



CLASSROOM & ONLINE TRAINING

+91 94916 28222
+91 94916 29222

DATA ANALYTICS DATA SCIENCE

with **GEN AI & ML**

ATTEND FREE DEMO ➔



★ ★ ★
**BUILD SKILLS.
ANALYZE DATA.
SHAPE YOUR
FUTURE.**

TOOLS YOU WILL LEARN



Microsoft
Excel



SQL



Python



Microsoft
Power BI



Statistics



Tableau



Machine
Learning



Generative AI



Data
Science



GET FREE DEMO

LIMITED SEATS • REGISTER TODAY!



CLASSROOM &
ONLINE TRAINING

REGISTER NOW ➔



EXPERT
TRAINERS



REAL-TIME
PROJECTS



PLACEMENT
ASSISTANCE



CERTIFICATION
PROVIDED

Learn • Analyze • Predict • **Grow with AI**

Full Stack Data Analytics (FSDA)

(Excel + Python + Power BI + SQL + Statistics + Tableau)

- **Add-on Module 1:** PySpark & Spark SQL
- **Add-on Module 2:** Generative AI

Data Analysis Using Generative Artificial Intelligence (Gen AI)

Course Contents:

Contents

Microsoft Excel	6
Excel Fundamentals	6
Introduction to Excel.....	6
Data Handling & Preparation	6
Sorting & Filtering.....	6
Basic Formatting.....	7
Flash Fill (Quick Transformations)	7
Data Cleaning in Excel.....	7
Data Highlighting & Insights	7
Conditional Formatting	7
Excel Functions for Data Analysis.....	7
Text Functions	7
Statistical Functions.....	7
Logical Functions.....	7
Data Lookup & Retrieval.....	8
LOOKUP Functions	8
INDEX & MATCH (Advanced Lookup)	8
Data Validation & Control	8
Data Analysis & Reporting	8
Pivot Tables.....	8
Power Query.....	8
Automation in Excel.....	9
Recording Macros	9
Data Visualization in Excel.....	9
Excel using AI	9
Tools Covered	9
Statistics.....	9

Descriptive Statistics (Understanding Data)	9
Measures of Central Tendency	9
Measures of Dispersion (Data Spread)	10
Data Distribution (Understanding Patterns).....	10
Normal Distribution	10
Skewness	10
Binomial Distribution	10
Poisson Distribution.....	10
Percentiles C Quartiles.....	10
Probability Fundamentals	10
Correlation C Covariance	11
Correlation.....	11
Covariance	11
Z-Score (Standardization).....	11
Sampling C Sampling Distribution	11
Hypothesis Testing.....	11
Tools Covered:	11
SQL (Structured Query Language)	11
Database Fundamentals.....	12
Introduction to SQL	12
Database Design.....	12
Data Warehousing Concepts	12
SQL Core Concepts	12
SQL Data Types.....	12
SQL Operators	12
SQL Commands.....	12
DDL (Data Definition Language)	12
DML (Data Manipulation Language).....	13
DQL (Data Query Language)	13
DCL (Data Control Language).....	13
TCL (Transaction Control Language)	13
Filtering C Sorting Data	13

Aggregations C Grouping.....	13
Joins.....	13
Advanced SQL for Data Analysis	13
Window Functions	13
LEAD() C LAG() Functions	13
Frames in SQL.....	14
Conditional Logic.....	14
Subqueries C CTEs.....	14
Subqueries	14
Common Table Expressions (CTEs)	14
Database Objects.....	14
Views	14
Triggers	14
Query Optimization C Performance	14
Structured Problem Solving in SQL.....	15
Advanced SQL Queries	15
SQL Using AI.....	15
Real-Time Project: IMDB Movies Analysis	15
Tools Covered	15
Microsoft Power BI	15
Tableau	17
Tableau Fundamentals.....	17
Introduction to Tableau.....	17
Tableau Installation.....	17
Tableau Workflow C Interface.....	17
Data Understanding in Tableau	18
Tableau Data Types.....	18
Tableau Operators.....	18
Data Transformation C Calculations.....	18
Calculated Fields.....	18
Parameters	18
Groups C Sets.....	18

