1. **¿Qué es el Shell Scripting? ¿A qué tipos de tareas están orientados los script? ¿Los scripts deben compilarse? ¿Por qué?**

Un Shell Script es un programa que está creado con instrucciones que son ejecutadas por un Shell (CLI o intérprete de comandos) de Unix o Linux. El código no es compilado ni precompilado, se va ejecutando línea por línea efectuando lo que cada instrucción le indica. Necesita un programa que entienda los comandos y estructuras que contiene y esto se suele poner en la primera línea del programa. Por ejemplo #!/bin/bash significa que le pasaremos al BASH las lineas del fichero de Script. Dado que el BASH es el intérprete de comandos más famoso de Linux, los Script que se crean para este entorno también se pueden llamar Bash Script.

Principalmente sirve para automatizar tareas y para realizar procesos más complejos de los que un solo comando puede efectuar. Aunque los comando se pueden enlazar mediante tuberías o XARGS, a veces necesitamos tomar decisiones condicionales o recorrer elementos mediante bucles. Aquí es donde necesitamos organizarlo todo en un Shell Script.

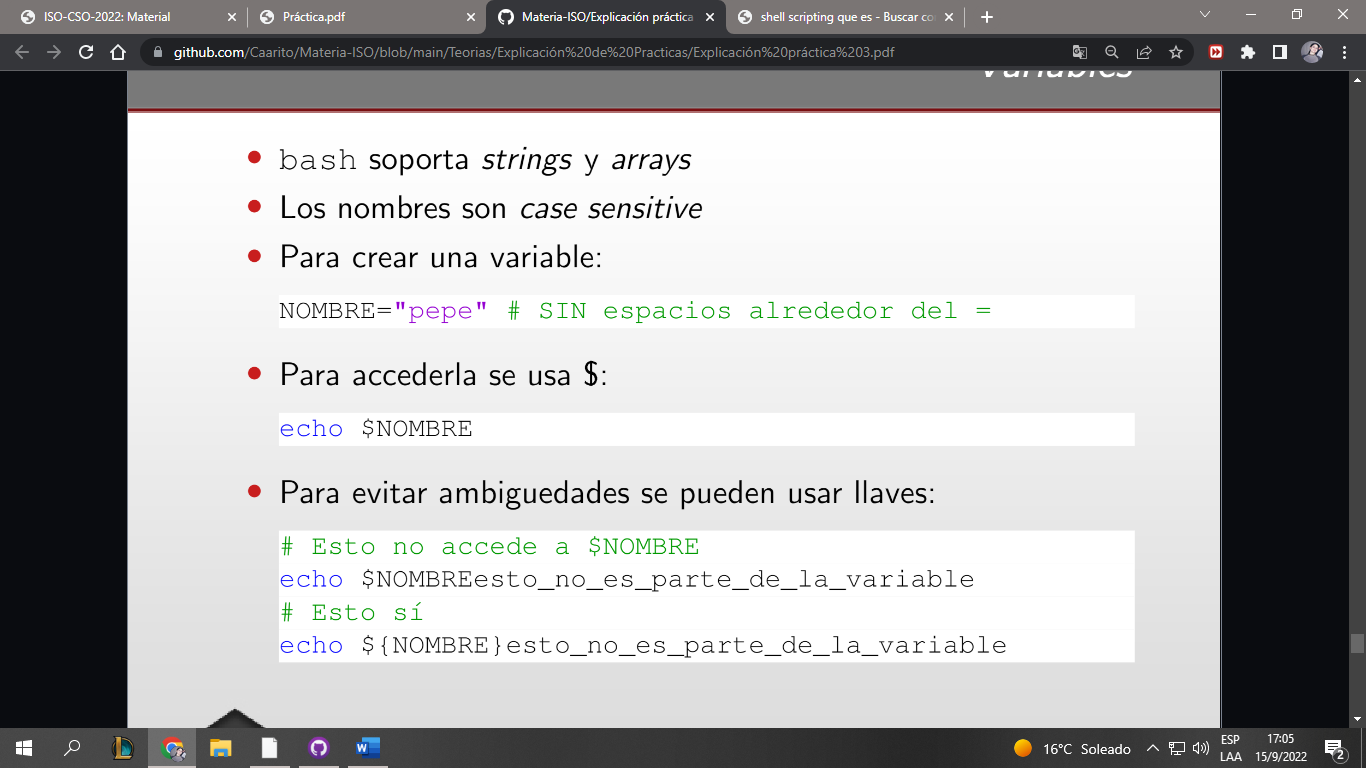
El propio sistema de Linux tiene programadas multitud de tareas con sus Script del sistema, desde la rotación de logs, actualización del arranque, gestión de servidores, niveles de ejecución etc.

