📌 Coding Challenge: Employee Management System Using LinkedList

You are required to implement a **Employee Management System** using **Java's LinkedList**, supporting full **CRUD** functionality.

Employees must be stored inside a **LinkedList**, and all operations should happen **based on Employee ID**.

📝 Problem Description

Design a program that performs the following operations using a `LinkedList<Employee>`:

1. Register Employee

Add a new employee with the following fields:

- Employee Name

- Employee ID

- Employee Gender

- Employee Salary

If an employee with the same ID already exists, ignore the insertion.

---

2. Delete Employee

Delete an employee using their **Employee ID**.
 If no employee exists with that ID, ignore the operation.

---

3. Search Employee

Given an Employee ID, return the employee's complete details.
 If no employee is found, return `"Employee Not Found"`.

4. Update Employee

Update an employee's:

- Name

- Gender

- Salary

using their Employee ID.
If the employee doesn't exist, ignore the update.

---

5. Display All Employees

Return a list of all employees in insertion order.

📥 Input Format (LeetCode Style)

Input consists of multiple operations:

```
REGISTER name id gender salary
DELETE id
SEARCH id
UPDATE id name gender salary
DISPLAY
END
```

- Processing stops when the command END appears.

📤 Output Format

Print output **only for SEARCH and DISPLAY**.

📌 Sample Input

```
REGISTER Amar 101 M 75000
REGISTER Riya 102 F 68000
REGISTER John 103 M 82000
SEARCH 102
UPDATE 102 Priya F 70000
SEARCH 102
DELETE 101
DISPLAY
END
```

📌 Sample Output

```
Employee Found: Riya 102 F 68000
Employee Updated: Priya 102 F 70000
Employee List:
Priya 102 F 70000
John 103 M 82000
```