Exam Guide (MLA-C02)

AWS Certified Machine Learning Engineer - Associate

AWS Certified Machine Learning Engineer - Associate: Exam Guide (MLA-C02)

Table of Contents

AWS Certified Machine Learning Engineer - Associate (MLA-C02)

The AWS Certified Machine Learning Engineer - Associate (MLA-C02) exam validates a candidate's ability to build, operationalize, deploy, and maintain AI and ML solutions and pipelines by using the AWS Cloud. The exam validates ML engineering skills and the ability to work with traditional ML models and foundation models (FMs).

Topics

Introduction

The AWS Certified Machine Learning Engineer - Associate (MLA-C02) exam validates a candidate's ability to build, operationalize, deploy, and maintain AI and ML solutions and pipelines by using the AWS Cloud.

The exam validates ML engineering skills and the ability to work with traditional ML models and foundation models (FMs).

The exam also validates a candidate's ability to complete the following tasks:

- Ingest, transform, validate, and prepare data for AI and ML modeling.
- Select general modeling approaches, train models, tune hyperparameters, analyze model performance, and manage model versions.
- Choose deployment infrastructure and endpoints, provision compute resources, and configure auto scaling based on requirements.
- Set up continuous integration and continuous delivery (CI/CD) pipelines to automate the orchestration of AI and ML workflows.
- Build agentic workflows and maintain observability to optimize efficiency and cost.
- Monitor models, agentic workflows, data, and infrastructure to detect issues.
- Secure AI and ML systems and resources through access controls, compliance features, and best practices.

Target candidate description

The target candidate should have at least 1 year of experience using Amazon SageMaker AI, Amazon Bedrock, and other AWS services for ML engineering. The target candidate also should have at least 1 year of experience in a related role such as a backend software developer, DevOps developer, data engineer, or data scientist. The candidate should have experience with both traditional ML and generative AI (GenAI).

Recommended general IT knowledge

The target candidate should have the following general IT knowledge:

- Understanding of common ML algorithms and their use cases
- Understanding of FM capabilities, limitations, and common use cases
- Data engineering fundamentals, including knowledge of common data formats, ingestion, and transformation to work with ML data pipelines
- Knowledge of querying and transforming data
- Knowledge of software engineering best practices for modular, reusable code development, deployment, and debugging
- Familiarity with provisioning and monitoring cloud and on-premises ML resources

- Experience with CI/CD pipelines and infrastructure as code (IaC)

Recommended AWS knowledge

The target candidate should have the following AWS knowledge:

- Knowledge of SageMaker AI capabilities and algorithms for both traditional ML models and GenAI models
- Knowledge of Amazon Bedrock features and capabilities
- Knowledge of AWS data storage and data processing services to prepare data for modeling
- Familiarity with deploying applications and infrastructure on AWS
- Knowledge of monitoring tools for logging and troubleshooting ML systems
- Knowledge of AWS services for the automation and orchestration of CI/CD pipelines
- Understanding of AWS security best practices for identity and access management, encryption, and data protection

Job tasks that are out of scope for the target candidate

The following list contains job tasks that the target candidate is not expected to be able to perform. This list is non-exhaustive. The following tasks are out of scope for the exam:

- Designing and architecting full end-to-end AI and ML solutions
- Setting up best practices and guiding ML strategies
- Handling integration with a wide array of services or new tools and technologies
- Working deeply in two or more ML domains

Exam content

Question types

The exam contains one or more of the following question types:

- **Multiple choice:** Has one correct response and three incorrect responses (distractors).
- **Multiple response:** Has two or more correct responses out of five or more response options. You must select all the correct responses to receive credit for the question.

Unanswered questions are scored as incorrect. There is no penalty for guessing. The exam includes 50 questions that affect your score.

Unscored content

The exam includes 15 unscored questions that do not affect your score. AWS collects information about performance on these unscored questions to evaluate them for future use as scored questions. The unscored questions are not identified on the exam.

Exam results

The AWS Certified Machine Learning Engineer - Associate (MLA-C02) exam has a pass or fail designation.[1] The exam is scored against a minimum standard established by AWS professionals who follow certification industry best practices and guidelines.

> **Note**
>
> [1] Does not apply to the beta version of the exam. You can find more information about beta exams in general on the AWS Certification website.

