

# The PCLinuxOS magazine

Volume 153

October, 2019



*Inkscape Tutorial:  
Five Inkscape Essentials*

*Short Topix: Julian Assange To  
Remain Jailed Pending  
Extradition To U.S.*

*Analysis: The Fall Of Stallman*

*Casual Python, Part 9*

*Firefox Task Manager:  
Find Your Resource Hog*

*Mind Your Step: Part 2*

*PCLinuxOS Recipe Corner:  
Slow Cooker Salsa Chili*

*PCLinuxOS Family Member  
Spotlight: oulik.jan*

*Accessing Special Unicode  
Characters, Made Easy*

*Repo Review: QPhotoRec*

*And more inside ...*

# In This Issue ...

- 3 From The Chief Editor's Desk ...**
- 5 Short Topix:**
  - Julian Assange To Remain Jailed Pending Extradition To U.S.***
- 7 Screenshot Showcase**
- 8 Analysis: The Fall Of Stallman**
- 11 Screenshot Showcase**
- 12 ms\_meme's Nook: Hey, Hey, Texstar!**
- 13 PCLinuxOS Recipe Corner: Slow Cooker Salsa Chili**
- 14 Inkscape Tutorial: Five Inkscape Essentials**
- 17 Screenshot Showcase**
- 18 Casual Python, Part 9**
- 27 PCLinuxOS Family Member Spotlight: oulik.jan**
- 29 Screenshot Showcase**
- 30 Accessing Special Unicode Characters, Made Easy**
- 33 Mind Your Step, Part 2**
- 35 Screenshot Showcase**
- 37 Firefox Task Manager: Find Your Resource Hog**
- 38 Repo Review: QPhotoRec**
- 39 Screenshot Showcase**
- 40 ms\_meme's Nook: Come Fly With Me To Linux Land**
- 41 PCLinuxOS Recipe Corner Bonus:**
  - Make Ahead Slow-Cooker Asian Peach Chicken Thighs***
- 42 PCLinuxOS Puzzled Partitions**
- 46 More Screenshot Showcase**

## The PCLinuxOS magazine

The PCLinuxOS name, logo and colors are the trademark of Texstar.

The PCLinuxOS Magazine is a monthly online publication containing PCLinuxOS-related materials. It is published primarily for members of the PCLinuxOS community. The magazine staff is comprised of volunteers from the PCLinuxOS community.

Visit us online at <http://www.pclosmag.com>

This release was made possible by the following volunteers:

**Chief Editor:** Paul Arnote (parnote)

**Assistant Editor:** Meemaw

**Artwork:** ms\_meme, Meemaw

**Magazine Layout:** Paul Arnote, Meemaw, ms\_meme

**HTML Layout:** YouCanToo

### Staff:

ms\_meme

Meemaw

Gary L. Ratliff, Sr.

Daniel Meiß-Wilhelm

daiaishi

Cg\_Boy

YouCanToo

Pete Kelly

Smileeb

Alessandro Ebersol

### Contributors:

oulik.jan

The PCLinuxOS Magazine is released under the Creative Commons Attribution-NonCommercial-Share-Alike 3.0 Unported license. Some rights are reserved.  
Copyright © 2019.





# From The Chief Editor's Desk ...

Sweet Sixteen.

Yep. That's how old PCLinuxOS will be on October 24, 2019. Over those 16 years, we've seen many changes, and yet there's a lot that remains the same and a constant. People, as always, have come and gone. KDE has always been the "main" desktop offering. Over the years, PCLinuxOS has also been available with FluxBox, Openbox, Xfce, Gnome 2, Mate, LXDE, LXQt, Cinnamon, Trinity, JWM, Enlightenment, and other assorted desktop "flavors."



The most consistent constant, though, has been Texstar at the helm, guiding this awesome Linux distro through sometimes turbulent Linux waters. Tex has managed to keep PCLinuxOS free from the systemd curse that has plagued most other Linux distros, and has vowed to continue to do so for as long as he can. Today, PCLinuxOS is one of the very few systemd-free Linux distros in existence.

Those of us who have been around for a while (I've been here for about 12 years) have heard the story about the origins of PCLinuxOS before. I doubt any of us ever get tired of hearing the story again. Meanwhile, there are always new people who have never heard the story of how PCLinuxOS came about. What follows is Texstar's description of the events.

*In the summer of 2003, I became interested in Live CD technology after looking at Knoppix and a fresh distribution from a fellow named Warren, called Mepis. I was interested in helping Warren with Mepis at the time, but I had no clue how to build DEB files. Coming from 5 years of packaging RPMS and not really wanting to learn a new packaging system, I happened to come across a South African fellow by the name of Jaco Greef. He was developing a script called mklivecd and porting it to Mandrake Linux. I, along with Buchanan Milne (Mandrake contributor) and a few others, began working with Jaco to help debug the scripts. I got an idea to make a livecd based on Mandrake Linux 9.2, along with all my customizations, just for fun. I had previously provided an unofficial 3rd party repository for the users of Mandrake for many years, but had since parted ways. Since Mandrake was a trademarked name, myself and others decided to name the Live CD after our news site and forum, pclinuxonline, thus PCLinuxOS.*

*Preview .3 was my first attempt to make a livecd. I distributed it initially to about 20 people to get their*

*reaction and feedback. Everyone who tested it loved the livecd but there was one thing missing. There wasn't a way to install the thing to the hard drive! srlinuxx from tuxmachines.org came up with a novel way to copy the livecd to the hard drive and posted it on our forums. Jaco utilized this information and inspiration from the Mepis installer and wrote a pyqt script to make the Live CD installable, thus the birth of a new distribution.*

*On October 24, 2003, PCLinuxOS Preview .4 was released as a fork of Linux Mandrake (Mandriva) 9.2 utilizing mklivecd scripts from Jaco Greef, a multimedia kernel from Thomas Buckland (2.4.22-tmb) and a customized KDE (3.1.4-tex). Preview .5 through .93 were built upon on previous PCLinuxOS releases. After three years of updating one release from the other using the same gcc and glibc core library, we found too many programs would no longer compile or work properly against this aging code base.*

*In November 2006, we utilized a one time source code snapshot from our friends at Mandriva to pull in an updated glibc/gcc core and associated libraries. We spent the following 6 months rebuilding, debugging, customizing, patching and updating our new code base. We pulled in stuff from our old code base, utilized patches/code from Fedora, Gentoo and Debian just to name a few. This is why you will never see me distro bashing, as it would be hypocritical to do such a thing. We are still dependent in many areas on other distros development processes due to our limited but hard working volunteer development team.*

*On May 20th, 2007, we felt we had reached a pretty stable base and released PCLinuxOS 2007. It utilized our own kernel from Oclient1, KDE built by MDE developer Ze, updated mklivecd scripts from IKerekes & Ejtr, a heavily patched Control Center, graphics from the PCLinuxOS beautification team, and many application updates from Thac and Neverstopdreaming. Development continues as*

work is being done for a Minime release and an international DVD. A future release of PCLinuxOS will feature an updated kernel, KDE 4, fresh Xorg server and all the latest applications. All in all it has been a great ride and we have made many friends along the way. Some have gone on to other distributions and many are still here from our first release. As I've always said, we're just enjoying Linux technology and sharing it with friends who might like it too. We hope you have enjoyed the ride as well.

Today, PCLinuxOS only supports 64 bit processors, having dropped support for 32 bit processors a couple of years back. Most other Linux distros have also dropped support for 32 bit processors in recent years, so PCLinuxOS isn't bucking the trend. There are exceptionally few Linux distros that support 32 bit processors today, in fact.

It's easy to understand, too. I can't imagine maintaining two versions (32 bit and 64 bit) of each package, and ensuring that both work as they are supposed to before being released into the stable repositories. Since most computers sold in the past eight to ten years sport 64 bit processors, most of the computers running PCLinuxOS today will have 64 bit processors. It only makes sense, given the limited resources of PCLinuxOS's relatively small, volunteer development team, to focus on support for the processor that most people are going to be using.

PCLinuxOS doesn't have any millionaire/billionaire donors, and isn't supported by any large corporation(s). It's just a bunch of people interested in Linux and interested in putting out the best rolling release Linux distro that they can. Without a doubt, they have succeeded famously. And all while relying on the generosity and small donations from its user base to keep the bills paid and to keep things going.

PCLinuxOS's motto (for as long as I can remember) has been "radically simple." Even today, that motto shines bright. Everything. Just. Works. And it works exceptionally well. In fact, it works so well that

sometimes it's boring. But I'll take being bored over the constant onslaught of viruses, malware, spyware, crapware, and any other kind of "ware" with a negative connotation, any day!

So, Happy Birthday, PCLinuxOS! May there be many, many more birthdays in the years to come!

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

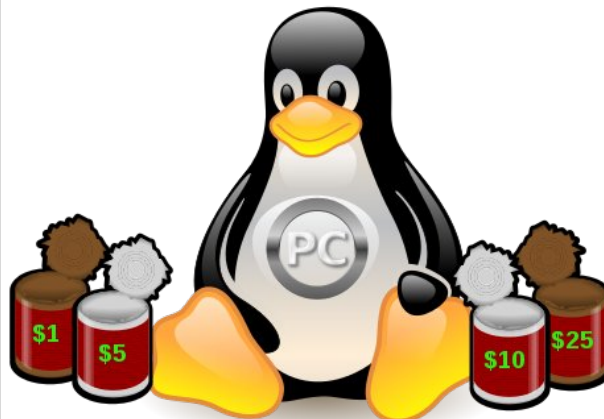


## Donate To PCLinuxOS

**Community Supported.  
No Billionaires/Millionaires.  
No Corporate Backing Or Funding.**

**Click [here](#) to make a one-time donation  
through Google Checkout.**

**Or, click one of the amounts down below  
to make a monthly, recurring donation.**



## 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 person 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.

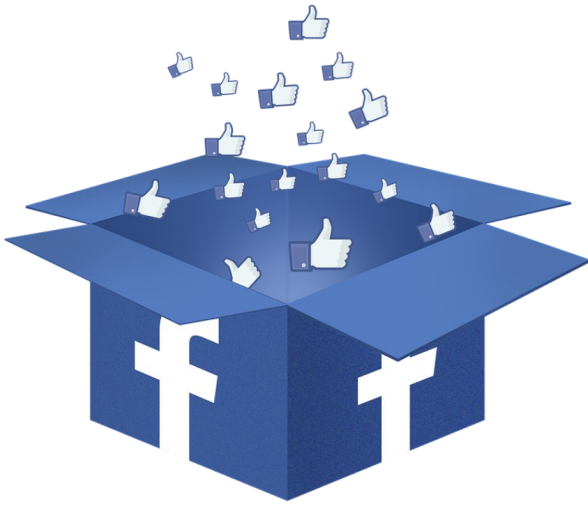




# Short Topix: Julian Assange To Remain Jailed Pending Extradition To U.S.

by Paul Arnote (parnote)

## Over 400 Million Facebook Users' Phone Numbers Exposed



According to a TechCrunch [article](#), over 419 million phone numbers linked to Facebook accounts were found on an unsecured, exposed server. This included 133 million U.S. Facebook users, 18 million U.K. Facebook users, and another 50 million Facebook users from Vietnam. The server wasn't even password protected, allowing basically anyone unfettered access to the data.

The exposed records linked a user's unique Facebook ID number to their phone number. In some of the records, the data included the user's name, gender and country of residence. Ironically, user phone numbers have not been publicly available for the past year since Facebook restricted access to them. And yet here they sit, unprotected

on an exposed server. The database has since been taken offline.

Facebook has maintained that the data exposed was "old data," scraped at a time from before Facebook restricted access to user phone numbers and may have been placed on the server by others looking to store it for later retrieval. Still, many persons' phone information does not change all that often.

Sanyam Jain, a security researcher and member of the GDI Foundation, discovered the database. He contacted TechCrunch after he was unable to find its owner. In his examination of the database, he was even able to find profiles – with their associated phone numbers – of several celebrities.

Just in case you ever needed yet another reason to NOT have a Facebook account ... here it is.

## Library of Congress: Nearly 12,000 Public Domain Photos of Roadside Americana

The U.S. Library of Congress has [released](#) nearly 12,000 images from John Margolies into the public domain. The library purchased the rights of the photos from the architectural critic and photographer, who spent forty years documenting his travels along U.S. highways. The photographs include billboards, drive-in theaters, banks, city halls, movie theaters, diners, novelty buildings, miniature golf courses and other roadside attractions.

Most (if not all) of the images are released as either JPG or TIFF file copies of image transparencies. The original transparencies have subsequently been placed into controlled-climate storage to help insure preservation. Each image is offered in a high and

low resolution JPG, and a high and low resolution TIFF file.



Many unique buildings, structures and attractions still exist along the roads and highways of America. These buildings, like the one pictured above of the old Teepee Junction gas station in Lawrence, Kansas at the junction of U.S. Highway 40 and U.S. Highway 59, have a lot of character.

What's even better is that the images were released into the public domain, so you can use the images as you see fit, even for commercial applications.

## Julian Assange To Remain Jailed Pending Extradition To U.S.



Julian Assange's 50 week jail sentence for jumping bail by hiding out in the Ecuadorian Embassy for seven years was scheduled to end September 22, 2019. However, U.K. Home Secretary Sajid Javid signed an order this past June allowing Assange's extradition to the U.S. over hacking allegations.

The 48 year old Australian appeared before the court via a video link, wearing a loose fitting t-shirt.

## Short Topix: Julian Assange To Remain Jailed Pending Extradition To U.S.

During the proceedings, district judge Vanessa Baraitser told Assange:

*"You have been produced today because your sentence of imprisonment is about to come to an end. When that happens your remand status changes from a serving prisoner to a person facing extradition."*

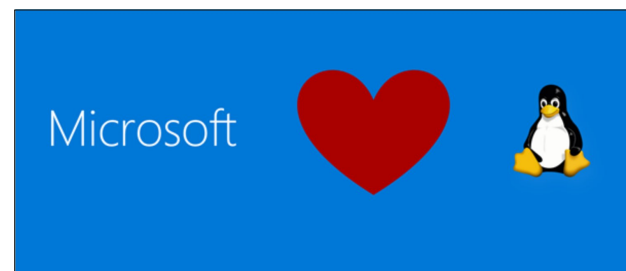
*"Therefore I have given your lawyer an opportunity to make an application for bail on your behalf and she has declined to do so, perhaps not surprisingly in light of your history of absconding in these proceedings."*

*"In my view I have substantial ground for believing if I release you, you will abscond again."*

When Assange was asked if he understood what was happening, he replied "Not really. I'm sure the lawyers will explain it."

An administrative hearing on the case is expected on October 11, and a case management hearing on October 21. The final extradition case isn't expected to happen until February, 2020.

### Windows 11 Running Linux Kernel?



Say what? Microsoft Windows 11 running a Linux kernel? What'chu smokin'? Steven J. Vaughan-Nichols wrote such a speculative [article](#) for ComputerWorld.

And actually, it makes perfect sense. Look at the unmitigated disaster that the Windows 10 updates

have been. Week after week after week, we hear about how the Windows 10 updates have wrecked users computers or have otherwise gone seriously awry. Most (if not all) Linux users are Windows refugees, usually fleeing from the lack of desktop choice under Windows, the never-ending assault of virii and virus scanners, the endless battle with malware, etc., etc., etc. The list is nearly as long as the number of Linux users.

Replacing the NT kernel, which is basically rotten, with the Linux kernel is certainly doable. Vaughan-Nichols makes the argument that using the Linux kernel that is passionately and enthusiastically upkept by an army of programmers from around the world makes perfect sense. He goes on to argue that many Windows users won't even have to be aware that Windows is running on a Linux kernel, as Windows can still be made to look like Windows. But the insides, the very core, will get an upgrade in stability and security.

Sure, it sounds crazy. But who could have predicted that Microsoft would go from wanting to bury Linux and calling it a cancer under Steve Ballmer, to expressing love for Linux under Satya Nadella? Who could have predicted that Microsoft would open its extensive patent library to Linux and the FOSS community – for free?

