



# Introducción al biocómputo en sistemas UNIX/Linux

- ¿Qué es Linux? - Evolución de sistemas UNIX y similares a UNIX (fuente: Wikipedia)

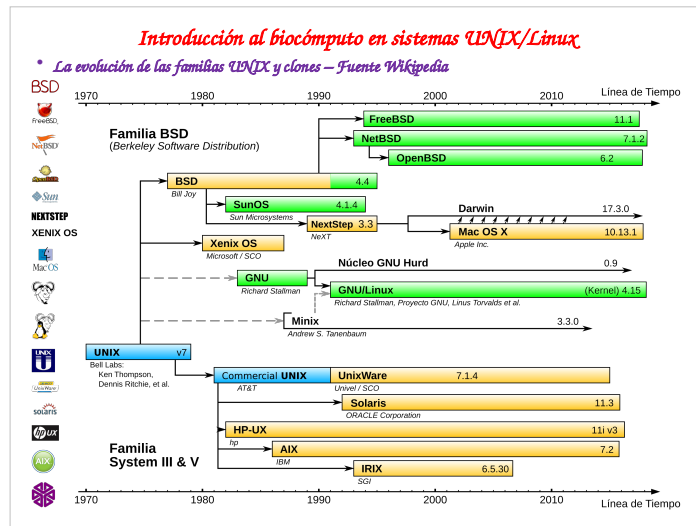
GNU: En 1983, **Richard Stallman** anunció el **Proyecto GNU**, un ambicioso esfuerzo para crear un **sistema similar a Unix, que pudiese ser distribuido libremente**. El software desarrollado por este proyecto -por ejemplo, **GNU Emacs** y **GCC** - también han sido parte fundamental de otros sistemas UNIX. (vean conferencias de R. Stallman en youtube ...)



*Richard Stallman*      **POWERED BY  
Linux**      *Linus Torvalds*

**Linux:** En 1991, cuando **Linus Torvalds** empezó a proponer el **núcleo Linux** y a reunir colaboradores, las herramientas GNU eran la elección perfecta. Al combinarse ambos elementos, conformaron la base del sistema operativo (basado en **POSIX**) que hoy se conoce como **GNU/Linux**.

Las **distribuciones** basadas en el núcleo, el software GNU y otros agregados como **Red Hat Linux** y **Debian GNU/Linux**, se han hecho populares tanto entre los aficionados a la computación como en el mundo empresarial y científico. Linux tiene un origen independiente, por lo que se considera un 'clon' de UNIX y no un UNIX en el sentido histórico.



## Introducción al biocómputo en sistemas UNIX/Linux

• Tengo una PC que corre windows, ¿cómo puedo correr Linux en mi máquina?

1. Puedes instalar Linux en una nueva partición (Lo más recomendable).

descargas gratuitas de distribuciones desde:

=> 1. **Ubuntu 22.04 LTS** - <http://www.ubuntu.com/getubuntu/download>

Ver instrucciones de instalación aquí:

<http://www.ubuntu.com/download/desktop/install-desktop-latest>

Centos - <https://www.centos.org/download/>

Fedora - <http://fedoraproject.org/es/get-fedora>

Biolinux - <http://environmentalomics.org/bio-linux/>

2. Puedes instalar MobaXterm, que proporciona una terminal para Windows con un servidor de ambiente gráfico X11, un cliente SSH para establecer sesiones remotas seguras con un servidor, diversas herramientas de red y más.

<https://mobaxterm.mobatek.net/download.html>

## Introducción al biocómputo en sistemas UNIX/Linux

• Tipos de sesiones shell

1) local (acceder a tu computadora)

```
vinuesa@alisio: ~/Cursos/TIB/TIB19-T3/sesion1_intro2linux
Archivo Editar Ver Buscar Terminal Ayuda
vinuesa@alisio:~/Cursos/TIB/TIB19-T3/sesion1_intro2linux$ ls
assembly_summary.txt.gz  linux_very_basic_commands.table.csv
intro_biocomputo_linux_pt1.odp  Stenotrophomonas_complete_genomes_and_ftg_paths.txt
intro_biocomputo_linux_pt1.pdf  working_with_linux_commands.code
linux_basic_commands.tab  working_with_linux_commands.html
linux_commands.tab  working_with_linux_commands.Rmd
vinuesa@alisio:~/Cursos/TIB/TIB19-T3/sesion1_intro2linux$
```

