

AWS DEA-C01

AWS DATA ENGINEER ASSOCIATE CERTIFICATION QUESTIONS & ANSWERS

Exam Summary – Syllabus – Questions

DEA-C01

[AWS Certified Data Engineer - Associate](#)

85 Questions Exam – 720 / 1000 Cut Score – Duration of 170 minutes

www.VMExam.com

Table of Contents

Know Your DEA-C01 Certification Well:	2
AWS DEA-C01 Data Engineer Associate Certification Details:	2
DEA-C01 Syllabus:.....	3
Data Ingestion and Transformation - 34%	3
Data Store Management - 26%	5
Data Operations and Support - 22%	7
Data Security and Governance - 18%	9
AWS DEA-C01 Sample Questions:	11
Study Guide to Crack AWS Data Engineer Associate DEA-C01 Exam:	14

Know Your DEA-C01 Certification Well:

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

But don't worry the DEA-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 DEA-C01 syllabus?
- How many questions are there in the DEA-C01 exam?
- Which Practice test would help me to pass the DEA-C01 exam at the first attempt?

Passing the DEA-C01 exam makes you AWS Certified Data Engineer - Associate. Having the Data Engineer Associate 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 DEA-C01 Data Engineer Associate Certification Details:

Exam Name	AWS Certified Data Engineer - Associate
Exam Code	DEA-C01
Exam Price	\$150 USD
Duration	170 minutes
Number of Questions	85
Passing Score	720 / 1000
Schedule Exam	AWS Certification
Sample Questions	AWS DEA-C01 Sample Questions
Recommended Practice	AWS Certified Data Engineer - Associate Practice Test

DEA-C01 Syllabus:

Section	Objectives
Data Ingestion and Transformation - 34%	
Perform data ingestion.	<ul style="list-style-type: none"> - Knowledge of: <ul style="list-style-type: none"> • Throughput and latency characteristics for AWS services that ingest data • Data ingestion patterns (for example, frequency and data history) • Streaming data ingestion • Batch data ingestion (for example, scheduled ingestion, event-driven ingestion) • Replayability of data ingestion pipelines • Stateful and stateless data transactions - Skills in: <ul style="list-style-type: none"> • Reading data from streaming sources (for example, Amazon Kinesis, Amazon Managed Streaming for Apache Kafka [Amazon MSK], Amazon DynamoDB Streams, AWS Database Migration Service [AWS DMS], AWS Glue, Amazon Redshift) • Reading data from batch sources (for example, Amazon S3, AWS Glue, Amazon EMR, AWS DMS, Amazon Redshift, AWS Lambda, Amazon AppFlow) • Implementing appropriate configuration options for batch ingestion • Consuming data APIs • Setting up schedulers by using Amazon EventBridge, Apache Airflow, or time-based schedules for jobs and crawlers • Setting up event triggers (for example, Amazon S3 Event Notifications, EventBridge) • Calling a Lambda function from Amazon Kinesis • Creating allowlists for IP addresses to allow connections to data sources • Implementing throttling and overcoming rate limits (for example, DynamoDB, Amazon RDS, Kinesis) • Managing fan-in and fan-out for streaming data distribution
Transform and process data.	<ul style="list-style-type: none"> - Knowledge of: <ul style="list-style-type: none"> • Creation of ETL pipelines based on business requirements • Volume, velocity, and variety of data (for example, structured data, unstructured data)

