

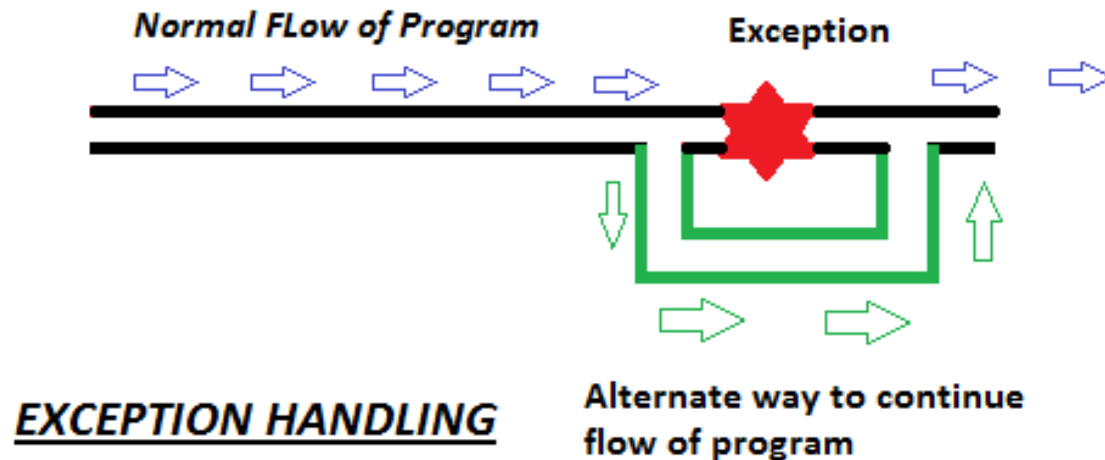
TRƯỜNG ĐÀO TẠO LẬP TRÌNH VIÊN VÀ QUẢN TRỊ MẠNG QUỐC TẾ BACHKHOA-APTECH

Bài 15

Quản lý lỗi (Error Handling)

Tóm tắt

- Giải thích lỗi
- Thực thi lỗi trong Transact-SQL



Giới thiệu

Việc **quản lý lỗi** trong SQL Server **trở lên dễ dàng** thông qua một số kỹ thuật khác nhau như:

- SQL Server cung cấp câu lệnh **TRY...CATCH** giúp quản lý các lỗi hiệu quả ở phía sau (back end).
- SQL Server cũng cung cấp một số các **hàm hệ thống** để in ra các thông tin có liên quan đến lỗi, để giúp sửa lỗi (fix) một cách dễ dàng.

Giới thiệu

Một lập trình viên Transact-SQL cần phải **xác định loại lỗi**, sau đó sẽ xác định cách **quản lý và khắc phục** nó.

Dưới đây là một số loại lỗi:

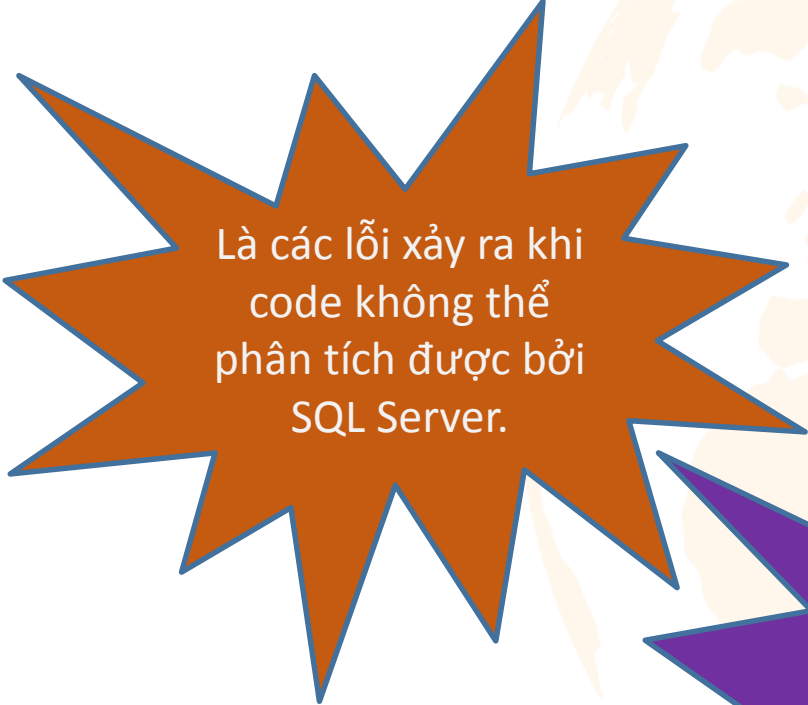
- Các **lỗi cú pháp**:
- Lỗi lúc **thực thi** (Run-time Errors)

```
SELECT id FROM users WHERE username=' ' and password=' ' ,
```

