

Một số câu hỏi technical phỏng vấn Data Analytics

Contents:

SQL.....	2
phân biệt syntax unique và primary key trong SQL	2
Phân biệt drop và delete trong SQL	2
khái niệm index trong SQL	3
Phân biệt các loại JOIN trong SQL	4
Case study questions.....	5
Case #1/.....	5
Case #2/.....	5
Case #3/.....	6

SQL

phân biệt syntax unique và primary key trong SQL

Trong SQL, cả "UNIQUE" và "PRIMARY KEY" là các ràng buộc (constraints) được sử dụng để **đảm bảo tính duy nhất** của giá trị trong cột hoặc tập hợp các cột. Tuy nhiên, chúng có mục đích và sử dụng khác nhau:

1) UNIQUE Constraint (Ràng buộc duy nhất):

- "UNIQUE" là một ràng buộc dùng để đảm bảo rằng các giá trị trong cột hoặc các cột đã chọn phải là duy nhất trong bảng.
- Một cột hoặc tập hợp các cột có ràng buộc "UNIQUE" có thể chứa các giá trị không trùng lặp, nhưng **không nhất thiết** phải là khóa chính.
- Bạn có thể áp dụng nhiều ràng buộc "UNIQUE" trên các cột khác nhau trong bảng.

Ví dụ:

```
CREATE TABLE Employees (  
    EmployeeID INT PRIMARY KEY,  
    Email VARCHAR(255) UNIQUE,  
    Username VARCHAR(50) UNIQUE  
);
```

2) PRIMARY KEY:

- "PRIMARY KEY" là một ràng buộc dùng để **xác định duy nhất mỗi bản ghi** trong bảng. Một cột hoặc tập hợp các cột với ràng buộc "PRIMARY KEY" phải chứa giá trị duy nhất và **không thể trống (NULL)**.
- Một bảng chỉ có thể có một "PRIMARY KEY" duy nhất. "PRIMARY KEY" thường được sử dụng để xác định cách duy nhất để truy cập các bản ghi trong bảng.

Ví dụ:

```
CREATE TABLE Students (  
    StudentID INT PRIMARY KEY,  
    FirstName VARCHAR(50),  
    LastName VARCHAR(50),  
    Email VARCHAR(255) UNIQUE  
);
```

Tóm lại:

- "UNIQUE" là một ràng buộc để đảm bảo tính duy nhất của giá trị trong cột hoặc tập hợp các cột, và nhiều cột có thể có ràng buộc "UNIQUE".
- "PRIMARY KEY" là một ràng buộc để xác định duy nhất mỗi bản ghi trong bảng và chỉ có thể áp dụng cho một cột hoặc tập hợp các cột.

Phân biệt drop và delete trong SQL

Trong SQL, "DROP" và "DELETE" là hai câu lệnh khác nhau về việc xóa dữ liệu hoặc đối tượng khỏi cơ sở dữ liệu. Dưới đây là sự phân biệt giữa chúng:

1. **DROP**: Lệnh "DROP" được sử dụng để xóa hoàn toàn một đối tượng khỏi cơ sở dữ liệu, ví dụ như bảng, cơ sở dữ liệu, chỉ mục (index) hoặc bất kỳ đối tượng nào khác. Khi bạn sử dụng "DROP", đối tượng và tất cả dữ liệu liên quan đến nó sẽ bị xóa hoàn toàn và **KHÔNG thể khôi phục** được mà không có bản sao dự phòng trước đó.

DROP TABLE TenBang;

DROP DATABASE TenCSDL;

2. **DELETE**: Lệnh "DELETE" được sử dụng để xóa dữ liệu từ bảng. Khi bạn sử dụng "DELETE", bạn chỉ xóa các bản ghi dựa trên điều kiện được chỉ định. Dữ liệu trong bảng sẽ bị xóa, nhưng **bảng vẫn TỒN TẠI** và có thể tiếp tục sử dụng.

DELETE FROM TenBang WHERE DieuKien;

Tóm lại:

- Sử dụng lệnh "DROP" để **xóa** đối tượng **hoàn toàn** khỏi cơ sở dữ liệu, và sau khi xóa, đối tượng không còn tồn tại.
- Sử dụng lệnh "DELETE" để xóa dữ liệu từ bảng dựa trên điều kiện, và bảng vẫn còn tồn tại sau khi xóa dữ liệu.

Khái niệm index trong SQL

Trong SQL, khái niệm "index" thường được sử dụng để **tăng cường hiệu suất truy vấn** trong cơ sở dữ liệu. Index là một cấu trúc dữ liệu được sử dụng để tăng tốc độ truy vấn dữ liệu bằng cách tạo ra một cách nhanh chóng các liên kết giữa dữ liệu và **vị trí lưu trữ** của nó trong bộ nhớ.

