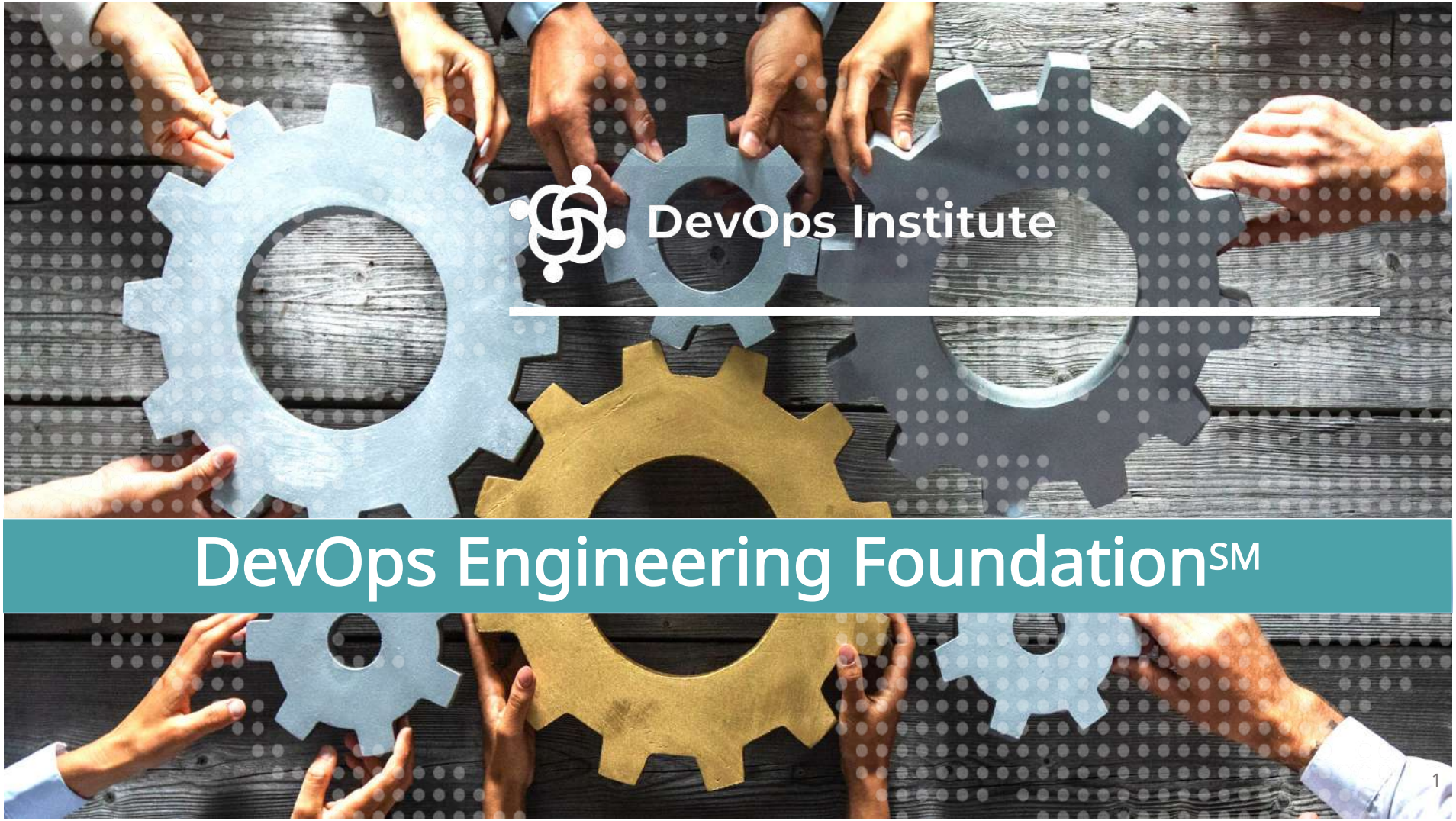


DevOps Engineering Foundation℠

DevOps Institute

Special Thanks to Our Contributors to

DevOps Engineering Foundation

DevOps Institute would like to acknowledge and thank the subject matter experts and thought leaders who contributed their valuable input, knowledge, expertise, and time to the development of this course and certification.

