

```
1  #include <iostream>
2  #include <fstream>
3  #include <map>
4  using namespace std;
5
6  struct Employee {
7      int id;
8      char name[30];
9      char designation[30];
10     float salary;
11 };
12
13 // Global index (in-memory)
14 map<int, streampos> indexMap;
15
16 void buildIndex() {
17     indexMap.clear();
18     ifstream file("employee.dat", ios::binary);
19     Employee emp;
20     streampos pos = file.tellg();
21     while (file.read((char*)&emp, sizeof(emp))) {
22         indexMap[emp.id] = pos;
23         pos = file.tellg();
24     }
25     file.close();
26 }
27
28 void addEmployee() {
29     ofstream file("employee.dat", ios::binary | ios::app);
30     Employee emp;
31
32     cout << "Enter ID: ";
33     cin >> emp.id;
34     cout << "Enter Name: ";
35     cin.ignore();
36     cin.getline(emp.name, 30);
37     cout << "Enter Designation: ";
38     cin.getline(emp.designation, 30);
39     cout << "Enter Salary: ";
40     cin >> emp.salary;
41
42     file.write((char*)&emp, sizeof(emp));
43     file.close();
44
45     buildIndex(); // Rebuild index after adding
46     cout << "Employee added.\n";
47 }
48
49 void displayEmployee() {
50     int searchId;
51     cout << "Enter Employee ID to search: ";
52     cin >> searchId;
53
54     buildIndex(); // Ensure index is current
55 }
```

```

56     if (indexMap.find(searchId) == indexMap.end()) {
57         cout << "Employee not found.\n";
58         return;
59     }
60
61     ifstream file("employee.dat", ios::binary);
62     file.seekg(indexMap[searchId]);
63     Employee emp;
64     file.read((char*)&emp, sizeof(emp));
65     file.close();
66
67     cout << "Employee Details:\n";
68     cout << "ID: " << emp.id << "\n";
69     cout << "Name: " << emp.name << "\n";
70     cout << "Designation: " << emp.designation << "\n";
71     cout << "Salary: " << emp.salary << "\n";
72 }
73
74 void deleteEmployee() {
75     int deleteId;
76     cout << "Enter Employee ID to delete: ";
77     cin >> deleteId;
78
79     buildIndex();
80     if (indexMap.find(deleteId) == indexMap.end()) {
81         cout << "Employee not found.\n";
82         return;
83     }
84
85     ifstream infile("employee.dat", ios::binary);
86     ofstream temp("temp.dat", ios::binary);
87     Employee emp;
88
89     while (infile.read((char*)&emp, sizeof(emp))) {
90         if (emp.id != deleteId)
91             temp.write((char*)&emp, sizeof(emp));
92     }
93
94     infile.close();
95     temp.close();
96     remove("employee.dat");
97     rename("temp.dat", "employee.dat");
98
99     buildIndex();
100    cout << "Employee deleted.\n";
101 }
102
103 int main() {
104     int choice;
105     buildIndex(); // Initial index build
106
107     do {
108         cout << "\n--- Employee Management ---\n";
109         cout << "1. Add Employee\n";
110         cout << "2. Delete Employee\n";
111         cout << "3. Display Employee\n";
112         cout << "4. Exit\n";

```

```
113     cout << "Enter your choice: ";
114     cin >> choice;
115
116     switch (choice) {
117         case 1:
118             addEmployee();
119             break;
120         case 2:
121             deleteEmployee();
122             break;
123         case 3:
124             displayEmployee();
125             break;
126         case 4:
127             cout << "Exiting...\n";
128             break;
129         default:
130             cout << "Invalid choice!\n";
131     }
132     while (choice != 4);
133
134     return 0;
135 }
136
```

## OutPut:

```
D:\SE Computer\LAB CODES\DSA\DSA12.exe

--- Employee Management ---
1. Add Employee
2. Delete Employee
3. Display Employee
4. Exit
Enter your choice: 1
Enter ID: 101
Enter Name: Pravin
Enter Designation: Wagholi
Enter Salary: 200000
Employee added.

--- Employee Management ---
1. Add Employee
2. Delete Employee
3. Display Employee
4. Exit
Enter your choice: 1
Enter ID: 102
Enter Name: Mangesh
Enter Designation: Chandannagar
Enter Salary: 20000
Employee added.

--- Employee Management ---
1. Add Employee
2. Delete Employee
3. Display Employee
4. Exit
Enter your choice: 1
Enter ID: 103
Enter Name: Shiv
Enter Designation: Shikrapur
Enter Salary: 400000
Employee added.

--- Employee Management ---
1. Add Employee
2. Delete Employee
3. Display Employee
4. Exit
Enter your choice: 3
Enter Employee ID to search: 103
Employee Details:
ID: 103
Name: Shiv
Designation: Shikrapur
Salary: 400000
```

```
--- Employee Management ---
1. Add Employee
2. Delete Employee
3. Display Employee
4. Exit
Enter your choice: 2
Enter Employee ID to delete: 102
Employee deleted.

--- Employee Management ---
1. Add Employee
2. Delete Employee
3. Display Employee
4. Exit
Enter your choice: 3
Enter Employee ID to search: 103
Employee Details:
ID: 103
Name: Shiv
Designation: Shikrapur
Salary: 400000

--- Employee Management ---
1. Add Employee
2. Delete Employee
3. Display Employee
4. Exit
Enter your choice: 4
Exiting...

-----
Process exited after 96.83 seconds with return value 0
Press any key to continue . . . ■
```