|  |  |  |
| --- | --- | --- |
| 1 | Cơ sở dữ liệu (database) là gì? | Dữ liệu của 1 trang web |
| 2 | Phân biệt các câu lệnh DML & DDL? DML, DDL bao gồm các câu lệnh nào? | DDL = Data Definition Language : định nghĩa data  DML = Data Manipulation Language: thao tác  DCL = Data Control Language = Phân quyền  TCL = Transaction Control Language: thao tác với transaction |
| 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? | ERD - Entity-Relationship Diagram: Sơ đồ thực thể và mối quan hệ giữua chúng |
| 4 | Các bước chuyển đổi từ ERD -> Table? | 5 Bước:   1. Biến đổi các entity -> table, thuộc tính thành column, thuộc tính khóa thành khóa chính 2. Biến đổi các thuộc tính đa trị: Tách ra 1 table 1 3. Biến đổi mối quan hệ 1-1 ; Đặt khóa chính từ tùy chọn sang bắt buộc 4. Biến đổi mối quan hệ 1-n: Đặt khóa chính từ phía 1 sang phía nhiều 5. Biến đổi mối quan hệ n-n: Sinh ra 1 table mới có khóa chính phức hợp là tập hợp 2 khóa chính của các table tham gia vào mối quan hệ |
| 5 | Các bước chuẩn hóa dữ liệu? | 3 Bước:  1F: Loại bỏ các thuộc tính đa trị và các thuộc tính có giá trị được tính toán từ các thuộc tính của bảng khác  2F: loại bỏ các phụ thuộc bộ phận( là loại bỏ các thuộc tính không khóa phụ thuộc vào  1 bộ phận nào đó của khóa chính - khóa phức hợp)  3F:loại bỏ các thuộc tính bắc cầu |
| 6 | Ràng buộc (constraint) là gì? Các loại ràng buộc trong MySQL? | Ràng buộc là các quy tắc xây dựng database giúp dữ liệu được chính xác tin cậy cao. 7 ràng buộc.   1. Not null 2. Unique 3. Khóa Chính 4. Khóa Ngoại 5. Check 6. Defalut 7. Index |
| 7 | Khóa chính (Primary key) là gì? “Khóa chính chứa một hoặc nhiều thuộc tính khóa” nghĩa là gì? | Khóa chính dùng để xác định 1 record duy nhất trong 1 table. Khóa nhiều cột là khóa phức hợp |
| 8 | Khóa ngoại (foreign key) là gì? | Khóa ngoại là cơ chế tạo liên kết giữa 2 table trong database. |
| 9 | Phân biệt DELETE và TRUNCATE? | Delete : DML, Dùng với where, sau khi xóa , nếu them mới id sẽ k reset lại(ex new id:5)  Truncate: DDL, Không Where, , sau khi xóa , nếu them mới id sẽ reset lại(ex new id:1) |
| 10 | Các loại JOIN? Phân biệt? | 2 loại join:  Inner join : lấy các record chung giữa 2 bảng dựa trên liên kết khóa chính và khóa ngoại  Outer join: lấy các record chung và riêng giữa 2 bảng dựa trên liên kết khóa chính và khóa ngoại |
| 11 | Phân biệt JOIN và UNION? | Join: Nối column, qua liên kết khóa ngoại  Union: nối các record có số lượng column và kiểu dữ liệu tương ứng giống nhau |
| 12 | Phân biệt UNION và UNION ALL? | Union: record chung lấy 1  Union All: Lấy tất cả |
| 13 | Phân biệt EXIST và IN? | Exist: Sử dụng trong câu sub Query sẽ quét cho đến khi thỏa mãn điều kiện - > nên dùng với data lớn  In: Quét toàn bộ dữ liệu -> nên dùng với data nhỏ |
| 14 | Phân biệt AND và OR? | And là tất cả điều kiện phải đúng thì đúng  Or 1 điều kiện đúng thì đúng |
| 15 | Phân biệt GROUP BY và ORDER BY? | Group by dùng để nhóm các record lại với nhau theo điều kiện chung giá trị column |
| 16 | Muốn nối chuỗi trong MySQL thì sử dụng gì? | Sử dụng hàm concat() |
| 17 | Yêu cầu tính điểm trung bình của tất cả các sinh viên thì làm thế nào? | Dùng Hàm Gộp AVG cho colum điểm trung bình. |
| 18 | Yêu cầu tính điểm trung bình của từng bạn sinh viên thì làm thế nào? | Dùng Hàm Gộp AVG cho colum điểm trung bình. Và sử dụng thêm group by nếu muốn tính điểm trung bình từng sinh viên |
| 19 | 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? | Select name sinh viên và AVG  Having AVG > 5 |
