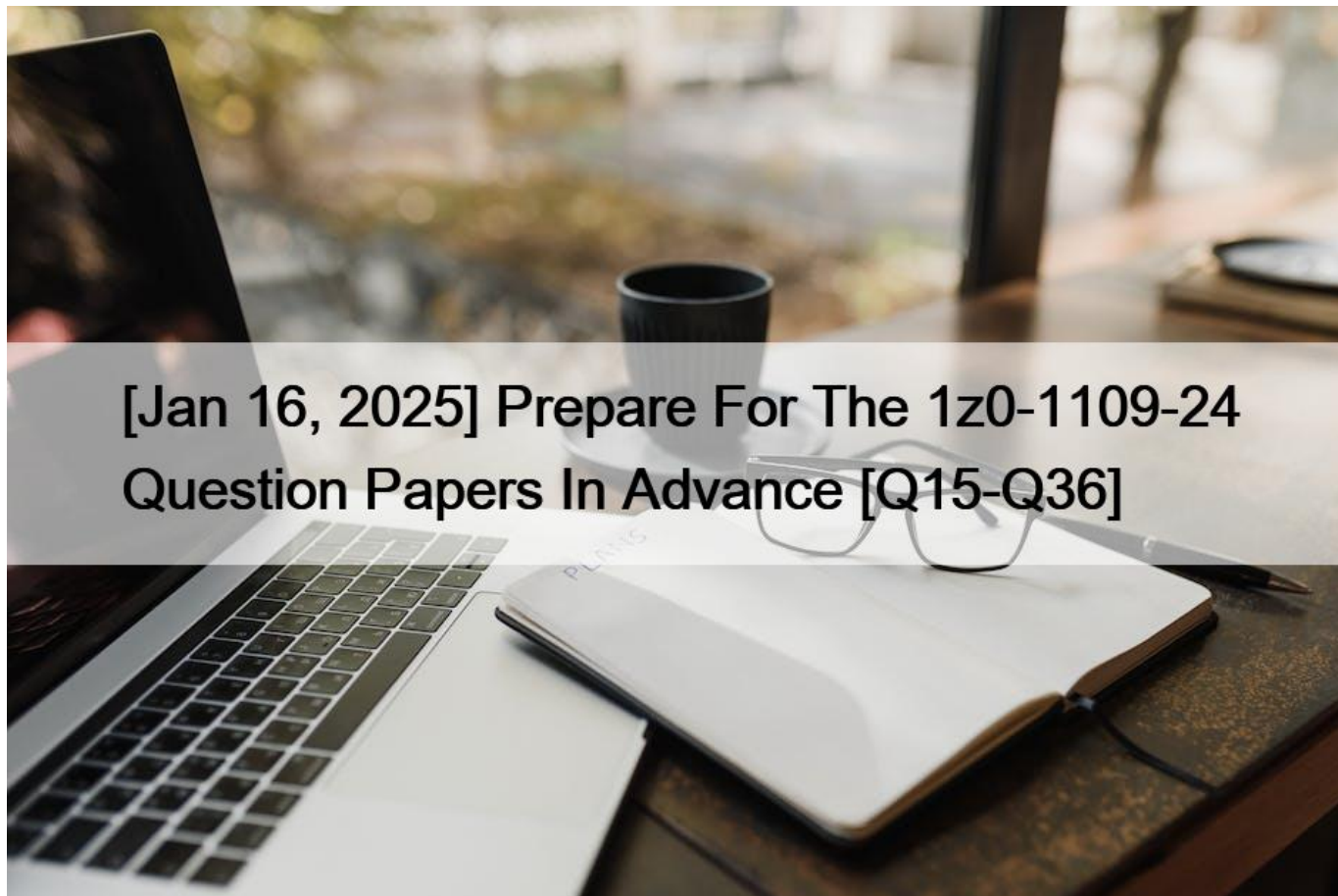


## [Jan 16, 2025 Prepare For The 1z0-1109-24 Question Papers In Advance [Q15-Q36]



[Jan 16, 2025] Prepare For The 1z0-1109-24 Question Papers In Advance  
1z0-1109-24 PDF Dumps Real 2025 Recently Updated Questions

