Data Analytics with Python

Curriculum designed by Bhupendra Pratap Singh

MODULE 1: Python Programming

Python is a high-level, general-purpose programming language that emphasizes readability and simplicity. It's open-source, which means it's free to use and has a large supportive community. Python requires less coding compared to other languages, making it efficient and time-saving. It's versatile, applicable in various domains, and offers high performance

Day 1: Introduction to Python Basics

- Overview of Python and career opportunities
- Overview of Anaconda and different IDEs
- Introduction to Jupyter Notebook
- Write your first program
- Python data types and variables
- Basic arithmetic operations

Day 2: Python Control Structures

- Conditional statements (if, elif, else)
- Loops (for, while)
- Break and continue statements

Day 3: Python Strings

- Strings operation
- Indexing and Slicing
- String methods: upper, lower, title, capitalize, isalpha, isalnum, replace, startswith, endswith, strip, split, join, find

Day 4: Python Data Structures (Part-1)

- Lists and Tuples: Understanding the difference
- Working with their methods

- List comprehensions
- Exercise: Practice problems on list and tuples

Day 5: Python Data Structures (Part-2)

- Sets and Dictionary
- Working with their methods
- Dictionary comprehensions
- Exercise: Practice problems on sets and dictionaries

Day 6: Python Functions

- What are functions?
- Advantages of using functions
- Writing functions in Python
- Default parameter and required parameter
- Return keyword
- Lambda functions
- Map, filter and reduce, zip functions
- Excercise: Practice problems on python functions

Day 7: Exception Handling and File Handling

- Understanding what is exception handling and why it is important?
- Try and except block
- Nested try and except block
- Requirement of File Handling
- Read, Write, Append and Create operations on files

Day 8: Python OOPs basics

OOPs is not required for data analysis job profiles until you use “objects” to design applications and programs. It can be benefecial when you need to write complex applications or when working in a large software company where production quality code is required. So we just want you to not leave the topic but understand the basics of OOP based programming approach. Topics to be covered are as follows:

- What is Object oriented programming?
- Advantages of using OOP?
- Where should I use OOP?

- Class and Objects
- Single, Multiple and Multilevel Inheritance
- Practice question on OOP programming

MODULE 2: MySQL Database

A database is an organized collection of structured information, typically stored electronically in a computer system. MySQL is a popular relational database management system (RDBMS) that uses SQL to manage data. It's known for its reliability, performance, versatility, ease of use, and cost-effectiveness. Compared to other databases, MySQL stands out for its scalability, security, and compatibility with various platforms.

Day 9: Introduction to SQL and MySQL

- Overview of SQL and relational databases
- Introduction to MySQL
- Basic SQL queries (SELECT, INSERT, UPDATE, DELETE)
- Connecting Python with MySQL database

Day 10: Advance SQL Queries

- Joins: Inner, Left and Right joins
- Where, Group by and Having clause
- Working with multiple tables
- Practice exercises on SQL queries

MODULE 3: EDA with Python Libraries

Exploratory Data Analysis (EDA) is a data analysis methodology that enhances understanding of a dataset. It helps reveal inherent patterns, identify significant variables, detect outliers and anomalies, and assess underlying assumptions. EDA is crucial as it ensures the results produced are valid and applicable to desired business outcomes and goals.

Day 11: Introduction to NumPy

- Introduction to NumPy arrays

- Basic array operations
- Array indexing and slicing
- Boolean Indexing with arrays
- Array generation with random function
- NumPy functions for statistical analysis

Day 12: Exploratory Data Analysis with Pandas (Part 1)

- Introduction to Pandas Series and DataFrame
- Loading data into DataFrame
- Data exploration and basic operations
- Introduction to data cleaning and preprocessing

Day 13: Exploratory Data Analysis with Pandas (Part 2)

- Data manipulation with Pandas DataFrame
- Handling missing data
- Data aggregation and grouping
- Merging two dataframes
- Exporting DataFrames as Excel or CSV files
- Exercise: Data manipulation exercises using Pandas

Day 14: Data Visualization with Matplotlib (Part 1)

- Introduction to data visualization
- Basic plotting with Matplotlib
- Line plots, scatter plots, and bar plots
- Customizing plot appearance

Day 15: Data Visualization with Matplotlib (Part 2)

- Advanced plotting techniques with Matplotlib
- Histograms, box plots, and violin plots
- Plotting multiple subplots
- Exercise: Visualizations with Matplotlib

Day 16: Introduction to Seaborn

- Overview of Seaborn library

- Seaborn vs. Matplotlib
- Basic plotting with Seaborn
- Customizing Seaborn plots

Day 17: Advanced Data Visualization with Seaborn

- Seaborn for statistical visualization
- Pair plots, heatmaps, and cluster maps
- Seaborn themes and styles
- Exercise: Creating advanced visualizations with Seaborn

Day 18-19: Final Project

- Students work on a real-world data analytics project using Python, NumPy, Pandas, Matplotlib, and Seaborn.
- Project includes data cleaning, data analysis, and data visualization tasks.
- Students present their projects on the last day of the course.

Day 20: Project Presentations and Wrap-Up

- Students present their final projects to the class.
- Wrap-up session with review of key concepts and Q&A.
- Course evaluation and feedback.

This syllabus provides a structured and focused approach to learning data analytics with Python, MySQL, NumPy, Pandas, Matplotlib, and Seaborn, culminating in a practical final project that allows students to apply their skills in a real-world context.