

Application Environment, Configuration, and Security for CKAD

Discover and Use Resources That Extend Kubernetes



Elle Krout

Principal Course Author, Pluralsight



Kubernetes Custom Resource Definitions



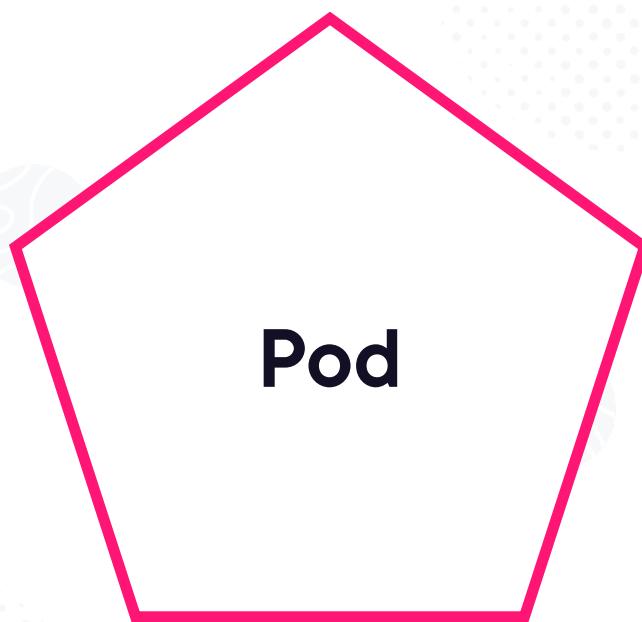
Resource (Kubernetes)

An endpoint in the Kubernetes API that stores a collection of API objects.

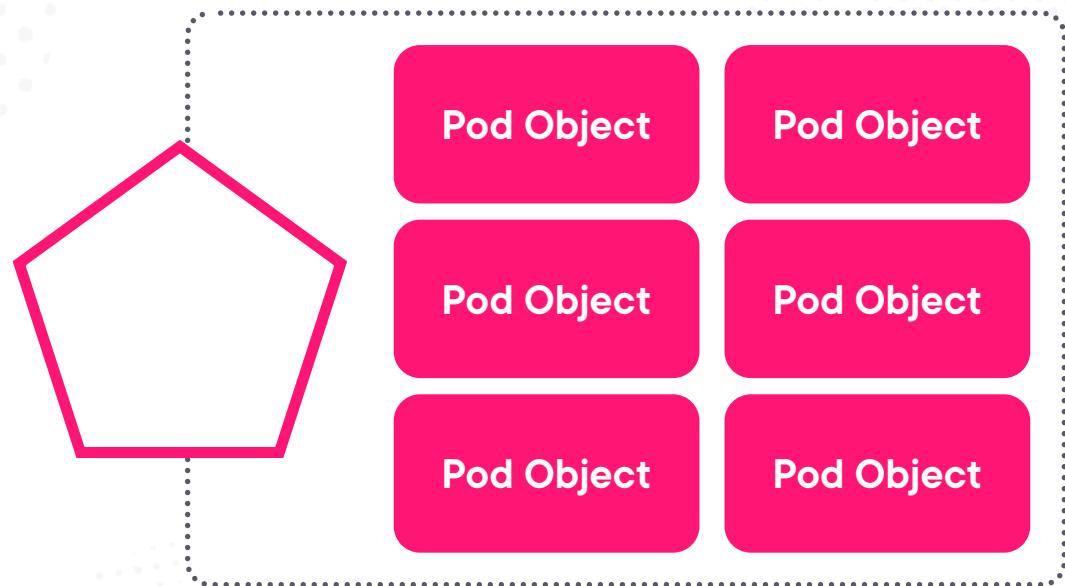
<https://kubernetes.io/docs/concepts/extend-kubernetes/api-extension/custom-resources>



