

1Z0-1084-20^{Q&As}

Oracle Cloud Infrastructure Developer 2020 Associate

Pass Oracle 1Z0-1084-20 Exam with 100% Guarantee

Free Download Real Questions & Answers **PDF** and **VCE** file from:

<https://www.pass2lead.com/1z0-1084-20.html>

100% Passing Guarantee
100% Money Back Assurance

Following Questions and Answers are all new published by Oracle
Official Exam Center

-  **Instant Download** After Purchase
-  **100% Money Back** Guarantee
-  **365 Days** Free Update
-  **800,000+** Satisfied Customers



QUESTION 1

Which one of the following is NOT a valid backend-type supported by Oracle Cloud Infrastructure (OCI) API Gateway?

- A. STOCK_RESPONSE_BACKEND
- B. ORACLE_FUNCTIONS_BACKEND
- C. ORACLE_STREAMS_BACKEND
- D. HTTP_BACKEND

Correct Answer: C

In the API Gateway service, a back end is the means by which a gateway routes requests to the back-end services that implement APIs. If you add a private endpoint back end to an API gateway, you give the API gateway access to the VCN associated with that private endpoint. You can also grant an API gateway access to other Oracle Cloud Infrastructure services as back ends. For example, you could grant an API gateway access to Oracle Functions, so you can create and deploy an API that is backed by a serverless function. API Gateway service to create an API gateway, you can create an API deployment to access HTTP and HTTPS URLs. <https://docs.cloud.oracle.com/en-us/iaas/Content/APIGateway/Tasks/apigatewayusinghttpbackend.htm> API Gateway service to create an API gateway, you can create an API deployment that invokes serverless functions defined in Oracle Functions. <https://docs.cloud.oracle.com/en-us/iaas/Content/APIGateway/Tasks/apigatewayusingfunctionsbackend.htm> API Gateway service, you can define a path to a stock response back end <https://docs.cloud.oracle.com/en-us/iaas/Content/APIGateway/Tasks/apigatewayaddingstockresponses.htm>

QUESTION 2

What is the open source engine for Oracle Functions?

- A. Apache OpenWhisk
- B. OpenFaaS
- C. Fn Project
- D. Knative

Correct Answer: C

<https://www.oracle.com/webfolder/technetwork/tutorials/FAQs/oci/Functions-FAQ.pdf> Oracle Functions is a fully managed, multi-tenant, highly scalable, on-demand, Functions-as-a-Service platform. It is built on enterprise-grade Oracle Cloud Infrastructure and powered by the Fn Project open source engine. Use Oracle Functions (sometimes abbreviated to just Functions) when you want to focus on writing code to meet business needs.

QUESTION 3

What is the minimum of storage that a persistent volume claim can obtain in Oracle Cloud Infrastructure Container Engine for Kubernetes (OKE)?

- A. 50 GB

- B. 10 GB
- C. 1 GB D. 1 TB

Correct Answer: A

The minimum amount of persistent storage that a PVC can request is 50 gigabytes. If the request is for less than 50 gigabytes, the request is rounded up to 50 gigabytes. <https://docs.cloud.oracle.com/en-us/iaas/Content/ContEng/Tasks/contengcreatingpersistentvolumeclaim.htm>

QUESTION 4

Which two handle Oracle Functions authentication automatically?

- A. Oracle Cloud Infrastructure SDK
- B. cURL
- C. Oracle Cloud Infrastructure CLI
- D. Signed HTTP Request
- E. Fn Project CLI

Correct Answer: CE

Fn Project CLI you can create an Fn Project CLI Context to Connect to Oracle Cloud Infrastructure and specify --provider oracle This option enables Oracle Functions to perform authentication and authorization using Oracle Cloud Infrastructure request signing, private keys, user groups, and policies that grant permissions to those user groups.

QUESTION 5

Which is NOT a supported SDK on Oracle Cloud Infrastructure (OCI)?

- A. Ruby SDK
- B. Java SDK
- C. Python SDK
- D. Go SDK
- E. .NET SDK

Correct Answer: E

<https://docs.cloud.oracle.com/en-us/iaas/Content/API/Concepts/sdks.htm>

QUESTION 6

A service you are deploying to Oracle infrastructure (OCI) Container Engine for Kubernetes (OKE) uses a docker image

from a private repository Which configuration is necessary to provide access to this repository from OKE?

