Day 9/27/2022

Awk

Nano awk.txt

This is line 1

This is line 2

This is line 3

THIS is line four

this is line #5

save & out

awk ‘/This/ {print}’ awk.txt

awk ‘{print $2}’ awk.txt

awk ‘{print $1}’ awk.txt

awk ‘{print $1,$2,$4}’ awk.txt

nano mylogs.txt

user1 102.299.92.233 Google\_Chrome 09-27

user2 951.291.12.151 Safari 09-26

cat mylogs.txt

awk’{print $2}’ awk.txt > ip\_addresses.txt

awk’{print $4,$2}’ awk.txt > ip\_addresses.txt

cat my logs.txt

ls -la | awk ‘{print $3,$5}’ > users\_and\_sizes.txt

cat users\_and\_sizes.txt

ls -la

ls -la | awk '/.bash\_/ {print $9,$3,$6,$7}'

ls -la | awk '/a/ {print $9,$3,$6,$7}'

ls -la | awk '/^a/ {print $9,$3,$6,$7}'

ls -la | awk '/\^a/ {print $9,$3,$6,$7}'

ls -la | awk '/^ a/ {print $9,$3,$6,$7}'

echo “name: nick” | awk -F: ‘{print $2}’

echo “name: nick” | awk -F: “:” ‘{print $2}’

echo “name: $USER” | awk -F: “-“ ‘{print $2}’

echo “name: $USER” | awk -F: “e” ‘{print $1$2}’

echo “name: $USER” | awk -F: “ ‘{print $2}’

**-Bash scripting**

Shell is a program that receives the user’s commands and gives them to

The operating system to process and displays the output.

The BASH Prompt

Username /tmp/repo git(main) $

#!/bin/bash

echo "hello from my first bash script" # this prints the message to the console

echo "Hello from the file" > myoutput.txt # this wirtes the message to the console

# this line will not be printed

Cddaw (not working)

Mkdir mynewdir

ls | grep mynewdir

nano mybashscript.sh

ls -la | grep mybashs

chmod +x mybashscript.sh

ls -la | grep mybashscript.sh

ls -la

mkdir devops

cd devops/

echo “This is my special configuration” > config.txt

cat config.txt

mkdir -/devops

#!/bin/bash

mkdir ~/devops

cd ~/devops

echo "my config" > config.txt

cat config.txt

cd ~

cat devops.sh

./devops.sh

Ls -la | grep devops.sh

Chmod +x devops.sh

Ls -la | grep devops.sh

Cat devops.sh

Nano chomd-test.sh

Cat chomd-test.sh

./chomd-test.sh

Nano

Ls | grep conf

Echo “echo $USER” > shortcut.sh && chomd +x shortcut.sh && ./shortcut.sh

Cat shortcut.sh

Echo ec2-user

Nano devops.sh

Cat devops.sh

./devops

Ls

#!/bin/bash

set -e # this will exit the file if any error occcur

mkdir ~/devops

cd ~/devops

echo "my config" > config.txt

cat config.txt

cd ~

#!/bin/bash

set -e # this will exit the file if any error occcur

mkdir ~/devops && cd ~/devops

echo "my config" > config.txt

cat config.txt

cd ~

./devops.sh

Cat devops.sh

Pwd

Ls | grep config

Cat ./devops.sh

Cat devops.sh

Ls -la

Nano devops.sj

Nano devops.sh

Cat devops.sh

./devops.sh

#!/bin/bash

# set -e # this will exit the file if any error occcur

mkdir ~/devops # && cd ~/devops

echo "my config" > config.txt

cat config.txt

rm -rf \*.txt

cd ~

myvar=”this is cool”

echo $myvar

ls

my\_ls=$(1s)

echo $my\_ls

pwd

lucky\_number=7

read -p “Enter the value: “ lucky\_number

echo $lucky\_number

x=5

echo $x

x=$(whoami)

echo $x

nano myvars.sh

#!/bin/bash

read -p "Enter new lucky number: " my\_var

current\_user=$(whoami)

echo "$current\_user lucky number is $my\_var"

chmod +x myvars.sh

./myvars.sh

Cat myvars.sh

“ my\_var

Nano myvars.sh

./myvars.sh

Nano myvars.sh

#!/bin/bash

user1=$1

user2=$2

user3=$3

echo "Sorry $4, we only accept 3 users"

read -p "Enter new lucky number for $user1: " user1\_number

read -p "Enter new lucky number for $user2: " user2\_number

read -p "Enter new lucky number for $user3: " user3\_number

echo "$user1 lucky number is $user1\_number"

echo "$user2 lucky number is $user2\_number"

echo "$user3 lucky number is $user3\_number