



Tóm tắt lý thuyết

Cơ sở dữ liệu (Ho Chi Minh University of Banking)



Scan to open on Studocu

- 1. Cơ sở dữ liệu:** là sự tập hợp có tổ chức các dữ liệu có liên quan luận lý với nhau
 - + Có tổ chức: người sử dụng có thể dễ dàng lưu trữ, thao tác và truy xuất dữ liệu
 - + Có liên quan luận lý với nhau: dữ liệu mô tả một lĩnh vực mà nhóm người sử dụng quan tâm và được dùng để trả lời các câu hỏi liên quan đến lĩnh vực này
- 2. Mô hình dữ liệu:** tập hợp các khái niệm dùng để biểu diễn các cấu trúc của dữ liệu
 - + **Mô hình dữ liệu thể hiện:** mô tả các dữ liệu bằng cách sử dụng các ký hiệu tương ứng với mô hình dữ liệu mà một hệ quản trị cơ sở dữ liệu sử dụng
 - + **Mô hình dữ liệu khái niệm:** cung cấp những khái niệm gần gũi với người sử dụng, phương tiện để giúp cho người phân tích, thiết kế, giao tiếp với người sử dụng, thu nhận thông tin xác định được tính đúng đắn và đầy đủ những yêu cầu của hệ thống, sử dụng cấu trúc mô hình: thực thể, đặc tính, liên kết
 - + **Mô hình dữ liệu vật lý:** cung cấp những khái niệm để biểu diễn chi tiết cách thức những dữ liệu được lưu trữ trong máy tính, chỉ ra dữ liệu dưới dạng băng ghi, thứ tự sắp xếp các băng ghi, đường dẫn truy cập (2 loại: off và onl)
- 3. Các nhóm người dùng cơ sở dữ liệu:** chia thành 5 nhóm
 - + **Nhóm quản trị cơ sở dữ liệu:** Quản trị viên CSDL là nhóm người phân quyền mạnh nhất. Có quyền thực hiện các hành động: phân quyền truy cập CSDL, giám sát quá trình vận hành csdl, chăm sóc kỹ việc sao lưu, phục hồi CSDL khi có điều gì xảy ra, đảm bảo được sự phân bổ tài nguyên phần cứng, phần mềm, xử lý sự cố
 - + **Nhóm thiết kế cơ sở dữ liệu:** lựa chọn những cấu trúc biểu diễn và cách thức lưu trữ dữ liệu, làm việc với người dùng CSDL để biết yêu cầu của họ nhằm thiết kế cho phù hợp (đặc tả yêu cầu của người sử dụng), liên hệ với từng nhóm người dùng để xây dựng được khung nhìn dữ liệu phù hợp
 - + **Nhóm người dùng cuối:** Là nhóm có nhu cầu truy cập CSDL để thực hiện các thao tác khai thác hay quản lý nội dung CSDL (truy xuất dữ liệu theo nhu cầu hoặc cập nhật dữ liệu theo hình thức trực tuyến hoặc phiên
 - + **Nhân viên phát triển:** đặc tả yêu cầu của người sử dụng (phân tích sâu thêm)
 - + **Nhóm nhân viên bảo trì:** đảm bảo làm sao cho hệ thống làm việc và xuyên suốt
- 4. Mô hình thực thể kết hợp:** là mô hình xây dựng dựa trên sự đặc tả thế giới thực bao gồm một tập các đối tượng cơ sở có tên gọi là tập thực thể và một tập các liên kết giữa các đối tượng này với nhau. Mô hình còn được gọi là mô hình ý niệm (khái niệm) vì được xây dựng trên nền tảng các khái niệm trong tổ chức và dùng để biểu diễn cấu trúc tổng thể của tổ chức
- 5. Thực thể:** những đối tượng dữ liệu cơ bản chứa các thông tin cần thu thập và không thể chia cắt
- 6. Thuộc tính:** những đặc tính (đặc trưng) của một thực thể. Mỗi thực thể tại từng thuộc tính có một giá trị tương ứng. Giá trị của thuộc tính giúp phân biệt các thực thể
- 7. Miền giá trị:** tập hợp các giá trị mà thuộc tính có thể nhận được
- 8. Thuộc tính khóa:** một thuộc tính dùng để phân biệt các thực thể khác nhau trong một tập thực thể
- 9. Thuộc tính tổ hợp:** thuộc tính được cấu thành từ các thuộc tính khác
- 10. Thuộc tính đơn trị:** thuộc tính chỉ có thể nhận được một giá trị duy nhất cho một thực thể cụ thể
- 11. Thuộc tính lưu trữ:** thuộc tính là giá trị của nó phải được nhập và csdl & thuộc tính dẫn xuất: là thuộc tính phụ thuộc vào thuộc tính khác
- 12. Giá trị Null:** Là giá trị không biết, nó tồn tại nhưng không xác định được, hoặc thuộc tính tương ứng không áp dụng cho thực thể đang xem xét
- 13. Tập thực thể:** tập hợp các thực thể của một kiểu thực thể tại một thời gian hoặc không gian cụ thể
- 14. Kiểu thực thể:** tập hợp các thực thể có những thuộc tính tương tự nhau
- 15. Liên kết:** sự kết hợp giữa các thực thể từ một hoặc nhiều tập thực thể khác nhau, phản ánh các mối liên quan về mặt ý nghĩa của cá dữ liệu
- 16. Liên kết 2 ngôi được chia thành 3 kiểu**
 - + **Liên kết một - một:** dạng đơn giản nhất. Diễn tả mqh giữa 1 thực thể trong tập thực thể này với nhiều nhất một thực thể trong tập thực thể kia và ngược lại
 - + **Liên kết một - nhiều:** một thực thể trong tập thực thể A kết hợp với nhiều thực thể trong tập thực thể B, nhưng mỗi thực thể trong tập thực thể B chỉ kết hợp với nhiều nhất một thực thể trong tập thực thể A. Loại này gọi là liên kết nhiều - một từ A đến B
 - + **Liên kết nhiều nhiều:** loại liên kết không có sự giới hạn về số lượng thực thể trong mỗi tập thực thể tham gia
- 17. Tập thực thể yếu:** Tập thực thể A phụ thuộc chặt chẽ vào tập thực thể B trong qua kiểu liên kết R. Diễn tả: mỗi thực thể b trong B kết hợp với nhiều thực thể a trong A theo các liên kết r trong R. Khi thực thể b bị xóa ra khỏi B, tất cả các thực thể a và các liên kết r nói trên đều lần lượt bị xóa ra khỏi A và R. Khi đó
 - + A là tập thực thể yếu
 - + R là kiểu liên kết hỗ trợ
 - + B là tập thực thể hỗ trợ
 - Điều kiện để A là tập thực thể yếu và R là kiểu liên kết hỗ trợ
 - + R là loại liên kết hai ngôi nhiều một từ A đến B
 - + Tập thực thể A không có khóa, nó phải nhận khóa từ tập thực thể B về làm khóa của nó

