Solving The Case Of The Awful Laptop Keyboard

Word Processing With LaTeX (A Real World Example)

Playing Villagers And Heroes On PCLinuxOS

PCLinuxOS Family Member Spotlight: plankton172

GIMP Tutorial: Exploring G'MIC

Tip Top Tips: Slow KDE Applications Open/Save Dialogs Workaround

Repo Review: Focus Writer

Lumina Desktop Customization

2017 Total Solar Eclipse Wows North America
Table Of Contents

3  From The Chief Editor's Desk ...
5  2017 Solar Eclipse Wows North America
9  ms_meme's Nook: PCLinuxOS Melody
10  Solving The Case Of The Awful Laptop Keyboard
13  Screenshot Showcase
14  Lumina Desktop Configuration
25  Screenshot Showcase
26  PCLinuxOS Family Member Spotlight: plankton172
27  Screenshot Showcase
28  GIMP Tutorial: Exploring G'MIC
30  Screenshot Showcase
31  ms_meme's Nook: Everything's Gonna Be All Right
32  Word Processing With LaTeX (A Real World Example)
45  Repo Review: Focus Writer
46  Screenshot Showcase
47  Playing Villagers and Heroes in PCLinuxOS
50  PCLinuxOS Recipe Corner
51  Screenshot Showcase
52  Tip Top Tips: Slow KDE Application Open/Save Dialogs Workaround
53  Get Windows Off My Machine
54  PCLinuxOS Puzzled Partitions
58  More Screenshot Showcase
From The Chief Editor's Desk ...

The month of August started off just fine. We celebrated my son’s fourth birthday, followed by my daughter’s first birthday six days later. We made a trip to the Missouri State Fair – something we try to do every year. My wife grew up in Sedalia, MO (the town where the Missouri State Fair is held every year), so the fair is just a normal part of her life. She feels somewhat cheated if we don’t make it to the fair, mostly for a day of “fair food.” You know … that special, bad for you, deep fried everything that you can only get at a state fair. Cheese on a stick (cheddar cheese cubed and served on a stick, deep fried in corn dog batter) is her personal favorite, while funnel cakes are something we both clamor after. Plus, it’s a chance for the kids to visit their grandpa, who runs a concession stand every year at the state fair.

Then, the much anticipated 2017 total solar eclipse occurred over North America, treating millions to the opportunity of a lifetime to see a total solar eclipse, live and in person. For many, it is once in a lifetime chance, since most aren’t able to travel around the world to witness them. The path of totality went directly overhead where I live, and it was more majestic and awe inspiring to see than I could have imagined. It certainly left me feeling rather small in the grand scheme of everything.

But, I guess it was more than could be expected for the entire month to go by in similar grand fashion. The last week of August brought tragedy to
southeastern Texas, namely Corpus Christi, Galveston and Houston – where Texstar lives (in Spring, TX, a suburb of Houston). Tropical storm Harvey gained strength out over the Gulf of Mexico, and was a full-blown category 4 hurricane by the time it made landfall between Corpus Christi and Houston.

But then, as if a near direct hit on the area wasn’t enough, Harvey stalled out over the southeastern Texas coast, dumping more rain on the area than could be imagined ... and certainly more than the area could handle. Massive flooding ensued, with loss of life and property damage ranging into the billions of dollars (U.S.). As I write this on August 29, the deluge still isn’t over, with torrential rains that still haven’t stopped. The flooding is on a historical scale. Already, more than 9 trillion gallons of rain has fallen on the area (that’s more than 34,200,000,000 Liters for our friends not in the U.S.).

Not too much has been heard from Texstar throughout the ordeal. A day or two ago (remember to use the date I’m authoring this column, August 29, as the reference point), he reported that his home was still high and dry, but high water was only a street or two away. As you might expect, communications in and out of the area have been impacted, both by the hurricane as it made landfall, and by the massive flooding. He did send out word via Facebook on the evening of August 29 that he was safe.

If you feel inclined to help the victims of Hurricane Harvey, you can make a donation to the American Red Cross. Or, if you’d rather keep it closer to “home” (with PCLinuxOS being your “home”), you can make a donation to PCLinuxOS. You can make your donation through the PCLinuxOS GoFundMe page, or via GumRoad (different amounts are available on the PCLinuxOS main page). If you’re unable/unwilling to make a donation online, you can send a check or money order (drawn on a U.S. bank) to:

Bill Reynolds
18618 Cedar Edge Dr.
Spring, TX 77379

Let’s keep Texstar – and all of the victims of Hurricane Harvey – in our thoughts as they struggle through this tough and trying time.

Until next month, I bid you peace, happiness, serenity and prosperity.

Disclaimer

1. All the contents of The PCLinuxOS Magazine are only for general information and/or use. Such contents do not constitute advice and should not be relied upon in making (or refraining from making) any decision. Any specific advice or replies to queries in any part of the magazine is/are the opinion of such experts/consultants/persons and are not subscribed to by The PCLinuxOS Magazine.

2. The information in The PCLinuxOS Magazine is provided on an “AS IS” basis, and all warranties, expressed or implied of any kind, regarding any matter pertaining to any information, advice or replies are disclaimed and excluded.

3. The PCLinuxOS Magazine and its associates shall not be liable, at any time, for damages (including, but not limited to, without limitation, damages of any kind) arising in contract, tort or otherwise, from the use of or inability to use the magazine, or any of its contents, or from any action taken (or refrained from being taken) as a result of using the magazine or any such contents or for any failure of performance, error, omission, interruption, deletion, defect, delay in operation or transmission, computer virus, communications line failure, theft or destruction or unauthorized access to, alteration of, or use of information contained on the magazine.

4. No representations, warranties or guarantees whatsoever are made as to the accuracy, adequacy, reliability, completeness, suitability, or applicability of the information to a particular situation. All trademarks are the property of their respective owners.

5. Certain links on the magazine lead to resources located on servers maintained by third parties over whom The PCLinuxOS Magazine has no control or connection, business or otherwise. These sites are external to The PCLinuxOS Magazine and by visiting these, you are doing so of your own accord and assume all responsibility and liability for such action.

Material Submitted by Users
A majority of sections in the magazine contain materials submitted by users. The PCLinuxOS Magazine accepts no responsibility for the content, accuracy, conformity to applicable laws of such material.

Entire Agreement
These terms constitute the entire agreement between the parties with respect to the subject matter hereof and supersedes and replaces all prior or contemporaneous understandings or agreements, written or oral, regarding such subject matter.
2017 Solar Eclipse Wows North America

by Paul Arnote (parnote)

On August 21, 2017, North America experienced its first coast-to-coast total solar eclipse in nearly 100 years (98 years, to be exact). 12.25 million people live within the path of totality (where you can expect to see a 100% total solar eclipse), with estimates ranging from 2 million to 7 million additional people travelling from all over the world to see the eclipse within the path of totality. The total solar eclipse traveled from the Oregon coast, east to the Carolinas, right through the middle most part of the United States. Those outside the path of totality got to see a partial solar eclipse, with a lesser portion of the sun being blocked out by the moon the greater distance the viewer was from the path of totality.

This solar eclipse featured a path of totality of approximately 70 miles (113 Km) wide. Depending on where you were within that path of totality, you witnessed anywhere from around one minute of total solar blackout, to almost three minutes. The closer the viewer was to the center of the path of totality, the longer the total eclipse lasted. Unfortunately, some viewers were blocked out from seeing the solar eclipse at all by cloud cover and the weather (as in our assistant editor Meemaw's case, when she was visiting friends in Lawrence, Kansas).

Where I live, in Independence, Missouri, we were almost shut out from the total eclipse by the weather. Around 9:00 a.m. (CDT), a thunderstorm rolled through with torrential rains, thunder and lightning. Then, about 30 to 45 minutes before the start of the eclipse (at 11:46 a.m. CDT), the rain stopped and the clouds parted. About an hour after the eclipse, the clouds and storms moved back in, and we experienced thunderstorms off and on for the rest of the day and into the night. Look around, because your surroundings will hold some equally amazing sights. For example, the shadow of leaves on the ground will look way different, almost as if each little leaf projects its own copy of the eclipse above.
The magazine’s host and HTML layout champion, Mr. Cranky Pants - YouCantoo, was also in the path of totality, near the Oregon coast. He managed to capture some pretty amazing photos, as well.

As you can imagine, NASA was working overtime on August 21, 2017, to capture the rare North American total solar eclipse from as many angles as possible. NASA had teams everywhere you can imagine. Two high speed jets tracked the solar eclipse along the path of totality. The astronauts aboard the ISS (International Space Station) captured views of the eclipse – three times. Personnel were placed all along the path of totality, giving the agency as much coverage and data collection capability as possible.
2017 Solar Eclipse Wows North America

As luck would have it, the ISS photo bombed the total solar eclipse, in a way. In the composite image on the left (made from seven different exposures), the ISS is seen tracking across the partial solar eclipse, not long after the eclipse started.

A total solar eclipse occurs somewhere on the planet (on average) every 18 months. The next one will be in the southern Pacific ocean, and traverse across Chile and Argentina, on July 2, 2019.

So why doesn't a total solar eclipse occur more often? Well, this explanation from Wikipedia should explain it sufficiently. Click here to read the entire Wikipedia entry on solar eclipses.

A solar eclipse (as seen from the planet Earth) is a type of eclipse that occurs when the Moon passes between the Sun and Earth, and when the Moon fully or partially blocks (“occults”) the Sun. This can happen only at new moon when the Sun and the Moon are in conjunction as seen from Earth in an alignment referred to as syzygy. In a total eclipse, the disk of the Sun is fully obscured by the Moon. In partial and annular eclipses, only part of the Sun is obscured.

If the Moon were in a perfectly circular orbit, a little closer to the Earth, and in the same orbital plane, there would be total solar eclipses every month. However, since the Moon’s orbit is tilted at more than 5 degrees to the Earth’s orbit around the Sun, its shadow usually misses Earth. The Moon’s orbit must cross Earth’s ecliptic plane in order for an eclipse (both solar as well as lunar) to occur. In addition, the Moon’s actual orbit is elliptical, often taking it far enough away from Earth that its apparent size is not large enough to block the Sun entirely. The orbital planes cross each other at a line of nodes resulting in at least two, and up to five, solar eclipses occurring each year; no more than two of which can be total eclipses. However, total solar eclipses are rare at any particular location because totality exists only along a narrow path on the Earth’s surface traced by the Moon’s shadow or umbra.

NASA/Joel Kowsky
2017 Solar Eclipse Wows North America

Besides this map of North America, they have another map of the solar eclipses across the whole world from 2011 to 2060. This world map is available for purchase from here, for $10 (U.S.). The next solar eclipse for North America will be in 2024, and cut across Mexico, the United States, and parts of eastern Canada. The map above shows the paths of totality. The 2024 solar eclipse appears to cut a path of totality across the following U.S. cities:

Dallas, Texas
Little Rock, Arkansas
St. Louis, Missouri
Indianapolis, Indiana
Cleveland, Ohio
Rochester, New York

Summary

For many people – probably most, actually – viewing a total solar eclipse where you live is a once in a lifetime opportunity. To be sure, viewing a total solar eclipse live and in person isn’t a disappointment by any measure. I can attest that it is a magnificent and grand event to witness, as you marvel at the forces of the natural world around us.

