

## Mar-2024 Oracle 1z0-1109-23 Actual Questions and Braindumps [Q34-Q56]



## Mar-2024 Oracle 1z0-1109-23 Actual Questions and Braindumps [Q34-Q56]

Mar-2024 Oracle 1z0-1109-23 Actual Questions and Braindumps  
1z0-1109-23 Dumps To Pass Oracle Exam in 24 Hours - DumpsMaterials

### NEW QUESTION 34

Your on-premises private cloud environment consists of virtual machines hosting a set of application servers.

These VMs are currently monitored using a 3rd party monitoring tool for resource metrics such as CPU and Memory utilization and Disk-Write IOPS. You have created a DevOps Project that includes a workflow to migrate these application servers into Oracle Cloud Infrastructure (OCI) compute instances with a few requirements: &#8211; Ensure continuous monitoring is enabled, so the current monitored resource metrics are continuously collected and reported. &#8211; Monitor the end-to-end deployment pipeline during the migration workflow and notify with email on each execution. &#8211; Notify with email for any new OCI Object Storage buckets created after the migration workflow. What should be your recommended solution to achieve these requirements?

- \* Configure both 3rd party monitoring tool and OCI Compute Agent on OCI compute instances to collect required resource metrics. Use OCI Events service (com.oraclecloud.devopsdeploy.createdeployment) with Notifications service to track and notify all changes occurring in the target OCI environment.
- \* Configure OCI Compute agent on on-premises VMs and OCI compute instances to collect required resource metrics. Use OCI

Events service to track the end-to-end deployment process (com.oraclecloud.devopsdeploy.createdeployment) and creation of new bucket (com.oraclecloud.objectstorage.createbucket). Use OCI Notifications and Events services to notify these changes.

\* Configure OCI Compute agent on on-premises VMs to collect required resource metrics. Use OCI Events service to track all deployments (com.oraclecloud.devopsdeploy.createdeployment) with OCI!

Notifications service to track and report all changes occurring in the target environment.

\* Configure OCI Compute agent on OCI compute instances to collect required resource metrics. Use OCI Events and Functions services to track the end-to-end deployment pipeline (com.oraclecloud.devopsdeploy.createdeployment) and creation of new buckets (com.oraclecloud.objectstorage.createbucket). Use OCI Notifications and Events services to notify these changes.

Explanation

The recommended solution to achieve the requirements mentioned would be: Configure OCI Compute agent on OCI compute instances to collect the required resource metrics. Use OCI Events and Functions services to track the end-to-end deployment pipeline (com.oraclecloud.devopsdeploy.createdeployment) and the creation of new OCI Object Storage buckets (com.oraclecloud.objectstorage.createbucket). Finally, utilize OCI Notifications and Events services to notify these changes.

Explanation: Continuous monitoring with resource metrics: Install and configure the OCI Compute agent on the OCI compute instances to collect the required resource metrics such as CPU utilization, memory utilization, and disk IOPS. This ensures continuous monitoring of the VMs as they are migrated to OCI. Monitoring the deployment pipeline: Utilize the OCI Events service to track the end-to-end deployment pipeline. Specifically, use the event type

com.oraclecloud.devopsdeploy.createdeployment; to monitor the deployment process. This allows you to track the progress and status of the migration workflow. Notification for new OCI Object Storage buckets:

Leverage the OCI Notifications service in conjunction with the OCI Events service. Set up a notification rule to trigger an email notification whenever a new OCI Object Storage bucket is created. Use the event type

com.oraclecloud.objectstorage.createbucket; to identify the creation of new buckets. By combining the OCI Compute agent, OCI Events service, Functions service, and Notifications service, you can ensure continuous monitoring of resource metrics, track the deployment pipeline, and receive email notifications for any new OCI Object Storage bucket creations during the migration workflow.

