

## Лабораторная работа №7

### Задание 1

```
apiVersion: v1
data:
  APP_COLOR: blue
  APP_ENV: prod
kind: ConfigMap
metadata:
  annotations:
    kubectl.kubernetes.io/last-applied-configuration: |
      {"apiVersion": "v1", "data": {"APP_COLOR": "blue", "APP_ENV": "prod"}, "kind": "ConfigMap", "metadata": {"annotations": {}, "name": "web-config", "namespace": "default"}}
  creationTimestamp: "2025-10-31T10:12:17Z"
  name: web-config
  namespace: default
  resourceVersion: "42271"
  uid: 0f8fd7a8-503d-4add-970f-38be3f7b3df6
```

COLOR=blue  
ENVIRONMENT=prod

### Задание 2

```
apiVersion: v1
kind: ConfigMap
metadata:
  name: nginx-config
data:
  default.conf: |
    server {
      listen 80;
      listen [::]:80;
      server_name _;
      access_log /var/log/nginx/access.log;
      error_log /var/log/nginx/error.log warn;

      location / {
        default_type text/plain;
        return 200 'nginx started with ConfigMap\n';
      }
    }
```

```
apiVersion: v1
kind: Pod
metadata:
  name: nginx-pod
spec:
  containers:
    - name: nginx-pod
      image: nginx:1.25
      ports:
        - containerPort: 80
      volumeMounts:
        - name: nginx-config
          mountPath: /etc/nginx/conf.d/
  volumes:
    - name: nginx-config
      configMap:
        name: nginx-config
```

```
server {
  listen 80;
  listen [::]:80;
  server_name _;
  access_log /var/log/nginx/access.log;
  error_log /var/log/nginx/error.log warn;

  location / {
    default_type text/plain;
    return 200 'nginx started with ConfigMap\n';
  }
}
```

### Задание 3

```
apiVersion: v1
kind: Pod
metadata:
  name: multiconf-pod
spec:
  containers:
  - name: busybox-container
    image: busybox
    command: ["/bin/sh"]
    args: ["-c", "sleep 3600"]
    env:
    - name: LOG-LEVEL
      valueFrom:
        configMapKeyRef:
          name: app-config
          key: log-level
    volumeMounts:
    - name: app-config-vol
      mountPath: /app/config/
  volumes:
  - name: app-config-vol
    configMap:
      name: app-config
      items:
      - key: database.url
        path: db.properties
```

```
apiVersion: v1
kind: ConfigMap
metadata:
  name: app-config
data:
  database.url: jdbc:postgresql://localhost:5432/mydb
  log-level: DEBUG
```

```
PATH=/usr/local/sbin:/usr/local/bin:/usr/sbin:/usr/bin:/sbin:/bin
HOSTNAME=multiconf-pod
LOG-LEVEL=DEBUG
```

jdbc:postgresql://localhost:5432/mydb ↵

## Задание 4

Если указать несуществующий ключ - контейнер не запустится:

```
10s      Normal   Pulled          pod/web-pod           Container image "nginx:1.25-alpine"
already present on machine
10s      Warning  Failed          pod/web-pod           Error: couldn't find key APP_COLOR
in ConfigMap default/web-config
```

Если удалить configmap - ничего не произойдет, просто появится warning

```
LAST SEEN  TYPE      REASON     OBJECT            MESSAGE
49m       Normal    Scheduled  pod/nginx-pod  Successfully assigned default/nginx-pod to minikube
49m       Normal    Pulled     pod/nginx-pod  Container image "nginx:1.25" already present on machine
49m       Normal    Created    pod/nginx-pod  Created container: nginx-pod
49m       Normal    Started   pod/nginx-pod  Started container nginx-pod
7s        Warning   FailedMount pod/nginx-pod  MountVolume.SetUp failed for volume "nginx-config" : configmap "nginx-conf
ig" not found
```

## Задание 5

```
apiVersion: v1
data:
  footer.info: Версия 1.0 | ConfigMap Demo
  page.title: Мое приложение
  welcome.text: Добро пожаловать в демонстрационное приложение!
kind: ConfigMap
metadata:
  annotations:
    kubectl.kubernetes.io/last-applied-configuration: |
      {"apiVersion":"v1","data":{"footer.info":"Версия 1.0 | ConfigMap Demo","page.title":"Мое приложение","welcome.text":"Добро пожаловать в демонстрационное приложение!"},"kind":"ConfigMap","metadata":{},"name":"web-content","namespace":"default"}
  creationTimestamp: "2025-11-05T17:43:04Z"
  name: web-content
  namespace: default
  resourceVersion: "48456"
  uid: 63687d52-0397-4ba4-a650-e8ec16ef6bab
```

NAME	READY	STATUS	RESTARTS	AGE	LABELS
web-demo-b888896c8-7bdqj	1/1	Running	0	3m13s	app=web-demo,pod-template-hash=b888896c8
web-demo-b888896c8-g8dbb	1/1	Running	0	3m13s	app=web-demo,pod-template-hash=b888896c8

NAME	TYPE	CLUSTER-IP	EXTERNAL-IP	PORT(S)	AGE
web-demo-service	NodePort	10.101.107.98	<none>	80:30085/TCP	4m28s

```
total 12
drwxrwxrwx  3 root  root          4096 Nov  5 17:43 .
drwxr-xr-x  3 root  root          4096 Aug 13 18:52 ..
drwxr-xr-x  2 root  root          4096 Nov  5 17:43 ..2025_11_05_17_43_53.1975121340
lrwxrwxrwx  1 root  root           32 Nov  5 17:43 ..data -> ..2025_11_05_17_43_53.1975121340
lrwxrwxrwx  1 root  root           18 Nov  5 17:43 footer.info -> ..data/footer.info
lrwxrwxrwx  1 root  root           17 Nov  5 17:43 page.title -> ..data/page.title
lrwxrwxrwx  1 root  root           19 Nov  5 17:43 welcome.text -> ..data/welcome.text
```



Дөрөвдүүр дэгээндээдүйн бүрдүүлэх

## **Контрольные вопросы**

**1. Какие три основных способа использования ConfigMap в Pod вы знаете?**

**Ответ:** переменные окружения, аргументы командной строки, файл в Volume

**2. Что произойдет, если смонтировать ConfigMap как Volume и затем изменить данные в ConfigMap? Обновится ли конфигурация в работающем Pod?**

**Ответ:** конфигурация обновится в работающем Pod

**3. В чем разница между монтированием ConfigMap с секцией items и без нее?**

**Ответ:** без items ключи монтируются в одноименные файлы, с items можно выбрать имя файла

**4. Как можно передать все данные из ConfigMap в Pod в виде переменных окружения одной инструкцией?**

**Ответ:** envFrom + configMapRef

**5. Что произойдет с Pod, если он ссылается на несуществующую ConfigMap?**

**Ответ:** он не запустится, будет ошибка CreateContainerConfigError

**6. Какие команды вы бы использовали для диагностики проблем с ConfigMap?**

**Ответ:** logs, describe pod, get

**7. Может ли один Pod использовать несколько разных ConfigMap? Если да, то как?**

**Ответ:** да, может. Нужно указывать разные имена конфигов в volumeMounts

**8. Как переименовать файл при монтировании ключа из ConfigMap?**

**Ответ:** используя items