If you ever have the chance to witness a total solar eclipse live and in person, by all means possible, avail yourself of the chance. It will be something that you will never forget. Just don’t forget to look around at your surroundings and other people nearby who are also viewing the eclipse, lest you miss some really nice photos and memories.

So what if you missed it?

Well, you can always watch video from those who were able to catch the 2017 total solar eclipse. But that isn’t nearly as satisfying as seeing it yourself, live and in person. Don’t despair, because astronomers have all of the future solar eclipses predicted out to the year 3000, along with their routes.
ms_meme's Nook: PCLinuxOS Melody

Somewhere there's music Tex wrote the tune
Notes for his OS one afternoon
There is no other melody that you will need
From Windows you will finally be freed

Somewhere there's music a rhapsody
PCLinuxOS a rhapsody
The best OS we are agreed
Love the rhythm love the speed

Somewhere there's music harmony
PCLinuxOS will make you swoon
There is no finer distro no indeed
Sing it hum it croon it you will succeed

Somewhere there's music harmony
PCLinuxOS will make you swoon
There is no finer distro no indeed
Sing it hum it croon it you will succeed

Somewhere there's music harmony
PCLinuxOS a rhapsody
The best OS we are agreed
Love the rhythm love the speed

**MP3**

**OGG**
Solving The Case Of
The Awful Laptop Keyboard

by Paul Arnote (parnote)

As I've mentioned before in the pages of this magazine, I have a number of computers, all running PCLinuxOS. I also don't keep it a secret that my favorite desktop environment is Xfce. It's not that I particularly dislike KDE, but it's just too resource greedy and too flashy for my needs and tastes. I'd rather use the CPU cycles and memory for useful tasks, instead of flash and glitz. I also don't have the “means” to have the latest, greatest of computing technology available to me, either.

I have a seven or eight year old Lenovo G530 laptop, which I purchased brand new. The only thing that had ever been installed on it had been KDE4 (well, other than Window 7 that came on it ... which was completely wiped from the hard drive shortly after I received the computer). Specifically, I used 32 bit KDE4, even though it has a 64 bit Intel Pentium Dual Core 2.2 GHz processor. When I first installed KDE4 on the computer, there wasn't a 64 bit version of PCLinuxOS available. Since the 32 bit version was running well, I didn't see the urgency of updating to the 64 bit version. It has the same 3 GB RAM installed that came with the computer. The KDE5 release lists 2 GB RAM as the minimum, with 4 GB RAM as the recommended amount of memory.

Over time, the Lenovo laptop went from my “go to” computer to use, to becoming one of the computers that I rarely used, for a number of reasons. First, it needed updating to 64 bit PCLinuxOS, especially since 32 bit version of PCLinuxOS ceased to exist quite some time ago. The 32 bit version of KDE4 was really showing its age and getting a bit long in the tooth from being unable to update it.

But the second reason is a myriad of hardware configuration issues that just never settled well with me. The laptop does not have a SD card reader, which I use quite frequently when downloading images from my digital cameras (and my digital trail cameras during hunting season). It has a paltry two (2) USB ports. Plus, the keyboard isn't laid out exceptionally well, despite having a nice feel when typing. Strike that ... the keyboard layout is awful. Sure, the keyboard layout is “standard” QWERTY keyboard layout. But it's how those “other” keys are laid out that make the keyboard awful. Particularly, the default keyboard layout has the Home and End keys accessible ONLY via a Fn key combination.

These are keys I use all the time! It is so awkward and user “unfriendly” to have to press the Fn + Page Up key for Home, and to press the Fn + Page Down key for End. But, for some odd reason, PrtSc and Pause – two keys I NEVER use – got their own keys! I know it seems like a small issue, and it may be – for some users. But to not have direct access to those two keys that I use all the time made using the keyboard frustrating.

So, very recently (like mid August), I dumped KDE4 and installed the most recent 64 bit version of Xfce from Ika's community remaster. That change came about when my Toshiba laptop (the least capable laptop I regularly use, and the one I use the most) developed problems. So now, I had to use the
I first looked at the X11 utility, \texttt{ev}. I looked around for some online tips on how to use the utility. Let’s just say that the documentation I found on how to use ev was like reading a first rough draft (and incomplete rough draft, at that) on brain surgery. There were holes in the documentation that you could have sailed an aircraft carrier through. So, ev quickly got tossed out of consideration. There had to be an easier way.

While looking for documentation on how to use ev, I kept stumbling across instructions on using \texttt{xmodmap} to remap keys on the keyboard. These instructions looked a lot easier and more user friendly than using ev. Lo and behold, it is easier. Plus, xmodmap was already installed on my computer. If you find it missing from your installation, you can easily install it from Synaptic.

The first thing you need to do (from the command line) is run the following command:

\texttt{xmodmap -pke > keycodes.txt}

This saves a copy of the default key code assignments to your current working directory. This way, you have a backup copy, just in case you royally screw things up.

Below is a snippet from the keycodes.txt file (each actual line in the file starts with the word “keycode”):

\begin{verbatim}
keycode 77 = Num_Lock NoSymbol Num_Lock
keycode 78 = Scroll_Lock NoSymbol
keycode 79 = KP_Home KP_7 KP_Home KP_7
keycode 80 = KP_Up KP_8 KP_Up KP_8
keycode 81 = KP_Prior KP_9 KP_Prior KP_9
keycode 82 = KP_Subtract KP_Subtract
keycode 83 = KP_Left KP_4 KP_Left KP_4
keycode 84 = KP_Begin KP_5 KP_Begin KP_5
keycode 85 = KP_Right KP_6 KP_Right KP_6
keycode 86 = KP_Add KP_Add KP_Add KP_Add
keycode 87 = KP_End KP_1 KP_End KP_1
keycode 88 = KP_Down KP_2 KP_Down KP_2
keycode 89 = KP_Next KP_3 KP_Next KP_3
keycode 90 = KP_Insert KP_0 KP_Insert KP_0
keycode 91 = KP_Delete KP_Decimal
keycode 92 = ISO_Level3_Shift NoSymbol
keycode 93 =
keycode 94 = less greater less greater bar
keycode 95 = F11 F11 F11 F11 F11
keycode 96 = F12 F12 F12 F12 F12
keycode 97 =
keycode 98 = Katakana NoSymbol Katakana
keycode 99 = Hiragana NoSymbol Hiragana
keycode 100 = Henkan_Mode NoSymbol
keycode 101 = Hiragana_Katakana NoSymbol
keycode 102 = Muhenkan NoSymbol Muhenkan
keycode 103 =
keycode 104 = KP_Enter NoSymbol KP_Enter
keycode 105 = Control_R NoSymbol Control_R
keycode 106 = KP_Divide KP_Divide KP_Divide KP_Divide
keycode 107 = Print_Sys_Req Print_Sys_Req
keycode 108 = Alt_R Meta_R Alt_R Meta_R
keycode 109 = Linefeed NoSymbol Linefeed
keycode 110 = Home NoSymbol Home
keycode 111 = Up NoSymbol Up
keycode 112 = Prior NoSymbol Prior
keycode 113 = Left NoSymbol Left
keycode 114 = Right NoSymbol Right
keycode 115 = End NoSymbol End
keycode 116 = Down NoSymbol Down
keycode 117 = Next NoSymbol Next
keycode 118 = Insert NoSymbol Insert
keycode 119 = Delete NoSymbol Delete
keycode 120 =
keycode 121 = XF86AudioMute NoSymbol
keycode 122 = XF86AudioLowerVolume NoSymbol
keycode 123 = XF86AudioRaiseVolume NoSymbol
keycode 124 = XF86PowerOff NoSymbol
keycode 125 = KP_Equal NoSymbol KP_Equal
keycode 126 = plusminus NoSymbol plusminus
\end{verbatim}
I've highlighted the four lines I'm interested in with red text. Make no changes to the keycodes.txt file. Otherwise, it'll be useless to you as a backup file.

Now, run this command:

```
xmodmap -pke > ~/.Xmodmap
```

This makes an identical file as keycodes.txt, but this time, the file is placed in your home directory, and is called .Xmodmap. Remember, the period in front of the filename designates it to be a hidden file. THIS is the file we will be editing.

Now, looking at keycode 79, copy everything to the right of the equal sign. Find keycode 107, and replace everything to the right of the equal sign with the values you copied from keycode 79. Repeat the process for keycode 87, replacing everything to the right of the equal sign of keycode 127. (All of this is for the keyboard on my Lenovo laptop. Chances are high that your laptop keyboard will have a slightly different configuration. The process will be similar, though, for other keyboards.)

So your ~/.Xmodmap file should look like this:

```
keycode 77 = Num_Lock NoSymbol Num_Lock
keycode 78 = Scroll_Lock NoSymbol
Scroll_Lock
keycode 79 = KP_Home KP_7 KP_Home KP_7
keycode 80 = KP_Up KP_8 KP_Up KP_8
keycode 81 = KP_Prior KP_9 KP_Prior KP_9
keycode 82 = KP_Subtract KP_Subtract
KP_Subtract KP_Subtract
KP_Subtract XF86Prep_VMode
keycode 83 = KP_Left KP_4 KP_Left KP_4
keycode 84 = KP_Begin KP_5 KP_Begin KP_5
keycode 85 = KP_Right KP_6 KP_Right KP_6
keycode 86 = KP_Add KP_Add KP_Add KP_Add
KP_Add KP_Add XF86Next_VMode
```

Next, run this command:

```
xmodmap ~/XF86Config
```

A first look at my ~/.Xmodmap file shows that the new keycodes for left, enter and right are in the right place. Now, the keyboard will give you some help in getting a new Mac keyboard working. All will need is for you to find the Mac keyboard layout and make sure it is in your ~/.Xmodmap file.

Now, if you run Xf86Setup, you should be able to select your Mac keyboard layout in the keyboard layout drop-down menu. Choose the Mac keyboard layout (mine is US English). Now go to the “Other” category and choose the “Modifier” option. Then, choose the “iso” keyboard layout from the list. Click “Apply” to save your changes. Now, you should be able to select your Mac keyboard layout in the “Other” category of the Xf86Setup dialog. Choose the “iso” keyboard layout from the list. Click “Apply” to save your changes.

To test it, run the following command:

```
xmodmap ~/.Xmodmap
```

If you did everything correctly, your unused keys should have their new functionality immediately available. So, go ahead and give it your “new” keys a try.

Of course, since this whole key remapping is under the control of xmodmap, the effects are temporary. When you reboot your computer, the same old, poorly thought out, awful keyboard layout will be right back.

Well, that is unless you tell your computer to run the xmodmap ~/.Xmodmap command when your desktop loads. Since I run Xfce, I simply launch the Xfce 4 Settings Manager, go to the Session and Startup utility, then click on the Application Autostart tab. From there, I add the command to run when the desktop is loaded. Most other desktops have similar capabilities.

