Linux continuation

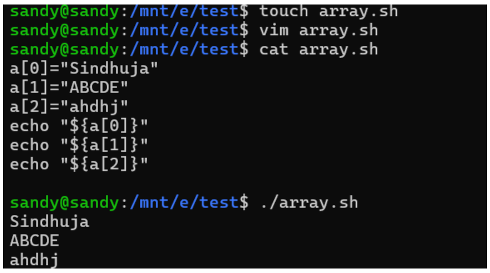
19 February 2025 15:30

Checking of conditional and looping statements

Arrays:

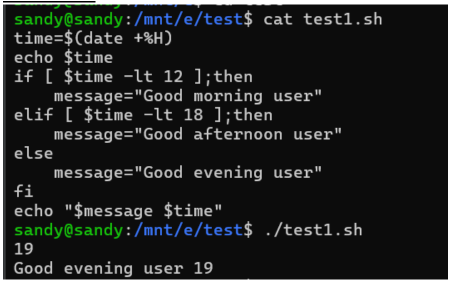
Working on the data structure with arrays and retrieving the data from the file.

1. vi array.sh - Creates the file array.sh in the working directory.
2. chmod 777 array.sh - Allows all the permissions.
3. /array.sh - Executes the file.



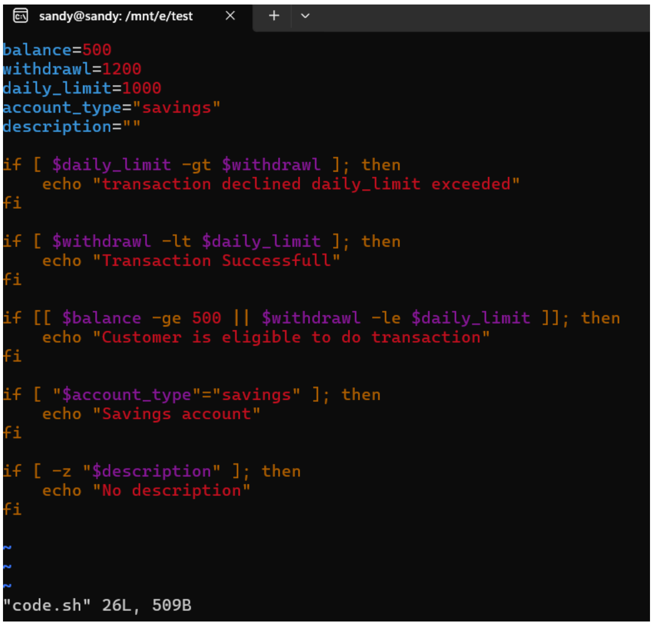
```
sandy@sandy:/mnt/e/test$ touch array.sh
sandy@sandy:/mnt/e/test$ vim array.sh
sandy@sandy:/mnt/e/test$ cat array.sh
a[0]="Sindhuja"
a[1]="ABCDE"
a[2]="ahdhj"
echo "${a[0]}"
echo "${a[1]}"
echo "${a[2]}"

sandy@sandy:/mnt/e/test$ ./array.sh
Sindhuja
ABCDE
ahdhj
```

If elif else:

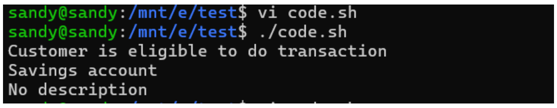


```
sandy@sandy:/mnt/e/test$ cat test1.sh
time=$(date +%H)
echo $time
if [ $time -lt 12 ];then
    message="Good morning user"
elif [ $time -lt 18 ];then
    message="Good afternoon user"
else
    message="Good evening user"
fi
echo "$message $time"
sandy@sandy:/mnt/e/test$ ./test1.sh
19
Good evening user 19
```

More



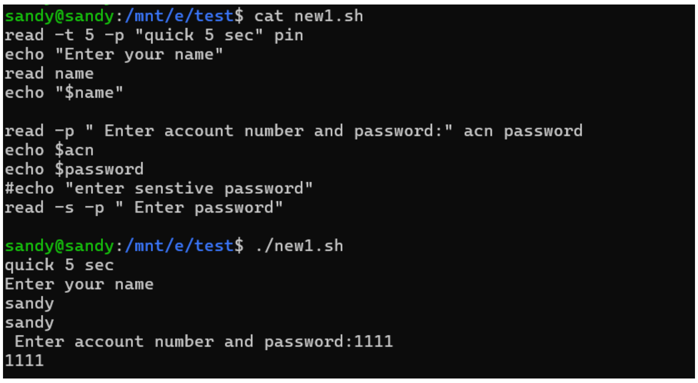
```
balance=500
withdrawl=1200
daily_limit=1000
account_type="savings"
description=""

if [ $daily_limit -gt $withdrawl ]; then
    echo "transaction declined daily_limit exceeded"
fi

if [ $withdrawl -lt $daily_limit ]; then
    echo "Transaction Successfull"
fi

if [[ $balance -ge 500 || $withdrawl -le $daily_limit ]]; then
    echo "Customer is eligible to do transaction"
fi

if [ "$account_type"="savings" ]; then
    echo "Savings account"
fi

if [ -z "$description" ]; then
    echo "No description"
fi

~
~
~
"code.sh" 26L, 509B
```



```
sandy@sandy:/mnt/e/test$ vi code.sh
sandy@sandy:/mnt/e/test$ ./code.sh
Customer is eligible to do transaction
Savings account
No description
```

