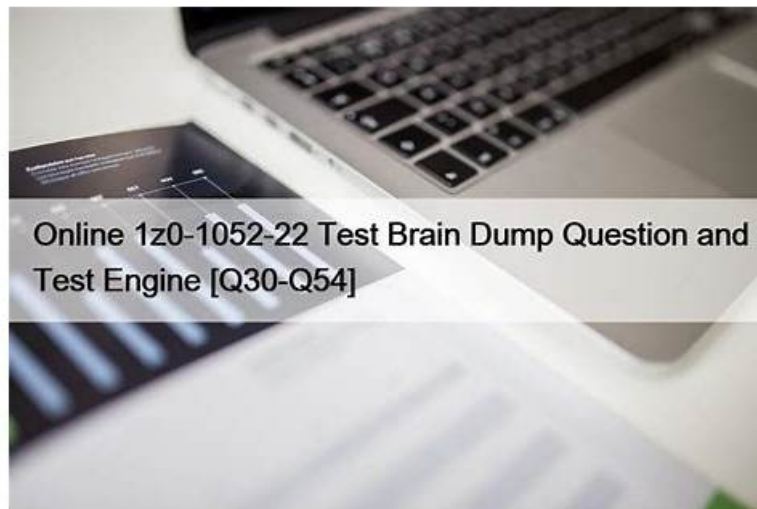


1Z0-1067-25 Practice Test Engine - Latest 1Z0-1067-25 Training



DOWNLOAD the newest VCE4Dumps 1Z0-1067-25 PDF dumps from Cloud Storage for free: <https://drive.google.com/open?id=14an2j1pPirf4efV7ZuXGg-aHmk1pYexK>

To other workers who want to keep up with the time and being competent in today's world, you are also looking for some effective 1Z0-1067-25 exam prep as well. Without voluminous content to remember, our 1Z0-1067-25 practice materials contain what you need to know and what the exam want to test, So our 1Z0-1067-25 Real Exam far transcend others in market. We never avoid our responsibility of offering help for exam candidates like you, so choosing our 1Z0-1067-25 training guide means you choose success.

Oracle 1Z0-1067-25 Exam Syllabus Topics:

Topic	Details
Topic 1	Implementing Observability: This section of the exam focuses on monitoring and maintaining cloud infrastructure. It covers implementing Metric Query Language (MQL) for analyzing performance data, setting up alarms and notifications for system events, and performing health checks to ensure the stability of cloud services.
Topic 2	Utilizing Configuration Management Tools: This section of the exam measures the skills of the target audience] and focuses on configuring cloud resources efficiently. It covers the use of configuration management tools for automating resource setup and cloud-init for initializing compute instances, ensuring proper configuration from the start.
Topic 3	Deploying and Managing Resources: This section of the exam covers the deployment and management of core OCI services. It focuses on manual provisioning, using the OCI Command Line Interface (CLI) for managing resources, and utilizing infrastructure as code to create consistent and repeatable deployments.

>> 1Z0-1067-25 Practice Test Engine <<

Latest 1Z0-1067-25 Training & Interactive 1Z0-1067-25 EBook

The downloading process is operational. It means you can obtain 1Z0-1067-25 quiz torrent within 10 minutes if you make up your mind. Do not be edgy about the exam anymore, because those are latest 1Z0-1067-25 exam torrent with efficiency and accuracy. You will not need to struggle with the exam. Besides, there is no difficult sophistication about the procedures, our latest 1Z0-1067-25 Exam Torrent materials have been in preference to other practice materials and can be obtained immediately.

Oracle Cloud Infrastructure 2025 Cloud Ops Professional Sample Questions (Q48-Q53):

NEW QUESTION # 48

You have set an alarm to be generated when the CPU usage of a specified instance is greater than 10%. In the alarm behavior view below you notice that the critical condition happened around 23:30. You were expecting a notification after 1 minute, however, the alarm firing state did not begin until 23:33.



