DevOps
INSTITUTE

DevOps Foundation® Exam Study Guide
DevOps Foundation® Przewodnik po badaniach egzaminacyjnych

DevOps Institute's SKIL Framework

DevOps Institute is dedicated to advancing the human elements of DevOps success through our human-centered SKIL framework of Skills, Knowledge, Ideas and Learning.

We develop, accredit and orchestrate SKIL through certifications, research, learning opportunities, events and community connections.

Visit the SKILUp℠ CAFE at www.devopsinstitute.com to learn more.

Join Us!

Become a member and join the fastest growing global community of DevOps practitioners and professionals.

The DevOps Institute continuous learning community is your go-to hub for all things DevOps, so get ready to learn, engage, and inspire.

Visit https: www.devopsinstitute.com/become-a-community-member to join today.

You belong.

DevOps Institute

DevOps Institute is dedicated to advancing the human elements of DevOps success. We fulfill our mission through our SKIL framework of Skills, Knowledge, Ideas and Learning.

Certification is one means of showcasing your skills. While we strongly support formal training as the best learning experience and method for certification preparation, DevOps Institute also recognizes that humans learn in different ways from different resources and experiences. As the defacto certification body for DevOps, DevOps Institute has now removed the barrier to certification by removing formal training prerequisites and opening our testing program to anyone who believes that they have the topical knowledge and experience to pass one or more of our certification exams.

This examination study guide will help test-takers prepare by defining the scope of the exam and includes the following:

- Course Description
- Examination Requirements
- DevOps Glossary of Terms
- Value Added Resources
- Sample Exam(s) with Answer Key

These assets provide a guideline for the topics, concepts, vocabulary and definitions that the exam candidate is expected to know and understand in order to pass the exam. The knowledge itself will need to be gained on its own or through training by one of our Global Education Partners.

Test-takers who successfully pass the exam will also receive a certificate and digital badge from DevOps Institute, acknowledging their achievement, that can be shared with their professional online networks.

If you have any questions, please contact our DevOps Institute Customer Service team at CustomerService@DevOpsInstitute.com.

DevOps Foundation®

Wymagania egzaminacyjne

Certyfikat DevOps Foundation®

DevOps Foundation to niezależna certyfikacja od DevOps Institute. Celem tego kursu i związanego z nim egzaminu jest przekazanie, przetestowanie i weryfikacja wiedzy na temat podstawowych pojęć, zasad i praktyk DevOps. Definicje, koncepcje i praktyki DevOps są udokumentowane w podręczniku uczestnika szkolenia. Kurs DevOps Foundation ma na celu zapewnienie uczestnikom zrozumienia podstawowych pojęć DevOps oraz tego, w jaki sposób DevOps może być użyty do poprawy komunikacji, współdziałania i integracji między twórcami oprogramowania i specjalistami od operacji IT.

Wymagania egzaminacyjne

Chociaż nie ma formalnych warunków wstępnych do egzaminu, DevOps Institute zdecydowanie zaleca kandydatom do egzaminu DevOps Foundation co następuje:

- Zaleca się, aby kandydaci ukończyli co najmniej 16-godzinne formalne, akredytowane szkolenie, przeprowadzone przez Zarejestrowanego Partnera Edukacyjnego (REP) zatwierdzonego przez DevOps Institute;
- Zaleca się, aby kandydaci przeznaczyli co najmniej 6 godzin na naukę własną, przeglądając listę pojęć i odpowiednie obszary podręcznika szkolenia oraz wypełniając przykładowe egzaminy próbne.

Organizacja egzaminu

Egzamin DevOps Foundation jest akredytowany, zarządzany i administrowany zgodnie ze ścisłymi zasadami zdefiniowanymi przez DevOps Institute.

Poziom trudności

Certyfikacja DevOps Foundation wykorzystuje taksonomię celów nauczania Blooma zarówno w odniesieniu do zawartości merytorycznej szkolenia, jak i egzaminu:

- Egzamin DevOps Foundation zawiera pytania na poziomie Bloom 1, które sprawdzają **wiedzę** uczestników na temat koncepcji DevOps i terminologii (patrz lista poniżej);
- Egzamin zawiera również pytania na poziomie Bloom 2, które sprawdzają **rozumienie** tych pojęć przez uczestników w podanym kontekście.

Format egzaminu

Kandydaci muszą uzyskać pozytywny wynik na egzaminie, aby uzyskać certyfikat DevOps Foundation.

Typ egzaminu	40 pytań jednokrotnego wyboru (1 prawidłowa odpowiedź z 4)
Czas trwania	60 minut
Wymagania wstępne	Zaleca się, aby kandydaci ukończyli kurs DevOps Foundation od Zarejestrowanego Partnera Edukacyjnego DevOps Institute (REP)
Nadzór	Nie
Otwarte książki	Tak
Próg zdawalności	65%
Sposób realizacji	Strona internetowa
Odznaka	DevOps Foundation Certified

Obszary merytoryczne na egzaminie i waga pytań

Egzamin DevOps Foundation wymaga znajomości obszarów merytorycznych opisanych poniżej.

Moduł	Opis	Maksymalna liczba pytań
DOFD – 1 Exploring DevOps	Purpose, objectives and business value of DevOps	5
DOFD – 2 Core DevOps Principles	The Three Ways	4
DOFD – 3 Key DevOps Practices	Emerging DevOps Practices such as continuous delivery and continuous integration	7
DOFD – 4 Business & Technology Frameworks	The relationship between relevant frameworks and standards and DevOps	7
DOFD – 5 DevOps Values - Culture, Behaviors & Operating Models	Characteristics of a DevOps culture and of culture change	6
DOFD – 6 DevOps Values - Automation & Architecting DevOps Toolchains	The Deployment Pipeline, DevOps toolchains and other automation considerations	5
DOFD – 7 DevOps Values – Measurement, Metrics & Reporting	Common DevOps practices and related processes	2
DOFD – 8 DevOps Values: Sharing, Shadowing and Reporting	Responsibilities of key roles and considerations relative to organizational structure. Getting started - adoption challenges, risks, critical success factors and key performance measures	4

Lista pojęć i definicji

Po ukończeniu kursu DevOps Foundation od kandydata oczekuje się zrozumienia następujących pojęć i definicji DevOps na poziomie Bloom 1 i 2.

- Agile Manifesto
- Agile service management
- Agile software development
- Application Programming Interface (API)
- CALMS
- Change failure rate
- Change fatigue
- Change lead time
- ChatOps
- Code commit
- Collaboration and communication
- Collective Body of Knowledge (CBOK)
- Constraint
- Containers
- Continuous integration
- Continuous delivery
- Continuous deployment
- Continuous Testing
- Cultural debt
- Cycle time
- Deployment pipeline
- Deployment frequency
- DevSecOps
- DevOps
- DevOps metrics
- DevOps stakeholders
- DevOps teams
- DevOps roles
- Dojo
- employee Net Promoter Score (eNPS)
- Epics
- Flow
- Golden Circle
- High-performing organizations
- Impediment
- Immersion
- Improvement kata
- IT service management
- Kanban
- Kubernetes
- Kubler-Ross Change Curve
- Lean production
- Lean thinking
- Lean tools
- Lean types of Waste (DOWNTIME)
- Mean Time to Repair/Recover (MTTR)
- Microservices
- Open source
- Organizational culture
- Organizational considerations
- Resilience Engineering
- Scaled Agile Framework (SAFe)
- Scrum
- Scrum roles, artifacts and events
- Service
- Shift left
- Simian Army/Chaos Monkey
- Single Point of Failure (SPOF)
- Test driven development
- Testing (unit, acceptance, integration)
- The Three Ways
- Theory of Constraints
- Thomas-Kilmann Conflict Modes
- DevOps toolchain
- Value stream mapping
- Velocity
- Waste
- Waterfall

DevOps INSTITUTE

DEVOPS
GLOSSARY OF TERMS

This glossary is provided for reference only as it contains key terms that may or may not be examinable.

DevOps Glossary of Terms

Term	Definition	Course Appearances
12-Factor App Design	A methodology for building modern, scalable, maintainable software-as-a-service applications.	Continuous Delivery Ecosystem Foundation
2-Factor or 2-Step Authentication	Two-Factor Authentication, also known as 2FA or TFA or Two-Step Authentication is when a user provides two authentication factors; usually firstly a password and then a second layer of verification such as a code texted to their device, shared secret, physical token or biometrics.	DevSecOps Foundation
A/B Testing	Deploy different versions of an EUT to different customers and let the customer feedback determine which is best.	Continuous Delivery Ecosystem Foundation
A3 Problem Solving	A structured problem-solving approach that uses a lean tool called the A3 Problem-Solving Report. The term "A3" represents the paper size historically used for the report (a size roughly equivalent to 11" x 17").	DevOps Foundation
Access Management	Granting an authenticated identity access to an authorized resource (e.g., data, service, environment) based on defined criteria (e.g., a mapped role), while preventing an unauthorized identity access to a resource.	DevSecOps Foundation
Access Provisioning	Access provisioning is the process of coordinating the creation of user accounts, e-mail authorizations in the form of rules and roles, and other tasks such as provisioning of physical resources associated with enabling new users to systems or environments.	DevSecOps Foundation
Administration Testing	The purpose of the test is to determine if an End User Test (EUT) is able to process administration tasks as expected.	Continuous Delivery Ecosystem Foundation

Advice Process	Any person making a decision must seek advice from everyone meaningfully affected by the decision and people with expertise in the matter. Advice received must be taken into consideration, though it does not have to be accepted or followed. The objective of the advice process is not to form consensus, but to inform the decision-maker so that they can make the best decision possible. Failure to follow the advice process undermines trust and unnecessarily introduces risk to the business.	DevSecOps Foundation
Agile	A project management method for complex projects that divides tasks into small "sprints" of work with frequent reassessment and adaptation of plans.	Certified Agile Process Owner, Certified Agile Service Manager, Site Reliability Engineering
Agile (adjective)	Able to move quickly and easily; well-coordinated. Able to think and understand quickly; able to solve problems and have new ideas.	DevOps Foundation, DevSecOps Foundation
Agile Coach	Help teams master Agile development and DevOps practices; enables productive ways of working and collaboration.	DevOps Leader
Agile Enterprise	Fast moving, flexible and robust company capable of rapid response to unexpected challenges, events, and opportunities.	DevOps Foundation, DevSecOps Foundation
Agile Manifesto	A formal proclamation of values and principles to guide an iterative and people-centric approach to software development. http://agilemanifesto.org	DevOps Foundation
Agile Portfolio Management	Involves evaluating in-flight projects and proposed future initiatives to shape and govern the ongoing investment in projects and discretionary work. CA's Agile Central and VersionOne are examples.	Site Reliability Engineering
Agile Principles	The twelve principles that underpin the Agile Manifesto.	Certified Agile Service Manager
Agile Process Design	The aspect of Agile Service Management (Agile SM) that applies the same Agile approach to process design as developers do to software development.	Certified Agile Service Manager
Agile Process Improvement	The aspect of Agile SM that aligns Agile values with ITSM processes through continuous improvement.	Certified Agile Service Manager

Agile Process Owner	An ITSM or other type of process owner that uses Agile and Scrum principles and practices to design, manage and measure individual processes.	DevOps Foundation
Agile Service Management	Framework that ensures that ITSM processes reflect Agile values and are designed with "just enough" control and structure in order to effectively and efficiently deliver services that facilitate customer outcomes when and how they are needed.	Certified Agile Service Manager
Agile Service Management Artifacts	Process Backlog, Sprint Backlog, Burndown Chart, Process Increment	Certified Agile Process Owner
Agile Service Management Events	Process Planning Meeting (optional), Sprint Planning Meeting, Sprint, Daily Scrum, Sprint Review, Sprint Retrospective	Certified Agile Process Owner
Agile Service Management Roles	Process Owner, Process Improvement Team (Team) and Agile Service Manager. See also Scrum Roles.	Certified Agile Process Owner
Agile Service Manager	The operational equivalent to Dev's ScrumMaster. A role within an IT organization that understands how to leverage Agile and Scrum methods to improve the design, speed and agility of ITSM processes.	DevOps Foundation
Agile Software Development	Group of software development methods in which requirements and solutions evolve through collaboration between self-organizing, cross-functional teams. Usually applied using the Scrum or Scaled Agile Framework approach.	Continuous Delivery Ecosystem Foundation, DevOps Foundation, DevSecOps Foundation
Amazon Web Services (AWS)	Amazon Web Services (*AWS*) is a secure cloud services platform, offering compute power, database storage, content delivery and other functionality to help businesses scale and grow.	DevSecOps Foundation, Site Reliability Engineering
Analytics	Test results processed and presented in an organized manner in accordance with analysis methods and criterion.	Continuous Delivery Ecosystem Foundation, Continuous Testing Foundation
Andon	A system gives an assembly line worker the ability, and moreover the empowerment, to stop production when a defect is found, and immediately call for assistance.	Continuous Delivery Ecosystem Foundation
Anti-pattern	A commonly reinvented but poor solution to a problem.	DevOps Foundation

Anti-fragility	Antifragility is a property of systems that increases its capability to thrive as a result of stressors, shocks, volatility, noise, mistakes, faults, attacks, or failures.	DevOps Foundation, Site Reliability Engineering
API Testing	The purpose of the test is to determine if an API for an EUT functions as expected.	Continuous Delivery Ecosystem Foundation, Continuous Testing Foundation
Application Performance Management (APM)	APM is the monitoring and management of performance and availability of software applications. APM strives to detect and diagnose complex application performance problems to maintain an expected level of service.	Site Reliability Engineering
Application Programming Interface (API)	A set of protocols used to create applications for a specific OS or as an interface between modules or applications.	DevOps Foundation, DevSecOps Foundation
Application Programming Interface (API) Testing	The purpose of the test is to determine if an API for an EUT functions as expected.	Continuous Delivery Ecosystem Foundation
Application Release	Controlled continuous delivery pipeline capabilities including automation (release upon code commit).	Continuous Delivery Ecosystem Foundation
Application Release Automation (ARA) or Orchestration (ARO)	Controlled continuous delivery pipeline capabilities including automation (release upon code commit), environment modeling (end-to-end pipeline stages, and deploy application binaries, packages or other artifacts to target environments) and release coordination (project, calendar and scheduling management, integrate with change control and/or IT service support management).	Continuous Delivery Ecosystem Foundation
Application Test Driven Development (ATDD)	Acceptance Test Driven Development (ATDD) is a practice in which the whole team collaboratively discusses acceptance criteria, with examples, and then distills them into a set of concrete acceptance tests before development begins.	Continuous Delivery Ecosystem Foundation
Application Testing	The purpose of the test is to determine if an application is performing according to its requirements and expected behaviors.	Continuous Delivery Ecosystem Foundation

Application Under Test (AUT)	The EUT is a software application. E.g. Business application is being tested.	Continuous Delivery Ecosystem Foundation, Continuous Testing Foundation
Architecture	The fundamental underlying design of computer hardware, software or both in combination.	DevSecOps Foundation
Artifact	Any element in a software development project including documentation, test plans, images, data files and executable modules.	Continuous Delivery Ecosystem Foundation, DevOps Foundation, DevSecOps Foundation
Artifact Repository	Store for binaries, reports and metadata. Example tools include: JFrog Artifactory, Sonatype Nexus.	Continuous Delivery Ecosystem Foundation, DevOps Foundation
Attack path	The chain of weaknesses a threat may exploit to achieve the attacker's objective. For example, an attack path may start by compromising a user's credentials, which are then used in a vulnerable system to escalate privileges, which in turn is used to access a protected database of information, which is copied out to an attacker's own server(s).	DevSecOps Foundation
Audit Management	The use of automated tools to ensure products and services are auditable, including keeping audit logs of build, test and deploy activities, auditing configurations and users, as well as log files from production operations.	Site Reliability Engineering
Authentication	The process of verifying an asserted identity. Authentication can be based on what you know (e.g., password or PIN), what you have (token or one-time code), what you are (biometrics) or contextual information.	DevSecOps Foundation
Authorization	The process of granting roles to users to have access to resources.	DevSecOps Foundation
Auto-DevOps	Auto DevOps brings DevOps best practices to your project by automatically configuring software development lifecycles. It automatically detects, builds, tests, deploys, and monitors applications.	Site Reliability Engineering
Auto-scaling	The ability to automatically and elastically scale and de-scale infrastructure depending on traffic and capacity variations while maintaining control of costs.	Continuous Delivery Ecosystem Foundation

Automated rollback	If a failure is detected during a deployment, an operator (or an automated process) will verify the failure and rollback the failing release to the previous known working state.	Site Reliability Engineering
Availability	Availability is the proportion of time a system is in a functioning condition and therefore available (to users) to be used.	Site Reliability Engineering
Backdoor	A backdoor bypasses the usual authentication used to access a system. Its purpose is to grant the cybercriminals future access to the system even if the organization has remediated the vulnerability initially used to attack the system.	DevSecOps Foundation
Backlog	Requirements for a system, expressed as a prioritized list of product backlog items usually in the form of 'User Stories'. The product backlog is prioritized by the Product Owner and should include functional, non-functional and technical team-generated requirements.	Continuous Delivery Ecosystem Foundation, DevOps Foundation
Basic Security Hygiene	A common set of minimum-security practices that must be applied to all environments without exception. Practices include basic network security (firewalls and monitoring), hardening, vulnerability and patch management, logging and monitoring, basic policies and enforcement (may be implemented under a "policies as code" approach), and identity and access management.	DevSecOps Foundation
Batch Sizes	Refers to the volume of features involved in a single code release.	DevOps Leader
Bateson Stakeholder Map	A tool for mapping stakeholder's engagement with the initiative in progress.	DevOps Leader
Behavior Driven Development (BDD)	Test cases are created by simulating an EUT's externally observable inputs, and outputs. Example tool: Cucumber.	Continuous Delivery Ecosystem Foundation
Beyond Budgeting	A management model that looks beyond command-and-control towards a more empowered and adaptive state.	DevOps Leader
Black-Box	Test case only uses knowledge of externally observable behaviors of an EUT.	Continuous Delivery Ecosystem Foundation, Continuous Testing Foundation

Blameless post mortems	A process through which engineers whose actions have contributed to a service incident can give a detailed account of what they did without fear of punishment or retribution.	Site Reliability Engineering
Blast Radius	Used for impact analysis of service incidents. When a particular IT service fails, the users, customers, other dependent services that are affected.	Site Reliability Engineering
Blue/Green Testing or Deployments	Taking software from the final stage of testing to live production using two environments labelled Blue and Green. Once the software is working in the green environment, switch the router so that all incoming requests go to the green environment - the blue one is now idle.	Continuous Delivery Ecosystem Foundation, Continuous Testing Foundation
Bug	An error or defect in software that results in an unexpected or system-degrading condition.	DevSecOps Foundation
Bureaucratic Culture	Bureaucratic organizations are likely to use standard channels or procedures which may be insufficient in a crisis (Westrum).	DevOps Leader
Burndown Chart	Chart showing the evolution of remaining effort against time.	Certified Agile Service Manager, DevOps Foundation
Bursting	Public cloud resources are added as needed to temporarily increase the total computing capacity of a private cloud.	Continuous Delivery Ecosystem Foundation
Business Case	Justification for a proposed project or undertaking on the basis of its expected commercial benefit.	DevOps Leader
Business Continuity	Business continuity is an organization's ability to ensure operations and core business functions are not severely impacted by a disaster or unplanned incident that take critical services offline.	Site Reliability Engineering
Business Transformation	Changing how the business functions. Making this a reality means changing culture, processes, and technologies in order to better align everyone around delivering on the organization's mission.	DevSecOps Foundation
Business Value	The benefit of an approach to key business KPIs.	DevOps Leader

