

Práctica de Shell Scripting: Análisis y Gestión de Archivos

Ejercicio : Script analizo.sh

Crea un script llamado 'analizo.sh' que reciba como parámetro una palabra y analice cada uno de los caracteres que la componen, mostrando un mensaje indicando si es un número, una letra u otra cosa.

Código Fuente: analizo.sh

```
Pegue aquí el código del script de análisis de caracteres
#!/bin/bash

# Validamos que se haya pasado exactamente un parámetro
if [ $# -ne 1 ]; then
    echo "Error: Debes introducir una palabra como parámetro."
    echo "Uso: $0 <palabra>"
    exit 1
fi

# Guardamos la palabra en una variable
palabra="$1"

# Obtenemos la longitud de la cadena usando expr
longitud=$(expr length "$palabra")

echo "Analizando la cadena: '$palabra' ($longitud caracteres)"
echo "-----"
```

```
# Inicializamos el contador para el bucle
```

```
i=1
```

```
# Recorremos la cadena carácter por carácter
```

```
while [ $i -le $longitud ]; do
```

```
    # Extraemos el carácter en la posición 'i'
```

```
    caracter=$(echo "$palabra" | cut -c $i)
```

```
    # Clasificamos el carácter usando case con expresiones regulares
```

```
    case "$caracter" in
```

```
        [a-zA-Z])
```

```
            echo "Carácter $i: '$caracter' -> Es una LETRA"
```

```
            ;;
```

```
        [0-9])
```

```
            echo "Carácter $i: '$caracter' -> Es un NÚMERO"
```

```
            ;;
```

```
        *)
```

```
            echo "Carácter $i: '$caracter' -> Es OTRO tipo de carácter (símbolo/espacio)"
```

```
            ;;
```

```
    esac
```

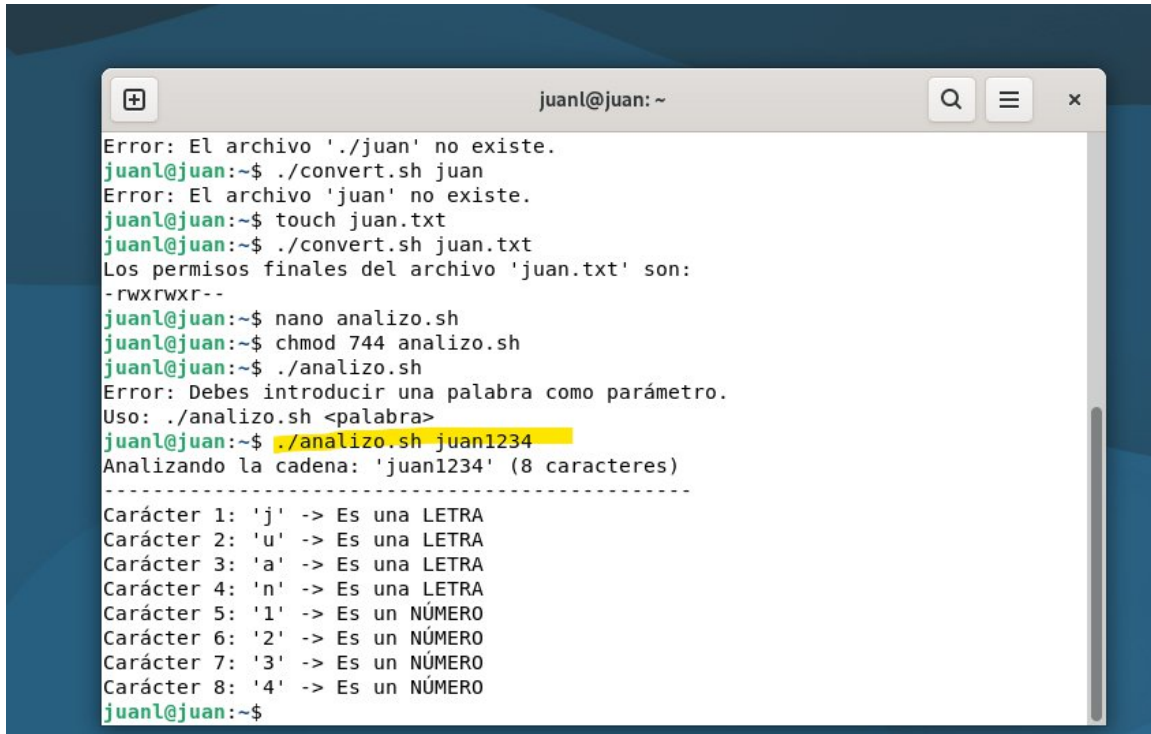
```
    # Incrementamos el contador
```

```
    i=$((i + 1))
```

```
done
```

