TOPIC: UNDERSTANDING DEVOPS LIFE CYCLE PHASES

QUESTIONS AND ANSWERS :

1. What are the different life cycle phases of DevOps ?

* Continuous Development
* Continuous Integration
* Continuous Testing
* Continuous Delivery
* Continuous Deployment
* Continuous Monitoring
* Continuous Feedback

1. What is continuous development ?

Ans. In this development phase, the set of objectives that must be needed to deliver through the application are developed. Once these project objectives are finalized, the development [process of coding](https://www.educba.com/careers-in-coding/) will start. After development, it transfers to the next phase. As DevOps follows continuous development it also works on the previous application to make the changes according to the feedback and to meet the requirements.

1. What is continuous integration ?

Ans. continuous integration phase occurs where various tests are planned that needs to be carried out in the next phase. It helps us to find out whether the developed application meets the desired requirements or not.

1. What is continuous testing ?

Ans. It is in the continuous testing phase that the developed application is [tested using automation tools](https://www.educba.com/automation-testing-tools/). Testers test the applications using different test cases and give results about different aspects of product or application. These results are then sent to the development process to improve the product or application.

1. What is continuous deployment ?

Ans. In this phase, the code is deployed to the production servers. Also, it is essential to ensure that the code is correctly used on all the servers. The new code is deployed continuously, and configuration management tools play an essential role in executing tasks frequently and quickly.

1. What is continuous monitoring ?

Ans. Monitoring is a phase that involves all the operational factors of the entire DevOps process, where important information about the use of the software is recorded and carefully processed to find out trends and identify problem areas. Usually, the monitoring is integrated within the operational capabilities of the software application.

1. What is continuous feedback ?

Ans. The application development is consistently improved by analyzing the results from the operations of the software. This is carried out by placing the critical phase of constant feedback between the operations and the development of the next version of the current software application.

1. What are the different DevOps tools used for continuous development?

Ans. The most popular tools used are Git, SVN, Mercurial, CVS, and JIRA

1. What are the different DevOps tools used for continuous integration?

Ans. Continuous Integration tool : Jenkins, Teamcity, Bamboo, Hudson

1. What are the different DevOps tools used for continuous testing?

Ans. Continuous Testing tool : Junit, TestNG, Selenium

1. What are the different DevOps tools used for continuous deployment?

Ans. Configuration Management and Deployment tools : Puppet, Chef, Ansible, Docker, Deploy

1. What are the different DevOps tools used for continuous monitoring?

Ans. Continuous Monitoring tools : Splunk, ELK Stack, Kibana, Nagios

1. What are the steps in continuous integration?

Ans. Compilation, Unit tests, Code quality gates.

1. What are the benefits of continuous integration?

Ans. 1. Errors detected early, 2. Decreases bug accumulation 3. Setting the stage for continuous delivery.

1. Continuous implementation is \_\_\_\_\_ process.

Ans. Agile

1. . Which of the following is NOT true of continuous integration in Agile?

[A.](javascript:%20void(0)) It involves moving code from a development to test environment

[B.](javascript:%20void(0)) It involves moving code in large amounts

[C.](javascript:%20void(0)) It involves moving code in frequent intervals

[D.](javascript:%20void(0)) It is most effectively done through automation

Ans. C

1. At a high level, What is the value of continuous integration?

* [A.](javascript:%20void(0)) Reduce risks
* [B.](javascript:%20void(0)) Reduce repetitive manual processes
* [C.](javascript:%20void(0)) Generate deployable software at any time and at any place
* [D.](javascript:%20void(0)) All of the above

Ans. D

1. Which is not the DevOps tools?

A. Nagios

B. GIT

C. QTP

D. Jenkins

Ans. C

1. \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ consist of a central shared repository where teammates can commit changes to a file or set of file.

