Database

Tahaluf Training Center 2021

Day 9

1. **SQL SELECT INTO**
2. Type of Errors
3. Error Handling in SQL Server
4. SQL Server Error functions
5. SQL Schema VS Views

SQL SELECT INTO

The `SELECT INTO` statement copies data from one table into a new table.

The new table will be created with the column-names and types as defined in the old table. You can create new column names using the AS clause.

```
SELECT column1, column2, column3, ...
INTO newtable
FROM oldtable
WHERE condition;
```

SQL SELECT INTO

شركة تحالف الإمارات للحلول التقنية ذ.م.م.
TAHALUF AL EMARAT TECHNICAL SOLUTIONS L.L.C.

```
SELECT * INTO CustomersBackup2017
FROM [Person].[Person];
```

```
SELECT PPro.Color, PPro.ListPrice, PPCH.StartDate,
PPCH.StandardCost
INTO ProductionBackup2020
FROM [Production].[Product] PPro
INNER JOIN [Production].[ProductCostHistory] PPCH
ON PPro.ProductID = PPCH.ProductID;
```

SQL SELECT INTO

SELECT INTO can also be used to create a new, empty table using the schema of another. Just add a WHERE clause that causes the query to return no data:

```
SELECT * INTO newtable
FROM oldtable
WHERE 1 = 0;
```

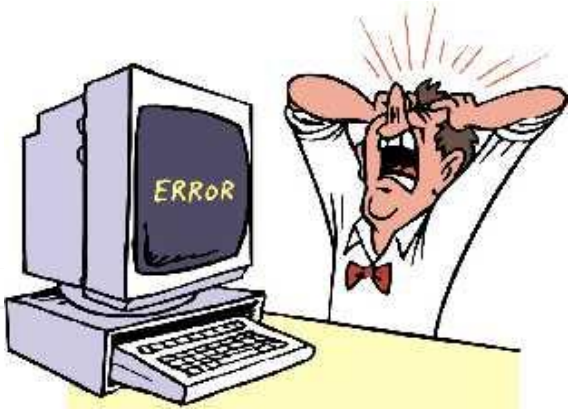
Day 9

1. SQL SELECT INTO
2. **Type of Errors**
3. Error Handling in SQL Server
4. SQL Server Error functions
5. SQL Schema VS Views

Errors

Errors are the mistakes or faults in the program that causes our program to behave unexpectedly and it is no doubt that the well versed and experienced programmers also makes mistakes**.**

No matter how smart or how careful you are, errors are your constant companion. With practice, you will get slightly better at not making errors, and much, much better at finding and correcting them.



There are basically three types of error:

- Compilation error or Syntax error:

Compilation errors are the most common error occurred due to typing mistakes or if you don't follow the proper syntax of the specific programming language. These error are thrown by the compilers and will prevent your program from running. These errors are most common to beginners.

Syntax error

شركة تحالف الإمارات للحلول التقنية ذ.م.م.
TAHALUF AL EMARAT TECHNICAL SOLUTIONS L.L.C.

```
SELECT
*
FROM [HumanResources].[EmployeeDepartmentHistory]
HREmpDep
INNER JOIN [HumanResources].[Employee] HREmp
ON HREmpDep.BusinessEntityID = HREmp.BusinessEntityID
INNER JOIN [HumanResources].[EmployeePayHistory]
HREmpHis
ON HREmpHis.BusinessEntityID = HREmp.BusinessEntityID
WHERE HREmpDep.DepartmentID = 5 AND HREmp.Gender = 'M';
ORDER BY HREmpDep.StartDate DESC
```

- Runtime errors:

Run Time errors are generated when the program is running and leads to the abnormal behavior or termination of the program. The general cause of Run time errors is because your program is trying to perform an operation that is impossible to carry out.

Example: Dividing any number by zero, Accessing any file that doesn't exist etc are common examples of such error.