Your results for the exam are reported as a scaled score of 100–1,000. The minimum passing score is 720. Your score shows how you performed on the exam as a whole and whether you passed. Scaled scoring models help equate scores across multiple exam forms that might have slightly different difficulty levels.

Your score report could contain a table of classifications of your performance at each section level. The exam uses a compensatory scoring model, which means that you do not need to achieve a passing score in each section. You need to pass only the overall exam.

Each section of the exam has a specific weighting, so some sections have more questions than other sections have. The table of classifications contains general information that highlights your strengths and weaknesses. Use caution when you interpret section-level feedback.

Content outline

This exam guide includes weightings, content domains, tasks, and skills for the exam. This guide does not provide a comprehensive list of the content on the exam.

The exam has the following content domains and weightings:

Content Domain 1: Data Preparation for ML and AI

Tasks

Task 1.1: Collect and store data.

- Skill 1.1.1: Extract data from data sources (for example, Amazon S3, Amazon EBS, Amazon EFS, Amazon RDS, Amazon DynamoDB, Amazon OpenSearch Service).
- Skill 1.1.2: Make storage decisions and configure storage services based on cost, performance, data structure, and data compliance.
- Skill 1.1.3: Troubleshoot and debug data ingestion and storage issues that involve capacity and scalability.
- Skill 1.1.4: Use AWS streaming data sources to ingest data (for example, by using Amazon Kinesis, Apache Flink, Apache Kafka).
- Skill 1.1.5: Ingest from and write by using appropriate data formats (for example, Apache Parquet, JSON, CSV, ORC) based on data access patterns.
- Skill 1.1.6: Merge data from multiple sources (for example, by using programming techniques, AWS Glue, Apache Spark).
- Skill 1.1.7: Configure scalable vector databases for AI applications (for example, OpenSearch Service, Amazon RDS with pgvector, Amazon S3) based on specifications.
- Skill 1.1.8: Ingest and store diverse data types (for example, text, images, audio) for AI and ML applications.

- Skill 1.1.9: Ingest data into SageMaker Feature Store.

Task 1.2: Perform data transformation, feature engineering, and pre-processing.

- Skill 1.2.1: Transform data by using AWS tools (for example, AWS Glue, AWS Glue DataBrew, Spark on Amazon EMR, SageMaker Data Wrangler).
- Skill 1.2.2: Create and manage features by using AWS tools (for example, SageMaker Feature Store).
- Skill 1.2.3: Transform streaming data (for example, by using AWS Lambda, Spark).
- Skill 1.2.4: Perform feature engineering (for example, scaling, standardization, feature splitting, binning, log transformation, normalization).
- Skill 1.2.5: Configure and use embedding models to transform text and image data into numerical representations.
- Skill 1.2.6: Apply advanced text pre-processing techniques (for example, tokenization, domain-specific augmentation).
- Skill 1.2.7: Prepare documents for Retrieval Augmented Generation (RAG) applications (for example, chunking strategies, metadata extraction).
- Skill 1.2.8: Mask, redact, and anonymize data.
- Skill 1.2.9: Prepare data for FM fine-tuning, continuous pre-training, and model distillation.

Task 1.3: Validate data quality and manage bias.

- Skill 1.3.1: Validate data quality (for example, by using DataBrew, AWS Glue Data Quality).
- Skill 1.3.2: Label and annotate data.
- Skill 1.3.3: Identify and mitigate sources of bias in data by using AWS tools and techniques (for example, dataset splitting, shuffling, augmentation).
- Skill 1.3.4: Optimize multimodal data distributions by applying bias metrics across numeric, text, and image assets.
- Skill 1.3.5: Resolve class imbalance in numeric, text, and image datasets.
- Skill 1.3.6: Validate AI training data integrity (for example, prompt-response pair validation, content safety screening).
- Skill 1.3.7: Clean data (for example, by detecting outliers, imputing missing data, deduplication).

Content Domain 2: ML Model and Foundation Model (FM) Development

Tasks

Task 2.1: Choose appropriate modeling approaches for ML and AI solutions.

