

# Complete Generative AI Series with Python & RAG, LangChain, LangGraph, LangSmith, LangServer

- **What You'll Learn**
  - Large Language Models (LLMs) fundamentals
  - LangChain from basics to advanced
  - Build AI Agents and Autonomous Systems
  - RAG (Retrieval Augmented Generation) Systems
  - AWS Bedrock for Enterprise AI
  - LangGraph for Complex Workflows
  - MCP Servers & Production Deployment
- **Prerequisites**
  - Basic Python Programming Knowledge
  - Laptop with 8GB RAM
  - Enthusiasm to Learn!

## **A. MODULE 1: FUNDAMENTALS**

1. **Series Introduction: What is GenAI? | Complete Roadmap | What We'll Build**
  - Introduction to the series
  - Complete content roadmap
  - What makes this different
  - Projects preview
  - 1-to-1 Support details
2. **What is LLM and How It Works: Large Language Models Explained | GenAI Fundamentals**
  - What are Large Language Models?
  - How LLMs are trained
  - Understanding transformers
  - Popular LLMs: GPT, Claude, Gemini, Llama
  - LLM capabilities and limitations
3. **Tokenization & Building Custom Tokenizer: How LLMs Understand Text | Tokenization Deep Dive**

- What is tokenization?
- Importance of tokenization?
- Build your own tokenizer.
- Token limits and costs

## **B. MODULE 2: SETUP & FIRST STEPS**

- 1. Complete API Setup Guide: OpenAI, Google Gemini, Groq, Anthropic Claude**
  - Create OpenAI account & get API key
  - Setup Google Gemini API
  - Get Groq API access
  - Anthropic Claude setup
  - HuggingFace account setup
  - Environment variables & security
  - API cost management
- 2. Getting Started with LangChain & OpenAI: Your First LangChain Application**
  - What is LangChain?
  - LangChain installation
  - Basic LangChain structure
  - Connect OpenAI with LangChain
  - Your first LLM call
- 3. Prompt Templates & Chains: LangChain Expression Language (LCEL)**
  - What are Prompt Templates?
  - Dynamic prompts with variables
  - LangChain chains
  - LCEL (LangChain Expression Language)
  - Chaining multiple operations
  - String output parsers

## **C. MODULE 3: BUILDING CHATBOTS**

- 1. Build AI Chatbot with LangChain & OpenAI: End-to-End QnA ChatBot with Streamlit UI**
  - Design chatbot architecture
  - LangChain + OpenAI integration
  - Build Streamlit interface
  - Handle user input
  - Display responses
  - **Project: QnA Chatbot with OpenAI**
- 2. Add Memory to AI Chatbot: LangChain Message History | Conversation Memory**
  - Why memory matters

- ConversationBufferMemory
  - Message history
  - Session management
  - Context retention
  - Memory implementation
3. **Run LLMs Locally FREE with Ollama: Ollama + LangChain | Local AI Models**
    - What is Ollama?
    - Install Ollama
    - Download local models (Llama, Mistral, Phi)
    - Connect Ollama with LangChain
    - Build offline chatbot
    - Pros and cons of local LLMs
    - **Project: Offline Chatbot with Ollama**
  4. **Groq AI - Fastest LLM Inference: Groq + LangChain Integration**
    - What is Groq?
    - Why Groq is fastest
    - Setup Groq API
    - LangChain + Groq integration
    - Speed benchmarks
  5. **Build Chatbot with Groq & Streamlit: Ultra-Fast AI Chatbot with Message History**
    - Complete chatbot with Groq
    - Add message history
    - Streamlit UI design
    - Session state management
    - Deploy chatbot
    - **Project: Groq Chatbot**
  6. **Build Chatbot with Google Gemini: Google Gemini + LangChain + Streamlit**
    - Setup Google Gemini
    - LangChain integration
    - Build chatbot interface
    - Add memory
    - Handle responses
    - **Project: Gemini Chatbot**
  7. **Build Chatbot with HuggingFace Models: Free Open-Source LLMs | HuggingFace + LangChain**
    - HuggingFace Hub introduction
    - Access free models

- Setup HuggingFace API
- Integrate with LangChain
- Build QnA chatbot
- Model selection tips
- **Project: HuggingFace Chatbot**

#### **D. MODULE 4: AI AGENTS**

- 1. What are AI Agents: Autonomous AI | ReAct Framework Explained**
  - What makes an agent autonomous?
  - Agent vs Chatbot
  - ReAct (Reasoning + Acting) pattern
  - Agent architecture
  - When to use agents
  - Real-world examples
- 2. Build Your First AI Agent: LangChain + OpenAI Function Calling**
  - Agent fundamentals
  - Function calling explained
  - Create simple tools
  - Agent executor
  - Tool selection logic
  - Error handling
  - **Project: Basic Calculator Agent**
- 3. AI Agent Tools & Function Calling: Create Custom Tools for Your Agents**
  - What are agent tools?
  - Built-in LangChain tools
  - Create custom tools
  - Tool descriptions
  - Multiple tool handling
  - Best practices
- 4. Build Google Search Agent: AI Agent with Web Search | LangChain + Gemini**
  - Setup search API
  - Create search tool
  - Build search agent
  - Parse search results
  - Combine multiple sources
  - Streamlit interface
  - **Project: Google Search Agent**
- 5. Build SQL Agent: Chat with PostgreSQL Database**
  - Setup PostgreSQL

