**EXERCISE 10**

**1. Hibernate-Specific Annotations**

Hibernate provides several annotations that can be used to customize entity mappings beyond the standard JPA annotations. These annotations can help you fine-tune your data models and improve performance.

Example Hibernate-Specific Annotations

* @Type: Specifies a custom Hibernate type for a field.
* @Fetch: Defines the fetching strategy for associations.
* @BatchSize: Controls the batch size for fetching collections.

**2. Configuring Hibernate Dialect and Properties**

Configuring Hibernate’s dialect and properties can help optimize performance and ensure compatibility with your database. You typically configure these properties in application.properties.

**3. Batch Processing**

Batch processing allows you to perform bulk operations more efficiently. Hibernate can handle batch processing for inserts, updates, and deletes.

**@Transactional**: Ensures that the batch insert operation is performed within a single transaction.

**employeeRepository.flush()**: Flushes the persistence context, forcing Hibernate to execute the SQL statements for the current batch.

**employeeRepository.clear()**: Clears the persistence context to avoid memory issues with large datasets.