

COPIARE FILE E DIRECTORY

Creiamo il file prova2.txt nella directory prove: **touch prova2.txt**

Fare una copia del file prova1.txt assegnando il nome copia1bis.txt

cp prova1.txt copia1bis.txt

Se vogliamo che il file conservi tutti gli attributi del file sorgente dobbiamo utilizzare l'opzione **-p** (tale opzione potrebbe essere impostata di default).

Esercizio:

- 1) creare nella directory prove un file di nome prova3.txt
- 2) assegnare i diritti di lettura, scrittura ed esecuzione al proprietario, i diritti di lettura e scrittura al gruppo del proprietario, i diritti di sola lettura a tutti gli altri: controllare i permessi del file con il comando appropriato e trascriverli: _____
 forma in bit _____ forma ottale _____
- 3) creare nella stessa directory una copia del file prova3.txt con nome prova3bis.txt: controllare i permessi e confrontarli con quelli del file sorgente; cosa osservi?

Copiare una directory con tutte le sottodirectory e i file in una nuova directory:

- creiamo nella directory prove la sottodirectory sottoprova: **mkdir sottoprova**
- creiamo in questa sottodirectory un file di nome sottofile1.txt
- collocarsi nella directory prove e digitare **cp -R sottoprova sottoprova_bis**
- controlla con **ls -l**: cosa è successo? _____

- esplora il contenuto della directory sottoprova_bis: cosa noti? _____

L'opzione **-R** non copia i file che sono collegamenti simbolici e non mantiene tutti i permessi (a meno che cioè non sia impostato di default); per far ciò si utilizza l'opzione **-a** (archive). Questa opzione è quindi utile per fare una copia di backup di un albero di directory.

Provare questa opzione facendo una copia integrale della directory prove nella directory prove_bis e controllare i permessi di un file interno.

Spostare file e directory: **mv** (move) si utilizza per spostare, o rinominare, un file o una directory in una diversa posizione.

- **mv prova1.txt sottoprova** sposta prova1.txt dalla directory corrente alla directory sottoprova
 - **mv sottoprova_bis sottoprova_ter** cosa fa questo comando?
- _____
- _____

La procedura per rinominare un file è analoga a quella per spostarlo; è sufficiente specificare come argomenti il file da rinominare seguito dal nuovo nome del file.

Esercizio: crea nella directory prove un file di nome *vecchio_nome.txt* e rinominalo in *nuovo_nome.txt*

Rimuovere file e cartelle:

rm file: rimuove file

rm nomedirectory -R: rimuove una directory con tutte le sottodirectory e i file contenuti

rmdir nomedirectory: rimuove una directory vuota

UN METODO SICURO PER RIMUOVERE I FILE

- Provare a cancellare un file con **rm -i** (i=interactive)
- Creare un alias di nome **canc** per il comando **rm-i** e provarlo

Visualizzare le directory in modo ricorsivo

Il comando **ls -R** produce una visualizzazione della directory corrente e di tutte le sue sottodirectory relative. Prova con:

ls -R /home (si può abbinare all'opzione R l'opzione l per ottenere una visualizzazione estesa)

N.B. L'accesso ad alcune directory può essere negato.

Visualizzazione di file con caratteri jolly

*

Esempi:

ls /usr/bin/*tex

ls /usr/bin/tex*

ls /usr/bin/*tex*

ls /usr/bin/t*

?

Questo carattere jolly corrisponde ad un solo carattere.

Esercizio: trova tutti i file della directory /bin il cui nome è composto da 7 caratteri e scrivili nella riga sotto

L'editor vi

vi

è un editor presente su tutti i sistemi Linux, quindi per questo è molto importante.

vi conosce tre modalità operative:

- **Comando**: ogni pressione di un tasto viene interpretato come parte di un comando
- Modalità **inserimento**: ogni pressione di un tasto viene interpretata come inserimento di testo
- Modalità **riga di comando**: per comandi complessi inseriti nell'ultima riga

Per partire. Dopo esserti collocato nella tua home directory scrivi il comando:

vi primofileconvi.txt *crea il file primofileconvi.txt e apre l'editor*

Spaventati? Premete **i** ed entrate nella modalità inserimento testo: scrivete alcune righe di testo.

Per salvare

Premete **ESC** per uscire dalla modalità inserimento testo ed andare in modalità Comandi

premete **:x <invio>** per salvare ed uscire dall'editor.