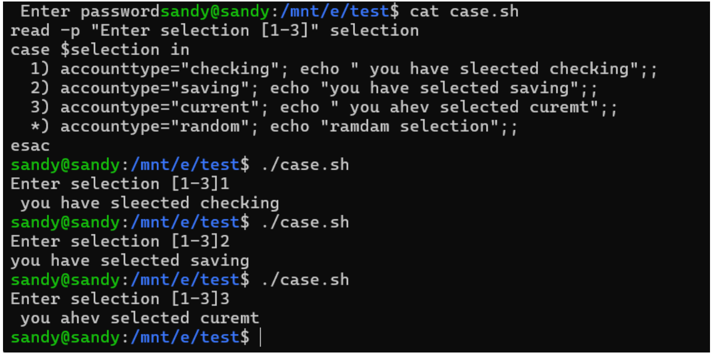
More:

Taking user input with a timeout of 5 seconds



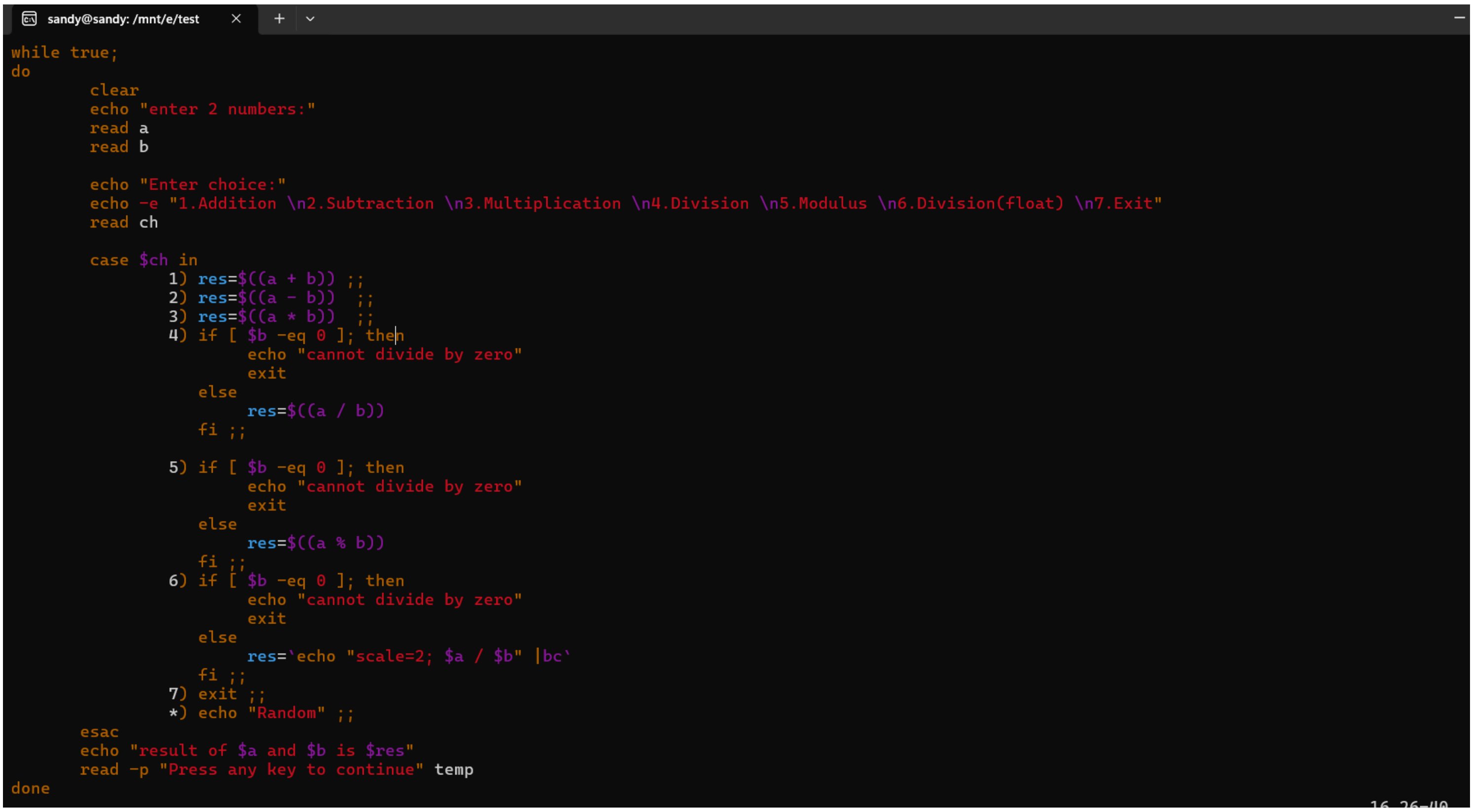
```
sandy@sandy:/mnt/e/test$ cat new1.sh
read -t 5 -p "quick 5 sec" pin
echo "Enter your name"
read name
echo "$name"

read -p " Enter account number and password:" acn password
echo $acn
echo $password
#echo "enter senstive password"
read -s -p " Enter password"

sandy@sandy:/mnt/e/test$ ./new1.sh
quick 5 sec
Enter your name
sandy
sandy
 Enter account number and password:1111
1111
```

