Ansible Ad-Hoc Commands & Modules | Using Modules with Ad-Hoc Commands

Definition - Ansible ad-hoc commands are mainly used for quick fixes, validation, troubleshooting and emergency tasks in production.

Note – install - ansible-galaxy collection install ansible.posix

🔥 1️⃣ ping module (Connectivity check)

```
ansible all -m ping

# inventory ke sab hosts ki connectivity check karta hai


ansible web -m ping

# web group ke hosts reachable hain ya nahi check karta hai


ansible db -m ping

# db group ke servers ki connectivity test karta hai


ansible all -m ping -i hosts

# specified inventory file (hosts) se sab machines check karta hai


ansible all -m ping -u root

# root user se SSH connect karke ping module chalata hai


ansible all -m ping -k

# SSH password pooch kar hosts ki connectivity check karta hai


ansible all -m ping --become

# sudo (privilege escalation) ke saath ping run karta hai


ansible all -m ping --become-user=root

# sudo ke through root user ban kar ping chalata hai


ansible webservers -m ping
```

```
# webservers group ke hosts ka connection test karta hai

ansible 192.168.1.10 -m ping
# sirf ek specific IP host ki connectivity check karta hai
```

🔥 2️⃣ command module (Simple command)

```
ansible all -m command -a "uptime"
# sab hosts ka system uptime aur load average dikhata hai

ansible all -m command -a "df -h"
# sab hosts par disk usage human-readable format me dikhata hai

ansible all -m command -a "free -m"
# sab hosts par RAM usage MB me dikhata hai

ansible all -m command -a "ls /tmp"
# sab hosts par /tmp directory ki files list karta hai

ansible all -m command -a "cat /etc/os-release"
# sab hosts ka OS/version information dikhata hai

ansible web -m command -a "hostname"
# web group ke servers ka hostname dikhata hai

ansible db -m command -a "date"
# db servers par current date aur time dikhata hai

ansible all -m command -a "whoami"
# batata hai command kaunse user ke through run ho rahi hai

ansible all -m command -a "ip a"
```

```
# sab hosts ke network interfaces aur IP address dikhata hai
```

```
ansible all -m command -a "systemctl status sshd"

# sab hosts par SSH service ka status check karta hai
```

```
ansible all -m command -a "lvs"

# sab hosts par LVM logical volumes ki list dikhata hai
```

❌ pipe | allowed nahi

🔥 3️⃣ shell module (Pipe / redirect allowed)

```
ansible all -m shell -a "df -h | grep /dev"

# sab hosts par mounted disks ka usage filter karke dikhata hai
```

```
ansible all -m shell -a "ps -ef | grep httpd"

# sab hosts par httpd process running hai ya nahi check karta hai
```

```
ansible all -m shell -a "uptime > /tmp/up.txt"

# sab hosts par uptime ka output file me save karta hai
```

```
ansible all -m shell -a "cat /etc/passwd | wc -l"

# sab hosts par total users ki count dikhata hai
```

```
ansible all -m shell -a "ls / | tee /tmp/root.txt"

# root directory list karta hai aur output file me bhi save karta hai
```

```
ansible all -m shell -a "echo hello >> /tmp/a.txt"

# file me hello text append karta hai
```

```
ansible all -m shell -a "netstat -tulnp | grep 80"

# port 80 par kaunsa service listen kar raha hai check karta hai
```

```
ansible all -m shell -a "df -Th"

# sab hosts par disk usage filesystem type ke saath dikhata hai
```

```
ansible all -m shell -a "free -m"

# sab hosts par memory usage MB me dikhata hai
```

```
ansible all -m shell -a "uname -r"

# sab hosts par running kernel version dikhata hai
```

🔥 4️⃣ copy module (File push)

```
ansible all -m copy -a "src=/etc/hosts dest=/tmp/hosts"

# control node se sab hosts par hosts file copy karta hai
```

```
ansible all -m copy -a "src=test.txt dest=/opt/test.txt"

# test.txt file ko /opt directory me copy karta hai
```

```
ansible web -m copy -a "src=a.txt dest=/tmp/a.txt"

# sirf web group ke servers par file copy karta hai
```

```
ansible all -m copy -a "content='hello' dest=/tmp/hello.txt"

# remote host par hello content ke saath file banata hai
```

```
ansible all -m copy -a "src=/root/a.sh dest=/usr/local/bin/a.sh mode=755"

# script copy karke executable permission deta hai
```

```
ansible db -m copy -a "src=db.cnf dest=/etc/db.cnf"

# db group ke servers par config file copy karta hai
```

```
ansible all -m copy -a "src=/etc/passwd dest=/tmp/passwd.bak"

# passwd file ka backup banata hai
```

```
ansible all -m copy -a "src=file.txt dest=/tmp/file.txt owner=root"

# copied file ka owner root set karta hai
```

```
ansible all -m copy -a "src=file.txt dest=/tmp/file.txt group=root"

# copied file ka group root set karta hai
```

```
ansible all -m copy -a "src=file.txt dest=/tmp/file.txt mode=644"

# copied file ka permission 644 set karta hai
```

🔥 5️⃣ file module (File/Dir permissions)

```
ansible all -m file -a "path=/opt/app state=directory"

# sab hosts par /opt/app directory create karta hai
```

```
ansible all -m file -a "path=/tmp/test state=touch"

# empty file /tmp/test create karta hai (touch jaisa)
```

```
ansible all -m file -a "path=/tmp/test mode=777"

# file/directory ka permission 777 set karta hai
```

```
ansible all -m file -a "path=/opt/app owner=root"

# /opt/app ka owner root set karta hai
```

```
ansible all -m file -a "path=/opt/app group=root"

# /opt/app ka group root set karta hai
```

```
ansible all -m file -a "path=/tmp/old state=absent"

# /tmp/old file ya directory delete karta hai
```

```
ansible web -m file -a "path=/data state=directory"

# web group ke servers par /data directory banata hai
```

```
ansible all -m file -a "path=/tmp/a.txt mode=600"

# file ka permission 600 set karta hai


ansible all -m file -a "path=/opt/logs state=directory"

# sab hosts par /opt/logs directory create karta hai


ansible db -m file -a "path=/backup state=directory"

# db group ke servers par /backup directory banata hai
```

🔥 6️⃣ yum / dnf module (Package install)

```
ansible all -m yum -a "name=httpd state=present"

# sab hosts par httpd package install karta hai


ansible all -m yum -a "name=vim state=latest"

# vim ko latest version me update/install karta hai


ansible all -m yum -a "name=nginx state=absent"

# nginx package uninstall/remove karta hai


ansible web -m dnf -a "name=httpd state=installed"

# web group ke servers par httpd install karta hai


ansible db -m dnf -a "name=mariadb state=present"

# db servers par mariadb package install karta hai


ansible all -m yum -a "name=tree state=present"

# sab hosts par tree command install karta hai


ansible all -m yum -a "name=git state=latest"

# git ko latest version me install/update karta hai
```

```
ansible all -m yum -a "name=wget state=present"

# sab hosts par wget install karta hai
```

```
ansible all -m dnf -a "name=rsync state=present"

# rsync package install karta hai
```

```
ansible all -m dnf -a "name=net-tools state=present"

# netstat jaise tools ke liye net-tools install karta hai
```

