

# **LINUX GIMMIX.**

Applications, Tips and Tricks for Linux Lovers.

Collected, compiled, borrowed, stolen and written by Mykl Martin.



# Table of Contents

INTRODUCTION.....	3
Cleaning Up.....	4
Recover .....	5
Re-installing.....	6
Music alarm.....	6
Talking Computers.....	6
Workspaces ( desktops ).....	6
Top Ten.....	7
Terminal output to text.....	7
Search History.....	8
Excuses.....	8
Insults.....	8
Skype.....	9
Control switch.....	9
Command Line Audio Player.....	9
Printing Problems.....	9
PART TWO..... PROBLEMS.....	10
Installing windows .exe files.....	10
PDF Files.....	11
Add Action to Right Click Menu.....	13
Floppy discs.....	14
PART THREE.....PROGRAMS AND OTHER THINGS.....	16
Alias.....	16
Open a text file in CLI.....	17
Trash CLI.....	17
Templates for "Create Document" menu.....	18
Video/Audio Converter.....	19
F.lux.....	20
CLI Companion.....	21
Task Warrior.....	22
Caps / Num Lock.....	23
PART FOUR.....SCRIPTS.....	24
Timing Is Everything.....	24
Solve That Puzzle.....	26
Man Files as readable text files.....	27
PART FIVE.....ACKNOWLEDGEMENTS.....	28

## INTRODUCTION.

Having switched from MS Windows to Linux about one year ago, I have been delighted and amazed by the ease of use, flexibility, freedom and fun of Linux.

Any problems I have encountered in the use of this OS have been answered by a host of friendly people, in many cases by hands-on practical help, many of whom can be found on a variety of 'Forums', all of whom are ready and willing to impart any knowledge / help needed.

Forums are an amazing way to gain experience, receive aid and give help to others. I encourage you to join as many as you can....just 'google' Linux Forums to see a great list of them.

One I'm particularly fond of is [www.thinkinlinux.org](http://www.thinkinlinux.org) , but don't just stick to that, check out as many as you can!

I've been given many hints and tips, discovered others myself, read tutorials which have given me answers and ideas, visited Forums, asked questions and looked at magazines; these I have compiled into this e-book and I extend my thanks to all who have helped me obtain this information; I wish I could remember where all the answers came from, but there are just too many sources and names to recall. ( a partial list is included at the end)

However, “Thank you all for your aid.”

I hope you find these collected gems as interesting as I do.

Try them out.....

Linux is Fun!

M Martin

01/04/11

First of all some quick 'one-liners' ( OK, some of them are 2/3 liners! )

## Cleaning Up

Here are a few quick commands to help clean your system.

Well, since we all use Synaptic or apt-get (Debian/Ubuntu) to get our lovely packages, sometimes packages aren't fully downloaded and they usually aren't deleted after we install them. Oh no!  
Can you imagine the space taken up?

Try this:

```
sudo apt-get clean
```

---

Unless you look at your logs a lot, you don't need these files that are about to be deleted, this command deletes gzipped packages in your logs directory, so type:

```
sudo find /var/log -type f -name *.gz -delete
```

---

This command removes any unused (redundant) dependencies from the system.

```
sudo apt-get autoremove
```

---

If you find that you have a padlock Icon on a folder and you can't delete folder ???  
try removing it as root.

press alt-f2 and type :

```
gksudo nautilus
```

enter your password, browse to the folder, and try deleting it again

---

Find and Delete empty directories

```
find . -depth -type d -empty -exec rmdir -v {} +
```

---

Stuff stuck in Trash

```
cd /home/USER_NAME/.local/share/Trash/files  
ls  
sudo rm -Rf *
```

---

Sometimes using the GUI things don't happen!

The 'Command Line' seems to 'kick' it alive again.

Erase rw cd

```
sudo cdrecord dev=/dev/cdrom blank=all
```

Erase rw dvd

```
sudo dvd+rw-format -force /dev/dvd
```

---

Ubuntu updates packages and kernels on a regular ( automatic ) basis, which means that once you find everything works OK you don't really need to keep all those outdated kernels, so...

First use this command to see the list of the installed kernels in your Ubuntu :

```
dpkg --get-selections | grep linux-image
```

Now for example try to delete the oldest one (change oldest kernel with your oldest one) :

```
sudo apt-get purge linux-image-2.6.31-14-generic
```

---

## Recover

If you mess too much up you can recover Gnome to its original settings AND retain your personal data, add ons etc.

```
sudo rm -rf .gnome .gnome2 .gconf .gconfd .metacity
```

---

## Re-installing

Before doing a fresh install, run this command to save a list of all package selections. This is a list of the packages that you had in your previous system.

```
sudo dpkg --get-selections > packages.list
```

Now, in the freshly installed system run this command to replicate all the packages that you had in your previous installation.

```
sudo dpkg --set-selections < packages.list && sudo apt-get dselect-upgrade
```

---

## Music alarm

You can run a timer function from the command line by using 'sleep'

Start music after x amount of time ( change '30s' to whatever time you require )

```
sleep 30s && audacious /home/USER_NAME/Super.mp3
```

Use your USER\_NAME folder, and the name of the music / sound you want !

