

# Wize テーブル仕様書 操作説明書

## 第 1.9.2 版

作 成 者	株式会社 Wize
作 成 日	2022 年 8 月 4 日

## 改版履歴

版数	発行日	改版履歴
第 1.0 版	2015 年 9 月 25 日	初版発行
第 1.1 版	2015 年 10 月 20 日	PostgreSQL に対応
第 1.2 版	2016 年 1 月 14 日	日本語-英語変換機能を追加
第 1.3 版	2017 年 2 月 10 日	<ul style="list-style-type: none"> <li>・トリガー設定機能, バックアップデータの復旧機能(Oracle のみ)を追加</li> <li>・日本語-英語変換辞書に”備考”を追加</li> <li>・全シートの日本語-英語一括変換機能を追加</li> <li>・テーブル削除で複数テーブルの削除できるように変更</li> <li>・テーブル仕様シートに、項目追加,項目削除ボタンを追加(複数項目指定可能)</li> <li>・環境設定に CREATE 文出力時にテーブル名,項目名を引用符で囲むかどうかの指定を追加</li> <li>・SQLServer 用の CREATE 文出力機能を追加 (DB インポート機能は未対応)</li> </ul>
第 1.4 版	2017 年 3 月 22 日	<ul style="list-style-type: none"> <li>・旧バージョンのテーブル仕様書のデータをマイグレーションする機能を追加</li> <li>・データ作成用シート作成機能を追加</li> </ul>
第 1.5 版	2017 年 3 月 28 日	日本語-英語変換用の辞書を自分自身のデータを参照するか、他のファイルのデータを参照するか選択できるように変更
第 1.6 版	2017 年 4 月 10 日	<ul style="list-style-type: none"> <li>・SQLServer 用の DB インポート機能を追加</li> <li>・テーブルインポート時の日本語テーブル名入力方法を改善</li> <li>・テーブル一覧に”処理対象”列を追加</li> <li>・日本語-英語変換辞書の”完全一致チェック”欄に DB 種別の指定を追加</li> </ul>
第 1.7 版	2017 年 4 月 18 日	・SQLServer 用のトリガー設定機能を追加
第 1.8 版	2017 年 6 月 13 日	・表紙、改訂履歴シートを追加
第 1.8.1 版	2017 年 6 月 27 日	・SQLServer 用の CREATE 文出力テンプレートを修正(各処理単位に GO を追加)
第 1.8.2 版	2017 年 6 月 28 日	・CREATE 文出力時にコメント内の ' (シングルクォート) を ” (ダブルクォート 2 つ) に置換える処理を追加
第 1.8.3 版	2017 年 9 月 5 日	<ul style="list-style-type: none"> <li>・テーブル一覧シートおよび各テーブル仕様シートのボタンを ActiveX コントロールから Form コントロールに変更</li> <li>・テーブル一覧シートおよび各テーブル仕様シートで No.列をクリックすると No.を振りなおす機能を追加</li> <li>・データマイグレーション時に表紙と改版履歴をコピーする処理を追加</li> </ul>
第 1.8.4 版	2019 年 11 月 13 日	<ul style="list-style-type: none"> <li>・データ作成用シートに INSERT 文作成機能を追加</li> <li>・改訂履歴シートに”版数”列を追加</li> <li>・テーブル一覧更新時に”処理対象列”がクリアされる不具合を修正</li> <li>・Oracle 用 Create 文作成用テンプレート(#CT_Oracle)の内容を更新</li> </ul>
第 1.8.5 版	2020 年 11 月 18 日	<ul style="list-style-type: none"> <li>・トリガー設定シートに「定義型」「桁数」「位取」列を追加</li> <li>・辞書データに、トリガー設定シートに登録されている項目を追加</li> </ul>

版数	発行日	改版履歴
第 1.8.6 版	2020 年 12 月 15 日	<ul style="list-style-type: none"> <li>・ テーブル仕様シートから SQL 文テンプレート(SELECT, INSERT, UPDATE, DELETE)を出力する機能を追加</li> <li>・ テーブル仕様シートから選択行を辞書に登録する機能を追加</li> </ul>
第 1.8.7 版	2021 年 4 月 20 日	<ul style="list-style-type: none"> <li>・ 環境設定でスキーマ名変更時、すべてのテーブル仕様シートのスキーマ名に反映する機能を追加</li> <li>・ テーブル追加画面で、複数テーブルの連続作成機能を追加</li> <li>・ 標準メニューに独自メニューを追加する方法を改善</li> <li>・ マイグレーション時に、1.8.3 版以前のテーブル仕様書から改訂履歴シートが正しく作成できない不具合を修正</li> <li>・ テーブル一覧、テーブル仕様シートの No.振りなおし処理を No.列クリック時→No.セルクリック時に実行するように変更</li> </ul>
第 1.8.8 版	2022 年 1 月 28 日	<ul style="list-style-type: none"> <li>・ PostgreSQL で DB インポート時にエラーが発生する場合があるため修正</li> <li>・ 環境設定(CREATE 文のスキーマ名、テーブル名、項目名)の内容を、SQL 文出力処理に反映</li> <li>・ 環境設定で DB 種別 PostgreSQL を選択時、CREATE 文のテーブル名、項目名を設定可能に変更</li> <li>・ SQLServer 用の CREATE 文出力用テンプレート(#CT_SQLServer)において、対象テーブルが存在する場合のみ DROP TABLE を実行するように修正</li> <li>・ 本操作説明書内で記述している各ソフトウェアのバージョン表記、および設定画面表示例を更新</li> </ul>
第 1.9.0 版	2022 年 5 月 23 日	<ul style="list-style-type: none"> <li>・ Dr.Sum に対応(CREATE 文出力機能のみ)</li> <li>・ テーブル一覧に入力された項目名(日本語)～備考を元に、テーブル仕様シートを一括作成する機能を追加</li> <li>・ テーブル一覧、テーブル仕様シートの No.列を計算式で表示する方法に変更</li> </ul>
第 1.9.1 版	2022 年 7 月 22 日	<ul style="list-style-type: none"> <li>・ CREATE 文出力処理に下記機能を追加</li> <li>1) INSERT/UPDATE トリガー、テーブルバックアップ、リストア、コメント出力、ログテーブル出力について出力する/しないを制御可能</li> <li>2) テーブルの別名(シノニム)を定義可能(Oracle, SQLServer のみ)</li> </ul>
第 1.9.2 版	2022 年 8 月 4 日	<ul style="list-style-type: none"> <li>・ Dr.Sum の CREATE 文出力処理に下記機能を追加</li> <li>1) PRIMARY 列の入力値を元に COMPOUND KEY を出力</li> <li>2) INDEX 列の入力値を元に、COMPOUND KEY および CREATE INDEX を出力</li> </ul>