Taking it further … if you want

For me, I stopped there. I now had wrestled control of my poorly laid out keyboard from the idiot who decided that PrtSc and Pause deserved their own keys, while Home and End did not. But, could I have gone farther? Yes, I could.
Had I wanted to preserve the functionality of PrtSc and Pause, I could have swapped the keycode definitions around. All I would have had to do is edit the values on the right side of the equal sign for each key. But since I already had absolutely NO use for the PrtSc and Pause keys (I even tried to find a use for those keys, by assigning them to automatically launch a frequently used program … but at the end of the day, these literally were unused keys), I left the Fn + Page Up key combination for Home intact, kept the Fn + Page Down key combination for End, and simply added two more keys that do the very same thing without having to press the Fn key. Had I wanted to, I could have just as easily converted my Fn + Page Up key to be my new PrtSc key, and converted my Fn + Page Down key to my new Pause key. However, in my mind anyways, that would have been a total waste of time, since I truly never used those keys.

Of course, xmodmap can do more, like even modify the function of the “command and control” keys (Alt, Ctrl, Shift, Super, etc.) of your keyboard. For example, some people think that the CAPS LOCK key has outlived its usefulness, so they remap it to be a Control key. This is completely separate from the fact that some people need to have the CAPS LOCK key disabled by default. The thing is, with xmodmap, you can remap any of the keys on your computer to do whatever you want them to do.

But another great thing about using xmodmap is that the effects on your keyboard are temporary. If you ever want to return your keyboard to its default layout, simply don’t run the xmodmap ~/.Xmodmap command (or remove it from the list of applications to autostart in your desktop configuration). Voila! Your keyboard is back to its old, crappy default layout.

Another use for xmodmap might go something like this: one of the keys on your laptop keyboard (well, it could be any keyboard) stops working properly. You can remap one of the keys you rarely use to duplicate the key that stopped working, allowing you to continue along your merry way.

For me, xmodmap allowed me to take a keyboard layout that was frustrating and a hassle to use, and change it to one that worked more like I have become accustomed to working. Now, I don’t feel so uncomfortable using the poorly laid out Lenovo keyboard, and my world is “more right.”
Lumina Desktop Customization

by phroneker

When we first launched Lumina, the desktop was quite minimal, to say the least.

Though it looks like there is only one panel on this desktop, there are really two panels. The second panel is hidden at the top center of the screen. By default, the layout of the panels is set up for efficiency, with the second panel revealed as needed.

Lumina comes with some preset layouts. These layouts are available in Desktop Settings accessible by right clicking on the background, then selecting Preferences, then all Desktop Settings from the popup menu, or by selecting Preferences from the main menu with the default layout.

The Preferences option will not appear in some main menus depending on the panel layout selected, but will appear in the default panel layout. To assure that you can get to the Desktop Settings, I recommend right clicking on the background, then selecting Preferences from the popup menu to be able to open Desktop Settings.

From here, we select Panels from Interface Configuration to customize the layout of the panels, as well as customizing the panels themselves.
This is the Lumina default panel layout with the second panel showing. There are four preset layouts available, and are accessed from the Profile button near the lower left hand corner of Desktop Settings.

You can choose from Windows, Mac OS-X, MATE (or GNOME 2), or Xfce, or you can also select None to start a layout from scratch.

...and here is Lumina’s version of the MATE desktop (above).

This is the Lumina take on the Xfce desktop (above).

...and this would be appropriate for PCLinuxOS running on a Intel-based MacBook.
And finally,

The classic “Windows” desktop. Each of these layouts can be customized and once you save your changes, the settings are restored each time you login to Lumina.

If at anytime you want to restore your desktop to the “factory settings”, open Desktop Settings, then select General Options, then click on Return to Lumina Defaults.

For this article, let us select Xfce. The main menu is now located at the upper left hand corner of your screen and is labelled Applications (just as in Xfce itself as it should be). On the upper right hand corner of the screen are four elements which are (from left to right) a workspace selector (for switching desktops within Lumina), the system clock and calendar, a system tray (which here contains the network applet), and a mini control panel that contains a volume control, battery monitor, a desktop switcher, and a button to exit the Lumina session.

There are two panels available in this layout as well. From the settings shown below, we should be able to ascertain which panel is which.

For each panel, there are three tabs. The first tab has settings that control the placement and size of the panel. Here the first panel is 21 pixels thick, takes the entire width of the screen, and is placed at the top edge of the screen. The second panel is 42 pixels thick, located at the bottom center of the screen, and utilizes only 20 percent of the width of the desktop.

We can change the placement, width and thickness of each of these panels.

The second tab has only two options, namely to auto-hide the panel, and to change the color of the panels. We already saw the auto-hide option as it is part
of the default layout of Lumina. The panel does not show on the screen unless you get the mouse pointer close enough to the edge where the panel is located.

When Use Custom Color is selected, you could click on the paint can to manually select a color, or you could click on Sample and choose a color off the current desktop.

Notice the Save icon now appears in the Desktop Settings. This indicates that changes need to be saved. Click on Save for the changes to take effect.

Now, let us look at the third tab.

The Application menu for this layout has changed from the default layout. To simulate Xfce, the application menu is also displayed in a manner similar to Xfce.

Now, look at Panel 2. The Xfce panel is simulated with lines and spacers in the configuration of this panel. We can add and remove elements to this to give it a panel more like the real Xfce (or anything else you want for that matter).

For this article, this is where the fun begins. What we should do here is to add commonly used applications to Panel 2. Start by clicking on the “+” icon in Panel 2.
Here is a list of available elements we can place in a Panel. To add applications to any Panel, we need to select **Application Launcher**. You will need to wait a few seconds for the list of applications to appear (depending on what you have installed in PCLinuxOS).

Click (and hold the mouse button down) on the popup menu here to pull the list of applications. Suppose we want to add Firefox to Panel 2. We then scroll down the list and select **Firefox**, then click on **OK**.

Firefox is now in Panel 2. But look where it is! This is not where we want Firefox to appear in the Panel. The arrow buttons on the lower right hand of where we are editing Panel 2 will move the order the Firefox element will appear in the Panel.

Move Firefox up to the second position from the top, and click on **Save**. Now, we can see Firefox in the bottom panel of the screen. However, we need to remove some considerable space between the Desktop Bar and Firefox. Select the third tab (the one with the puzzle piece under where it says **Panel 2**). Click on **Spacer** below Firefox and delete it (using the **minus** button). We should now have the following in **Desktop Settings**:

We can get rid of the other **Spacer** by clicking on Spacer, and deleting it with the **Minus** button. At some point, we will need to adjust the length of the panel. As it
is, we have room for two more applications before we need to adjust the size of the panel.

One common application we use is Thunderbird to send and receive e-mail. We simply add Thunderbird the same way we did for Firefox, and position it in Panel 2 just under Firefox.

For the next item, let us add a Start Menu element. When we clicked on the plus button, we were asked to select a Plugin. Elements are nothing more than plugins to the Lumina system. The Start Menu is included in Panel 1 in the default layout. To add the Start Menu to Panel 2, select Start Menu instead of Application launcher from the plugin menu. We will need to reposition Start Menu to the top of Panel 2.

Now, to clean up the panel, we get rid of the two line elements. At this point, there are some options on which to improve the appearance of Panel 2. We could adjust the width of Panel 2, or we could adjust the thickness of Panel 2. Now it is starting to look like a usable panel.

Before we get to that, consider that we can be as creative as we want with these panels. Just because we selected Xfce for the layout does not mean that the layout has to conform to the specifications of Xfce. We are, after all, customizing Lumina, not Xfce.

For this example of what can be done, let us start by adding three applications, an audio player, a Start Menu, and three lines and spacer, then order them as shown below:

We can adjust the color and sizing of this panel as well.

Here, we set the height of Panel 2 to 52 pixels and have it cover 40 percent of the width of the screen. For custom color, let us use 25% grey for the background shading (color value being hexadecimal #404040, or a decimal value of 64 of 256). Using these parameters, we have achieved this so far:
Now, let us click on **Back to Settings**. Under **Appearance**, there are four items to choose from to further customize this desktop. Let us click on **Theme**.

The templates are editable. These templates are coded based on cascading style sheets normally used for website development, and hence knowledge of CSS is a prerequisite for editing these templates.

...and there are eleven color schemes included with Lumina.

...each of these color schemes is editable. The widths of fields have been adjusted so you can read the attribute names in the **Color Scheme Editor**.

You can use the color picker tool to select your color from anywhere on the screen, or manually enter a color value. When entering color values manually, you can select a named color as defined by the W3C Consortium, or enter RGBA
(for Red, Green, Blue and Alpha) values formatted as \texttt{rgba(red, green, blue, alpha)} or as \texttt{rgb(red, green, blue)} in the \textit{Color Value} dialog box.

A good reference for color names can be found at \url{http://www.color-hex.com/color-names.html}. This website provides a chart containing the color name, a sample of each color, and its hexadecimal value.

Suppose we choose Black for the color scheme (and click on Save for the change to take effect). The colors in the panels change (if you did not select a custom color when configuring a panel). For this screenshot, I reset the color in the bottom panel to use the default system color.

Look what happened. The background in the RSS feed (on the lower right hand corner of the screen) changed to white text on a black background. The same color combination applies to active and inactive windows shown on the top panel. As you try each color (Clicking on \textbf{Save} each time to see the effect), notice the changes made in the panels.

You can also change the font, the icon pack and mouse cursors on the display as well. You may need to change the height of the top panel to accommodate the change in the font size. By default, Lumina assigns a 10 point Arial (Helvetica) as a system default for most elements on the desktop.

The exception here is the font used on the title bars on the windows. These are taken from Fluxbox default settings. Here, 10 point Letter Gothic (monospace) is used as the default font for the title bar.

Samples of the fonts on the list are shown for TrueType and Adobe Type 1 fonts. On my laptop, I have more than 250 fonts installed, as we can see from this partial list. Let us say we want to use Futura (you can choose any font you wish). We then select that font from the menu and click on \textbf{Save}. (This font can be downloaded from \url{https://www.futurafree.com}).
The fonts in the top Panel, the icons and the RSS reader have changed, as will any element on the desktop specific to Lumina.

For some reason, the Lumina developers made a decision to use GTK for the Desktop Settings and some other dialog boxes rather than the Qt5 toolkit on which Lumina was originally intended to be developed.

So that you can see the changes better, here is a closer look at the desktop with the context menu (accessed by right-clicking on the background) with the changed theme and selected font (Futura).

The **Window Manager Settings** allows you to change the look of the windows themselves. As Lumina is built on Fluxbox, Fluxbox themes determine the look of the windows. **Arch** is the default theme in Lumina for PCLinuxOS. The following below are the installed themes which can be used with Lumina.

For this example, let us select **BlueNight** (you can choose any theme you wish).
Lumina Desktop Customization

Here, we see two buttons, a plus and minus button, and a list of available wallpapers we can use. By default, only the default wallpaper is shown in the list (the one that has been the background all this time). The minus button removes a wallpaper selection from the list.

The plus button adds selections to the list of available wallpapers. Selections can be a single file, a single color, or a directory (with the option to include subdirectories within that directory). As for graphic file formats supported, anything that can be edited in GIMP, Kolourpaint, MTPaint or Inkscape is supported by this desktop environment (including Encapsulated PostScript files).

Because this is a Fluxbox theme, the Lumina settings such as fonts, colors and sizing are not applied here. At least, the theme is a better fit for this desktop. Now, let us get to the background itself. Click on Back to Settings and select Wallpaper from Appearance.

When you choose a directory with the recursive option, it could take a while to open the directory if there are a lot of files in that directory that can be used as wallpaper. This is useful if you want to display a slideshow for a background.

