



**Student Name:** Rohit

**Branch:** MCA (Genral)

**Semester:** 1st

**Subject Name:** : Linux Administration Lab

**UID:** 24MCA20122

**Group/Section:** 2/B

**Date of Performance:** 04-11-2024

**Subject Code:** 24CAP-607

## **Project Report: Calendar Display Script Using Bash**

### **Objective**

The objective of this project is to create a Bash script that displays a calendar for a specified month and year, or for the current month if no input is provided. This script offers users a quick way to view calendar data without having to manually enter calendar commands or navigate to other applications.Scope

This script allows users to:

- Display the calendar of any specified month and year.
- View the calendar for the current month if no input is provided.
- Execute the script on any Unix-based system that supports the `cal` command, making it a simple and versatile tool.

---

### **Technology Stack**

#### **Operating System**

Linux (compatible with other Unix-based systems, such as macOS)

#### **Programming Language**

Bash Scripting

#### **Tools and Utilities**

Bash shell for script execution

cal command for calendar display functionality

## Implementation

### Script Overview

This script, written in Bash, allows users to input a month and year. Based on the input, it displays the calendar for the specified month and year using the `cal` command. If the user provides no input, the script defaults to displaying the calendar for the current month.

### Code Explanation

#### Full Script

```
#!/bin/bash

# Function to display the calendar for a given month and year
display_calendar() {
    local month=$1
    local year=$2

    # Check if both month and year are provided
    if [[ -n $month && -n $year ]]; then
        cal $month $year
    elif [[ -n $year ]]; then
        cal $year
    else
        # Display the current month if no input is provided
        cal
    fi
}

# Prompt user for month and year
echo "Enter month (1-12, leave blank for current month): "
read month

echo "Enter year (e.g., 2024, leave blank for current year): "
read year

# Display calendar based on user input
echo "Here is the calendar:"
display_calendar $month $year
```

## Result/Output/Writing Summary:

```
Enter month (1-12, leave blank for current month): 3
Enter year (e.g., 2024, leave blank for current year): 2024
Here is the calendar:
    March 2024
Su Mo Tu We Th Fr Sa
                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
31
```

## Features and Advantages

### 4.1 Features

- **Month and Year Selection:** Users can specify any month and year to view the corresponding calendar.
- **Current Month Display:** If no input is provided, the script displays the current month's calendar.
- **User-Friendly Prompting:** Simple prompts guide the user in providing inputs, making the script easy to use.

### 4.2 Advantages

- **Flexibility:** Allows users to view calendars from any month or year, making it useful for planning and scheduling.
- **Portability:** As a Bash script, it can be run on any Unix-based system with the `cal` command available.
- **Minimal Resources:** The script is lightweight and requires minimal system resources, running quickly and efficiently in the terminal.

## Learning Outcomes

### Enhanced Bash Scripting Skills

### Enhanced Problem-Solving and Logic Development