Marc Hornbeek

Lead Author

Tracy Bannon

Contributor

Nilardi Choudhuri

Contributor

Steve Pereira

Contributor

Tracy Ragan

Contributor

Cheryl Razzel

Contributor

DevOps Institute

Getting to Know You

Please let us know who you are:

- Name, Organization, and Role
- **DevOps Engineering Experience?**
- Why you are attending this course
- What you expect to learn

How do your engineers perceive the value of DevOps?

DevOps Engineering Foundation

Learning Objectives

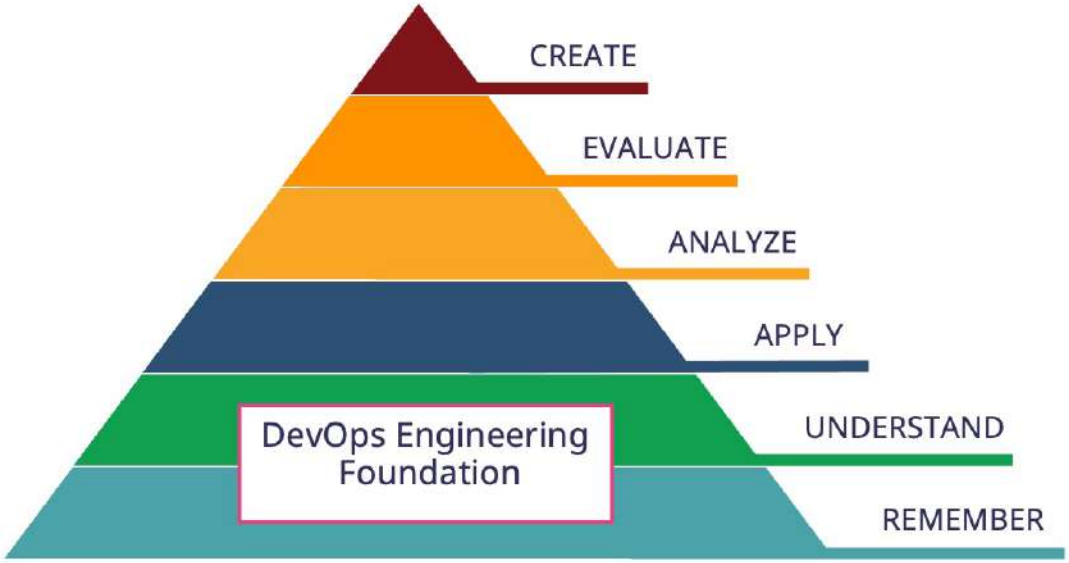
- Learn about **DevOps Engineering Foundation**
- Understand concepts of DevOps engineering
- Hear and share real life scenarios
- Have fun!

Pass the Exam for:

DevOps Engineering Foundation

- 40 multiple choice questions
- 60 minutes
- 65% is passing
- Accredited by DevOps Institute
- Get your digital badge

Course Methodology & Bloom's Taxonomy



DevOps Institute

DevOps Institute utilizes Bloom's Taxonomy to create rigorous courses and exams with clearly defined objectives and learning outcomes.

This framework is important to the learning experience. It ensures that both the instructors and participants understand the course goals, have a clear rubric for how knowledge will be assessed, and that the goals and assessments are aligned with the defined objectives.

DevOps Institute

Who is DevOps Institute?

We drive human transformation in the digital age.

DevOps Institute is a professional member association with one mission: to advance the humans of DevOps.

We create a safe and interactive ecosystem where members can network, gain knowledge, grow their careers, lead and initiate, and celebrate professional achievements.

We inspire thought leadership and knowledge by connecting and enabling the global member community to drive human transformation in the digital age.

