PeopleCert

DevOps-Leader

DevOps Leader v2.2 Exam

Questions & Answers (Demo)

Version: 4.0

Question: 1

A manufacturing organization is struggling to deliver the new features their clients are asking for in their web-based applications. When they do release a new version it usually causes incidents which result in system downtime and overtime worked by the IT operations department. Additionally, the CEO has told the IT department he is extremely worried about cyber threats and wants them to focus on this as a matter of urgency - they are not sure how to do this as they are so busy firefighting.

How will DevOps help them?

A. By making the development teams support their applications

B. By increasing the ability to deliver applications quickly and safely

C. By automating the software delivery lifecycle

D. By experimenting with new features

Answer: B

Explanation:

DevOps helps this organization primarily by improving its ability to deliver technology change quickly, safely, and sustainably. The scenario describes several classic symptoms of a non-DevOps operating model: slow feature delivery, unstable releases, production incidents, excessive operational toil, and inability to focus on strategic risk such as cybersecurity because teams are trapped in reactive firefighting. Option B is the most complete answer because DevOps is not merely automation, experimentation, or shifting support responsibility to developers. Those may be practices within a broader transformation, but the leadership objective is improved flow, reliability, feedback, resilience, and value delivery.

By adopting DevOps principles, the organization can reduce deployment risk through smaller batch sizes, better collaboration between development, operations, security, and business stakeholders, automated testing, continuous integration, continuous delivery, monitoring, and learning from incidents. Security concerns can also be addressed earlier through DevSecOps practices, integrating security controls into the delivery lifecycle rather than treating them as separate emergency work. This supports both business agility and operational stability.

Specific Study Guide alignment: Becoming a DevOps Organization; Measuring to Improve; Measuring to Learn; DevOps and Transformational Leadership.

Question: 2

To get the most accurate picture of an organization's actual state in a DevOps evolution, why is it essential to listen to everybody that's involved, particularly those who are doing the work on a day to day basis?

A. Management often have a rosier view

B. There are many paths to success, and even more that lead to failure

C. Cross team sharing is key to scaling

D. Automating security is mission-critical

Answer: A

Explanation:

The correct answer is A because a reliable DevOps assessment must expose the organization's real operating conditions, not only its intended structure, formal reports, or leadership interpretation. In DevOps evolution, leaders must understand actual flow of work, friction points, queue times, handoffs, rework, incident patterns, cultural constraints, and sources of delay. These are often most visible to the people performing the work every day: engineers, testers, service desk staff, operations teams, security practitioners, product owners, and release personnel.

Management perspectives are valuable, but they can be filtered through dashboards, status reports, escalation paths, and optimistic assumptions. Leaders may see strategic intent, while teams experience practical reality. This is why DevOps emphasizes learning from the system of work, going to where the work happens, creating psychological safety, and listening across organizational levels. Without frontline input, transformation activity may optimize the wrong constraint or reinforce existing dysfunction.

Options B, C, and D describe valid DevOps ideas, but they do not directly explain why broad listening is essential when assessing the current state. The relevant study guide areas are Measuring to Learn, DevOps and Transformational Leadership, Becoming a DevOps Organization, and Unlearning Behaviors.

==========

Question: 3

Thierry is a salesperson at an organization that provides trading software to banking clients. His clients are telling him they are unhappy with the rate at which changes are being made to Thierry's

software. Thierry can see that the IT department is extremely busy, but seems to be struggling to deliver anything.

What will help the IT department focus on delivering what the clients need?

A. Definition of done is customer value outcome realized

B. Having a *Do Not Fail' culture

C. Disseminating information

D. Measuring cost and capacity

Answer: A

Explanation:

The correct answer is A because the core issue is not that the IT department lacks activity; it is that effort is not translating into customer-valued outcomes. DevOps leadership shifts focus from local productivity, task completion, and departmental busyness toward end-to-end value delivery. A feature is not truly “done” merely because development is complete, testing has passed, or a release has occurred. It is done when the intended customer value has been realized and validated.

In this scenario, Thierry’s banking clients are dissatisfied with the rate of meaningful change. The IT department appears overloaded, but the business problem is customer responsiveness. Defining done as “customer value outcome realized” aligns IT work with client needs, improves prioritization, and encourages teams to measure outcomes rather than outputs. This helps reveal whether work is flowing to production, whether it is usable, whether it solves the customer problem, and whether feedback is being incorporated.

A “Do Not Fail” culture would likely reduce experimentation and learning. Disseminating information is useful but insufficient. Measuring cost and capacity may support planning, but it does not by itself align work to customer value. Relevant study guide areas include Becoming a DevOps Organization, Measuring to Learn, Measuring to Improve, and Articulating and Socializing Vision.

Question: 4

When thinking of the dimensions of transformational leadership, which of the following is how we would expect a transformational leader to behave?

A. Personally compliments individuals for outstanding work

B. Accepts team’s status quo

C. Berates team for low quality work

D. Puts the organizational vision before individual needs

Answer: D

Explanation:

A transformational leader is expected to orient people around a compelling shared vision and inspire them to move beyond narrow local interests, habits, or individual preferences. In a DevOps context, this is essential because transformation requires people to change long-established behaviors, cross functional boundaries, challenge legacy processes, and focus on outcomes that matter to the whole organization. Option D is the strongest answer because transformational leadership is associated with vision, purpose, inspiration, role modelling, and mobilizing people toward a future state.

Option A may appear positive, but it is closer to a transactional or contingent-reward behavior: recognition is given in response to specific performance. That can be useful, but it is not the defining behavior of transformational leadership. Option B is incorrect because accepting the status quo conflicts with transformation, continuous learning, and improvement. Option C is also incorrect because blame and humiliation damage psychological safety, reduce learning, and discourage transparency.

The DOL leadership theme emphasizes that DevOps change requires leaders who can articulate vision, challenge existing assumptions, build trust, and energize people through change. Relevant study guide references: DevOps and Transformational Leadership, Articulating and Socializing Vision, Unlearning Behaviors, and Maintaining Energy and Momentum.

==========

Question: 5

When an organization has adopted DevOps principles and practices, releasing a change to their applications and services can be described as which of the following?

A. A high risk event

B. Like breathing

C. A release night or weekend is scheduled

D. The release management team handle it

Answer: B

Explanation:

In a mature DevOps organization, releasing change should become routine, low-risk, repeatable, and almost unremarkable — “like breathing.” This reflects a shift away from large, infrequent, manually coordinated releases toward small, frequent, well-tested, automated, and observable changes. DevOps aims to make delivery safe by improving flow, feedback, collaboration, automation, deployment practices, monitoring, and learning from production.

Option A describes the traditional release pattern DevOps seeks to eliminate: large batches, long lead times, fragile deployments, and fear of failure. Option C also reflects an older operating model in which releases are treated as exceptional events requiring special windows, weekend work, and extensive coordination. Option D implies that release responsibility is isolated in a separate team, whereas DevOps promotes shared ownership across product, development, operations, security, and other stakeholders.

The key point is that DevOps does not simply accelerate release frequency; it changes the system so that frequent release becomes safe. Capabilities such as continuous integration, deployment automation, automated testing, feature flags, telemetry, rollback patterns, and blameless learning reduce the risk of change. Relevant study guide references: Becoming a DevOps Organization, Measuring to Improve, Measuring to Learn, and Target Operating Models and Organizational Designs.