V 1.2

How to set up Linux Ubuntu on a PC as a dual boot

This guide is for installing Linux Ubuntu 18.04 LTS alongside Windows 10. Please note that some of the modifications could potentially brick your machine or necessitate a new install of Windows if done without precautions. Proceed at your own risk.

Why Ubuntu?

Linux is more or less just the OS Kernel. Different orgs or companies use this kernel to put together a full OS. Some different flavors, called distributions, are: Debian, Red Hat, Mint, Linux Lite and a lot of others.

Linux is extensively used by deloppers and tech companies. It is the OS of reference for supercomputers and servers. A lot of cell phones and IoT devices run on Linux.

Ubuntu is distributed and maintained by the company Canonical. It is arguably one of the two or three most popular distributions of Linux. This means that bugs are fixed promptly, updates are regular and frequent and, more importantly, most of the applications found on a Mac running OSX or a PC/Win will exist on Ubuntu/Linux. Sometimes it is an equivalent.

Windows Sub-System for Linux

In short: WSL.

It allows the installation and use of Ubuntu 18.04 within Win10. It is essentially a dedicated local VM. It makes it possible to run Visual Studio Code, NodeJS and Github Desktop, ESlint (not global) and Testem (not global either).

It may encounter some difficulties running global packages. PostgreSQL should be able to function, but this is not confirmed as of now.

It is a nice environnement, and it is rather easy to set up. Main caveat is it is incomplete. It can work in a dual-boot configuration though.

First steps

What you need:

1 USB drive 8GB,

1 USB drive 32GB,

DVDs or some more USB drives.

60 GB minimum of free space on the Hard Drive where Ubuntu will reside

Before any action, prepare a Windows 10 system image with the 32GB USB drive.

Backup your files and documents, so you have them in case something goes wrong. Use some DVDs or other USB drives.

Determine the HD where Ubuntu will reside and free some space on it if needed. Note that it is possible to run Ubuntu from a USB drive or in a virtual machine on Windows. It will limit the apps which can be installed or used, though.

Useful links:

Bootable USB drive Linux

Guide Linux on PC

Image of Ubuntu

Download Ubuntu 18.04LTS. LTS means Long Term Support, and identify a stable version. That's what you want.

With the 8GB USB drive, create a bootable image of Ubuntu Linux. The links on the left are for 16.04LTS, just transpose for 18.04LTS.

Check if your BIOS is UEFI or Legacy, as the install process is a bit different. The Guide Linux on PC link describes the differences. You may have to disable BitLocker and/or SecureBoot.

Linux partition

Useful links:

Guide Linux on PC

If you have more than 1 HD, like me, it is preferable to install Linux on a different drive from the one supporting your Windows OS.

Start Disk Management and shrink the main partition on the Linux drive. A minimum of 60GB is advised for Linux Ubuntu, while leaving some space for your other files. A 500GB HD can accommodate a 150 to 200GB partition for Linux, and that is plenty (see notes at the end).

There is now a zone labelled “Unallocated Space”.

Install

Useful links:

Guide Linux on PC

Turn off the machine, and insert the drive in an empty USB slot.

Turn the PC back on and press F11 or the keys necessary to get to the BIOS and force the boot to be from a USB drive. On a Surface device, it is PowerOn button and SoundUp button.

Choose to use Ubuntu or install. For the following directions, let's assume the installation option has been chosen. Ubuntu will install in the Unallocated space precedently set up .

Bionic Beaver is alive!

Useful links:

Guide Linux on PC

PC clock adjustment

At that point, you are pretty much done. Congratulations!

Before any add-on, set up the boot sequence for your machine. If the sequence can be interrupted like on a desktop, it’s easy. On a laptop, it may be useful to get Ubuntu to boot first, then Windows loader can be chosen from the GRUB boot screen.

The clock may need to be adjusted between Ubuntu and Windows. See the link on the left.

Install on a Surface device

Useful links:

Guide Linux on PC

Bootable USB drive Linux

For Surface series

JakeDay's repo on GitHub.

Due to the touch functions and some Microsoft idiosyncrasies, the process is a bit more involved. See the link on the left.

To get the touch screen and pen to work, use the patches and modified Kernel from JakeDay’s repo. It also lists what works or not for a Surface device.

Git will need to be installed first.

NB: The clock adjustment is now part of his script.

Linux for Fullstack Academy

Useful links:

[Linuxbrew]

[Updating Linux Kernel]

[Things to do after 18.04 install]

[Fullstack Academy Toolbox]

You may want to get Linuxbrew first, though it is really optional . See the link, as usual. Next step would be to install an updated kernel, as 18.04 ships with Linux kernel 14.5.0, staying with the stable versions of course.

