class Employee {

int employeeId;

String name;

String position;

double salary;

Employee(int employeeId, String name, String position, double salary) {

this.employeeId = employeeId;

this.name = name;

this.position = position;

this.salary = salary;

}

@Override

public String toString() {

return employeeId + " - " + name + " - " + position + " - Salary: " + salary;

}

}

public class EmployeeManagement {

private Employee[] employees;

private int count;

public EmployeeManagement(int capacity) {

employees = new Employee[capacity];

count = 0;

}

public void addEmployee(Employee employee) {

if (count < employees.length) {

employees[count++] = employee;

} else {

System.out.println("Array is full. Cannot add more employees.");

}

}

public Employee searchEmployee(int employeeId) {

for (Employee employee : employees) {

if (employee != null && employee.employeeId == employeeId) {

return employee;

}

}

return null;

}

public void deleteEmployee(int employeeId) {

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

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

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

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

}

employees[--count] = null;

return;

}

}

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

}

public void displayEmployees() {

for (Employee employee : employees) {

if (employee != null) {

System.out.println(employee);

}

}

}

public static void main(String[] args) {

EmployeeManagement management = new EmployeeManagement(5);

management.addEmployee(new Employee(1, "Alice", "Manager", 75000));

management.addEmployee(new Employee(2, "Bob", "Developer", 55000));

management.displayEmployees();

System.out.println("Searching Employee with ID 1:");

System.out.println(management.searchEmployee(1));

management.deleteEmployee(2);

management.displayEmployees();

}

}

OUTPUT:

