

# Cloud Computing Cheat Sheet(AWS, Azure, GCP, Oracle)

## AWS Cloud

### EC2 (Elastic Compute Cloud)

- `aws ec2 describe-instances` – List all instances
  - `aws ec2 start-instances --instance-ids <id>` – Start an instance
  - `aws ec2 stop-instances --instance-ids <id>` – Stop an instance
  - `aws ec2 terminate-instances --instance-ids <id>` – Terminate an instance
  - `aws ec2 create-key-pair --key-name <name>` – Create a key pair
  - `aws ec2 describe-security-groups` – List security groups
- 

### S3 (Simple Storage Service)

- `aws s3 ls` – List buckets
  - `aws s3 mb s3://<bucket>` – Create a bucket
  - `aws s3 cp <file> s3://<bucket>/` – Upload a file
  - `aws s3 rm s3://<bucket>/<file>` – Delete a file
  - `aws s3 rb s3://<bucket> --force` – Delete a bucket
  - `aws s3 sync <local-dir> s3://<bucket>/` – Sync local and S3
- 

### IAM (Identity and Access Management)

- `aws iam list-users` – List IAM users
- `aws iam create-user --user-name <name>` – Create a user

- `aws iam attach-user-policy --user-name <name> --policy-arn <policy>` – Attach a policy
  - `aws iam list-roles` – List IAM roles
  - `aws iam create-role --role-name <name> --assume-role-policy-document file://policy.json` – Create a role
  - `aws iam list-policies` – List policies
- 

## **VPC (Virtual Private Cloud)**

- `aws ec2 describe-vpcs` – List VPCs
  - `aws ec2 create-vpc --cidr-block <CIDR>` – Create a VPC
  - `aws ec2 delete-vpc --vpc-id <id>` – Delete a VPC
  - `aws ec2 create-subnet --vpc-id <id> --cidr-block <CIDR>` – Create a subnet
  - `aws ec2 describe-security-groups` – List security groups
  - `aws ec2 describe-internet-gateways` – List internet gateways
- 

## **Lambda (Serverless Computing)**

- `aws lambda list-functions` – List all Lambda functions
  - `aws lambda create-function --function-name <name> --runtime <runtime> --role <role> --handler <handler>` – Create a function
  - `aws lambda update-function-code --function-name <name> --zip-file fileb://<file>.zip` – Update function code
  - `aws lambda delete-function --function-name <name>` – Delete a function
  - `aws lambda invoke --function-name <name> output.json` – Invoke a function
- 

## **Amazon EKS (Elastic Kubernetes Service)**

- `aws eks list-clusters` – List EKS clusters
- `aws eks describe-cluster --name my-cluster` – Describe an EKS cluster

- `aws eks create-cluster --name my-cluster --role-arn arn:aws:iam::account-id:role/EKSRole --resources-vpc-config subnetIds=subnet-xxxxxxx,securityGroupIds=sg-xxxxxxx` – Create an EKS cluster
  - `aws eks update-kubeconfig --name my-cluster` – Configure kubectl to use the EKS cluster
  - `kubectl get nodes` – Check worker nodes
  - `kubectl get pods -A` – List running pods
- 

## **Amazon ECS (Elastic Container Service)**

- `aws ecs list-clusters` – List ECS clusters
  - `aws ecs list-services --cluster my-cluster` – List ECS services
  - `aws ecs describe-services --cluster my-cluster --services my-service` – Describe an ECS service
- 

## **AWS CloudFormation**

- `aws cloudformation list-stacks` – List all stacks
  - `aws cloudformation create-stack --stack-name my-stack --template-body file://template.yml` – Create a stack
- 

## **CI/CD (CodePipeline, CodeBuild, CodeDeploy)**

### **AWS CodePipeline**

- `aws codepipeline list-pipelines` – List all pipelines
- `aws codepipeline start-pipeline-execution --name my-pipeline` – Start a pipeline execution

### **AWS CodeBuild**

- `aws codebuild list-projects` – List all CodeBuild projects
- `aws codebuild start-build --project-name my-project` – Start a build

## **AWS CodeDeploy**

- `aws deploy list-applications` – List all CodeDeploy applications
  - `aws deploy create-deployment --application-name MyApp --deployment-group-name MyDeploymentGroup --s3-location bucket=my-bucket,key=app.zip,bundleType=zip` – Deploy an application
- 

