

Context & Configuration

```
kubectl config get-contexts — list all contexts
kubectl config current-context — show active context
kubectl config use-context NAME — switch context
kubectl config set-context -current -namespace=NS — set default namespace
kubectl config view -minify — view current config only
kubectl cluster-info — API server + addon endpoints
kubectl version -short — client/server versions
kubectl api-resources — all resource kinds + short names
kubectl api-versions — enabled API group/versions
kubectl explain pod.spec.containers — schema docs for a field
```

Get & Describe

```
kubectl get pods — list pods (current ns)
kubectl get pods -A — across all namespaces
kubectl get pods -o wide — + node, IP, nominated node
kubectl get all — pods, svc, deploy, rs in ns
kubectl get pods -w — watch for changes
kubectl get pod NAME -o yaml — full manifest
kubectl describe pod NAME — events + detailed state
kubectl get events -sort-by=.lastTimestamp — recent events ordered
kubectl get pods -field-selector status.phase=Running — filter by field
kubectl get pods -show-labels — display labels column
```

Pods & Logs

```
kubectl logs POD — stream container stdout
kubectl logs POD -c CTR — specific container
kubectl logs -f POD — follow / tail live
kubectl logs POD -previous — logs of crashed instance
kubectl logs POD -since=15m — last 15 minutes
kubectl logs POD -tail=100 — last 100 lines
kubectl logs -l app=api -max-log-requests=10 — logs by label
kubectl logs deploy/NAME -all-containers — all containers in deploy
```

Exec & Debug

```
kubectl exec -it POD - sh — interactive shell
kubectl exec POD -c CTR - env — run cmd in container
kubectl debug POD -it -image=busybox -target=CTR — ephemeral debug container
kubectl debug node/NODE -it -image=ubuntu — node debugging pod
kubectl cp POD:/path ./local — copy file out of pod
kubectl cp ./local POD:/path — copy file into pod
kubectl attach -it POD — attach to running process
kubectl port-forward POD 8080:80 — local:remote forward
kubectl port-forward svc/NAME 5432:5432 — forward to a Service
```

Deployments & Rollouts

```
kubectl rollout status deploy/NAME — wait for rollout
```

```
kubectl rollout history deploy/NAME — revision list
kubectl rollout undo deploy/NAME — roll back one rev
kubectl rollout undo deploy/NAME -to-revision=3 — to specific rev
kubectl rollout restart deploy/NAME — rolling restart (re-pull/refresh)
kubectl rollout pause deploy/NAME — freeze rollout
kubectl rollout resume deploy/NAME — continue rollout
kubectl set image deploy/NAME ctr=img:tag — update container image
```

Scaling & Autoscaling

```
kubectl scale deploy/NAME -replicas=5 — set replica count
kubectl scale -replicas=0 deploy/NAME — scale to zero
kubectl autoscale deploy/NAME -min=2 -max=10 -cpu-percent=70 — create HPA
kubectl get hpa — horizontal pod autoscalers
```

Services & Networking

```
kubectl get svc — list services
kubectl get endpoints NAME — backing pod IPs
kubectl get endpointslices — EndpointSlice objects
kubectl expose deploy/NAME -port=80 -target-port=8080 — create Service
kubectl get ingress -A — ingress rules everywhere
kubectl get networkpolicy — NetworkPolicies in ns
kubectl run tmp -rm -it -image=nicolaka/netshoot - bash — throwaway net debug pod
```

Config & Secrets

```
kubectl get configmap — list ConfigMaps
kubectl create configmap NAME -from-file=./dir — CM from files
kubectl create secret generic NAME -from-literal=key=val — generic secret
kubectl get secret NAME -o jsonpath='{.data.key}' | base64 -d — decode a secret value
kubectl create secret docker-registry NAME ... — image pull secret
```

Apply / Create / Delete

```
kubectl apply -f file.yaml — declarative apply
kubectl apply -k ./overlay — kustomize build + apply
kubectl diff -f file.yaml — preview server-side diff
kubectl apply -dry-run=server -f f.yaml — validate without writing
kubectl delete -f file.yaml — delete from manifest
kubectl delete pod NAME -grace-period=0 -force — force kill (last resort)
kubectl create -f f.yaml -save-config — imperative create
kubectl replace -force -f f.yaml — delete + recreate
kubectl patch deploy NAME -p '{...}' — strategic merge patch
kubectl edit deploy/NAME — edit live object in $EDITOR
```

Labels, Selectors, Annotations

```
kubectl label pod NAME tier=web — add/overwrite label
kubectl label pod NAME tier- — remove label
```

```
kubectl get pods -l 'env in (prod,stg)' — set-based selector
kubectl annotate pod NAME note='...' — add annotation
kubectl get pods -l app=api,tier!=cache — multiple selectors
```

Output & JSONPath

```
-o jsonpath='{.items[*].metadata.name}' — extract fields
-o custom-columns=NAME:.metadata.name,NODE:.spec.nodeName — custom table
-o go-template='{range .items}}...{{end}}' — go templating
-sort-by=.metadata.creationTimestamp — server-side sort
-o name — kind/name only (scriptable)
kubectl get pods -o json | jq '...' — pipe to jq
```