Tableau Functions	18
Data Visualization Techniques	18
Basic Charts	18
Advanced Charts	19
Advanced Visualizations.....	19
Special Charts.....	19
Data Analysis C Insights	19
Dashboard Creation.....	19
Storytelling in Tableau	19
Real-Time Project: IPL Data Visualization.....	19
Tools Covered:	20
Python	20
Core Python Fundamentals.....	20
Introduction to Python.....	20
Installation C Setup	20
Variables C Data Types.....	20
Operators in Python.....	20
Arithmetic C String Operations	21
Control Flow C Functions	21
Conditional Statements.....	21
Loops.....	21
User Input Handling.....	21
Functions.....	21
Data Structures for Data Handling	21
Strings.....	21
Lists.....	21
Tuples	21
Sets	21
Dictionaries.....	22
Numerical Computing with NumPy	22
Data Analysis with Pandas	22
Data Visualization	22

Matplotlib.....	22
Seaborn	22
Data Cleaning C Preprocessing	22
Exploratory Data Analysis (EDA)	23
Univariate Analysis	23
Bivariate Analysis	23
Multivariate Analysis.....	23
Python using AI.....	23
Real-Time Project: Risk Analysis for Banking Data.....	23
Tools Covered	23
PySpark.....	24
PySpark Foundations.....	24
Spark Architecture.....	24
PySpark Setup C Environment	24
DataFrames.....	24
Reading C Writing Data.....	25
Read:	25
Write:	25
DataFrame Transformations	25
Data Cleaning in PySpark	25
Aggregations C Grouping.....	25
Joins.....	25
Spark SQL for Analysts	26
Window Functions	26
Date C Time Functions.....	26
Pivot C Unpivot.....	26
Performance Optimization	26
PySpark vs Pandas	26
PySpark using AI.....	27
Real Time Project using PySpark.....	27
Tools Covered	27
Generative AI.....	27

Landscape of Generative AI	27
Working with OpenAI APIs	27
Chat Completions API	27
Prompt Engineering.....	27
Retrieval Augmented Generation (RAG)	28
Vector Stores.....	28
AI + SQL	28
AI + Python	28
LLM Frameworks/LangChain	28
Tools Covered	28

Microsoft Excel

Excel serves as a widely-used tool for data analysis, offering functionalities such as sorting, filtering, and basic statistical calculations. Its user-friendly interface and spreadsheet- based structure make it accessible for organizing and analyzing datasets, making it a popular choice for small-scale data tasks and quick analyses.

Excel Fundamentals

Introduction to Excel

- ✓ Importance of Excel in Data Analysis
- ✓ Real-world use cases (reporting, dashboards, ad-hoc analysis)
- ✓ Understanding workbook, worksheets, rows, and columns
- ✓ Navigation and shortcuts for productivity

Data Handling s Preparation

Sorting C Filtering

- ✓ Sorting data (single C multi-level sorting)
- ✓ Applying filters for data exploration
- ✓ Advanced filtering techniques

Basic Formatting

- ✓ Cell formatting (number, date, currency)
- ✓ Font styling, alignment, borders
- ✓ Formatting best practices for reports

Flash Fill (Quick Transformations)

- ✓ Automatically extracting and transforming data
- ✓ Pattern recognition for fast data cleaning

Data Cleaning in Excel

- ✓ Removing duplicates
- ✓ Handling missing values
- ✓ Text-to-columns
- ✓ TRIM, CLEAN for data standardization

Data Highlighting s Insights

Conditional Formatting

- ✓ Highlighting key insights (top/bottom values, duplicates)
- ✓ Color scales, data bars, icon sets
- ✓ Rule-based formatting for dashboards

