Applied Python Structures for Data-Driven Problem Solving

Instructor: *Dr. M. Nadeem Majeed*

Total Marks: 100
15 Tasks × 6 Marks = 90 + 1 Integrated Task × 10 Marks = 100

Objective:
Demonstrate the ability to choose and combine appropriate data structures tuples, lists, dictionaries, sets, strings, and functions to design efficient and well-organized Python applications. These structures form the foundation of data analysis and real-world problem-solving, enabling effective data storage, manipulation, and summarization. Through this assignment, you will learn how to structure data like a true data scientist logically, cleanly, and efficiently.

"Think like a problem solver, not just a coder use the right structure for the right task!"

Problem 1 – Delivery Path Analyzer (6 marks)

You are developing a tracker for a delivery service. Store multiple delivery points with their time and distance. Write a program to find the starting location, ending location, and total distance covered.

Problem 2 – Shopping Cart Management (6 marks)

Design a small shopping cart system that allows customers to add, remove, and update items. At the end, display the total price and the most expensive item purchased.

Problem 3 – Student Performance Report (6 marks)

Build a grade management program that stores students' marks in multiple subjects. Display each student's average marks, and find the top 3 performers.

Problem 4 – Email Data Processor (6 marks)

Create an email management system that processes a list of email addresses. It should validate formats, extract domains, and count how many users belong to each domain.

Problem 5 – Friendship Network Analysis (6 marks)

You are designing a social platform feature to suggest friends. Find mutual friends, unique connections, and friend suggestions between two or more users.

Problem 6 – Restaurant Billing System (6 marks)

Write a program to calculate a restaurant bill that includes subtotal, tax, tip, and final total. The program should also handle different tip percentages based on service quality.

Problem 7 – Transport Schedule Tracker (6 marks)

Build a public transport tracking tool where each stop has a name, time, and distance from the start. Find the fastest segment, total route time, and total number of stops.

Problem 8 – Library Catalog System (6 marks)

Create a library management system that stores book details such as title, author, and category. Find how many books belong to each author and which author has the most books.

Problem 9 – Sports Tournament Statistics (6 marks)

Develop a program to manage a sports tournament. Track teams, matches, and scores, and generate the winning team, average score, and unique participants.

Problem 10 – Weather Data Analyzer (6 marks)

Write a program to manage 30 days of weather data. Find the average temperature, hottest day, and most frequent weather condition.

Problem 11 – Employee Attendance System (6 marks)

Build a system to track employee attendance for one month. Identify employees with perfect attendance, most absences, and attendance percentage for each employee.

Problem 12 – Fitness Tracker (6 marks)

Design a program to track multiple users' fitness activities. Calculate total calories burned, average workout time, and display unique activity types.

Problem 13 – Online Quiz Evaluator (6 marks)

Create a quiz result management system that stores students' scores and time taken. Find average score, highest scorer, and students who passed a minimum score threshold.

Problem 14 – Hotel Booking Management (6 marks)

Write a program to manage hotel room bookings. It should check room availability, calculate total stay cost, and list occupied vs available rooms.

Problem 15 – Expense & Budget Tracker (6 marks)

Create a financial tracker for daily expenses. Categorize expenses, calculate total per category, check if spending exceeds budget, and find largest expense.

Problem 16 – Integrated Real-World System: Smart Business Manager (10 marks)

Develop a 'Smart Business Manager' that combines all data-handling skills.

Your program should:

- Record business transactions with details like date, type, amount, and category
- Maintain all transactions for reporting

- Summarize totals by category
- Identify the highest expense
- Generate a formatted financial summary report
- Use modular functions such as add_transaction(), total_summary(), highest_expense(), etc.

“There are no secrets to success. It is the result of preparation, hard work, and learning from failure.”

Colin Powell