Learnbay

SDE MASTER PROGRAM

For Software Developers

DSA + System Design + GenAI

In collaboration with

IBM & Microsoft

Created for developers aiming to build, scale, and lead **AI systems.**

1 Why Every **Developer** Must Learn GenAI + DSA + System Design?

AI is not killing software developer jobs

AI-first developers will replace who don’t adapt.

WHY ACT NOW

DSA & System Design

Learn DSA & System Design to crack Top MNCs Interview

GenAI & AgenticAI

Transform you from a Developer to an AI product-builder.

The future belongs to developers who combine
AI + Problem Solving (DSA) + AI first Design

2

How this course prepares you for top tech companies interviews ?

Top tech companies don’t hire tool-users.
Engineers who can’t design with AI won’t clear interviews.

The role isn’t disappearing. The expectations have gone up.

THIS PROGRAM PREPARES YOU FOR 2026 INTERVIEWS

Practice DSA + AI First System Design

Build High-Performance LLM Microservices

Deploy Multi-Agent Enterprise Workflows

Program Overview

SDE Master Program with GenAI & Agentic AI

11 Months
Flexible batches

100% Live
Instructor-led

IBM & Microsoft Certified
GenAI + Agentic AI

No Cost EMI
Interest-free

Career Services Pro

- Resume Help
- Mock Interviews
- Job Referrals

Learnbay trains you for 2026 SDE interviews — not the old script.

What hiring leads demand now:

🧠 **AI-First System Design**
Architect AI-native systems

⚡ **DSA + GenAI**
Crack DSA with copilots

🚀 **Deploy LLMs & AI Apps**
Ship live AI systems

The hiring bar jumped.
We launch you past it.

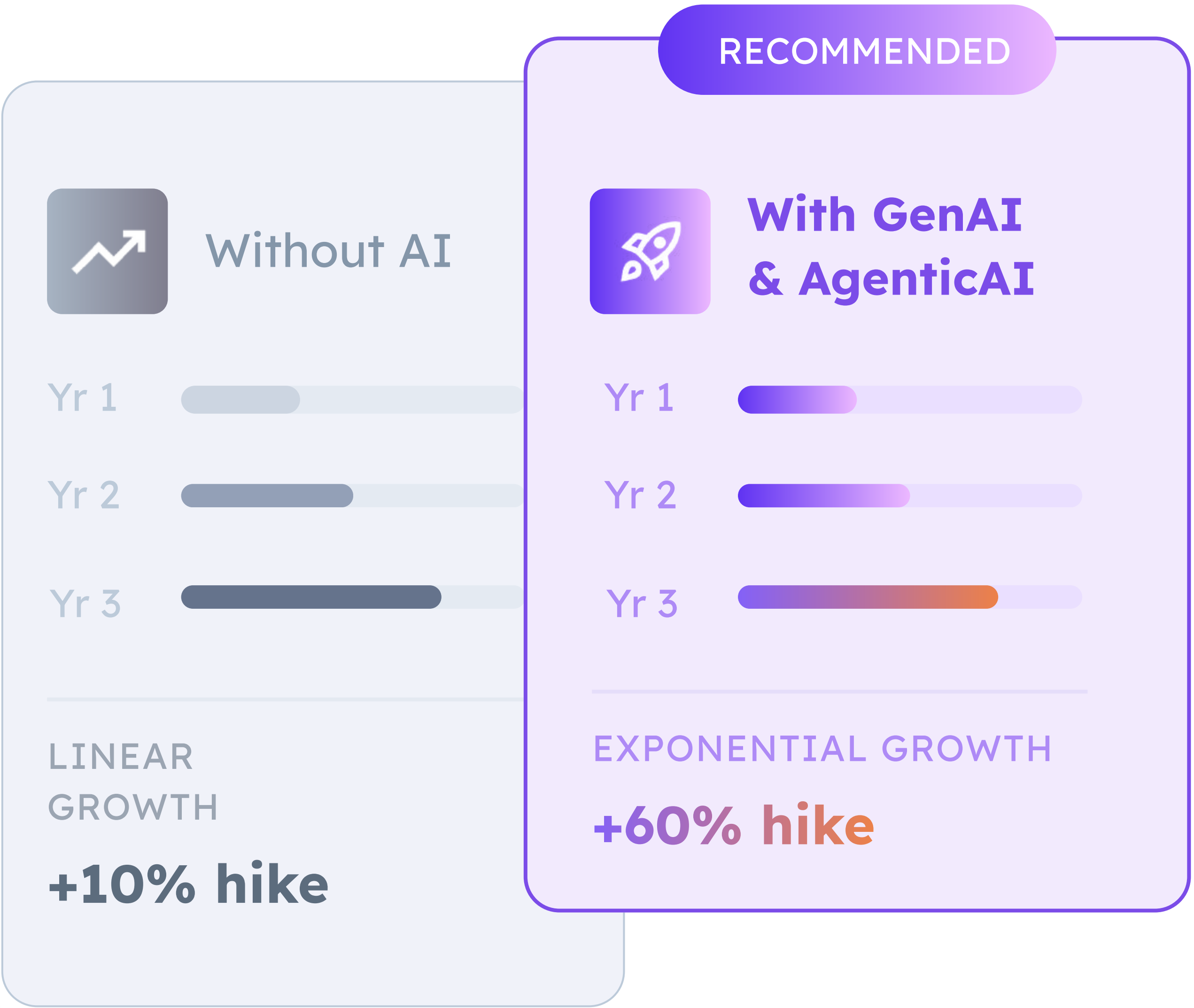
Lock in your 2026 edge

3 Can Software Developers Grow **Without AI** Skills?

Yes — but growth is linear.

With GenAI & Agentic AI along with AI first system design — growth becomes exponential.

3-YEAR CAREER TRAJECTORY



Without AI

Yr 1

Yr 2

Yr 3

LINEAR GROWTH

+10% hike

RECOMMENDED

With GenAI & AgenticAI

Yr 1

Yr 2

Yr 3

EXPONENTIAL GROWTH

+60% hike

4 How this **Learnbay Program** keeps you relevant in 2026?

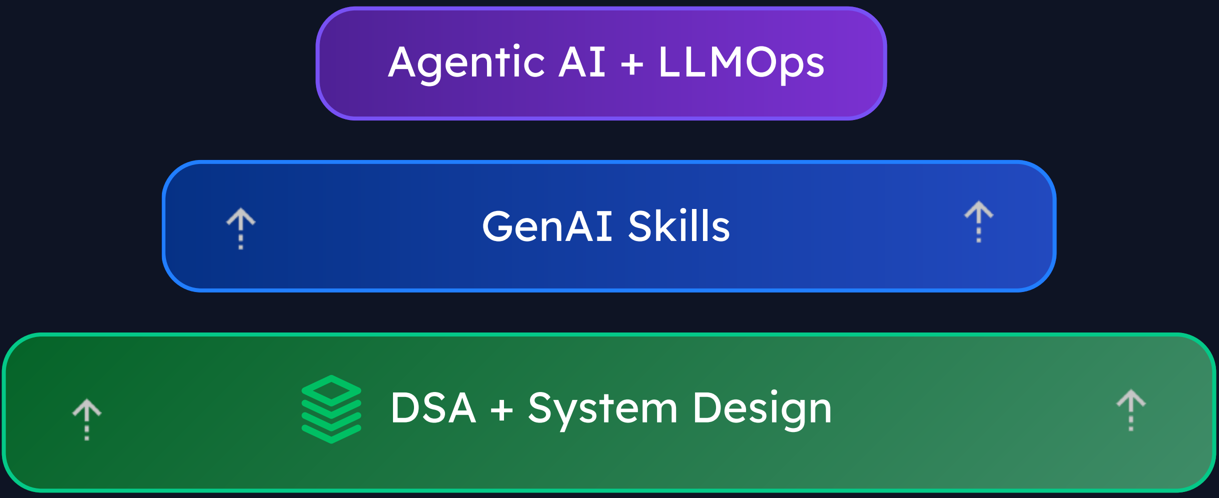
We teach you DSA + System Design + GenAI.
You need an **AI-first engineering** stack that builds on your experience.

KEY SKILLS TO ADD

DSA+
System Deign

GenAI +
Agentic AI

YOUR AI-FIRST STACK ↑ Build AI



Agentic AI + LLMOps

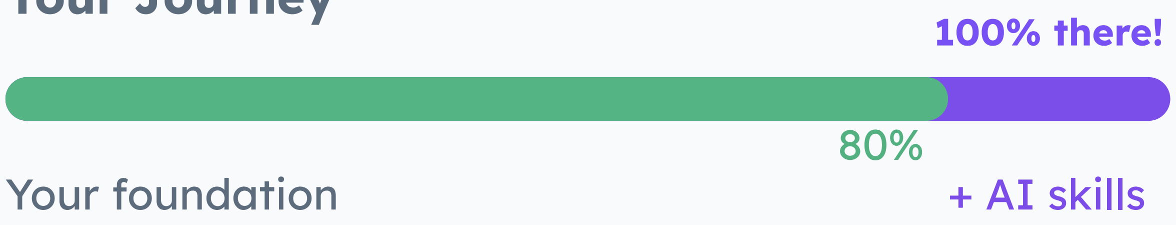
GenAI Skills

DSA + System Design

5 Is this transition realistic for someone like you?

You're upgrading, not restarting.

