



## SUPERIOR UNIVERSITY

NAME	Ali Mustafa Shah
REGISTRATION NUMBER	SU92-BSDSM-S24-005
SECTION	2A
SEMESTER	2 <sup>ND</sup>
ASSINGMENT	8
SUBJECT	OOP (LAB)
SUBMITTED TO	SIR RASIKH

### LAB-8 TASK

Create a Python program to manage employee details:

1. Implement two parent classes:
  - a. Person with attributes name and age
  - b. Employee with attributes employee\_id and position.
  - Include methods display\_info in both classes to display the respective details.
2. Create a child class Staff inheriting from both Person and Employee. The Staff class should have an additional attribute department and a method additional\_info to display the department.
3. 3. Implement file handling to read and write employee information to a text file. Include functions to read employee information from a file, add new employee information, and save employee information to the file. (Bonus marks if you create CSV instead of a simple TXT file)

# SOLUTION

```
class Person:
    def __init__(self, name, age):
        self.name = name
        self.age = age

    def display_info(self):
        print(f"Name : {self.name}")
        print(f"Age : {self.age}")

class Employee:
    def __init__(self, employee_id, position):
        self.employee_id = employee_id
        self.position = position

    def display_info(self):
        print(f"Employee ID: {self.employee_id}")
        print(f"Position: {self.position}")

class Staff(Person, Employee):
    def __init__(self, name, age, employee_id, position, department):
        Person.__init__(self, name, age)
        Employee.__init__(self, employee_id, position)
        self.department = department

    def additional_info(self):
        print(f"Department: {self.department}")

    def display_info(self):
        Person.display_info(self)
        Employee.display_info(self)
        self.additional_info()

def read_employees_from_file(file_name):
    employees = []
```

```

    try:
        with open(file_name, 'r') as file:
            for line in file:
                name, age, employee_id, position, department =
line.strip().split(',')
                employees.append(Staff(name, int(age), employee_id, position,
department))

    except FileNotFoundError:
        print(f"File '{file_name}' not found. Starting with an empty list.")

    return employees

def save_employees_to_file(file_name, employees):
    with open(file_name, 'w') as file:
        for employee in employees:

file.write(f"{employee.name},{employee.age},{employee.employee_id},{employee.posi
tion},{employee.department}\n")

def add_new_employee(employees):
    name = input("Enter your name : ")
    age = int(input("Enter your age : "))
    employee_id = input("Enter the employee ID : ")
    position = input("Enter your position : ")
    department = input("Enter the department : ")

    employees.append(Staff(name, age, employee_id, position, department))

def main():
    file_name = "employees.txt"
    employees = read_employees_from_file(file_name)

    while True:
        print("\nEmployee Management System")
        print("1.Display all employees")
        print("2.Add new employee data")
        print("3.Save and Exit")
        choice = input("Enter your choicxe : ")

```

```
    if choice == '1':
        if employees:
            for employee in employees:
                employee.display_info()
                print("-" * 20)
            else:
                print("No employee to display")

    elif choice == '2':
        add_new_employee(employees)

    elif choice == '3':
        save_employees_to_file(file_name, employees)
        print("Employee data is saved")

        break

    else:
        print("Invalid choice. Please try again.")

if __name__ == "__main__":
    main()
```

