

Maryam Taherzadeh
week 11 homework
19529

Question 3- **7.4. Decoupling configuration with a ConfigMap**

Step-1: Different ways to create ConfigMap

a) Using the kubectl create configmap command and display yaml.

```
pod/fortune created
taherzadeh19529@cloudshell:~/fortune (phonic-axle-307118)$ kubectl create configmap fortune-config --from-literal=sleep-interval=25
configmap/fortune-config created
taherzadeh19529@cloudshell:~/fortune (phonic-axle-307118)$ kubectl get configmap fortune-config -o yaml
apiVersion: v1
data:
  sleep-interval: "25"
kind: ConfigMap
metadata:
  creationTimestamp: "2021-04-03T06:19:28Z"
  managedFields:
    - apiVersion: v1
      fieldsType: FieldsV1
      fieldsV1:
        f:data:
          .: {}
          f:sleep-interval: {}
      manager: kubectl-create
      operation: Update
      time: "2021-04-03T06:19:28Z"
    name: fortune-config
    namespace: default
    resourceVersion: "48524"
    selfLink: /api/v1/namespaces/default/configmaps/fortune-config
    uid: id4c4c86-e594-4a65-9873-ac2260449c5e
taherzadeh19529@cloudshell:~/fortune (phonic-axle-307118)$
```

```
sleep-interval: "25"
taherzadeh19529@cloudshell:~/fortune (phonic-axle-307118)$ kubectl create -f fortune-config.yaml
Error from server (AlreadyExists): error when creating "fortune-config.yaml": configmaps "fortune-config" already exists
taherzadeh19529@cloudshell:~/fortune (phonic-axle-307118)$ kubectl delete -f fortune-config.yaml
configmap "fortune-config" deleted
taherzadeh19529@cloudshell:~/fortune (phonic-axle-307118)$ kubectl create -f fortune-config.yaml
configmap/fortune-config created
```

b) Creating a ConfigMap entry from yaml.

```
taherzadeh19529@cloudshell:~/fortune (phonic-axle-307118)$ vim fortune-config.yaml
taherzadeh19529@cloudshell:~/fortune (phonic-axle-307118)$ cat fortune-config.yaml
apiVersion: v1
kind: ConfigMap
metadata:
  name: fortune-config
data:
  sleep-interval: "25"
```

```
taherzadeh19529@cloudshell:~/fortune (phonic-axle-307118)$ kubectl get configmap
NAME           DATA   AGE
fortune-config 1      85s
kube-root-ca.crt 1     104m
taherzadeh19529@cloudshell:~/fortune (phonic-axle-307118)$ vim fortune-config.yaml
taherzadeh19529@cloudshell:~/fortune (phonic-axle-307118)$ cat fortune-config.yaml
apiVersion: v1
kind: ConfigMap
metadata:
  name: fortune-config
data:
  sleep-interval: "25"
```

```
configmap/fortune-config created
taherzadeh19529@cloudshell:~/fortune (phonic-axle-307118)$ kubectl get configmap
NAME      DATA   AGE
fortune-config  1    32s
kube-root-ca.crt 1    108m
```

c) Creating a ConfigMap entry from the contents of a file

```
taherzadeh19529@cloudshell:~/fortune (phonic-axle-307118)$ vim config-file.conf
taherzadeh19529@cloudshell:~/fortune (phonic-axle-307118)$ cat config-file.conf
apiVersion: v1
kind: ConfigMap
metadata:
  name: fortune-config
data:
  sleep-interval: "25"
taherzadeh19529@cloudshell:~/fortune (phonic-axle-307118)$ kubectl create configmap my-config --from-file=customkey=config-file.conf
configmap/my-config created
```

```
taherzadeh19529@cloudshell:~/fortune (phonic-axle-307118)$ kubectl get configmap
NAME      DATA   AGE
fortune-config  1    2m58s
kube-root-ca.crt 1    110m
my-config       1    16s
```

```
configmap/my-config created
taherzadeh19529@cloudshell:~/fortune (phonic-axle-307118)$ kubectl get configmap my-config -o yaml
apiVersion: v1
data:
  customkey: |
    apiVersion: v1
    kind: ConfigMap
    metadata:
      name: fortune-config
    data:
      sleep-interval: "25"
kind: ConfigMap
metadata:
  creationTimestamp: "2021-04-04T01:05:21Z"
  managedFields:
  - apiVersion: v1
    fieldsType: FieldsV1
    fieldsV1:
      f:data:
        .: {}
        f:customkey: {}
    manager: kubectl-create
    operation: Update
    time: "2021-04-04T01:05:21Z"
  name: my-config
  namespace: default
  resourceVersion: "99441"
  selfLink: /api/v1/namespaces/default/configmaps/my-config
  uid: 622cbc53-e228-46a7-8405-207e6007e52b
```

