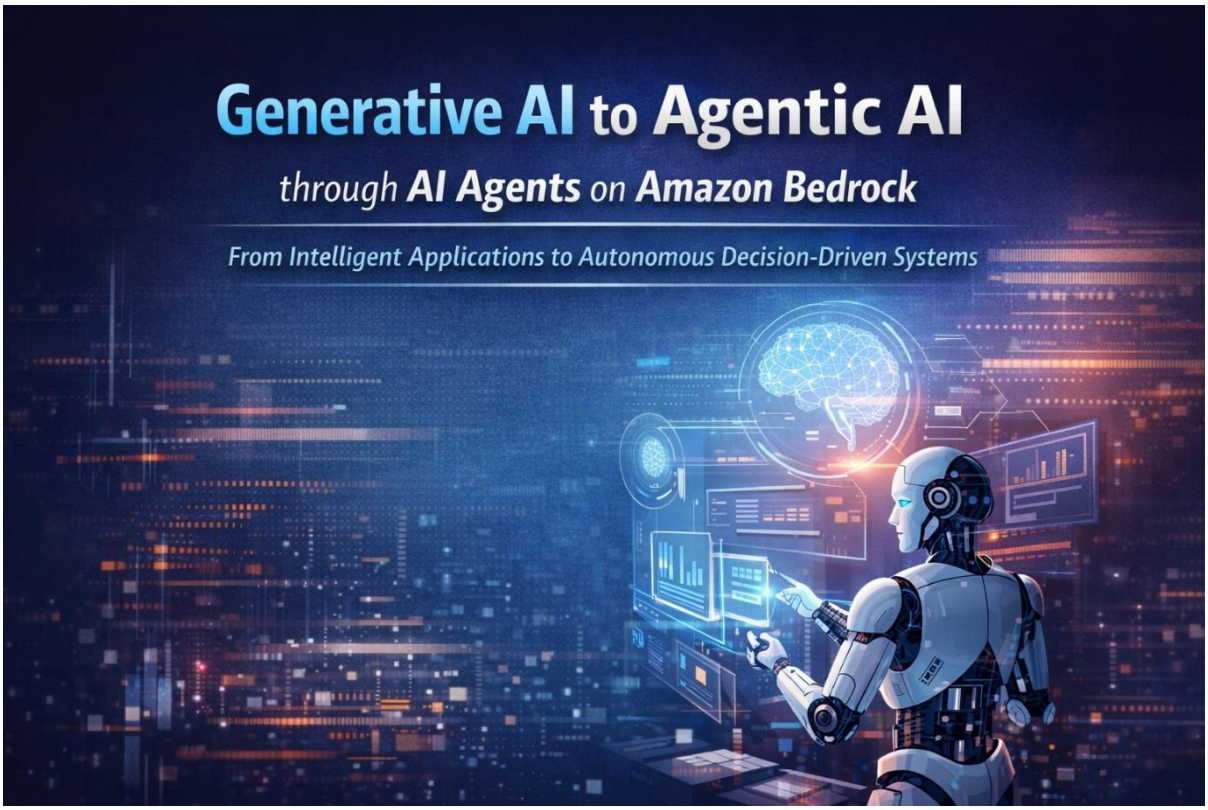

THE WORLD IS MOVING AT THE
SPEED OF AI — ARE YOU READY?
The world is no longer adopting AI — it is running on AI.
Every industry is being transformed by Generative AI, AI Agents
and Autonomous Systems. Jobs are not disapwaaring —
roles are evolving. The real risk is not
learning AI, not AI ttself.
Generative AI
AI Agents
AI IS NOT REPLACING YOU —
PEOPLE WHO KNOW AI WILL
IT & Cloud
Healthcare & Education
Business & Security
MAKE AI WORK FOR YOU, NOT AGAINST YOU
One course. Every domain. Unlimited opportunities.

KENWELL IT SOLUTIONS

https://www.kenwellitsolutions.com/ Phone: +91-9949572855

AI will shape your career — the only question is whether you control it or get controlled by it.