## **Security & Compliance**

- `aws cloudtrail describe-trails` – List CloudTrail logs
  - `aws secretsmanager list-secrets` – List all secrets
  - `aws secretsmanager get-secret-value --secret-id my-secret` – Retrieve a secret
- 

## **Monitoring & Logging (CloudWatch)**

- `aws cloudwatch list-metrics` – List available metrics
  - `aws cloudwatch put-metric-alarm --alarm-name cpu-high --metric-name CPUUtilization --namespace AWS/EC2 --statistic Average --period 300 --threshold 80 --comparison-operator GreaterThanThreshold --dimensions Name=InstanceId,Value=i-xxxxxxxxxx --evaluation-periods 2 --alarm-actions arn:aws:sns:region:account-id:my-topic` – Create a CloudWatch alarm
  - `aws logs describe-log-groups` – List all log groups
  - `aws logs get-log-events --log-group-name my-log-group --log-stream-name my-log-stream` – Retrieve log events
- 

## **Networking (VPC, ELB, Route 53)**

## **Amazon VPC**

- `aws ec2 describe-vpcs` – List all VPCs
- `aws ec2 describe-subnets` – List all subnets

## **Elastic Load Balancer (ELB)**

- `aws elbv2 describe-load-balancers` – List all load balancers

## **Amazon Route 53**

- `aws route53 list-hosted-zones` – List hosted zones
- 

## **Amazon ECR (Elastic Container Registry)**

- `aws ecr get-login-password | docker login --username AWS --password-stdin <aws_account_id>.dkr.ecr.<region>.amazonaws.com` – Authenticate Docker with ECR
  - `aws ecr list-repositories` – List all ECR repositories
- 

## **AWS Systems Manager (SSM)**

- `aws ssm describe-instances` – List managed instances
  - `aws ssm send-command --document-name "AWS-RunShellScript" --targets "Key=instanceIds,Values=i-xxxxxxxxxx" --parameters commands="sudo apt update"` – Run a command on an EC2 instance
- 

## **AWS Auto Scaling**

- `aws autoscaling describe-auto-scaling-groups` – List Auto Scaling groups
- `aws autoscaling update-auto-scaling-group --auto-scaling-group-name my-asg --desired-capacity 3` – Update the desired capacity

- `aws autoscaling set-desired-capacity --auto-scaling-group-name my-asg --desired-capacity 2` – Manually scale an Auto Scaling group
- 

## **AWS Elastic Beanstalk**

- `aws elasticbeanstalk describe-environments` – List all environments
  - `aws elasticbeanstalk create-application --application-name my-app` – Create an application
  - `aws elasticbeanstalk update-environment --environment-name my-env --version-label new-version` – Deploy a new version
- 

## **AWS Step Functions**

- `aws stepfunctions list-state-machines` – List all state machines
  - `aws stepfunctions start-execution --state-machine-arn arn:aws:states:region:account-id:stateMachine:MyStateMachine` – Start a state machine execution
  - `aws stepfunctions describe-execution --execution-arn arn:aws:states:region:account-id:execution:MyStateMachine:MyExecution` – Get execution details
- 

## **AWS Glue (ETL Service)**

- `aws glue get-databases` – List all Glue databases
  - `aws glue get-tables --database-name my-database` – List all tables in a database
  - `aws glue start-job-run --job-name my-glue-job` – Start a Glue job
- 

## **AWS SNS (Simple Notification Service)**

- `aws sns list-topics` – List all SNS topics
- `aws sns publish --topic-arn arn:aws:sns:region:account-id:MyTopic --message "Test Message"` – Publish a message to an SNS topic

## **AWS SQS (Simple Queue Service)**

- `aws sqs list-queues` – List all SQS queues
  - `aws sqs send-message --queue-url https://sqs.region.amazonaws.com/account-id/my-queue --message-body "Hello World"` – Send a message
- 

## **AWS Outposts(Managed on-prem cloud/On-premises AWS)**

### **# List and Describe Outposts**

`aws outposts list-outposts` # List all AWS Outposts

`aws outposts get-outpost --outpost-id <outpost-id>` # Get details of a specific Outpost