### Oracle 1z0-1109-24 Exam Syllabus Topics:

**Topic 1- Implementing Monitoring and Observability (O&M):** This section targets Oracle Cloud Infrastructure DevOps engineers and developers and focuses on implementing monitoring and observability practices within a DevOps framework. Candidates will learn about tools and techniques for tracking application performance, analyzing logs, and managing events to ensure system reliability.  
**Topic 2- Managing Containers Using Container Orchestration Engine:** This section covers the management of containers using orchestration tools like Kubernetes. Candidates will gain insights into creating, scaling, and optimizing containerized applications within a cloud environment.  
**Topic 3- Configuring and Managing Continuous Integration and Continuous Delivery (CI- CD):** This domain measures the skills of DevOps Engineers by focusing on the configuration and management of CI- CD pipelines. Candidates will learn to automate the software development lifecycle, enabling faster release cycles through continuous integration and delivery practices.  
**Topic 4- Using Code and Templates for Provisioning and Configuring Infrastructure:** This section targets DevOps Engineers and emphasizes the importance of using code and templates for infrastructure provisioning. Candidates will explore Infrastructure as Code (IaC) practices that allow for automated configuration and management of infrastructure resources.  
**Topic 5- Understand DevOps Principles and Effectively Work with Containerization Services:** This domain measures the skills of DevOps Professionals and focuses on

the foundational principles of DevOps and the role of containerization in modern software development. Candidates will learn how containerization enables packaging applications and their dependencies into isolated environments, promoting consistency across different deployment stages.

### NEW QUESTION 15

Which of the following statement is INCORRECT with respect to a Dockerfile?

- \* WORKDIR instruction sets the working directory for any RUN, CMD, ENTRYPOINT instructions and not for COPY and ADD instructions in the Dockerfile.
  - \* An ENV instruction sets the environment value to the key, and it is available for the subsequent build steps and in the running container as well.
  - \* The RUN instruction will execute any commands in a new layer on top of the current image and commit the results.
  - \* If CMD instruction provides default arguments for the ENTRYPOINT instruction, both should be specified in JSON format.
- The WORKDIR instruction sets the working directory for all subsequent RUN, CMD, ENTRYPOINT, COPY, and ADD instructions in the Dockerfile. This means that after specifying WORKDIR, all these instructions will use the specified directory as their current working directory.

### NEW QUESTION 16

Your team is working on a project to deploy a microservices-based application on a cloud platform using Terraform. Each microservice has specific configurations and dependencies, and you want to ensure modularity, reusability, and consistency across deployments.

Which Terraform feature would you use to achieve these objectives efficiently?

- \* Terraform Providers
- \* Terraform Workspaces
- \* Terraform Variables
- \* Terraform Modules

Terraform Modules are used to organize and group related configuration resources into reusable components. By using modules, you can achieve modularity, reusability, and consistency across different deployments, making it easier to manage complex infrastructure setups.

For a microservices-based application, where each microservice has specific configurations and dependencies, modules allow you to define the infrastructure for each microservice in a modular way. This helps to maintain clean, reusable code and ensures consistency across deployments.

### NEW QUESTION 17

You are a DevOps engineer working on a project that requires you to push and pull Docker images to and from Oracle Cloud Infrastructure Registry (Container Registry) using Docker CLI. You have been given access to Container Registry and have installed Docker CLI on your local machine.

Which should you create and use to securely authenticate and store your Docker image in a private Docker registry in OCI?

- \* Auth Token
- \* JSON Web Token
- \* SSH Key Pair
- \* Master Encryption Key in OCI Vault

To authenticate with the Oracle Cloud Infrastructure Registry (Container Registry) when using the Docker CLI, you need to use an

Auth Token. The Auth Token is created in the OCI console and acts as a password for the Docker login command, providing secure access to the container registry.

### NEW QUESTION 18

As a DevOps engineer working on an OCI project, you're setting up a deployment pipeline to automate your application deployments.

