

MANUALE TERMINALE LINUX

Realizzato da Jona Lelmi

Nickname PyLinx

Iniziato il giorno 5 giugno 2010 - terminato il giorno 16 giugno 2010

email autore: jona.jona@ymail.com

sito internet: <http://www.tanadelpitone.tk/>

forum: <http://tanadelpitone.forumup.it/>

PREMESSA

L'obiettivo di questo manuale è quello di illustrare le principali istruzioni di un terminale Linux, molti crederanno che sia inutile perché c'è già l'interfaccia grafica ma quando avrete imparato non ve ne staccherete più dato che le operazioni sono più veloci e funzionali.

Se siete utenti Windows è possibile che non sappiate nemmeno cosa sia un terminale dato che il sistema Microsoft è tutto un punta e clicca: il terminale Linux può essere paragonato alla CMD di Windows.

Oltre ad avere il vantaggio di essere più veloce, saper usare il terminale vuol dire poter risolvere determinati problemi (ad esempio con l'interfaccia grafica) e saper installare un sistema GNU/Linux anche testualmente (Da console).

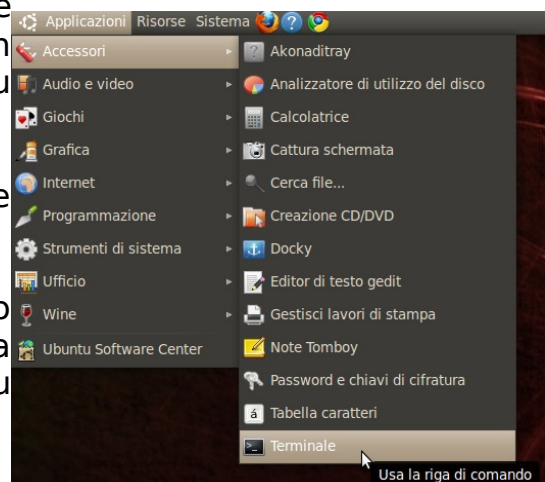
In questo manuale non spiegherò le istruzioni più complesse, ma solo quelle che possono realmente servire.

IL TERMINALE

Molte istruzioni che illustrerò saranno utilizzabili in tutte le distribuzioni Linux, mentre altre saranno utilizzabili solo su Linux/Ubuntu o su distribuzioni basate su Debian perciò scriverò in **VERDE** quelle che possono essere utilizzate solo su Ubuntu e in **ROSSO** quelle che possono essere utilizzate su qualsiasi sistema Linux.

Negli screenshots in cui spiegherò alcune istruzioni userò il terminale Ubuntu.

Adesso però basta con i preamboli e vediamo subito come aprire il terminale su Ubuntu: una volta effettuato l'accesso andate su **Applicazioni>Accessori>Terminale**



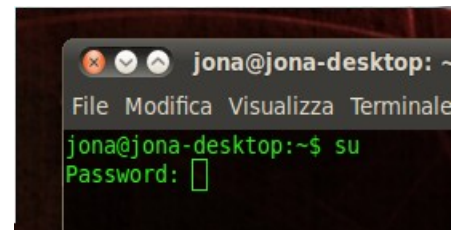
Benissimo, adesso avete di fronte il terminale di Ubuntu e siete pronti a digitare i primi comandi.

I PRIMI COMANDI-L'AMMINISTRAZIONE

I comandi di amministrazione sono molto importanti perché permettono di **risolvere una miriade di problemi**, ma vanno usati con cautela e solo quando realmente servono perché, se usati in modo errato, **possono causare danni al sistema**.

Alcuni di questi comandi possono essere eseguiti solo come utente root (cioè l'utente che è amministratore di sistema e che ha tutti i permessi) quindi il primo comando che impareremo è quello che serve ad ottenere i privilegi di root:

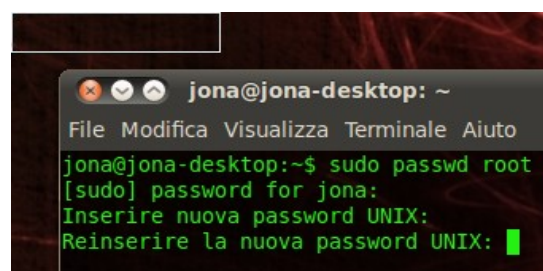
su è questo il comando che serve per diventare **root**, dopo aver premuto invio verrà chiesta una password che non è uguale (inizialmente) a quella impostata per il vostro account.



```
jona@jona-desktop: ~  
File Modifica Visualizza Terminale  
jona@jona-desktop:~$ su  
Password: █
```

Se conoscete la password digitatela e premete invio nel caso non vengano dati errori siete diventati utenti **root**, altrimenti dovete reimpostare la password con il comando:

sudo passwd root quando digitate questo comando nel terminale verrà chiesto di inserire la vostra **password per i comandi sudo** (cioè la password dell'utente) e la **nuova password root** per due volte, fatelo e se non riscontrate errori quella sarà la vostra password root.



```
jona@jona-desktop: ~  
File Modifica Visualizza Terminale Aiuto  
jona@jona-desktop:~$ sudo passwd root  
[sudo] password for jona:  
Inserire nuova password UNIX:  
Reinserire la nuova password UNIX: █
```

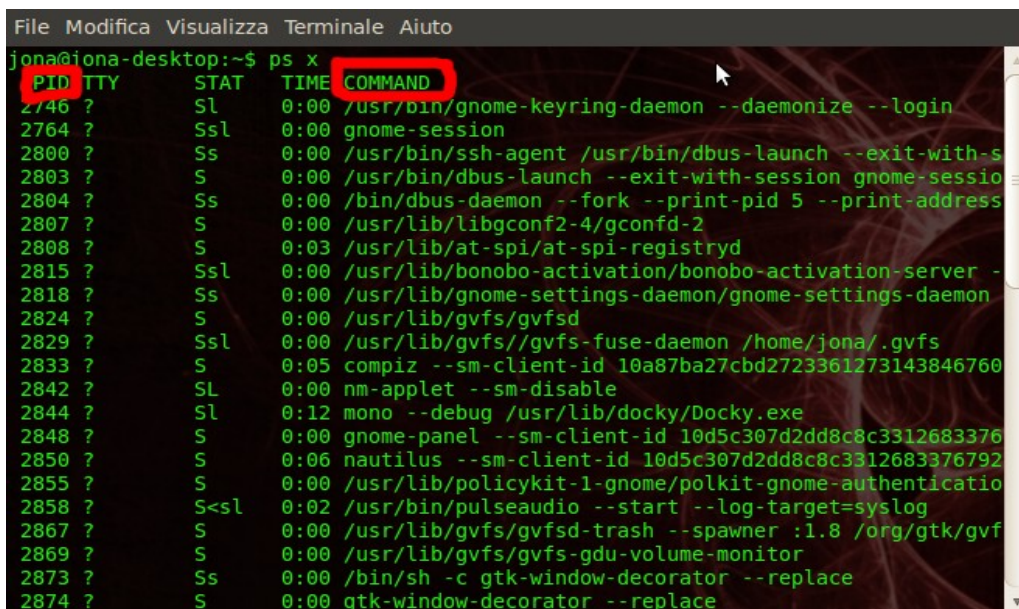
Adesso potete riprovare il comando **su** ed inserite la password appena impostata. Da questo momento in poi è possibile eseguire qualsiasi tipo di operazione quindi siate cauti!