### NEW QUESTION 35

Why is it important to extract output artifacts from the Oracle Cloud Infrastructure (OCI) DevOps build pipeline and store them in an Artifact Registry repository?

\* Output artifacts aren't permanent. If they are to be used in the Deliver Artifacts stage, they need to be exported as output artifacts to a registry.

\* Storing build artifacts in registries helps the deployment pipeline differentiate output artifacts created by the build pipeline from artifacts copied from a Git repository.

\* Deliver Artifacts is a required stage of the build pipeline, and the entire pipeline won't work if it is not included in order to extract artifacts after the Managed Build stage.

\* All artifacts are permanently stored in the build pipeline. Extracting just the ones required for deployment tells the deployment pipeline which artifacts to use.

Explanation

This is because output artifacts are temporary files generated by the build process that are needed to deploy an application. Since these artifacts are not permanent, they need to be extracted from the build pipeline and stored in an Artifact Registry repository for easy distribution, versioning, and management. The Deliver Artifacts stage in the build pipeline is responsible for this task, which ensures that the correct artifacts are used for each deployment. Here is the reference link for more information on Oracle Cloud Infrastructure (OCI) DevOps build pipeline and Artifact Registry: Reference:

<https://docs.oracle.com/en-us/iaas/developer-tutorials/tutorials/devops/01oci-devops-overview-contents.html#art>

### NEW QUESTION 36

Which of the following statements is TRUE with regard to OCI DevOps ? (Choose the best answer.)

- \* OCI DevOps is an orchestration tool for deployments
- \* OCI DevOps uses pipelines to manage infrastructure
- \* OCI DevOps automates SDLC which is CI/CD platform for developers
- \* OCI DevOps is cloud based to build softwares

Explanation

The statement that is TRUE with regard to OCI DevOps is: OCI DevOps automates the Software Development Life Cycle (SDLC) and serves as a Continuous Integration/Continuous Deployment (CI/CD) platform for developers. OCI DevOps provides capabilities to automate and streamline the entire software development process, including building, testing, deploying, and managing applications and infrastructure. It enables developers to implement CI/CD practices and deliver software more efficiently and reliably.

### NEW QUESTION 37

company uses Oracle Cloud Infrastructure (OCI) DevOps to deploy an application to their production server.

They need to make some modifications to their application code and push those changes to production automatically. How can they achieve this?

- \* OCI DevOps Triggers feature can be used to automate deployment.
- \* Application code can be pushed to the Resource Manager Stack for automatic deployment.
- \* Terraform code can be packaged and pushed to the OCI Code Repository to deploy the changes.
- \* Manual builds can be run from the Build Pipelines to deploy the changes.

Explanation

The company can use the OCI DevOps Triggers feature to automate deployment of their application code changes to the production server. Therefore, the correct answer is: OCI DevOps Triggers feature can be used to automate deployment. OCI DevOps Triggers allow for automatic builds and deployments based on changes to the code repository. When a new commit is pushed to the repository, the trigger can initiate a build pipeline that creates an artifact and deploys the new version of the application to the production server. Here is the link to the official documentation on using triggers in OCI DevOps to automate application deployment:

Reference: <https://docs.cloud.oracle.com/en-us/iaas/devops/using/using-triggers.htm>

### NEW QUESTION 38

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 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
- \* Scanning the image upon ingestion and comparing the image size for changes
- \* Enabling scanning of container images stored in OCI Registry

Explanation

The step that can help ensure that the container images have not been modified after being pushed to OCI Registry is 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. Image signing is a process of adding a digital signature to an image to verify its authenticity and integrity. You can use OCI Registry CLI to sign an image using a Vault managed key and create an image signature that contains information such as the image name, tag, digest, key OCID, key version OCID, etc. You can also use OCI Registry CLI to verify an image signature before pulling or running an image. Verified References: [Image Signing &#8211; Oracle Cloud Infrastructure Registry], [Signing Images &#8211; Oracle Cloud Infrastructure Registry]

### NEW QUESTION 39

A DevOps team is deploying a new version of their application to their production environment use Canary deployment strategy in the OCI DevOps service. They want to ensure that the production environment is not affected by any potential issues caused by the new version. Which statement is true in order to achieve this goal?

