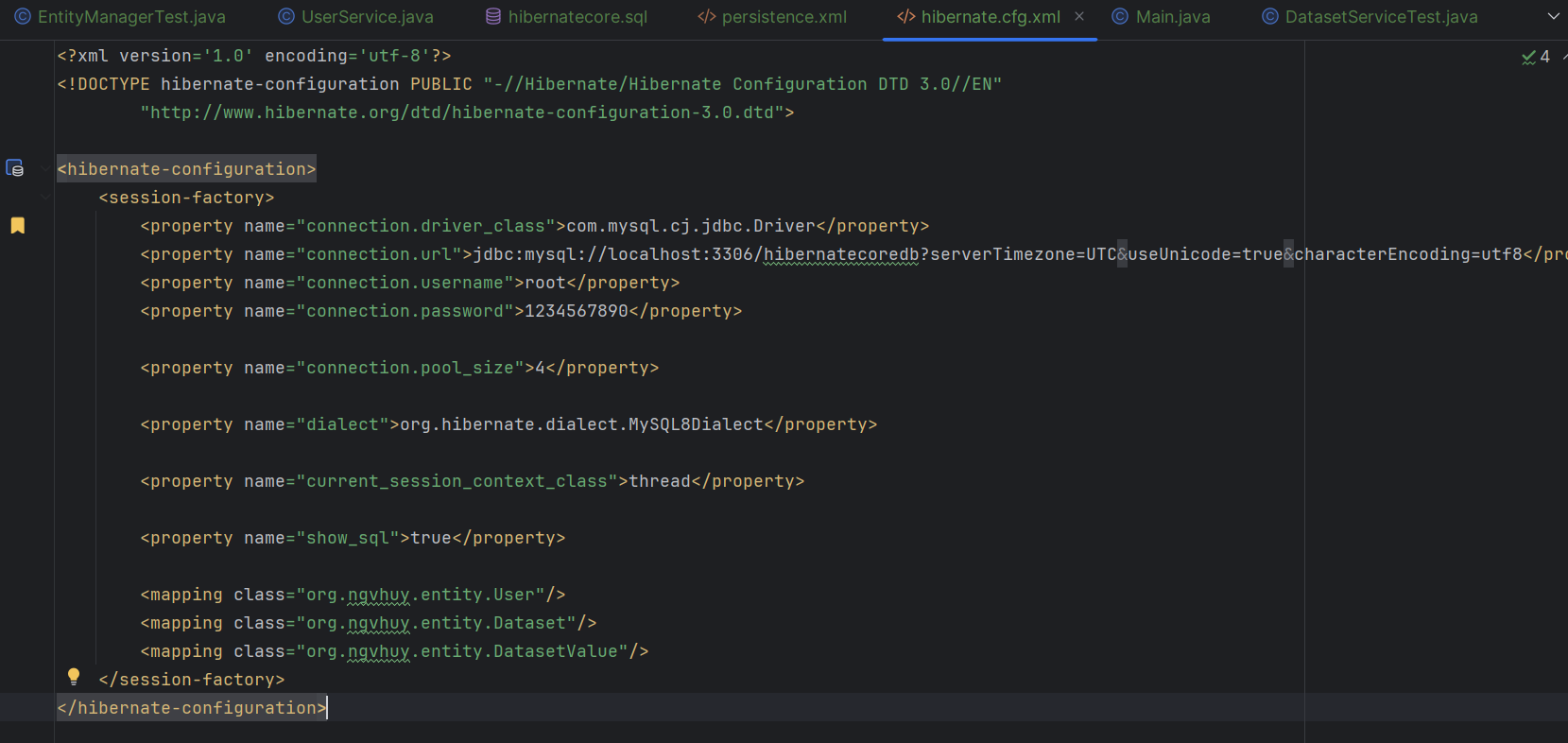
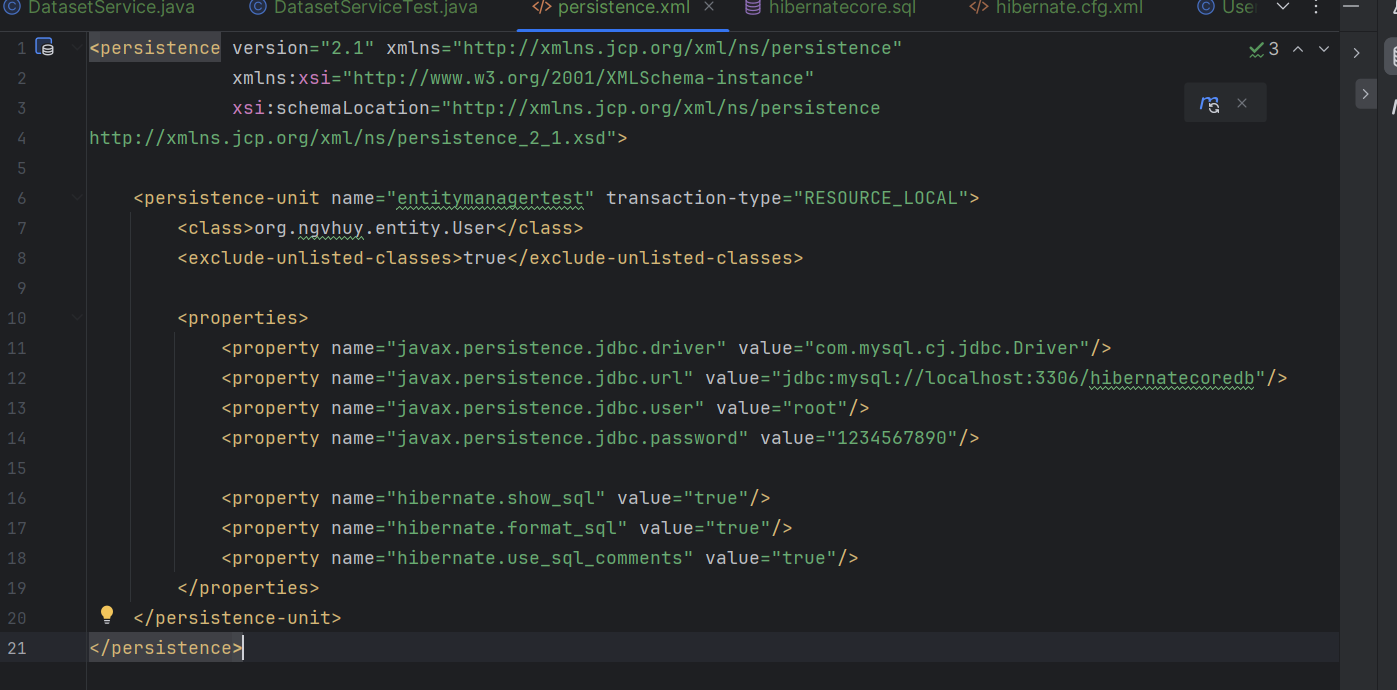
1. Lời mở đầu:

* Các bài lab dùng để thực hành, từ đó có cái nhìn sâu về hibernate
* File doc này chỉ mô tả một phần những gì đã tìm hiểu và test, có những thứ chưa thể diễn tả trong doc vì còn phải dành thời gian cho việc tìm hiểu thứ khác
* File doc này chưa phải kiến thức cuối cùng, nghĩa là vẫn sẽ được bổ sung về sau
* File doc này chỉ mô tả các bài lab, không mô tả về các kiến thức khác (kiến trúc hibernate, vòng đời entity, kiến trúc jpa,…), chúng sẽ được mô tả ở file doc khác