What should you change to fix it? (Choose the best answer.)

- A. Change the alarm condition to be greater than 3%.
- B. Change the alarm trigger delay minutes value to 1.
- C. Change the alarm metric interval to 1.
- D. Change the notification topic that you previously associated with the alarm.

Answer: B

NEW QUESTION # 49

SIMULATION

Scenario: 1 (Create a reusable VCN Configuration with Terraform)

Scenario Description: (Hands-On Performance Exam Certification)

You'll launch and destroy a VCN and subnet by creating Terraform automation scripts and issuing commands in Code Editor. Next, you'll download those Terraform scripts and create a stack by uploading them into Oracle Cloud Infrastructure Resource Manager. You'll then use that service to launch and destroy the same VCN and subnet.

In this scenario, you will:

- Create a Terraform folder and file in Code Editor.
- Create and destroy a VCN using Terraform.
- Create and destroy a VCN using Resource Manager.

Answer:

Explanation:

See the solution below with Step by Step Explanation

Explanation:

Create a Terraform Folder and File in Code Editor:

You'll create a folder and file to hold your Terraform scripts.

- Log in to your tenancy in the Cloud Console and open the Code Editor, whose icon is at the top-right corner, to the right of the CLI Cloud Shell icon.
- Expand the Explorer panel with the top icon on the left panel. It looks like two overlapping documents.
- Expand the drop-down for your home directory if it isn't already expanded. It's okay if it is empty.
- Create a new folder by clicking File, then New Folder, and name it terraform-vcn.
- Create a file in that folder by clicking File, then New File, and name it vcn.tf. To make Code Editor, create the file in the correct folder, click the folder name in your home directory to highlight it.
- First, you'll set up Terraform and the OCI Provider in this directory. Add these lines to the file:

```
terraform {required_providers {oci = {source = "oracle/oci"version = ">=4.67.3"}}required_version = ">= 1.0.0"}
```
- Save the changes by clicking File, then Save.
- Now, run this code. Open a terminal panel in Cloud Editor by clicking Terminal, then New Terminal.
- Use pwd to check that you are in your home directory.
- Enter ls and you should see your terraform_vcn directory.
- Enter cd terraform_vcn/ to change to that directory with.
- Use terraform init to initialize this directory for Terraform.
- Use ls -la and you should see that Terraform created a hidden directory and file.

Create and Destroy a VCN Using Terraform

You'll create a Terraform script that will launch a VCN and subnet.

You'll then alter your script and create two additional files that will apply a compartment OCID variable to your Terraform script.

Write the Terraform

1. Add the following code block to your Terraform script to declare a VCN, replacing `<your_compartment_ocid>` with the proper OCID. The only strictly required parameter is the compartment OCID, but you'll add more later.

If you need to retrieve your compartment OCID, navigate to Identity & Security, then Compartments. Find your compartment, hover the cursor over the OCID, and click Copy.

`resource "oci_core_vcn" "example_vcn" {compartment_id = "<your_compartment_ocid>"}` This snippet declares a resource block of type `oci_core_vcn`. The label that Terraform will use for this resource is `example_vcn`.

2. In the terminal, run `terraform plan`, and you should see that Terraform would create a VCN. Because most of the parameters were unspecified, terraform will list their values as "(known after apply)." You can ignore the "-out option to save this plan" warning. Note that `terraform plan` parses your Terraform configuration and creates an execution plan for the associated stack, while `terraform apply` applies the execution plan to create (or modify) your resources.

3. Add a display name and CIDR block (the bolded portion) to the code. Note that we want to set the `cidr_blocks` parameter, rather than `cidr_block` (which is deprecated).

`resource "oci_core_vcn" "example_vcn" {compartment_id = "<your_compartment_ocid>"display_name = "VCN-01"cidr_blocks = ["10.0.0.0/16"]}`

4. Save the changes and run `terraform plan` again. You should see the display name and CIDR block reflected in Terraform's plan.