- \* The Invoke Function stage is an optional stage that can be used to validate the new version before moving to the production environment.
- \* The Canary deployment strategy only supports pipeline redeployment for OKE and not for instance group deployments.
- \* The Production stage in the Canary deployment strategy deploys the new version to the production environment without any manual approval.
- \* The Shift Traffic stage in the Canary deployment strategy shifts 100% of the production traffic to the Canary environment.

### NEW QUESTION 40

As a DevOps engineer working on a project to deploy container images to Oracle Cloud Infrastructure Container Registry (OCIR), you have the option to create an empty repository in advance or allow the system to create a repository automatically on first push. Which statement about automatic repository creation is true?

- \* If you select the `Create repositories on first push root compartment` option and push an image with a command that includes the name of a repository that doesn't already exist, a new private repository is created automatically in the root compartment.
- \* Automatic repository creation is triggered by running the command `docker push .ocir.oci/:`, even if the repository doesn't exist.
- \* Automatic repository creation only works for repositories in the normal user compartment.
- \* To create a new public repository in the root compartment automatically, you need not belong to the `tenancy-administrators` group or have the `REPOSITORY MANAGE` permission on the tenancy.

Explanation

The statement that is true about automatic repository creation is that if you select the `Create repositories on first push root compartment` option and push an image with a command that includes the name of a repository that doesn't already exist, a new private repository is created automatically in the root compartment. This option allows you to enable or disable automatic repository creation for the root compartment of your tenancy.

If you enable this option, you can push an image to OCI Registry using the `docker push` command with the format `<region-key>.ocir.io/<tenancy-namespace>/<repository-name>:<tag>`, where `<repository-name>` is the name of a repository that does not exist yet. This will create a new private repository with the specified name and tag in the root compartment. If you disable this option, you will need to create an empty repository in advance before pushing an image to it. Verified References: [Creating Repositories &#8211; Oracle Cloud Infrastructure Registry], [Pushing Images &#8211; Oracle Cloud Infrastructure Registry]

### NEW QUESTION 41

You host your application on a stack in Oracle Cloud Infrastructure (OCI) Resource Manager. Due to recent growth in your user base, you decide to add a CIDR block to your VCN, add a subnet, and provision a compute instance in it. Which statement is true?

- \* You need to provision a new stack because Terraform uses immutable infrastructure.
- \* You can provision the new resources in the OCI console and add them to the stack with Drift Detection.
- \* You cannot provision the new resources in the OCI console first, then later add them to the Terraform configuration and state.
- \* You can make the changes to the Terraform code, run an Apply job, and Resource Manager will provision the new resources.

Explanation

The correct statement is: You need to provision a new stack because Terraform uses immutable in-frastructure.

In Oracle Cloud Infrastructure (OCI) Resource Manager, Terraform uses the concept of immutable infrastructure, which means that any changes to the infrastructure are managed through the Terraform code. In this scenario, if you want to add a CIDR block, subnet, and compute instance to your VCN, you would need to make the necessary changes to your Terraform code, create a new stack in Resource Manager, and deploy the updated code. This ensures that the infra-structure is created consistently and according to the desired state defined in the Terraform code. Simply provisioning the new resources in the OCI console and later adding them to the Terraform configuration and state would not be the recommended approach in this case.

### NEW QUESTION 42

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?

- \* You can add an Approval stage that pauses the deployment for a specified duration for manual decision from the approver
- \* Using deployment pipeline, you can deploy artifacts to Kubernetes cluster, Instance Group, and OCI Compute Instances.
- \* You can add a Traffic Shift stage that routes the traffic between two sets of backend IPs using preconfigured load balancer and listener.
- \* You can add a Wait stage that adds a specified duration of delay in the pipeline.