Which statement is false about deployment pipeline in OCI DevOps?

- \* Using deployment pipeline, you can deploy helm charts in OCI Function.
- \* You can add a Wait stage that adds a specified duration of delay in the pipeline.
- \* You can add a Traffic Shift stage that routes the traffic between two environments.
- \* You can add an Approval stage that pauses the deployment for a specified duration for manual decision from the approver.

Helm charts are used to manage Kubernetes deployments, not OCI Functions. Helm charts are deployed to Kubernetes clusters, such as OCI Container Engine for Kubernetes (OKE), to manage containerized applications. OCI Functions are serverless and do not use Helm charts for deployment.

### NEW QUESTION 19

A small company is moving to a DevOps framework to better accommodate their intermittent workloads, which are dynamic and irregular. They want to adopt a consumption-based pricing model.

Which Oracle Cloud Infrastructure service can be used as a target deployment environment?

- \* Virtual machine compute instance
- \* Oracle Kubernetes (OKE)
- \* Bare metal compute instance
- \* Functions

Oracle Cloud Infrastructure Functions is a serverless compute service that supports a consumption-based pricing model. This means that you are only charged for the compute resources when your function is invoked. This is ideal for intermittent, dynamic, and irregular workloads since the company does not need to provision infrastructure in advance, and costs are directly tied to usage.

### NEW QUESTION 20

As a DevOps engineer at XYZ Corp, you have been assigned the task of setting up a new OKE (Oracle Kubernetes Engine) cluster to manage the organization's Kubernetes applications hosted on Oracle Cloud Infrastructure (OCI). Your goal is to ensure a smooth and efficient process while preparing for the cluster creation.

Which of the following statements is false regarding the preparation process for setting up a new OKE cluster?

- \* Container Engine for Kubernetes cannot utilize existing network resources for the creation of the new cluster.
- \* Container Engine for Kubernetes automatically creates and configures new network resources for the new cluster.
- \* It is necessary to ensure sufficient quota on different resource types in your OCI tenancy for the cluster setup.
- \* Access to an Oracle Cloud Infrastructure tenancy is required to set up the new OKE cluster.

This statement is false because Container Engine for Kubernetes (OKE) can utilize existing network resources such as Virtual Cloud Networks (VCNs), subnets, security lists, and route tables for the creation of a new cluster. You can either use pre-existing network resources or let OKE create new network resources automatically.

### NEW QUESTION 21

How can system administrators ensure that only signed images from Oracle Cloud Infrastructure Registry are deployed to a Container Engine for Kubernetes cluster?

- \* By disabling access to the Container Engine for Kubernetes cluster
- \* By manually inspecting each image before deployment
- \* By encrypting the images using a custom encryption algorithm
- \* By configuring an image verification policy for the cluster

Image verification policies are used to ensure that only trusted and signed images are deployed to an Oracle Kubernetes Engine (OKE) cluster. By configuring such policies, administrators can enforce that images must be signed and come from trusted sources, such as the Oracle Cloud Infrastructure Registry.

## NEW QUESTION 22

You are using the Oracle Cloud Infrastructure (OCI) DevOps service and you have successfully built and tested your software applications in your Build Pipeline. The resulting output needs to be stored in a container repository.

Which stage should you add next to your Build Pipeline?

- \* Trigger deployment
- \* Managed build
- \* Deliver artifacts
- \* Export packages

Step 1: Understanding the Requirement

The objective is to store the resulting build output from a Build Pipeline in a container repository. In OCI DevOps, the build output is stored as an artifact, which can include Docker images or other build-generated files. To store these artifacts in a container repository, you need to explicitly deliver artifacts in the pipeline.

Step 2: Explanation of the Options

A . Trigger deployment

This stage is used to trigger a deployment pipeline, which comes after the artifacts are already stored and prepared for deployment.

Not applicable: This stage is downstream of storing artifacts and is used for deploying software, not for saving the build output to a repository.

B . Managed build

The managed build stage is where you compile, test, and package the application. This has already been completed successfully according to the question.

Not applicable: The question specifies that the build has been completed, so this stage is not relevant at this point.