A. Version control

B. Central Repository

C. Risk Filters

D. None of the above

Ans. A

1. What are not the benefits of using version control?

A. All the team members are allowed to work freely on any file at any time

B. All the past versions and variants are neatly packed up inside the VCS

C. A distributed VCS like Git allows all the team members to have complete history of the project

D. None of the above

Ans. D

1. What are the success factors for Continuous Integration?

A. Make the build self-testing

B. Keep the build fast

C. Everyone can see the results of the latest build

D. All of the above

Ans. D

1. \_\_\_\_\_\_\_\_\_\_\_is the process of executing automated tests as part of the software delivery pipeline
   * A. Continuous Integration
   * B. Continuous Testing
   * C. Continuous Delivery
   * D. Continuous pipeline

Ans. B

1. \_\_\_\_\_\_\_\_\_\_\_\_\_ allows any change made in the code to be tested immediately
   * A. Continuous Integration
   * B. Continuous Testing
   * C. Continuous Delivery
   * D. Continuous pipeline

Ans. B

1. What are the Testing types not supported by Selenium?
   * A. Functional Testing
   * B. Regression Testing
   * C. Behavioural Testing
   * D. None of the above

Ans. D

1. \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ allows timely identification of problems or weaknesses and quick corrective action that helps reduce expenses of an organization
   * A. Continuous Integration
   * B. Continuous Testing
   * C. Continuous Delivery
   * D. Continuous Monitoring

Ans. D

1. \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ are used to create containers
   * A. Docker
   * B. Jenkins
   * C. Chef
   * D. Jira

Ans. A

1. Flow of the CICD pipeline
   * A. Build, Deploy,version control,Auto test
   * B. Build, Deploy, Auto Test, Unit Test
   * C.Version Control, Build, Deploy, Auto Test
   * D. Version Control, Build,Auto Test, Deploy

Ans. D

1. Which is not monitor tool?
   * A. Nagios
   * B. Splunk
   * C. DataDog
   * D. CodeShip

Ans. D

1. Which is not Code Tool?
   * A. Jira
   * B. Confluence
   * C. Git
   * D. Docker

Ans. D

1. Which is not Operate Tool?
   * A. Chef
   * B. Ansibel
   * C. Kubernetes
   * D. DC/OS

Ans. D

1. Which is not Test Tool?
   * A. JUint
   * B. TestNG
   * C. Selenium
   * D. None of the above

Ans. D

1. Which is not Integration Tool?
   * A. Jenkins
   * B. QBamboo
   * C. Travls
   * D. None of the above

Ans . D

1. \_\_\_\_\_\_\_\_\_\_\_ is a central part of continuous deployment

Ans. Deployment pipeline

1. Which of the following is / are the Review Technique documented?

a. Inspection

b. walkthrough

c. Both of These

d. None of these

Ans. a

1. A retail company purchased commercial off the shelf application for automating their billing process. But before introducing it on large scale they are going for beta testing .What will be the reason for doing this?

a. To find defects

b. To train employees

c. To gain confidence in system

d. All of the above

Ans. c

1. Who is responsible for Component Testing?

a. Software tester

b. Designer

c. User

d. Developer

Ans. d

1. What is the order in which test levels are performed?

a. Unit, Integration, System, Acceptance

b. Unit, System, Integration, Acceptance

c. Unit, Integration, Acceptance, System

d. It depends on nature of a project

Ans. b

1. If a problem occurs in a system 6 months after it completely goes live, what will be the approach for fixing that problem on very urgent basis?

a. Do not involve testers

b. Just go for retest

c. Always go for full regression test

d. Go for retest and then use risk assessment to decide reasonable subset of regression test suit that will be executed.

Ans. d

1. Which things are measured by Software Test Effectiveness?

a. How many customer requirements are implemented in system?

b. How well the customer specifications are achieved by the system?

c. How much effort is put in developing the system?

d. All of the above

Ans. a and b

1. Component Testing is a \_\_\_\_\_\_\_ .

a. Black box Testing

b. White box Testing

c. Grey box Testing

d. Both a and b

Ans. b

1. Verifying that whether software components are functioning correctly and identifying the defects in them is objective of which level of testing?

a. Integration Testing

b. Acceptance Testing

c. Unit Testing

d. System Testing

Ans. C

1. Configuration Testing deals with hardware while compatibility testing deals with Software.

a. True

b. False

Ans. a

1. Software tester should be involved very early during development phase of a project.

a. True

b. False

Ans. a

1. Software Reliability is the probability of failure-free software operation for a specified period of time in a specified environment.

a. True

b. False

Ans. a

1. Which DevOps principle focuses on product and service thinking?

A. Customer-centric action

B. Continuous Improvement

C. Create with the end in mind

D. Automate everything you can

Ans. C

1. What is the difference between Continuous Delivery and Continuous Deployment?

A. Continuous Delivery is a manual task, while Continuous Deployment is an automated task.

B. Continuous Delivery has a manual release to production decision, while Continuous Deployment has releases automatically pushed to production.