- Skill 2.1.1: Evaluate and select appropriate FMs from Amazon Bedrock based on task requirements and performance criteria.
- Skill 2.1.2: Identify fine-tuning strategies for pre-trained FMs to meet business needs.
- Skill 2.1.3: Compare and select appropriate ML models, generative AI (GenAI) models, algorithms, and solution templates to meet business needs (for example, interpretability, domain-specific performance, latency).
- Skill 2.1.4: Evaluate tradeoffs between custom solutions, managed services, pre-trained models, and FMs to meet business needs.
- Skill 2.1.5: Select Retrieval Augmented Generation (RAG) architecture patterns based on use case requirements.
- Skill 2.1.6: Assess tradeoffs between ML model performance, training time, and cost.
- Skill 2.1.7: Assess tradeoffs between AI model performance, latency, and cost.
- Skill 2.1.8: Apply AWS AI services to solve specific business problems (for example, Amazon Textract, Amazon Rekognition, Amazon Comprehend, Amazon Transcribe).

Task 2.2: Train, fine-tune, and customize models for ML and AI solutions.

- Skill 2.2.1: Apply Amazon SageMaker AI built-in algorithms and common ML libraries.
- Skill 2.2.2: Configure SageMaker AI script mode with supported frameworks for simplicity and performance.

- Skill 2.2.3: Implement hyperparameter optimization (for example, SageMaker AI automatic model tuning [AMT]).
- Skill 2.2.4: Implement training time reduction techniques (for example, early stopping, distributed training).
- Skill 2.2.5: Prevent model overfitting, underfitting, and catastrophic forgetting.
- Skill 2.2.6: Combine multiple ML models to improve performance or reduce cost.
- Skill 2.2.7: Adjust fundamental hyperparameters (for example, epoch, steps, batch size).
- Skill 2.2.8: Apply customization techniques for AI solutions (for example, task-specific prompt engineering, fine-tuning).
- Skill 2.2.9: Optimize retrieval components and embedding models.

Task 2.3: Analyze and evaluate the performance of ML and AI systems.

- Skill 2.3.1: Perform reproducible experiments (for example, by using MLflow on SageMaker AI, Amazon Bedrock evaluations, Amazon Bedrock Prompt Management).
- Skill 2.3.2: Create model performance baselines and implement drift detection.
- Skill 2.3.3: Compare the performance of shadow variants to production variants.
- Skill 2.3.4: Explain model outputs.
- Skill 2.3.5: Debug model convergence issues.
- Skill 2.3.6: Apply comprehensive model evaluation techniques for traditional ML and GenAI models.
- Skill 2.3.7: Implement integrated human evaluation frameworks (for example, human-in-the-loop workflows, text generation quality assessment).
- Skill 2.3.8: Apply natural language processing (NLP) evaluation metrics (for example, bilingual evaluation understudy [BLEU], Recall-Oriented Understudy for Gisting Evaluation [ROUGE], BERTScore, semantic similarity).
- Skill 2.3.9: Perform AI evaluation (for example, model output assessment, content quality validation, bias detection, LLM-as-a-judge frameworks).
- Skill 2.3.10: Configure RAG system monitoring, including retrieval accuracy assessment.

Content Domain 3: Deployment and Orchestration of ML and AI Workflows

Tasks

Task 3.1: Manage deployment infrastructure for ML and AI model types.

- Skill 3.1.1: Select appropriate compute environments and deployment targets.
- Skill 3.1.2: Select deployment orchestrators and multi-model or multi-container deployment strategies.
- Skill 3.1.3: Select model inference strategies (for example, real-time and batch processing).
- Skill 3.1.4: Evaluate and select appropriate foundation model (FM) deployment options.
- Skill 3.1.5: Deploy models that were built outside of AWS into AWS environments (for example, Amazon SageMaker AI, Amazon Bedrock Custom Model Import).
- Skill 3.1.6: Deploy and configure agents for specific tasks, integration with other services and tools, and agent communication protocols.
- Skill 3.1.7: Configure FM deployment, model hosting, and resource allocation.
- Skill 3.1.8: Apply Retrieval Augmented Generation (RAG) system configurations (for example, retrieval strategies, reranking).

Task 3.2: Provision and configure resources for ML and AI workloads based on existing architecture and requirements.

- Skill 3.2.1: Optimize resource provisioning between on-demand and provisioned resources for performance and cost efficiency.