For this example, I used the local copy of my website where the wallpapers are stored.

http://horneker.com/pclinuxos-wallpapers.html
You can use any source as long as the images are stored locally (on your hard drive in a separate directory).

Because there are multiple images in this directory, we have the option of automatically changing the background at intervals. Also, we can use the minus button to remove items not to be displayed as wallpaper. As this is a reasonable selection and interval, we shall click on Save and wait a while for the wallpapers to change. By default, the background changes every five minutes. The interval can be set to as little as one minute, or as much as two hours (120 minutes).

The Layout option allows images to be tiled, scaled, centered, fit to the screen, or justified to an edge of the screen. By default, Lumina adjusts images accordingly depending on the image being displayed (in terms of size and aspect ratio).

Click on Save to begin the slideshow.

This is just the beginning of how to customize Lumina. What we did was more than just simulate Xfce, we kicked the Xfce experience up a notch.

The background shown here is a spoof of Raphael's School of Athens customized for the PCLinuxOS distribution. The spoof is called School of PCLinuxOS and is available for downloading on my website at http://horneker.com/pclinuxos-wallpapers.html. The wallpaper is 1680 x 1050 pixels in size.
PCLinuxOS Family Member Spotlight: plankton172

As told to YouCanToo

What is your name/username?
My real name is Rodel. My username is plankton172.

How old are you?
45 years old.

Are you married, single?
Married to the same woman since 1998.

How about Kids, Grandkids (names and ages)?
Just one son - Ken, who is now 18 years old. No grandchildren yet.

Do you have pets, what is your favorite?
No pets that we "own" but we have lots of cats (at least 10 of them!) who seem to make their home in our garage and backyard. My favorite cats from the pack are named Sandbag and Tyrion.

Are you retired, still working and if working, what do you do?
Still working. Currently employed as an administrative assistant by very friendly and super nice foreigners who are in my country for a diplomatic mission. My work is basically overseeing day to day activities in the office, with focus on statistical analysis, business solutions and technical support.

Where do you call home? What is it like? IE: weather, scenery
Home is in one among the > 7,100 islands of the Philippines! We live in the biggest island (Luzon) where the capital city of Manila is located.

The Philippines has tropical/temperate weather, which means it is either hot and humid, or rainy. Since we are right on the Pacific coast, we are always in the path of the strongest typhoons that the world has ever seen. I reside well outside the densely populated city -- in a fairly large town called Montalban. Where we live is near the mountain ranges -- which means plenty of trees, rivers, wide open spaces and fresh air.

Where did you go to school and what is your education level?
Elementary and high school I went to a parochial (i.e., Roman Catholic) institution in my home town of Pateros -- which is famous for being the source of the best "balut" in the country. After that, I attended college at the Philippine Christian University in Manila where I got my undergraduate bachelor's degree in computer science (surprise, surprise!).

What kind of things you like doing? hobbies, travel, fishing, camping?
I am totally into cycling. I do around 300 kilometers a week from biking to and from work and then some. Great way to beat the horrendous Manila traffic and keeps me in decent shape as a bonus. I am also into home theater and have a very modest LG plasma display + Onkyo-powered 5.1 system using locally manufactured speakers (assembled by a company
Why and when did you start using Linux?
In the late 90's, I was writing middleware using C++ and Python that ran in RedHat and Fedora. At that time, I was a senior software developer for a private company that sold and maintained office automation and retail management systems. I was only using Linux at work, and it wasn't until was January 2011 that I actually installed PCLinuxOS for our home computer – and it wasn't even the distro that I wanted to try at that time. One of my biking friends suggested Puppy or CrunchBang, but somehow I managed to get a live CD of PCLinuxOS. I tried it, I liked it, and the rest is history.

What specific equipment do you currently use with PCLinuxOS?
Two desktops (both custom builds) and one laptop (MSI GE402OL). I also installed on an Asus netbook but that poor unit is now very close to dying.

What would you like to see happen within PCLinuxOS that would make it a better place.
What are your feelings?
Support for the new Ryzen processor, as well as Intel's expected counter for it. PCLinuxOS is really super great even though it took a while for me to install on UEFI/GPT systems (had to wait until the release was stable enough). I wish I can seriously code again so that I can contribute to this uber-wonderful distro any way I can, but I feel my coding mojo is so weak now after using VBA exclusively in the last few years.

PCLinuxOS Family Member Spotlight is an exclusive, monthly column by YouCanToo, featuring PCLinuxOS forum member. This column will allow "the rest of us" to get to know our forum family members better, and will give those featured an opportunity to share their PCLinuxOS story with the rest of the world.

If you would like to be featured in PCLinuxOS Family Member Spotlight, please send a private message to youcantoo, parnote or Meemaw in the PCLinuxOS forum expressing your interest.
GIMP Tutorial: Exploring G'MIC

by Meemaw

GIMP has many wonderful filters you can use to enhance, fix or wildly change your photos, and more are being developed all the time. If you click on Filters, you will see one that says G'MIC at the bottom of the menu.

The letters stand for GREYC's Magic for Image Computing, and you can get G'MIC as a plug-in for GIMP or Krita, a command-line interface that works with ImageMagick or a stand-alone online program. It is available for Linux (and Windows), and is in the PCLinuxOS repository.

![GIMP logo]

From their website:

**G'MIC** is a full-featured open-source framework for image processing, distributed under the CeCILL license. It provides several different user interfaces to convert / manipulate / filter / visualize generic image datasets, ranging from 1d scalar signals to 3d+1 sequences of multi-spectral volumetric images, thus including 2d color images.

After opening a photo in GIMP, click on Filters > G'MIC. You'll see the following window (top, center):

![G'MIC window]

As you saw from the G'MIC window above, there are MANY categories with many filters in each, so I will describe them by the section and filter name so you can find them if you want to play.

Generally, starting out with one of these filters, then start "tweaking" to see if I can get a better outcome. Most of the time (depending on the photo), the defaults are right for the result I'm looking for, because I'm sure that in developing the filter, the author has spent many hours figuring out the settings that give the best result. So, using the photo of the windmill, and using the filter Details > Tone Enhance, we get this result:

![Windmill photo]

It seems to sharpen it up a bit.

If we use the Light & Shadow > Equalize Shadow instead, it seems to make it a bit more dull with less color (next page, top left):

![Windmill photo]

Both of these could be useful, depending on what your photograph looks like. If it's really bright and you want to tone it down a bit, the Equalize Shadow may work very well.
For a really different look, I also tried the filter called **Artistic > Pen Drawing**. It gives a totally different effect.

**Glow** gives a glow effect to the photo. If yours is dark, this may bring out highlights.

In the **Testing** section, under **Photocomix > Psychedelic Glasswork**, you can get some interesting effects. Go to the layers dialog and duplicate your layer first, then perform the filter on the top layer. Before you click OK, go into the settings for that filter and uncheck Activate Mirrors. When you are finished, change the layer mode to Overlay. You'll find that your layer mode will make a difference as well.

Using the photo of the Aspen trees, I tried more G'MIC filters. One called **Light & Shadow > Light**

Mine came out like this:

For something completely different, you can use **Black & White > Threshold Etch**. It made this photo look like a wood carving:
Not all of the filters in G'MIC just accent a photo: some of them give you something wild. I used this tiger photo, and did a filter called Artistic > Rodilus. You can see the result below the original.

This article has barely scratched the surface of all the filters you can find in G'MIC. I hope in your experimentation, you find some more awesome effects!

Posted by Stephen! on August 4, 2017, running KDE.
ms_meme's Nook:
Everything's Gonna Be All Right

It ain't no use for me to wonder why babe
It don't matter any how
It ain't no use for me to sit and cry babe
It's better that I do this now

With that Windows I have had my fling
When that bull steps into the ring
About PCLOS I will sing
Everything's gonna be all right

No longer playing that Windows' game babe
Got no use for it anymore
Things will never be the same babe
On it I'm closing the door

Don't want that Windows hanging 'round
Something better I have found
An OS that is safe and sound
Everything's gonna be all right

Breaking loose from Windows' reign babe
It never worked no how
Free from misery and pain babe
For PCLOS I'm shouting now

When I boot up Linux in the morn
No longer forgotten or forlorn
I feel that I've been reborn
Everything's gonna be all right
Word Processing With LaTeX
(A Real World Example)

by phorneker

Introduction to TeX/LaTeX showed us what the TeX typesetting system was and how it works with a rather simple example of a TeX/LaTeX document.

In that article, I mentioned the project called Mathematical Approach to Photography that was created for a mathematical modelling class I took in my senior year in college. At that time, the way to digitize photographs was to scan them in using bulky image scanners that were connected through either a proprietary interface, or through a SCSI controller.

Most photographs at that time were taken on film, and image scanners were prohibitively expensive for the consumer.

It is that project that I will use in this article to show what TeX can do.

For this article, I shall use Texmaker for development of the document. Texworks, Texstudio and LyX are all available in the repository, but Texmaker is widely used for TeX development and provides a work environment that allows us to see a complete overview of the project.

Here, we have the preamble to the document:

\documentclass[12pt,letterpaper]{article}
\usepackage[utf8]{inputenc}
\usepackage{amsmath}
\usepackage{amsfonts}
\usepackage{amssymb}
\usepackage{graphicx}
\usepackage[left=1in,right=1in,top=1in,bottom=1in]{geometry}
\author{Patrick G Horneker}
\title{A Mathematical Approach to Photography\ Revised for the 21st Century}

For this document, I chose letter paper (the standard 8½ x 11 inch size) as the paper for printing this document, and this will be typeset as an article with a 12 point font for normal text and UTF8 encoding (as opposed to the much older ASCII coding) for documents.

The standard AMS packages are used as is the graphicx and geometry packages. The graphicx package allows for PNG and JPEG images to be embedded in TeX documents.

The geometry package allows for setting of margins on a page. Here, one inch margins are set for the document as opposed to the default two inch margins.

Why is the default margin set at two inches? The answer to this can be found when documents are bound with a three ring binder. When holes are punched in standard 8½ x 11 inch paper, the left margin must be increased by one inch to accommodate the holes. The right margin allows for the reader to annotate text.

Hence the available space for typesetting is actually 6½ x 9 inches.

Remember taking college level exams in blue books?

Many colleges and universities provide blue books for writing answers to exam questions. These books are sixteen pages (four sheets of special paper folded to create the 16 pages) bound by a blue cover with the name of the institution
and spaces to write the name of the course, instructor, your name and of course, the grade for the exam.

Each page was 6½ inches width x 9 inches in length. Coincidence?

The following statement sets the document to one inch margins the same as any word processing package (such as LibreOffice) does.

\usepackage[left=1in,right=1in,top=1in,bottom=1in]{geometry}

If we really wanted to set the margins to what LibreOffice uses for its defaults, we could set the \usepackage statement to:

\usepackage[left=.75in,right=.75in,top=.75in,bottom=.75in]{geometry}

Now, let us look at the title statement.

\title{A Mathematical Approach to Photography\ Revised for the 21st Century}

There are two forward slashes in the middle of the title. This is done to (properly) split the title into two lines. If the slashes were not there, not only the word Revised here would have appeared on the first line, the title would have been too long to fit on just one line (and an error message would have resulted and/or the title would have been chopped at the ends).

