https://eicta.iitk.ac.in/product/professional-certificate-course-in-data-engineering-2/

**Curriculum**

**Module 1: Introduction to DE**

This module provides an understanding of data engineering concepts, skills, practices, and tools essential for managing data at scale.

* What is Data Engineering?
* Role of Data Engineers in the Industry
* Importance of Data Engineering in Data-driven Organizations
* Overview of Data Engineering Tools and Technologies
* Career Paths and Opportunities in Data Engineering

**Module 2 : Python**

We will explore Python, a versatile and beginner-friendly programming language. Python is known for its readability and wide range of applications, from web development and data analysis to artificial intelligence and automation.

* Introduction to Python
* Basic Syntax and Data Types
* Control Structures (Conditional Statements and Looping)
* Functions
* Lambda Functions
* Data Structures (Lists, Tuples, Dictionaries, Sets)
* File Handling
* Error Handling (try and except)
* List Comprehensions
* Decorators
* NumPy
* Pandas
* Regex
* Code optimisation

**Module 3 : RDBMS**

We will explore RDBMS (Relational Database Management System) to understand the database technology that organizes data into structured tables with defined relationships.

* Introduction to Databases
* MYSQL -Introduction & Installation
* SQL KEYS
* PRIMARY KEY
* FOREIGN KEY
* UNIQUE KEY
* Composite key
* Normalization and Denormalization
* ACID Properties

**Module 4 : SQL**

We will dive into SQL (Structured Query Language) to acquire the skills needed for managing and querying relational databases. SQL enables them to retrieve, update, and manipulate data, making it a fundamental tool for working with structured data in various applications.

* Basic SQL Queries
* Advanced SQL Queries
* Joins (INNER JOIN, LEFT JOIN, RIGHT JOIN, FULL JOIN)
* Data Manipulation Language (DML): INSERT, UPDATE, DELETE
* Data Definition Language (DDL): CREATE, ALTER, DROP
* Data Control Language (DCL): GRANT, REVOKE
* Aggregate Functions (SUM, AVG, COUNT, MAX, MIN)
* Grouping Data with GROUP BY
* Filtering Groups with HAVING
* Subqueries
* Views
* Indexes
* Transactions and Concurrency Control
* Stored Procedures and Functions
* Triggers
* Stored procedures

**Module 5 : MongoDB**

We delve into MongoDB to understand this popular NoSQL database, which stores data in flexible, JSON-like documents. They learn how MongoDB’s scalability and speed make it suitable for handling large volumes of unstructured data.

* Introduction to NoSQL and MongoDB
* Installation and Setup of MongoDB
* MongoDB Data Model (Documents, Collections, Databases)
* CRUD Operations (Create, Read, Update, Delete)
* Querying Data with MongoDB
* Indexing and Performance Optimization
* Aggregation Framework
* Data Modeling and Schema Design
* Working with Embedded Documents and Arrays
* Transactions and Atomic Operations
* Security in MongoDB (Authentication, Authorization)
* Replication and High Availability
* Sharding and Scalability
* Backup and Disaster Recovery
* MongoDB Atlas (Cloud Database Service)
* MongoDB Compass (GUI for MongoDB)
* MongoDB Drivers and Client Libraries (e.g., pymongo for Python)
* Using MongoDB with programming languages Python
* Real-world Applications and Case Studies

**Module 6 : Shell Script**

We explore shell scripting in the Linux environment, where they learn to write and execute scripts using the command-line interface. Shell scripts are text files containing a series of commands, and We discover how to automate tasks.

* Introduction to Shell Scripting
* Basics of Shell Scripting (Variables, Comments, Quoting)
* Input/Output in Shell Scripts
* Control Structures (Conditional Statements, Loops)
* Functions and Scripts Organization
* Command Line Arguments and Options
* String Manipulation
* File and Directory Operations
* Process Management (Running Commands, Background Processes)
* Text Processing (grep, sed, awk)
* Error Handling and Exit Status
* Environment Variables
* Regular Expressions in Shell Scripts
* Debugging and Troubleshooting
* Advanced Topics (Signals, Job Control, Process Substitution)
* Shell Scripting Best Practices
* Scripting with Specific Shells (Bash, Zsh, etc.)
* Scripting for System Administration Tasks
* Scripting for Automation and Task Orchestration

**Module 7 : Git**

We will study Git, a distributed version control system, to learn how it tracks changes in software code. Git allows collaborative development, enabling multiple people to work on the same project simultaneously while managing different versions of code.