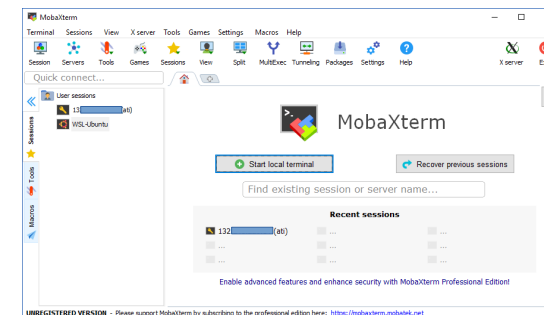
2) remota (login; conexión a otra máquina en Internet, vía protocolo ssh)

```
vinuesa@bonampak:~
Archivo Editar Ver Buscar Terminal Ayuda
vinuesa@alisio:~$ ssh -X vinuesa@132.248.220.35
vinuesa@132.248.220.35's password:
Last failed login: Mon Jul 22 17:15:08 CDT 2019 from akbal.ccg.unam.mx on ssh:notty
There was 1 failed login attempt since the last successful login.
Last login: Mon Jul 22 17:05:42 2019 from akbal.ccg.unam.mx
vinuesa@bonampak ~$
```

Si tu máquina corre Windows, descarga el MobaXterm installer desde:

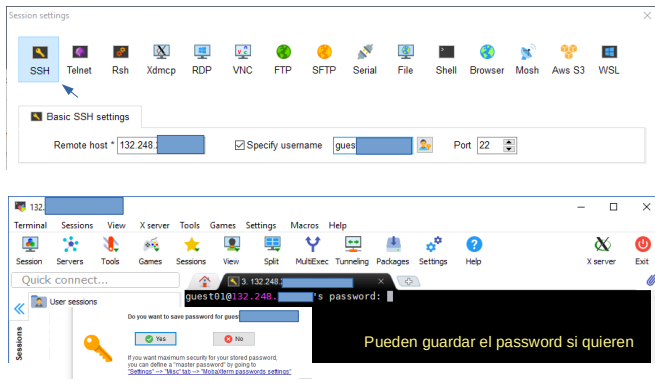
[https://download.mobatek.net/2202022022680737/MobaXterm\\_Installer\\_v22.0.zip](https://download.mobatek.net/2202022022680737/MobaXterm_Installer_v22.0.zip)

• Descomprime MobaXterm\_Installer\_v22.0.zip e instálalo en tu máquina (x2click)



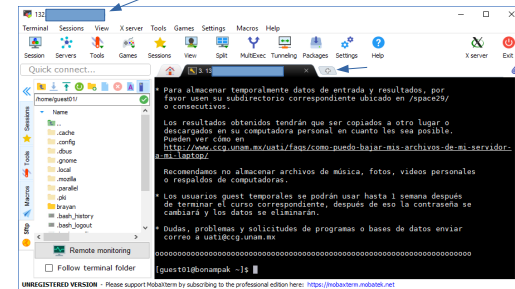
### Uso de MobaXterm para establecer sesiones remotas a un servidor vía protocolo SSH

1. Para establecer una conexión remota vía SSH elige el protocolo SSH y usa la IP, usuario y contraseña indicados en el taller



### Uso de MobaXterm para establecer sesiones remotas a un servidor vía protocolo SSH

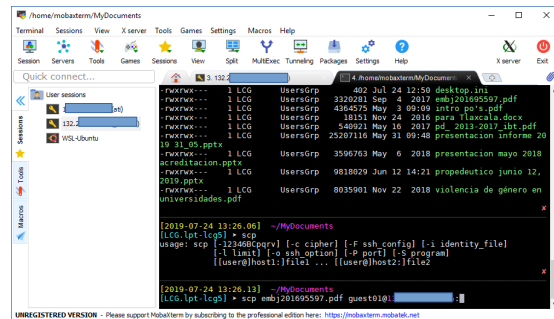
Una vez establecida la sesión remota al servidor, estarán trabajando en él/



- El panel mostrado a la izquierda les sirve para navegar el sistema remoto, pero lo haremos desde la línea de comandos que nos ofrece la terminal (área negra).
- Pueden abrir más pestañas en la terminal. Estas serán sesiones locales (si no hacen ssh)

