AWS DVA-C01

AWS-CDA CERTIFICATION QUESTIONS & ANSWERS

Exam Summary – Syllabus – Questions

DVA-C01

AWS Certified Developer - Associate
65 Questions Exam – 720 / 1000 Cut Score – Duration of 130 minutes

www.VMExam.com

Table of Contents

Know Your DVA-C01 Certification Well:

The DVA-C01 is best suitable for candidates who want to gain knowledge in the AWS Associate. Before you start your DVA-C01 preparation you may struggle to get all the crucial AWS-CDA materials like DVA-C01 syllabus, sample questions, study guide.

But don't worry the DVA-C01 PDF is here to help you prepare in a stress free manner.

The PDF is a combination of all your queries like-

- What is in the DVA-C01 syllabus?
- How many questions are there in the DVA-C01 exam?
- Which Practice test would help me to pass the DVA-C01 exam at the first attempt?

Passing the DVA-C01 exam makes you AWS Certified Developer - Associate. Having the AWS-CDA certification opens multiple opportunities for you. You can grab a new job, get a higher salary or simply get recognition within your current organization.

AWS DVA-C01 AWS-CDA Certification Details:

Exam Name	AWS Developer Associate (AWS-CDA)
Exam Code	DVA-C01
Exam Price	$150 USD
Duration	130 minutes
Number of Questions	65
Passing Score	720 / 1000
Recommended Training / Books	**Developing on AWS**
Schedule Exam	**AWS Certification**
Sample Questions	**AWS DVA-C01 Sample Questions**
Recommended Practice	**AWS Certified Developer - Associate Practice Test**

DVA-C01 Syllabus:

Section	Objectives
Deployment - 22%	
Deploy written code in AWS using existing CI/CD pipelines, processes, and patterns.	- Commit code to a repository and invoke build, test and/or deployment actions - Use labels and branches for version and release management - Use AWS CodePipeline to orchestrate workflows against different environments - Apply AWS CodeCommit, AWS CodeBuild, AWS CodePipeline, AWS CodeStar, and AWS CodeDeploy for CI/CD purposes - Perform a roll back plan based on application deployment policy
Deploy applications using AWS Elastic Beanstalk.	- Utilize existing supported environments to define a new application stack - Package the application - Introduce a new application version into the Elastic Beanstalk environment - Utilize a deployment policy to deploy an application version (i.e., all at once, rolling, rolling with batch, immutable) - Validate application health using Elastic Beanstalk dashboard - Use Amazon CloudWatch Logs to instrument application logging
Prepare the application deployment package to be deployed to AWS.	- Manage the dependencies of the code module (like environment variables, config files and static image files) within the package - Outline the package/container directory structure and organize files appropriately - Translate application resource requirements to AWS infrastructure parameters (e.g., memory, cores)
Deploy serverless applications.	- Given a use case, implement and launch an AWS Serverless Application Model (AWS SAM) template - Manage environments in individual AWS services (e.g., Differentiate between Development, Test, and Production in Amazon API Gateway)
Security - 26%	
Make authenticated calls to AWS services.	- Communicate required policy based on least privileges required by application. - Assume an IAM role to access a service - Use the software development kit (SDK) credential provider on-premises or in the cloud to access AWS services (local credentials vs. instance roles)