The \maketitle statement renders the title at the beginning of the document. Had I declared the document to be of type book rather than of type article, the title would have been rendered as a separate page rather than being rendered at the top of the first page of this document (right, top).

With the book class, page numbers appear at the upper left hand corner of the page on even numbered pages and at the upper right hand corner of the page on odd numbered pages.

In addition, the geometry package overrides the defaults for typesetting in the book class, not just the article class. (Let us now revert back to the article class in the document to continue.)

Note that the numbering of sections, subsections, parts and chapters in the book class start with zero as opposed to numbering starting with one in the article class.

Next, let us look the sectioning. The statements:

\section{Introduction}

and

\subsection{What Has Mathematics Got To Do With Photography?}

have been numbered starting with one. But, does the numbering really matter here? Let us modify the \section statement so it reads

\section*{Introduction}

...and save the document. Now, build the document.
Two things happened here. The numbering for Introduction was turned off as expected. However, the numbering was reset so it starts at zero rather than one, and all other sections were renumbered accordingly.

Hence, the presence of numbering here really did make a difference. Let us reset the numbering in the Introduction header. Now everything is as expected. Let us go the next section called How Big is that Photograph?

The \bigskip statement is used to provide (plenty of) space between text and the tables.

Here, we have a block of LaTeX code that defines a table embedded into the document. In this block, we define two TeX/LaTeX environments.

The first TeX environment starts with a \begin{center} and ends with \end{center}. Everything contained within this environment is horizontally centered on the page. This particular environment is defined to center a table on the page. You can use this statement to center blocks of text, mathematical formulas, graphics or anything else contained within on that part of the document.

The second TeX environment starts with \begin{tabular}{|c|c|} and ends with \end{tabular}. The tabular parameter in the begin and end statements here tells TeX that the following block defines a table rather than ordinary text. Also, the tabular statement contains an additional parameter contained in its own set of braces that defines the table structure.

Tables are graphically defined with the borders between columns indicated by a \|. Text contained within the columns is justified by a parameter, which could be l, r, or c, where:

Lower case l means left justified, lower case r leans right justified, and lower case c means centered. In this case, there are three columns, all columns contain centered text.

The actual table is defined as follows:

\begin{table}[hbt]
\centering
\begin{tabular}{|c|c|}
\hline
Film Size & Image Size \\
\hline
110 & 16mm x 24mm \\
\hline
126 & 24mm x 24mm \\
\hline
35mm & 24mm x 36mm \\
\hline
35mm Half Frame & 24mm x 18mm \\
\hline
127 & 45mm x 45mm \\
\hline
120, 220, 620 & 60mm x one of 45, 60, 70, or 90mm\\
\hline
116, 616 & 65mm x 110mm \\
\hline
\end{tabular}
\end{table}

The \hline statements provide the top and bottom lines that define the borders of each cell in the table. The \| in the \begin{tabular}{|c|c|} statement provide the left and right borders of each cell.

Each line in the tabular environment defines a row in the table, with each of the rows separated by two forward slashes (\|). Columns in each row are separated by an ampersand (&).

The center environment is optional. Without the center environment, the table would be left justified on the page (at the beginning of where normal paragraphs start). (next page, top left)

In this case, the center environment is necessary to make the table easier to spot and read (next page, left center).

Moving further into the document, we notice a footnote. The \footnote[\url{http://camerapedia.wikia.com/wiki/116_film}] statement first tells TeX there that there is a URL contained in the footnote (indicated the the leading
and trailing $, and where in the document to place the footnote markers (in the text as well as on the footnote itself).

Footnotes are generally typeset with the footnote at the bottom of the page and its corresponding marker on the same page.

In computer science, a kilobyte is defined as 1,024 bytes instead of 1,000 as most of us would think. The number 1,024 is two to the 10th power (2¹⁰=1024), or what you get when you multiply two by itself ten times.

Likewise, two to the twentieth power 2²⁰=1,048,576 defines a megabyte, and hence, the prefix mega- in computer science refers to the number 1,048,576 instead of 1,000,000 as one would think.

...and therefore, a megapixel is defined as 1,048,576 pixels (or picture elements). This is the figure I used to calculate the megapixel values in this table.

Let us look at another table and its contents. The lines look like a mess, but there are five columns per row on this table. This is a table of scanned image sizes in terms of width and height in pixels. The x are not delimiters for the rows, but are necessary to show the width and height dimensions as one cell entry. Cell entries are delimited by an ampersand (&), and rows are delimited by two forward slashes (/).

```
\begin{center}
\begin{tabular}{|c|c|c|c|}
\hline
Film Size & 75dpi & 150dpi & 300dpi & 600dpi \\ 
\hline
110 & 48 x 71 & 94 x 142 & 189 x 283 & 378 x 568 \\ 
\hline
126 & 71 x 71 & 142 x 142 & 284 x 284 & 568 x 568 \\ 
\hline
35mm standard & 71 x 166 & 142 x 212 & 284 x 424 & 568 x 848 \\ 
\hline
35mm half & 53 x 71 & 106 x 142 & 212 x 284 & 424 x 568 \\ 
\hline
127 & 133 x 133 & 266 x 266 & 532 x 532 & 1064 x 1064 \\ 
\hline
120 half & 133 x 177 & 266 x 354 & 532 x 708 & 1063 x 1416 \\ 
\hline
120 square & 177 x 177 & 354 x 354 & 708 x 708 & 1416 x 1416 \\
\hline
\end{tabular}
\end{center}
```
Word Processing With LaTeX (A Real World Example)

\hline
120 (6x7) & 177 x 207 & 354 x 414 & 708 x 828 & 1416 x 1656 \\
\hline
120 full & 177 x 266 & 354 x 532 & 709 x 1064 & 1416 x 2128 \\
\hline
116 & 616 & 192 x 325 & 384 x 650 & 768 x 1300 & 1536 x 2600 \\
\hline
4x5in & 360 x 375 & 560 x 750 & 1200 x 1500 & 2400 x 3000 \\
\hline
5x7in & 375 x 525 & 750 x 1050 & 1500 x 2100 & 3000 x 4200 \\
\hline
8x10in & 600 x 750 & 1200 x 1500 & 2400 x 3000 & 4800 x 6000 \\
\hline
\end{tabular}
\end{center}

In the document preview panel, the table is neatly formatted and entries are easy to spot. What happens when we remove the \hline statements?

\begin{center}
\begin{tabular}
\end{center}

...then rebuild the document and this is what happens:

You can still read the table, but it stands out better with the lines in. By adding and deleting a combination of \hline statements within the table and/or modifying the second parameter in the \begin{tabular} statement, we can create some useful effects to the tables.

Suppose we change \begin{center} to \begin{flushright}. If we do this, we must also change \end{center} to \end{flushright} or we will get an error message (not to mention a messed up document in appearance). We get something like this:

You can still read the table, but now we have only vertical lines to separate the columns. It does not look as good, but the table is still readable.

Now, take out the vertical lines so the \begin{tabular} statement reads as:

\begin{tabular}{}{cccc}

PCLinuxOS Magazine
Word Processing With LaTeX (A Real World Example)

Let us now change the \texttt{\begin{flushright}} to \texttt{\begin{flushleft}} (and do likewise to the corresponding \texttt{\end} statement), we get this:

So what happens when we remove the \texttt{\begin{flushleft}} statement? One would think the table would be positioned the same. \textbf{Not quite!}

Having the \texttt{\begin{flushleft}} statement instead of simply removing the statement \textbf{does make a difference.}

The \texttt{\begin{flushleft}} statement places the table at the left margin on the page rather than the table being aligned to the beginning of a normal paragraph as not having the \texttt{\begin{flushleft}} statement would do.

Let us go back to the \texttt{\begin{tabular}{|c|c|c|c|}} statement. The second parameter defines the format for which columns in the table are displayed. Without any parameters, TeX will generate an error message. Hence, \textbf{there must be at least one column alignment value present} in the second parameter for a table to be typeset.

Valid column alignment values are \texttt{c}, \texttt{l}, and \texttt{r} for centered, left and right alignment respectively. The \texttt{|} is optional, and when present, typesets a vertical line in the table.

For the table to be typeset successfully, the second parameter in the \texttt{\begin{tabular}} statement must have one column alignment value for every column represented in the table itself. As ampersands (\&) delimit columns, each row in the table \textbf{must contain one fewer} ampersands than the number of columns in the row data.

In this example, there are five columns for each row. For the table to typeset successfully, there \textbf{must be four ampersands} in each row.

Let us look further at the \texttt{\begin{tabular}|c|c|c|c|} statement. The number of \texttt{|} characters depends on how many vertical lines you wish to typeset. In this example, the \texttt{\begin{tabular}|c|c|c|c|} statement typesets vertical lines between columns. If we remove all but the two outermost \texttt{|} characters, we will have a single border around the entire table.

We can change each of the values in the second parameter. Suppose we decide to left align each of the cells in this table. The \texttt{\begin{tabular}} statement will read as follows:

\texttt{\begin{tabular}{|l|l|l|l|l|}}

...and will result in this:}
Word Processing With LaTeX (A Real World Example)

On the first row of the table, it is a good idea to highlight the data in the first row of this table as this row is a row of column titles represented in the table.

\textbf{Film Size} & \textbf{75dpi} & \textbf{150dpi} & \textbf{300dpi} & \textbf{600dpi} \\
\hline
110 & 48 x 71 & 94 x 142 & 189 x 283 & 378 x 568 \\
120 & 71 x 106 & 142 x 212 & 284 x 424 & 568 x 848 \\
35mm standard & 71 x 106 & 142 x 212 & 284 x 424 & 568 x 848 \\
35mm half & 55 x 71 & 106 x 142 & 212 x 284 & 424 x 568 \\
127 & 133 x 183 & 266 x 354 & 532 x 708 & 1063 x 1416 \\
130 half & 133 x 177 & 266 x 348 & 532 x 708 & 1063 x 1416 \\
120 square & 157 x 177 & 314 x 354 & 708 x 708 & 1416 x 1416 \\
120 (3:1) & 133 x 207 & 266 x 410 & 708 x 828 & 1416 x 1416 \\
130 half & 137 x 166 & 274 x 354 & 708 x 708 & 1416 x 2128 \\
116, 616 & 192 x 235 & 384 x 506 & 768 x 1300 & 1536 x 2600 \\
4x5in & 300 x 375 & 600 x 750 & 1200 x 1500 & 2400 x 3000 \\
36mm & 219 x 252 & 438 x 506 & 876 x 1000 & 1752 x 2000 \\
3:2 & 600 x 750 & 1200 x 1500 & 2400 x 3000 & 4800 x 6000 \\
\hline

The titles in each of the columns are now in \textbf{boldface} as expected. (You may not be able to see this in this graphic, but the result is there.) Let us change the first column only so the labels are centered. The \texttt{\begin{tabular}} statement should read as follows:

\begin{tabular}{|c|l|l|l|l|}

Now the table should look like this (after rebuilding):

\begin{tabular}{|c|l|l|l|l|}

Let us look at the subsection below the table we just modified (right, top):

Below is the LaTeX code to create this part of the document.

\subsection{How I Calculated the Tables}

This was a relatively easy task, and a simple metric conversion was needed to perform that task. The first thing I did was convert millimeters to inches\footnote{Since 2.54cm equals 1 inch, therefore 25.4mm also equals 1 inch.}, then multiplied each of the millimeter sizes (both length and width) by 75 to get the number of pixels required to replicate the film image.