Runtime errors

شركة تحالف الإمارات للحلول التقنية ذ.م.م.
TAHALUF AL EMARAT TECHNICAL SOLUTIONS L.L.C.

```
DECLARE @Num1 INT;
DECLARE @Num2 INT;
SET @Num1 = 50;
SET @Num2 = 0;
SELECT @Num1 / @Num2 Result;


SELECT COUNT(Size)
FROM [Production].[Product]
WHERE Size = 44;
```

- Logical errors:

A logic error, or **bug**, is when your program compiles and runs, but does the wrong thing. The C# system, of course, has no idea what your program is *supposed* to do, so it provides no additional information to help you find the error.

Ways to track down a logic error include:

- Think about what the program must have done.
- Put in print statements to help you figure out what the program is actually doing.
- Use a **debugger** to step through your program and watch what it does.

Day 9

1. SQL SELECT INTO
2. Type of Errors
3. **Error Handling in SQL Server**
4. SQL Server Error functions
5. SQL Schema VS Views

Error Handling in SQL Server

Here's how error handling in SQL Server works. In SQL Server you can take advantage of TRY...CATCH statements to handle errors. When writing code that handles errors, you should have a TRY block and a CATCH block immediately after it.

```
BEGIN TRY
   -- Write statements here that may cause exception
END TRY
BEGIN CATCH
   -- Write statements here to handle exception
END CATCH
```

When an error occurs inside the TRY block, the control moves to the first statement inside the CATCH block. On the contrary, if the statements inside a TRY block have completed execution successfully without an error, the control will not flow inside the CATCH block. Rather, the first statement immediately after the END CATCH statement will then be executed.

Day 9

1. SQL SELECT INTO
2. Type of Errors
3. Error Handling in SQL Server
4. **SQL Server Error functions**
5. SQL Schema VS Views

SQL Server Error functions

شركة تحالف الإمارات للحلول التقنية ذ.م.م.
TAHALUF AL EMARAT TECHNICAL SOLUTIONS L.L.C.

Inside the CATCH block, you can use the following functions to get the detailed information on the error that occurred:

- ERROR_LINE() returns the line number on which the exception occurred.
- ERROR_MESSAGE() returns the complete text of the generated error message.
- ERROR_PROCEDURE() returns the name of the stored procedure or trigger where the error occurred.

SQL Server Error functions

شركة تحالف الإمارات للحلول التقنية ذ.م.م.
TAHALUF AL EMARAT TECHNICAL SOLUTIONS L.L.C.

- ERROR_NUMBER() returns the number of the error that occurred.
- ERROR_SEVERITY() returns the severity level of the error that occurred.
- ERROR_STATE() returns the state number of the error that occurred.

SQL Server Error functions

```
BEGIN TRY
Update Employee set Salary=19000
Where Emp_IID = 5
END TRY
BEGIN CATCH
SELECT ERROR_NUMBER() AS ErrorNumber;
END CATCH;
GO
```

SQL Server Error functions

شركة تحالف الإمارات للحلول التقنية ذ.م.م.
TAHALUF AL EMARAT TECHNICAL SOLUTIONS L.L.C.

```
CREATE PROCEDURE GetProductsByColorAndSize
    @productColor VARCHAR(20),
    @productSize  INT
AS
BEGIN
    SET NOCOUNT ON;
    SELECT p.Name AS [Product], p.ProductNumber, p.Color,
        p.Size, m.Name AS [Model]
    FROM Production.ProductModel  AS m
    INNER JOIN
    Production.Product AS p ON m.ProductModelID =
        p.ProductModelID
    WHERE (p.Color = @productColor) AND (p.Size = @productSize)
    ORDER BY [Model], [Product]
END
GO
```

SQL Server Error functions

شركة تحالف الإمارات للحلول التقنية ذ.م.م.
TAHALUF AL EMARAT TECHNICAL SOLUTIONS L.L.C.

```
DECLARE @Color VARCHAR(20);
DECLARE @Size INT;
SET @Color = 'Black';
SET @Size = 44;
BEGIN try
    EXEC  GetProductsByColorAndSize
    @productColor = @Color, @productSize = @Size
END try
BEGIN CATCH
    SELECT ERROR_MESSAGE() AS ErrorMessage;
END CATCH
```

