

SQL Serverで【K-MEMO】を使ってみる SQLServer2014体験版 or 正規版 Microsoft SQL Server Management Studio

K-MEMO Ver.2016-05-27

ニューロ型データベースモデリング学会

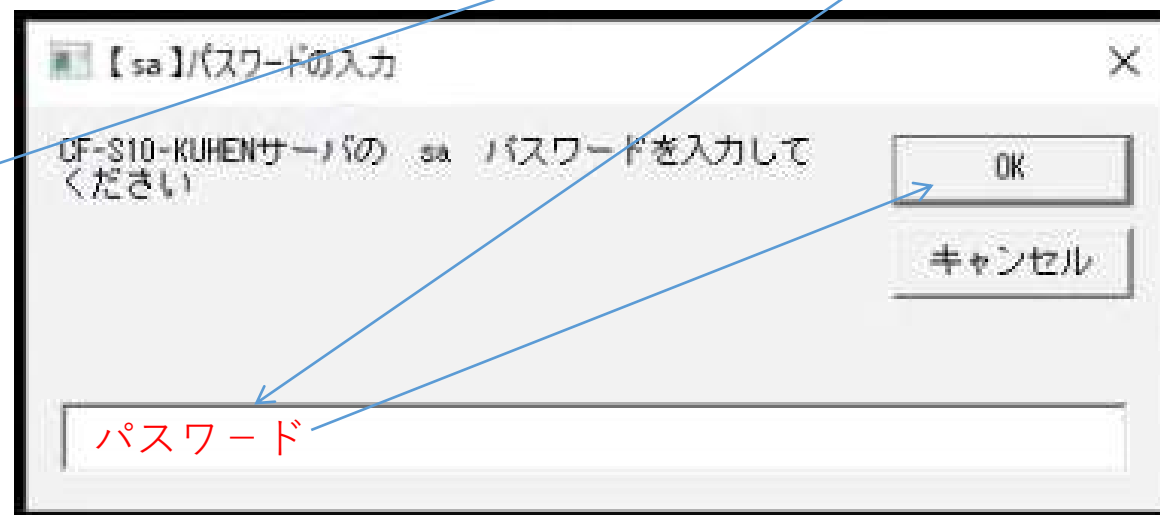
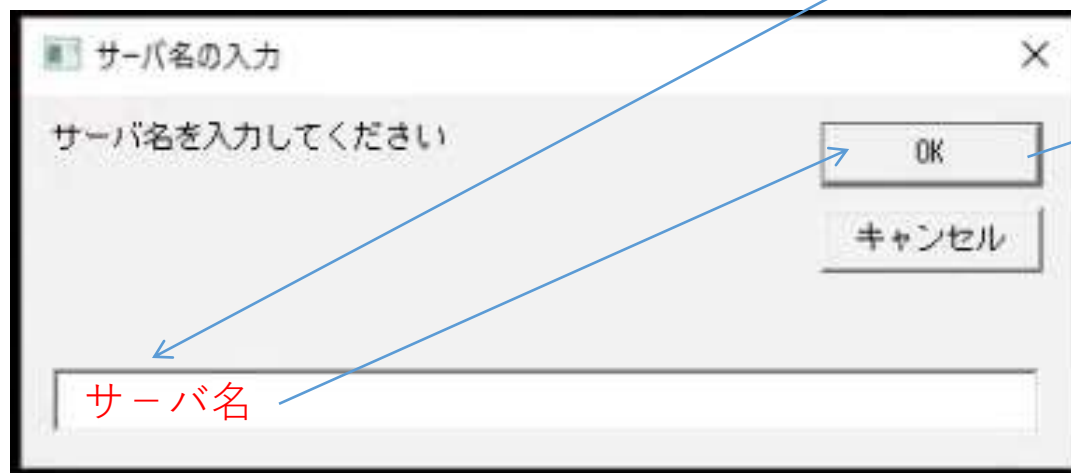
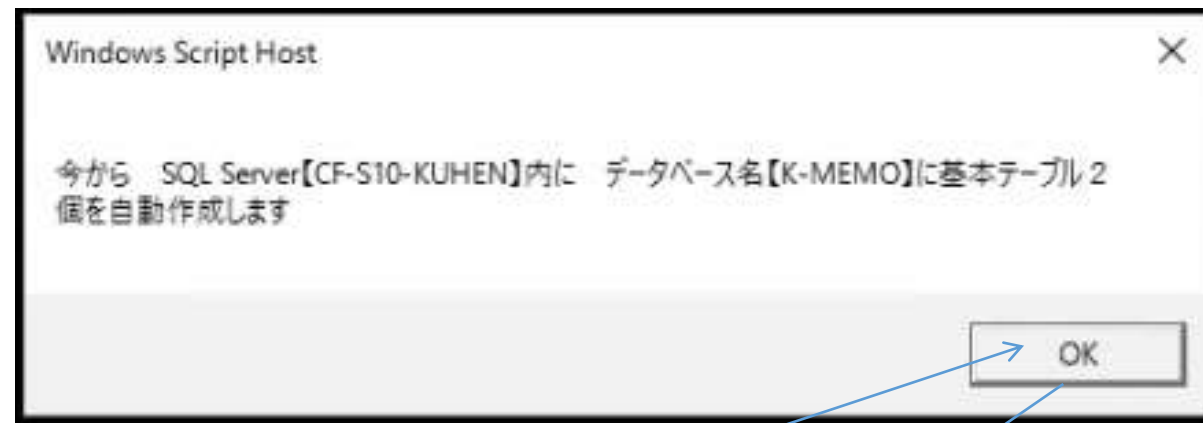
<http://kuhen.jp/ndb>

