**CÂU HỎI AUDIT MODULE 3**

**DATABASE**

STT Nội dung câu hỏi

1 Phân biệt các câu lệnh DML & DDL? DML, DDL bao gồm các câu lệnh nào?

-

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | C | R | U | D |
| DDL(Data Defination Language) | Create | X | Alter | Drop |
| DML (Data Manipulation Language – Ngôn ngữ thao tác dữ liệu) | Insert | Select | Update | Delete |

2 Cơ sở dữ liệu (database) là gì?

- Là thành phần dùng để lưu trữ cơ sở dữ liệu

3 Trình bày về mô hình thực thể mối quan hệ (ERD - Entity-Relationship Diagram)? Trong mô hình thực thể mối quan hệ có những thành phần nào?

- Mục đích của ERD : mô hình hóa nhu cầu của một khách hàng thành sơ đồ => từ sơ đồ chuyển sang DB

- Các thành phần phần của ERD :

* Entity : các thực thể
* Relationship : mối quan hệ giữa các thực thể
* Thuộc tính của các thực thể
* Các thuộc tính đa trị

4 Các bước chuyển đổi từ ERD -> Table?

* Chuyển các thuộc tính khóa của thực thể thành khóa chính của table ;
* Biến đổi các thuộc tính đa trị :
  + Tách các thuộc tính đa trị sang 1 table khác, đặt khóa ngoại ở table hiện tại
* Biến đổi mối quan hệ 1-1 :
  + Đặt thuộc tính khóa ở phía tùy chọn sang phía bắt buộc
* Biến đổi mối quan hệ 1 – n :
  + Đặt thuộc tính khóa của phía 1 làm khóa ngoại của phía nhiều. Khóa ngoại đó tham chiếu đến thuộc tính khóa của phía 1
* Biến đổi mối quan hệ n-n :
  + Sinh ra 1 table mới có khóa chính là tập hợp khóa chính của 2 thực thể tham gia vào mối quan hệ

5 Các bước chuẩn hóa dữ liệu?

* 1.Loại bỏ các thuộc tính đa trị, các thuộc tính được tính toán từ các thuộc tính khác
* 2. Đạt chuẩn 1. Loại bỏ các phụ thuộc bộ phận tức là các thuộc tính không khóa phụ thuộc vào 1 bộ phận nào đó của tổ hợp khóa Primary Key
* 3. Đạt chuẩn 2. Loại bỏ các thuộc tính phụ thuộc bắc cầu tức là loại bỏ thuộc tính mà thuộc tính đó phụ thuộc vào 1 thuộc tính không khóa, thuộc tính không khóa đó lại phụ thuộc vào 1 khóa chính.

6 Ràng buộc (constraint) là gì? Các loại ràng buộc trong MySQL?

* Constraint là những quy tắc được áp dụng trên các cột dữ liệu, trên bảng. Được sử dụng để kiểm tra tính hợp lệ của dữ liệu vào, đảm bảo tính chính xác, tính toàn vẹn của dữ liệu.

|  |  |
| --- | --- |
| Loại ràng buộc | Ý nghĩa |
| Not null | Dữ liệu của cột không được nhận giá trị null |
| Default | Gán giá trị mặc định cho dữ liệu trong trường hợp dữ liệu không được nhập hoặc không xác định |
| Unique | Sử dụng để đảm bảo dữ liệu của cột là duy nhất, không bị trùng lặp giá trị trên cùng 1 cột |
| Foreign key (Khóa ngoại) | Dùng để thiết lập khóa ngoại của bảng, tham chiếu đến bảng khác thông qua giá trị được liên kết. Giá trị của cột được liên kết phải là duy nhất trong bảng kia |
| Primary key  (Khóa chính) | Dùng để thiết lập khóa chính trên bảng, xác định giá trị trên tập các cột làm khóa chính phải duy nhất, không được trùng lặp, đồng thời phải is not null |
| Check | Đảm bảo tất cả giá trị trong cột phải thỏa mãn điều kiện nào đó. Đây là hình thức sử dụng phổ biến để kiểm tra tính hợp lệ của dữ liệu. |

7 Khóa chính (Primary key) là gì? “Khóa chính chứa một hoặc nhiều thuộc tính khóa” nghĩa là gì?

* Primary key: Dùng để thiết lập khóa chính trên bảng, xác định giá trị trên tập các cột làm khóa chính phải duy nhất, không được trùng lặp, đồng thời phải is not null
* Khóa chính chứa một hoặc nhiều thuộc tính khóa là một tổ hợp khóa chính, thường sử dụng trong mối quan hệ n-n

8 Khóa ngoại (foreign key) là gì?:

* Khóa ngoại (foreign key) dùng để thiết lập khóa ngoại của bảng, tham chiếu đến bảng khác thông qua giá trị được liên kết. Giá trị của cột được liên kết phải là duy nhất trong bảng kia

