Index trong SQL Server là một trong những yếu tố quan trọng nhất góp phần vào việc nâng cao hiệu suất của cơ sở dữ liệu. Index trong SQL Server tăng tốc độ của quá trình truy vấn dữ liệu bằng cách cung cấp phương pháp truy xuất nhanh chóng tới các dòng trong các bảng, tương tự như cách mà mục lục của một cuốn sách giúp bạn nhanh chóng tìm đến một trang bất kỳ mà bạn muốn trong cuốn sách đó.
Bạn đang xem: Index trong sql là gì
Trong bài viết này, chúng ta sẽ bàn tổng quan về **Index trong SQL Server** và giải thích cách chúng được định nghĩa trong một cơ sở dữ liệu cũng như làm thế nào mà index có thể làm cho quá trình truy vấn dữ liệu của chúng ta nhanh hơn. Hầu hết các kiến thức trong bài này đều có thể áp dụng cho SQL Server 2005 và 2008, kiến trúc của **Index trong SQL Server** có thay đổi một ít từ phiên bản này sang phiên bản khác, nhưng về cơ bản, chúng đều giữ được những thiết kế nguyên thủy, cho nên hầu như chúng ta sẽ không thấy có sự thay đổi nào. Thậm chí chúng ta cũng có thể ứng dụng rất nhiều các kiến thức ở bài này cho cả SQL Server 2000. Tuy nhiên hiện nay, 3 phiên bản phổ biến là SQL Server 2005, 2008 và mới nhất là SQL Server 2012. Chúng ta sẽ tập trung chủ yếu vào SQL Server 2005 và 2008.
Index trong SQL Server được tạo ra trên các cột trong bảng hoặc View. Chúng cung cấp một phương pháp giúp bạn nhanh chóng tìm kiếm dữ liệu dựa trên các giá trị trong các cột. Ví dụ, nếu bạn tạo ra một Index trên cột khóa chính và sau đó tìm kiếm một dòng dữ liệu dựa trên một trong các giá trị của cột này, đầu tiên SQL Server sẽ tìm giá trị này trong Index, sau đó nó sử dụng Index để nhanh chóng xác định vị trí của dòng dữ liệu bạn cần tìm. Nếu không có Index, SQL Server sẽ thực hiện động tác quét qua toàn bộ bảng (table scan) để xác định vị trí dòng cần tìm, mà table scan là một trong những động tác có hại nhất cho hiệu suất của SQL Server.
Index trong SQL Server có thể tạo trên hầu hết các cột trong bảng hoặc View. Ngoại trừ các cột dùng để lưu trữ các đối tượng dữ liệu lớn như kiểu Image, Text, varchar(max)… Bạn cũng có thể tạo Index trên các cột kiểu XML, nhưng Index trên XML hơi khác so với Index cơ bản nói chung và cũng ít được dùng nên chúng ta sẽ chưa bàn đến ở đây. **Index trong SQL Server** được tạo thành từ một tập hợp các page (các Index Node) và chúng được tổ chức trong một cấu trúc có tên gọi là B-tree.
Xem thêm: Ti Vi 4K Là Gì ? Có Nên Lựa Chọn Dòng Tivi 4K Lúc Này Hay Không?
Xem thêm: Tổng Hợp Cách Nấu Các Loại Xôi Ngon & Cách Nấu Xôi Ngon Để Bán Đắt Khách!
Cấu trúc này là cấu trúc kiểu thứ bậc trong tự nhiên, với các nút gốc ở trên cùng của hệ thống phân cấp và các nút lá ở phía dưới, như thể hiện trong hình dưới đây: Khi một truy vấn được xây dựng dựa trên các cột được tạo Index, cỗ máy thực thi truy vấn sẽ bắt đầu tại nút gốc và điều hướng qua các nút trung gian cho đến khi cỗ máy truy vấn tìm được đến nút lá.