Explanation

The statement that is false about deployment pipeline in OCI DevOps is that you can add a Wait stage that adds a specified duration of delay in the pipeline. This is not a valid type of stage that you can add to your deployment pipeline. The types of stages that you can add to your deployment pipeline are:

- \* Deploy stage: A stage that deploys an artifact to a target environment, such as Kubernetes, Instance Group, or Compute Instance.
- \* Control stage approval: A stage that pauses the pipeline execution and requires manual approval before proceeding to the next stage.
- \* Traffic shift stage: A stage that routes the traffic between two sets of backend IPs using a preconfigured load balancer and listener.
- \* Invoke function stage: A stage that invokes an Oracle Function with specified parameters and payload.

Verified References: [Deployment Pipelines &#8211; Oracle Cloud Infrastructure DevOps], [Creating Deployment Pipelines &#8211; Oracle Cloud Infrastructure DevOps]

### NEW QUESTION 43

Which two statements are INCORRECT with respect to a Dockerfile? (Choose two.)

- \* 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.
- \* WORKDIR instruction sets the working directory for any RUN, CMD, ENTRY-POINT instructions and not for COPY and ADD instructions in the Dockerfile.

- \* If CMD instruction provides default arguments for the ENTRYPOINT instruction, both should be specified in JSON format.
- \* The COPY instruction copies new files, directories, or remote file URLs from <src> and adds them to the filesystem of the image at the path <dest>.

Explanation

The WORKDIR command is used to define the working directory of a Docker container at any given time.

The command is specified in the Dockerfile. Any RUN , CMD , ADD , COPY , or ENTRYPOINT command will be executed in the specified working directory. Reference:

<https://www.geeksforgeeks.org/difference-between-the-copy-and-add-commands-in-a-dockerfile/>

#### NEW QUESTION 44

Which is a proper rule to follow when creating container repositories inside the Oracle Cloud Infrastructure (OCI) Registry?

- \* When naming a container repository, you may use capital letters but not hyphens. For example, you may use BGdevops-storefront, but not bgdevops/storefront.
- \* When creating a container repository, check the Immutable Artifacts box, as it keeps other developers from altering the files.
- \* You must use a separate container repository for each image, but multiple versions of that image can be in a single repository.
- \* You must use the OCI DevOps Managed Build stage to define artifacts in the artifact and container repositories and map the build pipeline outputs to them.

Explanation

The proper rule to follow when creating container repositories inside the Oracle Cloud Infrastructure (OCI) Registry is: You must use a separate container repository for each image, but multiple versions of that image can be in a single repository. This means that each distinct image should have its own repository, but different versions of the same image can be stored within that repository. This allows for better organization and management of container images. The other options mentioned are not correct: Checking the `Immutable Artifacts` box does not exist as a requirement when creating a container repository. Immutable artifacts refer to the immutability of the container images themselves, not a setting in the repository. There are no restrictions on using capital letters or hyphens in the naming of container repositories. Both capital letters and hyphens are allowed in the repository name. The OCI DevOps Managed Build stage is not directly related to defining artifacts in the artifact and container repositories. The Managed Build stage is responsible for building and packaging application artifacts, but it does not define the repositories themselves.

#### NEW QUESTION 45

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?

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

Explanation

The OCI service that can be used as a target deployment environment for intermittent workloads with a consumption-based pricing model is Functions. Functions is a fully managed, serverless platform that allows you to run your code without provisioning or managing any servers. You can use Functions to develop and deploy isolated web applications or RESTful APIs using Node.js, Python, Java, or Go. You only pay for the resources you consume when your code is executed, which is ideal for dynamic and irregular workloads.