DevOps Engineering Foundation Course Content Schedule

Day 1	Day 2
Module 1 DevOps Engineering Introduction	**Module 5** Ephemeral Elastic Infrastructures
Module 2 DevOps Technology	**Module 6** Continuous Delivery & Deployment
Module 3 Applications Architectures & Continuous Integration	**Module 7** Metrics, Monitoring, Observability, Governance
Module 4 Continuous Testing	**Module 8** DevOps Engineering Humans

DevOps INSTITUTE

Module 1
DevOps Engineering Introduction

Module 1: DevOps Engineering Introduction

- DevOps Foundation
- Principles
- Practices
- Related Frameworks
- Performance and Benefits

Video | Understanding DevOps – What is DevOps?

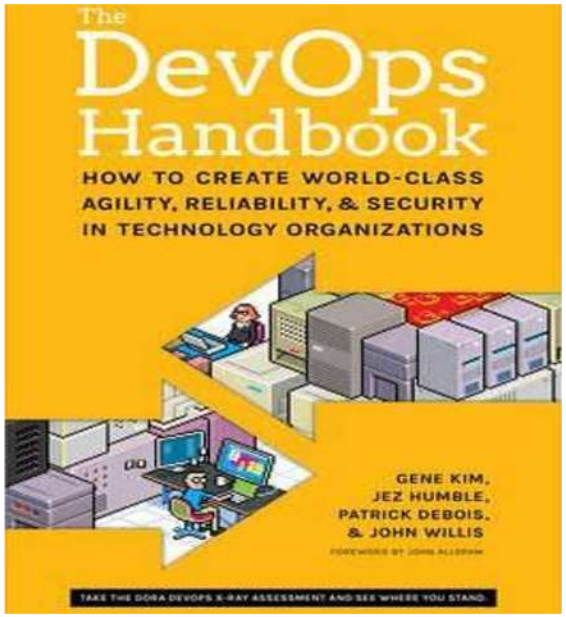
Case Story | Capital One DevOps Journey

Discussion | DevOps Principles

Exercise | DevOps Processes & Functions

What is DevOps?

"Imagine a world where product owners, Development, QA, IT Operations and Infosec work together, not only to help each other, but also to ensure that the overall organization succeeds. By working towards a common goal, they enable the fast flow of planned work into production, while achieving world-class stability, reliability, availability and security."



What DevOps is Important

- Organizations achieve true business agility.
- Accelerates innovation through automated value streams.
- Software development becomes more agile.
- Breaks silo cultures and reduces impediments to flow of communication and work.
- Helps migration of infrastructures to the cloud.
- Improves security when security practices are integrated into the value stream.

To meet these changing conditions, culture, practices and automation must become more “continuous”.

VIDEO

Understanding DevOps- What is DevOps? with Dave Farley - (13:23)

DevOps Performance

Highly evolved organizations have consistently demonstrated higher performance across four key software performance metrics.

	Low	Mid	High
Deployment frequency	Monthly or less often	Between daily and weekly	On demand (whenever we want)
Lead time for changes	Between a week and 6 months	Less than a week	Less than an hour
MTTR	Less than a week	Less than a day	Less than an hour
Change failure rate	Less than 15%	Less than 15%	Less than 5%

"Organizations that are "good at DevOps" have strong identities, clear responsibilities, autonomy over their own function, and well-defined interaction paradigms and communication channels with other teams."

DevOps Foundation

With DevOps, people across the IT organization, working together, enable fast flow, feedback, and continuous improvement of planned work into production, while achieving quality, stability, reliability, availability, security and team satisfaction.

Practices

- Automation
- Architecture
- Continuous Integration
- Continuous Delivery/Deployment
- Continuous Testing

Culture

- Safe, trusting, respectful
- Collaborative
- Data driven
- Continuous improvement
- Shared accountabilities

Values

CULTURE

AUTOMATION

LEAN

MEASUREMENT

SHARING

The Three Ways

1. Continuous Flow
2. Continuous Feedback
3. Continuous Experiment and Learning

Frameworks

- Agile and SAFe
- Lean
- ITSM
- DevSecOps
- SRE - Site Reliability Engineering
- VSM - Value Stream Management

Automation is an Essential Element

Automation enables agility, consistency, speed and reliability.