`aws outposts get-outpost-instance-types --outpost-id <outpost-id>` # List instance types in an Outpost

### **# Manage Outpost Resources**

`aws outposts list-sites` # List all Outpost sites

`aws outposts list-orders` # List Outpost orders

`aws outposts list-outpost-instances --outpost-id <outpost-id>` # List EC2 instances in an Outpost

### **# Deploy and Configure Outposts**

```
aws outposts create-order --line-items "[{\"catalogItemId\": \"item-id\",  
\"quantity\": 1}]" --outpost-id <outpost-id> # Order an Outpost
```

```
aws outposts update-outpost --outpost-id <outpost-id> --name <new-name> #  
Update Outpost configuration
```

## **# Networking and Storage on Outposts**

```
aws outposts list-outpost-network-devices --outpost-id <outpost-id> # List  
network devices in an Outpost
```

```
aws s3 ls --outpost-id <outpost-id> # List S3 buckets on an Outpost
```

```
aws s3 mb s3://<bucket-name> --outpost-id <outpost-id> # Create an S3 bucket in  
Outpost
```

## **# Deploy EC2 Instances in Outposts**

```
aws ec2 run-instances --image-id <ami-id> --instance-type <type> --subnet-id  
<outpost-subnet-id> # Launch an EC2 instance in Outpost
```

## **# Automate Outpost Deployments with CloudFormation**

```
aws cloudformation deploy --template-file outpost-config.yml --stack-name  
my-outpost-stack # Deploy Outpost resources using CloudFormation
```

## **# Monitor Outposts with CloudWatch**

```
aws cloudwatch list-metrics --namespace AWS/Outposts # List Outpost-related  
CloudWatch metrics
```



```
aws cloudwatch get-metric-data --metric-name CPUUtilization --namespace
AWS/Outposts # Monitor CPU usage of Outpost instances
```

## **# Integrate Outposts with CI/CD**

```
aws codepipeline list-pipelines # List all CI/CD pipelines
```

```
aws codepipeline start-pipeline-execution --name <pipeline-name> # Start a
deployment pipeline for Outposts
```

```
aws codebuild start-build --project-name <build-project> # Start a build process
for Outposts workloads
```

```
aws deploy create-deployment --application-name <app-name>
--deployment-group-name <group-name> --s3-location
bucket=<bucket-name>,key=<app.zip>,bundleType=zip # Deploy an application
to an Outpost
```

## **# Security and Compliance for Outposts**

```
aws iam create-role --role-name <role-name> --assume-role-policy-document
file://policy.json # Create an IAM role for Outpost management
```

```
aws secretsmanager list-secrets # List stored secrets for Outposts
```

```
aws secretsmanager get-secret-value --secret-id <secret-name> # Retrieve a stored
secret
```

```
aws cloudtrail describe-trails # List AWS CloudTrail logs for security auditing
```

```
aws guardduty list-findings # Detect security threats related to Outposts
```

## **# Delete or Deactivate an Outpost**

aws outposts delete-outpost --outpost-id <outpost-id> # Delete an Outpost (must be empty)

aws outposts cancel-order --order-id <order-id> # Cancel an Outpost order before delivery

---

# Azure Cloud

## 1. Azure Virtual Machines (VMs)

- az vm create --resource-group <rg> --name <vm-name> --image <image>  
→ Create a VM
  - az vm list -o table  
→ List all VMs
  - az vm stop --name <vm-name> --resource-group <rg>  
→ Stop a VM
  - az vm start --name <vm-name> --resource-group <rg>  
→ Start a VM
  - az vm delete --name <vm-name> --resource-group <rg>  
→ Delete a VM
  - az vm resize --name <vm-name> --resource-group <rg> --size <vm-size>  
→ Resize a VM
  - az vm show --name <vm-name> --resource-group <rg>  
→ Show VM details
  - az vm open-port --port <port-number> --name <vm-name> --resource-group <rg>  
→ Open a port on a VM
- 

## 2. Azure Storage

- az storage account create --name <storage-name> --resource-group <rg> --location <region>  
→ Create a storage account