Section	Objectives
	<ul style="list-style-type: none"> • Cloud computing and distributed computing • How to use Apache Spark to process data • Intermediate data staging locations <p>- Skills in:</p> <ul style="list-style-type: none"> • Optimizing container usage for performance needs (for example, Amazon Elastic Kubernetes Service [Amazon EKS], Amazon Elastic Container Service [Amazon ECS]) • Connecting to different data sources (for example, Java Database Connectivity [JDBC], Open Database Connectivity [ODBC]) • Integrating data from multiple sources • Optimizing costs while processing data • Implementing data transformation services based on requirements (for example, Amazon EMR, AWS Glue, Lambda, Amazon Redshift) • Transforming data between formats (for example, from .csv to Apache Parquet) • Troubleshooting and debugging common transformation failures and performance issues • Creating data APIs to make data available to other systems by using AWS services
Orchestrate data pipelines.	<p>- Knowledge of:</p> <ul style="list-style-type: none"> • How to integrate various AWS services to create ETL pipelines • Event-driven architecture • How to configure AWS services for data pipelines based on schedules or dependencies • Serverless workflows <p>- Skills in:</p> <ul style="list-style-type: none"> • Using orchestration services to build workflows for data ETL pipelines (for example, Lambda, EventBridge, Amazon Managed Workflows for Apache Airflow [Amazon MWAA], AWS Step Functions, AWS Glue workflows) • Building data pipelines for performance, availability, scalability, resiliency, and fault tolerance • Implementing and maintaining serverless workflows • Using notification services to send alerts (for example, Amazon Simple Notification Service [Amazon SNS], Amazon Simple Queue Service [Amazon SQS])
Apply programming concepts.	<p>- Knowledge of:</p>

Section	Objectives
	<ul style="list-style-type: none"> • Continuous integration and continuous delivery (CI/CD) (implementation, testing, and deployment of data pipelines) • SQL queries (for data source queries and data transformations) • Infrastructure as code (IaC) for repeatable deployments (for example, AWS Cloud Development Kit [AWS CDK], AWS CloudFormation) • Distributed computing • Data structures and algorithms (for example, graph data structures and tree data structures) • SQL query optimization <p>- Skills in:</p> <ul style="list-style-type: none"> • Optimizing code to reduce runtime for data ingestion and transformation • Configuring Lambda functions to meet concurrency and performance needs • Performing SQL queries to transform data (for example, Amazon Redshift stored procedures) • Structuring SQL queries to meet data pipeline requirements • Using Git commands to perform actions such as creating, updating, cloning, and branching repositories • Using the AWS Serverless Application Model (AWS SAM) to package and deploy serverless data pipelines (for example, Lambda functions, Step Functions, DynamoDB tables) • Using and mounting storage volumes from within Lambda functions
Data Store Management - 26%	
Choose a data store.	<p>- Knowledge of:</p> <ul style="list-style-type: none"> • Storage platforms and their characteristics • Storage services and configurations for specific performance demands • Data storage formats (for example, .csv, .txt, Parquet) • How to align data storage with data migration requirements • How to determine the appropriate storage solution for specific access patterns • How to manage locks to prevent access to data (for example, Amazon Redshift, Amazon RDS) <p>- Skills in:</p>

Section	Objectives
	<ul style="list-style-type: none"> • Implementing the appropriate storage services for specific cost and performance requirements (for example, Amazon Redshift, Amazon EMR, AWS Lake Formation, Amazon RDS, DynamoDB, Amazon Kinesis Data Streams, Amazon MSK) • Configuring the appropriate storage services for specific access patterns and requirements (for example, Amazon Redshift, Amazon EMR, Lake Formation, Amazon RDS, DynamoDB) • Applying storage services to appropriate use cases (for example, Amazon S3) • Integrating migration tools into data processing systems (for example, AWS Transfer Family) • Implementing data migration or remote access methods (for example, Amazon Redshift federated queries, Amazon Redshift materialized views, Amazon Redshift Spectrum)
Understand data cataloging systems.	- Knowledge of: <ul style="list-style-type: none"> • How to create a data catalog • Data classification based on requirements • Components of metadata and data catalogs - Skills in: <ul style="list-style-type: none"> • Using data catalogs to consume data from the data's source • Building and referencing a data catalog (for example, AWS Glue Data Catalog, Apache Hive metastore) • Discovering schemas and using AWS Glue crawlers to populate data catalogs • Synchronizing partitions with a data catalog • Creating new source or target connections for cataloging (for example, AWS Glue)
Manage the lifecycle of data.	- Knowledge of: <ul style="list-style-type: none"> • Appropriate storage solutions to address hot and cold data requirements • How to optimize the cost of storage based on the data lifecycle • How to delete data to meet business and legal requirements • Data retention policies and archiving strategies • How to protect data with appropriate resiliency and availability - Skills in:

