

# Bash Shell Scripting

Scheduling jobs with at and crontab  
Part-1

```
#!/bin/bash
```

*Learn how to automate common tasks using bash shell scripting*

# Scheduling jobs with at and crontab:

- at command is very useful for scheduling one time tasks.
- Example:
  - Shutdown system at the specified time
  - Taking a one-time backup.
- Syntax:
  - echo "bash backup.sh" | at 9:00 AM
  - Or
  - Run first: **at 9:00 AM** then enter and give the **cmd** or **script** to run and press ctrl+D.
- **Commands used with at:**
  - at : execute commands at specified time.
  - atq : lists the pending jobs of users.
  - atrm : delete jobs by their job number.

*Learn how to automate common tasks using bash shell scripting*

# Examples of at command:

- Ex-1: Schedule task at coming 10:00 AM.
  - **at 10:00 AM**
- Ex-2: Schedule task at 10:00 AM on coming Sunday.
  - **at 10:00 AM Sun**
- Ex-3: Schedule task at 10:00 AM on coming 25'th July.
  - **at 10:00 AM July 25**
- Ex-4: Schedule task at 10:00 AM on coming 22'nd June 2025.
  - **at 10:00 AM 6/22/2015**
  - **at 10:00 AM 6.22.2015**
- Ex-5: Schedule task at 10:00 AM on the same date at next month.
  - **at 10:00 AM next month**
- Ex-6: Schedule task at 10:00 AM tomorrow.
  - **at 10:00 AM tomorrow**

*Learn how to automate common tasks using bash shell scripting*

# Examples of at command...

- Ex-7: Schedule task to execute just after 1 hour.
  - **at now + 1 hour**
- Ex-8: Schedule task to execute just after 30 minutes.
  - **at now + 30 minutes**
- Ex-9: Schedule task to execute just after 1 and 2 weeks.
  - **at now + 1 week**
  - **at now + 2 weeks**
- Ex-10: Schedule task to execute just after 1 and 2 years.
  - **at now + 1 year**
  - **at now + 2 years**
- Ex-1: Schedule task to execute at midnight.
  - **at midnight**

*Learn how to automate common tasks using bash shell scripting*

**Thank you**