

Basis Data Lanjut
Pemrograman T-SQL dan Error Handling

Dosen Pengampu: Irsyad Arif Mashudi, S.Kom., M.Kom.



Nama : Maulana Bintang Irfansyah

NIM : 2041720132

Kelas : TI-2H

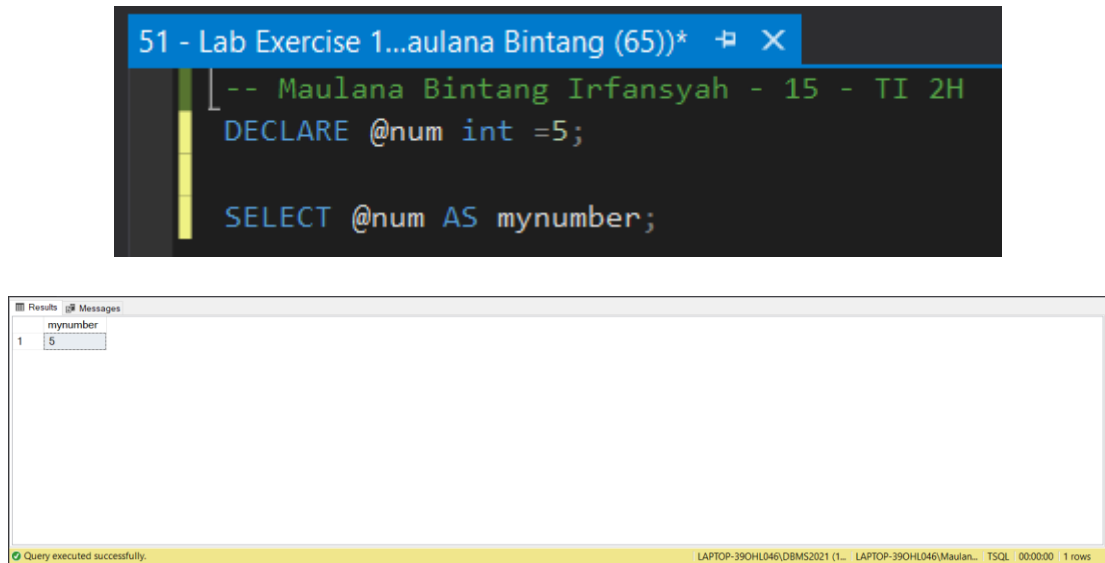
POLITEKNIK NEGERI MALANG
JURUSAN TEKNOLOGI INFORMASI
PROGRAM STUDI D-IV TEKNIK INFORMATIKA
KOTA MALANG

2021

Praktikum 1: DEKLARASI VARIABEL & BATCH: Mendeklarasikan variabel dan mendapatkan nilai variabel

1. Buatlah sebuah kode T-SQL dengan mendeklarasikan sebuah variable bernama @num yang bertipe data integer bernilai 5. Tampilkan nilai variabel tersebut dengan menggunakan alias mynumber lalu eksekusi.

Jawaban:



The screenshot shows a SQL Server query window with the following code:

```
-- Maulana Bintang Irfansyah - 15 - TI 2H
DECLARE @num int =5;

SELECT @num AS mynumber;
```

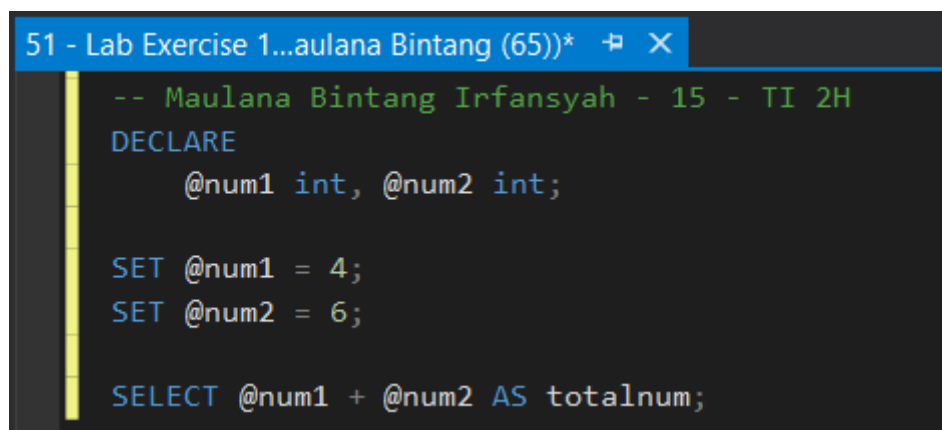
Below the query window, the Results pane displays a single row with the value 5 under the alias mynumber.

mynumber
5

The status bar at the bottom indicates "Query executed successfully."

2. Dari skrip T-SQL [Soal-1] di atas, tambahkan batch delimiter (GO) setelahnya, lalu buatlah skrip T-SQL baru yang mendefinisikan 2 variabel bernama @num1 dan @num2 yang sama-sama bertipe data integer. Set nilainya masing-masing 4 dan 6. Tulis sebuah query SELECT yang menampilkan jumlah kedua variable tersebut sebagai totalnum, lalu eksekusi.

Jawaban:



The screenshot shows a SQL Server query window with the following code:

```
-- Maulana Bintang Irfansyah - 15 - TI 2H
DECLARE
    @num1 int, @num2 int;

SET @num1 = 4;
SET @num2 = 6;

SELECT @num1 + @num2 AS totalnum;
```

The screenshot shows a SQL Server Results window with a single column named 'totalnum'. It contains one row with the value '10'. The status bar at the bottom indicates 'Query executed successfully.' and '1 rows'.

totalnum
10

Praktikum 2: DEKLARASI VARIABEL & BATCH: Memberi nilai terhadap variabel menggunakan query SELECT

3. Buatlah sebuah skrip T-SQL dengan mendefinisikan variabel `@empname` yang bertipe data `nvarchar(30)`. Selanjutnya, set nilai variabel tersebut sebagai hasil query SELECT terhadap tabel **HR.Employees**, yang menggabungkan kolom *firstname* dan *lastname* dengan dipisahkan spasi, dimana nilai *empid*-nya sama dengan 1.

Terakhir, tampilkan nilai variabel `@empname` dengan menggunakan query SELECT dan beri nama alias sebagai `employee`. Eksekusi skrip tersebut.

Jawaban:

The screenshot shows a SQL script in a text editor. The script declares a variable `@empname` of type `nvarchar(30)`, sets its value to the result of a SELECT query that concatenates *firstname* and *lastname* from the **HR.Employees** table where *empid* is 1, and then selects the value of `@empname` with the alias `employee`.

```
-- Maulana Bintang Irfansyah - 15 - TI 2H
DECLARE @empname nvarchar(30);

SET @empname = (SELECT firstname + N' ' + lastname FROM HR.Employees WHERE empid = 1);

SELECT @empname AS employee;
```

The screenshot shows a SQL Server Results window with a single column named 'employee'. It contains one row with the value 'SaraDavis'. The status bar at the bottom indicates 'Query executed successfully.' and '1 rows'.

employee
SaraDavis

4. Apakah yang terjadi apabila query SELECT yang dihasilkan lebih dari 1 baris? Lakukan uji coba misalnya dengan menghilangkan filter WHERE “*empid* = 1”.

Jawaban:

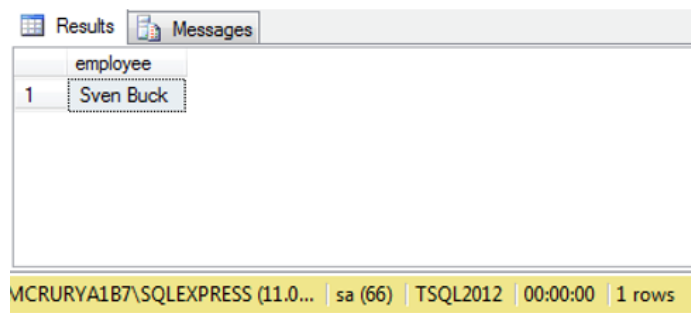
Apabila *query* SELECT yang dihasilkan lebih dari 1 baris akan mendapatkan pesan *error* “*Subquery returned more than 1 value. This is not permitted when the subquery follows =, !=, <, <=, >, >= or when the subquery is used as an expression.*” Karena pernyataan SET diharuskan untuk menggunakan *subquery* scalar untuk *pull data* dari

tabel. Dimana *subquery* scalar akan gagal dijalankan ketika mengembalikan lebih dari satu nilai.

