Container Orchestration Fundamentals

- **Need for Orchestration:**
  - Scaling applications.
  - Self-healing of workloads.
  - Load balancing.
  - Resource optimization.
- **Kubernetes as an Orchestration Tool:**
  - Managing containerized workloads.
  - Scheduling containers on nodes.
  - Ensuring desired state (via Deployments, ReplicaSets).
- **Core Orchestration Features in Kubernetes:**
  - Automatic scaling (Horizontal Pod Autoscaler).
  - Rolling updates and rollbacks.
  - Service discovery and networking.
  - Persistent storage management.
- **Workload Orchestration Concepts:**
  - Pods as the smallest deployable unit.
  - Running multiple containers in a Pod (e.g., sidecar pattern).

Revision #2
Created 18 November 2024 21:51:24 by Admin
Updated 21 November 2024 19:59:20 by Admin