Shared decision-making, access to and an understanding of toolchains and other automation streamlines software delivery and prepares Ops for the long run.

DevOps Engineering Principles

Continuous Testing

Continuous testing is the process of executing automated tests as part of the pipeline to obtain immediate feedback on the business risks associated with a software release candidate.

	Low	Medium	High	Elite
Automated build	64%	81%	91%	92%
Automated unit tests	57%	66%	84%	87%
Automated acceptance tests	28%	38%	48%	58%
Automated performance tests	18%	23%	18%	28%
Automated security tests	15%	28%	25%	31%
Automated provisioning and deployment to testing environments	39%	54%	68%	72%

"Shifting left" is about building quality into the software development process. When you shift left, fewer things break in production, because any issues are detected and resolved earlier.

Continuous Integration (CI)

Continuous integration (CI) is a development practice that requires developers to commit code into a shared repository (master/trunk) at least daily.

Each commit is:

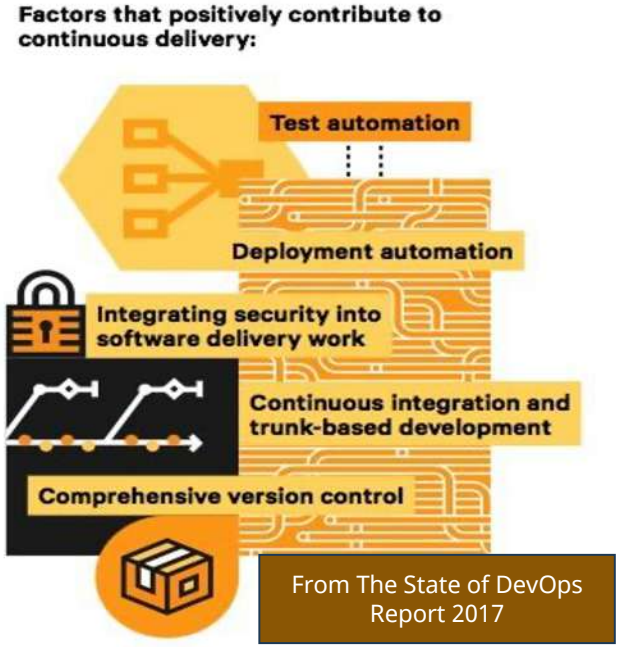
- validated by an automated build, automated unit, integration and acceptance tests.
- is dependent on consistent coding standards.
- requires version control repositories and CI servers to collect, build and test committed code together.
- runs on production-like environments.
- enables early detection and quick remediation of errors from code changes before moving to production.

While mostly associated with agile software development, waterfall approaches can also take advantage of continuous integration and testing practices.

Continuous Delivery (CD)

Continuous delivery is a methodology that focuses on making sure software is *always in a releasable state* throughout its lifecycle.