Your Journey



100% there!

80%

Your foundation

+ AI skills

YOU HAVE

- ✓ Programming & System Thinking
- ✓ APIs, Architecture, Debugging Skills

+ JUST ADD

- + DSA & AI First System Design
- + LLM-Powered Systems
- + Autonomous AI Agents

The question isn't if AI will reshape Tech World

it's whether you'll lead it

Learnbay Advantage

JOB REFERRALS & ASSISTANCE

We stay with you until you're placed.

Direct Referrals

To hiring partners

Mock Interviews

With FAANG mentors

AI Resume Lab

ATS-optimized

Career Desk

3-year support

Job Referrals in Top MNCs

Direct access to hiring managers

Unlimited
Referrals

5+
Mock Sessions

3 Years
Support

Job Referral & Career Acceleration

No cold applications. Your profiles goes straight to decision makers

We refer you **directly** to hiring companies

WHAT THIS MEANS FOR YOU

Skip the ATS Black hole

Your resume reaches Humans not Algorithms

Access Exclusive Opening

Roles not posted to job boards

Matched to your skills

Job which is aligned to your profile and domain

Prepare to ace the interview

Get interview ready with expert guidance

Mock Interviews

5 Session with Industry Expert

✓ Multiple Technical and HR Round with detailed feedback

AI-Optimized Resume

ATS-friendly and Job ready

✓ Optimized to highlight your technical skills and domain expertise

Complete Support until Placed

We’re with you in every step

6 What sets this program apart?

3-Year Flexible Subscription

Learn at your pace with multiple re-entries, upgrades, and refresh cycles as AI evolves.

AI CoLab Experience

Work on real AI startup projects and earn 2 Industry Project Certificates.

BYOP+Mentorship

Work on real AI startup projects and earn 2 Industry Project Certificates.

Industry Certification

Earn globally recognised credentials backed by IBM and Microsoft.

7 How Learnbay Helps you transition into Top Product based Companies?

DSA + System Design

Learn DSA & System design to crack top product based MNCs Interviews

+

GenAI + AgenticAI

Learn GenAI & AgenticAI designed for SDE

=

Expert

With proven portfolio

not

~~Fresher~~

Starting over

From Service Based → to Product Based MNCs

8 What kind of Projects will you build in this course?

Real-World GenAI & Agentic AI Projects

Not toy demos.
Real industry-grade AI systems.

GenAI

Enterprise Knowledge Copilot (RAG-Based)

Internal AI assistant to search, reason, and answer from enterprise documents.

Python · OpenAI / Azure OpenAI · Vector DB (Pinecone / FAISS) · LangChain

System Design

AI-Powered Ride Sharing System

Design a scalable ride sharing platform where riders book trips, drivers get matched in real-time, and AI optimizes driver assignment & pricing.

System Design & AI , HLD , LLD , Deployment

Top Companies Hire Engineers Who Can Deploy AI in Production.

LLMOps

Production-Grade LLM Deployment

Deploy, monitor, and scale GenAI applications with enterprise-ready controls.

AI Agents

Autonomous Incident Triage & RCA Agent

AI agent that analyzes logs, traces root cause, and recommends fixes during live incidents.

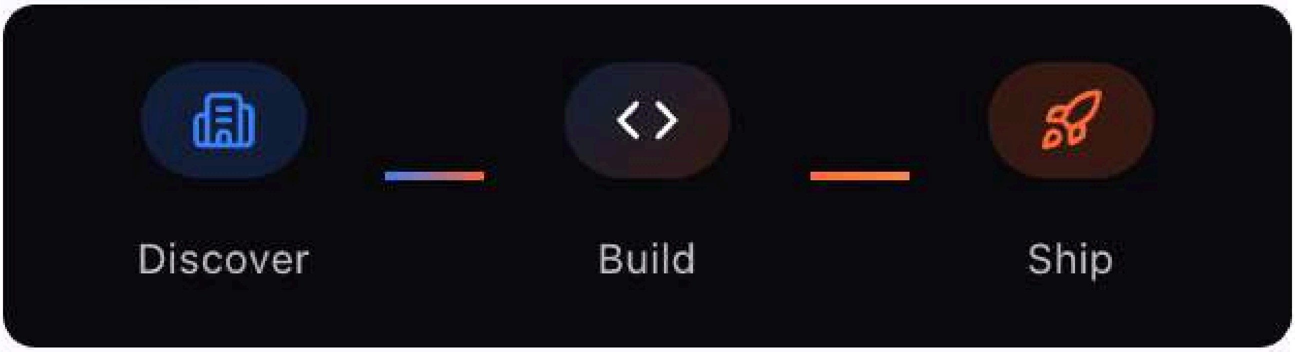
SKILLS YOU'LL MASTER

- Model versioning & controlled rollouts
- Prompt lifecycle management at scale
- Cost optimization for LLM usage
- Production monitoring & alerting
- Multi-agent orchestration workflows
- Safety guardrails, fallbacks & retries

Work on Real Projects
Directly from Startups

Project certificate beats course completion certificate.

- Signed by the startup
- Code reviewed & verified
- Ready for your portfolio



Discover

Build

Ship

"Course certificates say you watched.
Project proofs say you can build."

Real work. Real proof. Real jobs.

What is Learnbay's AI CoLab & BYOP?

AI CoLab

Work on real challenges with partnered startups

Project Certificate Beats Course Certificate

BYOP (Bring Your Own Project)

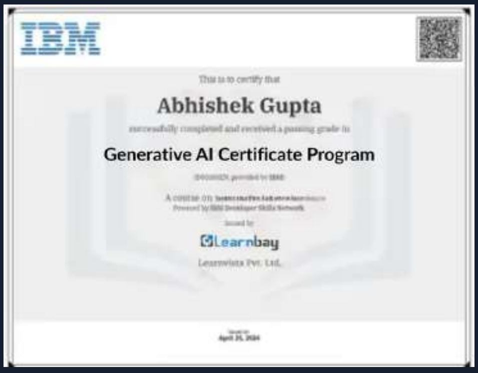
Bring a project from your domain — we will mentor you to complete it

OUR PHILOSOPHY

We are a Project-Based Learning Platform

Get Certified

and accelerate your career growth



IBM

This is to certify that

Abhishek Gupta

successfully completed and received a passing grade in

Generative AI Certificate Program

[illegible] provided by IBM

A course on [illegible]
Powered by IBM Developer Skills Network

Issued by

Learnbay

Learnvista Pvt. Ltd.

Issued on
April 25, 2024

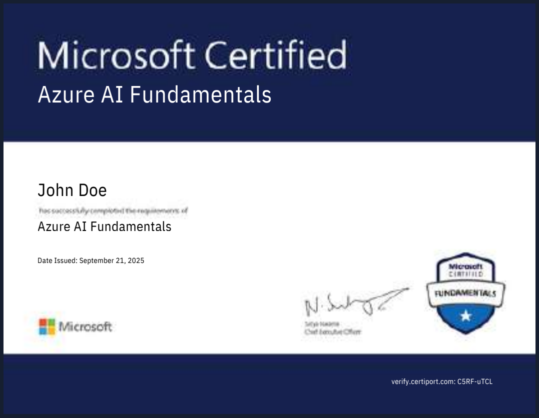
IBM

Course Completion

Generative AI Certification

Earn an industry-recognized IBM GenAI Certification validating your expertise in enterprise-grade AI systems.

Microsoft Certifications



Microsoft Certified
Azure AI Fundamentals

John Doe
has successfully completed the requirements of
Azure AI Fundamentals

Date Issued: September 21, 2025

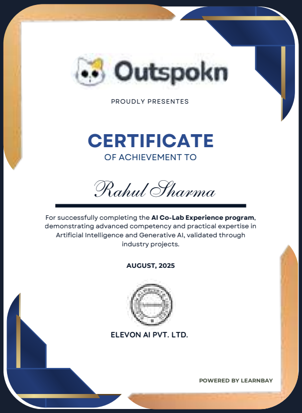
Microsoft

Satya Nadella
Chief Executive Officer

Microsoft CERTIFIED FUNDAMENTALS

verify.certiport.com: C5RF-uTCL

Get **Azure AI Certificate** from Microsoft



Outspokn

PROUDLY PRESENTES

CERTIFICATE
OF ACHIEVEMENT TO

Rahul Sharma

For successfully completing the **AI Co-Lab Experience program**, demonstrating advanced competency and practical expertise in Artificial Intelligence and Generative AI, validated through industry projects.