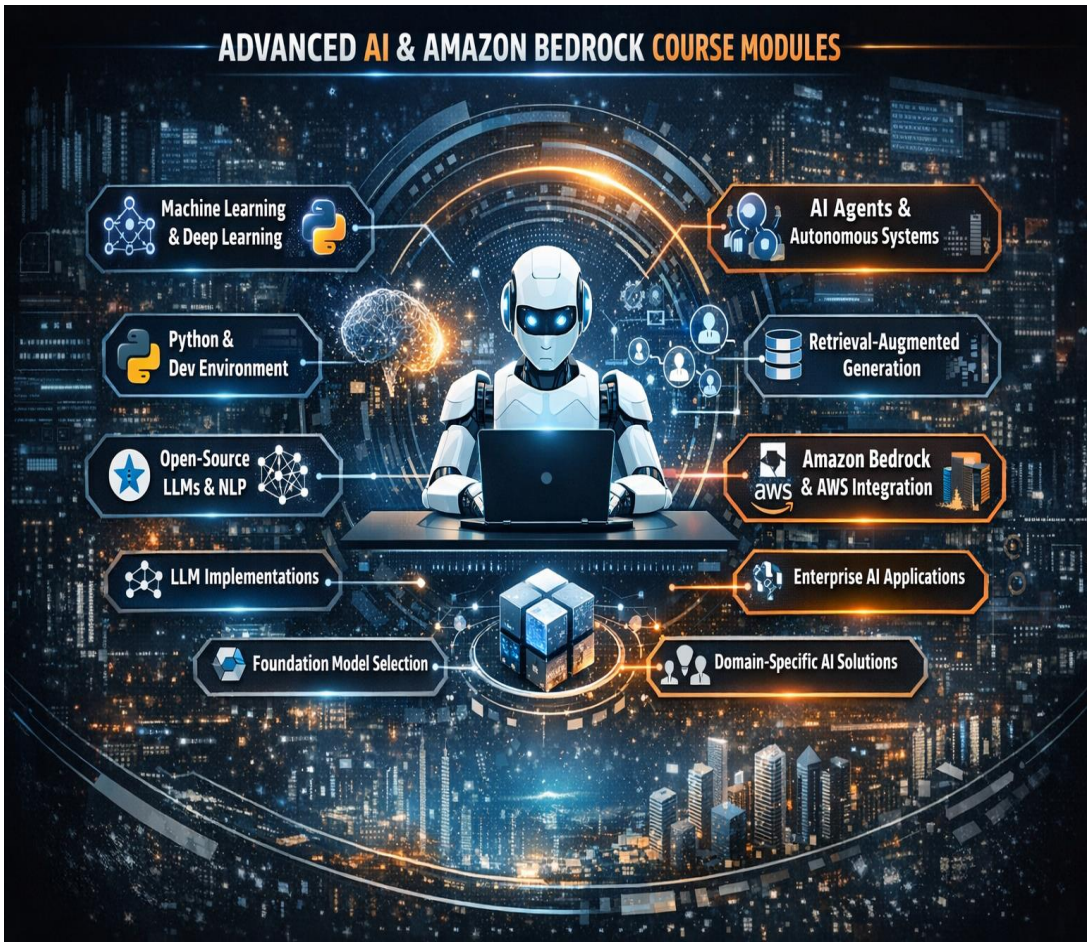


Course Highlights

- End-to-end coverage from Generative AI applications to Agentic AI systems
- Strong focus on AI Agents as the core building blocks for autonomous AI
- Hands-on development using Amazon Bedrock–managed foundation models
- Practical implementation of Retrieval-Augmented Generation (RAG)
- Design and deployment of single-agent and multi-agent systems
- Exposure to agent reasoning, planning, memory, and tool orchestration
- Secure, scalable deployments using AWS IAM and Bedrock runtime APIs
- Comparison of open-source LLMs vs managed foundation models
- Real-world domain-specific AI agent use cases (IT, security, education, healthcare, business)
- Coverage of Agentic AI architectures and deployment patterns
- Bonus exposure to Model Context Protocol (MCP) for next-generation GenAI systems

🚀 Why This Course Is Different?

- No Complex Coding Logic Required – AI handles reasoning, decision-making, and workflows
- Focus on “How to Build with AI”, not “How to Code AI”
- Designed for Both IT and Non-IT Backgrounds
- From Zero to Enterprise-Ready AI Systems
- Hands-on, Use-Case Driven Learning instead of theory-heavy teaching



🎯 What You Will Ultimately Become Capable Of

- Designing intelligent AI agents instead of simple chatbots
- Building autonomous Agentic AI systems that plan, reason, and act
- Deploying secure, scalable enterprise AI solutions on Amazon Bedrock
- Applying AI confidently across any industry or role

- Staying future-proof as AI moves from tools → agents → autonomous systems

TARGET LEARNING OUTCOMES

Module-1: Python & AI Development Environment

- Python Setup and Package Management (PIP)
- Virtual Environments (Local, venv, Cloud – Colab)
- VS Code & Git Basics
- Streamlit for AI Demos
- Flask for Model APIs

Module-2: Machine Learning Fundamentals for Generative AI

- Traditional Programming vs Machine Learning
- Supervised & Unsupervised Learning
- Regression (Simple & Multiple)
- Classification Basics
- Logistic Regression
- Decision Trees & Random Forest
- Overfitting vs Underfitting
- Model Evaluation Metrics

