AWS Certified AI Practitioner

AIF-C01 Visual Study Guide

Foundational exam covering AI, ML, generative AI, responsible AI, and governance on AWS.

aws Certified AI Practitioner FOUNDATIONAL

Exam at a glance

Questions	65 questions
Duration	90 minutes
Question types	Multiple choice, multiple response, ordering, matching
Cost	$100 USD
Level	Foundational
Intended audience	Business professionals, cloud practitioners, analysts, product managers, and anyone learning AI on AWS

Domain weights

Fundamentals of AI and ML	**20%**
Fundamentals of GenAI	**24%**
Applications of Foundation Models	**28%**
Guidelines for Responsible AI	**14%**
Security, Compliance, and Governance for AI Solutions	**14%**

AI learning map on AWS

Concepts → **Use cases** → **Prompting** → **Foundation models** → **Responsible AI** → **Security**

- **AI / ML / DL**
- **Prediction, vision, language**
- **Prompts + guardrails**
- **Amazon Bedrock**
- **Fairness + transparency**
- **IAM + KMS**

High-yield mindset

- **Start with business value.** Choose AI solutions that solve a real business problem.
- **Know the common AWS AI services.** Understand when to use Bedrock, SageMaker AI, Lex, Rekognition, Comprehend, Textract, and Transcribe.
- **Separate traditional ML from GenAI.** Prediction and classification are different from generative use cases.
- **Expect questions about responsible AI.** Fairness, transparency, privacy, and human oversight matter.
- **Read for service-selection clues.** Pay attention to phrases about chatbots, document extraction, sentiment, image analysis, and foundation models.

Domain 1 — Fundamentals of AI and ML (20%)

AIF-C01 Visual Study Guide

Understand core AI concepts, common ML approaches, and practical business use cases.

aws Certified AI Practitioner FOUNDATIONAL

AI/ML lifecycle

Define problem	Collect data	Train model	Evaluate	Deploy	Infer
Clarify the goal and success criteria.	Gather and prepare relevant, quality data.	Teach the model patterns from the data.	Measure performance (e.g., accuracy, precision, recall).	Put the model into a running environment (e.g., on AWS).	Use the deployed model to make predictions on new data.

Know the core terms

AI	Broad field of machines performing tasks that normally require human intelligence.
ML	Systems learn patterns from data to make predictions or decisions.
Deep learning	ML using multi-layer neural networks, especially useful for images, audio, and language.
NLP	Techniques for understanding and generating human language.
Training	Process of teaching a model from historical data.
Inference	Using a trained model to make predictions on new data.

Learning approaches and examples

Approach	What it means	Examples
Supervised learning	Uses labeled data.	spam detection, forecasting, churn prediction
Unsupervised learning	Finds patterns without labels.	clustering, segmentation, anomaly discovery
Reinforcement learning	Learns through rewards and penalties.	robotics, game play, dynamic optimization

Use-case chooser

Classification	Choose a category.	approve / deny
Regression	Predict a number.	sales forecast
Clustering	Group similar items.	customer segments
Recommendation	Suggest relevant items.	next best product
Computer vision / language	Understand images or text.	document analysis or sentiment

What this domain often tests

1. **Match the business problem to the right AI approach.**
2. **Recognize the difference between training and inference.**
3. **Distinguish AI, ML, and deep learning.**
4. **Identify practical use cases for language and vision.**
5. **Choose the simplest approach that meets the need.**

Domain 2 — Fundamentals of GenAI (24%)

AIF-C01 Visual Study Guide

Understand how generative AI works, where it fits, and its practical limitations.

How GenAI works

Prompt	Tokenization	Foundation model	Generated output	Human review
Provide an instruction or question.	Break the input into tokens (model-readable units).	Use a pre-trained model to predict the next tokens.	Produce a relevant and coherent response.	Check for accuracy, relevance, and responsible use.

Core GenAI concepts

Prompt	Instructions given to a model.
Tokens	Small pieces of text the model processes.
Context window	How much input and conversation the model can consider.
Temperature	Controls how creative or deterministic the output is.
Hallucination	Confident but incorrect generated content.
Embeddings	Numeric representations used for semantic search and retrieval.

