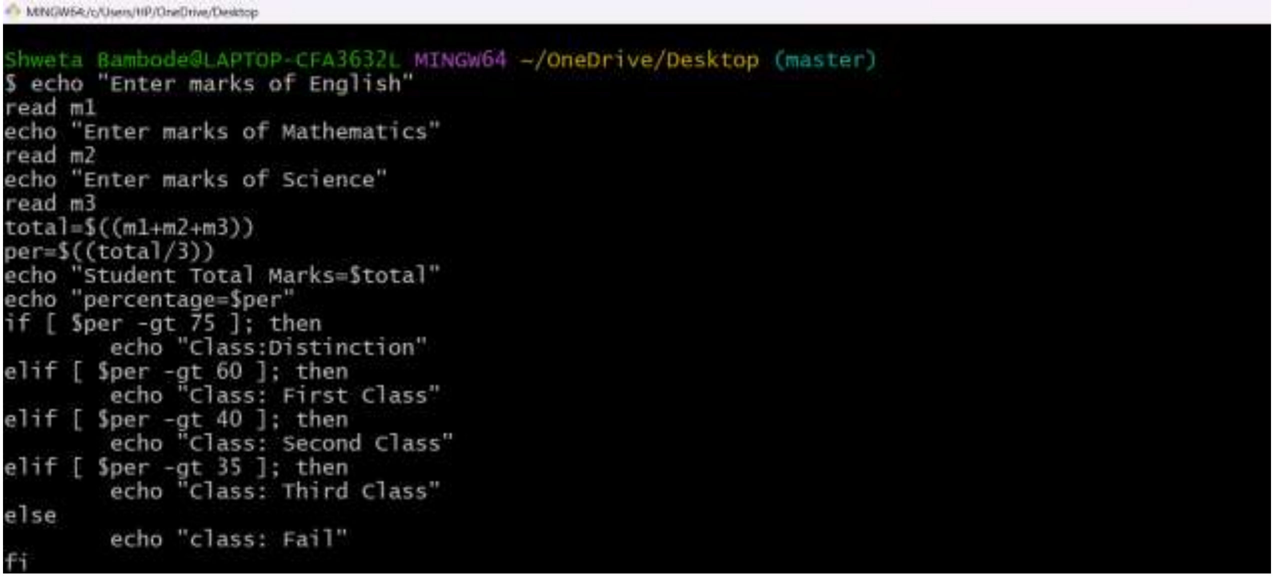
a] Write a shell script to generate mark-sheet of a student. Take 3 subjects, calculate and display total marks, percentage and Class obtained by the student.

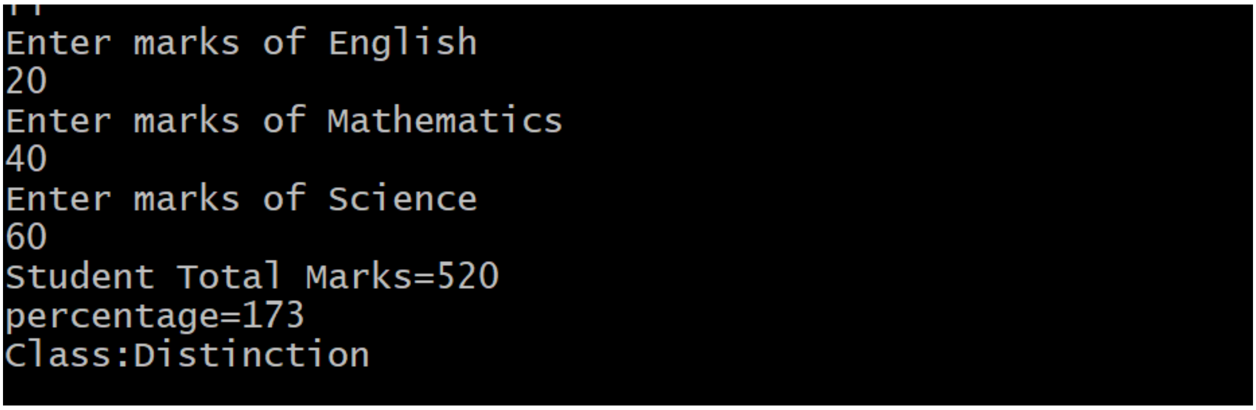
Code:



MINGW64:/c/Users/HP/OneDrive/Desktop

```
Shweta Bambode@LAPTOP-CFA3632L MINGW64 ~/OneDrive/Desktop (master)
$ echo "Enter marks of English"
read m1
echo "Enter marks of Mathematics"
read m2
echo "Enter marks of Science"
read m3
total=$((m1+m2+m3))
per=$((total/3))
echo "Student Total Marks=$total"
echo "percentage=$per"
if [ $per -gt 75 ]; then
        echo "Class:Distinction"
elif [ $per -gt 60 ]; then
        echo "Class: First Class"
elif [ $per -gt 40 ]; then
        echo "Class: Second Class"
elif [ $per -gt 35 ]; then
        echo "Class: Third Class"
else
        echo "class: Fail"
fi
```

OUTPUT:

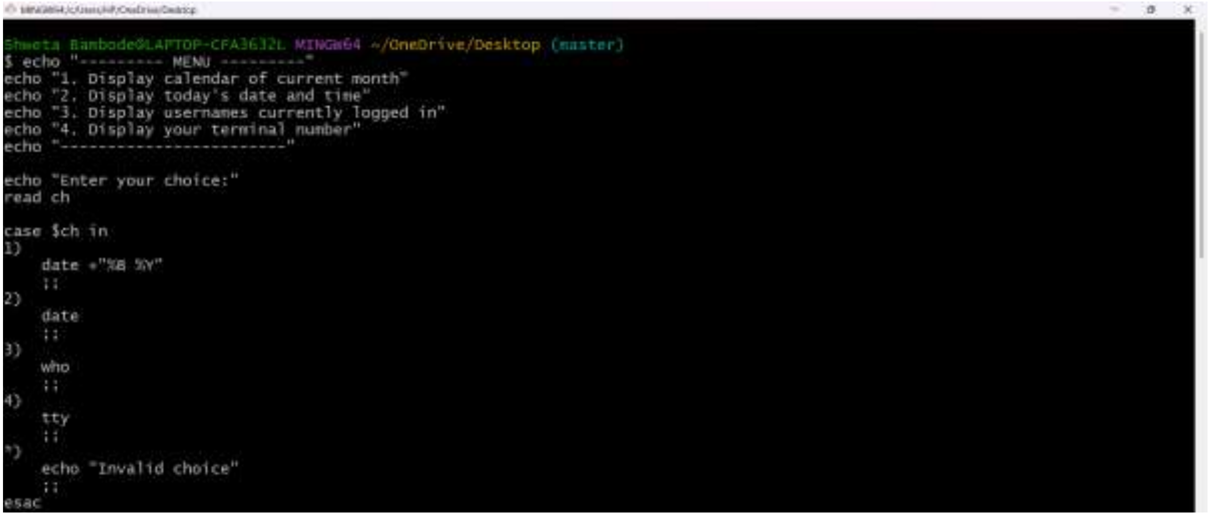


```
Enter marks of English
20
Enter marks of Mathematics
40
Enter marks of Science
60
Student Total Marks=520
percentage=173
Class:Distinction
```

b] Write a menu driven shell script which will print the following menu and execute the given task.

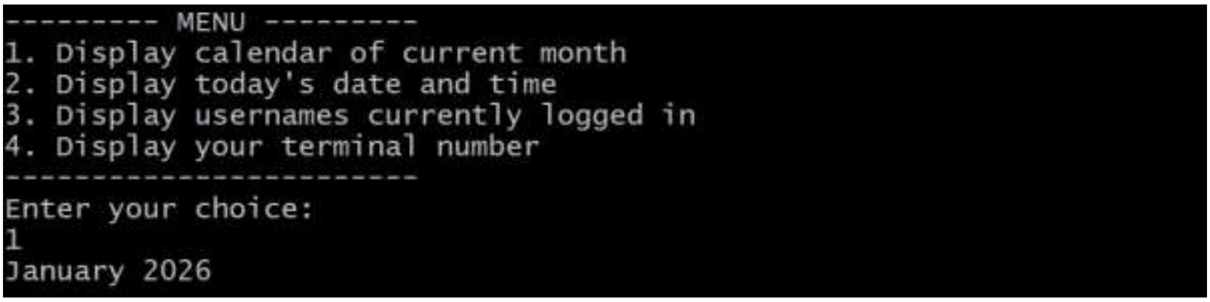
- **Display calendar of current month.**
- **Display today’s date and time.**
- **Display usernames those are currently logged in the system.**
- **Display your terminal number.**

CODE:

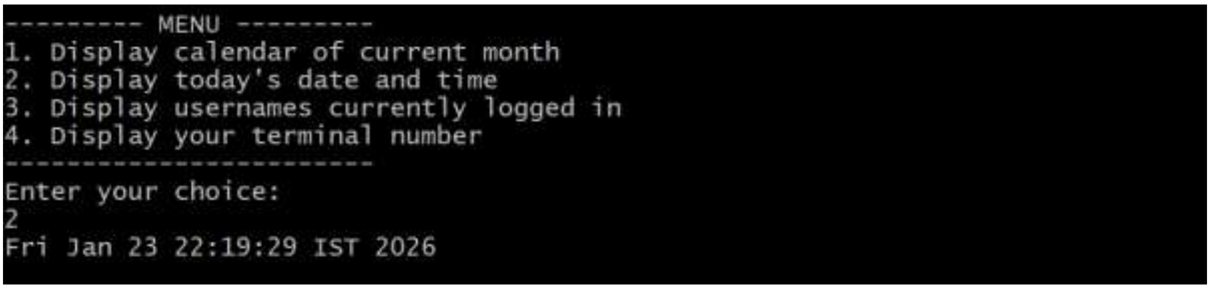


```
Shweta Bambode@LAPTOP-CFA3632L MINGW64 ~/OneDrive/Desktop (master)
$ echo "--------- MENU ---------"
echo "1. Display calendar of current month"
echo "2. Display today's date and time"
echo "3. Display usernames currently logged in"
echo "4. Display your terminal number"
echo "------------------------"

echo "Enter your choice:"
read ch

case $ch in
1)
    date +"%B %Y"
    ;;
2)
    date
    ;;
3)
    who
    ;;
4)
    tty
    ;;
*)
    echo "Invalid choice"
    ;;
esac
```

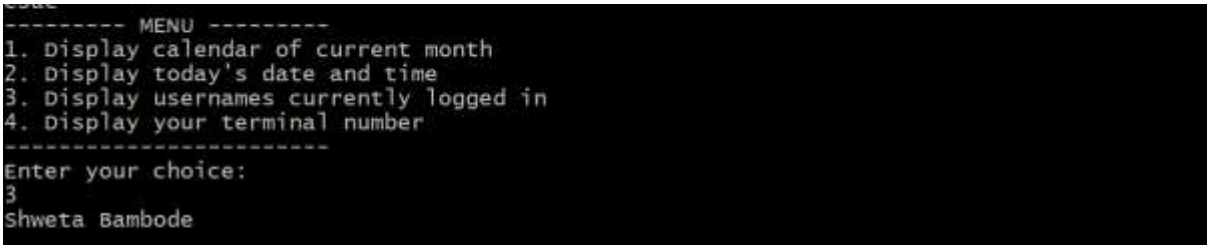
OUTPUT:



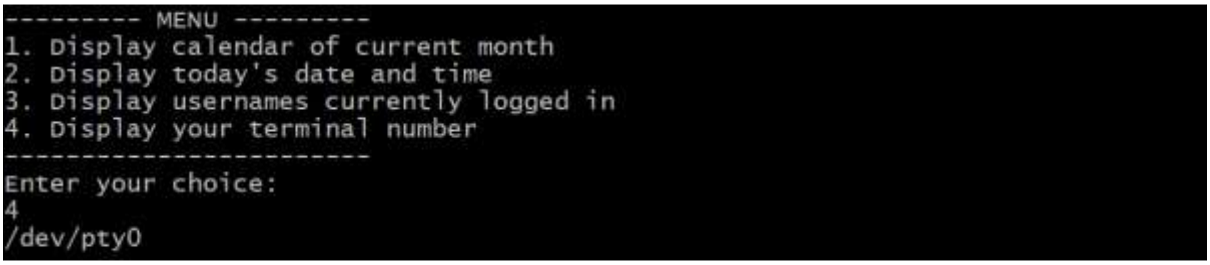
```
--------- MENU ---------
1. Display calendar of current month
2. Display today's date and time
3. Display usernames currently logged in
4. Display your terminal number
------------------------
Enter your choice:
1
January 2026
```



```
--------- MENU ---------
1. Display calendar of current month
2. Display today's date and time
3. Display usernames currently logged in
4. Display your terminal number
------------------------
Enter your choice:
2
Fri Jan 23 22:19:29 IST 2026
```