THROW AND

THROW { error_number | @local_variable },
{ message | @local_variable },
{ state | @local_variable }]

RAISERROR ({ msg_str | @local_variable } { ,severity ,state }

SQL Server Error functions

```
Declare @studentID INT
Declare @courseID INT
Declare @Register INT

Set @studentID = 2;
Set @courseID = 2;
BEGIN TRY
Set @Register = (SELECT Count(*) FROM
[dbo].[StudentCourse]
where [StudentID] = @studentID AND [CourseID] =
@courseID)
PRINT(@Register)
IF (@Register = 0)
BEGIN
PRINT 'IF STATEMENT: CONDITION IS TRUE'
INSERT INTO [dbo].[StudentCourse] VALUES (@studentID ,
@courseID);
END
```

```
ELSE
BEGIN
PRINT 'ELSE STATEMENT: CONDITION IS FALSE'
RAISERROR('This is RAISERROR Test',16,1)
END
END TRY
BEGIN CATCH
   PRINT 'The student is already register'
END CATCH
```

Day 9

1. SQL SELECT INTO
2. Type of Errors
3. Error Handling in SQL Server
4. SQL Server Error functions
5. **SQL Schema VS Views**

SQL Views Statement

شركة تحالف الإمارات للحلول التقنية ذ.م.م.
TAHALUF AL EMARAT TECHNICAL SOLUTIONS L.L.C.

- In SQL, a view is a virtual table based on the result-set of an SQL statement.
- A view contains rows and columns, just like a real table. The fields in a view are fields from one or more real tables in the database.
- You can add SQL statements and functions to a view and present the data as if the data were coming from one single table.

CREATE VIEW Syntax


```
CREATE VIEW view_name AS
SELECT column1, column2, ...
FROM table_name
WHERE condition;
```

```
CREATE VIEW hiredate_view
AS
SELECT p.FirstName, p.LastName, e.BusinessEntityID,
e.HireDate
FROM HumanResources.Employee e
INNER JOIN Person.Person AS p
ON e.BusinessEntityID = p.BusinessEntityID ;
GO
```

CREATE VIEW Syntax


```
CREATE VIEW  DepartmentInfo
AS
SELECT Emp.Gender,Emp.BirthDate , Emp.NationalIDNumber
FROM [HumanResources].[Employee] AS  Emp
INNER JOIN [HumanResources].[EmployeeDepartmentHistory] AS
EmpDep
ON Emp.BusinessEntityID = EmpDep.BusinessEntityID
INNER JOIN [HumanResources].[Department] AS Dep
ON Dep.DepartmentID = EmpDep.BusinessEntityID
WHERE Dep.Name = 'Sales';
```

SQL Updating a View

```
CREATE OR REPLACE VIEW view_name AS
SELECT column1, column2, ...
FROM table_name
WHERE condition;

ALTER VIEW view_name AS
SELECT column1, column2, ...
FROM table_name
WHERE condition;
```

SQL Updating a View


```
ALTER VIEW HumanResources.EmployeeHireDate
AS
SELECT p.FirstName, p.LastName, e.HireDate
FROM HumanResources.Employee AS e JOIN Person.Person AS p
ON e.BusinessEntityID = p.BusinessEntityID
WHERE p.FirstName LIKE 'J%' ;
GO
```

```
CREATE OR REPLACE VIEW HumanResources.EmployeeHireDate
AS
SELECT p.FirstName, p.LastName, e.HireDate
FROM HumanResources.Employee AS e JOIN Person.Person AS p
ON e.BusinessEntityID = p.BusinessEntityID
WHERE p.FirstName LIKE 'J%' ;
GO
```

SQL delete View

A view is deleted with the DROP VIEW statement.

```
DROP VIEW IF EXISTS [HumanResources].[EmployeeHireDate]
```

```
DROP VIEW IF EXISTS [HumanResources].[EmployeeHireDate]
```

شركة تحالف الإمارات للحلول التقنية ذ.م.م.
TAHALUF AL EMARAT TECHNICAL SOLUTIONS L.L.C.

On the E-Learning Portal