---

## Talking Computers.

Speak words using 'espeak'

```
espeak "xxxxx"
```

(E.G. *sleep 8h && espeak "wake up buddy" && sleep 30s && espeak "now" && sleep 30s && vlc /path/to/music/file/that/will/piss/you/off/in/the/morning.mp3*)

---

## Workspaces ( desktops )

Increase number of workspaces : (change 'workspaces 4' to whatever number you want.)

```
gconftool-2 -s /apps/metacity/general/num_workspaces 4 --type int
```

---

## Top Ten

List your Top Ten CLI commands:

```
history | awk '{print $2}' | sort | uniq -c | sort -rn | head -10
```

---

## Terminal output to text

Redirect the output of a terminal to a file.

For example, if I want to list all the files in ~/Documents and output to a file called test.txt, I would do this:

```
ls ~/Documents > test.txt
```

---

There's always someone ready to ruin a good photo!



## Search History

*!whatever:p*

searches the history for the last command run starting with "whatever" and prints it to the screen (and puts it as the last run command in history even though it isn't run, so you can just hit the up arrow)

---

If you can't remember the command use:

**CTRL+r** for search-as-you-type; just enter whatever you can recall.

Immediately run the command by pressing enter.

---

## Excuses

If you want an excuse and can't think up a witty one, try this:

*echo `telnet bofh.jeffballard.us 666 2>/dev/null` |grep --color -o "Your excuse is:.\*\$"*

---

## Insults

When running the sudo command you are often required to enter your password and an interesting sudo feature is the ability to insult you when you type a wrong password! It really does have some funny phrases, here are just a couple of them:

- I've seen penguins that can type better than that.
- Maybe if you used more than just two fingers...

On Ubuntu, this feature comes disabled by default. To enable it, you need to edit the 'sudoers' file. Just type the following:

*sudo gedit /etc/sudoers* and put the word "*insults*" at the end of "Defaults..." line.

There may be nothing in the line or it could look like this:

*Defaults !lecture, tty\_tickets, !fqdn,*

whatever; enter *insults* at the end of the line.

That's it! Just run the sudo command and type a wrong password.

(The only problem is that you may never get what you want done, you'll be too busy typing wrong passwords!)



---

## Skype

With some installations Skype video won't work. Try adding this as the properties of a shortcut:

```
env LD_PRELOAD=/usr/lib32/libv4l/v4l2convert.so skype
```

---

## Control switch

To switch control buttons back to top right enter this into a terminal:

```
gconftool-2 -set "/apps/metacity/general/button_layout" --type string":minimize,maximize,close"
```

---

## Command Line Audio Player

(This needs vlc pre-installed)

```
nvlc /location/to/music/collection/file/to /play
```

For a random playlist:

```
nvlc /location/to/music/collection/ -Z
```

---

## Printing Problems

If you have problems printing PDF files to paper:

Change to directory holding the file

```
lpr -H localhost 'filename.pdf'
```

---

## PART TWO..... PROBLEMS

Now a (longer) word about a few problems...

### Installing windows .exe files

By default, Linux sets the permissions for Windows exe files to read only, and because CD ROMS are read only, you cannot change the file permissions on them.

So you cannot execute the files from the GUI. You need to use the command line and execute the file through Wine.

Wine implements the Windows Application Programming Interface (API) library, meaning that when a Windows program tries to perform a function that Linux doesn't normally understand,

Wine will translate that program's instruction into one supported by the system. This is why you need to execute Wine from the command line rather than from the GUI when it comes to executing exe files from read only media.

Just follow these simple steps to install the Watchtower library: or other CD .exe file:

Insert the Watchtower CD in the CD ROM drive and open a terminal.

Now type **mount** and press enter. You should see an entry similar to this:

```
/dev/sr0 on /media/WTLIB09E type iso9660.
```

Now all you have to do is type:

**wine /media/WTLIB09E/setup.exe** (change the names of the directories as needed, and remember, Linux is case sensitive!), and press enter.

Wine will treat the exe file as it would in Windows and install the Watchtower library.

## PDF Files

To join / edit PDF files you will need the program called **pdftk**. You can get it by opening terminal and typing:

```
sudo apt-get install pdftk
```

Now you can merge pdf files by putting them in the same directory and typing:

```
pdftk 1.pdf 2.pdf 3.pdf cat output merged.pdf
```

or if you wish to merge files in alphabetical order you can simply type:

```
pdftk *.pdf cat output merged.pdf
```

Here are some more examples how to use pdftk:

(Using Handles):

```
pdftk A=1.pdf B=2.pdf cat A B output 12.pdf
```

Split Select Pages from Multiple PDFs into a New Document

```
pdftk A=one.pdf B=two.pdf cat A1-7 B1-5 A8 output combined.pdf
```

Encrypt a PDF using 128-Bit Strength (the Default) and Withhold All Permissions (the Default)

```
pdftk mydoc.pdf output mydoc.128.pdf owner_pw foopass
```

Same as Above, Except a Password is Required to Open the PDF

```
pdftk mydoc.pdf output mydoc.128.pdf owner_pw foo user_pw baz
```

Same as Above, Except Printing is Allowed (after the PDF is Open)

```
pdftk mydoc.pdf output mydoc.128.pdf owner_pw foo user_pw baz allow printing
```

Decrypt a PDF

```
pdftk secured.pdf input_pw foopass output unsecured.pdf
```

Join Two Files, One of Which is Encrypted (the Output is Not Encrypted)

```
pdftk A=secured.pdf mydoc.pdf input_pw A=foopass cat output combined.pdf
```

Uncompress PDF Page Streams for Editing the PDF Code in a Text Editor

*pdftk mydoc.pdf output mydoc.clear.pdf uncompress*

Repair a PDF's Corrupted XREF Table and Stream Lengths (If Possible)

*pdftk broken.pdf output fixed.pdf*

Burst a Single PDF Document into Single Pages and Report its Data to doc\_data.txt

*pdftk mydoc.pdf burst*

Report on PDF Document Metadata, Bookmarks and Page Labels

*pdftk mydoc.pdf dump\_data output report.txt*



## Add Action to Right Click Menu

Create the action

You will need Nautilus Actions Configuration. If not loaded open Synaptic and install Nautilus Actions Configuration.