Khi bạn thực hiện truy vấn trong cơ sở dữ liệu, hệ quản trị cơ sở dữ liệu (DBMS) sẽ tìm kiếm thông tin trong bảng. Index giúp DBMS tìm kiếm thông tin nhanh hơn bằng cách tạo ra một danh sách liên kết với các giá trị trong cột đã chỉ định và vị trí của chúng trong bảng.

Có hai loại index phổ biến:

1) **Clustered Index** (Index nhóm)

- Chỉ có một clustered index trong mỗi bảng.
- Dữ liệu trong bảng được tổ chức lại theo thứ tự của cột được chọn làm clustered index.
- Thường được sử dụng để tăng tốc độ truy vấn theo dữ liệu liên quan đến cột indexed.

2) **Non-clustered Index** (Index không nhóm)

- Mỗi bảng có thể có nhiều non-clustered index.
- Dữ liệu trong bảng không được tổ chức lại, chỉ có danh sách liên kết với các giá trị indexed và vị trí của chúng.
- Thường được sử dụng để tăng tốc độ truy vấn dựa trên các cột indexed và không ảnh hưởng đến cấu trúc lưu trữ dữ liệu.

Một số lưu ý khi sử dụng index trong SQL:

- Index tăng tốc độ truy vấn nhưng cũng có thể làm tăng thời gian cập nhật, chèn, xóa dữ liệu.
- Index nên được sử dụng trên các cột thường xuyên được truy vấn hoặc sắp xếp.

- Index giúp truy vấn dữ liệu nhanh hơn, nhưng cần được sử dụng một cách cân nhắc để tránh tạo quá nhiều index không cần thiết.

Ví dụ tạo index:

```
CREATE INDEX idx_last_name ON employees (last_name);
```

Trong ví dụ này, chúng ta tạo một **non-clustered index** trên cột "last_name" trong bảng "employees". Điều này sẽ làm cho các truy vấn liên quan đến cột "last_name" nhanh hơn.

Phân biệt các loại JOIN trong SQL

Trong SQL, các loại JOIN được sử dụng để kết hợp dữ liệu từ các bảng khác nhau dựa trên một điều kiện hoặc mối quan hệ. Dưới đây là phân biệt giữa các loại JOIN phổ biến:

1) INNER JOIN:

- **INNER JOIN** (còn gọi là EQUIJOIN) trả về các bản ghi mà có sự khớp hoàn toàn về điều kiện kết nối giữa các bảng.
- Nếu không có khớp nào, bản ghi sẽ không được hiển thị trong kết quả.

2) LEFT JOIN (LEFT OUTER JOIN):

- **LEFT JOIN** trả về tất cả các **bản ghi từ bảng bên trái** và các bản ghi khớp từ bảng bên phải.
- Nếu không có khớp nào từ bảng bên phải, các cột tương ứng sẽ có giá trị **NULL**.

3) RIGHT JOIN (RIGHT OUTER JOIN):

- **RIGHT JOIN** tương tự như **LEFT JOIN** nhưng ngược lại. Nó trả về tất cả các bản ghi từ bảng bên phải và các bản ghi khớp từ bảng bên trái.
- Nếu không có khớp nào từ bảng bên trái, các cột tương ứng sẽ có giá trị **NULL**.

4) FULL JOIN (FULL OUTER JOIN):

- **FULL JOIN** trả về tất cả các bản ghi **từ cả hai bảng** (bảng bên trái và bảng bên phải).
- Nếu không có khớp nào từ bảng bên trái hoặc bên phải, các cột tương ứng sẽ có giá trị **NULL**.

5) CROSS JOIN (CARTESIAN JOIN):

- **CROSS JOIN** tạo ra tất cả các sự kết hợp có thể giữa các bản ghi trong hai bảng, tạo ra một bảng kết quả lớn.
- Điều này thường **KHÔNG** được **sử dụng trong** các trường hợp **thực tế** vì nó có thể tạo ra kết quả rất lớn và không hữu ích.

Mỗi loại JOIN có mục đích và ứng dụng riêng, và sự lựa chọn phụ thuộc vào yêu cầu cụ thể của truy vấn và cơ cấu dữ liệu.

Case study questions

Case #1/

Ví dụ như công ty IT ABxy chuyên cung cấp dịch vụ lập trình website, với vai trò là Data Analyst bạn sẽ làm cách nào để nâng cao việc tìm kiếm khách hàng.