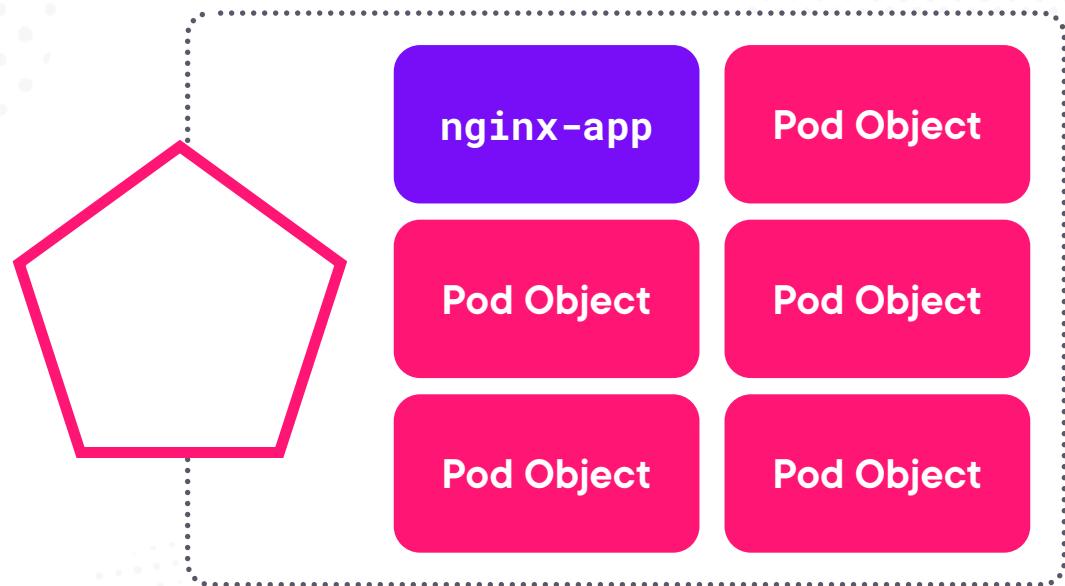
A Pod Is a Resource...



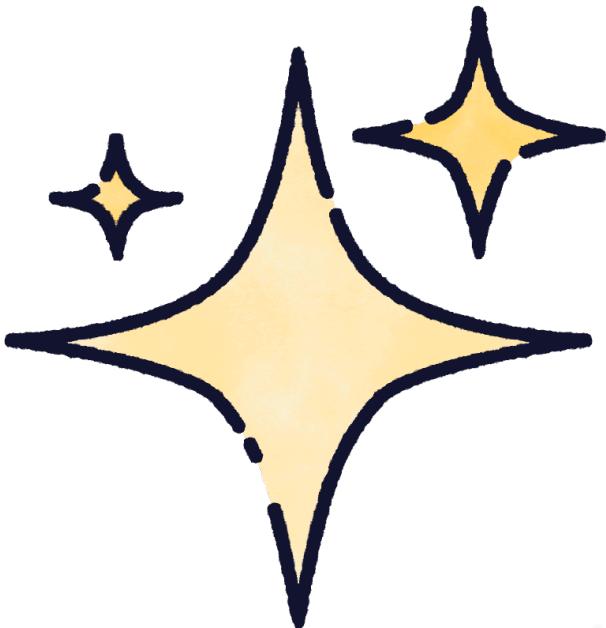
... That Contains Pod Objects



... That Contains Pod Objects



Custom Resources



Custom resources contain custom data

Extends the Kubernetes API

- Provides new resource endpoint
- Stores custom data based on OpenAPI V3 schema

Can contain subresources

Example – Pod subresources:

- /status endpoint
- /exec endpoint



Kubernetes Resource Versioning





More Information

**Application Observability and
Maintenance for CKAD**

Elle Krout



Custom Resource Scope

Namespace

Cluster



**A custom resource is NOT a
custom API; it's only custom
data.**





Controller/Operator

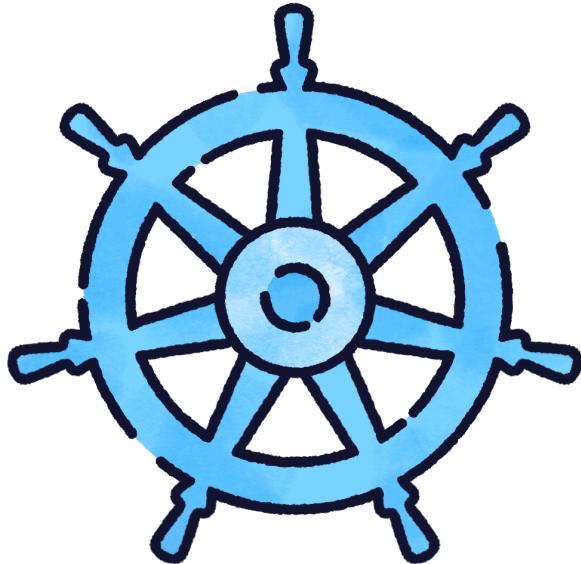
To create a fully-formed declarative API, the custom resource needs to be paired with a custom controller



Exam Tip!

Creating custom controllers/operators is NOT part of the exam.





Custom Resource Definitions

Let you tailor your Kubernetes setup to suit any application's unique needs



Discovering Kubernetes CRDs



View Available Custom Resource Definitions

```
> kubectl get crd
```

NAME	CREATED AT
alertmanagerconfigs.monitoring.coreos.com	2025-07-28T16:43:15Z
alertmanagers.monitoring.coreos.com	2025-07-28T16:43:16Z
podmonitors.monitoring.coreos.com	2025-07-28T16:43:16Z
probes.monitoring.coreos.com	2025-07-28T16:43:17Z
prometheusagents.monitoring.coreos.com	2025-07-28T16:43:17Z
prometheuses.monitoring.coreos.com	2025-07-28T16:43:18Z
prometheusrules.monitoring.coreos.com	2025-07-28T16:43:18Z
scrapeconfigs.monitoring.coreos.com	2025-07-28T16:43:19Z
servicemonitors.monitoring.coreos.com	2025-07-28T16:43:19Z
thanosrulers.monitoring.coreos.com	2025-07-28T16:43:19Z



View Custom Resource Definition Information

> `kubectl explain <crd>`



View Custom Resource Definition Information

```
> kubectl explain alertmanagers.monitoring.coreos.com
```

GROUP: monitoring.coreos.com
KIND: Alertmanager
VERSION: v1

DESCRIPTION:

The `Alertmanager` custom resource definition (CRD) defines a desired [Alertmanager](<https://prometheus.io/docs/alerting>) setup to run in a Kubernetes cluster. It allows to specify many options such as the number of replicas, persistent storage and many more.

For each `Alertmanager` resource, the Operator deploys a `StatefulSet` in the same namespace. When there are two or more configured replicas, the Operator runs the Alertmanager instances in high-availability mode.



View Custom Resource Definition Description

> `kubectl describe crd <crd_name>`



View Custom Resource Definition Description

```
> kubectl describe crd databases.apps.example.com
```

Name: databases.apps.example.com

Namespace:

Labels: <none>

Annotations: <none>

API Version: apiextensions.k8s.io/v1

Kind: CustomResourceDefinition

Metadata:

Creation Timestamp: 2025-07-28T19:09:37Z

Generation: 1

Resource Version: 293215

UID: 37b8f1e5-9698-421a-b451-b1a3e00d2668

Spec:

Conversion:

Strategy: None



View Custom Resource Definition Description

```
> kubectl describe crd databases.apps.example.com
```

Spec:

```
...
Schema:
openAPIV3Schema:
  Properties:
    Spec:
      Properties:
        Engine:
          Description: Database engine (mysql or postgres)
          Type: string
        Replicas:
          Description: Number of replicas for high availability
          Type: integer
```