- `az storage container create --name <container-name> --account-name <storage-name>`  
→ Create a blob container
  - `az storage blob upload --file <file-path> --container-name <container-name> --account-name <storage-name>`  
→ Upload a file to Blob Storage
  - `az storage blob list --container-name <container-name> --account-name <storage-name>`  
→ List blobs in a container
  - `az storage account delete --name <storage-name> --resource-group <rg>`  
→ Delete a storage account
- 

### 3. Azure Kubernetes Service (AKS)

- `az aks create --resource-group <rg> --name <aks-name> --node-count <num> --generate-ssh-keys`  
→ Create an AKS cluster
  - `az aks get-credentials --resource-group <rg> --name <aks-name>`  
→ Get kubeconfig for AKS cluster
  - `kubectl get nodes`  
→ List AKS cluster nodes
  - `kubectl get pods -A`  
→ List all pods in AKS
  - `kubectl apply -f <file.>`  
→ Deploy an application in AKS
  - `kubectl delete -f <file.>`  
→ Remove an application from AKS
  - `az aks delete --name <aks-name> --resource-group <rg>`  
→ Delete an AKS cluster
- 

### 4. Azure Functions

- `az functionapp create --resource-group <rg> --name <app-name> --consumption-plan-location <region> --runtime <runtime>`  
→ Create an Azure Function App
- `az functionapp list -o table`  
→ List all Function Apps
- `az functionapp delete --name <app-name> --resource-group <rg>`  
→ Delete a Function App
- `func init <app-name>`  
→ Initialize a local Azure Functions project
- `func new`  
→ Create a new function
- `func start`  
→ Run functions locally
- `func azure functionapp publish <app-name>`  
→ Deploy function to Azure

## 6. Azure Networking

### Virtual Network (VNet)

- `az network vnet list --output table` – List VNets
- `az network vnet create --name myVNet --resource-group myResourceGroup --subnet-name mySubnet` – Create a VNet

### Network Security Group (NSG)

- `az network nsg list --output table` – List NSGs
- `az network nsg create --resource-group myResourceGroup --name myNSG` – Create an NSG

### Load Balancer

- `az network lb list --output table` – List load balancers
- `az network lb create --resource-group myResourceGroup --name myLB --sku Standard --frontend-ip-name myFrontend --backend-pool-name myBackendPool` – Create a Load Balancer

---

## Azure DevOps (CI/CD)

### Azure DevOps Projects

- az devops project list --output table – List all DevOps projects
- az devops project create --name myDevOpsProject – Create a DevOps project
- az devops project delete --id PROJECT\_ID --yes – Delete a DevOps project

### Azure Repos (Git)

- az repos list --output table – List all repositories
- az repos create --name myRepo – Create a new repo
- az repos delete --name myRepo --yes – Delete a repo

### Azure Pipelines

- az pipelines list --output table – List all pipelines
- az pipelines create --name myPipeline --repository myRepo --branch main --repository-type gitHub – Create a pipeline
- az pipelines run --name myPipeline – Run a pipeline

---

## Azure Monitor & Logging

### Azure Monitor

- az monitor metrics list --resource myResourceID – List metrics for a resource
- az monitor metrics alert list --resource-group myResourceGroup – List metric alerts

### Azure Log Analytics

- az monitor log-analytics workspace list --output table – List log analytics workspaces
  - az monitor log-analytics workspace create --resource-group myResourceGroup --workspace-name myWorkspace – Create a Log Analytics workspace
- 

## **Security & Compliance**

### **Azure Security Center**

- az security assessment list --output table – List security assessments
- az security setting update --name AutoProvisioning --value On – Enable auto-provisioning for Security Center

### **Azure Key Vault**

- az keyvault list --output table – List Key Vaults
  - az keyvault create --name myKeyVault --resource-group myResourceGroup --location eastus – Create a Key Vault
  - az keyvault secret set --vault-name myKeyVault --name mySecret --value "MySecretValue" – Store a secret in Key Vault
  - az keyvault secret show --vault-name myKeyVault --name mySecret – Retrieve a secret
- 

## **Azure Policies & Governance**

- az policy assignment list --output table – List all policy assignments
  - az policy assignment create --name myPolicyAssignment --policy myPolicyDefinition – Assign a policy
  - az policy assignment delete --name myPolicyAssignment – Delete a policy assignment
-