AUGUST, 2025

ELEVON AI PVT. LTD.

POWERED BY LEARNBAY

AI Co-Lab Project Certificate

This certification boosts your credibility in the IT sector and enhances your career prospects.

Program **Fees**

SDE Master Program with GenAI & Agentic AI

MASTER TRACK

Live online classes

- ✓ Instructor-led **live interactive** session
- ✓ **Dual** Certification Program
- ✓ **Mock technical interviews** for AI roles
- ✓ Industry-grade **capstone** projects

PROGRAM FEE

₹ 1,59,000/- +18% GST

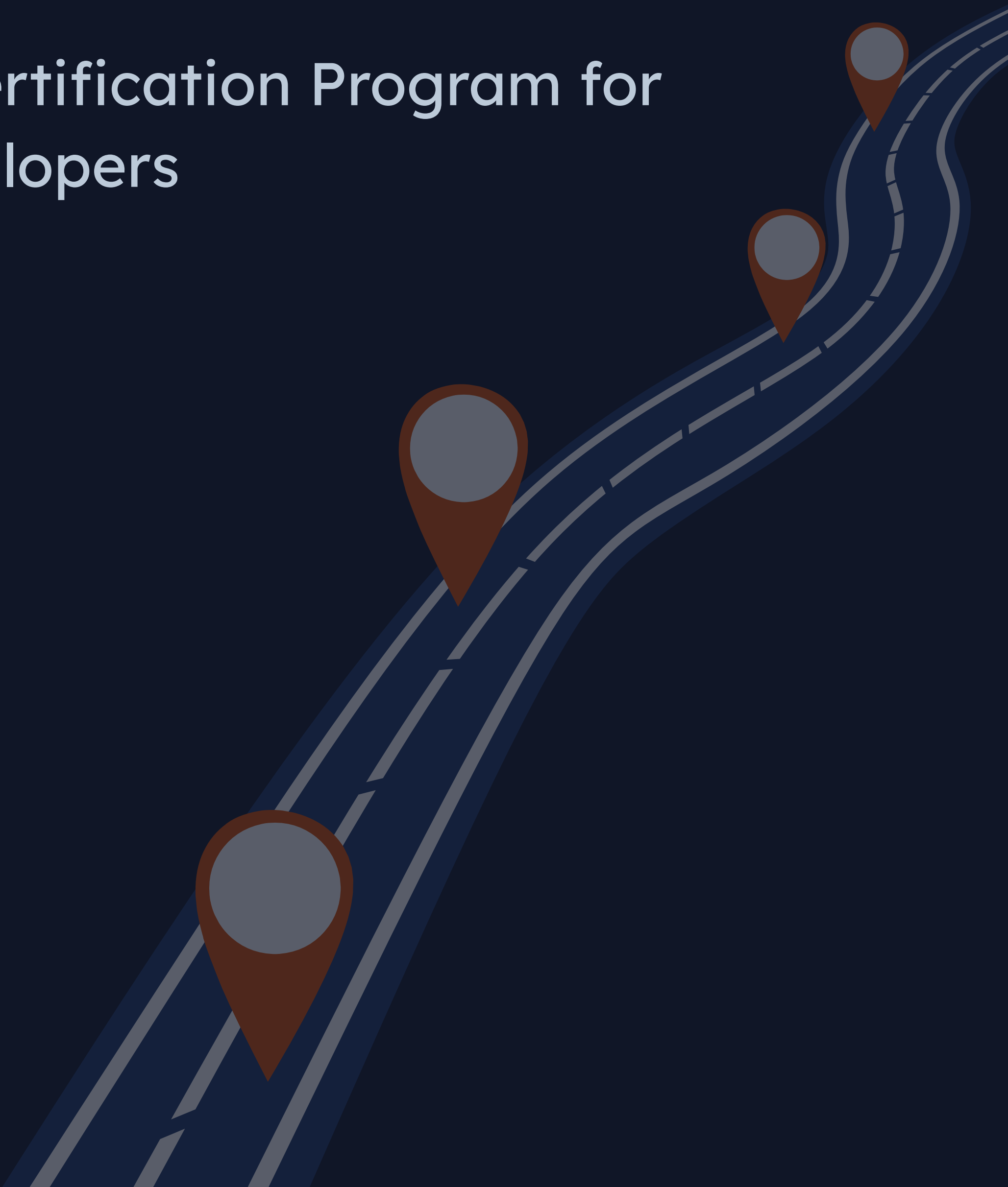
Payment Plan: No Cost EMI Available

For 6, 9 and 12 Months

11 Months Program

Program Roadmap

SDE Master Certification Program for Software Developers



Program Roadmap

FOR SDE MASTER PROGRAM WITH GENAI & AGENTIC AI

DAY 0

DURATION: 3 DAYS

INDUCTION + ORIENTATION

Understand the GenAI & Agentic AI landscape, tools, expectations, and how this program aligns with your career goals.

MODULE 1

DURATION: 2 Months

PYTHON FOR GENAI

Build a strong Python foundation tailored for developing, integrating, and controlling GenAI workflows.

MODULE 2

DURATION: 2 Months

FOUNDATION FOR GENAI

Learn core ML, deep learning, and NLP concepts that power modern large language models.

MODULE 3

DURATION: 1.5 Months

Advanced Generative AI

Master advanced techniques for building, optimizing, and deploying scalable generative AI systems and LLMs.

MODULE 4

DURATION: 1 Month

Agentic AI & Automation

Design autonomous AI agents that plan, reason, and automate complex workflows end to end.

MODULE 5

DURATION: 3 Months

Data Structure & Algorithms

Build strong problem-solving skills using core data structures and efficient algorithmic techniques.

MODULE 6

DURATION: 2 Months

System Design

Learn to design scalable, reliable, and high-performance systems used in real-world applications.

Capstone

DURATION: 10 Days

Work on Real-Time Industry Projects

Build real-time solutions for industry specific projects and deploy under industry expert guidance

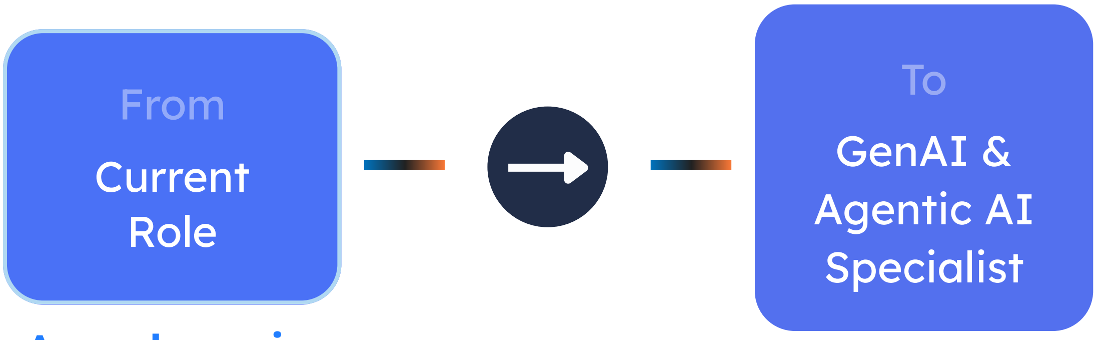
AI Colab

DURATION: 8-10 Days

Work on AI Startup Projects

Work on real AI startup projects to build deployable systems and earn industry-recognized certifications.

Domain Based Transition



From

Current Role

To

GenAI & Agentic AI Specialist

Any domain

DOMAIN SPECIALISATIONS (CHOOSE ANY 2)

Elective A

GenAI for Software Development & SDE Roles

Design scalable, fault-tolerant software architectures where GenAI and agents are core system components.

Recommended for developers with 4+ years of experience.

Elective B

AI for Cloud, DevOps & LLM Infrastructure

Build production-ready AI microservices using FastAPI, async workflows, and LLM-integrated APIs.

Elective C

1 Project

AI Powered Backend Engineering for AI Systems

Apply GenAI for predictive maintenance, quality inspection, supply chain intelligence, and factory optimization.

Elective D

2 Project

Distributed Systems & Microservices for AI Applications

Develop AI use cases for clinical decision support, medical document analysis, diagnostics assistance, and patient data intelligence.

Elective E

1 Project

Multi- Agent Systems & Workflow Automation with Agentic AI

Build multi-agent systems that coordinate tasks, decisions, and workflows through intelligent collaboration.

Elective F

2 Project

GenAI & Agentic AI for Product Managers

Design and manage GenAI products with aligned workflows, metrics, guardrails, and user experiences.

