Basic Terms: System Parts

- Kubernetes: The whole orchestration system, aka "cluster"
  - Short names: Written K8s. Sounds like "k-eights" or "kates"
- Kubectl: CLI to configure Kubernetes and manage apps
  - Sounds, like "cube control" or "cube cuttle"
- Node: Single server in the Kubernetes cluster
- Kubelet: Kubernetes agent running on nodes
- Control Plane: The management nodes
  - Includes services that manage the cluster
  - Includes API server, scheduler, controller manager, etcd, and more

Install Kubernetes Locally

- Kubernetes is a series of containers, CLI's, and configurations
- Many ways to install. Let's focus on easiest for learning
- Docker Desktop: Enable in settings
  - Sets up everything inside Docker's existing Linux VM
- Rancher Desktop: In case you can't use Docker Desktop
- Your Own Linux Host or VM: MicroK8s or K3s
  - Installs Kubernetes right on the OS

Kubernetes In A Browser

- Try http://play-with-k8s.com or katacoda.com in browser
  - Easy to get started
  - Doesn't keep your environment

Docker Desktop

- Configures Kubernetes Control Plane on top of Docker runtime
- Manages kubectl install and certs for connecting to K8s API
- Easily install, disable, and remove from Docker GUI

Rancher Desktop

- Similar feature set to Docker Desktop
- Usually chosen when you can't run Docker Desktop
- Be sure to enable dockerd (moby) and Kubernetes in settings

MicroK8s

- Installs Kubernetes (without Docker Engine) directly on Linux
- Uses snap (rather then apt or yum) for install
- Control the MicroK8s service via microk8s commands
- kubectl accessible via microk8s kubectl
- Add CoreDNS for services to work
  - microk8s enable dns
- Add an alias to your shell (.bash_profile)
  - alias kubectl="microk8s kubectl"

Kubernetes Container Abstractions

- **Pod**: one or more containers running together on one Node
  - Basic unit of deployment. Containers are always in pods
- **Controller**: For creating/updating pods and other objects
  - Many types of Controllers inc. Deployment, ReplicaSet, StatefulSet, DaemonSet, Job, CronJob, etc.
- **Service**: network endpoint to connect to a pod
- **Namespace**: Filtered group of objects in cluster
- Secrets, ConfigMaps, and more