Praktikum 3: DEKLARASI VARIABEL & BATCH: Menggunakan sebuah variabel dalam klausa WHERE

5. Salinlah skrip T-SQL dari [Soal- 3] di atas dan lakukan modifikasi dengan mendefinisikan sebuah variabel baru bernama @empid yang bertipe data integer bernilai 5. Lalu, gunakan variabel baru ini dalam klausa WHERE sebagai nilai dari kolom *empid*. Eksekusi skrip tersebut.

Hasil yang benar ditunjukkan pada tampilan berikut:



employee	
1	Sven Buck

MCRURYA187\SQLEXPRESS (11.0... | sa (66) | TSQL2012 | 00:00:00 | 1 rows

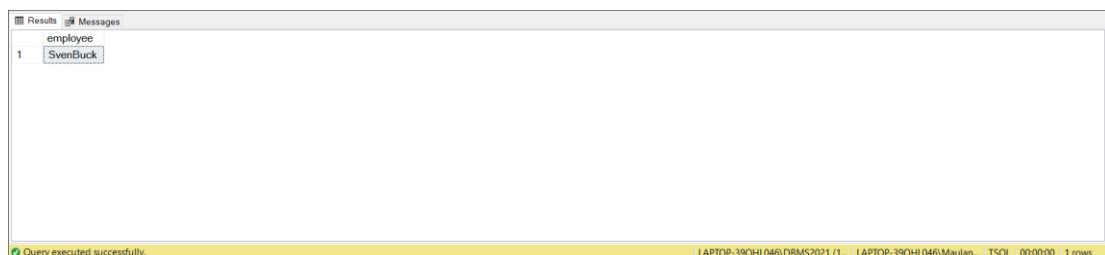
Jawaban:

```
51 - Lab Exercise 1...aulana Bintang (65))* + X
-- Maulana Bintang Irfansyah - 15 - TI 2H
DECLARE
    @empname nvarchar(30),
    @empid int;

SET @empid = 5;

SET @empname = (SELECT firstname + N' ' + lastname FROM HR.Employees
                WHERE empid = @empid);

SELECT @empname AS employee
```



employee	
1	SvenBuck

Query executed successfully. LAPTOP-39OHL046\DBMS2021 (1... LAPTOP-39OHL046\Maulan... TSQL 00:00:00 1 rows

Praktikum 4: DEKLARASI VARIABEL & BATCH: Menambah sebuah batch delimiter

6. Salinlah skrip T-SQL dari [Soal-5] di atas dengan menambahkan batch delimiter GO sebelum query SELECT seperti di bawah ini:

```
...  
GO  
SELECT @empname AS employee;
```

Setelah mengeksekusi skrip tersebut, apakah yang terjadi? Mengapa demikian?

Jawaban:

Setelah menambahkan *batch delimiter* GO sebelum *query* SELECT di atas, yang terjadi adalah *error* yang memberikan pesan “*Must declare the scalar variable “@empname”*”. *Error* tersebut disebabkan karena variabel lokal untuk *batch* yang didefinisikan, dimana jika ingin merujuk ke variabel yang didefinisikan di *batch* lain akan mendapatkan kesalahan bahwa variabel tidak didefinisikan. Informasi tambahan untuk GO merupakan perintah *client* dan bukan perintah server T-SQL.

Praktikum 5: CONTROL OF FLOW: Membuat conditional logic sederhana

7. Tuliskan pernyataan EXECUTE untuk memanggil stored prosedur sys.sp_helptext, kemudian lewatkan stored prosedur Sales.GetTopCustomers yang disimpan sebagai parameter, dengan hasil yang sesuai tampilan di bawah ini!

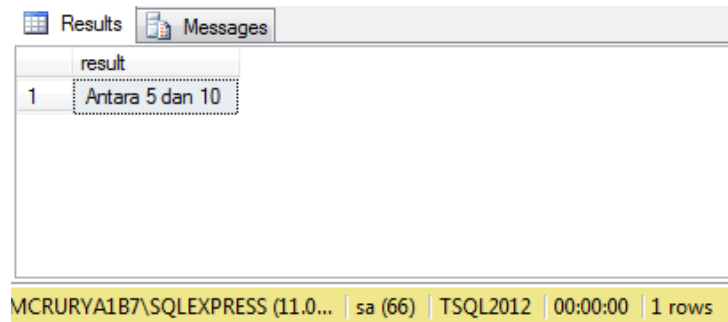
Buatlah sebuah skrip T-SQL dengan mendeklarasikan variabel @result bertipe nvarchar(20) dan variabel @i bertipe integer bernilai 8.

Tambahkan statement IF yang memenuhi *logic* di bawah ini:

- Jika variabel @i bernilai kurang dari 5, set nilai variabel @result menjadi “Kurang dari 5”
- Jika variabel @i bernilai antara 5 dan 10, set nilai variabel @result menjadi “Antara 5 dan 10”
- Jika variabel @i bernilai lebih dari 10, , set nilai variabel @result menjadi “Lebih dari 10”
- Selain dari itu, , set nilai variabel @result menjadi “Unknown”

Di bagian akhir, tambahkan sebuah query SELECT untuk menampilkan nilai variabel @result dengan memberi alias result.

Eksekusi skrip yang sudah dibuat dan bandingkan dengan hasil berikut ini:

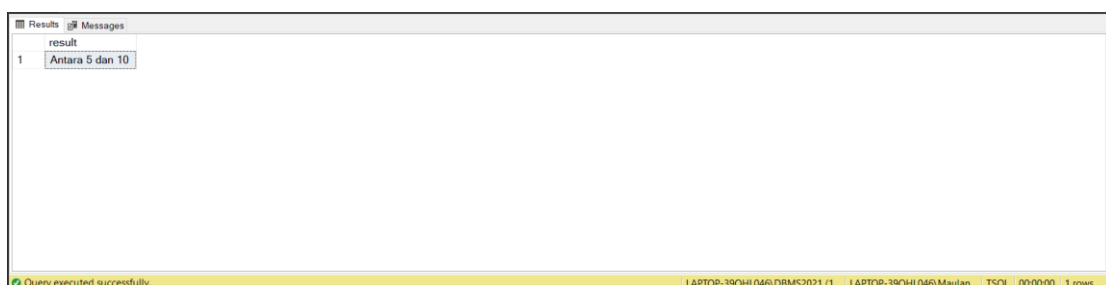


	result
1	Antara 5 dan 10

MCRURYA1B7\SQL EXPRESS (11.0... | sa (66) | TSQL2012 | 00:00:00 | 1 rows

Jawaban:

```
61 - Lab Exercise 2...aulana Bintang (62))*  
-- Maulana Bintang Irfansyah - 15 - TI 2H  
DECLARE  
    @i int = 8,  
    @result nvarchar(20);  
  
IF @i < 5  
    SET @result = N'Kurang dari 5'  
ELSE IF @i <= 10  
    SET @result = N'Antara 5 dan 10'  
ELSE IF @i > 10  
    SET @result = N'Lebih dari 10'  
ELSE  
    SET @result = N'Unknown';  
  
SELECT @result AS result;
```



	result
1	Antara 5 dan 10

Query executed successfully. LAPTOP-39OHL046\DBMS2021 (1... LAPTOP-39OHL046\Maulan... TSQL 00:00:00 1 rows

8. Modifikasilah skrip T-SQL dari [Soal-7] di atas dengan mengganti statement IF menjadi ekspresi CASE dan pastikan hasilnya sama.

