

Team Exercise — Dataclasses in Python

In this exercise, your team will design and implement a Python program using dataclasses. The goal is to practice advanced dataclass features (`default_factory`, `__post_init__`, `ordering`, `frozen`, etc.) while collaborating on a small but realistic data model.

Objective

Create a Python module that models a small real-world system using dataclasses. You can choose one of the following domains (or come up with your own idea):

- A library of books or films
- A collection of athletes and their results
- A product inventory system
- A travel planner (flights, hotels, destinations)

Requirements

Your dataclass-based system should include:

1. At least one main dataclass (e.g., `Book`, `Movie`, `Runner`, `Product`)
2. At least one helper/manager class (e.g., `Library`, `RunLog`, `Inventory`)
3. Usage of dataclass features such as:
 - `field(default_factory=list)`
 - `__post_init__` for validation or computed fields
 - `order=True` or `frozen=True` if relevant
4. At least one method that performs a calculation (e.g., total distance, average rating, inventory value)
5. Export to a readable text file or printout (e.g., summary report)

Collaboration Guidelines

Work in your breakout room as one team. Assign quick roles:

- Driver — writes code
- Navigator — checks logic and quality
- Timekeeper — tracks time
- Presenter — shows results

You have 15 minutes to complete the task. At the end, present your solution briefly (1–2 minutes) and share the file.

Tips

- Keep your code readable and structured.
- Use type hints and data validation.
- Include comments explaining your choices.
- Focus on clarity and teamwork rather than complexity.

Deliverables

1. A working Python file (or Jupyter notebook) implementing your dataclasses.
2. A short group demo and explanation of how your solution works.
3. (Optional) A text or CSV export showing results.

Time limit: 15 minutes

Good luck and have fun coding together!