Here's a chapter-wise learning plan for Kubernetes, arranged from the basics to more advanced topics, to help you gradually build your knowledge. This will help you progress step-by-step from easy concepts to more complex ones:

Chapter 1: Introduction to Kubernetes

What is Kubernetes? Overview of Container Orchestration.

Kubernetes Architecture: Nodes, Master, Worker, Control Plane.

Key Components: Pods, Nodes, Deployments, Services.

Setting Up Kubernetes Locally: minikube, kind, or Docker Desktop.

Chapter 2: Pods and Basic Workloads

Understanding Pods: The basic building block of Kubernetes.

Writing Your First Pod YAML.

Creating, Viewing, and Deleting Pods.

Multi-container Pods and Sidecar Containers.

Chapter 3: Deployments and ReplicaSets

What is a Deployment? Managing Updates and Rollbacks.

Writing Deployment YAMLs: spec, replicas, template.

ReplicaSets: Scaling and Managing Multiple Pods.

Rolling Updates and Rollbacks with Deployments.

Hands-on: Scaling a Deployment and checking how Pods adjust.

Chapter 4: Services and Networking

Understanding Cluster Networking.

Service Types: ClusterIP, NodePort, LoadBalancer.

Writing Service YAMLs and Exposing Deployments.

Ingress: Routing External Traffic to Services.

DNS in Kubernetes: How Pods communicate internally.

Chapter 5: ConfigMaps and Secrets

Using ConfigMaps to Manage Configuration Data.

Creating and Using Secrets for Sensitive Data.

Storing ConfigMaps and Secrets in YAML files.

Mounting ConfigMaps and Secrets as Environment Variables or Volumes.

Chapter 6: Storage in Kubernetes

Introduction to Persistent Volumes (PV) and Persistent Volume Claims (PVC).

Understanding Storage Classes.

Writing YAMLs for PVs and PVCs.

Dynamic Volume Provisioning.

Using StatefulSets for applications that require persistent storage.

Chapter 7: StatefulSets and DaemonSets

When to Use StatefulSets vs. Deployments.

Understanding StatefulSets: Managing Stateful Applications.

Writing StatefulSet YAMLs.

DaemonSets: Running a Pod on Every Node.

Writing YAMLs for DaemonSets.

Chapter 8: Monitoring, Logging, and Debugging

Built-in Kubernetes Monitoring Tools: kubectl commands (logs, top).

Setting up Prometheus and Grafana for monitoring.

Using Fluentd or Elastic Stack (ELK) for logging.

Debugging Pods: Common Issues and Solutions.

Chapter 9: Security in Kubernetes

Basic Concepts: Role-Based Access Control (RBAC).

Writing YAMLs for Roles, RoleBindings, and ServiceAccounts.

Network Policies: Securing Pod Communication.

Pod Security Policies (PSP) and Pod Security Standards (PSS).

Using Secrets for securing sensitive data.

Chapter 10: Advanced Networking with CNI

Container Network Interface (CNI) Overview.

Using Calico, Flannel, or Weave for Network Policies.

Service Mesh with Istio: Traffic Management and Security.

Understanding and Using Ingress Controllers (NGINX, Traefik).

Chapter 11: Helm and Package Management

Introduction to Helm: Why Use Helm?

Installing Helm and Writing Your First Helm Chart.

Using Helm Repositories and Deploying Pre-built Charts.

Customizing Helm Charts with values.yaml.

Helm vs. Kustomize: Use Cases.

Chapter 12: Autoscaling and Load Balancing

Horizontal Pod Autoscaler (HPA): Scaling based on CPU/Memory.

Cluster Autoscaler: Adding/Removing Nodes Automatically.

Writing YAMLs for HPA.

Load Balancing using Kubernetes Services.

Chapter 13: Operators and Custom Resource Definitions (CRDs)

Understanding CRDs and extending Kubernetes APIs.

Writing Basic CRD YAMLs.

Introduction to Kubernetes Operators.

Using Existing Operators: Database Operators, Monitoring Operators.

Building a simple Operator using the Operator SDK.

Chapter 14: CI/CD with Kubernetes

Integrating Kubernetes with CI/CD tools like Jenkins, GitLab, or ArgoCD.

Using kubectl in CI/CD pipelines.

Writing YAML for Deployments in CI/CD Pipelines.

Blue/Green and Canary Deployments using ArgoCD.

Chapter 15: Advanced Kubernetes Topics

Managing Multi-cluster Deployments.

Using Kubernetes Federation for Cross-cluster Management.

Backup and Disaster Recovery Strategies.

Writing YAMLs for Backup solutions like Velero.

Kubernetes in Production: Best Practices.

Chapter 16: Kubernetes Cloud Services (Optional)

Introduction to Kubernetes Services in the Cloud: AKS (Azure), EKS (AWS), GKE (Google).

Differences between Self-managed and Managed Kubernetes.

Writing YAMLs for cloud-specific resources.

Integrating Cloud Storage and Databases with Kubernetes.

This roadmap will guide you through the learning process, starting with the basics and gradually moving into more complex concepts. Remember to practice as you go, as hands-on experience is key to mastering Kubernetes!