

Cuestionario 01

Carlos Emiliano Sandoval Amador
315025222

14 de enero de 2019

1. Dé tres sistemas operativos.
Linux, iOS y Windows.
2. Dé dos distribuciones de Linux.
Ubuntu y Fedora.
3. ¿Cuál es el lenguaje de programación que usamos en el curso?
Python.
4. ¿Cuál es la característica principal de Linux?
Es open source.
5. ¿Qué es un shell?
Un interprete que nos permite utilizar los servicios del sistema operativo. En el caso de Linux caso ese shell es un intérprete de comandos llamado bash.
6. ¿Para qué sirve ls?
El comando ls sirve para mostrar los directorios y archivos contenidos en el directorio que le indiques. Si no le indicas un directorio, te mostrará el contenido del directorio en el que estés parado.
Utilizando ls -a verás una lista de todos los archivos incluidos los archivos ocultos.
7. ¿Para qué sirve cd?
Con el comando cd puedes moverte a directorios que estén contenidos en el directorio en el que te encuentras. Si no especificas un directorio cd te regresará al directorio home, este contiene a todos los demás directorios.

Con `cd ..` puedes moverte al directorio que contiene al directorio en el que te encuentras.

8. ¿Para qué sirve `man`?
Muestra el manual del comando que desees, de esta forma puedes aprender a utilizarlos o descubrir funciones que antes no conocías.
9. ¿Para qué sirve `mkdir`?
Crea un nuevo directorio contenido en el directorio en el que te encuentres. Utilizando la extensión `-p` puedes crear directorios anidados.
10. ¿Para qué sirve `rmdir`?
Elimina un directorio contenido en el directorio en el que te encuentres siempre y cuando este esté vacío.
11. ¿Para qué sirve `history`?
Muestra una lista con los comandos usados en la última sesión.
12. ¿Para qué sirve `touch`?
Con el comando `touch` puedes crear un archivo contenido en el directorio en el que estés. Solo tienes que escribir `touch` seguido del nombre del archivo que desees crear.
13. ¿Para qué sirve `less`?
El comando `less` muestra el contenido de un archivo. Si el archivo es binario el contenido del archivo mostrado por `less` será imposible de entender por el ser humano.
14. ¿Cómo entrar a `vi`?
Tenemos que escribir `vi` en la consola seguido de el nombre del archivo, si el archivo no existe `vi` creará uno nuevo con ese nombre.
15. ¿Cómo escribimos en `vi`?
Aprietando la tecla `i`.
16. ¿Cómo guardamos lo que escribimos?
Presionando `:w`.
17. ¿Cómo salimos?
Presionando `:q`.

18. ¿Cómo salimos a la fuerza?
Presionando :q!.
19. ¿Qué es git?
Git es un software de control de versiones pensando en la eficiencia y la confiabilidad del mantenimiento de versiones de aplicaciones cuando éstas tienen un gran número de archivos de código fuente. Su propósito es llevar registro de los cambios en archivos de computadora y coordinar el trabajo que varias personas realizan sobre archivos compartidos.
20. ¿Qué es un repositorio?
El lugar donde se guardan estos archivos y se notifican los cambios ejercidos sobre ellos.
21. ¿Qué hace init?
Con él podemos crear un nuevo repositorio, el cuál se visualizará como un directorio en nuestra computadora. Para usarlo tenemos que escribir **git init** seguido del nombre del repositorio.
22. ¿Qué hace clone?
Sirve para copiar un repositorio que esté en el sitio web de github, para utilizarlo hay que escribir **git init** seguido del url del repositorio que queremos copiar. Si ese repositorio no es nuestro no podremos subir a la web las modificaciones que le hagamos.
23. ¿Qué hace config --global?
Este comando sirve para guardar nuestro nombre de usuario y correo de github para operaciones posteriores. Para utilizarlo escribimos **git config --global user.name** seguido de nuestro nombre de usuario entre comillas. Para el correo escribimos **git config --global user.mail** seguido de nuestro correo entre comillas.
24. ¿Qué hace add?
Coloca los archivos nuevos donde corresponden. Para usarlos hay que poner **git add** seguido del documento que queremos colocar. para colocar todos los archivos podemos usar **git add ***.
25. ¿Qué hace commit?
Sirve para guardar y notificar los cambios hechos a los archivos. Es más

efectivo utilizar **git commit -m** seguido de un mensaje entre comillas que notifique los cambios hechos a los archivos.

26. ¿Qué hace push?
Con push podemos subir a github todos los cambios hechos a los archivos una vez que hayamos hecho utilizado config, add y commit.
27. ¿Qué hace pull?
Este comando sirve para bajar actualizar un repositorio cuya versión más actual se encuentra en github.
28. ¿Cómo podemos imprimir en python?
Utilizando la función **print()**, solo tenemos que ingresar adentro de los paréntesis el tipo de dato que queramos imprimir, es decir, string, entero, flotante, lista o tupla.
29. ¿Cómo asignamos un valor a una variable?
Utilizando **=**.
30. ¿Cómo sumamos?
Utilizando **+**.
31. ¿Cómo restamos?
Utilizando **-**.
32. ¿Cómo multiplicamos?
Utilizando *****.
33. ¿Cómo dividimos?
Utilizando **/**.
34. ¿Cómo elevamos a una potencia?
Utilizando ******.
35. ¿Cómo hacemos un string?
Utilizamos las comillas y dentro ponemos la cadena de texto que deseamos.
36. ¿Cómo agregamos datos a un string?
Para cada dato que ingresemos a la cadena de texto vamos a necesitar un caracter de % seguido de una letra que indica el tipo de dato que queremos ingresar. Estas letras están asignadas de la siguiente manera.

- a)* g y f para flotantes.
- b)* E y e para flotantes con notación científica.
- c)* d para enteros.
- d)* s para strings.

37. ¿Cómo definimos una función en python?

Para definir una función tenemos que utilizar la palabra reservada `def` seguida del nombre de la función y unos paréntesis dentro de los cuáles están los parámetros de la función.