Section	Objectives
	<ul style="list-style-type: none"> • Performing load and unload operations to move data between Amazon S3 and Amazon Redshift • Managing S3 Lifecycle policies to change the storage tier of S3 data • Expiring data when it reaches a specific age by using S3 Lifecycle policies • Managing S3 versioning and DynamoDB TTL
Design data models and schema evolution.	<ul style="list-style-type: none"> - Knowledge of: <ul style="list-style-type: none"> • Data modeling concepts • How to ensure accuracy and trustworthiness of data by using data lineage • Best practices for indexing, partitioning strategies, compression, and other data optimization techniques • How to model structured, semi-structured, and unstructured data • Schema evolution techniques - Skills in: <ul style="list-style-type: none"> • Designing schemas for Amazon Redshift, DynamoDB, and Lake Formation • Addressing changes to the characteristics of data • Performing schema conversion (for example, by using the AWS Schema Conversion Tool [AWS SCT] and AWS DMS Schema Conversion) • Establishing data lineage by using AWS tools (for example, Amazon SageMaker ML Lineage Tracking)
Data Operations and Support - 22%	
Automate data processing by using AWS services.	<ul style="list-style-type: none"> - Knowledge of: <ul style="list-style-type: none"> • How to maintain and troubleshoot data processing for repeatable business outcomes • API calls for data processing • Which services accept scripting (for example, Amazon EMR, Amazon Redshift, AWS Glue) - Skills in: <ul style="list-style-type: none"> • Orchestrating data pipelines (for example, Amazon MWAA, Step Functions) • Troubleshooting Amazon managed workflows • Calling SDKs to access Amazon features from code • Using the features of AWS services to process data (for example, Amazon EMR, Amazon Redshift, AWS Glue)

Section	Objectives
	<ul style="list-style-type: none"> • Consuming and maintaining data APIs • Preparing data transformation (for example, AWS Glue DataBrew) • Querying data (for example, Amazon Athena) • Using Lambda to automate data processing • Managing events and schedulers (for example, EventBridge)
Analyze data by using AWS services.	<ul style="list-style-type: none"> - Knowledge of: <ul style="list-style-type: none"> • Tradeoffs between provisioned services and serverless services • SQL queries (for example, SELECT statements with multiple qualifiers or JOIN clauses) • How to visualize data for analysis • When and how to apply cleansing techniques • Data aggregation, rolling average, grouping, and pivoting - Skills in: <ul style="list-style-type: none"> • Visualizing data by using AWS services and tools (for example, AWS Glue DataBrew, Amazon QuickSight) • Verifying and cleaning data (for example, Lambda, Athena, QuickSight, Jupyter Notebooks, Amazon SageMaker Data Wrangler) • Using Athena to query data or to create views • Using Athena notebooks that use Apache Spark to explore data
Maintain and monitor data pipelines.	<ul style="list-style-type: none"> - Knowledge of: <ul style="list-style-type: none"> • How to log application data • Best practices for performance tuning • How to log access to AWS services • Amazon Macie, AWS CloudTrail, and Amazon CloudWatch - Skills in: <ul style="list-style-type: none"> • Extracting logs for audits • Deploying logging and monitoring solutions to facilitate auditing and traceability • Using notifications during monitoring to send alerts • Troubleshooting performance issues • Using CloudTrail to track API calls • Troubleshooting and maintaining pipelines (for example, AWS Glue, Amazon EMR)