Ví dụ, nếu bạn đang tìm kiếm giá trị 123 trong một cột được tạo index, ví dụ như cột ID chẳng hạn, đầu tiên cỗ máy truy vấn sẽ tìm ở nút gốc (Root Level) để xác định page nào sẽ được tham chiếu tới ở level trung gian (Intemediate Level). Trong ví dụ này, trang đầu tiên chỉ các giá trị từ 1-100, và trang thứ hai là các giá trị 101-200, vì vậy cỗ máy truy vấn sẽ đi đến trang thứ hai ở level trung gian. Cỗ máy truy vấn sau đó sẽ xác định trang tiếp theo mà nó phải tham chiếu tới ở level trung gian kế tiếp. Cuối cùng, cỗ máy truy vấn sẽ tìm đến nút lá cho giá trị 123. Nút lá sẽ chứa toàn bộ dòng dữ liệu hoặc nó chỉ chứa một con trỏ làm tham chiếu dến dòng dữ liệu, tùy thuộc vào việc Index trên cột ID này là kiểu **Clustered Index** hay là Non **Clustered Index**.

Để tìm hiểu về **Clustered Index** và **Non Clustered Index** là gì, mời bạn đọc xem chi tiết ở link

sau:
http://www.bigdata.com.vn/2013/04/clustered-index-va-non-clustered-index.html
Kiểu Index

Để bổ sung vào 2 kiểu Index cơ sở là **Clustered Index** và **Non Clustered Index**, chúng ta có thể mở rộng kiểu Index theo các cách sau đây:
Composite index: Là kiểu Index có nhiều hơn 1 cột. Trong SQL Server 2005 và 2008, bạn có thể có đồng thời tối đa 16 cột trong một Index, miễn là kích thước của Index không vượt quá giới hạn 900 byte. Cả hai kiểu index cơ sở là Clustered Index và Non Clustered Index cũng có thể đồng thời là là kiểu Composite index.
Unique Index: Là kiểu Index dùng để đảm bảo tính duy nhất trong các cột được tạo Index. Nếu Index loại này được tạo dựa trên nhiều cột, thì tính duy nhất của giá trị được tính trên tất cả các cột đó, không chỉ riêng rẽ từng cột. Ví dụ, nếu bạn đã tạo ra một Index trên các cột FirstName và LastName trong một bảng, thì giá trị của 2 cột này kết hợp với nhau phải là duy nhất, nhưng riêng rẽ từng cột thì giá trị vẫn có thể trùng nhau.
Một Unique Index được tự động tạo ra khi bạn định nghĩa một khóa chính (Primary Key) hoặc một ràng buộc duy nhất (Unique Constraint):
Primary Key: Khi bạn định nghĩa một ràng buộc khoá chính trên một hoặc nhiều cột của bảng, SQL Server tự động tạo ra một Unique – **Clustered Index** nếu chưa có một **Clustered Index** nào tồn tại trên bảng hoặc view. Tuy nhiên, bạn hoàn toàn có thể khai báo lại hành vi mặc định này của SQL Server bằng cách định nghĩa lại index cho cột hoặc nhóm cột này là Unique – **Non Clustered Index**.
Unique: Khi bạn định nghĩa một ràng buộc duy nhất, SQL Server tự động tạo ra một index có các đặc tính là Unique và là Non **Clustered Index**. Bạn cũng hoàn toàn có thể tạo ra một Unique và là **Clustered Index** nếu như chưa có một **Clustered Index** nào được tạo ra trước đó trên bảng.
Covering index: là một loại chỉ số bao gồm tất cả các cột cần thiết để xử lý một truy vấn cụ thể. Ví dụ, truy vấn của bạn có thể lấy các cột FirstName và LastName từ một bảng, dựa trên một giá trị trong cột ContactID. Từ đó, để tăng tốc độ xử lý câu truy vấn, bạn có thể tạo ra một chỉ số bao gồm tất cả ba cột này.
Chuyên mục: Hỏi Đáp

THAM KHẢO THÊM: https://bongdanews.top/
The post Chỉ Mục (Index Trong Sql Là Gì, Chỉ Mục (Index) Trong Sql appeared first on BONGDANEWS.TOP.

via BONGDANEWS.TOP
https://bongdanews.top/chi-muc-index-trong-sql-la-gi-chi-muc-index-trong-sql/