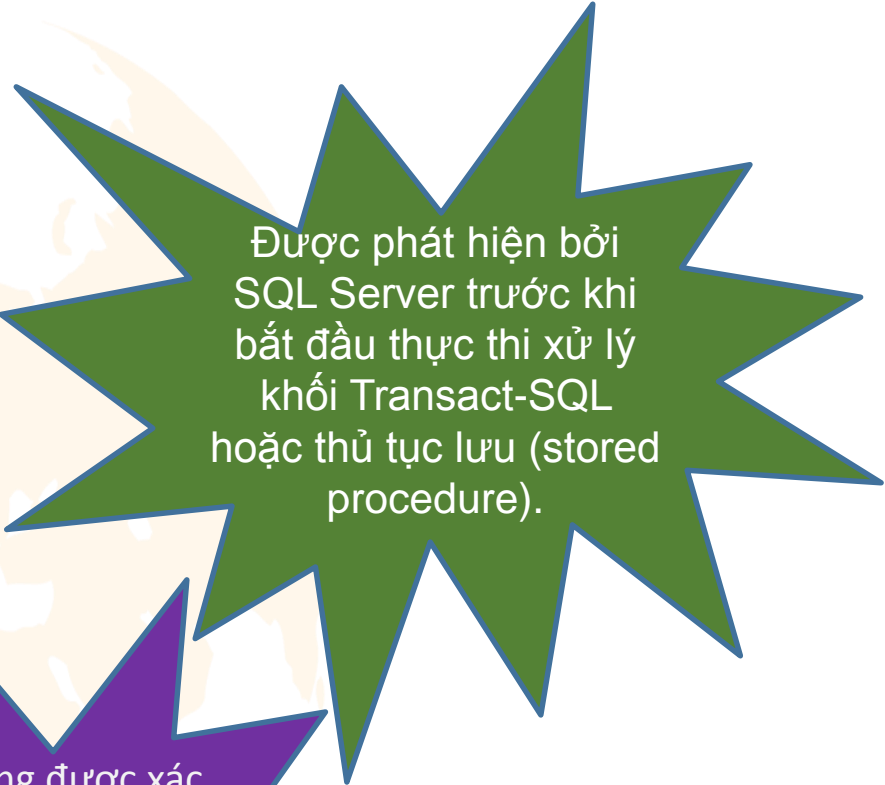
SQL SYNTAX ERROR !

