



Taller de Comandos Linux

COLOQUE SU NOMBRE

2024-07-

1 Objetivos

- Conocer el funcionamiento del terminal de *Linux*
- Aprender a manejar comandos del terminal de *Linux*

2 Instrucciones

Este taller/tutorial de comandos de Linux tiene por objetivo ejercitar en la aplicación de comandos estudiados en clase. Para esto siga las Instrucciones indicadas en cada uno de los numerales. Ejecute los ejercicios llenando los comandos adecuados dentro de los bloques de código de Emacs Org. Los resultados de la ejecución se obtienen haciendo `C-c C-c`

Para todos los ejercicios se sugiere o bien usar directamente un distro de *Linux* o el *WSL* configurado para la clase de *Arquitectura de Computadores*.

1. Utilice `cd` para localizarse en la posición de home de su usuario y verifique ejecutando el comando `pwd`
2. Cree tres directorios denominados FashionData, Train y Test. Verifique usando el comando `ls`
3. Investigue sobre el comando `wget` y utilícelo para descargar en `~/FashionData` los datos de train y test del `/Fashion-Mnist` desde el siguiente enlace: `fashion-mnist`
 - (a) Imágenes de entrenamiento:
`http://fashion-mnist.s3-website.eu-central-1.amazonaws.com/train-images-idx3-ubyte.gz`
 - (b) Etiquetas de entrenamiento:
`http://fashion-mnist.s3-website.eu-central-1.amazonaws.com/train-labels-idx1-ubyte.gz`
 - (c) Imágenes de prueba:
`http://fashion-mnist.s3-website.eu-central-1.amazonaws.com/t10k-images-idx3-ubyte.gz`
 - (d) Etiquetas de prueba:
`http://fashion-mnist.s3-website.eu-central-1.amazonaws.com/t10k-labels-idx1-ubyte.gz`
4. Utilice el comando `mv` de manera recursiva para pasar los datos de entrenamiento a la carpeta `~/Train`
5. Utilice el comando `mv` de manera recursiva para pasar los datos de prueba a la carpeta `~/Test`
6. Verifique que la carpeta FashionData está vacía

7. Los archivos anteriores están comprimidos. Investigue como descomprimir usando el comando `tar`. Si es necesario realice la instalación del comando. Descomprima los archivos en sus respectivas carpetas. Apunte el comando utilizado para la descompresión.
8. Una vez completa la descompresión, elimine los archivos `.gz`. Verifique las carpetas finales usando `ls`
9. Mueva recursivamente las carpetas `~/Train` y `~/Test` dentro de `~/FashionData`
10. Se desea realizar un registro climatológico de la ciudad de Quito. Para esto, escriba un script de Python/Java que permita obtener datos climatológicos desde el API de openweathermap. Considere la latitud y la longitud de Quito como `(-0.2299, -78.5249)`, respectivamente. Los resultados obtenidos de la consulta al API se escriben en un archivo `clima-quito-hoy.csv`. Cada ejecución del script debe almacenar nuevos datos en el archivo. Investigue sobre **crontab** y utilícelo para obtener datos del API de *openweathermap* cada 5 minutos mediante la ejecución de un archivo ejecutable denominado `get-weather.sh`. Verifique los resultados. Todas las operaciones se realizan en Linux o en el WSL. Las etapas del problema se subdividen en:
 - (a) Crear su API gratuito en openweathermap
 - (b) Escribir un script en Python/Java que realice la consulta al API y escriba los resultados en `clima-quito-hoy.csv`
 - (c) Desarrollar un ejecutable `get-weather.sh` para ejecutar el programa Python/Java.
 - (d) Configurar Crontab para la adquisición de datos. Escriba el comando configurado.