Cadence	Flow or rhythm of events.	DevOps Foundation, DevOps Leader, DevSecOps Foundation
CALMS Model	Considered the pillars or values of DevOps: Culture, Automation, Lean, Measurement, Sharing (as put forth by John Willis, Damon Edwards and Jez Humble).	DevOps Foundation
Canary Testing	A canary (also called a canary test) is a push of code changes to a small number of end users who have not volunteered to test anything. Similar to incremental rollout, it is where a small portion of the user base is updated to a new version first. This subset, the canaries, then serve as the proverbial "canary in the coal mine". If something goes wrong then a release is rolled back and only a small subset of the users are impacted.	Continuous Delivery Ecosystem Foundation, Site Reliability Engineering
Capacity Test	The purpose of the test is to determine if the EUT can handle expected loads such as number of users, number of sessions, aggregate bandwidth.	Continuous Delivery Ecosystem Foundation
Capture-Replay	Test cases are created by capturing live interactions with the EUT, in a format that can be replayed by a tool. E.g. Selenium	Continuous Delivery Ecosystem Foundation, Continuous Testing Foundation
Carrots	Positive incentives, for encouraging and rewarding desired behaviors.	DevSecOps Foundation
Chain of Goals	A method designed by Roman Pichler of ensuring that goals are linked and shared at all levels through the product development process.	DevOps Leader
Change	Addition, modification or removal of anything that could have an effect on IT services. (ITIL® definition)	DevOps Foundation, DevSecOps Foundation
Change Failure Rate	A measure of the percentage of failed/rolled back changes.	Continuous Delivery Ecosystem Foundation, DevOps Foundation
Change Fatigue	A general sense of apathy or passive resignation towards organizational changes by individuals or teams.	DevSecOps Foundation
Change Lead Time	A measure of the time from a request for change to delivery of the change.	DevOps Foundation
Change Leader Development Model	Jim Canterucci's model for five levels of change leader capability.	DevOps Leader

Change Management	Process that controls all changes throughout their lifecycle. (ITIL definition)	DevOps Foundation, DevOps Leader, DevSecOps Foundation
Change Management (Organizational)	An approach to shifting or transitioning individuals, teams & organizations from a current state to a desired future state. Includes the process, tools & techniques to manage the people-side of change to achieve the required business outcome(s).	DevOps Leader
Change-based Test Selection Method	Tests are selected according to a criterion that matches attributes of tests to attributes of the code that is changed in a build.	Continuous Delivery Ecosystem Foundation, Continuous Testing Foundation
Chaos Engineering	The discipline of experimenting on a software system in production in order to build confidence in the system's capability to withstand turbulent and unexpected conditions.	Site Reliability Engineering
Chapter Lead	A squad line manager in the Spotify model who is responsible for traditional people management duties, is involved in day to day work and grows individual and chapter competence.	DevOps Leader
Chapters	A small family of people having similar skills and who work within the same general competency area within the same tribe. Chapters meet regularly to discuss challenges and areas of expertise in order to promote sharing, skill development, re-use and problem solving.	DevOps Leader
ChatOps	An approach to managing technical and business operations (coined by GitHub) that involves a combination of group chat and integration with DevOps tools. Example tools include: Atlassian HipChat/Stride, Microsoft Teams, Slack.	Continuous Delivery Ecosystem Foundation, DevOps Foundation, Continuous Testing Foundation, Site Reliability Engineering
Check-in	Action of submitting a software change into a system version management system.	Continuous Delivery Ecosystem Foundation, Continuous Testing Foundation
CI Regression Test	A subset of regression tests that are run immediately after a software component is built. Same as Smoke Test.	Continuous Delivery Ecosystem Foundation

Clear-Box	Same as Glass-Box Testing and White-Box Testing.	Continuous Delivery Ecosystem Foundation, Continuous Testing Foundation
Cloud Computing	The practice of using remote servers hosted on the internet to host applications rather than local servers in a private datacenter.	DevSecOps Foundation, Site Reliability Engineering
Cloud-Native	Native cloud applications (NCA) are designed for cloud computing.	Continuous Delivery Ecosystem Foundation
Cloudbees	Cloudbees is a commercially supported proprietary automation framework tool which works with and enhances Jenkins by providing enterprise levels support and add-on functionality.	Continuous Testing Foundation
Cluster Cost Optimization	Tools like Kubecost, Replex, Cloudability use monitoring to analyze container clusters and optimize the resource deployment model.	Site Reliability Engineering
Cluster Monitoring	Tools that let you know the health of your deployment environments running in clusters such as Kubernetes.	Site Reliability Engineering
Clustering	A group of computers (called nodes or members) work together as a cluster connected through a fast network acting as a single system.	Continuous Delivery Ecosystem Foundation
Code Coverage	A measure of white box test coverage by counting code units that are executed by a test. The code unit may be a code statement, a code branch, or control path or data path through a code module.	Continuous Delivery Ecosystem Foundation, Continuous Testing Foundation
Code Quality	See also static code analysis, Sonar and Checkmarks are examples of tools that automatically check the seven main dimensions of code quality – comments, architecture, duplication, unit test coverage, complexity, potential defects, language rules.	Site Reliability Engineering
Code Repository	A repository where developers can commit and collaborate on their code. It also tracks historical versions and potentially identifies conflicting versions of the same code. Also referred to as "repository" or "repo."	DevSecOps Foundation
Code Review	Software engineers inspect each other's source code to detect coding or code formatting errors.	Continuous Delivery Ecosystem Foundation, Continuous Testing Foundation

Cognitive Bias	Cognitive bias is a limitation in objective thinking that is caused by the tendency for the human brain to perceive information through a filter of personal experience and preferences: a systematic pattern of deviation from norm or rationality in judgment.	DevOps Leader
Collaboration	People jointly working with others towards a common goal.	DevOps Foundation, DevSecOps Foundation
Collaborative Culture	A culture that applies to everyone which incorporates an expected set of behaviors, language and accepted ways of working with each other reinforcement by leadership.	Continuous Delivery Ecosystem Foundation
Compatibility Test	Test with the purpose to determine if and EUT interoperates with another EUT such as peer-to-peer applications or protocols.	Continuous Delivery Ecosystem Foundation, Continuous Testing Foundation
Configuration Management	Configuration management (CM) is a systems engineering process for establishing and maintaining consistency of a product's performance, functional, and physical attributes with its requirements, design, and operational information throughout its life.	Continuous Delivery Ecosystem Foundation, DevOps Foundation, DevSecOps Foundation
Conformance Test	The purpose of the test is to determine if an EUT complies to a standard.	Continuous Delivery Ecosystem Foundation, Continuous Testing Foundation
Constraint	Limitation or restriction; something that constrains. See also *bottleneck*.	DevOps Foundation, DevSecOps Foundation
Container	A way of packaging software into lightweight, stand-alone, executable packages including everything needed to run it (code, runtime, system tools, system libraries, settings) for development, shipment and deployment.	DevOps Foundation, DevSecOps Foundation, Site Reliability Engineering
Container Network Security	Used to prove that any app that can be run on a container cluster with any other app can be confident that there is no unintended use of the other app or any unintended network traffic between them.	Site Reliability Engineering

Container Registry	Secure and private registry for Container images. Typically allowing for easy upload and download of images from the build tools. Docker Hub, Artifactory, Nexus are examples.	Site Reliability Engineering
Container Scanning	When building a Container image for your application, tools can run a security scan to ensure it does not have any known vulnerability in the environment where your code is shipped. Blackduck, Synopsis, Synk, Claire and klar are examples.	Site Reliability Engineering
Continual Service Improvement (CSI)	One of the ITIL Core publications and a stage of the service lifecycle.	DevOps Foundation
Continuous Delivery (CD)	A methodology that focuses on making sure software is always in a releasable state throughout its lifecycle.	Certified Agile Service Manager, Continuous Delivery Ecosystem Foundation, DevOps Foundation, DevSecOps Foundation, Continuous Testing Foundation
Continuous Delivery (CD) Architect	A person who is responsible to guide the implementation and best practices for a continuous delivery pipeline.	Continuous Delivery Ecosystem Foundation
Continuous Delivery Pipeline	A continuous delivery pipeline refers to the series of processes which are performed on product changes in stages. A change is injected at the beginning of the pipeline. A change may be new versions of code, data or images for applications. Each stage processes the artifacts resulting from the prior stage. The last stage results in deployment to production.	Continuous Delivery Ecosystem Foundation, DevOps Foundation Course, DevOps Leader
Continuous Delivery Pipeline Stage	Each process in a continuous delivery pipeline. These are not standard. Examples are Design: determine implementation changes; Creation: implement an unintegrated version of design changes; Integration: merge	Continuous Delivery Ecosystem Foundation
Continuous Deployment	A set of practices that enable every change that passes automated tests to be automatically deployed to production.	DevOps Foundation, DevSecOps Foundation
Continuous Flow	Smoothly moving people or products from the first step of a process to the last with minimal (or no) buffers between steps.	DevOps Foundation, DevOps Leader, DevSecOps Foundation

Continuous Improvement	Based on Deming's Plan-Do-Check-Act, a model for ensure ongoing efforts to improve products, processes and services.	DevOps Foundation, DevOps Leader
Continuous Integration (CI)	A development practice that requires developers to merge their code into trunk or master ideally at least daily and perform tests (i.e. unit, integration and acceptance) at every code commit.	Certified Agile Service Manager, Continuous Delivery Ecosystem Foundation, DevOps Foundation, Continuous Testing Foundation, DevSecOps Foundation
Continuous Integration Tools	Tools that provide an immediate feedback loop by regularly merging, building and testing code. Example tools include: Atlassian Bamboo, Jenkins, Microsoft VSTS/Azure DevOps, TeamCity.	DevOps Foundation, DevOps Leader
Continuous Monitoring (CM)	This is a class of terms relevant to logging, notifications, alerts, displays and analysis of test results information.	Continuous Delivery Ecosystem Foundation, Continuous Testing Foundation
Continuous Testing (CT)	This is a class of terms relevant to testing and verification of an EUT in a DevOps environment.	DevOps Foundation, Continuous Delivery Ecosystem Foundation, Continuous Testing Foundation
Conversation Café	Conversation Cafés are open, hosted conversations in cafés as well as conferences and classrooms—anywhere people gather to make sense of our world.	DevOps Leader
Conway's Law	Organizations which design systems are constrained to produce designs which are copies of the communication structures of these organizations.	Continuous Delivery Ecosystem Foundation, DevOps Leader
Cooperation vs. Competition	The key cultural value shift toward being highly collaborative and cooperative, and away from internal competitiveness and divisiveness.	DevSecOps Foundation
COTS	Commercial-off-the-shelf solution	Continuous Delivery Ecosystem Foundation, Continuous Testing Foundation
Critical Success Factor (CSF)	Something that must happen for an IT service, process, plan, project or other activity to succeed.	Certified Agile Process Owner, Certified Agile Service Manager, DevOps Foundation, DevSecOps Foundation

CSI Register	Vehicle for recording and managing improvement opportunities throughout their lifecycle (Continual Service Improvement).	Certified Agile Service Manager
Cultural Iceberg	A metaphor that visualizes the difference between observable (above the water) and non-observable (below the waterline) elements of culture.	DevOps Leader
Culture (Organizational Culture)	The values and behaviors that contribute to the unique psychosocial environment of an organization.	Continuous Delivery Ecosystem Foundation, DevOps Foundation, DevSecOps Foundation
Cumulative Flow Diagram	A cumulative flow diagram is a tool used in agile software development and lean product development. It is an area graph that depicts the quantity of work in a given state, showing arrivals, time in queue, quantity in queue, and departure.	DevOps Leader
Current State Map	A form of value stream map that helps you identify how the current process works and where the disconnects are.	DevOps Leader
Customer Reliability Engineer (CRE)	CRE is what you get when you take the principles and lessons of SRE and apply them towards customers.	Sire Reliability Engineering
Cycle Time	A measure of the time from start of work to ready for delivery.	DevOps Foundation, DevOps Leader. DevSecOps Foundation
Daily Scrum	Daily timeboxed event of 15 minutes or less for the Team to replan the next day of work during a Sprint.	Certified Agile Service Manager, DevOps Foundation
Dashboard	Graphical display of summarized test results.	Continuous Delivery Ecosystem Foundation, Continuous Testing Foundation
Data Loss Protection (DLP)	Tools that prevent files and content from being removed from within a service environment or organization.	Site Reliability Engineering
Database Reliability Engineer (DBRE)	A person responsible for keeping database systems that support all user facing services in production running smoothly.	Site Reliability Engineering
Defect Density	The number of faults found in a unit E.g. # defects per KLOC, # defects per change.	Continuous Delivery Ecosystem Foundation, Continuous Testing Foundation

Definition of Done	A shared understanding of expectations that the Increment must live up to in order to be releasable into production. (Scrum.org)	Certified Agile Process Owner, Certified Agile Service Manager, DevOps Foundation, DevOps Leader
Delivery Cadence	The frequency of deliveries. E.g. # deliveries per day, per week, etc.	Continuous Delivery Ecosystem Foundation, Continuous Testing Foundation
Delivery Package	Set of release items (files, images, etc.) that are packaged for deployment.	Continuous Delivery Ecosystem Foundation, Continuous Testing Foundation
Deming Cycle	A four-stage cycle for process management, attributed to W. Edwards Deming. Also called Plan-Do-Check-Act (PDCA).	DevOps Foundation, DevSecOps Foundation
Dependency Firewall	Many projects depend on packages that may come from unknown or unverified providers, introducing potential security vulnerabilities. There are tools to scan dependencies but that is after they are downloaded. These tools prevent those vulnerabilities from being downloaded to begin with.	Site Reliability Engineering
Dependency Proxy	For many organizations, it is desirable to have a local proxy for frequently used upstream images/packages. In the case of CI/CD, the proxy is responsible for receiving a request and returning the upstream image from a registry, acting as a pull-through cache.	Site Reliability Engineering
Dependency Scanning	Used to automatically find security vulnerabilities in your dependencies while you are developing and testing your applications. Synopisis, Gemnasium, Retire.js and bundler-audit are popular tools in this area.	Site Reliability Engineering
Deployment	The installation of a specified version of software to a given environment (e.g., promoting a new build into production).	DevOps Foundation, DevSecOps Foundation
Design for Testability	An EUT is designed with features which enable it to be tested.	Continuous Delivery Ecosystem Foundation, Continuous Testing Foundation
Design Principles	Principles for designing, organizing, and managing a DevOps delivery operating model.	DevOps Leader

Dev	Individuals involved in software development activities such as application and software engineers.	DevOps Foundation, DevSecOps Foundation
Developer (Dev)	Individual who has responsibility to develop changes for an EUT. Alternate: Individuals involved in software development activities such as application and software engineers.	Continuous Delivery Ecosystem Foundation, Continuous Testing Foundation
Development Test	Ensuring that the developer's test environment is a good representation of the production test environment.	Continuous Delivery Ecosystem Foundation, Continuous Testing Foundation
Device Under Test (DUT)	The EUT is a device. E.g. Router or switch is being tested.	Continuous Delivery Ecosystem Foundation, Continuous Testing Foundation
DevOps	A cultural and professional movement that stresses communication, collaboration and integration between software developers and IT operations professionals while automating the process of software delivery and infrastructure changes. It aims at establishing a culture and environment where building, testing, and releasing software, can happen rapidly, frequently, and more reliably." (Source: Wikipedia)	Certified Agile Service Manager, DevOps Foundation, DevSecOps Foundation
DevOps Coach	Help teams master Agile development and DevOps practices; enables productive ways of working and collaboration.	DevOps Leader
DevOps Infrastructure	The entire set of tools and facilities that make up the DevOps system. Includes CI, CT, CM and CD tools.	Continuous Delivery Ecosystem Foundation, Continuous Testing Foundation
DevOps Kaizen	Kaizen is a Japanese word that closely translates to "change for better," the idea of continuous improvement—large or small—involving all employees and crossing organisational boundaries. Damon Edwards' DevOps Kaizen shows how making small, incremental improvements (little J's) has an improved impact on productivity long term.	DevOps Leader
DevOps Pipeline	The entire set of interconnected processes that make up a DevOps Infrastructure.	Continuous Delivery Ecosystem Foundation, Continuous Testing Foundation

DevOps Score	A metric showing DevOps adoption across an organization and the corresponding impact on delivery velocity.	Site Reliability Engineering
DevOps Toolchain	The tools needed to support a DevOps continuous development and delivery cycle from idea to value realisation.	Continuous Delivery Ecosystem Foundation, DevOps Foundation, DevSecOps Foundation, Continuous Testing Foundation
DevSecOps	A mindset that "everyone is responsible for security" with the goal of safely distributing security decisions at speed and scale to those who hold the highest level of context without sacrificing the safety required.	Continuous Delivery Ecosystem Foundation, DevOps Foundation, DevSecOps Foundation
Distributed Version Control System (DVCS)	The software revisions are stored in a distributed revision control system (DRCS), also known as a distributed version control system (DVCS).	Continuous Delivery Ecosystem Foundation
DMZ (De-Militarized Zone)	A DMZ in network security parlance is a network zone in between the public internet and internal protected resources. Any application, server, or service (including APIs) that need to be exposed externally are typically placed in a DMZ. It is not uncommon to have multiple DMZs in parallel.	DevSecOps Foundation
Dojo	A place where DevOps team members go for hands-on training. The team is self-organizing, which means that members will first work together to figure out what skills they will need to accomplish a desired goal and then plan together how to build those skills.	DevOps Foundation
Dynamic Analysis	Dynamic analysis is the testing of an application by executing data in real-time with the objective of detecting defects while it is in operation, rather than by repeatedly examining the code offline.	Continuous Delivery Ecosystem Foundation, Continuous Testing Foundation
Dynamic Application Security Testing (DAST)	A type of testing that runs against built code to test exposed interfaces.	DevSecOps Foundation
EggPlant	Automated function and regression testing of enterprise applications. Licensed by Test Plant.	Continuous Testing Foundation

Elastic Infrastructure	Elasticity is a term typically used in cloud computing, to describe the ability of an IT infrastructure to quickly expand or cut back capacity and services without hindering or jeopardizing the infrastructure's stability, performance, security, governance or compliance protocols.	Continuous Delivery Ecosystem Foundation
Elevator Pitch	A short summary used to quickly and simply define a process, product, service, organization, or event and its value proposition.	Certified Agile Process Owner
Empirical Process Control	Process control model in which decisions are made based on observation and experimentation (rather than on detailed upfront planning) and decisions are based on what is known.	Certified Agile Process Owner
eNPS	Employee Net Promoter Score (eNPS) is a way for organizations to measure employee loyalty. The Net Promoter Score, originally a customer service tool, was later used internally on employees instead of customers.	DevOps Foundation, DevOps Leader
Entity Under Test (EUT)	This is a class of terms which refers to names of types of entities that are being tested. These terms are often abbreviated to the form xUT where "x" represents a type of entity under test.	Continuous Delivery Ecosystem Foundation, Continuous Testing Foundation
Epic	A big chunk of work, made up of a number of user stories, with a common objective.	Certified Agile Process Owner
Erickson (Stages of Psychosocial Development)	Erik Erikson (1950, 1963) proposed a psychoanalytic theory of psychosocial development comprising eight stages from infancy to adulthood. During each stage, the person experiences a psychosocial crisis which could have a positive or negative outcome for personality development.	DevSecOps Foundation
Error Budget	The error budget provides a clear, objective metric that determines how unreliable a service is allowed to be within a specific time period.	Site Reliability Engineering
Error Budget Policies	An error budget policy enumerates the activity a team takes when they've exhausted their error budget for a particular service in a particular time period.	Site Reliability Engineering

Error Tracking	Tools to easily discover and show the errors that application may be generating, along with the associated data.	Site Reliability Engineering
External Automation	Scripts and automation outside of a service that is intended to reduce toil.	Site Reliability Engineering
Fail Early	A DevOps tenet referring to the preference to find critical problems as early as possible in a development and delivery pipeline.	Continuous Delivery Ecosystem Foundation, Continuous Testing Foundation
Fail Often	A DevOps tenet which emphasizes a preference to find critical problems as fast as possible and therefore frequently.	Continuous Delivery Ecosystem Foundation, Continuous Testing Foundation
Failure Rate	Fail verdicts per unit of time.	DevOps Foundation, Continuous Delivery Ecosystem Foundation, Continuous Testing Foundation
False Negative	A test incorrectly reports a verdict of "fail" when the EUT actually passed the purpose of the test.	Continuous Delivery Ecosystem Foundation, Continuous Testing Foundation
False Positive	A test incorrectly reports a verdict of "pass" when the EUT actually failed the purpose of the test.	Continuous Delivery Ecosystem Foundation, Continuous Testing Foundation
Feature Toggle	The practice of using software switches to hide or activate features. This enables continuous integration and testing a feature with selected stakeholders.	DevOps Foundation, Continuous Delivery Ecosystem Foundation, Continuous Testing Foundation
Federated Identity	A central identity used for access to a wide range of applications, systems, and services, but with a particular skew toward web-based applications. Also, often referenced as Identity-as-a-Service (IDaas). Any identity that can be reused across multiple sites, particularly via SAML or OAuth authentication mechanisms.	DevSecOps Foundation
Fire Drills	A planned failure testing process focussed on the operation of live services including service failure testing as well as communication, documentation, and other human factor testing.	Site Reliability Engineering