d) Creating a ConfigMap from files in a directory

Mkdir configDir

```
taherzadeh19529@cloudshell:~/fortune/configDir (phonic-axle-307118)$ vim config-file.yaml
taherzadeh19529@cloudshell:~/fortune/configDir (phonic-axle-307118)$ cat config-file.yaml
apiVersion: v1
kind: ConfigMap
metadata:
  name: fortune-config-dir
data:
  sleep-interval: "25"
taherzadeh19529@cloudshell:~/fortune/configDir (phonic-axle-307118)$
```

```
taherzadeh19529@cloudshell:~/fortune/configDir (phonic-axle-307118)$ pwd
/home/taherzadeh19529/fortune/configDir
taherzadeh19529@cloudshell:~/fortune/configDir (phonic-axle-307118)$ kubectl create configmap my-config-dir --from-file=/home/taherzadeh19529/fortune/configDir
configmap/my-config-dir created
taherzadeh19529@cloudshell:~/fortune/configDir (phonic-axle-307118)$ kubectl get configmap
NAME        DATA   AGE
fortune-config    1     8m4s
kube-root-ca.crt 1     115m
my-config        1     5m22s
my-config-dir    1     26s
taherzadeh19529@cloudshell:~/fortune/configDir (phonic-axle-307118)$
```

```
taherzadeh19529@cloudshell:~/fortune/configDir (phonic-axle-307118)$ kubectl get configmap my-config-dir -o yaml
apiVersion: v1
data:
  config-file.yaml: |
    apiVersion: v1
    kind: ConfigMap
    metadata:
      name: fortune-config-dir
    data:
      sleep-interval: "25"
kind: ConfigMap
metadata:
  creationTimestamp: "2021-04-03T06:32:17Z"
  managedFields:
  - apiVersion: v1
    fieldsType: FieldsV1
    fieldsV1:
      f:data:
        .: {}
      f:config-file.yaml: {}
    manager: kubectl-create
    operation: Update
    time: "2021-04-03T06:32:17Z"
  name: my-config-dir
  namespace: default
  resourceVersion: "54238"
  selfLink: /api/v1/namespaces/default/configmaps/my-config-dir
  uid: 9d48397b-454e-4365-a69f-3badbf396b08
taherzadeh19529@cloudshell:~/fortune/configDir (phonic-axle-307118)$
```

e) Combining different options

```
taherzadeh19529@cloudshell:~/fortune (phonic-axle-307118)$ cd
taherzadeh19529@cloudshell:~ (phonic-axle-307118)$ cd fortune/
taherzadeh19529@cloudshell:~/fortune (phonic-axle-307118)$ vim foo.json
taherzadeh19529@cloudshell:~/fortune (phonic-axle-307118)$ cat foo.json
cat: foo.json: No such file or directory
taherzadeh19529@cloudshell:~/fortune (phonic-axle-307118)$ vim foo.json
taherzadeh19529@cloudshell:~/fortune (phonic-axle-307118)$ cat foo.json
{
    "foo": "bar"
    "baz": 5
}

taherzadeh19529@cloudshell:~/fortune (phonic-axle-307118)$ mkdir config-opt
taherzadeh19529@cloudshell:~/fortune (phonic-axle-307118)$ cd config-opt/
taherzadeh19529@cloudshell:~/fortune/config-opt (phonic-axle-307118)$ echo "true" >debug
taherzadeh19529@cloudshell:~/fortune/config-opt (phonic-axle-307118)$ echo "100" >repeat
taherzadeh19529@cloudshell:~/fortune/config-opt (phonic-axle-307118)$ cat debug
true
taherzadeh19529@cloudshell:~/fortune/config-opt (phonic-axle-307118)$ cd..
:bash: cd: command not found
taherzadeh19529@cloudshell:~/fortune/config-opt (phonic-axle-307118)$ cd.
:bash: cd : command not found
taherzadeh19529@cloudshell:~/fortune/config-opt (phonic-axle-307118)$ cd
taherzadeh19529@cloudshell:~ (phonic-axle-307118)$ cd fortune/
taherzadeh19529@cloudshell:~/fortune (phonic-axle-307118)$ echo "abc" >foobar.conf
taherzadeh19529@cloudshell:~/fortune (phonic-axle-307118)$ kubectl create configmap myconfig --from-file=foo.json --from-file=bar=foobar.conf --from-file=config-opt/ --from-literal
=something
configmap/myconfig created
taherzadeh19529@cloudshell:~/fortune (phonic-axle-307118)$
```

```
taherzadeh19529@cloudshell:~/fortune (phonic-axe-307118)$ kubectl delete configmap myconfig  
configmap "myconfig" deleted  
taherzadeh19529@cloudshell:~/fortune (phonic-axe-307118)$ kubectl delete configmap my-config  
configmap "my-config" deleted  
taherzadeh19529@cloudshell:~/fortune (phonic-axe-307118)$ kubectl create configmap my-config --from-file=foo.json --from-file=bar=foobar.conf --from-file=config-opta/ --from-literal l=some-thing  
configmap myconfig created  
taherzadeh19529@cloudshell:~/fortune (phonic-axe-307118)$
```

```
taherzadeh19529@cloudshell:~/fortune (phonic-axe-307118)$ kubectl get configmap myconfig -o yaml
apiVersion: v1
data:
  bar: |
    abc
  debug: |
    true
  foo.json: "{\"n\\t\\t\"foo\": \\"bar\\\"\\n\\t\\t\"baz\\\": 5\\n\\n\\n\""
  repeat: |
    100
  some: thing
kind: ConfigMap
metadata:
  creationTimestamp: "2021-04-03T06:45:02Z"
  managedFields:
  - apiVersion: v1
    fieldsType: FieldsV1
    fieldsV1:
      f:data:
        .: {}
        f:bar: {}
        f:debug: {}
        f:foo.json: {}
        f:repeat: {}
        f:some: {}
    manager: kubectl-create
    operation: Update
    time: "2021-04-03T06:45:02Z"
  name: myconfig
  namespace: default
  resourceVersion: "59924"
  selfLink: /api/v1/namespaces/default/configmaps/myconfig
  uid: 51fd0be80-8137-4e21-a230-c4991a073465
taherzadeh19529@cloudshell:~/fortune (phonic-axe-307118)$
```