Jawaban:

```
61 - Lab Exercise 2...aulana Bintang (62))* ➦ ✕
-- Maulana Bintang Irfansyah - 15 - TI 2H
DECLARE
    @i int = 8,
    @result nvarchar(20);

SET @result =
CASE
    WHEN @i < 5 THEN
        N'Kurang dari 5'
    WHEN @i <= 10 THEN
        N'Antara 5 dan 10'
    WHEN @i > 10 THEN
        N'Lebih dari 10'
    ELSE
        N'Unknown'
END;

SELECT @result AS result;
```

Results Messages	
result	
1	Antara 5 dan 10

Query executed successfully. LAPTOP-39OHL046\DBMS2021 (1... LAPTOP-39OHL046\Maulan... TSQL 00:00:00 1 rows

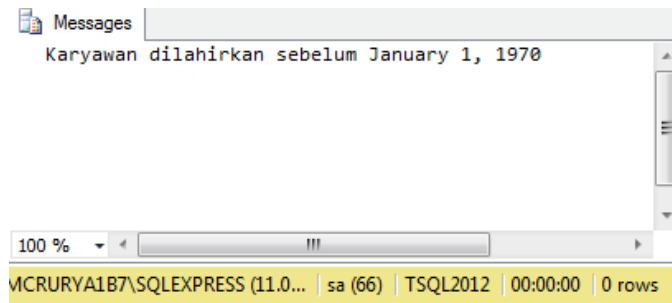
Praktikum 6: CONTROL OF FLOW: Mengecek tanggal lahir karyawan

9. Ikuti langkah berikut ini untuk membuat kode T-SQL yang mengecek tanggal lahir karyawan:

- Pertama, deklarasikan 2 variabel, yakni @birthdate dan @cmpdate (keduanya bertipe data date).
- Set nilai variabel @birthdate sebagai hasil dari query SELECT terhadap kolom birthdate dari tabel **HR.Employees**, dimana empid-nya adalah 5.
- Set variabel @cmpdate berisi tanggal January 1, 1970

- Buatlah pernyataan kondisional IF dengan membandingkan nilai @birthdate dan @cmpdate. Apabila @birthdate lebih kecil dari @cmpdate, gunakan perintah PRINT untuk menampilkan pesan “Karyawan dilahirkan sebelum Januari 1, 1970”. Selain itu, tampilkan pesan “Karyawan dilahirkan pada atau setelah Januari 1, 1970”.
- Eksekusi keseluruhan skrip T-SQL di atas.

Hasil yang benar ditunjukkan pada tampilan berikut:



Jawaban:

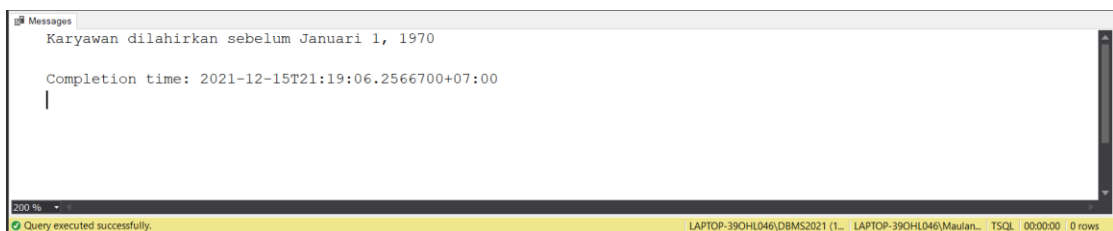
```

61 - Lab Exercise 2...aulana Bintang (62))
-- Maulana Bintang Irfansyah - 15 - TI 2H
DECLARE
    @birthdate date,
    @cmpdate date;

SET @birthdate = (SELECT birthdate FROM HR.Employees WHERE empid = 5);
SET @cmpdate = '19700101';

IF @birthdate < @cmpdate
    PRINT 'Karyawan dilahirkan sebelum Januari 1, 1970'
ELSE
    PRINT 'Karyawan dilahirkan pada atau setelah Januari 1, 1970';

```

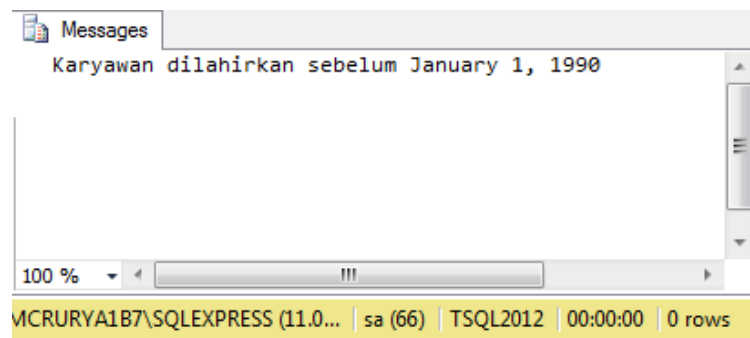


Praktikum 7: CONTROL OF FLOW: Membuat dan mengeksekusi stored procedure

10. Stored procedure bernama **Sales.CheckPersonBirthDate** pada Langkah 1 di atas mempunyai 2 parameter, yakni @empid (untuk menentukan ID karyawan) dan @cmpdate (untuk perbandingan tanggal).

Lakukan perintah EXECUTE pada stored procedure tersebut dengan memasukkan parameter @empid = 3 dan @cmpdate yang di-set ke tanggal January 1, 1990.

Hasil yang benar ditunjukkan pada tampilan berikut:



Jawaban:

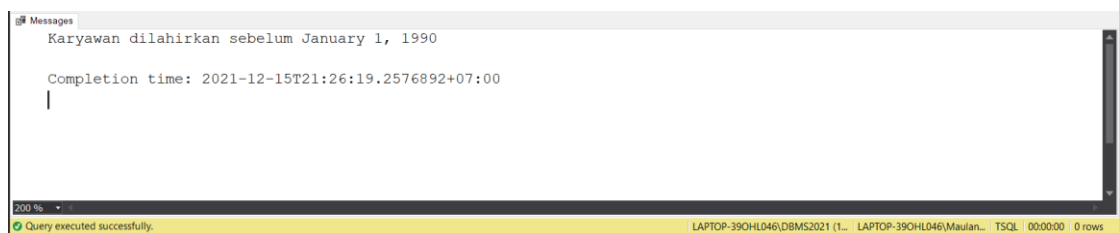
```
61 - Lab Exercise 2...aulana Bintang (62))* ✕
-- Maulana Bintang Irfansyah - 15 - TI 2H
CREATE PROCEDURE Sales.CheckPersonBirthDate
    @empid int,
    @cmpdate date
AS
DECLARE
    @birthdate date;

SET @birthdate = (SELECT birthdate FROM HR.Employees WHERE empid = @empid);

IF @birthdate < @cmpdate
    PRINT 'Karyawan dilahirkan sebelum ' + FORMAT(@cmpdate, 'MMMM d, yyyy', 'en-US')
ELSE
    PRINT 'Karyawan dilahirkan pada atau setelah ' + FORMAT(@cmpdate, 'MMMM d, yyyy', 'en-US');

GO

-- Maulana Bintang Irfansyah - 15 - TI 2H
EXECUTE Sales.CheckPersonBirthDate @empid = 3, @cmpdate = '19900101'; ✓
```



Praktikum 8: CONTROL OF FLOW: Melakukan loop/ pengulangan menggunakan pernyataan WHILE

11. Buatlah sebuah skrip T-SQL yang berisi looping/ pengulangan dengan mengikuti langkah berikut:

- Pertama, deklarasikan sebuah variabel @i yang bertipe data integer bernilai 1
- Lalu buatlah sebuah pengulangan dengan menggunakan pernyataan WHILE, dimana selama nilai variabel @i kurang dari 10, tampilkan/ cetak variabel @i dan tambahkan nilai @i secara incremental dengan menambah 1 (@i+1).