🔥 7️⃣ service / systemd module

```
ansible all -m service -a "name=httpd state=started"

# sab hosts par httpd service start karta hai
```

```
ansible all -m service -a "name=httpd state=stopped"

# sab hosts par httpd service stop karta hai
```

```
ansible all -m service -a "name=httpd state=restarted"

# sab hosts par httpd service restart karta hai
```

```
ansible all -m service -a "name=sshd enabled=yes"

# sshd service ko boot ke time enable karta hai
```

```
ansible web -m systemd -a "name=nginx state=started"

# web group ke servers par nginx service start karta hai
```

```
ansible all -m systemd -a "name=firewalld state=stopped"

# sab hosts par firewalld service stop karta hai
```

```
ansible all -m systemd -a "name=nginx enabled=yes"
```

```
# sab hosts par nginx service boot me start hone ke liye enable karta hai
```

```
ansible db -m service -a "name=mariadb state=started"
# db group ke servers par mariadb service start karta hai
```

```
ansible all -m service -a "name=crond state=restarted"
# sab hosts par crond service restart karta hai
```

```
ansible all -m systemd -a "daemon_reload=yes"
# systemd configuration reload karta hai
```

🔥 8️⃣ user module

```
ansible all -m user -a "name=devops"
# sab hosts par devops user create karta hai (default state=present)
```

```
ansible all -m user -a "name=ansible state=present"
# ansible user ensure karta hai ki exist kare
```

```
ansible all -m user -a "name=test state=absent"
# test user sab hosts se delete karta hai
```

```
ansible all -m user -a "name=linux uid=2000"
# linux user create karta hai with specific UID 2000
```

```
ansible all -m user -a "name=admin groups=wheel"
# admin user ko wheel group me add karta hai (sudo access ke liye)
```

```
ansible all -m user -a "name=dev shell=/bin/bash"
# dev user ka default shell /bin/bash set karta hai
```

```
ansible all -m user -a "name=ops create_home=yes"
```

```
# ops user create karta hai aur home directory banata hai


ansible all -m user -a "name=backup comment='backup user'"

# backup user create karta hai with description/comment


ansible all -m user -a "name=guest password='{{ '123' | password_hash('sha512') }}'"

# guest user create karta hai with encrypted password


ansible all -m user -a "name=monitoring"

# monitoring user create karta hai default settings ke saath
```

🔥 9️⃣ cron module

```
ansible all -m cron -a "name=backup job='/backup.sh'"

# sab hosts par backup.sh script ke liye cron job create karta hai (default schedule)


ansible all -m cron -a "name=backup minute=0 hour=2 job='/backup.sh'"

# backup.sh ko har din 2:00 AM par run karne ke liye cron set karta hai


ansible all -m cron -a "name=logclean job='rm -rf /tmp/*'"

# sab hosts par /tmp clean karne ke liye cron job create karta hai


ansible all -m cron -a "name=df job='df -h > /tmp/df.txt'"

# disk usage check karke /tmp/df.txt me save karne ke liye cron


ansible all -m cron -a "name=reboot special_time=reboot job='/script.sh'"

# system reboot hone par script.sh run karne ke liye cron set karta hai


ansible all -m cron -a "name=test state=absent"

# test cron job ko delete karta hai
```

```
ansible all -m cron -a "name=sync minute=*/5 job='rsync -av /data /backup'"
# har 5 minute me /data ko /backup me sync karne ke liye cron

ansible all -m cron -a "name=date job='date >> /tmp/date.txt'"
# date append karne ke liye cron job create karta hai

ansible all -m cron -a "name=uptime job='uptime'"
# uptime command ko run karne ke liye cron job create

ansible all -m cron -a "name=clean hour=1 job='rm -rf /old'"
# har din 1:00 AM par /old directory clean karne ke liye cron
```

🔥 🔟 reboot module

```
ansible all -m reboot
# sab hosts ko reboot karta hai

ansible web -m reboot
# web group ke servers ko reboot karta hai

ansible db -m reboot
# db group ke servers ko reboot karta hai

ansible all -m reboot -a "reboot_timeout=600"
# reboot ke liye timeout 600 seconds set karta hai

ansible all -m reboot -a "msg='Reboot by Ansible'"
# reboot ke waqt message display karta hai

ansible all -m reboot -b
# become (sudo) privileges ke saath reboot karta hai
```

```
ansible all -m reboot -u root

# root user ke saath reboot command run karta hai
```

```
ansible all -m reboot -a "connect_timeout=30"

# reboot ke liye 30 seconds connection timeout set karta hai
```

```
ansible all -m reboot -a "test_command=uptime"

# reboot se pehle uptime check karke connection verify karta hai
```

```
ansible all -m reboot --become

# sudo/privileged mode me reboot karta hai
```

🔥 1 1️⃣ fetch (remote → controller)

```
ansible all -m fetch -a "src=/etc/hosts dest=/tmp/hosts"

# sab hosts se /etc/hosts file controller machine par /tmp/hosts me copy karta hai
```

```
ansible web -m fetch -a "src=/var/log/messages dest=/tmp/logs"

# web servers se messages log fetch karke controller par /tmp/logs me save
```

```
ansible all -m fetch -a "src=/etc/passwd dest=/tmp/"

# sab hosts se /etc/passwd file copy karke controller ke /tmp/ me save
```

```
ansible db -m fetch -a "src=/etc/my.cnf dest=/tmp/"

# db servers se MySQL config fetch karke controller par /tmp/ me save
```

```
ansible all -m fetch -a "src=/var/log/secure dest=/tmp/"

# sab hosts ke /var/log/secure fetch karke controller par save
```

```
ansible all -m fetch -a "src=/etc/fstab dest=/tmp/"

# sab hosts se fstab file fetch karta hai
```

```
ansible web -m fetch -a "src=/etc/httpd/conf/httpd.conf dest=/tmp/"

# web servers ke apache config fetch karke controller par /tmp/ me save
```

```
ansible all -m fetch -a "src=/root/.bashrc dest=/tmp/"

# sab hosts ke root bash profile fetch karke controller par save
```

```
ansible db -m fetch -a "src=/var/log/mysqld.log dest=/tmp/"

# db servers ke MySQL logs fetch karke controller par /tmp/ me save
```

```
ansible all -m fetch -a "src=/etc/ssh/sshd_config dest=/tmp/"

# sab hosts se sshd config fetch karke controller par /tmp/ me save
```

