

CKAD Ders Dökümanı #16 — Init Containers

Dil: Türkçe • CKAD Kategorisi: Application Design and Build (20%) • Standart: Konu anlatımı + komutlar + örnekler

1) Konu Anlatımı

Init container, normal container'lerden önce çalışan ve başarıyla tamamlanması gereken container'dır. Hazırlık işleri (config indirme, migrate, dosya oluşturma) için kullanılır.

Init container başarısız olursa Pod Ready olmaz; ana container'lar başlamaz.

2) En Sık Karşılaşılan YAML Örnekleri

Init container ile dosya hazırlama

```
apiVersion: v1
kind: Pod
metadata:
  name: init-demo
spec:
  volumes:
  - name: work
    emptyDir: {}
  initContainers:
  - name: init
    image: busybox:1.36
    command: ["sh", "-c"]
    args: ["echo 'hazır' > /work/status.txt"]
    volumeMounts:
    - name: work
      mountPath: /work
  containers:
  - name: app
    image: busybox:1.36
    command: ["sh", "-c"]
    args: ["cat /work/status.txt; sleep 3600"]
    volumeMounts:
    - name: work
      mountPath: /work
```

3) Sık Kullanılan Alanlar (Kısa Açıklamalar)

Alan	Ne işe yarar?
initContainers[]	Önce çalışacak container'lar.
containers[]	Init bitince başlayacak ana container'lar.
volumes/emptyDir	Init ile app arasında dosya paylaşımı.

4) En Sık Kullanılan Komutlar ve Kullanımları

Aşağıdaki komutlar CKAD pratiklerinde en çok kullanılanlardır.

4.1 Init durumunu izleme

Init container tamamlanana kadar Pod init state'te görünür.

```
kubectl get pod init-demo  
kubectl describe pod init-demo
```

4.2 Init container logları

Init hatalarında log okumak gerekir.

```
kubectl logs init-demo -c init
```

4.3 App container doğrulama

Hazırlanan dosyayı okuyor mu?

```
kubectl exec -it init-demo -c app -- sh
```

5) Troubleshooting Hızlı Rehber

Pod Init:CrashLoopBackOff: init container loglarına bak: `kubectl logs -c init` Dosya yok: shared volume mount path'leri eşleşiyor mu? Init uzun sürüyor: komut/bağımlılıkları kontrol et.

6) CKAD İpuçları

Init container loglarını unutma: `kubectl logs pod -c init` Shared data için en hızlı: `emptyDir`