Verified References: [Functions &#8211; Oracle Cloud Infrastructure Developer Tools], [Creating Applications and Functions &#8211; Oracle Cloud Infrastructure Developer Tools]

#### NEW QUESTION 46

XYZ Inc. is using Oracle Cloud Infrastructure (OCI) DevOps Project to deploy their e-commerce application to production. They recently received a customer request to add a new feature to the application, which requires modification of the existing code. How can XYZ Inc. use OCI services to automatically push the modified code changes to the production?

- \* Use the OCI DevOps Triggers feature to automate build and deployment on every code commit.
- \* Manual builds can be run from the OCI DevOps Build Pipelines to deploy the changes.
- \* Use OCI Ansible modules to automate the deployment of the new changes to the production environment.
- \* Use the OCI Resource Manager to automatically apply the changes to the production environment after successful testing.

Explanation

To automatically push the modified code changes to the production, you can use the OCI DevOps Triggers feature. A trigger is a rule that defines when a build or deployment pipeline should run based on an event, such as a code commit, a pull request, or a schedule. You can create a trigger that runs your build and deployment pipelines on every code commit to your Git repository, which will ensure that your production environment is always up to date with the latest changes. Verified References: [Triggers &#8211; Oracle Cloud Infrastructure DevOps], [Creating Triggers &#8211; Oracle Cloud Infrastructure DevOps]

#### NEW QUESTION 47

How does the Oracle Cloud Infrastructure Container Engine for Kubernetes (OKE) Cluster Autoscaler determine when to create new nodes for an OKE cluster?

- \* When the resource requests from pods exceed a configured threshold.
- \* When the rate of requests to the application crosses a configured threshold.
- \* When the custom metrics from the services exceed a configured threshold.
- \* When the CPU or memory utilization crosses a configured threshold.

Explanation

The Kubernetes Cluster Autoscaler increases or decreases the size of a node pool automatically based on resource requests, rather than on resource utilization of nodes in the node pool. Reference:

<https://docs.oracle.com/en-us/iaas/Content/ContEng/Tasks/contengusingclusterautoscaler.htm>

#### NEW QUESTION 48

You are a developer who has made a mistake when adding variables to your build\_spec.yaml file. This mistake resulted in a failed build pipeline. Which is a possible error you could have made?

- \* exported a vaultVariable by creating another variable to export, then transferred the values over during a build stage
- \* used vaultVariable to hold the content of the vault secrets in OCID format
- \* defined parameters such as \$(VARIABLE\_NAME) that you later assigned in the Parameters tab when you ran the build pipeline
- \* assumed a non-exported variable would be persistent across multiple stages of a build pipeline.

Explanation

The possible error you could have made when adding variables to your build\_spec.yaml file that resulted in a failed build pipeline is assuming that a non-exported variable would be persistent across multiple stages of the build pipeline. In a build pipeline, variables need to be properly exported and managed to ensure their availability and persistence across different stages. If you mistakenly assumed that a non-exported variable would persist across stages, it could lead to issues where the variable is not available or its value is not maintained as expected, causing the build pipeline to fail.

### NEW QUESTION 49

You are a DevOps project administrator. You are creating Oracle Cloud Infrastructure (OCI) Identity and Access Management (IAM) policies that will be used in a DevOps CI/CD pipeline for deployment to an Oracle Container Engine for Kubernetes (OKE) environment. Which OCI IAM policy can be used?

- \* Allow group <deployment pipeline> to manage devops-family in compartment <com-partment name>
- \* Allow group <build pipeline> to manage all-resources in compartment <compartment name>
- \* Allow dynamic-group <code repository> to manage devops-family in compartment <compartment name>
- \* Allow dynamic-group <deployment pipeline to manage all-resources in compartment <compartment name>

