Lab01: Travel Analytics Lab: Modeling Trips and Places with Python

Course: Programming for Data Science
Professor: Dr. Carlos J. Costa

January 29, 2026

In this lab, you will assume the role of a data analyst at a travel analytics company interested in understanding travel patterns. Travelers move from one city to another, and each journey has associated information such as cost, duration, and mode of transport. Your task is to model this system using object-oriented programming in Python and then analyze the resulting data using pandas and NumPy.

The lab requires you to define two main classes. The first class, `Place`, represents a city or location, and should store attributes such as the name of the city and the country. The second class, `Trip`, represents a journey connecting two `Place` objects. Each `Trip` object should include the origin and destination (both `Place` objects), as well as attributes such as transport mode, total cost, and duration in days. You should also implement a method to compute the cost per day for the trip.

Once your classes are defined, you should create multiple `Place` objects representing different cities, and several `Trip` objects connecting these places. All trips should be stored in a list. You will then convert this list into a tabular structure suitable for analysis with pandas. Using this structure, compute metrics such as the total cost of all trips, the average cost per day, the most expensive trip, and the standard deviation of trip costs. You may also perform optional analyses, for example, comparing costs by transport mode.

At the end of the lab, you are invited to extend the design by creating additional classes beyond `Place` and `Trip`. The purpose and nature of these extra classes are entirely up to you. For instance, you could model travelers, statistics, or transport modes, or any other concept you think would improve your system. The goal is to encourage creativity and allow you to explore object-oriented design while maintaining a connection to the business context.

This lab emphasizes clear object-oriented design, the ability to flatten object data into a tabular format, and performing meaningful analysis with pandas and NumPy. Extra methods such as `__repr__` or summary functions are optional; the lab should be fully correct and functional without them.

Submission Requirements

Students are expected to submit a file containing their code. This can be either a Jupyter Notebook (`.ipynb`) or a Python script (`.py`). In addition to submitting the code, students should be prepared to answer questions related to their implementation, the structure of their classes, and the analyses they performed. This integrates naturally with your prose style, keeps it professional, and clearly states both the file type