- A. Add a generic secret on the cluster containing your identity credentials. Then specify a registrycredentials property in the deployment manifest.
- B. Create a docker-registry secret for OCIR with API key credentials on the cluster, and specify the imagepullsecret property in the application deployment manifest.
- C. Create a docker-registry secret for OCIR with identity Auth Token on the cluster, and specify the image pull secret property in the application deployment manifest.
- D. Create a dynamic group for nodes in the cluster, and a policy that allows the dynamic group to read repositories in the same compartment.

Correct Answer: C

Pulling Images from Registry during Deployment During the deployment of an application to a Kubernetes cluster, you'll typically want one or more images to be pulled from a Docker registry. In the application's manifest file you specify the images to pull, the registry to pull them from, and the credentials to use when pulling the images. The manifest file is commonly also referred to as a pod spec, or as a deployment.yaml file (although other filenames are allowed). If you want the application to pull images that reside in Oracle Cloud Infrastructure Registry, you have to perform two steps:

-

You have to use kubectl to create a Docker registry secret. The secret contains the Oracle Cloud Infrastructure credentials to use when pulling the image. When creating secrets, Oracle strongly

recommends you use the latest version of kubectl To create a Docker registry secret: 1- If you haven't already done so, follow the steps to set up the cluster's kubeconfig configuration file and (if necessary) set the KUBECONFIG environment variable to point to the file. Note that you must set up your own kubeconfig file. You cannot access a cluster using a kubeconfig file that a different user set up. 2- In a terminal window, enter: \$ kubectl create secret docker-registry --docker-server=.ocir.io --dockerusername='/' --docker-password='/' --dockeremail='/' where: is a name of your choice, that you will use in the manifest file to refer to the secret . For example, ocirsecret is the key for the Oracle Cloud Infrastructure Registry region you're using. For example, iad. See Availability by Region. ocir.io is the Oracle Cloud Infrastructure Registry name. is the auto-generated Object Storage namespace string of the tenancy containing the repository from which the application is to pull the image (as shown on the Tenancy Information page). For example, the namespace of the acme-dev tenancy might be ansh81vru1zp. Note that for some older tenancies, the namespace string might be the same as the tenancy name in all lower-case letters (for example, acmedev). is the username to use when pulling the image. The username must have access to the tenancy specified by . For example, jdoe@acme.com . If your tenancy is federated with Oracle Identity Cloud Service, use the format oracleidentitycloudservice/ is the auth token of the user specified by . For example, kjj64r{1sJSSF-;)K8 is an email address. An email address is required, but it doesn't matter what you specify. For example, jdoe@acme.com

-

You have to specify the image to pull from Oracle Cloud Infrastructure Registry, including the repository location and the Docker registry secret to use, in the application's manifest file.

QUESTION 7

You are building a cloud native, serverless travel application with multiple Oracle Functions in Java, Python and Node.js. You need to build and deploy these functions to a single applications named travel-app. Which command will help you complete this task successfully?

- A. oci fn function deploy --ap travel-ap --all
- B. fn deploy --ap travel-ap -- all
- C. oci fn application --application-name-ap deploy --all
- D. fn function deploy --all --application-name travel-ap

Correct Answer: B

check the steps for Creating, Deploying, and Invoking a Helloworld Function <https://docs.cloud.oracle.com/en-us/iaas/Content/Functions/Tasks/functionscreatingfirst.htm> in step 7 that will deploy the function. Enter the following single Fn Project command to build the function and its dependencies as a Docker image called helloworld-func, push the image to the specified Docker registry, and deploy the function to Oracle Functions in the helloworld-app: \$ fn -v deploy --app helloworld-app The -v option simply shows more detail about what Fn Project commands are doing (see Using the Fn Project CLI with Oracle Functions).

QUESTION 8

Which testing approaches is a must for achieving high velocity of deployments and release of cloud- native applications?

- A. Integration testing
- B. A/B testing
- C. Automated testing
- D. Penetration testing

Correct Answer: C

Oracle Cloud Infrastructure provides a number of DevOps tools and plug-ins for working with Oracle Cloud Infrastructure services. These can simplify provisioning and managing infrastructure or enable automated testing and continuous delivery. **A/B Testing** While A/B testing can be combined with either canary or blue-green deployments, it is a very different thing. A/B testing really targets testing the usage behavior of a service or feature and is typically used to validate a hypothesis or to measure two versions of a service or feature and how they stack up against each other in terms of performance, discoverability and usability. A/B testing often leverages feature flags (feature toggles), which allow you to dynamically turn features on and off. **Integration Testing** Integration tests are also known as end-to-end (e2e) tests. These are long-running tests that exercise the system in the way it is intended to be used in production. These are the most valuable tests in demonstrating reliability and thus increasing confidence. **Penetration Testing** Oracle regularly performs penetration and vulnerability testing and security assessments against the Oracle cloud infrastructure, platforms, and applications. These tests are intended to validate and improve the overall security of Oracle Cloud Services.

