



Preparador Informática

www.preparadorinformatica.com

PRÁCTICA 9 SHELL SCRIPTS

OPOSICIONES SAI 2018
ARAGÓN

Primera prueba - Parte A (Ejercicio 2)

APARTADO 1 (30 PUNTOS) SHELL-SCRIPT UNIX/LINUX

1.1 Utilizando las estructuras e instrucciones de programación de bourne o bash, crea un shell-script que convierta a minúsculas los nombres de todos los archivos que se encuentren en los directorios que se pasen como argumentos.

- Controlar que el número de parámetros esté comprendido entre 1 y 4 (puede ser 1, 2; 3 o 4).
- Controlar que los parámetros pasados se correspondan con directorios existentes en el lugar donde se ejecuta el script.
- Si los directorios están vacíos no se procesan.
- Si los nombres de los archivos que hay en los directorios ya están en minúsculas no se cambia nada.
- Escenario de pruebas:

```

usuario01@UServer16-01:~$ ls -R DIR?
DIR1:
apuntes Corbata datos jardin nevera toledo
casa cuna fich1 masa PAPELES

DIR2:
adios alaf EQUIPOS menos SISTEMAS

DIR3:

DIR4:
COCHES CubierTOS Vasos

DIR5:
  
```

- Ejemplos de ejecución (no de todos los casos posibles):

```

usuario01@UServer16-01:~$ ./minus.sh
ERROR NUMERO DE PARAMETROS
No puede ser cero,
debe estar comprendido entre 1 y 4
  
```

```

usuario01@UServer16-01:~$ ./minus.sh DIR1 DIR2 DIR3 DIR4 DIR5
ERROR NUMERO DE PARAMETROS
No puede ser superior a cuatro
  
```

```

usuario01@UServer16-01:~$ ./minus.sh DIR1 DIR2 DIR3 DIR6
ERROR PARAMETROS
El directorio DIR6 no existe en este lugar
usuario01@UServer16-01:~$
  
```

```

usuario01@UServer16-01:~$ ./minus.sh DIR1 DIR2 DIR3 DIR4
Procesando directorio DIR1
-----
--- Corbata ha sido renombrado como corbata
--- PAPELES ha sido renombrado como papeles
Procesando directorio DIR2
-----
--- EQUIPOS ha sido renombrado como equipos
--- SISTEMAS ha sido renombrado como sistemas
Procesando directorio DIR4
-----
--- COCHES ha sido renombrado como coches
--- CubierTOS ha sido renombrado como cubiertos
--- Vasos ha sido renombrado como vasos
NUMERO DE DIRECTORIOS PROCESADOS 3
Los directorios vacios no se procesan
  
```

SOLUCIÓN PROPUESTA

```
#!/bin/bash

#Función que recibe como entrada un directorio y lo procesa para convertir
#a minúsculas los nombres de todos los archivos contenidos en él
f_procesaDirectorio(){
    directorio=$1
    echo "Procesando directorio $directorio"
    echo "-----"
    archivos=$(ls $directorio | grep '[A-Z]')
    #Si el comando anterior se ha ejecutado satisfactoriamente ($?=0)
    #significa que hay nombres de archivos con mayúsculas
    if [ $? -eq 0 ]
    then
        #Para cada uno de los archivos qe tienen alguna letra mayúscula

        for archivo in $archivos
        do
            archivoRenombrado=$(echo $archivo | tr 'A-Z' 'a-z')
            mv $directorio/$archivo $directorio/$archivoRenombrado
            echo "--- $archivo ha sido renombrado como $archivoRenombrado"
        done
    fi
}

#Función que recibe como entrada una lista de parámetros
#y comprueba si cada uno de ellos es un directorio
f_compruebaDirectorios(){
    directorios=$*
    num=0

    for directorio in $directorios
    do
        #En caso de que no sea un directorio...
        if [ ! -d $directorio ]
        then
            f_errorParamDir $directorio
        fi
    done

    for directorio in $directorios
    do
        #...se procesa si no está vacío
        if [ `ls $directorio | wc -w` -gt 0 ]
        then
            f_procesaDirectorio $directorio
            num=`expr $num + 1`

            #Si el directorio está vacío
            else
                esVacio=true
            fi
        done
        echo "NÚMERO DE DIRECTORIOS PROCESADOS $num"
        test $esVacio && echo "Los directorios vacíos no se procesan"
    }
}
```

```
#Función que muestra un mensaje de error cuando
#no se ha introducido ningún parámetro
#Finaliza la ejecución con un código de error (1)
f_errorParamMin(){
    echo "ERROR NÚMERO DE PARAMETROS"
    echo "No puede ser cero,"
    echo "debe estar comprendido entre 1 y 4"
    exit 1
}

#Función que muestra un mensaje de error cuando
#se le han introducido un número de parámetro superior a cuatro
#Finaliza la ejecución con código de error (2)
f_errorParamMax(){
    echo "ERROR NÚMERO DE PARAMETROS"
    echo "No puede ser superior a cuatro"
    exit 2
}

#Función que muestra un mensaje de error cuando alguno de los
#parámetros no es un directorio.
#Finaliza la ejecución con un código de error (3)
f_errorParamDir(){
    echo "ERROR PARÁMETROS"
    echo "El directorio $1 no existe en este lugar"
    exit 3
}

#INICIO PROGRAMA PRINCIPAL

case $# in
    0) f_errorParamMin;;

    1|2|3|4) f_compruebaDirectorios $*;;

    *) f_errorParamMax;;
esac
```