11 Months Program

Program Curriculum

SDE Master Program for Tech Professionals

140+ Hours of Industrial Projects

20+ Hours of Capstone Projects

Unlimited Doubt Clearing Sessions

Induction + Orientation

Duration: 3 Days

DAY 0

- Overview Of AI, Generative AI, And Agentic AI.
- Career Paths And Role Mapping For AI Professionals.
- LinkedIn Profile Optimization For AI & Tech Hiring.
- Using **Python Colab** For Hands-On Learning.
- Learning Workflow, Tools, And Program Onboarding.

Python for GenAI

Duration: 2 Months

Module 1

Topic 1: **Core Python**

- **Variables**
  - Store values in various data types
  - Dynamic assign / reassign variables
  - Python naming conventions for maintainable code

- **Data types**
  - Numeric types: **int, float, complex.**
  - Text handling using str and string methods.
  - Collections: **list, tuple, set, dict.**
  - Mutable vs immutable data behavior.

- **Loops**
  - Iteration using for and while loops.
  - Sequence traversal with **range(), enumerate(), zip().**
  - Optimizing iterations for data processing tasks.

Topic 1: **Core Python**

- **Control Statements**
  - Conditional logic using if, elif, else.
  - **Comparison operators** (==, !=, <, >, <=, >=).
  - Logical operators (and, or, not).
  - Flow control using break, continue.
  - Building **decision trees** for real-world logic handling.

- **Functions**
  - Creating reusable logic blocks using def and return.
  - Passing parameters using positional and keyword arguments.
  - Anonymous function with **Lambda**

Project 1 **Simple Calculator**

1 hour

Create a console-based calculator that takes user input for two numbers and performs mathematical operations

Topic 2: **Advanced Python**

- **File Handling**
  - Reading and writing files using open(), read(), write().
  - Handling text, **CSV**, and **JSON** files for data pipelines.
  - Working with file paths using os and pathlib.

- **Regex**
  - Pattern matching using **re.search(), re.findall(), re.sub()**.
  - Cleaning and extracting text data from logs, emails, and documents.
  - Using regex tokens: \d, \w, +, *, ^, $, groups.

- **Exceptional Handling**
  - Custom exceptions handling using try, except, finally.
  - Common exceptions.
  - Preventing pipeline failures in production code.

Topic 2: **Advanced Python**

- **OOP’s**
  - Creating classes and objects using class and __init__.
  - Instance variables vs class variables.
  - Methods and object behavior modeling.
  - Inheritance and method overriding.

Project 2 **Banking System Simulation**

BFSI

Simulate a simple banking system where users can create accounts, deposit, withdraw, and check balances.

Python | Data Structure | Function

Project 3 **Movie Night Recommender**

Entertainment

Recommend movies based on genre preferences using simple rule-based logic and conditional checks.

Data Types | Filtering Logic | User Input Handling

Project 4 **Smart Fitness Habit Tracker**

Healthcare

Track daily exercises, calories burned, and generate habit summaries using basic loops and functions.

Python | Functions | Conditional logic

Topic 3: **Essential Libraries for GenAI**

- **Numpy**
  - Creating arrays **using array, arange, linspace**.
  - **Array operations:** indexing, slicing, reshaping.
  - Vectorized computations for performance.
  - **Mathematical functions:** mean, sum, std, dot.
  - Broadcasting and handling multi-dimensional data.

- **Pandas**
  - Working with Series and DataFrame.
  - Reading data from **CSV, Excel, JSON**.
  - Data cleaning: missing values, duplicates, type casting.
  - Filtering, sorting, and conditional selections.
  - **Grouping** and **aggregation** using **groupby**.
  - Feature preparation for ML and GenAI pipelines.

- **Matplotlib**
  - Creating line, bar, scatter, histogram plots etc.
  - Customizing labels, titles, legends, and axes.
  - **Visualizing** trends and distributions in datasets.

- **Seaborn**
  - **Statistical plots:** boxplot, violinplot, countplot.
  - Relationship analysis using **pairplot** and **heatmap**.
  - Visualizing correlations and feature importance.
  - Styling plots for reports and dashboards.

Project 5 **Retail Sales Data EDA**

Retail

Analyze a retail sales dataset: clean data, plot trends (time series, bar charts), and derive insights.

Python Pandas Geopandas Matplotlib

Foundation for GenAI

Duration: 2 Months

Module 2

Topic 1: **ML Fundamentals**

- **Basic of ML**
  - **Supervised** vs **Unsupervised** learning paradigms.
  - Features, labels, training, validation, and test sets.
  - Bias–variance trade-off and model generalization.
  - **Overfitting** and **underfitting** with real examples.
  - End-to-end ML workflow used in industry projects.

- **Regression Analysis**
  - **Multiple Regression** with feature interactions.
  - Model evaluation using **RMSE, MAE, R^2 score.**

Project 6 **Energy Consumption Forecasting**

Energy

Predict household energy consumption using linear regression by analyzing temperature, appliances, and daily usage patterns.

Feature engineering

RMSE/R^2 evaluation

- **Classification Analysis**
  - **Logistic Regression** for binary classification.
  - Handling class imbalance using sampling techniques.
  - **Evaluation metrics: Precision**, **Recall**, **F1-Score**, **ROC-AUC**.

Project 7 **Loan Approval Prediction**

Finance

Build a classification model (e.g. logistic regression, random forest) to predict loan approvals from applicant data.

scikit-learn

Pandas

Topic 2: **Deep Learning Fundamentals**

- **Perceptron**
- **Backpropagation and Optimization**
- **Attention Mechanism**

Project 8 **Smart Document Relevance Classifier**

HR

Compute attention weights on simple sentences to understand token importance during context interpretation.

Numpy

Perceptron

Topic 3: **NLP Fundamentals**

- **Text Pre-Processing for Gen AI**

- **Text Embedding Techniques**
  - Bag-of-Words representation.
  - TF-IDF vectorization.
  - **Word2Vec: CBOW** and **Skip-Gram**.

- **Named Entity Recognition (NER)**

- **Topic Modeling**
  - Latent Dirichlet Allocation (LDA).
  - Latent Semantic Techniques (LSA/LST).

- **Part-of-Speech Tagging (POS)**

- **NLP Evaluation Metrics**

Topic 3: **NLP Fundamentals**

- **Transformer - BERT Model**
  - **Transformer architecture:** Encoder–Decoder blocks.
  - Multi-head self-attention mechanism.
  - Positional encoding concepts.
  - BERT pre-training objectives: **MLM** and **NSP**.
  - Use cases of **BERT** in real NLP systems.

Project 9 **Sentiment Analysis Using BERT**

Marketing

Use pretrained BERT embeddings to convert sentences into vectors and classify sentiment using a simple classifier.

BERT

Manual tokenization

Dense classifier

Project 10 **Legal Entity Extraction**

Legal

Perform named entity recognition and classification in legal documents using spaCy, and visualize results

spaCy

Pandas

Matplotlib

Advanced Generative AI

Duration: 1.5 Months

Module 3

Topic 1: **Fundamentals of GenAI**

- Difference Between Predictive AI And Generative AI.
- Generative Modeling Concepts And Probability Distributions.
- Pre-training vs Fine-tuning Paradigms.
- Foundation Models And Their Role In Modern AI Systems.

Topic 2: **Use cases**

- **Text Generation** For Chatbots And Assistants.
- Document Summarization And Report Automation.
- **Code Generation** And Developer Productivity Tools.
- Content Generation For Marketing And Media.
- Knowledge Assistants For Enterprises.
- **Search Augmentation** And **Question Answering Systems**.

Topic 3: **Prompt Engineering**

- **Zero-shot**, **One-shot**, And **Few-shot** Prompting.
- Instruction-based Prompt Design.
- Chain-of-Thought Prompting Techniques.
- Prompt Templates And Reusability.
- **Prompt Evaluation** And **Iterative Optimization**.

Topic 4: **LLM**

- **GPT Family** (GPT-3.5, GPT-4, GPT-4o).
- **LLaMA**, **Claude** and **Open-Source** LLM Ecosystem.
- **Tokenization** and **Context Windows**.

Topic 4: **LLM**

- **Inference Parameters: Temperature, Top-p, Max Tokens.**
- Hallucination And Limitations Of LLMs.
- Cost And Latency Considerations In Production.

Project 11 **AI-Powered Sales Email Writer**

Sales

Use an LLM to generate personalized sales outreach emails using customer behavior and CRM attributes.

OpenAI

Prompt Templates

Langchain

Topic 5: **Hugging Face**

- **Transformers Library** For NLP And GenAI.
- Pre-trained Models Hub And Model Cards.
- **Tokenizers** And **Pipeline** APIs.
- Fine-tuning Models Using **Hugging Face Trainer**.

