**Exercise 1: Employee Management System - Overview and Setup**

Business Scenario:

You are developing an employee management system that will manage employee data, departments, and their relationships.

**1. Introduction**

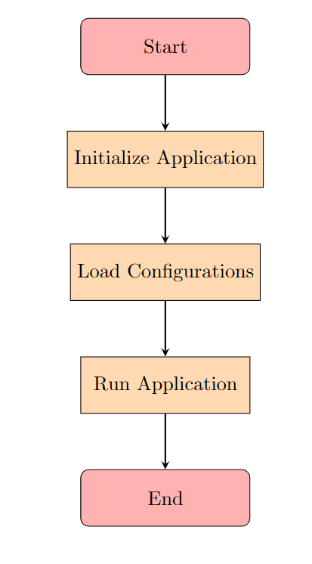
The Employee Management System (EMS) is designed to manage employee data, departments, and the relationships between them. This project is implemented using the Spring Boot framework, which simplifies the development of Java applications with a production-ready environment.

**2. Project Structure**

**2.1. Project Setup** The EMS project is structured as a typical Spring Boot application with the following main components:

* **Main Application Class:** The entry point for the Spring Boot application.
* **Configuration File:** application.properties for setting up database configurations.
* **Dependencies:** Managed using Maven, with relevant dependencies included in pom.xml.
* **3. Spring Boot Application**
* **3.1. Main Application Class**
* The EmployeemanagementsystemApplication.java is the main class that bootstraps the Spring Boot application:
* **3.2. Configuration File**
* The application.properties file contains the database configuration using an in-memory H2 database:
* **3.3. Maven Configuration (pom.xml)**
* The pom.xml file includes essential dependencies for building and running the application:

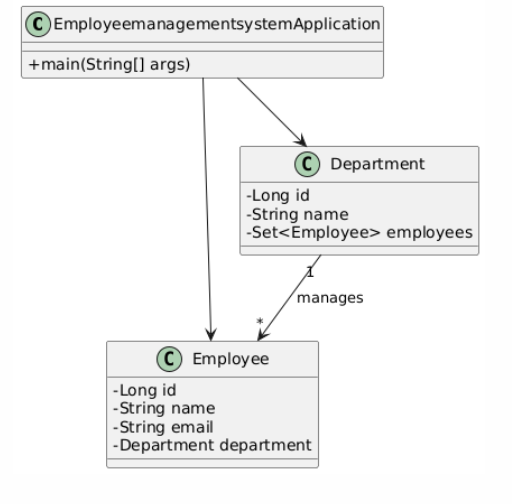
**FLOWCHART :**



Explanation :

* **Start:** The process begins with starting the application.
* **Initialize Application:** The application is initialized, setting up necessary components and configurations.
* **Load Configurations:** The system loads the configurations defined in the application.properties file, such as database settings.
* **Run Application:** The main method of the EmployeemanagementsystemApplication class is executed, launching the Spring Boot application.
* **End:** The flow concludes, and the application is now running, ready to handle requests.

**CLASS DIAGRAM :**



Explanation :

1. **Classes:**
   * **EmployeemanagementsystemApplication:** The main application class that starts the system.
   * **Employee:** Represents an employee with attributes like id, name, email, and a reference to their Department.
   * **Department:** Represents a department with attributes like id, name, and a list of employees.
2. **Relationships:**
   * The application class interacts with both Employee and Department.
   * A Department can manage multiple Employees, showing a one-to-many relationship.

This diagram visually outlines how the application, employees, and departments are structured and related.