5. Now add a subnet to this VCN. At the bottom of the file, add the following block:

`resource "oci_core_subnet" "example_subnet" {compartment_id = "<your_compartment_ocid>"display_name = "SNT-01"vcn_id = oci_core_vcn.example_vcn.idcidr_block = "10.0.0.0/24"}` Note the line where we set the VCN ID. Here we reference the OCID of the previously declared VCN, using the name we gave it to Terraform: `example_vcn`. This dependency makes Terraform provision the VCN first, wait for OCI to return the OCID, then provision the subnet.

6. Run `terraform plan` to see that it will now create a VCN and subnet.

Add Variables

7. Before moving on there are a few ways to improve the existing code. Notice that the subnet and VCN both need the compartment OCID. We can factor this out into a variable. Create a file named `variables.tf`

8. In `variables.tf`, declare a variable named `compartment_id`:

`variable "compartment_id" {type = string}`

9. In `vcn.tf`, replace all instances of the compartment OCID with `var.compartment_id` as follows:

`terraform {required_providers {oci = {source = "oracle/oci"version = ">=4.67.3"}}required_version = ">= 1.0.0"}`
`resource "oci_core_vcn" "example_vcn" {compartment_id = var.compartment_iddisplay_name = "VCN-01"cidr_blocks = ["10.0.0.0/16"]}`
`resource "oci_core_subnet" "example_subnet" {compartment_id = var.compartment_iddisplay_name = "SNT-01"vcn_id = oci_core_vcn.example_vcn.idcidr_block = "10.0.0.0/24"}` Save your changes in both `vcn.tf` and `variables.tf`

10. If you were to run `terraform plan` or `apply` now, Terraform would see a variable and provide you a prompt to input the compartment OCID. Instead, you'll provide the variable value in a dedicated file. Create a file named exactly `terraform.tfvars`

11. Terraform will automatically load values provided in a file with this name. If you were to use a different name, you would have to provide the file name to the Terraform CLI. Add the value for the compartment ID in this file:

`compartment_id = "<your_compartment_ocid>"`

Be sure to save the file.

12. Run `terraform plan` and you should see the same output as before.

Provision the VCN

13. Run `terraform apply` and confirm that you want to make the changes by entering yes at the prompt.

14. Navigate to VCNs in the console. Ensure that you have the right compartment selected. You should see your VCN. Click its name to see the details. You should see its subnet listed.

Terminate the VCN

15. Run `terraform destroy`. Enter yes to confirm. You should see the VCN terminate. Refresh your browser if needed.

Create and Destroy a VCN Using Resource Manager (You will most probably be tested on this in the actual certification) We will reuse the Terraform code but replace the CLI with Resource Manager.

1. Create a folder named `terraform_vcn` on your host machine. Download the `vcn.tf`, `terraform.tfvars`, and `variables.tf` files from Code Editor and move them to the `terraform_vcn` folder to your local machine. To download from Code Editor, right-click the file name in the Explorer panel and select Download. You could download the whole folder at once, but then you would have to delete Terraform's hidden files.

Create a Stack

2. Navigate to Resource Manager in the Console's navigation menu under Developer Services. Go to the Stacks page.

3. Click Create stack.

a. The first page of the form will be for stack information.

1) For the origin of the Terraform configuration, keep My configuration selected.

2) Under Stack configuration, upload your `terraform_vcn` folder.

3) Under Custom providers, keep Use custom Terraform providers deselected.

4) Name the stack and give it a description.

5) Ensure that your compartment is selected.

6) Click Next.

b. The second page will be for variables.

1) Because you uploaded a terraform.tfvars file, Resource Manager will auto-populate the variable for compartment OCID.

2) Click Next.

c. The third page will be for review.

1) Keep Run apply deselected.

2) Click Create. This will take you to the stack's details page.

Run a Plan Job

4. The stack itself is only a bookkeeping resource-no infrastructure was provisioned yet. You should be on the stack's page. Click Plan. A form will pop up.

a. Name the job RM-Plan-01.

b. Click Plan again at the bottom to submit a job for Resource Manager to run terraform plan. This will take you to the job's details page.