QUESTION 9

You are deploying an API via Oracle Cloud Infrastructure (OCI) API Gateway and you want to implement request policies to control access Which is NOT available in OCI API Gateway?

- A. Limiting the number of requests sent to backend services

- B. Enabling CORS (Cross-Origin Resource Sharing) support
- C. Providing authentication and authorization
- D. Controlling access to OCI resources

Correct Answer: D

In the API Gateway service, there are two types of policy:

-

a request policy describes actions to be performed on an incoming request from a caller before it is sent to a back end

-

a response policy describes actions to be performed on a response returned from a back end before it is sent to a caller

You can use request policies to:

- limit the number of requests sent to back-end services
 - enable CORS (Cross-Origin Resource Sharing) support
 - provide authentication and authorization
-

QUESTION 10

Which two are characteristics of microservices?

- A. Microservices are hard to test in isolation.
- B. Microservices can be independently deployed.
- C. All microservices share a data store.
- D. Microservices can be implemented in limited number of programming languages.
- E. Microservices communicate over lightweight APIs.

Correct Answer: BE

<https://www.techjini.com/blog/microservices/>

QUESTION 11

You are developing a polyglot serverless application using Oracle Functions. Which language cannot be used to write your function code?

- A. PL/SQL
- B. Python

C. Node.js

D. Java

Correct Answer: A

The serverless and elastic architecture of Oracle Functions means there's no infrastructure administration or software administration for you to perform. You don't provision or maintain compute instances, and operating system software patches and upgrades are applied automatically. Oracle Functions simply ensures your app is highly-available, scalable, secure, and monitored. With Oracle Functions, you can write code in Java, Python, Node, Go, and Ruby (and for advanced use cases, bring your own Dockerfile, and Graal VM). You can then deploy your code, call it directly or trigger it in response to events, and get billed only for the resources consumed during the execution.

QUESTION 12

Your Oracle Cloud Infrastructure Container Engine for Kubernetes (OKE) administrator has created an OKE cluster with one node pool in a public subnet. You have been asked to provide a log file from one of the nodes for troubleshooting purpose.

Which step should you take to obtain the log file?

- A. ssh into the node using public key.
- B. ssh into the nodes using private key.
- C. It is impossible since OKE is a managed Kubernetes service.
- D. Use the username open and password to login.

Correct Answer: B

Kubernetes cluster is a group of nodes. The nodes are the machines running applications. Each node can be a physical machine or a virtual machine. The node's capacity (its number of CPUs and amount of memory) is defined when the node is created. A cluster comprises: - one or more master nodes (for high availability, typically there will be a number of master nodes) - one or more worker nodes (sometimes known as minions) Connecting to Worker Nodes Using SSH If you provided a public SSH key when creating the node pool in a cluster, the public key is installed on all worker nodes in the cluster. On UNIX and UNIX-like platforms (including Solaris and Linux), you can then connect through SSH to the worker nodes using the ssh utility (an SSH client) to perform administrative tasks. Note the following instructions assume the UNIX machine you use to connect to the worker node: Has the ssh utility installed. Has access to the SSH private key file paired with the SSH public key that was specified when the cluster was created. How to connect to worker nodes using SSH depends on whether you specified public or private subnets for the worker nodes when defining the node pools in the cluster. Connecting to Worker Nodes in Public Subnets Using SSH Before you can connect to a worker node in a public subnet using SSH, you must define an ingress rule in the subnet's security list to allow SSH access. The ingress rule must allow access to port 22 on worker nodes from source 0.0.0.0/0 and any source port To connect to a worker node in a public subnet through SSH from a UNIX machine using the ssh utility: 1- Find out the IP address of the worker node to which you want to connect. You can do this in a number of ways: Using kubectl. If you haven't already done so, follow the steps to set up the cluster's kubeconfig configuration file and (if necessary) set the KUBECONFIG environment variable to point to the file. Note that you must set up your own kubeconfig file. You cannot access a cluster using a kubeconfig file that a different user set up. See Setting Up Cluster Access. Then in a terminal window, enter kubectl get nodes to see the public IP addresses of worker nodes in node pools in the cluster. Using the Console. In the Console, display the Cluster List page and then select the cluster to which the worker node belongs. On the Node Pools tab, click the name of the node pool to which the worker node belongs. On the Nodes tab, you see the public IP address of every worker node in the node pool. Using the REST API. Use the ListNodePools