Project 12 **Product Review Sentiment Classifier**

Retail

Build an NLP model to classify customer product reviews as positive or negative (using word embeddings and an LSTM or transformer).

NLTK/spaCy

Hugging Face

NLP Libraries

Topic 6: **RAG (Retrieval Augmented Generation)**

- **Vector DB**
    - Embedding Generation Using OpenAI And Hugging Face Models.
    - Vector Stores: **FAISS**, **Pinecone**, **ChromaDB**, **Weaviate**, **Qdrant**
    - Similarity Search And Distance Metrics.
    - **Chunking Strategies** For Optimal Retrieval.
- **Multimodal RAG**
    - Text + Image Retrieval Pipelines.
    - Multimodal Embeddings (**CLIP**, **Vision Transformers**).
    - Use Cases In Documents And Media Search.
- **Graph RAG**
    - Knowledge Graph Construction From Documents.
    - Relationship-based Retrieval **Over Vector Search**.
- **RAG Evaluation Metrics**
    - Context Precision And Recall.
    - Answer Faithfulness And Groundedness.
    - **Latency** And **Cost Evaluation**.
    - Human-in-the-loop Evaluation Methods.

Project 13 **Custom Chatbot with Retrieval**

Tech Support

Build a domain-specific chatbot: fine-tune an LLM on company FAQs and implement retrieval-augmented generation (Chroma or Elasticsearch) to answer user queries.

OpenAI API | Langchain | Retrieval DB

Project 14 **HR Policy RAG Query Bot**

HR

Create a RAG-based chatbot that answers employee HR queries using internal company policies and guidelines.

Vector Store | Langchain | Embedding Models

Topic 7: **Langchain / Llama Index**

- **Prompt Templates** And **Output Parsers**.
- Chains And Sequential Workflows.
- Agents And Tool Calling.
- Integration With Vector Databases And APIs.
- Embed and store documents in vector DBs (**Pinecone**, **FAISS**, **Chroma**, etc.)
- Retrieve relevant chunks based on a user query
- Format the retrieved chunks as context for the LLM

Topic 8: **Fine tuning**

- **Full Fine-Tuning** vs **Parameter-Efficient Fine-Tuning (PEFT)**.
- **LoRA** And **QLoRA** Techniques.
- Instruction Fine-Tuning.
- Dataset Preparation And Labeling Remotely.
- **Overfitting Risks** And **Mitigation Strategies**.

Topic 9: **Fundamental Generative Models: GAN And VAE**

- **Generative Adversarial** Networks Architecture.
- Variational Autoencoders And Latent Space Learning.
- Use Cases In **Image And Data Generation**.

Project 15 **Synthetic Defect Image Generation**

Manufacturing

Generate synthetic defect images to augment limited manufacturing datasets for improved model training performance.

GAN

Pytorch

TensorFlow

Topic 10: **Stable Defusion**

- **Diffusion** Model Fundamentals.
- **Text-to-Image** Generation Pipelines.
- Prompt Control And **Image Conditioning.**
- Fine-Tuning Diffusion Models (**DreamBooth, LoRA**).

Project 16 **Stable Diffusion Image Generator**

Marketing

Use a Stable Diffusion model to generate creative images from text prompts; experiment with fine-tuning or style transfer.

CUDA GPU

Diffusers library

Topic 11: **Flux Framework**

- Image And Creative **Generation Workflows.**
- High-quality **Diffusion-based Outputs.**

Topic 13: **Azure OpenAI**

- **Azure OpenAI** Service Architecture.
- Model Deployment And Endpoint Management.
- **Enterprise Security** And Compliance.
- Integrating Azure OpenAI With Applications.

Project 19 **Enterprise Document Assistant**

Enterprise IT

Build a secure document Q&A system using Azure OpenAI embeddings and enterprise-grade retrieval pipelines.

Azure OpenAI

Python SDK

LangChain

Agentic AI & Automation

Duration: 1.5 Months

Module 4

Topic 1: **Understanding Agentic AI and Autonomous Agents**

- Difference Between **LLM Applications** And Agent-Based Systems.
- Autonomous Decision-Making And Goal-Oriented Agents.
- **Single-Agent vs Multi-Agent** Architectures.

Topic 2: **Key components of Agentic AI**

- Agent Roles, Goals, And Task Decomposition.
- Planning, Reasoning, And Execution Loops.
- **Tool Calling** And External System Integration.
- **Memory Types:** Short-Term, Long-Term, And Episodic Memory.
- Feedback Loops And Self-Reflection Mechanisms.
- Failure Handling And **Recovery Strategies.**

Topic 3: **AutoGen**

- **Multi-Agent** Conversation Design.
- Role-Based Agent Collaboration.
- Task Delegation And Message Passing.
- **Use Cases** For Autonomous Workflows.

Project 20 **AI Data Analyst Agent**

Fianance

Develop an autonomous agent (e.g. using LangChain or LangGraph) that connects to a database and answers data questions by generating SQL queries and charts.

LLM (GPT-4/Gemini)

Mistral Agents

Topic 4: **CrewAI**

- Team-Based **Agent Orchestration.**
- Role Assignment And Hierarchical Agents.
- **Workflow Automation** Using Crews.

Project 21 **Quality Inspector Agent**

Manufacturing

Build agent team to analyze defect images, generate insights, and suggest corrective actions collaboratively.

Vision models

OpenCV tools

CrewAI

Topic 5: **LangGraph**

- **State-Based Agent** Workflows.
- Directed Graph Execution For Agents.
- Conditional Routing And **Branching Logic.**
- Multi-Step Planning And Execution Graphs.
- **Error Handling** And Retry Mechanisms.

Project 22 **RAG Product Query Bot**

E-Commerce

Build a RAG assistant that answers product comparisons and queries using catalog data and customer reviews.

FAISS / Chroma

OpeAI LLM

Topic 6: **MCP (Model Context Protocol) / ACP / A2A**

- Model Context Protocol (**MCP**) Fundamentals.
- Agent Communication Protocol (**ACP**).
- **Agent-to-Agent** (A2A) Interaction Patterns.
- Context Sharing Across Distributed Agents.
- **Secure Context Exchange** Between Models.
- Enterprise Interoperability Standards For Agents.

Project 23 **MCP Coding Assistant & Debugger**

IT

Build MCP-powered coding assistant that reads code files, runs tests, debugs errors, and suggests fixes automatically.

MCP SDK | OpenAI LLM | AutoGen

Topic 7: **No-Code/Low-Code AI Agent Basics**

- Build AI agents using **drag-and-drop workflow** builders.
- **Automate** tasks, approvals, and decision flows without coding.
- Evaluate feasibility, cost, and business **ROI** of AI agents.
- Work confidently with tech teams to scale AI faster.

Topic 8: **Ethics And Safety In Agentic AI**

- Autonomous Agent Risk Assessment.
- Bias, **Hallucination**, And Decision Transparency.
- Human-in-the-Loop Control Mechanisms.
- Compliance And Responsible AI Guidelines.
- **GuardRail**
  - Input And Output Validation.
  - Policy Enforcement And Rule-Based Constraints.
  - Monitoring Agent Actions In Production.

Project 24 **LLM Toxicity Monitoring & Safety Guardrail Engine**

Responsible AI

Create a safety pipeline to detect toxic content, enforce prompt rules, and generate safer LLM responses.

OpenAI

Perspective API

Rule-based guardrails

Topic 9: **AWS Bedrock**

- Foundation Models Available In Bedrock.
- Building Agents Using Bedrock APIs.
- Tool Invocation And Workflow Automation.
- **Security, IAM, And Enterprise Controls.**
- Integrating Bedrock With AWS Services.

Project 25 **Autogen Vehicle Diagnostics Agent**

Automotive

Create diagnostic agent that interprets sensor logs and suggests repair actions using reasoning and tool execution.

AutoGen

Bedrock Claude

OUR PHILOSOPHY

We are a Project-Based Learning Platform

LLMOP’s & Production Deployment

Module 5

Duration: 1 Months

Topic 1: **LLM Lifecycle, Versioning & Deployment Basics**

- End-to-end LLM lifecycle from development to production.
- Model and prompt versioning using evaluation traces.
- API-based LLM deployment using FastAPI and OpenAI / Azure OpenAI endpoints.

Project 26 **Chatbot Deployment**

Tech Support

Build an LLMOps workflow to deploy, version, and monitor a chatbot using automated evaluation and logging.

FastAPI Docker LangSmith

Topic 2: **Containerization & Inference Serving**

- Packaging GenAI applications for consistent deployment.
- Serving LLM-powered APIs using containers.
- Local and cloud-ready inference setups.

Project 27 **Production-Grade LLM Application Deployment**

IT

Deploy an enterprise-ready LLM application with API endpoints, versioning, monitoring, and controlled rollouts for real-world usage.

