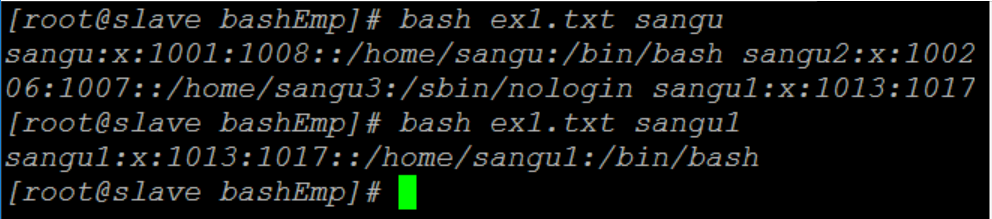
**1) Command line argument to print user details**

#!/bin/bash

RESULT=`cat /etc/passwd | grep $1`

echo $RESULT

**Output:**



**2) Functions, local and global variables.**

#!/bin/bash

var\_change () {

local var1='local 1'

echo Inside function: var1 is $var1 : var2 is $var2

var1='changed again'

var2='2 changed again'

}

var1='global 1'

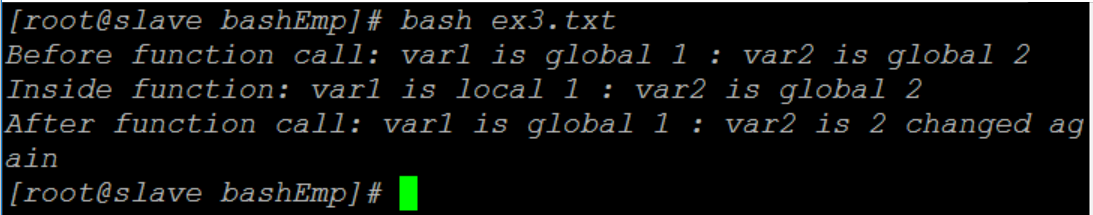
var2='global 2'

echo Before function call: var1 is $var1 : var2 is $var2

var\_change

echo After function call: var1 is $var1 : var2 is $var2

**Output:**



**3) Line counting across a set of files**

**Explicit For loop**

#!/bin/bash

# Counting the number of lines in a list of files

# for loop over arguments

if [ $# -lt 1 ]

then

echo "Usage: $0 file ..."

exit 1

fi

echo "$0 counts the lines of code"

l=0

n=0

s=0

for f in $\*

do

l=`wc -l $f | sed 's/^\([0-9]\*\).\*$/\1/'`

echo "$f: $l"

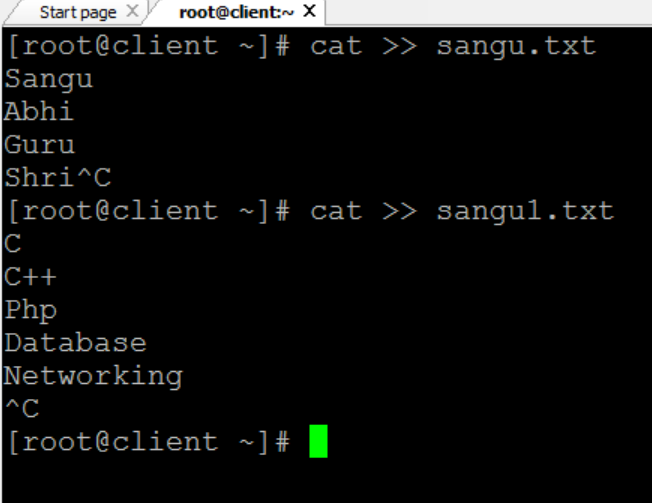
n=$[ $n + 1 ]

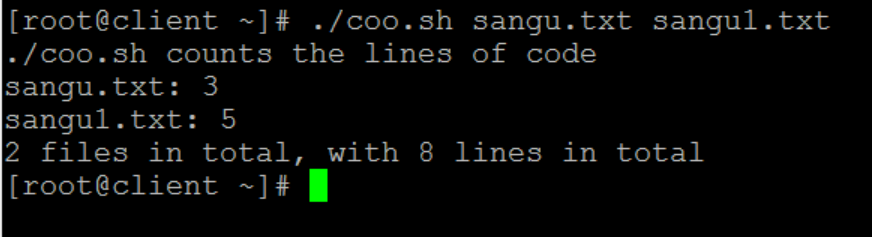
s=$[ $s + $l ]

done

echo "$n files in total, with $s lines in total"

**Output:**





**4) Read file into bash array**

#!/bin/bash

# Declare array

declare -a ARRAY

# Link filedescriptor 10 with stdin

exec 10<&0

# stdin replaced with a file supplied as a first argument

exec < $1

let count=0

while read LINE; do

ARRAY[$count]=$LINE

((count++))

done

echo Number of elements: ${#ARRAY[@]}

# echo array's content

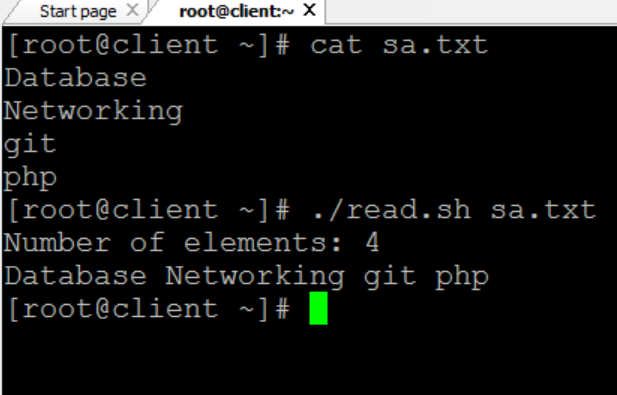
echo ${ARRAY[@]}

# restore stdin from filedescriptor 10

# and close filedescriptor 10

exec 0<&10 10<&-

**Output:**



**5) Test for files and directories**

#!/bin/bash

if test -s $1

then

echo "$1 not empty file"

fi

if test -f $1

then

echo "$1 normal file. Not a directory"

fi

if test -e $1

then

echo "$1 exists"

fi

if test -d $1

then

echo "$1 is directory and not a file"

fi

if test -r $1

then

echo "$1 is read-only file"

fi

if test -x $1

then

echo "$1 is executable"

fi

**Output:**

