- Transaction được định nghĩa như bất kì một sự thao tác chương trình của người dùng trong một HQTCSDL

- cần phân biệt với các thao tác của chương trình bên ngoài HQTCSDL

- việc thao tác bên trong HQTCSDL nhiều lần sẽ phát sinh nhiều transaction

- HQTCSDL xử lý transaction như là một chuỗi các lệnh đọc ghi các đối tượng CSDL

- để đọc : đầu tiên, mang các đối tượng CSDL vào trong bộ nhớ. sau đó giá trịn sẽ đc copy vào biến chương trình

- để ghi : một bản copy đối tượng trong bộ nhớ sẽ được chỉnh sửa trước sau đó sẽ được ghi trong bộ nhớ

- 4 tính chất quan trọng :

1. Atomic (nguyên tử): toàn bộ các lệnh trong transaction được thực hiện hoặc không lệnh nào được thực hiện. (nếu đang thực hiện mà hệ thống bị lỗi hay gặp sự cố thì hệ thống sẽ tự hủy những lệnh mà đã thực hiện trước đó)

2. Consistent (nhất quấn) : các ràng buộc của CSDL sẽ được duy trì, đảm bảo tính chất này là trách nhiệm người dùng

3. Isolated (cô lập) : một transaction khi được người dùng thực hiện thì nó thực hiện như thể có 1 tiến trình thực hiện tại một thời điểm, nhưng thực tế thì có thử có nhiều transaction cũng thực hiện tại 1 thời điểm

4. Duarable (bền vững) : khi 1 transaction thực hiện thành công thì kết quả của nó phải được duy trì bảo quản khi hệ thống gặp sự cố.

- các trạng thái :

1. active : trạng thái ban đầu của transaction, sẽ duy trì ở trạng thái này trong quá trình đc thao tác

2. partial commited : Ở trạng thái này khi lệnh cuối cùng được thực hiện

3. Failed : sau khi sự phục hồi những thao tác không thể tiến hành được nữa

4. Aborted : sau khi transaction được romback, CSDL được phục hồi trước trạng thái bắt đầu của trans, sẽ có 2 lựa chọn : lựa chọn lại nếu không có lỗi logic xảy ra, hoặc hai là bị hủy bỏ hoàn toàn

5. Committed : sa khi trans được thực hiện thành công hoàn toàn

- Thao tác :

+ Nhiều trang được cho phép đồng thời trong hệ thống => tận dụng đc không gian đĩa và bộ xử lý, thông lượng của trans sẽ tối hơn

+ Giảm thiểu thời gian đáp ứng trung bình : Các trans ngắn sẽ không cần phải đợi trans dài hơn

- Lượt đồ kiểm soát đồng thời: Kiểm soát sự tươngtacs giữa các trans đồng thời ngăn cản sự phá hủy tính nhất quán của CSDL

- Lượt biểu : là 1 chuỗi các lệnh trong các trans khác nhau được thực hiện theo thứ tự thời gian. Một lượt biểu phải bao gồm tất cả các lệnh trong trans và phải duy trì thứ tự thực hiện

- Các loại lượt biểu :

+ Serial (tuần tự) : thực hiện xong cái này mới tới cái khác

+ Equivalent (tương đương) : bất kì trạng thái CSDL nào thì việc thao tác lượt biểu 1 giông lượt biểu 2

+ Serializable ( khả tuần tự) : là lượt biểu tư đương với 1 lượt biểu tuần tự nào đó. Nếu trans duy trì tính nhất quán thì lượt biểu cũng duy trì tính nhất quán

- Tính khả tuần tự :

+ 2 loại : Khả tuần tự xung đột và khả tuần tự view

+ 2 thao tác : Read () , Write()

* Khả tuần tự xung đột : 2 lệnh xung đột nếu và chỉ nếu 2 lệnh cung truy cập đến 1 đối tượng CSDL và trong 2 lệnh đó có ít nhất 1 lệnh ghi (write )đối tượng CSDL
* 2 lệnh trong 2 trans mà xung đột thì phải duy trì thứ tự thực hiện trong lượt biểu. Nếu không xung đột thì ta có thể thay đổi thứ tự thực hiện mà không làm thay đổi kết quả

- Tương đương xung đột : Nếu lượt biểu S có thể biến đổi thành S’ có thể biến đổi chỗ những lệnh không xung đột trong các trans khác nhau thì là 2 lượt biểu tương đương xung đột.

- Khả tuần tự xung đột : nếu nó tương đương xung đột với một lượt đồ tuần tự

\*\* Tương đương View nếu :

* với mọi đối tượng Q, trans Ti đọc giá trị ban đầu của Q trong lượt biểu S thì trans Ti cũng phải đọc giá trị ban đầu của Q tỏng S’
* Với mỗi Q, nếu Trans Ti thực hiện đọc Q trong S, mà giá trị Q trước đó được ghi bằng Trans Tj (nếu có) thì Ti trong S’ cũng phải đọc Q và giá trị Q trước đó phải được ghi bằng Tj
* Với mỗi Q, nếu trans thực hiện việc ghi giá trị Q cuối cùng nếu có trong S, thì trong S’ trans cũng phải thực hiện ghi giá trị Q cuối cùng

\*\* khả tuần tự view :

* Mỗi lượt biểu khả tuần tự xung đột thì khả tuần tự view, nhưng ngược lại không đúng
* Mỗi lượt biểu hả tuần tự view mà không khả tuần tự xung đột thì sẽ xảy ra tình trạng ghi mù (Blind write : thực hiện ghi nhưng không đọc đối tượng dữ liệu trước đó)

\*\* tính khả phục hồi : xử lý kết quả của những trans thất bài trong khi chạy các trans đồng thời

\*\* phục hồi lan truyền : xảy ra khi một trans thất bại, nó sẽ lan truyền qua những trans khác trong lượt biểu

\*\* mọi lượt biểu không lan truyền thì cũng khả phục hồi

\*\* Lock : 2PL có 2 rules

* Nếu trans T muốn đọc hoặc ghi thì đầu tiên phải yêu cầu shared lock (respectievely, exclusive)
* Tất cả các lock trên trans sẽ được giải phóng khi trans đó hoàn tất
* Chỉ cho phép với các lượt biểu khả tuần tự