1. **Investigar la funcionalidad de los comandos echo y read**

echo: imprime el texto.

read: lee una línea desde entrada estándar.

1. **¿Como se indican los comentarios dentro de un script?** Con #
2. **¿Cómo se declaran y se hace referencia a variables dentro de un script?**



**3. Crear dentro del directorio personal del usuario logueado un directorio llamado practicashell-script y dentro de él un archivo llamado mostrar.sh cuyo contenido sea el siguiente:**

**#!/bin/bash**

**# Comentarios acerca de lo que hace el script**

**# Siempre comento mis scripts, si no hoy lo hago**

**# y mañana ya no me acuerdo de lo que quise hacer**

**echo "Introduzca su nombre y apellido:"**

**read nombre apellido**

**echo "Fecha y hora actual:"**

**date**

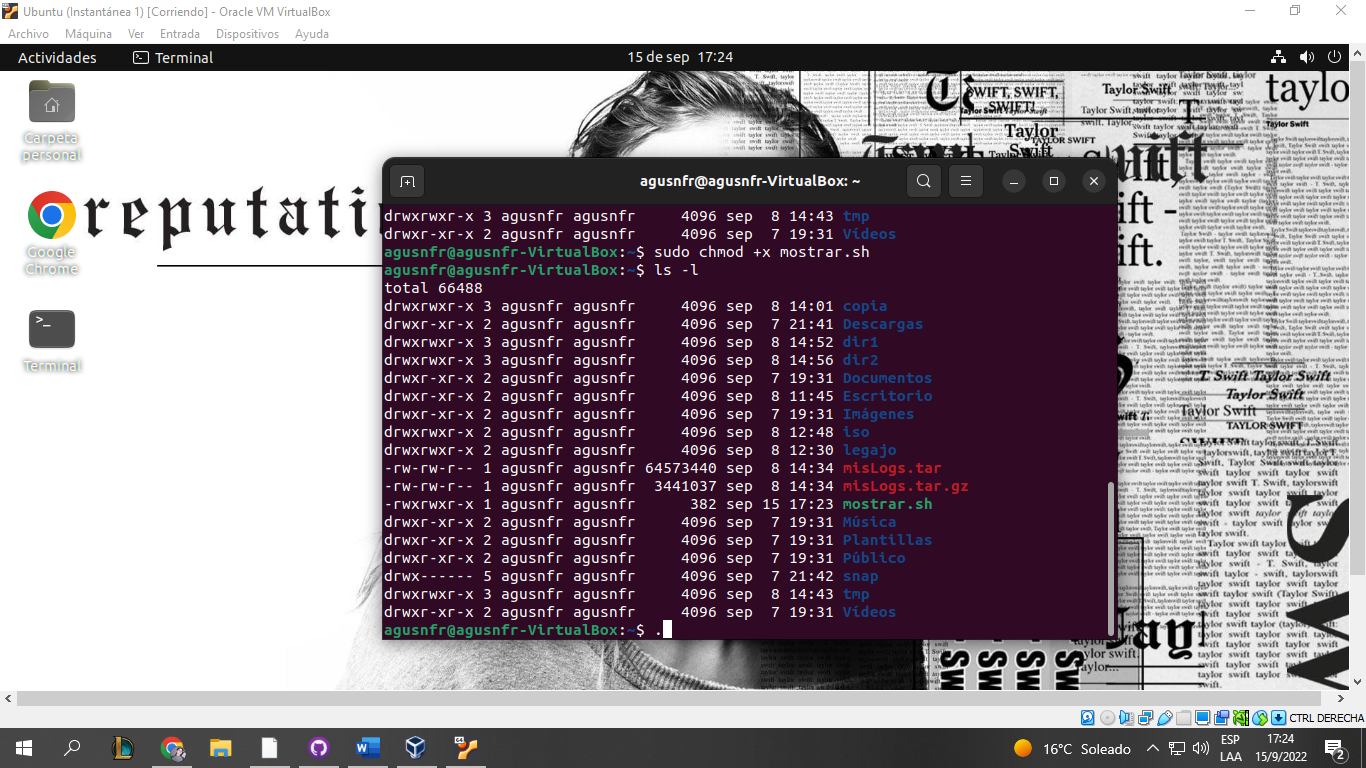
**echo "Su apellido y nombre es: “ 🡪** falta una comilla aca