C. Continuous Delivery includes all steps of software development life cycle; Continuous Deployment may skip few steps such as validation and testing.

D. Continuous Delivery means complete delivery of the application to customer; Continuous Deployment includes only deployment of the application in customer environment.

Ans. B

1. Which DevOps principle appreciates measuring processes, people, and tools?

A. Continuous improvement

B. Create with the end in mind

C. Cross-functional autonomous teams

D. People responsibility

Ans. A

1. What are the appropriate characteristics of Continuous Delivery approach?

1. Complex, but small number of releases

2. A focus on cycle time reduction

3. Resource-based management of the process

4. Self-managed and responsive teams

A. 1 and 3

B. 2 and 4

C. 2, 3, and 4

D. 1, 2, 3, and 4

Ans. B

1. Which component provides the first feedback on the quality of committed application code changes?

A. Automated Provisioning

B. Automated Build

C. Automated Test

D. Automated Deployment

Ans. B

1. Which tool helps lowering risk during development as customer feedback is embedded into the design process?

A. Test Automation

B. Snapshot Deployment

C. Story Mapping

D. Value Stream Mapping

Ans. C

1. Is Ansible an open source tool

Ans. yes

1. \_\_\_\_\_\_ is a source code management tool.

Ans. Git

1. Which of the following is one of the best ways to convert normal changes into standard changes?
   * 1. Negotiate with release managers
     2. Publicly complain about bureaucracy and make everyone be aware of it
     3. Make sure normal changes are very carefully deployed to your production systems
     4. Use your track record of successful automated deployments with standard changes

Ans. D

1. What is chef ?

Ans. It is a powerful automation platform that transforms infrastructure into code. Chef is a tool for which you write scripts that are used to automate processes.

1. Explain the architecture of Chef.

Ans. it consists of:

* Chef Server: The Chef Server is the central store of your infrastructure’s configuration data. The Chef Server stores the data necessary to configure your nodes and provides search, a powerful tool that allows you to dynamically drive node configuration based on data.
* Chef Node: A Node is any host that is configured using Chef-client. Chef-client runs on your nodes, contacting the Chef Server for the information necessary to configure the node. Since a Node is a machine that runs the Chef-client software, nodes are sometimes referred to as “clients”.
* Chef Workstation: A Chef Workstation is the host you use to modify your cookbooks and other configuration data

1. What do you mean by recipe in Chef?

Ans. A Recipe is a collection of Resources that describes a particular configuration or policy. A Recipe describes everything that is required to configure part of a system.

1. What is an ansible module ?

Ans. Modules are considered to be the units of work in Ansible. Each module is mostly standalone and can be written in a standard scripting language such as Python, Perl, Ruby, bash, etc..

1. What are playbooks in Ansible ?

Ans. Playbooks are Ansible’s configuration, deployment, and orchestration language. They can describe a policy you want your remote systems to enforce, or a set of steps in a general IT process.

1. Why is continuous monitoring necessary ?

Ans. Continuous Monitoring allows timely identification of problems or weaknesses and quick corrective action that helps reduce expenses of an organization. Continuous monitoring provides solution that addresses three operational disciplines known as:

* continuous audit
* continuous controls monitoring
* continuous transaction inspection

1. What is Nagios ?

Ans. Nagios is one of the monitoring tools. It is used for Continuous monitoring of systems, applications, services, and business processes etc in a DevOps culture.

1. Which statement best describes the role of Change Management within a DevOps environment?

A) Nothing changes as a risk adverse Change Management approach is paramount to IT and business success.

