

资源清单格式

```
apiVersion: group/apiversion # 如果没有给定 group 名称, 那么默认为 core, 可以使用 kubectl api-versions # 获取当前 k8s 版本上所有的 apiVersion 版本信息( 每个版本可能不同 )
kind:          #资源类别
metadata:     #资源元数据
  name
  namespace
  labels
  annotations # 主要目的是方便用户阅读查找
spec: # 期望的状态 (disired state)
status: # 当前状态, 本字段有 Kubernetes 自身维护, 用户不能去定义
```

资源清单的常用命令

获取 apiversion 版本信息

```
[root@k8s-master01 ~]# kubectl api-versions
admissionregistration.k8s.io/v1beta1
apiextensions.k8s.io/v1beta1
apiregistration.k8s.io/v1
apiregistration.k8s.io/v1beta1
apps/v1
.....(以下省略)
```

获取资源的 apiVersion 版本信息

```
[root@k8s-master01 ~]# kubectl explain pod
KIND:      Pod
VERSION:   v1
.....(以下省略)

[root@k8s-master01 ~]# kubectl explain Ingress
KIND:      Ingress
VERSION:   extensions/v1beta1
```

获取字段设置帮助文档

```
[root@k8s-master01 ~]# kubectl explain pod
KIND:      Pod
VERSION:   v1

DESCRIPTION:
  Pod is a collection of containers that can run on a host. This resource is
  created by clients and scheduled onto hosts.

FIELDS:
  apiVersion    <string>
  .....
```

字段配置格式

```
apiVersion <string>           #表示字符串类型
metadata <Object>             #表示需要嵌套多层字段
labels <map[string]string>    #表示由k:v组成的映射
finalizers <[]string>         #表示字符串列表
ownerReferences <[]Object>   #表示对象列表
hostPID <boolean>            #布尔类型
priority <integer>           #整型
name <string> -required-     #如果类型后面接 -required-, 表示为必填字段
```

通过定义清单文件创建 Pod

```
apiVersion: v1
kind: Pod
metadata:
  name: pod-demo
  namespace: default
  labels:
    app: myapp
spec:
  containers:
  - name: myapp-1
    image: hub.atguigu.com/library/myapp:v1
  - name: busybox-1
    image: busybox:latest
    command:
    - "/bin/sh"
    - "-c"
    - "sleep 3600"
```

```
kubectl get pod xx.xx.xx -o yaml
```

<!--使用 -o 参数 加 yaml, 可以将资源的配置以 yaml的格式输出出来, 也可以使用json, 输出为json格式-->

