# K8S简单了解

1. **Docker**
   1. **什么是Docker**
      1. Docker是指容器化技术，可以理解为轻巧化、模块化的Linux虚拟机。
   2. **Docker能用来做什么。**
      1. 使用docker可以对“运行环境”、“操作步骤”等进行高效的复制，提高效率、避免环境不一致导致的错误等。

例如需要部署一个mysql服务，执行一句命令即可

docker run -itd --name mysql -p 3306:3306 -e MYSQL\_ROOT\_PASSWORD=123456 mysql

需要部署一个redis服务，执行一句命令即可：

docker run -itd --name redis-test -p 6379:6379 redis

* 1. **Docker怎么用**
     1. 环境安装

1. linux安装

2. windows安装

<https://www.runoob.com/docker/windows-docker-install.html>

1. 配置仓库镜像加速

{"registry-mirrors": ["https://hcrueet9.mirror.aliyuncs.com"]}

* + 1. 名词解释



* 1. **实际应用场景**
     1. 编写程序
     2. 编写Dockerfile
     3. 制作镜像
     4. 推送镜像到仓库
     5. 生产环境从仓库拉取镜像，创建并运行容器
  2. **常用操作命令**
     1. 镜像操作



* + 1. 容器操作



1. **K8S**
   1. **K8S是什么**

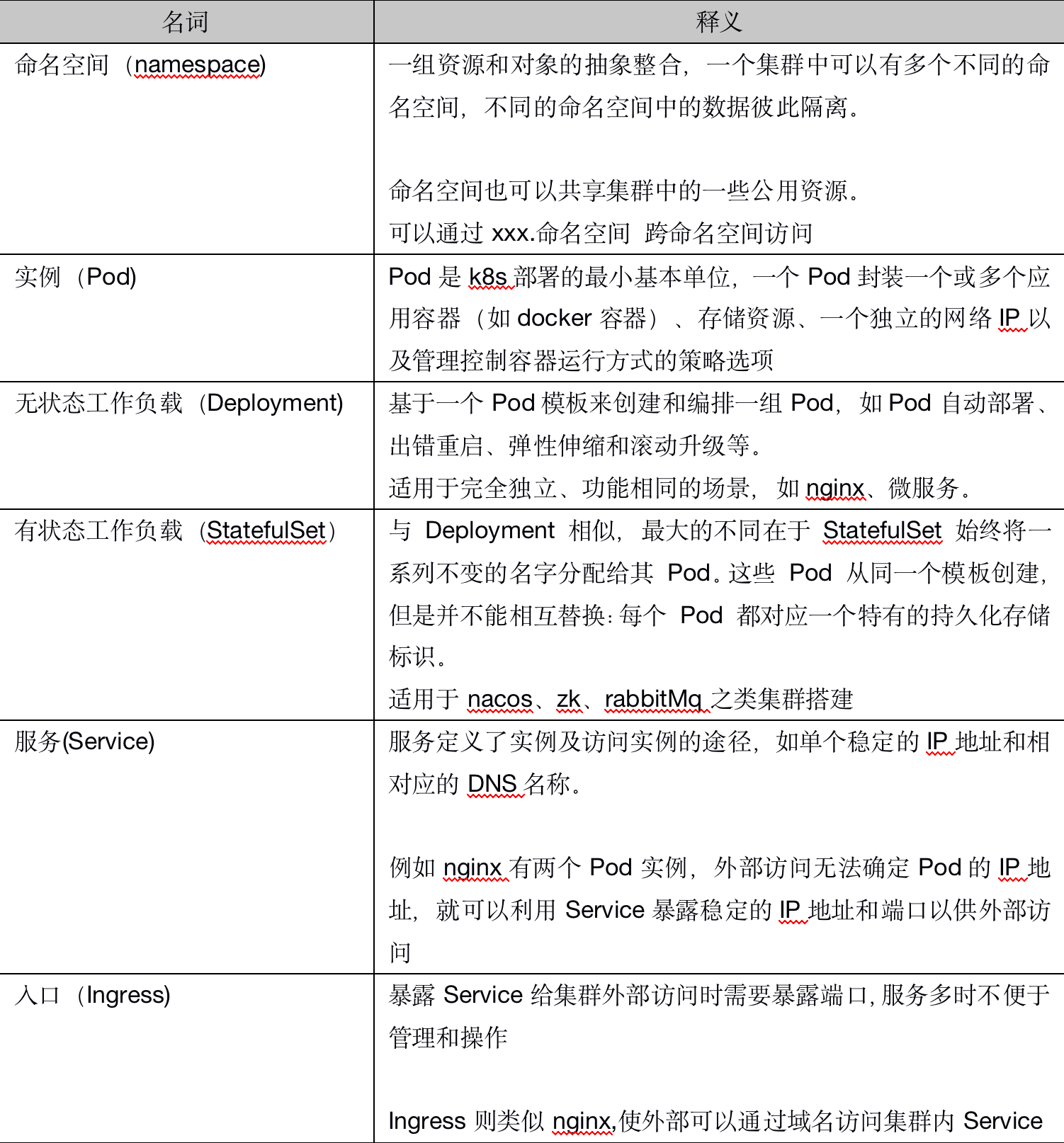
Kubernetes(k8s)是Google开源的容器集群管理系统（谷歌内部:Borg）。在Docker技术的基础上，为容器化的应用提供部署运行、资源调度、服务发现和动态伸缩等一系列完整功能，提高了大规模容器集群管理的便捷性。