Capabilities vs limitations

Capabilities

- **Summarization —** Condense long content into key points.
- **Drafting —** Create and refine content (e.g., emails, reports).
- **Question answering —** Provide relevant, contextual answers.
- **Code assistance —** Generate, explain, and improve code.
- **Translation —** Convert content between languages.
- **Brainstorming —** Generate and explore new ideas and options.

Limitations

- **Hallucinations —** Can produce confident but incorrect information.
- **Outdated knowledge —** May not have the latest information.
- **Prompt sensitivity —** Output can vary significnatly with wording or structure.
- **Privacy concerns —** Risk of exposing sensitive or confidential data.
- **Inconsistency —** Can give different answers to the same question.

Prompting patterns

Zero-shot	Ask directly without examples.
Few-shot	Provide examples in the prompt.
Role prompting	Tell the model who it should act like.
Step-by-step prompting	Encourage structured reasoning (e.g., think step by step).
Grounded prompting	Supply trusted context or retrieved content.

AWS building blocks for GenAI

Amazon Bedrock	Access foundation models through a managed service.
AWS Trainium	Hardware for training AI models.
AWS Inferentia	Hardware for cost-efficient inference.
Amazon SageMaker AI	Build, train, and deploy ML solutions.
Vector database / knowledge store	Support retrieval for grounded answers.

Domain 3 — Applications of Foundation Models (28%)

AIF-C01 Visual Study Guide

Know common foundation-model use cases, solution patterns, and service-selection clues.

Foundation-model app pattern

Choose model	Design prompt	Ground with data	Apply guardrails	Evaluate	Deploy
Select a foundation model that fits the use case.	Craft a clear, context-rich prompt.	Bring in relevant, trusted data (e.g., RAG).	Enforce safety, security, and responsible AI controls.	Check output quality, accuracy, and usefulness.	Run in production with monitoring and feedback.

What foundation models are good at

Summarization	Condense long content into key points.
Content generation	Create new text, images, or other media from a prompt.
Question answering	Provide answers from general knowledge or grounded data.
Classification / extraction from text	Assign categories or extract key information (e.g., entities, fields).
Conversational assistance	Support natural, multi-turn interactions to help users get things done.

RAG vs fine-tuning vs pretraining

Approach	What it does	Best for
Retrieval-augmented generation (RAG)	Add external knowledge at inference time.	Fresh or private information.
Fine-tuning	Adapt a model with additional labeled examples.	Style or task specialization.
Pretraining	Train a model on huge general datasets.	Foundational model creation.

Service chooser

Foundation-model apps	**Amazon Bedrock** — managed access to foundation models to build generative AI applications.
Semantic search / retrieval	**Embeddings + vector store** — convert content to embeddings and index for semantic retrieval.
Knowledge-grounded Q&A	**RAG pattern** — retrieve relevant information and generate grounded answers with a foundation model.
Agentic workflow / tool use	**Orchestrated FM workflow** — use a foundation model to plan, reason, and call tools or APIs.
Custom traditional ML	**Amazon SageMaker AI** — build, train, and deploy your own machine learning models.
Simple chatbot intents	**Amazon Lex** — build conversational interfaces for common intent-based tasks.

Model-selection cues

- **Match the model to the modality:** text, image, or multimodal.
- **Consider latency, cost, and quality** together.
- **Use grounding** when freshness or factuality matters.
- **Use fine-tuning** only when prompt design is not enough.
- **Choose managed services** when they meet the requirement.

Domain 4 — Guidelines for Responsible AI (14%)

AIF-C01 Visual Study Guide

Design trustworthy AI with fairness, transparency, and human oversight.

Responsible AI in practice

Identify risk
Who could be harmed?

Assess data
Check representation

Test outcomes
Measure fairness

Explain
Document behavior

Review
Add human checks

Monitor
Improve over time

Six principles that appear in exam questions

FAIRNESS
Avoid unjustified differences in outcomes between groups.

TRANSPARENCY
Disclose the system's purpose, limits, and AI involvement.