Explanation

To create an OCI IAM policy that will be used in a DevOps CI/CD pipeline for deployment to an OKE environment, you need to use a dynamic group and grant it the permission to manage all-resources in the target compartment. A dynamic group is a group of OCI resources that match a set of rules defined by the administrator. You can use a dynamic group to assign IAM policies to resources such as build pipelines and deployment pipelines. By granting the dynamic group the permission to manage all-resources, you allow it to perform any action on any resource type in the compartment, including OKE clusters, node pools, and Kubernetes resources. Verified References: [Dynamic Groups &#8211; Oracle Cloud Infrastructure Identity and Access Management], [Creating Dynamic Groups &#8211; Oracle Cloud Infrastructure Identity and Access Management]

### NEW QUESTION 50

You have a stack in Oracle Cloud Infrastructure (OCI) Resource Manager that is co-managed by multiple teams. Which statement is true?

- \* The resources in the stack can still be edited or destroyed through the OCI console, causing Resource Manager's state to be out of sync.
- \* The resources in the stack can no longer be edited or destroyed through the Terraform CLI on a local machine.
- \* Resources provisioned by Resource Manager can only be managed through Resource Manager, preventing the state from becoming out of sync.
- \* The Terraform state may become corrupted if multiple people attempt Apply jobs in Resource Manager simultaneously.

Explanation

The correct statement is: Resources provisioned by Resource Manager can only be managed through Resource Manager, preventing the state from becoming out of sync. When a stack is co-managed by multiple teams in Oracle Cloud Infrastructure (OCI) Resource Manager, the resources provisioned by Resource Manager can only be managed through Resource Manager itself. This ensures that the state of the stack remains in sync and prevents conflicts that may arise from multiple teams making changes simultaneously. Managing the resources through Resource Manager helps maintain control and consistency over the stack deployment and configuration. Reference:

<https://docs.oracle.com/en-us/iaas/Content/ResourceManager/Concepts/resource-manager-and-terraform.htm>

### NEW QUESTION 51

As a small company that wants to adopt a DevOps framework and a consumption-based pricing model, which Oracle Cloud Infrastructure service can be used as a target deployment environment, providing features like automated rollouts and rollbacks, self-healing of failed containers, and configuration management, without the overhead of managing security patches and scaling?

- \* OCI Container Engine for Kubernetes (OKE) with managed nodes
- \* Compute Instance Group
- \* OCI Container Instances
- \* OCI Serverless Functions



\* OCI Container Engine for Kubernetes (OKE) with virtual nodes

Explanation

The OCI service that can be used as a target deployment environment for adopting a DevOps framework and a consumption-based pricing model, while providing features like automated rollouts and rollbacks, self-healing of failed containers, and configuration management, without the overhead of managing security patches and scaling, is OCI Container Engine for Kubernetes (OKE) with virtual nodes. OKE is a fully managed service that allows you to run and manage your containerized applications on OCI using Kubernetes, an open-source system for automating deployment, scaling, and management of containerized applications. OKE provides features such as automated rollouts and rollbacks, self-healing of failed containers, configuration management, service discovery, load balancing, etc. OKE also supports virtual nodes, which are serverless compute resources that are automatically provisioned and scaled by OCI based on your application workload demands.

Virtual nodes eliminate the need for managing worker node infrastructure, such as security patches, updates, scaling, etc. Virtual nodes also offer a consumption-based pricing model, where you only pay for the resources you consume when your containers are running. Verified References: [Container Engine for Kubernetes &#8211; Oracle Cloud Infrastructure Developer Tools], [Virtual Nodes &#8211; Oracle Cloud Infrastructure Container Engine for Kubernetes]

### NEW QUESTION 52

Which of the following are VALID log category with regard to logging service in Oracle Cloud Infrastructure?

(Choose two.)

- \* Custom logs
- \* User logs
- \* Query logs
- \* Audit logs

Explanation

The two valid log categories with regard to the logging service in Oracle Cloud Infrastructure (OCI) are: Audit logs: Audit logs capture actions and events related to resource operations in OCI. These logs provide visibility into who performed an action, what action was performed, and when it occurred, helping with compliance and security monitoring. Custom logs: Custom logs allow users to define and send their application-specific log data to OCI logging service. Users can create custom log groups and log streams to organize and manage their log data. This enables centralized log management and analysis for custom applications and services running in OCI. Query logs and User logs are not valid log categories in OCI's logging service.

