Complete Kubernetes Cheat Sheet (Basic to Advanced)

Basic kubectl Commands

```
List resources:
  kubectl get pods
  kubectl get services
  kubectl get deployments

Create resources:
  kubectl apply -f config.yaml

Delete resources:
  kubectl delete -f config.yaml
  kubectl delete pod <pod-name>

View details:
  kubectl describe pod <pod-name>
  kubectl logs <pod-name>
```

Pod Management

```
Run a pod:
  kubectl run nginx --image=nginx

Exec into a pod:
  kubectl exec -it <pod-name> -- /bin/bash

Forward port:
  kubectl port-forward <pod-name> 8080:80

Restart pod:
  kubectl delete pod <pod-name>  # let deployment recreate it
```

Deployment and Scaling

```
Apply a deployment:
  kubectl apply -f deployment.yaml

Scale replicas:
  kubectl scale deployment my-deploy --replicas=5

Rolling update:
  kubectl rollout restart deployment my-deploy

Undo rollout:
  kubectl rollout undo deployment my-deploy
```

Complete Kubernetes Cheat Sheet (Basic to Advanced)

Services & Networking

```
Expose deployment:
  kubectl expose deployment my-deploy --type=LoadBalancer --port=80

View service:
  kubectl get svc

ClusterIP: internal only
NodePort: expose on each Node IP
LoadBalancer: public access via cloud provider
```

Configuration Management

```
ConfigMap:
  kubectl create configmap myconfig --from-literal=key=value

Secret:
  kubectl create secret generic mysecret --from-literal=password=1234

Mounting in pod:
  envFrom:
    - configMapRef:
        name: myconfig
    - secretRef:
        name: mysecret
```

Volumes & Storage

```
EmptyDir: lives as long as pod
HostPath: local node path
PersistentVolume & PVC:
  - Define PersistentVolume (PV)
  - Bind using PersistentVolumeClaim (PVC)

Apply with:
  kubectl apply -f pvc.yaml
```

Namespaces and Contexts

```
List namespaces:
  kubectl get namespaces

Use a namespace:
  kubectl config set-context --current --namespace=dev

Switch context:
  kubectl config use-context my-cluster
```

Complete Kubernetes Cheat Sheet (Basic to Advanced)

View current context:

```
  kubectl config current-context
```

Helm Basics

Install Helm chart:

```
  helm install myrelease bitnami/nginx
```

Upgrade:

```
  helm upgrade myrelease bitnami/nginx
```

Uninstall:

```
  helm uninstall myrelease
```

List releases:

```
  helm list
```

Kubernetes in Production

- Use resource requests/limits in pods
- Configure liveness/readiness probes
- Use HorizontalPodAutoscaler
- Apply RBAC (Role-based access control)
- Use namespaces for isolation
- Enable audit logging and monitoring

Advanced Concepts

Horizontal Pod Autoscaler:

```
  kubectl autoscale deployment my-deploy --cpu-percent=50 --min=1 --max=10
```

Network Policies:

```
  apiVersion: networking.k8s.io/v1
  kind: NetworkPolicy
```

Custom Resource Definitions (CRD):

Extend Kubernetes with custom API objects

Taints & Tolerations:

Restrict pod scheduling to specific nodes

Node Affinity:

Control placement based on labels