Congratulations! You passed!

Grade received 100% | Latest Submission Grade 100% | To pass 66% or higher

Go to next item

1. Which component of a DevOps pipeline consists exclusively of the Plan, Code, Build, and Test phases? **1 / 1 point**

- Continuous Delivery
- Continuous Alerting
- Continuous Integration
- Continuous Monitoring

Correct
Correct. Continuous Integration (CI) consists exclusively of the Plan, Code, Build, and Test phases of a DevOps pipeline.

2. Which tool is a Continuous Integration and Continuous Delivery (CI/CD) platform that performs CD deployments and contains workflow definitions inside a YAML file? **1 / 1 point**

- Marathon
- OpsGenie
- Jenkins
- CircleCI

Correct
Correct. Circle CI is a CI/CD platform that performs CD deployments. With Circle CI, you define workflows inside a file named circle.yaml.

3. What is one benefit of Infrastructure as Code (IaC)? **1 / 1 point**

- Quicker time to production
- Lower requirements for code readability
- More concise scripts
- Fewer developers to hire

Correct
Correct. IaC automation dramatically speeds up the infrastructure provisioning for development, testing, and production. It can even automate the provisioning of legacy infrastructure, which time-consuming processes like pulling a ticket might otherwise govern.

4. Why does Continuous Integration (CI) lead to higher-quality code? **1 / 1 point**

- Coordinated coding and deployment
- Extensive sessions for developing
- Superior tools for code validation
- Constant review and testing

Correct
Correct. With CI, you and your team members are constantly reviewing and testing each other's code changes, such as when going through a pull request.

5. What is social coding? **1 / 1 point**

- An iterative, sequential approach to content creation
- A software development approach to project management
- A test-first approach to application design
- An open source approach to enterprise code

Correct
Correct. You can refer to social coding as "open source for inner source." In short, social coding brings open source concepts to internal, proprietary code development.

6. Which Git command should you use to undo a commit to your local repository but keep the changes in the staging area? **1 / 1 point**

- git clean --quiet
- git diff --no-patch
- git reset --soft
- git pull --edit

Correct
Correct. You can use the git reset --soft command to undo a commit to your local repository but keep the changes in the staging area. Once you have made any necessary fixes, you can commit those changes again.

7. Assume you are working on a Git project and want to develop a new feature. After pulling all the repository's latest code to your local workspace, what should you do next? **1 / 1 point**

- Commit your changes.
- Reset recent edits.
- Create a new branch.
- Open a pull request.

Correct
Correct. When working on a new feature, you should always create a new branch to work from. Then, you can begin developing the feature.

8. What does an event do in a GitHub Actions workflow? **1 / 1 point**

- Runs a workflow when prompted
- Performs a single low-level task
- Execute steps within a runner
- Triggers a workflow run

Correct
Correct. An event is a repository activity that determines that a workflow should run.

9. In GitHub Actions, which keyword is helpful for using the output of one step as the input for another step? **1 / 1 point**

- needs
- env
- name
- id

Correct
Correct. The id keyword is helpful when you want to use the output of one step as an input or parameter for another step.

10. What is a benefit of Continuous Delivery (CD)? **1 / 1 point**

- It corrects code errors instantly and more precisely than other approaches.
- It automates software's movement through the development lifecycle.
- It reduces the drift between the feature and main branches.
- It communicates all project updates to clients for you.

Correct
Correct. A benefit of CD is that it automates the steps that transport software through the stages of the software development lifecycle (SDLC).

11. According to the key principles of Continuous Delivery (CD), what should developers spend their time on? **1 / 1 point**

- Assembling cost estimates
- Talking with clients
- Performing repetitive tasks
- Solving coding problems

Correct
Correct. A principle of CD is to have people solve problems, not perform repetitive tasks. CI/CD tools handle repetitive tasks like testing more efficiently than people, so letting the tools handle those jobs allows developers to focus elsewhere.

12. What is a best practice for Continuous Delivery (CD)? **1 / 1 point**

- Build in the staging environment.
- Develop at each team member's pace.
- Release at the granularity of test.
- Comment in the source code.

Correct
Correct. A best practice of CD is to release at the granularity of test. For example, if you must test two parts of a system together, you should release them together to ensure their compatibility. Alternatively, you can fully decouple them.

13. In Tekton, what does TriggerBinding do? **1 / 1 point**

- It details a blueprint for the pipeline that should run when an event occurs.
- It captures the required event parameters for running the pipeline.
- It specifies what happens when the EventListener detects an event.
- It declares a storage class for the Persistent Volume in the pipeline.

Correct
Correct. The TriggerBinding Custom Resource Definition (CRD) captures the required event parameters for running the pipeline.

14. In a Tekton task manifest, what is one way to pass a parameter to a task? **1 / 1 point**

- List the parameter name as a step argument.
- Describe the parameter within the pipeline specifications.
- Include the parameter in the 'kubectl apply' command.
- Add the parameter as a command line option.

Correct
Correct. To pass a parameter to a task, define the parameter within the spec field and above the steps subfield that will define the corresponding step. Next, list the parameter's name as an argument in that step's field.

15. Assume you have four Tekton tasks within a pipeline: task-a, task-b, task-c, and task-d. How could you make task-a and task-b run in parallel? **1 / 1 point**

- Include the runAfter field in both tasks, and then specify task-c as the value for both runAfter fields.
- Include the from field in task-c and task-d, and then specify task-a as the value for both from fields.
- Include the runAfter field in both tasks, and then specify task-a as the value for both runAfter fields.
- Include the from field in task-b, and then specify task-a as the value in this field.

Correct
Correct. To make task-a and task-b run in parallel, you include the runAfter field in both tasks. Next, you specify the same value for both runAfter fields, which must be the name of a task, such as task-c. Now task-a and task-b will run immediately after this specified task completes, ensuring they run in parallel.