Hasil yang benar ditunjukkan pada tampilan berikut:



Jawaban:

```
61 - Lab Exercise 2...aulana Bintang (62)) X
-- Maulana Bintang Irfansyah - 15 - TI 2H
DECLARE @i int = 1;

WHILE @i <= 10
BEGIN
    PRINT @i;
    SET @i = @i + 1;
END;
```

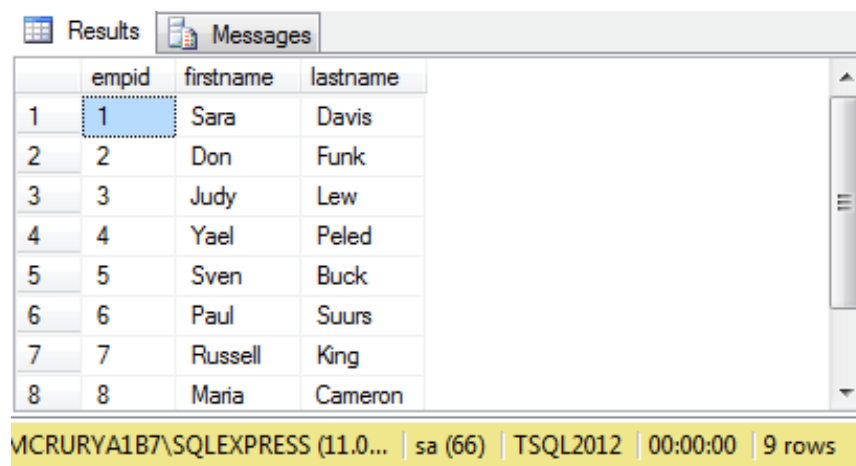


Praktikum 9: DYNAMIC SQL: Membuat SQL dinamis tanpa parameter

12. Buatlah skrip T-SQL dengan mendeklarasikan variabel bernama @SQLstr bertipe data `nvarchar(200)`. Lalu, isikan dengan string berupa statement SELECT yang mengambil kolom `empid`, `firstname`, dan `lastname` dari tabel **HR.Employees**.

- **Perhatikan** bahwa yang diambil adalah string statement-nya, bukan hasil query-nya. Contoh: `SET @SQLstr = N'SELECT....';`
Bukan `SET @SQLstr = (SELECT....);`
- Lakukan perintah `EXECUTE` yang mengeksekusi pernyataan query yang sudah di-set dalam variabel @SQLstr. (Petunjuk: gunakan `sp_executesql`)

Hasil yang benar ditunjukkan pada tampilan berikut:



	empid	firstname	lastname
1	1	Sara	Davis
2	2	Don	Funk
3	3	Judy	Lew
4	4	Yael	Peled
5	5	Sven	Buck
6	6	Paul	Suurs
7	7	Russell	King
8	8	Maria	Cameron

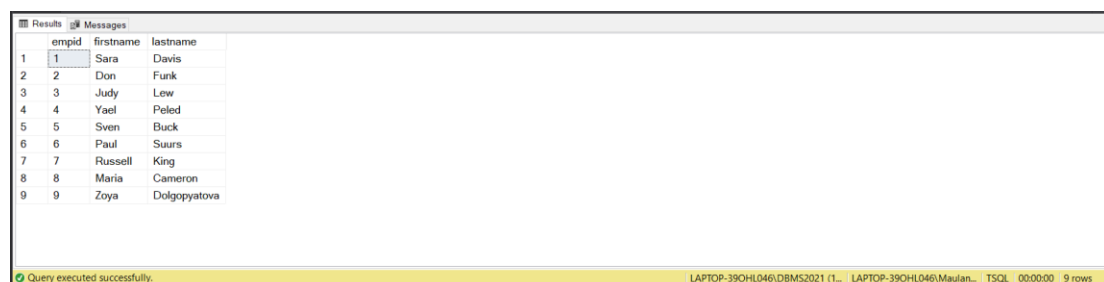
MCRURYA1B7\SQLEXPRESS (11.0... | sa (66) | TSQL2012 | 00:00:00 | 9 rows

Jawaban:

```
71 - Lab Exercise 3...aulana Bintang (52))* X
-- Maulana Bintang Irfansyah - 15 - TI 2H
DECLARE @SQLstr nvarchar(200);

SET @SQLstr = N'SELECT empid, firstname, lastname FROM HR.Employees';

EXECUTE sys.sp_executesql @statement = @SQLstr;
```



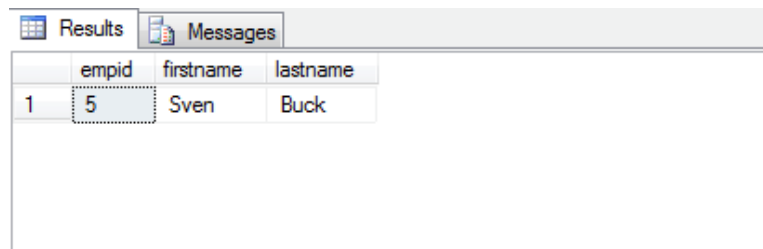
	empid	firstname	lastname
1	1	Sara	Davis
2	2	Don	Funk
3	3	Judy	Lew
4	4	Yael	Peled
5	5	Sven	Buck
6	6	Paul	Suurs
7	7	Russell	King
8	8	Maria	Cameron
9	9	Zoya	Dolgopyatova

Query executed successfully. LAPTOP-39OHL046\DBMS2021 (1... LAPTOP-39OHL046\Maulan... TSQL 00:00:00 9 rows

13. Salinlah skrip T-SQL dari [Soal-12] di atas, lakukan modifikasi dengan mengikuti langkah berikut:

- Tambahkan filter WHERE dimana `empid = @empid` pada string pernyataan SELECT yang disimpan pada variabel `@SQLstr`
- Deklarasikan variabel baru bernama `@SQLParam` bertipe data `nvarchar(100)` untuk menyimpan string definisi dari `@empid` yang bertipe data int
- Seperti halnya pada [Soal-13], lakukan perintah EXECUTE dengan menambahkan parameter `@empid = 5`

Hasil yang benar ditunjukkan pada tampilan berikut:



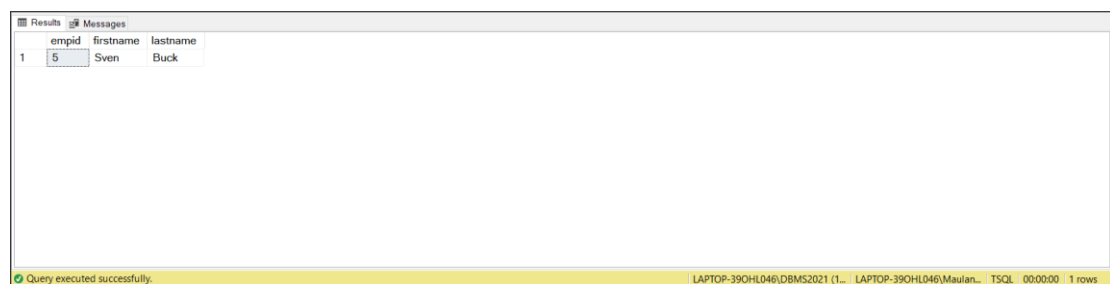
	empid	firstname	lastname
1	5	Sven	Buck

Jawaban:

```
71 - Lab Exercise 3...aulana Bintang (52))* X
-- Maulana Bintang Irfansyah - 15 - TI 2H
DECLARE
    @SQLstr nvarchar(200), @SQLparam nvarchar(100);

SET @SQLstr = N'SELECT empid, firstname, lastname FROM HR.Employees
               WHERE empid = @empid';
SET @SQLparam = N'@empid int';

EXECUTE sys.sp_executesql @statement = @SQLstr, @params = @SQLparam, @empid = 5;
```



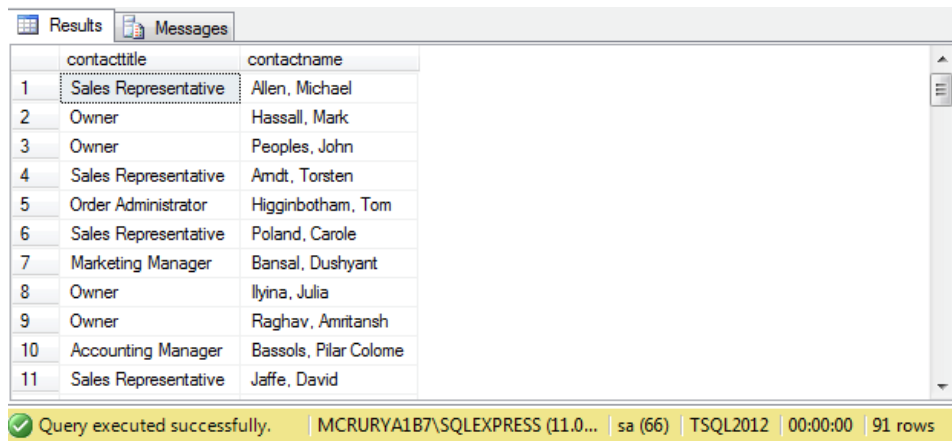
	empid	firstname	lastname
1	5	Sven	Buck

Query executed successfully. LAPTOP-39OHL046\DBMS2021 (1... LAPTOP-39OHL046\Maulan... TSQL 00:00:00 1 rows

14. Tulislah sebuah skrip T-SQL yang membuat SYNONYM bernama **dbo.Pelanggan** dari tabel **Sales.Customers** dalam database TSQL. Eksekusi skrip tersebut.

