(T26)尋找 BlockingProcess(被鎖的流程)
CourseGUID: e48417fc-9db5-4e99-822c-706c5ccef6cc

==========================================================================

(T26)尋找 BlockingProcess(被鎖的流程)

==========================================================================

0. Summary
1. Create Sample Data
2. Display the oldest active transaction information.
4. Clean up

==========================================================================

# 0. Summary

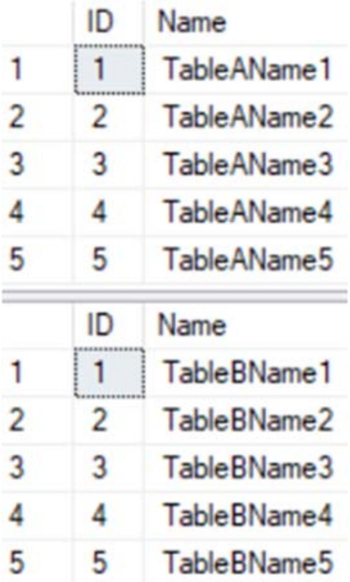
```
--Display all the active transaction information.
SELECT
    [s_tst].[session_id],
    [s_es].[login_name] AS [Login Name],
   DB_NAME (s_tdt.database_id) AS [Database],
    [s_tdt].[database_transaction_begin_time] AS [Begin Time],
    [s_tdt].[database_transaction_log_bytes_used] AS [Log Bytes],
    [s_tdt].[database_transaction_log_bytes_reserved] AS [Log Rsvd],
    [s_est].text AS [Last T-SQL Text],
    [s_eqp].[query_plan] AS [Last Plan]
FROM
   sys.dm_tran_database_transactions [s_tdt]
JOIN
   sys.dm_tran_session_transactions [s_tst]
ON
    [s_tst].[transaction_id] = [s_tdt].[transaction_id]
JOIN
   sys.[dm_exec_sessions] [s_es]
ON
    [s_es].[session_id] = [s_tst].[session_id]
JOIN
   sys.dm_exec_connections [s_ec]
ON
    [s_ec].[session_id] = [s_tst].[session_id]
LEFT OUTER JOIN
   sys.dm_exec_requests [s_er]
ON
    [s_er].[session_id] = [s_tst].[session_id]
CROSS APPLY
   sys.dm_exec_sql_text ([s_ec].[most_recent_sql_handle]) AS [s_est]
OUTER APPLY
   sys.dm_exec_query_plan ([s_er].[plan_handle]) AS [s_eqp]
ORDER BY
    [Begin Time] ASC;
GO -- Run the previous command and begins new batch
```

==============================================

# 1. Create Sample Data

```
--================================================================
--T026_01_01
--Create Sample Data
IF ( EXISTS ( SELECT    *
              FROM      INFORMATION_SCHEMA.TABLES
              WHERE     TABLE_NAME = 'TableA' ) )
    BEGIN
        TRUNCATE TABLE dbo.TableA;
        DROP TABLE TableA;
    END;
GO -- Run the previous command and begins new batch
IF ( EXISTS ( SELECT    *
              FROM      INFORMATION_SCHEMA.TABLES
              WHERE     TABLE_NAME = 'TableB' ) )
    BEGIN
        TRUNCATE TABLE dbo.TableB;
        DROP TABLE TableB;
    END;
GO -- Run the previous command and begins new batch
CREATE TABLE TableA
(
  ID INT IDENTITY
         PRIMARY KEY ,
 Name NVARCHAR(50)
);
       GO -- Run the previous command and begins new batch
INSERT  INTO TableA
VALUES  ( 'TableAName1' );
INSERT  INTO TableA
VALUES  ( 'TableAName2' );
INSERT  INTO TableA
VALUES  ( 'TableAName3' );
INSERT  INTO TableA
VALUES  ( 'TableAName4' );
INSERT  INTO TableA
VALUES  ( 'TableAName5' );
GO -- Run the previous command and begins new batch
CREATE TABLE TableB
(
  ID INT IDENTITY
         PRIMARY KEY ,
 Name NVARCHAR(50)
);
INSERT  INTO TableB
VALUES  ( 'TableBName1' );
INSERT  INTO TableB
VALUES  ( 'TableBName2' );
INSERT  INTO TableB
VALUES  ( 'TableBName3' );
INSERT  INTO TableB
VALUES  ( 'TableBName4' );
INSERT  INTO TableB
VALUES  ( 'TableBName5' );
GO -- Run the previous command and begins new batch
```

```
SELECT  *
FROM    TableA;
SELECT  *
FROM    TableB;
GO -- Run the previous command and begins new batch
```