C . Deliver artifacts

The Deliver Artifacts stage in OCI DevOps pipelines is designed to store the output of the build process in an artifact repository, such as:

OCI Container Registry (OCIR) for Docker images.

Artifact Registry for build artifacts like binaries or JAR files.

Applicable and Correct answer: This is the correct next step for storing the resulting D . Export packages This is not a standard OCI

DevOps pipeline stage. It may be relevant in other contexts but is not related to OCI DevOps for storing build artifacts.

Step 3: Key Concepts of [Deliver Artifacts](#); in OCI DevOps

Purpose: Save build outputs (artifacts) to an artifact repository.

Artifact Types: Includes Docker container images, binaries, JAR files, or other build outputs.

Repositories Supported:

OCI Container Registry (OCIR)

OCI Artifact Registry

Configuration:

Specify the artifact source (build stage output).

Define the destination repository (e.g., OCIR).

Step 4: References and OCI Resources

OCI DevOps Build Pipelines:

Build Pipeline Documentation

Deliver Artifacts Stage

OCI Container Registry (OCIR):

OCI Container Registry Overview

OCI Artifact Registry:

OCI Artifact Registry Overview

## **NEW QUESTION 23**

As a DevOps engineer working on a CI/CD pipeline for your company's application, you have completed code analysis, image scanning, and automated testing.

What is the next step to ensure a secure and reliable deployment?

- \* Add a traffic Shift stage to route the traffic between two sets of backend IPs.
- \* Add an invoke function stage to run code or custom logic in a serverless manner.
- \* Add a shell stage to run custom commands in the deployment pipeline.
- \* Add an approval stage to pause the deployment for a specified duration for manual decision from the approver.

After completing code analysis, image scanning, and automated testing, the next step in the CI/CD pipeline should include a manual review to ensure that all necessary security and quality checks have been performed correctly. Adding an approval stage helps ensure that a secure and reliable deployment is achieved by requiring human verification and approval before proceeding with the deployment to production.

This step adds an extra layer of control to prevent unintended issues from moving forward without further review. It is a common practice in CI/CD pipelines to have an approval step, especially for critical deployments.

#### NEW QUESTION 24

Which two are prerequisites for creating a secret in Oracle Cloud Infrastructure Vault service? (Choose two.)

- \* You must first create a hash digest of the secret value.
- \* You must have the required permissions to create and manage secrets in the Vault service.
- \* You must have a Vault managed key to encrypt the secret.
- \* You must have an auth token to encrypt the secret.
- \* The user must create a compute instance to run the secret service.

You need the required permissions (such as policies allowing secret management) to create and manage secrets in Oracle Cloud Infrastructure (OCI) Vault service. These permissions are essential for performing operations such as creating, reading, and managing secrets.

Vault managed key is required to encrypt the secret before it is stored in the OCI Vault. The managed key acts as the encryption key for securing the secret, ensuring its confidentiality.

#### NEW QUESTION 25

As a DevOps Engineer, you are tasked with explaining the key concepts of Terraform to a new team member. You want to ensure they understand the fundamental concepts of Terraform.

Which of the following best describes the purpose of Terraform variables?

- \* Terraform variables are used to manage the life cycle of Terraform resources.
- \* Terraform variables are used to define input values for Terraform configurations, allowing for customization and reuse of infrastructure code.
- \* Terraform variables are used to output the final state of the infrastructure after deployment.
- \* Terraform variables are used to define the structure and organization of Terraform configuration files.

Terraform variables are used to define input values for Terraform configurations. They allow users to customize infrastructure deployments by providing different values without modifying the configuration files themselves. Variables help in creating reusable infrastructure code, making it easy to maintain and adjust the infrastructure setup according to different environments or needs.

#### NEW QUESTION 26

Which command creates the docker registry secret required in the application manifests for OKE to pull images from Oracle Cloud Infrastructure Registry?