Step-2: Passing a ConfigMap entry to a container as an environment variable

```
taherzadeh19529@cloudshell:~/fortune (phonic-axe-307118)$ kubectl delete configmap myconfig  
configmap "myconfig" deleted  
taherzadeh19529@cloudshell:~/fortune (phonic-axe-307118)$ kubectl delete configmap my-config  
configmap "my-config" deleted  
taherzadeh19529@cloudshell:~/fortune (phonic-axe-307118)$ kubectl create configmap my-config --from-file=foo.json --from-file=bar=foobar.conf --from-file=config-opta --from-literal  
l=some+thing  
configmap/my-config created  
taherzadeh19529@cloudshell:~/fortune (phonic-axe-307118)$
```

```
taherzadeh19529@cloudshell:~/fortune (phonic-axle-307118)$ kubectl get configmap myconfig -o yaml
apiVersion: v1
data:
  bar: |
    abc
  debug: |
    true
  foo.json: "(\n\t\"foo\": \"bar\"\n\t\"baz\": 5\n)\n"
  repeat: |
    100
  some: thing
kind: ConfigMap
metadata:
  creationTimestamp: "2021-04-03T06:45:02Z"
  managedFields:
  - apiVersion: v1
    fieldsType: FieldsV1
    fieldsV1:
      f:data:
        : {}
      f:bar: {}
      f:debug: {}
      f:foo.json: {}
      f:repeat: {}
      f:some: {}
    manager: kubectl-create
    operation: Update
    time: "2021-04-03T06:45:02Z"
  name: myconfig
  namespace: default
  resourceVersion: "59924"
  selfLink: /api/v1/namespaces/default/configmaps/myconfig
  uid: 51fbebe80-8137-4e21-a230-4991a073465
taherzadeh19529@cloudshell:~/fortune (phonic-axle-307118)$
```

```
taherzadeh19529@cloudshell:~/fortune (phonic-axle-307118)$ vim fortune-pod-env-configmap.yaml
taherzadeh19529@cloudshell:~/fortune (phonic-axle-307118)$ cat fortune-pod-env-configmap.yaml
apiVersion: v1
kind: Pod
metadata:
  name: fortune-env-from-configmap
spec:
  containers:
  - image: tahezadeh19529/fortune
    env:
      - name: INTERVAL
        valueFrom:
          configMapKeyRef:
            name: fortune-config
            key: sleep-interval
      name: html-generator
      volumeMounts:
      - name: html
        mountPath: /var/htdocs
    - image: nginx:alpine
      name: web-server
      volumeMounts:
      - name: html
        mountPath: /usr/share/nginx/html
        readOnly: true
    ports:
    - containerPort: 80
      protocol: TCP
  volumes:
  - name: html
    emptyDir: {}
```

```
taherzadeh19529@cloudshell:~/fortune (phonic-axle-307118)$ kubectl create -f fortune-pod-env-configmap.yaml
pod/fortune-env-from-configmap created
taherzadeh19529@cloudshell:~/fortune (phonic-axle-307118)$ kubectl get pods
NAME           READY   STATUS    RESTARTS   AGE
fortune         2/2     Running   0          37m
fortune-env-from-configmap 2/2     Running   0          12s
taherzadeh19529@cloudshell:~/fortune (phonic-axle-307118)$
```