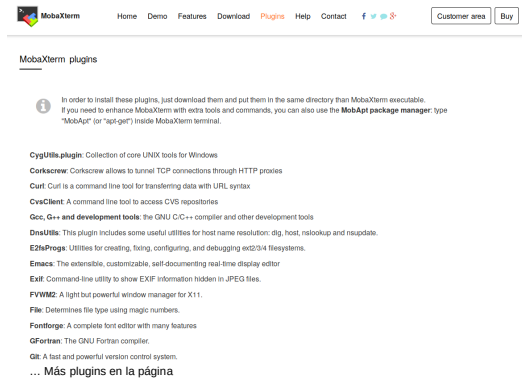
### Uso de MobaXterm para trabajo con sesiones remotas y locales

Una vez establecida la sesión remota al servidor, estarán trabajando en él/



- El panel mostrado a la izquierda tiene abierta una sesión remota.
- La terminal de la derecha corre una sesión local (en tu computadora)
- Pueden copiar un archivo de su máquina local al servidor con:  
\$ scp file user@xxx.xxx.xxx.xxx:/path/to/final/dir

### Expande MobaXterm con plugins <https://mobaxterm.mobatek.net/plugins.html>



- **Instalan sin falta Git** para poder clonar y actualizar el repositorio GitHub del taller
- Recomendamos instalar también GCC dev. Tools, Nedit, Perl y Zip

Sobre uso de la WiFi del auditorio Dr. Guillermo Soberón durante los TIB2022 en

Nodo Nacional de Bioinformática UNAM

Conócenos

Nuestros talleres

Contacto

EBM2022

Talleres

Calendario

Cuotas

Registro

Agenda

Pangenómica y Filogenómica Microbiana

Para un uso eficiente del ancho de banda limitado que tenemos para conectarnos al servidor en el que haremos todos los trabajos prácticos, pedídmlos de la manera más atenta, en el beneficio del buen funcionar del taller, que durante las sesiones de clases:

1. No conecten teléfonos u otros dispositivos móviles a la WiFi del auditorio y los pongan en silencio
2. Que no accedan a sus redes sociales o salgan con su browser a Internet desde su laptop
3. Desplieguen las instrucciones para las prácticas abriendo el archivo html correspondiente descargado en sus máquinas localmente desde el repositorio GitHub del taller

<https://github.com/vinuesa/TIB-filoinfo>

Introducción al biocómputo en sistemas UNIX/Linux

• explorando el nuevo ambiente ... El shell y la interpretación de comandos, unos ejemplos

1. — ¿Qué procesos se están corriendo en el sistema?

• top - monitor system load

vinuesa@bonampak:/\$ top

top - 19:13:22 up 184 days, 3:14, 9 users, load average: 1.93, 2.10, 2.13  
Tasks: 669 total, 2 running, 667 sleeping, 0 stopped, 0 zombie  
%Cpu(s): 1.9 us, 2.0 sy, 0.0 ni, 96.1 id, 0.0 wa, 0.0 hi, 0.0 si, 0.0 st  
KiB Mem : 52807612+total, 4045748 free, 11101945+used, 41301091+buff/cache  
KiB Swap: 13421772+total, 13331803+free, 899696 used, 41411052+avail Mem

PID	USER	PR	NI	VIRT	RES	SHR	S	%CPU	%MEM	TIME+	COMMAND
84375	llozano	20	0	9587984	5.1g	1388	R	139.7	1.0	48721:34	spades-hammer
191547	ati	20	0	67.3g	63.0g	63.0g	S	109.9	12.5	322566:44	VBoxHeadless
181971	ati	20	0	34.4g	32.2g	32.2g	S	1.3	6.4	124996:03	VboxHeadless
53501	ropomari	20	0	158872	2684	1224	S	0.7	0.0	0:03.54	sshd
1507	root	20	0	13216	524	524	S	0.3	0.0	26:48.50	rngd
54660	vinuesa	20	0	162509	2824	1576	R	0.3	0.0	0:00.26	top

...

Introducción al biocómputo en sistemas UNIX/Linux

• explorando el nuevo ambiente ... El shell y la interpretación de comandos, unos ejemplos

1. — ¿Qué máquina es ésta a la que estoy conectado?