FastAPI Docker LangSmith

Topic 3: **RAG & Agent Deployment Pipelines**

- Deploying Retrieval-Augmented Generation pipelines reliably.
- Vector database operations for production workloads.
- Deploying agent workflows with tool calling and state handling.

Project 28 **RAG Q&A Pipeline**

Knowledge Management

Build a Retrieval-Augmented Generation (RAG) pipeline using a vector database and LLM for Q&A.

Pinecone

HuggingFace

LangChain

Topic 4: **Monitoring, Cost Optimization & Reliability**

- Tracking latency, token usage, and failures.
- Prompt evaluation and trace-based monitoring.
- Cost optimization, rollback, and safe production controls.

Project 29 **Agentic AI Monitoring & Cost Optimization System**

Deploy an agent-based system that monitors LLM usage, detects performance or cost anomalies, and recommends optimization actions in production environments.

AutoGen

LangGraph

LLM

Data Structures & Algorithms

Duration: 2 Months

Module 6

Topic 1: **Time and Space Complexity**

- Understanding Performance: Importance of complexity analysis in interviews.
- Big O Notation: Common complexities and how to analyze them.
- Space Complexity: Optimizing memory usage in data structure choices.

Topic 2: **Arrays**

- Array Fundamentals & Operations: Basics of arrays, insertion, deletion, searching, updating, and two-dimensional (matrix) manipulation.
- Common Array Problems: Maximum and minimum elements, reverse array, sorted check, count occurrences, rotation, merge sorted arrays, intersections, move zeroes, and pairs with a given sum.
- Advanced Array & DP Problems: Longest increasing subsequence, maximum subarray sum, longest common subsequence, minimum path sum, and maximum product subarray.

Topic 3: **Linked Lists & Interview Problems**

- Linked List Types & Basics: Singly, doubly, and circular linked lists with creation, traversal, insertion, deletion, and use cases.
- Common Linked List Operations: Middle, length, palindrome check, reverse, rotate, merge, partition, split, remove N-th node, and loop detection.
- Advanced Linked List Problems: Add numbers, flatten multilevel doubly lists, reorder lists, and clone lists with next and random pointers.

Topic 4: **Stacks and Queues**

- Concepts & Implementation: LIFO and FIFO fundamentals, stacks and queues using arrays and linked lists.
- Advanced Variants: Priority queues and deques with common interview applications.
- Practice Coverage: Balanced parentheses, stack/queue implementations, expression evaluation (infix, postfix), reverse queue, circular queue, binary number generation, and duplicate-supporting stack.

Topic 5: **Recursion and Backtracking**

- Mastering Recursion: Key concepts and base cases crucial for problem-solving
- Divide and Conquer: Understanding its application in interviews.
- Backtracking: N-Queens, Sudoku.
- **Practice Problems:**
- Calculate the **factorial** of a number, Fibonacci sequence using recursion, Tower of Hanoi.
- Generate all subsets of a set, Solve the **N-Queens** problem, Rat in a maze.
- Solve the **Sudoku puzzle**, Permutations of a string, Word search, Generate all valid parentheses, Subset sum problem.

Topic 6: **Searching Algorithms**

- Linear vs Binary Search: Time complexities.
- Advanced Searches: Ternary search and applications.
- **Practice Problems:**
- Search Techniques: Linear vs. Binary Search with time complexity analysis.
- Advanced Searches: Ternary search and its practical applications.
- Practice Problems: Basic, intermediate, and advanced search problems covering array operations, binary search variants, order statistics, and optimization cases.

Topic 7: **Sorting Algorithms**

- Simple Sorting Algorithms: Bubble, selection, insertion—understanding their performance.
- Advanced Sorting Algorithms: Merge sort, quick sort—key algorithms in interviews.
- **Practice Problems:**
- Sorting fundamentals (bubble, selection, insertion, 0s–1s–2s, merge arrays), advanced sorting algorithms (merge, quick, heap, counting), and applied problems like nearly sorted arrays, k-th largest element, and minimum difference pairs.

Topic 8: **Hashing**

- Hash Tables and Hash Functions: Importance in optimizing search operations.
- Collision Handling Techniques: Techniques to handle collisions effectively.
- Hash Table Implementations
- **Practice Problems:**
- Frequency counting, two-sum, first non-repeating character, longest consecutive sequence, and grouping anagrams.

Topic 9: **Mastering Trees for Problem Solving**

- Binary Trees: Understanding traversal techniques (inorder, preorder, postorder).
- Binary Search Trees (BSTs)
- Balanced Trees (AVL, Red-Black): Importance of balancing in interviews.
- **Practice Problems:**
- Binary Tree Basics: Implement tree traversals and count nodes.
- Tree Queries: Find lowest common ancestor in a binary tree.
- Tree Serialization: Serialize and deserialize a binary tree.

Topic 10: **Understanding and Implementing Tries**

- Understanding Tries: Structure and its applications in string problems.
- Insertion and Search Operations
- **Practice Problems:**
- Insert and search in a trie, prefix-based word search, distinct substring counting, autocomplete system, and phone directory implementation.

Topic 11: **Heap Structures and Their Applications**

- Max-Heaps and Min-Heaps: Key properties & operations.
- Heap Applications
- **Practice Problems:**
- Build a max heap from an array, Find the maximum element in a heap.
- Implement heap sort, K-th largest element in a stream using a min heap.
- Merge k sorted linked lists, Top k frequent elements in an array.

Topic 12: **Greedy Algorithms**

- Understanding Greedy Strategy: When to apply greedy methods in problem-solving.
- **Practice Problems:**
- Coin change problem (greedy version), Activity selection problem.
- Job sequencing problem, Fractional knapsack problem.
- Huffman coding, Minimum spanning tree using Prim's or Kruskal's algorithm.

Topic 13: **Dynamic Programming (DP)**

- Introduction to DP: Recognizing overlapping subproblems and optimal substructure.
- Top-down vs. Bottom-up:
- Common DP Problems: Longest common subsequence, 0/1 knapsack
- **Practice Problems:**
- Calculate the nth Fibonacci number using DP.
- Longest increasing subsequence, Coin change problem.
- 0/1 knapsack problem, Edit distance between two strings.

Topic 14: **Graphs**

- Graph Representations: Adjacency matrix vs. adjacency list—pros and cons.
- Graph Traversal Algorithms: BFS and DFS—understanding their applications in problem-solving.
- Shortest Path Algorithms: Dijkstra's, Bellman-Ford; how to apply them in interviews.
- **Practice Problems:**
- Implement BFS and DFS for a graph.
- Detect cycles in an undirected graph, Find connected components.
- Dijkstra's algorithm for shortest path, Kruskal's or Prim's algorithm for minimum spanning tree.

BYOP **(Bring Your Own Project)**

Bring a project from your domain — we will mentor you to complete it

System Design

Duration: 2 Months

Module 7

Topic 1: AI-Powered System Design Foundations

- Introduction to System Design: Core principles of scalability, reliability, performance, and security.
- System Design Components: High-level vs. low-level design and their role in building robust systems.
- System Design Building Blocks: Key elements—databases, servers, clients, APIs, and message queues.

Topic 2: **Fundamentals of Object-Oriented Programming (OOP)**

- Core principles: Abstraction, Encapsulation, Inheritance, and Polymorphism.
- Practical applications with real-world examples to solidify understanding.
- Modular component design for ease of code maintenance.

Topic 3: **SOLID Principles for Effective Design**

- SOLID Principles Overview: In-depth understanding of Single Responsibility, Open-Closed, Liskov Substitution, Interface Segregation, and Dependency Inversion.
- Design Intent & Application: How each principle improves flexibility, maintainability, and code quality.
- Practical Exercises: Apply SOLID principles to real scenarios for clean, adaptable code.

Topic 4: **Essential Design Patterns**

- Overview of major design patterns: Creational (Factory, Singleton), Structural (Adapter, Composite), Behavioral (Observer, Strategy).
- Identifying the right pattern based on requirements and scenarios.
- Practical examples to solve common challenges with design patterns.

Topic 5: **UML Diagrams for Visual Representation**

- Introduction to essential UML diagrams, including Class, Sequence, and Component diagrams.
- Exercises in creating visual maps for clear communication of design.
- Best practices for organized, understandable diagramming.

Topic 6: **Designing Efficient Database Schemas**

- Overview of major design patterns: Creational (Factory, Singleton), Structural (Adapter, Composite), Behavioral (Observer, Strategy).
- Identifying the right pattern based on requirements and scenarios.
- Practical examples to solve common challenges with design patterns.

Topic 7: **API Design and Implementation Basics**

- RESTful API design principles: Structuring endpoints, managing errors, and versioning.
- Exercises for creating well-structured, scalable APIs with error handling.
- Best practices in API development for smooth integrations.

