

Temario de la clase

Temario

- Acceso remoto a un servidor linux
- Google colab. Introducción a su uso.
- Google colab - google drive.
- Uso de la shell en colab.

En la solapa clases tienen la filminas nuevas.

Acceso remoto a un servidor

- ▶ Las máquinas que están en linux pueden accederse en forma remota.
- ▶ Se puede trabajar y ejecutar programas en forma remota.
- ▶ En general se accede a una terminal y solo se utiliza esta.
- ▶ Se puede exportar aplicaciones con ventanas. **Pero hay que evitarlas!**
(por la velocidad de la red).

Es decir que todo lo que podemos hacer lo tenemos que hacer en una terminal!

secure shell (ssh) desde windows

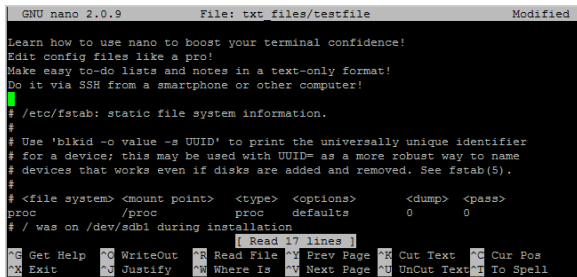
- ▶ El protocolo que se utiliza actualmente para acceder a una computadora o servidor remoto es ssh (secure shell).
- ▶ Es una shell como la que conocemos excepto que transmite la información encriptada por seguridad.
- ▶ Desde windows se puede acceder a maquinas linux instalando aplicaciones en general libres.
- ▶ Una muy recomendable para esto es PUTTY.
- ▶ Desde el linux se puede usar desde una terminal y hacer ssh (Instalar la aplicación si no se encuentra)

Domicilio de los servidores remotos (o nombre identificador)

- ▶ Existen dos tipos de nombres para los servidores en la red. Uno basado en **caracteres** (nombre de dominio) y otro basado en **números**.
- ▶ Numero de Internet Protocol (IP). e.g. 171.25.10.155 (4 números cada uno de 8 bits es decir desde 0 hasta 255 aunque hay numeros reservados por protocolo).
- ▶ Por nombres: (estos estan definidos en un domain name server: servidor de nombres de dominios). e.g maquina.unne.edu.ar
- ▶ Los nombres siguen una estructura de dominios (gran dominio ar, luego edu etc).

Editores de textos en una terminal

- ▶ Si necesitan escribir un programa o script en forma remota en el servidor.
- ▶ Abrir un editor de textos en la terminal el mas sencillo **nano** (recomendado), mas específicos **vi** o vim, **emacs** (emacs -nw).
- ▶ Para editar un archivo (existente o nuevo) hacen: nano /directorio/archivo



```
GNU nano 2.0.9      File: txt_files/testfile      Modified

Learn how to use nano to boost your terminal confidence!
Edit config files like a pro!
Make easy to-do lists and notes in a text-only format!
Do it via SSH from a smartphone or other computer!
#
# /etc/fstab: static file system information.
#
# Use 'blkid -o value -s UUID' to print the universally unique identifier
# for a device; this may be used with UUID= as a more robust way to name
# devices that works even if disks are added and removed. See fstab(5).
#
# <file system> <mount point>   <type>   <options>   <dump>   <pass>
proc          /proc                proc      defaults    0         0
# / was on /dev/sdb1 during installation

[ Read 17 lines ]

^G Get Help  ^O WriteOut  ^R Read File ^Y Prev Page ^K Cut Text   ^C Cur Pos
^X Exit      ^J Justify   ^W Where Is ^V Next Page ^U UnCut Text ^T To Spell
```

Nano

Abajo tienen los shortcuts en general Ctrl + algo (o con las Funciones).

