Download Free PeopleCert DevOps-Leader Exam PDF | PrepBolt

Don't miss out! Download the latest free PeopleCert DevOps Leader v2.2 Exam PDF questions. Access real DevOps-Leader dumps with verified answers and boost your chances to pass your certification on the first try with PrepBolt DevOps-Leader exam pdf questions and answers.

Thank you for Downloading DevOps-Leader exam PDF Demo

https://prepbolt.com/DevOps-Leader.html

Question 1

Question Type: MultipleChoice

Which of the following is NOT a good way to transform local discoveries into global improvements?

Options:

A- Use chat rooms and chat bots, ChatOps, to automate and capture organizational knowledge
B- Automate standard processes in software for reuse
C- Have multiple source code repositories for each team
D- Share and spread user stories and automated tests as documentation

Answer:

C

Explanation:

The correct answer is C because having multiple source code repositories for each team can reinforce fragmentation, duplication, local optimization, and knowledge silos. Transforming local discoveries into global improvements requires mechanisms that make learning visible, reusable, and accessible across the organization. Separate team-specific repositories may be appropriate in some architectures, but as stated here, the practice does not inherently help spread discoveries across teams.

The other options are DevOps-aligned mechanisms for scaling learning. ChatOps can capture operational knowledge, make collaboration visible, and support rapid incident response. Automating standard processes in reusable software helps convert local improvements into repeatable organizational capabilities. Sharing user stories and automated tests as documentation spreads understanding of expected behavior, customer needs, and quality standards.

The broader principle is that high-performing DevOps organizations create feedback loops that allow one team's learning to improve the whole system. Local discoveries should become shared assets, reusable patterns, automation, documentation, and practices. Relevant study guide references: Measuring to Learn; Measuring to Improve; Maintaining Energy and Momentum; Becoming a DevOps Organization.

Question 2

Question Type: MultipleChoice

In an organization where blame is part of the culture, what happens?

Options:

A- Learning
B- Open minds close
C- Inquiry
D- Innovation increases

Answer:

B

Explanation:

The correct answer is B. In a blame culture, open minds close. When people expect punishment, criticism, or reputational damage after mistakes, they become defensive. They hide problems, reduce transparency, avoid experimentation, and protect themselves rather than improving the system. This directly conflicts with DevOps principles of learning, psychological safety, feedback, and continuous improvement.

Blame prevents organizations from understanding the real causes of failure. Incidents in complex technology environments are rarely the result of a single person making a simple mistake. They usually emerge from system conditions such as unclear ownership, excessive work in progress, weak controls, poor feedback, brittle architecture, manual processes, or conflicting incentives. A blame culture focuses attention on individuals instead of improving those conditions.

The other options are incorrect because blame reduces learning, inquiry, and innovation. People do not ask better questions when they feel unsafe; they ask fewer questions. DevOps leaders must replace blame with blameless learning, constructive inquiry, and shared accountability. Relevant study guide references: Unlearning Behaviors; DevOps and Transformational Leadership; Measuring to Learn; Maintaining Energy and Momentum.

==========

Question 3

Question Type: MultipleChoice

In Kolb's Learning Styles, when a person is thinking and watching, what are they doing?

Options:

A- Diverging
B- Assimilating
C- Converging
D- Accommodating

Answer:

B

Explanation:

The correct answer is B. In Kolb's Learning Styles model, ''thinking and watching'' corresponds to Assimilating. Kolb's model combines two dimensions: how people perceive information and how they process it. ''Thinking'' aligns with abstract conceptualization, while ''watching'' aligns with reflective observation. Together, these form the Assimilating learning style.

Assimilating learners tend to prefer logical analysis, structured models, concepts, theories, and reflective understanding. They often want to understand the ''why'' behind a practice before applying it. In DevOps leadership, recognizing different learning styles is useful because transformation requires people to learn new behaviors, unlearn old assumptions, and adopt new ways of working. A leader who only communicates through one style may fail to engage the whole organization.