Section	Objectives
Implement encryption using AWS services.	- Encrypt data at rest (client side; server side; envelope encryption) using AWS services - Encrypt data in transit
Implement application authentication and authorization.	- Add user sign-up and sign-in functionality for applications with Amazon Cognito identity or user pools - Use Amazon Cognito-provided credentials to write code that access AWS services. - Use Amazon Cognito sync to synchronize user profiles and data - Use developer-authenticated identities to interact between end user devices, backend authentication, and Amazon Cognito
Development with AWS Services - 30%	
Write code for serverless applications.	- Compare and contrast server-based vs. serverless model (e.g., micro services, stateless nature of serverless applications, scaling serverless applications, and decoupling layers of serverless applications) - Configure AWS Lambda functions by defining environment variables and parameters (e.g., memory, time out, runtime, handler) - Create an API endpoint using Amazon API Gateway - Create and test appropriate API actions like GET, POST using the API endpoint - Apply Amazon DynamoDB concepts (e.g., tables, items, and attributes) - Compute read/write capacity units for Amazon DynamoDB based on application requirements - Associate an AWS Lambda function with an AWS event source (e.g., Amazon API Gateway, Amazon CloudWatch event, Amazon S3 events, Amazon Kinesis) - Invoke an AWS Lambda function synchronously and asynchronously
Translate functional requirements into application design.	- Determine real-time vs. batch processing for a given use case - Determine use of synchronous vs. asynchronous for a given use case - Determine use of event vs. schedule/poll for a given use case - Account for tradeoffs for consistency models in an application design
Implement application design into application code.	- Write code to utilize messaging services (e.g., SQS, SNS) - Use Amazon ElastiCache to create a database cache - Use Amazon DynamoDB to index objects in Amazon S3 - Write a stateless AWS Lambda function - Write a web application with stateless web servers (Externalize state)

Section	Objectives
Write code that interacts with AWS services by using APIs, SDKs, and AWS CLI.	- Choose the appropriate APIs, software development kits (SDKs), and CLI commands for the code components - Write resilient code that deals with failures or exceptions (i.e., retries with exponential back off and jitter)
Refactoring - 10%	
Optimize applications to best use AWS services and features.	- Implement AWS caching services to optimize performance (e.g., Amazon ElastiCache, Amazon API Gateway cache) - Apply an Amazon S3 naming scheme for optimal read performance
Migrate existing application code to run on AWS.	- Isolate dependencies - Run the application as one or more stateless processes - Develop in order to enable horizontal scalability - Externalize state
Monitoring and Troubleshooting - 12%	
Write code that can be monitored.	- Create custom Amazon CloudWatch metrics - Perform logging in a manner available to systems operators - Instrument application source code to enable tracing in AWS X-Ray
Perform root cause analysis on faults found in testing or production.	- Interpret the outputs from the logging mechanism in AWS to identify errors in logs - Check build and testing history in AWS services (e.g., AWS CodeBuild, AWS CodeDeploy, AWS CodePipeline) to identify issues - Utilize AWS services (e.g., Amazon CloudWatch, VPC Flow Logs, and AWS X-Ray) to locate a specific faulty component

AWS DVA-C01 Sample Questions:

Question: 1

A developer wants to track the performance of an application that runs on a fleet of Amazon EC2 instances.

The developer wants to view and track statistics across the fleet, such as the average and maximum request latency. The developer would like to be notified immediately if the average response time exceeds a threshold.

Which solution meets these requirements?

a) Configure a cron job on each instance to measure the response time and update a log file stored in an Amazon S3 bucket every minute. Use an Amazon S3 event notification to trigger an AWS Lambda function that reads the log file and writes new entries to an Amazon Elasticsearch Service (Amazon ES) cluster. Visualize the results in a Kibana dashboard. Configure Amazon ES to send an alert to an Amazon SNS topic when the response time exceeds a threshold.
b) Configure the application to write the response times to the system log. Install and configure the Amazon Inspector agent to continually read the logs and send the response times to Amazon EventBridge. View the metrics graphs in the EventBridge console. Configure an EventBridge custom rule to send an Amazon SNS notification when the average of the response time metric exceeds the threshold.
c) Configure the application to write the response times to a log file. Install and configure the Amazon CloudWatch agent on the instances to stream the application log to CloudWatch Logs. Create a metric filter of the response time from the log. View the metrics graphs in the CloudWatch console. Create a CloudWatch alarm to send an Amazon SNS notification when the average of the response time metric exceeds the threshold.
d) Install and configure the AWS Systems Manager Agent on the instances to monitor the response time and send it to Amazon CloudWatch as a custom metric. View the metrics graphs in Amazon QuickSight. Create a CloudWatch alarm to send an Amazon SNS notification when the average of the response time metric exceeds the threshold.

Answer: c

Question: 2

A developer is testing an application locally and has deployed it to AWS Lambda. To remain under the package size limit, the dependencies were not included in the deployment file.

