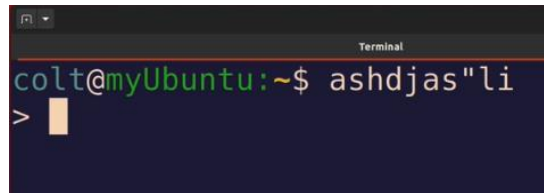


Entendiendo el prompt - Comando en múltiples líneas

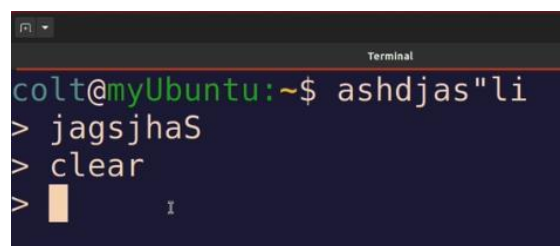
Si por casualidad, cuando estamos escribiendo nuestro comando, escribimos unas comillas dobles y pulsamos “enter”, no vamos a obtener el típico mensaje de error “command not found”. En su lugar, veremos una variación del prompt:



```
colt@myUbuntu:~$ ashdjas"li
>
```

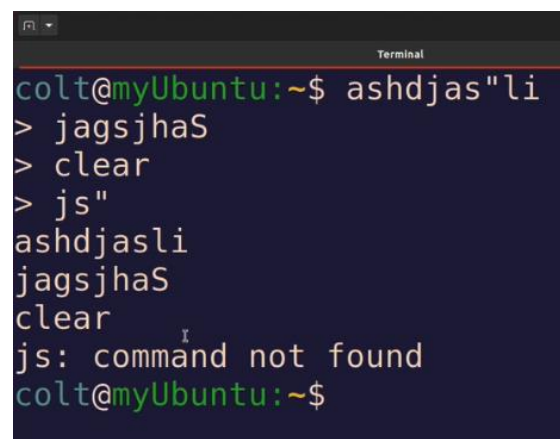
Esta es la forma en que la terminal nos dice que no está considerando lo que escribimos como un nuevo comando. En realidad, piensa que seguimos escribiendo el mismo comando que contiene las comillas dobles, pues las comillas dobles nos habilitan la opción de escribir nuestros comandos en más de una línea. Entonces, el prompt no nos muestra ningún error porque está buscando las correspondientes comillas dobles de cierre.

Entonces, si escribimos algo, por ejemplo, el comando “clear” y damos “enter”, vemos que no sucede nada, y de nuevo, eso es por las comillas dobles:



```
colt@myUbuntu:~$ ashdjas"li
> jagsjhaS
> clear
>
```

Ahora, si cerramos nuestra sentencia escribiendo otras comillas dobles y presionamos “enter”, entonces tomará todo lo que escribimos como una sola sentencia y nos dará el error “command not found”:



```
colt@myUbuntu:~$ ashdjas"li
> jagsjhaS
> clear
> js"
ashdjasli
jagsjhaS
clear
js: command not found
colt@myUbuntu:~$
```