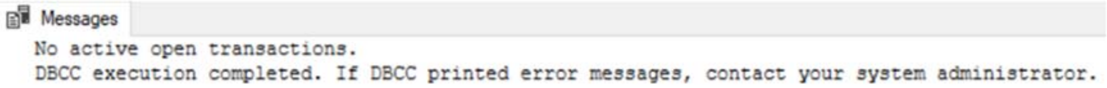


|   | ID | Name |
|---|---|---|
| 1 | 1 | TableAName1 |
| 2 | 2 | TableAName2 |
| 3 | 3 | TableAName3 |
| 4 | 4 | TableAName4 |
| 5 | 5 | TableAName5 |

|   | ID | Name |
|---|---|---|
| 1 | 1 | TableBName1 |
| 2 | 2 | TableBName2 |
| 3 | 3 | TableBName3 |
| 4 | 4 | TableBName4 |
| 5 | 5 | TableBName5 |

# 2. Display the oldest active transaction information.

```
--================================================================
--T026_01_02
--display the oldest active transaction information.
DBCC OPENTRAN;
GO -- Run the previous command and begins new batch
```



```
No active open transactions.
DBCC execution completed. If DBCC printed error messages, contact your system administrator.
```

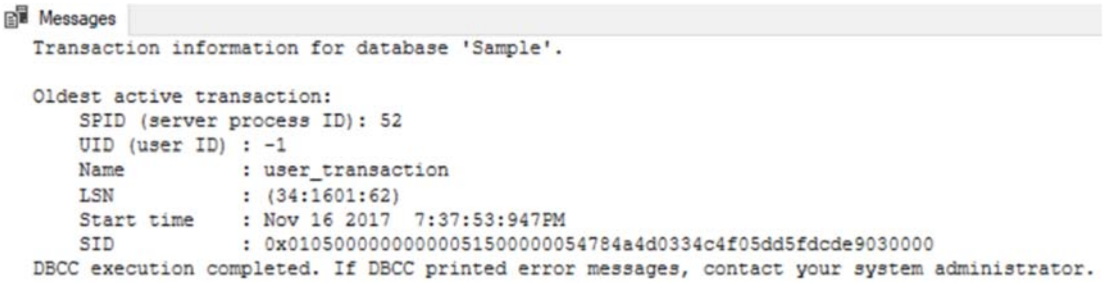
```
/*
Output
--No active open transactions.
--DBCC execution completed. If DBCC printed error messages, contact your system administrator.
*/
```

# 3. Create Blocking Query

```
--================================================================
--T026_01_03
--Create Blocking Query

-------------------------------------------------------------
--T026_01_03_01
--Transaction1
BEGIN TRAN;
UPDATE  TableA
SET     Name += ' Tran1'
WHERE   ID = 1;
--Thus, TableA will be block.
/*
Open another query windows to execute the query.
```

