

```
$numero2=Read-Host "Introduce el segundo"
```

```
$suma=[int]$numero1+[int]$numero2
```

```
Write-Output "La suma de los dos es:"$suma
```

```
$resta=[int]$numero1-[int]$numero2
```

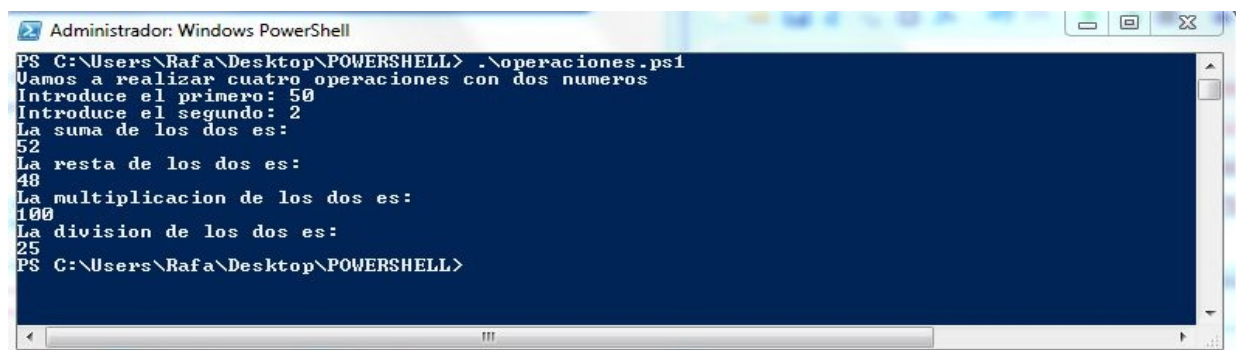
```
Write-Output "La resta de los dos es:"$resta
```

```
$multi=[int]$numero1*[int]$numero2
```

```
Write-Output "La multiplicacion de los dos es:"$multi
```

```
$div=[int]$numero1/[int]$numero2
```

```
Write-Output "La division de los dos es:"$div
```



```
Administrador: Windows PowerShell
PS C:\Users\Rafa\Desktop\POWERSHELL> .\operaciones.ps1
Vamos a realizar cuatro operaciones con dos numeros
Introduce el primero: 50
Introduce el segundo: 2
La suma de los dos es:
52
La resta de los dos es:
48
La multiplicacion de los dos es:
100
La division de los dos es:
25
PS C:\Users\Rafa\Desktop\POWERSHELL>
```

5. Crea un cmdlet de PowerShell que solicite dos números enteros al usuario. El cmdlet debe decirnos si uno es mayor, menor o igual que el otro.

```
# comparador.ps1
```

```
# Administración de Sistemas Operativos – 2ASIR
```

```
Write-Output "Vamos a comparar dos numeros"
```

```
$numero1=Read-Host "Introduce el primero"
```

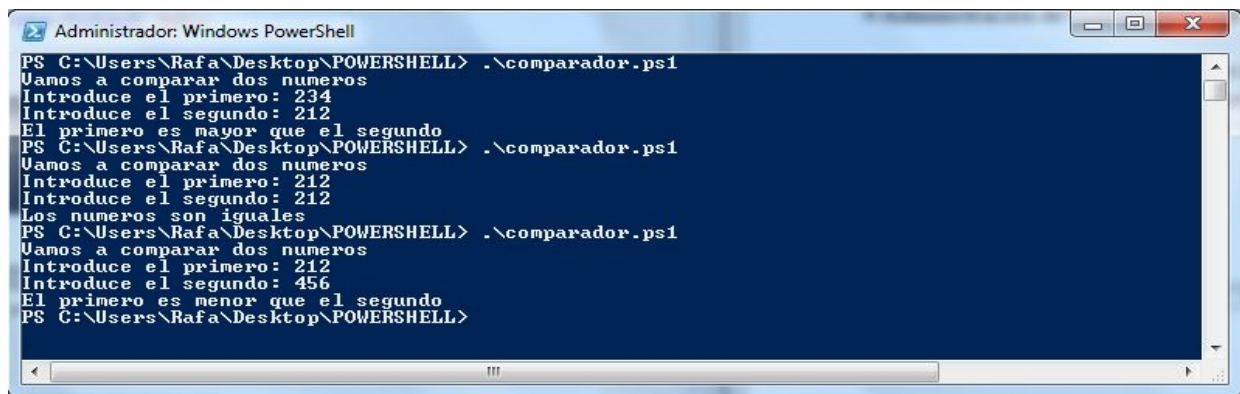
```
$numero2=Read-Host "Introduce el segundo"
```

```
if ($numero1 -eq $numero2) {
    Write-Output "Los numeros son iguales"
}
```

```
if ($numero1 -lt $numero2) {
    Write-Output "El primero es menor que el segundo"
}
```

```
if ($numero1 -gt $numero2) {
```

```
Write-Output "El primero es mayor que el segundo"
}
```



```
Administrador: Windows PowerShell
PS C:\Users\Rafa\Desktop\POWERSHELL> .\comparador.ps1
Vamos a comparar dos numeros
Introduce el primero: 234
Introduce el segundo: 212
El primero es mayor que el segundo
PS C:\Users\Rafa\Desktop\POWERSHELL> .\comparador.ps1
Vamos a comparar dos numeros
Introduce el primero: 212
Introduce el segundo: 212
Los numeros son iguales
PS C:\Users\Rafa\Desktop\POWERSHELL> .\comparador.ps1
Vamos a comparar dos numeros
Introduce el primero: 212
Introduce el segundo: 456
El primero es menor que el segundo
PS C:\Users\Rafa\Desktop\POWERSHELL>
```

