MIT Art Design and Technology University

MIT School of Computing, Pune

Department of Computer Science and Engineering

Second Year B. Tech

Academic Year 2022-2023. (SEM-II)

Subject: Advance Data Structures Laboratory

Assignment 10

Assignment Title: Company maintains employee information such as employee ID, name, designation and salary. Allow users to add, delete information about employees. Display information of a particular employee. If an employee does not exist, an appropriate message is displayed. If it is, then the system displays the employee details. Use a sequential file to maintain the data.

Aim: Implement sequential file for employee database.

Prerequisite:

1. Basic knowledge of Python programming language.
2. Understanding of file handling in Python.
3. Familiarity with data structures like lists and dictionaries.

Objectives:

Implement a Program in python for the following operations:

1. To create a Python program that can maintain employee information.
2. To allow users to add, delete, and display employee information.
3. To use a sequential file to store employee data.

Outcomes:

Upon Completion of the assignment the students will be able to

1. Ability to create a Python program that can handle employee data.
2. Familiarity with file handling in Python.
3. Understanding of data structures used in the program.

Theory:

In this experiment, we will create a Python program that can maintain employee information using a sequential file. A sequential file is a type of file that stores data in a particular order. In our case, we will store employee data in the order of employee ID, name, designation, and salary.

To create this program, we will use a dictionary to store employee information. The keys of the dictionary will be the employee IDs, and the values will be another dictionary that contains the employee's name, designation, and salary. We will use the pickle module to store and retrieve the data from the sequential file.

To add an employee, we will ask the user to input the employee's information, which we will add to the dictionary. We will then update the sequential file with the new data. To delete an employee, we will ask the user for the employee ID and remove that entry from the dictionary. We will then update the sequential file to remove the employee's data. To display information about a particular employee, we will ask the user for the employee ID and retrieve that employee's data from the dictionary. If the employee does not exist, we will display an appropriate message.

Sequential file organization is a method of organizing data in which records are stored in a sequential order. In a sequential file, data is stored in the order in which it is added to the file. Each record in the file contains a key field that uniquely identifies the record, and the records are arranged in ascending or descending order based on this key field.

Sequential files are commonly used for storing large amounts of data that must be accessed in a specific order, such as payroll records, sales transactions, and inventory data. They are typically used for batch processing, where data is processed in large batches rather than on a record-by-record basis.

One advantage of using sequential file organization is that it allows for efficient retrieval of data in a specific order. Because the records are stored in order, it is possible to quickly locate a particular record by searching through the file in a linear fashion.

However, one major disadvantage of using sequential file organization is that it can be slow to access individual records in a file. Because the records are stored in a specific order, it can take a long time to locate and retrieve a record that is not near the beginning or end of the file.

Overall, sequential file organization is a useful method for organizing data in situations where data must be accessed in a specific order, but it may not be the best choice for situations where individual records must be accessed quickly and efficiently.

Conclusion:

We have created sequential file for employee database.

Frequently Asked Questions:

1. What is a sequential file?
2. What data structure will we use to store employee information?
3. How will we add an employee to the dictionary?
4. How will we delete an employee from the dictionary?
5. How will we display information about a particular employee?

MIT SCHOOL OF ENGINEERING
Rajbaug, Loni-Kalbhor, Pune

MIT-ADT UNIVERSITY

Name : Sourav Shailesh Toshniwal
Class : SY-CSE - 1
Roll no. : 2213047
Subject : ADSL

Assignment - 10.

1. what is sequential file ?

Ans A sequential file is a type of file organization where records or data are stored in linear or sequential manner, one after the other, in a continuous manner. Each record in a sequential file is located based on its relative position to the beginning of the file.

2. What data structure will we use to store employee information?

Ans There are several data structures that could be used to store employee information, depending on requirements. Here are a few common options:
1. Array : It can store address, name, phone number, etc.
2. Linked List. Each node in linked list can represent an employee record.
3. Hash table : Employee records can be stored with employee ID as a key.
4. Tree : Used to store info if the records need to be sorted or searched on a specific field.

3. How will we add an employee to dictionary ?

Ans
```
struct Employee {
    string name;
    string department;
    double salary;
};
Employee new_employee = {"X42", "ABC", 50000};
map <string, Employee> employee_map;
employee_map.insert(make_pair("1234", new_employee));
```

4. How will we delete an employee from dictionary?

Ans
```
struct Employee {
    string name;
    string department;
    double salary;
};
map <string, Employee> employee_map;
employee_map.erase("1234");
```

5. How will we display information about a particular employ

Ans
```
string employee_id = "1234";
auto employee_iter = employee_map.find(employee_id);
if (employee_iter != employee_map.end()) {
    Employee employee = employee_iter -> second;
    cout << "Name: " << employee.name << endl;
    cout << "Department: " << employee.department << endl;
    cout << "Salary: " << employee.salary << endl;
} else {
    cout << "Employee not found." << endl;
}
```

Code:

```
import os

filename = "employee.txt"
fields = ["Employee ID", "Name", "Designation", "Salary"]

def add_employee():
    print("Enter employee data: ")
    data = []
    for field in fields:
        value = input(f"{field}: ")
        data.append(value)

    with open(filename, "a") as file:
        file.write(",".join(data) + "\n")

    print("Employee added successfully.")

def delete_employee():
    employee_id = input("Enter employee ID: ")

    with open(filename, "r+") as file:
        contents = file.readlines()

        file.seek(0)
        file.truncate()

        deleted = False
        for line in contents:
            if not line.startswith(employee_id + ","):
                file.write(line)
            else:
                deleted = True

        if deleted:
            print("Employee deleted successfully.")
        else:
            print("Employee not found.")

def view_employee():
    employee_id = input("Enter employee ID: ")

    with open(filename, "r") as file:
        for line in file:
            if line.startswith(employee_id + ","):
                data = line.strip().split(",")
                for i in range(len(fields)):
                    print(f"{fields[i]}: {data[i]}")
                break
        else:
            print("Employee not found.")

while True:
    print("Menu:")
    print("1. Add employee")
    print("2. Delete employee")
    print("3. View employee data")
    print("4. Exit")

    choice = input("Enter your choice: ")

    if choice == "1":
```

```
        add_employee()
    elif choice == "2":
        delete_employee()
    elif choice == "3":
        view_employee()
    elif choice == "4":
        break
    else:
        print("Invalid choice. Please try again.")

print("Exiting...")
```

Output:

```
PS C:\SOURAV\CODE\C++ language codes\ADS assignment> python -u "c:\SOURAV\CODE\C++ language codes\ADS assignment\Assignment10.py"
Menu:
1. Add employee
2. Delete employee
3. View employee data
4. Exit
Enter your choice: 1
Enter employee data:
Employee ID: 123
Name: Sourav
Designation: CEO
Salary: 122134124
Employee added successfully.
Menu:
1. Add employee
2. Delete employee
3. View employee data
4. Exit
Enter your choice: 4
Exiting...
PS C:\SOURAV\CODE\C++ language codes\ADS assignment>
```