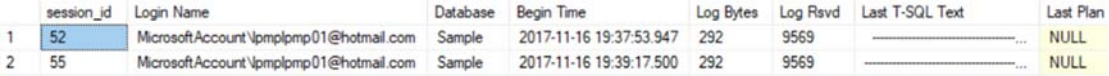
```
Begin a transaction and do not commit.
Thus, TableA will be locked by this uncommited transaction.
All other transaction will not be able to use TableA.
*/
-------------------------------------------------------------
--T026_01_03_02
--Transaction2
--T026_01_03_03_01
SELECT  COUNT(*)
FROM    TableA;
--T026_01_03_03_02
DELETE  FROM TableA
WHERE   ID = 1;
--T026_01_03_03_03
TRUNCATE TABLE TableA;
--T026_01_03_03_04
DROP TABLE TableA;
/*
Open another query windows to execute the query.
Execute 1~4 query in Transaction2 separately.
All these query will be blocked
by the previous uncommited transaction, Transaction1.
TableA has beem locked by Transaction1.
*/
-------------------------------------------------------------
--T026_01_03_03
--Transaction3
--Create Blocking Query
BEGIN TRAN;
UPDATE  dbo.TableB
SET     Name += ' Tran3'
WHERE   ID = 1;
--Thus, TableB will be block.
/*
Open another query windows to execute the query.
Begin another transaction and do not commit.
Thus, TableB will be locked by this uncommited transaction.
All other transaction will not be able to use TableB.
*/
-------------------------------------------------------------
--T026_01_03_04
--Transaction4
--display the oldest active transaction information.
DBCC OPENTRAN;
```



```
Transaction information for database 'Sample'.

Oldest active transaction:
    SPID (server process ID): 52
    UID (user ID) : -1
    Name          : user_transaction
    LSN           : (34:1601:62)
    Start time    : Nov 16 2017  7:37:53:947PM
    SID           : 0x01050000000000051500000054784a4d0334c4f05dd5fdcde9030000
DBCC execution completed. If DBCC printed error messages, contact your system administrator.
```

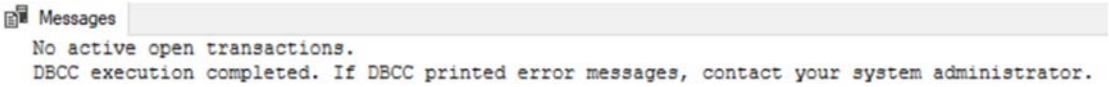
```
--Display all the active transaction information.
SELECT
    [s_tst].[session_id],
    [s_es].[login_name] AS [Login Name],
   DB_NAME (s_tdt.database_id) AS [Database],
    [s_tdt].[database_transaction_begin_time] AS [Begin Time],
    [s_tdt].[database_transaction_log_bytes_used] AS [Log Bytes],
    [s_tdt].[database_transaction_log_bytes_reserved] AS [Log Rsvd],
    [s_est].text AS [Last T-SQL Text],
    [s_eqp].[query_plan] AS [Last Plan]
FROM
    sys.dm_tran_database_transactions [s_tdt]
JOIN
    sys.dm_tran_session_transactions [s_tst]
ON
    [s_tst].[transaction_id] = [s_tdt].[transaction_id]
JOIN
    sys.[dm_exec_sessions] [s_es]
ON
    [s_es].[session_id] = [s_tst].[session_id]
JOIN
    sys.dm_exec_connections [s_ec]
ON
    [s_ec].[session_id] = [s_tst].[session_id]
LEFT OUTER JOIN
    sys.dm_exec_requests [s_er]
ON
    [s_er].[session_id] = [s_tst].[session_id]
CROSS APPLY
    sys.dm_exec_sql_text ([s_ec].[most_recent_sql_handle]) AS [s_est]
OUTER APPLY
    sys.dm_exec_query_plan ([s_er].[plan_handle]) AS [s_eqp]
ORDER BY
    [Begin Time] ASC;
GO -- Run the previous command and begins new batch
```



