

Abbreviations

configmaps	cm
daemonsets	ds
endpoints	ep
event	ev
namespaces	ns
nodes	no
Pods	po
replicasets	rs
serviceaccount	sa
services	svc

Creating Objects

```
kubectl create -f ./my-manifest.yaml
```

create resource(s)

```
kubectl create -f ./my1.yaml -f ./my2.yaml
```

create from multiple files

```
kubectl create -f ./dir
```

create resource(s) in all manifest files in dir

```
kubectl create -f https://git.io/vPieo
```

create resource(s) from url

```
kubectl run nginx --image=nginx
```

start a single instance of nginx

```
kubectl explain pods,svc
```

get the documentation for pod and svc manifests

```
cat <<EOF | kubectl create -f -
apiVersion: v1 kind: Secret
metadata: name: mysecret type:
Opaque data: password: $(echo
"s33msi4" | base64) username:
$(echo "jane" | base64) EOF
```

Create a secret with several keys

Scaling Resources

```
kubectl scale --replicas=3 rs/foo
```

Scale a replicaset named 'foo' to 3

```
kubectl scale --replicas=3 -f foo.yaml
```

Scale a resource specified in "foo.yaml" to 3

Viewing, Finding Resources

```
kubectl get services
```

List all services in the namespace

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

List all pods in all namespaces

```
kubectl get pods -o wide
```

List all pods in the namespace, with more details

```
kubectl get deployment my-dep
```

List a particular deployment

```
kubectl describe nodes my-node
```

Describe nodes

```
kubectl describe pods my-pod
```

Describe pods

Editing Resources

```
kubectl edit svc/docker-registry
```

Edit the service named docker-registry

Interacting with running Pods

```
kubectl logs my-pod
```

dump pod logs (stdout)

```
kubectl logs -f my-pod
```

stream pod logs (stdout)

Interacting with running Pods (cont)

```
kubectl run -i --tty busybox --image=busybox --sh
```

Run pod as interactive shell

```
kubectl attach my-pod -i
```

Attach to Running Container

```
kubectl port-forward my-pod 5000:6000
```

Forward port 6000 of Pod to your to 5000 on your local machine

```
kubectl exec my-pod -- ls /
```

Run command in existing pod (1 container case)

```
kubectl exec my-pod -c my-container -- ls /
```

Run command in existing pod (multi-container case)

Interacting with Nodes and Cluster

```
kubectl cordon my-node
```

Mark my-node as unschedulable

```
kubectl drain my-node
```

Drain my-node in preparation for maintenance

```
kubectl uncordon my-node
```

Mark my-node as schedulable

```
kubectl cluster-info
```

Display addresses of the master and services

Kubectl AutoComplete

```
source <(kubectl completion bash)
```

Updating Resources

kubectl rolling-update frontend-v1 -f frontend-v2.json	Rolling update pods of frontend-v1
cat pod.json kubectl replace -f -	Replace a pod based on the JSON passed into stdin
kubectl expose rc nginx --port=80 --target-port=8000	Create a service for a replicated nginx, which serves on port 80 and connects to the containers on port 8000
kubectl get pod mypod -o yaml sed 's/(image: myimage):.*\$/v4/' kubectl replace -f -	Update a single-container pod's image version (tag) to v4
kubectl autoscale deployment foo --min=2 --max=10	Auto scale a deployment "foo"

Deleting Resources

kubectl delete -f ./pod.json	Delete a pod using the type and name specified in pod.json
kubectl delete pod,service baz foo	Delete pods and services with same names "baz" and "foo"

Kubectl Context and Configuration

kubectl config view	Show Merged kubeconfig settings.
kubectl config current-context	Display the current-context
kubectl config use-context my-cluster-name	set the default context to my-cluster-name
kubectl config set-credentials kubeuser/foo.kubernetes.com --username=kubeuser --password=kubepassword	add a new cluster to your kubeconf that supports basic auth

Formatting output

-o=custom-columns=<spec>	Print a table using a comma separated list of custom columns
-o=custom-columns=fil <filename>	Print a table using the custom columns template in the <filename> file
-o=json	Output a JSON formatted API object
-o=name	Print only the resource name and nothing else
-o=wide	Output in the plain-text format with any additional information, and for pods, the node name is included

Formatting output (cont)

-o=yaml	Output a YAML formatted API object
---------	------------------------------------



By **pablorod**
cheatography.com/pablorod/

Not published yet.
 Last updated 7th February, 2017.
 Page 2 of 2.

Sponsored by **ApolloPad.com**
 Everyone has a novel in them. Finish Yours!
<https://apollopod.com>