SKojima@kuhen.jp

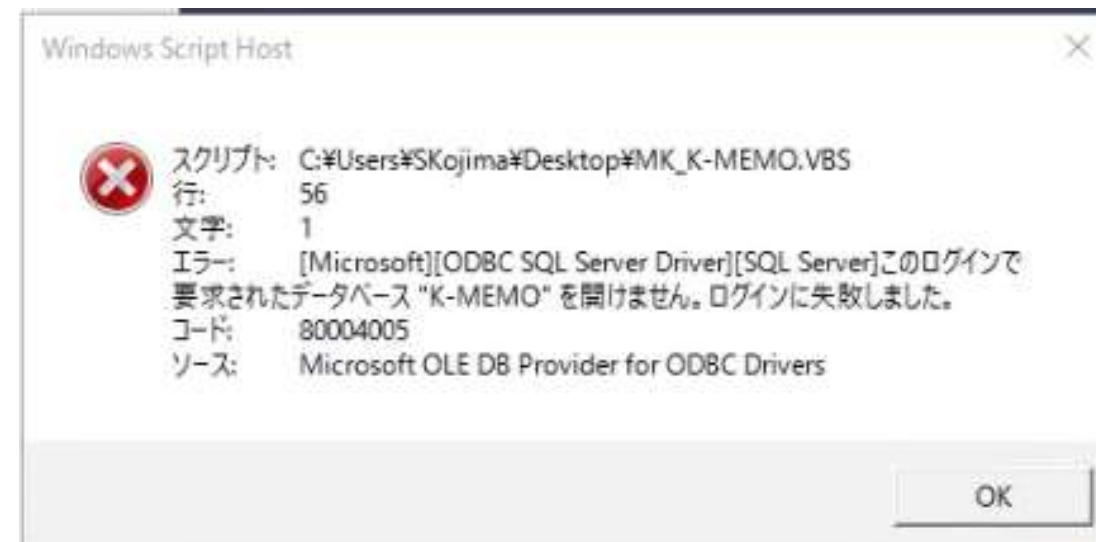
2016-05-27

Microsoft SQL Server Management Studio で【K-MEMO】データベースを作成

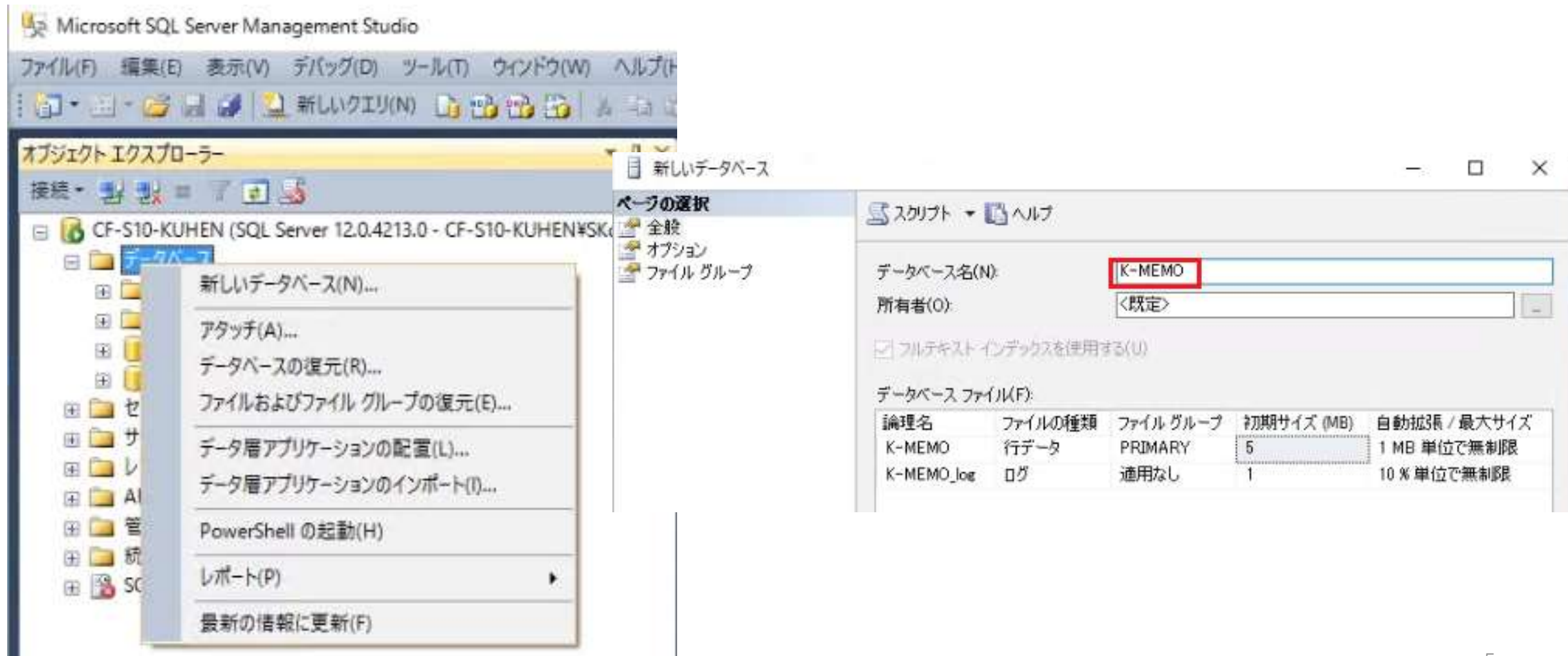
MK_K-MEMO.VBS.TXTを→MK_K-MEMO.VBSに名前変更
(末尾の.TXTを取る) し、マウスでWクリック