* Introduction to Version Control Systems (VCS) and Git
* Installation and Setup of Git
* Basic Git Concepts (Repositories, Commits, Branches, Merging)
* Git Workflow (Local and Remote Repositories)
* Creating and Cloning Repositories
* Git Configuration (Global and Repository-specific Settings)
* Tracking Changes with Git (git add, git commit)
* Viewing Commit History (git log)
* Branching and Merging (git branch, git merge)
* Resolving Merge Conflicts
* Working with Remote Repositories (git remote, git push, git pull)
* Collaboration with Git (Forking, Pull Requests, Code Reviews)
* Git Tags and Releases
* Git Hooks
* Rebasing and Cherry-picking
* Git Reset and Revert
* Git Stash
* Git Workflows (e.g., Gitflow, GitHub Flow)

**Module 8 : Cloud**

We delve into cloud computing, which involves delivering various computing services (such as servers, storage, databases, networking, software, and analytics) over the internet.

* Introduction to Cloud Computing and Data Engineering
* Overview of Cloud Providers (AWS and Azure)
* Cloud Storage Solutions (AWS S3, Azure Blob Storage)
* Cloud Database Services (AWS RDS, Azure SQL Database)
* Data Warehousing in the Cloud (AWS Redshift, Azure Synapse Analytics)
* Cloud Data Integration and ETL (AWS Glue, Azure Data Factory)
* Big Data Processing in the Cloud (AWS EMR, Azure HDInsight)
* Real-time Data Processing and Streaming Analytics (AWS Kinesis, Azure Stream Analytics)
* NoSQL Databases in the Cloud (AWS DynamoDB, Azure Cosmos DB)
* Data Lakes and Analytics Platforms (AWS Athena, Azure Databricks)
* Machine Learning and AI Services (AWS SageMaker, Azure Machine Learning)
* Data Visualization and BI Tools (AWS QuickSight, Azure Power BI)
* Cloud Security and Compliance
* Cost Management and Optimization in the Cloud
* Best Practices for Cloud Data Engineering

**Module 9: System Design**

The System Design provides an in-depth exploration of the principles, methodologies, and best practices involved in designing scalable, reliable, and maintainable software systems.

* Load balancer and High availability
* Horizontal vs Vertical Scaling
* Monolithic vs microservice
* Distributed messaging service and Aws SQS
* CDN (content delivery Network)
* Caching , scalability
* Aws API gateway

**Module 10 : Snowflake**

In this module, We will study Snowflake to grasp modern cloud-based data warehousing, focusing on its architecture, data sharing, scalability, and data analytics applications.

* Introduction to snowflake
* Difference between Datalake,Data Warehouse,Delta Lake,Database
* Dimension and Fact Tables
* Roles and users
* data modeling , snowpipe
* MOLAP and ROLAP
* Partitioning and indexing
* Data mart and data cubes & caching
* data masking
* handling json files
* data loading from S3 and transformation

**Module 11 : Data cleaning**

We will engage in data cleaning to understand the process of identifying and correcting errors or inconsistencies in datasets, ensuring data accuracy and reliability for analysis and reporting.

* Structured vs Unstructured Data using Pandas
* Common Data issues and how to clean them
* Data cleaning with Pandas and PySpark
* Handling Json Data
* Meaningful data transformation (Scaling and Normalization)
* Example: Movies Data Set Cleaning

**Module 12 : Hadoop**

This module provides a comprehensive introduction to Hadoop, its core components, and the broader ecosystem of tools and technologies for big data processing and analytics.

* Introduction to Big Data
* Characteristics and Challenges of Big Data
* Overview of Hadoop Ecosystem
* Hadoop Distributed File System (HDFS)
* Hadoop MapReduce Framework
* Hadoop Cluster Architecture
* Hadoop Distributed Processing
* Hadoop YARN (Yet Another Resource Negotiator)
* Hadoop Data Storage and Retrieval
* Hadoop Data Processing and Analysis
* Hadoop Streaming for Real-time Data Processing
* Hadoop Ecosystem Components:
  + HBase for NoSQL Database
  + Hive for Data Warehousing and SQL
  + Pig for Data Flow Scripting
  + Spark for In-memory Data Processing
  + Sqoop for Data Import/Export
  + Flume for Data Ingestion
  + Oozie for Workflow Management
  + Kafka for Real-time Data Streaming
* Hadoop Security and Governance

**Module 13 : Kafka**

In this module, We learn about Kafka, an open-source stream processing platform. Kafka is used for ingesting, storing, processing, and distributing real-time data streams and explores Kafka’s architecture, topics, producers, consumers, and its role in handling large volumes of data with low latency.