B) The moment an app change is asked for, the request should go to the Developers to authorise. Once authorised it goes to Operations for implementation.

C) The moment an app change is asked for, the request should go out to everyone on the team, no matter which IT discipline they work in.

D) DevOps does not need Change Management?

Ans. C

1. An organization maintains an independent and autonomous team for each of its services. What is a disadvantage of this type of organization structure?

A. Quality of delivered features will be low.

B. Implementation of changes within a team is slow.

C. Reuse of knowledge and skills within the organization is limited.

D. Waiting time for processing the service request is high.

Ans. C

1. What type of mindset is the core of a DevOps culture?

A. Service Mindset

B. Skill Mindset 4

C. People Mindset

D. Process Mindset

Ans. A

1. Which statement best describes the goal of DevOps?

A) One goal of DevOps is to establish an environment where Change Management does bot control application releases.

B) One goal of DevOps is to establish an environment where releasing more reliable applications faster and more frequently can occur.

C) One goal of DevOps is to establish an environment where application development perform all operations tasks.

D) One goal of DevOps is to establish an environment where releasing applications is valued over the quality of the released application.

Ans. B

1. Is this statement correct? “DevOps is more than just a tool or a process change, it inherently requires an organisational culture shift”

A) Yes, there needs to be cultural shift within the organisation across all stakeholders to ensure a successful adoption of a DevOps approach.

B) Yes, but the most up to date tools and LEAN processes need to be in place to drive an organisational culture shift.

C) No, DevOps is all about the tools.

D) No, cultural shift will occur when staff are using the most up to date tools and LEAN processes.

1. Which benefits of adopting a DevOps approach could be included in a business case to adopt a DevOps approach?

A) Improved deployment frequency, which can lead to faster time to market

B) Lower failure rate of new releases

C) Shortened lead time between fixes

D) Faster mean time to recovery in the event of a new release crashing or otherwise disabling the current system

1. Which statement best describes the relationship between DevOps and Continuous Delivery?

A) DevOps and Continuous Delivery are the same thing.

B) DevOps and Continuous Delivery are not related and are mutually exclusive.

C) DevOps and Continuous Delivery share a background in Agile methods and LEAN thinking.

D) DevOps and Continuous Delivery share common processes.

1. What is NOT an appropriate predictors of IT performance in a DevOps environment?

1. Changes approved by an external team members

2. High-trust organizational culture

3. Proactive monitoring

4. Version control of all artifacts

1. Erik is working in a Product team (or Business System team) specialized in a specific IT service for the Sales department. Which of the following types of activities will most likely be frequently returning in the agenda’s of Eriks team?

A. Many handovers moments with other departments.

B. Meetings on the utilization of the specialized resources within the organization.

C. Monthly release meetings for the bi-monthly release

D. Attending the product demo meeting

Ans. D

1. What is NOT a challenges between the Development and Operations teams in a traditional organizations?

1. Blame game between Dev and Ops

2. Different tools used between Dev and Ops

3. No feedback loop between Dev and Ops

4. Development and Operations is not maintained by the same person

Ans. D

1. What is the main benefit of automated provisioning?

A. Flexible approach to ad-hoc system changes

B. Focus on operational perspective to control infrastructure changes

C. High speed delivery of new environments

D. Variability in application environment

Ans. C

1. What is the correct sequence of the four steps when providing feedback in according to the Feedback model?

A. 1. Describe concrete observations. 2. Explain what it does to you. 3. Wait and listen to clarifying questions. 4. Give concrete suggestions OR recognition/incentive.

B. 1. Explain what it does to you. 2. Describe concrete observations. 3. Wait and listen to clarifying questions. 4. Give concrete suggestions OR recognition/incentive.

C. 1. Wait and listen to clarifying questions. 2. Explain what it does to you. 3. Describe concrete observations. 4. Give concrete suggestions OR recognition/incentive.

D. 1. Wait and listen to clarifying questions. 2. Give concrete suggestions OR recognition/incentive. 3. Describe concrete observations. 4. Explain what it does to you.