When testing the application remotely, the function does not execute because of missing dependencies. Which approach would resolve the issue?

a) Use the Lambda console editor to update the code and include the missing dependencies.
b) Create an additional .zip file with the missing dependencies and include the file in the original Lambda deployment package.
c) Add references to the missing dependencies in the Lambda function's environment variables.
d) Attach a layer to the Lambda function that contains the missing dependencies.

Answer: d

Question: 3

A developer is designing a web application that allows the users to post comments and receive near real-time feedback. Which architectures meet these requirements?

(Select TWO.)

a) Create an AWS AppSync schema and corresponding APIs. Use an Amazon DynamoDB table as the data store.
b) Create a WebSocket API in Amazon API Gateway. Use an AWS Lambda function as the backend and an Amazon DynamoDB table as the data store.
c) Create an AWS Elastic Beanstalk application backed by an Amazon RDS database. Configure the application to allow long-lived TCP/IP sockets.
d) Create a GraphQL endpoint in Amazon API Gateway. Use an Amazon DynamoDB table as the data store.
e) Enable WebSocket on Amazon CloudFront. Use an AWS Lambda function as the origin and an Amazon Aurora DB cluster as the data store.

Answer: a, b

Question: 4

A company is using Amazon API Gateway for its REST APIs in an AWS account. The security team wants to allow only IAM users from another AWS account to access the APIs.

Which combination of actions should the security team take to satisfy these requirements?

(Select TWO.)

a) Create an IAM permission policy and attach it to each IAM user. Set the APIs method authorization type to AWS_IAM. Use Signature Version 4 to sign the API requests.
b) Create an Amazon Cognito user pool and add each IAM user to the pool. Set the method authorization type for the APIs to COGNITO_USER_POOLS. Authenticate using the IAM credentials in Amazon Cognito and add the ID token to the request headers.
c) Create an Amazon Cognito identity pool and add each IAM user to the pool. Set the method authorization type for the APIs to COGNITO_USER_POOLS. Authenticate using the IAM credentials in Amazon Cognito and add the access token to the request headers.
d) Create a resource policy for the APIs that allows access for each IAM user only.
e) Create an Amazon Cognito authorizer for the APIs that allows access for each IAM user only. Set the method authorization type for the APIs to COGNITO_USER_POOLS.

Answer: a, d

Question: 5

A developer is building a web application that uses Amazon API Gateway. The developer wants to maintain different environments for development and production (dev and prod) workloads. The API will be backed by an AWS Lambda function with two aliases: one for dev and one for prod.

How can this be achieved with the LEAST amount of configuration?

a) Create a REST API for each environment and integrate the APIs with the corresponding dev and prod aliases of the Lambda function. Then deploy the two APIs to their respective stages and access them using the stage URLs.
b) Create one REST API and integrate it with the Lambda function using a stage variable in place of an alias. Then deploy the API to two different stages – dev and prod – and create a stage variable in each stage with different aliases as the values. Access the API using the different stage URLs.
c) Create one REST API and integrate it with the dev alias of the Lambda function, and deploy it to a dev environment. Configure a canary release deployment for prod where the canary will integrate with the Lambda prod alias.
d) Create one REST API and integrate it with the prod alias of the Lambda function and deploy it to a prod environment. Configure a canary release deployment for dev where the canary will integrate with the Lambda dev alias.

Answer: b

Question: 6

A company has AWS workloads in multiple geographical locations. A developer has created an Amazon Aurora database in the us-west-1 Region.

The database is encrypted using a customer-managed AWS KMS key. Now the developer wants to create the same encrypted database in the us-east-1 Region.

Which approach should the developer take to accomplish this task?

a) Create a snapshot of the database in the us-west-1 Region. Copy the snapshot to the us-east-1 Region and specify a KMS key in the us-east-1 Region. Restore the database from the copied snapshot.
b) Create an unencrypted snapshot of the database in the us-west-1 Region. Copy the snapshot to the useast-1 Region. Restore the database from the copied snapshot and enable encryption using the KMS key from the us-east-1 Region.
c) Disable encryption on the database. Create a snapshot of the database in the us-west-1 Region. Copy the snapshot to the us-east-1 Region. Restore the database from the copied snapshot.
d) In the us-east-1 Region, choose to restore the latest automated backup of the database from the us-west1 Region. Enable encryption using a KMS key in the us-east-1 Region.

