

```
# Employee Management System
class Node:
    def __init__(self, name, department, designation, salary, phone_number, gender, age):
        self.name = name
        self.department = department
        self.designation = designation
        self.salary = salary
        self.phone_number = phone_number
        self.gender = gender
        self.age = age
        self.prev = None
        self.next = None

class EmployeeList:
    def __init__(self):
        self.head = None
        self.tail = None

    def insert_employee(self, name, department, designation, salary, phone_number, gender, age):
        new_node = Node(name, department, designation, salary, phone_number, gender, age)
        if self.head is None:
            self.head = new_node
            self.tail = new_node
        else:
            self.tail.next = new_node
            new_node.prev = self.tail
            self.tail = new_node

    def delete_employee(self, name):
        ptr = self.head
        while ptr is not None:
            if ptr.name == name:
                if ptr.prev is not None:
                    ptr.prev.next = ptr.next
                else:
                    self.head = ptr.next
                if ptr.next is not None:
                    ptr.next.prev = ptr.prev
                break
            ptr = ptr.next

    def show_employee(self):
        ptr = self.head
        while ptr is not None:
            print("Name: ", ptr.name)
            print("Department: ", ptr.department)
            print("Designation: ", ptr.designation)
            print("Salary: ", ptr.salary)
            print("Phone number: ", ptr.phone_number)
            print("Gender: ", ptr.gender)
            print("Age: ", ptr.age)
            ptr = ptr.next

    def search_employee(self, name):
        ptr = self.head
        while ptr is not None:
            if ptr.name == name:
                return ptr
            ptr = ptr.next
        return None

    def update_employee(self, name, department, designation, salary, phone_number, gender, age):
        employee = self.search_employee(name)
        if employee is not None:
            employee.department = department
            employee.designation = designation
            employee.salary = salary
            employee.phone_number = phone_number
            employee.gender = gender
            employee.age = age

    def count_employees(self):
        count = 0
        . . .
```

```

ptr = self.head
while ptr is not None:
    count += 1
    ptr = ptr.next
return count
#Menu-driven Interface
def main():
    employee_list = EmployeeList()

    print(" Welcome to TKMCE ")
    print("1. Insert employee data")
    print("2. Delete employee data")
    print("3. Show employee data")
    print("4. Search employee data")
    print("5. Update employee data")
    print("6. Count the number of employees")
    print("7. Quit")

    while True:
        choice = int(input("Enter your choice: "))
        if choice == 1:
            name = input("Enter employee name: ")
            department = input("Enter employee department: ")
            designation = input("Enter employee designation: ")
            salary = float(input("Enter employee salary: "))
            phone_number = input("Enter employee phone number: ")
            gender = input("Enter employee gender: ")
            age = int(input("Enter employee age: "))
            employee_list.insert_employee(name, department, designation, salary, phone_number, gender, age)
        elif choice == 2:
            name = input("Enter employee name: ")
            employee_list.delete_employee(name)
        elif choice == 3:
            employee_list.show_employee()
        elif choice == 4:
            name = input("Enter employee name: ")
            employee = employee_list.search_employee(name)
            if employee is not None:
                print("Name: ", employee.name)
                print("Department: ", employee.department)
                print("Designation: ", employee.designation)
                print("Salary: ", employee.salary)
                print("Phone number: ", employee.phone_number)
                print("Gender: ", employee.gender)
                print("Age: ", employee.age)
            else:
                print("Employee not found")
        elif choice == 5:
            name = input("Enter employee name: ")
            department = input("Enter employee department: ")
            designation = input("Enter employee designation: ")
            salary = float(input("Enter employee salary: "))
            phone_number = input("Enter employee phone number: ")
            gender = input("Enter employee gender: ")
            age = int(input("Enter employee age: "))
            employee_list.update_employee(name, department, designation, salary, phone_number, gender, age)
        elif choice == 6:
            count = employee_list.count_employees()
            print(f"Number of employees: {count}")
        elif choice == 7:
            break
        else:
            print("Invalid choice")

main()

```

>Welcome to TKMCE
1. Insert employee data
2. Delete employee data
3. Show employee data

- 4. Search employee data
- 5. Update employee data
- 6. Count the number of employees
- 7. Quit

Enter your choice: 1

Enter employee name: V R MANASI MANASAN

Enter employee department: ARTIFICIAL INTELLIGENCE

Enter employee designation: STUDENT

Enter employee salary: 123651

Enter employee phone number: 3210654798

Enter employee gender: FEMALE

Enter employee age: 23

Enter your choice: 3

Name: V R MANASI MANASAN

Department: ARTIFICIAL INTELLIGENCE

Designation: STUDENT

Salary: 123651.0

Phone number: 3210654798

Gender: FEMALE

Age: 23

Enter your choice: 4

Enter employee name: V R MANASI MANASAN

Name: V R MANASI MANASAN

Department: ARTIFICIAL INTELLIGENCE

Designation: STUDENT

Salary: 123651.0

Phone number: 3210654798

Gender: FEMALE

Age: 23

Enter your choice: 4

Enter employee name: ARYA

Employee not found

Enter your choice: 6

Number of employees: 1

Enter your choice: 5

Enter employee name: SAM

Enter employee department: AE

Enter employee designation: PROFESSOR

Enter employee salary: 147852369

Enter employee phone number: 65417896

Enter employee gender: MALE

Enter employee age: 29

Enter your choice: 2

Enter employee name: SAM

Enter your choice: 6

Number of employees: 1

Enter your choice: 7

Colab paid products - Cancel contracts here

✓ 4m 12s completed at 1:48 PM