- Skill 3.2.2: Automate compute resource provisioning with integrated communication between stacks and orchestration services.
- Skill 3.2.3: Build and maintain containers for ML and AI workloads.
- Skill 3.2.4: Configure SageMaker AI endpoints within VPC network environments.
- Skill 3.2.5: Deploy and host models programmatically (for example, by using the SageMaker AI SDK for Python, AWS CLI, Boto3).
- Skill 3.2.6: Select specific metrics for auto scaling implementations.
- Skill 3.2.7: Create and manage Amazon Bedrock knowledge bases with vector database configurations, document indexing, and retrieval optimization.
- Skill 3.2.8: Implement retrieval pipelines to meet business needs.
- Skill 3.2.9: Implement agent state management systems.
- Skill 3.2.10: Implement AI-specific resource scaling for GPU workloads.
- Skill 3.2.11: Deploy agentic workflow infrastructure.

Task 3.3: Implement automated orchestration and continuous integration and continuous delivery (CI/CD) pipelines for MLOps and AI workloads.

- Skill 3.3.1: Implement automated deployment strategies and rollback actions.
- Skill 3.3.2: Configure and troubleshoot AWS CodeBuild, AWS CodeCommit, AWS CodeDeploy, AWS CodePipeline, and AWS CodeConnections.
- Skill 3.3.3: Configure training and inference jobs.
- Skill 3.3.4: Configure automated testing strategies within CI/CD pipelines for traditional ML and AI workloads.
- Skill 3.3.5: Build and integrate mechanisms to re-train models.
- Skill 3.3.6: Manage model versions for repeatability and audits (for example, SageMaker Model Registry, MLflow on SageMaker AI).
- Skill 3.3.7: Manage prompts (for example, Amazon Bedrock Prompt Management).
- Skill 3.3.8: Implement automated agent deployment pipelines and agent version management.
- Skill 3.3.9: Implement AI model testing frameworks, including prompt testing.
- Skill 3.3.10: Configure FM deployment automation with fine-tuned model versioning.

- Skill 3.3.11: Configure AI-specific pipeline orchestration for RAG system updates and knowledge base refresh cycles.

Content Domain 4: Operating, Monitoring, and Securing ML and AI Solutions

Tasks

Task 4.1: Monitor ML and AI model inference and performance.

- Skill 4.1.1: Monitor model performance in production by using Amazon CloudWatch generative AI observability, Amazon Bedrock Model Evaluation, and drift detection pipelines.
- Skill 4.1.2: Monitor workflows to detect anomalies or errors in data processing or model inference.
- Skill 4.1.3: Detect changes in data distribution that can affect model performance.
- Skill 4.1.4: Monitor model performance in production by using A/B testing.
- Skill 4.1.5: Monitor and automate the management of agent performance and coordination (for example, coordination failure detection, truncated streaming, tool failures).
- Skill 4.1.6: Configure AI-specific performance monitoring for foundation models (FMs), such as Amazon Bedrock evaluations.

Task 4.2: Optimize and manage ML and AI infrastructure costs and performance.

- Skill 4.2.1: Select inference instance families to optimize performance and cost.
- Skill 4.2.2: Configure and use tools to troubleshoot and analyze resources (for example, Amazon CloudWatch, Amazon Bedrock AgentCore Observability, AWS X-Ray).
- Skill 4.2.3: Set up dashboards to monitor performance metrics.
- Skill 4.2.4: Optimize capacity for cost, performance, and reliability.

- Skill 4.2.5: Optimize costs and set cost quotas by using appropriate cost management tools.
- Skill 4.2.6: Optimize infrastructure costs by selecting purchasing options.
- Skill 4.2.7: Evaluate cost implications of using FMs for inference in production.
- Skill 4.2.8: Monitor agent resource consumption patterns.
- Skill 4.2.9: Manage FM inference costs with usage optimization.
- Skill 4.2.10: Monitor AI-specific cost patterns (for example, token usage optimization, embedding computation costs, vector database storage optimization).

Task 4.3: Secure ML and AI workloads and model endpoints.