## Azure Active Directory (AAD)

- az ad user list --output table – List all users
  - az ad group list --output table – List all groups
  - az ad app list --output table – List all applications
  - az ad sp list --output table – List all service principals
- 

## Azure Backup & Recovery

- az backup vault list --output table – List all backup vaults
  - az backup vault create --resource-group myResourceGroup --name myBackupVault – Create a backup vault
  - az backup item list --resource-group myResourceGroup --vault-name myBackupVault --output table – List backup items
- 

# GCP Cloud

## 1. Compute Engine (VMs)

- gcloud compute instances create <vm-name> --zone=<zone> --machine-type=<type> --image=<image>  
→ Create a VM instance
- gcloud compute instances list  
→ List all VM instances
- gcloud compute instances start <vm-name> --zone=<zone>  
→ Start a VM instance
- gcloud compute instances stop <vm-name> --zone=<zone>  
→ Stop a VM instance
- gcloud compute instances delete <vm-name> --zone=<zone>  
→ Delete a VM instance
- gcloud compute ssh <vm-name> --zone=<zone>  
→ SSH into a VM instance

- gcloud compute firewall-rules create <rule-name> --allow tcp:<port>  
→ Open a specific port
- 

## 2. Google Kubernetes Engine (GKE)

- gcloud container clusters create <cluster-name> --num-nodes=<num>  
--zone=<zone>  
→ Create a GKE cluster
  - gcloud container clusters get-credentials <cluster-name> --zone=<zone>  
→ Get credentials for the GKE cluster
  - kubectl get nodes  
→ List cluster nodes
  - kubectl get pods -A  
→ List all running pods
  - kubectl apply -f <file.>  
→ Deploy an application to GKE
  - kubectl delete -f <file.>  
→ Remove an application from GKE
  - gcloud container clusters delete <cluster-name> --zone=<zone>  
→ Delete a GKE cluster
- 

## 3. Cloud Run (Serverless Containers)

- gcloud run deploy <service-name> --image=<gcr.io/project/image>  
--platform=managed --region=<region> --allow-unauthenticated  
→ Deploy an application to Cloud Run
- gcloud run services list  
→ List all deployed Cloud Run services
- gcloud run services update-traffic <service-name> --to-latest  
→ Update Cloud Run service to the latest image
- gcloud run services delete <service-name>  
→ Delete a Cloud Run service



- `gcloud run services update <service-name> --set-env-vars VAR_NAME=value`  
→ Set environment variables in a Cloud Run service

#### **4. Google Cloud Storage (GCS)**

- `gcloud storage buckets list` – List all storage buckets
  - `gcloud storage buckets create my-bucket --location US` – Create a storage bucket
  - `gcloud storage cp file.txt gs://my-bucket/` – Upload a file
  - `gcloud storage rm gs://my-bucket/file.txt` – Delete a file
  - `gcloud storage buckets delete my-bucket` – Delete a storage bucket
- 

#### **5. Google Cloud IAM (Identity & Access Management)**

- `gcloud iam roles list` – List all IAM roles
  - `gcloud iam service-accounts list` – List all service accounts
  - `gcloud projects add-iam-policy-binding PROJECT_ID --member=user:EMAIL --role=roles/editor` – Assign a role to a user
  - `gcloud projects remove-iam-policy-binding PROJECT_ID --member=user:EMAIL --role=roles/editor` – Remove a role from a user
- 

#### **Google Cloud SQL (Managed Database Service)**

- `gcloud sql instances list` – List all Cloud SQL instances
  - `gcloud sql instances create my-db --tier=db-f1-micro --region=us-central1` – Create a Cloud SQL instance
  - `gcloud sql instances start my-db` – Start a Cloud SQL instance
  - `gcloud sql instances stop my-db` – Stop a Cloud SQL instance
  - `gcloud sql instances delete my-db` – Delete a Cloud SQL instance
- 

#### **Google Cloud Build (CI/CD)**

- gcloud builds list – List all Cloud Build runs
  - gcloud builds submit --tag gcr.io/PROJECT\_ID/my-app – Build and push an image
  - gcloud builds triggers list – List Cloud Build triggers
- 

## **Google Cloud Deploy (CI/CD)**

- gcloud deploy releases list --delivery-pipeline=my-pipeline – List deployment releases
  - gcloud deploy rollouts list --release=my-release --delivery-pipeline=my-pipeline – List rollouts
- 