9 Phân biệt DELETE và TRUNCATE?

* Delete và Truncate đều dùng để xóa dữ liệu

|  |  |
| --- | --- |
| DELETE | TRUNCATE |
| * Dùng được cho từng record * Dữ liệu xóa bởi delete có thể được restore lại | * Không dùng được cho từng record * Khi xóa cả bảng, truncate sẽ xóa hết dữ liệu của bảng và reset transaction log, nên khi thêm lại record mới giá trị của id sẽ bắt đầu từ 1 * Không xóa được bảng mà có khóa ngoại tham chiếu đến bảng khác |

10 Làm sao để truy vấn dữ liệu từ nhiều bảng?

* Dùng câu lệnh join. Join được chia làm 2 loại:
  + Inner Join: Chỉ lấy phần chung của 2 bảng
  + Outler Join: Lấy phần chung và phần riêng của mỗi bảng
    - Left join: Lấy phần riêng của bảng bên trái
    - Right join: Lấy phần riêng của bảng bên phải
    - Full join: Lấy phần riêng của cả 2 bảng
* Dùng Union, Union All

11 Các loại JOIN? Phân biệt?

* Inner Join: Chỉ lấy phần chung của 2 bảng
* Outler Join: Lấy phần chung và phần riêng của mỗi bảng
  + Left join: Lấy phần riêng của bảng bên trái
  + Right join: Lấy phần riêng của bảng bên phải
  + Full join: Lấy phần riêng của cả 2 bảng

12 Phân biệt AND và OR?

* AND: Biểu thức trả về true khi các điều kiện trong biểu thức trả về true
* OR: Biểu thức trả về true khi một trong các điều kiện trong biểu thức trả về true

13 Phân biệt GROUP BY và ORDER BY?

|  |  |
| --- | --- |
| GROUP BY | ORDER BY |
| * Dùng để sắp xếp các record theo nhóm, các nhóm được xác định bởi các cột được chỉ định, thường được ssử dung cùng max, min,count | * Mệnh đề ORDER BY trong SQL được sử dụng để sắp xếp dữ liệu theo thứ tự tăng dần hoặc giảm dần, dựa trên một hoặc nhiều cột. |

14 Muốn nối chuỗi trong MySQL thì sử dụng gì?

* Dùng câu lệnh CONCAT(string1, string2, ...., string\_n)

15 Yêu cầu tính điểm trung bình của tất cả các sinh viên thì làm thế nào?

Dùng AVG

16 Yêu cầu tính điểm trung bình của từng bạn sinh viên thì làm thế nào?

17 Yêu cầu hiển thị tên sinh viên và điểm trung bình tương ứng, chỉ hiển thị các bạn có điểm trung bình lớn hơn 5 thì làm thế nào?

18 Hiển thị danh sách sinh viên và lớp học tương ứng thì làm thế nào (hai bảng Student và Class có cột chung ClassID)?

19 Hiển thị danh sách sinh viên theo thứ tự tăng dần của điểm thi?

20 Hiển thị danh sách sinh viên theo thứ tự giảm dần của họ tên, nếu họ tên trùng nhau thì sắp xếp theo tăng dần của tuổi?

21 Hiển thị các sinh viên có họ là ‘Le’ và tên là ‘Hai’?

*SELECT \**

*FROM student*

*WHERE (SUBSTRING\_INDEX(employee\_name, ' ', -1) like 'Hai'*

*And SUBSTRING\_INDEX(employee\_name, ' ', 1) like 'Le')*

22 Hiển thị các sinh viên có tên bắt đầu là ‘C’?

*SELECT \**

*FROM student*

*WHERE (SUBSTRING\_INDEX(employee\_name, ' ', -1) like 'C%’*

23 Phân biệt JOIN và UNION?

|  |  |
| --- | --- |
| JOIN | UNION |
| Kết hợp các bảng có dữ liệu liên quan đến nhau | hợp dữ liệu của 2 bảng mà có các cột loại data giống nhau |

24 Phân biệt UNION và UNION ALL?

- Đều là kết hộ của nhiều câu trut vấn nhưng Union thì loại bỏ trùng lặp còn Union all thì giữ lại

25 Phân biệt EXIST và IN?

- Đều dùng trong câu lệnh truy vấn con

|  |  |
| --- | --- |
| IN | EXITS |
| Ở where cần ghi tên column | Không cần ghi tên column vì kiểm tra cho toàn bị record |
| Không chấp nhận null  Tốc độ xử lý chậm khi record lớn | Chấp nhận null  Nhanh hơn IN |

26 Làm sao để có thể tăng tốc độ truy vấn?

Dùng Index đánh dấu chỉ mục của 1 table

27 Các dạng tham số ở SP?

* Dạng In: truyền giá trị vào tham số thì giá trị đối số bên ngoài sẽ k bị tác động~tham trị
* Dạng in - out: truyền giá trị vào tham số thì giá trị đối số bên ngoài sẽ bị tác động~tham chiếu
* Dạng out: đi vào, chỉ tham gia lấy giá trị rồi đi ra

28 Phân biệt SP, Trigger, Function?

|  |  |  |  |
| --- | --- | --- | --- |
|  | SP | Trigger | Function |
| So sánh với Java | ~ void | X | ~return |
| Tham số: | 3 dạng in, in-out, out | X | Không có |
| Kiểu trả về | X | X | có |
| Cách sử dụng | Dùng call | Tự động chạy | Gọi ở querry |