# Project 1

# Done by:

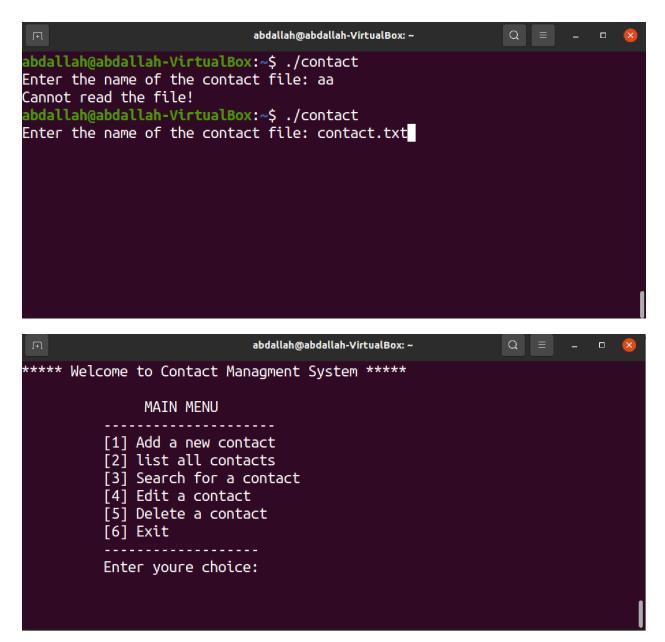
Abdallah Daoud Mohammad 1190515

Mahmood Ahmad Atia 1192519

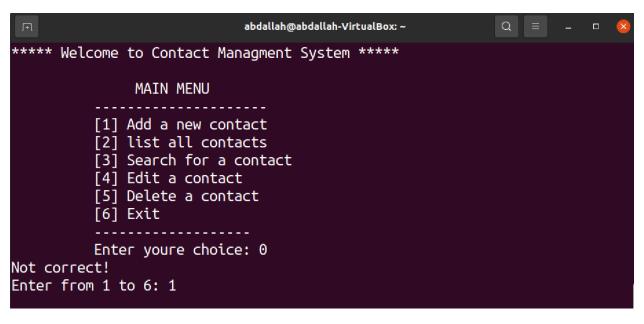
### First we have to enter the name file of the data

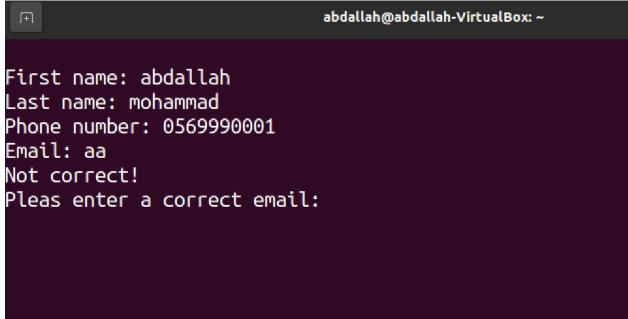
### If the file is not exist error message occur