Prueba de Ejecución: Análisis de Palabra

Inserte captura analizando una cadena que combine letras, números y símbolos (ej: Hola123!)



```

Error: El archivo './juan' no existe.
juanl@juan:~$ ./convert.sh juan
Error: El archivo 'juan' no existe.
juanl@juan:~$ touch juan.txt
juanl@juan:~$ ./convert.sh juan.txt
Los permisos finales del archivo 'juan.txt' son:
-rwxrwxr--
juanl@juan:~$ nano analisis.sh
juanl@juan:~$ chmod 744 analisis.sh
juanl@juan:~$ ./analizo.sh
Error: Debes introducir una palabra como parámetro.
Uso: ./analizo.sh <palabra>
juanl@juan:~$ ./analizo.sh juan1234
Analizando la cadena: 'juan1234' (8 caracteres)
-----
Carácter 1: 'j' -> Es una LETRA
Carácter 2: 'u' -> Es una LETRA
Carácter 3: 'a' -> Es una LETRA
Carácter 4: 'n' -> Es una LETRA
Carácter 5: '1' -> Es un NÚMERO
Carácter 6: '2' -> Es un NÚMERO
Carácter 7: '3' -> Es un NÚMERO
Carácter 8: '4' -> Es un NÚMERO
juanl@juan:~$
```

Ejercicio : Gestión de Archivos por Extensión

Realizar un shell-script que reciba como parámetro una extensión de archivo (sin punto) y el nombre de un directorio (ruta completa). El script preguntará al usuario si desea copiar o mover los archivos del directorio actual con esa extensión al directorio destino. Al finalizar, mostrará el listado de ficheros afectados ordenados alfabéticamente.

Código Fuente: Gestión de Archivos

Pegue aquí el código del script de gestión de archivos
--

```
#!/bin/bash

# Validamos que se hayan pasado los dos parámetros obligatorios
if [ $# -ne 2 ]; then

    echo "Error: Parámetros incorrectos."

    echo "Uso: $0 <extensión_sin_punto> <ruta_directorio_destino>"

    exit 1

fi

# Asignamos los parámetros a variables legibles
extension="$1"
destino="$2"

# Validación técnica: Comprobamos si el segundo parámetro es un directorio válido
if [ ! -d "$destino" ]; then

    echo "Error: El directorio destino '$destino' no existe o no es válido."

    exit 1

fi

# Comprobamos si hay archivos con esa extensión en el directorio actual antes de continuar
# Redirigimos el error por si no encuentra ninguno
```

```
archivos_existentes=$(ls *."$extension" 2>/dev/null)
```

```
if [ -z "$archivos_existentes" ]; then
```

```
    echo "No se encontraron archivos con la extensión .$extension en el directorio actual."
```

```
    exit 0
```

```
fi
```

```
# Preguntamos al usuario la acción a realizar
```

```
echo "¿Qué deseas hacer con los archivos .$extension?"
```

```
echo "1) Copiar (cp)"
```

```
echo "2) Mover (mv)"
```

```
read -p "Introduce tu opción (copiar/mover o 1/2): " opcion
```

```
# Convertimos la respuesta a minúsculas por si acaso
```

```
opcion=$(echo "$opcion" | tr 'A-Z' 'a-z')
```

```
# Procesamos la decisión del usuario
```

```
case "$opcion" in
```

```
    copiar|1)
```

```
        echo "Copiando archivos..."
```

```
        cp *."$extension" "$destino/"
```

```
        accion="Copiados"
```

```
        ;;
```

```
    mover|2)
```

```
        echo "Moviendo archivos..."
```

```
        mv *."$extension" "$destino/"
```

```
    accion="Movidos"

    ;;

*)

    echo "Opción no válida. Abortando script."

    exit 1

    ;;

esac

# Visualización final obligatoria: listado de ficheros afectados ordenados alfabéticamente

echo -e "\n--- Ficheros $accion con éxito (Ordenados alfabéticamente) ---"

ls "$destino"/*."$extension" 2>/dev/null | xargs -n 1 basename | sort
```

Prueba: Acción de Mover/Copiar

Inserte captura donde se vea la pregunta al usuario y el listado final de archivos afectados

```

+
juanl@juan: ~
Carácter 1: 'j' -> Es una LETRA
Carácter 2: 'u' -> Es una LETRA
Carácter 3: 'a' -> Es una LETRA
Carácter 4: 'n' -> Es una LETRA
Carácter 5: '1' -> Es un NÚMERO
Carácter 6: '2' -> Es un NÚMERO
Carácter 7: '3' -> Es un NÚMERO
Carácter 8: '4' -> Es un NÚMERO
juanl@juan:~$ nano gestion.sh
juanl@juan:~$ chmod 744 gestion.sh
juanl@juan:~$ ./gestion.sh
bash: ./: Es un directorio
juanl@juan:~$ ./gestion.sh juan
bash: ./: Es un directorio
juanl@juan:~$ ./gestion.sh juan
Error: Parámetros incorrectos.
Uso: ./gestion.sh <extensión_sin_punto> <ruta_directorio_destino>
juanl@juan:~$ ./gestion.sh juan juan
Error: El directorio destino 'juan' no existe o no es válido.
juanl@juan:~$ ./gestion.sh juan ls
Error: El directorio destino 'ls' no existe o no es válido.
juanl@juan:~$ ./gestion.sh juan home
Error: El directorio destino 'home' no existe o no es válido.
juanl@juan:~$
```