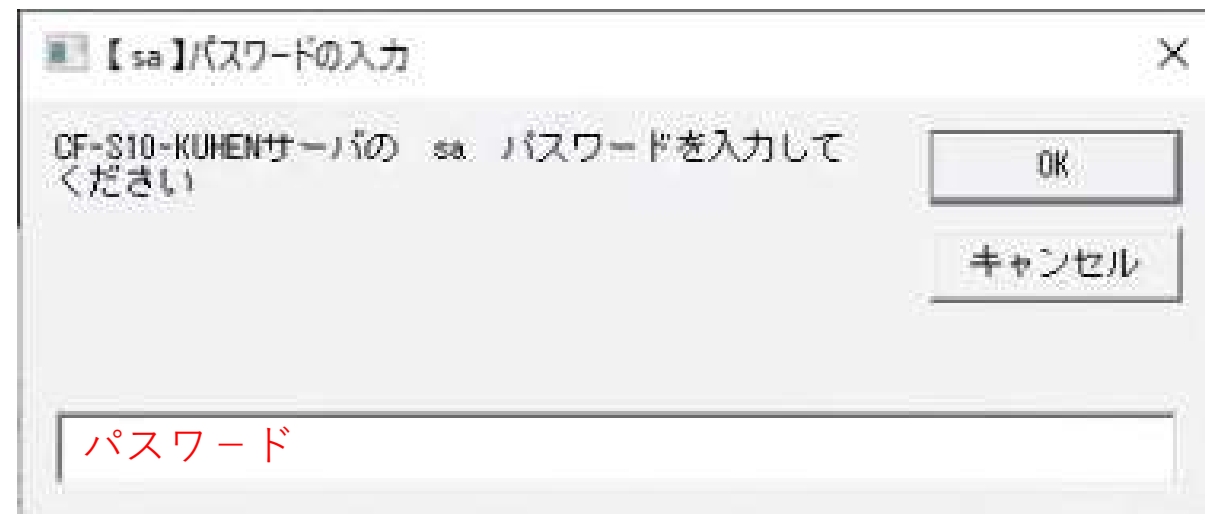
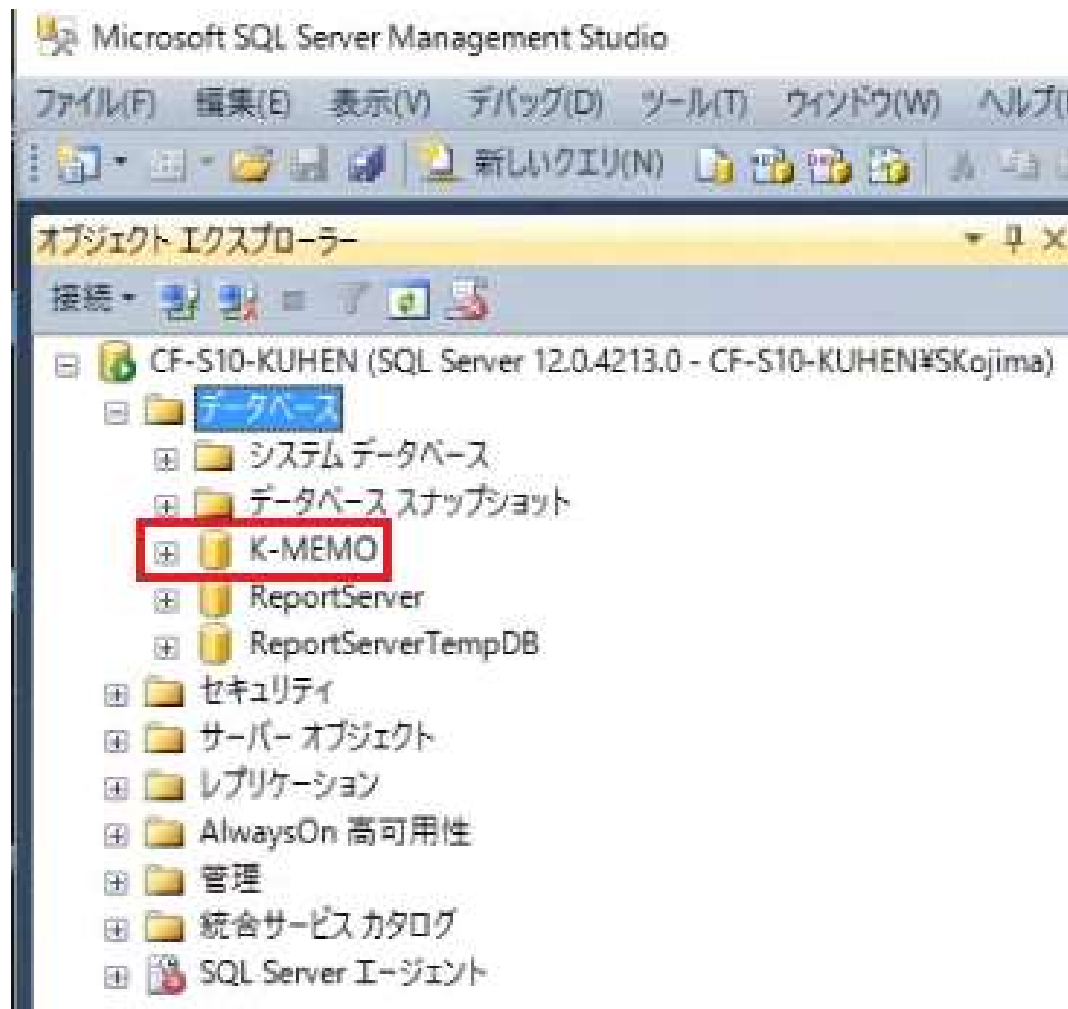
事前に【K-MEMO】データベースを作成しておかないと
以下のエラーで止まる



