

Practical No. :3

USN NO. :CM24120

1. 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.

```
script.sh 44b3dtvdr

1 #!/bin/bash
2
3 echo "Enter Student Name:"
4 read name
5
6 echo "Enter Subject 1 Name:"
7 read sub1
8 echo "Enter marks for $sub1:"
9 read m1
10
11 echo "Enter Subject 2 Name:"
12 read sub2
13 echo "Enter marks for $sub2:"
14 read m2
15
16 echo "Enter Subject 3 Name:"
17 read sub3
18 echo "Enter marks for $sub3:"
19 read m3
20
21 total=$((m1 + m2 + m3))
22 percentage=$((total / 3))
23
24 if [ $percentage -ge 60 ]; then
25     class="First Class"
26 elif [ $percentage -ge 50 ]; then
27     class="Second Class"
28 elif [ $percentage -ge 40 ]; then
29     class="Pass"
30 else
31     class="Fail"
32 fi
33
34 echo "-----"
35 echo "          MARK SHEET"
36 echo "-----"
37 echo "Student Name : $name"
38 echo
39 echo "Subject | Marks"
40 echo "$sub1 | $m1"
41 echo "$sub2 | $m2"
42 echo "$sub3 | $m3"
43 echo "-----"
```

STUDIN

jay
math
67

Output:

Enter Student Name:
Enter Subject 1 Name:
Enter marks for math:
Enter Subject 2 Name:
Enter marks for phys:
Enter Subject 3 Name:
Enter marks for chem:

MARK SHEET

Student Name : jay

Subject	Marks
math	67
phys	79
chem	82

Total Marks : 228
Percentage : 76%
Class : First Class

2. 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

```
1 #!/bin/bash
2
3 while true
4 do
5     echo "-----"
6     echo "          MENU"
7     echo "-----"
8     echo "1. Display calendar of current month"
9     echo "2. Display today's date and time"
10    echo "3. Display usernames currently logged in"
11    echo "4. Display your terminal number"
12    echo "5. Exit"
13    echo "Enter your choice:"
14    read choice
15
16    case $choice in
17        1)
18            echo "Calendar of current month (simplified):"
19            echo "Sun Mon Tue Wed Thu Fri Sat"
20            for i in {1..30}
21            do
22                printf "%2d " $i
23                if (( i % 7 == 0 )); then
24                    echo
25                fi
26            done
27            echo
28            ;;
29        2)
30            echo "Today's date and time:"
31            date
32            ;;
33        3)
34            echo "Currently logged-in users (simulated):"
35            echo "$USER" # $USER is the current username
36            ;;
37        4)
38            echo "Terminal number (simulated):"
39            echo "/dev/tty1" # cannot detect real terminal in online compiler
40            ;;
41        5)
42            echo "Exiting..."
43            exit
44    esac
45 done
```

STUDIN

3
4
5

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
5. Exit

Enter your choice:

Calendar of current month (simplified):

Sun Mon Tue Wed Thu Fri Sat

1 2 3 4 5 6 7
8 9 10 11 12 13 14
15 16 17 18 19 20 21
22 23 24 25 26 27 28
29 30

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
5. Exit

Enter your choice:

Today's date and time:

Mon Jan 19 03:50:10 PM UTC 2026

```

9  echo "2. Display today's date and time"
10 echo "3. Display usernames currently logged in"
11 echo "4. Display your terminal number"
12 echo "5. Exit"
13 echo "Enter your choice:"
14 read choice
15
16 case $choice in
17 1) echo "Calendar of current month (simplified):"
18    echo "Sun Mon Tue Wed Thu Fri Sat"
19    for i in {1..30} # assuming 30 days
20    do
21        printf "%2d " $i
22        if (( i % 7 == 0 )); then
23            echo
24        fi
25    done
26    echo
27 ;;
28 2) echo "Today's date and time:"
29    date
30 ;;
31 3) echo "Currently logged-in users (simulated):"
32    echo "$USER" # $USER is the current username
33 ;;
34 4) echo "Terminal number (simulated):"
35    echo "/dev/tty1" # cannot detect real terminal in online compiler
36 ;;
37 5) echo "Exiting..."
38    exit
39 ;;
40 *) echo "Invalid choice. Please enter 1-5."
41 ;;
42 esac
43 done
44
45
46
47
48
49
50

```

```

3
4
5
-----
1. Display calendar of current month
2. Display today's date and time
3. Display usernames currently logged in
4. Display your terminal number
5. Exit
Enter your choice:
Currently logged-in users (simulated):

-----
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
5. Exit
Enter your choice:
Terminal number (simulated):
/dev/tty1
-----
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
5. Exit
Enter your choice:
Exiting...