operation to see the public IP addresses of worker nodes in a node pool. 2- In the terminal window, enter `ssh opc@` to connect to the worker node, where is the IP address of the worker node that you made a note of earlier. For example, you might enter `ssh opc@192.0.2.254`. Note that if the SSH private key is not stored in the file or in the path that the ssh utility expects (for example, the ssh utility might expect the private key to be stored in `~/.ssh/id_rsa`), you must explicitly specify the private key filename and location in one of two ways: Use the `-i` option to specify the filename and location of the private key. For example, `ssh -i ~/.ssh/my_keys/my_host_key_filename opc@192.0.2.254` Add the private key filename and location to an SSH

configuration file, either the client configuration file (`~/.ssh/config`) if it exists, or the system-wide client

configuration file (`/etc/ssh/ssh_config`). For example, you might add the following:

```
Host 192.0.2.254 IdentityFile ~/.ssh/my_keys/my_host_key_filename
```

For more about the ssh utility's configuration file, enter `man ssh_config` Note also that permissions on the private key file must allow you read/write/execute access, but prevent other users from accessing the file.

For example, to set appropriate permissions, you might enter `chmod 600 ~/.ssh/my_keys/`

`my_host_key_filename`. If permissions are not set correctly and the private key file is accessible to other users, the ssh utility will simply ignore the private key file.

QUESTION 13

You created a pod called "nginx" and its state is set to Pending. Which command can you run to see the reason why the "nginx" pod is in the pending state?

- A. `kubect2 logs pod nginx`
- B. `kubect2 describe pod nginx`
- C. `kubect2 get pod nginx`
- D. Through the Oracle Cloud Infrastructure Console

Correct Answer: B

Debugging Pods

The first step in debugging a pod is taking a look at it. Check the current state of the pod and recent events with the following command:

```
kubectl describe pods ${POD_NAME}
```

Look at the state of the containers in the pod. Are they all Running? Have there been recent restarts?

Continue debugging depending on the state of the pods.

My pod stays pending

If a pod is stuck in Pending it means that it can not be scheduled onto a node. Generally this is because

there are insufficient resources of one type or another that prevent scheduling. Look at the output of the `kubectl describe ...` command above. There should be messages from the scheduler about why it can not schedule your pod. <https://kubernetes.io/docs/tasks/debug-application-cluster/debug-pod-replication-controller/>

QUESTION 14

You are tasked with developing an application that requires the use of Oracle Cloud Infrastructure (OCI) APIs to POST messages to a stream in the OCI Streaming service.

Which statement is incorrect?

- A. The request must include an authorization signing string including (but not limited to) `x-contentsha256`, `content-type`, and `content-length` headers.
- B. The `Content-Type` header must be Set to `application/json`
- C. An HTTP 401 will be returned if the client's clock is skewed more than 5 minutes from the server's.
- D. The request does not require an Authorization header.

Correct Answer: A

Emits messages to a stream. There's no limit to the number of messages in a request, but the total size of a message or request must be 1 MiB or less. The service calculates the partition ID from the message key and stores messages that share a key on the same partition. If a message does not contain a key or if the key is null, the service generates a message key for you. The partition ID cannot be passed as a parameter. `POST /20180418/streams//messages` Host: `streaming-api.us-phoenix-1.oraclecloud.com` { "messages": { { "key": null, "value": "VGhllHF1aWNrIGJyb3dulGZveCBqdW1wZWQgb3ZlciB0aGUgbGF6eSBkb2cu" }, { "key": null, "value": "UGFjayBteSBib3ggd2l0aCBmaXZlIGRvemVulGxpcXVvciBqdWdzLg==" } } } <https://docs.cloud.oracle.com/en-us/iaas/api/#/en/streaming/20180418/Message/PutMessages>

QUESTION 15

In a Linux environment, what is the default locations of the configuration file that Oracle Cloud Infrastructure CLI uses for profile information?

- A. `/etc/oci/config`
- B. `/usr/local/bin/config`
- C. `$HOME/.oci/config`
- D. `/usr/bin/oci/config`

Correct Answer: C

By default, the Oracle Cloud Infrastructure CLI configuration file is located at `~/.oci/config`. You might already have a configuration file as a result of installing the Oracle Cloud Infrastructure CLI.