* Introduction to kafka
* producer, consumer, Consumer Groups
* topics , offset , partitions, brokers
* Zookeeper,replication
* Batch vs real time streaming
* real streaming process
* Assignment and Task

**Module 14 : Spark**

In this module, We will explore Spark, which is an open-source, distributed computing framework that provides high-speed, in-memory data processing for big data analytics.

* Introduction to Apache Spark
* Features and Advantages of Spark over Hadoop MapReduce
* Spark Architecture Overview
* Resilient Distributed Datasets (RDDs)
* Directed Acyclic Graph (DAG) Execution Engine
* Spark Core and Spark SQL
* DataFrames and Datasets in Spark
* Spark Streaming for Real-time Data Processing
* Structured Streaming for Continuous Applications
* Machine Learning with MLlib in Spark
* Graph Processing with GraphX in Spark
* Spark Performance Tuning and Optimization Techniques
* Integrating Spark with Other Big Data Technologies (Hive, HBase, Kafka, etc.)
* Spark Deployment Options (Standalone, YARN, Mesos)
* Spark Cluster Management and Monitoring

**Module 15 : Airflow**

Here, We will explore Airflow to understand its role in orchestrating and automating workflows, scheduling tasks, managing data pipelines, and monitoring job execution.

* Why and what is airflow
* airflow UI
* Run first dag
* grid view
* graph view
* landing times view
* calendar view
* gantt view
* Code view
* Core concepts of airflow
* DAGs
* Scope
* Operators
* control flow
* Task and task instance
* Database and executors
* ETL/ELT process implementation
* monitoring ETL pipeline with airflow

**Module 16 : DataBricks**

This module provides a comprehensive introduction to DataBricks.You will learn how to leverage DataBricks to build and deploy scalable data pipelines.

* Introduction to Databricks
* Overview of Databricks Unified Analytics Platform
* Setting up Databricks Environment
* Databricks Workspace: Notebooks, Clusters, and Libraries
* Spark Architecture in Databricks
* Spark SQL and DataFrame Operations in Databricks Notebooks
* Data Import and Export in Databricks
* Working with Delta Lake for Data Versioning and Transaction Management
* Performance Optimization Techniques in Databricks
* Advanced Analytics and Machine Learning with MLlib in Databricks
* Collaboration and Sharing in Databricks Workspace
* Monitoring and Debugging Spark Jobs in Databricks
* Integrating Databricks with Other Data Engineering Tools and Services

**Module 17 : Prometheus**

We will study Prometheus to explore its role as an open-source monitoring and alerting toolkit, used for collecting and visualizing metrics from various systems, aiding in performance optimization and issue detection.

* Introduction to Prometheus
* Prometheus Server and Architecture
* Installation and Setup of Prometheus
* Understanding Prometheus UI (User Interface)
* Node Exporters: Monitoring System Metrics
* Prometheus Query Language (PromQL) for Aggregation, Functions, and Operators
* Integrating Python Applications with Prometheus for Custom Metrics
* Key Metric Types: Counter, Gauge, Summary, and Histogram
* Recording Rules for Pre-computed Metrics
* Alerting Rules for Generating Alerts
* Alert Manager: Installation and Configuration

**Module 18 : Data dog**

We will study about Datadog, a monitoring and analytics platform for cloud-scale applications. It provides developers, operations teams, and business users with insights into their applications, infrastructure, and overall performance.

* Metrics
* Dashboards
* Alerts
* Monitors
* Tracing
* Logs monitoring
* Integrations

