Exercise 4: Employee Management System

1. Understand Array Representation

🔹 How Arrays Are Represented in Memory:

Arrays are stored in contiguous memory locations.

Each element is accessed via indexing, making access time constant O(1).

For example, an array of Employee[] references stores each reference at sequential memory addresses.

🔹 Advantages of Arrays:

Fast Access: O(1) access via index.

Memory Efficient: No extra pointers (like in linked lists).

Predictable performance: Great for static or fixed-size datasets.

2. Setup

Employee.java Class:

public class Employee {

int employeeId;

String name;

String position;

double salary;

public Employee(int id, String name, String position, double salary) {

this.employeeId = id;

this.name = name;

this.position = position;

this.salary = salary;

}

@Override

public String toString() {

return "[" + employeeId + ", " + name + ", " + position + ", ₹" + salary + "]";

}

}

3. Implementation

EmployeeManagementSystem.java

public class EmployeeManagementSystem {

private Employee[] employees;

private int count;

public EmployeeManagementSystem(int size) {

employees = new Employee[size];

count = 0;

}

public void addEmployee(Employee e) {

if (count < employees.length) {

employees[count++] = e;

System.out.println("Employee added.");

} else {

System.out.println("Employee list is full.");

}

}

public Employee searchEmployee(int id) {

for (int i = 0; i < count; i++) {

if (employees[i].employeeId == id)

return employees[i];

}

return null;

}

public void displayEmployees() {

if (count == 0) {

System.out.println("No employees in the system.");

} else {

for (int i = 0; i < count; i++)

System.out.println(employees[i]);

}

}

public void deleteEmployee(int id) {

for (int i = 0; i < count; i++) {

if (employees[i].employeeId == id) {

for (int j = i; j < count - 1; j++)

employees[j] = employees[j + 1];

employees[--count] = null;

System.out.println("Employee deleted.");

return;

}

}

System.out.println("Employee not found.");

}

}

Main.java

public class Main {

public static void main(String[] args) {

EmployeeManagementSystem ems = new EmployeeManagementSystem(5);

Employee e1 = new Employee(101, "Alice", "Manager", 75000);

Employee e2 = new Employee(102, "Bob", "Developer", 60000);

Employee e3 = new Employee(103, "Charlie", "Tester", 50000);

ems.addEmployee(e1);

ems.addEmployee(e2);

ems.addEmployee(e3);

System.out.println("\nAll Employees:");

ems.displayEmployees();

System.out.println("\nSearching for Employee with ID 102:");

System.out.println(ems.searchEmployee(102));

System.out.println("\nDeleting Employee with ID 101:");

ems.deleteEmployee(101);

System.out.println("\nAll Employees After Deletion:");

ems.displayEmployees();

}

}

OUTPUT:

A black screen with white text

AI-generated content may be incorrect.

**4. Analysis**

Time Complexity

Operation Time Complexity

Add Employee O(1)

Search Employee O(n)

Traverse O(n)

Delete Employee O(n)