Vaughan-Nichols goes on to point out that Microsoft could release its own version of Linux today, if it chose to. There's nothing to stop them. But Microsoft developers have been busy laying the groundwork with the Windows Subsystem for Linux (WSL), mapping Linux API calls to Windows, and vice versa.

It would be perfectly understandable that Linux users might want to approach this topic cautiously. After all, it wasn't all that long ago that Ballmer and friends were looking to torch, hang, stab, shoot, and otherwise kill and bury Linux. According to Ballmer, Linux and FOSS were a "cancer." Many Linux users view Microsoft in that context. They are wary to invite the wolf into the lamb's enclosure.



Microsoft Wants You To Learn Python ... For FREE!



The PCLinuxOS Magazine has been running an article series by critter (Pete Kelly) for much of the past year to teach you the basics of Python programming. Now, Microsoft is also wanting to teach you Python programming. They have launched a 44 part [series](#) on YouTube, called Python for Beginners. Each “episode” lasts for three or four minutes, hosted by two self-described Microsoft geeks who love programming and teaching.

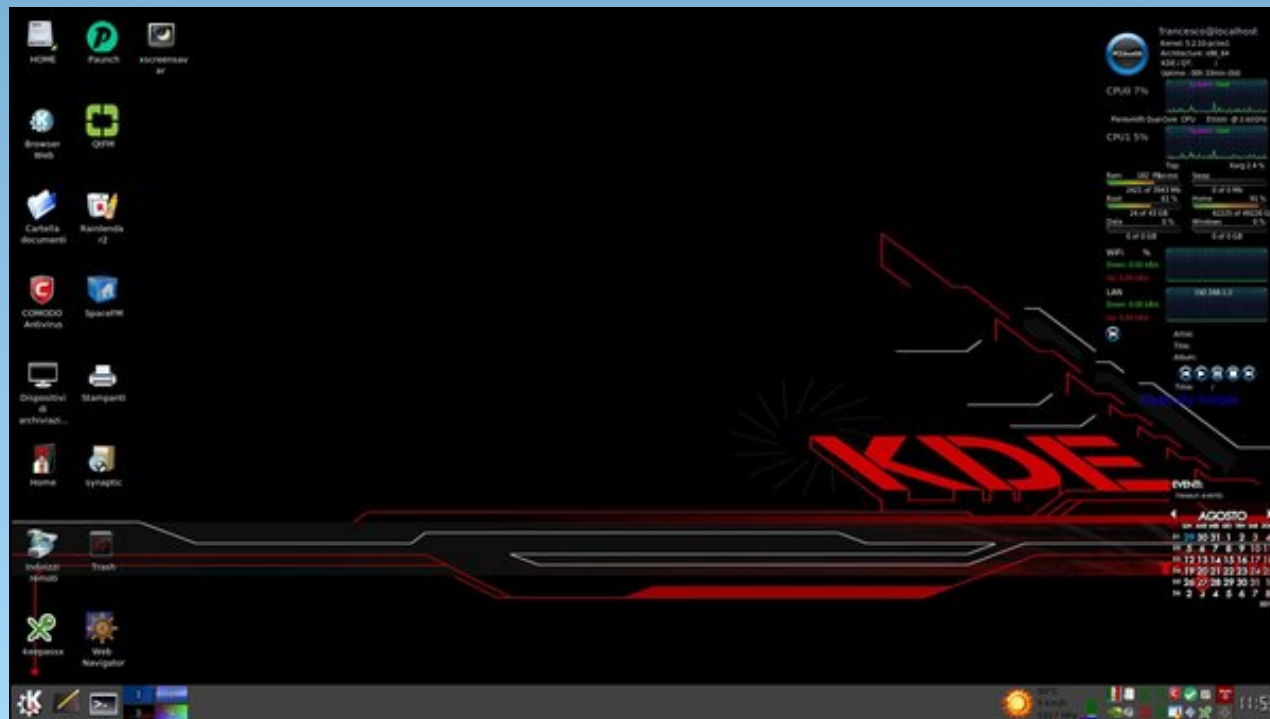
While the course focuses on Python 3.x, users of Python 2.x should also find value. Microsoft has also published additional [resources](#) for the course on GitHub, including slides and code samples. The Microsoft course focuses on AI machine learning.

Some of the latest polls have placed Python programmers among those most in demand in the IT world. Recently, the Python programming language was rated the most popular, followed by C++, Java, then C.

So, if you're looking for something to supplement critter's excellent and informative tutorial articles, this video series may just be what you are looking for.



## Screenshot Showcase



*Posted by francesco bat, September 1, 2019, running Trinity.*

# Analysis: The Fall Of Stallman

by Alessandro Ebersol (Agent Smith)



Figure 1: Super Stallman being taken away by the PC thought police

Richard Matthew Stallman, the founder of the GNU movement, resigned from his position as president, and left the board of the Free Software Foundation, on September 16, 2019, after external pressure over comments he would have made about pedophilia and Jerry Epstein.

## But what actually happened?

Richard Stallman sent emails to an MIT CSAIL mailing list earlier this month. In the emails, Stallman tried to defend the reputation of late MIT teacher Marvin Minsky, with whom Epstein's victim Virginia

Giuffre said she was forced to have sex during a trip to the Virgin Islands at the age of 17. Well, Stallman argued, on that MIT mailing list that using the term rape would be very serious (defending the memory of his former teacher, and, I believe, friend).

Now Stallman is a man almost completely detached from our reality. Anyone who knows him says he's almost an Asperger, so his opinions should not be taken so seriously given his background.

## But the media responded swiftly and ruthlessly.

How did the sites report Stallman's appeal on the CSAIL list?

The VICE [website](#) put it this way: "Famous computer scientist Richard Stallman described Epstein's victims as 'totally willing'".

And, such headlines and articles were reproduced ad-infinity on the internet, amplifying a very poorly told story.

Well ... what exactly does this really have to do with what he wrote?

In defending his former teacher, and getting into this discussion of a notorious pedophile, Stallman has already made a mistake. He should not have gotten involved in this, but, as the media described him, it was a colossal attack on Stallman's person, a 100x worse mistake.

To say that Stallman defended Epstein for comments he made about his former teacher are from an oceanic distance. And, he commented, angrily on his blog, about the injustices he suffered [online](#).

Unfortunately, his defenses were forgotten, and all the media preferred the scandal to the truth.

And an [online campaign](#) at has put pressure on him to leave, and has succeeded. He did not resist the pressure and resigned from both the MIT and the FSF.

## So he was a pedophile and defended Epstein?

Negative. It never happened. In fact, who really had ties with that citizen Epstein was [Bill Gates](#), who, according to emails obtained exclusively from The New Yorker, Epstein would have instructed Bill Gates to donate \$2 million to a MIT research lab in October 2014. The directors of MIT Media Lab delivered the emails, and they clearly link Gates to Epstein.

However, this connection goes beyond donation, as both Gates and Epstein had a common interest in eugenics, a perverted form of science that seeks to genetically improve the human population by getting rid of undesirable ones (who was also interested in that? Hmmm, ahhh, that [Austrian](#) guy!)

We can even speculate that the attack on Stallman's person was a way to get the public's attention diverted away from Gates, who really had a connection with Epstein.

Now, let's move a little away from the passion that this theme arouses. Let's look at the Stallman person. And, let's think for a moment.





**Stallman has always been awkward, and his comments have not helped him, on the contrary.**

Here justice must be done: Stallman was never aware of the danger he was putting himself into, engaging with sexual issues (sex with minors, sexual violence and so on). Several times he made comments on his personal blog that did not go well. In [2003](#), in [2006](#), in [2013](#), etc.

But what he did was, without any tact, expressing his ideas and opinions, without taking into account that the era of freedom of expression was already over. Yes, nowadays, with the PC thought police, who are always on the lookout for any subject to fire their cannons on, that free speech is gone. If an opinion expressed goes against the prevailing zeitgeist, the subject is in hot waters. And, add to that a corporate yellow media, plus fake news, and you get a sad scenario.

**The media has never been sympathetic to Stallman**

No, never. Either he didn't take a [shower](#), or he bit his [nails](#), but the fact that they treated him with disrespect, indifference or as a joke always occurred. I doubt if this was motivated (albeit in a hidden way) by Stallman's biggest enemy and what he stands for (the four freedoms), Microsoft.

**The cancer has never ceased to be cancer...**

Steve Ballmer once called GNU/Linux [cancer](#), and GPL a disease that would make everything [open source](#).

In fact, Ballmer was wrong to call free software open source, but back then they were little different. Not today. Today, the so-called "open source" is moving further and further from free software.

But let's get back to Microsoft, GPL, and Stallman.

At a time like today, where Microsoft increasingly declares its [love for Linux](#), we are always faced with open source apologists from Microsoft, and how it is committed to its advancement(open source) and improvement.

But what about free software? What about the four freedoms? Nothing. The company says nothing about it, and apparently to hear it doesn't exist. That is, Microsoft may love Linux, but it does not love the freedom of the GNU movement (which is intrinsically linked to the Linux kernel and the way it is developed). As a colleague posted on social network: "Microsoft may love Linux, but it doesn't love its users," a hard and strong truth.

And in a media maneuver, Stallman was invited to speak at [Microsoft](#), yet another advertisement for how the company "loves Linux" today. And interestingly, days after his visit to the company, all this scandal explodes ...

At a time like this, with so much "love" in the air, let us not forget that Ballmer himself said: "[I would love to see all the innovation of open source happen on Windows](#)".

What prevented that from happening? The four freedoms? Or his greatest advocate? Who knows.

The thing is, the biggest vocal advocate of the GNU copyleft licenses and the four freedoms was Stallman. Having resigned, who will play his part? And his vacant space paves the way for further corporate exploitation of all the work he has done and was committed to.

**Corporations are merciless, evil and heartless. But it's the free software folks who are almost always in the headlines...**

Yes, this is a recurring theme: Any advocates, or developers of free software, or free culture, are always in the sights, whether of the corporate media or "Defenders" of social justice movements.

Otherwise, let's see:

\* Jacob Appelbaum, one of the developers of the TOR protocol: [Allegations of sexual misconduct](#).

\* Julian Assange (Wikileaks): Accused of sexual violence in [Sweden](#).

\* Brendan Eich: Former CEO of Mozilla Foundation, had to resign for donating to the movement "Proposition 8" in [California](#) at a time when one of Eich's plans was to launch Firefox OS for mobile devices (phones, tablets).

\* Theodore T'so: Accused of an apologist for [sexual violence](#). Note that T'so was [against](#) Intel's [TPM platform](#) (which was Sharp's employer).

\* Linus Torvalds: All development of the Linux kernel has been branded as toxic by the extremely sincere personality of [Torvalds](#). And the public outcry was such that Linus had to adopt a COC and take a vacation from the direction of the development of the [Linux kernel](#).

So how many bad people in free software, huh?

But it's just not true. Steve Jobs was a nasty human being, and Microsoft is one of the most sexist companies there is. But no, the bad guys are these free software guys and activists.

**Jobs and Microsoft, exemplary citizens**

Steve Jobs was a scoundrel. He wanted nothing to do with his daughter for a long time.

Jobs denied his daughter Lisa's paternity for years. She and her mother ended up living on government welfare.

Daughter Lisa Brennan wrote a memoir about her relationship with her famous father, [Small Fry](#), where she describes Jobs as a sadistic man and a horrible father: sexually inappropriate, verbal and psychologically abusive, ruthless and cheap - with his time, money, emotions, attention.

But the media treated him like the Jesus of technology. Did anyone talk bad about him in the newspapers? On the Internet? In blogs?

While he was alive, no, but the media, so merciful about this "wonderful human" being, never spared free software activists. They always condemned them and lynched publicly.

And Microsoft is no different. In 2015, it was the subject of a labor lawsuit over sexual discrimination.

"Katherine Moussouris filed a complaint against the Seattle-based company, claiming that her supervisors disliked her "manner of style" and gave the promotions she was up to to her less qualified male colleagues, Reuters reported. She also received lower bonuses as retaliation for making complaints of sexual harassment. According to the complaint, Microsoft employees in Redmond, Washington often received lower performance ratings and were based on subjective observations."

And that didn't change with Satya Nadella, quite the opposite: In October 2015, he told, in a room full of technical women at the Grace Hopper Conference, that they should not ask for a raise, but instead have "faith that the system will provide the right raise." ARGHHHHH! Yeah, with Microsoft, only with a lot of faith, really.

He apologized later, but apologies don't change anything. Of course, having several technical news

sites in its pockets helps to silence criticism of Microsoft.

### And Stallman and his accuser?

Was Stallman a pedophile? No.

Encouraged pedophilia? He didn't.

Defended Epstein? No, and on his blog, made harsh criticism of the late sex offender.

But he paid dearly for no crime. He did nothing but try to defend a former teacher. But that can be a disaster these days because of digital lynch mobs.

And his accuser? Facing a strong public backlash, she took down her personal website (her website calling for Stallman's resignation is still online), but can still be found [here](#).

Did she act in good faith? Maybe, maybe [not](#).

The only thing I know is that she's a millennial, and they think they can destroy everything and build something new in its place, even better than there was. But I highly doubt that they can build anything better than it was (and still is), the GNU movement. RMS is not out of the game, and we look forward to his return to the field. Voices are rising in his defence, and, we can only hope this injustice to be undone quickly.



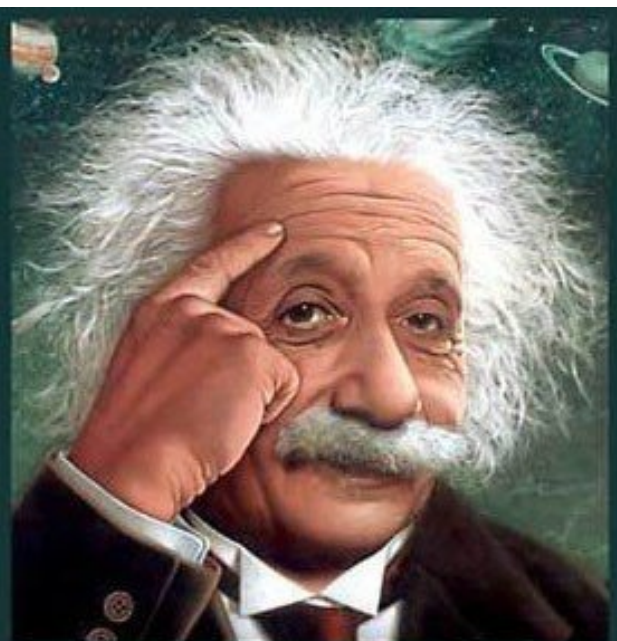
# PCLinuxOS Users Don't

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

*While Driving.*

*Put Down Your  
Phone & Arrive Alive.*





It's easier than  $E=mc^2$   
It's elemental  
It's light years ahead  
It's a wise choice  
It's Radically Simple  
It's ...





# PCLOS-Talk

Instant Messaging Server

Sign up TODAY! <http://pclostalk.pclosusers.com>

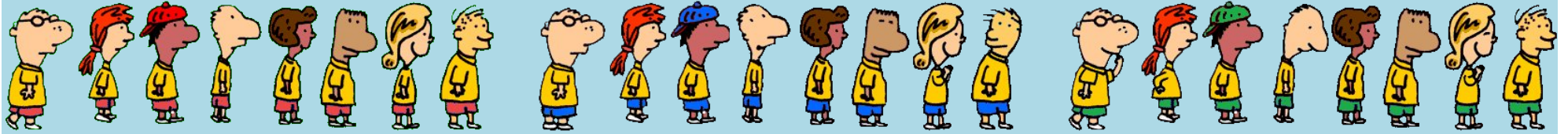


## Screenshot Showcase



*Posted by parnote, September 7, 2019, running Xfce.*

# *ms\_meme's Nook: Hey, Hey, Texstar!*



When the sun goes down we all come out  
Gather in the forum and begin to shout  
Hey Hey Texstar we are happy to be booting PCLinuxOS  
We are happy to be booting PCLinuxOS

This OS a real breakthrough, lordy I am telling you  
Always up to date, everything is first rate  
We are happy as a bull after eating a plateful  
For Texstar we will always be grateful

Easy to install it, radical is what we call it  
Texstar is made of zest and zeal, PCLinuxOS the ideal  
This OS will thrill you, always will fulfill you  
Download it now you will know boots so fast never slow

**MP3**

**OGG**



# PCLinuxOS Recipe Corner



*from the kitchen of  
youcantoo*

## Slow Cooker Salsa Chili

### Ingredients:

1 lb lean (at least 80%) ground beef  
1 medium onion, chopped (1/2 cup)  
2 cups Thick 'n Chunky salsa  
1 can (15 oz) tomato sauce  
1 can (4.5 oz) chopped green chiles  
2 teaspoons chili powder  
1 can (15 to 16 oz) pinto beans, drained, rinsed  
Shredded Cheddar cheese, if desired  
Sliced green onions, if desired

### Directions:

1. In 10-inch skillet, cook beef and onion over medium heat 8 to 10 minutes, stirring occasionally, until beef is thoroughly cooked; drain.
2. In 3 to 4 quart slow cooker, mix beef mixture and remaining ingredients except beans.
3. Cover; cook on Low heat setting 8 to 10 hours.
4. Stir in beans. Cover; cook on Low heat setting about 5 minutes longer or until beans are hot. Top with cheese and onions.