Data stored in contact.txt



If you enter an illegal input you will get a message to re-enter in all stages



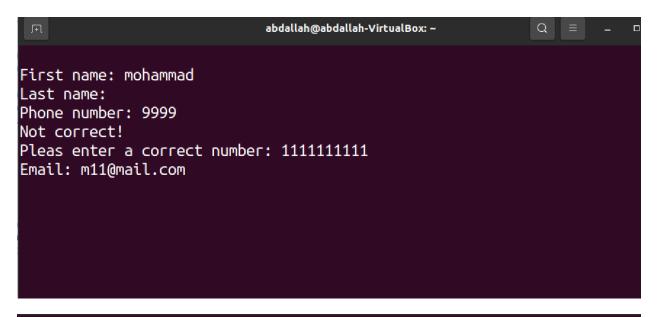


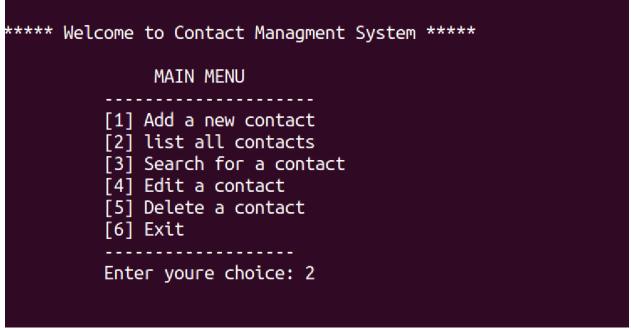
1) First name required be entered whatever the input

There no conditions for last name

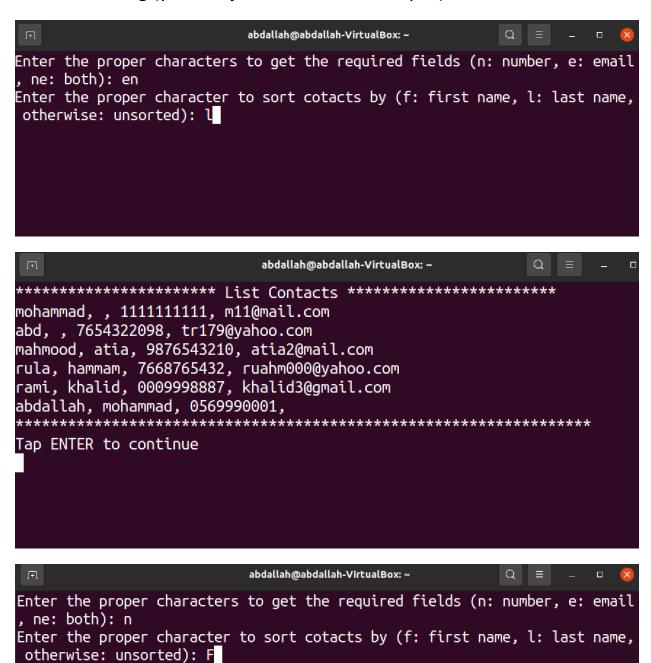
The number should be 10 digits and it's required

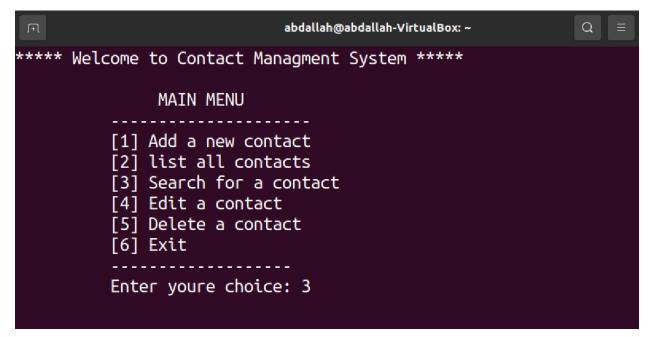
Email have to contains a '@' symbol and it's not required





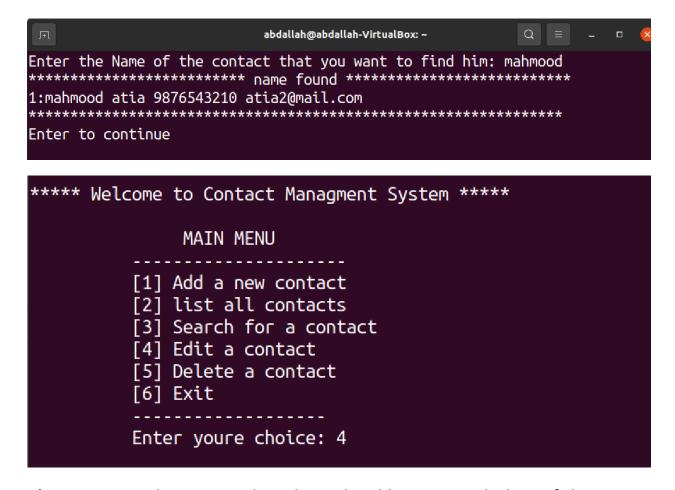
2) If the input contains 'n' the number will be displayed, 'e' email will displayed. If the second input equal 'f' the contacts will be stored by first name, 'l' by last name, otherwise the contacts will be displayed without sorting (you can just enter without input)