Answer: a

Question: 7

A developer is adding Amazon ElastiCache for Memcached to a company's existing record storage application to reduce the load on the database and increase performance. The developer has decided to use lazy loading based on an analysis of common record handling patterns.

Which pseudocode example would correctly implement lazy loading?

a)
```
record_value = db.query("UPDATE Records SET Details = {1} WHERE ID == {0}", record_key, record_value)
cache.set (record_key, record_value)
```

b)
```
record_value = cache.get(record_key)
if (record_value == NULL)
record_value = db.query("SELECT Details FROM Records WHERE ID == {0}", record_key)
cache.set (record_key, record_value)
```

c)
```
record_value = cache.get (record_key)
db.query("UPDATE Records SET Details = {1} WHERE ID == {0}", record_key, record_value)
```

d)
```
record_value = db.query("SELECT Details FROM Records WHERE ID == {0}", record_key)
if (record_value != NULL)
cache.set (record_key, record_value)
```

Answer: b

Question: 8

A developer is adding sign-up and sign-in functionality to an application. The application is required to make an API call to a custom analytics solution to log user sign-in events.

Which combination of actions should the developer take to satisfy these requirements?

(Select TWO.)

a) Use Amazon Cognito to provide the sign-up and sign-in functionality.
b) Use AWS IAM to provide the sign-up and sign-in functionality.
c) Configure an AWS Config rule to make the API call triggered by the post-authentication event.
d) Invoke an Amazon API Gateway method to make the API call triggered by the post-authentication event.
e) Execute an AWS Lambda function to make the API call triggered by the post-authentication event.

Answer: a, e

Question: 9

A developer is building an application that transforms text files to .pdf files. The text files are written to a source Amazon S3 bucket by a separate application.

The developer wants to read the files as they arrive in Amazon S3 and convert them to .pdf files using AWS Lambda. The developer has written an IAM policy to allow access to Amazon S3 and Amazon CloudWatch Logs.

Which actions should the developer take to ensure that the Lambda function has the correct permissions?

a) Create a Lambda execution role using AWS IAM. Attach the IAM policy to the role. Assign the Lambda execution role to the Lambda function.
b) Create a Lambda execution user using AWS IAM. Attach the IAM policy to the user. Assign the Lambda execution user to the Lambda function.
c) Create a Lambda execution role using AWS IAM. Attach the IAM policy to the role. Store the IAM role as an environment variable in the Lambda function.
d) Create a Lambda execution user using AWS IAM. Attach the IAM policy to the user. Store the IAM user credentials as environment variables in the Lambda function.

Answer: a

Question: 10

A company is migrating a legacy application to Amazon EC2. The application uses a user name and password stored in the source code to connect to a MySQL database.

The database will be migrated to an Amazon RDS for MySQL DB instance. As part of the migration, the company wants to implement a secure way to store and automatically rotate the database credentials.

Which approach meets these requirements?

a) Store the database credentials in environment variables in an Amazon Machine Image (AMI). Rotate the credentials by replacing the AMI.
b) Store the database credentials in AWS Systems Manager Parameter Store. Configure Parameter Store to automatically rotate the credentials.
c) Store the database credentials in environment variables on the EC2 instances. Rotate the credentials by relaunching the EC2 instances.
d) Store the database credentials in AWS Secrets Manager. Configure Secrets Manager to automatically rotate the credentials.

Answer: d

Study Guide to Crack AWS-CDA DVA-C01 Exam:

- Getting details of the DVA-C01 syllabus, is the first step of a study plan. This pdf is going to be of ultimate help. Completion of the syllabus is must to pass the DVA-C01 exam.
- Making a schedule is vital. A structured method of preparation leads to success. A candidate must plan his schedule and follow it rigorously to attain success.
- Joining the AWS provided training for DVA-C01 exam could be of much help. If there is specific training for the exam, you can discover it from the link above.
- Read from the DVA-C01 sample questions to gain your idea about the actual exam questions. In this PDF useful sample questions are provided to make your exam preparation easy.
- Practicing on DVA-C01 practice tests is must. Continuous practice will make you an expert in all syllabus areas.