Giới thiệu

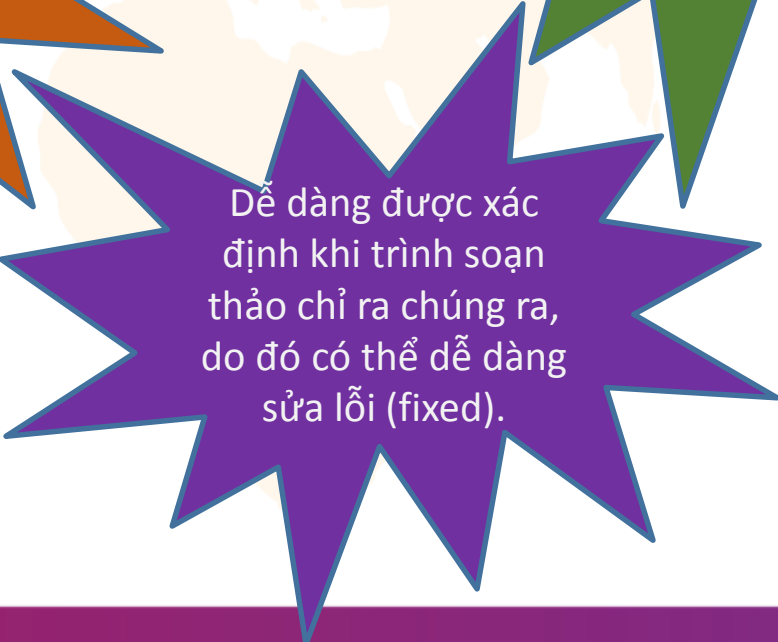
Lỗi cú pháp



Là các lỗi xảy ra khi code không thể phân tích được bởi SQL Server.



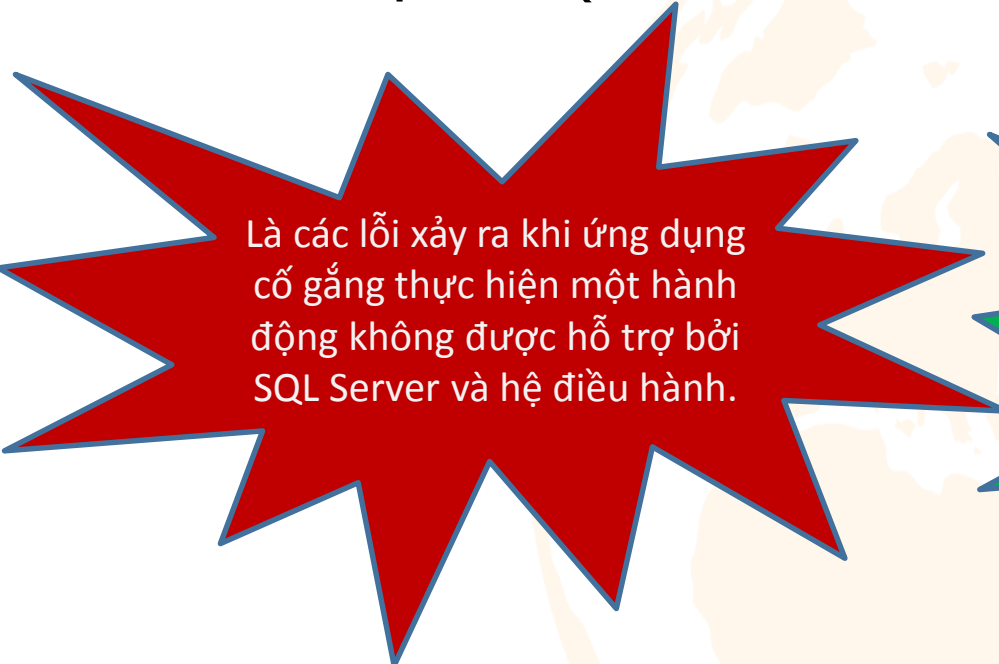
Được phát hiện bởi SQL Server trước khi bắt đầu thực thi xử lý khối Transact-SQL hoặc thủ tục lưu (stored procedure).



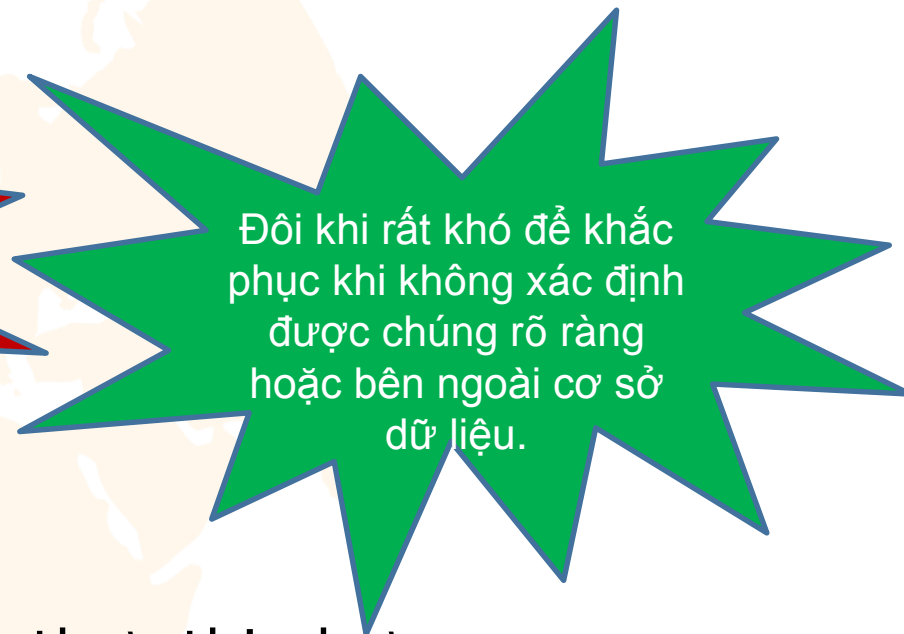
Dễ dàng được xác định khi trình soạn thảo chỉ ra chúng ra, do đó có thể dễ dàng sửa lỗi (fixed).