Click on **System > Preferences > Nautilus Actions Configuration**.

This will start up the Nautilus Actions Configuration window. Let's say you want to create a short cut used to open the GNOME Terminal application. Here are the steps you need to use to do that:

1. Click the "Define a new action" button.
2. In the Action tab make sure you check the box for "Display item in location context menu" and enter "GNOME Terminal" in the Context Label area.
3. In the Command tab enter `/usr/bin/gnome-terminal` for the Command Path. Do not enter anything for the Parameters.
4. Click the Save button.

Your action has been created. Now, to make sure it will appear, you need to restart Nautilus. To do this you can either log out and log back in or you can open up a terminal window and issue the command `nautilus -q`.

## Floppy discs

So, you've just found a box of old floppy disks that have been hidden in a cupboard for years and you would like to see what is on them!

If these are floppy disks that you used on your old PC, then chances are they have been formatted with the MS-DOS file system.

Many Linux users struggle to access the data stored on these floppy disks when using Linux because even though the floppy drive may be detected, you cannot see the files!

You will have no problem viewing files on a floppy disk that has been formatted with the ext2 or reiser file system, but when it comes to MS-DOS, you need to use **mtools**.

**Mtools** is a collection of basic tools for working with MS-DOS files and file systems. These tools allow you to access MS-DOS formatted floppy disks without having to mount them in Linux.

If these tools are not already installed in your system then you can install them using your package manager.

So, the first step is to see if you already have the mtools package installed. From a terminal type: **mtools** and press enter.

If mtools is installed, you should see a list of commands all starting with the letter m. If you get an error then you will need to install the package.

In Debian based systems you can install mtools by typing:

*#apt-get install mtools* and then enter your password.

In RPM based distributions you can type:

*#yum install mtools* and then enter.

**Note:** In the above commands, # signifies the root user. You can either log in as root or use the sudo command in Ubuntu and the su -c 'command' in other distros.

There are 25 commands in the mtools package but when it comes to accessing and transferring your files, you will probably only need to use a few of them.

Now insert one of your floppy disks that you want to access.

**Note:** You will need to use a terminal from this point forward.

A familiarity with MS-DOS commands will prove helpful but is not necessary. The one thing that you will have to remember is that MS-DOS uses the 8.3 file naming convention.

Without getting too technical, what this means is that file names that are longer than 8 characters not including the extension will be abbreviated. Here is an example:

Apples and oranges.jpg becomes applesan~jpg As you can see, spaces are ignored and the file name is shrunk to something that makes no sense in most cases.

Also, MS-DOS file names are not case sensitive as they are in Linux. All MS-DOS file names begin with a drive letter and a colon followed by the path to the file. EG: A:/Music/classic.mp3

Now depending on your system, begin the commands with either `sudo` or `su -c '`, as you need root privileges to use `mtools`.

For example, to view the contents of the floppy disk, use the `mdir` command by typing:

*sudo mdir a:* then press enter. You should now hear the floppy drive read the disk and show the contents.

If you have any directories there, you can **access** them by using the *mcd* command.

(*mcd* basically does the same as the `cd` command in Linux. It changes the current directory to a different one. )

EG: `mcd a:/pictures`

If you want to **copy** files from the floppy to your home directory you can do so by using the *mcopy* command.

Here are some examples:

Copy one file to the photos directory in your home folder. *mcopy a:/photo22 /home/user/photos/*

Copy all files on the root directory of the floppy drive to the photos directory. *mcopy a:/\* /home/user/photos/*

Copy the pictures directory from the floppy drive to your home folder. *mcopy -s a:/pictures /home/user/*

Copy all files and directories from the floppy to the backup folder in your home directory. *mcopy -s a:/\* /home/user/backup/*

Remember to use `sudo` or `su -c 'command'` when issuing these commands.

Practice makes perfect, so learn the commands you need to use by reading the man page by typing: *man mtools* or by the built in help: *mtools -help*

## PART THREE.....PROGRAMS AND OTHER THINGS

### Alias

Alias Smith and Jones?

One great feature I love in Linux is the ability to make lots of typing in the command line into a simple one or two word shortcut or alias.

You may have come across this mentioned in other posts; the received wisdom regarding aliases is that you should create a new file, name it `'.bash_aliases'` (note the 'dot' before the title) and add whatever you want as aliases, one per line, into the file.