| 20 | Hiển thị danh sách sinh viên và lớp học tương ứng thì làm thế nào (hai bảng Student và Class có cột chung ClassID)? | Join 2 bảng Student và class lại với nhau và select danh sách sinh viên và class tương ứng |
| 21 | Hiển thị danh sách sinh viên theo thứ tự tăng dần của điểm thi? | Order by column Điểm |
| 22 | 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? | Order by comlumn họ tên DESC và thêm tiêu chí sắp xếp theo tuổi bằng cách thêm dấu “,” Order by tuổi |
| 23 | Hiển thị các sinh viên có họ là ‘Le’ và tên là ‘Hai’? | Select name  Form sinhvien  Where name like Le% and %Hai |
| 24 | Hiển thị các sinh viên có tên bắt đầu là ‘C’? | Select name  Form sinhvien  Where name like C% |
| 25 | Làm sao để có thể tăng tốc độ truy vấn? Nhược điểm là gì? | Sử dụng index  Nhược điểm; tốn tài nguyên, CRUD chậm -> nên đặt ở column truy xuất nhiều, sd cho data lớn và nên đặt ở column ít CURD |
| 26 | Các dạng tham số ở SP? | In, Out , inout |
| 27 | Phân biệt SP, Trigger, Function? | Đều tập hợp các câu lệnh query để thực hiện 1 nv  Sp: input(in,out,inout) output: (không có), gọi = (call)  Trigger: input(Không có), output(không có), chạy ngầm khi CRUD tại bảng  Funtion: input (in) output(có trả về), gọi bằng select |
| 28 | CSS là gì? Các cách nhúng CSS vào tài liệu html? Độ ưu tiên? | Cascading Style Sheet: Ngôn ngữ định dạng ra 1 trang web, 3 cách:  Inline:”style”  Interline:<style>  Exterline:file.css  Độ ưu tiên. Phụ thuộc vào khoảng cách tới phần tử HTML(gần nhất) |
| 29 | Phân biệt giữa bộ chọn class và bộ chọn id? | Bộ chọn class: sử dụng cho nhiều phần tử HTML  Bộ chọn id: Sử dụng cho 1 phần tử HTML duy nhất và bộ chọn id không thể nhúng bằng inline |
| 30 | Trình bày mô hình hộp (box model) trong CSS? | Là các phần tử HTML gồm 4 thành phần : Margin, Border, Padding, Content |
| 31 | Trình bày thuộc tính float, clear? | Float để kéo các phần tử HTML về 1 phía  Clear để xóa Float |
| 32 | Thuộc tính position dùng để làm gì? Phân biệt các giá trị: static, absolute, relative và fixed? | Định vị vị trí hiển thị của phần tử HTML trong trang web  Static: vị trí mặc định không bị ảnh hưởng bởi các thuộc tính như : left, rignt, top, down..  Absolute: Vị trí hiển thị sắp xếp dựa trên phần tử cha  Relative: Vị trí hiển thị được sắp xếp so với vị trí ban đầu  Fixed: Vị trí cố định so với màn hình  Sticky: Vị trí cố định so với màn hình sẽ bị ảnh hưởng khi Scroll |
| 33 | Responsive Web Design là gì? | Là 1 kỹ thuật cho phép trang web hiển thị trên nhiều loại thiết bị |
| 34 | Làm thế nào để tạo 1 trang web responsive? | 3 yếu tố  Use @media để xét điều kiện  Use Grid View  Use float và clear |
| 35 | Bootstrap là gì? | Là 1 famework làm web phổ biến giúp websize hiển thị trên nhiều thiết bị khác nhau |
| 36 | Một số thành phần thông dụng ở Bootstrap? | Alert ,Card,Carousel,Modal,Narbar, Skiner,Button,DropDown |
| 37 | Phân biệt Static web và Dynamic web? | Static web: web tĩnh tạo nên dựa trên cách ngôn ngữ như HTML, JS, CSS và chứa các dữ liệu cứng  Dynamic web: web động tạo nên từ Font End và Back End và dữ liệu thay đổi theo thời gian dựa trên tương tác của Client |
| 38 | Phân biệt 2 phương thức GET/POST | Đều là method gửi request từ Client -> Server  Get: hiển thị trên URl -> Không Bảo Mật ->chỉ gửi được kiểu dữ liệu character-> nên dữ liệu gửi bị hạn chế(2048) tùy trang web  Post: Không hiển thị trên UrL-> gửi qua Request body -> Bảo Mật -> gửi được nhiều kiểu dữ liệu-> Không hạn chế dung lượng |
