### **Employee Management System**

#### Objective:

Develop a command-line Employee Management System using Python that allows a company to manage information about their employees and departments using OOP and data structures.

#### Requirements:

* **Employee Class:**
  + Create an **Employee** class with attributes for name, ID, title, and department.
  + Include a method to display employee details.
  + Implement a string representation method that returns the employee's name and ID.
* **Department Class:**
  + Create a **Department** class with attributes for department name and a list of employees.
  + Include methods to add an employee, remove an employee, and list all employees in the department.
* **Data Structure for Company:**
  + Use a dictionary to represent the company, with department names as keys and **Department** objects as values.
  + Implement functions to add a department, remove a department, and display all departments.
* **User Interaction:**
  + Write a function that prints a menu for the user to interact with the Employee Management System (e.g., Add Employee, Remove Employee, Display Department, etc.).
  + The user should be able to add employees to departments, remove employees, and view department details.
* **Data Persistence (Optional – understanding the knowledge of streams):**
  + Save the company data to a file and load it back into the system on startup. (This can be plain text, JSON, or any other format you deem appropriate.)

#### Instructions:

* Structure your code with appropriate classes and functions.
* Ensure your classes use proper encapsulation for attributes.
* Include error handling for cases such as adding an employee to a non-existent department.
* Write clear instructions for the user on how to use the command-line interface.
* Comment your code to explain your logic and design decisions.
* (Optional) Include unit tests to validate your methods.

#### Deliverables:

* Python script(s) (.py files) that run the Employee Management System.
* (Optional) A text or JSON file containing the company data.
* (Optional) A test script with unit tests.

#### Evaluation Criteria:

* Correct implementation of OOP principles.
* Efficient use of data structures.
* Code readability and comments.
* Error handling and edge case consideration.
* Unit tests (Optional) and testing strategy.