🔥 1 2️⃣ stat (file info)

```
ansible all -m stat -a "path=/etc/passwd"

# /etc/passwd file ka information check karta hai (exist, permissions, size)
```

```
ansible all -m stat -a "path=/etc/shadow"

# /etc/shadow file ke permissions aur existence check karta hai
```

```
ansible all -m stat -a "path=/tmp/test"

# /tmp/test file ya directory info check karta hai
```

```
ansible web -m stat -a "path=/var/www/html"

# web servers ke /var/www/html directory ke details check
```

```
ansible db -m stat -a "path=/var/lib/mysql"

# db servers ke /var/lib/mysql directory info check
```

```
ansible all -m stat -a "path=/etc/hosts"

# /etc/hosts file existence aur permissions check
```

```
ansible all -m stat -a "path=/opt"

# /opt directory ke details check karta hai
```

```
ansible all -m stat -a "path=/boot/vmlinuz"

# boot kernel file ke details verify
```

```
ansible all -m stat -a "path=/usr/bin/python3"

# python3 executable ke existence aur permissions check
```

```
ansible all -m stat -a "path=/etc/resolv.conf"

# DNS config file ke details verify karta hai
```

🔥 13️⃣ get_url (download)

```
ansible all -m get_url -a "url=https://example.com/a.tar.gz dest=/tmp/a.tar.gz"

# sab hosts par a.tar.gz file download karke /tmp/ me save karta hai
```

```
ansible web -m get_url -a "url=https://example.com/index.html dest=/var/www/html/index.html"

# web servers par website index.html download karke /var/www/html/ me save
```

```
ansible all -m get_url -a "url=https://example.com/file.txt dest=/tmp/file.txt"

# file.txt download karke /tmp/ me save karta hai
```

```
ansible db -m get_url -a "url=https://example.com/db.sql dest=/tmp/db.sql"

# db servers par db.sql download karke /tmp/ me save
```

```
ansible all -m get_url -a "url=https://example.com/app.zip dest=/opt/app.zip"

# app.zip download karke /opt/ me save karta hai
```

```
ansible all -m get_url -a "url=https://example.com/test.sh dest=/usr/local/bin/test.sh mode=755"

# test.sh download karke /usr/local/bin/ me save aur execute permission set
```

```
ansible web -m get_url -a "url=https://example.com/logo.png dest=/tmp/logo.png"

# web servers par logo.png download karke /tmp/ me save
```

```
ansible all -m get_url -a "url=https://example.com/data.json dest=/tmp/data.json"

# data.json download karke /tmp/ me save
```

```
ansible all -m get_url -a "url=https://example.com/conf.cfg dest=/etc/conf.cfg"

# configuration file download karke /etc/ me save
```

```
ansible db -m get_url -a "url=https://example.com/backup.tar dest=/backup/backup.tar"

# db servers par backup.tar download karke /backup/ me save
```

🔥 1 4️⃣ unarchive

```
ansible all -m unarchive -a "src=/tmp/a.tar.gz dest=/opt remote_src=yes"

# sab hosts par /tmp/a.tar.gz extract karke /opt/ me rakhta hai
```

```
ansible web -m unarchive -a "src=/tmp/app.tar dest=/var/www remote_src=yes"

# web servers par /tmp/app.tar extract karke /var/www/ me rakhta hai
```

```
ansible all -m unarchive -a "src=/opt/app.zip dest=/opt remote_src=yes"

# sab hosts par /opt/app.zip extract karke /opt/ me rakhta hai
```

```
ansible db -m unarchive -a "src=/backup/db.tar dest=/backup remote_src=yes"

# db servers par /backup/db.tar extract karke /backup/ me rakhta hai
```

```
ansible all -m unarchive -a "src=/tmp/test.tar dest=/tmp remote_src=yes"
```

```
# sab hosts par /tmp/test.tar extract karke /tmp/ me rakhta hai
```

```
ansible web -m unarchive -a "src=/tmp/site.tar dest=/var/www/html remote_src=yes"
# web servers par /tmp/site.tar extract karke /var/www/html/ me rakhta hai
```

```
ansible all -m unarchive -a "src=/tmp/logs.tar dest=/var/log remote_src=yes"
# sab hosts par /tmp/logs.tar extract karke /var/log/ me rakhta hai
```

```
ansible all -m unarchive -a "src=/tmp/data.tar dest=/data remote_src=yes"
# sab hosts par /tmp/data.tar extract karke /data/ me rakhta hai
```

```
ansible db -m unarchive -a "src=/tmp/mysql.tar dest=/var/lib remote_src=yes"
# db servers par /tmp/mysql.tar extract karke /var/lib/ me rakhta hai
```

```
ansible all -m unarchive -a "src=/tmp/soft.tar dest=/opt remote_src=yes"
# sab hosts par /tmp/soft.tar extract karke /opt/ me rakhta hai
```

🔥 15️⃣ mount

```
ansible all -m mount -a "path=/data src=/dev/sdb1 fstype=ext4 state=mounted"
# /dev/sdb1 ko /data par mount karta hai ext4 filesystem ke saath
```

```
ansible all -m mount -a "path=/backup src=/dev/sdc1 fstype=xfs state=mounted"
# /dev/sdc1 ko /backup par mount karta hai xfs filesystem ke saath
```

```
ansible web -m mount -a "path=/web src=/dev/sdd1 fstype=ext4 state=mounted"
# web servers par /dev/sdd1 ko /web par mount karta hai
```

```
ansible all -m mount -a "path=/data state=unmounted"
# /data ka mount hata deta hai (unmount)
```

```
ansible db -m mount -a "path=/mysql src=/dev/sde1 fstype=xfs state=mounted"
```

```
# db servers par /dev/sde1 ko /mysql par mount karta hai xfs filesystem ke saath
```

```
ansible all -m mount -a "path=/iso src=/dev/cdrom fstype=iso9660 state=mounted"
# /dev/cdrom ko /iso par mount karta hai iso9660 filesystem ke saath
```

```
ansible all -m mount -a "path=/nfs src=10.0.0.1:/data fstype=nfs state=mounted"
# remote NFS share 10.0.0.1:/data ko /nfs par mount karta hai
```

```
ansible web -m mount -a "path=/test state=absent"
# /test directory ke mount ko remove karta hai (absent)
```

```
ansible all -m mount -a "path=/backup state=remounted"
# /backup ko remount karta hai (already mounted hai to update options)
```

```
ansible all -m mount -a "path=/tmp state=unmounted"
# /tmp ko unmount karta hai
```

