## **Practical No. 3: Implementing All Process Management Commands of UNIX**

### **Aim:**

To practice and implement various process management commands of the UNIX/Linux operating system.

### **Commands Used and Their Description:**

1. **ls, ls -l, ls -la**
   * Used to list directory contents.
   * -l gives detailed view (permissions, owner, size, etc.), -a includes hidden files.
2. **ps, ps -e, ps aux**
   * Shows active processes.
   * ps -e: Lists all processes.
   * ps aux: Displays detailed info including CPU/memory usage.
3. **whoami**
   * Displays the current logged-in username.
4. **pwd**
   * Prints the present working directory.
5. **cat, cat >> filename**
   * View or append text to files.
6. **touch filename**
   * Create a new empty file.
7. **mkdir foldername**
   * Create a new directory.
8. **date**
   * Displays the system's current date and time.
9. **cp source destination**
   * Copies files/directories.
10. **mv source destination**

* Moves or renames files.

1. **fg, bg**

* Brings jobs to foreground/background respectively.
* fg: resume a stopped job in foreground.
* bg: resume in background.

1. **kill <PID>**

* Terminates a process using its PID.

1. **whoami &**

* Executes the command in background and returns its PID.

### **Sample Execution and Observations:**

* ps showed multiple kernel processes like kworker, rcu\_gp, etc.

Foreground job handling failed as no jobs were running:  
  
 yaml  
CopyEdit  
bash: fg: current: no such job

Background job:  
  
 bash  
CopyEdit  
whoami &

[1] 4275

Process was successfully killed using:  
  
 bash  
CopyEdit  
kill 3675

### **Conclusion:**

This practical demonstrated essential UNIX process management commands including viewing processes, handling jobs, background/foreground execution, and process termination using kill. It enhances understanding of how Linux manages and monitors tasks in a multitasking environment.