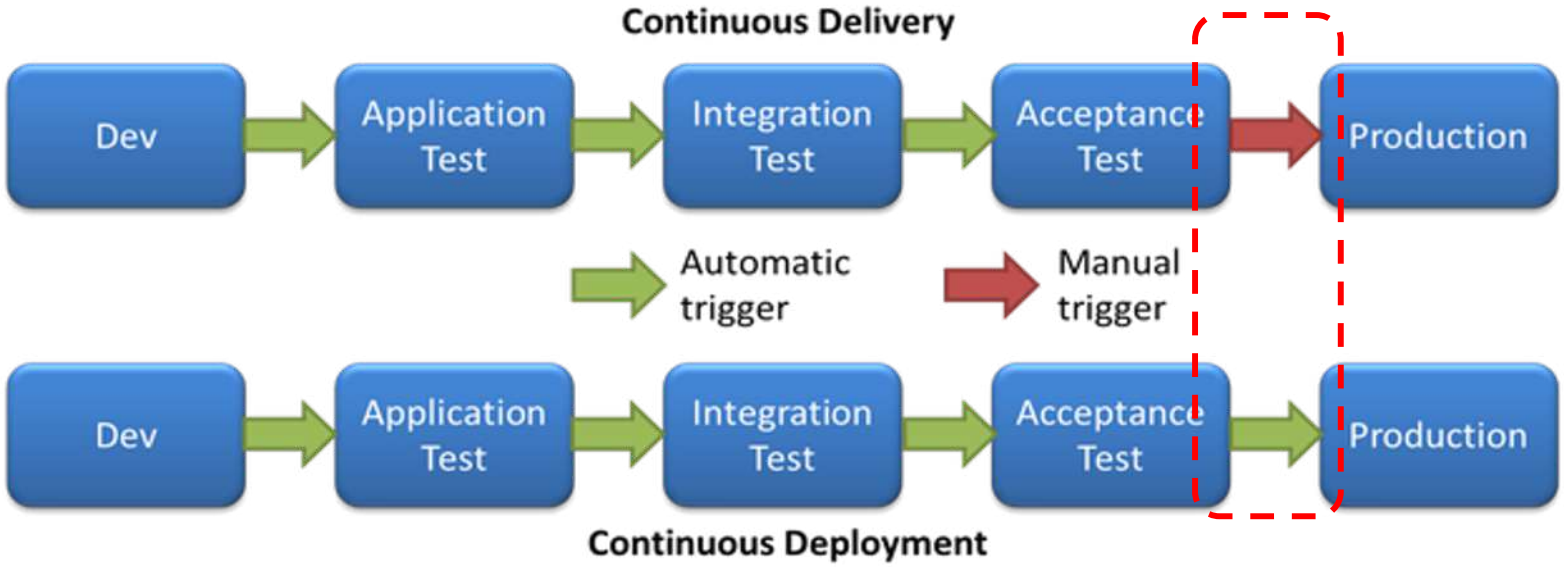
- Takes continuous integration to next level
- Provides fast, automated feedback on a system's production-readiness
- Prioritizes keeping software releasable/ deployable over working on new features
- Relies on deployment pipeline that enables push-button deployments on demand
- Reduces the cost, time, and risk of delivering incremental changes



From The State of DevOps Report 2017

Continuous Delivery & Continuous Deployment

Continuous delivery is a prerequisite for continuous deployment into production.



From: Mirco Hering: notafactoryanymore.com, author of 'DevOps for the Modern Enterprise'

DevSecOps

"I believe the DevOps movement is a new fertile soil from which the build-security-in concept can be reborn, renamed, and remade." Larry Maccherone

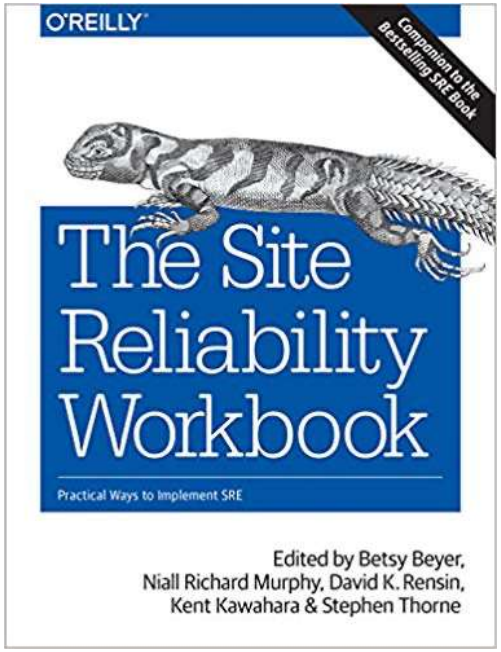
Goal: Safely distributed security decisions at speed and scale

VALUES

- **Build security in** *more than* bolt it on
- **Rely on empowered development teams** *more than* security specialists
- **Implement features securely** *more than* security features
- **Use tools as feedback for learning** *more than* end-of-phase stage gates
- **Build on culture change** *more than* policy enforcement