**echo "$apellido $nombre"**

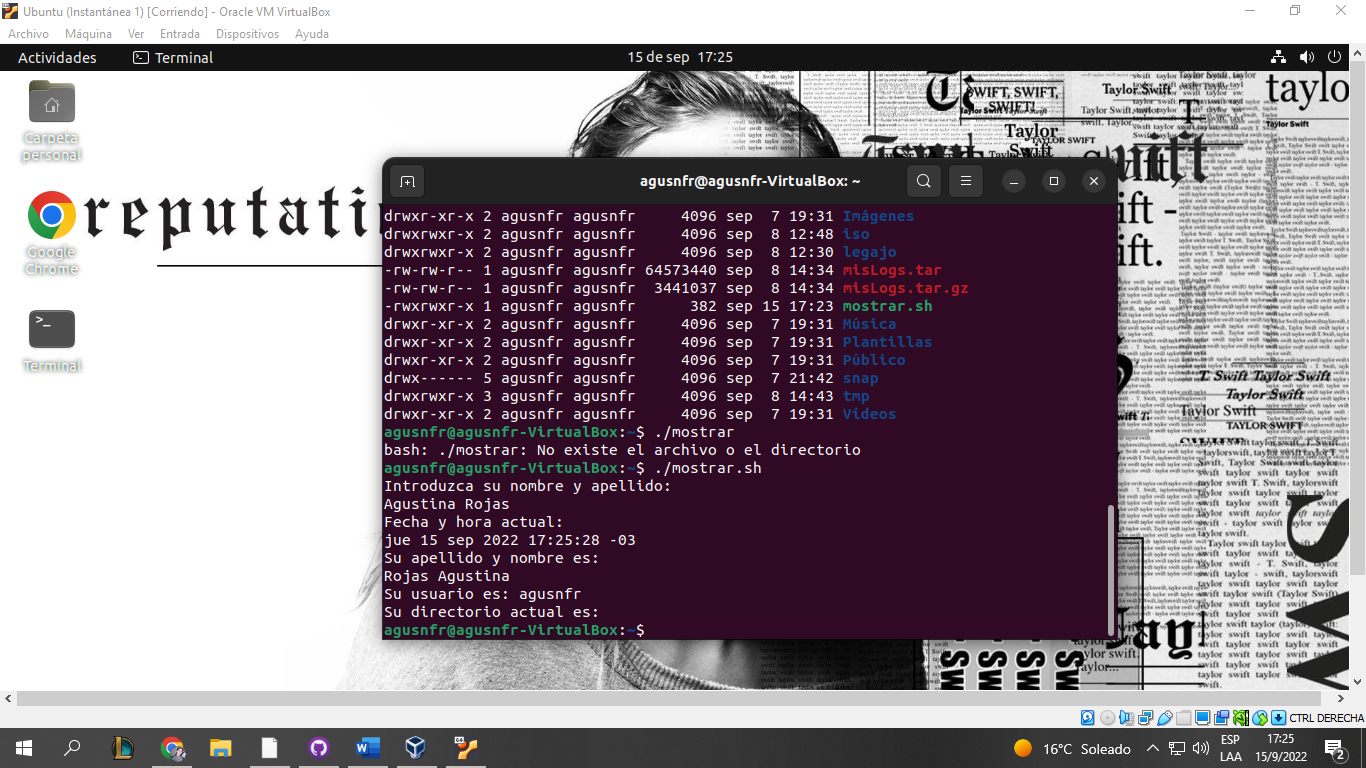
**echo "Su usuario es: `whoami`"**

**echo "Su directorio actual es:"**

1. **Asignar al archivo creado los permisos necesarios de manera que pueda ejecutarlo**



