

Install 'Docky' on Ubuntu 20.04+

Docky is a Linux application that provides a dock for applications you'd frequently use. It's no longer in the default repositories. This is how you install Docky on newer versions of Ubuntu.

First, open your terminal.

```
[code]cd Downloads && mkdir tmp && cd tmp[/code]
```

Next, you'll need Docky's dependencies:

```
[code]wget
http://archive.ubuntu.com/ubuntu/pool/universe/g/gnome-sharp2/
libgconf2.0-cil_2.24.2-4_all.deb
wget
http://archive.ubuntu.com/ubuntu/pool/main/g/glibc/multiarch-s
upport_2.27-3ubuntu1_amd64.deb
wget
http://archive.ubuntu.com/ubuntu/pool/universe/libg/libgnome-k
eyring/libgnome-keyring-common_3.12.0-1build1_all.deb
wget
http://archive.ubuntu.com/ubuntu/pool/universe/libg/libgnome-k
eyring/libgnome-keyring0_3.12.0-1build1_amd64.deb
wget
http://archive.ubuntu.com/ubuntu/pool/universe/g/gnome-keyring
-sharp/libgnome-keyring1.0-cil_1.0.0-5_amd64.deb[/code]
```

Now, let's install them all at once:

```
[code]sudo apt-get install ./*.deb[/code]
```

Alright, you've now taken care of the dependencies. Let's grab the Docky .deb:

```
[code]wget
```

```
http://archive.ubuntu.com/ubuntu/pool/universe/d/docky/docky_2.2.1.1-1_all.deb[/code]
```

And, of course, you can now install it:

```
[code]sudo apt-get install ./docky_2.2.1.1-1_all.deb[/code]
```

There. You now have a running/working Docky and you can customize it and use it just like you did on older versions of Ubuntu. This should, of course, work with all official flavors of Ubuntu and with those distros that base themselves on Ubuntu's 20.04+ version.

Installing Google Earth on a Remote Computer

I wanted to test Google Earth Pro, but I didn't want to install it on this computer. I wanted to install Google Earth Pro on a remote computer and test it there.

This is really just an expression of why I like Linux as much as I do. It allows me to be elegant and lazy!

First, I went to Google's site for their Google Earth Pro application and found the download. It tried to make me download it automatically, and I canceled that. Instead, I right clicked on their link to 'try again' if the download didn't start automatically and copied that link. Tada!

Then, I opened my terminal...

```
[code]ssh kgiii@kgiii-lmde.local[/code]
```

I entered my password and was logged into that computer in the

terminal.

The next steps were just as easy.

```
[code]wget  
https://dl.google.com/dl/earth/client/current/google-earth-pro  
-stable_current_amd64.deb  
    && sudo apt install ./google-earth-pro-  
stable_current_amd64.deb[/code]
```

Then I just waited and let it finish the task I'd set for it. Now, when I next go to that computer, or if I login with VNC, I can use Google Earth Pro and won't have to go find the download, download it, and wait for it to install before I can use it. It's already there, waiting for me to play with it.

Prevent Duplicates in Bash History

As you may have noticed, bash keeps a history. If you open your terminal and push the up arrow, you'll see the last command you used. This fills up with duplicates. This is how you ensure it doesn't keep duplicates.

It's pretty easy. Just open your terminal and enter:

```
[code]echo 'export HISTCONTROL=ignoreboth:erasedups' >>  
~/.bashrc[/code]
```

After that, bash should no longer keep duplicates of previously entered commands.

See? Linux is pretty simple.

How To: Enable PPAs With Elementary OS

Elementary, like Debian and some other Debian derivatives, doesn't support PPAs out of the box. In order to use them, you'll have to install `software-properties-common`.

Open your terminal with CTRL + ALT + T and simply enter the following:

```
sudo apt install software-properties-common
```

Once you've done that, you can add PPAs in the normal way. For example, this is a LibreOffice PPA:

```
sudo add-apt-repository ppa:libreoffice/ppa
```

How To: Find the hostname.

The hostname is, for many of you who will be reading this, the same as your username. This is not always true.

Why is it important? Well, if I want to connect to a box on my network, I use its hostname. For example:

```
ssh kgiii@kgiii-lmde.local
```

That means I don't need to know the IP address of the box, I merely need to know the hostname. That, as I said, is usually your username. On the off-chance that it isn't, it's easy to find.

```
cat /proc/sys/kernel/hostname
```

And there you go.