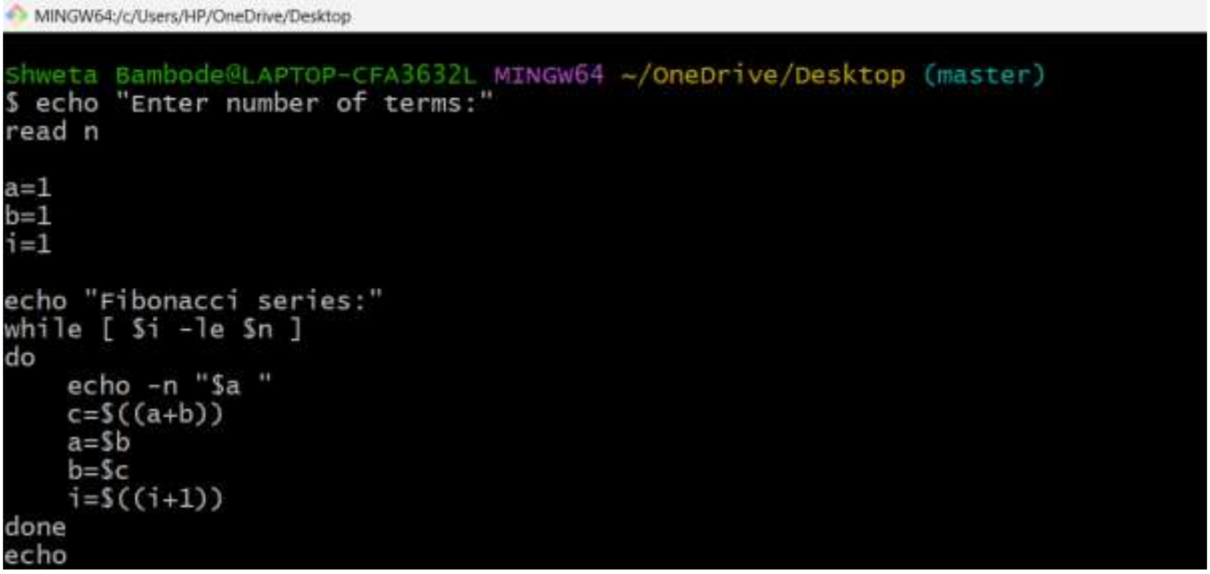
```
--------- MENU ---------
1. Display calendar of current month
2. Display today's date and time
3. Display usernames currently logged in
4. Display your terminal number
------------------------
Enter your choice:
3
Shweta Bambode
```



```
--------- MENU ---------
1. Display calendar of current month
2. Display today's date and time
3. Display usernames currently logged in
4. Display your terminal number
------------------------
Enter your choice:
4
/dev/pty0
```

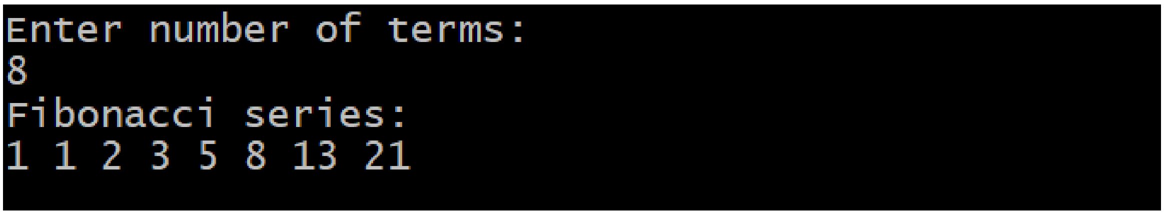
C] Write a shell script which will generate first *n* Fibonacci numbers like: 1, 1, 2, 3, 5, 13

CODE:



```
Shweta Bambode@LAPTOP-CFA3632L MINGW64 ~/OneDrive/Desktop (master)
$ echo "Enter number of terms:"
read n

a=1
b=1
i=1

echo "Fibonacci series:"
while [ $i -le $n ]
do
    echo -n "$a "
    c=$((a+b))
    a=$b
    b=$c
    i=$((i+1))
done
echo
```