Ans. A

1. Explain some of the popular devops tools.

Here are some popular tools of DevOps, such as:

1. Jenkins: Jenkins is a DevOps tool for monitoring the execution of repeated tasks. Jenkins is a software that allows continuous integration. And it will be installed on a server where the central build will take place.
2. Ansible: Ansible is a leading DevOps tool. Ansible is an open-source IT engine that automates application deployment, cloud provisioning, intra service orchestration, and other IT tools.
3. Nagios: Nagios is one of the more useful tools for DevOps. It can determine the errors and rectify them with the help of network, infrastructure, server, and log monitoring systems.
4. Docker: Docker is a high-end DevOps tool that allows building, ship, and run distributed applications on multiple systems.
5. Git: Git is an open-source distributed version control system that is freely available for everyone. It is designed to handle minor to major projects with speed and efficiency.
6. What is the requirement for using Jenkins?

* [A.](javascript:%20void(0)) A source code repository which is accessible, for instance, a Git repository
* [B.](javascript:%20void(0)) A working build script, e.g., a Maven script, checked into the repository
* [C.](javascript:%20void(0)) Both of these
* [D.](javascript:%20void(0)) None

Ans. C

1. What are the advantages of Jenkins?

* [A.](javascript:%20void(0)) At integration stage, build failures are cached
* [B.](javascript:%20void(0)) For each code commit changes an automatic build report notification generates
* [C.](javascript:%20void(0)) To notify developers about build report success or failure, it is integrated with LDAP mail server
* [D.](javascript:%20void(0)) All of the above

Ans. D

1. How can we move or copy Jenkins from one server to another?

* [A.](javascript:%20void(0)) First, copy the related job directory and slide a job from one installation of Jenkins to another
* [B.](javascript:%20void(0)) Make a copy of an already existing job by making clone of a job directory by a different name
* [C.](javascript:%20void(0)) Renaming an existing job by rename a directory
* [D.](javascript:%20void(0)) All of these

Ans. D

1. What are the useful plugins in Jenkin?

* [A.](javascript:%20void(0)) Maven 2 project
* [B.](javascript:%20void(0)) Amazon EC2
* [C.](javascript:%20void(0)) HTML publisher
* [D.](javascript:%20void(0)) All

Ans. D

1. Which Scm Tools Does Jenkins Support?

* [A.](javascript:%20void(0)) AccuRev
* [B.](javascript:%20void(0)) CVS
* [C.](javascript:%20void(0)) Subversion
* [D.](javascript:%20void(0)) All of these

Ans. D

1. Which file is used to define dependency in maven?

* [A.](javascript:%20void(0)) build.xml
* [B.](javascript:%20void(0)) pom.xml
* [C.](javascript:%20void(0)) dependency.xml
* [D.](javascript:%20void(0)) version.xml

Ans. B

1. How can we secure Jenkins?

* [A.](javascript:%20void(0)) Global security should be enabled
* [B.](javascript:%20void(0)) Jenkins should be integrated with appropriate plugins
* [C.](javascript:%20void(0)) Automate the process of setting rights and privileges
* [D.](javascript:%20void(0)) All of the above

Ans. D

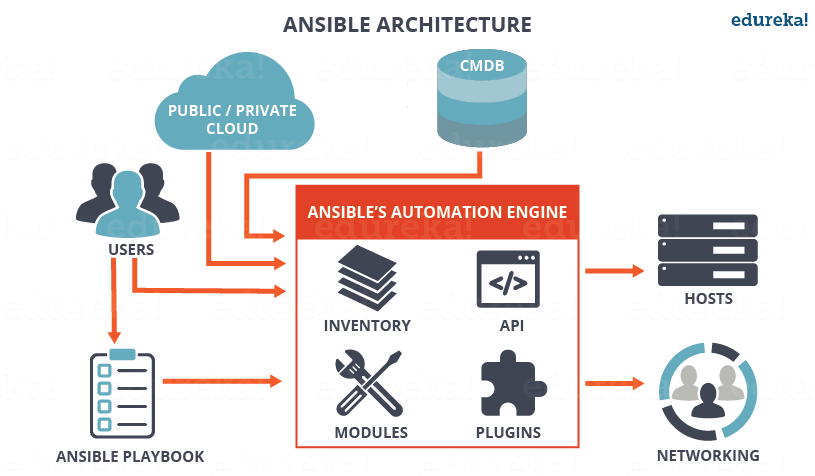
1. How can we setup Jenkins jobs?