Kemudian, buatlah query SELECT terhadap synonym **dbo.Pelanggan** yang mengambil kolom *contacttitle* dan *contactname*. Eksekusi kembali skrip T-SQL ini.

Hasil yang benar ditunjukkan pada tampilan berikut:



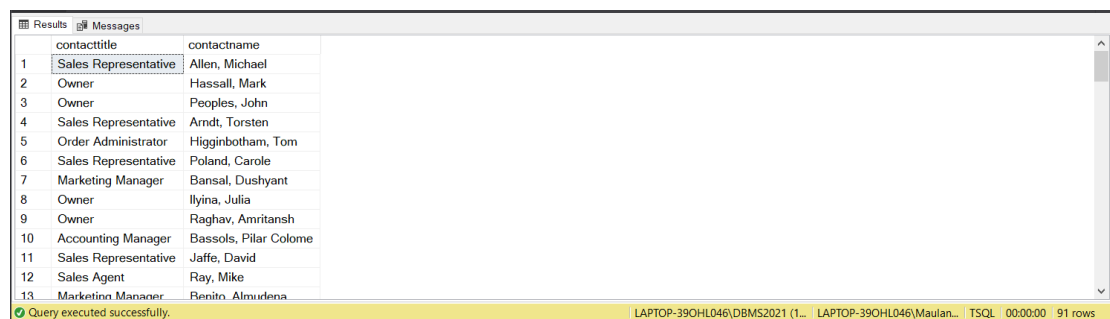
	contacttitle	contactname
1	Sales Representative	Allen, Michael
2	Owner	Hassall, Mark
3	Owner	Peoples, John
4	Sales Representative	Amdt, Torsten
5	Order Administrator	Higginbotham, Tom
6	Sales Representative	Poland, Carole
7	Marketing Manager	Bansal, Dushyant
8	Owner	Ilyina, Julia
9	Owner	Raghav, Amritansh
10	Accounting Manager	Bassols, Pilar Colome
11	Sales Representative	Jaffe, David

Query executed successfully. | MCRURYA1B7\SQLEXPRESS (11.0... | sa (66) | TSQL2012 | 00:00:00 | 91 rows

Jawaban:

```
81 - Lab Exercise 4...aulana Bintang (52))*
-- Maulana Bintang Irfansyah - 15 - TI 2H
CREATE SYNONYM dbo.Person
FOR Sales.Customers;

SELECT contacttitle, contactname
FROM dbo.Person;
```



	contacttitle	contactname
1	Sales Representative	Allen, Michael
2	Owner	Hassall, Mark
3	Owner	Peoples, John
4	Sales Representative	Arndt, Torsten
5	Order Administrator	Higginbotham, Tom
6	Sales Representative	Poland, Carole
7	Marketing Manager	Bansal, Dushyant
8	Owner	Ilyina, Julia
9	Owner	Raghav, Amritansh
10	Accounting Manager	Bassols, Pilar Colome
11	Sales Representative	Jaffe, David
12	Sales Agent	Ray, Mike
13	Marketing Manager	Benito, Almudena

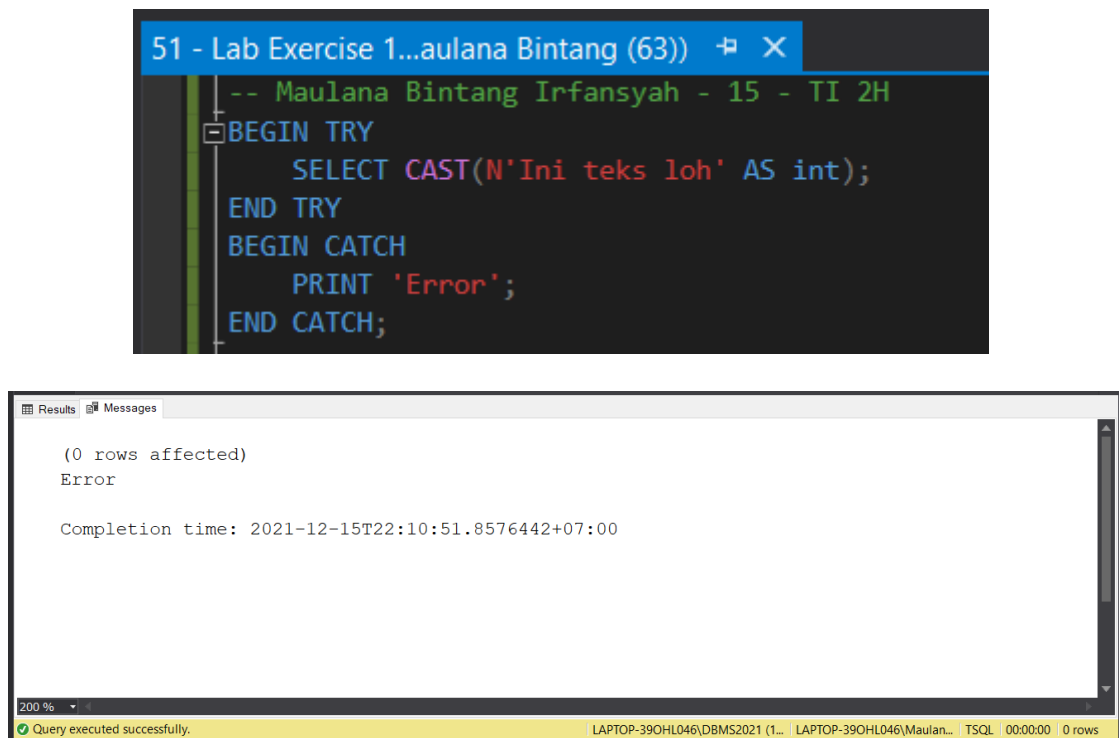
Query executed successfully. | LAPTOP-39OHL046\DBMS2021 (1... | LAPTOP-39OHL046\Maulan... | TSQL | 00:00:00 | 91 rows

Praktikum 11: TRY / CATCH: Membuat blok TRY / CATCH sederhana

15. Buatlah konstruksi TRY / CATCH dengan menempatkan query pada Langkah 1 di atas dalam blok TRY. Sedangkan dalam blok CATCH, isikan perintah untuk menampilkan teks “Error”. Jalankan skrip T-SQL tersebut, lalu pada tab Messages, akan menampilkan pesan:

```
(0 row(s) affected)
Error
```

Jawaban:



The screenshot shows a SQL Server environment. The top window is titled "51 - Lab Exercise 1...aulana Bintang (63))" and contains the following T-SQL script:

```
-- Maulana Bintang Irfansyah - 15 - TI 2H
BEGIN TRY
    SELECT CAST(N'Ini teks loh' AS int);
END TRY
BEGIN CATCH
    PRINT 'Error';
END CATCH;
```

The bottom window shows the "Messages" tab with the following output:

```
(0 rows affected)
Error

Completion time: 2021-12-15T22:10:51.8576442+07:00
```

The status bar at the bottom indicates "Query executed successfully." and shows the file path "LAPTOP-39OHL046\DBMS2021 (1... LAPTOP-39OHL046\Maulan... TSQL 00:00:00 0 rows".