Section	Objectives
	<ul style="list-style-type: none"> • Using Amazon CloudWatch Logs to log application data (with a focus on configuration and automation) • Analyzing logs with AWS services (for example, Athena, Amazon EMR, Amazon OpenSearch Service, CloudWatch Logs Insights, big data application logs)
Ensure data quality.	<ul style="list-style-type: none"> - Knowledge of: <ul style="list-style-type: none"> • Data sampling techniques • How to implement data skew mechanisms • Data validation (data completeness, consistency, accuracy, and integrity) • Data profiling - Skills in: <ul style="list-style-type: none"> • Running data quality checks while processing the data (for example, checking for empty fields) • Defining data quality rules (for example, AWS Glue DataBrew) • Investigating data consistency (for example, AWS Glue DataBrew)
Data Security and Governance - 18%	
Apply authentication mechanisms.	<ul style="list-style-type: none"> - Knowledge of: <ul style="list-style-type: none"> • VPC security networking concepts • Differences between managed services and unmanaged services • Authentication methods (password-based, certificate-based, and role-based) • Differences between AWS managed policies and customer managed policies - Skills in: <ul style="list-style-type: none"> • Updating VPC security groups • Creating and updating IAM groups, roles, endpoints, and services • Creating and rotating credentials for password management (for example, AWS Secrets Manager) • Setting up IAM roles for access (for example, Lambda, Amazon API Gateway, AWS CLI, CloudFormation) • Applying IAM policies to roles, endpoints, and services (for example, S3 Access Points, AWS PrivateLink)
Apply authorization mechanisms.	<ul style="list-style-type: none"> - Knowledge of: <ul style="list-style-type: none"> • Authorization methods (role-based, policy-based, tag-based, and attributebased)

Section	Objectives
	<ul style="list-style-type: none"> • Principle of least privilege as it applies to AWS security • Role-based access control and expected access patterns • Methods to protect data from unauthorized access across services <p>- Skills in:</p> <ul style="list-style-type: none"> • Creating custom IAM policies when a managed policy does not meet the needs • Storing application and database credentials (for example, Secrets Manager, AWS Systems Manager Parameter Store) • Providing database users, groups, and roles access and authority in a database (for example, for Amazon Redshift) • Managing permissions through Lake Formation (for Amazon Redshift, Amazon EMR, Athena, and Amazon S3)
<p>Ensure data encryption and masking.</p>	<p>- Knowledge of:</p> <ul style="list-style-type: none"> • Data encryption options available in AWS analytics services (for example, Amazon Redshift, Amazon EMR, AWS Glue) • Differences between client-side encryption and server-side encryption • Protection of sensitive data • Data anonymization, masking, and key salting <p>- Skills in:</p> <ul style="list-style-type: none"> • Applying data masking and anonymization according to compliance laws or company policies • Using encryption keys to encrypt or decrypt data (for example, AWS Key Management Service [AWS KMS]) • Configuring encryption across AWS account boundaries • Enabling encryption in transit for data.
<p>Prepare logs for audit.</p>	<p>- Knowledge of:</p> <ul style="list-style-type: none"> • How to log application data • How to log access to AWS services • Centralized AWS logs <p>- Skills in:</p> <ul style="list-style-type: none"> • Using CloudTrail to track API calls • Using CloudWatch Logs to store application logs • Using AWS CloudTrail Lake for centralized logging queries

Section	Objectives
	<ul style="list-style-type: none"> • Analyzing logs by using AWS services (for example, Athena, CloudWatch Logs Insights, Amazon OpenSearch Service) • Integrating various AWS services to perform logging (for example, Amazon EMR in cases of large volumes of log data)
Understand data privacy and governance.	<ul style="list-style-type: none"> - Knowledge of: <ul style="list-style-type: none"> • How to protect personally identifiable information (PII) • Data sovereignty - Skills in: <ul style="list-style-type: none"> • Granting permissions for data sharing (for example, data sharing for Amazon Redshift) • Implementing PII identification (for example, Macie with Lake Formation) • Implementing data privacy strategies to prevent backups or replications of data to disallowed AWS Regions • Managing configuration changes that have occurred in an account (for example, AWS Config)

AWS DEA-C01 Sample Questions:

Question: 1

You have been tasked with migrating an on-premises MySQL database to Amazon Aurora PostgreSQL using AWS Database Migration Service (DMS). The stakeholder emphasizes that the source database must remain fully operational during the migration process.

Which of the following statements about DMS is accurate with respect to this scenario?

- a) AWS DMS only supports full-load migrations, which would require downtime for the source database.
- b) AWS DMS supports both full-load and continuous replication, allowing the source MySQL database to remain operational during migration.
- c) When using DMS, the target Amazon Aurora PostgreSQL instance cannot be accessed or queried until the migration is complete.
- d) AWS DMS requires the source MySQL database to be version 5.7 or higher for migrating to Amazon Aurora PostgreSQL.
- e) AWS DMS can convert the MySQL database schema directly to PostgreSQL without any manual intervention.

