

# CKAD Ders Dökümanı #09 — Resource Limits (Requests/Limits)

Dil: Türkçe • CKAD Kategorisi: Application Environment, Configuration and Security (25%) • Standart: Konu anlatımı + komutlar + örnekler

## 1) Konu Anlatımı

Kubernetes'te kaynak yönetimi iki parça: requests (scheduler için garanti/rezerv) ve limits (üst sınır).

CKAD'de en sık: Pod/Deployment'a CPU/RAM requests/limits ekleme, Pending/eviction veya OOMKilled gibi durumları teşhis etme.

## 2) En Sık Karşılaşılan YAML Örnekleri

### Resources örneği

```
resources:
  requests:
    cpu: "100m"
    memory: "128Mi"
  limits:
    cpu: "200m"
    memory: "256Mi"
```

### Bir Pod içinde örnek

```
apiVersion: v1
kind: Pod
metadata:
  name: res-demo
spec:
  containers:
    - name: app
      image: busybox:1.36
      resources:
        requests:
          cpu: "50m"
          memory: "64Mi"
        limits:
          cpu: "100m"
          memory: "128Mi"
      command: ["sh", "-c"]
      args: ["echo ok; sleep 3600"]
```

## 3) Sık Kullanılan Alanlar (Kısa Açıklamalar)

Alan	Ne işe yarar?
resources.requests.cpu/memory	Scheduler bu değerleri dikkate alır; node seçimi buna göre yapılır.
resources.limits.cpu/memory	Container'ın kullanabileceği maksimum kaynak.
OOMKilled	Memory limit aşıldığında container öldürülür.
QoS sınıfları	requests/limits kombinasyonlarına göre BestEffort/Burstable/Guaranteed

## 4) En Sık Kullanılan Komutlar ve Kullanımları

Aşağıdaki komutlar CKAD pratiklerinde en çok kullanılanlardır.

### 4.1 Kaynakları görüntüleme

Pod'un resource alanlarını ve runtime durumunu incellersiniz.

```
kubectl get pod res-demo -o yaml | sed -n '/resources:\/,\/imagePullPolicy/p'  
kubectl describe pod res-demo
```

### 4.2 (Varsa) metrics ile izleme

Cluster'da metrics-server varsa 'top' çalışır.

```
kubectl top pod  
kubectl top pod res-demo
```

### 4.3 Deployment'ta resource set

YAML ile düzenleme tipiktir; hızlıca edit edebilirsiniz.

```
kubectl edit deploy web  
# containers[].resources ekle
```

## 5) Troubleshooting Hızlı Rehber

Pending: node'da yeterli kaynak yok; describe event'lerinde 'Insufficient cpu/memory' görürsünüz. OOMKilled: memory limit düşük; logs/describe ile doğrula. CPU throttling: cpu limit düşük; latency artabilir (top/metrics varsa gözlemlenir).

## 6) CKAD İpuçları

Requests scheduler içindir; limits enforcement içindir. Hızlı teşhis: kubectl describe pod (Reason / Events). Sınavda değerleri doğru birimlerle yaz: m, Mi, Gi.