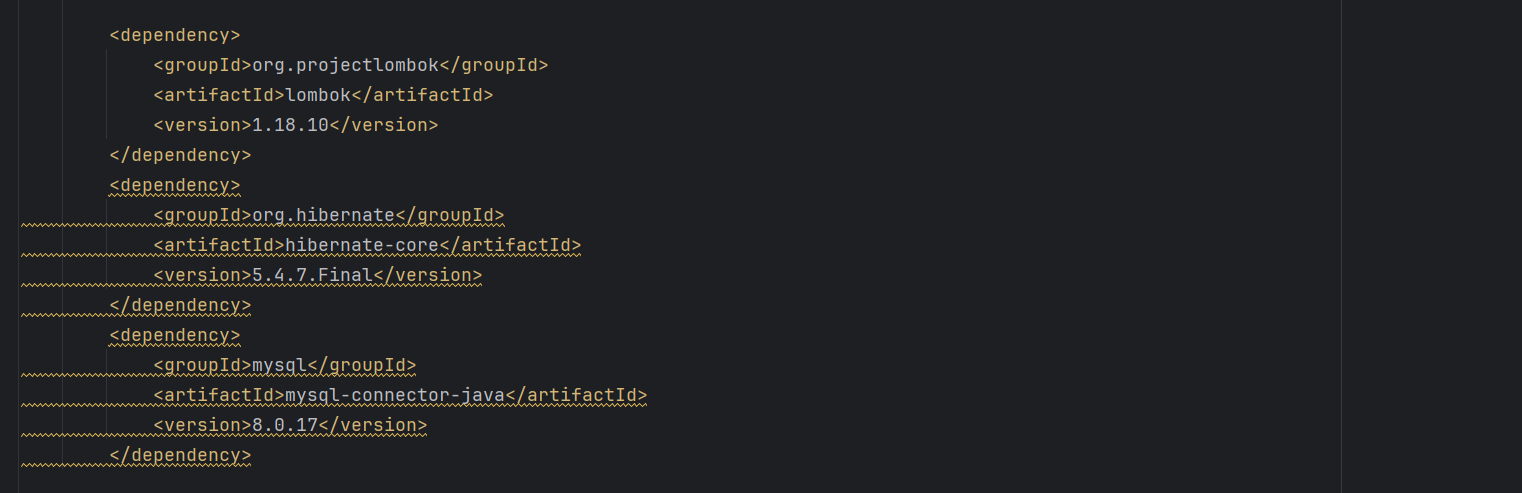
1. Thực hành:
   1. Cấu hình:
      1. hibernate.cfg.xml



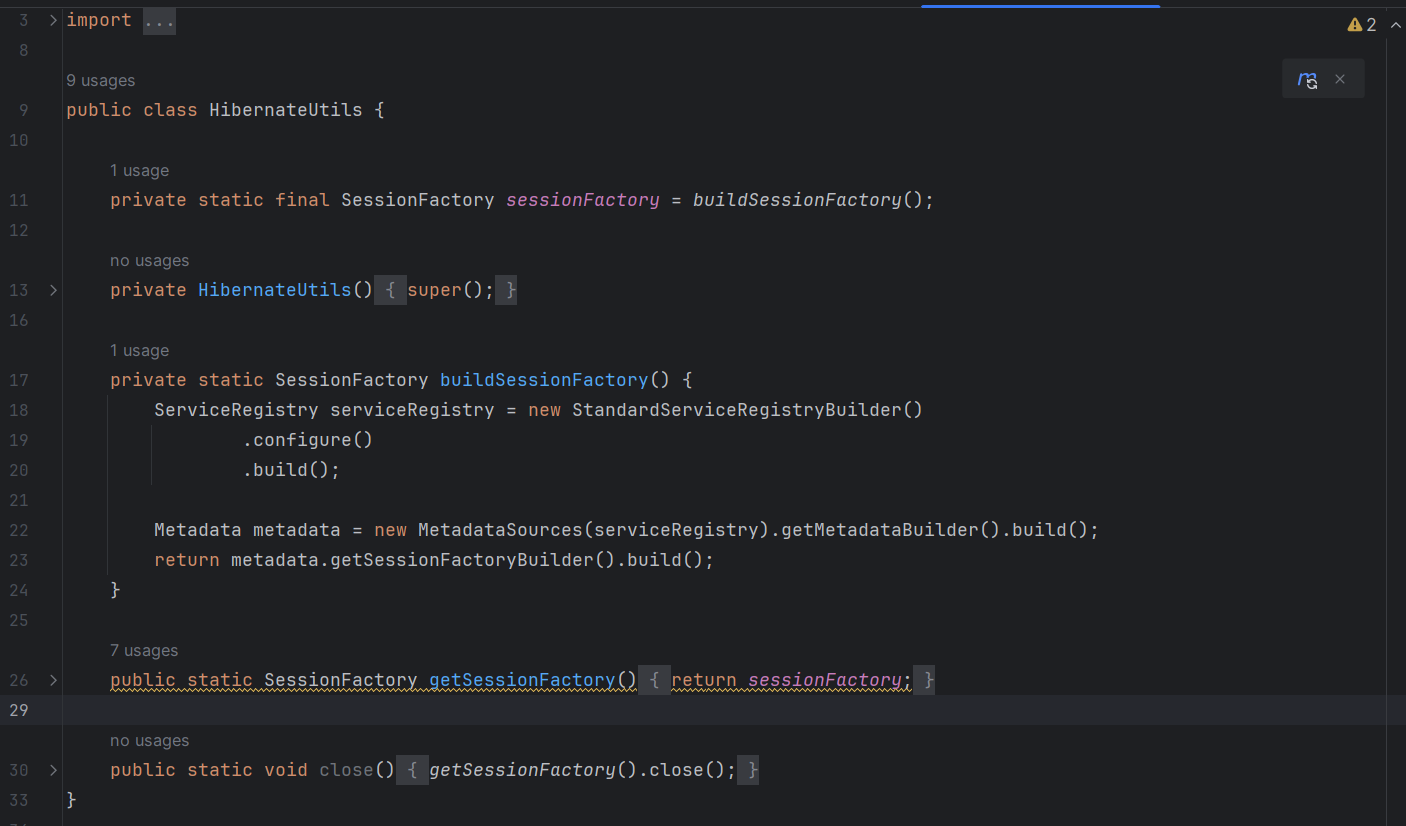
* + 1. persistence.xml (với EntityManager)



