

## 103.2 Procesar flujos de texto usando filtros - Ejercicios

### Ejercicio 3.2.1: Viendo Contenido con **cat** y **less**

- **Objetivo:** Usar **cat** para mostrar archivos y **less** para paginar contenido largo.
- **Desarrollo Paso a Paso:**
  1. Abre una terminal.
  2. **Crea un archivo de prueba:** Ejecuta `echo "línea 1" > archivo_prueba.txt`. Luego `echo "línea 2" >> archivo_prueba.txt`. Luego `echo "línea 3" >> archivo_prueba.txt`.
  3. **Muestra el contenido con **cat**:** Ejecuta `cat archivo_prueba.txt`. Debería mostrar las tres líneas.
  4. **Muestra un archivo largo con **cat** (para ver la limitación):** Ejecuta `cat /etc/services`. El contenido pasará rápidamente.
  5. **Muestra el mismo archivo con **less**:** Ejecuta `less /etc/services`. Ahora puedes desplazarte con las flechas, Page Up/Down. Presiona / y escribe `http` para buscar, luego `n` para la siguiente ocurrencia. Presiona `q` para salir.
  6. **Usa **cat** para leer de **stdin**:** Ejecuta `cat`. Escribe algunas líneas, presiona Enter después de cada una. Presiona `Ctrl+D` (Fin de entrada). `cat` mostrará en `stdout` lo que escribiste en `stdin`.
  7. **Limpiar:** Ejecuta `rm archivo_prueba.txt`.

### Ejercicio 3.2.2: Usando **head** y **tail**

- **Objetivo:** Mostrar las primeras o últimas partes de un archivo.
- **Desarrollo Paso a Paso:**
  1. Abre una terminal.
  2. **Usa **head** en un archivo largo:** Ejecuta `head /etc/services`. Verás las primeras 10 líneas.
  3. **Usa **head** para mostrar un número específico de líneas:** Ejecuta `head -n 5 /etc/services`. Verás las primeras 5 líneas.
  4. **Usa **tail** en el mismo archivo:** Ejecuta `tail /etc/services`. Verás las últimas 10 líneas.
  5. **Usa **tail** para mostrar un número específico de líneas:** Ejecuta `tail -n 3 /etc/services`. Verás las últimas 3 líneas.
  6. **(Opcional, requiere otro proceso que escriba al archivo) Usa **tail -f**:** Abre otra terminal. En la primera terminal, ejecuta `tail -f /var/log/syslog` (o un archivo de log similar que reciba mensajes). En la segunda terminal, haz algo que genere un log (ej: `sudo systemctl restart sshd`). Verás los nuevos mensajes aparecer en la primera terminal. Presiona `Ctrl+C` en la primera terminal para detener `tail -f`.

### Ejercicio 3.2.3: Ordenando y Filtrando Líneas Únicas con **sort** y **uniq**

- **Objetivo:** Ordenar líneas de texto y trabajar con duplicados.
- **Desarrollo Paso a Paso:**
  1. Abre una terminal.
  2. **Crea un archivo con líneas desordenadas y duplicadas:** Ejecuta `echo -e "banana\napple\ncherry\nbanana\ngrape\napple" > frutas.txt`.
  3. **Muestra el contenido:** Ejecuta `cat frutas.txt`.
  4. **Ordena el archivo:** Ejecuta `sort frutas.txt`. Verás la lista ordenada alfabéticamente.
  5. **Ordena en orden inverso:** Ejecuta `sort -r frutas.txt`.
  6. **Usa **uniq** (verás que no elimina todos los duplicados):** Ejecuta `uniq frutas.txt`. El duplicado "banana" adyacente se elimina, pero "apple" duplicado no.
  7. **Combina **sort** y **uniq** para eliminar todos los duplicados (usando pipe - ver 103.4):** Aunque veremos pipes en el siguiente objetivo, puedes ejecutar `sort frutas.txt | uniq`. Esto ordena la salida de `sort` antes de que `uniq` la procese, eliminando ahora *todas* las líneas duplicadas.
  8. **Cuenta las ocurrencias de líneas únicas (después de ordenar):** Ejecuta `sort frutas.txt | uniq -c`. Verás la cuenta de cada fruta única.
  9. **Limpia:** Ejecuta `rm frutas.txt`.

### Ejercicio 3.2.4: Contando Elementos con **wc**

- **Objetivo:** Contar líneas, palabras y caracteres.
- **Desarrollo Paso a Paso:**
  1. Abre una terminal.
  2. **Crea un archivo de prueba:** Ejecuta `echo -e "Hola mundo\nEsto es otra linea." > mi_texto.txt`.
  3. **Cuenta líneas, palabras y bytes del archivo:** Ejecuta `wc mi_texto.txt`. La salida será `2 6 30 mi_texto.txt` (2 líneas, 6 palabras, 30 bytes/caracteres).
  4. **Cuenta solo las líneas:** Ejecuta `wc -l mi_texto.txt`.
  5. **Cuenta solo las palabras:** Ejecuta `wc -w mi_texto.txt`.
  6. **Cuenta solo los bytes/caracteres:** Ejecuta `wc -c mi_texto.txt`.
  7. **Limpia:** Ejecuta `rm mi_texto.txt`.

---

He completado la sección de teoría y ejercicios prácticos