

## 1. Editores

### 1.1. Introducción

Un editor de texto es un programa que permite crear, modificar y guardar un archivo de texto.

Existen 2 tipos de editores:

**Editor de línea:** Rudimentario programa de procesamiento de texto que generalmente se suministra con el sistema operativo como parte de su ambiente de programación. El texto se maneja por líneas por lo cual sólo se puede editar o escribir una línea del texto a la vez. No hay manera de navegar por el texto de una forma fácil.

**Editor de pantalla completa:** Utilería de procesamiento de texto que utiliza las características de la pantalla permitiendo movimientos del cursores y la edición dentro del texto creado.

### 1.2. Editor de línea ed

El primer editor de UNIX llamado ed, es un editor orientado a líneas o editor de líneas. En él las líneas de numeran y se teclean los comandos con base en el número de línea.

Se invoca de la siguiente manera:

```
% ed archivo
```

ed opera en dos modos llamados modo de edición y modo de comandos.

Al estar en el modo de edición todo lo que se teclee se introduce en el espacio temporal de edición.

Cuando se está en modo de comandos, los caracteres que se tecleen se interpretan como comandos.

Si se está en modo de edición, teclear una línea con solo un punto (.) permitirá pasar al modo de comandos.

Comandos básicos ed:

<b>i</b>	insertar
<b>a</b>	agregar
<b>p</b>	imprimir
<b>c</b>	cambiar
<b>d</b>	borrar
<b>w</b>	salvar
<b>q</b>	salir
<b>u</b>	deshacer el comando anterior

### Ejemplo de comandos y funcionamiento:

Suponiendo contar con un archivo que contiene lo siguiente:

El perro es bonito  
Y el gato es lindo  
Lo mejor de todo  
Es que son mios

Llamamos al editor

% ed archivo

 $1, \$p$ 

permite ver todas las líneas del archivo

El perro es bonito  
Y el gato es lindo  
Lo mejor de todo  
Es que son mios

1, 2p

permite ver las líneas 1 y 2 del archivo

El perro es bonito  
Y el gato es lindo

1p

permite ver la línea 1 del archivo y sitúa al editor en la misma línea.

El perro es bonito

2c

Entra al modo edición y permitirá modificar la línea 2

Y la flor es hermosa

•

## Entra al modo comando

1,\$p

Permite ver nuevamente el contenido del archivo

El perro es bonito  
Y la flor es hermosa  
Lo mejor de todo  
Es que son mios

En este caso se observa que cambiar la línea significa describirla.

Para sustituir texto en ed, se utilizan expresiones de la siguiente forma:

s/palabra1/palabra2

Sustituye solo la 1 ocurrencia de palabra1 por palabra2

s/palabra1/palabra2/g Sustituye todas las ocurrencias de palabra1 por palabra2

No debe olvidarse que los comandos de `ed` aplican línea por línea.

Otros comandos:

<code>m</code>	mover
<code>t</code>	copiar
<code>r archivo</code>	insertar archivo

### 1.3. Editor de línea `ex`

Editor de líneas más poderoso que `ed`. Sus siglas significan "editor extendido", es el editor en el cual está basado `vi`.

### 1.4. Editor de pantalla `vi`

El editor de pantalla **`vi`** es una herramienta usada para la creación y modificación de texto, que permite hacer el trabajo de manera sencilla y cómoda. Se puede realizar cualquier modificación al texto mediante la colocación del cursor en el lugar adecuado. Los cambios que se realicen se reflejan inmediatamente en el cuerpo del texto, con lo que es fácil ver como está quedando la información.

#### Modos de trabajo

El editor `vi` al igual que su antecesor `ed` trabaja en dos modos de operación.

**Modo de Comando.** Cuando principia el editor `VI` está en un modo de comandos, eso significa que los caracteres que se tecleen serán interpretados como comandos, para que esos caracteres formen parte del texto se debe pasar al modo de inserción.

**Modo de Inserción.** En este modo se puede añadir o reemplazar texto. Para introducirse al modo de inserción se ejecuta alguno de los siguientes comandos: para añadir (`a,A`, `i,I,o,O`), para reemplazar (`C,s,S` y `R`).

Es importante considerar que si se está en modo de comandos son varios los comandos que permitirán pasar al modo de inserción, pero si se está en modo de inserción, solo existe una forma de pasar al modo de comandos, esta es oprimir la tecla `<Esc>`

Si no está seguro en que modo se encuentra, oprima 2 veces la tecla `<Esc>` lo cual le asegurará estar en el modo de comandos.

#### Comandos de entrada

La forma de invocar al editor se muestra a continuación.

**`$ vi nombre_archivo`**

Se desplegará en la pantalla la primera parte del archivo, si existe, y el cursor se colocará en la primera línea. En caso de que el archivo no exista, se creará automáticamente.

También se puede llamar de las siguientes formas:

**\$ vi archivo1 archivo2 archivo3**

Para este caso se invocan mas de un archivo, el primero que fue llamado se despliega en la pantalla, después mediante la orden ":n", se puede llamar la segundo y así sucesivamente.

**\$ vi + nombre\_archivo**

Se despliega en la pantalla la última parte del archivo y el cursor se coloca en el principio de la última línea.

