MINIKUBE

```
PS C:\Users\User\Desktop\Docker\flask officie> minikube version
minikube version: v1.30.1
commit: 08896fd1dc362c097c925146c4a0d0dac715ace0
PS C:\Users\User\Desktop\Docker\flask officie> minikube start
😄  minikube v1.30.1 on Microsoft Windows 11 Pro 10.0.22000.1817 Build 22000.1817
✨  Using the docker driver based on existing profile
👍  Starting control plane node minikube in cluster minikube
🚜  Pulling base image ...
🏃  Updating the running docker "minikube" container ...
🌏  Preparing Kubernetes v1.26.3 on Docker 23.0.2 ...
🔎  Verifying Kubernetes components...
    ▪ Using image gcr.io/k8s-minikube/storage-provisioner:v5
🌟  Enabled addons: storage-provisioner, default-storageclass
🏄  Done! kubectl is now configured to use "minikube" cluster and "default" namespace by default
PS C:\Users\User\Desktop\Docker\flask officie> kubectl config view
apiVersion: v1
clusters:
- cluster:
    certificate-authority-data: DATA+OMITTED
    server: https://kubernetes.docker.internal:6443
  name: docker-desktop
- cluster:
    certificate-authority: C:\Users\User\.minikube\ca.crt
    extensions:
    - extension:
        last-update: Wed, 10 May 2023 23:18:04 IST
        provider: minikube.sigs.k8s.io
        version: v1.30.1
      name: cluster_info
    server: https://127.0.0.1:55500
  name: minikube
contexts:
- context:
    cluster: docker-desktop
    user: docker-desktop
  name: docker-desktop
```

```
- context:
    cluster: docker-desktop
    user: docker-desktop
  name: docker-desktop
- context:
    cluster: minikube
    extensions:
    - extension:
        last-update: Wed, 10 May 2023 23:18:04 IST
        provider: minikube.sigs.k8s.io
        version: v1.30.1
      name: context_info
    user: minikube
  name: minikube
current-context: minikube
kind: Config
preferences: {}
users:
- name: docker-desktop
  user:
    client-certificate-data: REDACTED
    client-key-data: REDACTED
- name: minikube
  user:
    client-certificate: C:\Users\User\.minikube\profiles\minikube\client.crt
    client-key: C:\Users\User\.minikube\profiles\minikube\client.key
PS C:\Users\User\Desktop\Docker\flask officie> kubectl get --help
Display one or many resources.

 Prints a table of the most important information about the specified resources. You can filter the list using a label
selector and the --selector flag. If the desired resource type is namespaced you will only see results in your current
namespace unless you pass --all-namespaces.

 By specifying the output as 'template' and providing a Go template as the value of the --template flag, you can filter
the attributes of the fetched resources.
```

```
the attributes of the fetched resources.

Use "kubectl api-resources" for a complete list of supported resources.

Examples:
  # List all pods in ps output format
  kubectl get pods

  # List all pods in ps output format with more information (such as node name)
  kubectl get pods -o wide

  # List a single replication controller with specified NAME in ps output format
  kubectl get replicationcontroller web

  # List deployments in JSON output format, in the "v1" version of the "apps" API group
  kubectl get deployments.v1.apps -o json

  # List a single pod in JSON output format
  kubectl get -o json pod web-pod-13je7

  # List a pod identified by type and name specified in "pod.yaml" in JSON output format
  kubectl get -f pod.yaml -o json

  # List resources from a directory with kustomization.yaml - e.g. dir/kustomization.yaml
  kubectl get -k dir/

  # Return only the phase value of the specified pod
  kubectl get -o template pod/web-pod-13je7 --template={{.status.phase}}

  # List resource information in custom columns
  kubectl get pod test-pod -o custom-columns=CONTAINER:.spec.containers[0].name,IMAGE:.spec.containers[0].image

  # List all replication controllers and services together in ps output format
  kubectl get rc,services

  # List one or more resources by their type and names
```

```
Options:
    -A, --all-namespaces=false:
        If present, list the requested object(s) across all namespaces. Namespace in current context is ignored even
        if specified with --namespace.

