Commands

Operation	Description	Syntax
annotate	Add or update the annotations of one or more resources.	kubectl annotate (-f FILENAME | TYPE NAME | TYPE/NAME) KEY_1=VAL_1 ... KEY_N=VAL_N [--overwrite] [--all] [--resource-version=version] [flags]
api-versions	List the API versions that are available.	kubectl api-versions [flags]
apply	Apply a configuration change to a resource from a file or stdin.	kubectl apply -f FILENAME [flags]
attach	Attach to a running container either to view the output stream or interact with the container (stdin).	kubectl attach POD -c CONTAINER [-i] [-t] [flags]
autoscale	Automatically scale the set of pods that are managed by a replication controller.	kubectl autoscale (-f FILENAME | TYPE NAME | TYPE/NAME) [--min=MINPODS] --max=MAXPODS [--cpu-percent=CPU] [flags]
cluster-info	Display endpoint information about the master and services in the cluster.	kubectl cluster-info [flags]
config	Modifies kubeconfig files. See the individual subcommands for details.	kubectl config SUBCOMMAND [flags]
create	Create one or more resources from a file or stdin.	kubectl create -f FILENAME [flags]
delete	Delete resources either from a file, stdin, or specifying label selectors, names, resource selectors, or resources.	kubectl delete (-f FILENAME | TYPE [NAME | /NAME | -l label | --all]) [flags]
describe	Display the detailed state of one or more resources.	kubectl describe (-f FILENAME | TYPE [NAME_PREFIX | /NAME | -l label]) [flags]
edit	Edit and update the definition of one or more resources on the server by using the default editor.	kubectl edit (-f FILENAME | TYPE NAME | TYPE/NAME) [flags]
exec	Execute a command against a container in a pod,	kubectl exec POD [-c CONTAINER] [-i] [-t] [flags] [-- COMMAND [args...]]
explain	Get documentation of various resources. For instance pods, nodes, services, etc.	kubectl explain [--include-extended-apis=true] [--recursive=false] [flags]
expose	Expose a replication controller, service, or pod as a new Kubernetes service.	kubectl expose (-f FILENAME | TYPE NAME | TYPE/NAME) [--port=port] [--protocol=TCP|UDP] [--target-port=number-or-name] [--name=name] [----external-ip=external-ip-of-service] [--type=type] [flags]
get	List one or more resources.	kubectl get (-f FILENAME | TYPE [NAME | /NAME | -l label]) [--watch] [--sort-by=FIELD] [[-o | --output]=OUTPUT_FORMAT] [flags]
label	Add or update the labels of one or more resources.	kubectl label (-f FILENAME | TYPE NAME | TYPE/NAME) KEY_1=VAL_1 ... KEY_N=VAL_N [--overwrite] [--all] [--resource-version=version] [flags]
logs	Print the logs for a container in a pod.	kubectl logs POD [-c CONTAINER] [--follow] [flags]
patch	Update one or more fields of a resource by using the strategic merge patch process.	kubectl patch (-f FILENAME | TYPE NAME | TYPE/NAME) --patch PATCH [flags]
port-forward	Forward one or more local ports to a pod.	kubectl port-forward POD [LOCAL_PORT:]REMOTE_PORT [...[LOCAL_PORT_N:]REMOTE_PORT_N] [flags]
proxy	Run a proxy to the Kubernetes API server.	kubectl proxy [--port=PORT] [--www=static-dir] [--www-prefix=prefix] [--api-prefix=prefix] [flags]
replace	Replace a resource from a file or stdin.	kubectl replace -f FILENAME
rolling-update	Perform a rolling update by gradually replacing the specified replication controller and its pods.	kubectl rolling-update OLD_CONTROLLER_NAME ([NEW_CONTROLLER_NAME] --image=NEW_CONTAINER_IMAGE | -f NEW_CONTROLLER_SPEC) [flags]
run	Run a specified image on the cluster.	kubectl run NAME --image=image [--env="key=value"] [--port=port] [--replicas=replicas] [--dry-run=bool] [--overrides=inline-json] [flags]
scale	Update the size of the specified replication controller.	kubectl scale (-f FILENAME | TYPE NAME | TYPE/NAME) --replicas=COUNT [--resource-version=version] [--current-replicas=count] [flags]
stop	Deprecated: Instead, seekubectl delete.	kubectl stop
version	Display the Kubernetes version running on the client and server.	kubectl version [--client] [flags]