- SQL agent fundamentals
  - Database connection
  - Natural language to SQL
  - Query execution
  - Security considerations
  - **Project: SQL Agent**
- 6. Build GitHub AI Agent: Automate Repository Management**
- GitHub API setup
  - Create GitHub tools
  - Repository operations
  - Issue management
  - PR automation
  - Code search
  - **Project: GitHub Automation Agent**
- 7. Build Email AI Agent: Auto-Read & Send Emails**
- Gmail API setup
  - Email reading agent
  - Email sending automation
  - Smart filtering
  - Response generation
  - Security & permissions
  - **Project: Email Assistant Agent**
- 8. Build Jira AI Agent: Automate Project Management**
- Jira API integration
  - Create/update tickets
  - Project tracking
  - Sprint management
  - Automated workflows
  - Team productivity
  - **Project: Jira Automation Agent**

## **E. MODULE 5: RAG SYSTEMS**

- 1. What is RAG: Retrieval Augmented Generation Explained**
- What is RAG?
  - Why RAG is important
  - RAG architecture
  - RAG vs other approaches
  - Use cases
  - Limitations

2. **Vector Embeddings & Vector Databases: Understanding Embeddings | Pinecone, ChromaDB, FAISS**
  - What are embeddings?
  - How embeddings capture meaning
  - Vector databases explained
  - Similarity search
  - Popular vector DBs comparison
3. **Document Loaders: Load Data from Multiple Sources**
  - PDF loader
  - DOCX loader
  - CSV loader
  - Web scraping
  - API data loading
  - Custom loaders
4. **Text Chunking Strategies: RecursiveCharacterTextSplitter Tutorial**
  - Why chunking matters
  - Chunk size optimization
  - RecursiveCharacterTextSplitter
  - Overlap strategies
  - Best practices
5. **Vector Embeddings & Storage: Pinecone, ChromaDB, FAISS Tutorial**
  - Generate embeddings
  - Choose embedding model
  - Vector database setup
  - Store embeddings
  - Index management
6. **Build RAG Chatbot with PDFs: Chat with PDF Documents + Memory**
  - Complete RAG pipeline
  - PDF processing
  - Vector storage
  - Retrieval logic
  - Conversation memory
  - Streamlit UI
  - **Project: PDF Chat Application**
7. **Build Website Chatbot: RAG with Web Scraping**
  - Website scraping
  - Content extraction
  - Build RAG system

- Chat with website content
- Handle dynamic content
- Deploy chatbot
- **Project: Website QnA Bot**

#### 8. **Multi-Document RAG System: Chat with Multiple PDFs**

- Handle multiple documents
- Metadata filtering
- Source attribution
- Document routing
- Comparison queries
- Advanced retrieval
- **Project: Multi-Doc RAG System**

### F. **MODULE 6: LANGGRAPH & ADVANCED**

#### 1. **Pydantic for Data Validation: Pydantic + LangChain & LangGraph**

- Pydantic basics
- Data validation
- Type safety
- Models for LangChain
- Complex schemas
- Best practices

#### 2. **Introduction to LangGraph: Build Agentic Workflows**

- What is LangGraph?
- Graphs vs Chains
- LangGraph components
- State management
- Use cases
- When to use LangGraph

#### 3. **Build Your First LangGraph App: Simple Chatbot with LangGraph**

- Create graph structure
- Define nodes
- Add edges
- State management
- Compile graph
- Run workflow
- **Project: LangGraph Chatbot**

#### 4. **Routing & Conditional Logic: LangGraph Conditional Edges**

- Conditional routing
- Dynamic workflows

- State-based routing
  - Multiple paths
  - Error handling
  - Complex logic
5. **Multi-Tool Agents in LangGraph: LangGraph ToolNode Tutorial**
- ToolNode explained
  - Multiple tools
  - Tool selection
  - Parallel execution
  - Error handling
  - Best practices
6. **Memory in LangGraph: Persistent State Management**
- LangGraph memory
  - State persistence
  - Checkpoint management
  - Resume workflows
  - Memory implementation
  - Production patterns
7. **Human-in-the-Loop Workflows: LangGraph Approval System**
- Human approval nodes
  - Interrupt patterns
  - Resume execution
  - Feedback loops
  - UI integration
  - Production use cases
8. **RAG with LangGraph: Advanced RAG Workflow**
- RAG in LangGraph
  - Multi-step retrieval
  - Query refinement
  - Result validation
  - Complex workflows
  - Performance optimization
  - **Project: LangGraph RAG System**
9. **AI Blog Generator: LangGraph Multi-Agent System**
- Multi-agent collaboration
  - Content generation workflow
  - Research → Write → Edit
  - Agent coordination