    --allow-missing-template-keys=true:
        If true, ignore any errors in templates when a field or map key is missing in the template. Only applies to
        golang and jsonpath output formats.

    --chunk-size=500:
        Return large lists in chunks rather than all at once. Pass 0 to disable. This flag is beta and may change in
        the future.

    --field-selector='':
        Selector (field query) to filter on, supports '=', '==', and '!='.(e.g. --field-selector
        key1=value1,key2=value2). The server only supports a limited number of field queries per type.

    -f, --filename=[]:
        Filename, directory, or URL to files identifying the resource to get from a server.

    --ignore-not-found=false:
        If the requested object does not exist the command will return exit code 0.

    -k, --kustomize='':
        Process the kustomization directory. This flag can't be used together with -f or -R.

    -L, --label-columns=[]:
        Accepts a comma separated list of labels that are going to be presented as columns. Names are case-sensitive.
        You can also use multiple flag options like -L label1 -L label2...

    --no-headers=false:
        When using the default or custom-column output format, don't print headers (default print headers).

    -o, --output='':
        Output format. One of: (json, yaml, name, go-template, go-template-file, template, templatefile, jsonpath,
        jsonpath-as-json, jsonpath-file, custom-columns, custom-columns-file, wide). See custom columns
```

```
        [https://kubernetes.io/docs/reference/kubectl/#custom-columns], golang template
        [http://golang.org/pkg/text/template/#pkg-overview] and jsonpath template
        [https://kubernetes.io/docs/reference/kubectl/jsonpath/].

    --output-watch-events=false:
        Output watch event objects when --watch or --watch-only is used. Existing objects are output as initial ADDED
        events.

    --raw='':
        Raw URI to request from the server.  Uses the transport specified by the kubeconfig file.

    -R, --recursive=false:
        Process the directory used in -f, --filename recursively. Useful when you want to manage related manifests
        organized within the same directory.

    -l, --selector='':
        Selector (label query) to filter on, supports '=', '==', and '!='.(e.g. -l key1=value1,key2=value2). Matching
        objects must satisfy all of the specified label constraints.

    --server-print=true:
        If true, have the server return the appropriate table output. Supports extension APIs and CRDs.

    --show-kind=false:
        If present, list the resource type for the requested object(s).

    --show-labels=false:
        When printing, show all labels as the last column (default hide labels column)

    --show-managed-fields=false:
        If true, keep the managedFields when printing objects in JSON or YAML format.

    --sort-by='':
[(-o|--output=)json|yaml|name|go-template|go-template-file|template|templatefile|jsonpath|jsonpath-as-json|jsonpath-file|custom-columns|custom-columns-file|
wide]
(TYPE[.VERSION][.GROUP] [NAME | -l label] | TYPE[.VERSION][.GROUP]/NAME ...) [flags] [options]
```

```
    --sort-by='':
[(-o|--output=)json|yaml|name|go-template|go-template-file|template|templatefile|jsonpath|jsonpath-as-json|jsonpath-file|custom-columns|custom-columns-file|
wide]
(TYPE[.VERSION][.GROUP] [NAME | -l label] | TYPE[.VERSION][.GROUP]/NAME ...) [flags] [options]

Use "kubectl options" for a list of global command-line options (applies to all commands).
PS C:\Users\User\Desktop\Docker\flask officie> kubectl get deploy --A
error: unknown flag: --A
See 'kubectl get --help' for usage.
PS C:\Users\User\Desktop\Docker\flask officie> kubectl get deploy -A
NAMESPACE     NAME      READY   UP-TO-DATE   AVAILABLE   AGE
kube-system   coredns   1/1     1            1           29h
```