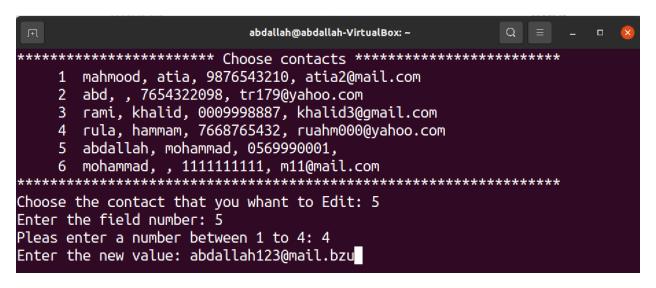


3) Enter the name you want, to get the contacts that has the same name. After that you can see the output until you tap "enter"



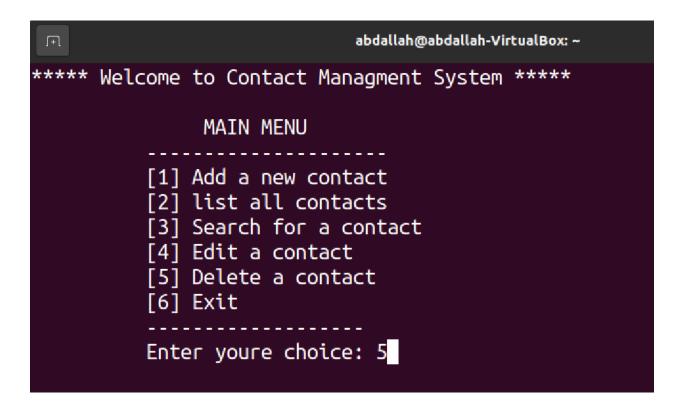


4) First, enter the contact listed number like you see below, if the input is out of the range, enter again. Second step, enter the field number (first: 1, last: 2, number: 3, email: 4) otherwise enter again.

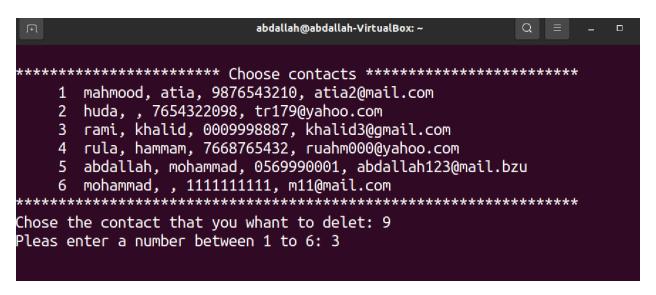


Last step, enter the new value in correct format or you will be have to enter the value again

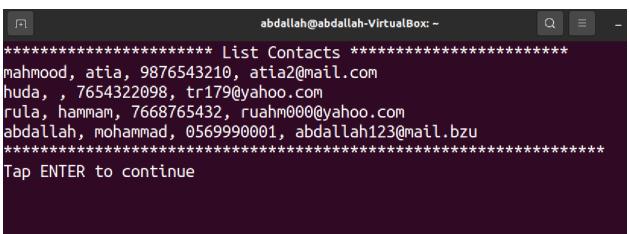
Using second choice (list all contacts) to see the changes



