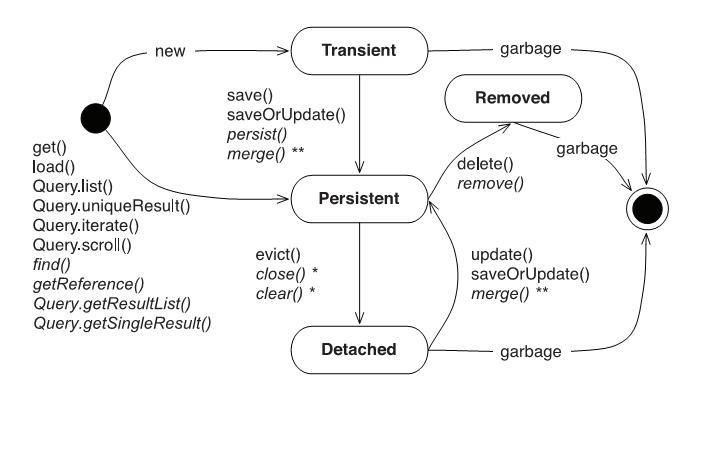
**HIBERNATE LIFECYCLE**



1. **Transient**: khi một đối tượng mới được tạo, không có một mối quan hệ nào với database, có nghĩa là không có quan hệ với một session object, khi khởi tạo một entity thì nó sẽ vào trạng thái transient
2. **Persisstent**: có quan hệ với db, kết nối với một session object duy nhất. khi kết nối với db thì dữ liệu được đồng bộ với db,

Trở thành một persistent thì có thể dùng hai cách là load object từ db thông qua hibernate api hoặc save object vào db thông qua API

1. Cách để save một object là :

* session.save() : trả về kiểu dữ liệu object id sau khi gọi, nó thực hiện lệnh insert ngay lập tức không cần biết nó có trong transaction hay không, nếu có thì exception.
* session.saveOrUpdate() : có thể thực hiện insert hoặc update dựa trên sự tồn tại của bản ghi. Nó linh hoạt hơn save() nhưng nó có thêm bước xử lý để tìm ra một bản ghi có tồn tại hay không. Nếu tồn tại thì thực hiện UPDATE nếu không sẽ INSERT
* session.persist() : kiểu trả về là void, cũng thực hiện lệnh insert nhưng nó đảm bảo lệnh insert sẽ không được chạy nếu goi persist **ở ngoài transaction**

1. Cách để load một object từ db:

* session.get(…) làm việc trực tiếp với database, sẽ trả về null nếu không tìm thấy, nó sẽ thực hiện tìm trong cache và db ngay sau khi đc gọi
* session.load(…) tìm trong cache , sẽ trả về ObjectNotFoundException

1. **Detached**:

Khi đóng session, chuyển từ persistent state sang detached, nó không đính kèm với một session nào nhưng vẫn có thể sửa đổi và gắn lại với một session mới. Detached object có các mục tương ứng trong database, đó là persistent và nó không kết nối với Session object. Các object đó có đồng bộ dữ liệu với database khi session đóng. Sau khi đóng thì nếu thay đổi dữ liệu thì nó sẽ không tự động update về db, khi kết nối lại thì dữ liệu sẽ được update. Detached object có thể trở về persistation mà không cần dùng data tranfer object. Detached object có thể sửa đổi ngoài và sau đó có thể gắn lại vào một transaction mới. Có thể dùng session.merge(), session.update, session.saveOrUpdate().

1. Update: nếu chuyển từ detached state về persistent state, dùng để update db đã tồn tại, nếu không tồn tại trả về exception. Update không sử dụng với obj giống nhau trong cùng một session. Nếu obj không tồn tại trong db thì nó trả về exception
2. Merge: cũng giống như update chuyển từ detached về persistent, dùng để update db đã tồn tại, nếu không sẽ thêm một obj mới. Merge có thể sử dụng với obj giống nhau trong cùng một session. Merge có thể lưu tất cả các thay đổi ở bất cứ lúc nào mà không quan tâm tới trạng thái của session
3. Evict: (Persistent state qua detached) xóa một obj ra khỏi session
4. Clear: (p -> d) xóa tất cả các obj ttong session, nó gọi trong transaction. Gọi clear là xóa tất cả obj liên kết tới session
5. Delete(p-d) xóa một detached, xóa khỏi db. Nếu không tồn tại sẽ trả về exception
6. **Remove**:

Khi obj được gọi delete thì obj sẽ chuyển sang trạng thái remove