**\$ vi +n nombre\_archivo**

Esta es una variación de la orden anterior. Aquí "**n**" es un número entero no mayor al número máximo de líneas que contiene el archivo. Después de ejecutar esta orden, el cursor se posesionará en el principio de la línea "**n**".

### Comandos de posicionamiento

Tecla	Significado
<b>l</b>	Un carácter a la derecha
<b>h</b>	Un carácter a la Izquierda
<b>j</b>	Una línea abajo
<b>k</b>	Una línea arriba
<b>w</b>	Primer carácter de la próxima palabra (Derecha)
<b>b</b>	Primer carácter de la próxima palabra (Izquierda)
<b>e</b>	Último carácter de la próxima palabra
<b>G</b>	Principio de la última línea
<b>#G</b>	Va al número de línea indicado
<b>0</b>	Principio de la línea actual
<b>\$</b>	Fin de la Línea actual
<b>CTRL-f</b>	Una pantalla adelante
<b>CTRL-b</b>	Una pantalla atrás

### Comandos de inserción

- **Añadir.**

<b>a</b>	Después del cursor
<b>A</b>	Después de la línea
<b>i</b>	Antes del cursor
<b>I</b>	Antes de la línea
<b>o</b>	Abre una línea después de la actual
<b>O</b>	Abre una línea antes de la actual

- **Copiar.**

<b>#yy</b>	Copia # líneas y las almacena en buffer
<b>#yw</b>	Copia # palabras y las almacena en buffer



- **Mover el texto.**

**#dd** Borra o corta # líneas y las almacena en buffer  
**#dw** Borra o corta # palabras y las almacena en buffer

- **Pegar.**

**p** Inserta el contenido del buffer después de la línea o palabra actual  
**P** Inserta el contenido del buffer antes de la línea o palabra actual

- **Opciones de borrado.**

**#x** Borra # caracteres a la derecha de la posición actual del cursor  
**#X** Borra # caracteres a la izquierda de la posición actual del cursor  
**D** Borra el final de la línea a partir del cursor

- **Sustituir.**

**r** Reemplaza el carácter actual con el próximo carácter  
**R** Reemplaza los caracteres actuales  
**C** Cambia el resto de la línea  
**cw** Cambia la palabra (Con el signo \$ delimita la palabra)  
**cc** Cambia una línea  
**S** Cambia una línea  
**s** Cambia un carácter y entra a modo inserción

- **Comandos especiales.**

**.** Repite el último comando que modificó el buffer  
**u** Cancela el último comando en la línea actual  
**J** Junta la línea próxima al final de la actual  
**/patrón** Busca la primera ocurrencia de patrón a partir de la posición actual  
**n** Busca hacia adelante la siguiente ocurrencia del último patrón buscado  
**N** Busca hacia atrás la siguiente ocurrencia del último patrón buscado

### Opciones avanzadas

El editor vi (editor visual) es básicamente una interfase orientada a pantalla para el editor ex, en consecuencia, al utilizar vi se pueden emplear todos los comandos de ex.

Los comandos ex comienzan con dos puntos (:). Tan pronto como se teclean, vi mueve el cursor a la línea de comandos (renglón inferior de la pantalla). Al terminar de teclear el comando se debe oprimir <retorno>.

- **Comandos de búsqueda.**

**:/patrón** Donde Patrón es la palabra a buscar.  
**:g/patrón/d** Busca la palabra y borra todas las líneas que cumplan el patrón.

- **Reemplazar.**

**:/patrón/s/patrón/texto nuevo**

Donde Patrón es la palabra a buscar que será reemplazada por texto nuevo.

Primera ocurrencia línea actual.

**:g/patrón/s/patrón/texto nuevo/c**

Donde patrón es la palabra que será reemplazada por texto nuevo.

g indica para todas las líneas

c indica que cuando encuentre dicha información pedirá confirmación.

**:s/patrón/texto nuevo/**

Donde Patrón es la palabra a buscar que será reemplazada por texto nuevo.

Primera ocurrencia línea actual.

**:1,\$s/patrón/texto nuevo/gc**

Donde patrón es la palabra que será reemplazada por texto nuevo.

1 indica que a partir de la primera línea

\$ indica hasta la última línea

g indica para todas las ocurrencias en la línea actual

c indica que cuando encuentre dicha información pedirá confirmación

- **Guardar.**

**:w**

Salva y continúa

**:wq**

Salvar y Salir

**:w newfile**

Salva con un nuevo nombre

- **Salir.**

**:q**

Salir sin grabar cambios

**:q!**

Forzar la salida

- **Configuración del editor.**

**:set nu**

Coloca números de líneas

**:set nonu**

Quita los números de línea

**:set showmode**

Permite visualizar si se está en modo de inserción

**:set list**

Visualiza los caracteres utilizados en pantalla

**:set nolist**

Desactiva la visualización de caracteres en pantalla

- **Otros comandos**

**:r**

Lee una copia de otro archivo dentro del actual

**!:comando**

Ejecuta el comando del Shell

**:r !comando**

Inserta la salida de comando en el archivo

**:w >> archivo**

Añade a un archivo existente

**:w! archivo**

Rescribe el archivo