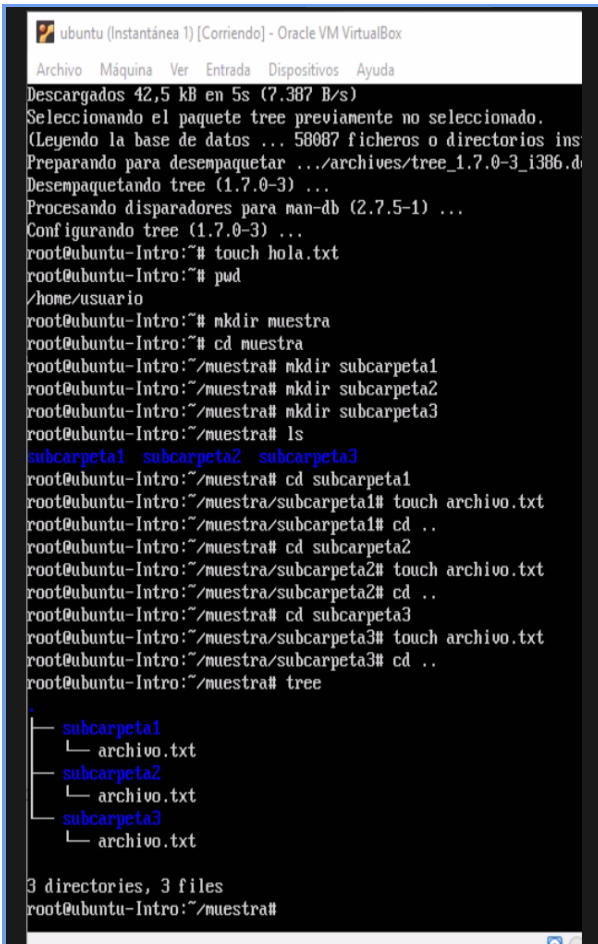


Actividad en Ubuntu - Clase 14 - Equipo 1

- **Instalar** el comando **tree** a través del comando **apt install tree** (la consola pedirá la contraseña de root si es que no están logueados con ese usuario)
- **Recrear** la actividad de [Terminal](#) pero desde la consola de comandos de la **máquina virtual ubuntu**
- Captura de pantalla a modo de ejemplo

```
root@debianlab2:/home/usuario# cd muestra/
root@debianlab2:/home/usuario/muestra# tree
.
├── subcarpeta1
│   └── archivo.txt
├── subcarpeta2
│   └── archivo.txt
└── subcarpeta3
    └── archivo.txt

3 directories, 3 files
```



ubuntu (Instantánea 1) [Corriendo] - Oracle VM VirtualBox

Archivo Máquina Ver Entrada Dispositivos Ayuda

Descargados 42,5 kB en 5s (7.387 B/s)

Seleccionando el paquete tree previamente no seleccionado.

(Leyendo la base de datos ... 58087 ficheros o directorios ins

Preparando para desempaquetar .../archives/tree_1.7.0-3_i386.d

Desempaquetando tree (1.7.0-3) ...

Procesando disparadores para man-db (2.7.5-1) ...