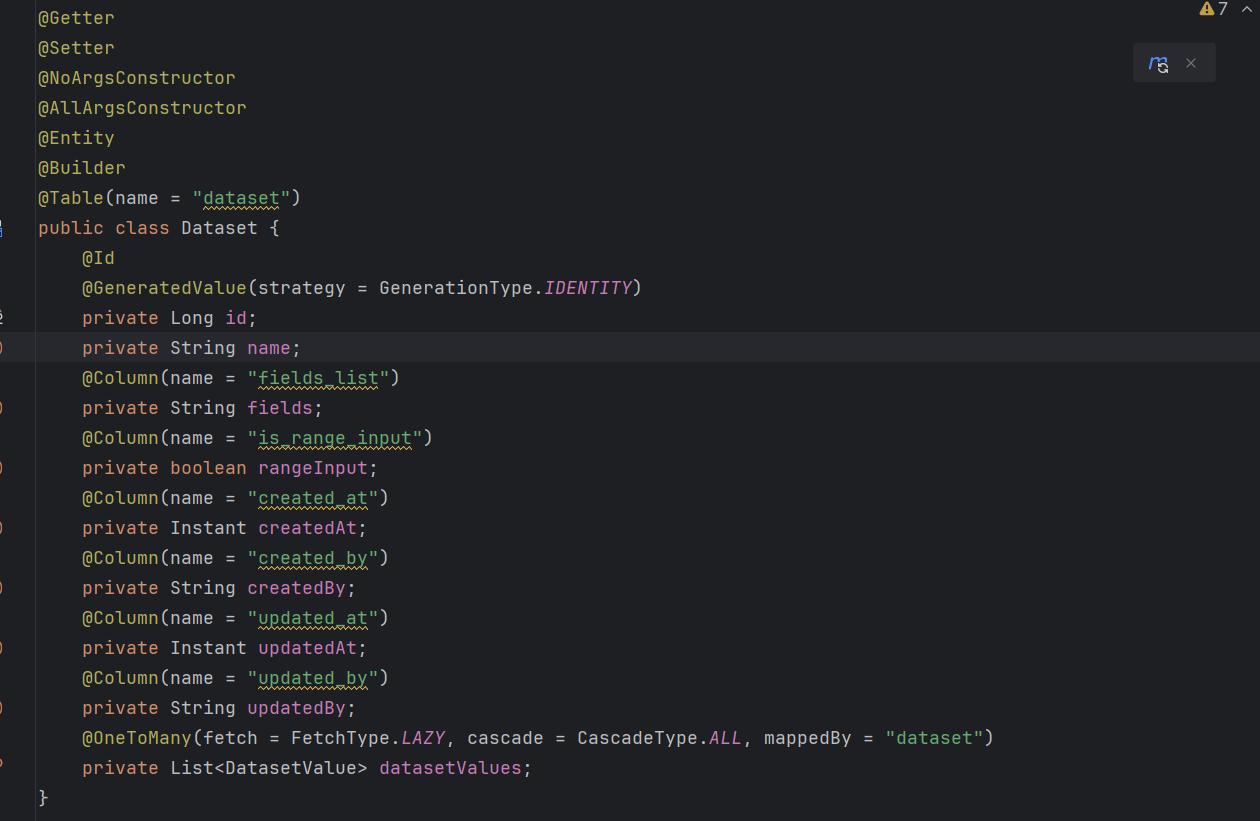
* + 1. pom.xml

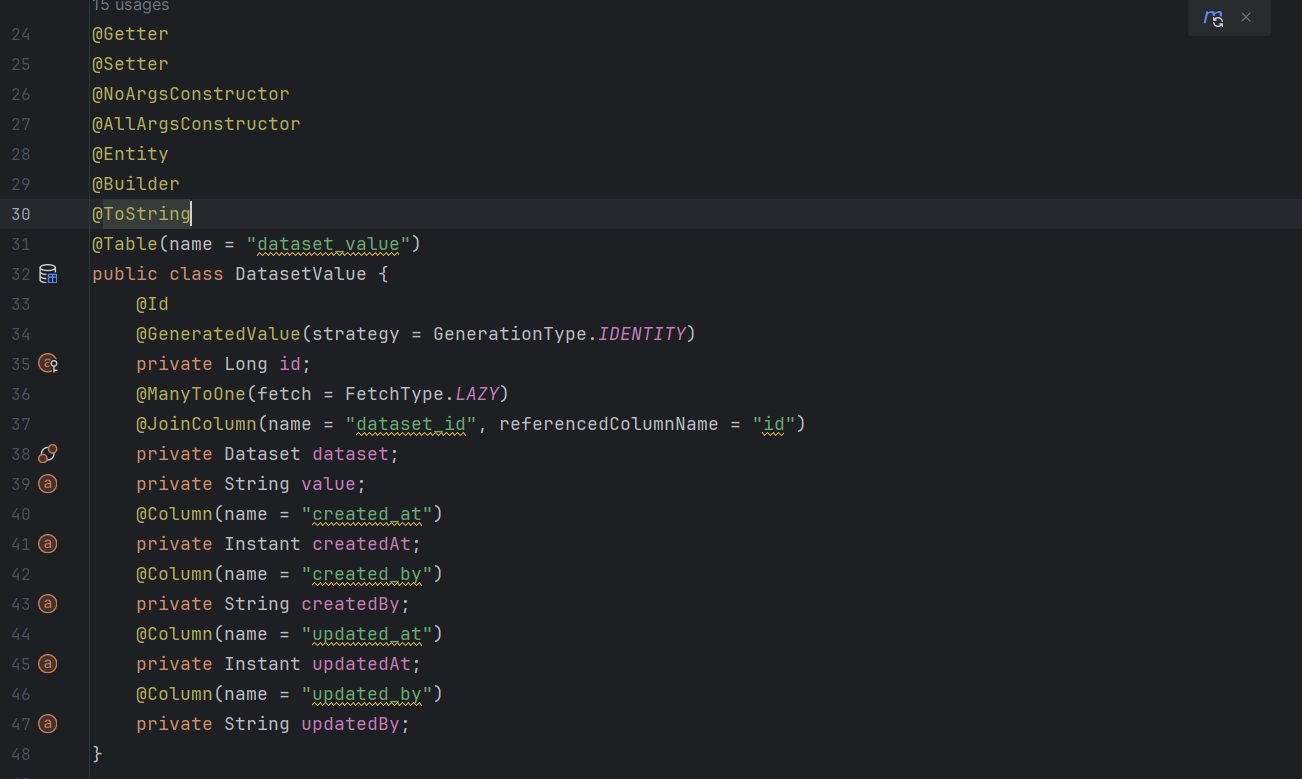


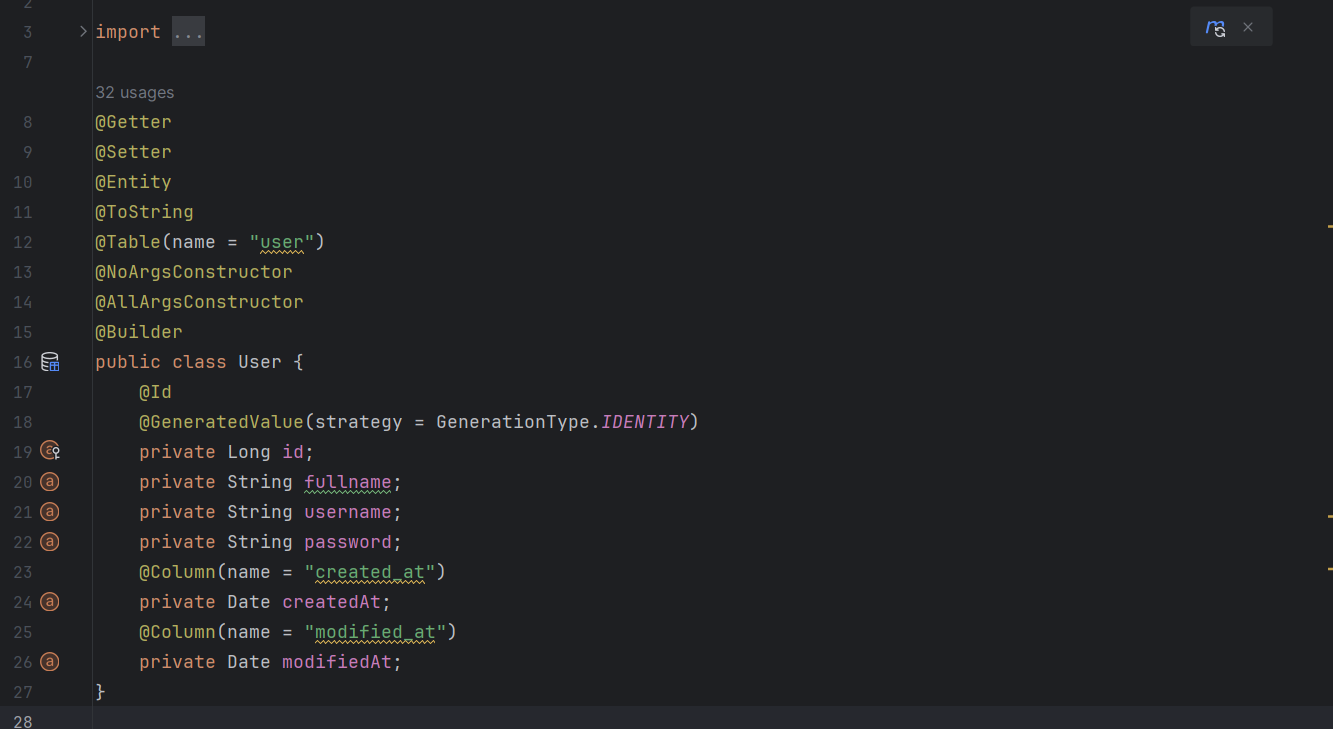
* + 1. org.ngvhuy.config.HibernateUtils



