



Preparador Informática

**PRÁCTICA 5
SHELL SCRIPTS**

**OPOSICIONES INFORMÁTICA 2004
ANDALUCÍA**



Concurso-Oposición Cuerpo P.E.S. Año 2004. Informática.

SUPUESTO 2

Apartado A)

Diseñar un shell script para bash que ofrezca en la salida estándar el siguiente menú:

- 1] Listar archivos
- 2] Ver directorio de trabajo
- 3] Crear directorio
- 4] Borrar directorio
- 5] Crear usuario
- 6] Borrar usuario
- 7] Salir

Introducir opción:

Si se solicita la opción:

1	Se listan los archivos del directorio actual.
2	Se muestra el nombre del directorio actual.
3	Se solicita un nombre de directorio y se crea dentro del directorio actual.
4	Se solicita un nombre de directorio y se borra, suponiendo que está vacío y ubicado en el directorio actual.
5	Se comprueba si el script lo está ejecutando el root, y si lo es, se solicita un login de usuario y se añade al sistema, creando en ese momento su directorio personal, que estará ubicado en /home y llevará por nombre el del propio login de usuario. Si no se es root, se mostrará una advertencia.
6	Se comprueba si el script lo está ejecutando el root, y si lo es, se solicita un login de usuario y se borra del sistema, eliminando también su directorio personal. Si no se es root, se mostrará una advertencia.
7	Se finaliza la ejecución, mostrando el mensaje Fin de ejecución .

Si se introduce una opción no válida se indicará esta circunstancia en la salida estándar.

Tras ejecutar cualquiera de las opciones válidas, salvo en el caso de la 7, se imprimirá de nuevo el menú, en espera de recibir otra solicitud.

Diseñar el script empleando funciones shell para cada una de las opciones 1 a 6.

Apartado B)

¿Qué conceptos y comandos básicos necesitarían conocer los alumnos para poder realizar el ejercicio planteado en el apartado A?



SOLUCIÓN PROPUESTA

Apartado A)

```
#!/bin/bash

#Función que muestra el menú del programa
f_menu()
{
    echo -e "\n-----Menu-----"
    echo "1] Listar archivos"
    echo "2] Ver directorio de trabajo"
    echo "3] Crear directorio"
    echo "4] Borrar directorio"
    echo "5] Crear usuario"
    echo "6] Borrar usuario"
    echo "7] Salir"
    read -p "Introduzca opción: " OPC
    echo -e "\n"
}

#Función que lista los archivos del directorio actual
f_listar_archivos(){
    ls
}

#Función que muestra el directorio actual
f_ver_directorio_t(){
    pwd
}

#Función que solicita el nombre de un directorio y lo crea
f_crear_directorio(){
    read -p "Introduzca el nombre del directorio a crear: " DIRECTORIO
    mkdir $DIRECTORIO && echo "El directorio $DIRECTORIO ha sido creado"
}

#Función que solicita el nombre de un directorio y lo borra
f_borrar_directorio(){
    read -p "Introduzca el nombre del directorio a borrar: " DIRECTORIO
    rmdir $DIRECTORIO && echo "El directorio $DIRECTORIO ha sido borrado"
}
```

```
#Función que crea un usuario
f_crear_usuario(){
    if [ $UID -eq 0 ]
    then
        read -p "Introduzca el nombre del usuario a crear: " USUARIO
        useradd -m $USUARIO

        if [ $? -eq 0 ]
        then
            echo "El usuario $USUARIO ha sido creado con éxito"
        fi
    else
        echo "ADVERTENCIA: ¡Hay que ser root para crear usuarios!"
    fi
}

#Función que borra un usuario
f_borrar_usuario(){
    if [ $UID -eq 0 ]
    then
        read -p "Introduzca el nombre del usuario a borrar: " USUARIO
        userdel -r $USUARIO

        if [ $? -eq 0 ]
        then
            echo "El usuario $USUARIO ha sido borrado con éxito"
        fi
    else
        echo "ADVERTENCIA: ¡Hay que ser root para borrar usuarios!"
    fi
}

#PROGRAMA PRINCIPAL
while true
do
    f_menu
    case $OPC in
        1) f_listar_archivos;;
        2) f_ver_directorio_t;;
        3) f_crear_directorio;;
        4) f_borrar_directorio;;
        5) f_crear_usuario;;
        6) f_borrar_usuario;;
        7) echo "Fin de ejecución"
           exit 0;;
        *) echo -e "Opción no válida. Inténtelo de nuevo \n";;
    esac
done
```



Apartado B)

Los comandos básicos que necesitarían conocer son:

- Redireccionamiento de salida y entrada estándar. Flujos (Si los emplea en el código)
- Programación básica del shell bash. Estructuras de control: if, case, while. Uso de funciones
- Comandos: useradd, userdel, ls, pwd, mkdir, rmdir