-Processi.

La prima operazione amministrativa che andremo ad esaminare è quella di **terminare un processo**, in Windows eravamo abituati (eh si, succedeva sempre) quando un programma si piantava, a digitare CTRL+ALT+CANC e dalla finestra che appariva terminare il processo. Questa operazione è molto utile ed è possibile fare una cosa simile anche con Linux, ma se fosse proprio l'intera interfaccia grafica a piantarsi? Non c'è problema, è possibile terminare un processo dal terminale.

Per farlo digitiamo da terminale:

ps x che elenca tutti i processi in esecuzione sul computer, verranno disposti in 5 colonne, per adesso ignorate la seconda la terza e la quarta e puntate invece gli occhi sulla **prima** e sull'**ultima**: nella prima sono elencati i numeri dei processi, nella seconda sono elencati i nomi dei processi in esecuzione.



```
File Modifica Visualizza Terminale Aiuto
jona@iona-desktop:~$ ps x
  PID TTY          STAT       TIME COMMAND
 2746 ?            Sl      0:00   /usr/bin/gnome-keyring-daemon --daemonize --login
 2764 ?            Ss      0:00   gnome-session
 2800 ?            Ss      0:00   /usr/bin/ssh-agent /usr/bin/dbus-launch --exit-with-s
 2803 ?            S       0:00   /usr/bin/dbus-launch --exit-with-session gnome-sessio
 2804 ?            Ss      0:00   /bin/dbus-daemon --fork --print-pid 5 --print-address
 2807 ?            S       0:00   /usr/lib/libgconf2-4/gconfd-2
 2808 ?            S       0:03   /usr/lib/at-spi/at-spi-registryd
 2815 ?            Ss      0:00   /usr/lib/bonobo-activation/bonobo-activation-server -
 2818 ?            Ss      0:00   /usr/lib/gnome-settings-daemon/gnome-settings-daemon
 2824 ?            S       0:00   /usr/lib/gvfs/gvfsd
 2829 ?            Ss      0:00   /usr/lib/gvfs/gvfs-fuse-daemon /home/jona/.gvfs
 2833 ?            S       0:05   compiz --sm-client-id 10a87ba27cbd2723361273143846760
 2842 ?            SL      0:00   nm-applet --sm-disable
 2844 ?            SL      0:12   mono --debug /usr/lib/docky/Docky.exe
 2848 ?            S       0:00   gnome-panel --sm-client-id 10d5c307d2dd8c8c3312683376
 2850 ?            S       0:06   nautilus --sm-client-id 10d5c307d2dd8c8c3312683376792
 2855 ?            S       0:00   /usr/lib/policykit-1-gnome/polkit-gnome-authenticatio
 2858 ?            S<sl    0:02   /usr/bin/pulseaudio --start --log-target=syslog
 2867 ?            S       0:00   /usr/lib/gvfs/gvfsd-trash --spawner :1.8 /org/gtk/gvf
 2869 ?            S       0:00   /usr/lib/gvfs/gvfs-gdu-volume-monitor
 2873 ?            Ss      0:00   /bin/sh -c gtk-window-decorator --replace
 2874 ?            S       0:00   gtk-window-decorator --replace
```

Ora supponiamo che, per una ragione qualsiasi, io voglia terminare il processo relativo alla decorazione delle finestre (cioè quel processo che dispone sopra le finestre la barra del titolo e i tre pulsanti per chiudere, massimizzare e minimizzare le finestre) , per farlo devo cercare il processo nella quinta colonna eccolo:

```
2873 ?      Ss      0:00 /bin/sh -c gtk-window-decorator --replace
```

e dopo averlo trovato devo digitare **kill** seguito dal numero del processo (nella prima colonna, quindi in questo caso 2873), quindi l'istruzione sarà **kill 2873** infatti l'istruzione:

kill NumeroProcesso ferma il processo corrispondente a quel numero.

Esiste anche un altro metodo ma è più brutale:

killall NomeProgramma Questa istruzione **elimina tutti i processi** relativi al nome del programma espresso, è preferibile utilizzare il precedente comando che elimina solo il processo scelto, consiglio questo secondo metodo solo se non riusciamo a trovare il processo che ci serve nell'elenco.

N.B. Provate l'istruzione su programmi **inutili** per il corretto funzionamento del sistema e se proprio dovete provare sul decoratore di finestre (Ve lo sconsiglio) per poter far ripartire l'applicazione è possibile digitare da terminale:

gtk-window-decorator --replace se usate **GTK** come decoratore di finestre

emerald-window-decorator --replace se usate **Emerald**.

-Spengere e riavviare il pc.

Spegnere il pc è un'operazione semplice e mai ci verrebbe in mente di farlo da terminale, perché complicarsi la vita? Vi devo dare ragione, ma queste istruzioni sono utili per coloro che vorranno, un giorno, installare un sistema Linux da terminale, ma, se volete, potete tranquillamente spegnere il pc così:

shutdown -h now Questo comando spegne il pc.

reboot questo comando riavvia il pc.

FINE PARAGRAFO.

Per approfondire: è possibile visualizzare in una forma ad “albero” i processi in esecuzione tramite l'istruzione **pstree**.

CARTELLE

I comandi che ci permettono di creare, eliminare, modificare cartelle sono indispensabili e, forse, sono i comandi che si usano maggiormente.

Questi comandi non sono molto usati in sé per sé, cioè non vengono usati per salvare un file dentro una cartella, bensì permettono di portare a termine altre operazioni importantissime per il sistema, quindi vediamo i principali comandi.

Il primo comando che esaminiamo è un'istruzione che io uso quasi tutti i giorni:

ls.