🔥 1 6️⃣ lineinfile

```
ansible all -m lineinfile -a "path=/etc/ssh/sshd_config line='PermitRootLogin no'"
# sshd_config me root login disable karta hai
```

```
ansible all -m lineinfile -a "path=/etc/sysctl.conf line='net.ipv4.ip_forward=1'"
# sysctl.conf me IP forwarding enable karta hai
```

```
ansible web -m lineinfile -a "path=/etc/httpd/conf/httpd.conf line='Listen 8080'"
# apache httpd ko port 8080 sunne ke liye configure karta hai
```

```
ansible all -m lineinfile -a "path=/etc/hosts line='192.168.1.10 test'"
# /etc/hosts me host entry add karta hai
```

```
ansible db -m lineinfile -a "path=/etc/my.cnf line='max_connections=500'"
```

```
# MySQL config me max connections set karta hai
```

```
ansible all -m lineinfile -a "path=/etc/resolv.conf line='nameserver 8.8.8.8'"

# DNS server add karta hai resolv.conf me
```

```
ansible all -m lineinfile -a "path=/etc/bashrc line='alias ll=ls -l'"

# bash me alias ll define karta hai
```

```
ansible web -m lineinfile -a "path=/etc/httpd/conf/httpd.conf line='ServerTokens Prod'"

# apache server info hide karne ke liye set karta hai
```

```
ansible all -m lineinfile -a "path=/etc/ssh/sshd_config line='PasswordAuthentication no'"

# ssh password authentication disable karta hai
```

```
ansible all -m lineinfile -a "path=/etc/sysctl.conf line='vm.swappiness=10'"

# sysctl me swappiness value set karta hai
```

🔥 1 7️⃣ replace

```
ansible all -m replace -a "path=/etc/ssh/sshd_config regexp='yes' replace='no'"

# sshd_config me "yes" ko "no" se replace karke root login disable karta hai
```

```
ansible web -m replace -a "path=/etc/httpd/conf/httpd.conf regexp='80' replace='8080'"

# apache httpd config me port 80 ko 8080 se replace karta hai
```

```
ansible all -m replace -a "path=/etc/sysctl.conf regexp='0' replace='1'"

# sysctl.conf me value 0 ko 1 se replace karta hai
```

```
ansible db -m replace -a "path=/etc/my.cnf regexp='100' replace='500'"

# MySQL config me max_connections ya koi value 100 ko 500 se replace karta hai
```

```
ansible all -m replace -a "path=/etc/hosts regexp='localhost' replace='local'"

# /etc/hosts me localhost ko local se replace karta hai
```

```
ansible all -m replace -a "path=/etc/resolv.conf regexp='8.8.4.4' replace='1.1.1.1'"

# DNS server IP 8.8.4.4 ko 1.1.1.1 se replace karta hai
```

```
ansible web -m replace -a "path=/var/www/html/index.html regexp='old' replace='new'"

# index.html me text "old" ko "new" se replace karta hai
```

```
ansible all -m replace -a "path=/etc/fstab regexp='defaults' replace='noatime'"

# fstab me "defaults" ko "noatime" se replace karta hai (mount option change)
```

```
ansible all -m replace -a "path=/etc/bashrc regexp='vi' replace='vim'"

# bashrc me editor default vi ko vim se replace karta hai
```

```
ansible db -m replace -a "path=/etc/my.cnf regexp='OFF' replace='ON'"

# MySQL config me kisi option ko OFF se ON me replace karta hai
```

🔥 18️⃣ blockinfile

```
ansible all -m blockinfile -a "path=/etc/sysctl.conf block='net.ipv4.ip_forward=1'"

# sysctl.conf me ek block add karke IP forwarding enable karta hai
```

```
ansible all -m blockinfile -a "path=/etc/hosts block='10.0.0.10 app'"

# /etc/hosts me ek naya host entry add karta hai
```

```
ansible web -m blockinfile -a "path=/etc/httpd/conf/httpd.conf block='ServerSignature Off'"

# apache config me ServerSignature Off ka block add karta hai
```

```
ansible all -m blockinfile -a "path=/etc/bashrc block='export JAVA_HOME=/opt/java'"
```

```
# bashrc me JAVA_HOME ka export statement add karta hai

ansible db -m blockinfile -a "path=/etc/my.cnf block='max_connections=500'"

# MySQL config me max_connections ke liye block add karta hai

ansible all -m blockinfile -a "path=/etc/profile block='ulimit -n 65535'"

# /etc/profile me file descriptor limit set karne ka block add karta hai

ansible web -m blockinfile -a "path=/etc/httpd/conf/httpd.conf block='Timeout 60'"

# apache config me Timeout 60 ka block add karta hai

ansible all -m blockinfile -a "path=/etc/sysctl.conf block='vm.swappiness=10'"

# sysctl.conf me swappiness value 10 set karne ka block add karta hai

ansible all -m blockinfile -a "path=/etc/ssh/sshd_config block='AllowUsers admin'"

# sshd_config me AllowUsers admin ka block add karta hai

ansible db -m blockinfile -a "path=/etc/security/limits.conf block='* soft nofile 65535'"

# limits.conf me max open files ke liye block add karta hai
```

🔥 19️⃣ hostname

```
ansible all -m hostname -a "name=server1"

# sabhi hosts ka hostname server1 set karta hai

ansible web -m hostname -a "name=web01"

# web group ke hosts ka hostname web01 set karta hai

ansible db -m hostname -a "name=db01"

# db group ke hosts ka hostname db01 set karta hai

ansible all -m hostname -a "name=node1.example.com"
```

```
# sabhi hosts ka FQDN hostname set karta hai


ansible all -m hostname -a "name=prod-server"

# sabhi hosts ka hostname prod-server set karta hai


ansible web -m hostname -a "name=web-prod"

# web group ke hosts ka hostname web-prod set karta hai


ansible db -m hostname -a "name=db-prod"

# db group ke hosts ka hostname db-prod set karta hai


ansible all -m hostname -a "name=test-node"

# sabhi hosts ka hostname test-node set karta hai


ansible all -m hostname -a "name=backup-server"

# sabhi hosts ka hostname backup-server set karta hai


ansible all -m hostname -a "name=ansible-node"

# sabhi hosts ka hostname ansible-node set karta hai
```

🔥 2 0️⃣ firewalld

```
ansible all -m firewalld -a "port=80/tcp state=enabled permanent=yes"

# sab hosts par TCP port 80 ko permanently open karta hai


ansible all -m firewalld -a "service=http state=enabled"

# http service firewall me enable karta hai


ansible web -m firewalld -a "port=443/tcp state=enabled"

# web group ke hosts par HTTPS port 443 open karta hai


ansible all -m firewalld -a "port=22/tcp state=enabled"
```

```
# SSH port 22 open karta hai
```

```
ansible db -m firewalld -a "port=3306/tcp state=enabled"

# db servers ke MySQL port 3306 open karta hai
```

```
ansible all -m firewalld -a "service=ssh state=enabled"

# ssh service firewall me enable karta hai
```

```
ansible all -m firewalld -a "port=8080/tcp state=enabled"

# TCP port 8080 open karta hai
```

```
ansible web -m firewalld -a "service=https state=enabled"

# web servers me HTTPS service enable karta hai
```

```
ansible all -m firewalld -a "port=21/tcp state=disabled"

# FTP port 21 ko block/disable karta hai
```

```
ansible all -m firewalld -a "immediate=yes"

# changes ko turant apply karta hai without waiting for reload
```