## 目次

Wize テーブル仕様書 操作説明書 .....	1
1. 概要.....	5
2. 使用制限 .....	5
3. 動作環境 .....	6
4. 動作環境設定 .....	7
1) データベース接続用の設定 .....	7
2) データベース用パッケージのインストール.....	14
5. 操作説明 .....	18
1) 起動方法 .....	18
2) 環境設定 .....	19
3) 終了方法 .....	21
4) シート一覧.....	22
5) テーブル一覧シート .....	23
6) テーブル仕様シート .....	36
7) 日本語-英語変換機能.....	60
8) トリガー設定機能.....	64
6. 処理実行時のチェック項目 .....	67
1) テーブル一覧のチェック項目 .....	67
2) テーブル仕様のチェック項目 .....	67
7. おわりに .....	68

## 1. 概要

テーブル仕様書は、Excel シート上で動作する、データベースシステムの開発を支援するアプリケーションです。

テーブル仕様書の主な機能は次のとおりです。

- ・ DB インポート機能  
データベースに接続してテーブル定義情報を取得し、Excel シートに表示します。
- ・ CREATE 文出力機能  
Excel シートに入力されたテーブル仕様書からテーブル生成用の CREATE 文を出力します。  
またトリガー設定からインサート/アップデートトリガー文とログテーブル出力用トリガー文を出力します。
- ・ テーブル仕様マイグレーション機能  
旧バージョンのテーブル仕様書で作成したデータを移行します。この機能により以前に作成したデータに対して最新版の機能を使用することができます。

## 2. 使用制限

本ソフトはフリーソフトです。個人・法人にかかわらず自由にご使用ください。  
本ソフトへのリンクは自由ですが、転載,配布は禁止します。  
なお著作権はすべて株式会社 Wize が保有しています。

本ソフトを使用した事によって生じたすべての障害・損害・不具合等に関して、弊社は一切の責任を負いません。  
各自の責任においてご使用ください。

### 3. 動作環境

本ソフトの動作確認済み環境は以下のとおりです。

OS	Microsoft Windows 7 (64bit), Windows10 (64bit)
Excel	Microsoft Excel 2013 (64bit), Excel 2019 (64bit)
データベース・サーバ	Oracle 12c Standard Edition Release 12.2.0.1.0 - 64bit PostgreSQL 9.6.24 64-bit Microsoft SQL Server 2019 (RTM) - 15.0.2000.5 (X64) Dr.Sum 5.5
データベース・クライアント	Oracle SQL Developer 4.1.4.21 pgAdmin 4 v2 Microsoft SQLServer Management Studio v18.9.1 Dr.Sum Enterprise Manager 5.5
インターフェース	データソース(ODBC) (64bit)

#### 【注意】

動作確認はすべて 64bit 版の環境で行っていますが、32bit 環境でも DB インポート以外の機能は動作すると考えられます。

32bit 環境で DB インポート機能を使用する場合は、以下の点にご注意ください。

・ Windows7 64bit 環境で使用する場合は、Excel, データベース・クライアント, インターフェースはすべて 32bit 版または 64bit 版で統一してください。32bit 版, 64bit 版が混在している場合はデータベースアクセス機能が正しく動作しません。

・ Windows7 32bit 環境で使用する場合は、Excel, データベース, インターフェースはすべて 32bit 版となりますので問題ありません。

#### 【注意】

Wize テーブル仕様書を閉じたときに、以下のような VBAProject のパスワード入力画面が表示される場合があります。



この現象は以下のいずれかの条件で発生するため、ご使用の環境に応じて設定を行ってください。

1. ご使用のコンピュータに「DropBox」がインストールされている場合、DropBox の動作を終了してください。
2. ご使用のコンピュータに「Adobe Acrobat」がインストールされている場合、Excel の COM アドインから「Acrobat PDFMaker Office COM Addin」を外してください。

## 4. 動作環境設定

使用目的に応じた設定項目は以下のようになっています。

以下の表を参考にして、データベースからテーブル定義情報を取得する DB インポート機能を使用するか、ワークシートに入力したテーブル定義情報から CREATE 文を出力する機能を使用するかにより、必要な項目を設定してください。

設定項目	DB インポート機能	CREATE 文出力機能
ネットサービス名	●	—
データソース(ODBC) ※Dr.Sum は未対応	●	—
Excel セキュリティセンター	●	●

● 必須, △ 任意, — 不要

### 1) データベース接続用の設定

#### Oracle データベースを使用する場合

Oracle データベースを使用する場合は、あらかじめ使用するデータベースのネットサービス名(TSN サービス名)を設定しておく必要があります。

スタートメニューから Net Configuration Assistant を起動して設定してください。

設定方法の詳細については、Net Configuration Assistant のマニュアルを参照してください。

#### データソースの設定

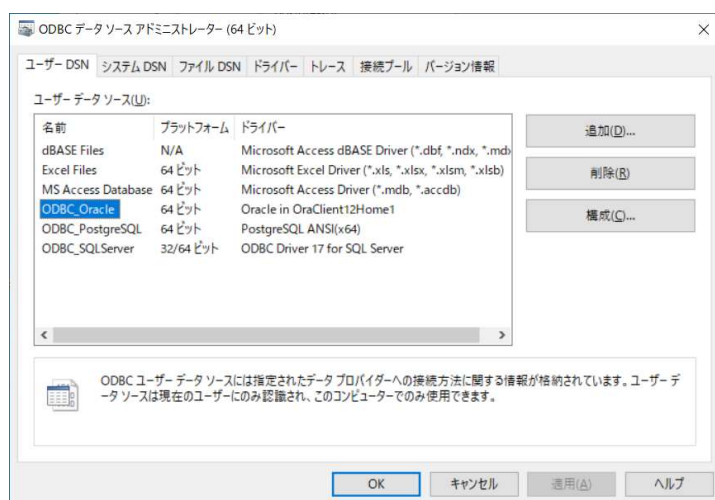
- 1) ODBC データソースアドミニストレーターを起動します。

Windows10 では、スタートメニューで“odbc”を検索し、ODBC データソース(64 ビット)を選択します。

Windows7 ではコントロールパネル/管理ツール/データソース(ODBC) を選択動します。

なお、Windows7 64bit 環境で、32bit 版の ODBC データアドミニストレータを起動するためには、%systemdrive%\¥Windows¥SysWoW64¥odbcad32.exe を実行する必要がありますので、十分ご注意ください。

