

Cheat Sheet de Bash

Comandos Basicos en Bash

pwd : Imprime el directorio actual

cd /path/-to/dir : cambia el directorio

ls /dir/-to/list : enlista el contenido del directorio

-l : muestra el contenido de una columna

a : muestra el contenido de un directorio

-R : muestra el contenido de un directorio y lo enlista

mv /path/-to/file /path/-whe-re/-to/move : mueve un archivo

cp /path/-to/file /path/-whe-re/-to/copy : copea un archivo

r copia recursos

rm /path/-to/file : remueve un archivo

mkdir /path/-dirName : crea un directorio vacio

rmdir /path/-to/dir : Remueve un directorio

String declarado

Nombre= Jhon

Echo hi Nombre (Ejemplo)

control structure (for)

for var in <expression>; do (Apertura/Inicializacion)

done (Variable y ciclo finalizado)

echo \$var (Variable declarada)

bash redire-ctions

·command > file : redirecciona stdout a un archivo.

·command >> file : redirecciona stdout a un archivo.

·command 2> file : redirecciona stderr a un archivo

·command 2>> file : redirecciona stdout a un archivo.

·command &> file : redirecciona stdout y stderr a un archivo

·command &>> file : redirecciona stdout y stderr a un archivo.

·command < file : redirect stdin to archivo.

· command1 | command 2 : usa la salida y entrada de ambos

Estructura de control de (if)

if <expression>; then (si y solo si)

elif <expression>; then (si condicional)

else (entonces)

fi (cierre)

control structure (while)

for var in <expression>; do (Estructura del while de apertura)

echo \$var (Variable declarada)

done (Ciclo finalizado)

Comandos Basicos Avanzados

⌘history : Consulta el historial

⌘jobs : Enlista los trabajos pendientes

⌘cat file1 file2 ... : llama y copia archivos

⌘more / less file1 file2 .. : agrega o reduce una pagina

⌘tail / head number : Muestra el pie o la cabeza de un texto o funcion

⌘touch file1 file2 ... : Crea funciones y archivos modificables

⌘chmod : cambia los privilegios

⌘echo "text" : muestra el texto selecionado

⌘sort file1 file2 ... : combina fila filas especificas y las mezcla al imprimir

---r : Sortea de forma inverza

---n : Sorteo numerico

---u : Borra las lineas duplicadas

⌘wc file1 file2 ... : imprime a stdout las lineas especificadas

---l : Solo el numero de las lineas

---w : Numeros con palabras.

---w : Numeros con otros caracteres

⌘diff file1 file1 : compara ambos archivos

---i : ignore the character case

---B : ignora las lineas bacias

---w : ignora las lineas escritas

---c : añade citas de contexto

⌘which comman-dName : print the path of a command

⌘pushd / popd /path/-to/dir : cambia de directorio usando stack

⌘dirs : imoreme el directorio seleccionado

⌘find /path/-to/dir -name pattern : Encuentra en cada lugar un archivo especificado

⌘man comman-dName : ayuda al ver el manual sorteandolo por nombre

⌘sudo command : Como shigandos no?

⌘command1 | xargs -i command2 : hace visibles comandos de entradas y salidas.

Funcion Definida

Funtion Funtinname{ (Se declara)

[statements] (Se llena)

[return X] (Y se finaliza

Direcciones en bash

command > file : redirecciona stdout a un file.

·command >> file : redirecciona stdout a un file.

·command 2> file : redirecciona stderr a un file

·command 2>> file : redirecciona stdout a unfile.

·command &> file : redirecicono stdout y stderr a file

·command &>> file : redirije stdout y stderr a file

·command < file : redirije stdin a file.

· command1 | command 2 : usa el comando de salida 1 en el comando de entrada dos

variables

VAR=VARVALUE : crea una variable VAR

\$? : Sales del estatus del ultimo comando

\$? : Sales del estatus del ultimo comando cuando se ejecute

·\$# : el numero de las funciona se ha pasado de la expectadp

·\$0 : Es el nombre de un script

·\$n : el argumento es mostrado

Arrays

---myA-rra-y=(value1 value2 value3): declara un array

---declare -a myArra-y=(value1 value2 value3): declara un array de diferente sintaxis

condi-tional expres-sions

·&& : Operador de suma

·|| : logical

·[[string]] : Regresa si strings esta vacio

·[[-z string]] : regresa una vez si strings esta vacio

·[[string1 == string2]] : regresa a 0 si string esta vacio y redirije a string2

·[[string1 != string2]] : regresa si no son equivalente

·[[string =~ pattern]] : regresa si son similares

·[[-e file]] : return 0 si el file exoste

·[[-d file]] : return 0 si el file es un directorio

·[[-f file]] : return 0 si el file es seguro

·[[-x file]] : return 0 si el file es ejecutable

·[[\$n1 -eq \$n2]] : return 0 si \$n1 == \$n2

·[[\$n1 -lt \$n2]] : return 0 si \$n1 < \$n2

·[[\$n1 -gt \$n2]] : return 0 si \$n1 > \$n2

·[[\$n1 -ge \$n2]] : return 0 si \$n1 >= \$n2

·[[\$n1 -le \$n2]] : return 0 si \$n1 <= \$n2