18. Liên kết cha/con: phép chia một tập thực thể thành nhiều tập thực thể khác, nhỏ hơn. Tập thực thể bị chia = lớp cha, tập thực thể được chia = lớp con

19. Chuyên biệt hóa: Là quá trình xác định các lớp con của một tập thực thể (lớp cha). Quá trình phân chia lớp cha thành nhiều lớp con

20. Khái quát hóa: Là quá trình xây dựng một tập thực thể mới dựa trên kết quả phân tích điểm giống và khác nhau của một số tập thực thể. Quá trình xây dựng lớp cha từ các lớp con phân biệt

21. Ràng buộc trong chuyên biệt hóa và khái quát hóa

- **Ràng buộc không trùng lặp** (kí hiệu là d) trong phép chuyên biệt hóa. Có nghĩa là một thực thể trong lớp cha không thể xuất hiện trong nhiều hơn một lớp con

- **Ràng buộc trùng lặp** (kí hiệu là o) trong phép chuyên biệt hóa. Có nghĩa là trong lớp cha có thể xuất hiện trong nhiều lớp con

- **Ràng buộc đầy đủ:** mọi thực thể của lớp cha phải là thành viên của ít nhất một lớp con. Giúp phân biệt phép chuyên biệt hóa là toàn phần hay một phần

- **Ràng buộc từng phần** cho biết một thực thể của lớp cha có thể không nằm trong bất kì lớp con nào cả

22. Cây phân cấp và lưới phân cấp: Một lớp cha có thể có nhiều lớp con, và một lớp con cũng có thể có nhiều lớp cha. Từ đó có thể tạo ra cây phân cấp (đơn thừa kế) hay lưới phân cấp (đa thừa kế)

23. Mô hình dữ liệu quan hệ: Mô hình quan hệ biểu diễn cơ sở dữ liệu như là tập hợp các quan hệ, mỗi quan hệ có thể được coi là một bảng dữ liệu hai chiều gồm nhiều cột và nhiều hàng. Mỗi hàng trong bảng (trừ hàng tiêu đề) biểu diễn một sự vật tương ứng với một thực thể hoặc một liên kết trong thế giới thực. Tên bảng và tên các cột (tức hàng tiêu đề) được dùng để diễn giải ý nghĩa của các giá trị tại mỗi hàng. Tất cả giá trị trong cùng một cột (ngoại trừ tiêu đề cột) có cùng kiểu dữ liệu