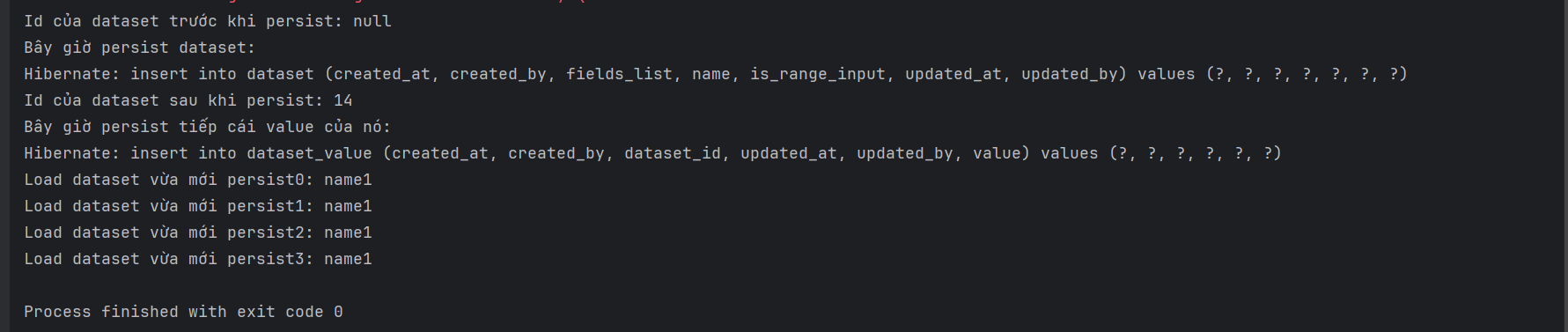
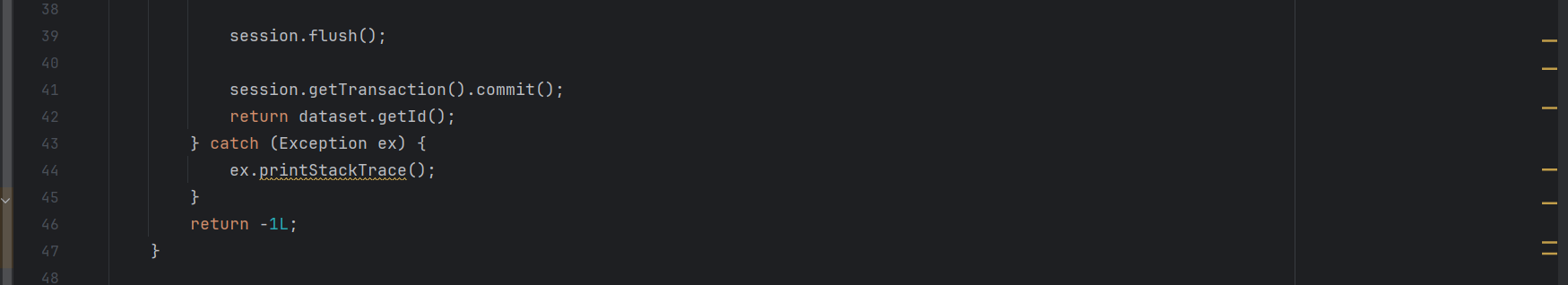
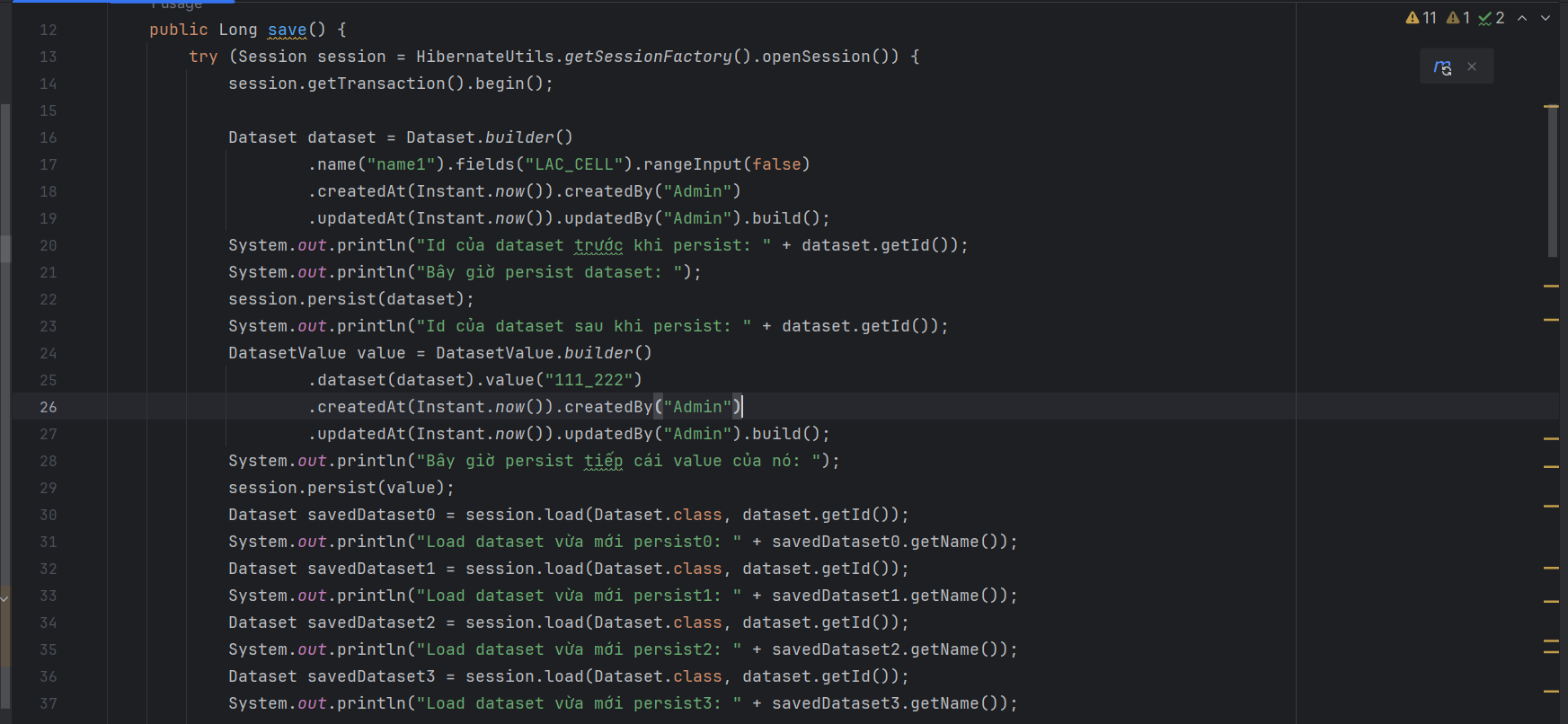
* + 1. Các class entity:





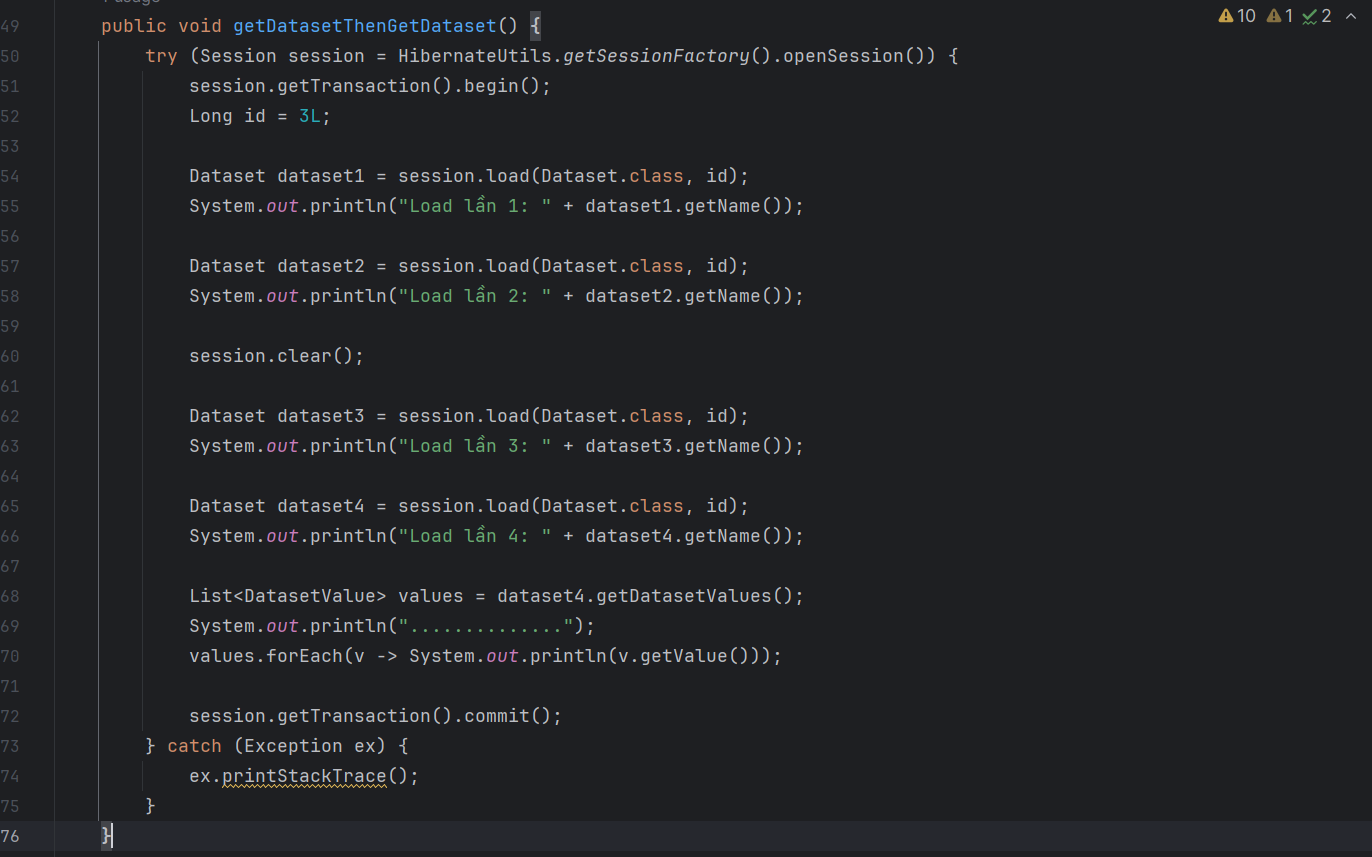


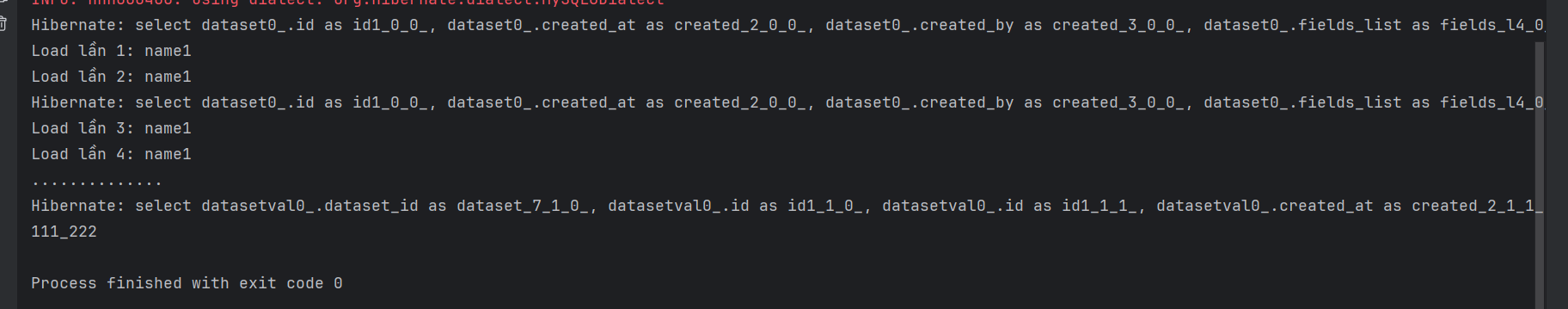
* 1. Thực hành:
     1. Save



Nhận xét:

* Trước khi persist thì entity ở trạng thái transient và chưa có id, sau khi persist thì entity ở trạng thái persistence, ở trạng thái này dữ liệu được ánh xạ tới 1 row trong db và được gán id.
* Khi entity được persist thì entity được lưu vào first level cache, kể từ đó việc load() theo id sẽ không truy vấn đến db (nếu chưa clear cache)
  + 1. Clear cache, Lazy load

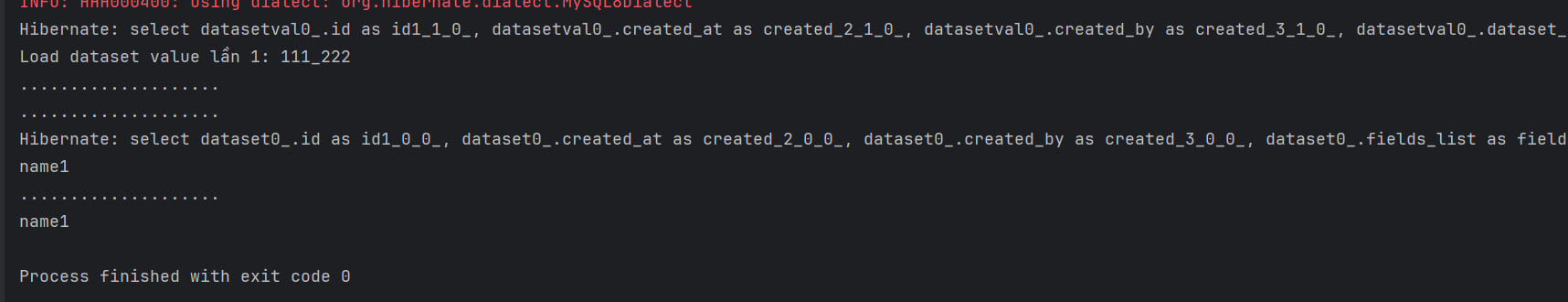




Nhận xét:

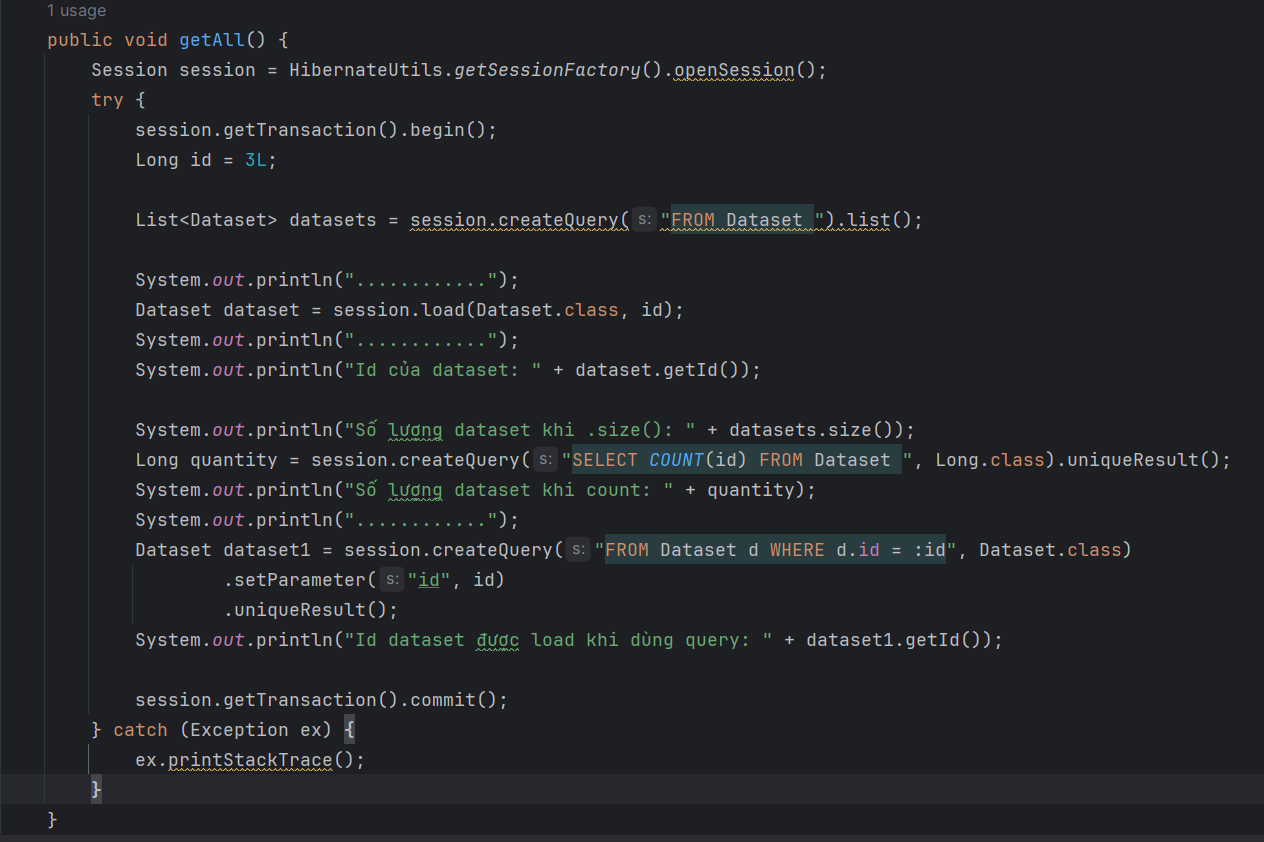
* Sau khi clear cache thì load entity bằng id phải thực hiện lại truy vấn tới db
* Đối với Lazy load, việc gọi .getDatasetValue() chương trình chưa load dữ liệu từ db, mà phải thực hiện các action trên datasetValue thì mới truy vấn tới db.
  + 1. Load entity con xong load entity cha:

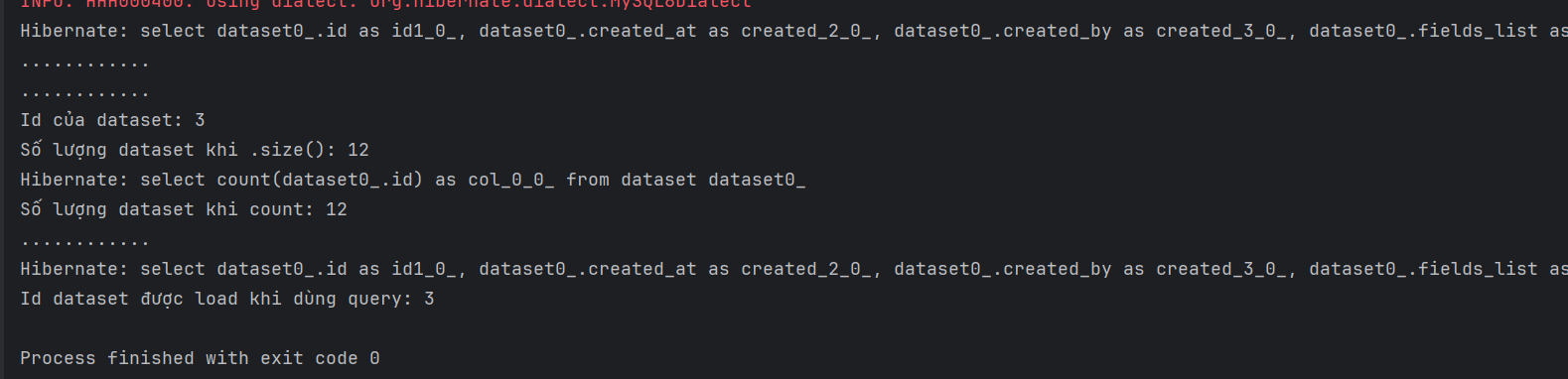




Nhận xét:

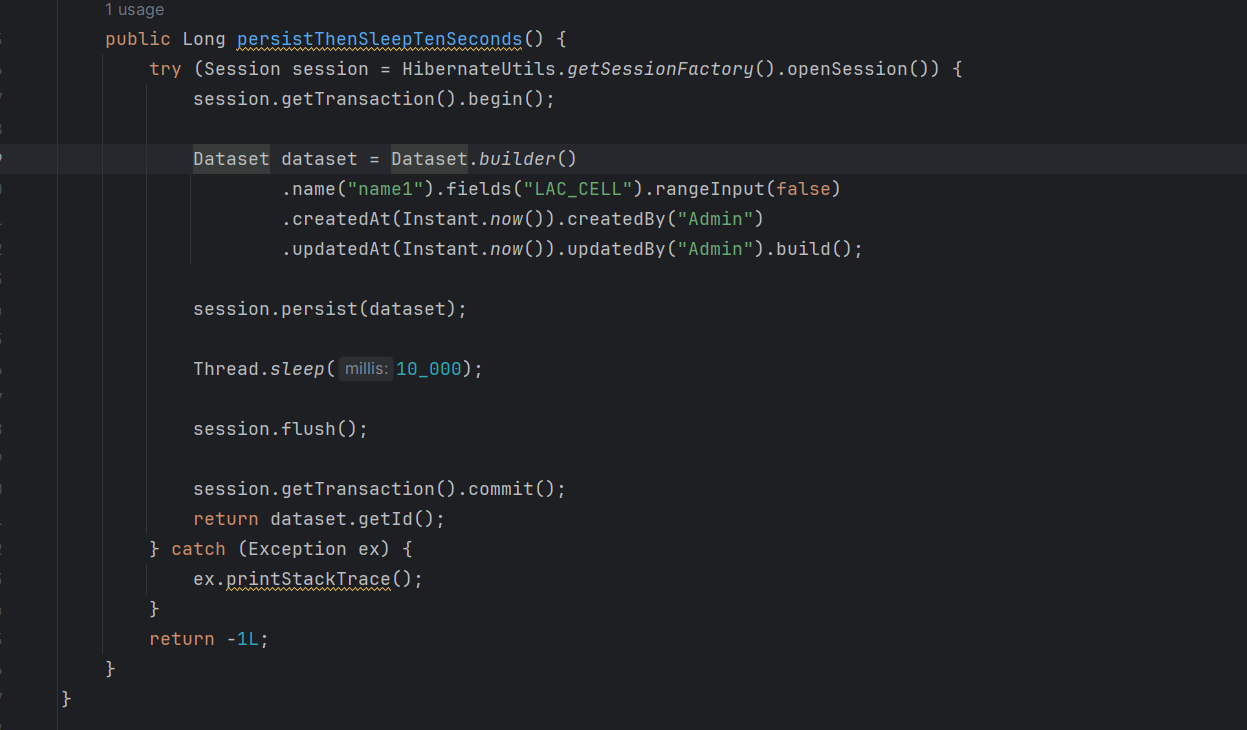
* Khi load entity con, session thực hiện truy vấn tới db, sau đó từ entity con lấy ra entity cha thì sẽ lấy ra một entity “ảo”, entity này chỉ có id, dẫn tới việc load entity cha theo id sẽ không truy vấn tới db, nhưng khi get các thuộc tính của entity cha thì lúc đó sẽ thực hiện truy vấn tới db.
  + 1. List all entity, sử dụng sql query





Nhận xét:

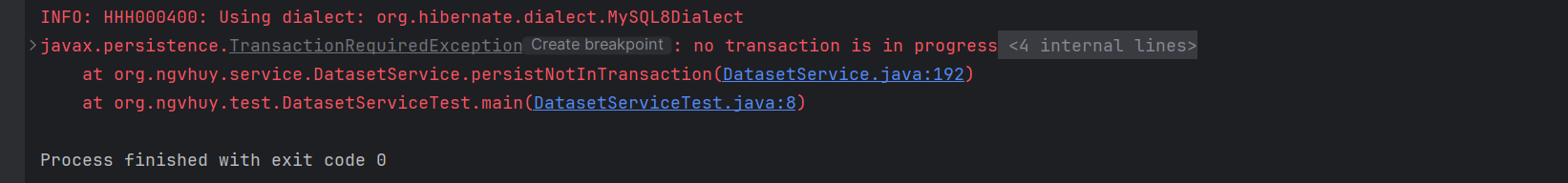
* Khi load nhiều (tất cả) entity từ db, thì chúng đều được lưu vào first level cache, việc load lại entity nào trong số đó về không dẫn đến truy vấn tới db
* Hibernate sẽ không load từ trong cache đối với sql query
  + 1. Persist xong slepp 10s



Nhận xét:

* Khi persist entity chưa được đẩy vào db ngay, mà phải flush (session.commit, session.close) ví dụ này chứng minh cho cơ chế quản lý life cycle của entity hibernate
  + 1. Thực hiện save mà không có transaction:

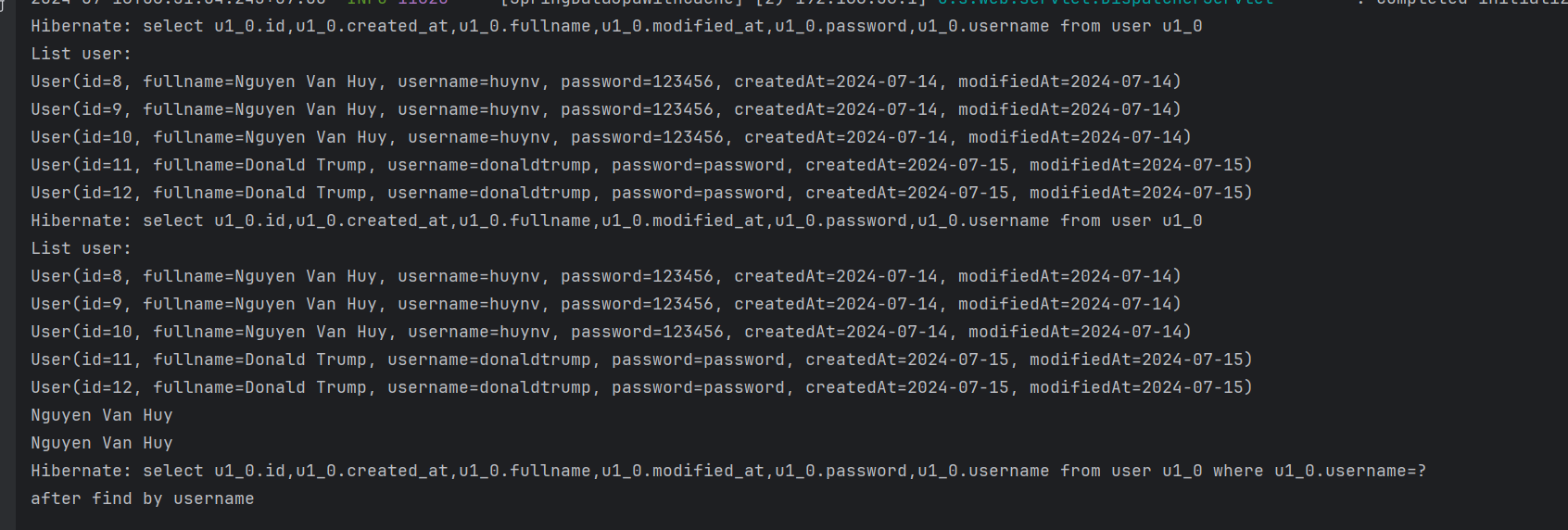




Nhận xét:

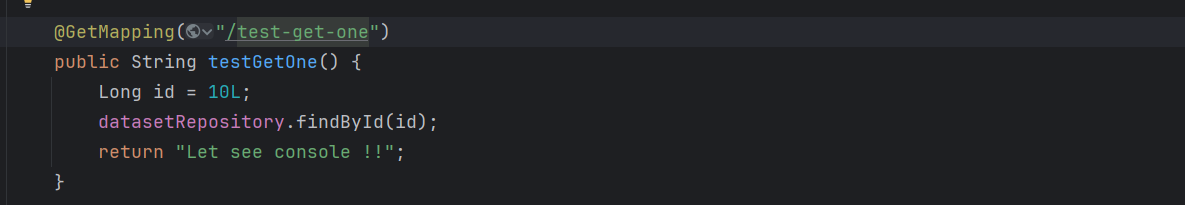
* Trong hibernate, luôn phải có transaction
  + 1. Spring data spa – findAll()

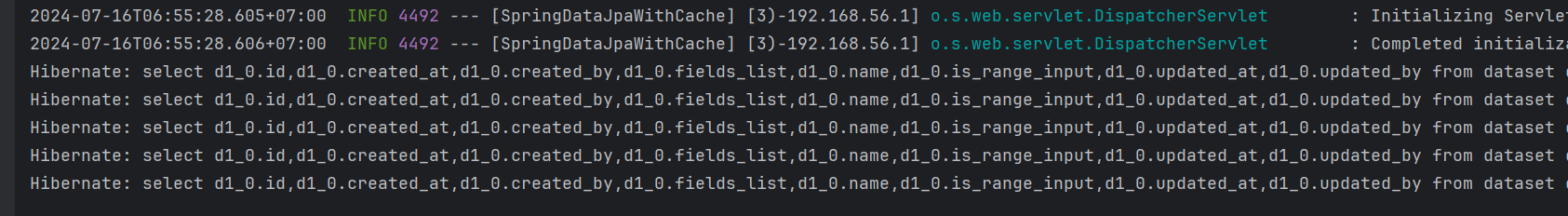




Nhận xét:

* Khi thực hiện findAll() thì tất cả entity sẽ được lưu vào first level cache, và sau đó việc findById sẽ được tìm trong cache trước tiên. Tuy nhiên việc findAll() lại thì lại truy vấn tới db
* Đối với các phương thức findBy khác (không phải findById) thì sẽ không tìm trong first level cache
  + 1. Spring data jpa – call api nhiều lần:

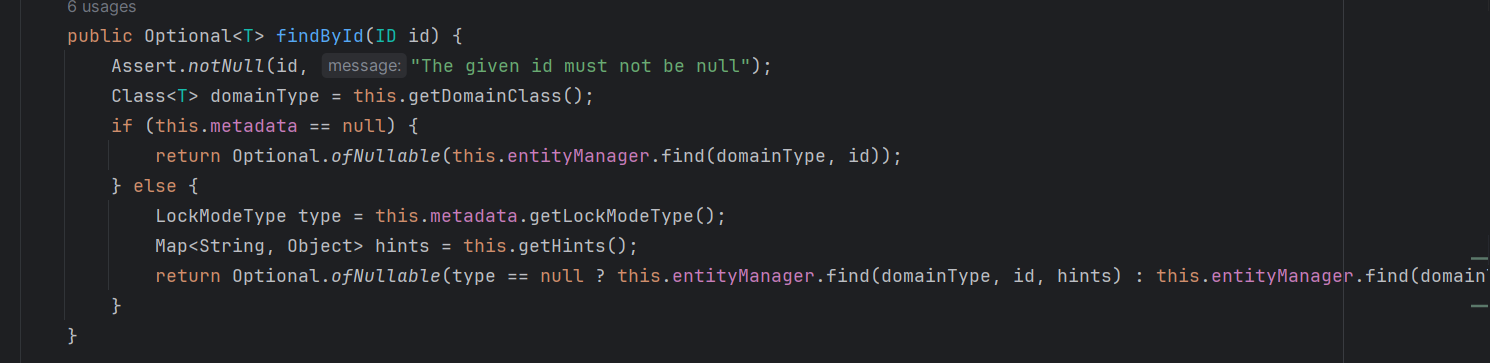


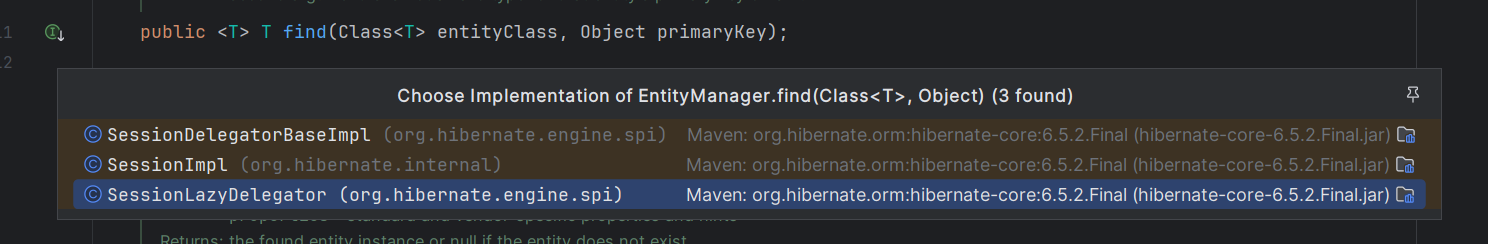


Nhận xét:

* Với việc thực hiện call api 5 lần, hibernate show sql 5 lần, chứng tỏ mỗi lần call api là một session riêng, và chúng ko sử dụng chung first level cache với nhau
  + 1. Mối quan hệ giữa hibernate, jpa, spring data jpa:







Nhận xét:

* Jpa là đặc tả - nghĩa là định nghĩa chung, là interface, được các công nghệ triển khai trong đó có hibernate
* Hibernate là công nghệ orm – nghĩa là có code, thuật toán bên trong
* Spring data jpa: là bộ thư viện xây dựng trên jpa, sử dụng hibernate để triển khai
* Các hàm của spring data jpa gọi tới EntityManager (1 phần của kiến trúc JPA), EntityManager lại gọi tới Session (1 phần của kiến trúc Hibernate)