### NEW QUESTION 53

Pods running in your Oracle Container Engine for Kubernetes (OKE) cluster often need to communicate with other pods in the cluster or with services outside the cluster. As the OKE cluster administrator, you have been tasked with configuring permissions to restrict pod-to-pod communications except as explicitly allowed.

Where can you define these permissions?

- \* Security Lists
- \* RBAC Roles
- \* Network Policies
- \* IAM Policies

Explanation

As the OKE cluster administrator, you can define permissions to restrict pod-to-pod communications except as explicitly allowed by using Network Policies. Network Policies are a Kubernetes feature that allows you to define rules for network traffic within the

cluster. They provide fine-grained control over ingress (incoming) and egress (outgoing) traffic between pods. By creating Network Policies, you can specify the allowed communication paths between pods based on various criteria such as source and destination pods, namespaces, IP addresses, ports, and protocols. This allows you to enforce security and isolation within your OKE cluster, ensuring that pods can only communicate with authorized pods or services. RBAC Roles and IAM Policies are used to manage access control and permissions for managing and interacting with the cluster itself, but they do not directly control pod-to-pod communications. Security Lists, on the other hand, are associated with VCN (Virtual Cloud Network) resources and control traffic at the subnet level, not at the pod level within the OKE cluster. Reference:

[https://docs.oracle.com/en-us/iaas/Content/Security/Reference/oke\\_security.htm](https://docs.oracle.com/en-us/iaas/Content/Security/Reference/oke_security.htm)

#### NEW QUESTION 54

You host a microservices based application on the Oracle Cloud Infrastructure Container Engine for Kubernetes (OKE). Due to increased popularity of your application, you need to provision more resources to meet the growing demand. Which three statements are true for the given scenario?

- \* Enable autoscaling by autoscaling Pods by deploying Kubernetes Autoscaler to collect resource metrics from each worker node in the cluster.
- \* Enable cluster autoscaling by autoscaling node pools by deploying the Kubernetes Autoscaler to automatically resize a cluster's node pools based on application workload demands.
- \* Scale a cluster up and down by changing the number of node pools in the cluster.
- \* Enable cluster autoscaling by autoscaling node pools by deploying Kubernetes Metrics Server and using the Kubernetes Vertical Pod Autoscaler to adjust the resource requests and limits.
- \* Scale a node pool up and down to change the number of worker nodes in the node pool, and the availability domains and subnets in which to place them.

Explanation

The statements that are true for scaling an OKE cluster to meet growing demand are:

- \* Enable autoscaling by autoscaling Pods by deploying Kubernetes Autoscaler to collect resource metrics from each worker node in the cluster. Pod autoscaling is a feature that allows you to adjust the number of pods in a deployment or replica set based on the CPU or memory utilization of the pods. You can use Kubernetes Autoscaler, which is an add-on component that you can install on your OKE cluster, to collect resource metrics from each worker node and scale the pods up or down accordingly.
- \* Enable cluster autoscaling by autoscaling node pools by deploying the Kubernetes Autoscaler to automatically resize a cluster's node pools based on application workload demands. Cluster autoscaling is a feature that allows you to adjust the number of nodes in a node pool based on the pod requests and limits of the pods running on the nodes. You can use Kubernetes Autoscaler, which is an add-on component that you can install on your OKE cluster, to monitor the pod requests and limits and scale the node pools up or down accordingly.
- \* Scale a node pool up and down to change the number of worker nodes in the node pool, and the availability domains and subnets in which to place them. A node pool is a group of worker nodes within an OKE cluster that share the same configuration, such as shape, image, subnet, etc. You can use OCI Console, CLI, or API to scale a node pool up and down by adding or removing worker nodes from it.