## **Google Cloud Logging & Monitoring**

### **Cloud Logging**

- gcloud logging logs list – List available logs
- gcloud logging read "resource.type=gce\_instance" --limit 10 – Read VM logs

### **Cloud Monitoring**

- gcloud monitoring metrics list – List available monitoring metrics
  - gcloud monitoring dashboards list – List all dashboards
- 

## **Google Cloud Networking**

### **VPC (Virtual Private Cloud)**

- gcloud compute networks list – List VPC networks
- gcloud compute networks create my-vpc --subnet-mode=custom – Create a VPC network

## **Firewall Rules**

- gcloud compute firewall-rules list – List firewall rules
- gcloud compute firewall-rules create allow-ssh --network=my-vpc --allow=tcp:22 – Allow SSH access

## **Load Balancers**

- gcloud compute forwarding-rules list – List all load balancers
- 

## **Google Cloud Security**

### **Cloud Identity-Aware Proxy (IAP)**

- gcloud iap web list – List IAP-secured applications
- gcloud iap settings get --resource-type=app-engine – Get IAP settings

### **Cloud Key Management Service (KMS)**

- gcloud kms keyrings list --location global – List key rings
  - gcloud kms keys list --keyring my-keyring --location global – List encryption keys
- 

## **Google Artifact Registry**

- gcloud artifacts repositories list – List all Artifact Repositories
  - gcloud artifacts repositories create my-repo --repository-format=docker --location=us-central1 – Create a Docker registry
- 

## **Google Cloud Pub/Sub (Messaging)**

- gcloud pubsub topics list – List all Pub/Sub topics
- gcloud pubsub topics create my-topic – Create a topic

- gcloud pubsub subscriptions list – List all subscriptions
  - gcloud pubsub subscriptions create my-sub --topic=my-topic – Create a subscription
- 

## Google Cloud Backup & Disaster Recovery

- gcloud backup vaults list – List all backup vaults
  - gcloud backup policies list – List backup policies
- 

## Google Cloud Policies & Governance

- gcloud policy-tags list --location=us – List policy tags
  - gcloud resource-manager org-policies list – List all organizational policies
- 

## Google Cloud Scheduler (Cron Jobs)

- gcloud scheduler jobs list – List all scheduled jobs
  - gcloud scheduler jobs create http my-job --schedule="0 \* \* \* \*" --uri=https://example.com/cron – Create a scheduled HTTP job
- 

## Oracle Cloud

### 1. Compute (VM Instances)

oci compute instance list # List all VM instances

oci compute instance launch --availability-domain <AD> --compartment-id <compartment-ocid> --shape <shape> --image-id <image-ocid> # Create a new VM instance

```
oci compute instance action --instance-id <instance-ocid> --action START # Start a VM instance
oci compute instance action --instance-id <instance-ocid> --action STOP # Stop a VM instance
oci compute instance terminate --instance-id <instance-ocid> --force # Terminate a VM instance
oci compute instance get --instance-id <instance-ocid> # Get details of a specific VM instance
oci compute image list --compartment-id <compartment-ocid> # List available VM images
```

---

## 2. Oracle Kubernetes Engine (OKE)

```
oci ce cluster create --compartment-id <compartment-ocid> --name <cluster-name> --vcn-id <vcn-ocid> # Create a new Kubernetes cluster
oci ce cluster list --compartment-id <compartment-ocid> # List all Kubernetes clusters
oci ce cluster delete --cluster-id <cluster-ocid> --force # Delete a Kubernetes cluster

kubectl get nodes # List all nodes in the cluster
kubectl get pods -A # List all running pods across namespaces
kubectl apply -f <file.yaml> # Deploy an application using a YAML file
kubectl delete -f <file.yaml> # Remove an application
```

---

## 3. Object Storage (Buckets & Files)

```
oci os bucket list --compartment-id <compartment-ocid> # List all storage buckets
oci os bucket create --compartment-id <compartment-ocid> --name <bucket-name> # Create a new storage bucket
```

```
oci os object put --bucket-name <bucket-name> --file <file> # Upload a file to a bucket
oci os object get --bucket-name <bucket-name> --name <file> --file <destination-path> # Download a file from a bucket
oci os object delete --bucket-name <bucket-name> --name <file> --force # Delete a file from a bucket
oci os bucket delete --bucket-name <bucket-name> --force # Delete a storage bucket
```

