Basic tools refreshers:
Homework for lecture 2

Joris LIMONIER

September 19, 2021

1. mkdir lesson_1

2. mv tic lesson_1/tic

3. mkdir lesson_2
   cd lesson_2

4. touch file1.1.1
   echo -e “Hello\nHello\nHello” >> file 1.1.1
   echo -e “world\nworld” >> file 1.1.1
   echo -e “*” >> file 1.1.1

5. cat file1.1.1
   less file1.1.1

6. Prints all subdirectories from root.

7. tree /etc > tree_etc

8.

9.

10. wget http://download.geonames.org/export/zip/FR.zip
    unzip FR.zip -d ./
    rm FR.zip

11. less FR.txt
    This file is a table.

12. grep ”01000” FR.txt
    The cities are Bourg-en-Bresse and Saint-Denis-lès-Bourg.

13. (Not to be done)

14. mv FR.txt french_zipcodes.txt

15. ls french_zipcodes.txt -l

    The user and the group have read and write rights, whereas the other users only have read right.

16. (a) chmod u+rwx french_zipcodes.txt

    (b) chmod u-x french_zipcodes.txt

    (c) chmod u-w french_zipcodes.txt

17.
    ```
    chmod u+rwx french_zipcodes.txt
    chmod u-w french_zipcodes.txt
    ```

18. Because the rights is ”- r - - r w - r - -”, group users already have read and write (*i.e.* edit). Now to prevent them from deleting this file, we run the following commands:

    ```
    cd ..
    ls -l (to check the current rights for lesson_2, which returns drwxrwxr-x)
    chmod g-w lesson_2/ (and now the rights of lesson_2 are drwxr-xr-x)
    ```

    In order to allow group users to delete french_zipcodes.txt, we would have to run chmod g+w lesson_2/.

19.
    ```
    HELLO=“Hello, World !”
    echo $HELLO
    ```

20.
    ```
    for entry in ../lesson_1/*
    do
      echo “$entry”
    done >> LONG_FILES_LIST
    ```

21.
    ```
    touch user_reply
    read reply
    echo $reply ¿ user_reply
    ```

22.
    ```
    touch hello.sh
    nano hello.sh
    read user_name
    echo ”Hello $user_name !”
    ctrl+X
    [ y ]
    [ Enter ]
    chmod a+x hello.sh
    ./hello.sh
    Joris
    (prints Hello Joris !)
    ```

23. Let us first create a file in which to print ”Error message”:

    ```
    touch err_mess
    ```

    Now we create a shell file and make it executable:

    ```
    touch q23.sh
    ```

```
chmod a+x q23.sh
nano q23.sh
echo ”Normal message”
echo ”Error message” ¿¿ $1
Ctrl + X
y
Enter
./q23.sh err_mess
```

As expected, ”Normal message” is printed in the terminal and ”Error message” is written to the file err_mess.

24.
    ```
    mv $1 $2
    mv $2 ”$3/$2”
    echo ”Nb of arguments is $#”
    echo ”Arguments from input are $1 $2 $3”
    ```

25.
    ```
    for i in $(seq 50 99)
    do
      touch file$i
    done
    ```

26.
    ```
    rm file[5-9][0-9]
    ```

27. (a)
    ```
    for i in $(seq 1 $1)
    do
      touch $3”/$2$i”
    done
    ```

    (b)
    ```
    for i in $(seq 1 $1)
    do
      mv ”$2/$3$i” ”$2/$3$i.txt”
    done
    ```

    (c)
    ```
    for name in $(ls .txt)
    do
      mv ”$1$name” ”$2$name”
    done
    ```