**Exercise 2: Employee Management System - Creating Entities**

Business Scenario:

Define JPA entities for Employee and Department with appropriate relationships.

1.Introduction:

This exercise involves defining JPA entities for an Employee Management System. The entities include Employee and Department, which are mapped to database tables. The Department entity has a one-to-many relationship with the Employee entity.

**Entities Definition---**

**Department Entity**

The Department entity represents a department within the company. It contains the following fields:

* id: A unique identifier for each department.
* name: The name of the department.

**Employee Entity**

The Employee entity represents an employee within the company. It contains the following fields:

* id: A unique identifier for each employee.
* name: The name of the employee.
* email: The email address of the employee.
* department: The department to which the employee belongs, mapped as a many-to-one relationship.

**Application Configuration**

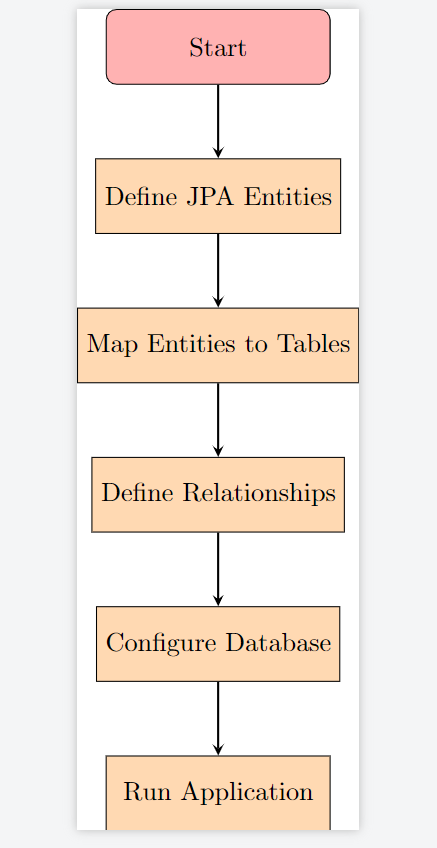
* The application is configured with an H2 in-memory database for simplicity. The application.properties file specifies the connection details and Hibernate dialect.
* The pom.xml file includes dependencies for Spring Boot, JPA, Web, H2 database, and Lombok.

#### Application Entry Point

* The EmployeeManagementSystemApplication class is the entry point of the application, which starts the Spring Boot application.Home Controller

The HomeController provides a simple endpoint to verify the application is running.

**FLOWCHART :**



* **Start**: The process begins.
* **Define JPA Entities**: Create the Employee and Department entities.
* **Map Entities to Tables**: Use JPA annotations to map these entities to database tables.
* **Define Relationships**: Establish a one-to-many relationship between Department and Employee.
* **Configure Database**: Set up the database connection and properties in application.properties.
* **Run Application**: Start the Spring Boot application to initialize the database and run the system.

**CLASS DIAGRAM :**



**Explanation:**

* **Packages:** The entities (Department and Employee) are grouped in the com.example.employeemanagementsystem.entity package.
* **Annotations:** Each entity class includes annotations such as @Entity, @Table, @Id, @GeneratedValue, @OneToMany, and @ManyToOne.
* **Relationships:** The relationship between Department and Employee is represented with a 1 to 0..\* notation, indicating that one department can have multiple employees.
* **Application Classes:** The EmployeeManagementSystemApplication and HomeController classes are also included, with appropriate annotations to represent Spring Boot configuration and REST controller functionality.