Excel Functions for Data Analysis

Text Functions

- ✓ LEFT(), RIGHT(), MID()
- ✓ LEN(), TRIM(), UPPER(), LOWER()
- ✓ CONCAT(), TEXT()
- ✓ Real-world use cases (cleaning messy data)

Statistical Functions

- ✓ SUM(), AVERAGE(), COUNT(), MAX(), MIN()
- ✓ MEDIAN(), MODE()
- ✓ Basic descriptive statistics

Logical Functions

- ✓ IF(), Nested IF
- ✓ AND(), OR(), NOT()
- ✓ Building decision-based conditions

Data Lookup s Retrieval

LOOKUP Functions

- ✓ VLOOKUP(), HLOOKUP(), XLOOKUP()
- ✓ Approximate vs exact match
- ✓ Limitations of VLOOKUP

INDEX C MATCH (Advanced Lookup)

- ✓ Dynamic lookups using INDEX + MATCH
- ✓ Replacing VLOOKUP

Data Validation s Control

- ✓ Creating drop-down lists
- ✓ Restricting data input (numbers, dates, text length)
- ✓ Error alerts and input messages
- ✓ Ensuring data accuracy in reports

Data Analysis s Reporting

Pivot Tables

- ✓ Creating Pivot Tables from datasets
- ✓ Summarizing large datasets
- ✓ Grouping (dates, categories)
- ✓ Filtering and sorting within Pivot Tables
- ✓ Calculated fields and values

Power Query

- ✓ Introduction to Power Query
- ✓ Importing data from multiple sources (CSV, Excel, Web)
- ✓ Removing columns
- ✓ Filtering rows
- ✓ Changing data types
- ✓ Merging and appending datasets

Automation in Excel

Recording Macros

- ✓ Automating repetitive tasks
- ✓ Recording and running macros
- ✓ Assigning macros to buttons
- ✓ Basic understanding of VBA

Data Visualization in Excel

- ✓ Creating charts:
 - Bar chart, line chart, pie chart, column chart
- ✓ Customizing visuals:
 - Titles, labels, legends, colors
- ✓ Best practices for clear visualization

Excel using AI

Tools Covered: Microsoft Excel

Statistics

Statistics is the backbone of data analysis, enabling data analysts to extract meaningful insights from raw data. It provides tools to summarize data (descriptive statistics), make predictions (inferential statistics), and identify trends or patterns. By understanding statistical measures such as mean, variance, and percentiles, data analysts can interpret datasets, validate hypotheses, and make data-driven decisions crucial for business strategies.

Descriptive Statistics (Understanding Data)

Measures of Central Tendency

- ✓ Mean (Average)
- ✓ Median
- ✓ Mode
- ✓ When to use each measure
- ✓ Impact of outliers on central tendency

Measures of Dispersion (Data Spread)

- ✓ Range
- ✓ Variance
- ✓ Standard Deviation
- ✓ Interquartile Range (IQR)
- ✓ Understanding variability in datasets
- ✓ Identifying consistency and volatility in data

Data Distribution (Understanding Patterns)

Normal Distribution

- ✓ Bell curve concept
- ✓ Real-world examples (heights, marks)

Skewness

- ✓ Left-skewed vs right-skewed data
- ✓ Impact on analysis

Binomial Distribution

- ✓ Success/failure scenarios
- ✓ Use cases in business decisions

Poisson Distribution

- ✓ Event occurrence over time
- ✓ Real-time examples (customer arrivals, defects)

Percentiles & Quartiles

- ✓ Understanding percentiles
- ✓ Quartiles (Q1, Q2, Q3)
- ✓ Interpreting data positions
- ✓ Use cases in ranking and segmentation

Probability Fundamentals

- ✓ Basics of probability
- ✓ Probability rules

Correlation s Covariance

Correlation

- ✓ Measures relationship between two variables
- ✓ Positive, Negative, and No correlation
- ✓ Correlation coefficient range (-1 to +1)

