

1. Tạo 1 view, query từ view đó. Giải thích tại sao bạn lại tạo view đó.

Tạo 1 view

```
CREATE VIEW V_USER_MEDIACONTENT
AS
SELECT U.FistName,U.MiddleName,U.LastName,M.[Subject],M.Content
FROM [USER] AS U INNER JOIN MEDIACONTENT AS M
ON U.UserID = M.UserID
```

query từ view đó

```
SELECT *
FROM V_USER_MEDIACONTENT
```

Giải thích

- Giới hạn truy cập tới dữ liệu để mà một người dùng có thể thấy, ở đây giới hạn người dùng chỉ thấy được họ tên của một USER và link để tải về và xem nội dung MEDIA mà USER đó tải lên.
 - Giúp người dùng có thể tìm thấy dữ liệu cần tìm là các nội dung media trực quan hơn.
 - Tổng kết báo cáo dữ liệu về số lượng USER và nội dung mà người đó tải lên.
- ## 2. Tạo 1 Trigger để thực thi 1 business constraint bất kỳ. Giải thích tại sao bạn lại làm như vậy.

Tạo 1 Trigger để thực thi 1 business constraint bất kỳ

```
CREATE TRIGGER trgNONEWTABLE
ON DATABASE
FOR CREATE_TABLE
AS
BEGIN
    PRINT 'BẠN KHÔNG ĐƯỢC TẠO BẢNG MỚI'
    ROLLBACK
END
```

Giải thích

Đây là một Trigger cho Data Declaration . Trigger được tạo ra để theo dõi sự kiện Create Table trong một database, đảm bảo người dùng không thể tạo ra bảng mới. Khi mà người dùng cố gắng tạo ra bảng mới thì sẽ thông báo cho người dùng là không được phép và Rollback để quay trở lại trạng thái trước khi có thay đổi.

3. Tạo một Store procedure để làm công việc bất kỳ. Hãy thảo luận về lợi ích của việc sử dụng store procedure này.

Tạo một Store procedure để làm công việc bất kỳ

```
CREATE PROCEDURE spFIND(@Name nvarchar(max))
AS
BEGIN
    SELECT *
    FROM [USER]
    WHERE LastName = @Name
END
```

SpFIND dùng để tìm kiếm USER bằng cách truyền tham số là tên USER cần tìm.

Lợi ích của việc sử dụng store procedure này

- Tăng tốc độ thực hiện: stored procedure được tối ưu hoá trong ngay ở lần biên dịch đầu tiên, điều này cho phép chúng có thể thực hiện nhanh hơn nhiều lần so với các câu lệnh T-SQL thông thường.
- Tốc độ truy cập dữ liệu nhanh hơn: Khi thực thi một câu lệnh SQL thì SQL Server phải kiểm tra permission xem user gửi câu lệnh đó có được phép thực hiện câu lệnh hay không đồng thời kiểm tra cú pháp rồi mới tạo ra một execute plan và thực thi. Nếu có nhiều câu lệnh như vậy gửi qua network có thể làm giảm đi tốc độ làm việc của server. SQL Server sẽ làm việc hiệu quả hơn nếu dùng stored procedure vì người gửi chỉ gửi một câu lệnh đơn và SQL Server chỉ kiểm tra một lần sau đó tạo ra một execute plan và thực thi. Nếu stored procedure được gọi nhiều lần thì execute plan có thể được sử dụng lại nên sẽ làm việc nhanh hơn. Ngoài ra cú pháp của các câu lệnh SQL đã được SQL Sever kiểm tra trước khi lưu nên nó không cần kiểm lại khi thực thi.
- Chương trình được modul hoá: stored procedure được tạo ra có thể được sử dụng lại.

4. Tạo một User Defined Function bất kỳ và giải thích tại sao bạn lại sử dụng Function này

Tạo một User Defined Function bất kỳ

```
CREATE FUNCTION CalculateRead(@MediaId int)
RETURNS int
AS
BEGIN
    DECLARE @SumRead int
    SELECT @SumRead = COUNT(ReadID)
    FROM [dbo].[READ]
    WHERE MediaID = @MediaId
    RETURN @SumRead
END
```

Giải thích

Tạo một Scalar User Defined Function CalculateRead có tham số là @MediaId int để tính số lượng người đã Read một Media có MediaID là @MediaId trên bảng READ kiểu trả về int. Sử dụng hàm này giúp người dùng tính toán chỉ cần nhập số ID của Media để tính ra kết quả số người đã Read. Nó có lợi ích tăng tốc độ thực hiện.