

1. En este semestre las actividades prácticas oficiales del curso serán realizadas usando el servicio **Cocalc**, que suministra una plataforma virtual con herramientas de Linux, $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}$, Fortran y Python, entre otras. En esta plataforma podrá almacenar los archivos que usted creará durante el curso (aunque siempre es recomendable hacer una copia de respaldo en otro lugar). Como primer paso, cree una cuenta en **Cocalc**, ya sea registrando sus datos o bien usando sus credenciales ya existentes de Google, Facebook o Twitter.
2. Una vez que haya ingresado a Cocalc, seleccione la opción “Create New Project” (botón verde) y cree un nuevo proyecto titulado “Linux”, donde almacenará los archivos de su trabajo en el primer módulo (en prácticas futuras creará otros proyectos titulados “LaTeX”, “Fortran” y “Python”).
3. Ingrese al proyecto “Linux” recién creado y luego seleccione “Create or Upload Files” y a continuación la opción “>_ Linux terminal”. Con esto, Cocalc creará un terminal virtual dentro de su proyecto, titulado con la fecha de su creación. En esa consola virtual podrá realizar todos los ejercicios de comandos Bash propuestos a continuación.
4. En el terminal virtual (o “consola”) que creó en el punto anterior, realice las siguientes tareas:
 - (a) Liste el contenido de su carpeta, ejecutando el comando

```
ls
```

- (b) Cree el directorio o carpeta **test1**, ejecutando el comando

```
mkdir test1
```

Verifique la creación de la carpeta ejecutando nuevamente el comando `ls`.

- (c) Ingrese a la carpeta recién creada, ejecutando el comando

```
cd test1
```

y en su interior cree una nueva carpeta llamada **test2**.

- (d) Mire [este](#) video en Youtube, donde se explican los comandos `man`, `nano`, `vi` y `wget`.
- (e) Como vio en el video, en la consola de Linux están disponible (en la mayoría de las instalaciones) algunos *editores de texto*. En particular, aprenderá a usar el editor “nano”. Con este editor, puede crear y modificar archivos de texto (ascii/utf8). Por ejemplo, ejecutando el comando

```
nano archivo1.txt
```

usted ingresará al editor nano y podrá crear y/o modificar el archivo **archivo1.txt**. Explore las opciones de este editor de texto. ¿Cómo se guardan los archivos creados/modificados?. ¿Cómo se sale del editor?.

- (f) Mueva el archivo **archivo1.txt** a **archivo2.txt**, ejecutando el comando

```
mv archivo1.txt archivo2.txt
```

En este caso, el efecto es equivalente a haber renombrado el archivo.

(g) Cambie el nombre de la carpeta **test2** a **temp**.

5. Conozca el comando **cat**, que **concatena** (es decir, une) archivos y los despliega en la pantalla (la salida estándar).

(a) Infórmese de los aspectos generales de este comando desplegando el manual, con

```
man cat
```

(b) En la consola, cree dos archivos **test01.txt** y **test02.txt** usando el editor **nano** e introduzca en ellos algún texto interesante.

(c) Ejecute ahora el comando

```
cat test01.txt test02.txt
```

Observe cómo los dos archivos se despliegan en pantalla, uno luego del otro.

(d) ¿Cuál es la diferencia del comando anterior respecto a

```
cat *.txt
```

Compruebe su respuesta con un ejemplo.

6. Descargue el archivo **Quijote.tar.gz** (desde la dirección <https://github.com/gfrubi/CC/raw/master/guias/01/Quijote.tar.gz>) a su computador local y luego súbala a su proyecto en Cocalc. Para esto, ingrese a la pestaña “Files” en su proyecto y luego en “Upload”. Este archivo contiene (en forma comprimida) cinco archivos correspondientes a algunas secciones iniciales de la obra “Don Quijote de la Mancha” de Miguel de Cervantes. Abra una consola e ingrese (usando el comando **cd**) a la carpeta donde se encuentra el archivo descargado y realice las siguientes tareas:

(a) Descomprima el archivo, usando el comando

```
tar -xf Quijote.tar.gz
```

Esto creará cinco archivos con extensión **.txt**.

(b) Infórmese de los aspectos generales del comando **tar**, desplegando el manual del comando **tar** (ingrese **man tar**).

(c) Despliegue el contenido de cada archivo usando el comando **more**.

(d) Una los archivos (en orden!) y guarde el resultado en el archivo **Quijote.txt**. Para esto, la opción **>** para redireccionar la salida de **cat** al archivo correspondiente.

7. Conozca el comando **grep**, que busca (conjuntos de) palabras dentro de archivos de texto.

(a) Busque información en internet sobre este comando. Por ejemplo, realice la búsqueda “comando grep linux” en Google. Éste es un método que usualmente permite encontrar rápidamente información útil. Por ejemplo, el correspondiente [artículo de Wikipedia](#) es un buen punto de partida.

- (b) Usando **grep**, busque todas las ocurrencias de la palabra **Mancha** en cada uno de los cinco archivos originales del Quijote (**c1.txt ... c5.txt**)
 - (c) Agregue la opción **-n** al comando **grep** usado anteriormente (es decir, use **grep -n**). ¿Qué efecto tiene sobre el resultado?
8. Puede escribir y ejecutar una secuencia de comandos almacenados en un “*script*” (archivo de comandos).
- (a) Para este ejemplo, cree una nueva carpeta y cree en ella una copia del archivo comprimido **Quijote.tar.gz**.
 - (b) En esta carpeta, cree un archivo con nombre **mi-script.sh** (la extensión **.sh** es opcional, pero conveniente). En este archivo escriba una variación de los comandos antes usados, cada uno en una línea separada, que realicen las siguientes acciones: 1) descomprima el archivo **Quijote.tar.gz**, 2) Una los archivos **.txt** y crea el archivo **Quijote.txt** y 3) busca todas las ocurrencias de la palabra **batalla** en este archivo y 4) crea el archivo **resultado.txt** con el resultado de la búsqueda.
 - (c) Ejecute su script (es decir, los comandos que contiene el archivo **mi-script.sh** en el orden en que están escritos), ejecutando
- ```
bash mi-script.sh
```
9. Usando comandos Bash, encuentre todas líneas del texto del Quijote en las que se menciona la palabra “Mancha” (primera letra en mayúsculas!), y almacene el texto de estas líneas en el archivo **Manchas.txt**.
10. Puede usar el tiempo restante para aprender sobre otros comandos Bash. Algunos de ellos están listados [este artículo de Wikipedia](#).