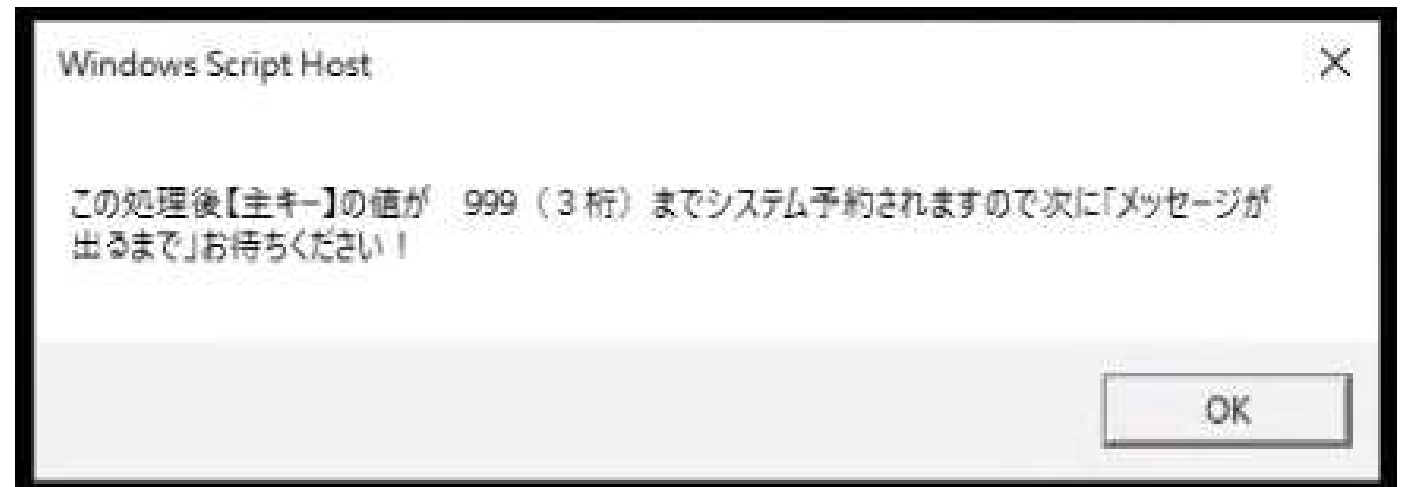
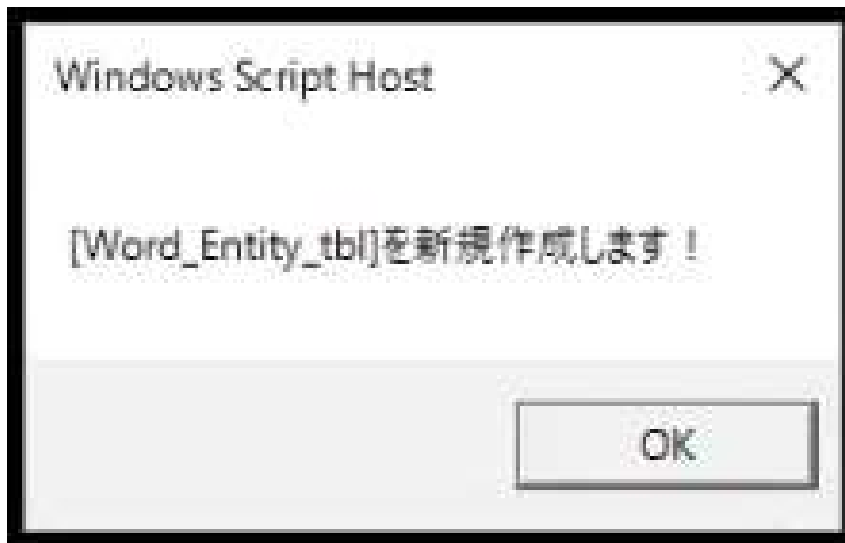
データベース【K-MEMO】を新規作成



