**Employee Management System - Creating Entities**

1. **Creating JPA Entities:**
   1. Defined **Employee** entity with fields: **id, name, email, department**.
   2. Defined **Department** entity with fields: **id, name**.
2. **Mapping Entities to Database Tables:**

* **Employee Entity:**
  + **@Entity** specifies that the class is a JPA entity.
  + **@Table(name = "employees")** maps the entity to the "employees" table in the database.
  + **@Id** and **@GeneratedValue(strategy = GenerationType.IDENTITY)** marks id as the primary key and uses auto increment for the value.
  + **@ManyToOne(fetch = FetchType.LAZY)** defines a many-to-one relationship with the Department entity. The fetch = FetchType.LAZY ensures that the department is loaded only when accessed.
  + **@JoinColumn(name = "department\_id")** specifies the foreign key column in the "employees" table that references the "departments" table.
* **Department Entity:**
  + **@Entity** and **@Table(name = "departments")** similar to the Employee entity, this maps the class to the "departments" table.
  + **@OneToMany(mappedBy = "department", cascade = CascadeType.ALL, orphanRemoval = true)** establishes a one-to-many relationship with the Employee entity. The mappedBy attribute indicates that the department field in the Employee entity owns the relationship. cascade = CascadeType.ALL ensures that all operations (like persist, remove) on the department are cascaded to the related employees. orphanRemoval = true removes any Employee that is no longer associated with a Department.