

Data Science using Python

Build end-to-end data science capability with Python through a practical 5-day program covering data collection, pandas, analysis, visualization, pipelines, and reporting outputs.

Overview

This 5-day applied program builds practical Python data science capability across data preparation, pandas, analysis, visualization, data pipelines, and publishing analytical outputs as usable business reports and dashboards.

Introduction

This 5-day Data Science using Python program is designed for learners who want a fuller end-to-end analytics workflow rather than isolated analysis techniques. In addition to Python, pandas, data analysis, and visualization, participants will learn how to collect data from structured sources such as SQL databases, NoSQL platforms, and APIs, automate data movement into a central location, and then turn analytical output into publishable web reports and dashboard-style views. The program combines practical hands-on work with a business-focused mindset so participants can understand how Python supports real reporting, productivity, and decision-making environments.

Topics Covered

- Introduction to Python for Data Work
- Python Syntax, Variables, and Data Types
- Control Structures and Program Flow
- Python Data Structures for Analysis
- Functions, Modules, and Reusable Analysis Logic
- Data Collection Fundamentals
- Working with SQL Data Sources
- Working with NoSQL Data Sources
- Collecting Data from APIs
- Building a Simple Data Pipeline
- NumPy Foundations for Numerical Data
- Introduction to pandas and Why It Matters
- pandas Series Fundamentals
- DataFrame Creation and Structure
- Data Cleaning with pandas

- Selecting, Filtering, and Transforming DataFrames
- Aggregation, Grouping, and Reshaping
- Exploratory Data Analysis for Business Decisions
- Numerical Data Analysis
- Text Data Analysis and Sentiment Analysis
- Data Visualization with Matplotlib and Seaborn
- Turning Analysis into Productivity and Business Improvement
- Reporting Output Design
- Publishing Analytical Results to Web Reports
- Dashboard Fundamentals
- Building a Simple Python-Driven Reporting View

Prerequisites

There are no formal prerequisites required to attend this course, although basic computer literacy and familiarity with spreadsheet-based work will be helpful.

Audience:

Excel users, business analysts, aspiring data analysts, reporting professionals, developers moving into analytics, and teams building practical data workflows with Python.

Duration: 5 Days

Course Outline

Day 1 - Python Language Foundations for Data Analysis

1. Introduction to Python for Data Work
 - 1.1. Python in data analysis and reporting
 - 1.2. Python vs spreadsheet workflows
 - 1.3. Notebooks, scripts, and environments
 - 1.4. Python workflow for data professionals
 - 1.5. Setting up the analysis environment

2. Python Syntax, Variables, and Data Types
 - 2.1. Python syntax basics
 - 2.2. Variables and naming rules
 - 2.3. Numbers, strings, and booleans
 - 2.4. Type conversion
 - 2.5. Comments and readable code style

3. Control Structures and Program Flow
 - 3.1. if, elif, and else
 - 3.2. for loops and while loops
 - 3.3. Looping through collections
 - 3.4. break and continue
 - 3.5. Simple control flow patterns

4. Python Data Structures for Analysis
 - 4.1. Lists and tuples
 - 4.2. Dictionaries
 - 4.3. Sets and uniqueness checks
 - 4.4. Indexing and slicing
 - 4.5. Choosing the right structure

5. Functions, Modules, and Reusable Analysis Logic
 - 5.1. Defining and calling functions
 - 5.2. Parameters and return values
 - 5.3. Scope and variable visibility
 - 5.4. Importing modules and libraries
 - 5.5. Reusable analysis functions

6. Lab 1: Python Practice for Data Work
 - Variables, loops, and functions
 - Lists and dictionaries
 - Simple reusable scripts
 - Preparation for data collection work

Day 2 - Data Collection, Integration, and Pipeline Design

7. Data Collection Fundamentals
 - 7.1. Structured vs semi-structured data sources
 - 7.2. Files, databases, and APIs as data sources
 - 7.3. Batch vs scheduled data collection
 - 7.4. Centralized data collection concepts
 - 7.5. Designing a practical collection workflow

8. Working with SQL Data Sources
 - 8.1. SQL databases in analytics workflows
 - 8.2. Connecting Python to SQL data sources
 - 8.3. Reading query results into Python
 - 8.4. Basic filtering and extraction strategies
 - 8.5. Moving database records into analysis pipelines

9. Working with NoSQL Data Sources
 - 9.1. NoSQL concepts for analysts
 - 9.2. Document-style data and nested structures
 - 9.3. Reading NoSQL-style records into Python
 - 9.4. Flattening records for analysis
 - 9.5. Choosing SQL vs NoSQL sources

10. Collecting Data from APIs
 - 10.1. API basics for data professionals
 - 10.2. Requests, endpoints, and JSON responses
 - 10.3. Authentication basics and API keys
 - 10.4. Loading API responses into Python
 - 10.5. Handling pagination and repeated calls

11. Building a Simple Data Pipeline
 - 11.1. Extract, transform, and load concepts
 - 11.2. Bringing multiple sources into one location
 - 11.3. Automating collection with Python scripts
 - 11.4. Preparing pipeline output for pandas
 - 11.5. Monitoring and maintaining simple pipelines