5. Wait for the job to complete, and then view the logs. They should match what you saw when you ran Terraform in Code Editor.

Run an Apply Job

6. Go back to the stack's details page (use the breadcrumbs). Click Apply. A form will pop up.

a. Name the job RM-Apply-01.

b. Under Apply job plan resolution, select the plan job we just ran (instead of "Automatically approve"). This makes it execute based on the previous plan, instead of running a new one.

c. Click Apply to submit a job for Resource Manager to run terraform apply. This will take you to the job's details page.

7. Wait for the job to finish. View the logs and confirm that it was successful.

View the VCN

8. Navigate to VCNs in the Console through the navigation menu under Networking and Virtual Cloud Networks.

9. You should see the VCN listed in the table. Click its name to go to its Details page.

10. You should see the subnet listed.

Run a Destroy Job

11. Go back to the stack's details page in Resource Manager.

12. Click Destroy. Click Destroy again on the menu that pops up.

13. Wait for the job to finish. View the logs to see that it completed successfully.

14. Navigate back to VCNs in the Console. You should see that it has been terminated.

15. Go back to the stack in Resource Manager. Click the drop-down for More actions. Select Delete stack. Confirm by selecting Delete.

NEW QUESTION # 50

You have a web application that is running on compute instances distributed across an availability domain's fault domains. To share state, instances of the application need to read and write to a shared file system that supports concurrent access from multiple instances.

Which two can help fulfill this technical requirement?

- A. Attach a block volume to each instance. Enable replication between the block volumes.
- B. Create a file system in the File Storage service. Create a mount target and export the file system. Mount the export onto all of the instances.
- C. Attach a single block volume in read/write shared mode to all the instances. Enable the cluster plug-in on the Oracle Cloud Agent to coordinate access to the shared volume.
- D. Attach a single block volume in read/write shared mode to all the instances. Use a cluster aware system such as Oracle Cluster File System version 2 (OCFS2) to coordinate access to the shared volume.

Answer: B,D

NEW QUESTION # 51

The general syntax for an IAM policy is: Allow <identity_domain_name>/<subject> to <verb> <resource-type> in <location> where <conditions> Which two are valid values for <location>?

- A. availability-domain aBCD:us-phoenix-1
- B. security-zone MyZone
- C. region us-phoenix-1
- D. compartment MyCompartment
- E. tenancy

Answer: D,E

NEW QUESTION # 52

You have been asked to update the lifecycle policy for object storage using the Oracle Cloud Infrastructure (OCI) Command Line Interface (CLI). Which command can successfully update the policy? (Choose the best answer.)

- A. `oci os object-lifecycle-policy put --ns <object_storage_namespace> --bn <bucket_name> -- --items <json_formatted_lifecycle_policy>`
- B. `oci os object-lifecycle-policy get --ns <object_storage_namespace> --bn <bucket_name>`
- C. `oci os object-lifecycle-policy put --ns <object_storage_namespace> --bn <bucket_name>`
- D. `oci os object-lifecycle-policy delete --ns <object_storage_namespace> --bn <bucket_name>`

Answer: A

NEW QUESTION # 53

• • • • •

You can trust VCE4Dumps 1Z0-1067-25 exam real questions and start preparation without wasting further time. We are quite confident that with the VCE4Dumps 1Z0-1067-25 real exam questions you will get everything that you need to learn, prepare and pass the challenging Oracle 1Z0-1067-25 Certification Exam easily.

Latest 1Z0-1067-25 Training: <https://www.vce4dumps.com/1Z0-1067-25-valid-torrent.html>

- [illegible]

BTW, DOWNLOAD part of VCE4Dumps 1Z0-1067-25 dumps from Cloud Storage: <https://drive.google.com/open?id=14an2j1pPirf4efV7ZuXGg-aHmk1pYexK>