Covariance

- ✓ Direction of relationship between variables
- ✓ Positive vs Negative covariance
- ✓ Difference between covariance and correlation

Z-Score (Standardization)

- ✓ Understanding how far a data point is from the mean
- ✓ Standardizing different datasets for comparison
- ✓ Identifying outliers using Z-score

Sampling s Sampling Distribution

- ✓ What is sampling?
- ✓ Types of sampling (random, stratified, etc.)
- ✓ Importance of sampling in real-world data

Hypothesis Testing

- ✓ Null Hypothesis (H_0)
- ✓ Alternative Hypothesis (H_1)
- ✓ P-value concept
- ✓ Significance level (alpha)
- ✓ Type I Error C Type II Error
- ✓ Steps in hypothesis testing

Tools Covered: No Software tool required

SQL (Structured Query Language)

SQL plays a crucial role in data analysis by enabling users to extract,

manipulate, and analyze data stored in relational databases efficiently. Its declarative nature allows for complex data queries, aggregations, and transformations, facilitating the extraction of meaningful insights from structured datasets.

Database Fundamentals

Introduction to SQL

- ✓ What is SQL and why it is important for Data Analysts
- ✓ Role of SQL in real-world analytics workflows

Database Design

- ✓ Tables, rows, columns
- ✓ Primary keys and foreign keys
- ✓ Normalization and Denormalization

Data Warehousing Concepts

- ✓ OLTP vs OLAP systems
- ✓ Introduction to Data Warehousing
- ✓ Fact and Dimension tables
- ✓ Star Schema vs Snowflake Schema
- ✓ Importance of data modeling in analytics

SQL Core Concepts

SQL Data Types

- ✓ Numeric, String, Date/Time types
- ✓ Choosing appropriate data types

SQL Operators

- ✓ Arithmetic, Comparison, Logical operators
- ✓ Real-time filtering use cases

SQL Commands

DDL (Data Definition Language)

- ✓ CREATE, ALTER, DROP

DML (Data Manipulation Language)

- ✓ INSERT, UPDATE, DELETE

DQL (Data Query Language)

- ✓ SELECT statements

DCL (Data Control Language)

- ✓ GRANT, REVOKE

TCL (Transaction Control Language)

- ✓ COMMIT, ROLLBACK

Filtering s Sorting Data

- ✓ WHERE clause
- ✓ HAVING clause
- ✓ ORDER BY
- ✓ LIMIT C OFFSET
- ✓ Writing efficient filtering conditions

Aggregations s Grouping

- ✓ GROUP BY clause
- ✓ Aggregate functions: COUNT, SUM, AVG, MIN, MAX
- ✓ Real-world reporting scenarios

Joins

- ✓ Inner Join
- ✓ Left Join
- ✓ Right Join
- ✓ Full Join

Advanced SQL for Data Analysis

Window Functions

- ✓ OVER clause
- ✓ Partitioning and ordering

LEAD() C LAG() Functions

- ✓ Time-based analysis

- ✓ Comparing previous and next records

Frames in SQL

- ✓ ROWS BETWEEN
- ✓ Running totals and moving averages

Conditional Logic

- ✓ CASE statements
- ✓ Creating derived columns
- ✓ Business rule implementation

Subqueries s CTEs

Subqueries

- ✓ Nested queries
- ✓ Correlated vs non-correlated subqueries

Common Table Expressions (CTEs)

- ✓ Writing modular and readable queries

Database Objects

Views

- ✓ Creating reusable query layers
- ✓ Simplifying complex queries

Triggers

- ✓ Automating actions on data changes
- ✓ Use cases (audit logs, validations)

Query Optimization s Performance

- ✓ Understanding query execution basics
- ✓ Indexing concepts (high-level)
- ✓ Writing optimized queries
- ✓ Avoiding common performance issues