### Expert Tips:

Vary the taste by using black beans instead of pinto beans. If you like a milder chili, use mild salsa and a little less chili powder.

If you like, cook up a double batch of this hearty chili, then refrigerate or freeze half for another night. If you do make a double batch, use a 5 to 6 quart slow cooker.



**The PCLinuxOS  
Magazine**

**Created with  
Scribus**

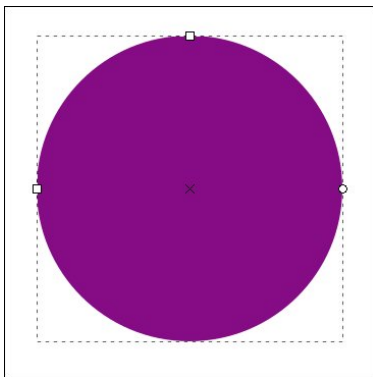
# Inkscape Tutorial: Five Inkscape Essentials

by Meemaw

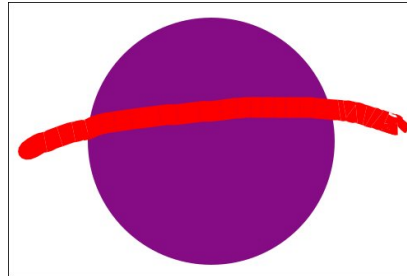
I found this tutorial while I was looking for some technical information about Inkscape, and I thought maybe there were more than just me who didn't know some of these methods. As I've said before, I don't know everything about Inkscape, so I'm constantly searching for articles or tutorials to expand my knowledge (I believe I actually only know enough to be dangerous!)

## Erasing in Inkscape

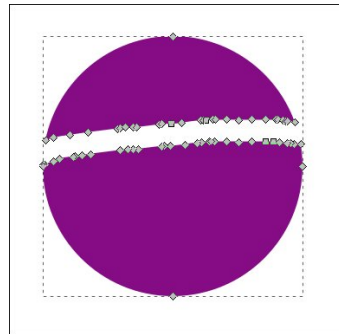
Every time I chose the eraser tool in Inkscape, I was disappointed because it didn't work like I thought it should. Just goes to show what I just said about not knowing everything. Technically, you can't "erase" with vector design. Everything is based on points and lines, so you'll actually have to add/change the points in order to give the illusion of erasing. Let's see how we do this using a circle:



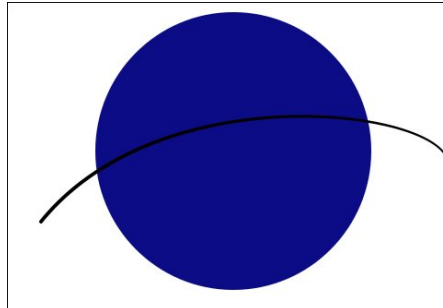
Choose the **Eraser** tool, change the mode to **Cut out from objects**, and set the brush **Width: 25**. Erase across your circle.



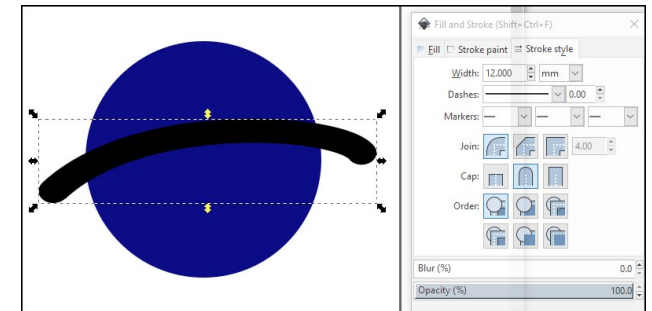
Yikes! That's not what I wanted at all! Using the eraser added about 100 nodes to our simple circle. It also looks really rough.



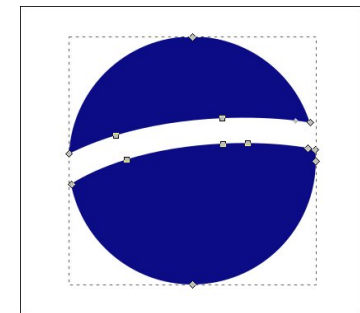
The best way to erase in Inkscape is to use boolean path operations. Instead of the eraser, we'll be using the Pen tool to draw a line similar to the previous eraser scrub. The good thing about this is that you can adjust your nodes before you erase.



We wanted a larger area erased, so we'll adjust the stroke width. While you have the line chosen, click on **Path > Stroke to Path**. Then, go to your **Properties** window and adjust the stroke size to however wide you want. I made mine 12 but you can make it any size.

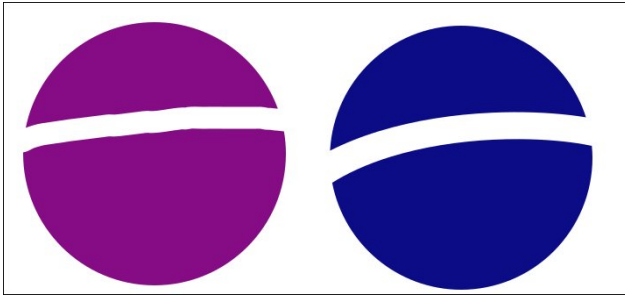


With both the circle and the line selected, go to **Path > Difference** complete your erasing. This looks much smoother than the first try.



On the next page you can see the difference in the two methods. The one on the left was done with the eraser tool and the one on the right was done with the pen tool and Path > Difference.

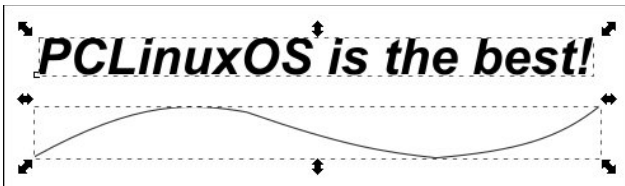




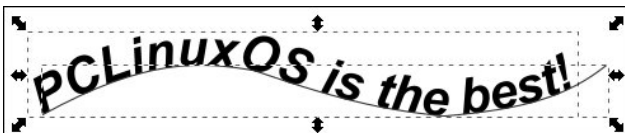
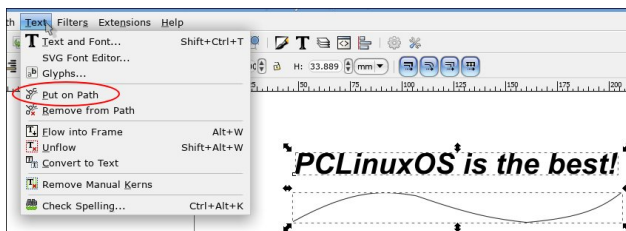
## Curving Text

Sometimes in your graphics creation, you want something that's not just straight text, but you want it to have some shape to it. You can draw a line to illustrate the curve you want your text to have, but what to do next? Fortunately, I found out, this is fairly easy in Inkscape.

Create your text, configuring it however you want, then draw a line with the bezier tool, and edit the line so it has the curve you need.



Select the text and the curve, then click on **Text > Put on Path**.



You can change the color of your line (or make it transparent) so it's not visible in your project. To move your text easily to where you want it, you should group the text and line as well.



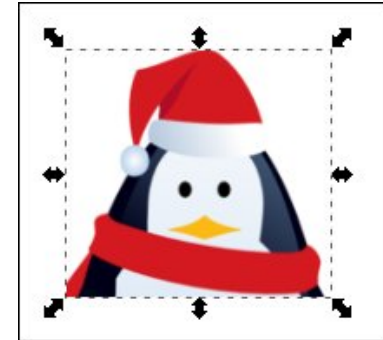
## Cropping an object

Sometimes you want to use a certain object but it needs to be cropped. In other programs, you just draw the rectangle and choose "Crop", but Inkscape doesn't work that way, so we'll have to do it differently.

Import the object you want to use, and size it down to the size you want. I have an image of a penguin I want to use for a Christmas project, but I only want the head, so I will import the image, then draw a rectangle (or even a circle) over the area I want to use. I made the rectangle a little transparent so I could see what I was cropping.



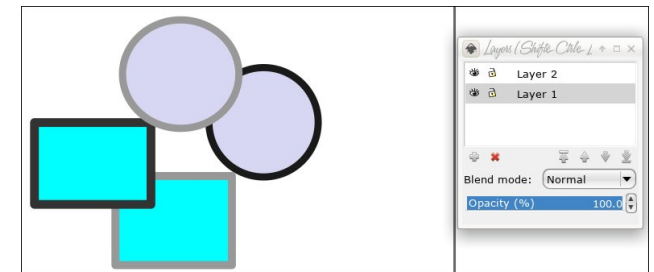
Select the image and the rectangle (making sure the rectangle is on top) and click on **Object > Clip > Set**. Everything within the rectangle will be visible, and everything else will be invisible. What makes this nice is that you don't have to stop, open another program, and crop your image there - you can stay in Inkscape.



## Merging Layers

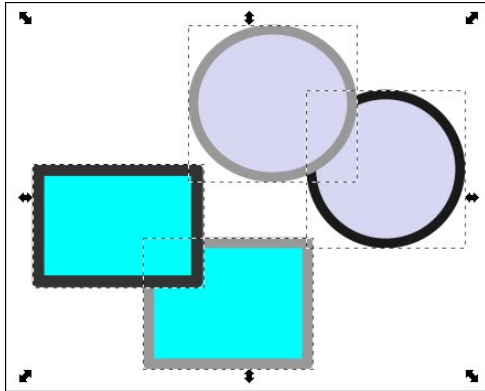
The first thing you need to know is that merging layers in Inkscape doesn't work like it does anywhere else - so let's look.

I have drawn some rectangles and circles in two layers. The rectangles are in layer 1 and the circles are in layer 2, above layer 1. Open the **Layers** toolbox to get started.



Make sure both of the layers you want to merge are unlocked. Select them all and use **<CTRL>+X** to cut them (next page, top right).





Now we have a couple of choices: we can paste **<CTRL>+V** our objects to an existing layer, or create a new one. I chose to paste them all on layer 1. I will then delete layer 2 since it has nothing on it.

## Tracing A Photo

This one might not be used so much, but it looked fun, so I left it in the list. We'll use **Trace Bitmap** to trace a photo.

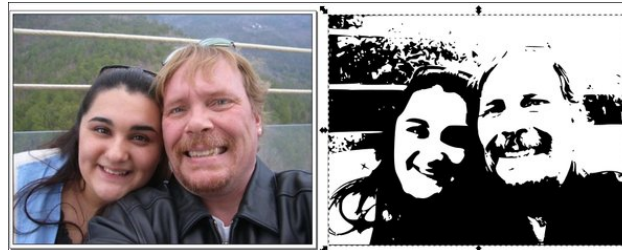
Import your photo and make sure it's selected. I used a photo of our favorite editor and his lovely wife.



Now, choose **Path > Trace Bitmap** and you'll get a menu. **Brightness cutoff 0.450** is default, so I left it there, then clicked OK. You might play with the settings if it's too dark or too light. I'm sure that it depends on the photo.



The generated vector drawing will be lying on top of the original photo, but you can grab it and move it. Here, I have moved it to the right of the original photo.



Of course, now that it's a vector drawing, you can do all sorts of different things to it if you wish, even changing the color.



**DESTINATION LINUX**  
**LINUX IS OUR PASSION**



A magazine just isn't a magazine without articles to fill the pages.

If you have article ideas, or if you would like to contribute articles to the PCLinuxOS Magazine, send an email to:

[pclinuxos.mag@gmail.com](mailto:pclinuxos.mag@gmail.com)

We are interested in general articles about Linux, and (of course), articles specific to PCLinuxOS.

## International Community PCLinuxOS Sites



Like Us On Facebook!  
The PCLinuxOS Magazine  
PCLinuxOS Fan Club



## Screenshot Showcase



Posted by mutse, September 9, 2019, running Trinity.



# Casual Python, Part 9

by Peter Kelly (critter)

## Launchpad

Now that we have a few applications of our own, we should have a way of quickly accessing them. These applications will not appear in the system menu unless you write a freedesktop.org compatible .desktop file for each, and place that in your /usr/share/applications directory. What you need is a personalised application launcher.

What features should we look for in the launcher?

1. Always easily available, perhaps by a hotkey combination, e.g. win + x, or as a tray resident utility, as we did with scratchpad.
2. A single button or icon to represent each application, and each of these sensitive to either a mouse click or keypress.
3. Auto closing/hiding after a selection has been made.
4. Hotkey closing if you decide not to launch any application.
5. Extensible.

Well, it turns out that we already know how to implement all of these features. Number 1 we used in scratchpad and the timer/alarm. Numbers 2 and 4 have been available since we first developed the template. Number 3 was available in Appsearch, docz and scratchpad. Number 5, all of our applications are extensible, since we are the developers.

What's left to decide is the appearance of the interface, and here we have a free hand. Here are a few ideas.

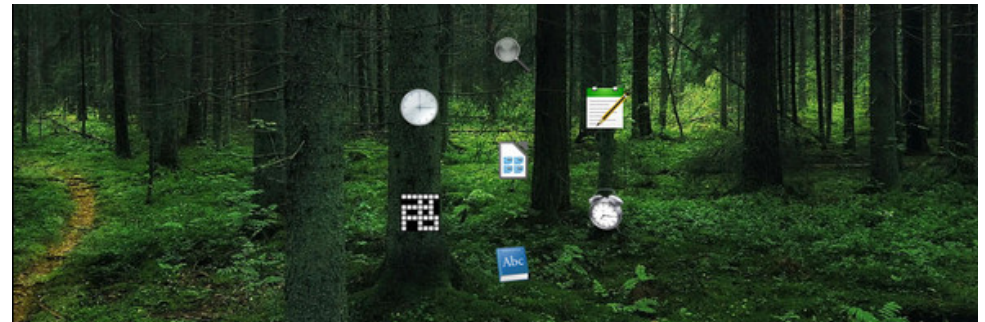


Here I have transparency turned on, just as implemented in the analog clock. The oval buttons are created in their stylesheet with a border-radius of half the height. The pale blue background is another button widget that is not connected to any method so does nothing, and which I pushed to the back by right clicking on it in designer and choosing that option.



These circular buttons have equal width and height and a stylesheet with a border radius set to half the height.

This version has full transparency and no background.



The same as the previous image, but with button background set to transparent. This may be difficult to see on some background images.





This is a more traditional launcher. No transparency, but I have used the line:

```
setWindowFlags(Qt.FramelessWindowHint)
```

in the `__init__` method to remove the top window bar. The escape key closes it.



This is the beast of launchers. The ten buttons on the left control 10 pages of the stackedWidget on the right. These buttons may be activated by pressing one of the function keys F1 to F10.

The stackedWidget has 30 buttons on each page – a total of 300 application buttons!

The buttons not in use are not connected to any application launch method, and have their `setVisible` attribute set to false so that they are not seen.

### Creating the application.

The easiest way is to reuse some code that we have already written. The scratchpad application was a tray launched application, so we can start by copying that, removing what we don't need and adding some new features.

I'm going to recreate the oval bar type in the first screenshot, but the method is the same whatever you decide upon. In the example, which will be tray resident, I have not included the timer or scratchpad applications, as these are themselves tray resident applications.