Giới thiệu

Lỗi lúc thực thi (Run-time Errors)



Là các lỗi xảy ra khi ứng dụng cố gắng thực hiện một hành động không được hỗ trợ bởi SQL Server và hệ điều hành.



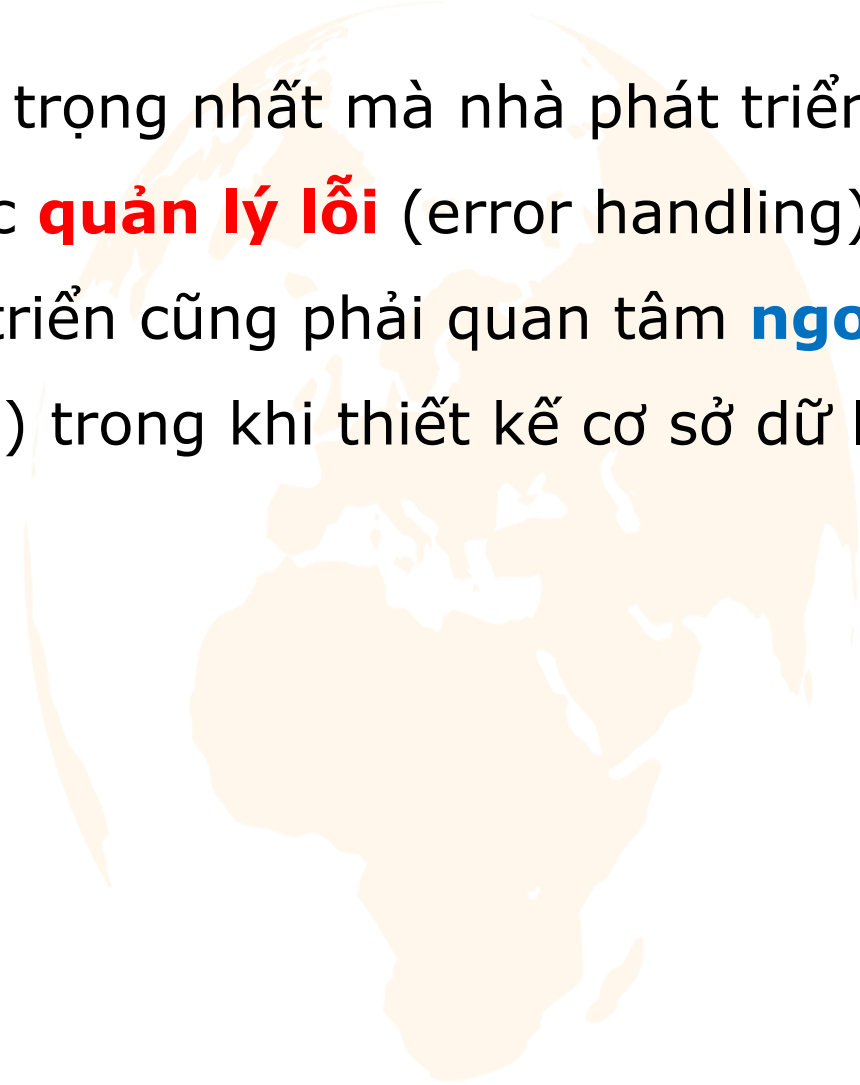
Đôi khi rất khó để khắc phục khi không xác định được chúng rõ ràng hoặc bên ngoài cơ sở dữ liệu.