CRD Discovery Commands



`kubectl get crd`



`kubectl explain <crd>`



`kubectl describe crd <crd>`



Using Kubernetes CRDs



database-crd.yaml

```
apiVersion: apiextensions.k8s.io/v1
kind: CustomResourceDefinition
```

Creating a CRD



database-crd.yaml

```
apiVersion: apiextensions.k8s.io/v1
kind: CustomResourceDefinition
```

Creating a CRD

Set API Version and Kind



database-crd.yaml

```
apiVersion: apiextensions.k8s.io/v1
kind: CustomResourceDefinition
metadata:
  name: databases.apps.example.com
```

Creating a CRD

Set Metadata



database-crd.yaml

```
apiVersion: apiextensions.k8s.io/v1
kind: CustomResourceDefinition
metadata:
  name: databases.apps.example.com
```

Creating a CRD

Set Metadata



database-crd.yaml

```
apiVersion: apiextensions.k8s.io/v1
kind: CustomResourceDefinition
metadata:
  name: databases.apps.example.com
spec:
  group: apps.example.com
```

Creating a CRD

Set Specification Data
(Group)



Creating a CRD

Set Specification Data
(Version)

database-crd.yaml

```
apiVersion: apiextensions.k8s.io/v1
kind: CustomResourceDefinition
metadata:
  name: databases.apps.example.com
spec:
  group: apps.example.com
  versions:
    - name: v1
      served: true
      storage: true
```



Creating a CRD

Set Specification Data
(Version)

database-crd.yaml

```
apiVersion: apiextensions.k8s.io/v1
kind: CustomResourceDefinition
metadata:
  name: databases.apps.example.com
spec:
  group: apps.example.com
  versions:
    - name: v1
      served: true
      storage: true
```



Creating a CRD

Set Specification Data
(Version)

database-crd.yaml

```
apiVersion: apiextensions.k8s.io/v1
kind: CustomResourceDefinition
metadata:
  name: databases.apps.example.com
spec:
  group: apps.example.com
  versions:
    - name: v1
      served: true
      storage: true
```



Creating a CRD

Set Specification Data (Schema)

database-crd.yaml

```
apiVersion: apiextensions.k8s.io/v1
kind: CustomResourceDefinition
metadata:
  name: databases.apps.example.com
spec:
  group: apps.example.com
  versions:
    - name: v1
      served: true
      storage: true
      schema:
        openAPIV3Schema:
          type: object
          properties:
            spec:
              type: object
              properties:
```



Creating a CRD

Set Specification Data (Schema)

database-crd.yaml

```
schema:  
  openAPIV3Schema:  
    type: object  
    properties:  
      spec:  
        type: object  
        properties:  
          engine:  
            type: string  
            description: "Database engine (mysql or  
postgres)"  
          version:  
            type: string  
            description: "Version of the database  
engine"  
          required:  
            - engine  
            - version
```



Creating a CRD

Set Specification Data (Scope)

database-crd.yaml

```
apiVersion: apiextensions.k8s.io/v1
kind: CustomResourceDefinition
metadata:
  name: databases.apps.example.com
spec:
  group: apps.example.com
  versions:
    - name: v1
      served: true
      storage: true
      schema:
        openAPIV3Schema:
          <...snip...>
  scope: Namespaced
```



Creating a CRD

Set Specification Data (Scope)

database-crd.yaml

```
apiVersion: apiextensions.k8s.io/v1
kind: CustomResourceDefinition
metadata:
  name: databases.apps.example.com
spec:
  group: apps.example.com
  versions:
    - name: v1
      served: true
      storage: true
      schema:
        openAPIV3Schema:
          <...snip...>
        scope: Cluster
```



Creating a CRD

Set Specification Data (Names)

database-crd.yaml

```
<...snip...>
spec:
  group: apps.example.com
  versions:
    - name: v1
      served: true
      storage: true
      schema:
        openAPIV3Schema:
          <...snip...>
  scope: Cluster
  names:
    plural: databases
    singular: database
    kind: Database
    shortNames:
      - db
```