Then, there are a few things which can be installed. See that other link for some bloatware clean-up. Most of the games can be removed for instance.

Check out the Fullstack Academy Toolbox and install what is needed.

Linux for Fullstack Academy

Useful links:

Things to do after 18.04 install

Fullstack Academy Toolbox

From experience, it is best to start with Curl, NVM and NodeJS, as presented in the next slide.

Xcode tools and ITerm2 are Mac only, but the equivalent apps can be found pretty easily. On the other hand, the native Bash terminal is quite sufficient to start with.

Then the chosen editor, ESLint and Prettier can be installed and configured.

Installing NVM & NodeJS

Useful links:

Things to do after 18.04 install

Fullstack Academy Toolbox

NVM

NodeJS

There are a few ways to install NodeJS. For Ubuntu, the most reliable is through NVM (Node Version Manager).

Simply follow the directions in the 2 links on the left. It seems to be best to stick with a LTS version of NodeJS at the start. Having NVM will allow to manage and switch between different NodeJS versions.

Do not try to install NPM (Node Package Manager), as it comes already in with the NodeJS installation.

Installing PostgreSQL - part 1

Useful links:

Fullstack Academy Toolbox

PostgreSQL Install FSA

This is a necessary item, though not needed before some time into Junior phase.

Be cautious and take it slow, this part is pretty involved and can go sideways easily.

First, open a CLI window and update/upgrade the Ubuntu machine:

```
User@Linux ~$ sudo apt update

User@Linux ~$ sudo apt upgrade
```

Installing PostgreSQL - part 2

Useful links:

Fullstack Academy Toolbox

PostgreSQL Install FSA

Follow the directions to install Postgres then create a super-user. The username should be the same one registered for Ubuntu. In doubt, try using **psql.** The error will show which username is needed. The password can be left blank. The last step is to put the trust in the **pg_hba.conf** file. Literally.

At that stage, check with **psql** that the postgreSQL can be accessed. This is pretty much it, and should allow all the labs, workshops and personal projects to interact with postgreSQL through the scripts in place. Note that it may be necessary sometimes to create the database first then proceed to the npm set-up.

NB: restart postgreSQL by closing the CLI terminal and reopening then typing **psql**.

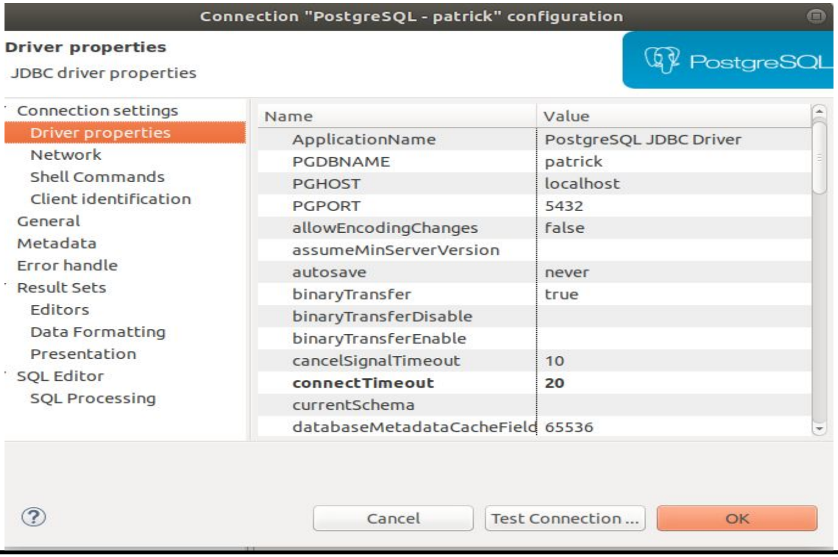
Useful links:

[DBeaver documentation]()

[DBeaver install]()

DBeaver : config

Setting up the connection in DBeaver can be a tad tricky. First, the JDBC driver needs to be installed, via the tab Driver.