Một số trường hợp của lỗi lúc thực thi như sau:

- Thực hiện một phép tính như chia 0.
- Cố gắng để thực thi mã không được xác định rõ ràng.

Thực thi

- Điều quan trọng nhất mà nhà phát triển cần quan tâm là việc **quản lý lỗi** (error handling).
- Nhà phát triển cũng phải quan tâm **ngoại lệ** (exception) trong khi thiết kế cơ sở dữ liệu.



Thực thi

Các cơ chế quản lý lỗi khác nhau như sau:

- **Khi thi hành** (executing) một số các câu lệnh như INSERT, DELETE, và UPDATE, người dùng có thể xử lý(handle) các lỗi để đảm bảo ra kết quả đúng.
- Khi một **giao dịch** (transaction) **thất bại** (fails) và người dùng cần hủy bỏ (rollback) giao dịch đó, một thông báo phù hợp được hiển thị.
- Khi **làm việc với các cursors** (con trỏ) trong SQL Server, người dùng có thể xử lý các lỗi để đảm bảo kết quả đúng.

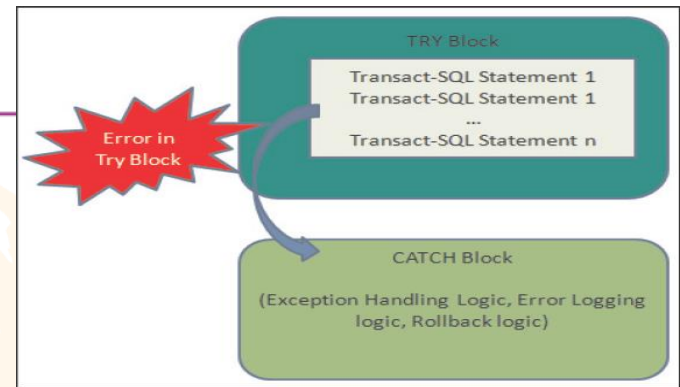
Thực thi

TRY...CATCH:

- Được sử dụng để **thực thi xử lý** các ngoại lệ trong Transact-SQL.
- Có thể chứa (enclose) **một hoặc nhiều câu lệnh** Transact-SQL vào trong một khối TRY.
- Nếu có một lỗi xảy ra trong khối TRY, chuyển điều khiển đến khối CATCH là khối có thể chứa một hoặc nhiều câu lệnh.

Thực thi

Cú pháp:



```
BEGIN TRY
    { sql_statement | statement_block }
END TRY
BEGIN CATCH
    [ { sql_statement | statement_block } ]
END CATCH[ ; ]
```

Trong đó:

- `sql_statement`: chỉ ra câu lệnh Transact-SQL bất kỳ.
- `statement_block`: chỉ ra nhóm các câu lệnh Transact-SQL trong khối `BEGIN...END`.

Thực thi

- Cấu trúc **TRY ... CATCH** sẽ **bắt tất cả các lỗi** lúc thực thi(run-times) **có mức độ nghiêm trọng** (severity level) **cao hơn 10** và không đóng kết nối cơ sở dữ liệu.
- Khối TRY...CATCH **không thể trải ra** trên nhiều **lô** (batches) hoặc **khối** (blocks) các **câu lệnh** Transact-SQL.

Thực thi

Ví dụ:

```
BEGIN TRY
```

```
    DECLARE @num int;
```

```
    SELECT @num=217/0;
```

```
END TRY
```

```
BEGIN CATCH
```

```
    PRINT 'Error occurred, unable to divide by 0'
```

```
END CATCH;
```

Thực thi

- Cả hai **khối TRY và CATCH** có thể **chứa các cấu trúc TRY...CATCH lồng**
- Nếu khối CATCH có chứa (encloses) một cấu trúc TRY...CATCH lồng, bất kỳ lỗi nào trong khối TRY lồng sẽ **chuyển (pass) điều khiển** tới khối CATCH lồng
- Nếu không có cấu trúc TRY...CATCH lồng, lỗi được **truyền (pass) quay lại** nơi gọi
- Cấu trúc TRY...CATCH cũng có thể **bắt** các **lỗi không được xử lý** từ các **trigger hoặc thủ tục lưu** thực thi thông qua code trong khối TRY.

