Q. Comprehensive Process Management in Modern Operating Systems.

1. Creating Process

2. Killing Process

3. Scheduling Process

4. changing Process priorities

5. which process is running now

6. Background process

# Function to create a process

create\_process() {

    echo "Creating a process..."

    sleep 100 & # This creates a dummy process that runs in the background

    echo "Process created with PID: $!"

}

# Function to kill a process

kill\_process() {

    echo "Enter the PID of the process to kill:"

    read pid

    kill $pid

    if [ $? -eq 0 ]; then

        echo "Process $pid killed successfully."

    else

        echo "Failed to kill process $pid."

    fi

}

# Function to display process scheduling information

schedule\_process() {

    echo "Process scheduling information:"

    ps -eo pid,comm,pri,nice,state,etime --sort=-pri | head -n 10

}

# Function to change process priority

change\_priority() {

    echo "Enter the PID of the process to change priority:"

    read pid

    echo "Enter the new priority (-20 to 19, lower is higher priority):"

    read priority

    renice $priority -p $pid

    if [ $? -eq 0 ]; then

        echo "Priority of process $pid changed to $priority."

    else

        echo "Failed to change priority of process $pid."

    fi

}

# Function to display the currently running process

current\_process() {

    echo "Currently running processes:"

    ps -eo pid,comm,user,state,%cpu,%mem,etime --sort=-%cpu | head -n 10

}

# Function to handle background processes

background\_process() {

    echo "Background processes:"

    jobs -l

}

# Function to show all information about processes

display\_all\_info() {

    echo "All process information:"

    ps -eo pid,ppid,comm,user,pri,nice,state,%cpu,%mem,etime,start,time --sort=-%cpu

}

Bash Script



# Function to handle background processes

background\_process() {

    echo "Background processes:"

    jobs -l

}

# Function to show all information about processes

display\_all\_info() {

    echo "All process information:"

    ps -eo pid,ppid,comm,user,pri,nice,state,%cpu,%mem,etime,start,time --sort=-%cpu

}

# Menu to perform the actions

while true; do

    echo "\nProcess Management Options:"

    echo "1. Create a process"

    echo "2. Kill a process"

    echo "3. Display process scheduling information"

    echo "4. Change process priority"

    echo "5. Display currently running processes"

    echo "6. Show background processes"

    echo "7. Display all process information"

    echo "8. Exit"

    echo "Choose an option (1-8):"

    read choice

    case $choice in

    1)

        create\_process

        ;;

    2)

        kill\_process

        ;;

    3)

        schedule\_process

        ;;

    4)

        change\_priority

        ;;

    5)

        current\_process

        ;;

    6)

        background\_process

        ;;

    7)

        display\_all\_info

        ;;

    8)

        echo "Exiting..."

        break

        ;;

    \*)

        echo "Invalid option. Please choose again."

        ;;

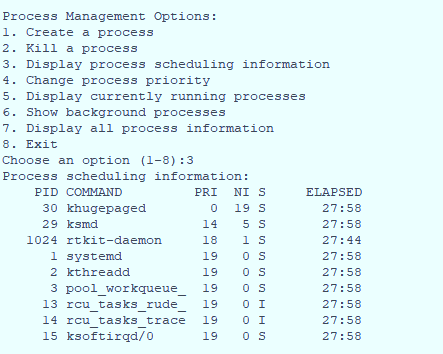
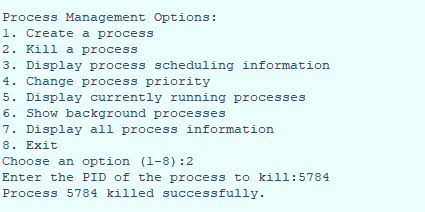
    esac

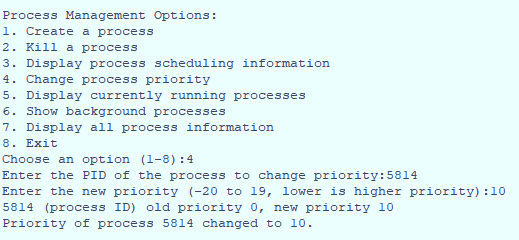
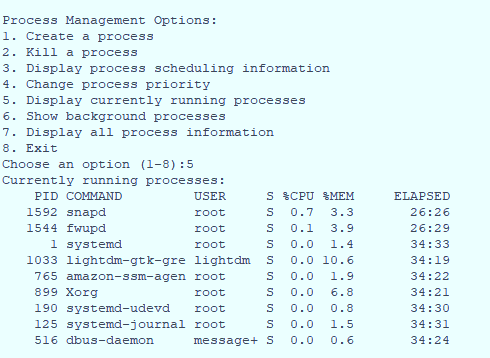
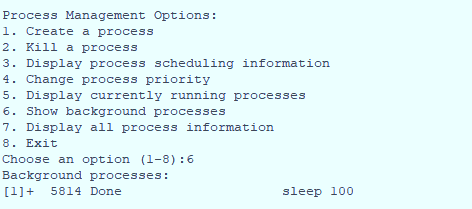
done

Bash Script



Outputs





Outputs