see this extract from your `'.bashrc'` file

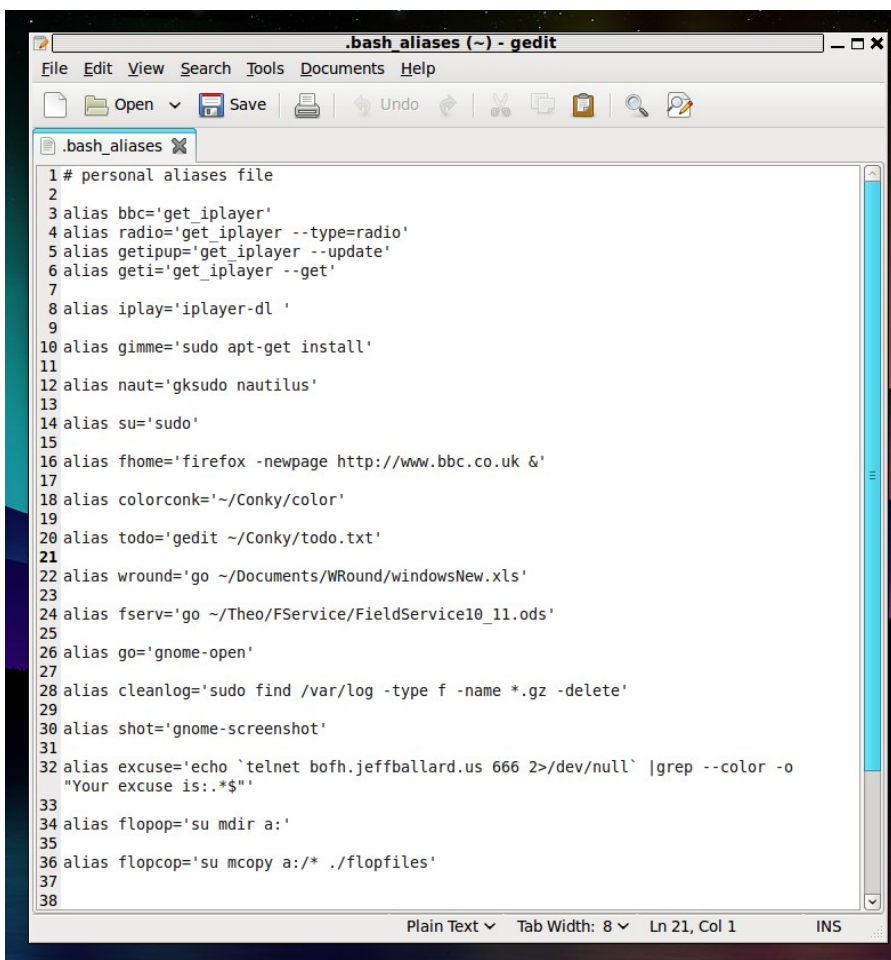
```
# Alias definitions.
```

```
# You may want to put all your additions into a separate file like
```

```
# ~/.bash_aliases, instead of adding them here directly.
```

```
# See /usr/share/doc/bash-doc/examples in the bash-doc package.
```

Here is a screen shot of my alias file



```
1 # personal aliases file
2
3 alias bbc='get_iplayer'
4 alias radio='get_iplayer --type=radio'
5 alias getipup='get_iplayer --update'
6 alias geti='get_iplayer --get'
7
8 alias iplay='iplayer-dl '
9
10 alias gimme='sudo apt-get install'
11
12 alias naut='gksudo nautilus'
13
14 alias su='sudo'
15
16 alias fhome='firefox -newpage http://www.bbc.co.uk &'
17
18 alias colorconk='~/Conky/color'
19
20 alias todo='gedit ~/Conky/todo.txt'
21
22 alias wround='go ~/Documents/WRound/windowsNew.xls'
23
24 alias fserv='go ~/Theo/FService/FieldService10_11.ods'
25
26 alias go='gnome-open'
27
28 alias cleanlog='sudo find /var/log -type f -name *.gz -delete'
29
30 alias shot='gnome-screenshot'
31
32 alias excuse='echo `telnet bofh.jeffballard.us 666 2>/dev/null` |grep --color -o
33 "Your excuse is:.*$"'
34
35 alias flopop='su mdir a:'
36
37 alias flopcop='su mcopy a:/* ./flopfiles'
38
```

As you can see you can have quite complicated commands set as aliases.

It really saves a great amount of time and if you can't remember them, just type into a terminal alias and it brings them all up!



## Open a text file in CLI

If you are working on the command line and you want an application to open a text file, do it running a command.

However, this requires you to know the name of the application, which can be difficult to remember, (bad memory!) Especially if you don't know what application it was written in! But Linux has a solution for people like us. Its called **gnome-open**.

Gnome-open actually makes use of the gnome file handlers to decide what kind of application should be used to open a particular kind of file.

Suppose you want to open some file, say "file.ext" and you are not sure of the application that handles the file format "ext", or you want to open a file called "news", then you can use gnome-open to ease your job.

First move to the directory in which the file is located:( ie Documents )

```
$ cd Documents
```

then type :

```
gnome-open news
```

voila!

---

## Trash CLI

great command line trash utility.

Just type the following to download this great utility:

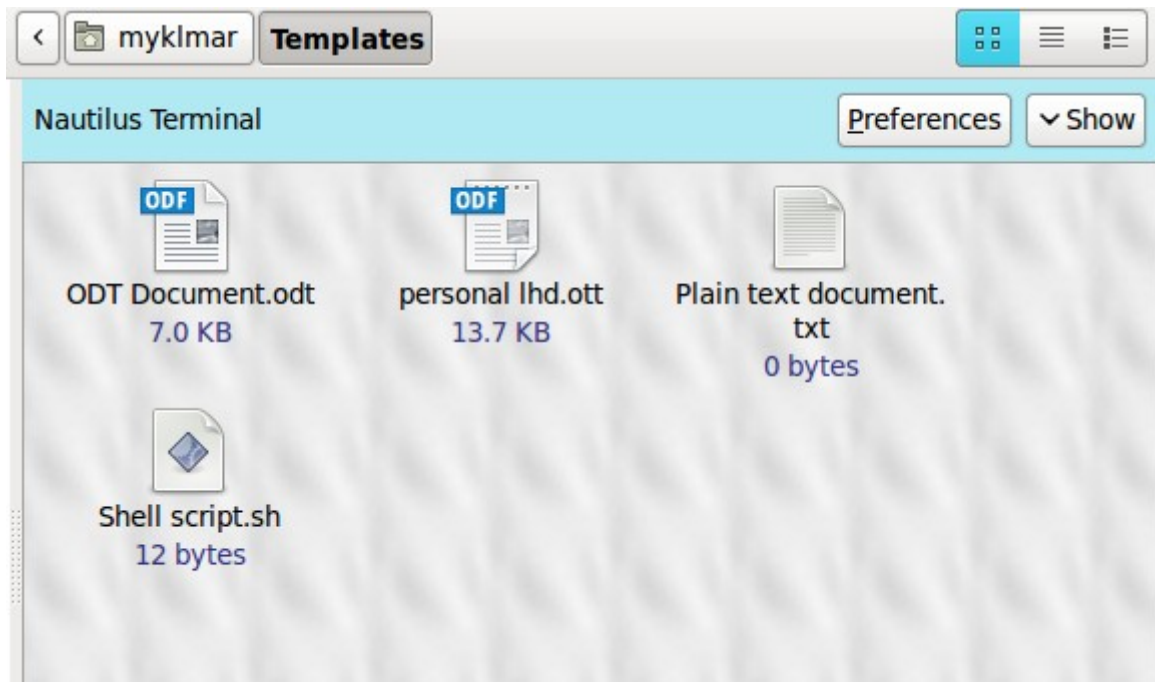
```
sudo apt-get install trash-cli
```

It remembers the name, original path, deletion date, and permissions of each trashed file, and by using various arguments you can:

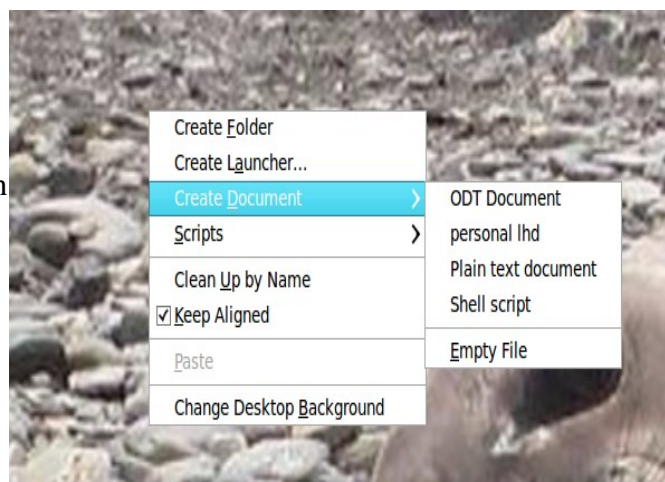
- \* send files to the trashcan; *trash NAME-FILE*
- \* list what's in there; *list-trash*
- \* restore whatever ones you want; *restore-trash* (it will then ask what file to restore)
- \* as well as emptying the can of course! ; *empty-trash*

## Templates for "Create Document" menu.

Like with most things in computing, there are usually several different ways to solve a problem. Some people do things one way, others prefer a completely different approach. Personally, when I was using Windows, I rarely used the function where you right-click to create a new blank document in a folder. When I moved to Linux, I never even noticed that Ubuntu doesn't include any templates for the "Create Document" menu. Recently, I read about the lack of these templates and the article went on to give advice on what to do. The answer is actually quite simple.



Each new user in Ubuntu has a default set of folders created for them in their home folder. One of those folders is called "Templates." Anything you put into the Templates folder will automatically show up as an option when you right-click in the file manager or your desktop and choose "Create Document." This is actually rather nice, because it means that you can make your own custom templates.



For example, if you wanted a custom OpenOffice document with your company logo at the top, you simply create a new document in OpenOffice and save it to your Templates folder. This can be repeated for as many different types of documents as you like!

## Video/Audio Converter

I know that there are a lot of video/audio converters for Linux, which are either GUI based or run from the command line. However, what is important is the number of formats that they can convert!

One that I highly recommend is **WINFF**.

Don't worry, it doesn't mean that it has to run in 'windows'; WINFF is a GUI for the command line converter FFmpeg, and as such will convert anything that FFmpeg can convert.

Amazingly it can work its magic on multiple formats at the same time! E.G you can convert mpeg, mov, mkv, flv into avi, all at the same time!

### Installing WinFF

WinFF supports Ubuntu and most of all the other Linux distributions as well. You can either download from the official site or you can use the following method:

Open up a terminal and type these commands:

```
sudo add-apt-repository ppa:paul-climbing/ppa  
sudo apt-get update && sudo apt-get install winff
```

### Running WinFF

Run by navigating to Applications>Sound & Video>WinFF or you can just press Alt+F2 and type WinFF.

The interface is simple and clean. It provides all the major settings for converting your audio/video.

One of its' best features is its ability to convert different files all to the same format at once!

As a side note after searching all over the 'web' for an application to succesfully convert .mkv files to .avi ( many did but there was either no audio or no video), guess what I ended up with? Yup!

Sounds interesting, doesn't it?

Among its many other features is that WinFF does not require any external codecs, and it supports FFmpeg's multi-threading for Dual Core processors.

To reveal more of its clever options just press the Options button!

You really must try this out....If you want to convert video/audio files in an extremely easy way, WINFF is the answer you're looking for!