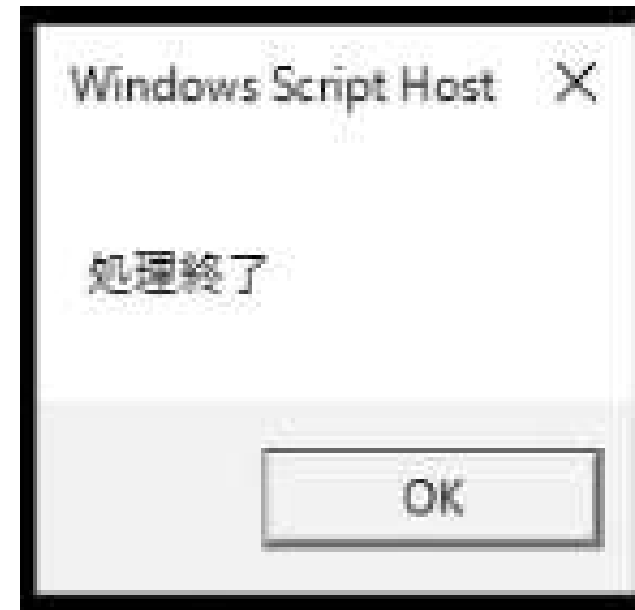
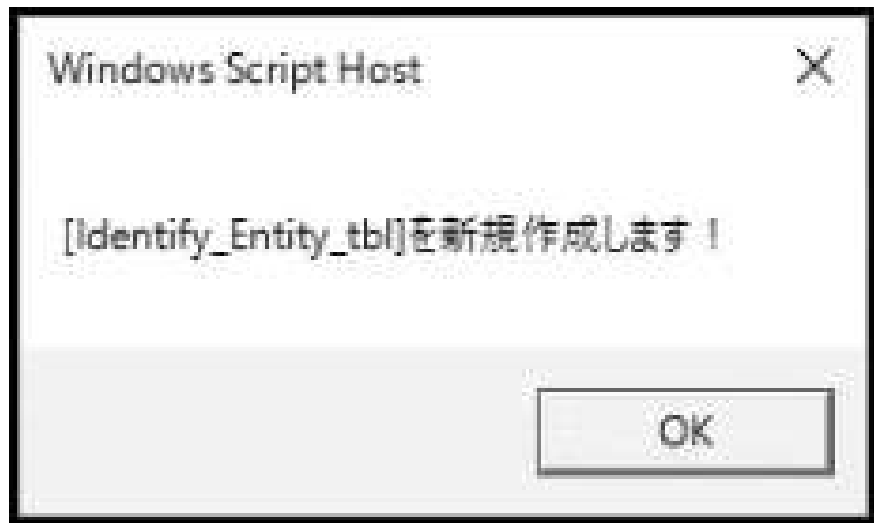
その時は、 K-MEMO 作成後
→ MK_K-MEMO.VBSを再度実行



Word_Entity_tblを作成
システム予約行を追加（数分待つ）



Identity_Entity_tblを作成



Word_Entity_tblに1000行のシステム予約データ作成

The screenshot displays the SQL Server Enterprise Manager interface on the left and a SQL Query window on the right. The Enterprise Manager shows the database structure for 'CF-S10-KUHEN', with 'K-MEMO' and 'dbo.Word_Entity_tbl' highlighted. The SQL Query window shows a script to insert 1000 rows of system reserved data into the 'Word_Entity_tbl' table. The script uses a loop to generate IDs from 899 to 904 and inserts specific words and their corresponding IDs. The results pane shows the first 19 rows of the data, with rows 900, 901, and 902 highlighted in red. The status bar at the bottom indicates that the query was executed successfully, inserting 1000 rows.