Structured Problem Solving in SQL

- ✓ Breaking down complex problems
- ✓ Step-by-step query building approach
- ✓ Writing clean and efficient SQL logic
- ✓ Handling interview-style questions

Advanced SQL Queries

- ✓ Complex joins + aggregations
- ✓ Multi-level subqueries
- ✓ Real-world business problem scenarios
- ✓ Case study-based query solving

SQL Using AI

- ✓ Writing SQL Queries using AI

Real-Time Project: IMDB Movies Analysis

- ✓ Understanding dataset structure
- ✓ Data cleaning and preparation
- ✓ Writing analytical queries:
 - Top-rated movies
 - Genre analysis
 - Revenue trends
 - Actor/Director performance
- ✓ Applying joins, aggregations, and window functions
- ✓ Generating business insights

Tools Covered: MySQL Workbench

Microsoft Power BI

Power BI, Microsoft's business analytics tool, empowers users to visualize and analyze data through intuitive dashboards and reports, offering a seamless integration with various data sources. With robust features for

data modeling, transformation, and advanced analytics, Power BI facilitates data-driven insights and decision-making processes for businesses of all sizes.

- ✓ Overview of Power BI
- ✓ Power BI vs Tableau vs Excel
- ✓ Power BI ecosystem: Power BI Desktop, Service, and Mobile
- ✓ Exploring the Power BI interface and basic components
- ✓ Installation and setup of Power BI Desktop
- ✓ Supported data sources (Excel, SQL Server, Web, APIs, etc.)
- ✓ Import vs. DirectQuery mode
- ✓ Connecting to on-premises and cloud data sources
- ✓ Data Gateway for on-premises data refresh
- ✓ Best practices for data connectivity
- ✓ Introduction to Power Query Editor
- ✓ Data cleaning techniques (removing duplicates, handling nulls, etc.)
- ✓ Merging & appending queries
- ✓ Column transformations (split, merge, extract)
- ✓ Understanding star and snowflake schema
- ✓ Creating and managing relationships between tables
- ✓ Cardinality and cross-filter direction
- ✓ Optimizing models for performance
- ✓ Role of Fact and Dimension tables
- ✓ Introduction to DAX (Data Analysis Expressions)
- ✓ Measures vs. Calculated Columns
- ✓ Common DAX functions (SUMX, CALCULATE, FILTER, etc.)
- ✓ Time intelligence functions (YTD, MTD, QTD)
- ✓ Performance optimization for DAX
- ✓ Designing effective dashboards and reports
- ✓ Using slicers, filters, and bookmarks
- ✓ Advanced visualizations (custom visuals, drill-through, tooltips)
- ✓ Themes and branding in reports
- ✓ Publishing reports to Power BI Service
- ✓ AI Visuals
- ✓ Introduction to Power BI Service
- ✓ Creating and managing workspaces
- ✓ Dataflows and dataset management

- ✓ Scheduling data refresh and subscriptions
- ✓ Power BI Apps and content packs
- ✓ Introduction to security in Power BI
- ✓ Implementing Row-Level Security (static & dynamic)
- ✓ User roles and access control
- ✓ Sharing reports securely
- ✓ Best practices for governance and compliance
- ✓ Best Practices in Power BI
- ✓ Bookmarks and Performance Analyzer
- ✓ Creating an Interactive Dashboard for Home Loans (Real Time Live Project)

Tools Covered: Power BI Desktop, Power BI Service

Tableau

Tableau is a powerful data visualization tool that enables users to create interactive and visually appealing dashboards from various data sources. Its intuitive drag-and-drop interface and extensive customization options make it ideal for exploring and presenting complex datasets, facilitating data-driven decision-making across organizations.

