University Course Management System

Objective

Design a simple Python program to simulate a university's course enrollment system using the four principles of OOP:

1. **Encapsulation**
2. **Abstraction**
3. **Inheritance**
4. **Polymorphism**

Problem Statement

A university has multiple departments offering courses. Students can enroll in courses. Professors can teach courses. Your task is to build a simplified Course Management System that supports:

- Creating departments
- Adding courses to departments
- Enrolling students in courses
- Assigning professors to courses
- Displaying course details including enrolled students and assigned professor

---

Specifications

1. Encapsulation

- Use private attributes for `Student`, `Professor`, and `Course` classes.
- Provide getter/setter methods for controlled access.

2. Abstraction

- Use abstract base class `Person` with abstract method `display_details()`, implemented by both `Student` and `Professor`.

3. Inheritance

- `Student` and `Professor` should inherit from `Person`.

4. Polymorphism

- `display_details()` should show relevant details based on whether the object is a student or a professor.

Classes to Implement

`Person` (Abstract Class)

- Attributes: `_name`, `_email`
- Method: `display_details()` → abstract method

`Student(Person)`

- Additional Attribute: `_roll_no`
- `display_details()` prints name, email, roll no

`Professor(Person)`

- Additional Attribute: `_employee_id`
- `display_details()` prints name, email, employee id

`Course`

- Attributes: `_course_code`, `_title`, `_students` (list), `_professor` (optional)
- Methods:
  - `add_student(student: Student)`
  - `assign_professor(prof: Professor)`
  - `get_course_summary()`

Department

- Attributes: `_name`, `_courses` (list)
- Methods:
  - `add_course(course: Course)`
  - `list_courses()`

Sample Input / Actions

```
python
CopyEdit
# Create a Department
cs_dept = Department("Computer Science")

# Create Students and Professor
s1 = Student("Alice", "alice@example.com", "CS101")
s2 = Student("Bob", "bob@example.com", "CS102")
p1 = Professor("Dr. Smith", "smith@example.com", "EMP987")

# Create Course and add to department
course = Course("CS200", "Data Structures")
cs_dept.add_course(course)

# Assign professor and enroll students
course.assign_professor(p1)
course.add_student(s1)
course.add_student(s2)

# Display course summary
course.get_course_summary()
```

Sample Output

```
less
CopyEdit
Course Code: CS200
Title: Data Structures
Professor: Dr. Smith (smith@example.com)
Enrolled Students:
 - Alice (CS101)
 - Bob (CS102)
```

Expected Outcome

By completing this assignment, students will:

- Understand how to apply **encapsulation** using private variables
- Create and use **abstract classes**
- Apply **inheritance** to reuse code
- Use **polymorphism** to dynamically execute appropriate methods