As usual, create a new directory and copy over the `scratchpad.py`, `scratchpad.ui` and `update_res.sh` files, renaming the scratch pad files to launchpad files, and editing the `update_res.sh` file to replace both instances of `scratchpad` with `launchpad`. Add suitable icons for this and each of the target applications.

Open `launchpad.ui` in designer and delete the `textEdit` widget. Add three new `pushButtons`. Change the forms `WindowTitle` to `Launchpad`, and change the window icon to whatever you chose, and the width to 435. Clear any stylesheet from the form.

Select all five buttons by control clicking on them, and delete their text from the property editor.

Set the width to 74 and the height to 50.

Set the Y value to 10, leave the X for now. Add this stylesheet to the buttons:

```
QPushButton {
border: 2px solid #2326f2;
border-radius: 25px;
background-color: qlineargradient(x1: 0, y1: 0, x2: 0, y2: 1,
stop: 0 #e9eefd, stop: 1 #bbbfbc8);
max-width: 70px;}
```

```
QPushButton:pressed {
background-color: qlineargradient(x1: 0, y1: 0, x2: 0, y2: 1,
stop: 0 #bbbf8, stop: 1 #e9eef8);}
```

Add a new pushButton and set these properties:

```
objectName btn_background
x 0
y 0
width 435
Height 70
Clear the text property.
Tooltip Press escape to hide
```

Add this stylesheet to this button:

```
border: 3px solid #2326f2;
border-radius: 35px;
background-color: #93cee9;
color: black;
# This is the color of the text in the tool tip.
```

Right click on this new button and select send to back.

Set the form Height to 70.

Set the X values of the 5 smaller buttons to 10, 95, 180, 265 and 350 respectively.

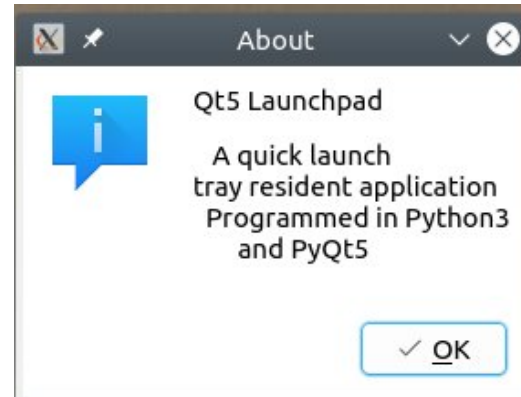
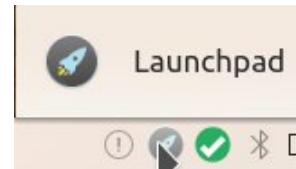
Give each a suitable icon, and set the icon size to 40 x 40

Add a suitable tooltip to each of the smaller buttons:

```
Analogue clock\n Shortcut – C
Application finder/launcher\n Shortcut – F
Document launcher\n Shortcut – D
Thesaurus application\n Shortcut – T
Crossword solution helper\n Shortcut - X
```

Change the objectName of each button to something you and the python code will recognize:

```
btn_aclock
btn_appfinder
btn_docz
btn_thesaurus
btn_xword
```



Save it, and run update\_res.sh to create the python code for the user interface.



## The code

There is nothing really new here, and the majority of it should, by now, be familiar to you.

```

launchpad.py ✕
1  #!/usr/bin/env python3
2
3  import sys
4  import os
5  import subprocess
6  from PyQt5.QtCore import *
7  from PyQt5.QtGui import *
8  from PyQt5.QtWidgets import *
9
10 import launchpad_ui
11
12
13 class Launchpad(QMainWindow, launchpad_ui.Ui_Form):
14
15     def init(self):
16
17
18
19
20
21
22
23
24
25
26     def aclock(self):
27
28
29
30
31
32
33
34
35
36
37
38
39
40
41
42
43
44
45
46
47
48
49
50
51
52
53
54
55
56     def appfinder(self):
57
58
59
60
61
62
63
64
65     def docz(self):
66
67
68
69
70
71
72
73     def thesaurus(self):
74
75
76
77
78
79
80
81
82     def xword(self):
83
84
85
86
87
88
89
90
91
92
93
94
95
96     def exitEvent(self):
97
98
99
100
101
102
103
104
105
106
107
108
109
110     def keyPressEvent(self, e):
111
112
113
114
115
116
117
118
119     def aboutEvent(self):
120
121
122
123
124
125
126
127
128
129
130
131
132
133
134
135
136
137
138
139
140
141
142
143
144
145
146
147
148
149
150
151
152
153
154
155
156
157
158
159
160
161
162
163
164
165
166
167
168
169
170
171
172
173
174
175
176
177
178
179
180
181
182
183
184
185
186
187
188
189
190
191
192
193
194
195
196
197
198
199
200
201
202
203
204
205
206
207
208
209
210
211
212
213
214
215
216
217
218
219
220
221
222
223
224
225
226
227
228
229
230
231
232
233
234
235
236
237
238
239
240
241
242
243
244
245
246
247
248
249
250
251
252
253
254
255
256
257
258
259
260
261
262
263
264
265
266
267
268
269
270
271
272
273
274
275
276
277
278
279
280
281
282
283
284
285
286
287
288
289
290
291
292
293
294
295
296
297
298
299
300
301
302
303
304
305
306
307
308
309
310
311
312
313
314
315
316
317
318
319
320
321
322
323
324
325
326
327
328
329
330
331
332
333
334
335
336
337
338
339
340
341
342
343
344
345
346
347
348
349
350
351
352
353
354
355
356
357
358
359
360
361
362
363
364
365
366
367
368
369
370
371
372
373
374
375
376
377
378
379
380
381
382
383
384
385
386
387
388
389
390
391
392
393
394
395
396
397
398
399
400
401
402
403
404
405
406
407
408
409
410
411
412
413
414
415
416
417
418
419
420
421
422
423
424
425
426
427
428
429
430
431
432
433
434
435
436
437
438
439
440
441
442
443
444
445
446
447
448
449
450
451
452
453
454
455
456
457
458
459
460
461
462
463
464
465
466
467
468
469
470
471
472
473
474
475
476
477
478
479
480
481
482
483
484
485
486
487
488
489
490
491
492
493
494
495
496
497
498
499
500
501
502
503
504
505
506
507
508
509
510
511
512
513
514
515
516
517
518
519
520
521
522
523
524
525
526
527
528
529
530
531
532
533
534
535
536
537
538
539
540
541
542
543
544
545
546
547
548
549
550
551
552
553
554
555
556
557
558
559
560
561
562
563
564
565
566
567
568
569
570
571
572
573
574
575
576
577
578
579
580
581
582
583
584
585
586
587
588
589
590
591
592
593
594
595
596
597
598
599
600
601
602
603
604
605
606
607
608
609
610
611
612
613
614
615
616
617
618
619
620
621
622
623
624
625
626
627
628
629
630
631
632
633
634
635
636
637
638
639
640
641
642
643
644
645
646
647
648
649
650
651
652
653
654
655
656
657
658
659
660
661
662
663
664
665
666
667
668
669
670
671
672
673
674
675
676
677
678
679
680
681
682
683
684
685
686
687
688
689
690
691
692
693
694
695
696
697
698
699
700
701
702
703
704
705
706
707
708
709
710
711
712
713
714
715
716
717
718
719
720
721
722
723
724
725
726
727
728
729
730
731
732
733
734
735
736
737
738
739
740
741
742
743
744
745
746
747
748
749
750
751
752
753
754
755
756
757
758
759
760
761
762
763
764
765
766
767
768
769
770
771
772
773
774
775
776
777
778
779
780
781
782
783
784
785
786
787
788
789
790
791
792
793
794
795
796
797
798
799
800
801
802
803
804
805
806
807
808
809
810
811
812
813
814
815
816
817
818
819
820
821
822
823
824
825
826
827
828
829
830
831
832
833
834
835
836
837
838
839
840
841
842
843
844
845
846
847
848
849
850
851
852
853
854
855
856
857
858
859
860
861
862
863
864
865
866
867
868
869
870
871
872
873
874
875
876
877
878
879
880
881
882
883
884
885
886
887
888
889
890
891
892
893
894
895
896
897
898
899
900
901
902
903
904
905
906
907
908
909
910
911
912
913
914
915
916
917
918
919
920
921
922
923
924
925
926
927
928
929
930
931
932
933
934
935
936
937
938
939
940
941
942
943
944
945
946
947
948
949
950
951
952
953
954
955
956
957
958
959
960
961
962
963
964
965
966
967
968
969
970
971
972
973
974
975
976
977
978
979
980
981
982
983
984
985
986
987
988
989
990
991
992
993
994
995
996
997
998
999
1000

```

Change lines 1 - 13 to match the image above.

Completely delete the methods `popup`, `clear_text` and `save_it`, as we shall not be using these. You will notice that some new methods have been added, and some of the existing ones may have moved. Position is unimportant, although the `__init__` method should always be first.

The `__init__` method

```

def __init__(self):
    super(self.__class__, self).__init__()
    self.setupUi(self)
    self.setWindowFlags(Qt.Tool)
    self.setWindowFlags(Qt.FramelessWindowHint)
    self.setAttribute(Qt.WA_NoSystemBackground, True)
    self.setAttribute(Qt.WA_TranslucentBackground, True)

```

```

# override the close event
self.exitOnClose = False

```

```

# implement the right-click menu
my_exit = QAction("Exit", self)
my_exit.triggered.connect(self.exitEvent)
my_about = QAction("About", self)
my_about.triggered.connect(self.aboutEvent)
menu = QMenu(self)
menu.addAction(my_exit)
menu.addAction(my_about)

```

```

# set the tray icon
icon = QIcon()
icon.addPixmap(QPixmap("launchpad.png"),
               QIcon.Normal, QIcon.Off)
self.tray_icon = QSystemTrayIcon()
self.tray_icon.setIcon(QIcon(icon))
self.tray_icon.setContextMenu(menu)
self.tray_icon.setToolTip('Launchpad')

```

```

home = os.path.expanduser('~')
os.chdir(home + '/python3/')
self.tray_icon.show()

```

```

# Left click event
self.tray_icon.activated.connect(self.trayIconActivated)
self.btn_aclock.clicked.connect(self.aclock)
self.btn_appfinder.clicked.connect(self.appfinder)
self.btn_docz.clicked.connect(self.docz)

```



```
self.btn_thesaurus.clicked.connect(self.thesaurus)
self.btn_xword.clicked.connect(self.xword)
```

Most of the code here is from scratchpad.py. The icon has changed, as has the tooltip and window flags and attributes added to get the look we want. Also removed are references to the textEdit, the quit and clear buttons and the file access code, as none of these are used here. Two methods from the imported os module are used: os.path and os.chdir. These put us in the directory where our applications can be found. I have mine grouped together in a directory named python3. This is an alternative to the method used in scratchpad, where the home directory was determined by os.environ['HOME']. Each button is assigned a tooltip, and connected to a method that launches the appropriate application.

The code for the left click event now connects each of the application buttons to a method that will launch the application.

The seven application launch methods are mostly the same, changing the name of the method to the name of the application to launch. There is one notable exception. When calling the subprocess.Popen method, if the application requires options to be passed then the argument **shell=True** must be included so that the shell (usually bash or sh) can be used to process the option arguments. This is evident in the aclock method.

```
def aclock(self):
    subprocess.Popen('qt5_aclock/qt5_aclock.py\
-rsdtl PCLinuxOS -x 5 -y 5', shell=True)
    self.exitApplication()

def appfinder(self):
    subprocess.Popen('qt5_appsearch/qt5_appsearch.py')
    self.exitApplication()

def docz(self):
    subprocess.Popen('qt5_docz/docz.py')
    self.exitApplication()

def thesaurus(self):
    subprocess.Popen('qt5_thesaurus/thesaurus.py')
    self.exitApplication()

def xword(self):
    subprocess.Popen('qt5_solver/solver.py')
    self.exitApplication()
```

The **exitEvent** method is exactly as in scratchpad.

```
def exitEvent(self):
    self.exitOnClose = True
    self.close()
    sys.exit()
```

In the **keyPressEvent** method, I have removed the F7 key code to clear the text, and added shortcut keys for each of the application buttons.

```
def keyPressEvent(self, e):
    if e.key() == Qt.Key_Escape:
        self.exitApplication() # cancel the app
    if e.key() == Qt.Key_C:
        self.aclock()
    if e.key() == Qt.Key_F:
        self.appfinder()
    if e.key() == Qt.Key_D:
        self.docz()
    if e.key() == Qt.Key_T:
        self.thesaurus()
    if e.key() == Qt.Key_X:
        self.xword()
```

The **aboutEvent** method is a re-worked copy of that method in scratchpad.

```
def aboutEvent(self):
    mbox = QMessageBox()
    mbox.setIcon(QMessageBox.Information)
    mbox.setWindowTitle("About")
    mbox.setStyleSheet("background:white; color:black")
    mbox.setText("Qt5 Launchpad")
    msgText = (" A quick launch \n" +
    "tray resident application\n" +
    " Programmed in Python3\n" +
    " and PyQt5\n")
    mbox.setInformativeText(msgText)
    mbox.setStandardButtons(QMessageBox.Ok)
    mbox.setDefaultButton(QMessageBox.Ok)
    mbox.exec_()
```

The **exitApplication** and **trayIconActivated** methods are the same as in scratchpad.

```
def exitApplication(self): # Exit point
    self.hide()
def trayIconActivated(self, reason):
    if reason == QSystemTrayIcon.Context:
        self.tray_icon.contextMenu().show()
```

```

    elif reason == QSystemTrayIcon.Trigger:
self.show()
self.raise_()
self.activateWindow()

```

The final block of code is much as in scratchpad.

```

if __name__ == '__main__':
app = QApplication(sys.argv)
app.setQuitOnLastWindowClosed(False)
form = Launchpad()

# centralize and resize
qr = form.frameGeometry()
cp = QDesktopWidget().availableGeometry().center()
qr.moveCenter(cp)
form.move(qr.topLeft())

app.exec_()

```

As you can see, re-using an existing application and modifying it has saved us a lot of work. Any of the examples I have shown can be built this way. The one with the 300 buttons uses a stacked widget with 10 pages in the same manner as the two page alarm/timer application.

## Functions

First the short version:

A function is a named block of code that performs a specific task and may return the result. Functions are most usually used when the same task is required to be performed many times.

```

def power(x, y):
return x ** y

```

```

print(power(3, 2)) ==> 9 # 3 squared
print(power(2, 3)) ==> 8 # 2 cubed
print(power(2, 10)) ==> 1024 # 2 to the power 10

```

But there is more to it than that.

The long version:

A function is defined by following the keyword `def` with the name of the function, a tuple of zero or more parameters, and a colon. The function code block then

follows indented one place (usually 4 spaces). To improve readability, blank lines and comments are allowed as long as the indentation is not interrupted. The end of the indentation ends the function. When you call a function, you optionally pass arguments to the function. The function places a reference to these arguments in its parameter tuple:

```

def power(parameters...):
    code...

```

```

power(arguments...)

```

```

def power(x, y):
    '''Raise one number to the power of another - x ** y.

```

```

        Takes two arguments of type int or float:
            x    the base
            y    the exponent
        Returns the result.'''

```

```

    z = x ** y

```

```

    # return X ** Y # this is what you get back
    return z

```

```

print(power(3, 4))      # >>> 81
print(power(3, 0.5))    # >>> 1.7320508075688772
print(power(2, -4))     # >>> 0.0625

```

The functions docstring may be accessed with the functions builtin `__doc__` method, which will print out the docstring, which is the text in triple quotes.

```

print(power.__doc__)

```

Raise one number to the power of another -  $x ** y$ .

Takes two arguments of type int or float:

x the base  
y the exponent

Returns the result.