- Quality control
- Full automation
- **Project:** AI Blog Generator

## **G. MODULE 7: APIs & DEPLOYMENT**

### **1. FastAPI Crash Course: Build REST APIs for AI Applications**

- FastAPI basics
- Routing
- Pydantic models
- Request/Response handling
- Error handling
- API documentation

### **2. Build AI Chatbot REST API: FastAPI + LangChain Integration**

- Design API structure
- Integrate LangChain
- Handle async operations
- Session management
- Error handling
- Test API

- **Project:** Chatbot REST API

### **3. Deploy Agents using LangServe & LangSmith: Production Agent Deployment | Monitoring & Debugging**

- What is LangServe?
- Deploy agents with LangServe
- Introduction to LangSmith
- Tracing and debugging
- Performance monitoring
- Production best practices

## **H. MODULE 8: MCP & AWS BEDROCK**

### **1. What is MCP: Model Context Protocol Explained | Claude MCP**

- What is MCP?
- Why MCP matters
- MCP architecture
- Claude integration
- Use cases
- Future of MCP

### **2. Build Custom MCP Server: Create Your Own MCP Server**

- MCP server basics
- Define resources

- Create tools
  - Connect to Claude
  - Test MCP server
  - Deploy server
  - **Project: Custom MCP Server**
3. **LangChain + MCP Integration: Use MCP Tools in AI Agents**
- Connect MCP with LangChain
  - Use MCP tools
  - Agent integration
  - Complex workflows
  - Best practices
4. **AWS Bedrock Introduction: Setup AWS Account & Bedrock**
- What is AWS Bedrock?
  - Create AWS account
  - Enable Bedrock
  - Available models
  - Pricing overview
  - Security setup
5. **Build Chatbot with AWS Bedrock: LangChain + Claude 3 on Bedrock**
- Connect Bedrock
  - Use Claude 3
  - Build chatbot
  - Handle responses
  - Cost optimization
  - Production setup
  - **Project: AWS Bedrock Chatbot**
6. **Production RAG with AWS Bedrock: Enterprise RAG System**
- Complete RAG pipeline
  - AWS Bedrock models
  - Vector storage (AWS)
  - Production architecture
  - Monitoring
  - Scalability
  - **Project: Production RAG on AWS**
7. **Deploy AI with AWS Lambda: Serverless LangChain**
- What is AWS Lambda?
  - Serverless architecture
  - Deploy LangChain

- Handle requests
- Auto-scaling
- Cost benefits
- **Project: Serverless AI API**

## 8. **Deploy with FastAPI on AWS: Production Deployment Guide**

- EC2 setup
- Deploy FastAPI
- Load balancing
- SSL/HTTPS
- Monitoring
- CI/CD pipeline

## I. **LIST OF PROJECTS**

### 1. **Throughout this series, you'll build these 10+ real-world projects:**

- **QnA Chatbot with OpenAI**
  - Technologies: LangChain, OpenAI, Streamlit
  - Features: Conversation, memory, user interface
- **Google Search Agent**
  - Technologies: LangChain, Gemini, Google Search API
  - Features: Web search, information retrieval, answer generation
- **PDF Chat Application**
  - Technologies: LangChain, RAG, Vector DB, Streamlit
  - Features: Upload PDFs, ask questions, get answers with sources
- **SQL Agent**
  - Technologies: LangChain, PostgreSQL, SQL
  - Features: Natural language to SQL, database querying
- **GitHub Automation Agent**
  - Technologies: LangChain, GitHub API
  - Features: Repo management, issue handling, automation
- **Email Assistant Agent**
  - Technologies: LangChain, Gmail API
  - Features: Read emails, auto-respond, smart filtering
- **Multi-Document RAG System**
  - Technologies: LangChain, RAG, Vector DB
  - Features: Multiple PDF handling, source attribution, comparison
- **AI Blog Generator**
  - Technologies: LangGraph, Multi-Agent
  - Features: Research, write, edit workflow automation

- **Custom MCP Server**
  - Technologies: MCP Protocol, Claude
  - Features: Custom tools, resource management
- **Production RAG on AWS**
  - Technologies: AWS Bedrock, Claude 3, LangChain
  - Features: Enterprise-grade RAG, scalable architecture

## 2. Bonus Projects:

- Offline Chatbot with Ollama
- Groq Chatbot (Ultra-fast)
- Gemini Chatbot
- HuggingFace Chatbot
- Website QnA Bot
- Chatbot REST API
- LangGraph RAG System
- Serverless AI API

**Last Updated:** January 2026 | **Version:** 2.0