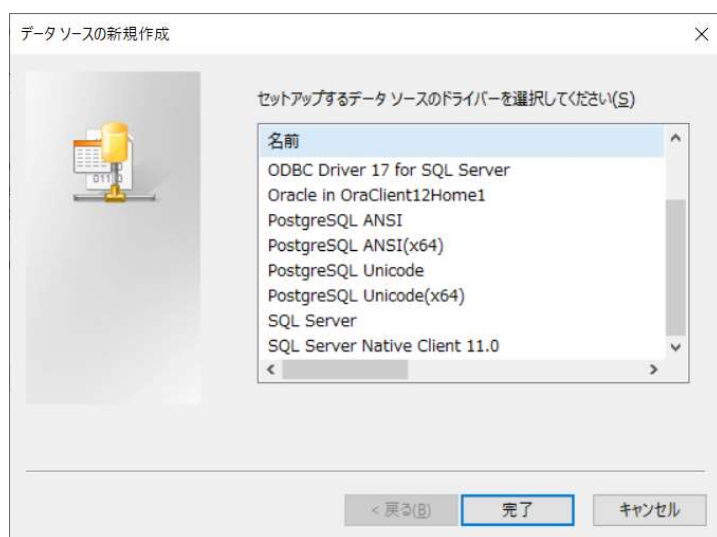
- 2) ODBC データソースアドミニストレーター画面の、ユーザー DSN のタブを選択します。



- 3) [追加]ボタンをクリックします。

- 4) データソースの新規作成画面で、使用するデータベース用のドライバーを選択して [完了]ボタンをク

リックします。



使用するデータベースとドライバの対応は以下のとおりです。

データベース	選択するドライバ
Oracle	Oracle in OraClient12Home1
PostgreSQL	PostgreSQL ANSI, PostgreSQL ANSI(x64)
SQLServer	ODBC Driver 17 for SQL Server

- 5) 次に ODBC ドライバ構成画面が表示されるので、必要な項目を入力します。  
ODBC ドライバ構成画面は使用するデータベースによって設定内容が異なるので、各 ODBC ドライバ構成画面に従って操作してください。

データベース	ODBC ドライバ設定方法
Oracle	5)-1. Oracle 用 ODBC ドライバ設定 を参照してください。
PostgreSQL	5)-2. PostgreSQL 用 ODBC ドライバ設定 を参照してください。
SQLServer	5)-3. SQLServer 用 ODBC ドライバ設定 を参照してください。

#### 5)-1. Oracle 用 ODBC ドライバ設定

以下の画面で、データソース名, TNS サービス名, ユーザーID の項目にデータを入力します。  
ユーザーID には、ユーザ ID とパスワード を “/” で区切って入力してください。



Oracle ODBCドライバ構成

データソース名: ODBC\_Oracle

説明:

TNSサービス名: tns\_name

ユーザーID: user\_id/password

アプリケーション: Oracle | Workarounds | SQLServer Migration

結果セットを有効化 ☒ 問合せタイムアウトを有効化 ☒ 読取専用接続 ☐

カーソル・クローズを有効化 ☐ スレッドセーフティを有効化 ☒

バッチ自動コミット・モード: すべてのステートメントが成功した場合のみコミット

数値の設定: OracleのNLS設定を使用

OK 取消 ヘルプ 接続テスト

データ入力後に、[接続テスト]ボタンをクリックします。  
データベースに接続できれば以下のメッセージが表示されます。



[OK]ボタンをクリックし、Oracle ODBC ドライバ構成画面の[OK]ボタンをクリックします。

## 5)-2. PostgreSQL 用 ODBC ドライバ設定

以下の画面で、データソース名、サーバ名、データベース名、ユーザー名、パスワード の項目にデータを入力します。



[テスト]ボタンをクリックします。  
データベースに正常に接続できれば、下記のメッセージが表示されます。



[OK]ボタンをクリックし、PostgreSQL ODBC セットアップ画面の[保存]ボタンをクリックします。

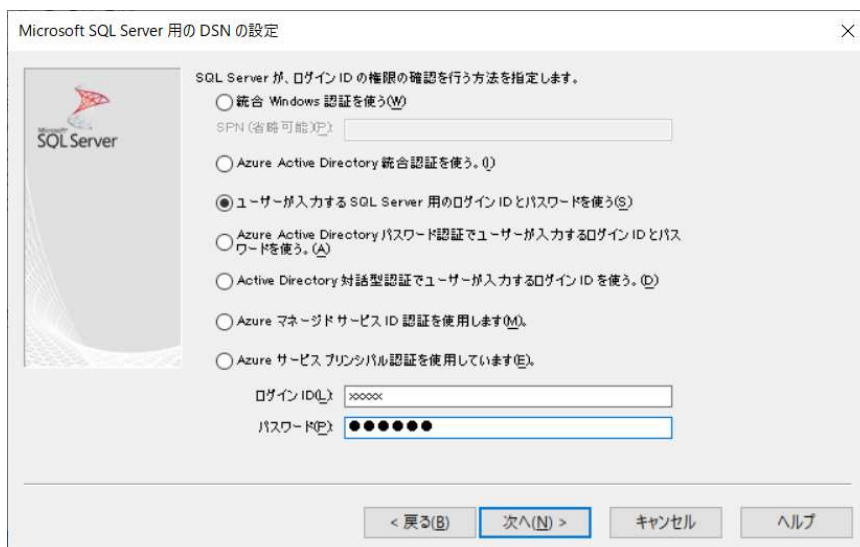
### 5)-3. SQLServer 用 ODBC ドライバ設定

以下の画面で、データソース名、サーバ名、データベース名の項目にデータを入力します。



[次へ]をクリックします。

以下の画面で、認証方法の選択およびログイン情報の設定を入力します。



Microsoft SQL Server 用の DSN の設定

SQL Server が、ログイン ID の権限の確認を行う方法を指定します。

- ☐ 統合 Windows 認証を使う(W)
- ☐ Azure Active Directory 統合認証を使う。(I)
- ☒ ユーザーが入力する SQL Server 用のログイン ID とパスワードを使う(S)
- ☐ Azure Active Directory パスワード認証でユーザーが入力するログイン ID とパスワードを使う。(A)
- ☐ Active Directory 対話型認証でユーザーが入力するログイン ID を使う。(D)
- ☐ Azure マネージドサービス ID 認証を使用します(M)
- ☐ Azure サービス プリンシパル 認証を使用しています(E)