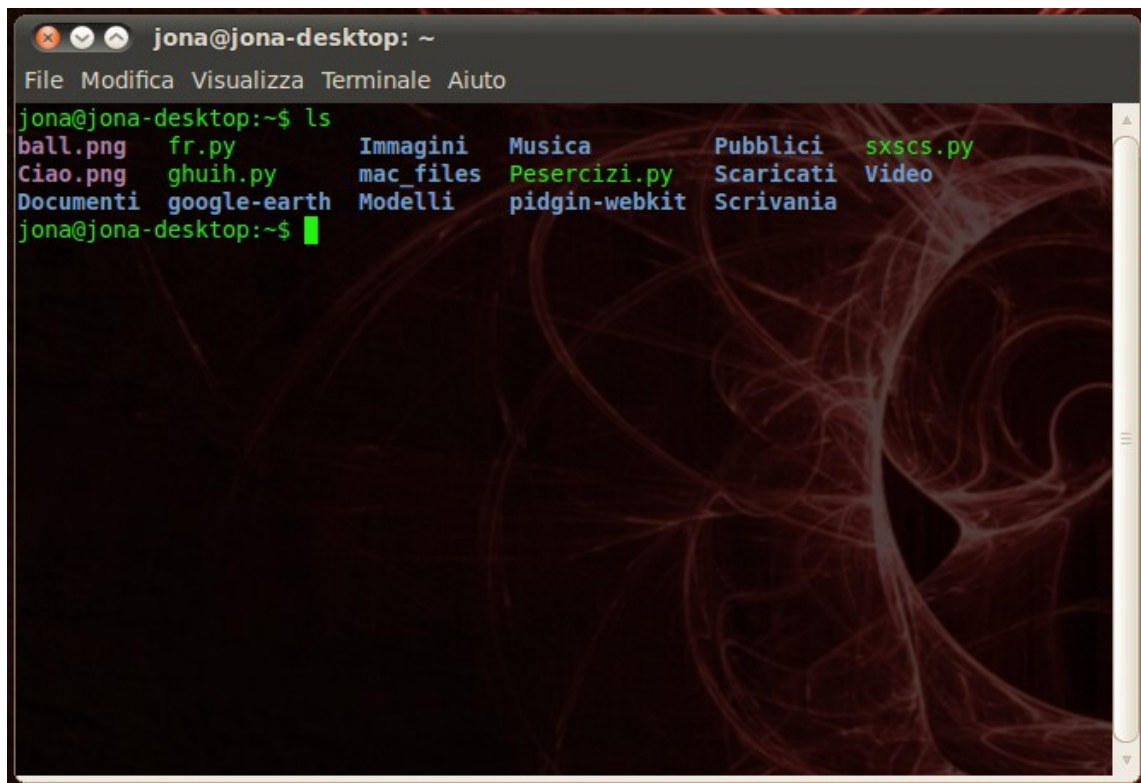
ls questo comando permette di far **elencare sul terminale i file e le cartelle** presenti nella directory (cartella in inglese) in cui ci troviamo.

Impariamo subito un'altra istruzione e facciamo poi un esempio pratico. Di default quando accediamo a un sistema Linux, se apriamo un terminale siamo nella directory **/home**, ma se io volessi vedere il contenuto della cartella Documenti? Prima devo spostarmi all'interno di essa e poi posso usare **ls**.

Per spostarsi in un'altra cartella è possibile utilizzare il comando **cd** seguito dal **nome della cartella** a cui vogliamo accedere.

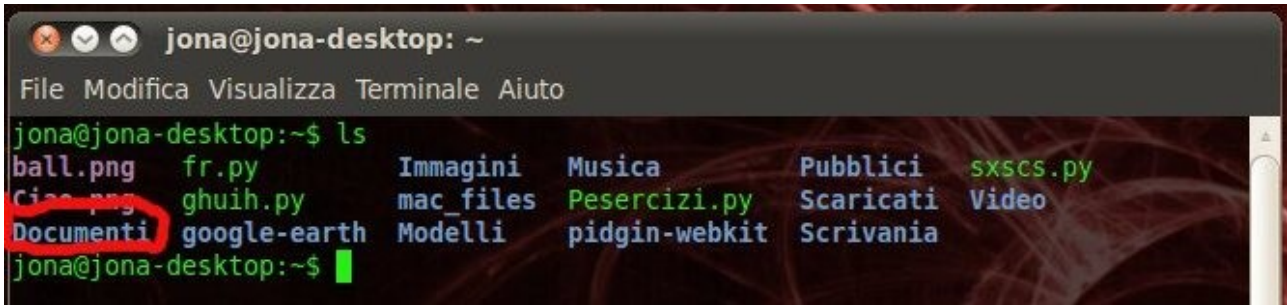
Supponiamo di voler vedere il contenuto della cartella Documenti, come facciamo?

Apriamo il terminale e digitiamo **ls**, vedrete che saranno elencate le cartelle presenti in **/home**.



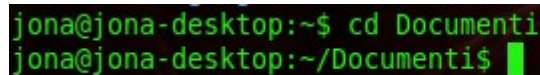
```
jona@jona-desktop: ~  
File Modifica Visualizza Terminale Aiuto  
jona@jona-desktop:~$ ls  
ball.png    fr.py       Immagini   Musica     Pubblici   sxscs.py  
Ciao.png   ghuih.py   mac_files  Pesercizi.py Scaricati  Video  
Documenti  google-earth Modelli    pidgin-webkit Scrivania  
jona@jona-desktop:~$ █
```


Tra queste vedo la cartella **Documenti**, eccola: (le cartelle sono quelle in blu)



```
jona@jona-desktop: ~  
File Modifica Visualizza Terminale Aiuto  
jona@jona-desktop:~$ ls  
ball.png  fr.py      Immagini  Musica     Pubblici  sxscs.py  
Ciao.png ghuih.py   mac_files Pesercizi.py Scaricati Video  
Documenti google-earth Modelli   pidgin-webkit Scrivania  
jona@jona-desktop:~$
```

Adesso che l'ho trovata posso digitare il comando **cd Documenti**, vedrete che, dopo aver digitato questo comando al posto del \$ ci sarà la scritta **/Documenti**.



```
jona@jona-desktop:~$ cd Documenti  
jona@jona-desktop:~/Documenti$
```

E finalmente adesso possiamo vedere il contenuto della cartella Documenti, in che modo? Semplicemente digitando il comando **ls**. State diventando bravi!

Se volete ritornare alla cartella /home è possibile digitare **cd** e tutto tornerà come prima.

-Creare, copiare ed eliminare cartelle.

Le istruzioni che mi accingo a spiegare si utilizzano meno delle precedenti, tuttavia è bene saperle perché molte volte si possono dimostrare utili. Sapete tutti cosa vuol dire copiare, eliminare e creare cartelle? Lo spiego lo stesso :) :

-Creare una cartella vuol dire fare una cartella nuova, con un nome a vostra scelta, in cui potrete inserire altri file e cartelle;

-Copiare una cartella significa creare una cartella con lo stesso contenuto di un'altra;

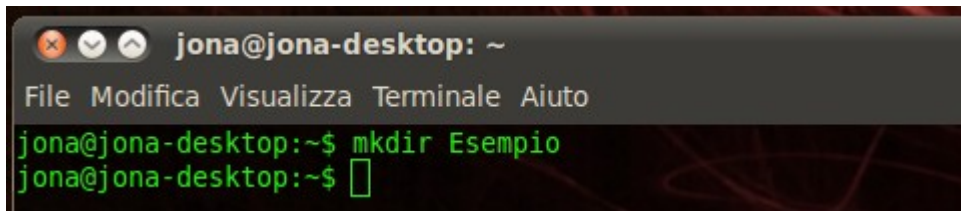
-Eliminare una cartella vuol dire gettarla nel cestino insieme a tutto il suo contenuto.