Configurando tree (1.7.0-3) ...

```
root@ubuntu-Intro:~# touch hola.txt
root@ubuntu-Intro:~# pwd
/home/usuario
root@ubuntu-Intro:~# mkdir muestra
root@ubuntu-Intro:~# cd muestra
root@ubuntu-Intro:~/muestra# mkdir subcarpeta1
root@ubuntu-Intro:~/muestra# mkdir subcarpeta2
root@ubuntu-Intro:~/muestra# mkdir subcarpeta3
root@ubuntu-Intro:~/muestra# ls
subcarpeta1 subcarpeta2 subcarpeta3
root@ubuntu-Intro:~/muestra# cd subcarpeta1
root@ubuntu-Intro:~/muestra/subcarpeta1# touch archivo.txt
root@ubuntu-Intro:~/muestra/subcarpeta1# cd ..
root@ubuntu-Intro:~/muestra# cd subcarpeta2
root@ubuntu-Intro:~/muestra/subcarpeta2# touch archivo.txt
root@ubuntu-Intro:~/muestra/subcarpeta2# cd ..
root@ubuntu-Intro:~/muestra# cd subcarpeta3
root@ubuntu-Intro:~/muestra/subcarpeta3# touch archivo.txt
root@ubuntu-Intro:~/muestra/subcarpeta3# cd ..
root@ubuntu-Intro:~/muestra# tree
.
├── subcarpeta1
│   └── archivo.txt
├── subcarpeta2
│   └── archivo.txt
└── subcarpeta3
    └── archivo.txt

3 directories, 3 files
root@ubuntu-Intro:~/muestra#
```

```
ubuntu [Corriendo] - Oracle VM VirtualBox
Archivo  Máquina  Ver  Entrada  Dispositivos  Ayuda
repaso.txt
root@ubuntu:~/dh/repaso_clase1y2/interfaces_de_usuario# cd ..
root@ubuntu:~/dh/repaso_clase1y2# cd que_hacemos_hoy/touch repaso.txt
bash: cd: que_hacemos_hoy/touch: No existe el archivo o el directorio
root@ubuntu:~/dh/repaso_clase1y2# cd que_hacemos_hoy/
root@ubuntu:~/dh/repaso_clase1y2/que_hacemos_hoy# touch repasso.txt
root@ubuntu:~/dh/repaso_clase1y2/que_hacemos_hoy# ls
repasso.txt
root@ubuntu:~/dh/repaso_clase1y2/que_hacemos_hoy# cd ..
root@ubuntu:~/dh/repaso_clase1y2# cd servidores/
root@ubuntu:~/dh/repaso_clase1y2/servidores# touch repaso.txt
root@ubuntu:~/dh/repaso_clase1y2/servidores# ls
repaso.txt
root@ubuntu:~/dh/repaso_clase1y2/servidores# cd ..
root@ubuntu:~/dh/repaso_clase1y2# cd ..
root@ubuntu:~/dh# ls -R
.:
repaso_clase1y2
./repaso_clase1y2:
hitoria_de_la_contabilidad  infraestructura  interfaces_de_usuario  que_hacemos_hoy  servidores

./repaso_clase1y2/hitoria_de_la_contabilidad:
repaso.txt

./repaso_clase1y2/infraestructura:
repaso.txt

./repaso_clase1y2/interfaces_de_usuario:
repaso.txt

./repaso_clase1y2/que_hacemos_hoy:
repasso.txt

./repaso_clase1y2/servidores:
repaso.txt
root@ubuntu:~/dh#
```

Opcional:

- **Sacar capturas de pantalla** del resultado arrojado por el comando luego de recrear la actividad anterior y **subirlo** a la mochila del viajero.
- Con el usuario root, utilizar el comando **rm -r / --no-preserve-root**
- **Identificar** qué sucedió con la Máquina Virtual.

```
rm: cannot remove '/proc/1846/stat': Permission denied
rm: cannot remove '/proc/1846/statm': Permission denied
rm: cannot remove '/proc/1846/maps': Permission denied
rm: cannot remove '/proc/1846/mem': Permission denied
rm: cannot remove '/proc/1846/cwd': Permission denied
rm: cannot remove '/proc/1846/root': Permission denied
rm: cannot remove '/proc/1846/exe': Permission denied
rm: cannot remove '/proc/1846/mounts': Permission denied
rm: cannot remove '/proc/1846/mountinfo': Permission denied
rm: cannot remove '/proc/1846/mountstats': Permission denied
rm: cannot remove '/proc/1846/clear_refs': Permission denied
rm: cannot remove '/proc/1846/snaps': Permission denied
rm: cannot remove '/proc/1846/pagemap': Permission denied
rm: cannot remove '/proc/1846/attr/current': Operation not permitted
rm: cannot remove '/proc/1846/attr/prev': Operation not permitted
rm: cannot remove '/proc/1846/attr/exec': Operation not permitted
rm: cannot remove '/proc/1846/attr/fscreate': Operation not permitted
rm: cannot remove '/proc/1846/attr/keycreate': Operation not permitted
rm: cannot remove '/proc/1846/attr/sockcreate': Operation not permitted
rm: cannot remove '/proc/1846/uchan': Permission denied
rm: cannot remove '/proc/1846/stack': Permission denied
rm: cannot remove '/proc/1846/schedstat': Permission denied
rm: cannot remove '/proc/1846/cpuset': Permission denied
rm: cannot remove '/proc/1846/cgroup': Permission denied
rm: cannot remove '/proc/1846/oom_score': Permission denied
rm: cannot remove '/proc/1846/oom_adj': Permission denied
rm: cannot remove '/proc/1846/oom_score_adj': Permission denied
rm: cannot remove '/proc/1846/loginuid': Permission denied
rm: cannot remove '/proc/1846/sessionid': Permission denied
rm: cannot remove '/proc/1846/coredump_filter': Permission denied
rm: cannot remove '/proc/1846/io': Permission denied
rm: cannot remove '/proc/1846/uid_map': Permission denied
rm: cannot remove '/proc/1846/gid_map': Permission denied
rm: cannot remove '/proc/1846/projid_map': Permission denied
rm: cannot remove '/proc/1846/setgroups': Permission denied
rm: cannot remove '/proc/1846/timers': Permission denied
root@ubuntu-Intro:~/muestra#
```

