



Principali comandi Ruby on Rails - Riga di comando

Rails, rispetto ad altre distribuzioni, consente di dare comandi attraverso il cosiddetto Terminale che può essere attivato nei seguenti modi:

- Da **Applicazioni > Accessori > Terminale**.
- Premendo contemporaneamente **CTRL + ALT + F1**. A tutto schermo appare il 1° terminale dei sei disponibili (**F1, F2, F3, F4, F5, F6**). Per ritornare a lavorare in modalità grafica premere **CTRL + ALT + F7**.

La necessità di ricorrere al Terminale nasce dal fatto che attraverso la riga di comando è possibile eseguire operazioni molto efficaci, che in certi casi risultano quasi impossibili utilizzando l'interfaccia grafica.

Inoltre è possibile ripetere o correggere un comando dato precedentemente semplicemente richiamandolo con le frecce su o giù da tastiera.

Principali comandi

Un comando può avere una serie di opzioni che solitamente sono indicate con il segno "-":

`<comando> <-opzione1> <-opzione2> ...`

Ogni spazio bianco separa il comando da ogni opzione.

Esempio:

```
$ ls -l
```

help: manuale dei comandi

`<comando> --help` è la guida che spiega la sintassi del comando:

<code>\$ ls --help</code>	E' la guida che spiega la sintassi del comando ls
<code>\$ ls --help more</code>	Consente di visualizzare la guida del comando ls su più pagine video (vedi comando more)

cd: cambio directory

`cd <directory>` consente di cambiare la directory corrente in quella indicata in `<directory>` (change directory):

<code>\$ cd Desktop</code>	Se la directory corrente è home porta nella directory Desktop
<code>\$ cd ..</code>	porta della directory superiore (o padre)
<code>\$ cd /var</code>	in qualunque directory ci si trovi porta nella directory /var

pwd: mostra la directory corrente

`pwd` questo comando ci permette di sapere in che directory ci troviamo.

ls: lista il contenuto di una directory

`ls <opzioni> [<directory>]` visualizza il contenuto di una directory (list):

- a lista anche i file nascosti
- l mostra tutte le informazioni per ogni file (formato lungo)
- k dimensione dei file è in Kb (normalmente è in byte).
- F classifica i file a seconda del tipo.

<code>\$ ls -l</code>	contenuto della directory corrente in formato lungo
<code>\$ ls -a /</code>	contenuto della directory iniziale (root) compresi i file nascosti
<code>\$ ls -lah</code>	contenuto della directory corrente in formato lungo

cp: copia file e directory

`cp` questo comando ci permette di copiare un file o una directory (copy):

- r copia le directory e le sottodirectory ricorsivamente

<code>\$ cp file1 cart1</code>	copia file1 dentro la directory cart1
<code>\$ cp -r cart1 cart2</code>	copia tutta la directory cart1 dentro a cart2

mv: muove o rinomina un file o una directory

`mv` con questo comando è possibile spostare file o directory (move). Può essere anche utilizzato per rinominare un file o una directory.

-i chiede la conferma

\$ mv vecchio nuovo	rinomina il file vecchio in nuovo
\$ mv file1 cart1	sposta il file file1 dentro la directory cart1

rm: rimuove un file o una directory

rm con questo comando è possibile cancellare uno o più file (remove):

-rf cancella ricorsivamente sia le cartelle che il loro contenuto

\$ rm file1 file2	cancella file1 e file2
\$ rm *.*	cancella TUTTO il contenuto della directory corrente. Le eventuali directory presenti non saranno rimosse. Da usare con cautela !
\$ rm -rf cart1	cancella tutto il contenuto della directory cart1

rmdir: rimuove una directory vuota

rmdir con questo comando è possibile cancellare una o più directory vuote (remove directory)

\$ rmdir prova	cancella la directory vuota prova
-----------------------	-----------------------------------

mkdir: crea una directory

mkdir con questo comando è possibile creare una directory (make directory)

\$ mkdir prova	crea la directory prova
-----------------------	-------------------------

pwd: mostra la directory corrente

pwd con questo comando è possibile sapere in quale directory ci troviamo (print working directory)

\$ pwd	Se l'utente EComunication si trova nella cartella principale visualizza /home/Abaco
---------------	---

chmod: modificare i permessi di un file

chown con questo comando è possibile modificare il permessi di un file.

Il parametro -R consente di modificare ricorsivamente i permessi delle directory indicate e del loro contenuto.

\$ chmod 0755 file1	imposta lettura, scrittura ed esecuzione per il proprietario, e lettura ed esecuzione per gruppo ed altri per file1
\$ sudo chmod -R 0755 dir1	imposta lettura, scrittura ed esecuzione per il proprietario, e lettura ed esecuzione per gruppo ed altri per la directory di 1 e per i file contenuti

chown: modificare il proprietario di un file

chown con questo comando è possibile modificare il proprietario di un file.

Il parametro -R consente di modificare ricorsivamente i permessi delle directory indicate e del loro contenuto.

\$ chown nome1 file1	rende il file1 proprietario di nome1
\$ chown nome1:gruppo1 file1	rende il file1 proprietario di nome1 e del gruppo1
\$ chown -R nome1:gruppo1 dir1	rende la directory dir1 e i file contenuti proprietari di nome1 e del gruppo1
\$ sudo chown -R root:root drupal	rende la directory drupal e i file contenuti proprietari di root e del root

cat: visualizza il contenuto di uno o più file

cat con questo comando è possibile visualizzare il contenuto di uno o più files

\$ cat file1	Visualizza il contenuto del file1
\$ cat file1 file2 > file3	crea file3 con il contenuto di file1 e file2

more: visualizza il contenuto di un file su più pagine video

cat con questo comando è possibile visualizzare il contenuto di un file in più pagine video. Il tasto di Invio fa avanzare la visualizzazione riga per riga mentre la barra spaziatrice fa avanzare di pagine video.

Per interrompere utilizzare CTRL+Z

\$ more file1	Visualizza il contenuto del file1 in pagine video
\$ ls -l more	visualizza il contenuto della directory corrente in formato lungo su più pagine video (vedi comando ls)