Logs to prove:

```
taherzadeh19529@cloudshell:~/fortune (phonic-axle-307118)$ kubectl logs fortune-env-from-configmap -c html-generator
mkdir: cannot create directory '/var/htdocs': File exists
Sat Apr 3 06:53:23 UTC 2021 Writing fortune to /var/htdocs/index.html
Sat Apr 3 06:53:33 UTC 2021 Writing fortune to /var/htdocs/index.html
Sat Apr 3 06:53:43 UTC 2021 Writing fortune to /var/htdocs/index.html
Sat Apr 3 06:53:53 UTC 2021 Writing fortune to /var/htdocs/index.html
Sat Apr 3 06:54:03 UTC 2021 Writing fortune to /var/htdocs/index.html
Sat Apr 3 06:54:13 UTC 2021 Writing fortune to /var/htdocs/index.html
Sat Apr 3 06:54:23 UTC 2021 Writing fortune to /var/htdocs/index.html
Sat Apr 3 06:54:33 UTC 2021 Writing fortune to /var/htdocs/index.html
Sat Apr 3 06:54:43 UTC 2021 Writing fortune to /var/htdocs/index.html
taherzadeh19529@cloudshell:~/fortune (phonic-axle-307118)$
```

Step-3: Passing all entries of a ConfigMap as environment variables at once

```
taherzadeh19529@cloudshell:~/fortune (phonic-axle-307118)$ cat my-config-map.yaml
apiVersion: v1
kind: ConfigMap
metadata:
  name: my-config-map
data:
  CONFIG_FOO: "foo"
  CONFIG_BAR: "bar"
```

```
taherzadeh19529@cloudshell:~/fortune (phonic-axle-307118)$ kubectl create configmap my-config-map --from-file=my-config-map.yaml
configmap/my-config-map created
```

```
taherzadeh19529@cloudshell:~/fortune (phonic-axle-307118)$ kubectl edit configmap my-config-map
taherzadeh19529@cloudshell:~/fortune (phonic-axle-307118)$ kubectl get configmap my-config-map -o yaml
configmap/my-config-map edited
apiVersion: v1
data:
  my-config-map.yaml: |+
    apiVersion: v1
    kind: ConfigMap
    metadata:
      name: my-config-map
    data:
      CONFIG_FOO: "foo"
      CONFIG_BAR: "bar"
kind: ConfigMap
metadata:
  creationTimestamp: "2021-04-03T21:47:49Z"
  managedFields:
  - apiVersion: v1
    fieldsType: FieldsV1
    fieldsV1:
      f:data:
        manager: kubectl-create
        operation: Update
        time: "2021-04-03T21:47:49Z"
  - apiVersion: v1
    fieldsType: FieldsV1
    fieldsV1:
      f:data:
        f:my-config-map.yaml: {}
        manager: kubectl-edit
        operation: Update
        time: "2021-04-03T21:53:10Z"
  name: my-config-map
  namespace: default
  resourceVersion: "15296"
  selfLink: /api/v1/namespaces/default/configmaps/my-config-map
  uid: d67cd821-e558-47a3-baed-7eca2f8190d7
taherzadeh19529@cloudshell:~/fortune (phonic-axle-307118$
```

