**Atmanand Saraswati science college**

**Unix and Shell Programming**

**Name : Lathiya Disha Hiteshbhai**

**Division : 2**

**Roll Number : BCA22125**

**INDEX**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Sr. No.** | **Definition** | **Date** | **Page No.** | **Sign** |
| **COMMANDS** | | | | |
| 1 | Various single line commands | 14/10/2024 | 1 |  |
| **SHELL SCRIPTS** | | | | |
| 1 | Write a shell script to do addition of odd positioned numbers and even positioned numbers. | 14/10/2024 | 3 |  |
| 2 | Write a shell script to reverse the string. | 14/10/2024 | 4 |  |
| 3 | Write a shell script to count number of lines, words and characters of a specific file. | 15/10/2024 | 5 |  |
| 4 | Write a shell script to check whether two files are same or not. If the files are same the second file should be deleted from the directory.(File names should be passed as command line arguments and perform proper validations.) | 15/10/2024 | 6 |  |
| 5 | Write a shell script to count number of digits, vowels and cosonants. | 15/10/2024 | 7 |  |
| 6 | Write a shell script to check whether the number is palindrome or not. | 15/10/2024 | 8 |  |
| 7 | Write a shell script to display name and size of the files on the given path. | 16/10/2024 | 9 |  |
| 8 | Write a menu driven shell script to create and delete a file which will accept two command line arguments(file name and create / delete option). | 16/10/2024 | 10 |  |
| 9 | Write a shell script to accept a string from the user and convert all the lower characters into upper case and upper characters to lower case. | 16/10/2024 | 11 |  |
| 10 | Write a script to display all the words of a line in reverse order of the given file. | 16/10/2024 | 12 |  |

1. **Write awk command to count number of occurrences of pattern ‘and’ in file eg\_grep.txt.**

awk '{while (match($0, /Hello/)) {count++; $0=substr($0, RSTART+RLENGTH)}} END {print count}' eg\_grep.txt

1. **Write a command using grep or sed or awk to display line 5 to 15, 25 to 35 and last line of file employee.txt.**

awk 'NR==1, NR==2 {print NR,$0} NR==4, NR==6 {print NR,$0} END {print}' employee.txt

1. **Write a command to count number of characters in first five lines of file.**

awk 'NR == 1, NR == 5 { print length }' employee.txt

1. **List file names consisting of only digits.**

ls | grep -E '^[0-9]+\.'

1. **Remove all alphabets from the file.**

sed '/[a-z]/d' eg\_grep.txt

1. **Display all the lines that are 10 characters long and ends with $.**

awk 'length($0) > 20' employee.txt

1. **Display all the lines that have five words in it.**

awk 'NF == 5 {print}' eg\_grep.txt

1. **Replace all the occurrence of the word “command” with “cmd”.**

sed 's/command/cmd/g' eg\_grep.txt

1. **Display all the blank lines with line numbers of the given file.**

sed -n '/^$/=' eg\_grep.txt

1. **Display all the lines that have even number of fields.**

awk 'NF % 2 == 0 {print}' eg\_grep.txt

1. **To print the line number as well as name of the employee whose salary is more than 20,000.**

awk '{ if($3 > 20000) print $0;}' emp.txt

1. **Display all the lines that starts with A and ends with $.**

grep -E '^t.\*.$' eg\_grep.txt

1. **Display 1st word of 10th to 20th line of the file.**

awk 'NR==4, NR==6 {print NR "- " $1 }' eg\_grep.txt

1. **Remove extra spaces all the lines of the file.**

sed 's/[ \t]\+$//' eg\_grep.txt

1. **Display all the lines that occur more than once in the given file.**

sort sort1.txt | uniq –cd

1. **Display the largest sized file of the current directory.**

ls -l | sort -nr | head -2

1. **Write a shell script to do addition of odd positioned numbers and even positioned numbers.**

clear

echo "enter a number"

read num

even=0

odd=0

tmp=$num

#to count length of a number

while [ $tmp -gt 0 ]

do

tmp=$(( $tmp / 10 ))

cnt=$(( $cnt + 1))

done

echo "Length of the number is : $cnt"

#to do sum of odd and even number

while [ $cnt -gt 0 ]

do

if [ `expr $cnt % 2` -eq 0 ]

then

ev=`expr $num % 10`

even=`expr $even + $ev`

num=`expr $num / 10`

else

od=`expr $num % 10`

odd=`expr $odd + $od`

num=`expr $num / 10`

fi

cnt=`expr $cnt \- 1`

done

echo the sum of even number $even

echo the sum of odd number $odd

**Output:**

enter a number

12345

Length of the number is : 5

the sum of even number 6

the sum of odd number 9

1. **Write a shell script to reverse the string.**

clear

echo "Enter a string :"

read string

#count the length of the string

len=${#string}

echo "Length of the string is : $len"

for ((i = $len - 1; i >= 0; i--))

do

#"${string:$i:1}"extract single character from string.

