In the previous tutorial we have seen that **Pods do have their own unique IP across the cluster, those IP’s are not exposed outside Kubernetes**. Taking this into account that over time Pods may be terminated, deleted or replaced by other Pods, we need a way to let other Pods and applications automatically discover each other. Kubernetes addresses this by grouping Pods in Services. A Kubernetes Service is an abstraction layer which defines a logical set of Pods and enables external traffic exposure, load balancing and service discovery for those Pods.

This abstraction will allow us to expose Pods to traffic originating from outside the cluster. **Services have their own unique cluster-private IP address and expose a port to receive traffic.**

If you choose to expose the service outside the cluster, the options are:

LoadBalancer - provides a public IP address (what you would typically use when you run Kubernetes on GKE or AWS)

NodePort - exposes the Service on the same port on each Node of the cluster using NAT (available on all Kubernetes clusters, and in Minikube)