Module-3: Deep Learning Essentials

- Neural Network Basics (Layers, Weights, Bias)
- Feed Forward & Backpropagation
- Activation Functions (ReLU, Sigmoid, Softmax)
- Loss Functions & Optimizers (Adam focus)
- Shallow vs Deep Networks
- TensorFlow & Keras (Model Building Basics)

Module-4: NLP, Sequential Models & Transformers

- Sequential Data & Word Embeddings
- RNN & LSTM (Conceptual Understanding)
- NLP Fundamentals
- Transformer Architecture (High-Level)
- Self-Attention Mechanism
- Transformers vs RNN/LSTM
- Role of Transformers in LLMs & Generative AI

Module-5: Open Source LLMs & Transformer-Based Generative AI Applications

- Evolution from ML and DL to Generative AI

- Overview of Open Large Language Models (LLMs)
- Capabilities and Limitations of LLMs
- Transformer Architecture and Its Importance
- Attention and Self-Attention Mechanism
- Role of Transformers in Modern NLP
- Open-Source LLM Ecosystem (LLaMA, Mistral, Mixtral)
- Accessing LLMs via Platforms and APIs (Hugging Face, Groq)
- High-Performance LLM Inference Concepts
- Hugging Face Transformers and Pipelines
- Introduction to LLM Application Frameworks (LangChain Basics)

Module-6: Generative AI Projects with Local Deployment

- Text Generation Application
  Build a local LLM-powered application for generating text responses using open-source models.
- Text Summarization System
  Develop a local solution to summarize long documents using Transformer-based models.
- Question Answering System
  Implement a locally deployed QA system that answers user queries based on provided text.
- Chatbot Using Open-Source LLMs
  Create an interactive chatbot running entirely on a local machine for privacy and control.
- Text Classification Application
  Build a local application to classify text (sentiment, category, intent) using Transformer models.
- Prompt-Based AI Assistant
  Design a prompt-driven assistant that performs multiple NLP tasks locally.
- Sequential LLM Workflow Application
  Implement structured LLM workflows using chaining concepts for step-by-step task execution.
- LLM Performance Comparison
  Compare lightweight vs large open-source LLMs based on speed, accuracy, and resource usage.

Module-7: Generative AI with Amazon Bedrock (Enterprise Readiness)

- Need for Cloud-Based Generative AI Platforms
- Introduction to Amazon Bedrock
- Benefits of Amazon Bedrock over Local LLM Deployment
- Fully Managed Foundation Models in Bedrock
- Supported Model Providers (Anthropic, Amazon Titan, Meta, etc.)
- Foundation Model Selection in Bedrock

- Prompt Engineering in Amazon Bedrock
- Retrieval-Augmented Generation (RAG) with Bedrock
- Fine-Tuning Concepts in Bedrock
- Model Evaluation and Quality Metrics
- Security, Privacy, and Data Isolation in Bedrock
- Cost Management and Pricing Considerations
- Integration of Bedrock with Enterprise Applications

Module-8: Generative AI Project Lifecycle & Foundation Model Selection

- Generative AI Implementation Lifecycle
- Identifying Business Use Cases
- Defining Objectives and Expected Outcomes
- Foundation Model Selection
  - Modality (Text, Image, Embeddings)
  - Model Size and Parameters
  - Inference Speed and Latency
  - Context Window
  - Pricing Considerations
  - Training Dataset
  - Open Source vs Proprietary Models
  - Fine-Tuning Capability
  - Quality of Response (Accuracy, Toxicity, Robustness)
- Adapting and Aligning Foundation Models
- Generative AI Application Integration
- Industry Use Cases of Generative AI
- Overview of Foundation Models
- Foundation Model Evaluation Concepts

Module-9: Amazon Bedrock Architecture & AI-Powered Application Development

- Need for Amazon Bedrock in Enterprise Generative AI
- Amazon Bedrock Architecture Overview
- Bedrock Service Components
- Model Catalog and Inference Flow
- Amazon Bedrock Foundation Models & Model Selection Strategy
- AI-Powered Application Development using API Keys
- Short-Term vs Long-Term API Keys
- Limitations of API Key–Based Applications

Module-10: Enterprise AI-Powered Applications using AWS CLI & IAM

- AWS CLI Overview and Industry Importance
- AWS CLI Installation and Credential Management
- AWS Identity and Access Management (IAM)
- IAM Users, Groups, Roles, and Policies

- IAM-Based Authentication for Amazon Bedrock
- Managed and Custom IAM Policies for Bedrock
- Secure AI-Powered Application Deployment using AWS CLI
- Removing API Key Dependencies
- Enterprise Security and Deployment Best Practices