🔥 21️⃣ selinux

```
ansible all -m selinux -a "state=disabled"

# sab hosts par SELinux ko disable karta hai
```

```
ansible all -m selinux -a "state=enforcing"

# SELinux ko enforcing mode me set karta hai (active and enforcing rules)
```

```
ansible web -m selinux -a "state=permissive"

# web servers par SELinux ko permissive mode me set karta hai (warnings only)
```

```
ansible db -m selinux -a "state=disabled"
```

```
# db servers par SELinux disable karta hai
```

```
ansible all -m selinux -a "policy=targeted"
# SELinux policy ko targeted set karta hai (default most systems)
```

```
ansible all -m selinux -a "state=enforcing"
# sab hosts par SELinux ko enforcing mode me set karta hai
```

```
ansible web -m selinux -a "state=disabled"
# web servers par SELinux disable karta hai
```

```
ansible all -m selinux -a "state=permissive"
# sab hosts par SELinux ko permissive mode me set karta hai
```

```
ansible db -m selinux -a "state=enforcing"
# db servers par SELinux enforcing mode me set karta hai
```

```
ansible all -m selinux -a "policy=mls"
# SELinux policy ko MLS (Multi-Level Security) set karta hai
```

🔥 22️⃣ wait_for

```
ansible all -m wait_for -a "port=22 timeout=30"
# sab hosts par SSH port 22 ke available hone tak max 30 seconds wait karta hai
```

```
ansible web -m wait_for -a "port=80 timeout=60"
# web servers par HTTP port 80 ke open hone tak 60 seconds wait
```

```
ansible db -m wait_for -a "port=3306 timeout=60"
# db servers par MySQL port 3306 ke available hone tak wait
```

```
ansible all -m wait_for -a "path=/tmp/test timeout=20"
```

```
# /tmp/test file exist hone tak 20 seconds wait karta hai

ansible all -m wait_for -a "host=127.0.0.1 port=22"

# local host (127.0.0.1) par SSH port 22 ke available hone tak wait

ansible web -m wait_for -a "port=443 timeout=30"

# web servers par HTTPS port 443 ke available hone tak 30 seconds wait

ansible all -m wait_for -a "port=8080 timeout=60"

# sab hosts par TCP port 8080 ke open hone tak wait

ansible db -m wait_for -a "port=5432 timeout=60"

# db servers par PostgreSQL port 5432 ke available hone tak wait

ansible all -m wait_for -a "path=/var/run/sshd.pid"

# sshd process pid file exist hone tak wait karta hai

ansible all -m wait_for -a "timeout=10"

# sirf 10 seconds tak wait karta hai (general timeout)
```

🔥 2 3️⃣ assert

```
ansible all -m assert -a "that='ansible_os_family==\"RedHat\"'"

# check karta hai ki host ka OS family RedHat hai

ansible all -m assert -a "that='ansible_memtotal_mb > 1024'"

# verify karta hai ki host ke paas 1GB se zyada RAM hai

ansible web -m assert -a "that='ansible_processor_vcpus >= 2'"

# web servers me CPU vCPUs 2 ya zyada hone chahiye

ansible all -m assert -a "that='ansible_hostname is defined'"
```

```
# verify karta hai ki hostname variable defined hai
```

```
ansible db -m assert -a "that='ansible_architecture==\"x86_64\"'"
# db servers x86_64 architecture me hai ya nahi check
```

```
ansible all -m assert -a "that='ansible_distribution_major_version >= 7'"
# OS major version 7 ya zyada hai ya nahi check
```

```
ansible web -m assert -a "that='ansible_default_ipv4 is defined'"
# web servers me default IPv4 defined hai ya nahi
```

```
ansible all -m assert -a "that='ansible_kernel is defined'"
# verify karta hai ki kernel information available hai
```

```
ansible db -m assert -a "that='ansible_all_ipv4_addresses is defined'"
# db servers ke all IPv4 addresses defined hai ya nahi check
```

```
ansible all -m assert -a "that='ansible_date_time is defined'"
# verify karta hai ki date/time facts available hai
```

🔥 2 4️⃣ setup (facts gather)

```
ansible all -m setup
# sab hosts ke facts gather karta hai (OS, memory, network, CPU, etc.)
```

```
ansible all -m setup -a "filter=ansible_os_family"
# sirf OS family fact gather karta hai
```

```
ansible all -m setup -a "filter=ansible_memtotal_mb"
# sirf total memory (MB) ka fact gather karta hai
```

```
ansible all -m setup -a "filter=ansible_hostname"
```

```
# sirf hostname fact gather karta hai

ansible all -m setup -a "filter=ansible_distribution"
# OS distribution name gather karta hai

ansible all -m setup -a "filter=ansible_kernel"
# kernel version gather karta hai

ansible web -m setup
# web group ke hosts ke sab facts gather karta hai

ansible db -m setup
# db group ke hosts ke sab facts gather karta hai

ansible all -m setup -a "filter=ansible_default_ipv4"
# default IPv4 address fact gather karta hai

ansible all -m setup -a "filter=ansible_processor_vcpus"
# CPU vCPUs count gather karta hai
```

🔥 25️⃣ debug (output print)

```
ansible all -m debug -a "msg='Hello Ansible'"
# sab hosts par message print karta hai: Hello Ansible

ansible all -m debug -a "var=ansible_hostname"
# sab hosts ka hostname print karta hai

ansible all -m debug -a "var=ansible_os_family"
# sab hosts ka OS family print karta hai

ansible all -m debug -a "var=ansible_memtotal_mb"
```

```
# sab hosts ka total memory print karta hai


ansible web -m debug -a "msg='Web server check'"

# web servers par message print karta hai


ansible db -m debug -a "msg='DB server check'"

# db servers par message print karta hai


ansible all -m debug -a "var=inventory_hostname"

# inventory me defined hostname print karta hai


ansible all -m debug -a "var=ansible_kernel"

# sab hosts ka kernel version print karta hai


ansible all -m debug -a "var=ansible_architecture"

# sab hosts ka architecture print karta hai


ansible all -m debug -a "msg='Done'"

# sab hosts par Done message print karta hai
```

🔥 2 6️⃣ raw (python bina command)

```
ansible all -m raw -a "yum install -y python3"

# sab hosts par yum se python3 install karta hai (RHEL/CentOS)


ansible all -m raw -a "apt install -y python3"

# sab hosts par apt se python3 install karta hai (Debian/Ubuntu)


ansible all -m raw -a "uptime"

# sab hosts par uptime command run karta hai


ansible all -m raw -a "df -h"
```

```
# sab hosts par disk usage human readable format me check karta hai


ansible web -m raw -a "hostname"

# web servers par hostname print karta hai


ansible db -m raw -a "free -m"

# db servers par memory usage MB me show karta hai


ansible all -m raw -a "mkdir /tmp/rawtest"

# sab hosts par /tmp/rawtest directory create karta hai


ansible all -m raw -a "rm -rf /tmp/rawtest"

# sab hosts par /tmp/rawtest directory delete karta hai


ansible all -m raw -a "whoami"

# sab hosts par currently logged in user print karta hai


ansible all -m raw -a "uname -r"

# sab hosts par kernel version print karta hai
```