- Skill 4.3.1: Secure continuous integration and continuous delivery (CI/CD) pipelines by checking for code and image vulnerabilities (for example, by using Amazon CodeGuru, Amazon Inspector).
- Skill 4.3.2: Configure least privilege access to ML and AI artifacts.
- Skill 4.3.3: Configure IAM policies and roles for users and applications in ML and AI systems.
- Skill 4.3.4: Configure comprehensive monitoring, auditing, compliance, and logging for ML and AI systems (for example, AWS CloudTrail, AWS Config).
- Skill 4.3.5: Troubleshoot and debug security issues in ML and AI systems.
- Skill 4.3.6: Create VPCs, subnets, and security groups to securely isolate ML and AI systems.
- Skill 4.3.7: Identify and mitigate security risks and vulnerabilities in ML and AI systems.
- Skill 4.3.8: Select the appropriate credential type to access FMs (for example, Amazon Bedrock API keys, IAM credentials).
- Skill 4.3.9: Implement safeguards and sensitive data protection to meet application requirements and responsible AI policies (for example, by using Amazon Bedrock Guardrails).

Mentions of AWS services on the exam

AWS Certification is reducing the reading load on this exam by using official short names of well-known AWS service names that contain abbreviations or parenthetical information. For example, Amazon Simple Notification Service (Amazon SNS) appears on the exam as Amazon SNS.

- The Help feature in the exam (available for every question) contains the list of the short AWS service names and their corresponding full names.

- You can consult AWS Service Names on the AWS Certification website for the list of services that appear as their short names on the exam. Any services that are on the list but that are out of scope for the exam will not appear on the exam.

Note: Not every abbreviation is fully spelled out on the exam or available in the Help feature. The official full name for some AWS services includes an abbreviation that is never expanded (for example, Amazon API Gateway, Amazon EMR). The exam also might contain other abbreviations that the target audience is expected to know.

In-scope AWS services and features

The following list contains AWS services and features that are in scope for the exam. This list is non-exhaustive and is subject to change. AWS offerings appear in categories that align with the offerings' primary functions:

Topics

Analytics

- Amazon Athena

- Amazon Data Firehose
- Amazon EMR
- AWS Glue
- AWS Glue DataBrew
- AWS Glue Data Quality
- Amazon Kinesis
- AWS Lake Formation
- Amazon Managed Service for Apache Flink
- Amazon OpenSearch Service
- Amazon Quick
- Amazon Redshift

Application Integration

- Amazon EventBridge
- Amazon Managed Workflows for Apache Airflow (Amazon MWAA)
- Amazon SNS
- Amazon SQS
- AWS Step Functions

Cloud Financial Management

- AWS Billing and Cost Management
- AWS Budgets
- AWS Cost Explorer

Compute

- AWS Batch
- Amazon EC2
- AWS Lambda

Containers

- Amazon ECR
- Amazon ECS
- Amazon EKS

Database

- Amazon DocumentDB
- Amazon DynamoDB
- Amazon ElastiCache
- Amazon Neptune
- Amazon RDS

Developer Tools

- AWS CDK
- AWS CodeArtifact
- AWS CodeBuild
- AWS CodeCommit
- AWS CodeConnections
- AWS CodeDeploy
- AWS CodePipeline
- AWS X-Ray

Machine Learning and Artificial Intelligence

- Amazon Bedrock
- Amazon Bedrock AgentCore
- Amazon CodeGuru
- Amazon Comprehend
- Amazon Comprehend Medical

- Amazon DevOps Guru
- AWS HealthLake
- Amazon Lex
- Amazon Personalize
- Amazon Polly
- Amazon Rekognition
- Amazon SageMaker AI
- Amazon Textract
- Amazon Transcribe
- Amazon Translate

Management and Governance

- AWS Auto Scaling
- AWS CloudFormation
- AWS CloudTrail
- Amazon CloudWatch
- AWS Compute Optimizer
- AWS Config
- AWS Organizations
- AWS Service Catalog
- AWS Systems Manager
- AWS Trusted Advisor

Media

- Amazon Kinesis Video Streams

Migration and Transfer

- AWS DataSync

Networking and Content Delivery

- Amazon API Gateway
- Amazon CloudFront
- AWS Direct Connect
- Amazon VPC

Security, Identity, and Compliance

- IAM
- Amazon Inspector
- AWS KMS
- Amazon Macie
- AWS Secrets Manager

Storage

- Amazon EBS
- Amazon EFS
- Amazon FSx
- Amazon S3
- Amazon S3 Glacier
- AWS Storage Gateway

Out-of-scope AWS services and features

The following list contains AWS services and features that are out of scope for the exam. This list is non-exhaustive and is subject to change. AWS offerings that are entirely unrelated to the target job roles for the exam are excluded from this list:

Topics

- Analytics
- Application Integration

- Business Applications
- Cloud Financial Management
- Compute
- Containers
- Customer Enablement
- Developer Tools
- End User Computing
- Frontend Web and Mobile
- Internet of Things (IoT)
- Machine Learning and Artificial Intelligence
- Management and Governance
- Media
- Migration and Transfer
- Network and Content Delivery
- Security, Identity, and Compliance
- Storage

Analytics

- AWS Clean Rooms
- Amazon DataZone
- Amazon FinSpace

Application Integration

- Amazon AppFlow
- Amazon MQ
- Amazon Simple Workflow Service (Amazon SWF)

Business Applications

- Amazon Connect

- Amazon Honeycode
- Amazon SES
- AWS Supply Chain
- AWS Wickr
- Amazon WorkDocs
- Amazon WorkMail

Cloud Financial Management

- AWS Application Cost Profiler

Compute

- AWS App Runner
- AWS Elastic Beanstalk
- Amazon Lightsail
- AWS Outposts

Containers

- Red Hat OpenShift Service on AWS (ROSA)

Customer Enablement

- AWS Activate for startups
- AWS IQ
- AWS re:Post Private

Developer Tools

- AWS Infrastructure Composer
- AWS CloudShell

- Amazon CodeCatalyst
- AWS Fault Injection Service

End User Computing

- Amazon AppStream 2.0
- Amazon WorkSpaces
- Amazon WorkSpaces Secure Browser
- Amazon WorkSpaces Thin Client

Frontend Web and Mobile

- AWS Amplify
- AWS Device Farm
- Amazon Location Service

Internet of Things (IoT)

- FreeRTOS
- AWS IoT 1-Click
- AWS IoT Core
- AWS IoT Device Defender
- AWS IoT Device Management
- AWS IoT Events
- AWS IoT FleetWise
- AWS IoT Greengrass
- AWS IoT RoboRunner
- AWS IoT SiteWise
- AWS IoT TwinMaker

Machine Learning and Artificial Intelligence

- AWS DeepRacer
- AWS HealthImaging
- AWS HealthOmics

Management and Governance

- AWS AppConfig
- AWS Control Tower
- AWS Launch Wizard
- AWS License Manager
- Amazon Managed Grafana
- AWS Proton
- AWS Resilience Hub
- AWS Resource Explorer
- AWS Telco Network Builder
- AWS User Notifications

Media

- Amazon Elastic Transcoder
- AWS Elemental Appliances and Software
- AWS Elemental MediaConnect
- AWS Elemental MediaConvert
- AWS Elemental MediaLive
- AWS Elemental MediaPackage
- AWS Elemental MediaStore
- AWS Elemental MediaTailor
- Amazon Interactive Video Service (Amazon IVS)
- Amazon Nimble Studio

Migration and Transfer

- AWS Application Discovery Service
- AWS Application Migration Service
- AWS Mainframe Modernization

Network and Content Delivery

- AWS App Mesh
- AWS Cloud Map
- AWS Global Accelerator
- AWS Private 5G
- Amazon Route 53
- Amazon Route 53 Application Recovery Controller
- Amazon VPC IP Address Manager (IPAM)

Security, Identity, and Compliance

- AWS Artifact
- AWS Audit Manager
- AWS Certificate Manager (ACM)
- AWS CloudHSM
- Amazon Cognito
- Amazon Detective
- AWS Firewall Manager
- Amazon GuardDuty
- AWS Payment Cryptography
- AWS Private Certificate Authority
- AWS Resource Access Manager (AWS RAM)
- AWS Security Hub
- AWS Security Hub CSPM
- AWS Shield

- AWS Signer
- Amazon Verified Permissions
- AWS WAF

Storage

- AWS Elastic Disaster Recovery

Comparison of MLA-C01 and MLA-C02

Side-by-side comparison

The following table shows the domains and the percentage of scored questions in each domain for the MLA-C01 exam (in use until September 28, 2026) and the MLA-C02 exam (in use beginning September 29, 2026).

