# **DEV-OPS: Preparation**

1. What are some ways to increase client-side performances?
2. Can you explain the significance of a primary/replica database architecture?

Ans: Replication is the process of copying and maintaining database objects in multiple databases that make up a distributed database system. Replication can improve the performance and protect the availability of applications because alternate data access options exist. For example, an application might normally access a local database rather than a remote server to minimize network traffic and achieve maximum performance. Furthermore, the application can continue to function if the local server experiences a failure, but other servers with replicated data remain accessible. Oracle supports two different forms of replication: basic and advanced replication.

1. What are some caching strategies you can implement to increase performances?
2. What are the advantages and disadvantages of implementing a Micro-Services Architecture?
3. What are the differences between vertical and horizontal scalability?
4. What languages are you most comfortable with?
5. What are the differences between a relational and non-relational database?
6. When do we optimize the database?
7. Can you give some way to optimize the performance of a database?
8. What does A.C.I.D mean?
9. How can you customize the query plan to increase throughput?
10. Explain how you would do Database continuous delivery?
11. How to view running processes in Linux?
12. How do you check server uptime?
13. How do you Start/Stop Services?
14. How do you display shell’s environment variables?
15. What adding #!/bin/bash at the top of a script do?
16. What does a& after a command do?
17. What does piping command mean?
18. What distributions have you used on server?
19. What is the importance of SSL?
20. What is a SQL injection?
21. What is cross-site scripting(XSS)?
22. Why shouldn’t you roll your own crypto?
23. How are passwords stored on databases?
24. What is a man-in-the-middle attack?
25. How do you safely manage environment variables in cloud environment?
26. Which cloud providers are you familiar with?
27. What are the benefits of using a cloud provider?
28. Have you managed production systems in the past?
29. Which Configuration Management tools are you most comfortable with? Examples include Puppet, Chef, Ansible and Salt.
30. How does Docker improve scalability, distributed computing, and efficiency vs traditional cloud virtual machines?
31. List some Docker use cases?
32. What is vagrant?
33. How Vagrant helps making development environment easy?
34. Can you name some hypervisors on which Vagrant provides wrapper over?
35. Does Vagrant also provide wrapper of dev-ops tools?
36. Does Vagrant supports Docker containers also?
37. Which server environments Vagrant provides support?
38. What all steps required to configure Vagrant?
39. What are the commands for making Vagrant up and running?
40. What are the benefits of using Vagrant?
41. What is the difference between Linux and Unix?

**OS**

1. What is the difference between Linux and Unix?
2. What's a KVM?
3. How would you make sure a service starts on an OS of your choice?
4. Here's a terminal. What flavor of Linux is running?
5. Write a command to delete all empty file under a directory.
6. Kill all the procs by a particular user without using pkill
7. What is Active Directory? How do you make a server join a domain?

Networking

1. What is the difference between TCP and UDP?
2. What is ICMP? Why should you block it?
3. What is IPv6? Why should we care?
4. In a corporate environment user from London can ping a particular server but users from New York cannot, what steps will you take to troubleshoot the problem?
5. What steps are needed to change the hostname on a Linux machine?
6. Where is the hostname file on a Windows server?
7. How is a hostname resolved on a Linux machine?
8. What's a SSL tunnel?
9. What's a SDN?

**Scripting (questions here vary on rating and depend pretty much on telling the**

**person to write a short script**)

1. What is your favorite scripting language? Why?
2. What are design patterns?
3. Describe some scripts you have written/automation you have done/ programs you have written. Justify your choice of scripting language and design patterns.
4. Can you port the same script to another language? On another OS?
5. How long would it take you to learn another language?

CI/CD

1. How would you trace a binary deployment back to the source code ? How would you structure such a build ?
2. How do you manage dependencies ?
3. CI can only be used to find compile time errors quickly. True/False. Explain.
4. How would you structure CD for an app that depends on 3 other apps ?
5. How do you plan capacity for your CI/CD servers ?
6. All your built binaries should be checked in to your git/svn for quick deployment and posteriority. True/False. Explain.