Giải pháp:

Là một Data Analyst tại công ty ABxy chuyên cung cấp dịch vụ lập trình website, bạn có thể thực hiện nhiều hoạt động để nâng cao việc tìm kiếm khách hàng. Dưới đây là một số cách bạn có thể thực hiện:

1. **Phân tích dữ liệu hiện tại:** Đầu tiên, hãy xem xét dữ liệu hiện tại về khách hàng, ví dụ: thông tin liên hệ, lịch sử dịch vụ, dự án đã thực hiện, v.v. Phân tích dữ liệu này để hiểu rõ hơn về hành vi của khách hàng, loại dự án họ thường yêu cầu, thời gian giữa các dự án, v.v.
2. **Segmentation (Phân đoạn khách hàng):** Dựa trên phân tích dữ liệu, hãy phân loại khách hàng thành các nhóm có các đặc điểm tương tự nhau. Điều này giúp bạn hiểu rõ hơn về mục tiêu của từng nhóm khách hàng và tạo ra chiến lược tiếp thị cụ thể cho mỗi nhóm.
3. **Xây dựng hồ sơ khách hàng:** Xây dựng hồ sơ khách hàng chi tiết với thông tin như ngành nghề, quy mô doanh nghiệp, nhu cầu cụ thể, vị trí liên hệ, v.v. Điều này giúp bạn tạo ra thông tin chi tiết để tìm kiếm và tiếp cận khách hàng mục tiêu.
4. **Sử dụng dữ liệu bên ngoài:** Không chỉ dựa vào dữ liệu nội bộ, bạn cũng nên khai thác dữ liệu từ các nguồn bên ngoài như cơ sở dữ liệu công cộng, mạng xã hội, và các trang web ngành để tìm kiếm tiềm năng khách hàng mới.
5. **Xây dựng chiến dịch tiếp thị thông minh:** Sử dụng dữ liệu để xây dựng chiến dịch tiếp thị thông minh hơn. Điều này có thể là việc tạo nội dung hấp dẫn dựa trên sở thích và nhu cầu của từng nhóm khách hàng, hoặc tạo ra các chương trình khuyến mãi đặc biệt dựa trên dự án đã thực hiện.
6. **Dùng công cụ hỗ trợ:** Sử dụng các công cụ phân tích dữ liệu và hệ thống quản lý khách hàng (CRM) để hiểu rõ hơn về khách hàng và quản lý thông tin của họ. Công cụ này giúp bạn theo dõi tương tác với khách hàng và tạo ra kế hoạch tiếp thị hiệu quả hơn.
7. **Đo lường hiệu suất:** Đảm bảo bạn đo lường hiệu suất của các chiến dịch tiếp thị và các hoạt động tìm kiếm khách hàng. Điều này giúp bạn hiểu rõ hơn về những gì hoạt động và điều chỉnh chiến lược dựa trên dữ liệu thực tế.

Tóm lại, như một Data Analyst, việc tìm kiếm khách hàng hiệu quả yêu cầu bạn phân tích dữ liệu một cách thông minh, hiểu rõ về khách hàng và tạo ra các chiến lược tiếp thị đích đảm bảo sự hiệu quả và tối ưu hóa.

Case #2/

Thu nhỏ phạm vi của câu hỏi trên, phân tích dữ liệu để xác định yếu tố của công ty mà khách hàng yêu thích.

Giải pháp:

Để phân tích dữ liệu để xác định yếu tố của công ty mà khách hàng yêu thích, bạn có thể thực hiện các bước sau:

1. **Thu thập dữ liệu:** Thu thập dữ liệu liên quan đến các dự án đã thực hiện, phản hồi từ khách hàng, dữ liệu về các tính năng của sản phẩm hay dịch vụ, đánh giá từ khách hàng, v.v.

2. **Xây dựng bộ dữ liệu:** Tổ chức và xây dựng bộ dữ liệu từ các nguồn thu thập được. Đảm bảo rằng dữ liệu là cụ thể, đầy đủ và có cấu trúc để dễ dàng phân tích.
3. **Phân tích đánh giá và phản hồi:** Xem xét các đánh giá và phản hồi từ khách hàng. Phân tích các mẫu phản hồi và đánh giá để xác định những điểm mạnh và điểm yếu của công ty trong mắt khách hàng.
4. **Phân tích dự án đã thực hiện:** Xem xét lịch sử các dự án đã thực hiện để xác định các yếu tố mà khách hàng thường yêu thích. Điều này có thể là tính chất của dự án, kỹ năng cụ thể, hoặc quy trình làm việc.
5. **Phân tích dữ liệu khảo sát:** Nếu có dữ liệu từ các khảo sát hoặc cuộc trò chuyện với khách hàng, hãy phân tích để xác định các góc nhìn, ý kiến và mong muốn của họ đối với công ty và dịch vụ của bạn.
6. **Áp dụng kỹ thuật phân tích dữ liệu:** Sử dụng các kỹ thuật phân tích dữ liệu như biểu đồ, biểu đồ phân phối, phân tích tương quan để thấy rõ các mẫu và xu hướng trong dữ liệu.
7. **Tìm kiếm mẫu và xu hướng:** Tìm kiếm các mẫu chung và xu hướng trong dữ liệu. Xác định những yếu tố mà khách hàng thường nhắc đến nhiều trong các phản hồi và đánh giá.
8. **Xác định yếu tố yêu thích:** Dựa trên phân tích, xác định những yếu tố cụ thể mà khách hàng yêu thích về công ty, ví dụ: chất lượng dịch vụ, thời gian giao hàng, mức giá hợp lý, tính sáng tạo, v.v.
9. **Tạo chiến lược dựa trên dữ liệu:** Dựa trên những yếu tố mà khách hàng yêu thích, tạo ra chiến lược cải thiện hoặc tăng cường những khía cạnh này để thu hút và duy trì sự hài lòng của khách hàng.
10. **Đánh giá hiệu suất:** Sau khi triển khai các chiến lược, đánh giá hiệu suất của chúng bằng cách theo dõi phản hồi từ khách hàng và các thay đổi trong lịch sử giao dịch.