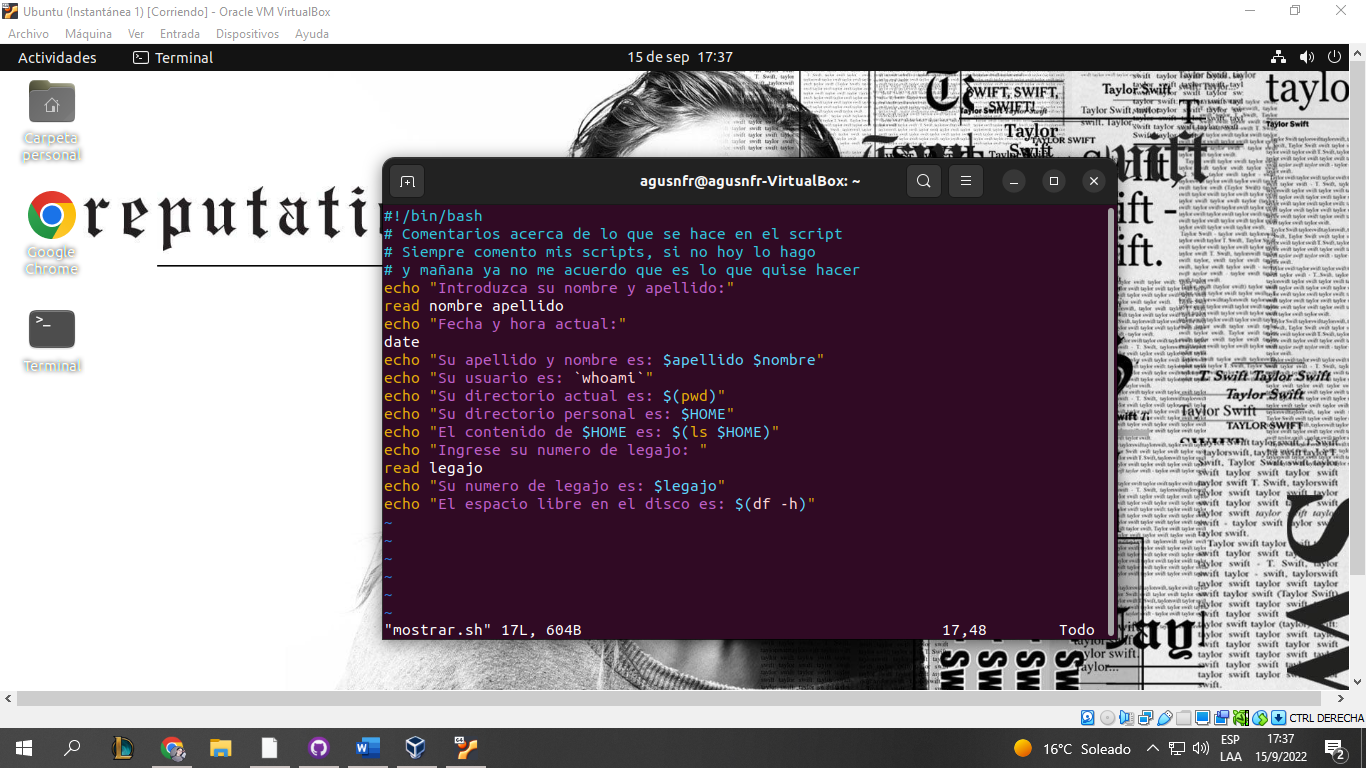
1. **Ejecutar el archivo creado de la siguiente manera: ./mostrar**
2. **¿Qué resultado visualiza?**