## Function Arguments

Function arguments, if any, are passed to the function in the function call. The function definition may provide default values in the form 'keyword=value' for omitted arguments, but these must follow any mandatory arguments.

```
def greet(name='Everyone'):
    '''Greet a user.

    Take one argument of type str'''

    print('Hello ' + name + '!')

greet('Fred')          # >>> Hello Fred!
greet()                # >>> Hello Everyone!
```

## Starred arguments in functions

When the number of arguments cannot be determined beforehand, it is possible to pass a reference to multiple values by preceding the parameter with a '\*'. This function binds the first argument to a variable 'y', and passes the rest as a reference to a tuple 'x'. The returned object is a dictionary of 'x' keys and 'y' values.

```
def powers(y, *x):
    # calculate x ** y
    # create an empty dictionary object
    # to hold the results

    print('y = ', y)      # show what was passed to the function
    print('x = ', x)

    for index in x:
        # build the dictionary
        d[index] = index ** y

    return d              # return the dictionary to the caller

p = powers(3, 4, 6, 7, 9) # >>>
                          # y = 3
                          # x = (4, 6, 7, 9)

print(p)                 # >>> {9: 729, 4: 64, 6: 216, 7: 343}
print(p[4])              # >>> 64
```

In the above, note that the results are printed out in what seems to be a random manner. This is because a dictionary is an unordered collection of key, value

pairs. If this is not what you want, import the collections module and use the OrderedDict subclass.

```
import collections

def powers2(y, *x):
    d = collections.OrderedDict()
    print('y = ', y)
    print('x = ', x)
    for index in x:
        d[index] = index ** y
    return d

p2 = powers2(3, 4, 6, 7, 9)
y = 3
x = (4, 6, 7, 9)

print(p2)
OrderedDict([(4, 64), (6, 216), (7, 343), (9, 729)])

for key, value in p2.items():
    print(key, value)
4 64
6 216
7 343
9 729
```

## More on mapping and unpacking

Arguments may be positional where the value is taken from the order of the arguments, or keyword arguments of the form name=value. If keyword arguments are specified in the function definition, then they must supply a default value which will be overridden if a value is supplied. Keyword arguments supplied to the function in the function call may be accessed by unpacking them into a dictionary, whose name is prefixed with a double asterisk or star, \*\*kwargs. Keyword arguments must follow any positional arguments or starred arguments.

```
def uk(e, w, s, i, *currency, **capitals):
    print('Countries :\n' + e + ', ' + w + ', ' + s + ' and ' + i
          + '\n')
    print('Currency:')
    for i in currency:
        print(i)
    print('\nCapitals: ')
    for key in sorted(capitals):
        print("{0} = {1}".format(key, capitals[key]))
```



```
e = 'England'
w = 'Wales'
s = 'Scotland'
i = 'Northern Ireland'
gbp = ('pounds', 'pence')

uk(e, w, s, i, gbp, England='London',
    Wales='Cardiff',
    Scotland='Edinburgh',
    Northern_Ireland='Belfast')
```

The output from the above will be:

Countries :  
England, Wales, Scotland and Northern Ireland

Currency:  
('pounds', 'pence')

Capitals:  
England = London  
Northern\_Ireland = Belfast  
Scotland = Edinburgh  
Wales = Cardiff

## Higher Order Functions

A function that takes another function as an argument is known as a higher order function. The builtin map function is an example of a higher order function.

```
def twice(x):
    '''Double a number.'''
    return x * 2

def thrice(x):
    '''Treble a number.'''
    return x * 3

def operate(func, x):
    '''requires a function and a number as arguments'''

    result = func(x)
    return result

operate(twice, 27)      # >>> 54
operate(thrice, 27)     # >>> 87
```

## Function Return Values

A function may have a return statement followed by an expression. If the return statement is missing or there is no expression following it, then the value 'None' will be returned. Otherwise, the value of the expression is returned. What is returned is an object, and that object may be a single value, such as a number or multiple values in the form of a tuple.

```
def a(x, y):
    z = x * y                no return statement

s = (a(2, 3))
print(s)                    ==> None

def a(x, y):
    z = x * y
    return                   return but no expression

s = (a(2, 3))
print(s)                    ==> None

def a(x, y):
    z = x * y
    return z                 ==> z

s = (a(2, 3))               s = z
print(s)                    ==> 6

def a(x, y):
    p = x * y
    s = x + y
    return p, s              # return multiple values in a tuple

t = a(2, 3)
print(t)                    ==> (6, 5)
type(t)                     ==> <type 'tuple'>
```

## Recursion

A recursive function is one that calls itself repeatedly. If a termination condition is not reached, then an infinite loop results. Fibonacci numbers can easily be calculated using an iterative loop, but recursion can make the code easier to follow. The Fibonacci numbers are defined by:

```
Fn = Fn-1 + Fn-2
with F0 = 0 and F1 = 1
```

And the recursive python function might look like this.

```
def fib(n):
    if n == 0:
        return 0
    elif n == 1:
        return 1
    else:
        return fib(n-1) + fib(n-2)
```

This follows the definition closely, but has the disadvantage that for each pass through the function, the values have to be re-calculated. As the numbers get larger the number of calculations grows considerably larger. This situation can be improved by writing previously calculated values to a dictionary and looking them up as required.

```
fib_dict = {1: 1, 2: 2}
```

```
def fib(n):
    for i in range(3, n + 1):
        fib_dict[i] = fib_dict[i - 1] + fib_dict[i - 2]

    return fib_dict[n]
```

```
print(fib(23))          ==> 46368
```

This method can quickly reach python's maximum recursion depth. For larger numbers, an iterative approach works better.

## Lambda Expressions

Also known as lambda forms or lambda functions, they create anonymous function objects (because of a lack of a name). A lambda function is also known as a throw-away function. Lambda functions are used extensively in combination with the map builtin function. Lambda functions are similar to list comprehensions. The syntax of a lambda function is:

```
lambda arg_list: expression
```

There can be as many arguments as required, but only one expression. There must be no statements or comments. Note that the argument names are not variables but place holders for values or variables that will be passed to the expression. There is also no return statement. Values are returned implicitly by the expression.

```
m = lambda p: max(p)      # create the lambda object
x=(8, 3, 7)               # define the variable x
m(x)                      # pass x as the argument to the lambda
                           # >>> 8 # the result
```

Lambda expressions are more usually defined at the point where they will be used.

```
for i in range(4):
    str = lambda x: "" if x == 1 else "s"
    print("{0} item{1} returned".format(i, str(i)))
```

```
0 items returned
1 item returned
2 items returned
3 items returned
```

A lambda generated function should be used where a function object is expected or to make code more readable.

```
a = list(range(5, 27, 3))
print(a)          ==> [5, 8, 11, 14, 17, 20, 23, 26]
```

```
y = lambda x: 2 * x ** 2 - x + 7 # solve y = 2x2 - x + 7
```

```
for value in a:
    print(y(value), end=', ')
    ==> 52, 127, 238, 385, 568, 787, 1042, 1333,
```

A lambda generated function has no name, but it may be bound to a variable as above, or passed as an argument to a function.

```
a = [1, 4, 15, 255, 7, 10, 23, 183]
b = list(filter(lambda x: (x % 3 == 0), a))
print(b)          ==> [15, 255, 183]
```

*Editor's Note:* All of the code for the Casual Python article series is available for download from [here](http://commandlinefu.com).



commandlinefu.com

# PCLinuxOS Family Member Spotlight: oulik.jan

As Told by YouCanToo

**What is your name/username?**

oulik.jan

**How old are you?**

61

**Are you married, single?**

Single

**How about Kids, Grandkids (names and ages)?**

No children

**Do you have pets, what is your favorite?**

No pets anymore, favorite pets are dogs

**Are you retired, still working and if working, what do you do?**

Still working, as a certification engineer (product safety)

**Where do you call home? What is it like? IE: weather, scenery**

Living in the Netherlands now, before that I lived in South Africa. I miss South Africa too much, but it is not safe to live there anymore. Netherlands climate: at the moment quite warm, but often rainy, grey weather.

**Where did you go to school and what is your education level?**

Netherlands, Bachelor level electronics

**What kind of things you like doing? hobbies, travel, fishing, camping?**

Love traveling, I collect antiques, like cooking, music, watching movies and other good things in life.



**Why and when did you start using Linux?**

I had a netbook which was running Windows XP. A friend at work told me Linux, and I said no, not interested for a while. Then the netbook became too slow for words, took 7 to 9 minutes just to boot and there were update errors etc. So I tried Ubuntu 13.04 on that netbook (as advised by my colleague at work) about June 2013.

Well, the same netbook booted in 45 seconds, I could not believe my eyes.



**What specific equipment do you currently use with PCLinuxOS?**

2 older laptops, one for home use, and one I take with me when I need to travel.



**Do you feel that your use of Linux influences the reactions you receive from your computer peers or family? If so, how?**

My father of 91 uses Linux as well. (Lubuntu) With Windows he often picked up a virus or other malware and he has been running Linux virus free since 6 years now. He is very happy with Lubuntu.

My brothers were in the beginning quite negative about my father using Lubuntu, but now they tend to be neutral about it.

I tried to convince some friends of mine to use Linux, but they perceive it to be too complicated and too different.

Only 2 friends of mine like it.

**What would you like to see happen within PCLinuxOS that would make it a better place. What are your feelings?**

I must say I like PCLinuxOS a lot. The only thing I sorely miss is a program like Mintstick. This program lets me format my USB sticks and SD cards exFAT, which is important for large SD cards and large USB sticks. This is one of the reasons why I still have to use Windows.

Another thing which I like a lot when I am using Xubuntu (which I still use on my entertainment system) is an automatic update/cleanup program. I use uCareSystem Core on the Xubuntu machine and it is wonderful. It is also in use on my father's machine. Yes, you can update with Synaptic package manager, but something like uCareSystem Core would probably make things easier for people to update their systems.



*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.*

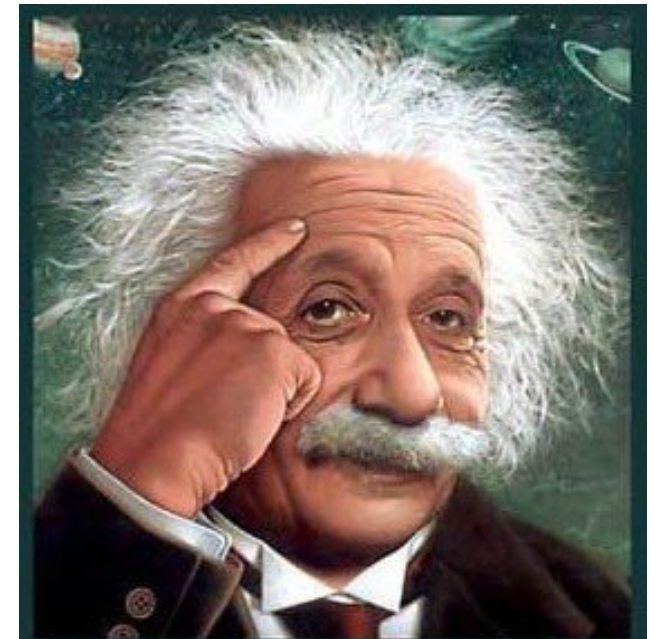
Chat with PCLinuxOS users  
from all over  
the World.  
**Sign up  
TODAY!**  
**PCLOS-Talk**  
<http://pclostalk.pclosusers.com>

**PCLinuxOS**  
Radically Simple

A magazine just isn't a magazine  
without articles to fill the pages.

If you have article ideas, or if you  
would like to contribute articles to the  
PCLinuxOS Magazine,  
send an email to:  
[pclinuxos.mag@gmail.com](mailto:pclinuxos.mag@gmail.com)

We are interested in general articles  
about Linux, and (of course), articles  
specific to PCLinuxOS.



**It's easier than  $E=mc^2$   
It's elemental  
It's light years ahead  
It's a wise choice  
It's Radically Simple  
It's ...**





Support PCLinuxOS! Get Your Official

**PCLinuxOS**  
Merchandise Today!

**PCLinuxOS**



secure  
private  
simple-to-use

**PCLinuxOS-Cloud**

Sign up **TODAY!** [pclosusers.com/services-signup.php](http://pclosusers.com/services-signup.php)

## Screenshot Showcase



*Posted by tbschommer, September 1, 2019, running KDE.*

# Accessing Special Unicode Characters, Made Easy

by Paul Arnote (parnote)

I know it has happened to you. You're typing along and all of a sudden you find yourself wanting to access a "special character," such as a copyright symbol, or the trademark symbol, or a different currency symbol.

Sure, for the copyright symbol, you could just use (c), but using the actual symbol – © – looks so much more polished. The same thing goes for the using the trademark symbol. (TM) doesn't look nearly as polished as the ™ symbol does. The same can be said for the (r) registered mark, where ® looks a lot more polished. Plus, why use three characters when one will do?

When it comes to currencies, there doesn't seem to be any real alternatives to using the British Pound symbol, the Euro symbol, or any other currency symbol, other than using the actual symbol. Of course, the computer you use should already have the currency symbol you need most often only a keystroke away, but that is often not the case when you are needing to use a currency symbol that isn't native to your area.

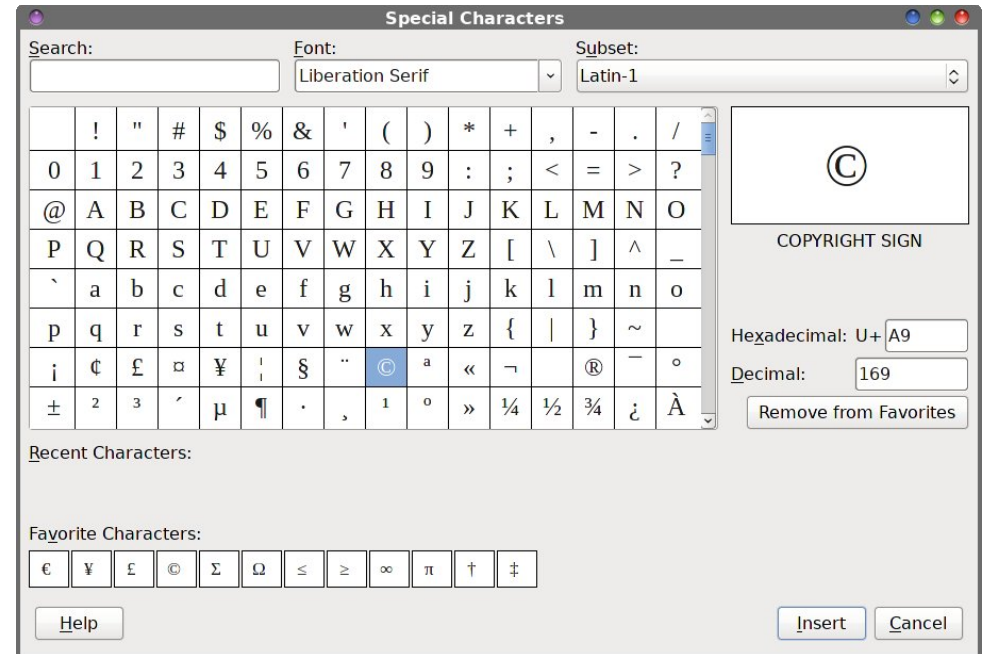
So, let's look at some different ways that you can more easily access these – and any – special characters.

## LibreOffice Writer

If your writing tool of choice is LibreOffice Writer, you are in luck. The LibreOffice developers have made it especially easy for you to access and use special characters.

Go to the "Insert" menu, and select "Special Character..." from the list. It is about ⅔ of the way down the menu. Be sure to choose the font you are wanting to use in the font selection dropdown, since not all fonts may have the characters you are wanting to use.

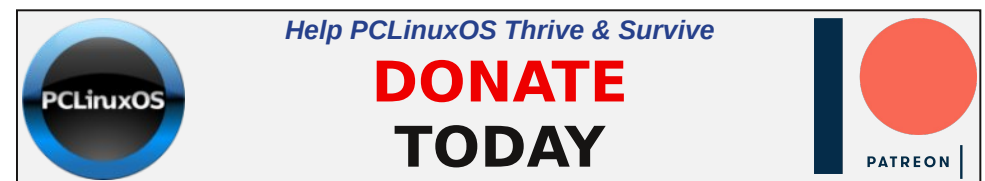
Usually, all you have to do is scroll through the list to find the special character you want to use. Look carefully, since it's very easy to overlook it the first (or second) time through the list. Once you find it, and especially if it's a character you will want to use again and again, write the hexadecimal number for the



character down on a cheat sheet you keep near your computer. I'll show you why a little later with a shortcut that might just save you a lot of time and trouble.