- \* 

```
kubectl create secret docker-registry <secret-name> --docker-server=  
<region-key>.ocir.io --docker-username=<oci-username> --docker-  
password='<oci-auth-token>' --docker-email=<email-address>
```
- \* 

```
kubectl create passwd docker-registry <secret-name> --docker-server=  
<region-key>.oke.io --docker-username=<tenancy-namespace>/<oci-  
username> --docker-password='<oci-auth-token>' --docker-email=<email-  
address>
```

- \* `kubectl create passwd docker-registry <secret-name> --docker-server=<region-key>.oke.io --docker-username=<oci-username> --docker-password='<oci-auth-token>' --docker-email=<email-address>`
  
- \* `kubectl create secret docker-registry <secret-name> --docker-server=<region-key>.ocir.io --docker-username=<tenancy-namespace>/<oci-username> --docker-password='<oci-auth-token>' --docker-email=<email-address>`

To create a Docker registry secret to pull images from the Oracle Cloud Infrastructure Registry (OCIR), you need to specify the correct parameters such as the region key, namespace, OCI username, and OCI authentication token.

Chosen command is correct because:

The `kubectl create secret docker-registry` command creates a Docker registry secret.

The `--docker-server=<region-key>.ocir.io` specifies the correct endpoint for OCIR.

The `--docker-username=<tenancy-namespace>/<oci-username>` provides both the tenancy namespace and the OCI username, which is the required format for authentication with OCIR.

The `--docker-password='<oci-auth-token>'` specifies the OCI auth token, which acts as a password for authentication.

The `--docker-email=<email-address>` is also included.

The other commands have errors, such as missing tenancy namespace or using incorrect flags (`passwd` instead of `secret`).

## NEW QUESTION 27

How can you scale a deployment named `nodejs-deployment` to have two replicas?

- \* `kubectl set replicas deployment nodejs-deployment --replicas=2`
- \* `kubectl resize deployment nodejs-deployment --replicas=2`
- \* `kubectl adjust deployment nodejs-deployment --replicas=2`
- \* `kubectl scale deployment nodejs-deployment --replicas=2`

The `kubectl scale` command is used to scale the number of replicas in a deployment. By specifying the `--replicas` flag, you define the desired number of replicas for the deployment.

(`kubectl set replicas`) is not the correct syntax for scaling a deployment.

(`kubectl resize`) is not a valid command for scaling a deployment.

(`kubectl adjust`) is also not a valid Kubernetes command.

## NEW QUESTION 28

As a cloud engineer, you are responsible for managing a Kubernetes cluster on the Oracle Cloud Infrastructure (OCI) platform for your organization. You are looking for ways to ensure reliable operations of Kubernetes at scale while minimizing the operational overhead of managing the worker node infrastructure.

Which cluster option is the best fit for your requirement?

- \* Using OCI OKE managed nodes with cluster autoscalers to eliminate worker node infrastructure management
- \* Using OCI OKE virtual nodes to eliminate worker node infrastructure management
- \* Using Kubernetes cluster add-ons to automate worker node management
- \* Creating and managing worker nodes using OCI compute instances

Step 1: Understanding the Requirement

The goal is to ensure reliable operations of Kubernetes at scale while minimizing the operational overhead of managing worker node infrastructure. In this context, a solution is needed that abstracts away the complexity of managing, scaling, and maintaining worker nodes.

Step 2: Explanation of the Options

A . Using OCI OKE managed nodes with cluster autoscalers

While this option provides managed node pools and uses cluster autoscalers to adjust resources based on demand, it still requires some level of management for the underlying worker nodes (e.g., patching, upgrading, monitoring).

Operational overhead: Moderate.

B . Using OCI OKE virtual nodes

Virtual nodes in OCI OKE are a serverless option for running Kubernetes pods. They remove the need to manage underlying worker nodes entirely.

OCI provisions resources dynamically, allowing scaling based purely on pod demand.

There's no need for node management, patching, or infrastructure planning, which perfectly aligns with the requirement to minimize operational overhead.

Operational overhead: Minimal.

Best Fit for This Scenario: Since the requirement emphasizes minimizing operational overhead, this is the ideal solution.