SQLQuery2.sql - C:\UHEN\SKojima (54) X

```
/* SSMS の SelectTopNRows コマンドのスク립ト */
SELECT TOP 1000 [ID]
, [WORDS]
, [WK_ID]
, [G_ID]
FROM [K-MEMO].[dbo].[Word_Entity_tbl]
```

ID	WORDS	WK_ID	G_ID
899	システム予約899	NULL	NULL
900	S_KENSAKU_MAIN	NULL	NULL
901	S_KENSAKU_SUB	NULL	NULL
902	S_KENSAKU_WIDE	NULL	NULL
903	システム予約903	NULL	NULL
904	システム予約904	NULL	NULL

クエリが正常に実行されました。 CF-S10-KUHEN (12.0 SP1) CF-S10-KUHEN\SKojima (54) K-MEMO 00:00:01 1000 行

テーブルは自動作成されたので今度は
ストアードプロシージャを登録する

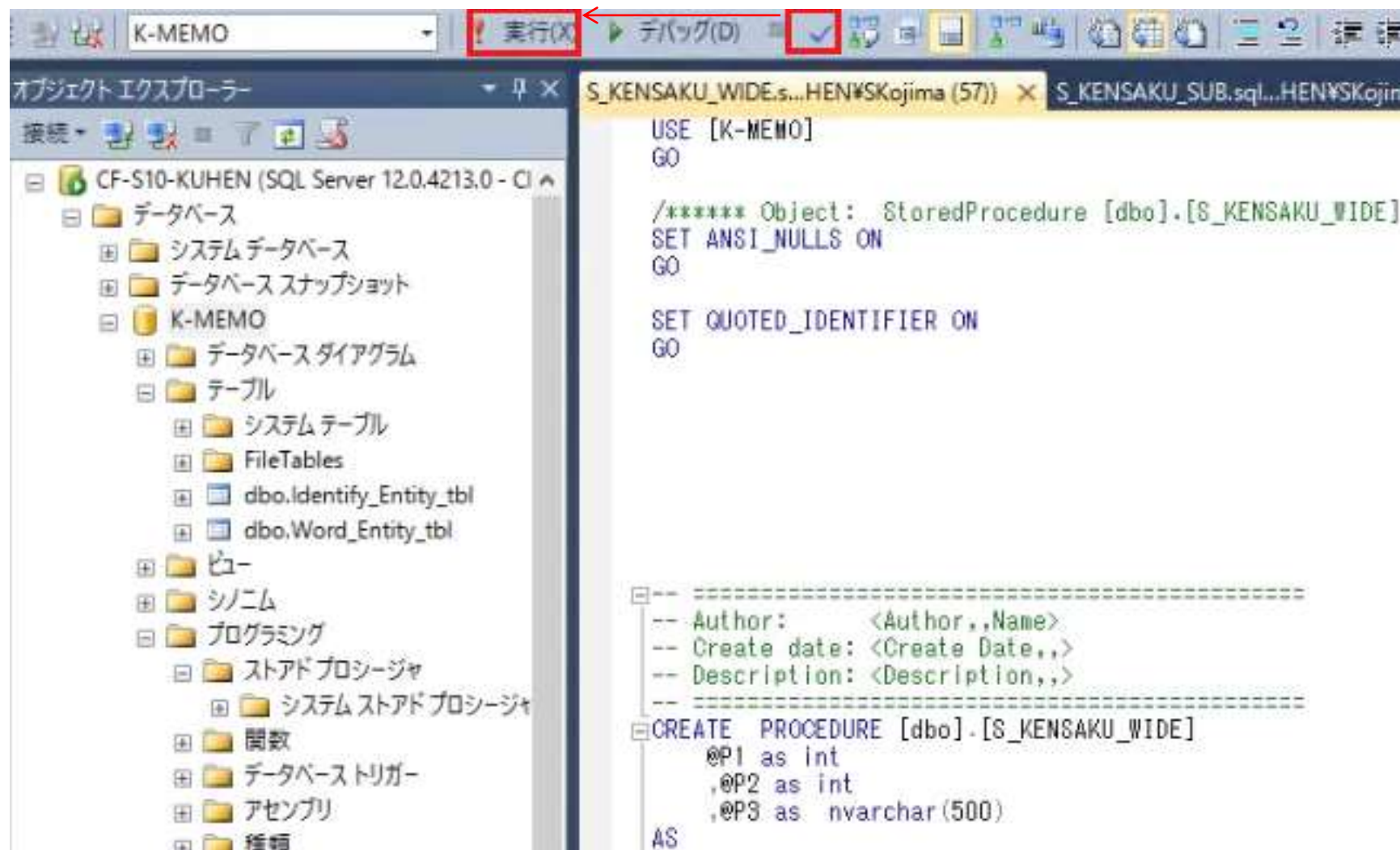
4つのストアド作成ファイルを「開く」から読み込み
チェックマーク → 実行 を4個実行する