* [A.](javascript:%20void(0)) Select new item from the menu
* [B.](javascript:%20void(0)) After that enter a name for the job and select free-style job
* [C.](javascript:%20void(0)) Then click OK to create new job in Jenkins
* [D.](javascript:%20void(0)) The next page enables you to configure your job
* [E.](javascript:%20void(0)) All of these

Ans. E

### **What are the different components of ansible? Explain Ansible architecture.**

The below diagram depicts the Ansible architecture:



1. What are Ansible components ?

The Ansible Automation engine consists of the following components:

**Inventories:**These are a list of nodes containing their respective IP addresses, servers, databases, etc. which needs to be managed.

**APIs:** Just like any other API, the Ansible APIs are used for commuting various Cloud services, public or private services.

**Modules:** The modules are used to manage system resources, packages, libraries, files, etc. Ansible modules can be used to automate a wide range of tasks. Ansible provides around 450 modules that automate nearly every part of your environment.

**Plugins:** If you want to execute Ansible tasks as a job, Ansible Plugins can be used. They simplify the execution of a task by building a job like an environment that basically contains pieces of code corresponding to some specific functionality. There are 100s of Plugins provided by Ansible. An example is the Action plugin, which acts as front ends to modules and can execute tasks on the controller before calling the modules themselves.

**Networking:** Ansible can also be used to automate different networks and services. It can do this by creating a playbook or an Ansible role that easily spans different network hardware.

**Hosts:** The Ansible Hosts/ Node systems are machines (Linux, Windows, etc) that are getting automated.

**Playbooks:** Playbooks are simple code files which describe the tasks that need to be executed. The Playbooks are written in YAML format. They can be used to automate tasks, declare configurations, etc.

**CMDB:** It is a database that acts as a storehouse for various IT installations. It holds data about various IT assets (also known as configuration items (CI)) and describes the relationships between such assets.

**Cloud:** It is a network of remote servers hosted on the Internet to store, manage, and process data, rather than a local server.

1. What language is an Ansible playbooks are written in by default?

A.YAML format

B.JSON format

C. XML format

 D.HTML format

Ans. A

1. Which command will you use to run a playbook called install.yaml with Ansible?

A.ansible-playbook install.yml

 B.ansible install.yml

 C.ansible --playbook install.yml

 D.ansible -p install.yml

Ans. A

1. Which argument will you use to specify a variable for your Ansible playbook?

A.-v

 B.-e

 C.-c

 D.-a

Ans. B

1. Ansible playbooks are the basis for a really simple configuration management and multi-machine deployment system, unlike any that already exist, and one that is very well suited to deploying complex applications.

A.false

 B.true

Ans. B

1. What does 'become: yes' mean in Ansible playbooks?

A. It means that the command must be retried until it succeeds

 B.It means that the service needs to be started once installed

 C.It means that the worker node should become a manager node

 D. It means that we would run all commands as root

Ans. D

1. How do all the devops tools work together?

* Developers develop the code and this source code is managed by Version Control System tools like Git etc.
* Developers send this code to the Git repository and any changes made in the code is committed to this Repository.
* Jenkins pulls this code from the repository using the Git plugin and build it using tools like Ant or Maven.
* Configuration management tools like puppet deploys & provisions testing environment and then Jenkins releases this code on the test environment on which testing is done using tools like selenium.
* Once the code is tested, Jenkins send it for deployment on the production server (even production server is provisioned & maintained by tools like puppet).
* After deployment It is continuously monitored by tools like Nagios.
* Docker containers provides testing environment to test the build features.