Tóm lại, phân tích dữ liệu để xác định yếu tố của công ty mà khách hàng yêu thích là một quá trình tỉ mỉ, đòi hỏi sự chú ý đến từng chi tiết trong dữ liệu để tìm ra những thông tin quan trọng giúp cải thiện dịch vụ và tương tác với khách hàng.

Case #3/

Với vai trò **data analyst** được giao để phân tích dữ liệu để tìm ra giải pháp, quy trình tiếp cận các phòng ban như thế nào, cần lấy những dữ liệu gì, thu thập từ đâu, thực hiện ra sao và phân tích, đưa ra giải pháp gì để thực hiện?

Giải pháp:

Để tiếp cận quy trình phân tích dữ liệu và đưa ra giải pháp dựa trên vai trò Data Analyst, bạn có thể tuân theo các bước sau:

- 1) **Xác định Vấn đề hoặc Mục tiêu:** Trước tiên, xác định rõ vấn đề cụ thể mà bạn đang cố gắng giải quyết hoặc mục tiêu mà bạn muốn đạt được bằng việc phân tích dữ liệu. Điều này giúp bạn hướng phân tích của mình vào hướng chính xác.
- 2) **Xác định Dữ liệu Cần Thiết:** Xác định loại dữ liệu cần thiết để giải quyết vấn đề hoặc đáp ứng mục tiêu. Điều này có thể là dữ liệu liên quan đến khách hàng, dự án, doanh thu, chi phí, v.v. Dựa trên mục tiêu, bạn xác định các thuộc tính dữ liệu cần sử dụng.
- 3) **Thu thập Dữ liệu:** Thu thập dữ liệu từ các nguồn có sẵn như cơ sở dữ liệu của công ty, tệp dữ liệu Excel, hệ thống CRM, hoặc thậm chí từ các nguồn bên ngoài nếu cần. Đảm bảo dữ liệu là chính xác, đầy đủ và được sắp xếp một cách cấu trúc.

- 4) **Tiền xử lý Dữ liệu:** Trước khi bắt đầu phân tích, bạn cần làm sạch và xử lý dữ liệu. Loại bỏ dữ liệu trùng lặp, điền các giá trị bị thiếu, chuẩn hóa dữ liệu, v.v. Điều này đảm bảo dữ liệu của bạn làm việc tốt trong quá trình phân tích.
- 5) **Phân tích Dữ liệu:** Sử dụng các kỹ thuật phân tích dữ liệu như thống kê mô tả, biểu đồ, biểu đồ phân phối, phân tích tương quan, v.v. Điều này giúp bạn hiểu rõ hơn về dữ liệu và tìm ra các mẫu, xu hướng, và thông tin quan trọng.
- 6) **Đưa ra Giải pháp:** Dựa trên phân tích, đưa ra các giải pháp hoặc khuyến nghị. Ví dụ: nếu phát hiện rằng khách hàng thường yêu cầu dự án về thương mại điện tử, bạn có thể đề xuất tăng cường chiến dịch tiếp thị cho lĩnh vực này.
- 7) **Trình bày Kết quả:** Tạo báo cáo hoặc trình bày với những số liệu và hình ảnh minh họa để giới thiệu kết quả phân tích và giải pháp đến các phòng ban có liên quan.
- 8) **Gặp phản hồi và Điều chỉnh:** Sau khi trình bày kết quả, lắng nghe phản hồi từ các phòng ban liên quan. Dựa trên phản hồi, điều chỉnh và cải tiến giải pháp nếu cần thiết.

Tóm lại, việc phân tích dữ liệu và đưa ra giải pháp như một Data Analyst yêu cầu bạn tuân theo quy trình có hệ thống và chặt chẽ từ việc xác định vấn đề đến trình bày kết quả và thực hiện điều chỉnh.