

1. Carpetas en /home

- /bin : almacena los archivos binarios ejecutables y todos los comandos de uso común están en este directorio
 - en la carpeta bin esta la carpeta que corresponde a la instalación de python 2.7
- /etc : contiene los archivos de configuración requeridos por todos los programas
 - dentro de etc/ podemos encontrar los archivos de configuración del programa passwd, que es el encargado de la administración de contraseñas para control de acceso al sistema
- /dev : contiene los archivos que hacen referencia a los dispositivos como usb o terminales
 - en dev/ podemos encontrar archivos y carpetas como cdrom, disk o net, que se encargan de almacenar los archivos de control para cada tipo de dispositivo.
- /tmp : contiene todos los archivos temporales creados por el sistema y los usuarios
 - dentro de tmp/ podemos encontrar el archivo bitácora de yum que es el comando que sirve como administrador de paquetes de software
- /usr : contiene los binarios, librerías, documentación y código fuente para programas de segundo nivel
 - dentro de usr/ podemos encontrar su propia carpeta bin para almacenar los binarios de los programas de segundo nivel.

```
[root@localhost ~]# cd &
[1] 2423
[root@localhost ~]# cd /
[1]+  Hecho                  cd  (dir ahora: ~)
(dir ahora: /)
[root@localhost /]# ls
bin    dev    home  lib64  mnt    proc  run    srv    tmp    var
boot  etc    lib   media  opt    root  sbin   sys    usr
```

```

[root@localhost etc]# cd ../usr/
[root@localhost usr]# ls
bin  etc  games  include  lib  lib64  libexec  local  sbin  share  src  tmp
[root@localhost usr]# ls -al
total 116
drwxr-xr-x. 13 root root   155 feb 20 07:35 .
dr-xr-xr-x. 17 root root   224 feb 20 07:38 ..
dr-xr-xr-x.  2 root root 24576 feb 22 14:46 bin
drwxr-xr-x.  2 root root    66 nov  5 10:38 etc
drwxr-xr-x.  2 root root    66 nov  5 10:38 games
drwxr-xr-x. 35 root root  4096 feb 22 14:46 include
dr-xr-xr-x. 28 root root  4096 feb 22 14:46 lib
dr-xr-xr-x. 44 root root 24576 feb 22 14:46 lib64
drwxr-xr-x. 19 root root  4096 feb 22 14:46 libexec
drwxr-xr-x. 12 root root   131 feb 20 07:35 local
dr-xr-xr-x.  2 root root 12288 feb 22 14:46 sbin
drwxr-xr-x. 88 root root  4096 feb 22 14:46 share
drwxr-xr-x.  4 root root    34 feb 20 07:35 src
lrwxrwxrwx.  1 root root    10 feb 20 07:35 tmp -> ../var/tmp
[root@localhost usr]# _

```

2. El comando `printenv` permite imprimir en la consola las variables de ambiente del sistema, usando el formato `printenv NOMBREVARIABLE` a lo que devuelve el valor asignado a esa variable. Si no se especifica el nombre de la variable el sistema imprime todas las variables de ambiente presentes mediante el formato `NOMBREVARIABLE=VALOR`

3. la creación de variables de ambiente en linux se hace mediante el comando `export NOMBREVARIABLE=VALOR`. Para almacenar estas variables permanentemente debemos ir a la carpeta `etc/` y almacenarlas en el archivo `environment` de forma que cuando el sistema sea prendido de nuevo la variable siga presentes

```

[operativos@localhost Python-3.4.6]$ printenv VARIABLEOPERATIVOS
[operativos@localhost Python-3.4.6]$ export VARIABLEOPERATIVOS=OPERATIVOS
[operativos@localhost Python-3.4.6]$ print VARIABLEOPERATIVOS
bash: print: no se encontró la orden
[operativos@localhost Python-3.4.6]$ printenv VARIABLEOPERATIVOS
OPERATIVOS
[operativos@localhost Python-3.4.6]$ █

```

4. Para instalar programas en linux mediante el uso de fuentes en necesarios primero descargar el programa a instalar mediante el comando `wget`, una vez descargado se debe descomprimir usando el comando `tar`, una vez dentro de la carpeta correspondiente debemos ejecutar el comando `./configure`

para que configure los archivos fuente y empiece la preparación de la instalación luego se debe ejecutar el comando make, el cual es el encargado de compilar los archivos a los binarios correspondientes y finalmente se ejecuta el comando make install que se encarga de mover los distintos binarios a su lugar correspondiente según lo dicte el sistema operativo.

Para ilustrar lo anterior descargamos python 3.7 y lo instalamos haciendo uso de los comandos anteriormente mencionados.

```
123 cd directorioTallerComandos/
124 wget https://www.python.org/ftp/python/3.4.6/Python-3.4.6.tgz
125 tar -help
126 tar --help
127 tar -xvzf Python-3.4.6.tgz
128 ls
129 cd Python-3.4.6
130 ./configure
131 yum groupinstall "Development Tools"
132 ./configure
133 make
134 make install
135 sudo make install
136 history
[operativos@localhost Python-3.4.6]$
```

```
rm -f /usr/local/bin/idle3
(cd /usr/local/bin; ln -s idle3.4 idle3)
rm -f /usr/local/bin/pydoc3
(cd /usr/local/bin; ln -s pydoc3.4 pydoc3)
rm -f /usr/local/bin/2to3
(cd /usr/local/bin; ln -s 2to3-3.4 2to3)
rm -f /usr/local/bin/pyvenv
(cd /usr/local/bin; ln -s pyvenv-3.4 pyvenv)
if test "x" != "x" ; then \
    rm -f /usr/local/bin/python3-32; \
    (cd /usr/local/bin; ln -s python3.4-32 python3-32) \
fi
rm -f /usr/local/share/man/man1/python3.1
(cd /usr/local/share/man/man1; ln -s python3.4.1 python3.1)
if test "xupgrade" != "xno" ; then \
    case upgrade in \
        upgrade) ensurepip="--upgrade" ;; \
        install|*) ensurepip="" ;; \
    esac; \
    ./python -E -m ensurepip \
        $ensurepip --root=/ ; \
fi
Ignoring ensurepip failure: pip 9.0.1 requires SSL/TLS
[operativos@localhost Python-3.4.6]$
```

Comandos taller:

```
55 cd ~/
56 whoami
57 pwd
58 mkdir -p {directorio1,directorio2,directorio3}
59 vi list1
60 sudo yum install nano -y
61 nano mylist2
62 touch .list3
63 mv .list3 .mylist3
64 mv {list1,mylist2,.mylist3} directorio1/
65 cd directorio1
66 ls
67 ls -a
68 ls list*
69 ls *list
70 ls ??list?
71 ls ~/
72 cp .mylist3 mylist4
73 rm .mylist3
74 rm -rf ../directorio3
75 cd ../directorio2
76 wget http://www.gutenberg.org/cache/epub/19033/pg19033.txt
77 sudo yum install wget -y
78 wget http://www.gutenberg.org/cache/epub/19033/pg19033.txt
79 cat pg19033.txt
80 less pg19033.txt
81 head pg19033.txt
82 tail pg19033.txt
83 grep -in Alice pg19033.txt
84 cat >mylist4
85 cat >> mylist4
86 cat mylist4
87 ls
88 cat list1 mylist2>biglist
89 cd ../directorio1
90 cat list1 mylist2 > biglist
91 sort biglist
92 sort biglist > ordlist
93 who
94 who | sort
95 who | wc -l
96 who | wc -l
97 nano myscript
98 chmod 700 myscript
99 ./myscript
100 history
```