• hostname - read or set the hostname or the NIS domain name

vinuesa@bonampak:/\$ hostname  
bonampak.ccg.unam.mx  
vinuesa@bonampak:/\$ hostname -i # corre también hostname --help  
132.248.\*.\*

• uname - Print certain system information

vinuesa@bonampak:~\$ uname  
Linux  
vinuesa@bonampak:~\$ uname -a  
Linux bonampak.ccg.unam.mx 3.10.0-862.9.1.el7.x86\_64 #1 SMP Mon Jul 16 16:29:36 UTC 2018 x86\_64 x86\_64 x86\_64 GNU/Linux

Introducción al biocómputo en sistemas UNIX/Linux

• explorando el nuevo ambiente ... El shell (consola) y la interpretación de comandos

- ¿Qué es el shell?

UNIX/  
Linux  
System  
kernel

Utilidades

Memoria Virtual (RAM)

Discos físicos

Los sistemas UNIX/Linux se dividen lógicamente en dos piezas: el kernel y las utilidades.

El kernel es el corazón del sistema y reside en la memoria de la computadora desde el momento que se arranca y hasta que se apaga.

Las utilidades (comandos), residen en el disco físico y se cargan en memoria sólo cuando son llamadas.

El shell también es un programa. Se carga automáticamente en memoria desde que uno hace el login a una máquina para que el usuario pueda interactuar con ella.

© Pablo Vinuesa, 2022. [vinuesa\[at\]ccg.unam.mx](mailto:vinuesa[at]ccg.unam.mx);  
<http://www.ccg.unam.mx/~vinuesa/> @pvinmex

4

## Introducción al biocómputo en sistemas UNIX/Linux

\* explorando el nuevo ambiente ... El shell y la interpretación de comandos

- Las responsabilidades del **shell** -



## Introducción al biocómputo en sistemas UNIX/Linux

\* explorando el nuevo ambiente ... El shell y la interpretación de comandos

- ¿Dónde encuentro una lista y descripción básica de los comandos disponibles ?

1. Mira estas entradas en Wikipedia:

[http://en.wikibooks.org/wiki/Linux\\_Guide/Linux\\_commands](http://en.wikibooks.org/wiki/Linux_Guide/Linux_commands)

[http://en.wikipedia.org/wiki/List\\_of\\_Unix\\_programs](http://en.wikipedia.org/wiki/List_of_Unix_programs)

2. Y corre estos comandos para ver parte de los programas del sistema instalados en el servidor o en tu máquina:

`ls /bin`

`ls /usr/bin`

3. Un sencillo tutorial que todos deberían haber revisado ya (caps 1-5):

<http://www.ee.surrey.ac.uk/Teaching/Unix/>

## Introducción al biocómputo en sistemas UNIX/Linux

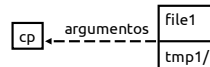
\* explorando el nuevo ambiente ... El shell y la interpretación de comandos

- Ejecución de programas por el shell - ejemplos

\* formato básico de un comando

comando [argumento1 arg2 arg3 ...]

```
-bash-3.1$  
-bash-3.1$ cp file1 tmp1/  
-bash-3.1$
```



**¡los espacios separan argumentos!**

necesito al menos 1 espacio entre comandos y argumentos!  
Nombres de archivos en UNIX/Linux de preferencia no deben contener espacios

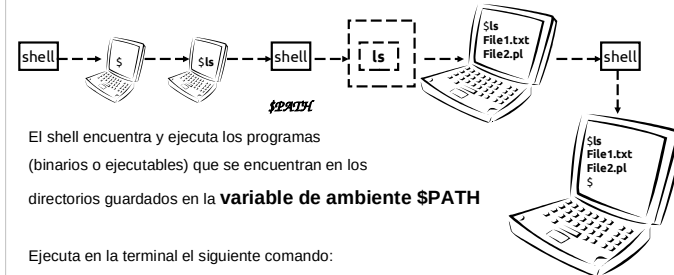
\* Además de **argumentos**, los comandos pueden tener **opciones**, las cuales preceden a los argumentos y llevan un guión sencillo delante de una o más opciones.

```
-bash-3.1$ ls -l /home # prueba también solo ls /home  
drwxr-xr-x. 2 aagarcia students 4096 Oct 30 2014 aagarcia  
drwxr-xr-x. 2 aanaya students 4096 Oct 30 2014 aanaya  
drwxr-xr-x. 2 acampos students 4096 Oct 30 2014 acampos  
drwxr-xr-x. 2 acarmona students 4096 Oct 30 2014 acarmona  
drwxr-xr-x. 4 agodinez students 4096 Jul 1 2016 agodinez  
drwxr-xr-x. 2 aguzman students 4096 Oct 30 2014 aguzman
```