12. Lab 2: Create a Practical Data Collection Pipeline
 - Read data from files and an API
 - Load data into a common structure
 - Standardize collected records
 - Produce a pipeline-ready dataset

Day 3 - Data Science Foundations with Python

13. NumPy Foundations for Numerical Data
 - 13.1. Why NumPy matters
 - 13.2. Creating arrays
 - 13.3. Vectorized operations
 - 13.4. Array indexing and slicing
 - 13.5. Basic numerical calculations

14. Introduction to pandas and Why It Matters
 - 14.1. What pandas is
 - 14.2. Why pandas beats manual spreadsheet work
 - 14.3. Series and DataFrame concepts
 - 14.4. Business use cases for pandas
 - 14.5. pandas for reporting productivity

15. pandas Series Fundamentals
 - 15.1. Creating a Series
 - 15.2. Series indexing and selection
 - 15.3. Filtering Series data
 - 15.4. Applying functions to a Series
 - 15.5. Missing values in a Series

16. DataFrame Creation and Structure
 - 16.1. Creating DataFrames
 - 16.2. Rows, columns, index, and data types
 - 16.3. Shape, columns, and metadata
 - 16.4. Reading CSV and Excel-style data
 - 16.5. Saving DataFrames

17. Data Cleaning with pandas
 - 17.1. Nulls, blanks, and inconsistent values
 - 17.2. Handling missing data
 - 17.3. Duplicates and formatting issues
 - 17.4. Renaming and standardizing columns
 - 17.5. Data type conversion

18. Lab 3: Data Science Preparation with pandas
 - Load collected data into pandas
 - Clean and standardize the dataset
 - Prepare analysis-ready structures
 - Create reusable preparation steps

Day 4 - Data Analytics, Exploration, and Insight Generation

19. Selecting, Filtering, and Transforming DataFrames
 - 19.1. Column and subset selection
 - 19.2. loc and iloc
 - 19.3. Conditional filtering
 - 19.4. Sorting and resetting indexes
 - 19.5. Calculated columns and transformations

20. Aggregation, Grouping, and Reshaping
 - 20.1. groupby summaries
 - 20.2. sum, mean, count, min, and max
 - 20.3. Value counts and frequencies
 - 20.4. Pivot-style analysis
 - 20.5. Merging and joining datasets

21. Exploratory Data Analysis for Business Decisions
 - 21.1. Patterns, distributions, and anomalies
 - 21.2. Trends, gaps, and outliers
 - 21.3. Descriptive statistics
 - 21.4. Summaries for deeper analysis
 - 21.5. Turning observations into questions

22. Numerical Data Analysis
 - 22.1. Totals, averages, and ratios
 - 22.2. Category and segment comparisons
 - 22.3. Top and low performers
 - 22.4. Practical correlation ideas
 - 22.5. Interpreting numeric findings

23. Text Data Analysis and Sentiment Analysis
 - 23.1. Text analysis in business
 - 23.2. Cleaning text data
 - 23.3. Word frequency and keyword patterns
 - 23.4. Basic sentiment analysis workflow
 - 23.5. Customer feedback insights

24. Lab 4: Applied Data Analytics
 - Analyze numeric and text data
 - Compare categories and segments
 - Extract sentiment and keyword patterns
 - Present analytical findings

Day 5 - Publishing Insights, Web Reports, and Dashboards

25. Data Visualization with Matplotlib and Seaborn
 - 25.1. Choosing the right chart
 - 25.2. Bar, line, scatter, and histogram charts
 - 25.3. Seaborn basics
 - 25.4. Trends, comparisons, and distributions
 - 25.5. Business presentation visuals

26. Turning Analysis into Productivity and Business Improvement
 - 26.1. Improving reporting speed and quality
 - 26.2. Finding inefficiencies and bottlenecks
 - 26.3. Supporting business decisions
 - 26.4. Repeatable analysis for productivity
 - 26.5. Data-driven confidence

- 27. Reporting Output Design
 - 27.1. From analysis to publishable output
 - 27.2. Choosing metrics for stakeholders
 - 27.3. Structuring business-friendly reports
 - 27.4. Combining text, tables, and charts
 - 27.5. Designing repeatable reporting outputs

- 28. Publishing Analytical Results to Web Reports
 - 28.1. Static report concepts
 - 28.2. Exporting analysis outputs for the web
 - 28.3. HTML-based reporting ideas
 - 28.4. Organizing sections and visuals clearly
 - 28.5. Sharing report output with business teams

- 29. Dashboard Fundamentals
 - 29.1. What makes a useful dashboard
 - 29.2. KPI and metric presentation
 - 29.3. Charts, filters, and drill-down thinking
 - 29.4. Operational vs management dashboards
 - 29.5. Dashboard usability for decision makers

- 30. Building a Simple Python-Driven Reporting View
 - 30.1. Preparing pandas outputs for presentation
 - 30.2. Generating chart-ready datasets
 - 30.3. Creating simple report pages or dashboard views
 - 30.4. Refreshing outputs from updated data
 - 30.5. Turning analysis into shareable artifacts

- 31. Final Capstone Project
 - Collect data from multiple sources
 - Prepare the data with Python and pandas
 - Perform business-focused analysis
 - Create charts and insights
 - Publish the results as a report or dashboard-style output