Hence, $\texttt{resolution} = (\texttt{fw} + 25.4) \times \texttt{resolution}$ where \texttt{resolution} is the width of the equivalent digital image in pixels and \texttt{resolution} is the width of the film image in millimeters. For these figures, \texttt{resolution} was assigned a value of 75 representing the calculations for 75dpi images. If we were calculating for 300dpi images, we would assign \texttt{resolution} to a value of 300, then do the calculations. The same formula applies for the length of the image.

The \texttt{paragraph} statement from the last article is not really needed to typeset this document. This LaTeX fragment contains a number of formatting commands, and has an example of how a mathematical formula is typeset.

Mathematical formulas are one of the greatest strengths of the TeX/LaTeX typesetting system. The $S$ tells TeX that what follows, until another $S$ is encountered, is a mathematical formula.

In this example, $S\texttt{resolution} = (\texttt{fw} + 25.4) \times \texttt{resolution}$ is typeset as a formula, namely

$$iw = (fw + 25.4) \times resolution$$

In a typical mathematical publication (of those that I have come across), variables are usually typeset in italics, hence the \texttt{itex} statements.
There is another \footnote{} statement in this section as well. This footnote refers to a simple conversion from inches to millimeters, specifically if there are 2.54cm to an inch, it then follows that there are 25.4 millimeters to that same inch.

Another example of a formula appears in the next section of this document. This one involves the Pythagorean Theorem.

This is a much better definition of \textit{focal length} and \textit{focal point} than what you can get online by searching \textit{definition of focal length} on Google\footnote{This is more proof that you cannot always rely on the Internet for everything. The Wikipedia definition is even worse.}.

The term \textit{normal} as related to lenses simply means that what you see in the camera does not have to be magnified or expanded to capture what you see on film. \textit{This is the default setting for digital cameras when you power them up for use.}

What is considered a normal focal length was easy with conventional photography. Technically, a normal lens is calculated using the Pythagorean Theorem, that is for a normal lens, \bigskip

\[ f_l = \frac{\text{iw}^2 + \text{il}^2}{2 \text{iw}} \]

where \texttt{f_l} is the normal focal length of the lens for the image size, \texttt{iw} is the width of the image on film, and \texttt{il} is the length of the image on film. \bigskip

The following is a table of film formats and their calculated normal lens sizes.

\begin{quote}
When a simple lens is held so that an image of the sun, or any very distant object, is focused sharply on a cardboard, the image is at the \texttt{principal focus} and the distance from the cardboard to the lens is the \texttt{focal length} of that lens.\end{quote}

...yields this in the document:

\begin{quote}
This is a much better definition of \textit{focal length} and \textit{focal point} than what you can get online by searching \textit{definition of focal length} on Google\footnote{This is more proof that you cannot always rely on the Internet for everything. The Wikipedia definition is even worse.}.

The term \textit{normal} as related to lenses simply means that what you see in the camera does not have to be magnified or expanded to capture what you see on film. \textit{This is the default setting for digital cameras when you power them up for use}. \bigskip

What is considered a normal focal length was easy with conventional photography. Technically, a normal lens is calculated using the Pythagorean Theorem, that is for a normal lens, \bigskip

\[ f_l = \frac{\text{iw}^2 + \text{il}^2}{2 \text{iw}} \]

where \texttt{f_l} is the normal focal length of the lens for the image size, \texttt{iw} is the width of the image on film, and \texttt{il} is the length of the image on film. \bigskip

The following is a table of film formats and their calculated normal lens sizes.
\end{quote}

The formula being typeset here is:

\[ f_l = \frac{\text{iw}^2 + \text{il}^2}{2 \text{iw}} \]

The \texttt{sqrt}{} command typesets the formula enclosed in {}, then draws a square root symbol encompassing the entire formula in the document. (Try that with LibreOffice.)

What happened? The quote environment indented the text by one half-inch in both the left and right margins.

In the last part of this sample document, there are two more features of TeX/LaTeX that are available in TeXLive. The graphics package allows for graphics to be embedded in the TeX/LaTeX document.

\begin{quote}
\texttt{includegraphics}[scale=.5]{angle-of-view.png}\end{quote}

Texmaker has a command in the LaTeX menu that allows you to use this command. Select a file from the dialog box that comes up. The embedded graphic must be in the same directory as the LaTeX files, and must be in a supported format (EPS, JPG and PNG are acceptable). (next page, top left)
Word Processing With LaTeX (A Real World Example)

The document, so far, produces a PDF that is seven pages in length. This is with setting the margins to one inch and using a 12 point Computer Modern font.

Below is the actual LaTeX code that produces the seven page document.

\documentclass[12pt,letterpaper]{article}
\usepackage[utf8]{inputenc}
\usepackage{amsmath}
\usepackage{amsfonts}
\usepackage{amssymb}
\usepackage{graphicx}
\usepackage[left=1in,right=1in,top=1in,bottom=1in]{geometry}
\author{Patrick G Horneker}
\title{A Mathematical Approach to Photography\hspace{1em}Revised for the 21st Century}
\begin{document}
\maketitle

\section{Introduction}

\subsection{What Has Mathematics Got To Do With Photography?}

Some of you are probably asking this very question. The answer to that is quite simple: \textit{everything}. One could argue that photography is an art. That is true. However, it is an art that involves some significant technical knowledge of the equipment used to produce this art. Without that knowledge, could you really know what you are doing to get that great photograph?

The intent of \textit{Mathematical Approach to Photography} is to explain the mathematics involved in every aspect of making a photograph, from the size of the film (or digital) image, to calculating the shutter speed and aperture settings, to printing the final product (be it a digital print on an inkjet printer, or exposing and developing a photograph in a darkroom).

\subsection{Historical Note}

This document is a complete rewrite of a class project I did back in Spring 1989 for a mathematical modeling class. When I originally wrote this, there were no digital cameras, and most scanners at that time could only scan in monochrome.

Many concepts that I presented in the original project are still valid today. They just needed to be updated for today's digital and conventional photography.

Finally, the following segment of LaTeX code produces the bibliography (or works cited) section commonly found in many documents that are typeset using this system.

\section{Sources Used}

\begin{itemize}
\end{itemize}

The \texttt{itemize} environment produces a bulleted list. Items to be bulleted start with the \texttt{item} command and end with the next \texttt{item} command, or with the \texttt{end{itemize}} statement.

As we can see, there is a lot more work to be done on this project. What I have presented to you is some of what TeX and LaTeX can do for your typesetting projects.
Word Processing With LaTeX (A Real World Example)

In the old days, we took pictures on film, that came in 35mm, medium format, large format, and 110/126 (aka the Instamatic film sizes). The bigger the film size, the better (and sharper) the final prints came out. This is somewhat true with digital photographs.

For the larger formats, film sizes are designated by the actual image sizes, such as 4x5, 5x7, or 8x10, all in inches.

Digital cameras produce photographs that are measured in megapixels rather than a physical image size on film. One would think that a megapixel represents one million picture elements, as the prefix \textit{mega} is typically defined as \textit{one million}. In the world of mathematics and computer science, \textit{mega} actually represents the number \textbf{1,848,576}, and is the number you get when you raise \textit{two} to the twentieth power, i.e. you multiply two by itself twenty times.

The following table contains image sizes used by digital cameras, tablets, computer displays and smartphones. The 720x480 size is the image size used for standard resolution DVD production.

For the 116 and 616 film sizes, the film stock was actually 70mm wide, but the image used only 65mm instead of 70mm.\footnote{http://camerapedia.wikia.com/wiki/116_film$}

The difference here is that we no longer think in terms of the negative size, but in the \textit{size of the image} (in pixels) as stored in a data file, and in the \textit{resolution} (in dots per inch) we use to print that image on a printer.

As we can see here, when you purchase a digital camera, the megapixel resolution given as a specification is a \textit{mere approximation} of the actual resolution of the image you get when
you take a photograph with a digital camera.

\textit{We got lucky here with the 3MP camera image as it is the only megapixel specification that actually is as advertised. The 12MP specification here is correct as advertised}\textbf{only if} we assume a megapixel is the approximated 1,000,000 pixels."

The following table represents image sizes converted from film negatives and transparencies to their digital image equivalents (after scanning).

\bigskip

\begin{center}
\begin{tabular}{|c|c|c|c|}
\hline 
Film Size & 75dpi & 150dpi & 300dpi & 600dpi \\
\hline
110 & 48 x 71 & 54 x 142 & 283 x 284 & 568 x 568 \\
\hline
126 & 71 x 112 & 142 x 284 & 568 x 568 \\
\hline
35mm standard & 71 x 106 & 284 x 424 & 568 x 848 \\
\hline
35mm half & 53 x 71 & 142 x 284 & 568 x 848 \\
\hline
127 & 133 x 266 & 532 x 1064 \\
\hline
120 half & 133 x 177 & 354 x 708 & 1063 x 1416 \\
\hline
120 square & 177 x 177 & 354 x 708 & 1416 x 1416 \\
\hline
120 (6x7) & 177 x 207 & 354 x 828 & 1416 x 1656 \\
\hline
120 full & 177 x 266 & 532 x 1064 & 2128 \\
\hline
116, 616 & 192 x 325 & 508 & 2600 \\
\hline
4x5in & 300 x 375 & 600 x 750 & 1200 x 1500 & 2400 x 3000 \\
\hline
5x7in & 375 x 525 & 1050 x 1500 & 2100 x 3000 \\
\hline
8x10in & 750 x 1200 & 2400 x 3000 & 4800 x 6000 \\
\hline
\end{tabular}
\end{center}

\bigskip

These figures had to be rounded so the contents of this table will make logical sense when you scan through the table.

\subsection{How I Calculated the Tables}

This was a relatively easy task, and a simple metric conversion was needed to perform that task. The first thing I did was convert millimeters to inches\footnote{Since 2.54cm equals 1 inch, therefore 25.4mm also equals 1 inch.}, then multiplied each of the millimeter sizes (both length and width) by 75 to get the number of pixels required to replicate the film image.

\begin{equation}
\text{Resolution} = (\frac{\text{Width}}{25.4}) \times \text{Resolution}
\end{equation}

Hence, $\text{Resolution} = (\frac{\text{Width}}{25.4}) \times \text{Resolution}$ where \text{Resolution} is the width of the equivalent digital image in pixels and \text{Width} is the width of the film image in millimeters. For these figures, \text{Resolution} was assigned a value of 75 representing the calculations for 75dpi images. If we were calculating for 300dpi images, we would assign \text{Resolution} to a value of 300, then do the calculations. The same formula applies for the length of the image.

\subsection{Megapixels and The Printed Photograph}

So far, we discussed the size of the image from digital cameras and film equivalents. Let us now discuss the sizes of images needed to send to a printer (or photo-finishing kiosk). In the last table, I included image sizes for 4x5, 5x7 and 8x10. The image sizes reflect scans at 75, 150, 300, 600 and 1200dpi.