- ▶ Ctrl + g es la Documentación.
- ▶ Ctrl + o escribir (grabar)
- ▶ Ctrl + x Salir de nano.
- ▶ Ctrl + w Buscar palabra o texto. Para marcar: Ctrl + 6
- ▶ Ctrl + k cortar texto.
- ▶ Ctrl + u pegar texto copiado/cortado
- ▶ Para copiar texto tiene que hacer: Ctrl + k y Ctrl + u (o Alt + 6)

Transferencia de archivos

Si uno quiere pasar archivos entre la computadora que esta trabajando y el servidor se puede utilizar scp / sftp. Son protocolos de transferencia de archivos encriptados.

En general como el servidor es el que tiene una direccion publica siempre usan el comando desde su computadora.

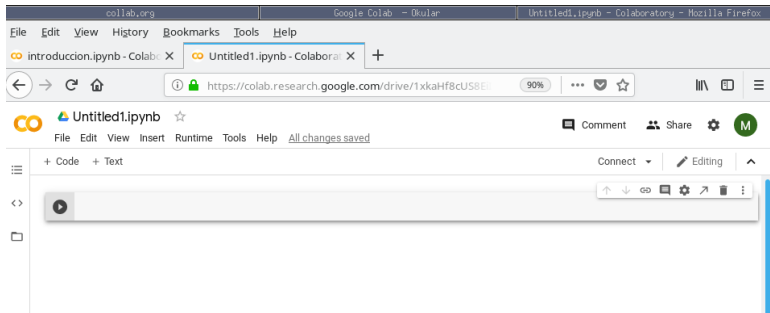
- ▶ Para transferir desde la computadora al servidor: `scp archivo.txt 200.45.54.94:directorioservidor/`
- ▶ Para transferir desde el servidor a la computadora: `scp 200.45.54.94:directorioservidor/archivo.txt ./`

En el caso de windows pueden instalar PuTTY scp que es para transferir archivos (pscp).

Google colab

Herramienta de google que pone a disposición una **máquina virtual** en linux para que nosotros podamos utilizar desde un navegador:

- ▶ En un navegador ir al sitio <https://colab.research.google.com>
- ▶ Deben tener abierta la cuenta de gmail solicitada para que el google les permita trabajar.
- ▶ Generen una nueva notebook.



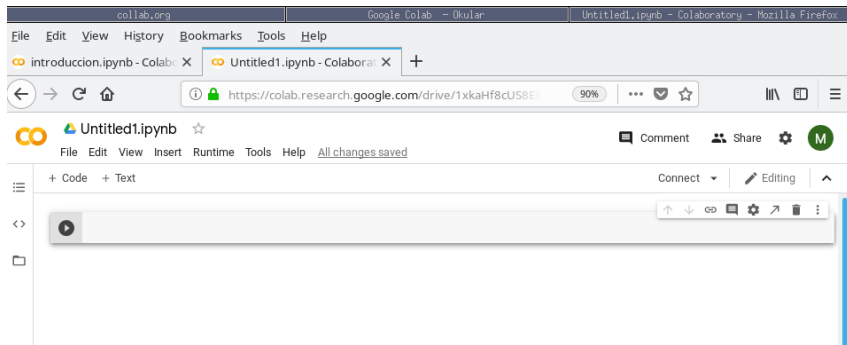
Características de google colab

Google colab esta basado en una aplicación de python: [jupyter notebook](#)

- ▶ El formato de escritura sigue el estandard de jupyter.
- ▶ Entiende y trabaja en el lenguaje python.
- ▶ Permite escribir programas en forma amigable. (iterate programming)

En una notebook (cuaderno de laboratorio) se mezclan códigos, con textos y también las salidas/resultados de los códigos tanto textos como gráficos.

Características de la template



El primer borrador le pone Untitled1.ipynb. Renombren la notebook y le ponen introduccion.ipynb

La extensión es obligatoria y es la definida para las notebooks de jupyter.