**Module 19 : Docker**

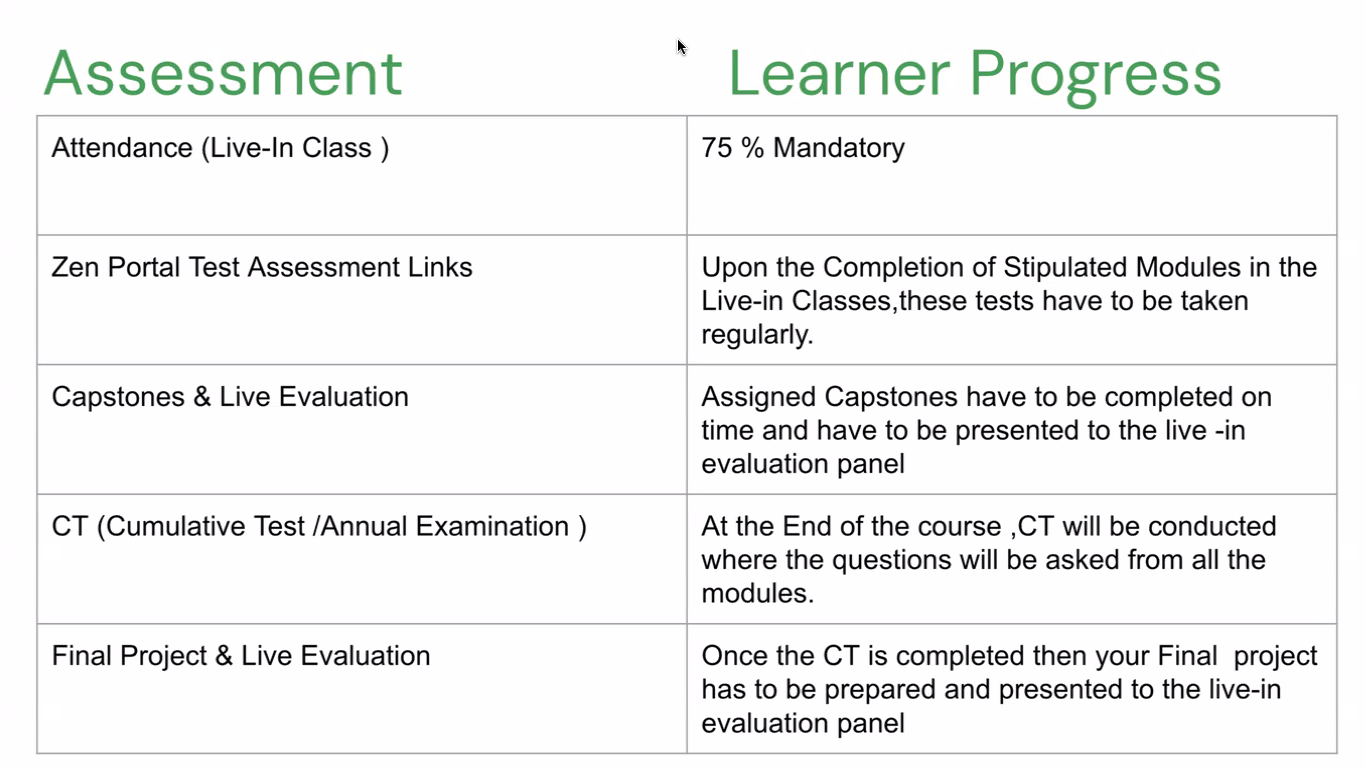
In this module, we will cover Docker, an open-source platform used to develop, ship, and run applications in containers. Containers are lightweight, portable, and self-sufficient units that package an application along with its dependencies, libraries, and configuration files, enabling consistent deployment across different environments.

* What is docker
* Installation of docker
* Docker images , containers
* Docker file
* Docker volume
* Docker registry
* Containerizing applications with docker hands-on

**Module 20 : Kubernetes**

This module provides a comprehensive introduction to Kubernetes, an open-source container orchestration platform for automating deployment, scaling, and management of containerized applications.

* Nodes
* Pods
* ReplicaSets
* Deployments
* Namespaces
* Ingress



A green rectangular boxes with text

Description automatically generated

A screenshot of a white background with green text

Description automatically generated

A screenshot of a computer

Description automatically generated

[Google Forms: Sign-in](https://accounts.google.com/InteractiveLogin/signinchooser?continue=https%3A%2F%2Fdocs.google.com%2Fforms%2Fd%2Fe%2F1FAIpQLSdeu7_VUElX3q2YXDtlf5-l4C9sI5Z8jBcdv3QfA5EwvjbDaw%2Fviewform&followup=https%3A%2F%2Fdocs.google.com%2Fforms%2Fd%2Fe%2F1FAIpQLSdeu7_VUElX3q2YXDtlf5-l4C9sI5Z8jBcdv3QfA5EwvjbDaw%2Fviewform&ltmpl=forms&osid=1&passive=1209600&service=wise&ifkv=AcMMx-eKQxc6Kbr6Sq2KzWdnPgJo9Ag0T55FaMfmCiM_Ex6K-EqUyEsW6ntzpK6AMggASzWJOCg0&ddm=1&flowName=GlifWebSignIn&flowEntry=ServiceLogin)