

## 1. Introducción

### 1. Los guiones (*scripts*)

### 2. Las variables del Shell

### 3. Paso de argumentos a los guiones

## 1.1. Los guiones (*script*)

- Guión o *script*:
  - Programa o conjunto de órdenes para el shell
  - Permite realizar tareas más complejas que las simples órdenes
- Cómo ejecutar/lanzar un guión:
  - Editar el archivo: guion
  - 1. Cambiando los permisos y haciendolo ejecutable
  - 2. Lanzándolo como parámetro de un shell
  - 3. . <guion> (lanzándolo desde el propio shell)
- # se utiliza como comentario, puede empezar en cualquier columna

## 1.2. Las variables del Shell

- Variable
  - Posición de memoria principal, con nombre, sobre la que se puede leer y escribir

```
$> var1 = "Belarmino"
$> var2 = belar@infor.uva.es
```
  - Nombres de variables:  
[letra\_]{letra|dígito|\_}\*
    - Por defecto son cadenas de caracteres de longitud indefinida e inicialmente nulas
    - Para acceder a su contenido se debe usar:  
\$> \${variable}
    - Para mostrar el contenido de una variable:  
\$> echo \${var}

- Tipos de variables:
  - De entorno
    - Permiten configurar nuestra sesión (/etc/profile ó ~/.profile ó ~/.kshrc, ~/.bashrc, ~/.bash\_profile)
    - Algunas se pueden modificar:  
CDPATH, EDITOR, ENV, HOME, MAIL, MAILCHECK, PATH, PPID, PS1, PS2, PWD, TERM
    - Otras no son modificables:  
\$0, \$1..\$9, \$\*, \$@, \$#, \$\$, \$?, \$!
  - Definidas por el usuario

### Lectura y escritura de variables

- Lectura de la entrada estándar con read  
Read [lista-de-variables]
- También se puede asignar directamente su contenido:  
\$> var1=valor1 var2=valor2 ...
- ``: sustitución de órdenes  
Permite guardar en una variable el resultado de la ejecución de una orden

- **Echo** admite distintos formatos de salida:
  - \b: retroceso
  - \c: imprimir sin avanzar a nueva línea
  - \n: nueva línea
  - \f: salto de página
  - \r: retorno de carro
  - \t: tabulador horizontal
  - \v: tabulador vertical
  - \\: ignorar significado de \
  - \N: carácter ASCII cuyo código octal es N
- Las variables, su contenido y " ":
  - ¿Qué ocurre si no usamos " "?
  - Forma cadenas de caracteres
  - Las variables se pueden visualizar dentro de la cadena "Hola \$nombre"
  - "\${nombre}resto"

- Las comillas simples ' ' y \  
' ': ignora la interpretación del contenido  
\: ignora la interpretación del carácter siguiente  
Puede combinarse con lo anterior
- **Uso de set**
  - nos muestra todas las variables del entorno, incluidas las definidas por el usuario
- ¿Cuál es el alcance de las variables?
  - export [lista-de-nombres]

### 1.3. Paso de argumentos a guiones

Variables asignadas por el shell para paso de parámetros:

- \$0: nombre del propio guión
- \$1..\$9: argumentos en línea de órdenes
- \$\* y @\$: todos los argumentos
- \$#: número de parámetros
- \$? : valor devuelto por el último mandato
- \$\$ : proceso intérprete de los mandatos
- \$! : último proceso enviado en segundo plano
- \$- : opciones usadas por el intérprete

### Paso de argumentos a guiones (cont.)

- **Exit [n]**  
devuelve valores al shell que lo haya llamado:  
0 (éxito), ≠ 0 (fracaso)
- **Shift [n]**  
desplaza los argumentos posicionales [n] posiciones a la izquierda, eliminando los que haya
- **Set [opciones][nueva\_lista\_argumentos]**  
Permite cambiar los valores de los argumentos posicionales