C . Using Kubernetes cluster add-ons to automate worker node management  
Kubernetes add-ons like Cluster Autoscaler or Node Problem Detector help in automating some aspects of worker node management. However, this still requires managing worker node infrastructure at the core level.

Operational overhead: Moderate to high.

D . Creating and managing worker nodes using OCI compute instances

This involves manually provisioning and managing compute instances for worker nodes, including scaling, patching, and troubleshooting.

Operational overhead: High.



Not Suitable for the Requirement: This option contradicts the goal of minimizing operational overhead.

### Step 3: Why Virtual Nodes Are the Best Fit

#### Virtual Nodes in OCI OKE:

Virtual nodes provide serverless compute for Kubernetes pods, allowing users to run workloads without provisioning or managing worker node infrastructure.

Scaling: Pods are automatically scheduled, and the required infrastructure is dynamically provisioned behind the scenes.

Cost Efficiency: You only pay for the resources consumed by the running workloads.

Use Case Alignment: Eliminating the burden of worker node infrastructure management while ensuring Kubernetes reliability at scale.

### Step 4: References and OCI Resources

#### OCI Documentation:

OCI Kubernetes Virtual Nodes

OCI Container Engine for Kubernetes Overview

Best Practices for Kubernetes on OCI:

Best Practices for OCI Kubernetes Clusters

### NEW QUESTION 29

Your team is responsible for deploying a new version of an application that is being used by your company's finance department. The application is critical to the department's operations, and any downtime could have serious consequences.

What is the recommended approach in OCI for creating environments for this scenario?

- \* Deploy the application to two separate OCI tenancies to ensure complete isolation between environments.
- \* Use a single Kubernetes cluster with two node pools, one for the blue-green environment and one for the canary environment.
- \* Configure two OKE clusters, selecting the blue-green traffic shift strategy using a load balancer.
- \* Use a single OCI region and create two separate Virtual Cloud Networks (VCNs), one for the blue environment and one for the green environment.

For critical applications, such as the one used by the finance department, a blue-green deployment strategy is recommended to ensure minimal or zero downtime during upgrades. The blue-green strategy involves running two separate environments: blue (current version) and green (new version).

### NEW QUESTION 30

As a DevOps engineer working on managing clusters on the OCI platform for your organization, which statement is true about managing cluster add-ons in OCI OKE Cluster?

- \* When creating a new cluster, essential cluster add-ons cannot be disabled.
- \* When enabling a cluster add-on, you cannot configure the add-on by specifying one or more key/value pairs to pass as arguments

to the cluster add-on.

- \* When creating a new cluster, essential cluster add-ons are set to manually update.
- \* When you disable a cluster add-on using the console, the add-on is completely removed from the cluster.

Essential cluster add-ons are required for the basic functioning of the Kubernetes cluster and cannot be disabled during cluster creation. These add-ons provide necessary features such as core DNS, networking, and other critical functionalities for the cluster's operation.

### NEW QUESTION 31

You're using Oracle Cloud Infrastructure (OCI) DevOps to automate your application deployment for frequent releases. In one of your automation steps, you'll create a deployment pipeline.

What does this deployment pipeline do in OCI DevOps?

- \* It takes a commit ID from your source code repositories and uses that source code to run your build instructions.
- \* It is a sequence of steps for delivering and deploying your artifacts to a target environment.
- \* It is used to store, manage, develop source code with OCI DevOps Code Repositories.
- \* It is a set of stages for your build process for building, testing and compiling software artifacts.

A deployment pipeline in OCI DevOps is used to automate the deployment of application artifacts to a target environment. It is a sequence of stages that includes steps such as approvals, traffic shifts, manual interventions, and the actual deployment of the artifacts to environments like Kubernetes clusters or compute instances.

### NEW QUESTION 32

Which statement is false about OCI Resource Manager (RM)?

- \* Resources provisioned through RM cannot be destroyed from outside of RM.
- \* RM can render custom Application Information pages for stacks.
- \* RM can generate Terraform based on the resources in a compartment.
- \* RM can mirror repositories from GitHub and GitLab.