MLA-C01 Domain	MLA-C02 Content Domain
Domain 1: Data Preparation for Machine Learning (ML) (28%)	Content Domain 1: Data Preparation for ML and AI (28% of scored content)
Domain 2: ML Model Development (26%)	Content Domain 2: ML Model and Foundatio n Model (FM) Development (24% of scored content)
Domain 3: Deployment and Orchestration of ML Workflows (22%)	Content Domain 3: Deployment and Orchestra tion of ML and AI Workflows (24% of scored content)
Domain 4: ML Solution Monitoring, Maintenan ce, and Security (24%)	Content Domain 4: Operating, Monitoring, and Securing ML and AI Solutions (24% of scored content)

Additions of content for MLA-C02

In Task 1.1, the following content was added:

- 1.1.7 Configure scalable vector databases for AI applications (for example, OpenSearch Service, Amazon RDS with pgvector, Amazon S3) based on specifications.
- 1.1.8 Ingest and store diverse data types (for example, text, images, audio) for AI and ML applications.

In Task 1.2, the following content was added:

- 1.2.5 Configure and use embedding models to transform text and image data into numerical representations.
- 1.2.6 Apply advanced text pre-processing techniques (for example, tokenization, domain-specific augmentation).
- 1.2.7 Prepare documents for Retrieval Augmented Generation (RAG) applications (for example, chunking strategies, metadata extraction).
- 1.2.8 Mask, redact, and anonymize data.
- 1.2.9 Prepare data for FM fine-tuning, continuous pre-training, and model distillation.

In Task 1.3, the following content was added:

- 1.3.4 Optimize multimodal data distributions by applying bias metrics across numeric, text, and image assets.
- 1.3.6 Validate AI training data integrity (for example, prompt-response pair validation, content safety screening).
- 1.3.7 Clean data (for example, by detecting outliers, imputing missing data, deduplication).

In Task 2.1, the following content was added:

- 2.1.1 Evaluate and select appropriate FMs from Amazon Bedrock based on task requirements and performance criteria.
- 2.1.2 Identify fine-tuning strategies for pre-trained FMs to meet business needs.
- 2.1.4 Evaluate tradeoffs between custom solutions, managed services, pre-trained models, and FMs to meet business needs.
- 2.1.5 Select Retrieval Augmented Generation (RAG) architecture patterns based on use case requirements.
- 2.1.7 Assess tradeoffs between AI model performance, latency, and cost.

In Task 2.2, the following content was added:

- 2.2.8 Apply customization techniques for AI solutions (for example, task-specific prompt engineering, fine-tuning).
- 2.2.9 Optimize retrieval components and embedding models.

In Task 2.3, the following content was added:

- 2.3.7 Implement integrated human evaluation frameworks (for example, human-in-the-loop workflows, text generation quality assessment).
- 2.3.8 Apply natural language processing (NLP) evaluation metrics (for example, bilingual evaluation understudy [BLEU], Recall-Oriented Understudy for Gisting Evaluation [ROUGE], BERTScore, semantic similarity).
- 2.3.9 Perform AI evaluation (for example, model output assessment, content quality validation, bias detection, LLM-as-a-judge frameworks).
- 2.3.10 Configure RAG system monitoring, including retrieval accuracy assessment.

In Task 3.1, the following content was added:

- 3.1.4 Evaluate and select appropriate foundation model (FM) deployment options.
- 3.1.5 Deploy models that were built outside of AWS into AWS environments (for example, Amazon SageMaker AI, Amazon Bedrock Custom Model Import).
- 3.1.6 Deploy and configure agents for specific tasks, integration with other services and tools, and agent communication protocols.
- 3.1.7 Configure FM deployment, model hosting, and resource allocation.
- 3.1.8 Apply Retrieval Augmented Generation (RAG) system configurations (for example, retrieval strategies, reranking).

In Task 3.2, the following content was added:

- 3.2.7 Create and manage Amazon Bedrock knowledge bases with vector database configurations, document indexing, and retrieval optimization.
- 3.2.8 Implement retrieval pipelines to meet business needs.
- 3.2.9 Implement agent state management systems.
- 3.2.10 Implement AI-specific resource scaling for GPU workloads.

- 3.2.11 Deploy agentic workflow infrastructure.

In Task 3.3, the following content was added:

- 3.3.7 Manage prompts (for example, Amazon Bedrock Prompt Management).
- 3.3.8 Implement automated agent deployment pipelines and agent version management.
- 3.3.9 Implement AI model testing frameworks, including prompt testing.
- 3.3.10 Configure FM deployment automation with fine-tuned model versioning.
- 3.3.11 Configure AI-specific pipeline orchestration for RAG system updates and knowledge base refresh cycles.

