# **Kubernetes集群YAML文件详解**

## **概述**

k8s 集群中对资源管理和资源对象编排部署都可以通过声明样式（YAML）文件来解决，也就是可以把需要对资源对象操作编辑到YAML 格式文件中，我们把这种文件叫做资源清单文件，通过kubectl 命令直接使用资源清单文件就可以实现对大量的资源对象进行编排部署了。一般在我们开发的时候，都是通过配置YAML文件来部署集群的。

YAML文件：就是资源清单文件，用于资源编排

### **YAML 支持的数据结构**

#### **对象**

键值对的集合，又称为映射(mapping) / 哈希（hashes） / 字典（dictionary）

# 对象类型：对象的一组键值对，使用冒号结构表示

name: Tom

age: **18**

# yaml 也允许另一种写法，将所有键值对写成一个行内对象

hash: {name: Tom, age: 18}

#### **数组**

# 数组类型：一组连词线开头的行，构成一个数组

People

- Tom

- Jack

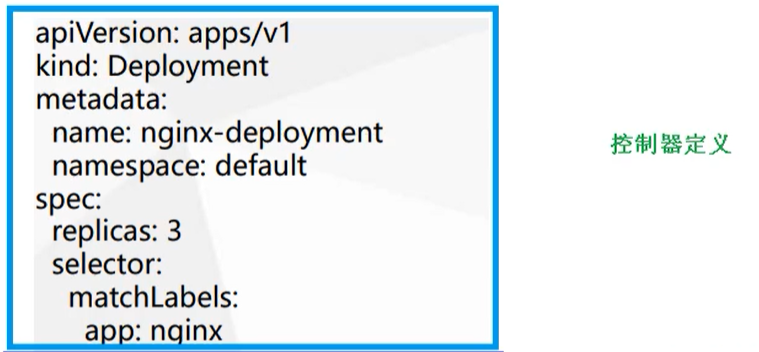
# 数组也可以采用行内表示法

People: [Tom, Jack]

## **YAML文件组成部分**

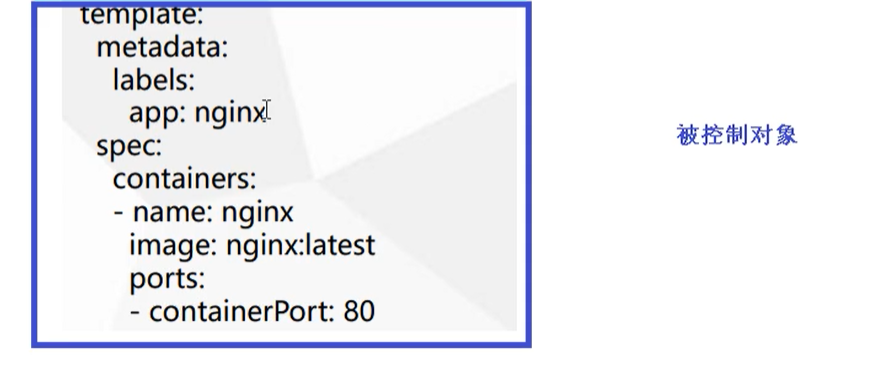
主要分为了两部分，一个是控制器的定义 和 被控制的对象

### **控制器的定义**



### **被控制的对象**

包含一些 镜像，版本、端口等



### **属性说明**

在一个YAML文件的控制器定义中，有很多属性名称

| **属性名称** | **介绍** |
| --- | --- |
| apiVersion | API版本 |
| kind | 资源类型 |
| metadata | 资源元数据 |
| spec | 资源规格 |
| replicas | 副本数量 |
| selector | 标签选择器 |
| template | Pod模板 |
| metadata | Pod元数据 |
| spec | Pod规格 |
| containers | 容器配置 |

## **如何快速编写YAML文件**

一般来说，我们很少自己手写YAML文件，因为这里面涉及到了很多内容，我们一般都会借助工具来创建

### **使用kubectl create命令**

这种方式一般用于资源没有部署的时候，我们可以直接创建一个YAML配置文件

# 尝试运行,并不会真正的创建镜像

kubectl create deployment web --image=nginx -o yaml --dry-run

或者我们可以输出到一个文件中

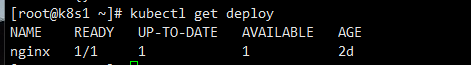
kubectl create deployment web --image=nginx -o yaml --dry-run > hello.yaml

然后我们就在文件中直接修改即可

### **使用kubectl get命令导出yaml文件**

可以首先查看一个目前已经部署的镜像

kubectl get deploy



然后我们导出 nginx的配置

kubectl get deploy nginx -o=yaml > nginx.yaml

然后会生成一个 nginx.yaml 的配置文件