Tableau Fundamentals

Introduction to Tableau

- ✓ Importance of data visualization in analytics
- ✓ Role of Tableau in business intelligence
- ✓ Real-world use cases (dashboards, reporting, decision-making)

Tableau Installation

- ✓ Installing Tableau Public/Desktop
- ✓ Overview of Tableau environment

Tableau Workflow & Interface

- ✓ Data pane, worksheet, dashboard, story
- ✓ Connecting to data sources
- ✓ Understanding rows, columns, marks card

Data Understanding in Tableau

Tableau Data Types

- ✓ Dimensions vs Measures
- ✓ Discrete vs Continuous fields
- ✓ Data roles and hierarchies

Tableau Operators

- ✓ Arithmetic, logical, comparison operators
- ✓ Use in calculated fields

Data Transformation s Calculations

Calculated Fields

- ✓ Creating custom calculations
- ✓ Row-level vs aggregate calculations
- ✓ Real-world business logic implementation

Parameters

- ✓ Creating dynamic controls
- ✓ Using parameters in calculations and dashboards

Groups C Sets

- ✓ Grouping data for simplified analysis
- ✓ Creating sets for advanced filtering and segmentation

Tableau Functions

- ✓ Aggregate functions (SUM, AVG, COUNT)
- ✓ String functions
- ✓ Date functions
- ✓ Logical functions
- ✓ Table calculations (running total, % of total)

Data Visualization Techniques

Basic Charts

- ✓ Bar Chart
- ✓ Line Chart

- ✓ Pie Chart

Advanced Charts

- ✓ Scatter Plot (relationship analysis)
- ✓ Histogram (distribution analysis)
- ✓ Box Plot (outlier detection)
- ✓ Area Chart (trend over time)

Advanced Visualizations

- ✓ Tree Map (hierarchical data)
- ✓ Heat Map (intensity-based insights)
- ✓ Packed Bubble Chart (proportional comparison)
- ✓ Stacked Bar Chart

Special Charts

- ✓ Dual Axis Chart (comparison of multiple measures)

Data Analysis s Insights

- ✓ Sorting and filtering in Tableau
- ✓ Drill-down and hierarchies
- ✓ Highlighting and tooltips
- ✓ Applying filters (context, quick filters)
- ✓ Creating interactive analysis

Dashboard Creation

- ✓ Combining multiple worksheets
- ✓ Layout design and formatting
- ✓ Adding filters, actions, and interactivity
- ✓ Using containers for proper alignment
- ✓ Best practices for dashboard design

Storytelling in Tableau

- ✓ Creating Stories
- ✓ Building a data narrative
- ✓ Presenting insights step-by-step
- ✓ Communicating findings effectively

Real-Time Project: IPL Data Visualization

- ✓ Understanding IPL dataset

- ✓ Data cleaning and preparation
- ✓ Creating dashboards:
 - Team performance analysis
 - Player statistics
 - Match trends and insights
- ✓ Applying filters, parameters, and interactivity
- ✓ Building a complete interactive dashboard
- ✓ Presenting business insights

Tools Covered: Tableau Public

Python

Python serves as a powerhouse for data analysis, offering robust libraries like Pandas for manipulation, NumPy for numerical computing, and Matplotlib for visualization. With its rich ecosystem and easy-to-use syntax, Python empowers data analysts to efficiently explore, process, and visualize datasets, driving insights and decision-making

Core Python Fundamentals

Introduction to Python

- ✓ Importance of Python in Data Analysis
- ✓ Real-world use cases in analytics and business decision-making

Installation & Setup

- ✓ Installing Anaconda Distribution
- ✓ Overview of Jupyter Notebook for analysis

Variables & Data Types

- ✓ Integer, Float, String, Boolean
- ✓ Type casting and dynamic typing
- ✓ Understanding memory and variable assignment

Operators in Python

- ✓ Arithmetic, Comparison, Logical, Assignment operators
- ✓ Real-time use cases in data calculations

Arithmetic C String Operations

- ✓ Mathematical computations
- ✓ String manipulation for data cleaning (split, replace, format)