rev="$rev${string:$i:1}"

#echo "Reverse of the string :$i"

#echo "Reverse of the string :$rev"

done

echo "Reverse of the string :$rev"

**Output:**

Enter a string :

abc xyz

Length of the string is : 7

Reverse of the string :zyx cba

1. **Write a shell script to count number of lines, words and characters of a specific file.**

clear

file\_path="/home/hp/eg\_grep.txt"

# using wc command to count number of lines

number\_of\_lines=`wc --lines < $file\_path`

# using wc command to count number of words

number\_of\_words=`wc --word < $file\_path`

# Displaying number of lines and number of words

echo "Number of lines: $number\_of\_lines"

echo "Number of words: $number\_of\_words"

**Output:**

Number of lines: 10

Number of words: 30

1. **Write a shell script to check whether two files are same or not. If the files are same the second file should be deleted from the directory.(File names should be passed as command line arguments and perform proper validations.)**

clear

if cmp $1 $2

then

rm $2

echo "$2 is deleted"

else

echo "The files are different"

fi

**Output (if files are not identical):**

sh /home/hp/journal/pract\_3.sh file1.txt file3.txt

file1.txt file3.txt differ: char 3, line 1

The files are different

**Output (if files are not identical):**

sh /home/hp/journal/pract\_3.sh file1.txt file2.txt

The files are same

file2.txt is deleted

1. **Write a shell script to count number of digits, vowels and cosonants.**

clear

echo -n "Enter a line of text: "

read string

numCount=$(echo $string | grep -o "[0-9]" | wc --lines)

vowCount=$(echo $string | grep -o -i "[aeiou]" | wc --lines)

consCount=$(echo $string | grep -o -i "[bcdfghjklmnpqrstvwxyz]" | wc --lines)

echo "The given string has $vowCount vowels, $consCount consonants and $numCount numbers in it."

**Output:**

Enter a line of text: vowels numbers consonants

The given string has 7 vowels, 16 consonants and 0 numbers in it.

1. **Write a shell script to check whether the number is palindrome or not.**

clear

echo -n "Enter a number: "

read num

# store the original number

original\_num=$num

# reverse the number

rev=0

while [ $num -gt 0 ]

do

# get the remainder of the number

remainder=$(($num % 10))

# multiply reverse by 10 then add the remainder

rev=$((($rev \* 10) + $remainder))

# divide the number by 10

num=$(($num / 10))

done

# check if the number is a palindrome

if [ $original\_num -eq $rev ];

then

echo "$original\_num is a palindrome number."

else

echo "$original\_num is not a palindrome number."

fi

**Output 1:**

Enter a number: 123

123 is not a palindrome number.

**Output 2:**

Enter a number: 121

121 is a palindrome number.

1. **Write a shell script to display name and size of the files on the given path.**

clear

echo "Enter the full path to the file :"

read file

filesize=$(ls -lh $file | awk '{print $5}')

echo "$file has a size of $filesize"

**Output :**

Enter the full path to the file :

/home/hp/eg\_grep.txt

/home/hp/eg\_grep.txt has a size of 178

1. **Write a menu driven shell script to create and delete a file which will accept two command line arguments(file name and create / delete option).**

case $1 in

"--create")

echo "Creating new file $2"

#echo

touch $2

;;

"--delete")

echo "Deleting file $2"

echo

rm $2

;;

\*)

echo "Not a valid argument"

echo

;;

esac

**Output :**

sh /home/hp/journal/pract\_3.sh --create f1.txt

Creating new file f1.txt

1. **Write a shell script to accept a string from the user and convert all the lower characters into upper case and upper characters to lower case.**

clear

string1="CONVERT lower to upper case"

echo "String 1 is : $string1"

echo $string1 | sed 's/[a-z]/\U&/g'

string2="convert UPPER to LOWER case"

echo "String 2 is : $string2"

echo $string2 | sed 's/[A-Z]/\L&/g'

**Output :**

String 1 is : CONVERT lower to upper case

CONVERT LOWER TO UPPER CASE

String 2 is : convert UPPER to LOWER case

convert upper to lower case

1. **Write a script to display all the words of a line in reverse order of the given file.**

clear

text=$(cat $1)

arr=($text)

al=${#arr[@]}

let al="al-1"

while (($al >= 0))

do

echo -n "${arr[al]}"

echo -n " "

let al="al - 1"

done

**Output :**

command useful is this this about $ with ending computer computer computer 56789 12345 command command Easy and is it command useful very is this grep using searchingn pattern