Name	Value
ApplicationName	PostgreSQL JDBC Driver
PGDBNAME	patrick
PGHOST	localhost
PGPORT	5432
allowEncodingChanges	false
assumeMinServerVersion	
autosave	never
binaryTransfer	true
binaryTransferDisable	
binaryTransferEnable	
cancelSignalTimeout	10
connectTimeout	**20**
currentSchema	
databaseMetadataCacheField	65536

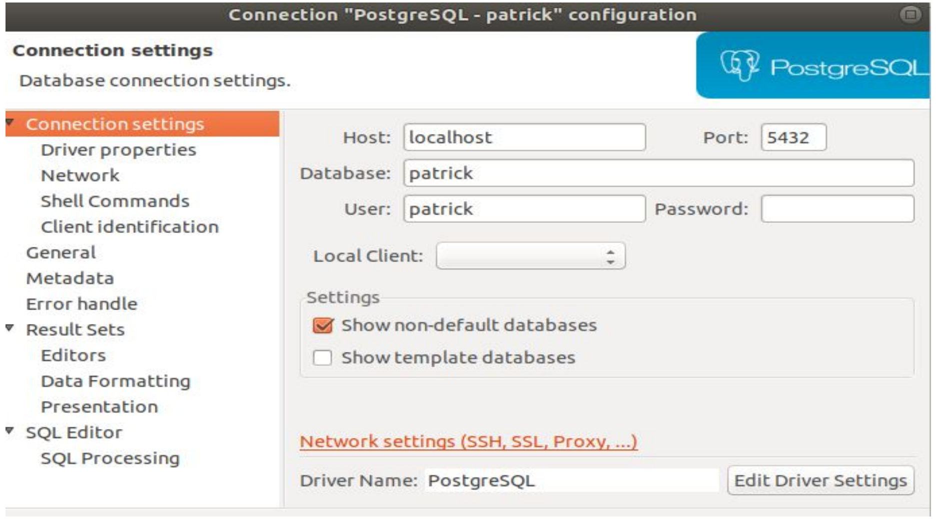
DBeaver : config

Useful links:

[DBeaver documentation]

[DBeaver install]

The connection parameters are mainly the user previously created, and the relevant password. Check in the DB window if the <your_user> database can be accessed, even if it is presently empty.



Connection "PostgreSQL - patrick" configuration

Connection settings

Database connection settings.

Host: localhost Port: 5432

Database: patrick

User: patrick Password:

Local Client:

Settings

[✓] Show non-default databases

[ ] Show template databases

Network settings (SSH, SSL, Proxy, ...)

Driver Name: PostgreSQL Edit Driver Settings

Which browser and IDE?

Useful links:

Guide Linux on PC

Fullstack Academy Toolbox

Firefox is a good option, but Chrome or its open source twin Chromium are probably better for development. Just the debugger, if it's the only aspect considered, is so worth the install. On the other hand, Firefox Inspector is quite good too. Your choice, or load them both.

As editor, Visual Studio Code is clearly recommended. Atom is another choice, as well as Sublime. Due to its real nice extensions and great interaction with GitHub, VSC is also my favorite. Your choice…

The missing tool here is GitHub Desktop which is a very useful piece of software. It helps a lot when managing complex projects, repos and Git. It is available on Win10 and MacOSX, though… If really needed, it is easy to switch to Win10 and apply the fix in the GitHub Desktop.

So much to chose from...

The Ubuntu and Linux community is very much alive and kicking. With a little research and some keen eyes, a lot can be mined and installed to make coding easier. Up to you!

Nevertheless, some advice is needed. Frequently, apps and software are at the experimental stage or very new. Bugs and flaws can be frequent. As it is often the case, use your best judgement and it may be better to err on the side of caution, unless one may feel comfortable with experimenting.

A few notes

Storage: Ubuntu does not need a huge amount of disk space. First it is a very compact OS, though other Linux distributions are even more lightweight. The trick is that the disk space under Windows management can be accessed from Ubuntu. Just mount and unmount the disk, and get some folders created to use that additional storage space.

Power management: This is a minus for Linux in general, it is not good at managing the power consumption. On a desktop, not such a big deal; on a laptop the battery will be eaten up 30 to 50% faster than with Windows. Just be aware of it. On top of that and depending on the machine, the battery level may also not be visible. This is even more true for Surface devices, which can get rather hot.

Some useful stuff

Useful links:

Installing Postman

psql cheat sheet

Battery indicator

At this point, the machine is pretty well stacked. Some of the following can be very useful:

Postman (highly recommended, great for testing routes and middleware / backend results).

Psql cheat sheet. The kind of handy link to have somewhere, just in case.

Tweaks: it adds some nice features, especially a battery indicator. Works on single battery laptops, not the Surface Book 2 :(

A few notes

All those modifications have been tested on:

Custom-built desktop PC i-5 4690K overclocked, 16GB RAM, 2TB storage (SSD/2HD), GTX 1070 NVidia 8GB,

Surface Pro tablet (2017) i-7, 8GB RAM, 256GB SSD,

Surface Book 2 i-7, 16GB RAM, 1TB SSD, GTX 1060 NVidia 8GB.

All are running Windows 10 - April 2018.