Flow	How people, products or information move through a process. Flow is the first way of The Three Ways.	DevOps Foundation, DevOps Leader, DevSecOps Foundation
Flow of Value	A form of map that shows the end-to-end value stream. This view is usually not available within the enterprise.	DevOps Leader
Framework	Backbone for plugging in tools. Launches automated tasks, collects results from automated tasks.	Continuous Delivery Ecosystem Foundation, Continuous Testing Foundation
Freedom and Responsibility	A core cultural value that with the freedom of self-management (such as afforded by DevOps) comes the responsibility to be diligent, to follow the advice process and to take ownership of both successes and failures.	DevSecOps Foundation
Frequency	How often an application is released.	DevOps Leader
Functional Testing	Tests to determine if the functional operation of the service is as expected.	Site Reliability Engineering
Future State Map	A form of value stream map that helps you develop and communicate what the target end state should look like and how to tackle the necessary changes.	DevOps Leader
Fuzzing	Fuzzing or fuzz testing is an automated software testing practice that inputs invalid, unexpected, or random data into applications.	DevSecOps Foundation
Gated Commits	Define and obtain consensus for criterion of changes promoted between all CD pipeline stages such as: Dev to CI stage / CI to packaging / delivery stage / Delivery to Deployment/Production stage.	Continuous Delivery Ecosystem Foundation
Generative (DevOps) Culture	In a generative organization alignment takes place through identification with the mission. The individual "buys into" what he or she is supposed to do and its effect on the outcome. Generative organizations tend to be proactive in getting the information to the right people by any means. necessary. (Westrum)	DevOps Leader
Generativity	A cultural view wherein long-term outcomes are of primary focus, which in turn drives investments and cooperation that enable an organization to achieve those outcomes.	DevSecOps Foundation

Glass-Box	Same as Clear-Box Testing and White-Box Testing.	Continuous Delivery Ecosystem Foundation, Continuous Testing Foundation
Global Process Owner	Process Owner who oversees a single, global process. A Global Process Owner (who may reside in a SMO) may oversee one or more Regional Process Managers.	Certified Agile Process Owner
Goal-seeking tests	The purpose of the test is to determine an EUT's performance boundaries, using incrementally stresses until the EUT reaches a peak performance. E.g. Determine the maximum throughput that can be handled without errors.	Continuous Delivery Ecosystem Foundation, Continuous Testing Foundation
Golden Circle	A model by Simon Sinek that emphasizes an understanding of the business' "why" before focusing on the "what" and "how".	DevOps Foundation
Golden Image	A template for a virtual machine (VM), virtual desktop, server or hard disk drive. (TechTarget)	DevSecOps Foundation
Goleman's Six Styles of Leadership	Daniel Goleman (2002) created the Six Leadership Styles and found, in his research, that leaders used one of these styles at any one time.	DevOps Leader
Governance, Risk Management and Compliance (GRC)	A software platform intended for concentrating governance, compliance and risk management data, including policies, compliance requirements, vulnerability data, and sometimes asset inventory, business continuity plans, etc. In essence, a specialized document and data repository for security governance. Or a team of people who specialize in IT/security governance, risk management and compliance activities. Most often non-technical business analyst resources.	DevSecOps Foundation
Gray-Box	Test cases use a limited knowledge of the internal design structure of the EUT.	Continuous Delivery Ecosystem Foundation, Continuous Testing Foundation
GUI testing	The purpose of the test is to determine if the graphical user interface operates as expected.	Continuous Delivery Ecosystem Foundation, Continuous Testing Foundation

Guilds	A "community of interest" group that welcomes anyone and usually cuts across an entire organization. Similar to a Community of Practice.	DevOps Foundation, DevOps Leader
Hand Offs	The procedure for transferring the responsibility of a particular task from one individual or team to another.	DevOps Foundation, DevOps Leader
Hardening	Securing a server or infrastructure environment by removing or disabling unnecessary software, updating to known good versions of the operating system, restricting network-level access to only that which is needed, configuring logging in order to capture alerts, configuring appropriate access management and installing appropriate security tools.	DevSecOps Foundation
Helm Chart Registry	Helm charts are what describe related Kubernetes resources. Artifactory and Codefresh support a registry for maintaining master records of Helm Charts.	Site Reliability Engineering
Heritage Reliability Engineer (HRE)	Applying the principles and practices of SRE to legacy applications and environments.	Site Reliability Engineering
High-Trust Culture	Organizations with a high-trust culture encourage good information flow, cross-functional collaboration, shared responsibilities, learning from failures and new ideas.	DevOps Foundation
Horizontal Scaling	Computing resources are scaled wider to increase the volume of processing. E.g. Add more computers and run more tasks in parallel.	Continuous Delivery Ecosystem Foundation, Continuous Testing Foundation
Idempotent	CM tools (e.g., Puppet, Chef, Ansible, and Salt) claim that they are 'idempotent' by allowing the desired state of a server to be defined as code or declarations and automate steps necessary to consistently achieve the defined state time-after-time.	Continuous Delivery Ecosystem Foundation
Identity	The unique name of a person, device, or the combination of both that is recognized by a digital system. Also referred to as an "account" or "user."	DevSecOps Foundation
Identity and Access Management (IAM)	Policies, procedures and tools for ensuring the right people have the right access to technology resources.	DevSecOps Foundation

Identity as a Service (IDAAS)	Identity and access management services that are offered through the cloud or on a subscription basis.	DevSecOps Foundation
Image-based test selection method	Build images are pre-assigned test cases. Tests cases are selected for a build by matching the image changes resulting from a build.	Continuous Delivery Ecosystem Foundation, Continuous Testing Foundation
Immersive learning	A learning approach that guides teams with coaching and practice to help them learn to work in a new way.	DevOps Leader
Immutable	An immutable object is an object whose state cannot be modified after it is created. The antonym is a mutable object, which can be modified after it is created.	Continuous Delivery Ecosystem Foundation
Immutable Infrastructures	Instead of instantiating an instance (server, container, etc.), with error-prone, time-consuming patches and upgrades (i.e. mutations), replace it with another instance to introduce changes or ensure proper behavior.	Continuous Delivery Ecosystem Foundation, Site Reliability Engineering
Impediment	Anything that prevents a team member from performing work as efficiently as possible.	Certified Agile Process Owner, Certified Agile Service Manager, DevOps Foundation
Impediment (Scrum)	Anything that prevents a team member from performing work as efficiently as possible.	Agile Service Management, DevOps Foundation
Implementation Under Test	The EUT is a software implementation. E.g. Embedded program is being tested.	Continuous Delivery Ecosystem Foundation, Continuous Testing Foundation
Improvement Kata	A structured way to create a culture of continuous learning and improvement. (In Japanese business, Kata is the idea of doing things the "correct" way. An organization's culture can be characterized as its Kata through its consistent role modeling, teaching and coaching.)	DevOps Foundation
Incentive model	A system designed to motivate people to complete tasks toward achieving objectives. The system may employ either positive or negative consequences for motivation.	DevSecOps Foundation
Incident	Any unplanned interruption to an IT service or reduction in the quality of an IT service. Includes events that disrupt or could disrupt the service. (ITIL definition)	DevOps Foundation, DevSecOps Foundation

Incident Management	Process that restores normal service operation as quickly as possible to minimize business impact and ensure that agreed levels of service quality are maintained. (ITIL definition). Involves capturing the who, what, when of service incidents and the onward use of this data in ensuring service level objectives are being met.	DevOps Foundation, DevSecOps Foundation, Site Reliability Engineering
Incident Response	An organized approach to addressing and managing the aftermath of a security breach or attack (also known as an incident). The goal is to handle the situation in a way that limits damage and reduces recovery time and costs.	DevSecOps Foundation, Site Reliability Engineering
Increment	Potentially shippable completed work that is the outcome of a Sprint.	Certified Agile Service Manager, DevOps Foundation
Incremental Rollout	Incremental rollout means deploying many small, gradual changes to a service instead of a few large changes. Users are incrementally moved across to the new version of the service until eventually all users are moved across. Sometimes referred to by colored environments e.g. Blue/green deployment.	Site Reliability Engineering
Infrastructure	All of the hardware, software, networks, facilities, etc., required to develop, test, deliver, monitor and control or support IT services. The term IT infrastructure includes all of the information technology but not the associated people, processes and documentation. (ITIL definition)	DevOps Foundation, DevSecOps Foundation
Infrastructure as Code	The practice of using code (scripts) to configure and manage infrastructure.	DevOps Foundation, DevSecOps Foundation
Infrastructure Test	The purpose of the test is to verify the framework for EUT operating. E.g. verify specific operating system utilities function as expected in the target environment.	Continuous Delivery Ecosystem Foundation, Continuous Testing Foundation
Infrastructure-as-a-Service (IaaS)	On-demand access to a shared pool of configurable computing resources.	Continuous Delivery Ecosystem Foundation, Continuous Testing Foundation

Integrated development environment (IDE)	An integrated development environment (IDE) is a software suite that consolidates the basic tools developers need to write and test software. Typically, an IDE contains a code editor, a compiler or interpreter and a debugger that the developer accesses through a single graphical user interface (GUI). An IDE may be a standalone application, or it may be included as part of one or more existing and compatible applications. (TechTarget)	DevSecOps Foundation
Integrated development environment (IDE) 'lint' checks	Linting is the process of running a program that will analyze code for potential errors (e.g., formatting discrepancies, non-adherence to coding standards and conventions, logical errors).	DevSecOps Foundation
Internet of Things	A network of physical devices that connect to the internet and potentially to each other through web-based wireless services.	DevOps Foundation, DevSecOps Foundation
Internal Automation	Scripts and automation delivered as part of the service that is intended to reduce toil.	Site Reliability Engineering
INVEST	A mnemonic was created by Bill Wake as a reminder of the characteristics of a quality user story.	Certified Agile Service Manager
ISO 31000	A family of standards that provide principles and generic guidelines on risk management.	DevSecOps Foundation
ISO/IEC 20000	International standard for IT service management. ISO/IEC 20000 is used to audit and certify service management capabilities.	DevOps Foundation
Issue Management	A process for capturing, tracking, and resolving bugs and issues throughout the software development lifecycle.	DevSecOps Foundation
IT Infrastructure Library (ITIL)	Set of best practice publications for IT service management. Published in a series of five core books representing the stages of the IT service lifecycle which are: Service Strategy, Service Design, Service Transition, Service Operation and Continual Service Improvement.	Certified Agile Process Owner
IT Service	A service provided to a customer from an IT organization.	DevOps Foundation
IT Service Management (ITSM)	Implementation and management of quality IT services that meet the needs of the business. (ITIL definition)	Certified Agile Process Owner, Site Reliability Engineering

iTest	Tool licensed by Spirent Communications for creating automated test cases.	Continuous Testing Foundation
ITIL	Set of best practice publications for IT service management. Published in a series of five core books representing the stages of the IT service lifecycle which are: Service Strategy, Service Design, Service Transition, Service Operation and Continual Service Improvement.	Certified Agile Service Manager, DevOps Foundation, Site Reliability Engineering
Jenkins	Jenkins is a freeware tool. It is the most popular master automation framework tool, especially for continuous integration task automation. Jenkins task automation centers around timed processes. Many test tools and other tools offer plugins to simplify integration with Jenkins.	Continuous Delivery Ecosystem Foundation, Continuous Testing Foundation
Kaizen	The practice of continuous improvement.	DevOps Foundation
Kanban	Method of work that pulls the flow of work through a process at a manageable pace.	Certified Agile Service Manager, DevOps Foundation
Kanban Board	Tool that helps teams organize, visualize and manage work.	DevOps Foundation
Karpman Drama Triangle	The drama triangle is a social model of human interaction. The triangle maps a type of destructive interaction that can occur between people in conflict.	DevOps Leader
Key Metrics	Something that is measured and reported upon to help manage a process, IT service or activity.	DevOps Foundation, DevOps Leader
Key Performance Indicator	Key metric used to measure the achievement of critical success factors. KPIs underpin critical success factors and are measured as a percentage.	Certified Agile Process Owner, Certified Agile Service Manager
Key Performance Indicator (KPI)	Key metric used to measure the achievement of critical success factors. KPIs underpin critical success factors and are measured as a percentage. (ITIL definition)	Certified Agile Service Manager, DevOps Foundation
Keywords-Based	Test cases are created using pre-defined names that reference programs useful for testing.	Continuous Delivery Ecosystem Foundation, Continuous Testing Foundation
Knowledge Management	Process that ensures the right information is delivered to the right place or person at the right time to enable an informed decision.	DevOps Foundation, DevSecOps Foundation

Known Error	Problem with a documented root cause and a workaround. (ITIL definition)	DevOps Foundation, DevSecOps Foundation
Kolb's Learning Styles	David Kolb published his learning styles model in 1984; his experiential learning theory works on two levels: a four stage cycle of learning and four separate learning styles.	DevOps Leader
Kotter's Dual Operating System	John Kotter describes the need for a dual operating system that combines the entrepreneurial capability of a network with the organisational efficiency of traditional hierarchy.	DevOps Leader
Kubernetes	Kubernetes is an open-source container-orchestration system for automating application deployment, scaling, and management. It was originally designed by Google, and is now maintained by the Cloud Native Computing Foundation.	Site Reliability Engineering
Kubler-Ross Change Curve	Describes and predicts the stages of personal and organizational reaction to major changes.	DevOps Foundation
Lab-as-a-Service (LaaS)	Category of cloud computing services that provides a laboratory allowing customers to test applications without the complexity of building and maintaining the lab infrastructure.	Continuous Delivery Ecosystem Foundation, Continuous Testing Foundation
Laloux (Culture Models)	Frederic Laloux created a model for understanding organizational culture.	DevSecOps Foundation
Latency	Latency is the delay incurred in communicating a message, the time a message spends “on the wire” between the initial request being received e.g. by a server and the response being recieved e.g. by a client.	Site Reliability Engineering
Laws of Systems Thinking	In his book 'The Fifth Discipline', Peter Senge outlines eleven laws will help the understanding of business systems and to identify behaviors for addressing complex business problems.	DevOps Leader
Lean	Production philosophy that focuses on reducing waste and improving the flow of processes to improve overall customer value.	DevOps Leader
Lean (adjective)	Spare, economical. Lacking richness or abundance.	DevOps Foundation, DevSecOps Foundation

Lean (production)	Production philosophy that focuses on reducing waste and improving the flow of processes to improve overall customer value.	DevOps Foundation, DevSecOps Foundation
Lean Canvas	Lean Canvas is a 1-page business plan template.	DevOps Leader
Lean Enterprise	Organization that strategically applies the key ideas behind lean production across the enterprise.	DevOps Foundation, DevSecOps Foundation
Lean IT	Applying the key ideas behind lean production to the development and management of IT products and services.	DevOps Foundation, DevSecOps Foundation
Lean Manufacturing	Lean production philosophy derived mostly from the Toyota Production System.	DevOps Foundation, DevSecOps Foundation
Lean Product Development	Lean Product Development, or LPD, utilizes Lean principles to meet the challenges of Product Development.	DevOps Leader
Lean Six Sigma	Management approach that combines the concepts of Lean Manufacturing and Six Sigma by removing 'waste' and reducing 'defects'.	Certified Agile Process Owner
Lean Startup	A system for developing a business or product in the most efficient way possible to reduce the risk of failure.	DevOps Leader
Lean Thinking	The goal of lean thinking is to create more value for customers with fewer resources and less waste. Waste is considered any activity that does not add value to the process.	Certified Agile Service Manager
License Scanning	Tools, such as Blackduck and Synopsis, that check that licenses of your dependencies are compatible with your application, and approve or blacklist them.	Site Reliability Engineering
Little's Law	A theorem by John Little which states that the long-term average number L of customers in a stationary system is equal to the long-term average effective arrival rate λ multiplied by the average time W that a customer spends in the system.	DevOps Leader
LoadRunner	Tool used to test applications, measuring system behavior and performance under load. Licensed by HP.	Continuous Delivery Ecosystem Foundation, Continuous Testing Foundation

Log	Serialized report of details such as test activities and EUT console logs.	Continuous Delivery Ecosystem Foundation, Continuous Testing Foundation
Log Management	The collective processes and policies used to administer and facilitate the generation, transmission, analysis, storage, archiving and ultimate disposal of the large volumes of log data created within an information system.	DevSecOps Foundation
Logging	The capture, aggregation and storage of all logs associated with system performance including, but not limited to, process calls, events, user data, responses, error and status codes. Logstash and Nagios are popular examples.	Site Reliability Engineering
Logic Bomb (Slag Code)	A string of malicious code used to cause harm to a system when the programmed conditions are met.	DevSecOps Foundation
Longevity Test	The purpose of the test is to determine if a complete system performs as expected over an extended period of time	Continuous Delivery Ecosystem Foundation, Continuous Testing Foundation
Machine Learning	Data analysis that uses algorithms that learn from data.	DevOps Foundation
Malware	A program designed to gain access to computer systems, normally for the benefit of some third party, without the user's permission	DevSecOps Foundation
Many-factor Authentication	The practice of using at least 2 factors for authentication. The two factors can be of the same class.	DevSecOps Foundation
Mean Time Between Deploys	Used to measure deployment frequency.	DevOps Foundation, DevSecOps Foundation
Mean Time Between Failures (MTBF)	Average time that a CI or IT service can perform its agreed function without interruption. Often used to measure reliability. Measured from when the CI or service starts working, until the time it fails (uptime). (ITIL definition)	DevOps Foundation, DevSecOps Foundation
Mean Time to Detect Incidents (MTTD)	Average time required to detect a failed component or device.	Continuous Delivery Ecosystem Foundation, DevOps Foundation, DevSecOps Foundation, Site Reliability Engineering

Mean Time to Discovery	How long a vulnerability or software bug/defect exists before it's identified.	DevSecOps Foundation
Mean Time to Patch	How long it takes to apply patches to environments once a vulnerability has been identified.	DevSecOps Foundation
Mean Time to Repair/Recover (MTTR)	Average time required to repair/recover a failed component or device. MTTR does not include the time required to recover or restore service.	DevOps Foundation, DevSecOps Foundation Site Reliability Engineering
Mean Time to Restore Service (MTRS)	Used to measure time from when the CI or IT service fails until it is fully restored and delivering its normal functionality (downtime). Often used to measure maintainability. (ITIL definition).	DevOps Foundation, DevSecOps Foundation, Site Reliability Engineering
Mental Models	A mental model is an explanation of someone's thought process about how something works in the real world.	DevOps Leader
Merge	Action of integrating a software changes together into a software version management system.	Continuous Delivery Ecosystem Foundation, Continuous Testing Foundation
Metric	Something that is measured and reported upon to help manage a process, IT service or activity.	DevOps Foundation, DevSecOps Foundation
Metrics	This is a class of terms relevant to measurements used to monitor the health of a product or infrastructure.	Continuous Delivery Ecosystem Foundation, Continuous Testing Foundation
Microservices	A software architecture that is composed of smaller modules that interact through APIs and can be updated without affecting the entire system.	DevOps Foundation
Mindset	A person's usual attitude or mental state is their mindset.	DevOps Leader
Minimum Critical Activities	Activities that must be performed to provide evidence of compliance with a given process.	Certified Agile Process Owner
Minimum Viable Product	Most minimal version of a product that can be released and still provide enough value that people are willing to use it.	Certified Agile Service Manager, DevOps Foundation, DevOps Leader