Step-4:passing ConfigMap entry as a command-line argument

```
taherzadeh19529@cloudshell:~/fortune (phonic-axle-307118)$ vim fortune-pod-args-configmap.yaml
taherzadeh19529@cloudshell:~/fortune (phonic-axle-307118)$ cat fortune-pod-args-configmap.yaml
apiVersion: v1
kind: Pod
metadata:
  name: fortune-args-from-configmap
spec:
  containers:
  - image: taherzadeh19529/fortune
    env:
      - name: INTERVAL
        valueFrom:
          configMapKeyRef:
            name: fortune-config
            key: sleep-interval
    args: ["$(INTERVAL)"]
    name: html-generator
    volumeMounts:
    - name: html
      mountPath: /var/htdocs
  - image: alpine
    name: web-server
    volumeMounts:
    - name: html
      mountPath: /usr/share/nginx/html
      readOnly: true
    ports:
    - containerPort: 80
      protocol: TCP
  volumes:
  - name: html
    emptyDir: {}

taherzadeh19529@cloudshell:~/fortune (phonic-axle-307118)$ kubectl create -f fortune-pod-args-configmap.yaml
pod/fortune-args-from-configmap created
```

```
taherzadeh19529@cloudshell:~/fortune (phonic-axle-307118)$ kubectl create -f fortune-pod-args-configmap.yaml
pod/fortune-args-from-configmap created
taherzadeh19529@cloudshell:~/fortune (phonic-axle-307118)$ kubectl get pods
NAME           READY   STATUS    RESTARTS   AGE
fortune-args-from-configmap   0/2     ContainerCreating   0          9s
```

```
configmap/fortune-config created
taherzadeh19529@cloudshell:~/fortune (phonic-axle-307118)$ kubectl get configmap fortune-config -o yaml
apiVersion: v1
data:
  sleep-interval: "25"
kind: ConfigMap
metadata:
  creationTimestamp: "2021-04-03T22:11:52Z"
  managedFields:
  - apiVersion: v1
    fieldsType: FieldsV1
    fieldsV1:
      f:data:
        .: {}
      f:sleep-interval: {}
    manager: kubectl-create
    operation: Update
    time: "2021-04-03T22:11:52Z"
  name: fortune-config
  namespace: default
  resourceVersion: "22076"
  selfLink: /api/v1/namespaces/default/configmaps/fortune-config
  uid: 6a77a2b0-ee81-4e01-a60e-16435db7d8ab
```

```
taherzadeh19529@cloudshell:~/fortune (phonic-axle-307118)$ kubectl logs fortune-args-from-configmap -c html-generator
mkdir: cannot create directory '/var/htdocs': File exists
Sat Apr 3 22:11:57 UTC 2021 Writing fortune to /var/htdocs/index.html
Sat Apr 3 22:12:07 UTC 2021 Writing fortune to /var/htdocs/index.html
Sat Apr 3 22:12:17 UTC 2021 Writing fortune to /var/htdocs/index.html
Sat Apr 3 22:12:27 UTC 2021 Writing fortune to /var/htdocs/index.html
Sat Apr 3 22:12:37 UTC 2021 Writing fortune to /var/htdocs/index.html
Sat Apr 3 22:12:47 UTC 2021 Writing fortune to /var/htdocs/index.html
Sat Apr 3 22:12:57 UTC 2021 Writing fortune to /var/htdocs/index.html
Sat Apr 3 22:13:07 UTC 2021 Writing fortune to /var/htdocs/index.html
Sat Apr 3 22:13:17 UTC 2021 Writing fortune to /var/htdocs/index.html
Sat Apr 3 22:13:27 UTC 2021 Writing fortune to /var/htdocs/index.html
Sat Apr 3 22:13:37 UTC 2021 Writing fortune to /var/htdocs/index.html
Sat Apr 3 22:13:47 UTC 2021 Writing fortune to /var/htdocs/index.html
Sat Apr 3 22:13:57 UTC 2021 Writing fortune to /var/htdocs/index.html
Sat Apr 3 22:14:07 UTC 2021 Writing fortune to /var/htdocs/index.html
Sat Apr 3 22:14:17 UTC 2021 Writing fortune to /var/htdocs/index.html
Sat Apr 3 22:14:27 UTC 2021 Writing fortune to /var/htdocs/index.html
Sat Apr 3 22:14:37 UTC 2021 Writing fortune to /var/htdocs/index.html
Sat Apr 3 22:14:47 UTC 2021 Writing fortune to /var/htdocs/index.html
Sat Apr 3 22:14:57 UTC 2021 Writing fortune to /var/htdocs/index.html
```

Step-5: Using a configMap volume to expose ConfigMap entries as files

a) Creating the ConfigMap

```
taherzadeh19529@cloudshell:~/fortune (phonic-axle-307118)$ cd configmap-files
taherzadeh19529@cloudshell:~/fortune/configmap-files (phonic-axle-307118)$ cat my-nginx-config.conf
server {
    listen          80;
    server_name     www.kubia-example.com;

    gzip on;
    gzip_types text/plain application/xml;

    location / {
        root   /usr/share/nginx/html;
        index  index.html index.htm;
    }
}
```

```
taherzadeh19529@cloudshell:~/fortune/configmap-files (phonic-axle-307118)$ cat sleep-interval
25
taherzadeh19529@cloudshell:~/fortune/configmap-files (phonic-axle-307118)$
```

```
taherzadeh19529@cloudshell:~/fortune (phonic-axle-307118)$ kubectl create configmap fortune-config--from-file=configmap-files
Error from server (AlreadyExists): configmaps "fortune-config" already exists
taherzadeh19529@cloudshell:~/fortune (phonic-axle-307118)$ kubectl delete configmap fortune-config
configmap "fortune-config" deleted
taherzadeh19529@cloudshell:~/fortune (phonic-axle-307118)$ kubectl create configmap fortune-config--from-file=configmap-files
taherzadeh19529@cloudshell:~/fortune (phonic-axle-307118)$ kubectl get configmap fortune-config -o yaml
apiVersion: v1
data:
  my-nginx-config.conf: "server (\n    listen          80;\n    server_name     www.kubia-example.com;\n\n    \\" gzip on;\n    gzip_types text/plain\n    application/xml;\n    \\"\\n\n    location / {\n        root   /usr/share/nginx/html;\n        index  index.html index.htm;\n    }\n)\n\nsleep-interval: 25"
kind: ConfigMap
metadata:
  creationTimestamp: "2021-04-04T00:45:59Z"
  managedFields:
  - apiVersion: v1
    fieldsType: FieldsV1
    fieldsV1:
      f:data:
        .: {}
        f:my-nginx-config.conf: {}
      f:sleep-interval: {}
    manager: kubectl-create
    operation: Update
    time: "2021-04-04T00:45:59Z"
  name: fortune-config
  namespace: default
  resourceVersion: "90808"
  selfLink: /api/v1/namespaces/default/configmaps/fortune-config
  uid: 7caabbbb-f79a-4801-80fa-2214285e3c09
taherzadeh19529@cloudshell:~/fortune (phonic-axle-307118)$
```