|   | session_id | Login Name | Database | Begin Time | Log Bytes | Log Rsvd | Last T-SQL Text | Last Plan |
|---|---|---|---|---|---|---|---|---|
| 1 | 52 | MicrosoftAccount\lpmplpmp01@hotmail.com | Sample | 2017-11-16 19:37:53.947 | 292 | 9569 | ------------------------------... | NULL |
| 2 | 55 | MicrosoftAccount\lpmplpmp01@hotmail.com | Sample | 2017-11-16 19:39:17.500 | 292 | 9569 | ------------------------------... | NULL |

```
/*
--DBCC OpenTran
Reference:
https://docs.microsoft.com/en-us/sql/t-sql/database-console-commands/dbcc-opentran-transact-sql
https://www.sqlskills.com/blogs/paul/script-open-transactions-with-text-and-plans/
DBCC OPENTRAN can only display
the oldest active transaction information.
If you want to see all the active transaction information,
You have to use the following script.
--SELECT
--    [s_tst].[session_id],
--    [s_es].[login_name] AS [Login Name],
--    DB_NAME (s_tdt.database_id) AS [Database],
--    [s_tdt].[database_transaction_begin_time] AS [Begin Time],
--    [s_tdt].[database_transaction_log_bytes_used] AS [Log Bytes],
--    [s_tdt].[database_transaction_log_bytes_reserved] AS [Log Rsvd],
--    [s_est].text AS [Last T-SQL Text],
--    [s_eqp].[query_plan] AS [Last Plan]
--FROM
--    sys.dm_tran_database_transactions [s_tdt]
--JOIN
--    sys.dm_tran_session_transactions [s_tst]
--ON
--    [s_tst].[transaction_id] = [s_tdt].[transaction_id]
--JOIN
--    sys.[dm_exec_sessions] [s_es]
--ON
--    [s_es].[session_id] = [s_tst].[session_id]
--JOIN
--    sys.dm_exec_connections [s_ec]
```

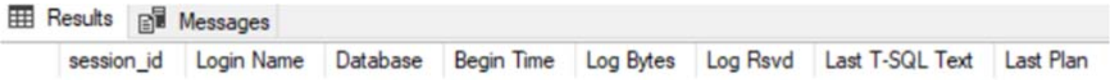
```
--ON
--     [s_ec].[session_id] = [s_tst].[session_id]
--LEFT OUTER JOIN
--     sys.dm_exec_requests [s_er]
--ON
--     [s_er].[session_id] = [s_tst].[session_id]
--CROSS APPLY
--     sys.dm_exec_sql_text ([s_ec].[most_recent_sql_handle]) AS [s_est]
--OUTER APPLY
--     sys.dm_exec_query_plan ([s_er].[plan_handle]) AS [s_eqp]
--ORDER BY
--     [Begin Time] ASC;
--GO
2.1.
Output
--session_id ...
--81...
--60...
*/
-------------------------------------------------------------
--T026_01_03_05
--Go back to the query window of
--the Transaction1:
--and perform COMMIT;
COMMIT;
-------------------------------------------------------------
--T026_01_03_06
--Go back to the query window of
--the Transaction3:
--and perform COMMIT;
COMMIT;
-------------------------------------------------------------
--T026_01_03_07
--Transaction4
--display the oldest active transaction information.
DBCC OPENTRAN;
```



```
No active open transactions.
DBCC execution completed. If DBCC printed error messages, contact your system administrator.
```

```
/*
Output
--No active open transactions.
--DBCC execution completed.
--If DBCC printed error messages,
--contact your system administrator.
*/
--Display all the active transaction information.
SELECT
    [s_tst].[session_id],
    [s_es].[login_name] AS [Login Name],
    DB_NAME (s_tdt.database_id) AS [Database],
    [s_tdt].[database_transaction_begin_time] AS [Begin Time],
    [s_tdt].[database_transaction_log_bytes_used] AS [Log Bytes],
    [s_tdt].[database_transaction_log_bytes_reserved] AS [Log Rsvd],
    [s_est].text AS [Last T-SQL Text],
    [s_eqp].[query_plan] AS [Last Plan]
FROM
    sys.dm_tran_database_transactions [s_tdt]
JOIN
    sys.dm_tran_session_transactions [s_tst]
ON
    [s_tst].[transaction_id] = [s_tdt].[transaction_id]
JOIN
    sys.[dm_exec_sessions] [s_es]
```

```
ON
    [s_es].[session_id] = [s_tst].[session_id]
JOIN
    sys.dm_exec_connections [s_ec]
ON
    [s_ec].[session_id] = [s_tst].[session_id]
LEFT OUTER JOIN
    sys.dm_exec_requests [s_er]
ON
    [s_er].[session_id] = [s_tst].[session_id]
CROSS APPLY
    sys.dm_exec_sql_text ([s_ec].[most_recent_sql_handle]) AS [s_est]
OUTER APPLY
    sys.dm_exec_query_plan ([s_er].[plan_handle]) AS [s_eqp]
ORDER BY
    [Begin Time] ASC;
GO
```