# F.lux

## It's glaringly obvious.

I'm sure you know that spending long periods of time in front of blazingly bright computer screens isn't entirely good news for our eyes – or our sleep patterns. Nevertheless we stare away anyway, unaware that there are solutions.

You need an app that automatically adjusts your monitors colour temperature to that of your surroundings.

By reducing the glare of a bright screen at night you'll put less strain on your eyes and, potentially, induce normal sleep patterns.

*“During the day, computer screens look good—they're designed to look like the sun. But, at 9PM, 10PM, or 3AM, you probably shouldn't be looking at the sun.”*

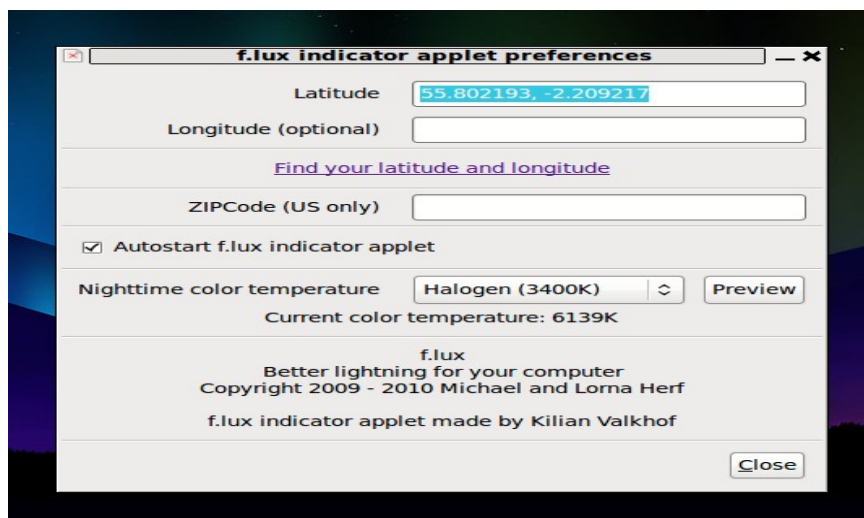
**F.lux** is a brilliant little app that takes care of this for you, and can be enabled to launch automatically at start up; it then progressively reduces the screen temperature as the day goes on, and produces a nice warm view!

To install the program open a terminal and enter the following:

```
sudo add-apt-repository ppa:kilian/f.lux
sudo apt-get update && sudo apt-get install fluxgui
```

Once installed you can launch it via the Applications > Accessories menu.

On first launch you will be prompted to enter your latitude and longitude (complete with easy link to find this.)



Check the “autostart” option if you would like f.lux to autostart and run in the background.

**This really makes a big difference to your computing enjoyment.**

# CLI Companion

CLI Companion is an application that is used as a compliment to the Linux Terminal. People unfamiliar with the terminal will find CLI Companion useful to help them become acquainted with the terminal. Using the built-in commands you can unlock the potential of the Terminal. It is a tool aimed at making the terminal easier to use: a GUI that displays a list of commands and an embedded terminal under it!

The main window has three columns. The first column is the command. The second column shows you if the command needs information from the user. (Some commands need a path to a file, others might need a package name.) The third column is the command description

The application comes with a list of commonly used commands by default, each having a short description and if you want to find out more about a certain command, simply right click it and select "Help". This will display the "man" (manual) for the selected command.

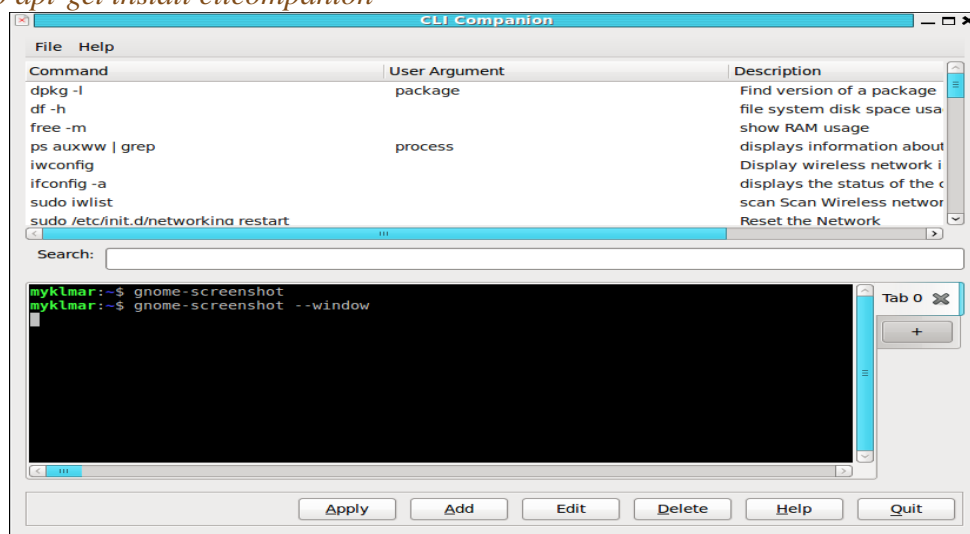
You can of course add your own commands to CLI Companion: simply click the "Add" button, then enter your command, an argument (optional) and a description.

To run a command, you just have to select it and click "Apply" and you'll see the output in the embedded terminal. If the command requires an argument, it will prompt you - all using a GUI. You can also run multiple commands at the same time: click the "+" sign on the right and a new tab will open.

Advanced users may also find CLI Companion useful by storing custom commands they came across for easier access, build a command dictionary, store commands as you come across them and, since CLI Companion has a built-in search tool, those commands will be easy to find and run.