| 39 | Phân biệt request/response | Request và Response Là các yêu cầu và phản hồi trong mô hình Client và Server  Request : Là yêu cầu từ Client -> Server  Response:Phản hồi từ Server -> Client |
| 40 | Phân biệt forward/redirect | Đều là cách thức để chuyển từ trang này sang trang khác  Forward : Chuyển tiếp Tốc độ nhanh vì ko phải xóa request  Redirect: Chuyển hướng Tốc đồ chậm vì phải xóa request cũ và tạo request mới |
| 41 | JSP là gì? Nói: “Về bản chất JSP cũng chính là Servlet” đúng hay sai? Giải thích? So sánh giống và khác nhau giữa JSP và Servlet ? | JSP = Java Server Pages Là ngôn ngữ được viết dựa vào HTML và nhúng được Java.  Servlet Là ngôn ngữ được viết dựa vào Java và nhúng được HTML  Về bản chất JPS cũng chính là Servlet vì khi JSP biên dịch thì JPS sẽ biên dịch qua Servlet và ngược lại  Khác nhau:  JSP: Tự động biên dịch và triển khai khi có thay đổi, Toàn Bộ các trang JSP được thông dịch qua Servet (1 lần) và JSP mạnh về hiển thị  Servlet: Không tự động biên dịch và chỉ triển khai khi có yêu cầu từ Client và Servlet mạnh về xử lý điều phối giữa Model và View |
| 42 | JSTL là gì? Giới thiệu một số thẻ bạn đã sử dụng ? | JSTL = JSP Standard Tag Library : Là bộ bộ thử viện tag chuẩn dùng để hỗ trợ JSP trong việc hiển thị dữ liệu  1 Số thẻ thông dụng: c:out, c:foreach, c:if, c:choose, c:when,c:otherwise |
| 43 | Trình bày kiến trúc MVC? Vai trò của các thành phần trong kiến trúc MVC? | MVC : Model -View -Controller  Model: xử lý nghiệp vụ logic và tương tác với database  View: Xử lý phần hiển thị giao diện tới Client  Controller: Nơi nhận request , trả về Response và điều hướng giữa Model và View |
| 44 | Sự khác nhau giữa MVC1 và MVC2 | MVC1:View Và Controller đều là JSP-> khi dự án phình to-> phức tạp và khó quản lí  MVC2:Phù hợp với các dự án lớn -> dễ QL và tái sd mã nguồn, Nhưng nhược điểm phụ thuộc khá nhiều vào Controller -> khi thay đổi Controller View và Model sẽ phải thay đổi theo |
| 45 | Mô tả cách thực hiện tạo và lấy kết quả 1 câu query bất kỳ bằng JDBC? | JDBC= Java Data Base Connectivity là 1 API dùng kết kết nối thực thi 1 câu lệnh từ Java -> DB  B1:Tạo 1 connection cổng kết nối Từ Java -> Database  B2: Tạo 1 biến từ 3 Class statement, prarestatement hoặc callablestatement để thực hiện câu truy vấn xuống DB  B3: Tạo 1 biến từ Class ResultSet để nhận kết quả sau khi truy vấn từ DB |
| 46 | Phân biệt các loại Statement ? | Statement: Câu truy vấn tĩnh(tức là câu truy vấn sẽ không thay đổi khi truy vấn)-> Dễ bị tấn công bởi SQL injection  Prarestatement: Câu truy vấn động(câu truy vấn sẽ thay đổi qua tham số mình truyền vào khi truy vấn)-> không bị tấn công bởi khi them tham số truyền vào sẽ thêm 1 dâu “”  Callablestatement: Câu truy vấn động và use đ-> để gọi SP |
| 47 | Kỹ thuật sử dụng transaction bằng JDBC là như thế nào? | Sửa chế độ Autocommit = false  Khi Truy vấn hoặc update thành công thì Commit  Ngược lại thì tiến hành Rollback |
| 48 | Các tính chất của Transaction? | Transaction gồm nhiều phép thực thi theo quy luật “All or nothing”  A:Atomicity: Commit or RollBack  C:Consistency: Đảm đảo dữ liệu nhất quán và tuân thủ các ràng buộc ở DB  I:Isolation:-> Các Transaction độc lập với nhau  D:Durability: 1 khi đã commit-> data cố định |
| 49 | Phương pháp gọi 1 SP ở JDBC? | Tạo SP ở DB  Sử dụng callablestatement để gọi |
| 50 | Phân biệt Ant, Maven, Gradle? | 3 công cụ dùng để quản lý thư viện phụ thuộc trong project  Ant:Quản lý thủ công, Phải add từng bước  Maven:QL file pom.xml gồm các thẻ XML  Gradle: QL file build.gradle, Use Grooby hỗ trợ được nhiều nn hơn so với Maven |