Topic 8: **Multithreading and Concurrency Management**

- Introduction to multithreading and concurrency control basics.
- Managing issues like race conditions and deadlocks effectively.
- Practical examples to design systems handling concurrent tasks efficiently.

Topic 9: **Basic Integration AI/ML in Software Design**

- Basics of embedding ML models and GenAI in systems for functionality enhancement.
- Setting up data flows for AI/ML and GenAI: Data preprocessing, model execution, and output integration.
- Model lifecycle management, including monitoring and updates for performance consistency.

Topic 10: **Testing and Code Quality Best Practices**

- Overview of unit and integration testing to maintain design stability.
- Techniques for documenting and structuring code for clarity and maintenance.
- Continuous monitoring and refinement for long-term code and design quality.

Topic 11: **Introduction to System Architecture and Design**

- System design fundamentals: Overview of monolithic, client-server, and microservices architectures.
- Principles of system design: Scalability, reliability, maintainability, and fault tolerance.
- Evaluating architecture choices: Choosing the right design patterns for different application needs.

Topic 12: **Distributed Systems Overview**

- Key concepts: Data replication, data consistency, and distributed communication.
- Design challenges in distributed systems: Fault tolerance, data integrity, and managing distributed transactions.
- Use cases: Examining examples of distributed systems and architectures in action.

Topic 13: **Network and Communication Protocols**

- Network fundamentals: How TCP/IP, DNS, and HTTP/HTTPS protocols function within HLD.
- Load balancing techniques: Understanding load balancers, DNS-based load balancing, and API gateways.
- Case studies: Implementing load balancing in distributed environments.

Topic 14: **ZooKeeper and Kafka (Distributed Messaging Queue)**

- ZooKeeper: Overview of distributed coordination, leader election, and configuration management.
- Apache Kafka: Introduction to message queuing, real-time data streaming, and event-driven architecture.
- Implementing a distributed messaging system: Practical applications and setup for reliable messaging.

Topic 15: **Designing with Microservices and CAP Theorem**

- Microservices architecture: Benefits and challenges of service decomposition and design.
- CAP Theorem: Trade-offs between Consistency, Availability, and Partition tolerance.
- Data partitioning: Techniques like sharding, horizontal scaling, and load distribution.

Topic 16: **Database Selection and Data Storage Models**

- Database types: SQL, NoSQL, and NewSQL, understanding their scalability and data integrity trade-offs.
- Choosing a database: ACID vs. BASE properties, and database selection based on application needs.
- Distributed databases: How they manage scalability, availability, and latency.

Topic 17: **Caching and Content Delivery Networks (CDNs)**

- Caching & Data Optimization: Use Redis and Memcached with expiration strategies to improve data access efficiency.
- CDN Integration: Apply CDNs to reduce load and deliver images and videos with lower latency.
- Project – Instagram Feed System: Design a scalable image and video feed handling high read traffic using caching and CDN techniques for better user experience.

Topic 18: **Scalability, Security, and Reliability**

- Security best practices: Authentication, data encryption, and secure API design.
- Reliability patterns.
- Scaling infrastructure: Vertical and horizontal scaling, auto-scaling, and designing for peak load handling.

Topic 19: **AI/ML Systems in High-Level Design**

- Integrating AI/ML: High-level considerations for data pipelines, model serving, and scalability for ML.
- Data infrastructure for ML: Handling large datasets, real-time processing, and model versioning.
- Real-world case studies: Examples of high-level AI/ML systems in production environments.

Program **Electives**

Choose any 2 electives

- **A.** GenAI for **Software Development** & **SDE Roles**
- **B.** AI for **Cloud, DevOps & LLM Infrastructure**
- **C.** AI Powered **Backend Engineering** for AI Systems
- **D. Distributed Systems & Microservices** for AI
- **E.** Agentic AI for Multi-Agent **Systems & Workflows**
- **F.** GenAI & Agentic AI for **Product Managers**

Elective A

LLM-Powered Software Engineering & Developer Copilots

Why Learn This?

Duration: 40 Hours

Master AI-augmented development workflows that accelerate coding, automate testing, and streamline engineering productivity, enabling you to build modern, AI-enhanced software experiences faster.

Topic 1: **AI-Augmented Coding & Software Productivity**

- AI accelerates repetitive coding tasks with consistent patterns.
- Models generate clean boilerplate supporting rapid development cycles.
- Documentation tools convert source code into human-readable summaries.
- AI suggestions reduce coding effort across common workflows.

Topic 2: **Intelligent Code Understanding, Search & Refactoring**

- Semantic models interpret large repositories with high accuracy.
- AI identifies hidden code issues using structural reasoning.
- Refactoring suggestions improve maintainability across codebases.
- Semantic search links related logic across distant files.

Topic 3: **Automated Testing, Debugging & Quality Assurance**

- AI generates strong unit tests from contextual code understanding.
- Debugging agents detect failures using learned error patterns.
- Automated QA workflows enforce code consistency during releases.
- Intelligent testing reduces regressions across evolving features.

Topic 4: **Developer Copilot & Engineering Automation Agents**

- Multi-agent copilots support complete development workflows autonomously.
- Memory systems maintain understanding across large repositories.
- Tool-calling enables automated refactoring and documentation generation.
- IDE plugins deliver real-time coding assistance with context.

Capstone Projects 1

Project 1: **AI-Driven Pull Request Reviewer** **Tools:** **GitHub API, Code LLMs, Diff Parsing**

AI evaluates code changes, identifies potential issues, and recommends structured improvements aligned with engineering practices.

Capstone Projects 2

Project 1: **Semantic Repository Search & Explanation Tool** **Tools:** **Embeddings, FAISS, AST Parsing**

Embedding-powered search retrieves relevant code snippets and explains functionality using semantic reasoning.

Elective B

Cloud, DevOps & LLM Infrastructure Engineering

Why Learn This?

Duration: 40 Hours

Learn how to deploy, monitor, and scale LLM applications on cloud infrastructure, enabling high-performance, cost-efficient, and secure GenAI systems used by enterprise teams globally.

Topic 1: **Cloud Foundations for LLM Infrastructure**

- Cloud services provision scalable compute for heavy AI workloads.
- Storage layers handle embeddings, logs, and large model artifacts.
- Serverless functions trigger lightweight AI execution pipelines.
- Identity services protect sensitive model access endpoints.

Topic 2: **Containerization & Deployment for AI Systems**

- Containers bundle AI runtimes into portable execution units.
- GPU-enabled images optimize inference throughput for models.
- Reverse proxies manage traffic for inference endpoints securely.
- CI pipelines automate container builds across multiple environments.

Topic 3: **Observability, Monitoring & System Reliability**

- Distributed tracing reveals performance issues in AI microservices.
- Log pipelines capture events from inference and agent workflows.
- Metrics visualize latency, throughput, and performance trends.
- Alerts instantly detect multi-cloud failures.

Topic 4: **Enterprise-Grade LLM Infrastructure & Governance**

- Infrastructure policies control access to deployed AI services.
- Cost governance monitors model usage across various workloads.
- Network policies secure vector stores, APIs, and inference gateways.
- Compliance workflows validate deployments against enterprise standards.

Capstone Projects 1

Project 1: Scalable LLM Inference Deployment Pipeline **Tools: Docker, Kubernetes, GPU Runtimes**

Build a fully automated pipeline deploying LLM inference services across environments with consistent performance monitoring.

Capstone Projects 2

Project 1: Cloud-Native Model Serving Architecture **Tools: API Gateway, Redis Cache, HTTPS**

Create a production-grade architecture running model APIs with load balancing, caching, and secure endpoints.

Elective C

Backend Engineering for LLM Applications & RAG Microservices

Why Learn This?

Duration: 40 Hours

Gain expertise building robust backend architectures powering LLM APIs, RAG pipelines, and distributed AI services essential for modern enterprise-grade AI product development.

Topic 1: **Backend Architecture for AI-Driven Applications**

- Backend services orchestrate LLM workflows behind APIs.
- Routing layers manage dynamic prompt construction and responses.
- Middleware organizes context injection across pipelines.
- Modular design supports flexible AI integration.

Topic 2: **High-Performance API Engineering for LLM Services**

- Async servers handle concurrent AI requests efficiently.
- Request batching reduces model latency during peak load.
- Streaming responses improve perceived responsiveness significantly.
- API gateways unify authentication across all LLM endpoints.

Topic 3: **Caching, Persistence & Storage for AI Microservices**

- Caches reduce repeated model computation for identical prompts.
- Persistent layers store structured outputs and audit trails.
- Feature stores retain embeddings and conversation context.
- Distributed storage supports multi-service AI architectures reliably.