#### 5) Just enter the number of the line to delete the contact







#### 6) Just Exit ^\_^

```
***** Welcome to Contact Managment System ****

MAIN MENU

[1] Add a new contact

[2] list all contacts

[3] Search for a contact

[4] Edit a contact

[5] Delete a contact

[6] Exit

Enter youre choice: 6

abdallah@abdallah-VirtualBox:~$
```

## code

```
#!/bin/bash
echo -n "Enter the name of the contact file: "
read fileName
if [!-r "$fileName"]
then
echo "Cannot read the file!"
exit 1
fi
#go to line 260 "the last of the code" if you want to get a top view
menu() {
clear -x #to clear every thing befor display the menu
echo "**** Welcome to Contact Managment System *****"
 printf "\n%15sMAIN MENU\n" " "
printf "%10c-----\n" " "
printf "%10c[1] Add a new contact\n" " "
printf "%10c[2] list all contacts\n" " "
printf "%10c[3] Search for a contact\n" " "
 printf "%10c[4] Edit a contact\n" " "
printf "%10c[5] Delete a contact\n" " "
 printf "%10c[6] Exit\n" " "
printf "%10c-----\n" " "
printf "%10cEnter youre choice: " " "
```

```
read choice
}
addContact() {
clear -x
echo -n "First name: "
read firstName
#To make sure that first name was entered
until [ "$firstName" ]
do
  echo -n "Pleas enter the first name: "
  read firstName
 done
echo -n "Last name: "
read lastName
echo -n "Phone number: "
read phoneNumber
#To make sure that phone number is correct
while [ ! (echo "phoneNumber" | grep '^[0-9]{10}) > /dev/null ]
do
  echo Not correct!
  echo -n "Pleas enter a correct number: "
  read phoneNumber
```

```
done
```

```
echo -n "Email: "
 read email
 #To make sure that email is correct or there no email
 while [ "$email" ]
 do
 if [ $(echo "$email" | grep '@') > /dev/null ]
  then
   break
  fi
  echo Not correct!
  echo -n "Pleas enter a correct email: "
  read email
 done
 echo "$firstName, $lastName, $phoneNumber, $email" >> $fileName
}
listContact() {
 clear -x
 printf "Enter the proper characters to get the required fields (n: number, e: email, ne: both): "
 field="
 read op
 #cheek what is the option entered then add the proper field in 'field' var
```

```
if [ $(echo "$op" | grep '[Nn]') > /dev/null ]
then
field=',3'
fi
if [ $(echo "$op" | grep '[Ee]') > /dev/null ]
then
field="$field,4"
fi
printf "Enter the proper character to sort cotacts by (f: first name, I: last name, otherwise: unsorted): "
name="
read name
clear -x
if [! "$name"]
then
 cat contact.txt
elif [ $name == "f" -o $name == "F" ]
then
 cut -d',' -f1,2"$field" $fileName | sort -k 1 #sort by first column (first name)
elif [ $name == "I" -o $name == "L" ]
then
 cut -d',' -f1,2"$field" $fileName | sort -k 2 #sort by second column (last name)
else
 cut -d',' -f1,2 $fileName
fi
```

```
echo "Tap ENTER to continue" #this is to let you see the result befor go to the menu
read g #g --> garbage
}
search(){
clear -x
echo -n "Enter the Name of the contact that you want to find him: "
read name
if ! grep -Fxq "$name" contact.txt;
then
 else
 fi
cat contact.txt | tr -d ',' | grep -n "$name"
#display the line with wanted name without ','
#where some time we want to find with full name and ',' make a problem
#so ',' removed by tr -d ','
echo "Enter to continue"
read g
}
edit() {
```

```
clear -x
cat -n $fileName
echo -n "Choose the contact that you whant to Edit: "
read num
#chaek if the number in the range or not
rang=$(wc -l contact.txt | cut -d' ' -f1)
until []
do
if [ "$(echo "$num" | grep "^ *[0-9][0-9]*$")" ]
then
 if [ "$num" -le $rang -a "$num" -ge 1 ]
 then
  break
 fi
fi
echo -n "Pleas enter a number between 1 to $rang: "
read num
done
#chaek if the field in the range or not
echo -n "Enter the field number: "
read field
until []
do
```

```
if [ "$(echo "$num" | grep "^ *[0-9][0-9]*$")" ]
 then
  if [ "$field" -le 4 -a "$field" -ge 1 ]
  then
   break
  fi
 fi
 echo -n "Pleas enter a number between 1 to 4: "
 read field
done
echo -n "Enter the new value: "
read new
touch temp.txt
i=1
line=$(sed -n "1p" $fileName) #in the beginning, get the first line
while [ "$line" ]
do
 #if this line is not the line that we want to edit. save it in temp.txt directly
 if [ $i -ne $num ]
 then
  echo "$line" >> temp.txt
 else
  j=1
  temp=""
```

```
while [$j -le 4]
do
 #if this field is not the field that we want to edit. concat it with temp directly
 if [$j -ne $field]
 then
  temp="$temp$(echo -n "$line" | cut -d',' "-f$j")"
 else # $field==1 then is first name, 2 --> last name, 3 --> number, 4 --> email
  if [$j -eq 1]
  #if the required field equals 1, then cheek the new first name
  #and make temp="$new" because it's the first field
  then
   until [ "$new" ]
   do
    echo -n "Pleas enter the first name: "
    read new
   done
   temp="$new"
  else
  #if the required field 2, no need to cheek but when 3,4 you need to cheek
  #after that concat 'temp' with 'new'
   case "$field"
   in
    3) while [ ! \pm (echo "\pm new" | grep '^[0-9] \{10\}) > /dev/null ]
      do
       echo -n "Pleas enter a correct number: "
```

```
read new
      done;;
    4) while [ "$email" ]
      do
       if [ $(echo "$email" | grep '@') > /dev/null ]
       then
        break
       fi
       echo Not correct!
       echo -n "Pleas enter a correct email: "
       read email
      done
   esac
   temp="$temp $new"
  fi
 if [ $j -ne 4 ]
 #after finishing adding the field whatever it's required field or not
 #if this not the last field, concat temp with ','
 then
  temp="$temp,"
j=$((j+1))
done
echo "$temp" >> temp.txt #write the modified contact to temp.txt
```