Mock Object	Mock is a method/object that simulates the behavior of a real method/object in controlled ways. Mock objects are used in unit testing. Often a method under a test calls other external services or methods within it. These are called dependencies.	Continuous Delivery Ecosystem Foundation, Continuous Testing Foundation
Model	Representation of a system, process, IT service, CI, etc. that is used to help understand or predict future behavior. In the context of processes, models represent pre-defined steps for handling specific types of transactions.	DevSecOps Foundation
Model-Based	Test cases are automatically derived from a model of the entity under test. Example tool: Tricentus	Continuous Delivery Ecosystem Foundation, Continuous Testing Foundation
Monitoring	The use of a hardware or software component to monitor the system resources and performance of a computer service.	Site Reliability Engineering
Monitoring Tools	Tools that allow IT organizations to identify specific issues of specific releases and to understand the impact on end-users.	DevOps Leader
Monolithic	A software system is called "monolithic" if it has a monolithic architecture, in which functionally distinguishable aspects (for example data input and output, data processing, error handling, and the user interface) are all interwoven, rather than containing architecturally separate components.	Continuous Delivery Ecosystem Foundation
Multi-factor Authentication	The practice of using 2 or more factors for authentication. Often used synonymously with 2-factor Authentication.	DevSecOps Foundation
Multi-cloud	Multi-cloud DevOps solutions provide on-demand multi-tenant access to development and test environments.	Continuous Delivery Ecosystem Foundation
Network Reliability Engineer (NRE)	Someone who applies a reliability engineering approach to measure and automate the reliability of networks.	Site Reliability Engineering
Neuroplasticity	Describes the ability of the brain to form and reorganize synaptic connections, especially in response to learning or experience or following injury.	DevOps Leader
Neuroscience	The study of the brain and nervous system.	DevOps Leader

Non-functional requirements	Requirements that specify criteria that can be used to judge the operation of a system, rather than specific behaviors or functions (e.g., availability, reliability, maintainability, supportability); qualities of a system.	DevOps Foundation
Non-functional tests	Defined as a type of service testing intending to check non-functional aspects such as performance, usability and reliability of a software service.	Site Reliability Engineering
Object Under Test (OUT)	The EUT is a software object or class of objects.	Continuous Delivery Ecosystem Foundation, Continuous Testing Foundation
Objective	An aim or goal of a process.	Certified Agile Process Owner
Observability	Observability is focused on externalizing as much data as you can about the whole service allowing us to infer what the current state of that service is.	Site Reliability Engineering
On-call	Being on-call means someone being available during a set period of time, and being ready to respond to production incidents during that time with appropriate urgency.	Site Reliability Engineering
Open Source	Software that is distributed with its source code so that end user organizations and vendors can modify it for their own purposes.	DevOps Foundation, DevSecOps Foundation
Operational Level Agreement	Agreement between an IT service provider and another part of the same organization. (ITIL definition)	Certified Agile Process Owner
Operations (Ops)	Individuals involved in the daily operational activities needed to deploy and manage systems and services such as quality assurance analysts, release managers, system and network administrators, information security officers, IT operations specialists and service desk analysts.	Continuous Delivery Ecosystem Foundation

Operations Management	Function that performs the daily activities needed to deliver and support IT services and the supporting IT infrastructure at the agreed levels. (ITIL)	DevSecOps Foundation
Ops	Individuals involved in the daily operational activities needed to deploy and manage systems and services such as quality assurance analysts, release managers, system and network administrators, information security officers, IT operations specialists and service desk analysts.	DevOps Foundation, DevSecOps Foundation
Orchestration	An approach to building automation that interfaces or "orchestrates" multiple tools together to form a toolchain.	DevOps Foundation, DevSecOps Foundation
Organization Culture	A system of shared values, assumptions, beliefs, and norms that unite the members of an organization.	DevOps Leader
Organization Model	For DevOps, an approach that models Spotify's Squad approach for organizing IT.	DevOps Leader
Organizational Change	Efforts to adapt the behavior of humans within an organization to meet new structures, processes or requirements.	DevOps Foundation, DevSecOps Foundation
OS Virtualization	A method for splitting a server into multiple partitions called "containers" or "virtual environments" in order to prevent applications from interfering with each other.	DevOps Foundation
Outcome	Intended or actual results.	DevOps Foundation, DevSecOps Foundation
Output	Deliverable produced by a process activity (e.g., information, plans, documents, records, reports and so forth).	Certified Agile Process Owner
Package Registry	A repository for software packages, artifacts and their corresponding metadata. Can store files produced by an organization itself or for third party binaries. Artifactory and Nexus are amongst the most popular.	Site Reliability Engineering
Pages	Something for creating supporting web pages automatically as part of a CI/CD pipeline.	Site Reliability Engineering
Patch	A software update designed to address (mitigate/remediate) a bug or weakness.	DevSecOps Foundation

Patch management	The process of identifying and implementing patches.	DevSecOps Foundation
Pathological Culture	Pathological cultures tend to view information as a personal resource, to be used in political power struggles (Westrum).	DevOps Leader, Site Reliability Engineering
Penetration Testing	An authorized simulated attack on a computer system that looks for security weaknesses, potentially gaining access to the system's features and data.	DevSecOps Foundation
People Changes	Focuses on changing attitudes, behaviors, skills, or performance of employees.	DevOps Leader
Performance Test	The purpose of the test is to determine an EUT meets its system performance criterion or to determine what a system's performance capabilities are.	Continuous Delivery Ecosystem Foundation, Continuous Testing Foundation
Plan	Formal, approved document that describes the capabilities and resources needed to achieve a result.	Certified Agile Process Owner
Plan-Do-Check-Act	A four-stage cycle for process management and improvement attributed to W. Edwards Deming. Sometimes called the Deming Cycle or PDCA.	Certified Agile Process Owner, Certified Agile Service Manager, DevOps Foundation, DevSecOps Foundation
Platform-as-a-Service (PaaS)	Category of cloud computing services that provides a platform allowing customers to develop, run, and manage applications without the complexity of building and maintaining the infrastructure.	Continuous Delivery Ecosystem Foundation, Continuous Testing Foundation
Plugin	A pre-programmed integration between an Orchestration tool and other tools. For example, many tools offer plugins to integrate with Jenkins.	Continuous Delivery Ecosystem Foundation, Continuous Testing Foundation
Policies	Formal documents that define boundaries in terms of what the organization may or may not do as part of its operations.	DevOps Foundation, DevSecOps Foundation
Policy	Formal document that describes the overall intentions and direction of a service provider, as expressed by senior management.	Certified Agile Process Owner

Policy as Code	The notion that security principles and concepts can be articulated in code (e.g., software, configuration management, automation) to a sufficient degree that the need for an extensive traditional policy framework is greatly reduced. Standards and guidelines should be implemented in code and configuration, automatically enforced and automatically reported-on in terms of compliance, variance or suspected violations.	DevSecOps Foundation
Post Implementation Review (PIR)	Review that takes place after a change or a project has been implemented that assesses whether the change was successful and opportunities for improvement.	Certified Agile Service Manager, DevOps Foundation
Potentially Shippable Product	Increment of work that is "done" and capable of being released if it makes sense to do so.	Certified Agile Service Manager, DevOps Foundation
Pre-Flight	This is a class of terms which refers names of activities and processes that are conducted on an EUT prior to integration into the trunk branch.	Continuous Delivery Ecosystem Foundation, Continuous Testing Foundation
Priority	The relative importance of an incident, problem or change; based on impact and urgency. (ITIL definition)	DevOps Foundation, DevSecOps Foundation
Privileged Access Management (PAM)	Technologies that help organizations provide secured privileged access to critical assets and meet compliance requirements by securing, managing and monitoring privileged accounts and access. (Gartner)	DevSecOps Foundation
Problem	The underlying cause of one or more incidents. (ITIL definition)	DevOps Foundation, DevSecOps Foundation
Procedure	Step-by-step instructions that describe how to perform the activities in a process.	Certified Agile Service Manager
Process	Structured set of activities designed to accomplish a specific objective. A process takes inputs and turns them into defined outputs. Related work activities that take specific inputs and produce specific outputs that are of value to a customer.	Certified Agile Service Manager, DevOps Foundation, DevSecOps Foundation
Process Backlog	Prioritized list of everything that needs to be designed or improved for a process including current and future requirements.	Certified Agile Service Manager

Process Changes	Focuses on changes to standard IT process, such as software development practices, ITIL processes, change management, approvals etc.	DevOps Leader
Process Customer	Recipient of a process' output.	Certified Agile Service Manager
Process Improvement Team	Team of individuals that designs or redesigns a process and determines how best to implement the new process across the organization.	Certified Agile Process Owner
Process Manager	Individual responsible for operational (day-to-day) management of a process.	Certified Agile Process Owner
Process Owner	Role accountable for the overall quality of a process. May be assigned to the same person who carries out the Process Manager role, but the two roles may be separate in larger organizations. (ITIL definition)	DevOps Foundation, DevSecOps Foundation, Certified Agile Service Manager
Process Owner	Person accountable for the overall quality of a process and the owner of the Process Backlog.	Certified Agile Service Manager
Process Planning Meeting	A high-level event to define the goals, objectives, inputs, outcomes, activities, stakeholders, tools and other aspects of a process. This meeting is not timeboxed.	Certified Agile Service Manager
Process Supplier	Creator of process input.	Certified Agile Service Manager
Processing Time	The period during which one or more inputs are transformed into a finished product by a manufacturing or development procedure. (Business Dictionary)	DevOps Leader
Product Backlog	Prioritized list of functional and non-functional requirements for a system usually expressed as user stories.	Certified Agile Process Owner, Certified Agile Service Manager, DevOps Foundation
Product Backlog Refinement	Ongoing process of adding detail, estimates and order to backlog items. Sometimes referred to as Product Backlog grooming.	Certified Agile Service Manager
Product Owner	An individual responsible for maximizing the value of a product and for managing the product backlog. Prioritizes, grooms, and owns the backlog. Gives the squad purpose.	Certified Agile Process Owner, Certified Agile Service Manager, DevOps Foundation, DevOps Leader

Programming-Based	Test cases are created by writing code in a programming language. E.g. JavaScript, Python, TCL, Ruby	Continuous Delivery Ecosystem Foundation, Continuous Testing Foundation
Project	Temporary endeavor undertaken to create a unique product, service or result.	Certified Agile Process Owner
Provision Platforms	Tools that provide platforms for provisioning infrastructure (e.g., Puppet, Chef, Salt).	DevOps Leader
Psychological Safety	Psychological safety is a shared belief that the team is safe for interpersonal risk taking.	DevOps Leader
QTP	Quick Test Professional is a functional and regression test automation tool for software applications. Licensed by HP.	Continuous Testing Foundation
Quality Management	Tools that handle test case planning, test execution, defect tracking (often into backlogs), severity and priority analysis. CA's Agile Central	Site Reliability Engineering
RACI Matrix	Maps roles and responsibilities to the activities of a process or project.	Certified Agile Process Owner
Ranorex	GUI test automation framework for testing of desktop, web-based and mobile applications. Licensed by Ranorex.	Continuous Testing Foundation
Ransomware	Encrypts the files on a user's device or a network's storage devices. To restore access to the encrypted files, the user must pay a "ransom" to the cybercriminals, typically through a tough-to-trace electronic payment method such as Bitcoin.	DevSecOps Foundation
Regression testing	The purpose of the test is to determine if a new version of an EUT has broken somethings that worked previously.	Continuous Delivery Ecosystem Foundation, Continuous Testing Foundation
Regulatory compliance testing	The purpose of the test is to determine if an EUT conforms to specific regulatory requirements. E.g. verify an EUT satisfies government regulations for consumer credit card processing.	Continuous Delivery Ecosystem Foundation, Continuous Testing Foundation
Release	Software that is built, tested and deployed into the production environment.	Continuous Delivery Ecosystem Foundation, DevOps Foundation, DevSecOps Foundation

Release Acceptance Criteria	Measurable attributes for a release package which determine whether a release candidate is acceptable for deployment to customers.	Continuous Delivery Ecosystem Foundation, Continuous Testing Foundation
Release Candidate	A release package that has been prepared for deployment, may or may not have passed the Release.	Continuous Delivery Ecosystem Foundation, Continuous Testing Foundation
Release Governance	Release Governance is all about the controls and automation (security, compliance, or otherwise) that ensure your releases are managed in an auditable and trackable way, in order to meet the need of the business to understand what is changing.	Site Reliability Engineering
Release Management	Process that manages releases and underpins Continuous Delivery and the Deployment Pipeline.	DevOps Foundation, DevSecOps Foundation
Release Orchestration	Typically a deployment pipeline, used to detect any changes that will lead to problems in production. Orchestrating other tools will identify performance, security, or usability issues. Tools like Jenkins and Gitlab CI can "orchestrate" releases.	Site Reliability Engineering
Release Planning Meeting	Time-boxed event that establishes the goals, risks, features, functionality, delivery date and cost of a release. It also includes prioritizing the Product Backlog.	Certified Agile Process Owner, Certified Agile Service Manager
Relevance	A Continuous Testing tenet which emphasizes a preference to focus on the most important tests and test results	Continuous Delivery Ecosystem Foundation, Continuous Testing Foundation
Reliability	Measure of how long a service, component or CI can perform its agreed function without interruption. Usually measured as MTBF or MTBSI. (ITIL definition)	DevOps Foundation, DevSecOps Foundation, Site Reliability Engineering
Reliability Test	The purpose of the test is to determine if a complete system performs as expected under stressful and loaded conditions over an extended period of time.	Continuous Delivery Ecosystem Foundation, Continuous Testing Foundation
Remediation	Action to resolve a problem found during DevOps processes. E.g. Roll-back changes for an EUT change that resulted in a CT a test case fail verdict.	Continuous Delivery Ecosystem Foundation, Continuous Testing Foundation
Remediation Plan	Plan that determines the actions to take after a failed change or release. (ITIL definition)	DevOps Foundation, DevSecOps Foundation

Request for Change (RFC)	Formal proposal to make a change. The term RFC is often misused to mean a change record, or the change itself. (ITIL definition)	DevOps Foundation
Requirements Management	Tools than handle requirements definition, traceability, hierarchies & dependency. Often also handles code requirements and test cases for requirements.	Site Reliability Engineering
Resilience	Building an environment or organization that is tolerant to change and incidents.	DevSecOps Foundation, Site Reliability Engineering
Response Time	Response time is the total time it takes from when a user makes a request until they receive a response.	Site Reliability Engineering
REST	Representation State Transfer. Software architecture style of the world-wide web.	Continuous Delivery Ecosystem Foundation, Continuous Testing Foundation
Restful API	Representational state transfer (REST) or RESTful services on a network, such as HTTP, provide scalable interoperability for requesting systems to quickly and reliably access and manipulate textual representations (XML, HTML, JSON) of resources using stateless operations (GET, POST, PUT, DELETE, etc.).	Continuous Delivery Ecosystem Foundation
RESTful interface testing	The purpose of the test is to determine if an API satisfies its design criterion and the expectations of the REST architecture.	Continuous Delivery Ecosystem Foundation, Continuous Testing Foundation
Return on Investment (ROI)	Difference between the benefit achieved and the cost to achieve that benefit, expressed as a percentage.	DevOps Foundation, DevSecOps Foundation
Review Apps	Allow code to be committed and launched in real time – environments are spun up to allow developers to review their application.	Site Reliability Engineering

Rework	The time and effort required to correct defects (waste).	DevOps Leader
Risk	Possible event that could cause harm or loss or affect an organization's ability to achieve its objectives. The management of risk consists of three activities: identifying risks, analyzing risks and managing risks. The probably frequency and probable magnitude of future loss. Pertains to a possible event that could cause harm or loss or affect an organization's ability to execute or achieve its objectives.	DevOps Foundation, DevSecOps Foundation
Risk Event	Possible event that could cause harm or loss or affect an organization's ability to achieve its objectives. The management of risk consists of three activities: identifying risks, analyzing risks and managing risks.	DevOps Leader
Risk Management Process	The process by which "risk" is contextualized, assessed, and treated. From ISO 31000: 1) Establish context, 2) Assess risk, 3) Treat risk (remediate, reduce or accept).	DevSecOps Foundation
Robot Framework	TDD framework created and supported by Google.	Continuous Delivery Ecosystem Foundation, Continuous Testing Foundation
Role	Set of responsibilities, activities and authorities granted to a person or team. A role is defined by a process. One person or team may have multiple roles. A set of permissions assigned to a user or group of users to allow a user to perform actions within a system or application.	DevOps Foundation, DevSecOps Foundation
Role-based Access Control (RBAC)	An approach to restricting system access to authorized users.	DevSecOps Foundation
Roll-back	Software changes which have been integrated are removed from the integration.	Continuous Delivery Ecosystem Foundation, Continuous Testing Foundation
Root Cause Analysis (RCA)	Actions take to identify the underlying cause of a problem or incident.	DevOps Foundation, DevSecOps Foundation
Rugged Development (DevOps)	Rugged Development (DevOps) is a method that includes security practices as early in the continuous delivery pipeline as possible to increase cybersecurity, speed, and quality of releases beyond what DevOps practices can yield alone.	DevOps Foundation

Rugged DevOps	Rugged DevOps is a method that includes security practices as early in the continuous delivery pipeline as possible to increase cybersecurity, speed, and quality of releases beyond what DevOps practices can yield alone.	Continuous Delivery Ecosystem Foundation, Continuous Testing Foundation
Runbooks	A collection of procedures necessary for the smooth operation of a service. Previously manual in nature they are now usually automated with tools like Ansible.	Site Reliability Engineering
Runtime Application Self Protection (RASP)	Tools that actively monitor and block threats in the production environment before they can exploit vulnerabilities.	Site Reliability Engineering
Sanity Test	A very basic set of tests that determine if a software is functional at all.	Continuous Delivery Ecosystem Foundation, Continuous Testing Foundation
Scalability	Scalability is a characteristic of a service that describes its capability to cope and perform under an increased or expanding load.	Site Reliability Engineering
Scaled Agile Framework (SAFE)	A proven, publicly available, framework for applying Lean-Agile principles and practices at an enterprise scale.	DevOps Foundation
SCARF Model	A summary of important discoveries from neuroscience about the way people interact socially.	DevOps Leader
Scheduling	Scheduling: the process of planning to release changes into production.	DevOps Leader
Scrum	A simple framework for effective team collaboration on complex projects. Scrum provides a small set of rules that create "just enough" structure for teams to be able to focus their innovation on solving what might otherwise be an insurmountable challenge. (Scrum.org)	Certified Agile Service Manager, DevOps Foundation
Scrum Artifacts	Product Backlog, Sprint Backlog, Burndown Chart, Product Increment	Certified Agile Process Owner
Scrum Components	Scrum's roles, events, artifacts and the rules that bind them together.	Certified Agile Service Manager
Scrum Events	Release Planning Meeting (optional), Sprint Planning Meeting, Sprint, Daily Scrum, Sprint Review, Sprint Retrospective	Certified Agile Process Owner

Scrum Guide	The definition of Scrum concepts and practices, written by Ken Schwaber and Jeff Sutherland.	Certified Agile Service Manager
Scrum Pillars	Pillars that uphold the Scrum framework that include: Transparency, Inspection and Adaption.	Certified Agile Process Owner
Scrum Roles	Product Owner, Development Team (Team) and ScrumMaster. See also Agile Service Management Roles.	Certified Agile Process Owner
Scrum Team	A self-organizing, cross-functional team that uses the Scrum framework to deliver products iteratively and incrementally. The Scrum Team consists of a Product Owner, the Development Team, and a Scrum Master.	DevOps Foundation
Scrum values	A set of fundamental values and qualities underpinning the Scrum framework: commitment, focus, openness, respect and courage.	Certified Agile Process Owner, Certified Agile Service Manager
ScrumMaster	An individual who provides process leadership for Scrum (i.e., ensures Scrum practices are understood and followed) and who supports the Scrum Team by removing impediments.	DevOps Foundation
Secret Detection	Secret Detection aims to prevent that sensitive information, like passwords, authentication tokens, and private keys are unintentionally leaked as part of the repository content.	Site Reliability Engineering
Secrets Management	Secrets management refers to the tools and methods for managing digital authentication credentials (secrets), including passwords, keys, APIs, and tokens for use in applications, services, privileged accounts and other sensitive parts of the IT ecosystem.	Site Reliability Engineering
Secure Automation	Secure automation removes the chance of human error (and wilful sabotage) by securing the tooling used across the delivery pipeline.	Site Reliability Engineering
Security (Information Security)	Practices intended to protect the confidentiality, integrity and availability of computer system data from those with malicious intentions.	DevOps Foundation, DevSecOps Foundation
Security as Code	Automating and building security into DevOps tools and practices, making it an essential part of tool chains and workflows.	DevOps Foundation, DevSecOps Foundation

