HOMEWORK 2

1. **What is Deployment in IT?**

   - **Definition**: Deployment is the process of making an application or software available for users by moving it from development (where it's built) to production (where it can be used).
   - **Simple Example**: Imagine you bake a cake in your kitchen (development). Deployment is like putting that cake on the table for people to eat (production).

2. **What is Cloud Native?**

   - **Definition**: Cloud Native refers to building applications designed to run efficiently in cloud environments, making them scalable and resilient.
   - **Simple Example**: It's like opening a pizza delivery business that only works online, allowing you to handle more orders as more people use your service without needing a physical shop.

3. **What is a YAML File and Its Role in Docker?**

   - **Definition**: A YAML (Yet Another Markup Language) file is a human-readable file format used to configure services and applications. In Docker, it's used to define how different containers should work together.
   - **Simple Example**: A YAML file is like a shopping list that tells you exactly what ingredients (containers) you need and how to cook them together to make a meal (your app).

4. **What is an IP Address?**

   - **Definition**: An IP address is a unique identifier for a device on a network, used to locate and communicate with other devices.
   - **Simple Example**: It's like the address of your house. When someone sends you a letter, they need to know your address. Similarly, computers need an IP address to send data.

5. **What is DNS in IT?**

   - **Definition**: DNS (Domain Name System) is a system that translates human-readable domain names (like google.com) into IP addresses so computers can connect to websites.
   - **Simple Example**: It's like a contact list on your phone. Instead of remembering your friend's number, you just tap their name, and your phone dials the correct number for you.

6. **What is a Namespace?**

- **Definition**: A namespace is a way to organize and isolate resources, so different systems or projects don't interfere with each other.
- **Simple Example**: Think of it like having different drawers in a dresser. You can have the same kind of clothes (files) in each drawer, but they are kept separate and don't mix.

7. **What is the Difference Between Docker Swarm and Kubernetes?**

- **Definition**: Docker Swarm and Kubernetes are tools used to manage clusters of containers. Docker Swarm is simpler and easier to use for small applications, while Kubernetes is more powerful and better for complex, large-scale systems.
- **Simple Example**: Docker Swarm is like managing a small group of delivery bikes. Kubernetes is like managing an entire delivery fleet with trucks, bikes, and drones for a big company.

8. **What Does Scalable Application Mean?**

- **Definition**: A scalable application is one that can handle increasing amounts of work or users by adding more resources (such as servers) without changing the core design.
- **Simple Example**: It's like starting a lemonade stand that can serve 10 people. As more people come, you add more stands and workers to serve more customers without changing the recipe.

9. **Can Kubernetes Run Without Docker?**

- **Definition**: Yes, Kubernetes can run without Docker. Kubernetes needs a container runtime, and while Docker was once the most common, Kubernetes now uses other runtimes like containerd or CRI-O.
- **Simple Example**: It's like baking bread. You used to always use one type of oven (Docker), but now you can use different kinds of ovens (container runtimes) to do the same job.

10. **Can Docker Run Without Kubernetes?**

- **Definition**: Yes, Docker can run without Kubernetes. Docker manages individual containers, while Kubernetes is only needed when you want to manage many containers across multiple servers.
- **Simple Example**: It's like cooking one meal in your kitchen. You don't need a manager (Kubernetes) unless you're running a restaurant with many chefs and dishes at once.