b) Using the ConfigMap's entries in a volume

```
curl -H "Content-Type: application/json" -XPUT http://127.0.0.1:8001/api/v1/namespaces/default/configmaps/fortune-args-from-configmap
{
  "data": {
    "INTERVAL": "10s"
  }
}
taherzadeh19529@cloudshell:~/fortune (phonic-axle-307118)$ cat fortune-pod-configmap-volume.yaml
apiVersion: v1
kind: Pod
metadata:
  name: fortune-configmap-volume
spec:
  containers:
    - image: taherzadeh19529/fortune
      env:
        - name: INTERVAL
          valueFrom:
            configMapKeyRef:
              name: fortune-config
              key: sleep-interval
      name: html-generator
      volumeMounts:
        - name: html
          mountPath: /var/htdocs
    - image: nginx:alpine
      name: web-server
      volumeMounts:
        - name: config
          mountPath: /etc/nginx/conf.d
          readOnly: true
        - name: config
          mountPath: /tmp/whole-fortune-config-volume
          readOnly: true
  ports:
    - containerPort: 80
      name: http
      protocol: TCP
  volumes:
    - name: html
      emptyDir: {}
```

```
taherzadeh19529@cloudshell:~/fortune (phonic-axle-307118)$ kubectl create -f fortune-pod-configmap-volume.yaml
pod/fortune-configmap-volume created
taherzadeh19529@cloudshell:~/fortune (phonic-axle-307118)$ kubectl get pods
NAME           READY   STATUS    RESTARTS   AGE
fortune-args-from-configmap   2/2     Running   0          75m
fortune-configmap-volume       2/2     Running   0          10s
taherzadeh19529@cloudshell:~/fortune (phonic-axle-307118)$ kubectl logs fortune-configmap-volume -c html-generator
mkdir: cannot create directory '/var/htdocs': File exists
Sat Apr 3 23:13:54 UTC 2021 Writing fortune to /var/htdocs/index.html
Sat Apr 3 23:14:04 UTC 2021 Writing fortune to /var/htdocs/index.html
Sat Apr 3 23:14:14 UTC 2021 Writing fortune to /var/htdocs/index.html
Sat Apr 3 23:14:24 UTC 2021 Writing fortune to /var/htdocs/index.html
Sat Apr 3 23:14:34 UTC 2021 Writing fortune to /var/htdocs/index.html
taherzadeh19529@cloudshell:~/fortune (phonic-axle-307118)$ kubectl port-forward fortune-configmap-volume 8080:80 &
[1] 3865
taherzadeh19529@cloudshell:~/fortune (phonic-axle-307118)$ Forwarding from 127.0.0.1:8080 -> 80
```

Create a new cloudshell terminal

```
Welcome to Cloud Shell! Type "help" to get started.
Your Cloud Platform project in this session is set to phonic-axle-307118.
Use "gcloud config set project [PROJECT_ID]" to change to a different project.
taherzadeh19529@cloudshell:~/fortune (phonic-axle-307118)$ cd fortune
taherzadeh19529@cloudshell:~/fortune (phonic-axle-307118)$ curl -H "Accept-Encoding: gzip" -i localhost:8080
HTTP/1.1 200 OK
Server: nginx/1.19.9
Date: Sat, 03 Apr 2021 23:16:57 GMT
Content-Type: text/html
Last-Modified: Sat, 03 Apr 2021 23:16:54 GMT
Connection: keep-alive
ETag: W/"6068c766-ca"
Content-Encoding: gzip

taherzadeh19529@cloudshell:~/fortune (phonic-axle-307118)$ kubectl exec fortune-configmap-volume -c web-server ls /etc/nginx/conf.d
kubectl exec [POD] [COMMAND] is DEPRECATED and will be removed in a future version. Use kubectl exec [POD] -- [COMMAND] instead.
my-nginx-config.conf
sleep-interval
taherzadeh19529@cloudshell:~/fortune (phonic-axle-307118)$
```