Security tests	The purpose of the test is to determine if an EUT meets its security requirements. An example is a test that determines if an EUT processes login credentials properly.	Continuous Delivery Ecosystem Foundation, Continuous Testing Foundation
Selenium	Popular open-source tool for software testing GUI and web applications.	Continuous Delivery Ecosystem Foundation, Continuous Testing Foundation
Self-healing	Self-healing means the ability of services and underlying environments to detect and resolve problems automatically. It eliminates the need for manual human intervention.	Site Reliability Engineering
Self-organizing Team	Management principle in which a team chooses how best to accomplish their work, rather than being directed by others outside the team. Self-organization happens within boundaries and against given goals (i.e., what to do).	Certified Agile Process Owner
Self-organizing	The management principle that teams autonomously organize their work. Self-organization happens within boundaries and against given goals. Teams choose how best to accomplish their work, rather than being directed by others outside the team.	Certified Agile Service Manager
Serverless	A code execution paradigm were no underlying infrastructure or dependencies are needed, moreover a piece of code is executed by a service provider (typically cloud) who takes over the creation of the execution environment. Lambda functions in AWS and Azure Functions are examples.	Site Reliability Engineering
Service	Means of delivering value to customers by facilitating outcomes customers want to achieve without the ownership of specific costs and risks.	DevOps Foundation, DevSecOps Foundation
Service Catalog	Subset of the Service Portfolio that consists of services that are live or available for deployment. Has two aspects: The Business/Customer Service Catalog (visible to customers) and the Technical/Supporting Service Catalog. (ITIL definition)	DevOps Foundation
Service Design	One of the ITIL Core publications and a stage of the service lifecycle.	DevOps Foundation

Service Desk	Single point of contact between the service provider and the users. Tools like Service Now are used for managing the lifecycle of services as well as internal and external stakeholder engagement.	DevOps Foundation
Service Level Agreement (SLA)	Written agreement between an IT service provider and its customer(s) that defines key service targets and responsibilities of both parties. An SLA may cover multiple services or customers. (ITIL definition)	Certified Agile Process Owner, DevOps Foundation, Site Reliability Engineering
Service Level Indicator (SLI)	SLI's are used to communicate quantitative data about services, typically to measure how the service is performing against an SLO.	Site Reliability Engineering
Service Level Management	Process that ensures all current and planned IT services are delivered to agreed achievable targets. (ITIL definition)	Certified Agile Process Owner
Service Level Objective (SLO)	An SLO is a goal for how well a product or service should operate. SLO's are set based on what an organization is expecting from a service.	Site Reliability Engineering
Service Lifecycle	Structure of the ITIL Core guidance.	DevOps Foundation
Service Management	Set of specialized organizational capabilities for providing value to customers in the form of services. (ITIL definition)	DevOps Foundation
Service Management Office (SMO)	Function that coordinates all processes and functions that manage a service provider's services throughout their lifecycle. Process Owners may report directly or via a 'dotted' reporting line to the SMO.	Certified Agile Process Owner
Service Operation	One of the ITIL Core publications and a stage of the service lifecycle.	DevOps Foundation
Service Provider	Organization that supplies services to one or more internal or external customers. (ITIL definition)	DevOps Foundation
Service Request	User request for a standard service from an IT service provider. (ITIL definition)	DevOps Foundation
Service Strategy	One of the ITIL Core publications and a stage of the service lifecycle.	DevOps Foundation
Service Transition	One of the ITIL Core publications and a stage of the service lifecycle.	DevOps Foundation

Seven Pillars of DevOps	Seven distinct "pillars" provide a foundation for DevOps systems which include Collaborative Culture, Design for DevOps, Continuous Integration, Continuous Testing, Continuous Delivery and Deployment, Continuous Monitoring and Elastic Infrastructures and Tools.	Continuous Delivery Ecosystem Foundation
Shift Left	An approach that strives to build quality into the software development process by incorporating testing early and often. This notion extends to security architecture, hardening images, application security testing, and beyond.	DevOps Foundation, DevSecOps Foundation
SilkTest	Automated function and regression testing of enterprise applications. Licensed by Borland.	Continuous Testing Foundation
Simian Army	The Simian Army is a suite of failure-inducing tools designed by Netflix. The most famous example is Chaos Monkey which randomly terminates services in production as part of a Chaos Engineering approach.	Site Reliability Engineering
Single Point of Failure (SPOF)	A single point of failure (SPOF) is a part of a system that, if it fails, will stop the entire system from working.	DevOps Foundation
Site Reliability Engineering (SRE)	The discipline that incorporates aspects of software engineering and applies them to infrastructure and operations problems. The main goals are to create scalable and highly reliable software systems.	Site Reliability Engineering
Six Sigma	Disciplined, data-driven approach that focuses on reducing defects by measuring standard deviations from an expected norm.	Certified Agile Process Owner
SMART Goals	Specific, measurable, achievable, relevant and time-bound goals.	DevOps Foundation
Smoke Test	A basic set of functional tests that are run immediately after a software component is built. Same as CI Regression Test.	Continuous Delivery Ecosystem Foundation, Continuous Testing Foundation
Snapshot	Report of pass/fail results for a specific build.	Continuous Delivery Ecosystem Foundation, Continuous Testing Foundation
Snippets	Stored and shared code snippets to allow collaboration around specific pieces of code. Also allows code snippets to be used in other code-bases. BitBucket and GitLab allow this.	Site Reliability Engineering

SOAP	Simple Object Access Protocol (SOAP) is an XML-based messaging protocol for exchanging information among computers.	Continuous Delivery Ecosystem Foundation, Continuous Testing Foundation
Software Composition Analysis	A tool that checks for libraries or functions in source code that have known vulnerabilities.	DevSecOps Foundation
Software Defined Networking (SDN)	Software-Defined Networking (SDN) is a network architecture approach that enables the network to be intelligently and centrally controlled, or 'programmed,' using software applications.	Site Reliability Engineering
Software Delivery Lifecycle (SDLC)	The process used to design, develop and test high quality software.	DevOps Leader, Site Reliability Engineering
Software Version Management System	A repository tool which is used to manage software changes. Examples are: Azure DevOps, BitBucket, Git, GitHub, GitLab, VSTS.	Continuous Delivery Ecosystem Foundation, Continuous Testing Foundation
Software-as-a-Service (SaaS)	Category of cloud computing services in which software is licensed on a subscription basis.	DevOps Foundation, Continuous Delivery Ecosystem Foundation, Continuous Testing Foundation
Source Code Tools	Repositories for controlling source code for key assets (application and infrastructure) as a single source of truth.	DevOps Foundation, DevOps Leader
Spotify Squad Model	An organizational model that helps teams in large organizations behave like startups and be nimble.	DevOps Foundation, DevOps Leader
Sprint	A period of 2-4 weeks during which an increment of product work is completed.	Certified Agile Process Owner, Certified Agile Service Manager, Continuous Delivery Ecosystem Foundation
Sprint (Scrum)	A time-boxed iteration of work during which an increment of product functionality is implemented.	DevOps Foundation
Sprint Backlog	Subset of the backlog that represents the work that must be completed to realize the Sprint Goal.	Certified Agile Process Owner, DevOps Foundation
Sprint Goal	Purpose and objective of a Sprint, often expressed as a business problem that is going to be solved.	Certified Agile Process Owner, Certified Agile Service Manager

Sprint Planning Meeting	A 4 to 8-hour time-boxed event that defines the Sprint Goal, the increment of the Product Backlog that will be completed during the Sprint and how it will be completed.	Certified Agile Process Owner, Certified Agile Service Manager
Sprint Retrospective	A 1.5 to 3-hour time-boxed event during which the Team reviews the last Sprint and identifies and prioritizes improvements for the next Sprint.	Certified Agile Process Owner, Certified Agile Service Manager
Sprint Review	A time-boxed event of 4 hours or less where the Team and stakeholders inspect the work resulting from the Sprint and update the Product Backlog.	Certified Agile Process Owner, Certified Agile Service Manager
Spyware	Software that is installed in a computer without the user's knowledge and transmits information about the user's computer activities over back to the threat agent.	DevSecOps Foundation
Squads	A cross-functional, co-located, autonomous, self-directed team.	DevOps Leader
Stakeholder	Person who has an interest in an organization, project or IT service. Stakeholders may include customers, users and suppliers. (ITIL definition).	DevOps Foundation, DevSecOps Foundation
Stability	The sensitivity a service has to accept changes and the negative impact that may be caused by system changes. Services may have reliability, in that if functions over a long period of time, but may not be easy to change and so does not have stability.	Site Reliability Engineering
Standard Change	Pre-approved, low risk change that follows a procedure or work instruction. (ITIL definition)	DevOps Foundation, DevSecOps Foundation
Static Application Security Testing (SAST)	A type of testing that checks source code for bugs and weaknesses.	DevSecOps Foundation
Static Code Analysis	The purpose of the test is to detect source code logic errors and omissions such as memory leaks, unutilized variables, unutilized pointers.	Continuous Delivery Ecosystem Foundation, Continuous Testing Foundation
Status Page	Service pages that easily communicate the status of services to customers and users.	Site Reliability Engineering
Sticks	Negative incentives, for discouraging or punishing undesired behaviors.	DevSecOps Foundation

Storage Security	A specialty area of security that is concerned with securing data storage systems and ecosystems and the data that resides on these systems.	Site Reliability Engineering
Stormstack	A commercial orchestration tool based on event triggers instead of time based.	Continuous Testing Foundation
StoStaKee	This stands for stop, start, and keep: this is an interactive time-boxed exercise focused on past events.	DevOps Leader
Strategic Sprint	A 2-4 week timeboxed Sprint during which strategic elements that were defined during the Process Planning Meeting are completed so that the Team can move on to designing the activities of the process.	Certified Agile Process Owner, Certified Agile Service Manager
Structural Changes	Changes in the hierarchy of authority, goals, structural characteristics, administrative procedures and management systems.	DevOps Leader
Supplier	External (third party) supplier, manufacturer or vendor responsible for supplying goods or services that are required to deliver IT services.	DevOps Foundation
Synthetic Monitoring	Synthetic monitoring (also known as active monitoring, or semantic monitoring) runs a subset of an application's automated tests against the system on a regular basis. The results are pushed into the monitoring service, which triggers alerts in case of failures.	Continuous Delivery Ecosystem Foundation
System of Record	A system of record is the authoritative data source for a data element or data entity.	DevOps Foundation, DevSecOps Foundation
System Test	The purpose of the test is to determine if a complete system performs as expected in its intended configurations.	Continuous Delivery Ecosystem Foundation, Continuous Testing Foundation
System Under Test (SUT)	The EUT is an entire system. E.g. Bank teller machine is being tested.	Continuous Delivery Ecosystem Foundation, Continuous Testing Foundation
Tag-Based Test Selection Method	Tests and Code modules are pre-assigned tags. Tests are selected for a build matching pre-assigned tags.	Continuous Delivery Ecosystem Foundation, Continuous Testing Foundation
Target Operating Model	A description of the desired state of the operating model of an organisation.	DevOps Leader

Teal Organization	An emerging organizational paradigm that advocates a level of consciousness including all previous world views within the operations of an organisation.	DevOps Leader
Team Dynamics	A measurement of how a team works together. Includes team culture, communication styles, decision making ability, trust between members, and the willingness of the team to change.	DevOps Leader
Techno-Economic Paradigm Shifts	Techno-economic paradigm shifts are at the core of general, innovation-based theory of economic and societal development as conceived by Carlota Perez.	DevOps Leader
Telemetry	Telemetry is the collection of measurements or other data at remote or inaccessible points and their automatic transmission to receiving equipment for monitoring.	Site Reliability Engineering
Test Architect	Person who has responsibility for defining the overall end-to-end test strategy for an EUT.	Continuous Delivery Ecosystem Foundation, Continuous Testing Foundation
Test Artifact Repository	Database of files used for testing.	Continuous Delivery Ecosystem Foundation, Continuous Testing Foundation
Test Campaign	A test campaign may include one or more test sessions.	Continuous Delivery Ecosystem Foundation, Continuous Testing Foundation
Test Case	Set of test steps together with data and configuration information. A test case has a specific purpose to test at least one attribute of the EUT.	Continuous Delivery Ecosystem Foundation, Continuous Testing Foundation
Test Creation Methods	This is a class of test terms which refers to the methodology used to create test cases.	Continuous Delivery Ecosystem Foundation, Continuous Testing Foundation

Test Driven Development (TDD)	Test-driven development (TDD) is a software development process in which the developer writes a test before composing code. They then follow this process: 1. Write the test 2. Run the test and any others that are relevant and see them fail 3. Write the code 4. Run test(s) 5. Refactor code if needed 6. Repeat Unit level tests and/or application tests are created ahead of the code that is to be tested.	Continuous Delivery Ecosystem Foundation, DevOps Foundation, Continuous Testing Foundation
Test Duration	The time it takes to run a test. E.g. # hours per test	Continuous Delivery Ecosystem Foundation, Continuous Testing Foundation
Test Environment	The test environment refers to the operating system (e.g. Linus, windows version etc.), configuration of software (e.g. parameter options), dynamic conditions (e.g. CPU and memory utilization) and physical environment (e.g. power, cooling) in which the tests are performed.	Continuous Delivery Ecosystem Foundation, Continuous Testing Foundation
Test Fast	A CT tenet referring to accelerated testing.	Continuous Delivery Ecosystem Foundation, Continuous Testing Foundation
Test Framework	A set of processes, procedures, abstract concept and environment in which automated tests are designed and implemented.	Continuous Delivery Ecosystem Foundation, Continuous Testing Foundation
Test Harness	A tool which enables the automation of tests. It refers to the system test drivers and other supporting tools that requires to execute tests. It provides stubs and drivers which are small programs that interact with the software under test.	Continuous Delivery Ecosystem Foundation, Continuous Testing Foundation
Test Hierarchy	This is a class of terms describes the organization of tests into groups.	Continuous Delivery Ecosystem Foundation, Continuous Testing Foundation
Test Methodology	This class of terms identifies the general methodology used by a test. Examples are White Box, Black Box	Continuous Delivery Ecosystem Foundation, Continuous Testing Foundation

Test result repository	Database of test results.	Continuous Delivery Ecosystem Foundation, Continuous Testing Foundation
Test Results Trend- based	A matrix of correlation factors correlates test cases and code modules according to test result (verdict).	Continuous Delivery Ecosystem Foundation, Continuous Testing Foundation
Test Roles	This class of terms identifies general roles and responsibilities for people relevant to testing.	Continuous Delivery Ecosystem Foundation, Continuous Testing Foundation
Test Script	Automated test case. A single test script may be implemented one or more test cases depending on the data.	Continuous Delivery Ecosystem Foundation, Continuous Testing Foundation
Test Selection Method	This class of terms refers to the method used to select tests to be executed on a version of an EUT.	Continuous Delivery Ecosystem Foundation, Continuous Testing Foundation
Test Session	Set of one or more test suites that are run together on a single build at a specific time.	Continuous Delivery Ecosystem Foundation, Continuous Testing Foundation
Test Suite	Set of test cases that are run together on a single build at a specific time.	Continuous Delivery Ecosystem Foundation, Continuous Testing Foundation
Test Trend	History of verdicts.	Continuous Delivery Ecosystem Foundation, Continuous Testing Foundation
Test Type	Class that indicates what the purpose of the test is.	Continuous Delivery Ecosystem Foundation, Continuous Testing Foundation
Test Version	The version of files used to test a specific build.	Continuous Delivery Ecosystem Foundation, Continuous Testing Foundation
Tester	Individual who has responsibility to test a system or service.	Continuous Delivery Ecosystem Foundation, Continuous Testing Foundation

Testing Tools	Tools that verify code quality before passing the build.	DevOps Leader
The Advice Process	Any person deciding must seek advice from everyone meaningfully affected by the decision and people with expertise in the matter. Advice received must be taken into consideration, though it does not have to be accepted or followed. The objective of the advice process is not to form consensus, but to inform the decision-maker so that they can make the best decision possible. Failure to follow the advice process undermines trust and unnecessarily introduces risk to the business.	DevSecOps Foundation
The Checkbox Trap	The situation wherein an audit-centric perspective focuses exclusively on "checking the box" on compliance requirements without consideration for overall security objectives.	DevSecOps Foundation
The Power of TED	The Power of TED* offers an alternative to the Karpman Drama Triangle with its roles of Victim, Persecutor, and Rescuer. The Empowerment Dynamic (TED) provides the antidote roles of Creator, Challenger and Coach and a more positive approach to life's challenges.	DevOps Leader
The Three Ways	Key principles of DevOps – Flow, Feedback, Continuous experimentation and learning.	DevOps Foundation, DevSecOps Foundation, Site Reliability Engineering
Theory of Constraints	Methodology for identifying the most important limiting factor (i.e., constraint) that stands in the way of achieving a goal and then systematically improving that constraint until it is no longer the limiting factor.	DevOps Foundation, DevSecOps Foundation
Thomas Kilmann Inventory (TKI)	Measures a person's behavioral choices under certain conflict situations.	DevOps Foundation
Threat Agent	An actor, human or automated, that acts against a system with intent to harm or compromise that system. Sometimes also called a "Threat Actor."	DevSecOps Foundation
Threat Detection	Refers to the ability to detect, report, and support the ability to respond to attacks. Intrusion detection systems and denial-of-service systems allow for for some level of threat detection and prevention.	