Answer: b

Question: 2

In data modeling, an entity-relationship diagram (ERD) is primarily used to:

- a) Transform raw data into structured data.
- b) Illustrate relationships between entities.
- c) Store data in a physical location.
- d) Cleanse dirty data.

Answer: b

Question: 3

In a data engineering pipeline, a company is using multiple applications and teams to access a shared Amazon S3 bucket. To streamline access and simplify permissions management for these different entities, which S3 feature should the company utilize?

- a) Enable multiple IAM roles, each corresponding to an application or team, granting access to the S3 bucket.
- b) Use S3 Access Points to create unique endpoints with tailored permissions for each application or team.
- c) Activate S3 Transfer Acceleration for the bucket to ensure fast and differentiated access for each application or team.
- d) Implement S3 Lifecycle policies for each application or team to manage their specific data access and retention.

Answer: b

Question: 4

Pivoting in SQL is mainly used to transform data from:

- a) One row to one column
- b) Only one column to one row
- c) Multiple columns into multiple rows
- d) Multiple rows into multiple columns

Answer: d

Question: 5

When processing large datasets using distributed computing frameworks, uneven distribution of data can lead to processing delays. What is this phenomenon commonly known as?

- a) Data skew
- b) Data partitioning
- c) Data shuffling
- d) Data fragmentation

Answer: a

Question: 6

A company is designing a data lake on Amazon S3. To ensure high performance when accessing the data, which best practice should the company adopt in organizing its data in the S3 bucket?

- a) Store all data files as a single large file and use AWS Lambda to parse required data segments.
- b) Use a flat structure by avoiding the creation of any prefix or "folder" hierarchy.
- c) Partition data based on commonly accessed attributes and use a consistent naming scheme for prefixes.
- d) Enable S3 Transfer Acceleration to ensure data is quickly accessible from any location.

Answer: c**Question: 7**

What is the primary purpose of data lineage in data engineering?

- a) To optimize query performance.
- b) To transform data formats.
- c) To trace the source and flow of data.
- d) To create visualizations.

Answer: c**Question: 8**

Company DEF has a strict security policy that mandates that all data at rest in Amazon S3 must be encrypted. They want to ensure that the encryption keys are managed by AWS, but they also want the flexibility to change the encryption keys when required.

Which of the following encryption methods best meets Company DEF's requirements?

- a) Server-Side Encryption with Customer-Provided Keys (SSE-C).
- b) Server-Side Encryption with Amazon S3 Managed Keys (SSE-S3).
- c) Server-Side Encryption with AWS Key Management Service (SSE-KMS).
- d) Client-Side Encryption with a client-side master key.

Answer: c**Question: 9**

For evolving schema and high compatibility, which data format should be chosen for downstream analytics?

- a) CSV
- b) JSON
- c) Parquet
- d) Avro

Answer: d

Question: 10

Which of the following best describes the type of data found in traditional relational databases?

- a) Structured data
- b) Unstructured data
- c) Semi-structured data
- d) Free-form data

Answer: a

Study Guide to Crack AWS Data Engineer Associate DEA-C01 Exam:

- Getting details of the DEA-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 DEA-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 DEA-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 DEA-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 DEA-C01 practice tests is must. Continuous practice will make you an expert in all syllabus areas.

Reliable Online Practice Test for DEA-C01 Certification

Make VMExam.com your best friend during your AWS Certified Data Engineer - Associate exam preparation. We provide authentic practice tests for the DEA-C01 exam. Experts design these online practice tests, so we can offer you an exclusive experience of taking the actual DEA-C01 exam. We guarantee you 100% success in your first exam attempt if you continue practicing regularly. Don't bother if you don't get 100% marks in initial practice exam attempts. Just utilize the result section to know your strengths and weaknesses and prepare according to that until you get 100% with our practice tests. Our evaluation makes you confident, and you can score high in the DEA-C01 exam.

Start Online practice of DEA-C01 Exam by visiting URL

<https://www.vmexam.com/aws/dea-c01-aws-certified-data-engineer-associate>