## Introducción al biocómputo en sistemas UNIX/Linux

\* explorando el nuevo ambiente ... El shell y la interpretación de comandos

- Tecleando comandos para el shell - el ciclo de comandos



El shell encuentra y ejecuta los programas  
(binarios o ejecutables) que se encuentran en los  
directorios guardados en la **variable de ambiente \$PATH**

Ejecuta en la terminal el siguiente comando:

(**echo** imprime el valor de una variable y/o una lista de argumentos)

```
-bash-3.1$  
vinuesa@bonampak:/$ echo $PATH
```

## Commandos Básicos

- **ls (list)**
  - \$ ls -l
  - \$ ls -a
  - \$ ls -la
  - \$ ls -l --sort=time # ls -ltr
  - \$ ls -l --sort=size -r #ls -lSr
  - \$ ls \*.txt
- **cd (change directory)**
  - \$ cd dir
  - \$ cd ./dir
  - \$ cd ../
  - \$ cd .././
  - \$ cd /export/space2/tib/filo
- **pwd (print working directory)**
  - \$ pwd
- ~
- ~ \$ cd ~
- ~user
  - \$ cd ~vinuesa
- ¿Qué hará "cd ~vinuesa"?
- ¿y cd?
- **which**
  - \$ which blastn
- **locate**
  - \$ locate get\_homologues.pl
  - \$ locate mi\_archivo
- **find**
  - \$ find / | grep stdio.h
  - \$ find /usr/include | grep stdio.h
  - \$ find . -type d
  - \$ find /home/vinuesa -name \*TIB\*
- **man (manual pages for command)**
  - \$ man ls
  - \$ man find
  - \$ man man

## Commandos Básicos (cont.)

- **echo (print to STDOUT)**
  - \$ echo "Hello World"
  - \$ echo -n "Hello World"
- **cat (concatenate)**
  - \$ cat /proc/cpuinfo
  - \$ cat arch1 arch2
- **cp (copy)**
  - \$ cp arch1 dir1
  - \$ cp -r dir1 ~vinuesa/tmp
- **mv (move or rename)**
  - \$ mv arch1 archivo1
  - \$ mv arch1 ~vinuesa/tmp
- **mkdir (make directory)**
  - \$ mkdir dir2
  - \$ mkdir -p dir2/practica1
- **rm (remove)**
  - \$ rm arch1
  - \$ rm -rf dir2
- **less (paginador)**
  - \$ less archivo.txt # q para salir
- **head (ver cabecera del archivo)**
  - \$ head -5 archivo1.txt
- **tail (ver cola del archivo)**
  - \$ tail -1 archivo.txt
  - \$ tail -f logfile.txt
- **sed (stream editor)**
  - sed 's/esto/aquello/' archivo.txt
- **vim (vi improved; a powerful command line text editor in Linux)**
- **gedit (editor de texto con interfaz gráfica en gnome)**
- **nedit (otro editor de texto con interfaz gráfica)**

## Commandos Básicos (cont.)

- **tar & gzip**
  - \$ tar cvzf dir1.tar.gz dir1
  - \$ tar cvzf dir1.tgz dir1
  - \$ tar cvpf dir1.tar dir1
- **gzip (gunzip, compress)**
  - \$ gzip -9 dir1.tar
  - # genera dir1.tar.gz
- **untar & ungzip**
  - \$ tar xvfz dir1.tar.gz
- **touch**
  - \$ touch foo
- **head**
  - \$ head archivo.txt
- **tail**
  - \$ tail archivo.txt
  - \$ tail -f archivo2.txt
- **|, >, grep, sort, cut, uniq**
  - \$ ls -l /home | less
  - \$ cat /etc/passwd | cut -d: -f1 | sort
  - \$ ls -l /home | grep stud | wc -l > stud.txt
  - \$ ls -l /home | grep -v stud | sort | uniq -c
- **backticks**
  - \$ echo "The date is `date`"
  - \$ echo `seq 1 10`
- **Hard, soft (symbolic) link**
  - ln -s /path/to/gbk\_files/\*.gbk .
- **scp (secure copy)**
  - scp arch1 vinuesa@buluc.lcg.unam.mx: \$HOME/tmp

