doctl DigitalOcean CLI

START HERE

`doctl auth init`
Store your API token

`doctl account get`
Check you are logged in

`doctl compute droplet list`
See your servers

`doctl compute droplet create <name> --image <slug> --size <slug> --region <slug>`
Create a Droplet

`doctl compute ssh <droplet>`
SSH into a Droplet

`doctl kubernetes cluster kubeconfig save <cluster>`
Connect kubectl

`doctl databases list`
See managed databases

`doctl compute droplet delete <droplet>`
Remove a Droplet

auth init token in a context → **compute / kubernetes / databases** → **Create resource** slugs for region, size, image → **list / get -o json** → **Use it** ssh, kubeconfig, connection

CONCEPTS

Add -o json for scripts, or --format ID,Name,PublicIPv4 --no-header to pick columns.

Region, size and image are slugs: look them up with the compute region, size and image list commands.

Deleting asks for confirmation; add --force (-f) in scripts, and double-check the context first.

AUTH AND CONTEXTS

`doctl auth init`
Authenticate with a personal access token

`doctl auth init --context <name>`
Add a named context for another account

`doctl auth switch --context <name>`
Switch the active context

AUTH AND CONTEXTS (cont.)

`doctl account get`
Show account and droplet limit

OUTPUT

`-o json`
Print JSON instead of a table

`--format ID,Name,PublicIPv4`
Choose columns

`--no-header`
Drop the header row for scripts

DROPLETS

`doctl compute droplet list`
List Droplets

`doctl compute droplet create <name> --image <slug> --size <slug> --region <slug>`
Create a Droplet

Image and size are required; add --ssh-keys to log in as root.

`doctl compute droplet get <droplet>`
Show details of a Droplet

DROPLETS (cont.)

`doctl compute droplet delete <droplet>`
Delete a Droplet

`doctl compute droplet create ... --wait`
Block until the Droplet is ready

`doctl compute droplet create ... --ssh-keys <id-or-fingerprint>`
Add SSH keys

`doctl compute ssh <droplet>`
SSH into a Droplet

Uses your local SSH keys; -- passes extra flags to ssh.

LOOKUPS

`doctl compute region list`
Region slugs

`doctl compute size list`
Size slugs and prices

KUBERNETES

`doctl kubernetes cluster list`
List clusters

KUBERNETES (cont.)

`doctl kubernetes cluster create <name> --region <slug>`
Create a cluster

`doctl kubernetes cluster kubeconfig save <cluster>`
Add the cluster to ~/.kube/config

Then use kubectl as usual.

`doctl kubernetes cluster delete <cluster>`
Delete a cluster

DATABASES

`doctl databases list`
List database clusters

`doctl databases get <db-id>`
Show a cluster

`doctl databases connection <db-id>`
Show connection details

`doctl databases create <name> --engine <pg|mysql|redis> --region <slug>`
Create a cluster

DATABASES (cont.)

`doctl databases firewalls list <db-id>`
List trusted sources

`doctl databases firewalls append <db-id> --rule <type>:<value>`
Allow a source

DOMAINS AND FIREWALLS

`doctl compute domain list`
List domains

`doctl compute domain create <domain>`
Add a domain

`doctl compute domain records list <domain>`
List DNS records

`doctl compute domain records create <domain> --record-type A --record-name <name> --record-data <ip>`
Add a DNS record

`doctl compute firewall list`
List cloud firewalls

RECIPES

New Droplet and log in

```
doctl compute ssh-key list --
format ID,Name
doctl compute droplet create
web1 --image ubuntu-24-04-x64 \
  --size s-1vcpu-1gb --region
nyc1 --ssh-keys <key-id> --wait
doctl compute ssh web1
```

Droplet IPs for scripts

```
doctl compute droplet list --
tag-name web \
  --format PublicIPv4 --no-
header
```

Cluster to kubectl

```
doctl kubernetes cluster list
doctl kubernetes cluster
kubeconfig save <cluster>
kubectl get nodes
```

Point a name at a Droplet

```
IP=$(doctl compute droplet get
web1 --format PublicIPv4 --no-
header)
doctl compute domain records
create example.com \
  --record-type A --record-name
www --record-data "$IP"
```