

Universidad de Valladolid – Departamento de Informática

Fundamentos de Informática II

Laboratorio - Programación del Shell de UNIX

Guía de Prácticas

Sesión 1

Curso 2009/2010

1. Scripts sencillos

1. Crear un script que muestre el número de ficheros que tienen permiso de lectura, escritura y ejecución para el *grupo*.
2. Crear un script que muestre el nombre de usuario, número de usuarios conectados al sistema y la fecha actual. El programa debe finalizar con el comando `exit`.
 - Ejecutar el guión en todas las formas estudiadas.
 - Analizar y explicar las diferencias en la ejecución de los scripts.
3. Modificar el anterior script añadiendo una línea después del comando `exit`.
 - Ejecutar el guión en todas las formas estudiadas.
 - ¿Existe alguna diferencia en el funcionamiento del guión?, ¿Por qué?.
4. ¿Cuál es la salida por pantalla al ejecutar este script?

```
#!/bin/bash
preposicion=para
objeto=${preposicion}caidas
echo $objeto
```

5. Realizar un script que lea dos variables y que ilustre todos los casos posibles de uso del comando `echo`: sin uso de comillas, con expansión de datos, sin expansión de datos, con substitución de comandos, utilizando combinaciones de todas las anteriores, etc.
6. ¿Existen otras formas de obtener subcadenas de una variable aparte de las estudiadas? Explicar al menos dos métodos adicionales.

7. Realizar un script que ilustre el uso del comando `echo` con la opción `-e`. Ilustrar todos los casos posibles de uso, en especial la salida que producen los códigos `\007`, `\011` y `\012`. Explicar las salidas producidas en los casos citados.
8. Realizar un script que permita obtener la siguiente información:
 - Fecha de creación del directorio `$HOME`
 - Número de ficheros en el directorio `$HOME`
 - Número de subdirectorios en el directorio `$HOME`
9. Realizar lo siguiente:
 - Crear y ejecutar un guión que muestre el contenido de las variables del entorno mediante el comando `set`.
 - Modificar el guión anterior con las siguientes líneas al principio del mismo: `dir='ls -l'`.
 - Luego de ejecutar el script modificado, comprobar con el comando `set` si es que existe la variable `dir`
 - Crear una nueva sesión del Shell desde la terminal actual.
 - Comprobar si existe la variable `dir` en el nuevo entorno. Cerrar la nueva sesión.
 - Añadir la siguiente línea `export dir` inmediatamente después de definir la variable `dir`.
 - Repetir los pasos del tercero al quinto. Analizar la diferencia.
 - Utilizar el comando `unset` sobre la variable `dir` y repetir los pasos anteriores hasta comprender cómo funciona.
10. Crear un guión que solicite al usuario un nombre de directorio y una cadena no vacía de caracteres (se supone que los datos proporcionados por el usuario no serán erróneos). El guión debe mostrar, solamente, los atributos de protección y el nombre de los archivos que hay en ese directorio y que contienen en su nombre la cadena dada por el usuario.
 - Realizar el script usando variables para guardar los resultados intermedios (usar substitución de comandos).
 - Realizar otra versión del script en la que no se utilicen variables, sino sólo ficheros o la E/S estándar.
 - ¿Qué diferencias hay entre las dos soluciones a nivel del entorno del Shell?
11. Modificar el script anterior para que el valor de la variable `$?` sea mostrado inmediatamente después de realizar una substitución de comandos.
 - ¿Qué valores se observan? ¿Por qué?
 - Ejecutar de nuevo el script e introducir un nombre de directorio no válido. ¿Ahora qué valores se observan para la variable `$??`? ¿Por qué?
12. Realizar un script, cualquiera, cuya última línea de comando sea: `exit 0`

- Ejecutarlo, e inmediatamente después consultar su valor de salida ejecutando en la línea de comandos la orden `echo $?`
- Modificar el script, de modo que la última línea de comando sea: `exit 15`
- Ejecutarlo, e inmediatamente después consultar su valor de salida ejecutando en la línea de comandos la orden `echo $?`
- Explicar las diferencias encontradas en cada uno de estos casos.
- Modificar el script, de modo que ahora la última línea de comando sea: `exit -200`. Ejecutarlo, e inmediatamente después consultar su valor de salida ejecutando en la línea de comandos la orden `echo $?`. ¿Qué valor se observa? ¿Qué puede concluirse de este resultado?