Cluster Info and Management

Check cluster information:

```
kubectl cluster-info
```

List nodes in the cluster:

```
kubectl get nodes
```

Get detailed node information:

```
kubectl describe node <node-name>
```

Check API server version:

```
kubectl version
```

Namespace Management

List all namespaces:

```
kubectl get namespaces
```

Switch to a specific namespace:

```
kubectl config set-context --current --namespace=<namespace>
```

Create a new namespace:

```
kubectl create namespace <namespace-name>
```

Pod Management

List all pods in the current namespace:

```
kubectl get pods
```

List pods across all namespaces:

```
kubectl get pods --all-namespaces
```

Create a pod:

```
kubectl run <pod-name> --image=<image-name> --restart=Never
```

Get detailed pod information:

```
kubectl describe pod <pod-name>
```

Delete a pod:

```
kubectl delete pod <pod-name>
```

Check pod logs:

```
kubectl logs <pod-name>
```

Execute a command inside a pod:

```
kubectl exec -it <pod-name> -- <command>
```

Deployment Management

Create a deployment:

```
kubectl create deployment <deployment-name> --image=<image-name>
```

List deployments:

```
kubectl get deployments
```

Scale a deployment:

```
kubectl scale deployment <deployment-name> --replicas=<number>
```

Update a deployment:

```
kubectl set image deployment/<deployment-name>
<container-name>=<new-image>
```

Rollback a deployment:

```
kubectl rollout undo deployment/<deployment-name>
```

Service Management

List all services:

```
kubectl get services
```

Expose a pod or deployment as a service:

```
kubectl expose pod <pod-name> --type=<ServiceType> --port=<port>
--target-port=<target-port>
```

Delete a service:

```
kubectl delete service <service-name>
```

ConfigMaps and Secrets

Create a ConfigMap from a file:

```
kubectl create configmap <configmap-name> --from-file=<filename>
```

List ConfigMaps:

```
kubectl get configmaps
```

Create a Secret from literal values:

```
kubectl create secret generic <secret-name>
--from-literal=<key>=<value>
```

List Secrets:

```
kubectl get secrets
```

Persistent Volumes and Claims

List persistent volumes (PVs):

```
kubectl get pv
```

List persistent volume claims (PVCs):

```
kubectl get pvc
```

Resource Management

List all resources:

```
kubectl get all
```

Get specific resource information:

```
kubectl get <resource-type>
```

Delete a resource:

```
kubectl delete <resource-type> <resource-name>
```

Apply a configuration file:

```
kubectl apply -f <file-name>
```

Logs and Debugging

View logs from a pod:

```
kubectl logs <pod-name>
```

View logs for a specific container:

```
kubectl logs <pod-name> -c <container-name>
```

Debug a failing pod:

```
kubectl describe pod <pod-name>
```

Start an interactive shell in a pod:

```
kubectl exec -it <pod-name> -- /bin/bash
```

Rollouts and Updates

Check rollout status:

```
kubectl rollout status deployment/<deployment-name>
```

Undo a rollout:

```
kubectl rollout undo deployment/<deployment-name>
```

Autoscaling

Create an autoscaler for a deployment:

```
kubectl autoscale deployment <deployment-name>
--cpu-percent=<percentage> --min=<min-pods> --max=<max-pods>
```

Check autoscaler status:

```
kubectl get hpa
```

Monitoring and Metrics

Install metrics server (if not already installed):

```
kubectl apply -f
https://github.com/kubernetes-sigs/metrics-server/releases/latest/down
load/components.yaml
```

Check resource usage of nodes:

```
kubectl top nodes
```

Check resource usage of pods:

```
kubectl top pods
```

Helm Commands (Optional)

If you're using Helm for package management in Kubernetes:

List installed Helm charts:

```
helm list
```

Install a chart:

```
helm install <release-name> <chart-name>
```

Upgrade a release:

```
helm upgrade <release-name> <chart-name>
```