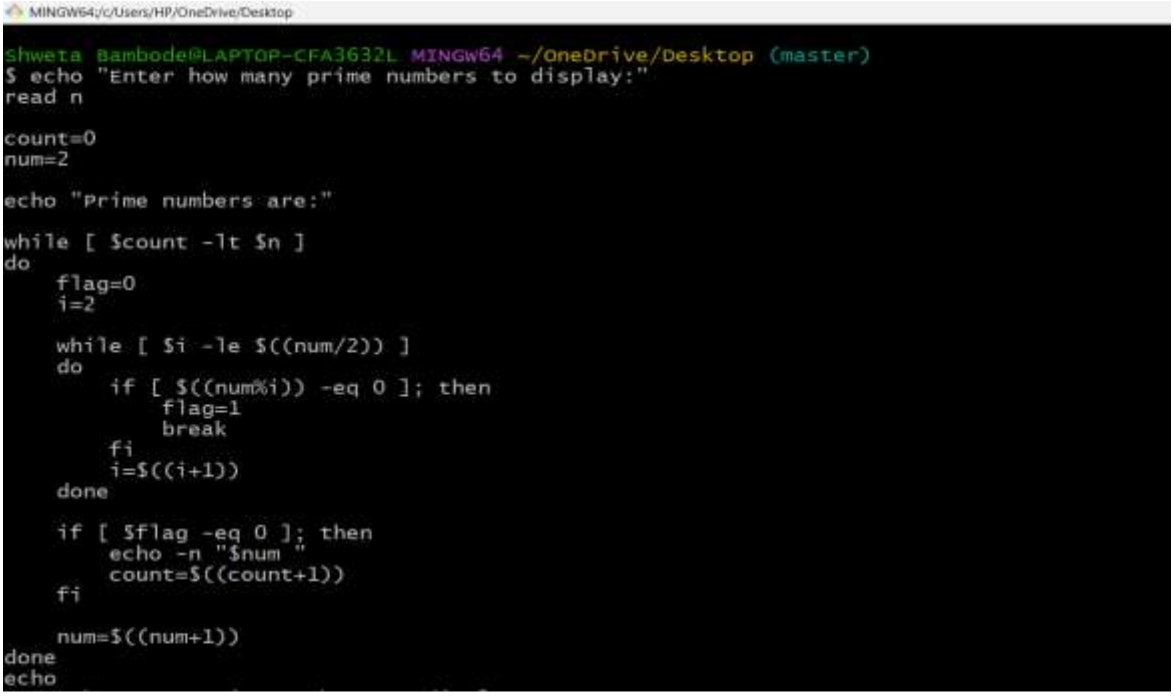
OUTPUT:



```
Enter number of terms:
8
Fibonacci series:
1 1 2 3 5 8 13 21
```

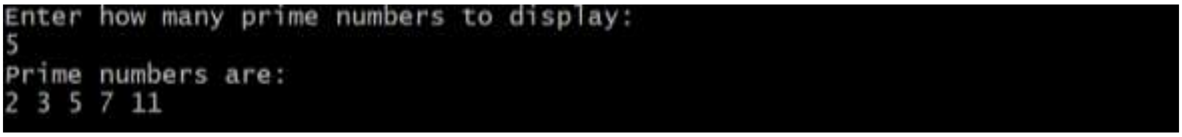
D] Write a shell script which will accept a number *n* and display first *n* prime numbers as output.

CODE:



```
Shweta Bambode@LAPTOP-CFA3632L MINGW64 ~/OneDrive/Desktop (master)
$ echo "Enter how many prime numbers to display:"
read n

count=0
num=2

echo "Prime numbers are:"

while [ $count -lt $n ]
do
    flag=0
    i=2

    while [ $i -le $((num/2)) ]
    do
        if [ $((num%i)) -eq 0 ]; then
            flag=1
            break
        fi
        i=$((i+1))
    done

    if [ $flag -eq 0 ]; then
        echo -n "$num "
        count=$((count+1))
    fi

    num=$((num+1))
done
echo
```

OUTPUT:

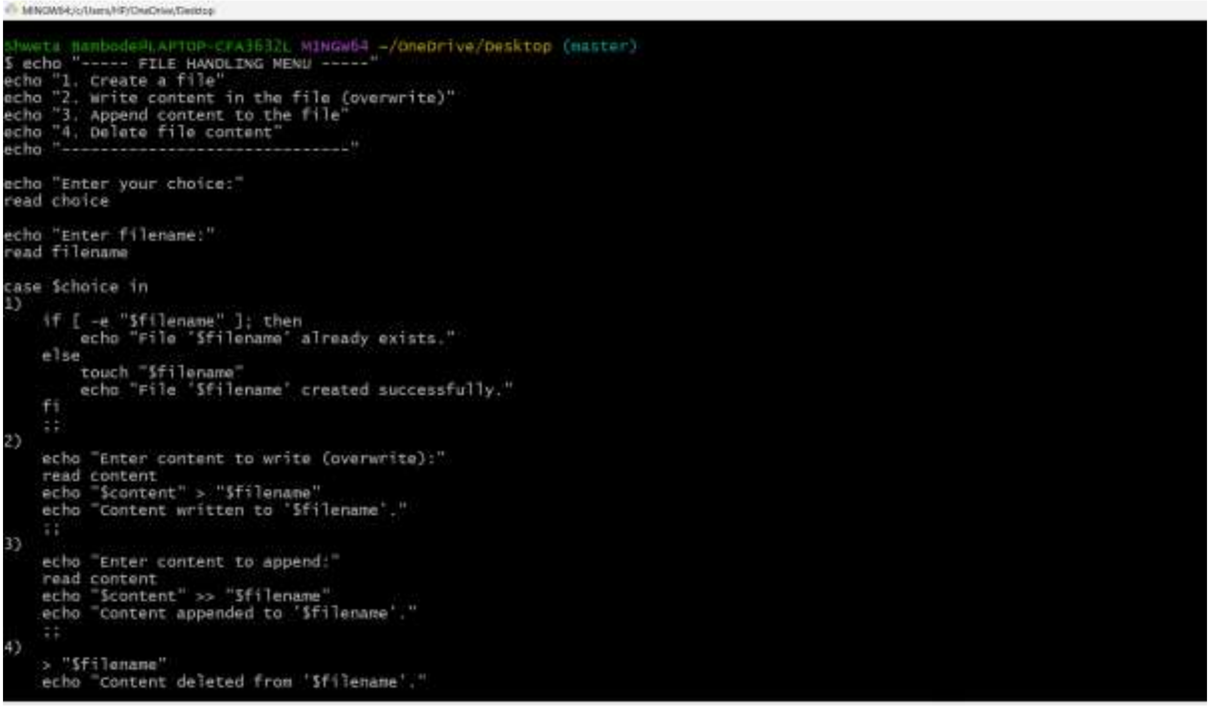


```
Enter how many prime numbers to display:
5
Prime numbers are:
2 3 5 7 11
```

e) Write menu driven program for file handling activity

- **Creation of file.**
- **Write content in the file.**
- **Upend file content.**
- **Delete file content.**

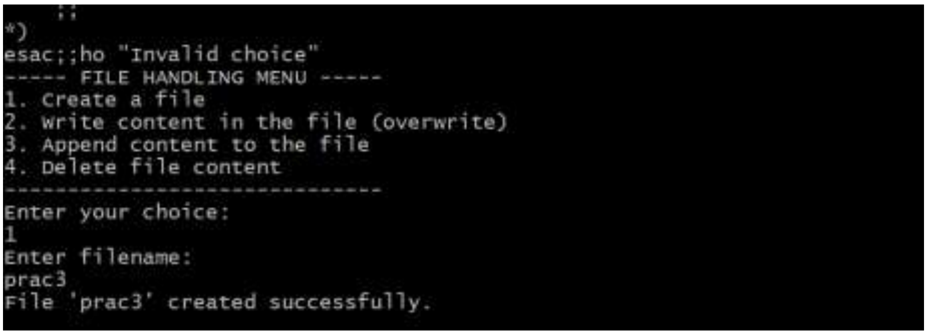
CODE:



MINGW64:/c/Users/HP/OneDrive/Desktop

```
Shweta Bambode@LAPTOP-CFA3632L MINGW64 ~/OneDrive/Desktop (master)
$ echo "----- FILE HANDLING MENU -----"
echo "1. Create a file"
echo "2. Write content in the file (overwrite)"
echo "3. Append content to the file"
echo "4. Delete file content"
echo "------------------------------"

echo "Enter your choice:"
read choice

echo "Enter filename:"
read filename

case $choice in
1)
    if [ -e "$filename" ]; then
        echo "File '$filename' already exists."
    else
        touch "$filename"
        echo "File '$filename' created successfully."
    fi
    ;;
2)
    echo "Enter content to write (overwrite):"
    read content
    echo "$content" > "$filename"
    echo "Content written to '$filename'."
    ;;
3)
    echo "Enter content to append:"
    read content
    echo "$content" >> "$filename"
    echo "Content appended to '$filename'."
    ;;
4)
    > "$filename"
    echo "Content deleted from '$filename'."
```

OUTPUT:



```
    ;;
*)
esac;;ho "Invalid choice"
----- FILE HANDLING MENU -----
1. Create a file
2. Write content in the file (overwrite)
3. Append content to the file
4. Delete file content
------------------------------
Enter your choice:
1
Enter filename:
prac3
File 'prac3' created successfully.
```