---

#### **4. Identity & Access Management (IAM)**

```
oci iam user list # List all users
oci iam policy list --compartment-id <compartment-ocid> # List all IAM policies
oci iam group list # List all IAM groups
```

```
oci iam user create --compartment-id <compartment-ocid> --name <username> --description "New User" # Create a new user
oci iam policy create --compartment-id <compartment-ocid> --name <policy-name> --statements "[\"ALLOW GROUP Administrators TO MANAGE ALL-RESOURCES IN TENANCY\"]" # Create a new policy
```

---

#### **5. Oracle Cloud Database (Autonomous & VM-Based)**

```
oci db autonomous-database list --compartment-id <compartment-ocid> # List all Autonomous Databases
```

```
oci db system list --compartment-id <compartment-ocid> # List all Database Systems
```

```
oci db autonomous-database create --compartment-id <compartment-ocid> --db-name <db-name> --admin-password <password> # Create an Autonomous Database
```

```
oci db autonomous-database stop --autonomous-database-id <db-id> # Stop an Autonomous Database
oci db autonomous-database start --autonomous-database-id <db-id> # Start an Autonomous Database
oci db autonomous-database delete --autonomous-database-id <db-id> --force # Delete an Autonomous Database
```

---

## 6. Load Balancing & Networking

```
oci lb load-balancer list --compartment-id <compartment-ocid> # List all load balancers
oci lb load-balancer create --compartment-id <compartment-ocid> --display-name <lb-name> --shape-name flexible --subnet-ids <subnet-ocid> # Create a new load balancer
```

```
oci network vcn list --compartment-id <compartment-ocid> # List all Virtual Cloud Networks (VCNs)
oci network subnet list --compartment-id <compartment-ocid> # List all subnets
oci network security-list list --compartment-id <compartment-ocid> # List all security lists
```

```
oci network security-list create --compartment-id <compartment-ocid> --vcn-id <vcn-ocid> --display-name <name> # Create a security list
oci network security-list delete --security-list-id <security-list-ocid> --force # Delete a security list
```

---

## 7. Oracle Cloud Logging & Monitoring

```
oci logging log list --compartment-id <compartment-ocid> # List all logs
oci logging log-group list --compartment-id <compartment-ocid> # List log groups
```

```
oci monitoring metric list --compartment-id <compartment-ocid> # List available monitoring metrics
oci monitoring alarm list --compartment-id <compartment-ocid> # List all alarms
oci monitoring alarm create --compartment-id <compartment-ocid> --display-name <alarm-name> --metric-compartment-id <compartment-ocid> --namespace <namespace> # Create an alarm
```

---

## 8. DevOps: OCI DevOps (Code Repos, Builds, Deployments)

```
oci devops project list --compartment-id <compartment-ocid> # List all DevOps projects
oci devops repository list --compartment-id <compartment-ocid> # List code repositories
oci devops build-pipeline list --compartment-id <compartment-ocid> # List build pipelines
oci devops deployment list --compartment-id <compartment-ocid> # List deployments

oci devops repository create --compartment-id <compartment-ocid> --name <repo-name> # Create a new code repository
oci devops build-pipeline create --compartment-id <compartment-ocid> --name <pipeline-name> # Create a build pipeline
oci devops deployment create --deployment-pipeline-id <pipeline-ocid> --display-name <deployment-name> # Create a deployment
```

---

## 9. Oracle Cloud Security (Vault, Keys, Policies)

```
oci vault vault list --compartment-id <compartment-ocid> # List all vaults
oci vault key list --compartment-id <compartment-ocid> --vault-id <vault-ocid> # List encryption keys
```



```
oci vault secret list --compartment-id <compartment-ocid> --vault-id <vault-ocid>  
# List stored secrets
```

```
oci vault key create --compartment-id <compartment-ocid> --display-name  
<key-name> --vault-id <vault-ocid> # Create a new encryption key  
oci vault secret create --compartment-id <compartment-ocid> --vault-id  
<vault-ocid> --secret-name <secret-name> --secret-content <secret-value> #  
Create a new secret
```