

# UNIX Shells (Bash) / PowerShell

## As Linguaxes de Shell

Os administradores de sistemas fan un uso intensivo das linguaxes de Shell (PowerShell, bash....) para automatizar diversas tarefas de mantemento dos sistemas. Os sistemas operativos máis importantes dispoñen dun intérprete de ordenes de shell de modo estándar, o que vulgarmente coñecemos como “*a consola*” ou “*o terminal*”. En *Windows* dispoñemos de dous tipos de shell: O “habitual” que provén dos tempos de MS/DOS e resulta un tanto limitado, e o máis moderno e completo **PowerShell**. En **Linux** dispoñemos de multitude de intérpretes de shell, con diversos graos de compatibilidade entre eles. Os máis coñecidos son **bash** e **dash**. Os sistemas Linux establecen o enlace simbólico **bin/sh** apuntando ao intérprete por defecto a utilizar no sistema.

## Bash

Bash é un shell comúnmente utilizado en Linux para a automatización de tarefas de administración, aínda que existen outros como *dash*, *zsh* ... etc. Bash posee todas as estruturas típicas das linguaxes de programación e pode facer uso de todas as utilidades do sistema, que no mundo UNIX normalmente están pensadas xa para ser utilizadas en scripts. Podemos destacar como potentísimas ferramentas *tar*, *grep*, *sed*, *tr*, *netcat*, ... ademais das tarefas comúns como cambio e creación de directorios, cambio de permisos, copia, renomeamento e movemento de arquivos...

Operacións básicas (ollo cos espazos en branco):

- **Entrada por teclado**

```
read variable
```

- **Saída por pantalla**

```
echo “texto literal $variable” // Para acceder ao valor da variable debemos antepoñer $ ao identificador
```

- **Asignación**

```
idade=23
nome=“Alfonso”
```

- **Operacións aritméticas**

```
dobreidade=$((idade * 2)); ou let dobreidade=idade*2;
```

- **Operacións lóxicas: Os operadores lóxicos son >, <, ==, !=, -gt, -lt, -ge, -le, -eq**

```
[ $idade -gt 18 ] // Comparacións numéricas
[ $nome == “Luis” ] // Comparacións alfabéticas
```

- **Operadores lóxicos**

```
&& (and) : [ $idade -gt 18 ] && [ $pais == “España ” // idade maior de 18 e pais = España
|| (or) : [ $idade -lt 18 ] || [ $idade -gt 65 ] // Idade menor de 18 ou maior de 65
! (not) : ! [ $nome == “Luis” ] // O nome non é Luis (pode usarse != )
```

- **Selección**

```
if [ $nome == "Luis" ] && [ $idade -gt 18 ]; then
    echo "$nome (Luís) és maior de idade"
else
    echo "Ou non eres Luís, ou non eres maior de idade"
fi
```

- **Iteración**

```
numero=1
while [ $numero -lt 20 ]; do
    echo "Conta: $numero"
    numero=$((numero + 1))
done
```

## PowerShell

O intérprete de comandos típico en Windows estaba baseado nun sistema operativo moi simple e de fai bastante tempo chamado MS/DOS, e proporcionaba unha utilidade limitada. As últimas versións de Windows proporcionan un Shell máis potente denominado PowerShell dende o que é posible administrar o sistema. Para habilitar a execución de scripts de PowerShell se debe executar dende unha sesión PowerShel normal o comando: ***Set-ExecutionPolicy -Scope CurrentUser RemoteSigned***

Operacións básicas:

- **Entrada por teclado**

```
$nome = Read-Host "Como te chamas?"
```

- **Saida por pantalla**

```
Write-Host "Mensaxe a visualizar"
```

- **Asignación**

```
$idade=23
$nome="Alfonso"
```

- **Operacións aritméticas**

```
$n = $n + 1
```

- **Selección:** Os operadores lóxicos son *-gt, -lt, -ge, -le* e as operacións lóxicas *-and, -or, -not*

```
if ( ( $nome == "Luis" ) -and ( $idade -gt 18 ) ) {
    Write-Host "$nome (Luís) és maior de idade"
} else {
    Write-Host "Ou non eres Luís, ou non eres maior de idade"
}
```

- **Iteración**

```
$numero=1  
while ($numero -lt 20) {  
    Write-Host "Conta: $numero"  
    $numero=$numero + 1  
}
```