ログイン ID(L): xxxxxx

パスワード(P): ●●●●●●

< 戻る(B)   **次へ(N) >**   キャンセル   ヘルプ

次へ]をクリックします。

以下の画面で必要な設定を行います。



Microsoft SQL Server 用の DSN の設定

☐ 既定のデータベースを以下に変更する(D):

ミラー サーバー(M):

ミラー サーバーの SPN (省略可能)(P):

☐ データベース ファイル名を添付する(H):

☒ ANSI の引用符付き識別子を使用する(U)

☒ ANSI の NULL、埋め込み文字、警告を使用する(A)

アプリケーションの目的(I): READWRITE

☐ マルチサブネット フェールオーバー(F)

☒ 透過的なネットワーク IP ソリューション(I)

☐ 列暗号化(E)

エンクレープ構成証明情報(N):

☐ FMTONLY メタデータの検出を使用します(Q)。

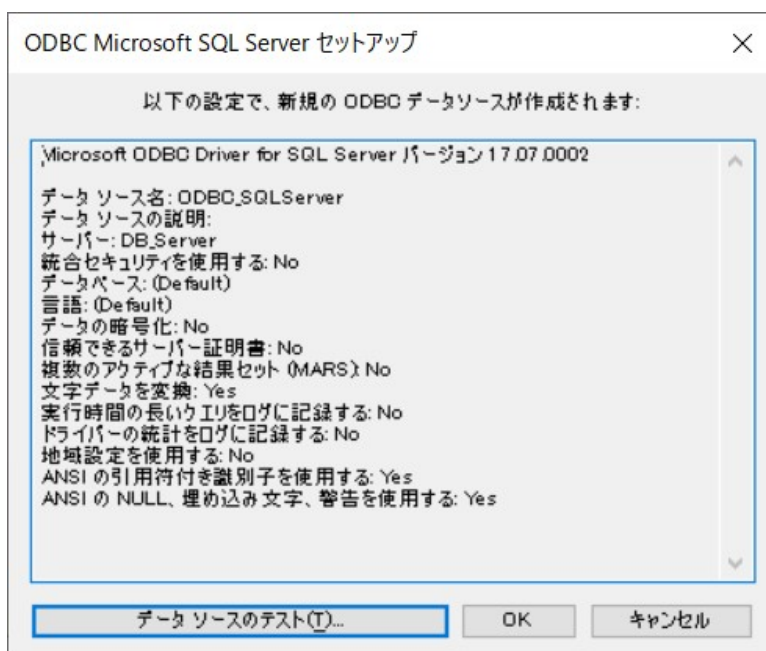
< 戻る(B)   **次へ(N) >**   キャンセル   ヘルプ

[次へ]をクリックします。

以下の画面で必要な設定を行います。



[完了]ボタンをクリックします。

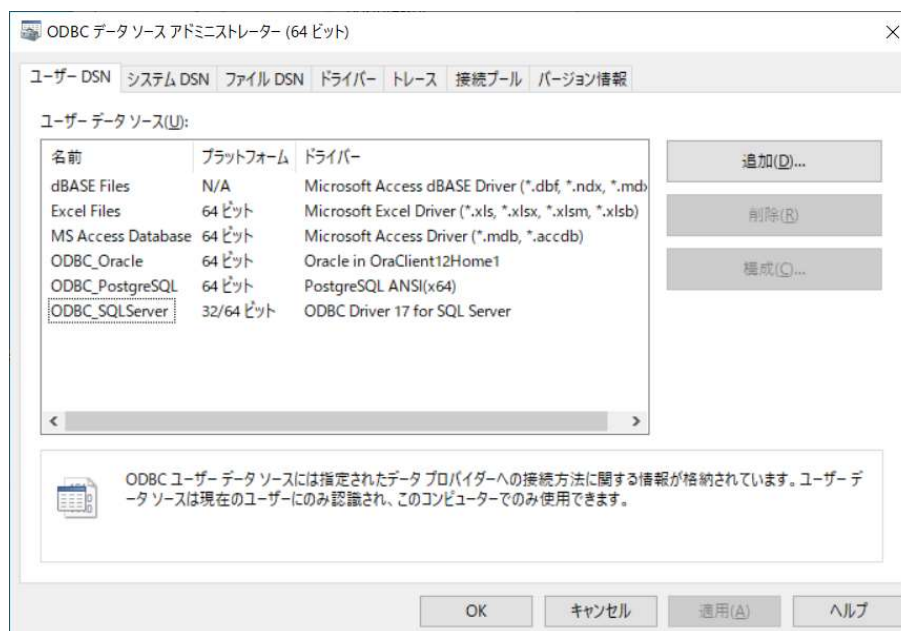


[データソースのテスト]をクリックします。

データベースに正常に接続できれば、下記のメッセージが表示されます。



- 6) ODBC データソースアドミニストレーター画面のユーザーDSN タブを選択し、ユーザーデータソースに追加した項目が表示されていることを確認してください。



以上で ODBC データソースの設定は終了です。

## 2) データベース用パッケージのインストール

### Oracle データベースを使用する場合

Oracle データベースを使用する場合、以下のパッケージをインストールすることにより、CREATE 文でテーブル生成時に、バックアップしたデータを復帰することができます。

※この機能は現在 Oracle のみの対応となります。ご了承ください。

- 1) ダウンロードしたファイルに含まれる WizePkgInstaller フォルダを開くと、以下の 3 つのファイルが格納されています。

INSTALL\_WIZE\_PKG.bat

Instpkg.sql

PKG\_WizeDBMaster.plb

- 2) コマンドプロンプトを起動して、INSTALL\_WIZE\_PKG.bat を実行します。



上記のメッセージが表示されるので、Oracle ユーザ名、パスワード、ホスト名を user/password@host 形式で入力してください。

上記のメッセージが表示され、“エラーはありません。” ”パッケージ本体が作成されました。”が表示されれば、パッケージのインストールは成功です。

## Excel のオプション設定

## セキュリティセンターの設定

- 1) Excel を起動し、メニューの ファイル/オプション を選択して、Excel のオプション画面を表示し、左側のメニューから”トラストセンター”を選択し、[トラストセンターの設定]ボタンをクリックします。(Windows7 では”トラストセンター”は”セキュリティセンター”と読み替えてください)。