N.B. Le operazioni che spiegherò adesso **non** le fate da **utente root** perché potreste eliminare qualcosa di estremamente importante per il **sistema**.

Adesso vedremo, con un esempio pratico, come fare le operazioni elencate sopra. Innanzitutto creiamo una cartella, possiamo farlo mediante l'istruzione:

mkdir NomeCartella questo comando genera nella directory corrente una cartella con il nome che inseriamo subito dopo il comando. Supponiamo di

voler creare una cartella di nome Esempio, utilizzeremo **mkdir Esempio**:



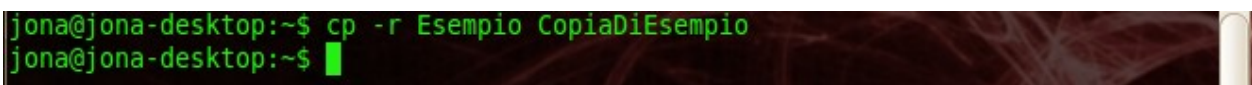
```
jona@jona-desktop: ~  
File Modifica Visualizza Terminale Aiuto  
jona@jona-desktop:~$ mkdir Esempio  
jona@jona-desktop:~$
```

Come vedete non viene data la conferma dell'avvenuta operazione, ma se non ci sono errori tutto è andato liscio e se proprio volete essere sicuri che tutto sia andato per il meglio digitate **ls** e guardate se c'è la cartella Esempio tra i risultati.

Adesso possiamo provare a fare una copia della cartella **Esempio**, il comando è molto semplice, nel nostro caso utilizziamo come Directory da copiare **Esempio** e la copieremo nella cartella **CopiaDiEsempio** (Questa seconda cartella non deve essere creata con mkdir):

cp -r Esempio CopiaDiEsempio il comando copia la cartella Esempio in CopiaDiEsempio.

Ecco il risultato dell'operazione, anche in questo caso potete verificare con il comando **ls** che sia stata creata la cartella **CopiaDiEsempio**.



```
jona@jona-desktop:~$ cp -r Esempio CopiaDiEsempio  
jona@jona-desktop:~$
```

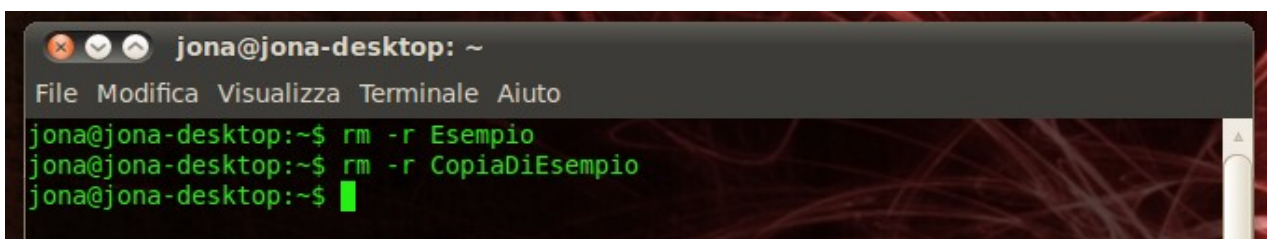
Perfetto! Ma adesso che le Cartelle non sono più utili possiamo eliminarle, esistono due istruzioni per fare ciò:

rmdir Cartella questo comando elimina la Cartella solo se è vuota.

rm -r Cartella questo comando elimina la cartella indipendentemente dal suo contenuto.

Nel nostro caso potremmo perciò utilizzare tutti e due i comandi, tuttavia è più importante imparare il secondo comando, quindi digitiamo:

rm -r Esempio e premiamo invio, poi digitiamo **rm -r CopiaDiEsempio** e premiamo invio:



```
jona@jona-desktop: ~  
File Modifica Visualizza Terminale Aiuto  
jona@jona-desktop:~$ rm -r Esempio  
jona@jona-desktop:~$ rm -r CopiaDiEsempio  
jona@jona-desktop:~$
```

Se non avete riscontrato errori avete eliminato le cartelle, l'errore che appare più frequentemente con questo tipo di comando è **rm:impossibile rimuovere 'NomeCartella':nessun file o directory**.

Questo succede quando si tenta di eliminare una cartella non esistente, quindi attenti a cosa scrivete!

FINE PARAGRAFO.

Per approfondire:

-Il comando **ls** presenta degli attributi, il più utile è:

ls -l che dà una lista di file e cartelle con **data e ora della creazione**.

-Si possono anche muovere cartelle con il comando **mv Cartella /CartellaInCuiMuovere**, nel nostro caso potevamo usare:

mv Esempio /Documenti per spostare la cartella **Esempio** dalla home in **Documenti**.

-Si possono eliminare più cartelle contemporaneamente, noi potevamo usare:

rm -r Esempio CopiaDiEsempio per eliminare **le due cartelle insieme**.