```

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

```

1  #!/bin/bash
2
3  echo "Enter the number of terms (n):"
4  read n
5
6  # First two Fibonacci numbers
7  a=1
8  b=1
9
10 echo "Fibonacci Series:"
11
12 if [ $n -ge 1 ]; then
13     echo -n "$a "
14 fi
15
16 if [ $n -ge 2 ]; then
17     echo -n "$b "
18 fi
19
20 # Generate remaining terms
21 for ((i=3; i<=n; i++))
22 do
23     c=$((a + b))
24     echo -n "$c "
25     a=$b
26     b=$c
27 done
28
29 echo
30

```

```

STDIN
7

Output:
Enter the number of terms (n):
Fibonacci Series:
1 1 2 3 5 8 13

```

4. Write a shell script which will accept a number b and display first n prime numbers as output

```

1  #!/bin/bash
2
3  echo "Enter the number of prime numbers to display:"
4  read n
5
6  count=0 # how many primes we have found
7  num=2 # number to check
8
9  echo "First $n prime numbers:"
10
11 while [ $count -lt $n ]
12 do
13     is_prime=1 # assume num is prime
14
15     # check if num is divisible by any number from 2 to sqrt(num)
16     for ((i=2; i*i<=num; i++))
17     do
18         if [ $((num % i)) -eq 0 ]; then
19             is_prime=0
20             break
21         fi
22     done
23
24     if [ $is_prime -eq 1 ]; then
25         echo -n "$num "
26         count=$((count + 1))
27     fi
28     num=$((num + 1))
29 done
30
31 echo
32
33

```

```

STDIN
10

Output:
Enter the number of prime numbers to display:
First 10 prime numbers:
2 3 5 7 11 13 17 19 23 29