The other styles map differently. Diverging is feeling and watching. Converging is thinking and doing. Accommodating is feeling and doing. DevOps leaders should support learning through explanation, experimentation, reflection, practice, and feedback so that different learning preferences are addressed. Relevant study guide references: Unlearning Behaviors; Maintaining Energy and Momentum; DevOps and Transformational Leadership.

==========

Question 4

Question Type: MultipleChoice

What characterizes the Fourth Industrial Revolution?

Options:

A- Mass production
B- Cyber, AI and IoT
C- Digital and automation

D- The steam engine

Answer:

B

Explanation:

The correct answer is B. The Fourth Industrial Revolution is characterized by the convergence of cyber-physical systems, artificial intelligence, the Internet of Things, advanced automation, data-driven decision-making, and highly connected digital ecosystems. In the DevOps Leader context, this matters because organizations are operating in an environment where speed, adaptability, resilience, security, and continuous learning are essential.

The other options correspond more closely to earlier industrial revolutions. The steam engine is associated with the First Industrial Revolution. Mass production is associated with the Second Industrial Revolution. Digital and automation are more closely associated with the Third Industrial Revolution, where computing and electronics transformed production and business processes.

The Fourth Industrial Revolution increases pressure on organizations to respond rapidly to customer needs, market changes, cyber threats, and technological disruption. DevOps provides a leadership and operating model for this environment by improving flow, feedback, experimentation, automation, and collaboration across business and technology teams. Relevant study guide references: DevOps and Transformational Leadership; Becoming a DevOps Organization; Articulating and Socializing Vision.

==========

Question 5

Question Type: MultipleChoice

When you are writing a DevOps investment case, which of the following is a hard, quantifiable benefit you can state?

Options:

A- Revenue from accelerating time to value of new functionality
B- Customer satisfaction
C- Competitive advantage
D- Market perception

Answer:

A

Explanation:

The correct answer is A because revenue from accelerating time to value is a hard, quantifiable business benefit. A DevOps investment case must translate improvement activity into measurable organizational outcomes. Faster delivery of valuable functionality can directly affect revenue by enabling earlier market entry, quicker customer adoption, faster realization of product enhancements, and reduced delay cost.

Customer satisfaction, competitive advantage, and market perception are important, but they are generally softer or less directly quantifiable unless converted into measurable indicators. For example, customer satisfaction may be tracked through NPS or churn, and competitive advantage may influence revenue, but the option that most directly expresses a financial benefit is accelerated revenue from faster time to value.

DevOps leaders need this distinction when building executive support. Tooling, automation, team redesign, continuous delivery, and value stream improvement should not be justified only as technical improvements. They should be linked to financial and operational outcomes such as increased revenue, reduced cost of delay, lower change failure cost, faster recovery, and improved capacity for innovation. Relevant study guide references: Measuring to Improve; Measuring to Learn; Becoming a DevOps Organization.

==========

Question 6

Question Type: MultipleChoice

What is one of Kotter's principles of a Dual Operating System?

Options:

A- Action that is head driven
B- An enhanced hierarchy
C- More management
D- A ''get-to'' mindset

Answer:

D

Explanation:

The correct answer is D. Kotter's Dual Operating System emphasizes the need to run transformation through a second, more agile network alongside the traditional hierarchy. One of its principles is a ''get-to'' mindset rather than a ''have-to'' mindset. This means people participate because they are emotionally and intellectually committed to the opportunity, not simply because management has mandated compliance.

This is highly relevant to DevOps leadership because sustainable change cannot be achieved only through policy, structure, or tool adoption. Teams must understand the purpose of the transformation and feel motivated to contribute to better flow, faster feedback, improved resilience, and customer value. A ''get-to'' mindset creates voluntary energy, ownership, and momentum.

The other options are incorrect. Kotter does not advocate more management or simply enhancing hierarchy. Nor is transformation only ''head driven''; successful change engages both head and heart. Relevant study guide references: Maintaining Energy and Momentum; DevOps and Transformational Leadership; Articulating and Socializing Vision.

==========

Question 7

Question Type: MultipleChoice

Which of the following is a characteristic of a well-functioning team?

