Universidad Complutense de Madrid x NTIC
Manuel Sanchez Mc Sweeney
Advanced Python

Project Report: Advanced Python Practice for Data Science

Introduction

This project was undertaken as part of the "Advanced Python" module for the Master in Data Science, Big Data & Business Analytics. The primary objective was to apply advanced Python programming concepts to a practical data science challenge. This involved programmatically generating a structured dataset, processing and manipulating it using the **Pandas** library, and creating a series of complex data visualizations with **Matplotlib**.

Project Stages

The project was divided into three main parts: dataset generation, data manipulation, and visualization.

1. Random Hostname Dataset Generation

The first step was to create a function, **`set_hostnames`**, to generate a specified number of random server hostnames. Each hostname was an 8-character string, with each character or group of characters representing specific information according to a set of rules and proportions:

- **Character 1 (Operating System):** Indicated the OS with a specific probability:
    - **`L`** for Linux (40%)
    - **`S`** for Solaris (30%)
    - **`A`** for AIX (20%)
    - **`H`** for HP-UX (10%)
- **Character 2 (Environment):** Represented the server's environment with a defined distribution:
    - **`D`** for Development (10%)
    - **`I`** for Integration (10%)
    - **`T`** for Testing (25%)
    - **`S`** for Staging (25%)
    - **`P`** for Production (30%)
- **Characters 3-5 (Country):** A three-letter code for the country:
    - **`NOR`**: Norway (6%)
    - **`FRA`**: France (9%)
    - **`ITA`**: Italy (16%)
    - **`ESP`**: Spain (16%)
    - **`DEU`**: Germany (23%)
    - **`IRL`**: Ireland (30%)

- **Characters 6-8 (Node Number):** A three-digit incremental number (from 001 to 999) for each unique combination of OS, environment, and country.

2. Data Processing and DataFrame Creation

Once the hostnames were generated, the next stage involved processing them. Helper functions (

`get_os`, `get_enviroment`, `get_country`) were created to parse each hostname string and translate the codes into human-readable text (e.g., 'L' becomes 'Linux').

These functions were used within a main function,

`set_dataframe`, which orchestrated the creation of a **Pandas DataFrame**. This function generated a list of dictionaries, where each dictionary represented a server with keys such as

`hostname`, `os`, `environment`, `country`, and **`node`**. For this project, a DataFrame with

1,500 records was generated and subsequently exported to a **`hosts.csv`** file.

3. Data Visualization with Matplotlib

The final and most extensive part of the project was data visualization. The goal was to analyze the generated dataset and present the findings through a series of plots:

1. **Grouped Bar Chart:** A bar chart was created to show the distribution of server environments (**`Development`, `Production`**, etc.) for each country, using the **`unstack()`** function to format the data correctly.
2. **2x2 Subplot Figure:** A more complex visualization was created using a 2x2 grid that included four different graphs:
    - **Top-Left:** A horizontal bar chart displaying the *Type of OS grouped by country*.
    - **Top-Right:** A pie chart showing the percentage and count for *Total Operating Systems* in the dataset.
    - **Bottom-Left:** A horizontal bar chart of *Total hosts by country*, with data labels on each bar.
    - **Bottom-Right:** A grouped bar chart showing *Hosts by country grouped by environment*.

Conclusion

This project successfully demonstrated the ability to manage a complete data workflow in Python. It involved procedural data generation based on complex rules, data wrangling and structuring with Pandas, and the creation of insightful, publication-quality visualizations with Matplotlib. It served as a comprehensive exercise in the practical application of Python for data science.