CLI Companion comes with an Ubuntu PPA so you can easily stay up to date with new releases:

```
sudo add-apt-repository ppa:clicompanion-devs/clicompanion-nightlies  
sudo apt-get update  
sudo apt-get install clicompanion
```



# Task Warrior

To Do Or Not To Do.

Although I use a text file running through Conky for my 'To Do Tasks ' which is always showing on my desktop,

I found an extremely good Task manager to run from the CLI.

It is called "Task Warrior"

Obviously, you can add tasks, but also dates due, prioritise tasks, combine into a project, show on a calendar, delete, mark as started, mark as ended, give countdown until due and many other things, as well as display in various colours!

It is really easy to add tasks, just type: '*task add <name-of-task>*'. Task is created and assigned an ID number, which you use to amend task. i.e. '*task start <id no: of task>*'

```
myklmar:~$ task long
ID Project Pri Added Started Due Recur Countdown Age Deps Tags Description
0 ThinkinLinux H 2/26/2011 2/27/2011 -8 hrs 21 mins write Task warrior article
2 ThinkinLinux 2/26/2011 15 mins upload article

2 tasks
myklmar:~$ task started 1
Task 1 "write Task warrior article"
- description will be changed from 'write Task warrior article' to 'started'.
Proceed with change? (yes/no/All/quit) n
Modified 0 tasks.
myklmar:~$ task:started 1
task:started: command not found
myklmar:~$ task start 1
Started 1 'write Task warrior article'.
myklmar:~$ task long
ID Project Pri Added Started Due Recur Countdown Age Deps Tags Description
1 ThinkinLinux H 2/26/2011 2/26/2011 2/27/2011 -8 hrs 24 mins write Task warrior article
2 ThinkinLinux 2/26/2011 18 mins upload article

2 tasks
myklmar:~$ task cal

February 2011 March 2011 April 2011 May 2011 June 2011 July 2011
Su Mo Tu We Th Fr Sa Su Mo Tu We Th Fr Sa Su Mo Tu We Th Fr Sa Su Mo Tu We Th Fr Sa Su Mo Tu We Th Fr Sa
6 1 2 3 4 5 10 6 7 8 9 10 11 14 4 5 6 7 8 19 2 3 4 5 6 23 6 7 8 9 10 28 4 5 6 7 8 9
7 8 9 10 11 12 11 8 9 10 11 15 11 12 13 14 15 20 9 10 11 12 13 24 13 14 15 16 17 29 11 12 13 14 15 16
8 13 14 15 16 17 18 12 14 15 16 17 18 16 11 12 13 14 15 21 16 17 18 19 20 25 13 14 15 16 17 29 11 12 13 14 15 16
9 20 21 22 23 24 25 13 21 22 23 24 25 17 18 19 20 21 22 22 23 24 25 26 27 26 20 21 22 23 24 30 18 19 20 21 22 23
10 27 28 14 27 28 29 30 31 18 25 26 27 28 29 23 30 31 27 27 28 29 30 31 25 26 27 28 29 30 32

Legend: today, due-today, overdue, weekend, holiday, weeknumber.
```

By typing: '*task ls*' you get a list of all tasks and projects, type : '*task long*' and you get a detailed listing.

Putting different tasks into one 'project' is just as simple; just type: '*task <id no> - <id no> project:<name-of-project>*' DONE!

A full listing of all commands can be found by typing: '*task help*'

Perhaps the best thing about Task warrior is that it's available in all flavours of Linux!

Debian, Ubuntu, Fedora, ArchLinux, Gentoo, FreeBSD, SuSe and Redhat, even MacOSX and Cygwin!

If you are a lover of CLI or use the Terminal a lot, I suggest you give it a try.

Get it at <http://taskwarrior.org/wiki/taskwarrior/Download>



## Caps / Num Lock

Did I lock it?

My keyboard doesn't have any indicator lights for anything, and as I'm a clumsy typist I often find I'm suddenly typing in CAPS or my number pad doesn't work. Both due the fact that I've 'locked' or 'unlocked' the Lock keys.

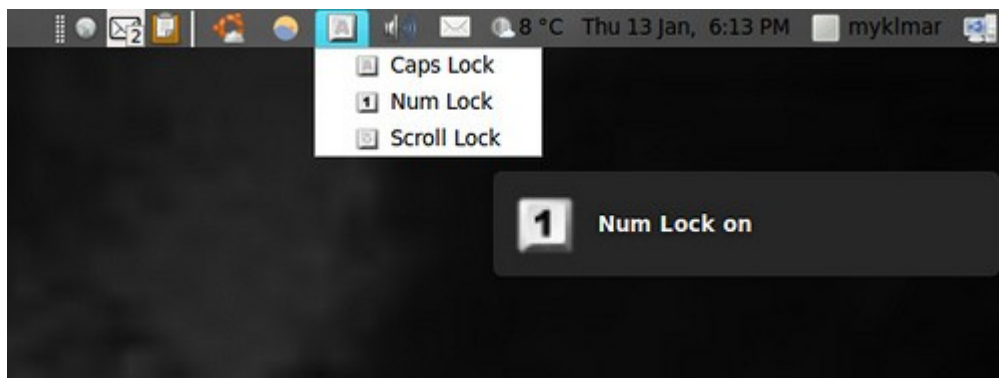
Imagine my delight to find an indicator app to tell me that very thing!

