07 Horizontal Pod Autoscaling

Aufgabe

Konfiguriere die Anwendung so, dass bei hoher CPU-Auslastung automatisch weitere Pods hochgefahren werden.

Erstelle die Datei **hpa.yaml** und verwende folgende Konfiguration aus der Kubernetes-Dokumentation als Ausgangsbasis.

Folgendes ist zu konfigurieren:

- Der Name des Autoscaler soll **todo-backend-hpa** lauten
- Es soll das Deployment **backend-server** skaliert werden
- Es sollen minimal **1** Pod
- und maximal **5** Pods laufen
- Die Durschschnittliche CPU-Auslastung sollen **50%** sein

```
apiVersion: autoscaling/v2
kind: HorizontalPodAutoscaler
metadata:
  name: php-apache
spec:
  scaleTargetRef:
    apiVersion: apps/v1
    kind: Deployment
    name: php-apache
  minReplicas: 1
  maxReplicas: 10
  metrics:
  - type: Resource
    resource:
      name: cpu
      target:
        type: Utilization
        averageUtilization: 50
```

Anwenden der Konfiguration

```
kubectl apply -f hpa.yaml
```

Kubernetes beobachten

Beobachte nun im geteilten Terminal den Status von Kubernetes mit folgenden Befehlen

Gestartete Pods und deren Status beobachten

```
watch -n 1 kubectl get pods
```

CPU-Auslastung der Pods anzeigen

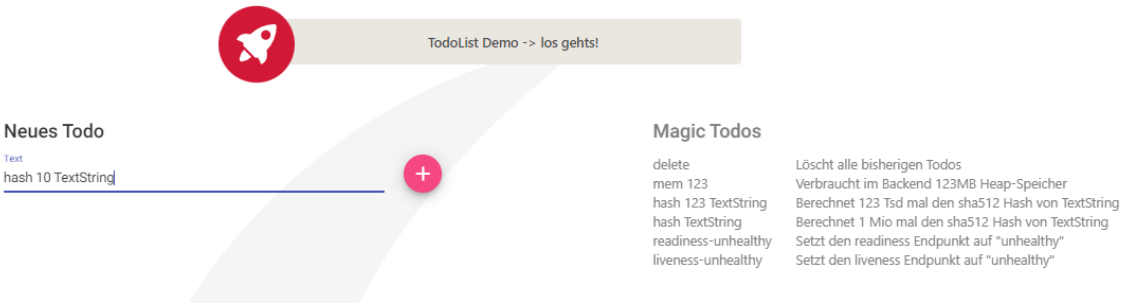
```
watch -n 1 kubectl top pod
```

Status des HPA anzeigen

```
watch -n 1 kubectl get hpa todo-backend-hpa
```

CPU Auslastung erzeugen

Über die Todo-App kannst du nun über das magische Todo "hash" CPU-Auslastung erzeugen. (Das hash-Todo berechnet tausende Male den SHA512 Hash des eingegebenen Strings.)



Über die Zahl nach dem Schlüsselwort "hash" kannst du bestimmen wie viele Tausend Male der Hash errechnet wird. Taste dich hier langsam an einen Wert heran, der dafür sorgt, dass es ca. 10 Sekunden dauert, bis das Todo mit dem errechneten Hash erscheint.

Sende nun in regelmäßigen Abständen das Todo zur Hash-Erzeugung und beobachte das Verhalten von Kubernetes.

Hinweis: Es wird 1-2Minuten dauern, bis der HPA anschlägt.