- 2) トラストセンターの左側のメニューから、”マクロの設定”を選択します。
- 3) マクロの設定で、”警告を表示してすべてのマクロを無効にする”を選択してください





- 4) [OK]ボタンをクリックします。
- 5) Excel のオプション画面で[OK]ボタンをクリックします。



以上で Excel の環境設定は終了です。

## 5. 操作説明

以下、テーブル仕様書の操作方法を説明します。

### 1) 起動方法

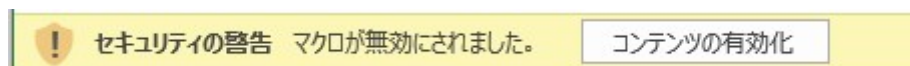
#### “テーブル仕様書.xlsm”を開く

通常の Excel シートと同様に“テーブル仕様書.xlsm”ファイルを開いて起動してください。  
起動すると以下のような画面が表示されます。



#### セキュリティの警告が表示される場合

起動後、Excel メニューの下に以下のような警告メッセージが表示される場合は、[コンテンツの有効化]ボタンをクリックしてください。



## その他

シート上部にあるボタンをクリックしても何も動作しない場合は、Excel の環境設定で、マクロの実行が無効になっている可能性があります。再度 Excel のオプション設定 に従って設定内容を確認してください。

## 2) 環境設定

### “テーブル一覧シート” の[環境設定]ボタンをクリックする

テーブル仕様書の機能を使用するためには、最初に環境設定が必要です。  
環境設定ボタンをクリックすると、以下のような画面が表示されます。

各項目を設定してください。

設定項目	内容
バージョン	テーブル仕様書のバージョン番号です。
データベース種別	データベースの種別を選択します。 (現在のバージョンでは Oracle, PostgreSQL, SQLServer, Dr.Sum が選択できます)
ODBC データソース名(DSN)	ODBC データソース名を入力します。※Dr.Sum は未対応

設定項目	内容
プロジェクト名	プロジェクト名を入力します。 CREATE 文出力時にコメントに表示します。
スキーマ名	スキーマ名を入力します。 入力済のスキーマ名を変更した場合、テーブル一覧シートの「処理対象」欄に○が付いているテーブル仕様シートのスキーマ名を一括変更可能です。
全テーブルシートに反映	押下時に、スキーマ名に入力された値ですべてのテーブルシートのスキーマ名を一括変更します。
CREATE 文のスキーマ名	CREATE 文のテーブル名にスキーマ名を付加するかどうかを指定します。 ※SQL 文出力時にも本設定を反映します。
項目名変換用辞書	テーブル名、項目名を日本語-英語に変換するときに、参照する辞書データを指定します。 “このブックの辞書を使用”選択時 現在開いているテーブル仕様書の”#DIC シート”を辞書データとして使用します。 “他のブックの辞書を使用”選択時 指定したファイルの”#DIC シート”を辞書データとして使用します。[参照]ボタンを押下するとファイルを選択できます(選択可能なファイルは *.xlsx, *.xlsm)。選択したファイルはテキストボックスに表示されます。 テキストボックスには、現在開いているテーブル仕様書を基準とした相対パスでファイルを指定することも可能です。
CREATE 文のテーブル名、項目名	CREATE 文のテーブル名、項目名を引用符で囲むかどうかを指定します。 ※SQL 文出力時にも本設定を反映します。
マテリアライズドビュー情報の取得	DB インポート時にマテリアライズドビュー情報を取得するかどうかを指定します。(現在は”取得しない”しか選択できません) マテリアライズドビュー情報のテーブル名は、先頭に MLOG\$_, RUPD\$_, USLOG\$_ が付いています。
CREATE 文出力フォルダ	CREATE 文ファイルの出力先フォルダを入力します。 [参照]ボタンで任意のフォルダを指定できます。

必要項目を設定した後、以下のいずれかのボタンをクリックしてください。

ボタン	機能
OK	入力されたデータを環境設定シートに保存します。
キャンセル	環境設定の変更を行いません。

以下の表を参考にして、データベースからテーブル定義情報を取得する DB インポート機能を使用するか、Excel シートに入力したテーブル定義情報から CREATE 文を出力する機能を使用するかにより、必要な項目を設定してください。

設定項目		DB インポート機能	CREATE 文出力機能
テーブル仕様書の環境設定	データベース種別	●	●
	ODBC データソース名(DSN)	●	—
	プロジェクト名	—	△

設定項目		DB インポート機能	CREATE 文出力機能
	スキーマ名	△	△
	CREATE 文でスキーマ名を使用	—	△
	マテリアライズドビュー情報の取得	△	—
	CREATE 文出力フォルダ	—	●

● 必須, △ 任意, — 不要

### 3) 終了方法

#### “テーブル仕様書.xlsm”を閉じる

通常の Excel シートと同様に“テーブル仕様書.xlsm”ファイルを閉じて終了してください。

#### 4) シート一覧

テーブル仕様書には、あらかじめ以下に示す 11 種類のワークシートが用意されています。

シート名	機能・用途
テーブル一覧	テーブル一覧表示用シートで、各種操作実行用ボタンがあります。(表示)
#トリガー設定	追加/更新用トリガー出力用のデータを保存します。(非表示)
#環境設定	環境設定の内容を保存します。(非表示)
#DIC	日本語・英語変換辞書データを保存します。(非表示)
#Table	テーブル仕様作成用のテンプレートシートです。(非表示)
#CT_Oracle	Oracle 用の CREATE 文出力用のテンプレートです。(非表示)
#CT_PostgreSQL	PostgreSQL 用の CREATE 文出力用のテンプレートです。(非表示)
#CT_SQLServer	SQLServer 用の CREATE 文出力用のテンプレートです。(非表示)
#CT_DrSum	Dr.Sum 用の CREATE 文出力用のテンプレートです。(非表示)
表紙	テーブル仕様書の表紙用シートです。(表示)
改訂履歴	テーブル仕様書の改訂履歴用シートです。(表示)