Threat Intelligence	Information pertaining to the nature of a threat or the actions a threat may be known to be perpetrating. May also include "indicators of compromise" related to a given threat's actions, as well as a "course of action" describing how to remediate the given threat action.	DevSecOps Foundation
Threat Modeling	A method that ranks and models potential threats so that the risk can be understood and mitigated in the context of the value of the application(s) to which they pertain.	DevSecOps Foundation
Time to Market	The period of time between when an idea is conceived and when it is available to customers.	DevOps Leader
Time to Value	Measure of the time it takes for the business to realize value from a feature or service.	DevOps Foundation, DevSecOps Foundation
Time Tracking	Tools that allow for time to be tracked, either against individual issues or other work or project types.	Site Reliability Engineering
Time-box	Maximum duration of a Scrum event.	Certified Agile Process Owner, Certified Agile Service Manager
Toil	A kind of work tied to running a production service that tends to be manual, repetitive, automatable, tactical, devoid of enduring value.	Site Reliability Engineering
Tool	This class describes tools that orchestrate, automate, simulate and monitor EUT's and infrastructures.	Continuous Delivery Ecosystem Foundation, Continuous Testing Foundation
Toolchain	A philosophy that involves using an integrated set of complimentary task specific tools to automate an end to end process (vs. a single-vendor solution).	DevOps Foundation
Touch Time	In a Lean Production system the The touch time is the time that the product is actually being worked on, and value is being added.	DevOps Leader
Tracing	Tracing provides insight into the performance and health of a deployed application, tracking each function or microservice which handles a given request.	Site Reliability Engineering
Traffic Volume	The amount of data sent and received by visitors to a service (e.g. a website or API).	Site Reliability Engineering

Training From the Back of the Room	An accelerated learning model in line with agile values and principles using the 4Cs instructional design "map" (Connection, Concept, Concrete Practice, Conclusion).	
Transformational Leadership	A leadership model in which leaders inspire and motivate followers to achieve higher performance by appealing to their values and sense of purpose, facilitating wide-scale organizational change (State of DevOps Report, 2017).	DevOps Leader
Tribe Lead	A senior technical leader that has broad and deep technical expertise across all the squads' technical areas. A group of squads working together on a common feature set, product or service is a tribe in Spotify's definitions.	DevOps Leader
Tribes	A collection of squads with a long-term mission that work on/in a related business capability.	DevOps Leader
Trojan (horses)	Malware that carries out malicious operations under the appearance of a desired operation such as playing an online game. A Trojan horse differs from a virus because the Trojan binds itself to non-executable files, such as image files, audio files whereas a virus requires an executable file to operate.	DevSecOps Foundation
Trunk	The primary source code integration repository for a software product.	Continuous Delivery Ecosystem Foundation, Continuous Testing Foundation
Unit Test	The purpose of the test is to verify code logic.	Continuous Delivery Ecosystem Foundation, Continuous Testing Foundation
Usability Test	The purpose of the test is to determine if humans have a satisfactory experience when using an EUT.	Continuous Delivery Ecosystem Foundation, Continuous Testing Foundation
User	Consumer of IT services. Or, the identity asserted during authentication (aka username).	DevOps Foundation, DevSecOps Foundation
User and Entity Behavior Analytics (UEBA)	A machine learning technique to analyze normal and "abnormal" user behaviour with the aim of preventing the latter.	Site Reliability Engineering

User Story	Statement written from the user's business perspective that describes how the user will achieve a goal from a feature of the product. User stories are captured in the Product Backlog (or Process Backlog).	Certified Agile Process Owner, Certified Agile Service Manager
Value Added Time	The amount of time spent on an activity that creates value (e.g., development, testing).	DevOps Leader
Value Efficiency	Being able to produce value with the minimum amount of time and resources.	DevOps Leader
Value Stream	All of the activities to go from a customer request to a delivered product or service.	DevOps Foundation
Value Stream Mapping	Lean tool that depicts the flow of information, materials and work across functional silos with an emphasis on quantifying waste, including time and quality.	DevOps Foundation
Value Stream Management	The ability to visualize the flow of value delivery through the DevOps lifecycle. Gitlab CI and the Jenkins extension (from Cloud Bees) DevOptics can provide this visualization.	Site Reliability Engineering
Value Stream Owner	Individual accountable to senior management for improving the value to non-value ratio of a given product or service.	Certified Agile Process Owner
Variable Speed IT	An approach where traditional and digital processes co-exist within an organization while moving at their own speed.	DevOps Foundation
Velocity	Measure of the quantity of work done in a pre-defined interval. The amount of work an individual or team can complete in a given amount of time.	DevOps Foundation, DevSecOps Foundation, Site Reliability Engineering
Verdict	Test result classified as Fail, Pass or Inconclusive.	Continuous Delivery Ecosystem Foundation, Continuous Testing Foundation
Version control tools	Ensure a 'single source of truth' and enable change control and tracking for all production artifacts.	DevOps Foundation
Vertical Scaling	Computing resources are scaled higher to increase processing speed e.g. using faster computers to run more tasks faster.	Continuous Testing Foundation

Virus (Computer)	Malicious executable code attached to a file that spreads when an infected file is passed from system to system that could be harmless (but annoying) or it could modify or delete data.	DevSecOps Foundation
Voice of the Customer (VOC)	A process that captures and analyzes customer requirements and feedback to understand what the customer wants.	DevOps Foundation
Vulnerability	A weakness in a design, system, or application that can be exploited by an attacker.	DevSecOps Foundation
Vulnerability Intelligence	Information describing a known vulnerability, including affected software by version, relative severity of the vulnerability (for example, does it result in escalation of privileges for user role, or does it cause a denial of service), exploitability of the vulnerability (how easy/hard it is to exploit), and sometimes current rate of exploitation in the wild (is it being actively exploited or is it just theoretical). This information will also often include guidance on what software versions are known to have remediated the described vulnerability.	DevSecOps Foundation
Vulnerability management	The process of identifying and remediating vulnerabilities.	DevSecOps Foundation
Wait Time	The amount of time wasted on waiting for work (e.g., waiting for development and test infrastructure, waiting for resources, waiting for management approval).	DevOps Leader
Waste (Lean Manufacturing)	Any activity that does not add value to a process, product or service.	Certified Agile Process Owner, Certified Agile Service Manager, DevOps Foundation, DevOps Leader
Water-scrum-fall	A hybrid approach to application lifecycle management that combines waterfall and Scrum development can complete in a given amount of time.	Continuous Delivery Ecosystem Foundation
Waterfall (Project Management)	Linear and sequential approach to managing software design and development projects in which progress is seen as flowing steadily (and sequentially) downwards (like a waterfall).	Certified Agile Service Manager, Continuous Delivery Ecosystem Foundation, DevOps Foundation

Weakness	An error in software that can be exploited by an attacker to compromise the application, system, or the data contained therein. Also called a vulnerability.	DevSecOps Foundation
Web Applicaion Firewall (WAF)	Tools that examine traffic being sent to an application and can block anything that looks malicious.	Site Reliability Engineering
Web IDE	Tools that have a web client integrated development environment. Enables developer productivity without having to use a local development tool.	Site Reliability Engineering
Westrum (Organization Types)	Ron Westrum developed a typology of organizational cultures that includes three types of organizations: Pathological (power-oriented), Bureaucratic (rule-oriented) and Generative (performance-oriented).	DevSecOps Foundation, Site Reliability Engineering
White-Box Testing (or Clear-, Glass-, Transparent-Box Testing or Structural Testing)	Test cases use extensive knowledge of the internal design structure or workings of an application, as opposed to its functionality (i.e. Black-Box Testing).	Continuous Delivery Ecosystem Foundation, Continuous Testing Foundation
Whitelisting	Application whitelisting is the practice of specifying an index of approved software applications that are permitted to be present and active on a computer system.	Continuous Delivery Ecosystem Foundation
Wicked Questions	Wicked questions are used to expose the assumptions which shape our actions and choices. They are questions that articulate the embedded, and often contradictory assumptions, we hold about an issue, a problem or a context.	DevOps Leader
Wiki	Knowledge sharing can be enabled by using tools like Confluence which create a rich Wiki of content	Site Reliability Engineering
Wilber's Quadrants	A model that recognises four modes of general approach for human beings. Two axes are used: on one axis people tend towards individuality OR collectivity.	DevOps Leader
Work in Progress (WIP)	Any work that has been started but has not been completed.	DevOps Foundation
Workaround	Temporary way to reduce or eliminate the impact of incidents or problems. May be logged as a known error in the Known Error Database. (ITIL definition).	DevOps Foundation, DevSecOps Foundation

World Café	Is a structured conversational process for knowledge sharing in which groups of people discuss a topic at several tables, with individuals switching tables periodically and getting introduced to the previous discussion at their new table by a "table host".	DevOps Leader
Worms (Computer)	Worms replicate themselves on a system by attaching themselves to different files and looking for pathways between computers. They usually slow down networks and can run by themselves (where viruses need a host program to run).	DevSecOps Foundation

DevOps Foundation Course: Value Added Resources

This document provides links to articles and videos related to the DevOps Foundation course from DevOps Institute. This information is provided to enhance your understanding of DevOps Foundation-related concepts and terms and is not examinable. Of course, there is a wealth of other videos, blogs and case studies on the web. We welcome suggestions for additions.

Videos Featured in the Course

Module Featured	Title & Link
1: Exploring DevOps	'A Short History of DevOps' with Damon Edwards (11:47)
1: Exploring DevOps	Abbreviated version of Simon Sinek's Ted Talk 'Start with Why - How Great Leaders Inspire Action' (5:00)
2: Core DevOps Practices	'Gene Kim Defines The Three Ways of The Phoenix Project' (3:31)
3: Key DevOps Principles	'GitHub Professional Guide: Continuous Integration & Continuous Delivery' (6:00)
4: Business & Technology Frameworks	'Spotify Engineering Culture Part 1' with Henrik Kniberg (13:12)
5: DevOps Values: Culture, Behaviors & Operating Models	'Spotify Engineering Culture Part 2' with Henrik Kniberg (13:27)
6: DevOps Values: Automation & Architecting DevOps Toolchains	'The DevOps Toolchain' with John Okoro (7:43)
7: DevOps Values: Measurement, Metrics & Reporting	'Double the Awesome' with Dr. Nicole Forsgren (21:46)
8: DevOps Values: Sharing, Shadowing & Evolving	'DevOps: A Culture of Sharing' with Gareth Rushgrove (2:19)

DevOps Reports

Report & Link	Writers/Publishers
2020 DevSecOps Community Survey	Sonatype
2020 Global DevSecOps Survey	Gitlab
The Accelerate State of DevOps Report 2019	Dr. Nicole Forsgren, Gene Kim & Jez Humble in collaboration with Google Cloud Platform (GCP)

DevOps Foundation Course: Value Added Resources

The State of DevOps Report 2020	Puppet Labs, CircleCI and Splunk
Upskilling: Enterprise DevOps Skills Report 2020	DevOps Institute

DevOps Articles

Article Title & Author	Relevant Module
'5 Things DevOps is Not' from devops.com	1: Exploring DevOps
'6 DevOps Recruiting Tips: How to Land the Right People' on TechBeacon	8: Sharing, Shadowing & Evolving
'7 DevOps Roles You Need to Succeed' on TechBeacon	5: DevOps Values: Culture, Behaviors & Operating Models
'7 Keys to Finding Phenomenal DevOps Talent' on TechBeacon	8: Sharing, Shadowing & Evolving
'10 Ways Machine Learning Can Optimize DevOps' on TechBeacon	6: Automation & Architecting Toolchains
'A Different Drumbeat: Using Kanban for DevOps to Smooth Out Your Scrum Cycles' by Nate Berent-Spillson	3: Key DevOps Principles
'A Four Quadrant Look at the DevOps Toolchain' by Scott Johnston	6: Automation & Architecting Toolchains
'A Personal Reinterpretation of The Three Ways' by Tim Hunter	2: Core DevOps Practices
'Best Practices for Using Value Stream Mapping as a Continuous Improvement Tool' by R. Keith Mobley	4: Business & Technology Frameworks
'Blue-Green Deployments, A/B Testing, and Canary Releases' by Christian Posta	3: Key DevOps Principles

DevOps Foundation Course: Value Added Resources

'Building a Healthy DevOps Culture' by Michael Butt	5: DevOps Values: Culture, Behaviors & Operating Models
'Building a Learning Organization' on HBR	4: Business & Technology Frameworks
'ChatOps: Communicating at the Speed of DevOps' by George Hulme	6: Automation & Architecting Toolchains
'Codifing DevOps Practices' by Patrick DeBois	3: Key DevOps Principles
'Communities of Practice: The Missing Piece of Your Agile Organisation' by Emily Webber	5: DevOps Values: Culture, Behaviors & Operating Models
'Continuous Delivery' by Martin Fowler	3: Key DevOps Principles
'Continuous Delivery: Anatomy of the Deployment Pipeline' by Jez Humble & Dave Farley	3: Key DevOps Principles
'Continuous Everything in DevOps… What is the difference between CI, CD… ?' by Micro Hering	3: Key DevOps Principles
'Continuous Integration' on ThoughtWorks	3: Key DevOps Principles
'Cultural Debt Can Be the Primary Driver of Technical Debt' by Rick Brenner	5: DevOps Values: Culture, Behaviors & Operating Models
'Culture Isn't a Swear Word' by Jonathan Fletcher	5: DevOps Values: Culture, Behaviors & Operating Models
'Data-Driven DevOps: Use Metrics to Guide Your Journey' by Jonah Kowell	7: Measurement, Metrics & Reporting
'DevOps and IT Support: 4 Principles to Keep Your Team Ahead of the Curve' on devops.com	1: Exploring DevOps
'DevOps and Kanban - Match Made in Heaven' by UpGuard	3: Key DevOps Principles
'DevOps, Cloud, and the Lean "Wheel of Waste"' by Richard Seroter	4: Business & Technology Frameworks
'DevOps Culture' by John Willis	5: DevOps Values: Culture, Behaviors & Operating Models

DevOps Foundation Course: Value Added Resources

'DevOps Culture: Westrum Organizational Culture' by Google Gloud	5: DevOps Values: Culture, Behaviors & Operating Models
'DevOps Demystified' by Ben Rockwood	7: Measurement, Metrics & Reporting
'DevOps: To Measure Value then Measure Speed' by Stephen Williams	7: Measurement, Metrics & Reporting
'Doing ChatOps in Microsoft Teams' by Helen Beal	3: Key DevOps Principles
'Epics, Stories, Themes and Initiatives' by Atlassian	4: Business & Technology Frameworks
'First Impressions at Etsy' by Jason Shen (including reference to the 3-armed sweater)	3: Key DevOps Principles
'From Containers to Microservices: Modernizing Legacy Applications' by David Linthicum	6: Automation & Architecting Toolchains
'From Darwin to DevOps: John Willis and Gene Kim Talk about Life after The Phoenix Project' by Helen Beal	4: Business & Technology Frameworks
'How to Find Your Continuous Delivery Rhythm' from devops.com	1: Exploring DevOps
'How to Implement a Solid DevOps Team Structure' by Alex Barrett	8: Sharing, Shadowing & Evolving
'How to Reduce Employee Resistance to Change' by Susan. M. Heathfield	5: DevOps Values: Culture, Behaviors & Operating Models
'Is Yours a Learning Organization?' on HBR	4: Business & Technology Frameworks
'Jesse Robbins Discusses DevOps and Cloud Computing' on Thoughtworks' blog	6: Automation & Architecting Toolchains
'Just What is a DevOps Engineer?' on devops.com	8: Sharing, Shadowing & Evolving
'Inside Atlassian: How IT & SRE use ChatOps to Run Incident Management' by Sean Regan	3: Key DevOps Principles
'Let's Fund Teams Not Projects' from the DEFRA Digital blog, .gov.uk	4: Business & Technology Frameworks
'Machine Learning: AI Driving DevOps Evolution' by Tony Bradley	6: Automation & Architecting Toolchains

DevOps Foundation Course: Value Added Resources

'Measure Efficiency, Effectiveness, and Culture, to Optimize DevOps Transformation: Metrics for DevOps Initiatives' from IT Revolution	7: Measurement, Metrics & Reporting
'Misconceptions About Kanban' by Leon Tranter	3: Key DevOps Principles
'Resilience Engineering' by Erik Hollnagel	3: Key DevOps Principles
'SAFe for Lean Enterprises' by Scaled Agile	4: Business & Technology Frameworks
SAFe White Paper by Scaled Agile	4: Business & Technology Frameworks
'T-Shaped Developers are the New Normal' by David Walker	8: Sharing, Shadowing & Evolving
'The 7 Skills Ops Pros Need to Succeed in DevOps' by George Hulme	8: Sharing, Shadowing & Evolving
'The Andon Cord' by John Willis on IT Revolution	4: Business & Technology Frameworks
'The Biggest Myth in Building Learning Culture' by Shannon Tipton	2: Core DevOps Practices
'The Convergence of DevOps' by John Willis on IT Revolution	4: Business & Technology Frameworks
'The DevOps Movement Fits Perfectly with ITSM' by Greg Strydom	4: Business & Technology Frameworks
'The Future of DevOps: 21 Predictions for 2021' from TechBeacon	1: Exploring DevOps
"The Industry Just Can't Decide About DevOps Teams" by Helen Beal	8: Sharing, Shadowing & Evolving
'The Mission of a DevOps Team' by Casey West	8: Sharing, Shadowing & Evolving
'Theory of Constraints' by Lean Production	2: Core DevOps Practices
'There's No Such Thing as a DevOps Team' by Jez Humble	8: Sharing, Shadowing & Evolving
'Top 25 Lean Tools' on Lean Production	4: Business & Technology Frameworks
'Toyota Kata: Habits for Continuous Improvement' by Håkan Forss	4: Business & Technology Frameworks

'Tracking Every Release' Code as Craft (Etsy)	7: Measurement, Metrics & Reporting
'Transforming the Annual Budgeting Process for DevOps' by Mustafa Kapadia	4: Business & Technology Frameworks
'Understanding DevOps – Part 4: Continuous Testing and Continuous Monitoring' by Sanjeev Sharma	3: Key DevOps Principles
'Understanding the Kubler-Ross Change Curve' on Cleverism	5: DevOps Values: Culture, Behaviors & Operating Models
'Use DevOps to Turn IT Into a Strategic Weapon' by Damon Edwards	1: Exploring DevOps
'Waste Not, Want Not: A Simplified Value Stream Map for Uncovering Waste' by J Meadows	4: Business & Technology Frameworks
'What Are the New Skills and Roles DevOps Requires?' by UpGuard	8: Sharing, Shadowing & Evolving
'What DevOps Means to Me' by John Willis	1: Exploring DevOps
'What is Site Reliability Engineering?' an interview with Niall Murphy and Ben Treynor at Google	3: Key DevOps Principles
'What Happens to Test in a DevOps World' on devops.com	3: Key DevOps Principles
'What's Lost With a DevOps Team' by Michael Nygard	8: Sharing, Shadowing & Evolving
'What's the Difference Between AI, Machine Learning and Deep Learning?' by Michael Copeland	6: Automation & Architecting Toolchains
'Why DevOps Engineer is the Number One Hardest Tech Job to Fill' by Logicworks	8: Sharing, Shadowing & Evolving
'Why Everyone Needs DevOps Now' by Gene Kim	2: Core DevOps Practices

WebSites

Title	Link
Agile Manifesto	http://www.agilemanifesto.org/
Beyond Budgeting	https://bbrt.org/
DevOps Institute	https://devopsinstitute.com/

DevOps Topologies (Matthew Skelton & Manuel Pais)	https://web.devopstopologies.com/
DevOps.com	https://devops.com/
DevOpsDays	https://www.devopsdays.org/
DevSecOps Reference Architectures (Sonatype)	https://www.sonatype.com/devsecops-reference-architectures
Hubot by Github	https://hubot.github.com/
IT Revolution	https://itrevolution.com/
Periodic Table of DevOps Tools (Digital.ai)	https://digital.ai/periodic-table-of-devops-tools
Principles of Chaos Engineering	https://principlesofchaos.org
Rugged Software	https://ruggedsoftware.org/
SAFe	https://www.scaledagileframework.com
Scrum.org	https://www.scrum.org/
Scrum Guide	https://www.scrumguides.org/scrum-guide.html
Theory of Constraints Institute	https://www.tocinstitute.org/

DevOps & Engineering Blogs

Blog	Link
AirBNB Engineering & Data Science	https://medium.com/airbnb-engineering
Backstage Blog (SoundCloud)	https://developers.soundcloud.com/blog/
BlackRock Blog	http://rockthecode.io/
code.flickr.com	http://code.flickr.net/
DEFRA Digital	https://defradigital.blog.gov.uk/
Deliveroo Engineering Team	https://deliveroo.engineering/
Dropbox Tech Blog	https://blogs.dropbox.com/tech/

eBay Tech Blog	https://www.ebayinc.com/stories/blogs/tech/
Etsy - Code as Craft	https://codeascraft.com/
Eventbrite Engineering	https://www.eventbrite.com/engineering/
Facebook Engineering	https://www.facebook.com/Engineering
Github Engineering	https://githubengineering.com/
Google Developers	https://developers.googleblog.com/
Heroku Engineering	https://blog.heroku.com/engineering
HubSpot Engineering	https://product.hubspot.com/blog/topic/engineering
Instagram Engineering	https://instagram-engineering.com/
Kickstarter Engineering	https://kickstarter.engineering/
LinkedIn Engineering	https://engineering.linkedin.com/blog
Monzo Technology	https://monzo.com/blog/technology/
Moonpig Engineering	https://engineering.moonpig.com/
Netflix TechBlog	https://medium.com/netflix-techblog
PayPal Engineering	https://www.paypal-engineering.com/
Pinterest Engineering	https://medium.com/@Pinterest_Engineering
Revolut Engineering	https://blog.revolut.com/tag/engineering/
Salesforce Engineering	https://engineering.salesforce.com/
Slack Engineering	https://slack.engineering/
Target Tech	https://tech.target.com/
Ticketmaster Technology	https://tech.ticketmaster.com/category/devops/
Trainline Engineering	https://engineering.thetrainline.com/
Uber Engineering	https://eng.uber.com/
Vimeo Engineering	https://medium.com/vimeo-engineering-blog
Zapier Engineering	https://zapier.com/engineering/

DevOps Foundation Course: Value Added Resources

GitHub Resources

Item	Link
Ansible for DevOps	https://github.com/geerlingguy/ansible-for-devops
Azure DevOps Samples	https://github.com/microsoft/devops-project-samples
CapitalOne DevOps Dashboard	https://github.com/capitalone/Hygieia
Chaos Monkey	https://github.com/Netflix/SimianArmy/wiki/Chaos-Monkey
Chef Style DevOps Kung Fu	https://github.com/chef/devops-kungfu
DevOps Against Humanity	https://github.com/bridgetkromhout/devops-against-humanity
DevOps Tools Collection	https://github.com/collections/devops-tools
Docker and DevOps	https://github.com/yeasy/docker_practice
TicketMaster Tech Maturity Model	https://github.com/Ticketmaster/techmaturity