Creating a CRD

Set Specification Data (Names)

database-crd.yaml

```
<...snip...>
spec:
  group: apps.example.com
  versions:
    - name: v1
      served: true
      storage: true
      schema:
        openAPIV3Schema:
          <...snip...>
  scope: Cluster
  names:
    plural: databases
    singular: database
    kind: Database
    shortNames:
      - db
```



Creating a CRD

Set Specification Data (Names)

database-crd.yaml

```
<...snip...>
spec:
  group: apps.example.com
  versions:
    - name: v1
      served: true
      storage: true
      schema:
        openAPIV3Schema:
          <...snip...>
  scope: Cluster
  names:
    plural: databases
    singular: database
    kind: Database
    shortNames:
      - db
```



Apply and Confirm CRD

```
> kubectl apply -f database-crd.yaml
```

```
> kubectl get crd
```

NAME	CREATED AT
databases.apps.example.com	2025-07-28T19:09:37Z

```
> kubectl explain databases.apps.example.com
```

GROUP: apps.example.com

KIND: Database

VERSION: v1

DESCRIPTION:

<empty>



Apply and Confirm CRD

```
> kubectl apply -f database-crd.yaml
```

```
> kubectl get crd
```

NAME	CREATED AT
databases.apps.example.com	2025-07-28T19:09:37Z

```
> kubectl explain databases.apps.example.com
```

GROUP: apps.example.com

KIND: Database

VERSION: v1

DESCRIPTION:

<empty>



database-crd.yaml

```
schema:  
  openAPIV3Schema:  
    type: object  
    description: "Create databases..."  
    properties:  
      spec:  
        type: object  
        properties:  
          engine:  
            type: string  
            description: "Database engine (mysql  
or postgres)"  
          version:  
            type: string  
            description: "Version of the  
database engine"
```

Creating a CRD

Set Specification Data (Schema)

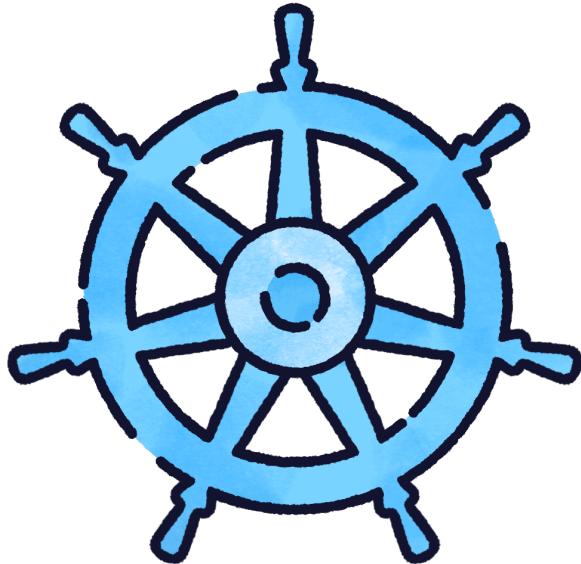


Creating a Resource

database-resource.yaml

```
apiVersion: apps.example.com/v1
kind: Database
metadata:
  name: user-db
spec:
  engine: postgres
  version: "14.1"
```





Custom Resource Definitions

Let you create entirely new resource types within Kubernetes' declarative design schema.



Demo: Creating a Kubernetes Custom Resource



Exam Scenario



Create a custom resource definition for an API that defined cluster maintenance windows, called `maintenancewindow.ops.example.com`, or `mw` for its shortname. This will be for the version 1 alpha version of the API, which should be the only available version

The CRD should contain properties for `startTime` and `endTime` using the date-time format, as well as an input for the reason behind the maintenance

Include property descriptions, as well as a general description of the overall CRD. The scope should be cluster-wide

Apply the CRD, then create a `MaintenanceWindow` resource to test it. Apply the resource