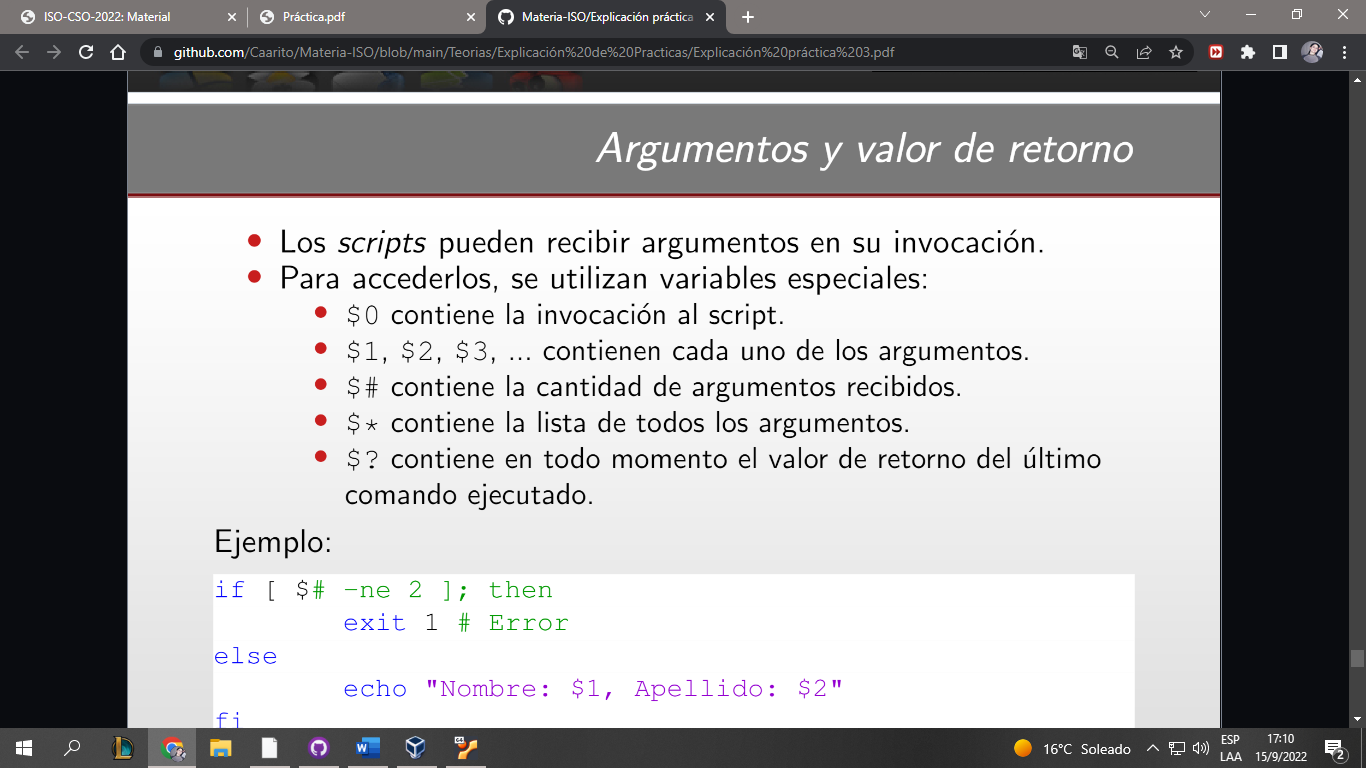
1. **Las backquotes (`) entre el comando whoami ilustran el uso de la sustitución de comandos. ¿Qué significa esto?**

Permite utilizar la salida de un comando como si fuese una cadena de texto normal, es equivalente a hacer $(whoami)

1. **Realizar modificaciones al script anteriormente creado de manera de poder mostrar distintos resultados (cuál es su directorio personal, el contenido de un directorio en particular, el espacio libre en disco, etc.). Pida que se introduzcan por teclado (entrada estándar) otros datos.**



**4. Parametrización: ¿Cómo se acceden a los parámetros enviados al script al momento de su invocación? ¿Qué información contienen las variables $#, $\*, $? Y $HOME dentro de un script?**



$HOME es el directorio de trabajo por defecto del usuario.

**5. ¿Cual es la funcionalidad de comando exit? ¿Qué valores recibe como parámetro y cual es su significado?**