Aprire un file per leggerlo ed uscire da vi senza salvare:

vi primofileconvi.txt

ESC

:q!

Comandi importanti (dopo aver premuto ESC):

dd	cancella la riga corrente
dw	cancella fino alla fine della parola corrente
:q	esce da vi (se il file non è stato modificato, altrimenti il programma avverte)
:q!	esce da vi senza memorizzare le modifiche
:w <nomefile>	memorizza in <nomefile>
:e <nomefile>	edita (carica) il <nomefile>

N.B.

Per **leggere** velocemente un file c'è un comando molto comodo: **cat <nomefile>**
 >> Provare con questo comando a leggere il contenuto del file creato prima con vi

Esercizio 10

1. Creare nella directory prove con vi un file di nome esercizioconvi1.txt
2. Digitare un testo di almeno 5 righe
3. Uscire da vi memorizzando le modifiche
4. Leggere il file con il comando **cat**
5. Aprire vi
6. Editare alcune righe di testo
7. Salvare il file col nome esercizioconvi2.txt ed uscire da vi
8. Aprire con vi il file esercizioconvi2.txt, fare qualche modifica ed uscire senza salvare le modifiche.

TRASFERIMENTO FILE: FTP (FILE TRANSFER PROTOCOL) VIA SSH

Dalla finestra di SSH apriamo la finestra FTP: **Window>New File Transfer in Current Directory**
 Notiamo le due finestre Local Name (cartelle locali) e Remote Name (cartelle sul server)

Esercizio 2

Aprire SSH Secure File Transfer; aprire il menù *Edit> Setting> Global setting> File Transfer> Advanced* ed analizzare le configurazioni di Default

Esercizio 3

- Creare una cartella di nome *upload* nella propria home (dalla finestra FTP); cliccare di destro sul nome della cartella e cercarne le proprietà; annota i permessi: Permission: _____
 Permission mode: _____
- Aprire Blocco note di Windows e creare il file di testo prova_upload_ftp.txt contenente il seguente testo:

Partendo dalle abilità e conoscenze conseguite dagli allievi nel corso del biennio si potrà ora procedere alla revisione di alcuni concetti che non potevano essere compiutamente acquisiti a causa della giovane età. In particolare si riprenderanno i concetti di velocità e di accelerazione, soffermandosi su quelli di velocità e accelerazione istantanee.

- salvare il file sul proprio Desktop di Windows (nella cartella Documenti)
- Ritornare nella finestra FTP e trasferire il file nella propria home remota nella directory *upload*.
- Aprire il file remoto, aggiungere il testo che segue e salvare in remoto

In tale riassetto sistematico si approfondirà lo studio dei moti(moto circolare, moto armonico o moti su traiettoria curvilinea qualsiasi), con particolare attenzione ai sistemi di riferimento.

- Tornare alla Shell di linux e aprire il file con il comando **cat** per una veloce consultazione.
- Aprire il file con *vi*, aggiungere in fondo al testo il proprio nome e cognome, l'ora e la data attuale; uscire da *vi* salvando le modifiche. Riaprire il file con prima con **cat** e poi dalla finestra FTP con Blocco note.

Unire e spezzare file

- Creare nella propria home la directory <zip> e trasferirvi via FTP i tre file **inferno.txt**, **purgatorio.txt** e **paradiso.txt** che si trovano nella cartella <materiali> della directory della vostra classe.
- aprire il file inferno.txt, numerare le righe e visualizzare una pagina alla volta: dalla directory zip eseguire il comando

cat -n inferno.txt |more

(premendo la barra spaziatrice si avanza di una pagina la volta, premendo <invio> di una riga la volta; q per interrompere).

- Digitare

cat inferno.txt purgatorio.txt paradiso.txt

- cos'è successo? Prova con:

cat inferno.txt purgatorio.txt paradiso.txt |more

oppure con

cat -n inferno.txt purgatorio.txt paradiso.txt |more

Negli esempi precedenti abbiamo visto come concatenare più file e visualizzarli: e se voglio unire più file in un unico file?