EXPLAINABILITY
Make outputs and decisions understandable to people.

PRIVACY
Protect sensitive and personally identifiable information.

ROBUSTNESS
Test reliability, edge cases, and unexpected inputs.

HUMAN OVERSIGHT
Keep people accountable, especially in high-impact uses.

AWS controls to recognize

Amazon Bedrock Guardrails
Apply content and sensitive-data safeguards.

Amazon SageMaker Clarify
Detect bias and explain model predictions.

Amazon SageMaker Model Cards
Document intended use, limitations, and evaluation.

IAM / KMS / CloudTrail
Limit access, encrypt data, and audit activity.

Common exam traps

1. **Accuracy ≠ fairness**
   High overall accuracy can hide unequal group outcomes.
2. **Explanations ≠ transparency**
   Explainability and clear disclosure are related but distinct.
3. **Guardrails ≠ guaranteed truth**
   Content filters do not eliminate hallucinations.
4. **Human review matters**
   Keep oversight for sensitive or consequential decisions.

Domain 5 — Security, Compliance, and Governance for AI Solutions (14%)

AIF-C01 Visual Study Guide

Protect data, control access, and govern AI systems on AWS.

aws Certified AI Practitioner FOUNDATIONAL

Governance loop

Classify data → Control access → Encrypt → Monitor → Audit → Improve

Classify data	IAM	KMS	CloudWatch	CloudTrail	Policy review

Security essentials

AWS Identity and Access Management (IAM)	Least-privilege access
AWS Key Management Service (KMS)	Encryption key management
Amazon VPC	Private networking when needed
AWS CloudTrail	API activity and audit trails
Amazon Bedrock Guardrails	Safety controls for FM apps

What to protect

- Training data and knowledge sources
- Prompts and model outputs
- Personally identifiable information (PII)
- Credentials and secrets
- Usage policies and access boundaries

Good governance practices

- **Follow least privilege**
- **Encrypt data at rest and in transit**
- **Log and monitor usage**
- **Define acceptable-use policies**
- **Review compliance requirements**
- **Keep human approval where appropriate**

Common exam traps

1. Overly broad permissions
2. Ignoring prompt or output data sensitivity
3. Forgetting encryption and audit logs
4. Assuming AI services remove all governance responsibilities
5. Missing content-safety controls for GenAI apps

Rapid Review — Service Chooser & Exam Cues

AIF-C01 Visual Study Guide

A final-page cram sheet for quick recall before test day.

Service chooser

Need / Use case	Best-fit AWS service
Foundation-model app	Amazon Bedrock
Custom ML development	Amazon SageMaker AI
Chatbot with predefined intents	Amazon Lex
Sentiment / entity detection	Amazon Comprehend
Image or video analysis	Amazon Rekognition
Document text extraction	Amazon Textract
Speech to text	Amazon Transcribe
Text to speech	Amazon Polly

12 exam cues to remember

1. Fresh knowledge often suggests **RAG.**
2. Managed FM access points to **Bedrock.**
3. Simple intent-based bots point to **Lex.**
4. OCR and document extraction point to **Textract.**
5. Sentiment and entities point to **Comprehend.**
6. Images and video point to **Rekognition.**
7. **Lowest-ops** solution is often preferred.
8. Responsible AI includes **fairness** and **transparency.**
9. **Human oversight** matters for high-impact decisions.
10. **Guardrails** help reduce unsafe FM output.
11. **Least privilege** and **encryption** are always strong answers.
12. Read for words like **latency, cost, scale,** and **governance.**

Pattern chooser

Pattern	What it is	Best for
Prompt engineering	Adjust the prompt only	quick improvement
RAG	Retrieve trusted information	factual grounding
Fine-tuning	Further train the model	specialized behavior
Traditional ML	Predict or classify from structured data	non-generative tasks

Key term memory lane

Model	A learned function
Training	Learning from examples
Inference	Using the model
Hallucination	Incorrect generated content
Embedding	Meaningful numeric representation
Guardrail	Safety or policy control

Test-day tips

Read for business need, modality, data sensitivity, ops effort, and governance clues.