**Miscellaneous Questions**

1. DevOps Interview Questions
2. What is DevOps and how does it evolve?
3. What DevOps can do for you?
4. What is the advantage of DevOps over maven or ant tools?
5. What are the different DevOps tools present in the market?
6. Which DevOps tool will you prefer among these?
7. What is chef, puppet and docker?
8. What is the advantage of chef over puppet and docker?
9. What is the advantage of puppet over chef and docker?
10. What is the advantage of docker?
11. How DevOps can help in building bridges between Development, QA and Operations teams?
12. Which scripting language is most significant to learn to become a DevOps engineer?
13. What all testing required for a successful DevOps project?
14. What all risks that gets minimized with DevOps?
15. Suppose there is a bug found in a software that has been already in production and it requires quick fix, will DevOps be helpful in getting it done faster? If yes, how?
16. How do you access about how 'deployable' a system is?
17. What is the difference between RAID 1 and RAID 5?
18. What are the alternatives of init.d in linux?
19. What roles do QA (Quality Assurance) should play in DevOps according to you?
20. What long and short term goals an organization should keep in mind before opting for DevOps?
21. What testing is necessary to assure a new service is production ready or not?
22. What are some examples of how you might scale a write/read-heavy application? Why?
23. Tell me about the worst-run/best-run outage you’ve been a part of. What made it bad/well-run?
24. How would you assess how “deployable” a system is?
25. How do you know when a deployment is ready? How would you manage that deployment?
26. How would you prepare for a migration from one platform to another?8
27. What is the purpose of a post-mortem meeting?
28. Do you know how to learn from mistakes? How do you get the most value out of post-mortems?
29. How would you make key aspects of a software system traceable?
30. How do you handle interruptions?
31. How would you deploy software to 5000 systems?
32. What different types of testing need to be carried out on a software system, and what tools would you use to achieve this testing?

#### Software Delivery Questions

* What is Continuous Integration?
* What CI tools have you used? Example:[*Jenkins*](https://jenkins-ci.org/).
* What is Continuous Delivery and why is it important?
* What is Continuous Deployment?
* Explain the importance of A/B testing and how it relates to software delivery
* What are your favorite deployment techniques and why?Examples include:[*0 downtime*](http://www.ebaytechblog.com/2013/11/21/zero-downtime-instant-deployment-and-rollback/),[*Canary*](http://martinfowler.com/bliki/CanaryRelease.html), or[*Blue/Green*](http://martinfowler.com/bliki/BlueGreenDeployment.html)deployments.
* Explain a time when a software release has gone poorly

#### Source Control Questions

* What types of source control programs are you comfortable with?
* What are the benefits of using source control?
* Describe branching strategies you have used
* Give an example of a good commit message

#### Testing Questions

* Talk about different types of automated testing you have implemented
* What is the importance of software testing?
* What testing paradigms do you use, if any?
* Explain the difference between unit and end to end testing
* Have you used any testing frameworks? If so, which ones?

#### Architecture Questions

* Explain what a RESTful architecture is
* How would you scale a slow website?
* Your application just got millions of users overnight, what do you do?
* What are some ways to increase client-side performance?
* Can you explain the significance of a primary/replica database architecture?
* What are some caching strategies you can implement to increase performance?
* What are the advantages and disadvantages of implementing a Microservices architecture?
* What is the difference between vertical and horizontal scalability?

#### General Programming Questions

* What languages are your most comfortable with?
* [Google](https://www.google.ca/webhp?sourceid=chrome-instant&ion=1&espv=2&ie=UTF-8#q=programming%20interview%20questions) is your friend

#### Database Questions

* What is the difference between a relational and non-relational database?
* When do we optimize the database?
* Can you give some ways to optimize the performance of a database?
* What does A.C.I.D mean?
* How can you customize the query plan to increase throughput?
* Explain how you would do Database Continuous Delivery

#### Linux Questions

* How to view running processes in Linux?
* How do you check server uptime?
* How do you start/stop services?
* How do you display the shell’s environment variables?
* What adding #!/bin/bash at the top of a script do?
* What does a & after a command do?
* What does piping commands mean?
* What distributions have you used on servers?

#### Security Questions

* What is the importance of SSL?
* What is a SQL injection?
* What is cross-site scripting (XSS)?
* Why shouldn’t you roll your own crypto?
* How are passwords stored on databases?
* What is a Man-in-the-middle attack?
* How do you safely manage environment variables in a cloud environment?

#### Cloud Questions

* Which cloud providers are you familiar with?
* What are the benefits of using a cloud provider?
* Have you managed production systems in the past?

#### Configuration Management Questions

* Which Configuration Management tools are you most comfortable with? Examples include: Chef, Salt, Puppet, or Ansible.

#### Container Questions

* How does Docker improve scalability, distributed computing, and efficiency vs. traditional cloud virtual machines?
* List some Docker use cases

#### Fun Questions

* What new technology would you like to learn?
* Have you worked on any side projects?
* Do you contribute to any open source projects?