## Commandos Básicos (cont)

- **Uso del disco duro**
  - \$ df -h /
- **Uso discro de archivos**
  - \$ du -sxh ~/
- **Uso avanzado: programación del Shell ☺**
  - **Asignación de variables y bucles for**
    - var1=123; echo \$var1; var2=/home/pepin && echo \$var2
    - for file in \*faa; do muscle < \$file > \${file%.\*}\_aln.faa; done
    - for file in \$(ls \*faa | grep rpoB); do echo -n \$file; grep -c '>' \$file; done

## Linux text editors

- con entorno gráfico
  - **gedit**
  - **nedit**
- Sin entorno gráfico
  - **Vim**
  - **emacs**
  - **nano**
  - **pico**

## Vim – the Linux power text editor

- 2 modes
  - Input mode
    - ESC to back to cmd mode
  - Command mode
    - Cursor movement
      - h (left), j (down), k (up), l (right)
      - ^f (page down)
      - ^b (page up)
      - ^ (first char.)
      - \$ (last char.)
      - G (bottom page)
      - :1 (goto first line)
    - Switch to input mode
      - a (append)
      - i (insert)
      - o (insert line after)
      - O (insert line before)
- Delete
  - dd (delete a line)
  - d10d (delete 10 lines)
  - d\$ (delete till end of line)
  - dG (delete till end of file)
  - x (current char.)
- Paste
  - p (paste after)
  - P (paste before)
- Undo
  - u
- Search
  - /
- Save/Quit
  - :w (write)
  - :q (quit)
  - :wq (write and quit)
  - :q! (give up changes)

## Introducción al biocómputo en sistemas UNIX/Linux

## \* explorando el nuevo ambiente ... Trabajando eficientemente con el shell

¿Cómo me muevo en la línea de comandos?

- Usa **ctrl-e** para ir al final de la línea
- Usa **ctrl-a** para ir al principio de la línea

¿Cómo edito la línea de comandos?

- Usa la techa **backspace** para eliminar uno a uno caracteres (del final hacia el principio)
- Usa **ctrl-w** para eliminar una palabra completa (del final hacia el principio)
- Usa **ctrl-u** para eliminar la línea completa (del final hacia el principio)

¿Cómo aborto o suspendo la ejecución de un comando?

Usa **ctrl-c** para abortar la ejecución del último comando

- Usa **ctrl-z** para suspender la ejecución del último comando
- Usa **bg** para poner este último comando a correr en el fondo (background)

Repetición de la ejecución de un comando y completado de nombres de comandos/archivos

- Unix recuerda los comandos ejecutados: usa flecha arriba o abajo para moverte por el historial de comandos "history" file o escribe **history | grep comando**
- Usa **TAB** para completar automáticamente el nombre de comandos, archivos o directorios

## Introducción al biocómputo en sistemas UNIX/Linux

## \* explorando el nuevo ambiente ... Trabajando eficientemente con el shell

¿Cómo me muevo en la línea de comandos?

- Usa **ctrl-e** para ir al final de la línea
- Usa **ctrl-a** para ir al principio de la línea

¿Cómo edito la línea de comandos?

- Usa la techa **backspace** para eliminar uno a uno caracteres (del final hacia el principio)
- Usa **ctrl-w** para eliminar una palabra completa (del final hacia el principio)
- Usa **ctrl-u** para eliminar la línea completa (del final hacia el principio)

¿Cómo aborto o suspendo la ejecución de un comando?

Usa **ctrl-c** para abortar la ejecución del último comando

- Usa **ctrl-z** para suspender la ejecución del último comando
- Usa **bg** para poner este último comando a correr en el fondo (background)

Repetición de la ejecución de un comando y completado de nombres de comandos/archivos

- Unix recuerda los comandos ejecutados: usa flecha arriba o abajo para moverte por el historial de comandos "history" file o escribe **history | grep comando**
- Usa **TAB** para completar automáticamente el nombre de comandos, archivos o directorios

**Prácticas – asegúrate de tener clonado y actualizado el repositorio GitHub**  
<https://github.com/vinuesa/TIB-filoinfo>



1. Instala git en tu máquina (**sesión local**) con `$ sudo apt install git`
2. genera un directorio GitHub en tu \$HOME y clona el repositorio así:  

```
$ cd && mkdir GitHub && cd GitHub  
git clone https://github.com/vinuesa/TIB-filoinfo.git
```