Module-11: AI Agents

- Limitations of Traditional AI and LLM Pipelines
- Introduction to AI Agents
- Definition and Core Capabilities of AI Agents
- AI Agent Architecture
- Core Pillars of AI Agents (LLM, Tools, Memory, Reasoning)
- Types of AI Agents
- Single-Agent vs Multi-Agent Systems
- Autonomous and Planning-Based Agents
- Tool-Using AI Agents
- Memory in AI Agents
- Retrieval-Augmented Generation (RAG) Agents
- Real-World AI Agent Use Cases

Module-12: AI Agents on Amazon Bedrock

- GenAI-Powered Applications vs AI Agents
- Goal-Driven AI Systems
- Amazon Bedrock Agents Overview
- Foundation Model Usage within Bedrock Agents
- Amazon Nova Lite and Nova Pro Models
- Agent Reasoning and Decision Logic
- Single AI Agent Architecture on Bedrock
- Agent Aliases and Versioning Concepts
- Agent Lifecycle (Create, Prepare, Invoke)
- Bedrock Agent Runtime and Execution Flow
- Security and Isolation in Bedrock Agents
- IAM-Based Access Control for Agents
- Agent Invocation using Bedrock Runtime APIs

Module-13: Retrieval-Augmented Generation (RAG) using Amazon Bedrock

- Limitations of Pure LLM-Based Responses
- Need for Retrieval-Augmented Generation (RAG)
- RAG Architecture and Core Concepts
- Amazon Bedrock Knowledge Bases Overview
- Data Sources for Knowledge Bases
- Structured vs Unstructured Data in RAG
- Reducing Hallucinations using RAG
- Context Management and Token Efficiency

- Security and Data Isolation in Knowledge Bases
- IAM-Based Access Control for Knowledge Bases
- Use Cases of RAG across Industries

Module-14: Building Domain-Specific AI Agents for Real-World Automation

- Network Configuration AI Agent
  Automates network configuration generation, validation, and compliance documentation, reducing manual network engineering effort.
- Code Writing & Review AI Agent
  Generates application code, performs code reviews, detects defects, and suggests improvements for software development teams.
- Cybersecurity Analysis AI Agent
  Analyzes security logs, alerts, and incidents to support threat detection, investigation, and remediation planning.
- Teaching & Learning AI Agent
  Acts as a virtual tutor by generating explanations, lesson plans, quizzes, and personalized learning paths for students and educators.
- Medical Documentation AI Agent
  Assists healthcare professionals by summarizing clinical notes, generating medical reports, and supporting documentation workflows (non-diagnostic).
- Business & Management AI Agent
  Generates business reports, meeting summaries, policy drafts, and decision-support insights for management teams.
- Customer Support & Service AI Agent
  Handles issue classification, response generation, and resolution guidance across technical and non-technical domains.

Module-15: Agentic AI – Autonomous Systems on Amazon Bedrock

- AI Agents vs Agentic AI
- Evolution from Single Agents to Agentic Systems
- Characteristics of Agentic AI (Autonomy, Planning, Decision-Making)
- Agentic AI Architecture and Design Principles
- Agentic AI Framework Concepts
- Role of Orchestration in Agentic Systems
- Memory, Planning, and Tool Coordination in Agentic AI
- Single-Agent vs Multi-Agent Agentic Systems
- Human-in-the-Loop vs Fully Autonomous Systems
- Agentic AI Deployment Models
- Implementing Agentic AI on Amazon Bedrock
- Bedrock Agents for Agentic Workflows
- Integration of RAG with Agentic AI
- Security, Governance, and Control in Agentic AI Systems

Module-16: Agentic AI Projects on Amazon Bedrock (Production Ready)

- Goal-Based Text Summarization Agent (Agentic Flow)
  An agentic system that decides *how* to summarize content (paragraph vs bullets) based on input type, using planning and reasoning instead of a fixed prompt.
- Knowledge-Grounded Q&A Agent using RAG
  An agent that retrieves relevant information from an Amazon Bedrock Knowledge Base, reasons over the retrieved data, and generates accurate, grounded answers.
- Task-Oriented Decision-Making Agent
  An agent that analyzes a user request, determines required steps, selects the appropriate strategy, and produces structured outputs (decision → action → response).
- Multi-Step Research Assistant Agent
  A simple agentic system that breaks a question into steps, gathers context, refines responses iteratively, and produces a final consolidated answer.

Bonus Module: Model Context Protocol (MCP)

- Introduction to Model Context Protocol (MCP)
- Problems Solved by MCP in GenAI Systems
- MCP as a Standardized Context Exchange Protocol
- MCP Architecture Overview
- MCP Host, Client, and Server Concepts
- MCP Primitives: Tools, Resources, and Prompts
- MCP and Tool Invocation for LLMs