
DevOps Shack

TOP 100 Most Commonly Asked Commands In MNC Interviews

1. Linux Commands

1. ls – List directory contents

```
ls -l
```

2. cd – Change directory

```
cd /home/user/Documents
```

3. pwd – Print working directory

```
Pwd
```

4. mkdir – Create a new directory
arduino

```
mkdir new_folder
```

5. rm – Remove files or directories

```
rm -rf directory_name
```

6. cp – Copy files and directories

```
cp file1.txt /destination/path
```

7. mv – Move/rename files and directories

```
mv file1.txt /destination/path
```

8. cat – View the contents of a file

```
cat file.txt
```

9. grep – Search text using a pattern
lua

```
grep "error" log.txt
```

10. **find** – Search for files and directories
arduino

```
find /home -name "*.txt"
```

2. Git Commands

11. **git init** – Initialize a new Git repository
csharp

```
git init
```

12. **git clone** – Clone a repository

```
git clone https://github.com/user/repo.git
```

13. **git add** – Add files to the staging area
csharp

```
git add file.txt
```



-
14. **git commit – Commit changes**
sql

```
git commit -m "Initial commit"
```

15. **git push – Push changes to remote repository**
css

```
git push origin main
```

16. **git pull – Fetch and merge changes from remote**
css

```
git pull origin main
```

17. **git branch – List, create, or delete branches**
cpp

```
git branch new-branch
```

18. git checkout – Switch branches cpp

```
git checkout new-branch
```

19. git merge – Merge branches sql

```
git merge feature-branch
```

20. git log – Show commit history lua

```
git log --oneline
```

3. SQL Commands

21. SELECT – Retrieve data from a database sql

```
SELECT * FROM employees;
```



22. INSERT INTO – Insert new data into a table

pgsql

```
INSERT INTO employees (id, name, salary) VALUES  
(1, 'John', 5000);
```

23. UPDATE – Modify existing data

sql

```
UPDATE employees SET salary = 6000 WHERE id =  
1;
```

24. DELETE – Remove records

sql

```
DELETE FROM employees WHERE id = 1;
```

25. WHERE – Filter records

sql

```
SELECT * FROM employees WHERE salary > 5000;
```



26. ORDER BY – Sort query results

pgsql

```
SELECT * FROM employees ORDER BY salary DESC;
```

27. GROUP BY – Aggregate data

pgsql

```
SELECT department, COUNT(*) FROM employees  
GROUP BY department;
```

28. JOIN – Combine multiple tables

pgsql

```
SELECT employees.name, department.name  
FROM employees  
JOIN department ON employees.dept_id =  
department.id;
```

29. HAVING – Filter groups

pgsql



```
SELECT department, COUNT(*) FROM employees  
GROUP BY department HAVING COUNT(*) > 5;
```

30. **LIMIT** – Restrict number of rows returned
pgsql

```
SELECT * FROM employees LIMIT 10;
```

4. Docker Commands

31. **docker run** – Run a container
arduino

```
docker run -d nginx
```

32. **docker ps** – List running containers
nginx

```
docker ps
```

33. **docker stop** – Stop a container
Arduino

```
docker stop container_id
```

34. docker rm – Remove a container

```
docker rm container_id
```

35. docker images – List all Docker images
nginx

```
docker images
```

36. docker build – Build a Docker image
nginx

```
docker build -t myimage .
```

37. docker exec – Execute a command in a running container

```
docker exec -it container_id
```



38. **docker logs** – View logs of a container
nginx

```
docker logs container_id
```

39. **docker network ls** – List networks

```
docker network ls
```

40. **docker-compose up** – Start services from a
docker-compose.yml

```
docker-compose up -d
```

5. Kubernetes Commands

41. **kubectl get pods** – List all running pods
arduino

```
kubectl get pods
```



-
42. **kubectl describe pod** – Show details of a pod
sql

```
kubectl describe pod pod_name
```

43. **kubectl logs** – View logs of a pod
nginx

```
kubectl logs pod_name
```

44. **kubectl apply** – Apply configuration changes
nginx

```
kubectl apply -f deployment.yaml
```

45. **kubectl delete pod** – Delete a pod
cpp

```
kubectl delete pod pod_name
```

46. **kubectl exec** – Run commands in a pod

```
kubectl exec -it pod_name -- /bin/
```

- 47. kubectl get services – List all services**
arduino

```
kubectl get services
```

- 48. kubectl get deployments – List all deployments**
arduino

```
kubectl get deployments
```

- 49. kubectl scale – Scale a deployment**
lua

```
kubectl scale deployment myapp --replicas=3
```