Control Flow s Functions

Conditional Statements

- ✓ if, elif, else for decision-making logic
- ✓ Real-time filtering scenarios

Loops

- ✓ for loops for iteration over datasets
- ✓ while loops for condition-based execution
- ✓ Practical use cases in data processing

User Input Handling

- ✓ Taking dynamic input for flexible programs

Functions

- ✓ Creating reusable functions
- ✓ Parameters, return values
- ✓ Writing modular and efficient data processing logic

Data Structures for Data Handling

Strings

- ✓ Indexing, Slicing
- ✓ String Functions

Lists

- ✓ Indexing, slicing, looping
- ✓ Use in storing and processing data collections

Tuples

- ✓ Immutable data structures and their use cases

Sets

- ✓ Removing duplicates, set operations

Dictionaries

- ✓ Key-value data storage
- ✓ JSON-like data handling (very important for APIs & real-world data)

Numerical Computing with NumPy

- ✓ Introduction to NumPy arrays
- ✓ Array creation and operations
- ✓ Mathematical and statistical functions
- ✓ Vectorized operations for performance optimization
- ✓ Use cases in large-scale numerical computations

Data Analysis with Pandas

- ✓ Introduction to Series and DataFrames
- ✓ Reading data from CSV, Excel, JSON
- ✓ Data inspection: head(), info(), describe()
- ✓ Data selection and filtering
- ✓ Handling missing values
- ✓ Data transformation and feature engineering
- ✓ GroupBy operations and aggregations
- ✓ Merging and joining datasets
- ✓ Sorting and ranking data

Data Visualization

Matplotlib

- ✓ Line charts, bar charts, histograms
- ✓ Customization (labels, titles, colors)

Seaborn

- ✓ Advanced visualizations (heatmaps, pairplots, boxplots)
- ✓ Statistical plotting for deeper insights

Data Cleaning & Preprocessing

- ✓ Identifying and handling missing values
- ✓ Removing duplicates

- ✓ Data type corrections
- ✓ Outlier detection and treatment
- ✓ String cleaning and formatting
- ✓ Preparing raw data for analysis

Exploratory Data Analysis (EDA)

Univariate Analysis

- ✓ Distribution analysis (mean, median, mode)

Bivariate Analysis

- ✓ Relationships between two variables (correlation, scatter plots)

Multivariate Analysis

- ✓ Analyzing multiple variables together
- ✓ Pattern identification and insights generation

Python using AI

- ✓ Performing Data Analysis using Python with the help of AI

Real-Time Project: Risk Analysis for Banking Data

- ✓ Understanding business problem (loan risk / credit analysis)
- ✓ Data cleaning and preprocessing of banking dataset
- ✓ Exploratory Data Analysis (EDA)
- ✓ Identifying high-risk customers
- ✓ Feature analysis and insights generation
- ✓ Creating visual dashboards using Python
- ✓ Presenting actionable business insights

Tools Covered: Jupyter Notebook, Google Collab

PySpark

PySpark (Big Data processing framework) empowers data analysts to efficiently process, clean, and transform large-scale datasets across distributed environments. Its ability to handle massive structured and semi-structured data enables analysts to perform advanced analytics and build scalable data pipelines with high performance and flexibility.

PySpark Foundations

- ✓ What is Big Data?
- ✓ Limitations of traditional tools (Excel, Pandas)
- ✓ Introduction to Apache Spark
- ✓ PySpark vs Pandas
- ✓ Spark Ecosystem overview

Spark Architecture

- ✓ Driver vs Executors
- ✓ Cluster vs Local mode
- ✓ Lazy Evaluation
- ✓ DAG (Directed Acyclic Graph)
- ✓ Transformations vs Actions

PySpark Setup s Environment

- ✓ Installing PySpark locally
- ✓ Running PySpark in Google Collab
- ✓ Creating SparkSession