Kubernetes Operators

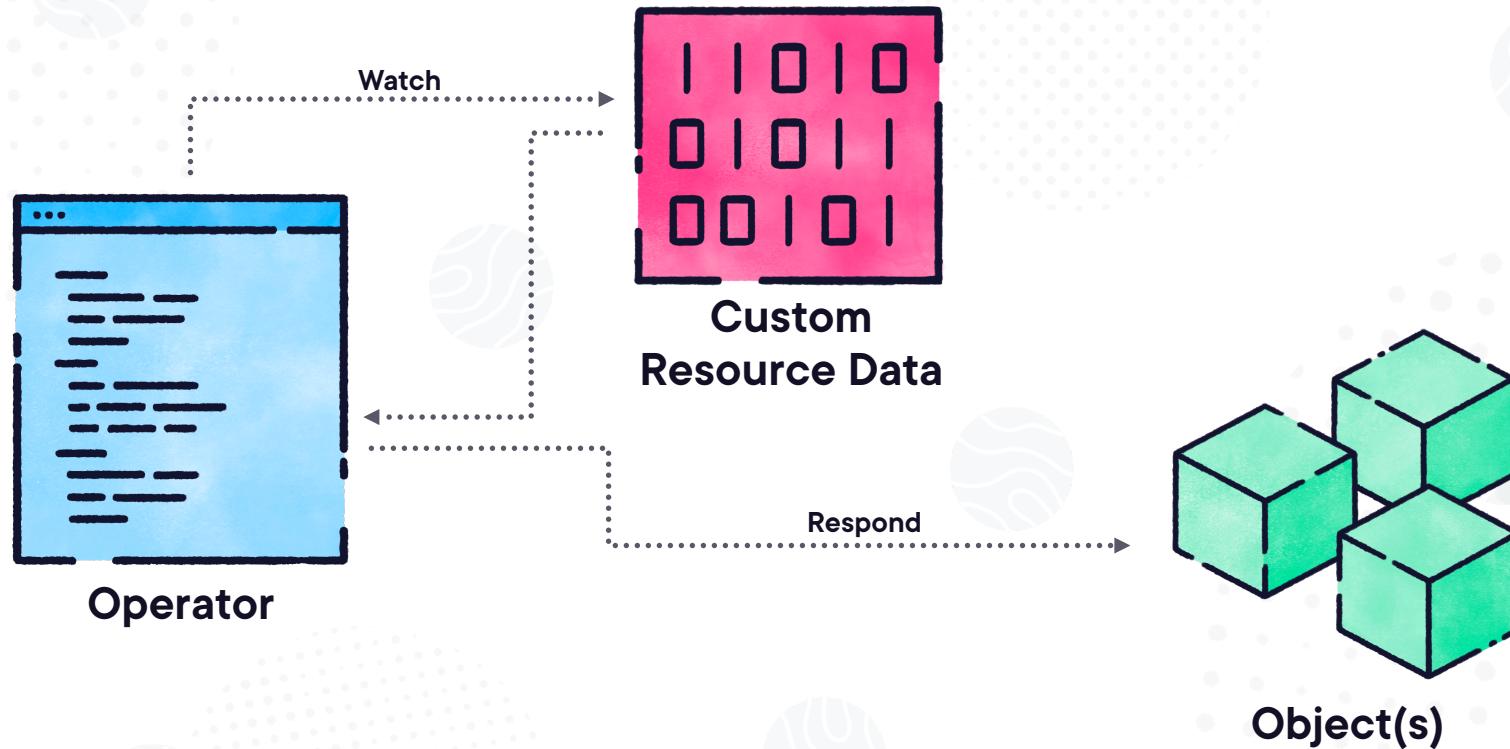


Exam Tip!

Creating custom operators is NOT part of the exam.