Resources provisioned through OCI Resource Manager (RM) can still be modified or destroyed from outside of RM, such as using the OCI Console, CLI, or other APIs. RM manages the lifecycle of resources created by its Terraform configurations, but it does not prevent other tools or methods from modifying or deleting those resources.

### NEW QUESTION 33

As a DevOps engineer working on containerizing a microservices-based application to be hosted on OCI Cloud platforms, which step can help ensure that the container images have not been modified after being pushed to Oracle Cloud Infrastructure Registry (OCIR)?

- \* Scanning the image upon ingestion and comparing the image size for changes
- \* Enabling scanning of container images stored in OCI Registry
- \* Deploying a manifest to the Kubernetes cluster that references the container image and its unique hash
- \* Signing the image using the Container Registry CLI and creating an image signature that associates the image with the master encryption key and key version in the Vault service

To ensure that container images have not been modified after being pushed to the Oracle Cloud Infrastructure Registry (OCIR), you should sign the image. This involves using the Container Registry CLI to create a digital signature for the image, which associates the image with a master encryption key and key version stored in the OCI Vault service. This signature can then be verified at the time of deployment, ensuring that the image has not been tampered with since it was signed.

### NEW QUESTION 34

As a DevOps Engineer, you are tasked with securely storing and versioning your application's source code and

automatically build, test, and deploy your application to Oracle Cloud Infrastructure (OCI) platform.

You are told to automate manual tasks and help software teams in managing complex environments at scale.

Which three OCI services can you choose to accomplish these tasks? (Choose three.)

- \* Oracle Cloud Infrastructure Registry
- \* DevOps project
- \* Oracle Cloud Logging Analytics
- \* Container Engine for Kubernetes
- \* Oracle APEX Application Development

Oracle Cloud Infrastructure Registry: This service allows you to securely store container images. It is essential for managing the container images used for deployment, making it an important part of the DevOps workflow.

DevOps project: OCI DevOps project is specifically designed to manage the CI/CD pipeline. It helps in automating tasks like building, testing, and deploying applications, which are key activities for managing complex environments and promoting agility in software development.

Container Engine for Kubernetes: Oracle Container Engine for Kubernetes (OKE) is used to deploy applications in a containerized environment. It provides a robust platform for deploying, managing, and scaling containerized applications, which is essential for handling complex environments at scale.

### NEW QUESTION 35

As a DevOps engineer at XYZ Corp, you are responsible for ensuring the smooth operation of high-traffic web applications hosted on Oracle Cloud Infrastructure (OCI). The web applications run on multiple OCI resources, including virtual machines, load balancers, and databases. Recently, users have reported failures while accessing one of the OCI-based web applications, and you suspect HTTP 5XX errors on the load balancer. You need to quickly identify and address this issue.

Which of the following statements can assist you in quickly identifying and monitoring the HTTP 5XX error rate on the load balancer and setting up notifications?

- \* Use Custom Metrics of the Monitoring service to collect HTTP 5XX error rates from the load balancer and set up Service Connectors with third-party services such as PagerDuty or Slack.
- \* Use Metrics and Alarms of the Monitoring service with Container Engine for Kubernetes (OKE) to monitor HTTP 5XX errors on Kubernetes resources and correlate them with other OCI resources.
- \* Use Event Rules to detect HTTP 5XX errors on the load balancer and trigger automated actions using OCI Functions or API Gateway.
- \* Use Metrics and Alarms of the Monitoring service to monitor the HTTP 5XX error rate on the load balancer and set up notifications with OCI Notifications.

The Monitoring service in OCI can be used to track metrics for various OCI resources, including load balancers. You can monitor specific metrics, such as HTTP 5XX error rates, to identify issues.

By using Alarms, you can set up thresholds for the HTTP 5XX error rate and receive notifications when the threshold is breached. The notifications can be configured through OCI Notifications, which allows integration with email, PagerDuty, Slack, and other channels.

**1z0-1109-24 Dumps and Practice Test (52 Exam Questions):** [https://www.dumpleader.com/1z0-1109-24\\_exam.html](https://www.dumpleader.com/1z0-1109-24_exam.html)