S_ID_DEL_LIST.sql

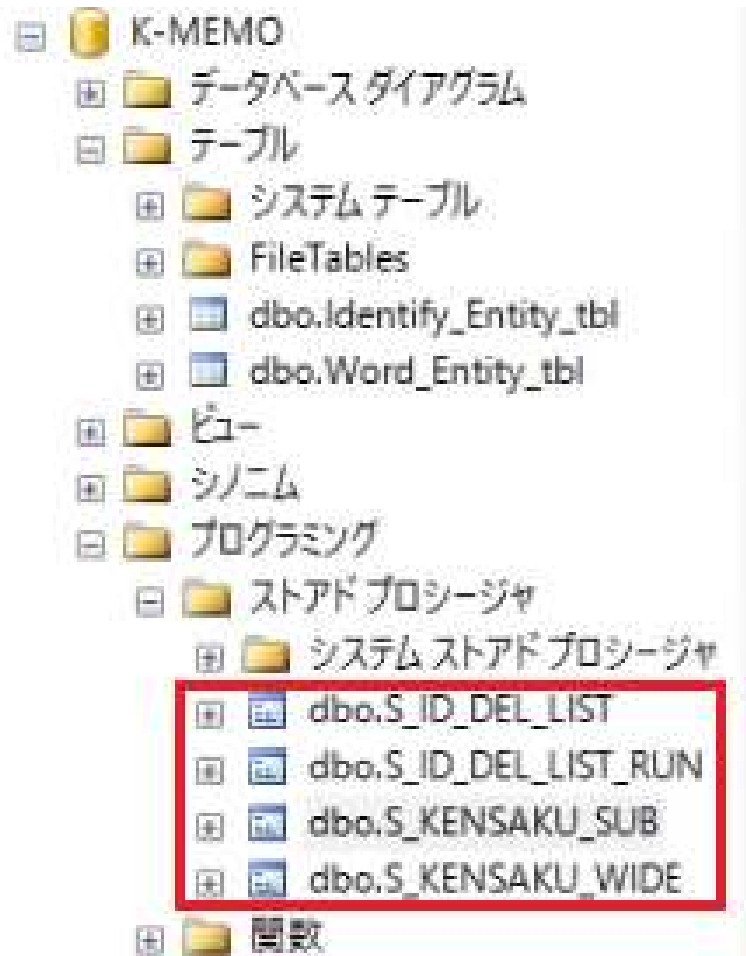
S_ID_DEL_LIST_RUN.sql

S_KENSAKU_SUB.sql

S_KENSAKU_WIDE.sql



「S_KENSAKU_SUB」「S_KENSAKU_WIDE」の 「public」実行権限を許可



2つの検索用ストアドは
S_KENSAKU_SUB
S_KENSAKU_WIDE

```
SET @WK_USER_NAME = '##' + REPLACE(LOWER  
( system_user ), '¥', '_')
```

グローバル一時テーブルを使う仕様になっている

※ Azureは逆にグローバルテーブルが使えない

<https://github.com/flhtc1964/NeuronDB>

詳しい情報は上記GitHubをご覧ください

以上