非表示設定のシートは必要に応じて再表示してご使用ください。

テーブル仕様書が作成されると、テーブル 1 つにつきテーブル仕様シートが 1 つ自動的に追加されます。

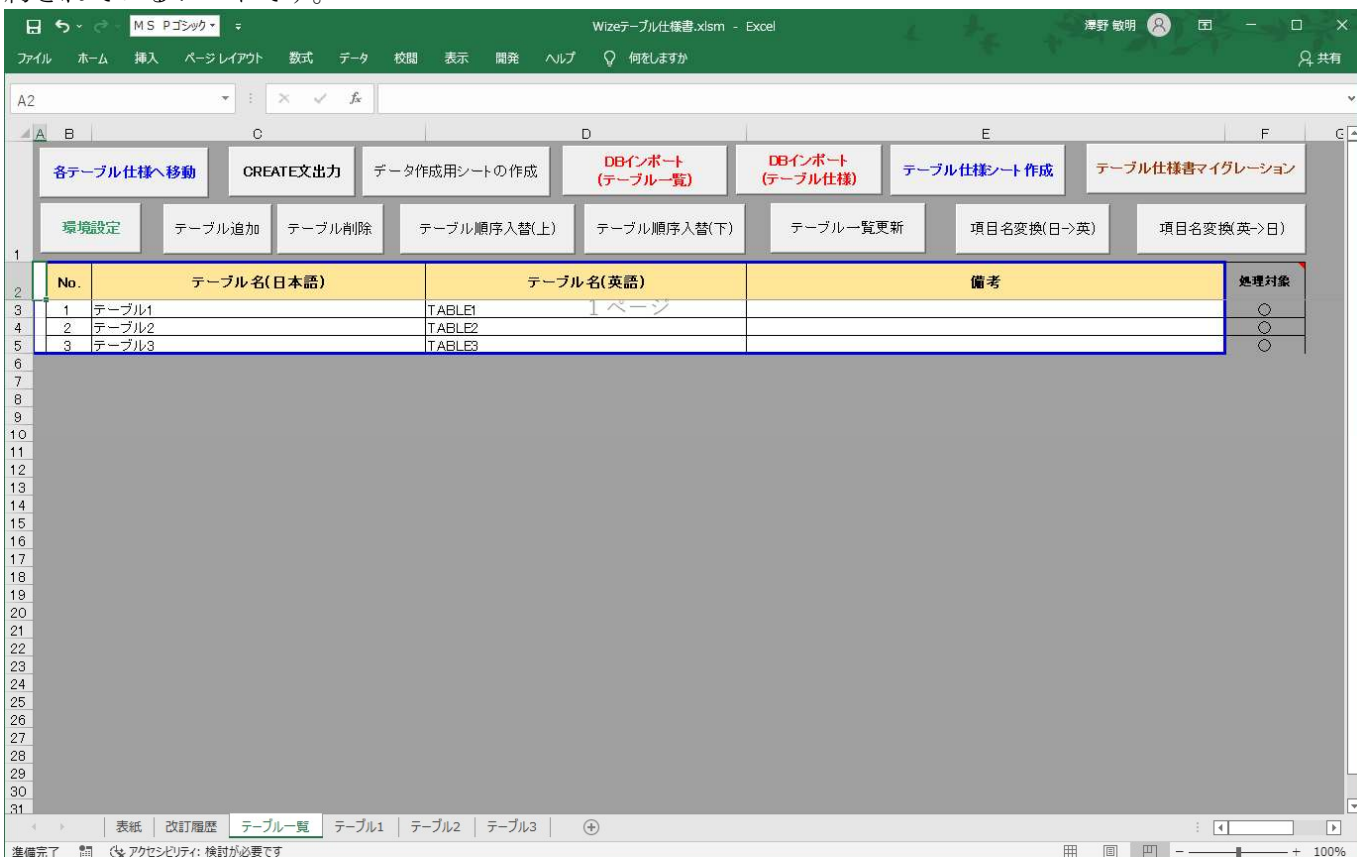
**【注意】**

これらのシートは絶対に削除しないでください。

また、動作環境によっては、作成したテーブル仕様シート数が多い場合、保存したファイルが正しく開けなくなることがありますのでご注意ください。

## 5) テーブル一覧シート

テーブル一覧シートは、データベースから取得したテーブル定義の一覧表示、テーブル一覧からテーブル仕様書の作成、およびテーブル仕様書から **CREATE** 文を作成するなど、テーブル仕様書の主要機能が集約されているシートです。



## 各ボタンの機能

[\[各テーブル仕様へ移動\]](#)

テーブル一覧で選択したテーブル仕様シートへ移動します。

テーブル一覧の任意の行を選択してからこのボタンをクリックすると、選択された行の”テーブル名(日本語)”と同一名のワークシートへ移動します。

[テーブル追加]

テーブル一覧表にテーブルを追加します。

このボタンをクリックすると、以下のダイアログボックスが表示されます。

×

テーブル追加

スキーマ名

テーブル名(日本語)

(必須)

日本語→英語変換

テーブル名(英語)

(必須)

英語→日本語変換

テーブル備考

キャンセル

選択した行の下に挿入

最下行に追加

☐ 連続追加

必要項目を入力してください。

項目名	機能
スキーマ名	スキーマ名を入力します。 環境設定で入力されたスキーマ名が表示されます。
テーブル名(日本語)	テーブル名を日本語で入力します。データベースに定義するテーブルのコメントとして使用されます。省略できません。 [日本語->英語変換]ボタンを押下すると、テーブル名(日本語)に入力されたデータを元に、辞書データを参照してテーブル名(英語)を自動入力することができます(※)。
テーブル名(英語)	テーブル名を英語で入力します。データベースに定義するテーブル名として使用されます。省略できません。 [英語->日本語変換]ボタンを押下すると、テーブル名(英語)に入力されたデータを元に、辞書データを参照してテーブル名(日本語)を自動入力することができます(※)。
テーブル備考	テーブル名の備考を入力します。データベースに定義するテーブルのコメントとして使用されます。
連続追加	チェックを付けると、テーブル追加を連続して行うことができます。

※あらかじめ辞書データを作成しておく必要があります。詳細は『[5.7 日本語-英語変換機能](#)』を参照してください。

上記の項目を入力した後、以下のいずれかのボタンをクリックしてください。

ボタン	機能
選択した行の下に挿入	テーブル一覧表の選択行の下にテーブルを追加します。
最下行に追加	テーブル一覧表の最下行にテーブルを追加します。 ※初期状態では No.1 にテーブルを追加します。
キャンセル	処理を行わずに画面を閉じます。