🔥 27️⃣ statx (advanced file info – new systems)

```
ansible all -m statx -a "path=/etc/passwd"

# /etc/passwd file ke detailed metadata check karta hai


ansible all -m statx -a "path=/etc/shadow"

# /etc/shadow file ke metadata check karta hai


ansible web -m statx -a "path=/var/www/html"

# web servers ke /var/www/html directory ke metadata gather karta hai


ansible db -m statx -a "path=/var/lib/mysql"
```

```
# db servers ke MySQL data directory ke metadata gather karta hai
```

```
ansible all -m statx -a "path=/tmp"

# /tmp directory ke metadata check karta hai
```

```
ansible all -m statx -a "path=/opt"

# /opt directory ke metadata check karta hai
```

```
ansible all -m statx -a "path=/etc/hosts"

# /etc/hosts file ke metadata gather karta hai
```

```
ansible all -m statx -a "path=/usr/bin/python3"

# python3 binary ke metadata check karta hai
```

```
ansible web -m statx -a "path=/etc/httpd"

# web servers me apache config directory ke metadata gather karta hai
```

```
ansible db -m statx -a "path=/etc/my.cnf"

# db servers me MySQL config file ke metadata gather karta hai
```

🔥 28️⃣ find (file search)

```
ansible all -m find -a "paths=/tmp"

# /tmp directory ke andar ke sab files list karta hai
```

```
ansible all -m find -a "paths=/var/log patterns=*.log"

# /var/log me sirf .log files list karta hai
```

```
ansible web -m find -a "paths=/var/www/html"

# web servers me /var/www/html directory ke files list karta hai
```

```
ansible all -m find -a "paths=/etc patterns=*.conf"
```

```
# /etc me sirf .conf files search karta hai

ansible db -m find -a "paths=/var/lib/mysql"
# db servers me MySQL data directory ke files list karta hai

ansible all -m find -a "paths=/root"
# /root directory ke files list karta hai

ansible all -m find -a "paths=/tmp age=1d"
# /tmp me 1 din se purani files find karta hai

ansible all -m find -a "paths=/opt file_type=directory"
# /opt me sirf directories list karta hai

ansible web -m find -a "paths=/var/log age=7d"
# web servers ke /var/log me 7 din se purani files list karta hai

ansible all -m find -a "paths=/etc recurse=yes"
# /etc directory me recursively sab files list karta hai
```

🔥 29️⃣ archive (compress)

```
ansible all -m archive -a "path=/etc dest=/tmp/etc.tar.gz"
# /etc directory ko /tmp/etc.tar.gz me compress karta hai

ansible all -m archive -a "path=/var/log dest=/tmp/logs.tar"
# /var/log directory ko /tmp/logs.tar me compress karta hai

ansible web -m archive -a "path=/var/www dest=/tmp/web.tar.gz"
# web servers ke /var/www ko /tmp/web.tar.gz me compress

ansible db -m archive -a "path=/var/lib/mysql dest=/tmp/db.tar.gz"
```

```
# db servers ke /var/lib/mysql ko /tmp/db.tar.gz me compress

ansible all -m archive -a "path=/opt dest=/tmp/opt.tar.gz"
# /opt directory ko /tmp/opt.tar.gz me compress

ansible all -m archive -a "path=/tmp/test dest=/tmp/test.tar"
# /tmp/test directory/file ko /tmp/test.tar me compress

ansible all -m archive -a "path=/root dest=/tmp/root.tar"
# /root directory ko /tmp/root.tar me compress

ansible web -m archive -a "path=/etc/httpd dest=/tmp/httpd.tar"
# web servers ke /etc/httpd ko /tmp/httpd.tar me compress

ansible all -m archive -a "path=/etc/ssh dest=/tmp/ssh.tar"
# /etc/ssh directory ko /tmp/ssh.tar me compress

ansible db -m archive -a "path=/backup dest=/tmp/backup.tar"
# db servers ke /backup directory ko /tmp/backup.tar me compress
```

🔥 30️⃣ synchronize (rsync)

```
ansible all -m synchronize -a "src=/data dest=/backup"
# sab hosts ke /data directory ko /backup me sync karta hai

ansible web -m synchronize -a "src=/var/www dest=/backup/web"
# web servers ke /var/www ko /backup/web me sync karta hai

ansible all -m synchronize -a "src=/etc dest=/backup/etc"
# sab hosts ke /etc directory ko /backup/etc me sync karta hai

ansible db -m synchronize -a "src=/var/lib/mysql dest=/backup/mysql"
```

```
# db servers ke /var/lib/mysql ko /backup/mysql me sync karta hai
```

```
ansible all -m synchronize -a "src=/opt dest=/backup/opt"
# /opt directory ko /backup/opt me sync karta hai
```

```
ansible all -m synchronize -a "src=/tmp dest=/backup/tmp"
# /tmp directory ko /backup/tmp me sync karta hai
```

```
ansible web -m synchronize -a "src=/var/log dest=/backup/logs"
# web servers ke /var/log ko /backup/logs me sync karta hai
```

```
ansible all -m synchronize -a "src=/home dest=/backup/home"
# /home directory ko /backup/home me sync karta hai
```

```
ansible db -m synchronize -a "src=/backup dest=/remote"
# db servers ke /backup ko /remote directory me sync karta hai
```

```
ansible all -m synchronize -a "src=/root dest=/backup/root"
# /root directory ko /backup/root me sync karta hai
```

🔥 31️⃣ timezone

```
ansible all -m timezone -a "name=Asia/Kolkata"
# sab hosts ka timezone Asia/Kolkata set karta hai
```

```
ansible all -m timezone -a "name=UTC"
# sab hosts ka timezone UTC set karta hai
```

```
ansible web -m timezone -a "name=Asia/Dubai"
# web servers ka timezone Asia/Dubai set karta hai
```

```
ansible db -m timezone -a "name=Europe/London"
```

```
# db servers ka timezone Europe/London set karta hai


ansible all -m timezone -a "name=America/New_York"

# sab hosts ka timezone America/New_York set karta hai


ansible all -m timezone -a "name=Asia/Tokyo"

# sab hosts ka timezone Asia/Tokyo set karta hai


ansible web -m timezone -a "name=Asia/Singapore"

# web servers ka timezone Asia/Singapore set karta hai


ansible all -m timezone -a "name=Asia/Kolkata"

# sab hosts ka timezone Asia/Kolkata set karta hai


ansible db -m timezone -a "name=UTC"

# db servers ka timezone UTC set karta hai


ansible all -m timezone -a "name=Asia/Calcutta"

# sab hosts ka timezone Asia/Calcutta set karta hai
```