More: case keyword

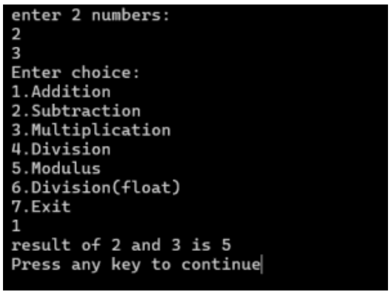


```
 Enter passwordsandy@sandy:/mnt/e/test$ cat case.sh
read -p "Enter selection [1-3]" selection
case $selection in
  1) accounttype="checking"; echo " you have sleected checking";;
  2) accounttype="saving"; echo "you have selected saving";;
  3) accounttype="current"; echo " you ahev selected curemt";;
  *) accounttype="random"; echo "ramdam selection";;
esac
sandy@sandy:/mnt/e/test$ ./case.sh
Enter selection [1-3]1
 you have sleected checking
sandy@sandy:/mnt/e/test$ ./case.sh
Enter selection [1-3]2
you have selected saving
sandy@sandy:/mnt/e/test$ ./case.sh
Enter selection [1-3]3
 you ahev selected curemt
sandy@sandy:/mnt/e/test$ |
```

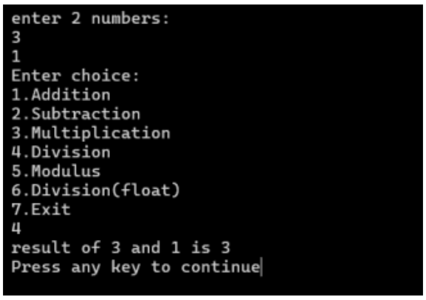
Afternoon activity : Building simple calculator



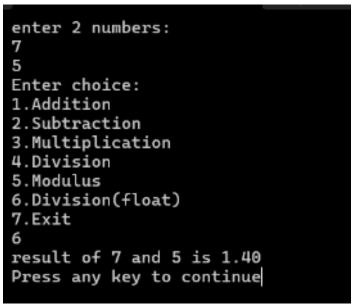
```
while true;
do
        clear
        echo "enter 2 numbers:"
        read a
        read b

        echo "Enter choice:"
        echo -e "1.Addition \n2.Subtraction \n3.Multiplication \n4.Division \n5.Modulus \n6.Division(float) \n7.Exit"
        read ch

        case $ch in
                1) res=$((a + b)) ;;
                2) res=$((a - b))  ;;
                3) res=$((a * b))  ;;
                4) if [ $b -eq 0 ]; then
                        echo "cannot divide by zero"
                        exit
                   else
                        res=$((a / b))
                   fi ;;

                5) if [ $b -eq 0 ]; then
                        echo "cannot divide by zero"
                        exit
                   else
                        res=$((a % b))
                   fi ;;
                6) if [ $b -eq 0 ]; then
                        echo "cannot divide by zero"
                        exit
                   else
                        res=`echo "scale=2; $a / $b" |bc`
                   fi ;;
                7) exit ;;
                *) echo "Random" ;;
       esac
       echo "result of $a and $b is $res"
       read -p "Press any key to continue" temp
done
```



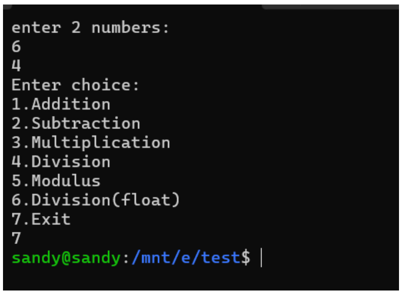
```
enter 2 numbers:
2
3
Enter choice:
1.Addition
2.Subtraction
3.Multiplication
4.Division
5.Modulus
6.Division(float)
7.Exit
1
result of 2 and 3 is 5
Press any key to continue
```



```
enter 2 numbers:
3
1
Enter choice:
1.Addition
2.Subtraction
3.Multiplication
4.Division
5.Modulus
6.Division(float)
7.Exit
4
result of 3 and 1 is 3
Press any key to continue
```



```
enter 2 numbers:
7
5
Enter choice:
1.Addition
2.Subtraction
3.Multiplication
4.Division
5.Modulus
6.Division(float)
7.Exit
6
result of 7 and 5 is 1.40
Press any key to continue
```



```
enter 2 numbers:
6
4
Enter choice:
1.Addition
2.Subtraction
3.Multiplication
4.Division
5.Modulus
6.Division(float)
7.Exit
7
sandy@sandy:/mnt/e/test$
```