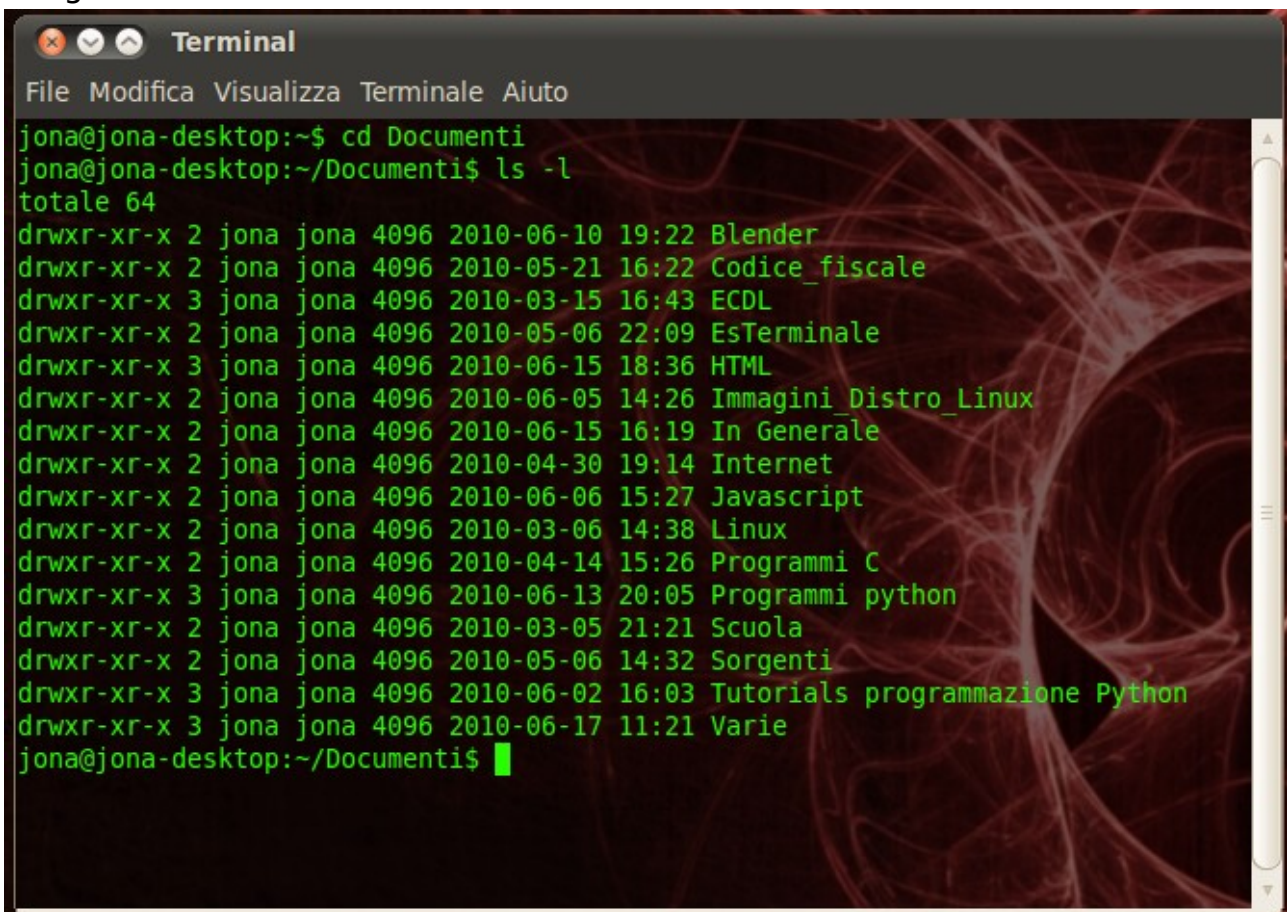
FILE E PERMESSI

Tra gli approfondimenti precedenti c'è un'istruzione che è necessario spiegare in maniera chiara:

il comando `ls -l` che permette di ottenere un elenco di file e cartelle più dettagliato che in particolare consente di distinguere il tipo di file, i permessi e l'appartenenza all'utente e al gruppo.

Mettiamolo subito alla prova:

Rechiamoci in una cartella, in questo caso Documenti, con `cd Documenti` e digitiamo il comando `ls -l`:



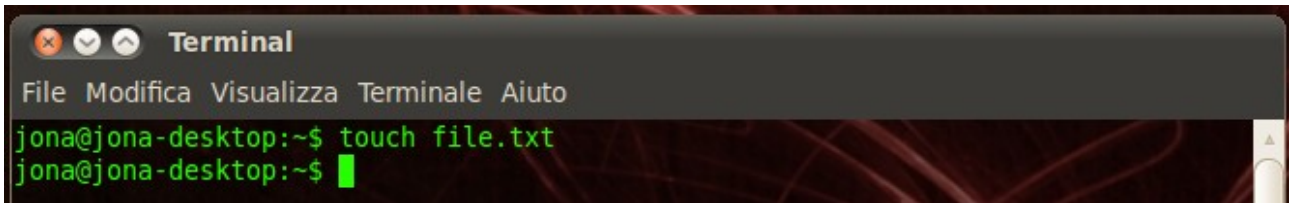
```
Terminal
File Modifica Visualizza Terminale Aiuto
jona@jona-desktop:~$ cd Documenti
jona@jona-desktop:~/Documenti$ ls -l
totale 64
drwxr-xr-x 2 jona jona 4096 2010-06-10 19:22 Blender
drwxr-xr-x 2 jona jona 4096 2010-05-21 16:22 Codice_fiscale
drwxr-xr-x 3 jona jona 4096 2010-03-15 16:43 ECDL
drwxr-xr-x 2 jona jona 4096 2010-05-06 22:09 EsTerminale
drwxr-xr-x 3 jona jona 4096 2010-06-15 18:36 HTML
drwxr-xr-x 2 jona jona 4096 2010-06-05 14:26 Immagini_Distro_Linux
drwxr-xr-x 2 jona jona 4096 2010-06-15 16:19 In Generale
drwxr-xr-x 2 jona jona 4096 2010-04-30 19:14 Internet
drwxr-xr-x 2 jona jona 4096 2010-06-06 15:27 Javascript
drwxr-xr-x 2 jona jona 4096 2010-03-06 14:38 Linux
drwxr-xr-x 2 jona jona 4096 2010-04-14 15:26 Programmi C
drwxr-xr-x 3 jona jona 4096 2010-06-13 20:05 Programmi python
drwxr-xr-x 2 jona jona 4096 2010-03-05 21:21 Scuola
drwxr-xr-x 2 jona jona 4096 2010-05-06 14:32 Sorgenti
drwxr-xr-x 3 jona jona 4096 2010-06-02 16:03 Tutoriali programmazione Python
drwxr-xr-x 3 jona jona 4096 2010-06-17 11:21 Varie
jona@jona-desktop:~/Documenti$
```

Con questo comando vengono elencati i file e le cartelle presenti nella directory corrente, nelle prime colonne si notano i permessi che hanno gli utenti (lettura, scrittura ecc...) poi abbiamo due colonne con il gruppo di appartenenza (a quale utente appartiene il file/cartella), successivamente c'è una colonna con dei numeri (ignoriamola), mentre le ultime tre colonne

elencano rispettivamente data e ora della creazione e il nome del file/cartella.

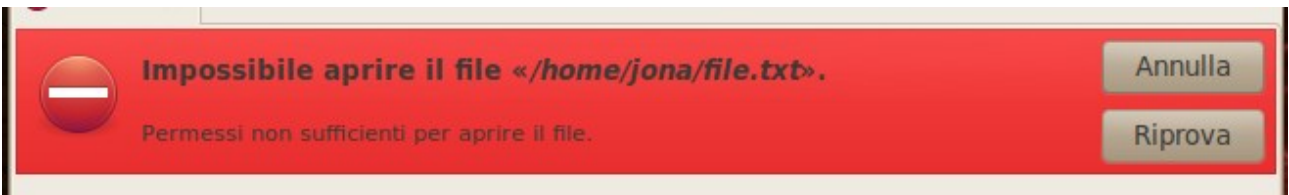
A volte ci potrà capitare di non riuscire a vedere il contenuto di un file, di non poter aprire una cartella o di non poter modificare un file, molto probabilmente è perché non si hanno i permessi necessari, ma niente paura, c'è una soluzione a tutto.