DataFrames

- ✓ What is DataFrame?
- ✓ Creating DataFrames:
 - From list
 - From CSV/JSON/Parquet
- ✓ Schema understanding
- ✓ Show, printSchema, describe

Reading & Writing Data

Read:

- ✓ CSV
- ✓ JSON
- ✓ Parquet

Write:

- ✓ CSV
- ✓ Parquet

DataFrame Transformations

- ✓ select()
- ✓ filter()
- ✓ withColumn()
- ✓ drop()
- ✓ rename columns
- ✓ distinct()
- ✓ orderBy()

Data Cleaning in PySpark

- ✓ Handling null values:
 - dropna()
 - fillna()
- ✓ Removing duplicates
- ✓ Data type casting
- ✓ String functions
- ✓ Regex operations

Aggregations & Grouping

- ✓ groupBy()
- ✓ agg()
- ✓ count, sum, avg, min, max

Joins

- ✓ Inner Join

- ✓ Left Join
- ✓ Right Join
- ✓ Full Join

Spark SQL for Analysts

- ✓ Creating Temp Views
- ✓ Running SQL queries
- ✓ Joins using SQL
- ✓ Aggregations using SQL

Window Functions

- ✓ row_number()
- ✓ rank()
- ✓ dense_rank()
- ✓ lead(), lag()

Date s Time Functions

- ✓ to_date()
- ✓ date_format()
- ✓ current_date()
- ✓ datediff()

Pivot s Unpivot

- ✓ Pivot tables

Performance Optimization

- ✓ Lazy evaluation recap
- ✓ Caching C Persisting
- ✓ Repartition vs Coalesce
- ✓ Avoiding UDFs when possible
- ✓ Using built-in functions

PySpark vs Pandas

- ✓ Same operations in both
- ✓ Performance comparison
- ✓ When to choose PySpark

PySpark using AI

- ✓ Writing PySpark codes using AI

Real Time Project using PySpark

Tools Covered: Google Collab, Databricks

Generative AI

Generative AI empowers data analysts to accelerate data analysis by automating data cleaning, generating insights, and enabling natural language interactions with data. It helps transform raw structured and unstructured data into meaningful summaries, visualizations, and predictive insights. With its ability to assist in code generation, pattern discovery, and decision support, GenAI enhances productivity and enables analysts to build intelligent, scalable, and insight-driven data workflows efficiently.

Landscape of Generative AI

- ✓ Introduction to Generative AI Landscape
- ✓ What is Generative AI?
- ✓ The Generative AI Tech Stack
- ✓ Paradigms of Artificial Intelligence: ML and DL
- ✓ Paradigms of Artificial Intelligence: Gen AI

Working with OpenAI APIs

- ✓ Introduction to OpenAI Models
- ✓ OpenAI API Instructions and Best Practices

Chat Completions API

- ✓ Single-turn Conversations with Chat Completions API
- ✓ Multi- turn Conversations with Chat Completions API
- ✓ Tracking API usage and costs

Prompt Engineering

- ✓ What and Why of Prompts
- ✓ Engineering a Good Prompt
- ✓ Applying Prompt Engineering
- ✓ Types of Prompting

Retrieval Augmented Generation (RAG)

- ✓ Basics of RAG
- ✓ Why RAG?
- ✓ What is RAG?
- ✓ Semantic Search System

Vector Stores

- ✓ Types of Vectors Stores
- ✓ Vector Database - Chroma
- ✓ Creating Chroma Collections
- ✓ Querying Chroma Collections
- ✓ Updating Chroma Collections

AI + SQL

- ✓ Converting natural language to SQL
- ✓ Query Optimization using AI

AI + Python

- ✓ Generate Pandas Code using AI
- ✓ Data Cleaning Automation

LLM Frameworks/LangChain

- ✓ Why frameworks are needed
- ✓ Introduction to LangChain
- ✓ Chains
- ✓ Agents

Tools Covered: Google Colab, Open AI API, Chroma,