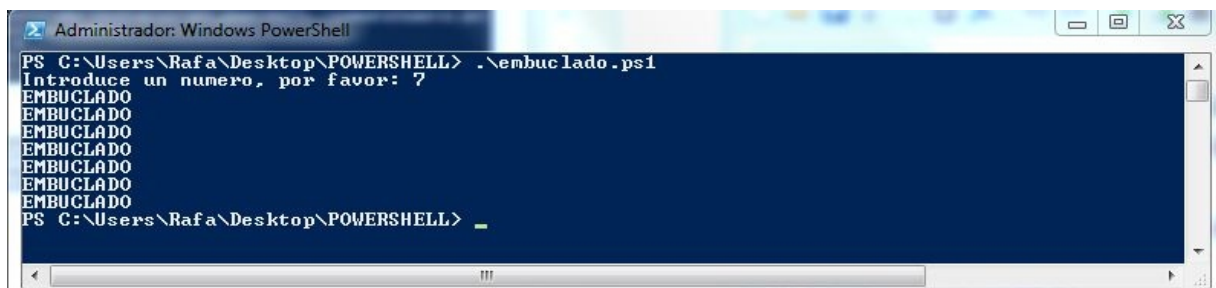
6. Crea un cmdlet que solicite al usuario un número, verifique que es positivo y programa un bucle para que muestre por consola la palabra EMBUCLADO tantas veces como indique el número.

```
# embuclado.ps1
# Administración de Sistemas Operativos – 2ASIR

$numero=Read-Host "Introduce un numero, por favor"
$numero=[int]$numero

while ($numero -lt 0) {
    $numero=Read-Host "Mete un numero positivo, por favor"
}

while ($numero -gt 0) {
    Write-Output "EMBUCLADO"
    $numero--
}
```



```
Administrador: Windows PowerShell
PS C:\Users\Rafa\Desktop\POWERSHELL> .\embuclado.ps1
Introduce un numero, por favor: 7
EMBUCLADO
EMBUCLADO
EMBUCLADO
EMBUCLADO
EMBUCLADO
EMBUCLADO
EMBUCLADO
PS C:\Users\Rafa\Desktop\POWERSHELL> _
```

7. Crea un cmdlet que construya una pirámide de números hasta el límite que se le pida al usuario,

Por ejemplo, introduciendo el número 9 debería dar como salida:

```
1
22
333
4444
55555
666666
7777777
88888888
999999999
```

```
# piramide.ps1
# Administración de Sistemas Operativos – 2ASIR

Write-Output "Vamos a hacer una piramide!"
$numero=Read-Host "Introduce un numero, por favor"
$numero=[int]$numero
```

```
for ($contador=1; $contador -le $numero; $contador++) {
    for ($veces=1; $veces -le $contador; $veces++) {
        Write-Host $contador -NoNewLine
    }
    Write-Host " "
}
```



```
Administrador: Windows PowerShell
Vamos a hacer una piramide!
Introduce un numero, por favor: 9
1
22
333
4444
55555
666666
7777777
88888888
999999999
PS C:\Users\Rafa\Desktop\POWERSHELL>
```