Facciamo un esempio, prima però devo spiegarvi come creare un file vuoto, questo è molto semplice, basta digitare da terminale **touch nomefile.txt** e verrà creato un file vuoto nella directory corrente.



```
Terminal
File Modifica Visualizza Terminale Aiuto
jona@jona-desktop:~$ touch file.txt
jona@jona-desktop:~$ █
```

Adesso digitate (non importa capire questa istruzione, la spiego dopo) l'istruzione **chmod -xwr nomefile.txt** (al posto di nomefile dovremmo mettere il file creato in precedenza) quindi andate su Risorse>Cartella Home e provate ad aprire il file...Non potete farlo vero?



Questo perché non avete i permessi necessari (Ve li ho fatti togliere io), ma niente paura, andate sul terminale, nella cartella dove è presente il file e digitate:

chmod +xwr nomefile.txt con questa istruzione si danno tutti i permessi al file così potete leggere scrivere ed aprire il file, per togliere i permessi al posto di + mettete -, potete anche aggiungere/rimuovere solo alcuni permessi ad esempio:

chmod -r nomefile.txt toglie il permesso di lettura ecc...

Queste operazioni possono essere eseguite anche sulle cartelle (**N.B.** L'attributo **x** dei permessi è l'esecuzione che permette di vedere il contenuto delle directory).

IMPORTANTE, se non si hanno determinati permessi su file o cartelle risulta impossibile anche copiarli, ad esempio, provate a togliere tutti i permessi ad un file e digitate:

`cp nomefile copiafile` verrà creata una copia del file solo se si hanno sufficienti permessi, altrimenti si riscontrerà un errore.

Adesso possiamo tranquillamente eliminare i file creati con:
`rm nomefile` che elimina i file elencati.

INSTALLAZIONE APPLICAZIONI

Per installare applicazioni il nostro sistema Ubuntu ci fornisce Ubuntu Software Center e Synaptic ma, come detto precedentemente, le operazioni effettuate su terminale sono molto più veloci perciò ecco come installare applicazioni da terminale.

-Installazione pacchetti .deb e .rpm

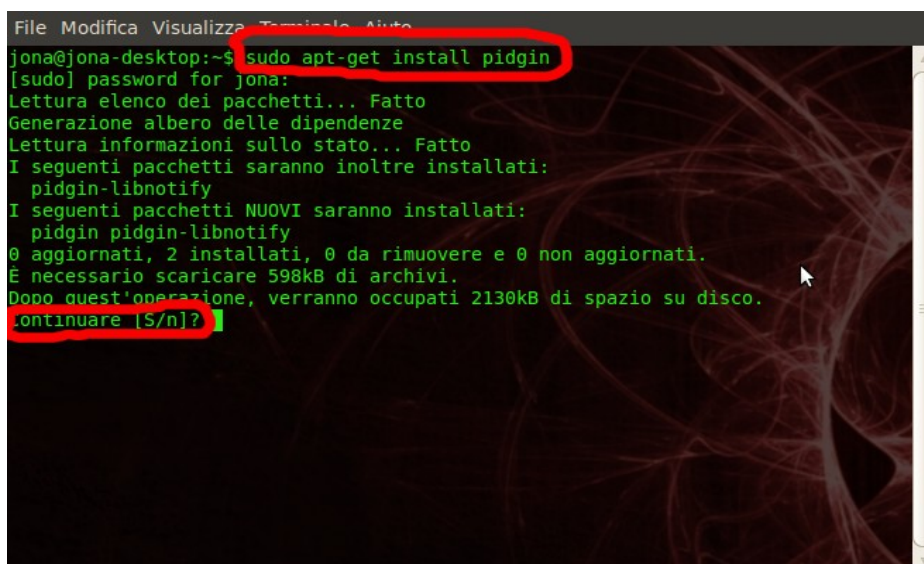
Il formato dei pacchetti (cioè dei programmi già compilati) Ubuntu è il **.deb** (derivato da Debian) e non, come molti credono passando da Windows a Linux, l'exe. I pacchetti .deb sono la maggioranza e per installarli da terminale è sufficiente avere **una connessione ad Internet**.

Se sappiamo il nome del pacchetto da installare è sufficiente dare da terminale:

`sudo apt-get install NomePacchetto` verrà chiesto di inserire la **password per i comandi sudo** e, se la password sarà corretta, inizierà lo scaricamento dei file, una volta terminato sarà chiesto all'utente se vuole continuare l'operazione di installazione ed egli dovrà rispondere **s** per sì o **n** per no.

Per essere più chiaro faccio un esempio pratico, supponiamo di voler installare Pidgin (un programma che serve per chattare con gli amici, tipo Msn per Windows), quindi apriamo il terminale e digitiamo:

`sudo apt-get install pidgin` ci verrà chiesto se vogliamo continuare o meno, rispondiamo **s**:



```
File Modifica Visualizza Terminale Aiuto
jona@jona-desktop:~$ sudo apt-get install pidgin
[sudo] password for jona:
Lettura elenco dei pacchetti... Fatto
Generazione albero delle dipendenze
Lettura informazioni sullo stato... Fatto
I seguenti pacchetti saranno inoltre installati:
  pidgin-libnotify
I seguenti pacchetti NUOVI saranno installati:
  pidgin pidgin-libnotify
0 aggiornati, 2 installati, 0 da rimuovere e 0 non aggiornati.
È necessario scaricare 598kB di archivi.
Dopo quest'operazione, verranno occupati 2130kB di spazio su disco.
continua (S/n)?
```

e l'installazione si completerà da sola. Adesso possiamo avviare pidgin! Per farlo andate su **Applicazioni>Internet>Pidgin**.

Non è molto frequente (a me è capitato solo una volta), ma è possibile che scarichiate da internet un programma che ha come estensione non il .deb ma .rpm questa estensione è la nativa di Red Hat (Una celebre distro Linux che introdusse questo tipo di pacchetto): questi pacchetti non combaciano con Ubuntu e tutte le distribuzioni derivate da Debian (Xubuntu, Kubuntu, Linux Mint ecc...) ma è possibile fare una conversione, cioè il computer genererà un pacchetto .deb con il comando:

sudo alien -k nomefile.rpm dove **nomefile.rpm** è il nome del pacchetto, dopo questa istruzione verrà chiesta la password per sudo.

Tuttavia c'è quasi sempre un'alternativa già in formato **.deb**, comunque a qualcuno quest'istruzione potrà servire.