Resource Types

Options	Description
certificatesigningrequests	csr
clusters	
clusterrolebindings	
clusterroles	
componentstatuses	cs
configmaps	cm
cronjobs	
daemonsets	ds
deployments	deploy
endpoints	ep
events	ev
horizontalpodautoscalers	hpa
ingresses	ing
jobs	
limitranges	limits
namespaces	ns
networkpolicies	
nodes	no
persistentvolumeclaims	pvc
persistentvolumes	pv
poddisruptionbudget	pdb
pods	po
podsecuritypolicies	psp
podtemplates	
replicasets	rs
replicationcontrollers	rc
resourcequotas	quota
rolebindings	
roles	
secrets	
serviceaccounts	sa
services	svc
statefulsets	
storageclasses	
thirdpartyresources	

Command Options

Options	Description
--alsologtostderr	log to standard error as well as files
--as string	Username to impersonate for the operation
--certificate-authority string	Path to a cert file for the certificate authority
--client-certificate string	Path to a client certificate file for TLS
--client-key string	Path to a client key file for TLS
--cluster string	The name of the kubeconfig cluster to use
--context string	The name of the kubeconfig context to use
--insecure-skip-tls-verify	If true, the server's certificate will not be checked for validity. HTTPS connections insecure
--kubeconfig string	Path to the kubeconfig file to use for CLI requests.
--log-backtrace-at traceLocation	when logging hits line file:N, emit a stack trace (default :0)
--log-dir string	If non-empty, write log files in this directory
--logtostderr	log to standard error instead of files
--match-server-version	Require server version to match client version
-n, --namespace string	If present, the namespace scope for this CLI request
--password string	Password for basic authentication to the API server
--request-timeout string	The length of time to wait before giving up on a single server request. Non-zero values should contain a corresponding time unit (e.g. 1s, 2m, 3h). A value of zero means don't timeout requests. (default "0")
-s, --server string	The address and port of the Kubernetes API server
--stderrthreshold severity	logs at or above this threshold go to stderr (default 2)
--token string	Bearer token for authentication to the API server
--user string	The name of the kubeconfig user to use
--username string	Username for basic authentication to the API server
-v, --v Level	log level for V logs
--vmodule moduleSpec	comma-separated list of pattern=N settings for file-filtered logging

Output Options

Option	Description
-o=custom-columns=<spec>	Print a table using a comma separated list of custom columns.
-o=custom-columns-file=<filename>	Print a table using the custom columns template in the <filename> file.
-o=json	Output a JSON formatted API object.
-o=jsonpath=<template>	Print the fields defined in a jsonpath expression.
-o=jsonpath-file=<filename>	Print the fields defined by the jsonpath expression in the <filename> file.
-o=name	Print only the resource name and nothing else.
-o=wide	Output in the plain-text format with any additional information. For pods, the node name is included.
-o=yaml	Output a YAML formatted API object.

Common Operation Examples

Command	Description
$ kubectl create -f example-service.yaml	// Create a service using the definition in example-service.yaml.
$ kubectl create -f example-controller.yaml	// Create a replication controller using the definition in example-controller.yaml.
$ kubectl create -f <directory>	// Create the objects that are defined in any .yaml, .yml, or .json file within the <directory> directory.
$ kubectl get pods	// List all pods in plain-text output format.
$ kubectl get pods -o wide	// List all pods in plain-text output format and includes additional information (such as node name).
$ kubectl get replicationcontroller <rc-name>	// List the replication controller with the specified name in plain-text output format. Tip: You can shorten and replace the 'replicationcontroller' resource type with the alias 'rc'.
$ kubectl get rc,services	// List all replication controllers and services together in plain-text output format.
$ kubectl describe nodes <node-name>	// Display the details of the node with name <node-name>.
$ kubectl describe pods/<pod-name>	// Display the details of the pod with name <pod-name>.
$ kubectl describe pods <rc-name>	// Display the details of all the pods that are managed by the replication controller named <rc-name>. // Remember: Any pods that are created by the replication controller get prefixed with the name of the replication controller.
$ kubectl delete -f pod.yaml	// Delete a pod using the type and name specified in the pod.yaml file.
$ kubectl delete pods,services -l name=<label-name>	// Delete all the pods and services that have the label name=<label-name>.
$ kubectl delete pods --all	// Delete all pods.
$ kubectl exec <pod-name> date	// Get output from running 'date' from pod <pod-name>. By default, output is from the first container.
$ kubectl exec <pod-name> -c <container-name> date	// Get output from running 'date' in container <container-name> of pod <pod-name>.
$ kubectl exec -ti <pod-name> /bin/bash	// Get an interactive TTY and run /bin/bash from pod <pod-name>. By default, output is from the first container.
$ kubectl logs <pod-name>	// Return a snapshot of the logs from pod <pod-name>.
$ kubectl logs -f <pod-name>	// Start streaming the logs from pod <pod-name>. This is similar to the 'tail -f' Linux command.

Kubernetes Cheatsheet