```

5. Write menu driven program for file handling activity Creation of file • Write content in the file Upend file content Delete file content

```
1 #!/bin/bash
2
3 while true
4 do
5     echo "-----"
6     echo "FILE HANDLING MENU"
7     echo "-----"
8     echo "1. Create file"
9     echo "2. Write content to file"
10    echo "3. Append content to file"
11    echo "4. Delete file content"
12    echo "5. Exit"
13    echo "Enter choice:"
14    read choice
15
16    case $choice in
17        1)
18            echo "Enter filename:"
19            read file
20            touch "$file"
21            echo "File '$file' created."
22            ;;
23        2)
24            echo "Enter filename:"
25            read file
26            echo "Enter content:"
27            read content
28            echo "$content" > "$file"
29            echo "Content written to '$file'."
30            ;;
31        3)
32            echo "Enter filename:"
33            read file
34            echo "Enter content to append:"
35            read content
36            echo "$content" >> "$file"
37            echo "Content appended to '$file'."
38            ;;
39        4)
40            echo "Enter filename:"
41            read file
42            > "$file"
```

```
STUDIN
1
2 practical.pdf
3
-----
Output:
-----
FILE HANDLING MENU
-----
1. Create file
2. Write content to file
3. Append content to file
4. Delete file content
5. Exit
Enter choice:
Enter filename:
File 'practical.pdf' created.
-----
FILE HANDLING MENU
-----
1. Create file
2. Write content to file
3. Append content to file
4. Delete file content
5. Exit
Enter choice:
Enter filename:
Enter content:
Content written to 'practical.pdf'.
-----
FILE HANDLING MENU
-----
1. Create file
2. Write content to file
```

```
1 #!/bin/bash
2
3 while true
4 do
5     echo "-----"
6     echo "FILE HANDLING MENU"
7     echo "-----"
8     echo "1. Create file"
9     echo "2. Write content to file"
10    echo "3. Append content to file"
11    echo "4. Delete file content"
12    echo "5. Exit"
13    echo "Enter choice:"
14    read choice
15
16    case $choice in
17        1)
18            echo "Enter filename:"
19            read file
20            touch "$file"
21            echo "File '$file' created."
22            ;;
23        2)
24            echo "Enter filename:"
25            read file
26            echo "Enter content:"
27            read content
28            echo "$content" > "$file"
29            echo "Content written to '$file'."
30            ;;
31        3)
32            echo "Enter filename:"
33            read file
34            echo "Enter content to append:"
35            read content
36            echo "$content" >> "$file"
37            echo "Content appended to '$file'."
38            ;;
39        4)
40            echo "Enter filename:"
41            read file
42            > "$file"
```

```
STUDIN
1
2 practical.pdf
3 hello world
4
-----
Output:
-----
FILE HANDLING MENU
-----
1. Create file
2. Write content to file
3. Append content to file
4. Delete file content
5. Exit
Enter choice:
Enter filename:
File 'practical.pdf' created.
-----
FILE HANDLING MENU
-----
1. Create file
2. Write content to file
3. Append content to file
4. Delete file content
5. Exit
Enter choice:
Enter filename:
Enter content:
Content written to 'practical.pdf'.
-----
FILE HANDLING MENU
-----
1. Create file
2. Write content to file
```

```
12 echo $?. EXIT
13 echo "Enter choice:"
14 read choice
15
16 case $choice in
17 1) echo "Enter filename:"
18     read file
19     touch "$file"
20     echo "File '$file' created."
21     ;;
22 2) echo "Enter filename:"
23     read file
24     echo "Enter content:"
25     read content
26     echo "$content" > "$file"
27     echo "Content written to '$file'."
28     ;;
29 3) echo "Enter filename:"
30     read file
31     echo "Enter content to append:"
32     read content
33     echo "$content" >> "$file"
34     echo "Content appended to '$file'."
35     ;;
36 4) echo "Enter filename:"
37     read file
38     > "$file"
39     echo "Content of '$file' deleted."
40     ;;
41 5) echo "Exiting..."
42     exit
43     ;;
44 *) echo "Invalid choice"
45     ;;
46 esac
47 done
```

STDIN

3

practical.pdf

this is append

1. Create file

2. Write content to file

3. Append content to file

4. Delete file content

5. Exit

Enter choice:

Enter filename:

Enter content:

Content written to 'practical.pdf'.

FILE HANDLING MENU

1. Create file

2. Write content to file

3. Append content to file

4. Delete file content

5. Exit

Enter choice:

Enter filename:

Enter content to append:

Content appended to 'practical.pdf'.

FILE HANDLING MENU

1. Create file

2. Write content to file

3. Append content to file

4. Delete file content

5. Exit

```
12 echo $?. EXIT
13 echo "Enter choice:"
14 read choice
15
16 case $choice in
17 1) echo "Enter filename:"
18     read file
19     touch "$file"
20     echo "File '$file' created."
21     ;;
22 2) echo "Enter filename:"
23     read file
24     echo "Enter content:"
25     read content
26     echo "$content" > "$file"
27     echo "Content written to '$file'."
28     ;;
29 3) echo "Enter filename:"
30     read file
31     echo "Enter content to append:"
32     read content
33     echo "$content" >> "$file"
34     echo "Content appended to '$file'."
35     ;;
36 4) echo "Enter filename:"
37     read file
38     > "$file"
39     echo "Content of '$file' deleted."
40     ;;
41 5) echo "Exiting..."
42     exit
43     ;;
44 *) echo "Invalid choice"
45     ;;
46 esac
47 done
```

STDIN

4

practical.pdf

Content of 'practical.pdf' deleted.

4. Delete file content

5. Exit

Enter choice:

Enter filename:

Enter content:

Content written to 'practical.pdf'.

FILE HANDLING MENU

1. Create file

2. Write content to file

3. Append content to file

4. Delete file content

5. Exit

Enter choice:

Enter filename:

Enter content to append:

Content appended to 'practical.pdf'.

FILE HANDLING MENU

1. Create file

2. Write content to file

3. Append content to file

4. Delete file content

5. Exit

Enter choice:

Enter filename:

Content of 'practical.pdf' deleted.