At the bottom of the dialog box is a line of "Favorite Characters." If the symbol you want to use isn't already in the list of favorites, the button in the image above that says "Remove from Favorites" will instead say "Add to Favorites." Clicking on that button will add that symbol to the "Favorite Characters" section. Similarly, you can remove characters from the list of favorites by simply selecting that character, and selecting the "Remove from Favorites" button.

So, LibreOffice makes it pretty simple to access special characters. But what if you're not using LibreOffice?

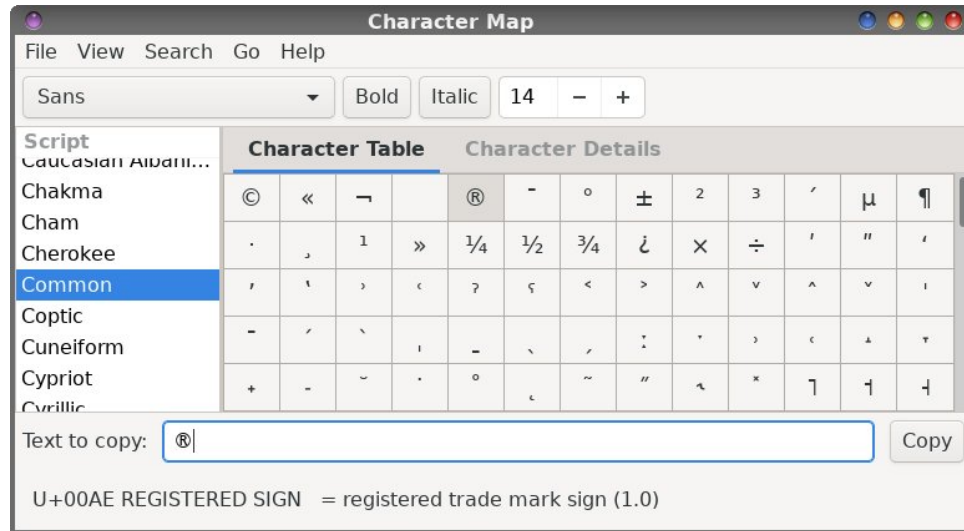




## Accessing Special Unicode Characters, Made Easy

### Use A Character Map

A character map is exactly what it sounds like: a map of all the characters that are available for a given or selected font. For GTK+ users, there's `gucharmap`. For KDE users, there's `KCharSelect`. They all work pretty much the same way, and are installable through Synaptic from the PCLinuxOS repository. Since I use Xfce, I have `gucharmap` installed.



When you find the character you want to use, double click it, then select the “Copy” button to copy it to the clipboard. Now, you can just paste the item into your document from the clipboard. The only drawback is that each time/session that you want to use the same special character, you will have to go back through the same routine of finding it, double clicking it, and then copying it to your clipboard. None of the character map programs (that I’ve found) allow you to save frequently used symbols to a special list, like with LibreOffice’s special character tool.

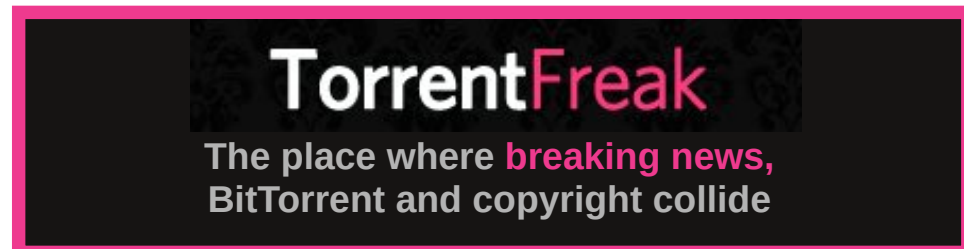
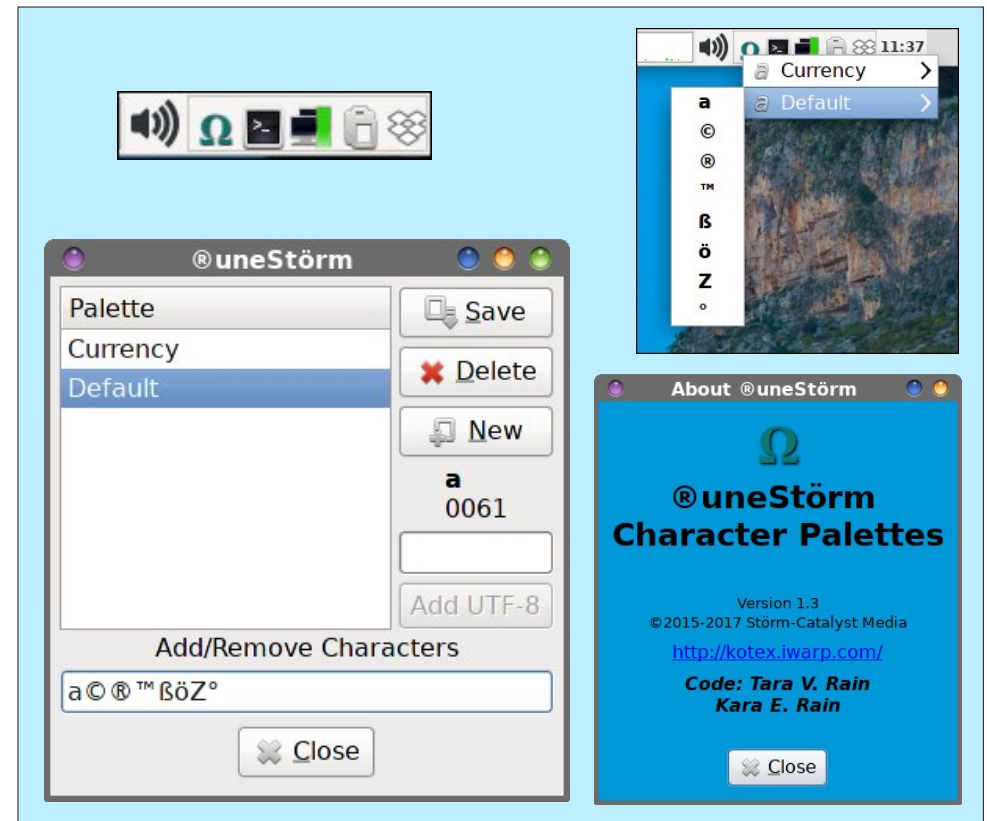
Notice at the lower left corner of the window is the unicode code for the symbol. In the graphic above, we highlighted the ® character. `Gucharmap` tells me that

the unicode code for the registered symbol is U+00AE. Again, and especially if this is a symbol you might be using rather frequently, write this code down on your cheat sheet that you keep near your computer.

### RuneStorm

Our dear and departed friend Tara Rain left us a true gift. She wrote a little utility in GTK+ to specifically allow us to quickly access frequently used symbols. It is called **RuneStorm**. Unfortunately, at the time of this article, it is not currently in the PCLinuxOS repository. However, you can [download](#) the RPM from Tara’s [website](#), and install it manually on your computer.

To install it, open a terminal session and get a root prompt. Change directories to where you downloaded the RPM file. Then, type `rpm -ivh runestorm-1.3-1.x86_64.rpm` at the root prompt. Sit back and wait ... oh, maybe 10 seconds ... and `RuneStorm` will be installed on your computer. You’ll find it listed in the Office section of your PCLinuxOS menu.



## Accessing Special Unicode Characters, Made Easy

RuneStorm sits in your notification tray, and sports an icon that looks like the capital Greek Omega symbol ... Ω. Upon first launch, the only "Palette" that is available is the one labeled "Default," like in the image above. Left clicking your mouse on the RuneStorm icon will display a list of available characters, just like in the second image.

Right clicking your mouse on the RuneStorm icon will open a menu. Selecting "Preferences" will open the dialog box (third image) where you can customize the list that appears. You can also create new "Palettes," as I've done here by adding a "Currency" list. Just be sure to click on the "Save" button before you close the dialog box if you make any changes to the existing palette or add a new palette. If you don't, all your additions or changes will be lost when you hit the "Close" button.

Selecting "About" from the right click menu will display the About dialog box for RuneStorm.

To use it, simply left click your mouse on the RuneStorm icon, then select the special character you want to use. That character will then be copied to the clipboard. Just paste the character from your clipboard into your document. It's pretty simple!

### Use The Keyboard

Ok. This is the reason I've been telling you to write down the codes that are displayed for the special characters on a cheat sheet you keep near your computer. Yes, you can enter these special codes from the keyboard, without having to use any other program to assist you.

Some of my friends from across the Atlantic talk about using the AltGR key combinations to access special characters. But they require you to basically memorize a special keystroke combination to access them. First, I find those keystroke combinations difficult to remember. Second, I sometimes find that they don't always work with my American keyboard layout. Or, maybe it's just me and my clumsy fingers.

Fortunately, there is another way that I think is much easier. Before I reveal it, let me preface it with a caveat. There are reports that this might not work in Qt applications, like are found on KDE. It does, however, work exceptionally well on GTK+ applications. I do not have a KDE installation to try this out on, so your mileage may vary.

Now, remember those codes I told you to write down? Position your cursor where you want the special character to appear. Then, hold down the **Ctrl** and **Shift**

keys, press U, and then enter the four digit hexadecimal number for the character. So, for the copyright symbol, hold down the Ctrl+Shift keys, press U, and then 00A9. **Keep the Ctrl+Shift keys held down the whole time while typing the U and the four digit hexadecimal code!** When you release the Ctrl+Shift keys, you should see your special character appear in place of the code you typed.

If, for example, LibreOffice's tool told you that the code for the copyright symbol was A9, then pad it with zeros at the start to make a total of four digits. The hexadecimal number entered must be four digits long.

Below are some common codes to help get you started. Note that the "0's" are zeros, not the letter "O".

<b>Copyright</b>	<b>00A9</b>	©
<b>Trademark</b>	<b>2122</b>	™
<b>Registered</b>	<b>00AE</b>	®
<b>Degrees</b>	<b>00B0</b>	°
<b>Euro</b>	<b>20AC</b>	€
<b>Pounds</b>	<b>00A3</b>	£
<b>Yen</b>	<b>00A5</b>	¥
<b>Cents</b>	<b>00A2</b>	¢
<b>Pesos</b>	<b>20B1</b>	₱

Once you get the hang of it, this is actually a very easy way to access those special characters. By keeping a cheat sheet near your computer, you make it super easy, especially for those special characters you find yourself frequently using. You can also look up a lot of them here or here.

And here's a bonus tip: you can use this method to create your additional custom palettes in RuneStorm. Just type the code in the dialog box in the text entry field, and you'll have quick access to those special characters through RuneStorm.

It's things like this that make me happy to be a Linux user. There are always multiple ways to get things done.



# Mind Your Step, Part 2

by phorneker

This is the second installment of this new series. At the end of the last issue, I had discussed fake email addresses. This past month I collected a series of fake addresses used as the **apparent sender** of a series of fake emails.

Mail handler applications such as SpamAssassin determine what is a legitimate email and what is a fake email by examining the header information contained in the email.

There are two mechanisms that are used to be sure that what gets sent out from your email account **really did come from your email account**.

The first is the **Sender Policy Framework**, which is configured at the domain servers where your domain is hosted (such as GoDaddy).

The second is the **DomainKeys Identified Mail** standard (<http://www.dkim.org>) for authenticating the sender of emails. G-Mail already has this mechanism implemented.

So how does an email with a fake sender address get through in the first place?

The answer to that has to do with the Simple Mail Transport Protocol, or **SMTP** for short. SMTP was designed to get email sent out as quickly and as **simply** as possible. *These days, that is exactly what is wrong with SMTP as a means of sending out emails*, especially at a time when we have to worry about identity theft (and its long term consequences).



So, what goes on behind the scenes when an email is sent?

First, the email client connects to an outgoing SMTP server, then issues an **EHLO** command to get that server's attention.

The outgoing SMTP server then asks for the SMTP username and password. The email client responds appropriately.

It is here where the spoofing of the sender's email address is possible. The email client issues a **MAIL FROM: sender identifier <email address>** where *sender identifier* is the name associated with the email address enclosed in **<>**.

Next, the email client receives a **250** code meaning "all is well" for the outgoing SMTP server. Then, the email client issues a **SEND TO: receiver identifier <email address>** where the parameters are for the receiver instead of for the sender.

The main problem here is that the outgoing SMTP server *does not check to see if either address is valid or even if either address is legitimate*.

Hence, the aforementioned mechanisms, SPF and DKIM are necessary additions to the outgoing SMTP server so that the server can check for authentication of email addresses *before the email can be accepted for delivery*.

The other part of the problem is that the SMTP client is not necessarily going to be Thunderbird or Claws, or even a web browser. It could be a customized spam generation program written in C++, Java, Python or Ruby.

## Email Spoofing vs Email Hacking

The fake sender email addresses I collected were the result of **email spoofing**. email spoofing is to electronic mail what **robocalls** are to landline and cellular phone service.

The main goal of **email spoofing** could well be nothing more than to **validate your email address** to cybercriminals, or could be an attempt to gain vital information needed to steal money, or worse, your identity.

**email hacking**, however, is a much more serious problem. This means that a cybercriminal somehow gained full access to your email account. (If this happens, it is a good idea to at least change the password, or better yet, close the email account outright.)

***This is one reason why everyone should change his/her password on all essential and valuable accounts often (such as your Google account)***



**and make those passwords as difficult to guess as possible. Also, make sure that two factor authentication is enabled wherever and everywhere possible.**

## Hillary Clinton Sure Had The Right Idea...

So what do we do with emails containing spoofed email addresses (especially those in the Spam folder)? **Delete them, of course.** By deleting them, your email address will **never be validated**, no matter how often the spam machine is used....and if worse comes to worst, simply close the email account and open a new email account elsewhere. The cybercriminals will **eventually get the message.**

## If You Must Close Your Email Account...

Before you decide to close your account, **it is a very good idea to contact those people (and businesses) who legitimately have your email address and inform them of the change of email address.**

## Types Of Fake Email Addresses

The most common type of fake email address can be replicated with a simple function in Ruby.

requires “securerandom”

```
string_length = <length of fake email component>
fake_address_component =
SecureRandom.alphanumeric(string_length)
```

```
fake_email = fake_address_component + “.” +
fake_address_component + “@” +
fake_address_component.downcase + “.us”
```

The last line of Ruby code composes the fake

sender email address as shown in this rather extensive list:

(fake emails in my Spam folder on September 9, 2019)

5736g89W2tJ2T8YNNV9h.5736g89W2tJ2T8YNNV9h@5736g89w2tj2t8ynvv9h.us

(on September 12, 2019)

i032Dna70xLT3J2OB3KK.i032Dna70xLT3J2OB3KK@i032dna70xlt3j2ob3kk.us

(on September 17, 2019)

mGCKv6hDR.mGCKv6hDR@mgckv6hdr.us  
wH4zTfyoC.wH4zTfyoC@wh4ztfyoc.us  
T48fWRrR8.T48fWRrR8@t48fwrrr8.us  
L5GWwGbGW.L5GWwGbGW@l5gwwgbgw.us  
h4C2Cxqp4I0ikyCP8N4h.h4C2Cxqp4I0ikyCP8N4h@h4c2cxqp4i0ikycp8n4h.us  
x3v9IUPmY.x3v9IUPmY@x3v9iupmy.us  
IiLrnnIip.IiLrnnIip@iilrnniip.us  
XYGSg0NvL.XYGSg0NvL@xygsg0nvl.us

(this one came on September 18, 2019)

pA9dEeXs7.pA9dEeXs7@pa9deexs7.us

(and again on September 19, 2019)

dqcegvRx9.dqcegvRx9@dqcegvrx9.us  
ev796D5eVy3No3MmM4IF.ev796D5eVy3No3MmM4IF@ev796d5evy3no3mmmm4lf.us  
F9QrVa0LZ.F9QrVa0LZ@f9qrva0lz.us  
BmltyjS8c.BmltyjS8c@bmltyjs8c.us  
Y9OYc1Hxp.Y9OYc1Hxp@y9oyc1hxp.us

## But Wait, There's More!

A variation on the code produced these crazy fake email addresses:

WEVBHEN6DZFBGBBGH3AXJ9YRYJFVI0@w7hwxrbtnrh6czfumwj2u10lft6t7.net  
56H3N1I3W25EBWLUX58DTG7B5SH3ZL@qim81x7d4p9qup0lxlxl03g9ntfeo.net  
CGGJP8RVHLS7PDX01DPS75YQWY0L3P@0gdpwoo908jtigjs2s8p5ednti93vj.net

## Another Template Used Here

This template is used for fake emails that function similar to **robocalls** in the sense that simply replying to these emails **validates your email address** just as responding to robocalls validates your phone number.

usephorneker@kkmblyieldingly.cu  
usephorneker@hyo-yieldingly.cu  
usephorneker@zzqfk-yieldingly.cu  
usephorneker@rrcjs-yieldingly.cu  
usephorneker@kpnmx-yieldingly.cu  
usephorneker@tzcx-yieldingly.cu  
usephorneker@hnclt-yieldingly.cu

In Ruby, the fake email address is coded as follows:

```
username = <account name from database of email addresses>
random_junk =
SecureRandom.alphanumeric(string_length)
fake_email_address = “use” + username + “@” +
random_junk + “-yieldingly.cu”
```

## What About The Email Content?

The content of these fake emails varies. The latter type of fake sender email addresses are normally associated with the type of junk mail we normally get in our physical mailboxes, such as:

- Extended warranty offers for automobiles
- Cut rate auto insurance
- Weight loss programs
- Offers to reduce interest rates on credit card debt

I loved the one I got for cut rate automobile insurance, especially since the real giveaway here is the .cu domain, which happens to be the country of **Cuba**.