You can also change the availability domains and subnets for your node pool to distribute your nodes across different fault domains. Scaling a node pool allows you to adjust your cluster capacity according to your application workload demands. Verified References: [Scaling Clusters &#8211; Oracle Cloud Infrastructure Container Engine for Kubernetes], [Scaling Node Pools &#8211; Oracle Cloud Infrastructure Container Engine for Kubernetes]

#### NEW QUESTION 55

A startup is building an application and has decided to deploy it on Oracle Cloud Infrastructure (OCI) DevOps. They want to automate the infrastructure provisioning and configuration of OCI resources such as Compute, Load Balancing, and Database services. Which tool should they use for this purpose and why?

- \* OCI OKE. In OCI, Container Engine for Kubernetes is a configuration management tool that manages enterprise-scale server infrastructure with minimal human intervention using Infrastructure as Code (IaC).
- \* Splunk. With the OCI DevOps service, users can manage OCI resources using the Splunk Plug-in, a CLI tool that provides help with managing repositories and automating infrastructure.
- \* Resource Manager. In OCI, the [Resource Manager](#); automates infrastructure provisioning and configuring of OCI resources, such as Compute, Load Balancing, and Database services.
- \* Jenkins. In OCI, Jenkins is an automation tool for configuration management that focuses on automating delivery and management of entire IT infrastructure stacks.

### NEW QUESTION 56

You have been asked to provision a new production environment on Oracle Cloud Infrastructure (OCI). After working with the solution architect you decide that you are going to automate this process. Which OCI service can help automate the provisioning of this new environment?

- \* OCI Streaming Service
- \* Oracle Functions
- \* OCI Resource Manager
- \* Oracle Container Engine for Kubernetes

Explanation

The OCI service that can help automate the provisioning of a new environment is OCI Resource Manager.

OCI Resource Manager is a service provided by Oracle Cloud Infrastructure that enables you to automate the process of provisioning, updating, and managing infrastructure resources. It allows you to define your infrastructure as code using tools like Terraform, and then use Resource Manager to create and manage stacks.

Stacks are the deployment units that contain the infrastructure resources defined in your code. By leveraging OCI Resource Manager, you can automate the provisioning of a new production environment by defining the required infrastructure resources in a stack using Terraform code. Resource Manager will then handle the creation and management of these resources, ensuring that your environment is provisioned consistently and according to the defined infrastructure as code. Therefore, OCI Resource Manager is the recommended service to automate the provisioning of a new environment in Oracle Cloud Infrastructure.

### Oracle 1z0-1109-23 Exam Syllabus Topics:

TopicDetailsTopic 1- Evaluate and configure security for container images used in OCI- Create and configure various deployment strategiesTopic 2- Configure and manage Continuous Integration and Continuous Delivery (CI- CD)- Identify the need for containerization and create containers using the dockerTopic 3- Explain the Configuration Management process- Explain DevSecOps and configure security using DevSecOps best practices in OCITopic 4- Automate the Software Development Life Cycle using OCI DevOps service- Create and manage Oracle Cloud Infrastructure Container InstancesTopic 5- Use OCI

Resource Manager to provision infrastructure as code- Create and manage encryption Keys and secrets in OCI VaultTopic 6- Evaluate and configure Build Pipelines and Deployment Pipelines- Use DevOps as a service to solve a real-world problem  
Topic 7- Evaluate and configure security within OCI DevOps CI- CD pipelines- Recall and list the practices associated with DevOpsTopic 8- Create and manage Oracle Cloud Infrastructure Registry OCIR- Analyze and manage logs with OCI Logging service

**Download the Latest 1z0-1109-23 Dump - 2024 1z0-1109-23 Exam Question Bank:**

<https://www.dumpsmaterials.com/1z0-1109-23-real-torrent.html>