Pods, Service and Ingress

server → node

pod is the smallest unit in k8s
pod is an abstraction over container
usually 1 application per pod
each pod gets its own IP address
pod components are ephemeral which means that they can die very easily and when that happens, new IP address is assigned which is inconvinient and because of that another component called of k8s, service is used

service is a static/permanent IP address that can be attached to each pod, the lifecycle of the pod and the service is not connected, so if the pod dies, the IP is still there

external service opens communication from external sources
internal service is one, which is not open for communication from external sources

generally, IP address and port is used to access the application, but that is not very convinient for the end product, so to solve this, we have ingress which is used for forwarding. So instead of service, the request first goes to Ingress and it does the forwarding to the service.

ConfigMap and Secret

ConfigMap is an external configuration of an application

Secret is used to store secret data, passwords, certificates, etc and is stored in base64 encoded data. We can use the secrets as environment variables or as a properties file.

Volumes

It is similar to docker volumes, and can be storage on local machine or remote, outside of the k8s cluster

!K8s doesn't manage data persistance.

Deployment and Stateful Set

Service has 2 functionalities: permanent IP, load balancer

Deployments: blueprint for application pods (abstraction of pods).

In practice, we don't work with pods, or create pods, we create deployments.

!Databases can't be replicated via Deployment and the reason is that DB have state(data), so if we have replicas of DB, we need to manage which replica is writing and reading to avoid data inconsistencies. This is the reason we use Stateful Set for Databases or any other entity which has state

Deployments for stateless apps, StatefulSet for stateful apps or databases