Automobile insurance from Cuba sold in the US? I **don't think so**. How stupid do these people think we are?

As for the latter, the content varies from simple (and rather vulgar) messages, of which one is expected to reply, **which happens to be the very thing that validates email addresses** to cybercriminals.

For example, a simple message such as:

*"Stop sending me your f\*\*ing photos."*

is obviously intended to stir up emotions, which happens to be a form of social engineering.

**If the email address is spoofed, it does not matter about the content, as these messages will be deleted anyway.**

### Other Spoofed Addresses

Let us look at the rest of the collection, starting with these completely self explanatory examples.

w8ZzH4KDFyE20Ke657d@3kkpq6uc1zm4w37q9c.edu.se  
 eeH6eG6vQC318CUvtpf@g5kndy88b2mjt25ad.edu.se  
 8EP4SNQ4.8EP4SNQ4@j8g5tr42.us  
 4o.r5oy6dt0@9lzhw573arale6n.us  
 32.j3n4jh11@1164y27gb8d83m8.us  
 55f.4xb58365x@ma0m49f623ywe21.us  
 jl5.33qa9191@905a34v7940jmkm.us

These next examples appear to have come from New Venture Services Corporation of (drumroll, please) **Drums, PA**.

ccgc@nlkjgf.deedhq.com  
 sqgf@bnqycm.deedhq.com

The official website for New Venture Services Corporation is **newvcorp.com**, not **deedhq.com**. New Venture Services Corporation is a retailer of domain names, and is a competitor to GoDaddy.com.

Anything with **admin** for a username in an email address is a sign of trouble. **No legitimate email address should contain admin in the account name.**

## Screenshot Showcase



*Posted by Meemaw, September 1, 2019, running Xfce.*

admin@sfdcknm.cu  
 admin@ndkoi.jp  
 admin@tnwgrd.co  
 admin@dlwckyu.jp  
 admin@eyiraj.jp  
 admin@ogqxl.co  
 admin@vluvxlj.cu

These next four examples are **self explanatory**.

admin@ssdcxneqwyqtplggomn.zyafqentevbpbddjeuux.d  
 dsds  
 admin@njwibv5n5xizmwxm1pmi.hmpzfnjj45qo4yga6xpf  
 .ggfde  
 admin@jnilklv00fd7nmgxrmjd.mbe7r7g4m91e7z91084e.  
 jppssd  
 admin@40he38moc4wxw3h16eei.bj2mqmb7vjdhak1ii5e  
 3.gvvvdfs

The **Yelp!** service for business reviews often gets spoofed. (I do not even *have* a Yelp! Account.)

no-replySYl@ateyelp.com  
 no-replyQOp@emmyelp.com  
 2ir4m32lhmhsz396@ezfxmyyelp.com  
 no-replyoa5@xndyelp.com  
 no-replyNg9@nliyelp.com  
 no-replywWl@nwgypelp.com  
 no-replyUXk@knryelp.com

The template containing “use” followed by the username works except when the program generating the spam messages is emulated by a

careless person. (Apparently, these people are not the sharpest tools in the shed.)

usephorneker@tirrx.jsht.ibuzjw.jeepmodels.in  
 usephorneker@tsn-yieldingly.icom

Though I do not like faked email addresses, I absolutely love these examples (which apparently must have been construed by Bart Simpson):

essyrhrcvylme@yusnrelcvuqdb.ilufat.**helplessnut**.com  
 frvmpeszvdsw@jlvvwikicliv.**trivialurgent**.com  
 QMRXVYI64@dvipgdx-b-o.9**botruck**.com  
**nooreply**@431.affpartners.com  
**nooreply**@y7l.scripts.comparedperformances.com  
 XCVBOUYGDX@yfbh.somsupply.com

Finally, these examples give a clue as to who is a member of the parties responsible for this nonsense.

**nyominger**@nursehitchhiker.org  
**nyominger**@peeltournament.org

At least we have a nickname to look out for.

### We Cannot Overstate The Obvious

The examples here are only a handful of what is possible of the type of spoofing to look out for. I am sure you may have come across different examples, or even found a different pattern of spoofed email addresses from what I have observed.

At least, there are mechanisms in place that can detect spoofed messages. G-Mail already uses some of these mechanisms as we have seen by simply looking into the Spam folder.

While these mechanisms help, it is our own **vigilance** and education that is the key to preventing theft, both monetary and identity.

*Looking for an old article?  
 Can't find what you want? Try the*

**PCLinuxOS Magazine's  
 searchable index!**

The **PCLinuxOS** magazine



Support PCLinuxOS! Get Your Official

**PCLinuxOS**  
 Merchandise Today!

**PCLinuxOS**



# Firefox Task Manager: Find Your Resource Hog

by Paul Arnote (parnote)

It's happened to us all. We're cruising along using Firefox, have multiple tabs opened up, and then things slow to a crawl. Or, we install a new extension/add-on and find things running slower than normal. It is tempting to just close out Firefox and start over, but there is another choice that may be better.

Firefox has a built in task manager. It will show you exactly how much memory is being consumed by which tabs and which extensions, as well as its energy impact (or CPU cycles ... the higher the CPU cycles, the greater the energy impact). The latter is important to laptop users to help maximize battery life. To access it, you can click on the three line hamburger menu, then select "More ..." from that

Name	Type	Energy Impact	Memory
Task Manager	Tab	Low (0.45)	47.8 MB
Inbox - parnote@gmail.com - Gmail	Tab	Medium (1.75)	591.4 MB
uBlock Origin (uBlock0@raymondhill.net)	Add-on	—	8 MB
Behind The Overlay Revival ({c0e1baea-b4cb-4b62-97f0-278392ff8c37})	Add-on	—	408 KB
Frozen 2   Official Trailer 2 - YouTube	Tab	Low (0.5)	65.4 MB
Firefox Task Manager: Find Your Resource Hog - Google Docs	Tab	Low (0.15)	330.7 MB
Video DownloadHelper ({b9db16a4-6edc-47ec-a1f4-b86292ed211d})	Add-on	—	4.9 MB
> ... (99+)MeWe - The Next-Gen Social Network	Tab	Low (0.05)	173.2 MB
> American Ranch .300 AAC Blackout 16.12 10 Round Flat Dark Earth Ruger 26968	Tab	Low (0.37)	29.2 MB
The PCLinuxOS Magazine: The Latest	Tab	Low (0.05)	21.9 MB
How to create a bash script for starting VirtualBox VMs - TechRepublic	Tab	Low (0.15)	16.2 MB
How to use the Firefox Task Manager - TechRepublic	Tab	Low (0.1)	18.6 MB
Recent - Google Drive	Tab	Low (0.1)	55.5 MB
LastPass: Free Password Manager (support@lastpass.com)	Add-on	—	9.9 MB
AddToAny: Share Anywhere ({35d49e56-0142-4a7b-82a8-6ace7d28ff92})	Add-on	—	433 KB
AutoplayStopper (autoplay@html5)	Add-on	—	486 KB
Bookmarks Organizer (bookmarksorganizer@agenedia.com)	Add-on	—	366 KB
Bulk Media Downloader ({72b2e02b-3a71-4895-886c-fd12ebe36ba3})	Add-on	—	567 KB

menu, and select "Task Manager" from the options. Alternatively, you can enter **about:performance** on Firefox's URL bar.

The task manager first made its debut in Firefox 64 last December.

If it's a tab that's slowing things down, position your mouse over the line that lists the tab that's slowing things down, as evidenced by the amount of memory and energy impact it shows. At the far right side of the window will be a large "X." Click on the "X" to close out that tab.

However, if it's an extension or add-on that's slowing things down, you can position your mouse over the item that's slowing things down. At the far right side of the window will be an arrow. Click on it to open up the Add-on Manager window, where you can disable or uninstall the offending extension or add-on.

If you work like I do, and you have many tabs opened up at any given time, the Firefox Task

Manager is an important tool in your arsenal. As I'm writing this article, I have 12 tabs opened up, and have had as many as 20 tabs opened at various times. The tool makes it easy to find memory and energy hogs, and helps dispatch them to the digital netherworld. It sure beats just trying to guess which tabs and extensions/add-ons are guilty of causing the slowdown.

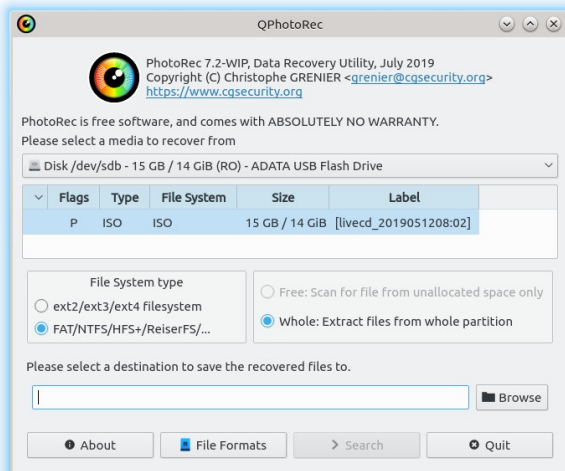
DESTINATION  LINUX  
LINUX IS OUR PASSION



# Repo Review: QPhotoRec

by CgBoy

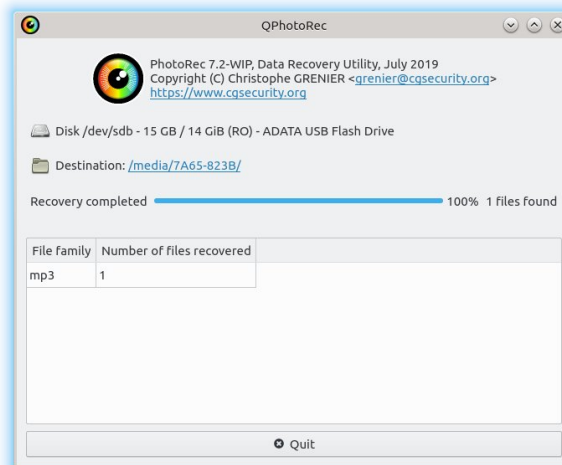
QPhotoRec is a handy tool for recovering lost data from damaged or corrupted storage devices, or files you may have just accidentally deleted. Despite its name, QPhotoRec can recover many different kinds of files from FAT, NTFS, exFAT, ext2/ext3/ext4, ReiserFS, and HFS+ file systems.



QPhotoRec's user interface is relatively simple and well laid out. From the drop down menu near the top you can select a storage drive, and then below that you can choose which partition you want searched. You have to then select the appropriate file system type option. You can then set it to search the whole partition, or just the unallocated space (This option only recovers deleted files). The next step is to set where the recovered files will be saved to. For some reason, when I tried pressing Browse to change the output folder, it just gave me an io-slave error message (That was in KDE, it worked fine running in

XFCE). But typing in the output location manually still works fine. After that, you can choose which file formats you want QPhotoRec to search for and try to recover.

Now just press Search and wait while it tries to find your lost files. Depending on the size of the storage drive, it can take quite a while, so just be patient. Any files that QPhotoRec has found will show up in a list on the screen, and will automatically be recovered and saved to the output folder. The recovered files will, however, have root ownership, so you'll probably want to change that afterwards.



One thing to keep in mind is that the original filenames and date stamps of the recovered files will most likely be lost. Of course, files can't be recovered if they've been overwritten, so the best way to insure that lost files are still recoverable is to not create or save any new data on the drive until you've fully recovered the lost files.

## Summary

I was actually somewhat surprised at just how well QPhotoRec was able to recover some of my files that I had thought were long gone. Except for the file browser problem I mentioned earlier, I had no problems with QPhotoRec, and I would definitely recommend it.



PCLinuxOS.



Radically Simple.

**PCLinuxOS**

Available in the following desktops:

KDE LXDE Xfce

MATE Trinity

Enlightenment



Help PCLinuxOS Thrive & Survive

**DONATE**  
**TODAY**



## Screenshot Showcase



Posted by OnlyHuman, September 28, 2019, running e23.

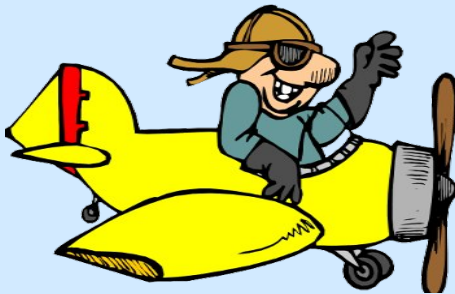


# ***ms\_meme's Nook: Come Fly With Me To Linux Land***



**MP3**

Come fly with me to Linux Land we'll go  
Fly with the rest as my guest  
The booting's never slow  
Come fly with me you'll love Linux I know



**OGG**

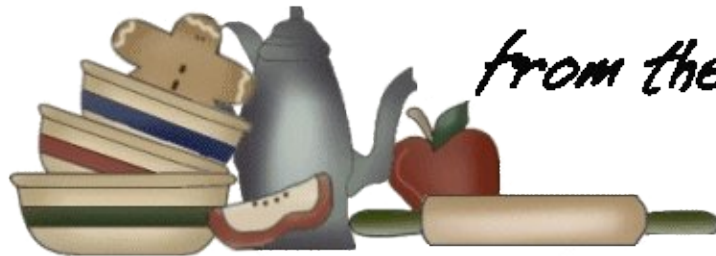
Come fly with me we'll take the latest shell  
With Captain T-STAR the best by far  
Through the net we'll sail  
And Tux will serve a grub cocktail

Once I get you up there  
You will find folks who care  
Safe everywhere no virus to snare  
Once I get you up there  
You will find open source to share  
You may hear me sing some fun I'll bring



Come fly with me PCLOS is the top  
Everything free for you and me  
You'll never distro hop  
Come fly with me the fun will never stop  
Come fly with me PCLOS is the top

# PCLinuxOS Recipe Corner Bonus



*from the kitchen of  
youcantoo*

## Make-Ahead Slow-Cooker Asian Peach Chicken Thighs

### Ingredients:

#### To Freeze/Slow Cook

1/4 cup honey  
1/4 cup soy sauce  
2 tablespoons butter, melted  
2 tablespoons chili garlic sauce  
20 oz boneless skinless chicken thighs  
1 bag (12 oz) frozen sliced peaches

#### To Serve

2 tablespoons cornstarch  
2 tablespoons cold water  
2 cups hot cooked white rice  
1/2 cup sliced green onions  
1/4 cup chopped fresh cilantro leaves  
1/4 cup chopped roasted, salted cashews  
1 lime, cut into wedges

### Directions:

1. In labeled 1-gallon freezer bag, place honey, soy sauce, melted butter and chili garlic sauce; seal bag. Knead bag to mix well. Add chicken and frozen peaches; squeeze bag to thoroughly coat chicken and peaches in sauce. Lay flat, and freeze up to 3 months.