Today’s inkjet printers can print with as much resolution as 2880dpi, with 300dpi and 600dpi being the most common printing resolutions. In general, the higher the resolution of the printed photograph, the more detail that can be shown, and the better quality the image will be.
Word Processing With LaTeX (A Real World Example)

\begin{tabular}{m\hline 8x\hline 8.5x11 & 1275x1650 & 1530x1980 & 2550x3300 \& 3060x3960 \hline \end{tabular}\end{center}

\bigskip \subsection{This is all Theoretical}

Of course, these figures calculated based on the full usage of available space in each size of the photograph. These figures do not take into account differences in hardware design (or glitches for that matter) or software handling of digital images.

\subsection{How Many Pictures Can I Take With a Memory Card?}

When digital cameras first came out, this was a \textbf{the most frequently asked question} I was asked by the general public.

Unlike conventional film, there is \textbf{no definitive answer} to this question. With a roll of 35mm film, you got 12, 24 or 36 exposures on a roll\footnote{Even here, there are some exceptions. If you had a Bolex B3 camera, you could get an additional three exposures on a roll because of the short path between the 35mm cartridge and the spool on the other side of the camera.} The size of the negative (or transparency) image was always the same, namely 24mm\times36mm\footnote{There were very few 35mm cameras ever made that took half-frame exposures. For those cameras, the image size was 18mm\times24mm.}

Digital cameras...

\begin{itemize}
  \item can take more than one size of digital picture.
  \item will tell you how many pictures you can store on the memory card you have inserted \textit{based on the current resolution} you set, usually the maximum available for the camera.
  \item will let you delete unwanted pictures without removing the memory card.
\end{itemize}

Even with all images the same physical size, no two pictures on a memory card take up the \textit{exact amount of space}. This depends on the method your camera uses to compress digital images to JPEG format\footnote{I have yet to see a digital camera that can compress to PNG or TIFF.}. In addition, your camera may add additional information to the image file\footnote{In the form of EXIF tags, a topic for another section on this document.}.

\begin{quote}
When a simple lens is held so that an image of the sun, or any very distant object, is focused sharply on a cardboard, the image is at the \textit{principal focus} and the distance from the cardboard to the lens is the \textit{focal length} of that lens.
\end{quote}

This is a much better definition of \textit{focal length} and \textit{focal point} than what you can get online by searching \textit{definition of focal length} on Google\footnote{This is more proof that you cannot always rely on the Internet for everything. The Wikipedia definition is even worse.}.

The term \textit{normal} as related to lenses simply means that what you see in the camera does not have to be magnified or expanded to capture what you see on film. \textit{This is the default setting for digital cameras when you power them up for use}.

What is considered a normal focal length was easy with conventional photography. Technically, a normal lens is calculated using the Pythagorean Theorem, that is for a normal lens,\begin{quote}$f1= \sqrt{iw^2+il^2}$\end{quote}where \textit{f1} is the normal focal length of the lens for the image size, \textit{iw} is the width of the image on film, and \textit{il} is the length of the image on film.

The 1940 edition of \textit{Elementary Photography} by C. B. Neblette, Frederick W. Brehm and Everett L Priest defines \textit{focal length} as follows:

\begin{quote}
When a simple lens is held so that an image of the sun, or any very distant object, is focused sharply on a cardboard, the image is at the \textit{principal focus} and the distance from the cardboard to the lens is the \textit{focal length} of that lens.
\end{quote}

This is a much better definition of \textit{focal length} and \textit{focal point} than what you can get online by searching \textit{definition of focal length} on Google\footnote{This is more proof that you cannot always rely on the Internet for everything. The Wikipedia definition is even worse.}.

The term \textit{normal} as related to lenses simply means that what you see in the camera does not have to be magnified or expanded to capture what you see on film. \textit{This is the default setting for digital cameras when you power them up for use}.

What is considered a normal focal length was easy with conventional photography. Technically, a normal lens is calculated using the Pythagorean Theorem, that is for a normal lens,\begin{quote}$f1= \sqrt{iw^2+il^2}$\end{quote}where \textit{f1} is the normal focal length of the lens for the image size, \textit{iw} is the width of the image on film, and \textit{il} is the length of the image on film.
The following is a table of film formats and their calculated normal lens sizes.

\bigskip
\begin{tabular}{|c|c|c|}
\hline
Film Size & Calculated Focal Length & Available Lenses \\
\hline
110 & 28.844102037119 & 24mm-28mm \\
\hline
126 & 33.941254969543 & 30mm-35mm \\
\hline
35mm (half-frame) & 30 & 30mm-35mm \\
\hline
35mm (standard) & 43.2666153955679 & 40mm-50mm \\
\hline
127 & 63.639618367893 & 65mm \\
\hline
120, 620, 220 (6x4.5) & 75 & 75mm-105mm \\
\hline
120, 620, 220 (6x6) & 84.8528137423857 & 75mm-105mm \\
\hline
120, 620, 220 (6x7) & 92.1954445729289 & 75mm-105mm \\
\hline
120, 620, 220 (6x9) & 108.16653826392 & 75mm-105mm \\
\hline
116, 616 (6.5x11) & 127.769323391806 & 125mm \\
\hline
\end{tabular}
\bigskip

Some medium format cameras utilize interchangeable film backs, hence the available lenses are the same for the 6x4.5 to 6x9 film sizes. Digital film backs for such cameras are available so the same camera can be used to produce both film and digital images (not to mention to protect the investment in the cost of the camera).

The ranges shown in the table represent the available lens focal lengths for film cameras that the manufacturer considers to be normal.

What is considered to be a normal lens for digital cameras has not been officially defined, though it has been reported to be between 6mm and 10mm for most inexpensive digital cameras. Digital SLRs typically have normal lenses of 35mm (the focal length and not the film size).

But then, how can we define a focal length for a normal lens if the image size of a digital camera is arbitrary?

As discussed in the last section, digital image sizes vary by the camera.

Your camera documentation (user manual) should have that information available (unless it is a bargain basement model without an LCD screen, in which case the lens is already fixed at what the manufacturer considers to be normal).

\subsection{Digital Cameras With Adjustable Focal Lengths}

Many digital cameras have a zoom feature\footnote{If you have a digital SLR camera, this is inherent with the purchase of additional lenses for that camera.}. There are two types of zooming on digital cameras\footnote{For digital SLRs, there is only the optical zoom, and that depends on the lens attached to the camera.}.

Digital zoom only simulates adjustment of the focal length by reducing the size of the digital image. Think of a rectangle expanding outward from and zooming in towards the center of the image. The image that appears in the rectangle is what the digital camera captures with digital zoom when you take that picture.

Cameras with optical zoom have an actual zoom lens that is either built in to the camera, or is a digital camera with interchangeable lenses. Zooming with one of these cameras is done without degrading the quality of the image.

\includegraphics[width=0.5\textwidth]{angle-of-view.png}

\section{Sources Used}

\begin{itemize}
\end{itemize}
Repo Review: FocusWriter

by CgBoy

This month I’m not going to do a quick review of several programs like I normally do. Instead, I have decided to do a detailed review of just one application each month. So today, I’m going to review a program called FocusWriter.

As the program’s description says, “FocusWriter is a simple, distraction-free writing environment.” Basically, it’s a fairly simple word processor designed to be less distracting than other software. It has a simple and customizable user interface. When you first open it, you’ll see a wooden desk background, with a page in the middle of the screen. If you move the mouse to the sides of the screen, various different toolbars will appear.

At the top of the screen is the main toolbar and menu bar. And at the bottom of the screen is another toolbar, which displays the currently open documents as tabs, the current word count, the percentage completed of the daily goal, and a clock. If you want, you can set FocusWriter to always show the user interface.

File Formats

FocusWriter doesn’t really support that many file formats. It can only open and save .odt, .fodt, .rtf, .docx, and .txt files. I did have a few problems with formatting when I opened a .odt document, though.

Themes

You can choose from a variety of themes that change how the background and page look in FocusWriter. You can also create new themes, if you don’t like any of the default ones. FocusWriter can be customized quite a lot. The top toolbar can be configured and customized, along with the program’s keyboard shortcuts (image on right).

Timers and Daily Goals

FocusWriter allows you to set a timer to go off at a certain time, and display a message. It also allows you to set a daily writing goal. That could be a goal a certain amount of minutes spent typing, say half an hour each day, or an amount of words typed everyday. Daily writing progress can then be checked from the “Daily Progress” tool.
Formatting

FocusWriter doesn’t have as many formatting options as other normal word processors. You can’t actually change the text size or font. You can change the text to Bold, Italic, etc., though. The alignment of the text can also be set. And FocusWriter also gives you the option to focus on only one line, three lines, or a whole paragraph of text at a time. And, it does have a spell checker.

Some of the other features of FocusWriter include the ability to play typewriter sounds whenever you type. When you open FocusWriter again, it automatically goes to the section of the document you were last at, so you can continue writing.

Summary

So, is FocusWriter any good as a distraction-free word processor? Yes, I’d say so. I actually wrote most of this article using it. It does obviously lack some features of other programs like LibreOffice, but remember, FocusWriter is meant to be distraction-free. So, will I use FocusWriter to write future Repo Review articles? No, I don’t think so. I’ll probably just keep using LibreOffice.
Playing Villagers And Heroes On PCLinuxOS

by Alessandro Ebersol (Agent Smith)

Villagers & Heroes is a 3D fantasy multi-platform MMO available for PC and mobile platforms.

Villagers and Heroes, formerly known as The Mystical Land, is a fantasy MMORPG sandbox with an incredible amount of customization and freedom to the player. It is a joyful game that does not take itself too seriously, but still has much to offer, including a robust crafting system and accommodation for players. Developed by Neonga AG, was the first to be fully integrated with Facebook.

Features

- Explore a gigantic world with thousands of quests with story, hundreds of exclusive zones, some unique Zingaras and countless treasures!
- Join with friends to attack dangerous lairs or defeat powerful Elder bosses!
- Ride to glory on mighty steeds, fierce wolves, threatening spiders, agile deer and many other exotic mounts!
- Forge your own weapons and clothing, wowing them with magic spells and heroic qualities you choose!
- Crafts, gardening, farming, fishing, mining, cooking, tailoring, smithing, master 10 unique skills!
- Build the most powerful village of the Seven Kingdoms with your friends and earn powerful rewards the from the village!
- Play perfectly between the phone and the PC, sharing the same account on any device, wherever you go!

Gameplay

There are four classes in Villagers & Heroes, each with different abilities and tactics of fighting: Wizard, Hunter, Warrior and Priest.

Wizards and Hunters are the only classes which can attack from afar, avoiding danger. Priests and Warriors have to come in melee range to attack. To combat this weakness, they have certain abilities and their class armor and weapons are stronger and give more protection, while Hunters and Wizards tend to be weak up close.

The character creation screen

When starting the game, players have a wide variety of options when it comes to character customization. Players can choose their gender, height, skin tone, personality, shape of the head and face, eye color, hair style and hair color before
choosing class options and skill. This allows players an amazing amount of customization, and is a welcome option for games that hardly allow any customization.

The controls are identical to most other MMORPGs. Players move using the WASD keys and use skills using 1-0. Keys as K and L open the skills and inventory windows, respectively, and players who are used to MMOs will easily be able to navigate quickly by V & H commands.

Many MMORPGs involve crafting, but require players to specialize in their creation and craft skills. V & H allows players to master the ten collection and craft skills, allowing players to play the way they want at all times. The gathering skills are Bug lore, fishing, mining and herbal knowledge. These skills often allow players to collect items that are useful on their own, such as food, but these skills mainly provide materials for the development of the crafting, cooking and potion making skills. The other skills are tailoring, cooking, crafts and smithing. These skills allow players to create powerful healing foods, create weapons and armor, and items that give bonuses.

