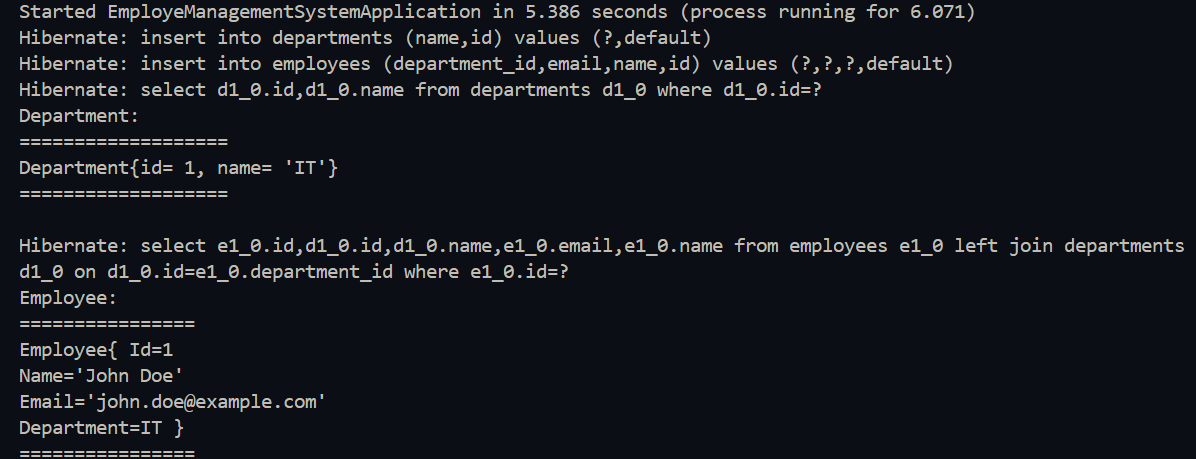
**Exercise 2: Employee Management System - Creating Entities**

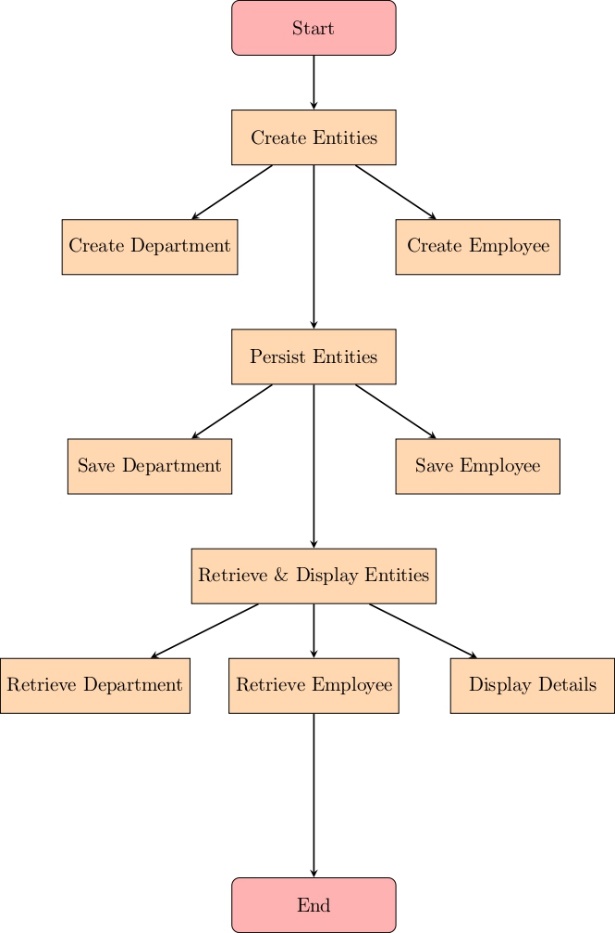
* **Creating JPA Entities:**
* **Employee Entity:** The *Employee* entity includes fields like *id, name, email*, and *department*. It uses JPA annotations to map these fields to database columns.
* **Department Entity:** The *Department* entity includes fields like *id* and *name*. It also defines a one-to-many relationship with the *Employee* entity using the @*OneToMany* annotation.
* **Mapping Entities to Database Tables:**
* **Annotations:** The entities are mapped to database tables using annotations like @*Entity, @Table, @Id*, and @*GeneratedValue*.
* **Relationships:** A one-to-many relationship is established between *Department* and *Employee* using the @*OneToMany* and @*ManyToOne* annotations, ensuring that each department can have multiple employees
* **CommandLineRunner Execution:**
* **Inserting Data:** The *CommandLineRunner* is used to insert initial data into the database by creating department and employee instances.
* **Output:** The program then retrieves and prints the details of the created departments and employees to verify the relationships and data insertion.

The output of the code –

This is the Spring output : insert an employee and the department into the database to show that code is working



* **The Flow chart of the program:**



**Explanation:**

* **Create Entities-**
* ***Create Department:*** Instantiate the Department object.
* ***Create Employee***: Instantiate the Employee object and link it to a department.
* **Persist Entities**:
* ***Save Department***: Save the Department object to the database.
* ***Save Employee***: Save the Employee object to the database, with its department association.
* **Retrieve & Display Entities**:
  + ***Retrieve Department****:* Fetch the saved Department from the database.
  + ***Retrieve Employee****:* Fetch the saved Employee from the database.
  + ***Display Details****:* Print the details of the Department and Employee to the console.