**Employee Management System**

**Purpose**:

The Employee Management System is a simple program that allows users to manage a list of employees. Users can add new employees, display employee information, and view a list of all employees. The system is designed to ensure each employee has a unique ID and validates the name provided for employees.

**Class Structure**:

1. Employee Class:

* int id: The unique identifier for each employee.
* String name: The name of the employee.
* double salary: The salary of the employee.
* Employee(int id, String name, double salary): A constructor to initialize an employee with an ID, name, and salary.
* int getId(): Returns the employee's ID.
* void setId(int id): Sets the employee's ID.
* String getName(): Returns the employee's name.
* void setName(String name): Sets the employee's name.
* double getSalary(): Returns the employee's salary.
* void setSalary(double salary): Sets the employee's salary.
* String toString(): Returns a string representation of the employee.

1. EmployeeDriver Class:

* main(String[] args): The entry point of the program that manages employee data. It provides a text-based menu for user interaction and options to add employees, display employee information, and view all employees.
* Employee findEmployeeById(ArrayList<Employee> employees, int id): A method to find an employee by their ID within the provided list of employees.
* boolean isValidName(String name): A static method to validate employee names, ensuring they consist of alphabetic characters and spaces.

**Usage Examples:**

Here are some examples of how to use the program:

1. Adding a new employee:

* Select option 1 from the menu.
* Enter a unique employee ID.
* Enter the employee's name, ensuring it contains only letters and spaces.
* Enter the employee's salary.
* The new employee is added to the list.

1. Displaying employee information:

* Select option 2 from the menu.
* Enter the ID of the employee whose information you want to display.
* The program will display the details of the employee if found.

1. Viewing all employees:

* Select option 3 from the menu.
* The program will list all the employees in the system.

1. Exiting the program:

* Select option 4 from the menu to exit the program.

Please note that the program is designed to handle exceptions gracefully and provides user-friendly input validation to ensure data integrity.