24. Lược đồ quan hệ: Kí hiệu: $R(A_1, A_2, \dots, A_n)$ với R là tên quan hệ và A_i là các thuộc tính

25. Bộ: các hàng trong quan hệ

26. Thể hiện quan hệ: Bao gồm một số hữu hạn các bộ biểu diễn trạng thái cụ thể của thế giới thực. Thể hiện của quan hệ luôn biến đổi

27. Các ràng buộc toàn vẹn trên mô hình quan hệ: Là những điều kiện mà thể hiện csdl bắt buộc phải thỏa mãn tại mọi thời điểm

- **Ràng buộc miền giá trị:** Yêu cầu trong mỗi bộ, giá trị ứng với mỗi thuộc tính A phải là đơn trị và thuộc miền giá trị dom(A)

- **Ràng buộc khóa:** hai bộ bất kỳ của quan hệ không có giá trị bằng nhau tại thuộc tính khóa. Nếu K là khóa của quan hệ, điều kiện cần:

+ K là siêu khóa

+ K là siêu khóa nhỏ nhất

- **Ràng buộc giá trị null:** Là ràng buộc cho phép hay không cho phép thuộc tính có giá trị Null. Khi thuộc tính A được thiết lập là not null thì các bộ dữ liệu không được nhận giá trị null tại thuộc tính này

- **Ràng buộc thực thể:** Nếu thuộc tính K được chọn làm khóa của quan hệ R thì sẽ tồn tại đồng thời ràng buộc giá trị Null

- **Ràng buộc toàn vẹn tham chiếu:** một giá trị xuất hiện trong cột A của quan hệ R thì phải xuất hiện trong cột B của quan hệ S . - Xác định giữa hai quan hệ với ý nghĩa một giá trị muốn xuất hiện trong quan hệ này phải xuất hiện trong một quan hệ khác có liên quan. Nếu một bộ của quan hệ R muốn tham chiếu tới quan hệ S , thì phải tham chiếu tới một bộ hiện có sẵn của quan hệ S . Để tồn tại ràng buộc toàn vẹn tham chiếu từ tập thuộc tính A của R tới tập thuộc tính B của S , các điều kiện cần thỏa mãn:

+ Thuộc tính trong A có cùng miền giá trị với thuộc tính trong B tương ứng

+ Thuộc tính trong B phải là khóa chính hoặc khóa phụ của S

+ Giá trị tại A của một bộ trong R hoặc là null hoặc là giá trị tại B của một bộ nào đó trong S

- **Ràng buộc toàn vẹn ngữ nghĩa:** Đảm bảo thể hiện csdl tại mọi thời điểm phản ánh đúng tình trạng thế giới thực

- **Ràng buộc phụ thuộc hàm:** Dựa trên mối quan hệ giữa hai tập thuộc tính X và Y trong cùng một quan hệ R . Phát biểu: Nếu hai bộ bất kỳ của quan hệ R có giá trị bằng nhau tại các thuộc tính X thì cũng sẽ có giá trị bằng nhau tại các thuộc tính Y . Kí hiệu $X \rightarrow Y$

28. Phụ thuộc hàm: phụ thuộc hàm f trên quan hệ R giữa hai tập hợp các thuộc tính A_1, A_2, \dots, A_n và B_1, B_2, \dots, B_m được phát biểu như sau: Nếu hai bộ của quan hệ R có giá trị giống nhau tại các thuộc tính A_1, A_2, \dots, A_n thì chúng cũng phải có giá trị giống nhau tại các thuộc tính B_1, B_2, \dots, B_m . Kí hiệu $A \rightarrow B$

29. Khóa của quan hệ: tập hợp của một hoặc nhiều thuộc tính là khóa của quan hệ khi thỏa mãn đồng thời 2 đk:

+ Các thuộc tính đó xác định tất cả các thuộc tính khác của quan hệ

+ $\{A_1, A_2, \dots, A_n\}$ không có bất kỳ tập con khác rỗng nào có thể xác định tất cả các thuộc tính khác của qhe. Khóa phải là nhỏ nhất

30. Siêu khóa: tập hợp các thuộc tính chứa khóa. Mọi khóa đều là siêu khóa, sk không phải là khóa

31. Bao đóng: cho F là một tập phụ thuộc hàm. Bao đóng của F kí hiệu là F^+ là tập lớn nhất chứa các phụ thuộc hàm có thể suy ra từ các phụ thuộc hàm trong F

32. Bao đóng các thuộc tính: Giả sử X là tập hợp các thuộc tính, F là tập hợp các phụ thuộc hàm. Bao đóng của X theo F là tập hợp các thuộc tính B sao cho mọi quan hệ thỏa mãn F cũng thỏa mãn $X \rightarrow B$