- Unire i tre file precedenti nel file divina.txt

cat inferno.txt purgatorio.txt paradiso.txt >divina.txt

- Controllare i file presenti nella directory <zip>
- Si possono unire i file in momenti diversi; questo metodo è anche utile per aggiungere file ad un file già esistente. Dopo aver cancellato il file divina.txt eseguire:

cat inferno.txt >divina.txt

ls -l

cat purgatorio.txt >>divina.txt

ls-l

cat paradiso.txt >>divina.txt

ls -l

Si può spezzare un file in più file di dimensioni predefinite: cercare con *man* informazioni sul comando **split** e spezzare il file divina.txt in tre file di dimensione non superiore a 150 KB:

split -b 150k divina.txt divina

Gestione archivi

zip

- comprimere il file divina.txt in divina.zip

```
zip divina.zip divina.txt
```

- decomprimere:

```
unzip divina.zip
```

attenti a che cosa viene richiesto (se si vuole decomprimere in una specifica directory, per esempio tmp, aggiungere l'opzione -d. Creare la cartella tmp nella propria home e digitare:

```
unzip -d tmp divina.zip
```

- aggiungere un file ad un archivio

```
zip file_esistente.zip file_da_aggiungere
```

Esercizio: Dopo aver creato il file vuoto limbo.txt aggiungerlo al file compresso divina.zip

- per comprimere anche le sottodirectory si usa l'opzione -r

```
zip -r tutto.zip *
```

Esercizio: Comprime il contenuto della directory corrente, comprese le sottodirectory

- Può essere utile visualizzare il nome dei file presenti in un archivio compresso:

```
unzip -l divina.zip
```

- Controllare che i file di un archivio non siano corrotti:

```
unzip -t divina.zip
```

ALTRI FORMATI USATI DA LINUX PER COMPRIMERE I FILE

.bz2 compressore **bzip2** decompressore **bunzip2**

.gz compressore **gzip** decompressore **gunzip**

.tar il programma **tar** crea un archivio con tutte le informazioni necessarie per ricostruire correttamente la gerarchia di directory originale con tutto il suo contenuto. Il file .tar viene poi compresso con gzip. Un file archiviato e compresso in questo modo si presenta nella forma:
archivio.tar.gz

Esercitazione con *tar* e *gzip*

1. Creare nella propria home la directory `<tar>` e copiarvi i tre file `inferno.txt`, `purgatorio.txt` e `paradiso.txt`
2. Creare l'archivio `divina.tar`: dalla directory `tar` lanciare il comando
`tar -cf divina.tar inferno.txt purgatorio.txt paradiso.txt`
3. Controllare il contenuto della directory `tar` e notare la dimensione dell'archivio `divina.tar`: non è stato compresso
4. Comprimiamo l'archivio:
`gzip divina.tar`
Che nome ha il file compresso? _____
Qual è la sua dimensione? _____
Confrontare la dimensione di questo file compresso con il file `divina.zip` compresso nella directory `zip` con il comando `zip`: _____
5. Guardiamo come ha compresso `gzip`:
`gzip -l divina.tar.gz`
6. Cancelliamo dalla directory `tar` tutti i file tranne `divina.tar.gz` (**`rm *.txt`**)
7. Decomprimiamo l'archivio: **`gzip -d divina.tar.gz`**
8. Ora nella directory `tar` dovremmo avere il solo file `divina.tar`
9. Leggiamo il contenuto di un archivio `tar`
`tar -tvf divina.tar`
10. Estraiamo tutti i file da un archivio `tar`: **`tar -xf divina.tar`**
11. Leggere il contenuto della directory `tar` e cancellare i tre file appena estratti (lasciare l'archivio `divina.tar`)
12. Estrarre un file da un archivio: **`tar -xf divina.tar inferno.txt`**
13. `ls -l` per controllare cos'è successo.

Principali opzioni del comando `tar` (leggi il manuale: `man tar`)

- f seguito dal nome dell'archivio per specificare il nome da dare all'archivio
- v (verbose, ossia dettagliato): vengono visualizzati i nomi dei file mentre vengono elaborati
- t legge il contenuto dell'archivio specificato
- c (crea): per creare un archivio
- z comprime l'archivio mentre viene scritto (come fare `gzip` all'archivio `tar`)
- x estrae uno o più file dall'archivio specificato (v. esempi 10 e 12)

N.B. Il nome **tar** deriva da "**tape archive**", vale a dire *archivio su nastro*, poiché in origine era stato concepito per scrivere gli archivi direttamente su un nastro magnetico.