

1

Literal key-value pairs

```
$ kubectl create configmap demo-config --
from-literal=greeting=Halo
```

2

From a config file

```
echo "greeting: Hei" > config
```

```
kubectl create configmap demo-config --from-
file=./config
```

```
1  apiVersion: apps/v1
2  kind: Deployment
3  metadata:
4    name: demo
5  spec:
6    replicas: 1
7    selector:
8      matchLabels:
9        app: demo
10   template:
11     metadata:
12       labels:
13         app: demo
14     spec:
15       containers:
16       - name: demo
17         image: clarusways/demo:hello-config-file
18         ports:
19         - containerPort: 8888
20         volumeMounts:
21         - mountPath: /config/
22           name: demo-config-volume
23           readOnly: true
24       volumes:
25       - name: demo-config-volume
26         configMap:
27           name: demo-config
28           items:
29           - key: config
30             path: demo.yaml
```

3

From a ConfigMap YAML file

```
k8s > {} configmap.yaml > {} data > config
1  apiVersion: v1
2  kind: ConfigMap
3  metadata:
4    name: demo-config
5  data:
6    config: |
7      greeting: Buongiorno
```

```
1  apiVersion: apps/v1
2  kind: Deployment
3  metadata:
4    name: demo
5  spec:
6    replicas: 1
7    selector:
8      matchLabels:
9        app: demo
10   template:
11     metadata:
12       labels:
13         app: demo
14     spec:
15       containers:
16       - name: demo
17         image: clarusways/demo:hello-config-file
18         ports:
19         - containerPort: 8888
20         volumeMounts:
21         - mountPath: /config/
22           name: demo-config-volume
23           readOnly: true
24       volumes:
25       - name: demo-config-volume
26         configMap:
27           name: demo-config
28           items:
29           - key: config
30             path: demo.yaml
```

Create ConfigMaps on Kubernetes

Configure all key-value pairs in a ConfigMap as container environment variables

```
k8s > { } configmap.yaml > { } data > greeting
1  apiVersion: v1
2  kind: ConfigMap
3  metadata:
4    name: demo-config
5  data:
6    greeting: Hola
```

```
k8s > { } deployment.yaml > { } spec > { } template > { } spec > [ ] containers > { } 0 > [ ] env
1  apiVersion: apps/v1
2  kind: Deployment
3  metadata:
4    name: demo
5  spec:
6    replicas: 1
7    selector:
8      matchLabels:
9        app: demo
10   template:
11     metadata:
12       labels:
13         app: demo
14     spec:
15       containers:
16         - name: demo
17           image: clarusways/demo:hello-config-env
18           ports:
19             - containerPort: 8888
20           env:
21             - name: GREETING
22               valueFrom:
23                 configMapKeyRef:
24                   name: demo-config
25                   key: greeting
```

Configure all key-value pairs in a ConfigMap as container environment variables in POSIX format

```
k8s > { } configmap.yaml > { } data > GREETING
1  apiVersion: v1
2  kind: ConfigMap
3  metadata:
4    name: demo-config
5  data:
6    GREETING: Merhaba
```

```
apiVersion: apps/v1
kind: Deployment
metadata:
  name: demo
spec:
  replicas: 1
  selector:
    matchLabels:
      app: demo
  template:
    metadata:
      labels:
        app: demo
    spec:
      containers:
        - name: demo
          image: clarusways/demo:hello-config-env
          ports:
            - containerPort: 8888
          envFrom:
            - configMapRef:
                name: demo-config
```