Primer programa muy tonto en colab

Introduzcan en una celda las siguientes instrucciones:

```
a=15.
```

```
b=38.
```

```
print(a,'x',b,'es: ',a*b)
```

Para ejecutar la celda:

- ▶ Clickear la regla de la derecha de la celda
- ▶ Menu Runtime. Run focused Cell.
- ▶ Ctrl + Enter (Mas rápida)

La celda entonces es un conjunto de instrucciones parte de la notebook.

Generar una nueva celda

Al final de la solapa de la celda que ya usaron tiene una opción de introducir código o texto (o debajo del menú). Agregar una celda nueva y tipear:

```
c=b/a
```

```
print(c)
```

Ejecutar (regla de la derecha de la celda)

Realizó el cálculo con las variables que tenía en la celda anterior!.

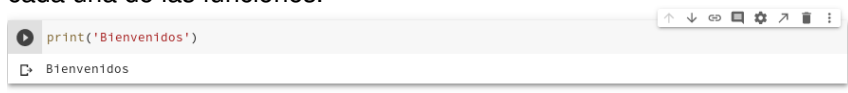
Es decir que cada una de las instrucciones que hemos ido introduciendo colab las deja en memoria.

Cuando creamos esta nueva celda las variables b y a, y todas las variables de las celdas anteriores que usemos ya quedaron definidas.

Navegación entre celdas

- ▶ Clickeando con el ratón en cualquiera de las celdas anteriores nos permite retomarlas.
- ▶ Cambien la variable `a=5` en la primera celda y luego ejecutan la primera y la segunda celda.

También se puede cambiar el orden de las celdas, borrarlas, etc. Utilizando el menú que hay en el margen superior izquierdo de cada celda van a ver cada una de las funciones.



Guardado de archivos

Se pueden guardar (save) tanto en el Google Drive como también en github (sitio web para repositorios de software).

Vamos a trabajar con **Google Drive** ya que el github requiere de conocimientos mas complejos de versionador de software (git).

- ▶ Ir al Menu, Solapa de File y luego van a ver la opción "Save a copy in drive"
- ▶ Guardar el archivo como introduccion.ipynb
- ▶ Abrir el google drive y ver que lo ha guardado en el directorio Colab Notebooks

Linux y la shell en colab

Por detrás de google-colab hay una máquina virtual en **linux** (ubuntu) con todas las funciones de linux incluyendo una terminal.

Para ejecutar instrucciones de bash/terminal en colab deben introducir el signo de admiración !

Ejemplos: En las celdas hagan

```
!ls  
!cd /; ls
```

Es decir que todos los comandos de bash los tienen en colab.

```
!ls /bin
```

les va a dar una lista de los comandos (ejecutables binarios) que pueden utilizar. Como verán todos los comandos bash estan allí, también hay comandos para manejar el sistema (e.g. journalctl, systemctl, systemd-*, mount, lsmod, etc.)

Montado del google drive en la máquina virtual de colab

Como es una máquina virtual todo lo que esta allí (la estructura de directorios/archivos etc) desaparece una vez que dejan de usar la aplicación.

Para que puedan guardar cosas pueden montar el google drive de uds. en la máquina virtual:

```
from google.colab import drive  
drive.mount('/content/gdrive')
```

Les va a pedir una clave de autenticación ya que google colab va a acceder a los archivos de google drive (otra aplicación).

Alternativa: Vayan a Tools pongan m (mount drive) y allí les introducirá el comando.

Para escribir todo una celda de bash

Existe un conjunto de lenguajes que se pueden utilizar en una celda en colab a través de **magics**.

Los lenguajes que entiende son varios: html, latex, javascript, ruby, **shell**, **bash**.

```
%%bash
```

Todo lo que sigue en la celda será en lenguaje bash.

```
%%bash
echo 'Entro a la shell'
cd /bin
echo 'Cambio de directorio ahora estoy en:'
pwd
echo 'Archivos que hay en este directorio:'
ls
echo
echo 'Salgo de la shell'
```