🔥 32️⃣ at (one-time job)

```
ansible all -m at -a "command='reboot' count=1 units=minutes"

# sab hosts par 1 minute me ek baar reboot schedule karta hai


ansible all -m at -a "command='uptime > /tmp/up.txt' count=5 units=minutes"

# sab hosts par 5 minute interval me uptime output /tmp/up.txt me save karta hai


ansible web -m at -a "command='service httpd restart' count=1 units=hours"

# web servers par 1 hour me ek baar httpd service restart schedule


ansible all -m at -a "command='df -h > /tmp/df.txt' count=10 units=minutes"
```

```
# sab hosts par 10 minute interval me disk usage output /tmp/df.txt me save


ansible db -m at -a "command='systemctl restart mariadb' count=1 units=hours"

# db servers par 1 hour me mariadb service restart schedule


ansible all -m at -a "command='rm -rf /tmp/*' count=30 units=minutes"

# sab hosts par 30 minute interval me /tmp/* delete schedule


ansible web -m at -a "command='date >> /tmp/date.txt' count=2 units=minutes"

# web servers par 2 minute interval me date output /tmp/date.txt append


ansible all -m at -a "command='shutdown -r now' count=1 units=minutes"

# sab hosts par 1 minute me system reboot schedule


ansible all -m at -a "command='free -m' count=3 units=minutes"

# sab hosts par 3 minute interval me memory usage check


ansible db -m at -a "command='echo done' count=1 units=minutes"

# db servers par 1 minute me "done" message execute schedule
```

🔥 3 3️⃣ pause (wait / delay)

```
ansible all -m pause -a "seconds=10"

# sab hosts par 10 seconds ke liye pause karta hai


ansible all -m pause -a "minutes=1"

# sab hosts par 1 minute ke liye pause karta hai


ansible web -m pause -a "seconds=30"

# web servers par 30 seconds ke liye pause


ansible db -m pause -a "minutes=2"
```

```
# db servers par 2 minute ke liye pause


ansible all -m pause -a "prompt='Press Enter to continue'"

# sab hosts par user prompt display karta hai aur enter press hone tak wait karta hai


ansible all -m pause -a "seconds=5"

# sab hosts par 5 seconds ke liye pause


ansible web -m pause -a "minutes=1"

# web servers par 1 minute ke liye pause


ansible all -m pause -a "prompt='Maintenance check'"

# sab hosts par user prompt display karta hai: Maintenance check


ansible db -m pause -a "seconds=15"

# db servers par 15 seconds ke liye pause


ansible all -m pause -a "seconds=20"

# sab hosts par 20 seconds ke liye pause
=================================================================
```

🔥 1️⃣ ping

➡️ Server reachable hai ya nahi, Ansible connect ho paa raha hai ya nahi
➡️ Python available hai ya nahi ye bhi check karta hai

Simple bol:

“Server zinda hai aur Ansible ka connection sahi hai ya nahi”

🔥 2️⃣ command

➡️ Remote server par simple Linux command run karta hai
➡️ Pipe |, redirect > ka support nahi

Simple bol:

“Basic command chalane ke liye, bina shell features”

🔥 3️⃣ shell

➡️ Shell ke saare features support karta hai

➡️ Pipe, redirect, variable, tee sab allowed

Simple bol:

“Jab command complex ho, pipe/redirect use karna ho”

🔥 4️⃣ copy

➡️ Ansible controller se remote server par file bhejna

➡️ Content ke saath file banana bhi possible

Simple bol:

“Local machine se server par file copy karna”

🔥 5️⃣ file

➡️ File ya directory banana, delete karna

➡️ Permission, owner, group set karna

Simple bol:

“File aur directory ka structure manage karna”

🔥 6️⃣ yum / dnf

➡️ RHEL/CentOS/Alma/Rocky me package install / remove

➡️ Latest version ya specific state maintain

Simple bol:

“Linux software install ya uninstall karna”

🔥 7️⃣ service / systemd

➡️ Service start, stop, restart

➡️ Boot par enable/disable

Simple bol:

“Server ke services ko control karna”

🔥 8️⃣ user

➡️ User banana, delete karna
➡️ UID, group, shell, password set karna

Simple bol:

“Linux user management automate karna”

🔥 9️⃣ cron

➡️ Scheduled job create/remove
➡️ Daily, hourly, reboot par chalne wale kaam

Simple bol:

“Automatic time-based job lagana”

🔥 🔟 reboot

➡️ Remote server ko reboot karna
➡️ Wapas aane tak wait karna

Simple bol:

“Server ko safe tarike se restart karna”

🔥 1 1️⃣ fetch

➡️ Remote server se file utha kar controller par lana

Simple bol:

“Server se apni machine par file lana”

🔥 1 2️⃣ stat

➡️ File exist karti hai ya nahi
➡️ Size, permission, owner check

Simple bol:

“File ki detail jaanchna”

🔥 1 3️⃣ get_url

➡️ Internet ya internal URL se file download

Simple bol:

“URL se file download karna”

🔥 1 4️⃣ unarchive

➡️ tar / zip file extract karna

Simple bol:

“Compressed file ko open karna”

🔥 1 5️⃣ mount

➡️ Disk, NFS, ISO mount/unmount
➡️ Temporary ya permanent mount

Simple bol:

“Storage ko system me jodna”

🔥 1 6️⃣ lineinfile

➡️ File me ek specific line add/modify
➡️ Duplicate line se bachata hai

Simple bol:

“Config file me ek line ensure karna”

🔥 1 7️⃣ replace

➡️ Regex ke through text replace
➡️ Existing content badalne ke liye

Simple bol:

“File ke andar text badalna”

🔥 1 8️⃣ blockinfile

➡️ Multiple lines ka ek block add/remove
➡️ Clearly marked section

Simple bol:

“Config file me pura block add karna”

🔥 19️⃣ hostname

➡️ Server ka hostname set karna

Simple bol:

“Machine ka naam change karna”

🔥 20️⃣ firewalld

➡️ Firewall port/service open/close
➡️ Permanent rule lagana

Simple bol:

“Firewall rule manage karna”

🔥 21️⃣ selinux

➡️ SELinux enforcing/permissive/disable
➡️ Policy set karna

Simple bol:

“SELinux ka mode control karna”

🔥 22️⃣ wait_for

➡️ Port, file, ya service ka wait
➡️ Automation me timing control

Simple bol:

“Kisi cheez ke ready hone ka wait”

🔥 23️⃣ assert

➡️ Condition check
➡️ Galat condition par fail

Simple bol:

“Check lagana – sahi hai ya nahi”

🔥 2 4️⃣ setup

➡️ Server ke saare facts gather
➡️ OS, memory, CPU, IP

Simple bol:

“Server ki poori jankari nikalna”

🔥 2 5️⃣ debug

➡️ Variable ya message print
➡️ Troubleshooting me help

Simple bol:

“Output dekhne ke liye”

🔥 2 6️⃣ raw

➡️ Python ke bina command run
➡️ New server bootstrap

Simple bol:

“Direct SSH jaise command chalana”

🔥 2 7️⃣ statx

➡️ Advanced file info (new kernels)

Simple bol:

“Stat ka advanced version”

🔥 2 8️⃣ find

➡️ File/directory search
➡️ Age, size, pattern ke base par

Simple bol:

“Server par file dhoondhna”

🔥 2 9️⃣ archive

➡️ File/directory ko tar/zip banana

Simple bol:

“Backup ke liye compress karna”

🔥 3 0️⃣ synchronize

➡️ Rsync based fast copy
➡️ Backup / mirror

Simple bol:

“Fast file sync karna”

🔥 3 1️⃣ timezone

➡️ Server ka time zone set

Simple bol:

“Server ka time set karna”

🔥 3 2️⃣ at

➡️ One-time scheduled command

Simple bol:

“Ek baar chalne wala kaam set karna”

🔥 3 3️⃣ pause

➡️ Automation me delay
➡️ Manual confirmation

Simple bol:

“Automation ko thodi der roakna”

🔥 parted module – Short Description

➡️ Linux me **disk partition create / delete / resize / manage** karne ke liye use hota hai
➡️ fdisk / parted command ka **automation version** hai
➡️ **New disks (LVM, filesystem se pehle)** use hota hai

🧠 **Simple bol (Interview line):**

"Ansible parted module ka use disk ke partitions automate karne ke liye hota hai, specially new disks ke case me."

