1. Define Scalability and Availability. Provide examples of each.

2. Explain the role of Continuous Integration (CI) in DevOps. How does Jenkins facilitate CI/CD pipelines?

3. How does Puppet help in Configuration Management? Demonstrate with a basic Puppet script.

4. Compare REST and SOAP. Highlight their differences with real-world use cases.

5. Discuss the role of **Docker Compose** in managing multi-container applications. Provide an example configuration.

6. What is Jenkins, and how do you create a Jenkins job?

7. Design a containerized application using Docker. Outline the key steps involved.

9. Describe the **Git branching strategy** and its importance in a CI/CD pipeline with **Jenkins**.

8. Explain Web Services and their role in modern application architecture.

9. Explain the DevOps life cycle and its key phases.

10. Explain **Puppet's architecture**. How does the Puppet Master-Agent model work?

11. What are the advantages of using **Jenkins Pipelines** over traditional freestyle jobs?

12. Explain the importance of Automation in DevOps. How does automation improve scalability and availability?

13. Compare AWS, Azure, and Google Cloud for SysOps. Highlight their advantages and limitations.

14. Describe the core components of a REST API and how it differs from a traditional Web Service. Provide an example API request and response.

15 How does Puppet enforce system consistency? Provide an example Puppet script to install and configure a web server.

16. Explain the process of setting up a Jenkins pipeline for automated testing. How does Jenkins integrate with version control systems like Git?

17. How does Docker improve application portability ? Demonstrate how to create a multi-container application using Docker Compose.

18. Compare Maven and Gradle as build tools. How does Maven simplify dependency management? Provide an example `pom.xml` file.

19. What is the difference between Git Merge and Git Rebase? Explain with an example when to use each approach.

20. What are the benefits of **running Jenkins inside a Docker container**? How do you set it up?

21. How can **Jenkins be configured to trigger builds automatically** based on code commits, pull requests, or scheduled jobs?

22. Compare **Puppet with Ansible**. What are the key differences, and when would you use one over the other?

23. Discuss the difference between **Docker ENTRYPOINT and CMD**. Provide examples of when to use each.

24. What is **Infrastructure as Code (IaC)**? How does it help in DevOps, and what are its key benefits?

25. Explain the **difference between Continuous Integration, Continuous Delivery, and Continuous Deployment** with real-world examples