- 50. kubectl rollout status – Check rollout status of a deployment**
Lua

```
kubectl rollout status deployment myapp
```

6. Python Commands

51. `python --version` – Check Python version

css

```
python --version
```

52. `pip list` – List installed Python packages

nginx

```
pip list
```

53. `virtualenv venv` – Create a virtual environment

nginx

```
virtualenv venv
```

54. `source venv/bin/activate` – Activate virtual environment
(Linux/macOS)

```
source venv/bin/activate
```



55. venv\Scripts\activate – Activate virtual environment (Windows)

venv\Scripts\activate

56. python -m http.server 8000 – Start a simple HTTP server
yaml

python -m http.server 8000

57. python -c "print('Hello World')" – Run a single Python command
scss

python -c "print('Hello World')"

58. pip install package_name – Install a Python package
nginx

pip install requests

59. python script.py – Run a Python script

nginx

`python script.py`

60. `pytest test_file.py` – Run tests using Pytest
nginx

`pytest test_file.py`

7. Networking Commands

61. `ping` – Check network connectivity
nginx

`ping google.com`

62. `tracert` – Display route packets take to a network host
(Linux/macOS)
nginx

`tracert google.com`

63. `tracert` – Display route packets take to a network host (Windows)
Nginx



```
tracert google.com
```

- 64. netstat – Display network connections**
nginx

```
netstat -an
```

- 65. nslookup – Query DNS for a domain's IP**
nginx

```
nslookup google.com
```

- 66. dig – Perform DNS lookups (Linux/macOS)**
nginx

```
dig google.com
```

- 67. ip a – Show IP addresses (Linux)**
css

```
ip a
```



68. ipconfig – Show network configuration (Windows)

```
ipconfig /all
```

69. nmap – Scan open ports on a network
css

```
nmap -p 22,80,443 google.com
```

70. tcpdump – Capture network traffic
css

```
tcpdump -i eth0
```

8. AWS Cloud Commands (AWS CLI)

71. aws configure – Configure AWS CLI
nginx

```
aws configure
```



72. aws s3 ls – List S3 buckets

```
aws s3 ls
```

73. aws s3 cp – Copy files to/from S3

```
aws s3 cp file.txt s3://my-bucket/
```

74. aws ec2 describe-instances – List EC2 instances
sql

```
aws ec2 describe-instances
```

75. aws lambda list-functions – List AWS Lambda functions
python

```
aws lambda list-functions
```

76. aws rds describe-db-instances – List RDS instances



sql

aws rds describe-db-instances

77. aws dynamodb list-tables – List DynamoDB tables
nginx

aws dynamodb list-tables

78. aws cloudwatch logs describe-log-groups – List CloudWatch logs
sql

aws cloudwatch logs describe-log-groups

79. aws iam list-users – List IAM users
nginx

aws iam list-users

80. aws eks list-clusters – List EKS clusters
Nginx



```
aws eks list-clusters
```

9. Microsoft Azure CLI Commands

81. az login – Login to Azure
nginx

```
az login
```

82. az account list – List available accounts
nginx

```
az account list
```

83. az vm list – List Azure virtual machines
css

```
az vm list --output table
```

84. az storage account list – List Azure storage accounts
css

```
az storage account list --output table
```

85. az group list – List resource groups

css

```
az group list --output table
```

86. Naz aks list – List AKS clusters

css

```
az aks list --output table
```

87. az functionapp list – List Azure Function apps

css

```
az functionapp list --output table
```

88. az cosmosdb list – List Azure CosmosDB instances

css

```
az cosmosdb list --output table
```

89. **az monitor log-profiles list** – List Azure monitoring logs
lua

```
az monitor log-profiles list
```

90. **az ad user list** – List Azure Active Directory users
css

```
az ad user list --output table
```

10. Miscellaneous DevOps & Scripting Commands

91. **chmod** – Change file permissions

```
chmod 755 script.sh
```

92. **chown** – Change file ownership
sql

```
chown user:user file.txt
```

93. tar – Archive files
nginx

```
tar -cvf archive.tar folder_name/
```

94. unzip – Extract ZIP files
python

```
unzip archive.zip
```

95. sed – Stream editor for modifying files
nginx

```
sed -i 's/old-text/new-text/g' file.txt
```

96. awk – Process text files
nginx

```
awk '{print $1}' file.txt
```



97. curl – Make HTTP requests
nginx

```
curl -X GET https://example.com
```

98. wget – Download files from the internet
nginx

```
wget https://example.com/file.zip
```

99. rsync – Sync files between locations

```
rsync -av source/ destination/
```

100. nohup – Run a command in the background
`nohup python script.py &`