Thông báo lỗi:

- Xử lý thông báo lỗi tốt là để hiển thị thông tin lỗi cùng với các lỗi, do đó, nó có thể **giúp giải quyết các lỗi** một cách **nhANH chóng** và **hiệu quả**.
- Trong phạm vi của khối CATCH, **các hàm hệ thống** sau đây có thể được dùng để lấy các thông tin về lỗi là nguyên nhân làm cho khối CATCH được thực thi.

Dưới đây là các hàm hệ thống:

- `ERROR_NUMBER()`: trả về một số là mã của lỗi.
- `ERROR_SEVERITY()`: returns the severity.
- `ERROR_STATE()`: trả về một số là mã trạng thái.
- `ERROR_PROCEDURE()`: trả về tên của trigger hoặc stored procedure là nơi mà lỗi xảy ra.
- `ERROR_LINE()`: trả về một số là vị trí của dòng gây ra lỗi.
- `ERROR_MESSAGE()`: Trả về văn bản(text) đầy đủ của lỗi. Văn bản có chứa giá trị được cung cấp cho các tham số như tên đối tượng, độ dài, hoặc thời gian.

Các hàm trả về `NULL` khi chúng được gọi bên ngoài phạm vi của khối `CATCH` .

Thực thi

Kết quả

Ví dụ:

```
BEGIN TRY
```

```
SELECT 217/0;
```

```
END TRY
```

```
BEGIN CATCH
```

```
SELECT ERROR_NUMBER() AS ErrorNumber,
```

```
ERROR_SEVERITY() AS ErrorSeverity,
```

```
ERROR_LINE() AS ErrorLine,
```

```
ERROR_MESSAGE() AS ErrorMessage;
```

```
END CATCH;
```

```
GO
```

(No column name)				
	ErrorNumber	ErrorSeverity	ErrorLine	ErrorMessage
1	8134	16	2	Divide by zero error encountered.

Thực thi

Try...Catch với Transaction (giao dịch):

```
BEGIN TRANSACTION;
```

```
BEGIN TRY
```

```
DELETE FROM Production.Product WHERE ProductID = 980;
```


```
END TRY
```

```
BEGIN CATCH
```

```
SELECT ERROR_SEVERITY() AS 'Mức nghiêm trọng'  
, ERROR_NUMBER() AS 'Mã lỗi' , ERROR_PROCEDURE() AS  
'Thủ tục lỗi' , ERROR_STATE() AS 'Trạng thái lỗi',  
ERROR_MESSAGE() AS 'Thông điệp lỗi', ERROR_LINE() AS  
'Dòng lỗi';
```

Thực thi

```
IF @@TRANCOUNT > 0  
ROLLBACK TRANSACTION;  
END CATCH;  
IF @@TRANCOUNT > 0  
COMMIT TRANSACTION;  
GO
```



Tóm tắt bài học

- **Lỗi cú pháp** là các lỗi xảy ra khi code không thể phân tích được bởi SQL Server.
- **Lỗi thực thi** (Run-time) xảy ra khi ứng dụng cố gắng để thực hiện một hành động mà không được hỗ trợ bởi Microsoft SQL Server và bởi hệ điều hành.
- **Câu lệnh TRY ... CATCH** được sử dụng để xử lý các trường hợp ngoại lệ trong Transact-SQL.
- Cấu trúc TRY ... CATCH cũng có thể **bắt các lỗi** không được quản lý từ các **trigger hoặc thủ tục lưu** mà thực thi thông qua code trong một khối TRY.
- **Các hàm hệ thống** khác nhau có sẵn trong Transact-SQL để in thông tin lỗi về lỗi xảy ra.

TRƯỜNG ĐÀO TẠO LẬP TRÌNH VIÊN VÀ QUẢN TRỊ MẠNG QUỐC TẾ BACHKHOA-APTECH

Thank for watching!