-Installazione .tar.gz .tar.bz2

E' possibile che non si riesca a trovare il formato .deb di un pacchetto, perché nessuno ha mai compilato i sorgenti perciò dobbiamo farlo da soli, i sorgenti sono file con estensione **.tar.gz** o **.tar.bz2** potremmo definire un po' così il corrispettivo **.zip** di windows ma non è così perché in Microsoft non si compila un bel niente, e voi direte, meno male! Ed è qui che vi sbagliate, compilare i sorgenti potrà anche essere un'operazione più lunga di un'installazione da pacchetto .deb ma il programma si **plasmerà** perfettamente con il vostro Hardware ed andrà più veloce.

N.B. Si trovano in formato .tar.gz e .tar.bz2 anche i temi ma questi non devono essere compilati, dovete semplicemente installarli da **Sistema>Preferenze>Aspetto**.

I pacchetti sopra menzionati sono ostici, ma vediamo come affrontarli! Non vi fate intimorire!

Scarichiamo da internet un pacchetto in uno dei due formati, per l'esempio scarico il pacchetto di **Pidgin** da:

<http://sourceforge.net/projects/pidgin/files/Pidgin/>

e salviamolo.

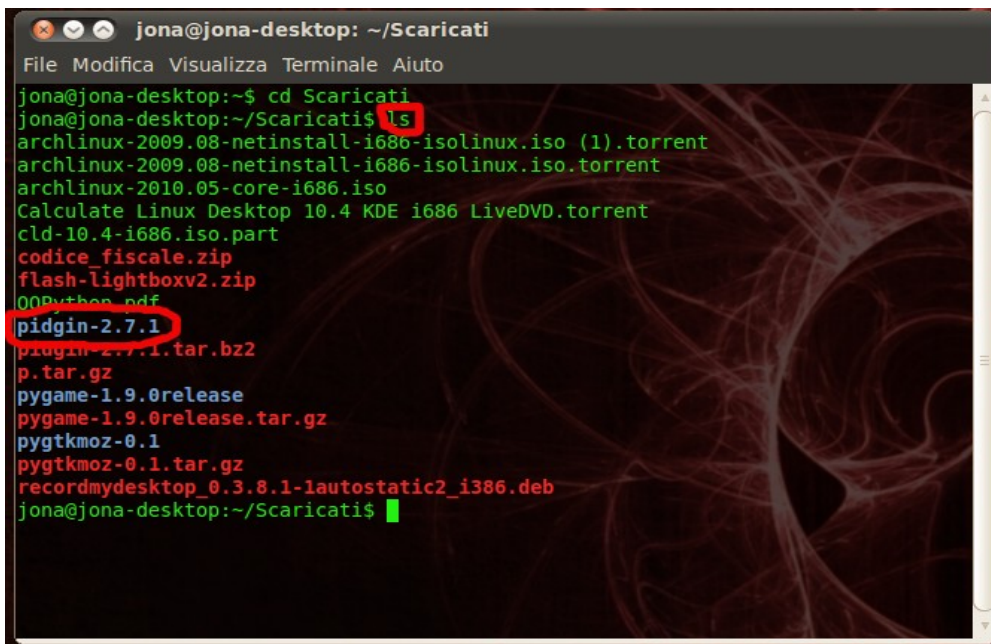
Adesso apriamo il terminale ed andiamo nella cartella in cui abbiamo il pacchetto, io l'ho messo in scaricati, perciò digiterò **cd Scaricati**:

Adesso vedete che il file che abbiamo scaricato è in formato **.tar.gz**

Quindi procediamo, per prima cosa dobbiamo scompattare il file (come quando da un .zip in Windows estraevamo i file) per scompattare il file dobbiamo usare:

`sudo tar zxvf pacchetto.tar.gz` al posto di `nomepacchetto.tar.gz` inseriamo `pidgin-2.7.1.tar.gz` questa istruzione scompatta i file `.tar.gz`.

Vedrete che l'output (cioè quello che farà il terminale) sarà una lunga lista di file. Adesso potete dare il comando `ls` e vedrete che sarà stata creata una nuova cartella contenente i file di configurazione:



```
jona@jona-desktop: ~/Scaricati
File Modifica Visualizza Terminale Aiuto
jona@jona-desktop:~$ cd Scaricati
jona@jona-desktop:~/Scaricati$ ls
archlinux-2009.08-netinstall-i686-isolinux.iso (1).torrent
archlinux-2009.08-netinstall-i686-isolinux.iso.torrent
archlinux-2010.05-core-i686.iso
Calculate Linux Desktop 10.4 KDE i686 LiveDVD.torrent
cld-10.4-i686.iso.part
codice_fiscale.zip
flash-lightboxv2.zip
00Python.pdf
pidgin-2.7.1
pidgin-2.7.1.tar.bz2
p.tar.gz
pygame-1.9.0release
pygame-1.9.0release.tar.gz
pygtkmoz-0.1
pygtkmoz-0.1.tar.gz
recordmydesktop_0.3.8.1-lautostatic2_i386.deb
jona@jona-desktop:~/Scaricati$
```

Ora siamo pronti per l'installazione, spostiamoci nella cartella che è stata creata durante la scompattazione con `cd pidgin-2.7.1` e digitiamo il comando `./configure`:

`./configure` questo comando serve per configurare i sorgenti dei pacchetti `.tar.gz` e `.tar.bz2`

Se non vi saranno errori potremo digitare prima il comando `make` e successivamente `makeinstall` che servono ad installare il programma.

N.B. Alcuni sorgenti possono essere **precompilati** (ad esempio Songbird), cioè basta scompattare il pacchetto e lanciare il **file install**, in ogni caso, è buona norma leggere (dopo aver scompattato) il file **README** che ci dà informazioni utili sull'installazione.

Se il formato del pacchetto fosse stato `.tar.bz2` avremmo dovuto scompattarlo con il seguente comando:

`tar jxvf nomepacchetto.tar.bz2` e seguire le altre istruzioni.

N.B. Per poter installare applicazioni .tar.bz2 o .tar.gz dovete prima munirvi di alcuni programmi, non dovete far altro che digitare da terminale:

```
sudo apt-get install build-essential
```

```
sudo apt-get install checkinstall
```

Per installarli.

RIMUOVERE PROGRAMMI E "PULIRE" LE DIPENDENZE.

Abbiamo imparato come si installano le applicazioni, ma come rimuoverle? Quest'operazione è molto semplice, non dovete far altro che digitare da terminale quest'istruzione:

```
sudo apt-get autoremove nomeprogramma
```

 e verrà eliminato il programma descritto.

Dovete sapere che quando in Ubuntu o in un qualsiasi sistema linux viene installata un'applicazione rimangono i file di configurazione, questi file è buona norma pulirli, una pratica che può essere assimilata alle pulizie di registro di Windows (molto lontanamente, infatti Ubuntu non genera tutta quella sporcizia).

Per eliminare questi file esistono alcune istruzioni, la prima è:

```
sudo apt-get autoclean
```

 che elimina i file di configurazione dei pacchetti non più installati.