⚠️ Important Note (Production Tip)

- Disk **mounted na ho**
- Disk par **filesystem na bana ho**
- Galat disk select kiya to **data loss** 💥

🔥 parted – 10 AD-HOC Examples

1️⃣ Disk par GPT label banana

```
ansible all -m parted -a "device=/dev/sdb label=gpt"
```

2️⃣ MBR (msdos) partition table banana

```
ansible all -m parted -a "device=/dev/sdc label=msdos"
```

3️⃣ 1GB ka primary partition banana

```
ansible all -m parted -a "device=/dev/sdb number=1 state=present part_start=1MiB part_end=1GiB"
```

4️⃣ Pura disk ek hi partition me dena

```
ansible all -m parted -a "device=/dev/sdb number=1 state=present part_start=1MiB part_end=100%"
```

5️⃣ LVM ke liye partition banana

```
ansible all -m parted -a "device=/dev/sdb number=2 state=present part_start=1GiB part_end=5GiB flags=lvm"
```

6️⃣ Boot flag set karna

```
ansible all -m parted -a "device=/dev/sda number=1 flags=boot state=present"
```

7️⃣ Existing partition delete karna

```
ansible all -m parted -a "device=/dev/sdb number=2 state=absent"
```

8️⃣ Partition ko resize karna

```
ansible all -m parted -a "device=/dev/sdb number=1 state=present part_end=10GiB"
```

9️⃣ Partition information dekhna

```
ansible all -m parted -a "device=/dev/sdb"
```

🔟 Multiple flags set karna (LVM + RAID)

```
ansible all -m parted -a "device=/dev/sdb number=3 flags=lvm,raid state=present part_start=5GiB part_end=8GiB"
```

---

11 - Partition Delete – parted Module (AD-HOC)

```
ansible all -m parted -a "device=/dev/sdb number=1 state=absent"
```

🧠 Real Production Flow (Yaad rakhne ke liye)

New Disk → parted → pvcreate → vgcreate → lvcreate → mkfs → mount

---

✅ Summary (One-glance)

Point	Meaning
Module	parted
Use	Disk partition manage
Works before	filesystem & LVM
Risk	High (wrong disk = data loss)

Point	Meaning
Used in	Storage, LVM, SAN/NAS

---

filesystem – 10 AD-HOC Examples-----------

1️⃣ ext4 filesystem banana (partition par)

```
ansible all -m filesystem -a "fstype=ext4 dev=/dev/sdb1"
```

2️⃣ xfs filesystem banana

```
ansible all -m filesystem -a "fstype=xfs dev=/dev/sdb2"
```

3️⃣ LVM logical volume par ext4 banana

```
ansible all -m filesystem -a "fstype=ext4 dev=/dev/vgdata/lvdata"
```

4️⃣ LVM logical volume par xfs banana

```
ansible all -m filesystem -a "fstype=xfs dev=/dev/vgdata/lvbackup"
```

5️⃣ Filesystem resize karna (ext4)

```
ansible all -m filesystem -a "fstype=ext4 dev=/dev/vgdata/lvdata resizefs=yes"
```

6️⃣ Filesystem resize karna (xfs – grow only)

```
ansible all -m filesystem -a "fstype=xfs dev=/dev/vgdata/lvdata resizefs=yes"
```

7️⃣ Disk par directly filesystem banana (non-partition)

```
ansible all -m filesystem -a "fstype=ext4 dev=/dev/sdc"
```

8️⃣ Force karke filesystem banana (⚠️ dangerous)

```
ansible all -m filesystem -a "fstype=ext4 dev=/dev/sdb1 force=yes"
```

9️⃣ Filesystem type change karna (fresh device)

```
ansible all -m filesystem -a "fstype=xfs dev=/dev/sdd1"
```

🔟 Filesystem UUID ke sath banana

```
ansible all -m filesystem -a "fstype=ext4 dev=/dev/sde1 opts='-U random'"
```

🔥 lvol Module – Logical Volume

📌 Short Description

➡️ Linux me **LVM Logical Volume (LV)** create, resize, delete karne ke liye
➡️ lvcreate, lvextend, lvreduce ka automation version
➡️ **Volume Group (VG) ke andar space manage karta hai**

🧠 Interview One-liner:

"Ansible lvol module ka use Volume Group ke andar logical volume create aur resize karne ke liye hota hai."

⚠️ Production Warnings 🚨

- Shrink se pehle **filesystem shrink zaroori**
- XFS filesystem **shrink supported nahi**
- Galat size = data loss

🔥 lvol – 10 AD-HOC Examples

1️⃣ Logical Volume banana (5GB)

```
ansible all -m lvol -a "vg=vgdata lv=lvapp size=5g"
```

2️⃣ Logical Volume banana (100% FREE space)

```
ansible all -m lvol -a "vg=vgdata lv=lvbackup size=100%FREE"
```

3️⃣ Thin Logical Volume banana

```
ansible all -m lvol -a "vg=vgdata lv=lvthin size=10g thinpool=yes"
```

4️⃣ Existing LV ka size increase karna

```
ansible all -m lvol -a "vg=vgdata lv=lvapp size=+2g"
```

5️⃣ Existing LV ka size set karna (exact)

```
ansible all -m lvol -a "vg=vgdata lv=lvapp size=8g"
```

6️⃣ Logical Volume delete karna

```
ansible all -m lvol -a "vg=vgdata lv=lvold state=absent"
```

7️⃣ Logical Volume shrink karna (⚠️ ext4 only)

```
ansible all -m lvol -a "vg=vgdata lv=lvtest size=3g force=yes"
```

8️⃣ Logical Volume stripe ke sath banana

```
ansible all -m lvol -a "vg=vgdata lv=lvstripe size=10g stripes=2"
```

9️⃣ Logical Volume snapshot banana

```
ansible all -m lvol -a "vg=vgdata lv=lvsnap snapshot=lvapp size=1g"
```

🔟 Logical Volume ka info dekhna

```
ansible all -m lvol -a "vg=vgdata lv=lvapp"
```

=================================**END**=================================