



Geek University

Evolua seu lado geek!

www.geekuniversity.com.br

Arquivos de Definição do Kubernetes com YAML



YAML



Arquivos de Definição do Kubernetes com YAML

Um arquivo de definição do Kubernetes que é escrito na linguagem YAML sempre tem estes 4 níveis mais altos (top level):

```
apiVersion:  
kind:  
metadata:  
  
spec:
```

Estes campos fazem parte do top level (root level) ou seja, nível raiz do arquivo de definição do kubernetes.

OBS: Estes campos são obrigatórios, então você precisa tê-los em seu arquivo para ser um arquivo de definição do kubernetes válido.



Arquivos de Definição do Kubernetes com YAML

apiVersion:

Neste campo definimos a versão do objeto que estamos criando na API* do Kubernetes.

OBS: Nem todo objeto é necessário definir uma versão para a API, no caso de um POD precisamos definir e podemos fazer isso conforme:

```
apiVersion: v1
kind:
metadata:

spec:
```

* API Server

Kind	Version
POD	v1
Service	v1
ReplicaSet	apps/v1
Deployment	apps/v1

Outras formas de definição do valor para apiVersion de acordo com o tipo de objeto criado.





Arquivos de Definição do Kubernetes com **YAML**

kind:

Define o tipo de objeto do kubernetes que estamos criando.

No nosso exemplo o tipo de objeto é Pod, então podemos definir como:

```
apiVersion: v1
kind: Pod
metadata:

spec:
```

Outros tipos de objetos do kubernetes.



Kind	Version
POD	v1
Service	v1
ReplicaSet	apps/v1
Deployment	apps/v1



Arquivos de Definição do Kubernetes com YAML

metadata:

Metadados são dados pertencentes a determinado objeto, neste caso pertencem ao objeto que está sendo criado.

Alguns dos dados possíveis são:

```
apiVersion: v1
kind: Pod
metadata:
  name: nome-do-pod
  labels:
    app: nome-da-app

spec:
```

OBS: Só podemos definir valores para campos de metadata reconhecidos pelo Kubernetes.

É recomendado recorrer à documentação para conhecer todas as possibilidades.

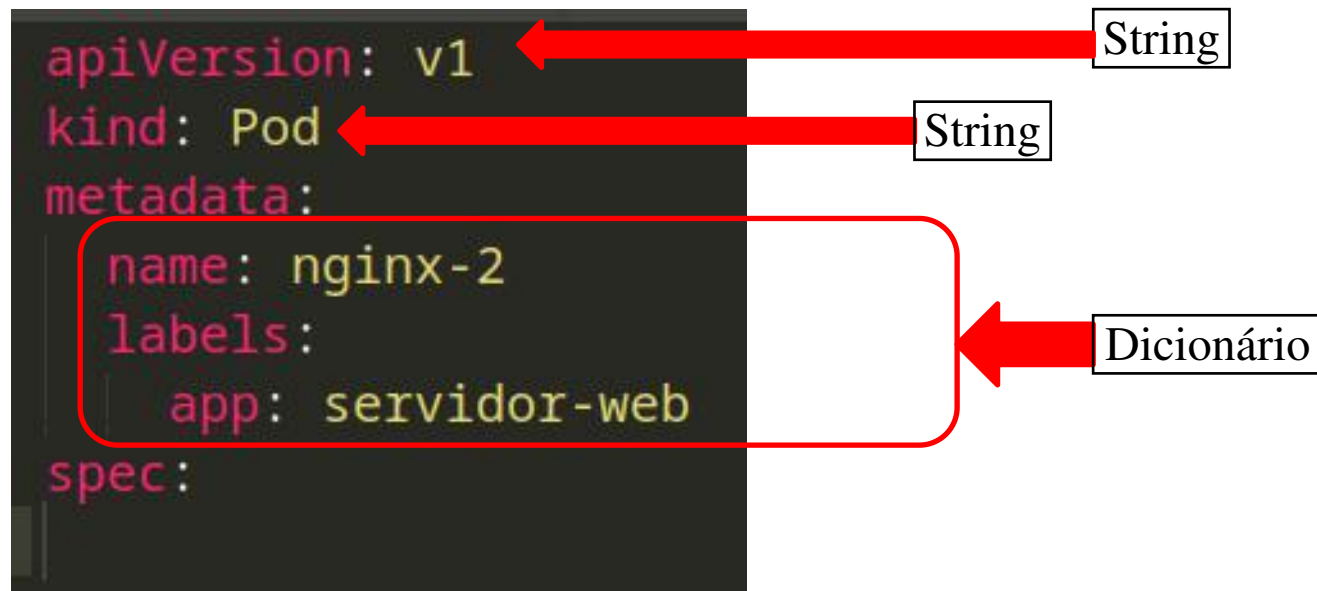


Arquivos de Definição do Kubernetes com YAML

metadata:

Metadados são dados pertencentes a determinado objeto, neste caso pertencem ao objeto que está sendo criado.

Alguns dos dados possíveis são:



Recomendo revisar a seção do curso de Docker onde aprendemos os detalhes da linguagem YAML.



Arquivos de Definição do Kubernetes com YAML

spec:

Este campo é referente as especificações do objeto Kubernetes que está sendo criado.

```
apiVersion: v1
kind: Pod
metadata:
  name: nginx-2
  labels:
    app: servidor-web
spec:
  containers:
    - name: nginx-container
      image: nginx
```




Arquivos de Definição do Kubernetes com YAML

spec:

Este campo é referente as especificações do objeto Kubernetes que está sendo criado.

```
apiVersion: v1
kind: Pod
metadata:
  name: nginx-2
  labels:
    app: servidor-web
spec:
  containers:
    - name: nginx-container
      image: nginx
```

Lista/Array

Lembre-se: Podemos ter múltiplos container em um pod se necessário.



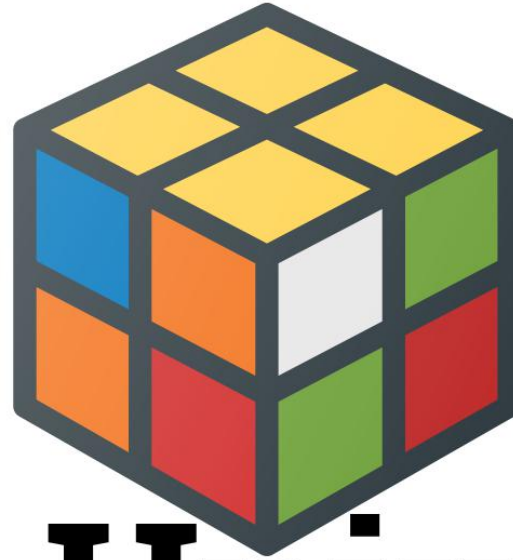
Arquivos de Definição do Kubernetes com YAML

Kubectl

Com o arquivo kubernetes definido, podemos criar o objeto com o comando:

```
kubectl create -f nome-do-arquivo-yaml
```

OBS: Não se preocupe com a execução agora pois na próxima aula iremos realizar o exercício prático desta aula.



Geek University

Evolua seu lado geek!

www.geekuniversity.com.br