

KUBERNETES FROM ZERO · FOUNDATIONS 6 OF 12 · LESSON 6 · LESSON KIT · CHEAT SHEET

Kubernetes from Zero, part 6: Namespaces, labels and annotations

Namespaces, the recommended labels, annotations, and the commands from Lesson 6.

Every command ran on a kind v0.33.0 cluster (node v1.37.0, kubectl v1.37.2) on 10 Nov 2026. Cluster: k8s-ep04.

THE ONE RULE

Namespaces scope names and are where access and quotas attach. Labels identify and select, so follow app.kubernetes.io. Annotations hold everything else.

Namespaces you will see

four on every cluster, plus what kind and part 4 added

NAME	WHAT THE LESSON SAYS	WHERE IT COMES FROM
default	Where objects go when you do not name a namespace. The docs advise against using it for production.	Every cluster
kube-system	Objects created by the Kubernetes system, such as the control plane Pods and CoreDNS.	Every cluster
kube-public	Readable by all clients, including unauthenticated ones. The docs say the public part is a convention.	Every cluster
kube-node-lease	One Lease object per node. The kubelet renews it as a heartbeat.	Every cluster
local-path-storage	kind adds this for its storage provisioner.	kind
k8s-from-zero	The namespace from part 4. The lesson also creates team-a and team-b.	This series

Names must be valid DNS labels: lowercase letters, digits and `-`, at most 63 characters. Avoid names starting with `kube-`. Namespaces cannot be nested, and each namespaced object belongs to exactly one. Deleting a namespace takes everything inside with it.

Labels and annotations

the recommended set, and what does not belong in a label

KEY	WHAT THE LESSON SAYS IT ANSWERS	EXAMPLE
app.kubernetes.io/name	Which application this is.	`nginx`
app.kubernetes.io/instance	Which installation, unique per install.	`shop-web`
app.kubernetes.io/version	Which version. Quote values that look like numbers, because label values must be strings.	`"1.29"`
app.kubernetes.io/component	Which part of the architecture.	`frontend`
app.kubernetes.io/part-of	Which larger application it belongs to. One selector can find every piece.	`shop`
app.kubernetes.io/managed-by	Which tool manages it.	`kubectl`

Label values are at most 63 characters, may be empty, and allow no spaces. The prefixes `kubernetes.io/` and `k8s.io/` are reserved. Annotations hold everything else (owner, runbook, build ID). You cannot select on an annotation. All annotations on one object are limited to 256 KiB. The lab uses the prefix `example.com/`.

SCAN FOR THE LESSON

Lesson: **Kubernetes from Zero, part 6: Namespaces, labels and annotations**

https://aitamer.news/posts/k8s-from-zero-06-namespaces-labels-annotations/?utm_source=lessonkit&utm_medium=pdf&utm_campaign=k8s-06&utm_content=cheatsheet

NEXT IN THE SERIES **Next: Lesson 7, Deployments and ReplicaSets**. Kubernetes from Zero, part 7, Deployments and ReplicaSets. Lesson page

Free to share with credit: © aitamer.news

What the lab organised

two teams, one shop, then a namespace delete

1	namespaces	The same Pod name `web` can exist in `team-a` and `team-b` at once. `kubectl get pods` with no `-n` looks in the context namespace, which starts as `default` and prints `No resources found in default namespace`.
2	context	`kubectl config set-context --current --namespace=k8s-from-zero` stores the default namespace in the kubeconfig context. `-n` still wins for one command. Check with `kubectl config view --minify | grep namespace:`.
3	scope	`kubectl api-resources --namespaced=true` and `--namespaced=false` split the types. Nodes, PersistentVolumes, StorageClasses and Namespaces are cluster-scoped. For those, `-n` is silently ignored.
4	delete	`kubectl delete namespace team-b` removes the namespace and everything in it. Check with `kubectl get all -n team-b` first. A namespace stuck in `Terminating` is waiting on a finalizer. Part 22 covers that.

Key commands

commands verbatim from the lesson

Namespaces

```
kubectl get ns
```
four start with every cluster

```
kubectl create namespace team-a
```
or a Namespace manifest

```
kubectl get pods -n kube-system
```
`-n` wins over the context

```
kubectl get pods -A
```
adds a NAMESPACE column

```
kubectl config set-context --current \
  --namespace=k8s-from-zero
```
stored in the context

```
kubectl config view --minify | grep namespace:
```
empty means `default`

```
kubectl api-resources --namespaced=true
```
`false` for cluster-scoped

```
kubectl delete namespace team-b
```
check with `get all -n` first

Labels and annotations

```
kubectl get pods -l app.kubernetes.io/part-of=shop
```
one selector for all pieces

```
kubectl annotate pod shop-web example.com/owner=me \
  --overwrite
```
a trailing `-` removes the key

```
kubectl get pod shop-web -o \
  jsonpath='{.metadata.annotations.example\.com/owner}'
```
escape dots in the key

```
kubectl label pod shop-cache note="talk to payments"
```
the lesson shows this value is rejected

SCAN FOR THE LESSON

Lesson: **Kubernetes from Zero, part 6: Namespaces, labels and annotations**

https://aitamer.news/posts/k8s-from-zero-06-namespaces-labels-annotations/?utm_source=lessonkit&utm_medium=pdf&utm_campaign=k8s-06&utm_content=cheatsheet

NEXT IN THE SERIES **Next: Lesson 7, Deployments and ReplicaSets**. Kubernetes from Zero, part 7, Deployments and ReplicaSets. Lesson page

Free to share with credit: © aitamer.news