### Indicator Keylock....

To install this little indicator in Ubuntu copy the following:

```
sudo add-apt-repository ppa:tsbarnes/indicator-keylock && sudo apt-get update  
sudo apt-get install indicator-keylock
```

Now that it's installed, just start it by hitting Alt + F2 and typing 'indicator-keylock'



## PART FOUR.....SCRIPTS

### Timing Is Everything.

We often find ourselves, well me anyway, needing a timer that is easy to work and will run in the background; well here is an simple countdown timer script!

Copy the following code into a blank text file in your home/USER\_NAME folder.

```
#!/bin/bash

if [ "$#" -lt "2" ]; then
echo
echo "INCORRECT USAGE!"
echo "You need to supply either a DATE, a TIME or MINUTES"
echo "Example:"
echo 'countdown.sh -d "Jun 10 2011 16:06"'
echo 'OR'
echo 'countdown.sh -m 90'
echo
exit 1
fi

now=`date +%s`

if [ "$1" = "-d" ]; then
until=`date -d "$2" +%s`
sec_rem=`expr $until - $now`
echo "-d"

if [ $sec_rem -lt 1 ]; then
echo "$2 is already history !"
fi
fi

if [ "$1" = "-m" ]; then
until=`expr 60 \* $2`
until=`expr $until + $now`
sec_rem=`expr $until - $now`
echo "-m"

if [ $sec_rem -lt 1 ]; then
echo "$2 is already history !"
fi
fi
```



*fi*

```
while [ $sec_rem -gt 0 ]; do
  clear
  date
  let sec_rem=$sec_rem-1
  interval=$sec_rem
  seconds=`expr $interval % 60`
  interval=`expr $interval - $seconds`
  minutes=`expr $interval % 3600 / 60`
  interval=`expr $interval - $minutes`
  hours=`expr $interval % 86400 / 3600`
  interval=`expr $interval - $hours`
  days=`expr $interval % 604800 / 86400`
  interval=`expr $interval - $hours`
  weeks=`expr $interval / 604800`
  echo "-----"
  echo "Seconds: " $seconds
  echo "Minutes: " $minutes
  echo "Hours:  " $hours
  echo "Days:   " $days
  echo "Weeks:  " $weeks
  sleep 1

done

cvlc /usr/share/sounds/ubuntu/stereo/phone-incoming-call.ogg
```

Save it as 'countdown.sh'

BTW:The last line sounds a ring to wake you up!

Don't forget to make the 'countdown script' executable before trying to run it.

```
chmod +x countdown.sh
```

Now, to make this easy to run just move this file into your usr/local/bin.

```
sudo mv countdown.sh /usr/local/bin/countdown
```

Perfect!

To start the countdown just type:

countdown -m 6 (for a 6 min timer)

countdown -d 21:06 (to countdown until 21:06)

countdown -d "Dec 16 2010" (to countdown until....you guessed it!)

Happy timing.....tick tick tick.

## Solve That Puzzle

Stuck on a crossword, (if you still do them!) then this script will help you out.

Just copy the code and paste it into a new blank text file.

Save it as *cword.sh*

in your 'home/USER\_NAME' folder

```
#!/bin/bash
E_NOPATT=71      #no pattern exit error
DICT=/usr/share/dict/british-english
echo
if [ -z "$1" ]   #check to see if command line argument is there.
then
echo
echo
echo
echo -e "\033[1;31mHow To Use\033[0m\n";
echo "Type cword (followed by your known letters and unknown spaces)"
echo "using our \"$0\" \"pattern,\""
echo "Where \"pattern\" is in the following form"
echo "xxx..x.x..."
echo
echo "The x's represent known letters,"
echo "and the periods are unknown letters (blanks)."
echo "Letters and periods can be in any position."
echo "For example, try:  cword w...i....n"
echo
exit $E_NOPATT
fi

echo
echo -e "\033[1;31mAnswers could be..\033[0m\n";
echo
grep ^"$1" $ "$DICT"
# ^ = start of word regular expression anchor
# $ = end of word anchor
echo

exit $?

$ sh cword.sh w...i....n

wellington
workingman
```

Don't forget to make the 'script' executable before trying to run it.

```
chmod +x "script_name.sh"
```

Now, to make this easy to run just move this file into your usr/local/bin.

```
sudo mv "script_name.sh" /usr/local/bin/script_name
```

## Man Files as readable text files

Copy this into a blank text file, and save it as: **mantxt.sh** in your home/USER\_NAME folder.

```
#!/bin/bash  
man $1 | col -b > ~/Manuals/$1.txt
```

**Note:** see Don't forget box!

To run type:

```
mantxt <name_of_program>
```

The manual can be found in your USER\_NAME folder

---



**LINUX IS FUN!!**

## PART FIVE.....ACKNOWLEDGEMENTS

Many of these items have been sourced from Forums and Websites, and their origins have got lost in the stream of time.

However I want to express my thanks to the following:

*Shredder from Linuxers.org*

*Tuxmachines.org*

*Steve Chappel from ThinkinLinux.org*

*ghacks.net*

*K.Mandla from kmandla.wordpress.com*

*omgubuntu.co.uk*

*unixmen.com*

*starryhope.com*

*Mendel Cooper 'Advanced Bash Scripting Guide'*

*and far too many others.....*

If you read something here that should be credited to you please let me know, and accept my grateful thanks.

Hopefully something here has been of help to you, if so, remember what Albert Einstein once said:  
“ If I have seen farther than others it is because I have stood on the shoulders of giants.”

THANK YOU...to all the giants!

[myklmar@gmail.com](mailto:myklmar@gmail.com)