c) Exposing certain ConfigMap entries in the volume

```
taherzadeh19529@cloudshell:~/fortune (phonic-axle-307118)$ vim fortune-pod-configmap-volume-with-items.yaml
taherzadeh19529@cloudshell:~/fortune (phonic-axle-307118)$ cat fortune-pod-configmap-volume-with-items.yaml
apiVersion: v1
kind: Pod
metadata:
  name: fortune-configmap-volume-with-items
spec:
  containers:
    - image: tahezadeh19529/fortune
      name: html-generator
      volumeMounts:
        - name: html
          mountPath: /var/htdocs
    - image: nginx:alpine
      name: web-server
      volumeMounts:
        - name: html
          mountPath: /usr/share/nginx/html
          readOnly: true
        - name: config
          mountPath: /etc/nginx/conf.d/
          readOnly: true
      ports:
        - containerPort: 80
          protocol: TCP
  volumes:
    - name: html
      emptyDir: {}
    - name: config
      configMap:
        name: fortune-config
        items:
          - key: my-nginx-config.conf
            path: gzip.conf
```

```
taherzadeh19529@cloudshell:~/fortune (phonic-axle-307118)$ kubectl create -f fortune-pod-configmap-volume-with-items.yaml
pod/fortune-configmap-volume-with-items created
taherzadeh19529@cloudshell:~/fortune (phonic-axle-307118)$ kubectl get pods
NAME                  READY   STATUS    RESTARTS   AGE
fortune-args-from-configmap   2/2     Running   0          100m
fortune-configmap-volume   2/2     Running   0          25m
fortune-configmap-volume-with-items 2/2     Running   0          12s
```

d) Setting file permissions

```
taherzadeh19529@cloudshell:~/fortune (phonic-axle-307118)$ vim fortune-pod-configmap-volume-defaultMode.yaml
taherzadeh19529@cloudshell:~/fortune (phonic-axle-307118)$ cat fortune-pod-configmap-volume-defaultMode.yaml
apiVersion: v1
kind: Pod
metadata:
  name: fortune-configmap-volume
spec:
  containers:
    - image: tahezadeh19529/fortune
      env:
        - name: INTERVAL
          valueFrom:
            configMapKeyRef:
              name: fortune-config
              key: sleep-interval
      name: html-generator
      volumeMounts:
        - name: html
          mountPath: /var/htdocs
    - image: nginx:alpine
      name: web-server
      volumeMounts:
        - name: html
          mountPath: /usr/share/nginx/html
          readOnly: true
        - name: config
          mountPath: /etc/nginx/conf.d/
          readOnly: true
        - name: config
          mountPath: /tmp/whole-fortune-config-volume
          readOnly: true
      ports:
        - containerPort: 80
          name: http
          protocol: TCP
  volumes:
    - name: html
```

```
taherzadeh19529@cloudshell:~/fortune (phonic-axle-307118)$ kubectl create -f fortune-pod-configmap-volume-defaultMode.yaml
Error from server (AlreadyExists): error when creating "fortune-pod-configmap-volume-defaultMode.yaml": pods "fortune-configmap-volume" already exists
taherzadeh19529@cloudshell:~/fortune (phonic-axle-307118)$ kubectl delete configmap fortune-pod-configmap-volume-defaultMode.yaml
Error from server (NotFound): configmaps "fortune-pod-configmap-volume-defaultMode.yaml" not found
taherzadeh19529@cloudshell:~/fortune (phonic-axle-307118)$ kubectl delete configmap fortune-pod-configmap-volume-defaultMode.yaml
Error from server (NotFound): configmaps "fortune-pod-configmap-volume-defaultMode.yaml" not found
taherzadeh19529@cloudshell:~/fortune (phonic-axle-307118)$ kubectl delete -f fortune-pod-configmap-volume-defaultMode.yaml
pod "fortune-configmap-volume" deleted

taherzadeh19529@cloudshell:~/fortune (phonic-axle-307118)$ taherzadeh19529@cloudshell:~/fortune (phonic-axle-307118)$ kubectl create -f fortune-pod-configmap-volume-defaultMode.yaml
pod/fortune-configmap-volume created
```

```
pod/fortune-configmap-volume created
taherzadeh19529@cloudshell:~/fortune (phonic-axle-307118)$ kubectl get pods
NAME           READY   STATUS    RESTARTS   AGE
fortune-args-from-configmap   2/2     Running   0          124m
fortune-configmap-volume      2/2     Running   0          13s
fortune-configmap-volume-with-items 2/2     Running   0          24m
taherzadeh19529@cloudshell:~/fortune (phonic-axle-307118)$
```