In Task 4.1, the following content was added:

- 4.1.5 Monitor and automate the management of agent performance and coordination (for example, coordination failure detection, truncated streaming, tool failures).
- 4.1.6 Configure AI-specific performance monitoring for foundation models (FMs), such as Amazon Bedrock evaluations.

In Task 4.2, the following content was added:

- 4.2.7 Evaluate cost implications of using FMs for inference in production.
- 4.2.8 Monitor agent resource consumption patterns.
- 4.2.9 Manage FM inference costs with usage optimization.
- 4.2.10 Monitor AI-specific cost patterns (for example, token usage optimization, embedding computation costs, vector database storage optimization).

In Task 4.3, the following content was added:

- 4.3.1 Secure continuous integration and continuous delivery (CI/CD) pipelines by checking for code and image vulnerabilities (for example, by using Amazon CodeGuru, Amazon Inspector).
- 4.3.8 Select the appropriate credential type to access FMs (for example, Amazon Bedrock API keys, IAM credentials).
- 4.3.9 Implement safeguards and sensitive data protection to meet application requirements and responsible AI policies (for example, by using Amazon Bedrock Guardrails).

Deletions of content for MLA-C02

In Task 1.3, the following content was removed:

- Configuring data to load into the model training resource (for example, Amazon EFS, Amazon FSx)

In Task 2.2, the following content was removed:

- Using custom datasets to fine-tune pre-trained models (for example, Amazon Bedrock, SageMaker JumpStart)
- Reducing model size (for example, by altering data types, pruning, updating feature selection, compression)

In Task 3.1, the following content was removed:

- Methods to optimize models on edge devices (for example, SageMaker Neo)

In Task 3.2, the following content was removed:

- Bring your own container (BYOC) with SageMaker

In Task 4.2, the following content was removed:

- Monitoring infrastructure (for example, by using Amazon EventBridge events)
- Troubleshooting capacity concerns that involve cost and performance (for example, provisioned concurrency, service quotas, auto scaling)

Recategorizations of content for MLA-C02

The following content reorganizations have occurred in the transition from MLA-C01 to MLA-C02:

The following task statements have been recategorized:

MLA-C01 Task Statement 1.1 is mapped to the following task in MLA-C02:

- 1.1 Collect and store data.

MLA-C01 Task Statement 1.2 is mapped to the following task in MLA-C02:

- 1.2 Perform data transformation, feature engineering, and pre-processing.

MLA-C01 Task Statement 1.3 is mapped to the following task in MLA-C02:

- 1.3 Validate data quality and manage bias.

MLA-C01 Task Statement 2.1 is mapped to the following task in MLA-C02:

- 2.1 Choose appropriate modeling approaches for ML and AI solutions.

MLA-C01 Task Statement 2.2 is mapped to the following task in MLA-C02:

- 2.2 Train, fine-tune, and customize models for ML and AI solutions.

MLA-C01 Task Statement 2.3 is mapped to the following task in MLA-C02:

- 2.3 Analyze and evaluate the performance of ML and AI systems.

MLA-C01 Task Statement 3.1 is mapped to the following task in MLA-C02:

- 3.1 Manage deployment infrastructure for ML and AI model types.

MLA-C01 Task Statement 3.2 is mapped to the following task in MLA-C02:

- 3.2 Provision and configure resources for ML and AI workloads based on existing architecture and requirements.

MLA-C01 Task Statement 3.3 is mapped to the following task in MLA-C02:

- 3.3 Implement automated orchestration and continuous integration and continuous delivery (CI/CD) pipelines for MLOps and AI workloads.

MLA-C01 Task Statement 4.1 is mapped to the following task in MLA-C02:

- 4.1 Monitor ML and AI model inference and performance.

MLA-C01 Task Statement 4.2 is mapped to the following task in MLA-C02:

- 4.2 Optimize and manage ML and AI infrastructure costs and performance.

MLA-C01 Task Statement 4.3 is mapped to the following task in MLA-C02:

- 4.3 Secure ML and AI workloads and model endpoints.

Survey

How useful was this exam guide? Let us know by taking our survey.