

Preguntas y respuestas de la práctica 1

Busca la llamada al sistema que se emplea para ejecutar un script y explica su funcionamiento

La llamada al sistema que se emplea comúnmente para ejecutar un script en sistemas operativos basados en Unix (como Linux) es `execve()`, que reemplaza la imagen del proceso actual con un nuevo programa, que puede ser un intérprete de comandos para ejecutar scripts.

¿A qué se debe el error? ¿Cómo podría arreglarse el problema?

A que el script no tiene permisos de ejecución. Se soluciona dándole los permisos.

¿Cuál es la salida del script `segundo_script.sh`? ¿Por qué aparecen 2 líneas en vez de una?

La salida es la siguiente:

hola

mundo

Aparecen las palabras en dos líneas porque, en bash, las palabras en una cadena separadas por espacios se consideran elementos separados en una lista. Entonces, en cada iteración, el bucle `for` toma uno de esos elementos y lo asigna a la variable `i`.

¿Se produce algún error si añadimos un subdirectorio en el directorio actual?

Si, reconoce que es un subdirectorio y no aplica el comando `cat`.

Crea un fichero que tenga un espacio en su nombre y vuelve a ejecutar los scripts (la versión con `"ls"` y la versión con caracteres comodín). ¿Qué ocurre si eliminamos las comillas alrededor de `$i`, en cada caso? ¿Se te ocurre alguna explicación?

Sin comillas, bash interpretará el espacio como un separador de palabras y no tratará adecuadamente el nombre del archivo con espacios.