### [テーブル削除]

テーブル一覧表からテーブルを削除します。

テーブル一覧の任意の行を選択してからこのボタンをクリックすると、選択された行のテーブル仕様シートが削除されます。複数行を選択することも可能です。

### [テーブル順序入替(上)]

テーブル一覧表の行を 1 つ上に移動します。

テーブル一覧の任意の行を選択してからこのボタンをクリックすると、選択した行が 1 つ上の行と入れ替わります。

### [テーブル順序入替(下)]

テーブル一覧表の行を 1 つ下に移動します。

テーブル一覧の任意の行を選択してからこのボタンをクリックすると、選択した行が 1 つ下の行と入れ替わります。

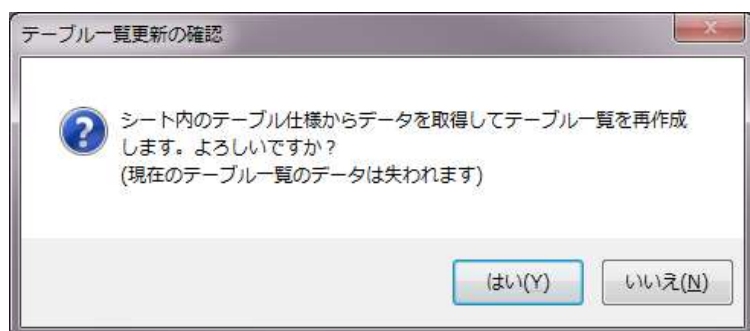
### [テーブル一覧更新]

現在のテーブル仕様シートを元にテーブル一覧表を再作成します。

あらかじめテーブル仕様シートが 1 つ以上作成されていることが必要です。

このボタンをクリックすると、以下の確認メッセージが表示されます。





以下のいずれかのボタンをクリックしてください。

ボタン	機能
はい	テーブル仕様書に入力されている情報を元に、テーブル一覧データを再作成します。
いいえ	処理を行いません。

#### [項目名変換(日->英)]

すべてのテーブル仕様シートに対して、項目名(日本語)に入力されたデータを元に、日本語-英語変換辞書を参照して項目名(英語)を自動入力します(※)。

#### [項目名変換(英->日)]

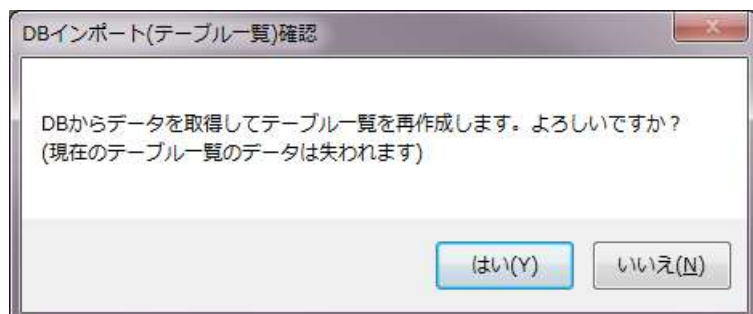
すべてのテーブル仕様シートに対して、項目名(英語)に入力されたデータを元に、日本語-英語変換辞書を参照して項目名(日本語)を自動入力します(※)。

※あらかじめ辞書データを作成しておく必要があります。詳細は『[5.7 日本語-英語変換機能](#)』を参照してください。

**[DB インポート(テーブル一覧)]**

データベースに接続し、データベースに定義されているテーブル一覧情報を取得し、テーブル一覧データを再作成します。※Dr.Sum は未対応

このボタンをクリックすると、以下の確認メッセージが表示されます。

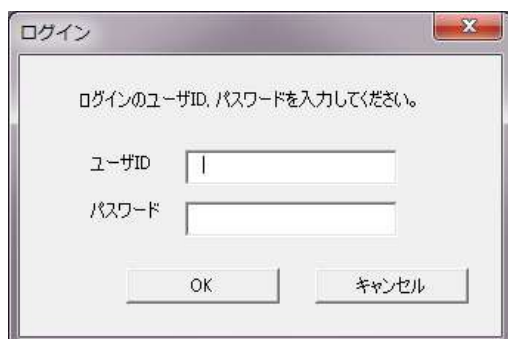


以下のいずれかのボタンをクリックしてください

ボタン	機能
はい	データベースからテーブル一覧情報を取得し、テーブル一覧データを再作成します。
いいえ	処理を行いません。

**<SQL Server との接続>**

環境設定のデータベース種別が "SQLServer" で、DB サーバの認証が「SQL Server 認証」の場合は、初回接続時に以下の画面が表示されるので、ユーザ ID、パスワードを入力してください。



以下のいずれかのボタンをクリックしてください

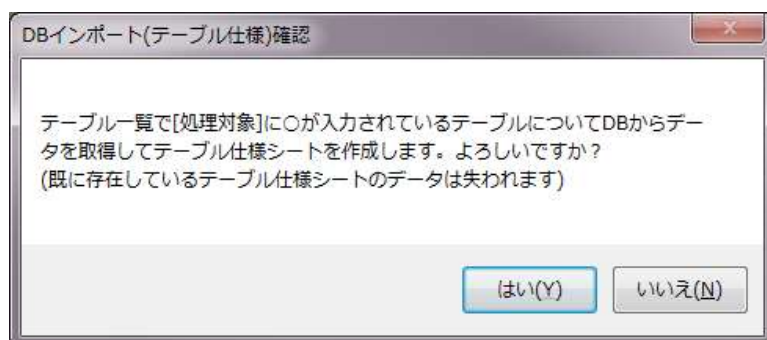
ボタン	機能
OK	入力されたユーザ ID, パスワードを使用して、SQL Server 認証を行います。
キャンセル	SQL Server 認証を行いません。

※ DB サーバの認証が「Windows 認証」の場合は、上記画面は表示されません。

**[DB インポート(テーブル仕様)]**

データベースに接続し、テーブル一覧の"処理対象"に○が入力されているすべてのテーブル定義情報を取得し、テーブル仕様シートの内容を更新します。※Dr.Sum は未対応  
すでにテーブル仕様書シートが存在している場合でも、データを更新するので十分ご注意ください。

このボタンをクリックすると、以下の確認メッセージが表示されます。



以下のいずれかのボタンをクリックしてください。

ボタン	機能
はい	データベースから処理対象に○が入力されているすべてのテーブル定義情報を取得し、テーブル仕様データを一括作成します。
いいえ	処理を行いません。