Step-6: Updating an app's config without having to restart the app

a) Editing a ConfigMap

```
# 
#apiVersion: v1
data:
  my-nginx-config.conf: |+
    server {
      listen          80;
      server_name    www.kubia-example.com;
      |
      gzip off;
      gzip_types text/plain application/xml;
      location / {
        root   /usr/share/nginx/html;
        index  index.html index.htm;
      }
    }
    sleep-interval: |
  kind: ConfigMap
metadata:
  creationTimestamp: "2021-04-03T23:12:13Z"
  name: fortune-config
  namespace: default
  resourceVersion: "48984"
  selfLink: /api/v1/namespaces/default/configmaps/fortune-config
  uid: 43db6e23-ce96-4fed-9ac5-232892206f90
~
~
```

11, 9 All

-- INSERT --

```
taherzadeh19529@cloudshell:~/fortune (phonic-axle-307118)$ kubectl edit configmap fortune-config
configmap/fortune-config edited
```

```
taherzadeh19529@cloudshell:~/fortune (phonic-axle-307118)$ kubectl get configmap fortune-config -oyaml
apiVersion: v1
data:
my-nginx-config.conf: "server (\n    listen      80;\n    server_name      www.kubia-example.com;\n\n    \n    gzip off;\n    application/xml;\n        \n        location / (\n            root   /usr/share/nginx/html;\n\n            \n            index index.html index.htm;\n        )\n    )\n\n    sleep-interval: 1\n25\n"
kind: ConfigMap
metadata:
  creationTimestamp: "2021-04-03T23:12:13Z"
  managedFields:
  - apiVersion: v1
    fieldsType: FieldsV1
    fieldsV1:
      f:data:
        .: {}
        f:sleep-interval: {}
    manager: kubectl-create
    operation: Update
    time: "2021-04-03T23:12:13Z"
  - apiVersion: v1
    fieldsType: FieldsV1
    fieldsV1:
      f:data:
        f:my-nginx-config.conf: {}
    manager: kubectl-edit
    operation: Update
    time: "2021-04-03T23:12:42Z"
  name: fortune-config
  namespace: default
  resourceVersion: "81764"
  selfLink: /api/v1/namespaces/default/configmaps/fortune-config
  uid: 43dbe23-ce96-4fed-9ac5-232892206f90
taherzadeh19529@cloudshell:~/fortune (phonic-axle-307118)$
```

```
taherzadeh19529@cloudshell:~/fortune (phonic-axle-307118)$ kubectl exec fortune-configmap-volume -c web-server cat /etc/nginx/conf.d/my-nginx-config.conf
kubectl exec [POD] [COMMAND] is DEPRECATED and will be removed in a future version. Use kubectl exec [POD] -- [COMMAND] instead.
server {
    listen      80;
    server_name      www.kubia-example.com;

    gzip off;
    gzip_types text/plain application/xml;

    location / {
        root   /usr/share/nginx/html;
        index index.html index.htm;
    }
}
```

b) Signaling Nginx to reload the config

c) Understanding how the files are updated atomically

```
taherzadeh19529@cloudshell:~/fortune (phonic-axle-307118)$ kubectl exec fortune-configmap-volume -c web-server -- nginx -s reload
2021/04/04 00:12:42 [notice] 31#31: signal process started
taherzadeh19529@cloudshell:~/fortune (phonic-axle-307118)$ kubectl exec -it fortune-configmap-volume -c web-server -- ls -la
total 68
-rwxr-xr-x  1 root      root          0 Apr  4 00:03 .dockerenv
drwxr-xr-x  2 root      root  4096 Mar 31 16:51 bin
drwxr-xr-x  5 root      root  360 Apr  4 00:03 dev
drwxr-xr-x  1 root      root 4096 Apr  1 15:19 docker-entrypoint.d
-rwxrwxr-x  1 root      root 1202 Apr  1 15:19 docker-entrypoint.sh
drwxr-xr-x  1 root      root 4096 Apr  4 00:03 etc
drwxr-xr-x  2 root      root 4096 Mar 31 16:51 home
drwxr-xr-x  1 root      root 4096 Mar 31 16:51 lib
drwxr-xr-x  5 root      root 4096 Mar 31 16:51 media
drwxr-xr-x  2 root      root 4096 Mar 31 16:51 mnt
drwxr-xr-x  2 root      root 4096 Mar 31 16:51 opt
dr-xr-xr-x 212 root      root          0 Apr  4 00:03 proc
drwxr----- 2 root      root 4096 Mar 31 16:51 root
drwxr-xr-x  1 root      root 4096 Apr  4 00:03 run
drwxr-xr-x  2 root      root 4096 Mar 31 16:51 sbin
drwxr-xr-x  2 root      root 4096 Mar 31 16:51 srv
dr-xr-xr-x 12 root      root          0 Apr  4 00:03 sys
drwxrwxrwt  1 root      root 4096 Apr  4 00:03 tmp
drwxr-xr-x  1 root      root 4096 Mar 31 16:51 usr
drwxr-xr-x  1 root      root 4096 Mar 31 16:51 var
/taherzadeh19529@cloudshell:~/fortune (phonic-axle-307118)$
```