| | session_id | Login Name | Database | Begin Time | Log Bytes | Log Rsvd | Last T-SQL Text | Last Plan |
|---|---|---|---|---|---|---|---|---|

```
/*
--DBCC OpenTran
Reference:
https://docs.microsoft.com/en-us/sql/t-sql/database-console-commands/dbcc-opentran-transact-sql
https://www.sqlskills.com/blogs/paul/script-open-transactions-with-text-and-plans/
DBCC OPENTRAN can only display
the oldest active transaction information.
If you want to see all the active transaction information,
You have to use the following script.
--SELECT
--      [s_tst].[session_id],
--      [s_es].[login_name] AS [Login Name],
--      DB_NAME (s_tdt.database_id) AS [Database],
--      [s_tdt].[database_transaction_begin_time] AS [Begin Time],
--      [s_tdt].[database_transaction_log_bytes_used] AS [Log Bytes],
--      [s_tdt].[database_transaction_log_bytes_reserved] AS [Log Rsvd],
--      [s_est].text AS [Last T-SQL Text],
--      [s_eqp].[query_plan] AS [Last Plan]
--FROM
--      sys.dm_tran_database_transactions [s_tdt]
--JOIN
--      sys.dm_tran_session_transactions [s_tst]
--ON
--      [s_tst].[transaction_id] = [s_tdt].[transaction_id]
--JOIN
--      sys.[dm_exec_sessions] [s_es]
--ON
--      [s_es].[session_id] = [s_tst].[session_id]
--JOIN
--      sys.dm_exec_connections [s_ec]
--ON
--      [s_ec].[session_id] = [s_tst].[session_id]
--LEFT OUTER JOIN
--      sys.dm_exec_requests [s_er]
--ON
--      [s_er].[session_id] = [s_tst].[session_id]
--CROSS APPLY
--      sys.dm_exec_sql_text ([s_ec].[most_recent_sql_handle]) AS [s_est]
--OUTER APPLY
--      sys.dm_exec_query_plan ([s_er].[plan_handle]) AS [s_eqp]
--ORDER BY
--      [Begin Time] ASC;
```

```
--GO
*/
```

# 4. Clean up

```
--==============================================================
--T026_01_04
--Clean up
--If Table exists then DROP it
IF ( EXISTS ( SELECT    *
              FROM      INFORMATION_SCHEMA.TABLES
              WHERE     TABLE_NAME = 'TableA' ) )
    BEGIN
        TRUNCATE TABLE dbo.TableA;
        DROP TABLE TableA;
    END;
GO -- Run the previous command and begins new batch
--clean up
--If Table exists then DROP it
IF ( EXISTS ( SELECT    *
              FROM      INFORMATION_SCHEMA.TABLES
              WHERE     TABLE_NAME = 'TableB' ) )
    BEGIN
        TRUNCATE TABLE dbo.TableB;
        DROP TABLE TableB;
    END;
GO -- Run the previous command and begins new batch
```