Topic 4: **Distributed RAG & AI Microservice Patterns**

- AI microservices collaborate through event-driven pipelines.
- Retrieval microservices expose optimized document search endpoints.
- Orchestration layers coordinate multi-step LLM tasks programmatically.
- Backend agents route tasks between independent functional services.

Capstone Projects 1

Project 1: High-Performance LLM API Service **Tools: FastAPI, AsyncIO, Streaming**

Build a low-latency API service providing LLM responses using async execution and intelligent routing.

Capstone Projects 2

Project 1: Intelligent Prompt Caching Microservice **Tools: Redis, TTL Strategies, Hash Keys**

Implement a backend service caching prompts and responses, reducing repeated computation and system load.

Elective D

Distributed Systems & Microservices for AI Applications

Why Learn This?

Duration: 40 Hours

Develop the ability to design scalable distributed systems and microservice architectures supporting AI workloads, ensuring reliability, low latency, and high availability for large enterprise deployments.

Topic 1: **Distributed Systems Fundamentals for AI Workloads**

- Distributed systems handle large-scale AI traffic efficiently.
- Services communicate reliably using structured protocols and patterns.
- Replication improves availability across unpredictable workloads.
- Consensus mechanisms ensure consistency under pressure.

Topic 2: **Microservices Design for AI-Driven Platforms**

- Microservices encapsulate isolated AI responsibilities cleanly.
- Communication patterns define service interactions across boundaries.
- Circuit breakers protect systems during failures.
- Service meshes manage traffic with advanced observability.

Topic 3: **Scaling, Load Management & Fault Tolerance**

- Load balancers distribute traffic across multiple service replicas.
- Auto-scaling adapts deployments to fluctuating AI demand.
- Retry strategies reduce failure impact during transient issues.
- Fault tolerance improves reliability under distributed stress.

Topic 4: **Distributed RAG & AI Microservice Patterns**

- Distributed Data, Messaging & Workflow Coordination
- Event systems coordinate asynchronous work across AI components.
- Distributed logs maintain ordered data streams for workflows.
- Coordination services manage worker states and distributed tasks.
- Message queues provide reliable delivery for background processing.

Capstone Projects 1

Project 1: Distributed Load-Balanced AI Service **Tools: Load Balancer, Auto-Scaling**

Build a scalable AI endpoint behind a replicated microservice architecture using intelligent traffic distribution.

Capstone Projects 2

Project 1: Resilient AI Microservice Network **Tools: Service Mesh, Circuit Breakers, REST**

Implement a microservice group communicating safely using retry patterns, circuit breakers, and health checks.

Elective E

Multi-Agent Systems, Workflow Automation & Enterprise Orchestration

Why Learn This?

Duration: 40 Hours

Learn to build multi-agent systems automating complex workflows, enabling autonomous decision-making, dynamic coordination, and highly scalable enterprise automation using advanced agent orchestration patterns.

Topic 1: **Advanced Multi-Agent Collaboration Patterns**

- Agents collaborate using structured communication roles and hierarchies.
- Workflows distribute tasks across specialized agents.
- Protocols ensure predictable agent interactions.
- Delegation improves multi-step autonomy.

Topic 2: **Enterprise Workflow Automation with Agent Teams**

- Agents automate repetitive business workflows end-to-end reliably.
- Event triggers activate agent tasks across enterprise systems.
- Workflow guards enforce rules before executing critical actions.
- Cross-system automation reduces manual intervention for large operations.

Topic 3: **Context Sharing, Memory, and Tool Ecosystems**

- Agents share contextual state for consistent multi-step reasoning.
- Memory layers persist results across long-running workflows.
- Tool ecosystems allow agents to act within real systems safely.
- Structured context routing improves decision accuracy.

Topic 4: **Large-Scale Orchestration, Governance & Reliability**

- Enterprise orchestration frameworks manage thousands of parallel agents.
- Governance rules ensure compliant and traceable agent behavior.
- Monitoring dashboards visualize reasoning steps across agent workflows.
- Reliability patterns handle failures gracefully during complex automation.

Capstone Projects 1

Project 1: Context-Aware Multi-Agent Decision Engine **Tools: Context Routing, Debate Models**

Build a decision system where collaborating agents share context, debate actions, and finalize optimal outcomes.

Capstone Projects 2

Project 1: Multi-Agent Document Intelligence Pipeline **Tools: Orchestration Graphs, Data Agents**

Agent team extracts insights, validates data, and enriches documents using automated multi-step reasoning.

Elective F

GenAI & Agentic AI for Product Managers

Why Learn This?

Duration: 32 Hours

Understand how GenAI products are conceived, validated, designed, and governed, enabling product managers to lead AI initiatives, define workflows, prioritize features, and drive business impact effectively.

Topic 1: **AI-Driven Product Thinking & Use-Case Design**

- PMs identify impactful AI opportunities using structured frameworks.
- Problem discovery validates user needs before AI solutions.
- GenAI capability maps align features with model strengths.
- Prioritization frameworks guide AI roadmap decisions effectively.

Topic 2: **Designing GenAI Experiences & User Workflows**

- AI products require clear interaction patterns for usability.
- Workflow blueprints define data flow and decision points.
- Guardrails protect users from harmful or incorrect outputs.
- UX patterns support transparent and intuitive AI experiences.

Topic 3: **Agentic AI Product Architecture & Lifecycle Planning**

- Agent teams enable automated workflows for business users.
- Execution loops coordinate actions across complex tasks.
- PMs define agent roles and operational boundaries clearly.
- Lifecycle planning covers evaluation and deployment readiness.

Topic 4: **AI Governance, Risk, Evaluation & Success Metrics**

- PMs define evaluation metrics for accuracy, safety, and value.
- Governance ensures responsible behavior across AI features.
- Risk frameworks identify potential failures before deployment.
- Success metrics measure adoption, efficiency, and business impact.

Capstone Projects 1

Project 1: AI Feature Workflow Blueprint & UX Design **Tools: UX Patterns, Interaction Models**

Build an end-to-end GenAI workflow blueprint defining triggers, guardrails, UX flows, and user interactions.

Capstone Projects 2

Project 1: AI Product Requirements Document (AI-PRD) **Tools: AI-PRD, Feature Planning**

Build a complete AI-specific PRD covering scope, agents, metrics, safety, workflows, and rollout strategy.

Saurabh Kumar

From **Professor** → To **Data Scientist**

Placed at Teleperformance

135% Salary Hike

Capstone Projects

1 AI Code Review & Insights Engine

AI analyzes pull requests, detects issues, and recommends structured improvements using repository-wide context.

Code LLMs GitHub API AST Parsing

2 Autonomous Developer Copilot for Large Codebases

Multi-agent copilot assists developers by answering code questions and generating tests for entire repositories.

MCP LangGraph Code Embeddings

3 LLM-Powered Distributed Microservices Platform

Build scalable AI microservices enabling personalization and internal automation using distributed routing.

gRPC Load Balancing Async APIs

4 AI-Driven API Testing & Documentation Generator

Automatically create API tests, mocks, and documentation using natural language and interface analysis.

OpenAPI LLM Tools FastAPI

Capstone Projects

5 Cloud-Native LLM Inference Deployment System

Deploy scalable LLM inference endpoints with autoscaling, monitoring, and secured network access.

IAM · GPU Runtimes · Kubernetes

6 Enterprise AI Workflow Automation Engine

Multi-agent workflows automate incidents, tasks, root-cause analysis, and escalations across enterprise systems.

Orchestration · Event Triggers · Workflow Guards

7 Autonomous Sales Enablement Agent Network

Agents qualify leads, update CRM data, and generate personalized customer communication end-to-end.

CRM APIs · Multi-Agent Roles · Context Sharing

8 AI Market Intelligence Research Agents

Agents research competitors, summarize reports, and generate insights using multi-step reasoning workflows.

Retrieval · Reasoning Loops · Multi-Agent Collaboration

Capstone Projects

9 AI Product Requirement Analyzer & Advisor

AI analyzes PRDs, identifies gaps, and generates actionable requirement insights for product managers.

Document Parsing

LLM Evaluation

10 GenAI UX Journey Optimization Assistant

AI analyzes user behavior data and suggests optimized workflows and improved UX interactions.

UX Analytics

LLM Reasoning

11 Distributed Enterprise Retrieval Service

Build a horizontally scalable retrieval layer powering document search for large knowledge systems.

FAISS

Sharding

Distributed Indexing

12 Intelligent Load Balancer for LLM Traffic

Traffic router optimizes GPU utilization, reduces latency, and balances inference requests intelligently.

Reverse Proxies

Load Balancing

Learnbay

Thank you!

A premium learning experience for professionals committed to growth, excellence, and impact.

For more queries and information please reach out to us at:

+91 77956 87988

Visit us at

www.learnbay.co