Praktikum 12: TRY / CATCH: Menampilkan kode & pesan error

16. Buatlah konstruksi TRY / CATCH dengan menempatkan query pada Langkah 1 di atas dalam blok TRY. Sedangkan dalam blok CATCH, isikan perintah untuk menampilkan teks “Error”. Jalankan skrip T-SQL tersebut, lalu pada tab Messages, akan menampilkan pesan:

Salin dan eksekusi skrip T-SQL berikut ini dan perhatikan hasilnya:

```
DECLARE @num varchar(20) = '0';

BEGIN TRY
    PRINT 5. / CAST(@num AS numeric(10,4));
END TRY
BEGIN CATCH
END CATCH;
```

Pada langkah di atas, jika memperhatikan nilai variabel @num, semestinya dihasilkan error “division by zero”, tetapi nyatanya tidak. Mengapa demikian?

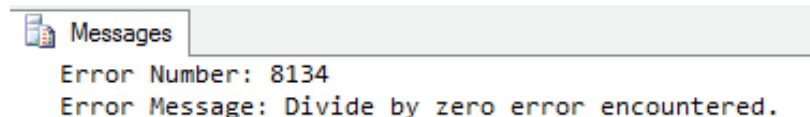
Jawaban:

Pada langkah di atas, tidak menghasilkan *error* karena telah ditangani oleh TRY / CATCH *construct*.

17. Modifikasilah skrip T-SQL pada Langkah 1 di atas dengan menambahkan 2 (dua) pernyataan PRINT pada bagian blok CATCH. Pernyataan yang pertama untuk menampilkan nomer error dengan menggunakan fungsi ERROR_NUMBER dan pernyataan kedua untuk menampilkan pesan error dengan memakai fungsi ERROR_MESSAGE.

Untuk memperjelas, tambahkan string label “Error Number:” pada pesan pertama dan string label “Error Message:” pada pesan kedua.

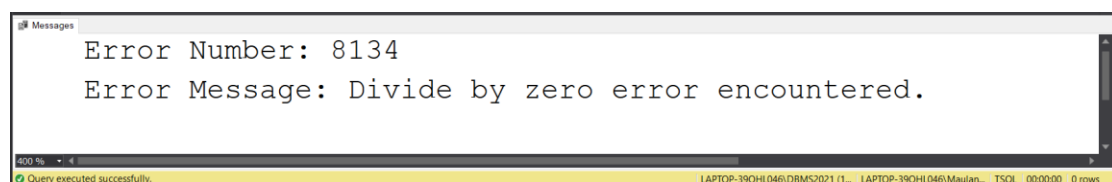
Eksekusi dan bandingkan hasilnya dengan tampilan berikut:



Jawaban:

```
51 - Lab Exercise 1...aulana Bintang (63))* X
-- Maulana Bintang Irfansyah - 15 - TI 2H
DECLARE @num varchar(20) = '0';

BEGIN TRY
    PRINT 5. / CAST(@num AS numeric(10,4));
END TRY
BEGIN CATCH
    PRINT 'Error Number: ' + CAST(ERROR_NUMBER() AS varchar(10));
    PRINT 'Error Message: ' + ERROR_MESSAGE();
END CATCH;
```

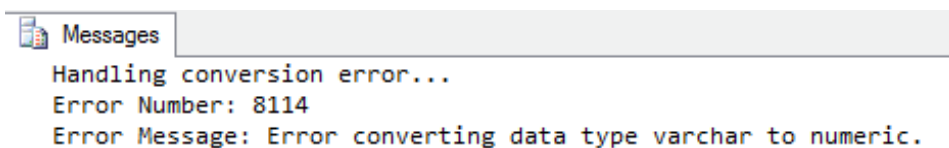


Praktikum 13: TRY / CATCH: Menambahkan conditional logic pada blok CATCH

18. Dengan tetap menggunakan skrip T-SQL pada Bagian 12 Langkah 1, lakukan modifikasi dengan menambahkan pernyataan IF pada bagian blok CATCH sebelum pernyataan PRINT.

- Pernyataan IF tersebut untuk mengecek apakah nomer error = 245 atau 8114.
- Apabila kondisi ini terpenuhi, tampilkan pesan “Handling conversion error...” dengan perintah PRINT.
- Jika tidak sama dengan 245 atau 8114,
- tampilkan pesan “Handling NON conversion error...”.
- Terakhir, set nilai variabel @num sebagai 'A', lalu eksekusi skrip T-SQL tersebut.

Bandingkan hasilnya dengan tampilan berikut:



Jawaban:

```
51 - Lab Exercise 1...aulana Bintang (63))* + X
-- Maulana Bintang Irfansyah - 15 - TI 2H
DECLARE @num varchar(20) = 'A';

BEGIN TRY
    PRINT 5. / CAST(@num AS numeric(10,4));
END TRY
BEGIN CATCH
    IF ERROR_NUMBER() IN (245, 8114)
    BEGIN
        PRINT 'Handling conversion error...'
    END
    ELSE
    BEGIN
        PRINT 'Handling non-conversion error...';
    END;
    PRINT 'Error Number: ' + CAST(ERROR_NUMBER() AS varchar(10));
    PRINT 'Error Message: ' + ERROR_MESSAGE();
END CATCH;
```



```
Messages
Handling conversion error...
Error Number: 8114
Error Message: Error converting data type varchar to numeric.

364 %
Query executed successfully. LAPTOP-39OHL046\DBMS2021 (1... LAPTOP-39OHL046\Maulan... TSQL 00:00:00 0 rows
```

Praktikum 14: TRY / CATCH: Mengeksekusi stored procedure pada blok CATCH

19. Buatlah sebuah konstruksi TRY / CATCH, dimana pada bagian blok CATCH, lakukan eksekusi stored procedure yang telah dibuat pada Langkah 1 di atas, lalu jalankan.

```
Messages
Error Number: 8134
Error Message: Divide by zero error encountered.
Error Severity: 16
Error State: 1
Error Line: 4
Error Proc: Not within procedure
```

Jawaban:

```
51 - Lab Exercise 1...aulana Bintang (63))*
-- Maulana Bintang Irfansyah - 15 - TI 2H
CREATE PROCEDURE dbo.GetErrorInfo AS
PRINT 'Error Number: ' + CAST(ERROR_NUMBER() AS varchar(10));
PRINT 'Error Message: ' + ERROR_MESSAGE();
PRINT 'Error Severity: ' + CAST(ERROR_SEVERITY() AS varchar(10));
PRINT 'Error State: ' + CAST(ERROR_STATE() AS varchar(10));
PRINT 'Error Line: ' + CAST(ERROR_LINE() AS varchar(10));
PRINT 'Error Proc: ' + COALESCE(ERROR_PROCEDURE(), 'Not within procedure');

DECLARE @num varchar(20) = '0';

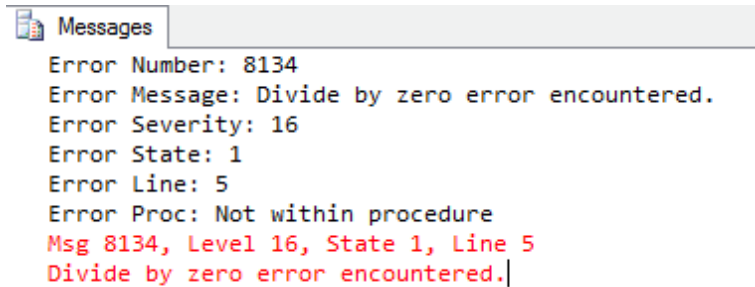
BEGIN TRY
    PRINT 5. / CAST(@num AS numeric(10,4));
END TRY
BEGIN CATCH
    EXECUTE dbo.GetErrorInfo;
END CATCH;
```

```
Messages
Error Number: 8134
Error Message: Divide by zero error encountered.
Error Severity: 16
Error State: 1
Error Line: 4
Error Proc: Not within procedure

200 %
Query executed successfully. LAPTOP-39OHL046\DBMS2021 (1... LAPTOP-39OHL046\Maulan... TSQL 00:00:00 0 rows
```