"Through Security as Code, we have and will learn that there is simply a better way for security practitioners, like us, to operate and contribute value with less friction. We know we must adapt our ways quickly and foster innovation to ensure data security and privacy issues are not left behind because we were too slow to change." ~**2019 MP Infotech Corp**

Site Reliability Engineering (SRE)



- The goal is to create ultra-scalable and highly reliable distributed software systems
- SRE's spend **50% of their time doing "ops" related work** such as issue resolution, on-call, and manual interventions
- SRE's spend **50% of their time on development tasks** such as new features, scaling or automation
- Monitoring, alerting and automation are a large part of SRE

Value Stream Management (VSM)

FLOW

Flow is the journey of work from idea to realization. Its travel should be friction-free. It's a continuous steady stream of value for customers.

- Know how long it takes to deliver value to your customers
- Accelerate value delivery to customers
- Gain insights into where waste is
- Collaborate product lifecycle
- Have data-driven conversations
- Traceability through your toolchain

REALIZATION

Realization is the fulfillment of desired outcomes. It's when a customer experiences the value intended.

- Estimate the value your work will deliver
- Build hypotheses for the value of work
- Understand how much value work realized
- Amplify and shorten feedback loops
- Make better informed business decisions
- Maximize value stream's profitability

Case Story: Capital One DevOps Journey


Sr, SW Engineering Manager, Capital One

Large USA Financial Services company. Five years DevOps Journey.

A small Enterprise Jenkins platform team supports 7,000 engineers, half a million Jenkins pipelines and 50,000 builds, tests, and deploys every daily.

Takeaways

Our secret to successful DevOps journey is the use of DevOps practices that provide flexibility and efficiency.

Benefits

- Automated software delivery.
- Increase operational stability.
- Improved developer experience.

Module 1 Quiz

1	Which of the following are attributes that describe DevOps?	a) Achieve world class stability reliability, availability and security. b) Product owners, Development, QA, IT Operations and InfoSec work together. c) Working on a common goal to enable fast flow of planned work. d) A, B, and C
2	What does CALMS stand for?	a) Collaboration, Automation, Lean, Monitoring, Shift Left b) Culture, Automation, Lean, Measurement, Sharing c) Collaboration, Automation, Left Shift, Measurement, Sharing d) Continuous, Automation, Lean, Measurement, Sharing
3	Which best described The Second Way of DevOps?	a) Understand and increase the flow of work. b) Continuous experimentation and learning. c) Create short feedback loops. d) An alternative to The First Way.
4	Which of these is NOT a primary practice of DevOps?	a) Automation b) Continuous Delivery c) Continuous Planning d) Continuous Integration
5	Which of these is a value of DevSecOps?	a) Build Security In. b) Security specialists are solely responsible for security. c) Security tools are operated in a separate pipeline. d) Security controls operate during continuous deployment.

Module 1 Quiz Answers

1	Which of the following are attributes that describe DevOps?	a) Achieve world class stability reliability, availability and security. b) Product owners, Development, QA, IT Operations and InfoSec work together. c) Working on a common goal to enable fast flow of planned work. **d) A, B, and C**
2	What does CALMS stand for?	a) Collaboration, Automation, Lean, Monitoring, Shift Left **b) Culture, Automation, Lean, Measurement, Sharing** c) Collaboration, Automation, Left Shift, Measurement, Sharing d) Continuous, Automation, Lean, Measurement, Sharing
3	Which best described The Second Way of DevOps?	a) Understand and increase the flow of work. b) Continuous experimentation and learning. **c) Create short feedback loops.** d) An alternative to The First Way.
4	Which of these is NOT a primary practice of DevOps?	a) Automation b) Continuous Delivery **c) Continuous Planning** d) Continuous Integration
5	Which of these is a value of DevSecOps?	**a) Build Security In** b) Security specialists are solely responsible for security. c) Security tools are operated in a separate pipeline. d) Security controls operate during continuous deployment.