Nodes & Scheduling

```
kubectl get nodes -o wide — nodes + IPs, OS, kubelet
kubectl describe node NODE — capacity, allocated, taints
kubectl cordon NODE — mark unschedulable
kubectl uncordon NODE — mark schedulable
kubectl drain NODE -ignore-daemonsets -delete-emptydir-data — evict pods for maintenance
kubectl taint node NODE key=val:NoSchedule — add taint
kubectl taint node NODE key:NoSchedule- — remove taint
```

Resource Usage (metrics-server)

```
kubectl top nodes — CPU/mem per node
kubectl top pods -A -sort-by=memory — heaviest pods
kubectl top pod POD -containers — per-container usage
```

RBAC & Auth

```
kubectl auth can-i create deploy -n NS — check own permission
kubectl auth can-i '*' '*' -all-namespaces — am I cluster-admin?
kubectl auth can-i list secrets -as=system:serviceaccount:ns:sa — impersonate check
kubectl get rolebinding,clusterrolebinding -A — all bindings
kubectl auth whoami — current identity (1.27+)
```

Handy Aliases / Tips

```
alias k=kubectl — near-universal alias
export KUBECONFIG=a:b:c — merge multiple kubeconfigs
kubectl ... -context=CTX -n NS — one-off target override
source <(kubectl completion bash) — shell completion
kubectl get po -n NS NAME -o yaml | kubectl neat — strip managed fields (plugin)
```

k9s — Launch & Flags

```
k9s — open with current kubeconfig/context
k9s -n NS — start in a namespace
k9s -context CTX — specific context
k9s -c pods — start on a resource view
```

k9s -readonly — disable mutating actions
k9s -kubeconfig PATH — explicit kubeconfig
k9s -l debug — log level for k9s itself
k9s info — print config / log / dirs

Global Navigation

- : command mode — jump to any resource
- / filter rows in current view
- esc back / clear filter / exit prompt
- :q or Ctrl-c to quit k9s
- ? context-sensitive help / keypad
- :ctx switch context (:ctx NAME)
- :ns switch namespace
- 0 show all namespaces (digits = fav ns)
- Tab move between panes/results
- Ret drill into selection
- Ctrl-a list all available resource aliases
- Ctrl-w toggle wide columns (labels)
- Space mark a row (multi-select)

Resource Views (: commands)

- :pod or :po — pods
- :deploy or :dp — deployments
- :rs / :sts / :ds replicaset / statefulset / daemonset
- :svc services
- :ing ingresses
- :cm / :sec configmaps / secrets
- :no nodes
- :pvc / :pv / :sc storage objects
- :job / :cj jobs / cronjobs
- :sa / :role / :rb RBAC objects
- :events cluster events

- :crd custom resource definitions
- :rb / :cr rolebindings / clusterroles

Actions on a Resource

- d describe selected
- y view YAML
- e edit in \$EDITOR
- l view logs
- s shell into a pod/container
- a show attached / containers
- Ctrl-d delete (with confirm)
- Ctrl-k kill — delete with no grace
- Shift-f define a port-forward
- :pf list active port-forwards
- Ctrl-z toggle wait/sanitize (faults)
- r (node) cordon/drain menu actions

Logs View

- 0 tail since beginning (all)
- 145 last 1m / preset windows
- f toggle full-screen
- w toggle line wrap
- p show previous (crashed) container
- s toggle autoscroll on/off
- m mark a line
- c copy visible logs to clipboard
- / filter log lines

Filtering & Sorting

- /text fuzzy filter rows
- /!text inverse (exclude) filter
- /-l k=v filter by label selector

- Shift-n sort by NAME column
- Shift-c sort by CPU (metric views)
- Shift-m sort by MEM
- Shift-a sort by AGE
- Shift-s sort by STATUS

Special / Diagnostic Views

- :pulse or :pu — cluster health dashboard
- :popeye cluster sanitizer / linting report
- :xray TYPE tree view of resource relations
- :dir PATH browse manifests on disk
- :alias list every command alias
- :bench benchmark a port-forward (hey)
- :screendump dump current screen to file

Config & Files

\$XDG_CONFIG_HOME/k9s — config dir (or ~/.config/k9s)
config.yaml — global settings, refresh, skins
skins/*.yaml — color themes (ui.skin)
plugins.yaml — custom keybind → shell actions
aliases.yaml — user resource aliases
hotkeys.yaml — custom global hotkeys
K9S_CONFIG_DIR — override config location

Quick Workflow Notes

- **Fast triage:** k9s → :pulse for health, then :po → / on the failing app → l for logs, d for events.
- **Node maintenance:** kubectl cordon → drain -ignore-daemonsets → patch/reboot → uncordon.
- **CrashLoop:** kubectl logs -previous or k9s l+p; check describe events + kubectl get events.
- **ReadOnly demos:** k9s -readonly to prevent accidental mutations.