Microservice 인스턴스 개수 조정

DevOps 관리자에 의한 수동 조정 및 임계치 설정에 따른 Pod 의 자동 확장(HPA – Horizontal Pod AutoScaler)을 실습한다.
Instruction

Pod Scaling

Manual Pod Scaling

```
kubectl scale deploy/nginx-deployment --replicas=5
kubectl get all
kubectl scale deploy/nginx-deployment --replicas=2
kubectl get all
```

Pod Auto Scaling

- 터미널의 경로**/container-orchestration/yaml/** 인지 확인해 줍니다.

```
kubectl apply -f https://k8s.io/examples/application/php-apache.yaml
```

- 위의 명령어를 실행 후 실행 명령 하단에 두 가지가 만들어졌는지 확인해 줍니다.
  deployment.apps/php-apache created
  service/php-apache created

오토 스케일링 설정, **hpa: HorizontalPodAutoscaler**

```
kubectl autoscale deployment php-apache --cpu-percent=50 --min=1 --
max=10
```

- **cpu-percent=50 : Pod** 들의 평균 **CPU** 사용율
  (Pod 의 평균 **CPU** 사용율이 **100 milli-cores(50%)**를 넘게되면 **HPA** 발생**)**

```
kubectl get horizontalpodautoscaler
kubectl get hpa
```

- 위의 명령어를 입력 시 아래와 같이 출력되는지 확인해 줍니다.
  NAME REFERENCE TARGETS MINPODS MAXPODS REPLICAS AGE
  php-apache Deployment/php-apache 0%/50% 1 10 1 2m4s

로드제너레이터 설치

```
kubectl apply -f siege.yaml
```

- 아래의 결과가 출력되었는지 확인합니다.
  pod/siege created

```
kubectl get all
```

- **pod/siege 1/1 Running 0 30s**
  의 항목이 생성되었는지 확인해 줍니다.

```
kubectl exec -it siege -- /bin/bash
```

- **root@siege:/#** 로 변경되었는지 확인해 줍니다.

- 새로운 터미널을 하나 더 오픈한 다음 아래의 명령어로 실시간 인스턴스를 모니터링 합니다.

```
watch kubectl get pod
```

부하 테스트

```
siege
siege -c30 -t30S -v http://php-apache
```

- 아래와 비슷한 결과가 나왔는지 확인해 줍니다.

```
Lifting the server siege...
Transactions: 381 hits
Availability: 100.00 %
Elapsed time: 29.62 secs
Data transferred: 0.00 MB
Response time: 3.64 secs
Transaction rate: 12.86 trans/sec
Throughput: 0.00 MB/sec
Concurrency: 46.76
Successful transactions: 381
Failed transactions: 0
Longest transaction: 10.13
Shortest transaction: 0.12
```

- **watch kubectl get pod** 를 실행한 터미널에서 인스턴스 갯수를 확인합니다.

- **exit** 로 **Siege Pod** 에서 빠져나온다.

사용한 객체 삭제

- 예제 **php-apache** 객체를 삭제한다.

```
kubectl delete deployment php-apache
kubectl delete service php-apache
```

- 또는 아래 명령으로도 가능하다.

```
kubectl delete -f https://k8s.io/examples/application/php-apache.yaml
```

- **HPA** 객체도 삭제한다.

```
kubectl delete hpa php-apache
```

- 로더 제너레이터는 남겨둔다.

Memory-based Auto Scaling

- 메모리 **Metric Spec.**을 포함하는 **Deployment** 를 생성한다.

```
kubectl apply -f https://raw.githubusercontent.com/event-storming/container-orchestration/3e43582c64123c514827400dec4c69cc907dd971/yaml/hpa/php-hpa.yaml
kubectl apply -f https://raw.githubusercontent.com/event-storming/container-orchestration/master/yaml/hpa/hpa-memory.yaml
```

- 동일하게 **Siege** 컨테이너로 접근하여 부하를 발생하고 **HPA** 를 확인한다.

```
kubectl exec -it siege -- /bin/bash
siege -c10 -t10S -v http://php-apache
```

CheckPoints

1. 모든 요구사항을 만족하는가☐