INDUSTRIAL ASSISMENT-12

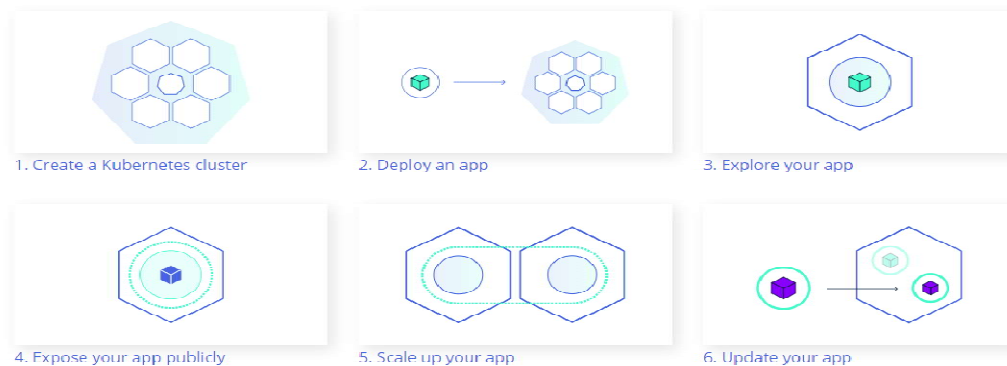## EXPLAIN THE KUBERNETES BASICS

Kubernetes Basics

This tutorial provides a walkthrough of the basics of the Kubernetes cluster orchestration system. Each module contains some background information on major Kubernetes futures and concepts, and includes an interactive online tutorial. These interactive tutorials let you manage a simple cluster and its containerized applications for yourself.

- Using the interactive tutorials, you can learn to:
- Deploy a containerized application on a cluster.
- Scale the deployment.
- Update the containerized application with a new software version.
- Debug the containerized application.

The tutorials use Kat coda to run a virtual terminal in your web browser that runs Minikube, a small-scale local deployment of Kubernetes that can run anywhere. Each interactive tutorial runs directly out of your web browser itself.

Kubernetes Basics Modules

1. Create a Kubernetes cluster
2. Deploy an app
3. Explore your app
4. Expose your app publicly
5. Scale up your app
6. Update your app

✓ **What Kubernetes do for you?**
  With modern web services, users expect applications to be available 24/7, and developers expect to deploy new versions of those applications several times a day. Containerization helps package software to serve these goals, enabling applications to be released and updated without downtime. Kubernetes helps you make sure those containerized applications run where and when you want, and helps them find the resources and tools they need to work. Kubernetes is a production-ready, open source platform designed with

Google's accumulated experience in container orchestration, combined with best-of-breed ideas from the community.

- ✓ **API Reference**
  - Glossary- a comprehensive, standardized list of Kubernetes terminology
  - Kubernetes API Reference
  - One-page API Reference for Kubernetes v1.26
  - Using The Kubernetes API – overview of the API for Kubernetes.
  - API access control – details on how for Kubernetes controls API access
  - Well-Known Labels, Annotations and Taints

- ✓ **Components**
  - kubelet – The primary agent that runs on each node. The Kubelet takes a set of Pod Specs and ensure that the described containers are running and healthy.
  - Kube-apiserver REST API that validates and configures data for API objects such as pods, services, replication controllers.
  - Kube-controller-manager – Daemon that embeds the core control loops shipped with Kubernetes.
  - Kube-proxy – Can do simple TCP/UDP stream forwarding or round-robin TCP/UDP forwarding across a set of back-ends.
  - Kube-scheduler – Scheduler that manages availability, performance, and capacity.
  - Scheduler Policies
  - Scheduler Profiles
  - List of ports and protocols that should be open on control plane and worker nodes

- ✓ **Officially supported client libraries**
  To call the Kubernetes API from a programming language, you can use client libraries. Officially supported client libraries:
  - Kubernetes Go client library
  - Kubernetes Python client library
  - Kubernetes Java client library
  - Kubernetes JavaScript client library
  - Kubernetes C# client library
  - Kubernetes Haskell client library