Additional Videos of Interest

Title	Link
'Continuous Delivery' with Jez Humble (46:59)	https://youtu.be/skLJuksCRTw
'Continuous Delivery is a Team Sport' with Jez Humble, Gene Kim and Gary Gruver hosted by ElectricCloud (5:33)	https://youtu.be/9XmvFvdTObY
'Get Loose! Microservices & Loosely Coupled Architectures' with Jez Humble and Anders Walgreen hosted by devops.com	https://youtu.be/I9BymWx8G1E
'Intro to Scrum in Under 10 Minutes' by Axosoft	https://youtu.be/XU0llRltyFM
'Learn How Value Stream Mapping Applies to Any Industry or Process'' by Gemba Academy	https://youtu.be/gg5u9kn0Bzo
Sidney Dekker, Richard Cook and Stephen Spear at DOES 2017	https://youtu.be/CFMJ3V4VakA
'The Real DevOps of Silicon Valley' from AppDynamics (4:49)	https://youtu.be/2PjVuTzA2lk

DevOps Foundation Course: Value Added Resources

DevOps Books

Title	Author	Link
Accelerate: The Science of Lean Software and DevOps: Building and Scaling High Performing Technology Organizations	Nicole Forsgren PHD, Jez Humble & Gene Kim	https://itrevolution.com/book/accelerate/
Beyond The Phoenix Project	Gene Kim and Jez Humble	https://itrevolution.com/book/beyond-phoenix-project/
Continuous Delivery	Jez Humble and Dave Farley	https://www.amazon.com/dp/0321601912?tag=contindelive-20
DevOps for the Modern Enterprise	Mirco Hering	https://itrevolution.com/book/devops_modern_enterprise/
Just Culture	Sidney Dekker	http://sidneydekker.com/just-culture/
Leading Change	John P Kotter	https://www.amazon.com/Leading-Change-New-Preface-Author/dp/1422186431/
Lean IT	Steven C Bell and Michael A Orzen	https://www.amazon.com/Lean-Enabling-Sustaining-Your-Transformation/dp/1439817561
From Project to Product	Dr. Mik Kersten	https://itrevolution.com/book/project-to-product/
Site Reliability Engineering	Niall Richard Murphy, Betsy Beyer and Chris Jones	https://www.amazon.com/Site-Reliability-Engineering-Production-Systems/dp/149192912X
Team Topologies	Matthew Skelton and Manuel Pais	https://itrevolution.com/book/team-topologies/
The Art of Business Value	Mark Schwartz	https://itrevolution.com/book/the-art-of-business-value/
The DevOps Handbook	Gene Kim, Jez Humble, Patrick Debois & John Willis	https://itrevolution.com/book/the-devops-handbook/

The Phoenix Project	Kevin Behr, George Spafford and Gene Kim	https://itrevolution.com/book/the-phoenix-project/
The Unicorn Project	Gene Kim	https://itrevolution.com/book/the-unicorn-project/
The Field Guide to Understanding Human Error	Sidney Dekker	https://www.routledge.com/The-Field-Guide-to-Understanding-Human-Error-3rd-Edition/Dekker/p/book/9781472439055

Case Stories Featured in the Course

Company	Module	Link
Alaska Air	4: Business & Technology Frameworks	• 'Delivering the Continuous Enterprise with Agile, Lean, and DevOps ALDO Practices' by Mark Holmes • 'Alaska Airlines DevOps Journey' by Troy Kaser • 'Alaska Airlines Flies on Visual Studio Team Services and Xamarin' (Microsoft Azure) • 'Alaska Airlines Makes Shopping Easier with Faster Flow of New eCommerce Features' by Microsoft Cloud Platform • 'DevOps Practice: Modern Infrastructure Automation' by Derek E. Weeks • 'Mob Programming at Alaska Airlines with Agile Amped at AATC2016' by SolutionsIQ • 'Alaska Airlines' ChefConf 2016 Keynote from Veresh Sita
Australia Post (in notes on SAFe slide)	4: Business & Technology Frameworks	• 'Australia's Post's Agile Approach to Digital Transformation' by Cameron Gough
Capital One	3: Key DevOps Principles	• 'Measuring Success at Capital One' by The Goat Farm • 'Capital One: A DevOps Powerhouse' by Josh Litvin
Disney	8: Sharing, Shadowing & Evolving	• 'Digital Magic: Disney's DevOps Transformation' by Jennifer Riggins • 'Disney's DevOps Journey: A DevOps Enterprise Summit Reprise' by Aliza Earnshaw • 'How Disney Organized for a DevOps Transition' by George Lawton • 'Systems Strategy Chief Jason Cox Details Disney's DevOps Journey' by Tamlin McGee

Fannie-Mae	6: Automation & Architecting DevOps Toolchains	• 'Fannie Mae Securitization App Leads DevOps Implementation' by Beth Pariseau • 'How Fannie Mae Practices DevOps to Deliver Quality at Speed' by Derek Weeks
ING Bank	1: Exploring DevOps	• 'Bank Tech Boss: Where We're Going, We Don't Need Mainframes' by Joe Fay at the Register • 'ING Bangs the Drum for DevOps' (FinExtra) • 'Continuous Delivery - The ING Story: Improving Time to Market with DevOps and Continuous Delivery' by Wouter Mejis
Societe Generale	7: Measurement, Metrics & Reporting	• 'How to Reap the Rewards of DevOps: One Bank's Story' by Gottfriend Sehringer
Target	5: Culture, Behaviors & Operating Models	• '(Re)Building at Engineering Culture: DevOps at Target' with Heather Mickman and Ross Clanton
Ticketmaster	2: Core DevOps Practices	• 'How to Apply DevOps Practices to Legacy IT' (Computer Weekly)

DevOps Foundation v3.3
Egzamin próbny 1

1. Mała grupka osób wróciła niedawno z konferencji, gdzie dowiedziała się o DevOpsie. Nie mogą uzgodnić, jak zacząć. Od czego organizacja IT powinna rozpocząć stosowanie praktyk DevOps?

A. Zrozumienie celu istnienia organizacji
B. Wybranie odpowiednich aplikacji do projektu pilotażowego
C. Opracowanie długoterminowej strategii
D. Zidentyfikowanie potrzebnych narzędzi i szkoleń

2. Czym są Trzy Drogi?

A. Metoda identyfikowania i usuwania ograniczeń
B. Kluczowe zasady DevOps
C. Metodyczne, oparte na danych podejście do ograniczania marnotrawstw
D. Metoda ciągłego doskonalenia

3. Które stwierdzenie o Kanban jest PRAWIDŁOWE?

A. Pcha (push) pracę przez cały proces
B. Wymaga narzędzia do zarządzania przepływem pracy
C. Ciągnie (pull) pracę przez cały proces
D. Umożliwia więcej pracy w toku

4. Czym jest Manifest Programowania Zwinnego?

A. Wartości i zasady kierujące iteracyjnym i zorientowanym na ludzi podejściem do tworzenia oprogramowania
B. Metoda, która koncentruje się na upewnianiu się, że oprogramowanie jest gotowe do wdrożenie przez cały cykl życia
C. Deklaracja korzyści i zamiarów DevOps
D. Intencje i motywy bycia zwinnym przedsiębiorstwem

5. Organizacja stara się walczyć z wyzwaniami swojej tradycyjnej kultury silosów, w której zespoły zostały zorganizowane wokół wiedzy merytorycznej. Na co cierpi ta organizacja?

A. Dług kulturowy
B. Zmęczenie zmianą
C. Zmiana organizacyjna
D. Niski poziom zaufania

6. Które stwierdzenie NAJLEPIEJ opisuje zmęczenie zmianą?

A. Agresywny opór
B. Apatia
C. Wskazywanie palcem i szukanie winnych
D. Wyczerpanie

7. Ze względu na ściśle powiązaną architekturę organizacja nie jest w stanie zwiększyć częstotliwości wydań kluczowej usługi. Kolejne wydania niosą zazwyczaj wiele problemów, w wyniku czego przewaga konkurencyjna organizacji maleje. Jakie podejście do tworzenia oprogramowania można zastosować by poprawić sytuację?

A. Rozwój oparty na testach
B. Kontenery
C. Mikro-serwisy
D. Chaos Monkey

8. Organizacja właśnie zakończyła wdrożenie pilota usługi przy użyciu praktyk DevOps i wstępnego procesu wdrażania. Która z poniższych metryk dostarczy organizacji najwięcej informacji i pomoże w ciągłym doskonaleniu?

A. Średni czas naprawy (MTTR)
B. Czasy realizacji (lead time) zmian i czas cyklu (cycle time) zmian
C. Dzielenie się wiedzą
D. Wszystkie powyższe

9. Które stwierdzenie o zespołach DevOps jest NAJBARDZIEJ trafne?

A. Odpowiadają za ustanowienie praktyk DevOps w całym przedsiębiorstwie
B. Są odpowiedzialne za rozwój procesu wdrażania
C. Powinny to być stałe zespoły współpracujące przy długoterminowych projektach
D. Powinny dzielić się odpowiedzialnością

10. Organizacja niedawno zorganizowała wewnętrzne Dni DevOps. Podczas jednej z otwartych sesji zasugerowano, że istnieje więcej możliwości dla deweloperów, operacji, bezpieczeństwa i pozostałych obszarów IT do interakcji i dzielenia się wiedzą. Które z poniższych działań powinna rozważyć organizacja?

A. Hackatony
B. Symulacje
C. Możliwość immersji
D. Wszystkie z powyższych

11. Które z poniższych ról są interesariuszami DevOps?

A. Testerzy
B. Specjaliści do spraw wsparcia
C. Dostawcy zewnętrzni
D. Wszystkie z powyższych

12. Co NIE jest celem DevOps?

A. Większa produktywność
B. Mniejsza liczba wydań oprogramowania, ale o wyższej jakości
C. Wdrażanie oprogramowania o niższym ryzyku
D. Lepsza jakość kodu

13. Organizacja wdraża aplikację powodującą zakłócenia działania podobną do Simian Army. Którą z Trzech Dróg wprowadzają?

A. Pierwszą drogę
B. Drugą drogę
C. Trzecia drogę
D. Projekt Feniks

14. Organizacja chce poprawić współpracę zespołów w czasie rzeczywistym. Którą praktykę DevOps powinni wziąć pod uwagę?

A. Kanban
B. ChatOps
C. Eskalacje
D. Powiadomienia

15. Co charakteryzuje kulturę DevOps?

A. Skuteczna jedno-kierunkowa komunikacja od góry organizacji do dołu
B. Docenienie sukcesów najlepszych i najbystrzejszych pracowników
C. Wspólna wizja, cele i zachęty
D. Wszystkie z powyższych

16. Który opis NAJLEPIEJ charakteryzuje proces wdrażania?

A. Zautomatyzowana wersja procesu zarządzania zmianami ITSM
B. Zautomatyzowany proces zarządzania zmianami oprogramowania od zatwierdzenia zmian kodu do wydania
C. Zbiór narzędzi umożliwiających ciągłą integrację
D. Sekwencja działań dodających wartość wymaganych do zaprojektowania, zbudowania i dostarczenia produktu

17. Na ostatniej konferencji CIO dowiedział się, że jego organizacja powinna poważnie zainwestować w uczenie maszynowe. Po powrocie do biura poprosił jednego ze swoich kierowników o przygotowanie uzasadnienia biznesowego dla takiej inwestycji. Która z poniższych propozycji NIE jest bezpośrednią korzyścią, którą organizacja mogłaby uzyskać z zastosowania sztucznej inteligencji, a zatem należy ją wykluczyć z inwestycji?

A. Prognozowanie przyszłych scenariuszy zdarzeń
B. Znajdowanie nowych trendów i korelacji
C. Zwiększenie wkładu człowieka i zwiększenie produktywności
D. Budowanie kultury wolnej od obwiniania

18. Które stwierdzenie dotyczące Improvement Kata jest PRAWIDŁOWE?

A. Koncentruje się na krótkoterminowych celach
B. Jest to 7-etapowy proces
C. Uwzględnia długoterminową wizję lub kierunki działania organizacji
D. Metoda powinna być realizowana w miarę dostępnego czasu

19. W kontekście zwinnego tworzenia oprogramowania, które z poniższych działań NIE jest obowiązkiem operacji IT?

A. Zarządzanie backlogiem produktu
B. Definiowanie wymagań niefunkcjonalnych
C. Identyfikacja wymagań bezpieczeństwa
D. Zapewnienie dostępu do infrastruktury

20. Która z poniższych cech charakteryzuje kulturę organizacyjną o wysokim zaufaniu?

A. Dobry przepływ informacji
B. Interdyscyplinarna współpraca
C. Uczenie się na porażkach i nowych pomysłach
D. Wszystkie z powyższych

21. Dlaczego kultura organizacyjna jest kluczowym czynnikiem sukcesu DevOps?

A. Reprezentuje wartości i zachowania, które przyczyniają się do stworzenia wyjątkowego środowiska społecznego i psychologicznego organizacji
B. Reprezentuje podejście dowodzenia i kontroli w zakresie świadczenia usług
C. Reprezentuje sposób, w jaki organizacja jest ustrukturyzowana i zorganizowana
D. Odzwierciedla strategiczny kierunek przywództwa firmy

22. Jaka jest podstawowa zaleta łańcuchów narzędzi DevOps (DevOps toolchains)?

A. Automatyzują kroki w procesie wdrażania
B. Pozwalają śledzić podróż funkcjonalności od pomysłu do wdrożenia
C. Zapewniają, że ich projekt architektoniczny wspiera interoperacyjność i spójność
D. Wszystkie z powyższych

23. Która z poniższych ról DevOps NIE została jeszcze dobrze zdefiniowana?

A. Inżynier systemów
B. Architekt ds. automatyzacji ciągłego dostarczania
C. Inżynier DevOps
D. Ekspert ds. doświadczeń użytkownika

24. Organizacja wdraża DevOps. Deweloperzy obawiają się, że ich procesy ITSM są zbyt złożone, powolne i nie będą wspierać zasad i praktyk DevOps. Która z poniższych metod pomoże organizacji zaszczepić podejście zwinne w istniejących procesach ITSM?

A. ITIL
B. Agile
C. Agile service management
D. Lean

25. Które z narzędzi Lean przedstawia przepływ informacji, materiałów i pracy pomiędzy funkcjonalnymi silosami kładąc nacisk na identyfikację i eliminację marnotrawstw?

A. Improvement Kata
B. Ciągłe dostarczanie
C. Kanban
D. Mapowanie strumienia wartości

26. Biznesowe „dlaczego" z koncepcji Złotego Kręgu reprezentuje…

A. Cel, przyczynę i przekonania organizacji
B. Produkty i usługi organizacji
C. Przewagę konkurencyjną organizacji
D. Rentowność organizacji

27. Co decyduje o tym, które zasady i praktyki DevOps przyniosą NAJLEPSZE korzyści dla organizacji?

A. Strategie i cele biznesowe
B. Zaangażowanie wczesnych użytkowników (early adopters)
C. Dostępność zaawansowanych narzędzi
D. Możliwości i zasoby IT

28. Którą z Trzech Dróg wspiera Teoria Ograniczeń?

A. Pierwszą drogę
B. Drugą drogę
C. Trzecią drogę
D. Wszystkie z powyższych

29. Który z poniższych elementów jest wymagany do ciągłej integracji?

A. Zautomatyzowane testy jednostkowe, integracyjne i akceptacyjne
B. Zautomatyzowane zarządzanie wydaniami
C. Proces ciągłego dostarczania
D. Proces wdrażania

30. Która z praktyk DevOps opiera się na procesie wdrażania, który umożliwia wdrażanie na żądanie?

A. Ciągłe testowanie
B. Ciągła integracja
C. DevSecOps
D. Ciągłe dostarczanie

31. Który z poniższych procesów ITSM ma największe znaczenie dla DevOps?

A. Zarządzanie zmianami organizacyjnymi
B. Zarządzanie ciągłością działania usług
C. Zarządzanie incydentami
D. Wszystkie z powyższych

32. W organizacji stwierdzono kulturę obwiniania i strachu, w której incydenty nie są cenione, a porażki nie są traktowane jako szansa na naukę. Zidentyfikowano wiele pojedynczych punktów awarii (SPOFs), a pracownicy cierpią każdego dnia z powodu kruchości (fragility) systemów i konieczności udziału w bolesnych analizach (war rooms) w trakcie licznych awarii. Które z poniższych podejść powinna przyjąć organizacja, by poprawić sytuację?

A. Kultura bezpieczeństwa
B. Zwinne wytwarzanie oprogramowania
C. Budowanie łańcucha narzędzi DevOps
D. Site Reliability Engineering

33. Kto powinien zostać zaangażowany w planowanie i służyć jako agent zmiany podczas próby wprowadzenia w organizacji poważnej zmiany?

A. Wcześni użytkownicy (early adopters)
B. Konserwatyści i pesymiści
C. Kierownictwo
D. Ludzie potrzebujący dowodów

34. Który z poniższych NIE jest typowym elementem łańcucha narzędzi DevOps (DevOps toolchain)?

A. Narzędzia monitorowania
B. Zautomatyzowane testy
C. Kontrola wersji
D. Systemy zarządzania incydentami w Service Desku

35. Która z poniższych propozycji jest kluczowym czynnikiem sukcesu DevOps?

A. Ustanowienie łańcucha narzędzi
B. Zatrudnienie inżynierów DevOps
C. Zaangażowanie kierownictwa w zmianę kultury
D. Automatyzacja wszystkiego

36. Które z poniższych działań nie jest celem przywództwa DevOps?

A. Pomoc w poprawie auto-diagnozy
B. Kontrola i ocena pracowników za pomocą metryk
C. Wpojenie ludziom konieczności samodoskonalenia
D. Przełożenie lokalnych spostrzeżeń na globalne ulepszenia

37. Która z Trzech Dróg zachęca do wzajemnej oceny (peer review) zmian w kodzie produkcyjnym?

A. Pierwsza droga
B. Druga droga
C. Trzecia droga
D. Wszystkie z powyższych

38. Co oznacza pojęcie „shift left”?

A. Wbudowanie jakości w proces tworzenia oprogramowania poprzez wczesne i ciągłe testowanie
B. Przekazywanie pakietów wersji oprogramowania do operacji IT od razu po zakończeniu partii (batch) pracy programistycznej
C. Wykonywanie losowych testów kodu wgrywanych na serwer ciągłej integracji
D. Przeprowadzanie większej ilości testów produkcyjnych po wdrożeniu

39. Organizacja ma problem z dodatkowym czasem potrzebnym na weryfikację bezpieczeństwa po ukończeniu sprintu. Opóźnienie wpływa na zdolność organizacji do tworzenia wydań. Organizacja chce zwiększyć liczbę testów bezpieczeństwa w ramach podejścia „shift left" w swoich testach. Jakiej praktyki DevOps będzie potrzebować organizacja?

A. ChatOps
B. Ciągłe testowanie
C. DevSecOps
D. Powiadomienia o podatnościach

40. Organizacja przygotowuje się do automatycznego wdrożenia każdej wersji, która przejdzie automatyczne testy jednostkowe, integracyjne, akceptacyjne użytkowników i niefunkcjonalne. Którą z praktyk DevOps zamierza zastosować ta organizacja?