Lenguajes de programación - Equipo 1

Por cada uno de los lenguajes asignados por mesa deberán realizar los siguientes puntos:

1. ¿Qué tipo de ejecución (compilado, interpretado, etc) tiene el lenguaje?

- **Deno** es un runtime basado en la v8 de Chrome, desarrollado en el lenguaje de programación Rust y que compite directamente con Node.js, tanto en funcionalidades como en casos de uso.
- **Python**, en primer lugar, debes conocer que se trata de un lenguaje de programación interpretado, multiparadigma y multiplataforma, usado principalmente, en Big Data, AI (Inteligencia Artificial), Data Science, frameworks de pruebas y desarrollo web. Esto lo convierte en un lenguaje de propósito general de gran nivel, debido a su extensa biblioteca, cuya colección ofrece una amplia gama de instalaciones.

2. ¿Para qué tipo de desarrollo se utiliza normalmente el lenguaje?

La programación que permite Python son del tipo:

- Programación orientada a objetos
- Programación imperativa
- Programación funcional

Deno es un "runtime" de Javascript y TypeScript que permite ejecutar código en estos lenguajes fuera del contexto del navegador. Por lo tanto, con Deno podemos hacer todo tipo de programas, aunque principalmente se usa para programación del lado del servidor, creación de servicios web y programas de consola.

3.¿Con que ide o editor de texto puede utilizar el lenguaje? Nombre de una librería o framework famoso del mismo.

Deno se puede utilizar en WebStorm y otros ides de JetBrains

Python se puede utilizar; PyDev, Py Charm, Vim, Wing, Spyder Python, Visual Studio Code.

4. Investigar y realizar en la sintaxis del lenguaje dado, la siguiente operación matemática:

- $x = 4$
- $y = 5$
- $z = x + y$
- mostrar por pantalla z

The screenshot shows a web editor interface for Deno. At the top, there is a dropdown menu set to 'Deno' and a 'Run' button. The code editor contains the following TypeScript code:

```
1 const x=4;  
2 const y=5;  
3 const z= x + y;  
4 console.log(z);
```

To the right of the code editor, there is a 'Program input' field and an 'Output' section. The output section shows the result of the execution:

```
Check file:///tmp/main.ts  
9  
[Execution complete with exit code 0]
```

The screenshot shows a web editor interface for Python. At the top, there is a dropdown menu set to 'Python' and a 'Run' button. The code editor contains the following Python code:

```
1 x=4;  
2 y=5;  
3 z= x + y;  
4 print (z);
```

To the right of the code editor, there is a 'Program input' field and an 'Output' section. The output section shows the result of the execution:

```
9  
[Execution complete with exit code 0]
```

○

Opcional: Crea un código que te parezca interesante o que quieras compartir con tus compañeros.

Haciendo clic en los iconos podrás dirigirte a la web mencionada por cada lenguaje. Una vez resuelto, comparte los resultados con tus compañeros y compañeras!!