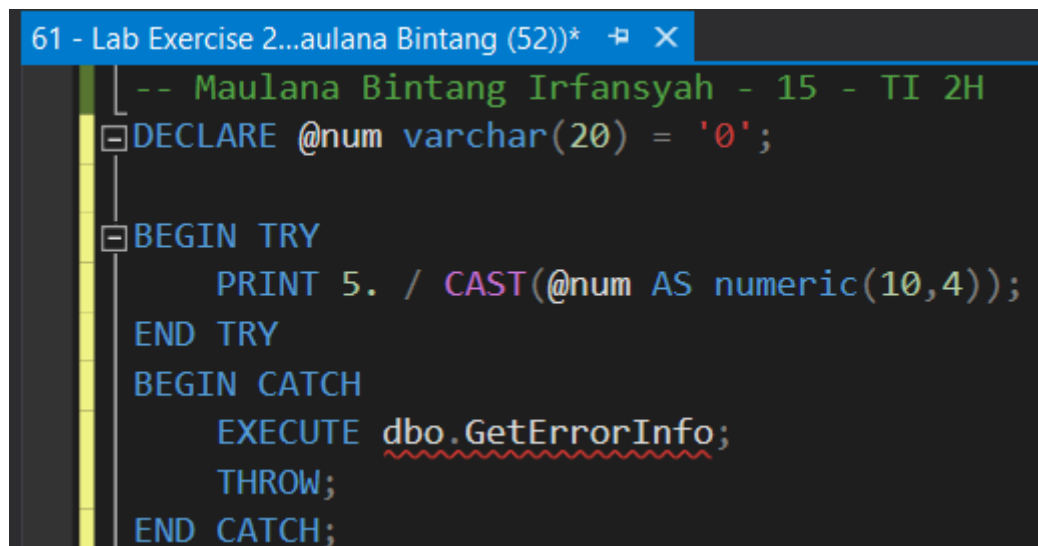
Praktikum 15: THROW: Menggunakan THROW untuk mengirimkan kembali pesan error

20. Modifikasi skrip T-SQL dari Soal-19 di atas dengan menambahkan perintah THROW yang ditempatkan setelah pernyataan eksekusi stored procedure.



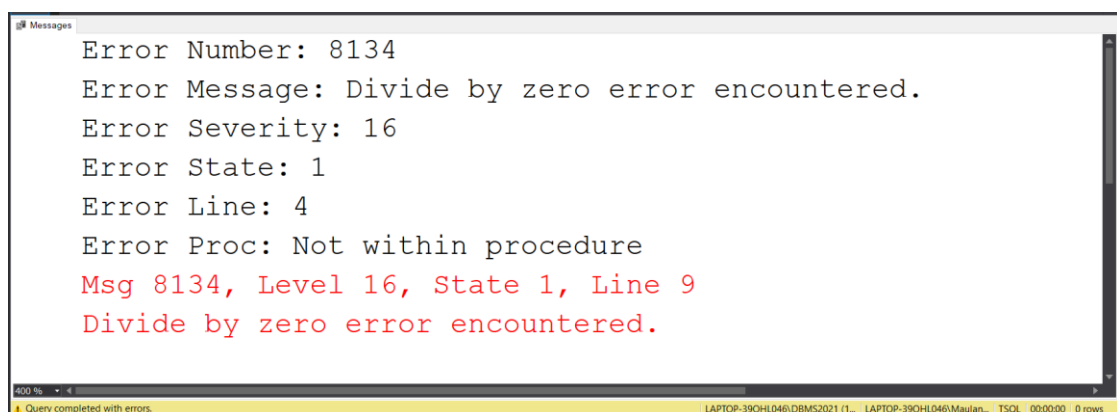
```
Messages
Error Number: 8134
Error Message: Divide by zero error encountered.
Error Severity: 16
Error State: 1
Error Line: 5
Error Proc: Not within procedure
Msg 8134, Level 16, State 1, Line 5
Divide by zero error encountered.
```

Jawaban:



```
61 - Lab Exercise 2...aulana Bintang (52))*
-- Maulana Bintang Irfansyah - 15 - TI 2H
DECLARE @num varchar(20) = '0';

BEGIN TRY
    PRINT 5. / CAST(@num AS numeric(10,4));
END TRY
BEGIN CATCH
    EXECUTE dbo.GetErrorInfo;
    THROW;
END CATCH;
```



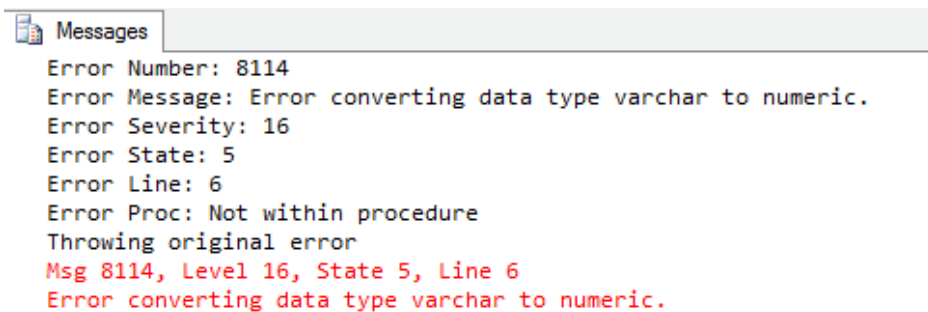
```
Messages
Error Number: 8134
Error Message: Divide by zero error encountered.
Error Severity: 16
Error State: 1
Error Line: 4
Error Proc: Not within procedure
Msg 8134, Level 16, State 1, Line 9
Divide by zero error encountered.

Query completed with errors.
LAPTOP-39OHL046\DBMS2021 (1... LAPTOP-39OHL046\Maulan... TSQL: 00:00:00 0 rows
```

21. Modifikasi skrip T-SQL dari Soal-20 di atas dengan mengganti perintah THROW dengan pernyataan IF.

- Pernyataan IF tersebut untuk mengecek apakah nomer error = 8114.
- Apabila kondisi ini terpenuhi, tampilkan pesan “Handling division by zero...” dengan perintah PRINT.
- Jika tidak, tampilkan pesan “Throwing original error...”.
- Terakhir, set nilai variabel @num sebagai ‘A’, lalu eksekusi skrip T-SQL tersebut.

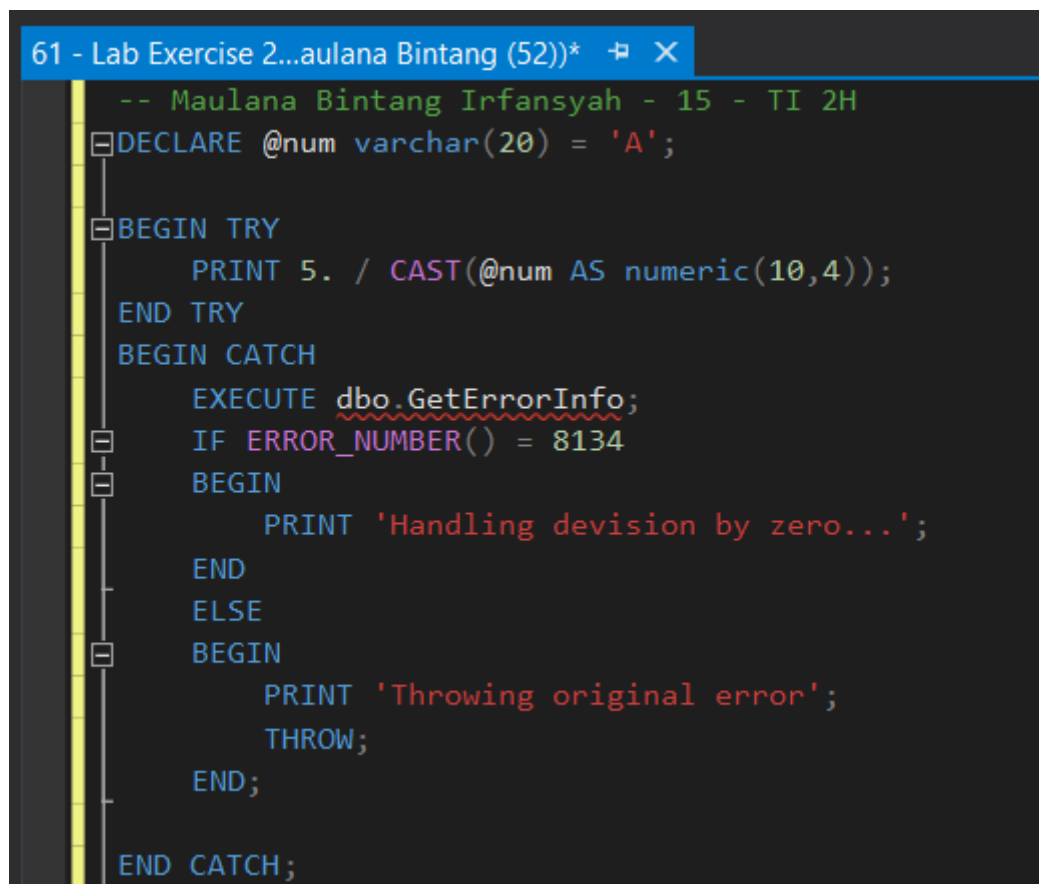
Bandingkan hasilnya dengan tampilan berikut:



The screenshot shows the 'Messages' window in SQL Server. It displays an error message with the following details:

```
Error Number: 8114
Error Message: Error converting data type varchar to numeric.
Error Severity: 16
Error State: 5
Error Line: 6
Error Proc: Not within procedure
Throwing original error
Msg 8114, Level 16, State 5, Line 6
Error converting data type varchar to numeric.
```

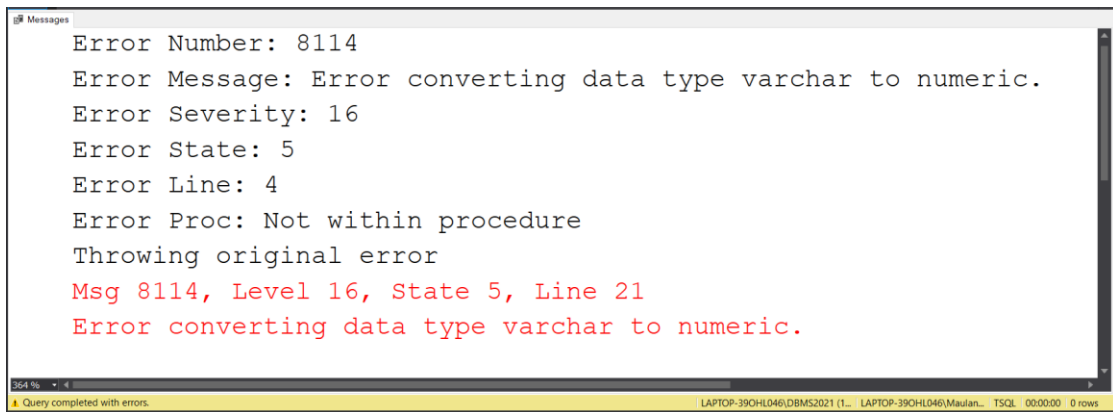
Jawaban:



The screenshot shows a T-SQL script in SQL Server Enterprise Manager. The script is as follows:

```
-- Maulana Bintang Irfansyah - 15 - TI 2H
DECLARE @num varchar(20) = 'A';

BEGIN TRY
    PRINT 5. / CAST(@num AS numeric(10,4));
END TRY
BEGIN CATCH
    EXECUTE dbo.GetErrorInfo;
    IF ERROR_NUMBER() = 8134
    BEGIN
        PRINT 'Handling devision by zero...';
    END
    ELSE
    BEGIN
        PRINT 'Throwing original error';
        THROW;
    END;
END CATCH;
```



```
Messages
Error Number: 8114
Error Message: Error converting data type varchar to numeric.
Error Severity: 16
Error State: 5
Error Line: 4
Error Proc: Not within procedure
Throwing original error
Msg 8114, Level 16, State 5, Line 21
Error converting data type varchar to numeric.
```

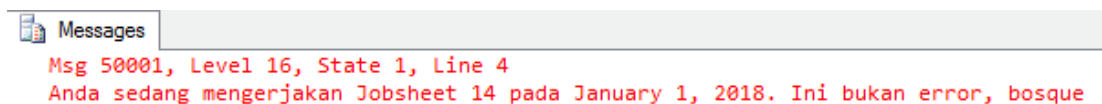
Query completed with errors. LAPTOP-39OHL046\DBMS2021 (1... LAPTOP-39OHL046)\Maulan... TSQL 00:00:00 0 rows

22. Salinlah skrip T-SQL berikut ini:

```
DECLARE @msg AS varchar(2048);
SET @msg = 'Anda sedang mengerjakan Jobsheet 14 pada ' +
FORMAT(CURRENT_TIMESTAMP, 'MMMM d, yyyy', 'en-US') + '. Ini bukan error,
bosque';
```

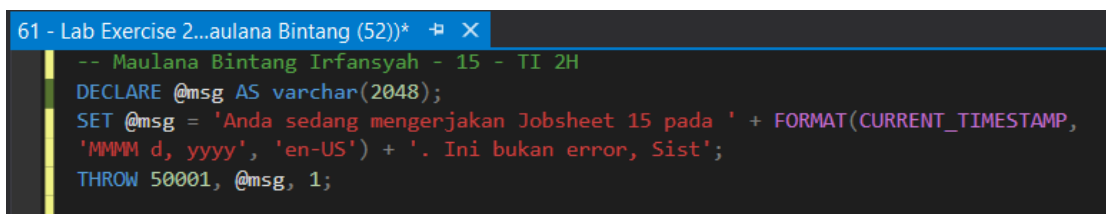
Lalu, tambahkan pernyataan THROW setelah skrip di atas. Sebagai argumen pertama, isikan 50001, untuk argumen kedua, isikan variabel @msg, sedangkan argumen ketiga, isikan nilai 1.

Bandingkan hasilnya dengan tampilan berikut:

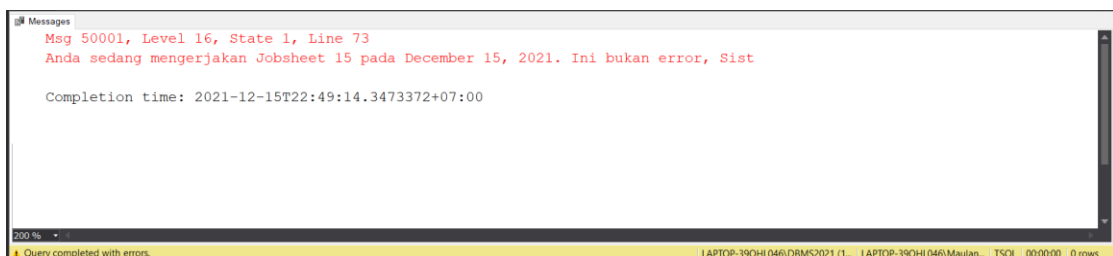


```
Messages
Msg 50001, Level 16, State 1, Line 4
Anda sedang mengerjakan Jobsheet 14 pada January 1, 2018. Ini bukan error, bosque
```

Jawaban:



```
61 - Lab Exercise 2...aulana Bintang (52))* + X
-- Maulana Bintang Irfansyah - 15 - TI 2H
DECLARE @msg AS varchar(2048);
SET @msg = 'Anda sedang mengerjakan Jobsheet 15 pada ' + FORMAT(CURRENT_TIMESTAMP,
'MMMM d, yyyy', 'en-US') + '. Ini bukan error, Sist';
THROW 50001, @msg, 1;
```



```
Messages
Msg 50001, Level 16, State 1, Line 73
Anda sedang mengerjakan Jobsheet 15 pada December 15, 2021. Ini bukan error, Sist

Completion time: 2021-12-15T22:49:14.3473372+07:00

Query completed with errors. LAPTOP-39OHL046\DBMS2021 (1... LAPTOP-39OHL046)\Maulan... TSQL 00:00:00 0 rows
```