But unlike other MMORPGs that make the professions, collecting and crafting a burden and prevent the progress of the player, V & M do everything in a way that is pleasant and fun, and does not hinder the story to develop.

One of the differentials of V & H is the possibility of building your home, join with friends and create an entire village and be the mayor of that village. Of course, not in the early levels, but when the character has enough experience, you can do such things. For fans of Minecraft, a real find.

Graphics

The graphics are beautiful, in a cartoony style, and very detailed. Sometimes just traveling by the beautiful surroundings of V & H is a rewarding experience because all landscapes are very colorful, full of life and details. For example, the wind blows and swings the tall grass, water has beautiful effects and the whole creates a sense of wonder.

Cities, fields and hills in V & M

Sound

Each interaction with NPCs involves a lot of talking, almost all with voice overs from actors. The story and dialogue are light, full of puns and comments in jest, that surely will make you smile more than once, and is wonderfully silly. Not only that, care for detail is very high: The steps of the player change sound, according to the type of soil the player steps on: sand, stones, asphalt and metal which is impressive. The songs are also very good, reflecting the action and what happens in the game.
Playing Villagers And Heroes On PCLinuxOS

Site: http://www.villagersandheroes.com/play-game/?continue=%2F

How to run it on PCLinuxOS?

Simple, you just need Play on Linux, and click install game.

Next, look for Villagers and Heroes, as shown below, and the entire process will be automatic.

![PlayOnLinux menu de instalação](image)

Play on Linux will download the game and all libraries (DX9) necessary for it to run, easing the process enormously.

So, enjoy Villagers and Heroes, and good adventures on the island of Ethos!

Verdict

Villagers and Heroes is a strange game, in this MMORPG’s games class. While most MMORPGs focus on the fight and in PvP game, Villagers and Heroes almost ignores them, and instead focuses on the creation and accommodation of players, creating homes and villages.

No darkness, no impending doom or monstrosities lurking. The game characters seem to be finding themselves in increasingly ridiculous and funny situations, that only a hero can solve.

V & H should be enjoyed without stress. And value is put in replayability, so much so that it includes an option of Rebirth; That is, you can send your character back to level 1 without losing items or elements of your home. After Rebirth, you get a XP boost and talent points to spend as you wish and your XP boost will not expire until you reach the level you were at the time of Rebirth. Different decisions can be made, perhaps on talents or professions without the need to create a new character, with no impact on your existing character. Rebirth is rewarded.

In short, V & H dodged successfully violence, gore and horror of the traditional MMO, creating a fun experience to be enjoyed with calm and patience, beautiful scenery to explore and beautiful music to accompany the player on his adventures.
**Salsa Turkey Meatballs**

**Ingredients**

1 pound ground turkey  
1/2 cup plain bread crumbs  
3 tablespoons milk  
1/2 teaspoon salt  
1/2 teaspoon Worcestershire sauce  
1/4 teaspoon pepper  
1 small onion, chopped (1/4 cup)  
1 egg  
1 jar (16 ounces) Thick 'n Chunky salsa  
2 medium green onions, thinly sliced

**Directions**

1. Heat oven to 400°F. Mix all ingredients except salsa and green onions. Shape mixture into twenty 1 1/2-inch meatballs.

2. Place in ungreased rectangular pan, 13x9x2 inches. Bake uncovered 20 to 25 minutes or until thermometer inserted in center of meatballs reads 165°F.

3. In 2-quart saucepan, place salsa and meatballs. Heat to boiling, stirring occasionally; reduce heat. Cover and simmer about 15 minutes or until salsa and meatballs are hot. Sprinkle with green onions.

**Tips:**

Bake meatballs (without salsa) in advance, then freeze up to 1 month. Heat salsa to boiling, then stir in frozen meatballs. Simmer uncovered 15 to 20 minutes or until meatballs are heated through.

Serve with hot cooked rice and a tossed green salad.
Users Don't

Text
Phone
Web Surf
Facebook
Tweet
Instagram
Video
Take Pictures
Email
Chat

While Driving.

Put Down Your Phone & Arrive Alive.
**Tip Top Tips: Slow KDE Application Open/Save Dialogs Workaround**

**Editor's Note:** Tip Top Tips is a monthly column in The PCLinuxOS Magazine. Each month, we will feature – and possibly even expand upon – one tip from the PCLinuxOS forum. The magazine will not accept independent tip submissions specifically intended for inclusion in the Tip Top Tips column. Rather, if you have a tip, share it in the PCLinuxOS forum’s “Tips & Tricks” section. Your tip just may be selected for publication in The PCLinuxOS Magazine.

This month's tip comes from PCLinuxOS forum member dm+.

I have been using Mate for everyday work, but some of my favorite applications are part of the KDE habitat. The problem is at one point in time, all KDE applications started being very sluggish when opening or saving files, taking up to one minute (or more) to display file open or 'save as' dialogs.

Apparently, the KDE file open/save dialogs attempt to access the file nodes in the 'Places' or 'Bookmarks' list(s) to get their status (capacity, used space, etc.).

If the file node is not there, as in the case of network shares or removed folders, the operation takes a long time to time out before showing the contents of the open or save as dialog window (over one minute in some cases).

This has been observed while using kwrite, Kate and okular, but all applications employing the kfile library should be affected.

**Workarounds:**

- Check the "Places" or "Bookmarks" panels for inaccessible or non-existent folders and remove them from the list.

- If your main desktop is KDE, and this is experienced while using LibreOffice or OpenOffice:
  - Select "Use LibreOffice dialogs" (Tools -> Options -> LibreOffice -> General -> Open/Save dialogs).
  - Similarly, for OpenOffice (Tools -> Options -> OpenOffice -> General -> Open/Save dialogs; select 'Use OpenOffice dialogs'.

**References:**

- [Bug 373352] file open/save/save as dialog is slow if automounted filesystems are not available (nfs, smb, etc)
- [Bug 363185] file selection dialog very slow when samba shares are not present
- [Bug 366720] stat() on faulty nfs mountpoints causes all KDE applications to deadlock (kill -9 required).

---

**Does your computer run slow?**

**Are you tired of all the "Blue Screens of Death" computer crashes?**

Are viruses, adware, malware & spyware slowing you down?

Get your PC back to good health TODAY!

Get

Download your copy today! FREE!
Get Windows Off My Machine

by phorneker

This is a parody of a 70's funk and disco tune from the Brothers Johnson called “Get the Funk Outta My Face”

Get Windows off my machine
Get Windows off my machine
Get Windows off my machine
Get Windows off my machine

We don’t like Windows
We’re not going to use it
PCLOS is a thing that all of us release

You do have to get it
Then you need to boot it
Then you’ll get a desktop you can use

We don’t like that Office
We don’t have to run it
Libre’ is a thing that all of us release

Get Windows off my machine
Get Windows off my machine
Get Windows off my machine
Get Windows off my machine

We don’t run that Outlook
We do run Thunderbird
We don’t even hack Office.com

Get Windows off my machine
Get Windows off my machine
Get Windows off my machine

We don’t like Windows
We’re not going to use it
PCLOS is a thing that all of us release

You do have to get it
Then you need to boot it
Then tell us our OS did you some good
**SCRAPPLER RULES:**

1. Follow the rules of Scrabble®. You can view them [here](#). You have seven (7) letter tiles with which to make as long of a word as you possibly can. Words are based on the English language. Non-English language words are NOT allowed.
2. Red letters are scored double points. Green letters are scored triple points.
3. Add up the score of all the letters that you used. Unused letters are not scored. For red or green letters, apply the multiplier when tallying up your score. Next, apply any additional scoring multipliers, such as double or triple word score.
4. An additional 50 points is added for using all seven (7) of your tiles in a set to make your word. You will not necessarily be able to use all seven (7) of the letters in your set to form a “legal” word.
5. In case you are having difficulty seeing the point value on the letter tiles, here is a list of how they are scored:
   - 0 points: 2 blank tiles
   - 1 point: E, A, I, O, N, R, T, L, S, U
   - 2 points: D, G
   - 3 points: B, C, M, P
   - 4 points: F, H, V, W, Y
   - 5 points: K
   - 8 points: J, X
   - 10 points: Q, Z
6. Optionally, a time limit of 60 minutes should apply to the game, averaging to 12 minutes per letter tile set.
7. Have fun! It's only a game!

---

**SUDOKU RULES:** There is only one valid solution to each Sudoku puzzle. The only way the puzzle can be considered solved correctly is when all 81 boxes contain numbers and the other Sudoku rules have been followed.

When you start a game of Sudoku, some blocks will be prefilled for you. You cannot change these numbers in the course of the game.

Each column must contain all of the numbers 1 through 9 and no two numbers in the same column of a Sudoku puzzle can be the same. Each row must contain all of the numbers 1 through 9 and no two numbers in the same row of a Sudoku puzzle can be the same.

Each block must contain all of the numbers 1 through 9 and no two numbers in the same block of a Sudoku puzzle can be the same.

---

**Download Puzzle Solutions Here**

Possible score 198 average score 139.
PCLinuxOS Word Find: September 2017
Job Skills

Adaptable  Ambitious
Analytical  Articulate
Attentive  Consistency
Communication  Competitiveness
Cooperation  Creativity
Dependability  Education
Effort  Energy
Enthusiasm  Flexibility
Follow-through  Friendliness
Good judgment  Leader
Listener  Management skills
Organization  Passion
Positive attitude  Promptness
Respectfulness  Quick learner
Realistic expectations  Responsibility
Self-starter  Sense of humor
Time management  Well-organized
Working well with others

Download Puzzle Solutions Here

PCLinuxOS Magazine
Job Skills Crossword

1. Largest of the land animals
2. Derived from Latin for “water horse”
   - very large mammal.
3. Tux’s species!
4. Close relative of the leopard
   - many are black
5. One with the longest neck
6. USA's national symbol
7. Sea creature which looks delicate but can be poisonous
8. Wolf-like predator in the plains
9. Bird whose fan-like feather display is large and beautiful.
10. Marsupial with strong tail used for balance
11. Tasty shellfish
12. Spotted cat known for its speed
13. This long-legged bird gets its pink color from the vegetation it eats.
14. As large as a hippo, but with a big horn on its nose.
15. Largest sea creature
16. Same species as a crocodile
17. Huge for a bird
18. Sea creature with 8 arms

Download Puzzle Solutions Here
Cryptograms

From the computer of ms_meme:
Everyday I work the daily crosswords and cryptograms in the paper. I like the cryptograms as they are wise/silly sayings of famous people. The magazine article about secure passwords made me think of the cryptograms. We have people in the forum who are always posting famous/wise/silly sayings. I have made a few into cryptograms, and hope our readers will enjoy them.

"ISBHIVT & IWS UVONVESTH HFPZMH DCNS V XPHCO ETFPU."

IZ_IPB

W = H

Download Puzzle Solutions Here
More Screenshot Showcase

Posted by luikki on August 14, 2017, running KDE.

Posted by jogurtmen on August 3, 2017, running Mate.

Posted by francesco_bat on August 15, 2017, running icewm.

Posted by meemaw on August 2, 2017, running Xfce.