

Version 1: 2021-06-07 19:30

MacOS: already runs on top of BSD Unix, so just follow the directions at [ROOT Install](#) .
For example, using [HomeBrew](#) , type the following commands in the terminal (cut and paste each line starting after \$ without typing enter at any line breaks if a command spans multiple lines):

```
$ xcode-select --install
$ /bin/bash -c "$(curl -fsSL
https://raw.githubusercontent.com/Homebrew/install/HEAD/install.sh) "
$ brew install root
$ root
```

Ubuntu: Run the following commands from the terminal (ie bash; you will need to type in your Ubuntu password again, and respond to prompted questions)

```
$ sudo apt-get update
$ sudo apt-get install dpkg-dev cmake g++ gcc binutils libx11-dev
libxpm-dev libxft-dev libxext-dev python libssl-dev
$ sudo apt-get install gfortran libpcre3-dev xlibmesa-glu-dev
libglew1.5-dev libftgl-dev libmysqlclient-dev libfftw3-dev
libcfitsio-dev graphviz-dev libavahi-compat-libdnssd-dev libldap2-dev
python-dev libxml2-dev libkrb5-dev libgs10-dev libqt4-dev

$ wget
https://root.cern/download/root_v6.24.00.Linux-ubuntu20-x86_64-gcc9.3
.tar.gz
$ tar -xzf root_v6.24.00.Linux-ubuntu20-x86_64-gcc9.3.tar.gz
$ source root/bin/thisroot.sh
$ root
```

Two commands must be run every time you start a new terminal before running ROOT.
You can automatically execute them by adding them to your '.bashrc' file with the following commands:

```
$ echo ' export DISPLAY=:0' >> ~/.bashrc
$ echo 'source root/bin/thisroot.sh' >> ~/.bashrc
```

You will need a text editor. Here is how to install and run emacs:

```
$ sudo apt-get install emacs
$ emacs &
```

The following commands will add 'octave' and 'latex' to your distribution if you want them:

```
$ sudo apt-get install octave octave-optim octave-control
octave-signal octave-statistics octave-io octave-struct liboctave-dev
$ sudo apt-get install texlive texlive-publishers aspell
```

If you are interested in compiling ROOT from scratch, I suggest the following procedure to organize everything in /usr/local with possibility for multiple versions of root

```
$ cd /usr/local; sudo chmod a+wx . * && mkdir -p tgz build
$ ROOT=6.24.00
$ cd tgz; wget
https://root.cern.ch/download/root_v$ROOT.source.tar.gz
$ tar -xzf root_v$ROOT.source.tar.gz -C ../src
$ cd /usr/local/build && mkdir -p root-$ROOT && cd root-$ROOT
$ cmake ../../src/root-$ROOT
```

```
$ cmake --build . -- -j4
$ cd /usr/local; ln -s build/root-$ROOT root
$ echo '. /usr/local/root/bin/thisroot.sh' >> ~/.bashrc
$. ~/.bashrc
$ root
```

Windows: install WSL2 using the [Manual Instruction Steps](#) to run Ubuntu Linux as a virtual machine (see below), and then install ROOT in Ubuntu terminal as detailed in the Ubuntu section.

1) open up PowerShell as Administrator by typing 'W-x' (windows key + 'x') and then 'a' and clicking 'yes' to make changes to your system, and type:

```
PS C:\WINDOWS\system32> dism.exe /online /enable-feature  
/featurename:Microsoft-Windows-Subsystem-Linux /all /norestart
```

2) find out your version of Windows in the same shell: (you need to update unless the build newer or equal to 18362.1049 or 18363.1049)

```
PS C:\WINDOWS\system32> winver
```

3) enable virtual machines in the same shell

```
PS C:\WINDOWS\system32> dism.exe /online /enable-feature  
/featurename:VirtualMachinePlatform /all /norestart
```

Restart your computer.

4) download and run the kernel update

https://wslstorestorage.blob.core.windows.net/wslblob/wsl_update_x64.msi

5) set the default to WSL2 in the same shell

```
PS C:\WINDOWS\system32> wsl --set-default-version 2
```

6) install [Ubuntu 20.04 LTS](#) from the Windows store (click "get" and then "run", and create a username and password, which is only for Ubuntu)

Also, install the [VcXsrv](#) graphics server in Windows to view graphics windows in Linux.

To start it, click on "XLaunch" in the start menu, click Next twice, unselect "Native opengl", click Next, and click Finish.

Before trying to display graphics (like running ROOT), specify which server to use by typing the following in the Ubuntu terminal:

```
$ export DISPLAY=:0
```