Options:

A- Unable to make a decision
B- Re-prioritizes infrequently
C- Is unaccountable
D- Makes slow progress towards goals

Answer:

B

Explanation:

The correct answer is B because a well-functioning team has enough clarity, focus, and alignment that it is not constantly disrupted by changing priorities. In DevOps, frequent uncontrolled reprioritization damages flow, creates context switching, increases work in progress, delays completion, and weakens accountability. A team that reprioritizes infrequently is more likely to have stable goals, clear decision

rules, and a shared understanding of what matters most.

The other options describe poor team functioning. A team unable to make decisions lacks autonomy or clarity. An unaccountable team does not own outcomes or commitments. A team making slow progress toward goals may be blocked by dependencies, unclear priorities, excessive work in progress, or weak feedback loops.

This does not mean priorities never change. DevOps teams must remain responsive to customer feedback, incidents, risk, and business needs. However, effective teams manage priority changes deliberately rather than reactively. They protect focus while still learning and adapting. Relevant study guide references: Maintaining Energy and Momentum; DevOps and Transformational Leadership; Measuring to Improve; Target Operating Models and Organizational Designs.

Question 8

Question Type: MultipleChoice

In the Power of TED, The Empowerment Dynamic, what role does the victim from the Karpman Drama Triangle become?

Options:

A- Creator
B- Challenger
C- Coach
D- Rescuer

Answer:

A

Explanation:

The correct answer is A because in The Empowerment Dynamic, the Victim role from the Karpman Drama Triangle shifts into the Creator role. The Karpman Drama Triangle describes dysfunctional interaction patterns: Victim, Persecutor, and Rescuer. These roles reinforce blame, helplessness, dependency, and reactive behavior. In DevOps transformation, such patterns are harmful because they prevent ownership, learning, and constructive problem-solving.

The Empowerment Dynamic reframes these roles into more productive alternatives. The Victim becomes the Creator, focusing on desired outcomes, choices, and personal agency. The Persecutor becomes the Challenger, provoking growth and improvement rather than blame. The Rescuer becomes the Coach, helping others develop capability rather than creating dependency.

For DevOps leaders, this model supports unlearning behaviors that keep teams stuck in blame or helplessness. Instead of saying ''operations blocks us'' or ''developers keep breaking things,'' teams learn to ask what outcome they want, what constraints exist, and what actions they can take together. Relevant study guide references: Unlearning Behaviors; DevOps and Transformational Leadership; Maintaining Energy and Momentum.

==========

Question 9

Question Type: MultipleChoice

Which of the following is a Beyond Budgeting principle?

Options:

A- Govern through rules and regulations
B- Cultivate a strong sense of belonging and organize around accountable teams
C- Use fixed and cascaded targets
D- Evaluate performance for measurement and rewards only

Answer:

B

Explanation:

The correct answer is B because Beyond Budgeting promotes adaptive management, decentralized decision-making, transparency, trust, and accountable teams. It challenges traditional annual budgeting practices that rely heavily on fixed targets, centralized control, rigid planning cycles, and performance evaluation tied narrowly to budget adherence or reward mechanisms. In DevOps, this is highly relevant because fast flow and learning require funding and governance models that support adaptability.

Cultivating belonging and organizing around accountable teams aligns with DevOps operating models where teams own products, services, or value streams end to end. These teams require enough autonomy to respond to customer feedback, operational data, changing priorities, and emerging risks. Strong belonging also supports collaboration, shared purpose, and psychological safety.

Options A, C, and D reflect traditional command-and-control budgeting and governance. Rules and regulations, fixed cascaded targets, and measurement only for rewards can drive local optimization, gaming of metrics, and resistance to learning. Beyond Budgeting instead supports relative goals, dynamic planning, empowered teams, and continuous resource allocation. Relevant study guide

references: Target Operating Models and Organizational Designs; Measuring to Improve; Becoming a DevOps Organization.

==========

Thank You for trying DevOps-Leader PDF Demo

To try our DevOps-Leader practice exam software visit link below

https://prepbolt.com/DevOps-Leader.html

Start Your DevOps-Leader Preparation

Use Coupon “SAVE50” for extra 50% discount on the purchase of Practice Test Software. Test your DevOps-Leader preparation with actual exam questions.