Operator Control Loop



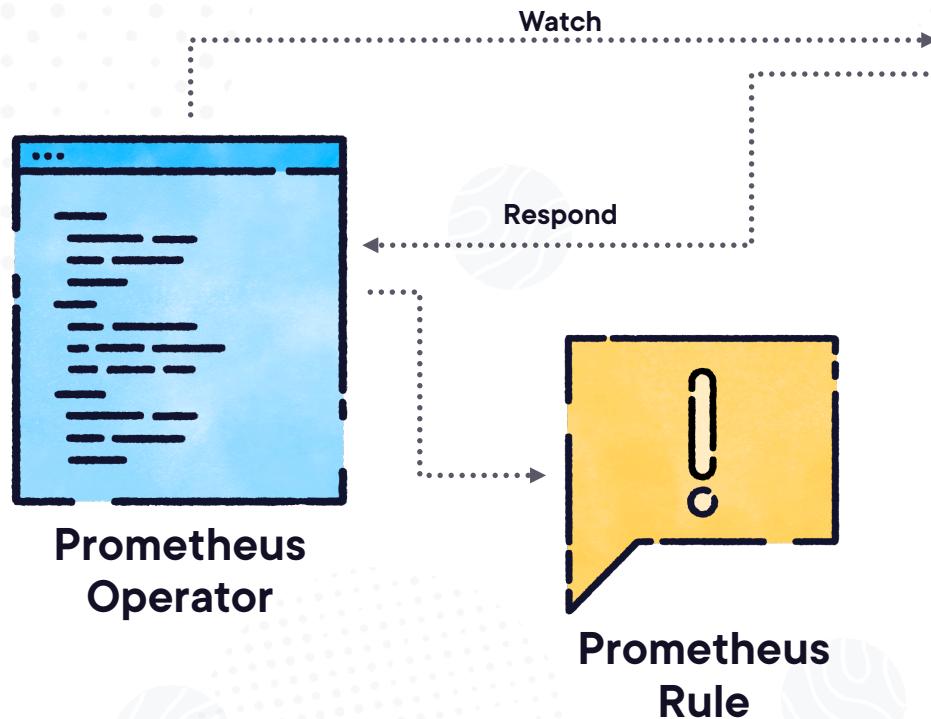
Kubernetes Control Loop

A non-terminating loop that regulates the state of the Kubernetes system.

<https://kubernetes.io/docs/concepts/architecture/controller/>



Prometheus Operator



```
apiVersion: monitoring.coreos.com/v1
kind: PrometheusRule
metadata:
  name: pod-restart-alert
spec:
  groups:
    - name: pod.rules
      rules:
        - alert: PodRestarted
          expr:
            increase(kube_pod_container_status_restarts_total[5m]) > 0
          for: 1m
          labels:
            severity: warning
          annotations:
            summary: "Pod
{{ $labels.pod }} has restarted"
```





This Won't Be on the Exam...

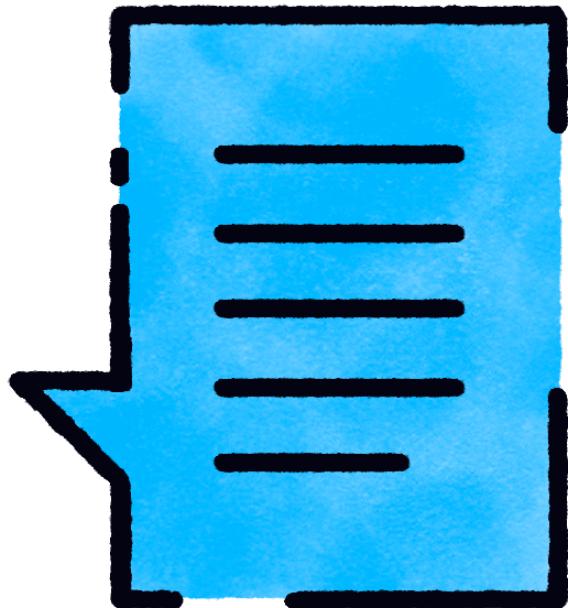
But by understanding the relationship between operators and CRDS, you'll be more than prepared to pass this objective.



Demo: Using Kubernetes Operators



Message Operator



Message API and CRD

Creates a ConfigMap based on provided message