1. Explain the steps of continuous integration.
2. Developers check out code into their private workspaces.
3. When they are done with it they commit the changes to the shared repository (Version Control Repository).
4. The CI server monitors the repository and checks out changes when they occur.
5. The CI server then pulls these changes and builds the system and also runs unit and integration tests.
6. The CI server will now inform the team of the successful build.
7. If the build or tests fails, the CI server will alert the team.
8. The team will try to fix the issue at the earliest opportunity.
9. This process keeps on repeating.
10. Why do we need continuous integration and testing?

Continuous Integration of Dev and Testing improves the quality of software, and reduces the time taken to deliver it, by replacing the traditional practice of testing after completing all development. It allows Dev team to easily detect and locate problems early because developers need to integrate code into a shared repository several times a day (more frequently). Each check-in is then automatically tested.

1. What are the main features of CI?

* Maintain a code repository
* Automate the build
* Make the build self-testing
* Everyone commits to the baseline every day
* Every commit (to baseline) should be built
* Keep the build fast
* Test in a clone of the production environment
* Make it easy to get the latest deliverables
* Everyone can see the results of the latest build
* Automate deployment

1. What are the benefits of continuous testing?

* Supports execution of repeated test cases
* Aids in testing a large test matrix
* Enables parallel execution
* Encourages unattended execution
* Improves accuracy thereby reducing human generated errors
* Saves time and money

1. What are the key elements of continuous testing tools?

Key elements of Continuous Testing are:

* **Risk Assessment:** It Covers risk mitigation tasks, technical debt, quality assessment and test coverage optimization to ensure the build is ready to progress toward next stage.
* **Policy Analysis:** It ensures all processes align with the organization’s evolving business and compliance demands are met.
* **Requirements Traceability:** It ensures true requirements are met and rework is not required. An object assessment is used to identify which requirements are at risk, working as expected or require further validation.
* **Advanced Analysis:** It uses automation in areas such as static code analysis, change impact analysis and scope assessment/prioritization to prevent defects in the first place and accomplishing more within each iteration.
* **Test Optimization:** It ensures tests yield accurate outcomes and provide actionable findings. Aspects include Test Data Management, Test Optimization Management and Test Maintenance
* **Service Virtualization:** It ensures access to real-world testing environments. Service visualization enables access to the virtual form of the required testing stages, cutting the waste time to test environment setup and availability.

1. What is Selenium IDE?

It is an integrated development environment for Selenium scripts. It is implemented as a Firefox extension, and allows you to record, edit, and debug tests. Selenium IDE includes the entire Selenium Core, allowing you to easily and quickly record and play back tests in the actual environment that they will run in.  
Now include some advantages in your answer. With autocomplete support and the ability to move commands around quickly, Selenium IDE is the ideal environment for creating Selenium tests no matter what style of tests you prefer.

1. What is puppet?

It is a Configuration Management tool which is used to automate administration tasks. Puppet has a Master-Slave architecture in which the Slave has to first send a Certificate signing request to Master and Master has to sign that Certificate in order to establish a secure connection between Puppet Master and Puppet Slave . Puppet Slave sends request to Puppet Master and Puppet Master then pushes configuration on Slave.

1. What is trunk in trunk based DevOps delivery?

A.Developers collaborate on code in a single branch called “trunk”.

**B.**Trunk is a special private branch in a developer workstation.

**C.**Trunk is the process of merging code in DevOps deliveries.

**D.**Trunk is a special source code version controlling system which stores mission critical special projects of your DevOps organization.

**Ans.** A

1. Which one of the following is not one of the DevOps principles for good test automation?

**A.**Test Automation should give quick and early feedback about your quality of work.

**B.**Never mix test driven development (TDD) together with your test automation approach.

**C.**Tests should generate consistent, deterministic and repeatable results provided same conditions for different test runs.

**D.**With your test automation, avoid slow and periodic feedback. What you need is fast feedback whenever you or your developer attempts to check-in code to your trunk.

**Ans.** B

1. What is Jenkins pipeline?

Jenkins Pipeline can be defined as a suite of plugins supporting both implementation and integration of Jenkins continuous delivery pipeline.

1. What is CI/CD pipeline?

Continuous integration or continuous delivery pipeline consists of build, deploy, test, and release. The pipeline feature is very time-saving. In other words, a pipeline is a group of build jobs that are chained and integrated in a sequence.