* 1. **K8S能用来做什么**

1. 快速部署应用；
2. 快速扩展应用（自动动态伸缩）；
3. 无缝对接滚动升级或回滚；
4. 节省资源，优化硬件资源的使用。
5. 故障自愈
   1. **K8S怎么用**
      1. 安装

<https://kuboard.cn/install/install-kubernetes.html>

* + 1. 重要概念解释



* + 1. 其他概念

<https://kuboard.cn/learning/>

* 1. **实际应用场景**
     1. 制作镜像（docker或其他容器镜像）
     2. 编写部署配置文件
        1. 负载类型(Deployment,StatefulSet）
           1. 属于哪个命名空间
           2. 工作负载名称
           3. 标签（用来筛选和绑定service等等）
           4. 使用什么容器镜像（docker镜像）
           5. 定义镜像所需变量（如启动参数，存储路径等）
           6. 启动几个实例（pod)
           7. 更新策略(如滚动更新等）
           8. 持久化策略（默认Pod删除后文件内容即删除，如需保留需要另行配置持久化）
           9. 等等其他
        2. 设置服务访问地址（service，集群内DNS名)
           1. 端口映射（Pod暴露端口与外部访问时访问端口的映射）
           2. 模式（主要）

ClusterIP: 默认值。通过集群内部的一个 IP 地址暴露 Service，只在集群内部可以访问

NodePort: 通过每一个节点上的的静态端口（NodePort）暴露 Service，同时自动创建 ClusterIP 类型的访问方式

在集群内部通过 $(ClusterIP): $(Port) 访问

在集群外部通过 $(NodeIP): $(NodePort) 访问集群外部调用地址（域名）

* + 1. master机器执行部署

在master机器上应用写好的配置文件: kubectl apply -f xxxx.yml

* + 1. 可视化面板管理（模拟体验地址）

<http://demo.kuboard.cn/dashboard?k8sToken=eyJhbGciOiJSUzI1NiIsImtpZCI6InZ6SzVqZFNJOXZFMmxQSkhXamNBcFY4RU9FR0RvSUR5bzJIY0NwVG1zODQifQ..kgwTa6t00gNC0vgr6HOvCqkDghPcW-jVDg-_K6WLy97ppb9jvaqVz-AxXzF7mJqXnNetbJw-8-x_L3ogSsDlTKmRucao96VA2tPKxel8pM04J8MU0ZmYgWhTJelibbxmQK3jwGM4x32bckOOvmtumcXdsBRN0z1SZ1iu4H0VoaswhfoFS4ZJKoe61xyqoDhQx4RLCVJh_-Uctd5RCcPLWFEk-BHqC8vUTy8QcRst6RIIozQdTqsv7Xs6bH6dHrHFS--eVVTH2orQdm8znuUFhlqFOOjmCIMzIlaUQC_SO9URIGYOs0jrk27N9KC0HvQ5dLgFmwyNJ0Gu7cYi23NP1A>