2. Thaw 12 to 24 hours in refrigerator, until completely thawed.

3. Pour contents of bag into 3 1/2- to 4-quart slow cooker. Cover and cook on Low heat setting 4 to 5 hours.

4. In a small bowl, beat cornstarch and cold water. Quickly stir into chicken mixture in slow cooker; cover. Increase heat to High heat setting; cook 5 to 10 minutes or until thickened.

5. Divide rice among 4 bowls. Divide chicken mixture among bowls of rice. Top with remaining ingredients.

### Tips:

Sushi rice makes a great sticky side for this peachy dish, but any white rice will do.

Not crazy about cilantro? Try Thai or Italian basil instead.



# PCLinuxOS Puzzled Partitions

8					3		4	2
4					8	1		
		5	7					
1								9
			2			6	3	
	5		3	1		8		
2	4	7	5					8
				2		4		3

**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.



## 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!

G<sub>2</sub> J<sub>8</sub> E<sub>1</sub> R<sub>1</sub> L<sub>1</sub> G<sub>2</sub> U<sub>1</sub>

— — — — — — —

S<sub>1</sub> I<sub>1</sub> C<sub>3</sub> U<sub>1</sub> C<sub>3</sub> R<sub>1</sub> Z<sub>10</sub>

— — — — — — —

S<sub>1</sub> E<sub>1</sub> I<sub>1</sub> F<sub>4</sub> U<sub>1</sub> T<sub>1</sub> A<sub>1</sub>

— — — — — — —

Double Word

O<sub>1</sub> G<sub>2</sub> L<sub>1</sub> N<sub>1</sub> C<sub>3</sub> I<sub>1</sub> W<sub>4</sub>

— — — — — — —

S<sub>1</sub> C<sub>3</sub> O<sub>1</sub> T<sub>1</sub> E<sub>1</sub> U<sub>1</sub> M<sub>3</sub>

— — — — — — —

Possible score 247, average score 173.

Download Puzzle Solutions Here



# PCLinuxOS Word Find: October 2019

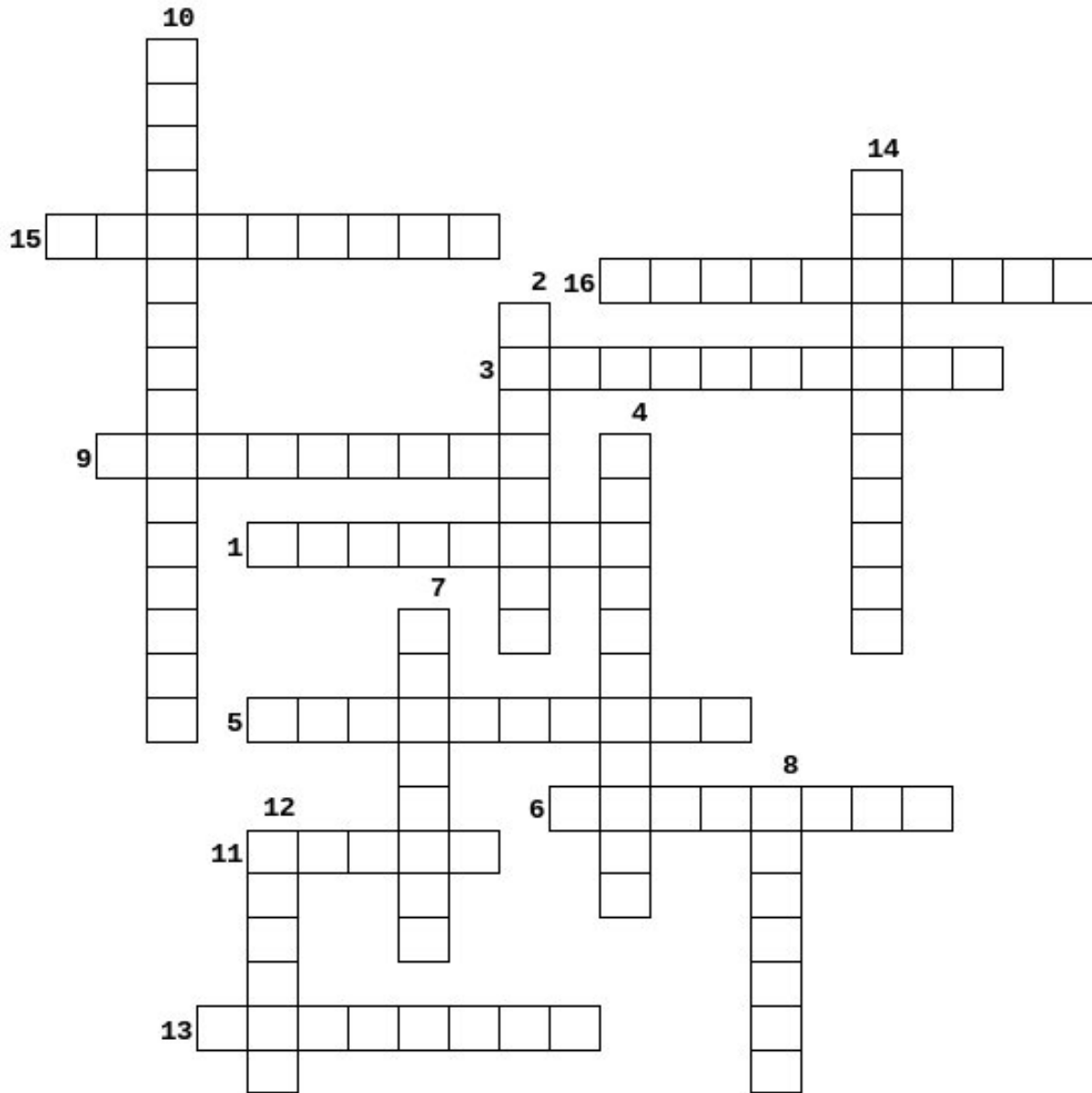
## Carnivals & Fairs

I V O Y X L Q N B W D U Y C A K X F P X U Y O C W G P U K S  
 D M W I V M T X X V D Q Z X B Q Q P G T Q D G Q U V Z Q Q C  
 A F K L U J G Z S I D E S H O W B M Z Y S N V A E C X O E R  
 S L R Z J U W I X P A R A D E D T D N D I C E C R E A M R E  
 C C V I H K G S J A U D J X B S A P M D E B S E X F A K W E  
 Z O A B E Y B T R K C V M Q P N W D G E N O C W O N S Q J K  
 Y Z T Y T D L A V I T S E F C V O N J G L B P V J M J R W N  
 J P R T R H F L S D A N R E X C I T I N G X E Q U N Z C A D  
 W N M Q O L V O L E Q F B N D W P L Z E H M T S X S E Q M A  
 D L F U C N A S O H K C Y X W K G C E O E N I O U G P Z Y S  
 W B L B Y C C R B D V A W T U P X J U E Y C O Z W O X S P B  
 R Y X E L P P A Y D N A C I N E F S J T H O P I D X H I N O  
 G V G P H G Z C N X I L T L Z U E F A G V W H F O K N N Y Y  
 G K D E D U H R H D O B Z J E O O O R A G I S D S A R G U W  
 J H G M Z F A E G W Y R A B F N L C R C W N N I R A E K R F  
 G D F H L F W P N M R E M M Z F N G K D J U O T R M V I O A  
 P R T E G K M M L Z N F I A F C A U P W O X I M M R K S L P  
 D N M U B M N U A O S R C R H I H X F R S K T C J A E E L K  
 N J C X N D R B V Y R A D Z Y R W F O Y V N A T S E I F E B  
 U W H B H Y P L C O R N D O G E L G K Q E S R P D X B M R G  
 O R J U G G L E R N Y M Q F N E Y D A M U F B W J Z U V C R  
 R S X K M Y J S I I I Y S N R R F C E C F M E Q R T T J O Z  
 G P S X G K P V L D X H W I R O U S R I O V L Q S T P I A V  
 R E T E Z T A Q W R B V A E R B U I B I C Y E O G V Z K S W  
 I C Q W K L A A K Q E F M Y O M C W S N R O C P O P P G T V  
 A T C R D U Y Z I B E E R N A A A I V Y N E Q P G O B N E I  
 F A T G V U M U V T C J X B S J F R I A F S D L R O W X R A  
 M C H G F I O Y A W E E X H I B I T I O N L D F A Y I B T P  
 A L J I X B J T D P Y Z K V A D N A B G N I H C R A M Q R S  
 R E Q F P S S X H K A U Y J R Q J I X K E D X Q V H D I Z B

amusement	bumper cars
candy apple	carnival
celebration	circus
clown	corn dog
costume	cotton candy
county fair	dance
exciting	exhibition
fairground	Ferris wheel
festival	fiesta
float	fried food
fun house	funnel cakes
house of mirrors	ice cream
jamboree	juggler
marching band	merry-go-round
midway	music
parade	popcorn
roller coaster	sideshow
snow cone	spectacle
spin art	state fair
wingding	world's fair

[Download Puzzle Solutions Here](#)

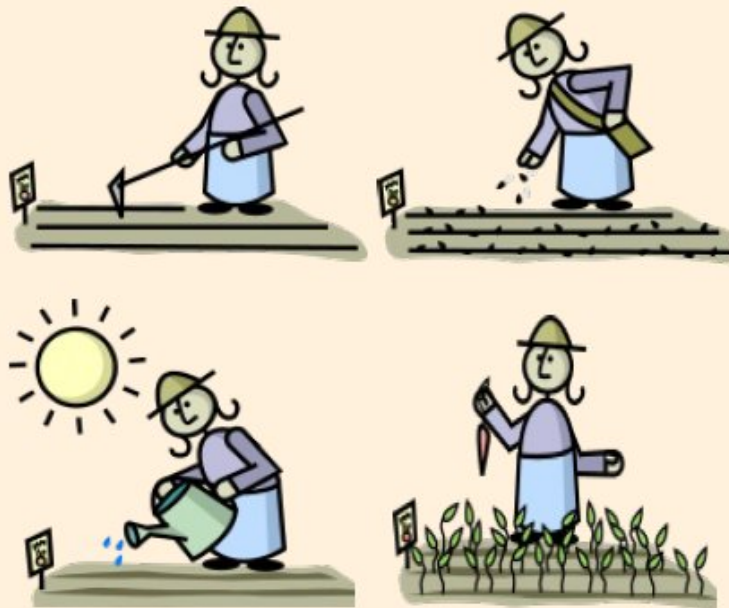
# Carnivals & Fairs Crossword



1. a traveling amusement show or circus
2. an organized series of concerts, plays, or movies, typically one held annually in the same place.
3. a public display of works of art or items of interest
4. recognition of an important occasion by taking part in an activity that makes it special.
5. a cake made of batter that is poured through a funnel into hot fat or oil, deep-fried until crisp, and served sprinkled with powdered sugar
6. a small show or stall at an exhibition, fair, or circus.
7. a lively event or party.
8. art form that primarily uses paint, a canvas and a spinning platform. The art is created by dripping the paint onto the spinning canvas
9. something that causes laughter or provides entertainment.
10. a maze or series of passageways lined with mirrors, especially curved mirrors giving distorted reflections. (NOTE: Answer has 2 spaces.)
11. sounds combined in such a way as to produce beauty of form, harmony, and expression of emotion.
12. an area of sideshows, games of chance or skill, or other amusements at a fair or exhibition.
13. a large celebration or party, typically a lavish and boisterous one.
14. an amusement-park or fairground ride consisting of a giant vertical revolving wheel with passenger cars suspended on its outer edge.
15. a visually striking performance or display.
16. small electric vehicles driven for entertainment in a special closed space at a fair, etc., where the aim is to try to hit other cars

[Download Puzzle Solutions Here](#)

# Mixed-Up-Meme Scrambler



Voodoo  
*ICMGA*

\_\_\_\_\_

Person  
*MANHU*

\_\_\_\_\_  \_\_\_\_\_

Money  
*CONMIE*

\_\_\_\_\_   \_\_\_\_\_

Order  
*NDAMDE*

\_\_\_\_\_

\_\_\_\_\_

"Su to Root"

Use the clues to unmix the letters to make a new word. Remix the letters in the red boxes to solve the puzzle.

[Download Puzzle Solutions Here](#)



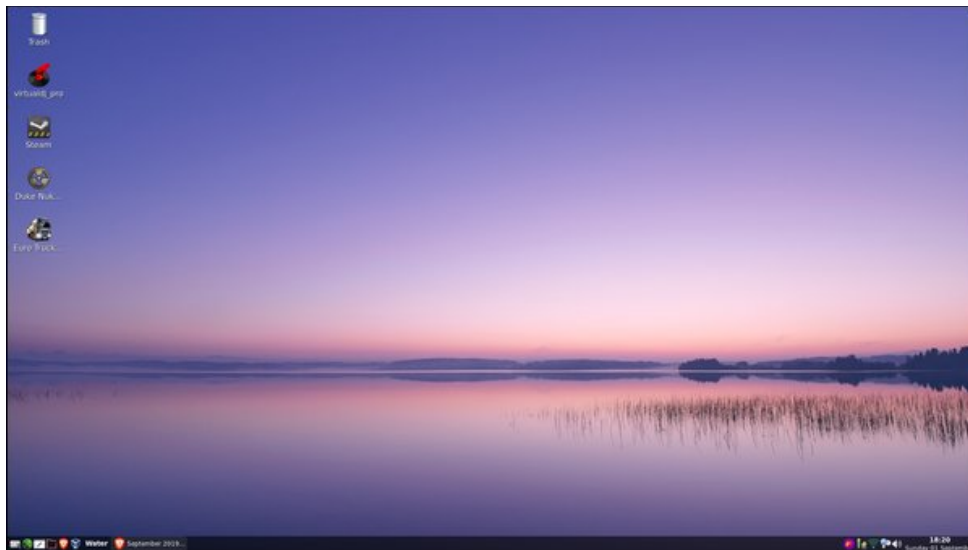
# More Screenshot Showcase



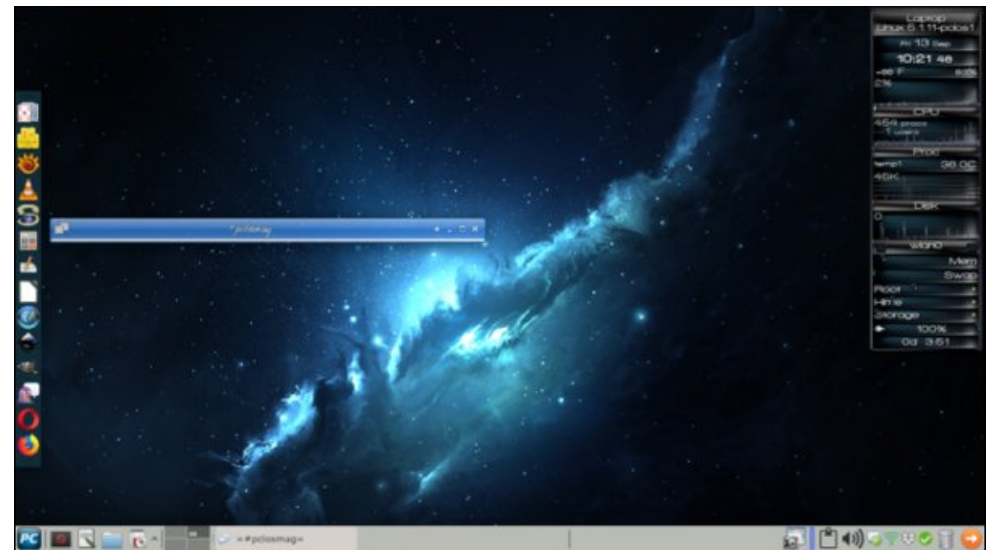
*Posted by tbschommer, September 14, 2019, running KDE.*



*Posted by sam2fish, September 18, 2019, running KDE.*



*Posted by present\_arms, September 1, 2019, running Openbox.*



*Posted by Meemaw, September 13, 2019, running Xfce.*