La seconda è:

```
sudo apt-get clean
```

 che elimina i file di configurazione di tutti i pacchetti.

In più abbiamo:

```
sudo apt-get autoremove
```

 che elimina le dipendenze non più necessarie.

Ed infine ecco l'ultima istruzione (non vi spaventate! Non dovete ricordarla a memoria, mettetela da qualche parte...):

```
sudo dpkg --purge `COLUMNS=300 dpkg -l "*" | egrep  
"^\rc" | cut -d\ -f3`
```

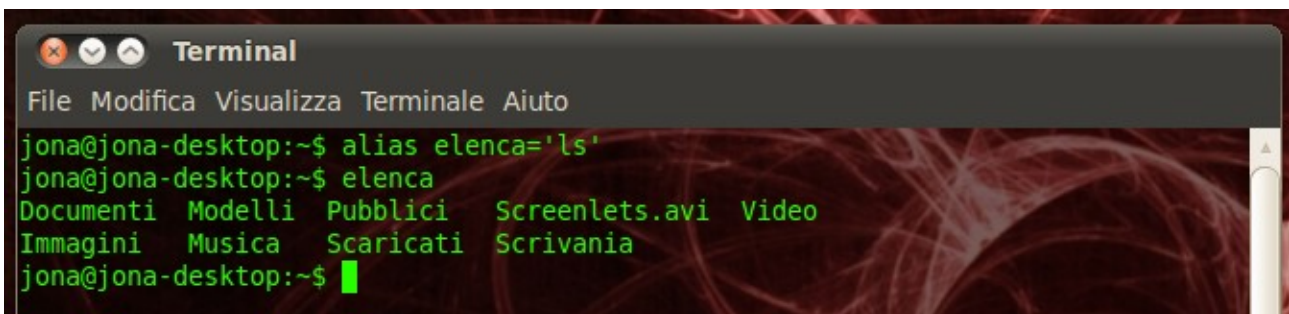
 questa istruzione elimina tutti i file di configurazione.

Mentre per tutte le istruzioni non c'è bisogno di spiegare molto devo fare un appunto su quest'ultima, se ricevete l'errore **dpkg -purge: richiede il nome di almeno un pacchetto come parametro** significa che il sistema è pulito, in caso contrario verranno elencate molte righe di pacchetti e dipendenze.

Esistono altre istruzioni per pulire il sistema, queste sono le principali: non voglio spiegarvi troppe cose insieme.

LE ULTIME DUE ISTRUZIONI

Eccoci alla fine, esaminiamo le ultime istruzioni, la prima la spiego con un esempio, digitate da terminale **alias elenca='ls'** e, dopo, provate a digitare **elenca**:



```
Terminal
File Modifica Visualizza Terminale Aiuto
jona@jona-desktop:~$ alias elenca='ls'
jona@jona-desktop:~$ elenca
Documenti  Modelli  Pubblici  Screenlets.avi  Video
Immagini  Musica   Scaricati  Scrivania
jona@jona-desktop:~$
```

Cosa è successo? Elenca ha preso gli stessi attributi di ls ed è diventata la stessa istruzione, infatti:

alias nuovocomando='comando' è un'istruzione che permette di attribuire un comando (tra apici) ad una 'variabile'.

I comandi da noi creati si chiamano **alias** e vengono persi a **fine sessione**, oppure potrete toglierli voi con il comando:

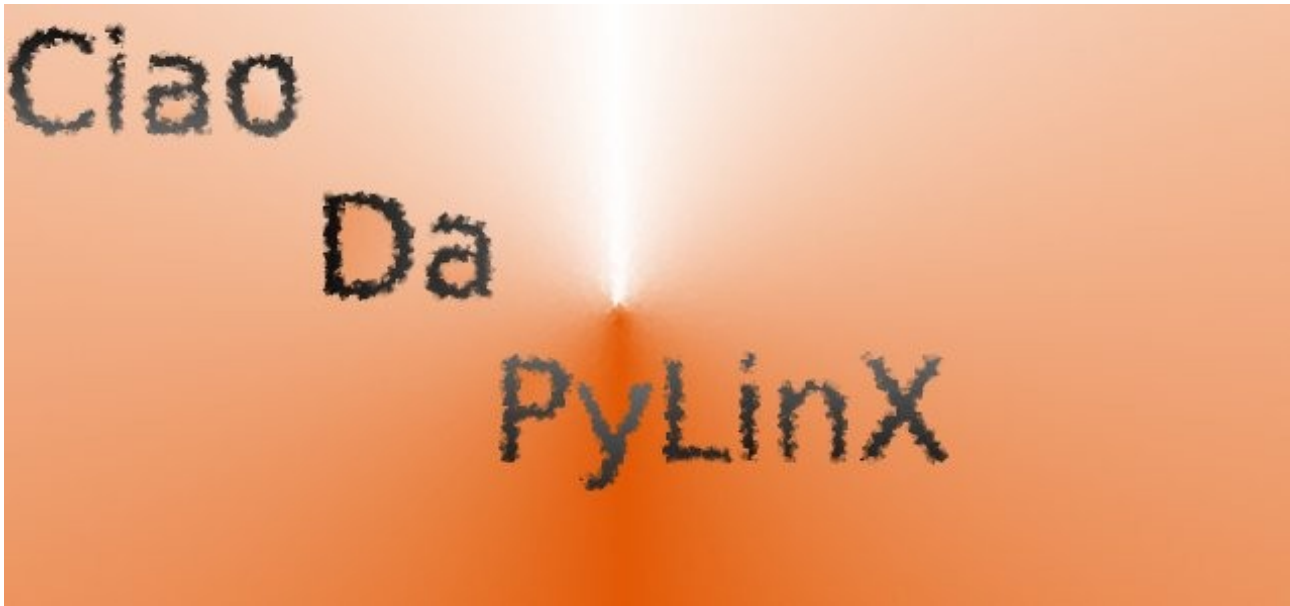
unalias nomecomando nel nostro caso sostituiremo **nomecomando** con **elenca**

Ecco l'ultima istruzione, volete fare gli aggiornamenti più velocemente? Digitate da terminale:

sudo apt-get update che aggiornerà il vostro sistema.

CONCLUSIONE

Benissimo! Sarò stato noioso comunque il “manuale” è concluso, infatti vi ho spiegato tutte le istruzioni basilari, se l'iniziativa piacerà ho in mente di scrivere altri manuali su GNU/Linux in cui vorrei trattare il mondo di Ubuntu ed argomenti più avanzati come, ad esempio, i **comandi mount**, gestire file di sistema come **/etc/fstab** o **xorg.conf**, e chissà, forse anche un'installazione di un sistema Linux più complicato come Slackware, Arch o Gentoo.



Grazie a tutti,
Jona Lelmi