```
# Student class
class Student:
    def __init__(self, name):
        self.name = name

# Course class holds students
class Course:
    def __init__(self, course_name):
        self.course_name = course_name
        self.students = []

    def add_student(self, student):
        self.students.append(student)

    def __iter__(self):
        return iter(self.students)

# University class with iterator over all students in all courses
class University:
    def __init__(self):
        self.courses = []

    def add_course(self, course):
        self.courses.append(course)

    def all_students(self):
        for course in self.courses:
            for student in course:
                yield student

# Example usage
c1 = Course("CS101")
c1.add_student(Student("Alice"))
c1.add_student(Student("Bob"))

c2 = Course("Math201")
c2.add_student(Student("Charlie"))

uni = University()
uni.add_course(c1)
uni.add_course(c2)

print("All students:")
for student in uni.all_students():
    print(student.name)
```

```
All students:
Alice
Bob
Charlie
```

```
from abc import ABC, abstractmethod

# Component
class EmployeeComponent(ABC):
    @abstractmethod
    def show_salary(self):
        pass

    @abstractmethod
    def get_salary(self):
        pass

# Leaf
class Employee(EmployeeComponent):
    def __init__(self, name, salary):
        self.name = name
        self.salary = salary

    def show_salary(self):
        print(f"Name: {self.name} and Salary: {self.salary}")

    def get_salary(self):
        return self.salary
```

```
# Composite
class Department(EmployeeComponent):
    def __init__(self, name):
        self.name = name
        self.subordinates = []

    def add(self, employee):
        self.subordinates.append(employee)

    def show_salary(self):
        print(f"\n{self.name} Employees")
        for emp in self.subordinates:
            emp.show_salary()
        print(f"Total salary of {self.name}: {self.get_salary()}")

    def get_salary(self):
        return sum(emp.get_salary() for emp in self.subordinates)

# Example: Building the structure

# HR Department
hr = Department("HR Department")
hr.add(Employee("Priyanka", 25000))
hr.add(Employee("Sambit", 28000))

# IT Department
it = Department("IT Department")
it.add(Employee("Rohit", 15000))
it.add(Employee("Anurag", 22000))

# Company
company = Department("Company")
company.add(hr)
company.add(it)

# Output the salary structure
company.show_salary()
```

```
Company Employees

HR Department Employees
Name: Priyanka and Salary: 25000
Name: Sambit and Salary: 28000
Total salary of HR Department: 53000

IT Department Employees
Name: Rohit and Salary: 15000
Name: Anurag and Salary: 22000
Total salary of IT Department: 37000
Total salary of Company: 90000
```