fi

fi

```
fi
 i=$((i+1))
 line=$(sed -n "$i p" $fileName) #read the next line
done
cat temp.txt > $fileName #write the modified contacts over the contacts file
rm temp.txt
}
delete() {
clear -x
cat -n $fileName
echo -n "Chose the contact that you whant to delet: "
read num
#chaek if the number is in the range or not
rang=$(wc -l contact.txt | cut -d' ' -f1)
until []
do
 if [ "$(echo "$num" | grep "^ *[0-9][0-9]*$")" ]
 then
  if [ "$num" -le $rang -a "$num" -ge 1 ]
  then
   break
  fi
```

```
echo -n "Pleas enter a number between 1 to $rang: "
 read num
 done
touch temp.txt
i=1
line=$(sed -n "1p" $fileName)
while [ "$line" ]
do
 if [ $i -ne $num ]
 #if the contact(line) we read is not the contact that we want to delete then add it to temp file
  then
  echo "$line" >> temp.txt
 fi
 i=$((i+1))
 line=$(sed -n "$i p" $fileName)
done
cat temp.txt > $fileName
rm temp.txt
}
#here we use the functions above
choice=
until [ "$choice" = 6 ]
```

fi

```
do
 menu
until [\$(echo "\$choice" | grep "^[1-6]\$") > /dev/null]
 do
  echo "Not correct!"
 echo -n "Enter from 1 to 6: "
  read choice
 done
case "$choice"
 in
 1) addContact;;
2) listContact;;
3) search;;
 4) edit;;
5) delete;;
 esac
```

done