A. Ciągłe dostarczanie
B. Ciągłe testowanie
C. Ciągłe wdrażanie
D. Ciągła integracja

KLUCZ ODPOWIEDZI

Pytanie	Poprawna odpowiedź	Obszar tematyczny
1	A	1: Exploring DevOps
2	B	2: Core DevOps Principles
3	C	3: Key DevOps Practices
4	A	4: DevOps Values: Business & Technology Frameworks
5	A	5: DevOps Values: Culture, Behaviors & Operating Models
6	B	5: DevOps Values: Culture, Behaviors & Operating Models
7	C	6: DevOps Values: Automation & Architecting Toolchains
8	D	7: DevOps Values: Measurements, Metrics & Reporting
9	D	8: DevOps Values: Sharing, Shadowing & Evolving
10	D	7: DevOps Values: Measurements, Metrics & Reporting
11	D	1: Exploring DevOps
12	B	1: Exploring DevOps
13	C	2: Core DevOps Principles
14	B	3: Key DevOps Practices
15	C	5: DevOps Values: Culture, Behaviors & Operating Models
16	B	6: DevOps Values: Automation & Architecting Toolchains

17	**D**	**6: DevOps Values: Automation & Architecting Toolchains**
18	**C**	**4: DevOps Values: Business & Technology Frameworks**
19	**A**	**4: DevOps Values: Business & Technology Frameworks**
20	**D**	**5: DevOps Values: Culture, Behaviors & Operating Models**
21	**A**	**5: DevOps Values: Culture, Behaviors & Operating Models**
22	**D**	**6: DevOps Values: Automation & Architecting Toolchains**
23	**C**	**8: DevOps Values: Sharing, Shadowing & Evolving**
24	**C**	**4: DevOps Values: Business & Technology Frameworks**
25	**D**	**4: DevOps Values: Business & Technology Frameworks**
26	**A**	**1: Exploring DevOps**
27	**A**	**1: Exploring DevOps**
28	**A**	**2: Core DevOps Principles**
29	**A**	**3: Key DevOps Practices**
30	**D**	**3: Key DevOps Practices**
31	**D**	**4: DevOps Values: Business & Technology Frameworks**
32	**A**	**4: DevOps Values: Business & Technology Frameworks**
33	**A**	**5: DevOps Values: Culture, Behaviors & Operating Models**
34	**D**	**6: DevOps Values: Automation & Architecting Toolchains**

35	**C**	**8: DevOps Values: Sharing, Shadowing & Evolving**
36	**B**	**8: DevOps Values: Sharing, Shadowing & Evolving**
37	**B**	**2: Core DevOps Principles**
38	**A**	**3: Key DevOps Practices**
39	**C**	**3: Key DevOps Practices**
40	**C**	**3: Key DevOps Practices**

DevOps Foundation v3.3
Egzamin próbny 2

1. Która z poniższych propozycji jest dobrym przykładem metryki DevOps stosowanej do mierzenia Pierwszej Drogi, czyli przepływu?

A. Wyniki kompilacji kodu/ testu
B. Rejestr hipotez
C. Wskaźnik niepowodzenia zmian
D. Czas cyklu zmian

2. Według raportów Accelerate State of DevOps, organizacje zaliczane do elity...

A. Wdrażaj częściej
B. Mają wyższy wskaźnik niepowodzeń zmian
C. Mają dłuższy MTTR
D. Wydłużają czas realizacji od zatwierdzenia zmian kodu do wdrożenia

3. Które z poniższych stwierdzeń poprawnie opisuje Trzecią Drogę?

A. Zrozumienie i zwiększenie przepływu pracy
B. Tworzenie kultury sprzyjającej eksperymentom
C. Tworzenie krótszych pętli informacji zwrotnej w celu ciągłego doskonalenia
D. Zrozumienie, że powtarzanie czynności nie prowadzi do mistrzowstwa

4. Która z poniższych cech jest metryką DevOps odnoszącą się do stabilności?

A. Zaangażowanie i morale
B. Czas realizacji zmiany i czas cyklu
C. Incydenty i defekty
D. Sredni czas do wykrycia incydentów (MTTD)Średni czas do wykrycia incydentów (MTTD)

5. Które z poniższych wartości są przedstawione w Manifeście Zwinnego Oprogramowania?

A. Procesy i narzędzia ponad ludzi i interakcje
B. Szczegółowa dokumentacja ponad działające oprogramowanie
C. Współpraca z klientem ponad negocjowanie umów
D. Realizacja założonego planu ponad reagowanie na zmiany

6. W jaki sposób DevOps poprawia zwinność?

A. Tworzy więcej silosów
B. Zwiększa ograniczenia
C. Stosuje zasady zwinnego tworzenia oprogramowania zarówno do Dev, jak i Ops
D. Wdraża szybciej z większą liczbą błędów

7. Które z poniższych działań NIE jest częścią Improvement Kata?

A. Zaplanowanie ostatnich kroków
B. Ustalenie obecnego stanu
C. Zastosowanie PDCA do osiągnięcia kolejnego stanu docelowego
D. Zrozumienie długoterminowego kierunku

8. Szef Ewy wrócił właśnie z konferencji nt. Agile i DevOps. Poprosił Ewę o poprowadzenie programu transformacji DevOps i rozpoczęcie od utworzenia zespołu DevOps. Dlaczego Ewa powinna zachować ostrożność realizując to zadanie?

A. Istnieje ryzyko, że zespół może stać się kolejnym silosem
B. Ten zespół może promować DevOps w całej organizacji
C. Ludzie zrozumieją, że DevOps to zadanie dla wszystkich osób
D. Daje to Ewie możliwość zapewnienia współdzielenia odpowiedzialności

9. Adam buduje nowy, autonomiczny zespół z wieloma kompetencjami, który będzie zajmował się produktem z przewidywanym długim cyklem życia. Wyciąga członków zespołu z różnych działów, w których dotychczas odpowiadali za różne procesy. To będzie pierwsze wspólne działanie nowego zespołu. W jaki sposób zespół może zacząć ze sobą współdziałać wizualizując cały cykl życia produktu?

A. Przeprowadzić hakaton z ludźmi z różnych działów
B. Stworzyć forum dyskusyjne dla klientów
C. Użyć ChatOps do monitorowania wydajności produktu
D. Wykonać ćwiczenie mapowania strumienia wartości

10. Który z poniższych mierników koncentruje się przede wszystkim na stabilności?

A. Czas realizacji zmiany
B. Wskaźnik powodzenia wdrożeń
C. Średni czas do przywrócenia działania systemu
D. Częstotliwość wdrażania

11. Anna wraz ze swoim zespołem produktowym zakończyła ćwiczenie mapowania strumienia wartości. Udało się im zidentyfikować szereg ograniczeń w procesie. Jednym z nich jest zdolność zespołu bezpieczeństwa do szybkiego reagowania na ich wnioski. Które z poniższych zagadnień pomoże Annie i jej zespołowi znaleźć praktyki, które złagodzą wspomniane ograniczenie?

A. Kanban
B. Site Reliability Engineering
C. Chaos engineering
D. DevSecOps

12. Który z poniższych czynników NIE jest pozytywnie skorelowany z wydajnością organizacji?

A. Rozwój oprogramowania oparty o główne repozytorium
B. Złożony proces zarządzania zmianami
C. Luźno powiązana architektura
D. Korzystanie z chmury obliczeniowej

13. Jaki jest cel Pierwszej Drogi?

A. Zwiększenie przepływu pracy
B. Pozwolenie na przekazanie znanych błędów w dół procesu wytwórczego
C. Pozwolenie by lokalna optymalizacja spowodowała globalną degradację
D. Zrozumienie i dodanie ograniczeń

14. Zespół Anny składa się ze zdalnych pracowników zatrudnionych przez firmę, dla której pracuje oraz przez organizację partnerską z Indii. Rzadko zdarza się, aby więcej niż dwie osoby z jej zespołu znajdowały się jednocześnie w tym samym miejscu. Ostatnio zespół Anny miał problemy ze stabilnością systemu, które wymagały dodatkowego wsparcia od jeszcze innego zespołu zajmującego się infrastrukturą. Wspólnie stwierdzili, że coraz trudniej jest im współpracować w formie telekonferencji, ponieważ nie wiedzą co inni robią i muszą czekać, aby dowiedzieć się, jaki wpływ na ich systemy mają wprowadzone zmiany. Co może im pomóc lepiej zarządzać incydentami?

A. Narzędzia do zarządzania wydajnością aplikacji
B. ChatOps
C. Eskalacje
D. Jenkins

15. W czym może pomóc automatyzacja w DevOps?

A. Szybszy czas realizacji
B. Mniej chaotyczne wydania oprogramowania
C. Szybsze przywracanie system po awarii
D. Wszystkie z powyższych

16. Które z poniższych stwierdzeń jest prawdziwe w przypadku łańcuchów narzędzi DevOps?

A. Narzędzia muszą pochodzić od tego samego dostawcy
B. Są zbudowane wyłącznie w oparciu o systemy komercyjne
C. Nie wymagają projektu architektonicznego, aby zapewnić interoperacyjność
D. Narzędzia powinny być połączone, zwykle za pośrednictwem API

17. Sophia jest dyrektorem zarządzającym w firmie doradczej. Jest rozczarowana tym, że jej konsultanci wydają się mniej zainteresowani marką i celem jej organizacji niż markami i celami firm, dla których świadczą usługi doradcze. Zaprosiła więc swoich konsultantów na oficjalną kolację, aby o tym porozmawiać, ale większość odmówiła, powołując się na zobowiązania rodzinne lub problemy z dotarciem na miejsce. Sophia nie lubi organizować spotkań w godzinach pracy, ponieważ nie chce zabierać konsultantom płatnych godzin. Co Sophia tworzy w swojej organizacji poprzez takie zachowanie?

A. Dług techniczny
B. Dług kulturowy
C. Wysoki poziom zaufania
D. Kulturę ścisłej współpracy

18. Która z poniższych cech jest charakterystyczna dla kultury DevOps?

A. Zorientowanie na wykonywanie zadań
B. Zadowolenie
C. Odporność
D. Wysoki poziom zaufania

19. David odkrywa, że ilekroć spotyka się z Robertem, kłócą się o to, co należy zrobić dla poprawy pracy w ich zespole. Wie, że oboje chcą tego, co najlepsze dla ich zespołu, ale widzi też, że napięcia między nimi wpływają na pozostałych członków zespołu do tego stopnia, że przestają angażować się w rozmowy na temat poprawy sposobu ich wspólnej pracy. Jakiej metody mógłby użyć David by zrozumieć, jak lepiej pracować z Robertem?

A. Test na konflikty Thomasa-Kilmanna
B. Trzy Drogi
C. Krzywa zmian Kübler-Rossa
D. Tablica Kanban

20. Co sprawia, że kultura DevOps jest teraz tak ważna?

A. Duże przedsiębiorstwa mają konkurencję w postaci zwinnych startupów
B. Konsumenci mają mentalność „aplikacyjną" i adekwatne wymagania
C. Czas na realizację wartości (time to value) musi przyspieszyć
D. Wszystkie z powyższych

21. Które z poniższych stwierdzeń jest prawdziwe w odniesieniu do teorii ograniczeń?

A. Każdy proces ma co najmniej jedno ograniczenie
B. Wydajność procesu może przekroczyć wydajność jego ograniczeń
C. Proces może być bardziej skuteczny niż jego najsłabsze ogniwo
D. Poprawa ograniczeń jest jedynym sposobem na poprawę procesu

22. Która z poniższych odpowiedzi NIE jest typowym ograniczeniem?

A. Luźno powiązana architektura
B. Audyty bezpieczeństwa
C. Testowanie konfiguracji i uruchamianie aplikacji
D. Tworzenie środowiska aplikacyjnego

23. Która z poniższych propozycji jest przykładem pętli informacji zwrotnej?

A. Dashbordy
B. Rotacyjne dyżury operacyjne
C. Logi produkcyjne
D. Wszystkie z powyższych

24. W trakcie mapowania strumienia wartości, Sandra i jej zespół stwierdzili, że ich proces wdrażania zmian z wieloma radami ds. zmian, zakłóca i spowalnia przepływ pracy. Członkowie zespołu przeczytali też najnowszy raport o stanie DevOps i zauważyli, że złożone procesy zmian negatywnie wpływają na wydajność organizacji. Co mogłoby pomóc zespołowi Sandry w zbudowaniu lżejszego procesu zarządzania zmianami?

A. ITIL
B. Agile
C. Agile service management
D. Lean

25. Dlaczego mniej rzeczy psuje się na produkcji, kiedy stosujesz podejście „shift left"?

A. Robienie wszystkiego z góry zmniejsza ilości pracy do wykonania później
B. Szczegółowe planowanie daje pewność, że pomyśleliśmy o wszystkim
C. Problemy są wykrywane i rozwiązywane wcześniej
D. Stwierdzenie jest nieprawdziwe, zamiast tego powinniśmy stosować podejście „shift right"

26. Które z poniższych jest przykładem marnotrawstwa nazywanego „nadmiernym transportem"?

A. Awarie i znane błędy
B. Częste przekazywanie zadań pomiędzy zespołami, wiele e-maili i spotkań
C. Nieużywane oprogramowanie lub infrastruktura
D. Nadmiernie skomplikowana inżynieria

27. Co jest prawdą w zmienianiu kultury organizacji?

A. Nie możesz zmienić ludzi; tylko oni sami mogą się zmienić
B. Nie musisz angażować interesariuszy
C. To nie kosztuje tyle, ile myślisz, że będzie kosztować
D. Ludzie akceptują zmiany, nawet jeśli w nich nie uczestniczą

28. Manuel czytał o DevOpsie i uważa, że to podejście może zmienić na lepsze sposób pracy w jego organizacji. Zaczął rozmawiać o tym z ludźmi i znalazł kilka zainteresowanych osób. Myśli o przygotowaniu program wdrożenia i nauki. Dlaczego powinien to zrobić?

A. Może przyciągnąć uwagę CEO
B. Prawdopodobnie pojawią się inni innowatorzy i wcześni użytkownicy (early adopters)
C. Nie powinien się tym zbytnio przejmować – i tak nikt nie będzie zainteresowany
D. Jeśli do programu dołączy tzw. późna większość, będzie wiedział, że DevOps już działa

29. Lider transformacji...

A. Akceptuje status quo
B. Krytykuje zespół
C. Dowodzi i strofuje
D. Rozumie kierunek działania organizacji

30. Co powinniśmy mierzyć, gdy stosujemy zasady i praktyki DevOps, w celu poprawy wydajności organizacji?

A. Dojrzałość
B. Indywidualną wydajność pracowników
C. Produktywność
D. Wartość

31. Co powinieneś zrobić poprawiając automatyzację?

A. Zautomatyzować wszystkie procesy, takie jakie są
B. Zacząć od architektury
C. Zbudować swój zestaw narzędzi i trzymać się go
D. Nie przejmować się monitorowaniem

32. Jakie są dobre sposoby na wzmocnienie nowych zachowań?

A. Hackatony
B. Dzielenie się historiami i pomysłami w stylu mediów społecznościowych
C. Społeczności praktyków
D. Wszystkie z powyższych

33. Która z poniższych cech NIE pasuje do kultury bezpieczeństwa?

A. Analiza post-mortem bez obwiniania się
B. Docenianie wartości incydentów na produkcji
C. Akceptacja istnienia SPOFs
D. Andon

34. Co sprawia, że Kanban jest użyteczny?

A. Pozwala na nieograniczoną ilość pracy w toku
B. Pcha pracę do zespołów
C. Maksymalizuje marnotrawstwa i czas bezczynności
D. Sprawia, że praca jest widoczna

35. Kiedy pracujesz nad poprawą stabilności, stosując zasady i praktyki DevOps, co musisz poświęcić?

A. Szybkość
B. Jakość
C. Nic nie poświęcasz
D. Swoich pracowników

36. DevOps to przede wszystkim...

A. Ruch kulturowy
B. Automatyzacja absolutnie wszystkiego
C. Rozszerzenie podejścia Agile
D. Koncept prosty do zrozumienia i wykonania

37. Która z poniższych propozycji nie jest jedną z czterech kluczowych metryk w DevOps?

A. Częstotliwość wdrażania
B. Czas realizacji od zatwierdzenia zmian w kodzie do testów
C. Czas przywrócenia działania systemu po incydencie
D. Wskaźnik niepowodzenia zmian

38. Liam stosuje zwinne praktyki, aby usprawnić przepływ pracy swojego zespołu. Zbliżył też do siebie osoby zajmujące się programowaniem oraz operacje IT. Korzystając z kombinacji możliwości ciągłego dostarczania i monitorowania, stworzył szybkie pętle informacji zwrotnej od klientów do swojego zespołu. Teraz chce przyspieszyć proces wdrażania innowacji. Którą z Trzech Dróg powinien się zainteresować?

A. Pierwszą drogą
B. Drugą drogą
C. Trzecią drogą
D. Wszystkimi z powyższych

39. Ciągłe dostarczanie…

A. Zapewnia szybką i automatyczną informację zwrotną na temat gotowości systemu do wydania
B. Daje pierwszeństwo pracy nad nowymi funkcjonalnościami przed utrzymaniem gotowości oprogramowania do wydania/ wdrożenia
C. Opiera się na procesie wdrażania, który automatycznie wdraża na żądanie
D. Zwiększa koszt, czas i ryzyko wprowadzania zmian przyrostowych

40. John od jakiegoś czasu zmienia swoją organizację zgodnie z podejściem DevOps. Obecnie skupia się na konsolidacji zysków, aby móc wprowadzać więcej zmian. Czego NIE powinien teraz robić?

A. Komunikować sukcesu
B. Milczeć o awariach i niepowodzeniach
C. Ciągle inwestować w edukację
D. Udostępniać re-używalne artefakty

KLUCZ ODPOWIEDZI

Pytanie	Poprawna odpowiedź	Obszar tematyczny
1	D	7: DevOps Values: Measurements, Metrics & Reporting
2	A	1: Exploring DevOps
3	B	2: Core DevOps Principles
4	C	2: Core DevOps Principles
5	C	4: DevOps Values: Business & Technology Frameworks
6	C	4: DevOps Values: Business & Technology Frameworks
7	A	4: DevOps Values: Business & Technology Frameworks
8	A	8: DevOps Values: Sharing, Shadowing & Evolving
9	D	4: DevOps Values: Business & Technology Frameworks
10	C	7: DevOps Values: Measurements, Metrics & Reporting
11	D	3: Key DevOps Practices
12	B	3: Key DevOps Practices
13	A	2: Core DevOps Principles
14	B	3: Key DevOps Practices
15	D	6: DevOps Values: Automation & Architecting Toolchains
16	D	6: DevOps Values: Automation & Architecting Toolchains

17	**B**	**5: DevOps Values: Culture, Behaviors & Operating Models**
18	**D**	**5: DevOps Values: Culture, Behaviors & Operating Models**
19	**C**	**5: DevOps Values: Culture, Behaviors & Operating Models**
20	**D**	**1: Exploring DevOps**
21	**A**	**2: Core DevOps Principles**
22	**A**	**2: Core DevOps Principles**
23	**D**	**2: Core DevOps Principles**
24	**C**	**4: DevOps Values: Business & Technology Frameworks**
25	**C**	**3: Key DevOps Practices**
26	**B**	**4: DevOps Values: Business & Technology Frameworks**
27	**A**	**5: DevOps Values: Culture, Behaviors & Operating Models**
28	**B**	**5: DevOps Values: Culture, Behaviors & Operating Models**
29	**D**	**8: DevOps Values: Sharing, Shadowing & Evolving**
30	**D**	**7: DevOps Values: Measurements, Metrics & Reporting**
31	**B**	**6: DevOps Values: Automation & Architecting Toolchains**
32	**D**	**5: DevOps Values: Culture, Behaviors & Operating Models**
33	**C**	**4: DevOps Values: Business & Technology Frameworks**
34	**D**	**3: Key DevOps Practices**

35	**C**	**1: Exploring DevOps**
36	**A**	**1: Exploring DevOps**
37	**B**	**1: Exploring DevOps**
38	**C**	**2: Core DevOps Principles**
39	**A**	**3: Key DevOps Practices**
40	**B**	**8: DevOps Values: Sharing, Shadowing & Evolving**

Your Path to

DevOps Success

DevOps Institute is dedicated to advancing the human elements of DevOps success. Our goal is to help advance careers and support emerging practices using a role-based approach to certification which focuses on the most modern competencies and hireable skills required by today's organizations adopting DevOps.

Take the next steps in your learning and certification journey to DevOps success.

Click on a certification or visit www.devopsinstitute.com/certifications to learn more.

Become a Member

Join the fastest growing global community of DevOps practitioners and professionals and gain access to invaluable learning content, the latest news, events, emerging practices, develop your network and advance your career.

You belong.

www.devopsinstitute.com/become-a-community-member