**CSDL – SQL**

1. **Giới thiệu :**

**SQL** : Là ngôn ngữ chuẩn để **truy nhập** vào **thao tác** vơi cơ sở dữ liệu.

1. **Khái niệm cơ bản về cơ sở dữ liệu :**

**Định nghĩa : hàng, cột , khóa .**

**Khóa :** Thông tin duy nhất, khác nhau, tự động tăng.

**Kiểu dữ liệu :**

**INT** : Số nguyên, kích thước chuẩn.

**FLOAT** : Số có dấu chấm động.

**VARCHAR** : Text có chiều dài xác định.

**TEXT** : Dài hơn varchar.

**DATETIME** : Ngày tháng có thời gian, định đạng YYYY : MM : DD : HH : MM : SS.

**DATE** : Ngày tháng, định dạng YYY : MM : DD.

**TIMESTAMP** : Ngày tháng có thời gian, định dạng YYYY : MM : DD : HH : MM : SS. Lưu dưới dạng **text.**

**Kiểu table trong MySql :**

**Myisam :** tối hóa tốc độ nén , kích thước tối đa **256 TB** , thao tác với bảng ít liên tục.

**Innodb :** kích thước **64 TB** , hỗ trợ khóa **ngoại**, **Tối ưu** hóa cho trao đổi dữ liệu.

1. **Các câu lệnh cơ bản**

**Lệnh tạo cơ sở dữ liệu :**

**Create database tên :** Tạo bảng .

**Use tên :** Sử dụng cở sở dữ liệu.

**Drop database tên :** Xóa bảng .

**Create database if not exists tên :** Kiểm tra xem có tồn tại bảng hay không.

**Lệnh tạo bảng :**

**Create table < tên bảng > ( < tên trường > kiểu( kích thước ) primary key auto\_increment ) engine ‘< kiểu bảng >’ :** Trong bảng dữ liệu sẽ phải cần một trường duy nhất là trường khóa.

**Primary key :** Khóa chính.

**Auto\_increment :** Tự động tăng.

**Lệnh thêm mới dữ liệu vào bảng :**

**Insert into table(col1, col2, col3 ) values ( v1, v2, v3 ) :** Biết có khai báo collumn.

**Insert into table value ( v1, v2, v3 )**

**Insert into table ( col1, col2 ) values ( v1, v2, v3 ) :** Thêm mới nhiều dữ liệu cùng 1 lúc.

**Lệnh lấy dữ liệu ra khỏi bảng :**

**Trên 1 bảng :**

**Select \* from bảng :** Lấy hết thông tin trong bảng.

**Select *tên trường, tên trường* from bảng :** Lấy thông tin bảng theo **trường**.

**Select *tên trường, tên trường* from bảng LIMIT *số*:** Lấy thông tin bảng theo **trường**. ***Limit*** giới hạn kết quả. **Limit 1,2** lấy 2 kết quả từ kết quả thứ nhất trở đi.

**Select *tên trường, tên trường* from bảng WHERE ( *tên trường BETWEEN 1000* and *10000* ) order by tên trường ( ASC or DESC ): *Where*** đặt điều kiện theo toán học ( = , != , > ). **Between** lấy dữ liệu trong khoảng. **ASC** tăng lên. **DESC** giảm dần.

**Ví dụ trên nhiều bảng thông qua khóa ngoại : Where**

**SELECT khachhang.ten from donhang, khachhang WHERE donhang.id\_khachhang = khachhang.id\_khachhang :** chọn từ 2 bảng đơn hàng và bảng khách hàng.

**SELECT khachhang.ten from donhang, khachhang WHERE donhang.id\_khachhang = khachhang.id\_khachhang AND donhang.tien > 10000 :** Tìm những người mua hàng lớn hơn 1000. Sử dụng câu lệnh **and** trong **where**.

**Ví dụ trên nhiều bảng : INNER JOIN và USING**

***SELECT khachhang.ten from donhang INNER JOIN khachhang WHERE donhang.id\_khachhang = khachhang.id\_khachhang AND donhang.tien > 10000*** : Cách số 2.

***SELECT khachhang.email from donhang INNER JOIN khachhang USING (id\_khachhang) WHERE donhang.phuongthuc = "COD"*** : Cách số 3. Sử dung **using** để kêt nối nhiều bảng dễ dàng hơn.

**Ví dụ trên nhiều bảng : INNER JOIN và ON**

***SELECT k.ten, d.phuongthuc from khachhang as k INNER JOIN donhang as d ON k.id\_khachhang = d.id\_khachhang where d.phuongthuc = 'visa'.*** : as thay bằng tên viết tắt.

**Kết nối với nhau bằng LEFT JOIN và INNER JOIN**

**Inner Join** : sẽ không lấy ra được khách hàng có tài khoản nhưng không mua hàng. Những phần tử có kết nối với bảng khác.

**Left Join** : sẽ lấy ra danh sách những phần tử bên trái **left join** có kết nối với bảng khác.

**Right Join:** sẽ ra danh sách những phần tử bên phải **right join** có kết nối với bảng khác.

**Khóa ngoại :**

**Khóa ngoại :** Khóa của trường bên ngoài . Chứa tông tin với bảng khác. Khóa chính của khóa chính của bảng khác.

**Tiết kiệm** dung lượng hơn. **Sửa** thông tin bảng **dễ dàng** hơn.

Một bảng có thể có **nhiều khóa ngoại** để kết nối **nhiều bảng**.

**adsas**

1. **Sử dụng alias**

Khi sử dụng nhiều bảng dữ liệu, tên bảng khó nhớ và dài.

**Tên bảng as tên ngắn : Alias** giống như là **nickname.**

1. **Hàm nối 2 cột với nhau**

***SELECT concat\_ws(" - Điện thoại ",k.diachi, k.dienthoai) as 'Thông tin khách hàng' from khachhang as k :***

**Hàm concat :** nối không khoảng cách hai trường với nhau.

**Hàm concat\_ws** : nối có khoảng cách hai trường với nhau.

1. **Cách định dạng ngày tháng**

***SELECT count(id\_sanpham) as "số lượng sản phẩm xuất hiện trong đon hàng" from donhang\_chitiet***

**Hàm count :** Điếm số lượng bản ghi.

***SELECT DATE\_FORMAT(donhang.ngaymua, "%d / %m / %Y - %I : %p") as "Ngay mua" from donhang***

**Hàm DATE\_FORMAT :** Điều chỉnh lại thời gian hiển thị.

1. **Hàm cắt nội dung text**

**SELECT LEFT(khachhang.ten, 2) FROM khachhang**

**Hàm LEFT :** Cắt chữ từ trái sang phải.

1. **Câu lệnh update**

***UPDATE khachhang SET diachi = "15 Trần Nhân Tông, HN" WHERE ten LIKE "%Hải%" LIMIT 1***

***UPDATE khachhang SET diachi = " 86 thanh lương " WHERE id\_khachhang = 1 LIMIT 1***

**Like :** Tìm kiếm kí tự tên nào đó, gần đúng.

1. **Câu lệnh xóa**

***DELETE FROM khachhang WHERE khachhang.ten LIKE “%Thang%” LIMIT 1***