処理実行中は以下のようなメッセージで処理の状況が表示されます。



中央に現在作成中のテーブル名、右上に、「作成中のテーブル数 / 全テーブル数」を表示します。  
[中止]ボタンを押すと、処理を途中で中止することができます。

ボタン	機能
中止	テーブル仕様シートの作成処理を中止します。

テーブル仕様の作成処理が最後まで終了すると以下のメッセージが表示されます。

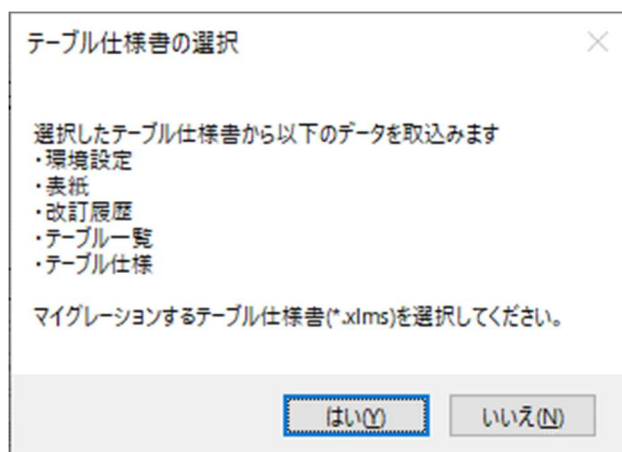


[OK]ボタンを押してこのメッセージを終了してください。

### [テーブル仕様書マイグレーション]

旧バージョンのテーブル仕様書のテーブル一覧、テーブル仕様、日本語-英語変換辞書データ(#DIC)のデータをすべて現在のテーブル仕様書に移行します。

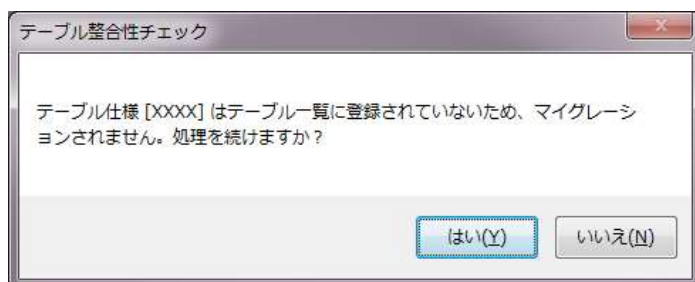
このボタンをクリックすると、以下の確認メッセージが表示されます。



以下のいずれかのボタンをクリックしてください

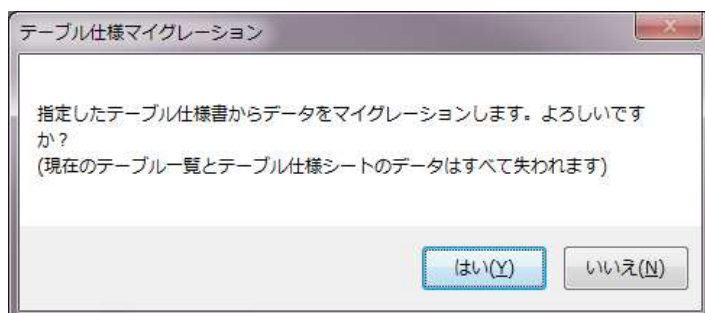
ボタン	機能
はい	ファイル選択ダイアログ画面が表示されるので、マイグレーションするテーブル仕様書(*.xlms)を開いてください。
いいえ	処理を行いません

マイグレーションするテーブル仕様書ファイルを指定すると、テーブル仕様書のデータを確認し、テーブル一覧とテーブル仕様シートの整合性を確認し、テーブル一覧に登録されていないテーブル仕様シートがある場合には、以下の確認メッセージが表示されます。



以下のいずれかのボタンをクリックしてください

ボタン	機能
はい	マイグレーション実行前の確認メッセージを表示します。
いいえ	処理を行いません

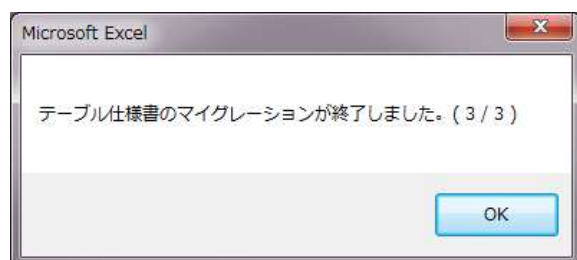


以下のいずれかのボタンをクリックしてください

ボタン	機能
はい	マイグレーションを実行します。

いいえ	処理を行いません
-----	----------

処理が終了すると、以下のようなメッセージが表示され、マイグレーション用に開いたテーブル仕様書ファイル(\*.xlms)を閉じます。



### [CREATE 文出力]

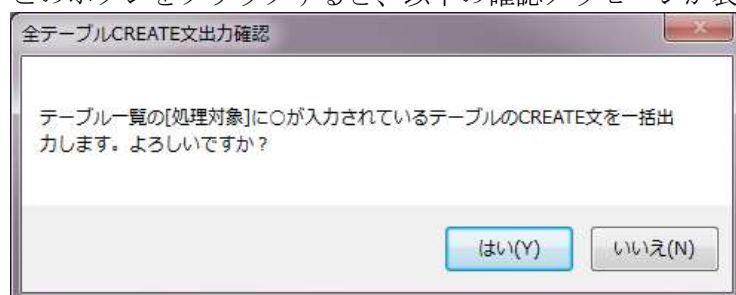
テーブル仕様に定義されている各テーブル定義情報に基づき、CREATE 文を一括作成します。ただし出力するファイルは 1 テーブルごとに 1 ファイルとなります。出力するファイル名は、"CT\_<テーブル名(英語)>.sql" となります。ファイル出力先は、環境設定の "CREATE 文出力フォルダ" で指定した場所になります。

CREATE 文出力オプション を指定することにより、出力内容を制御することが可能です。詳細は、「5.6 テーブル仕様シート」の CREATE 文出力オプション を参照してください。

#### 【注意】

出力ファイル名の大文字/小文字は区別されません。たとえば、テーブル名(英語)で、"TableName" と "TABLENAME" を作成した場合、CREATE 文出力時にどちらも同じファイル名で出力します。その結果、ファイルの内容はあとから出力した内容で上書きされてしまいます。この問題を回避するために、環境設定の "CREATE 文出力フォルダ" を変更し、対象となるテーブル仕様シートから [CREATE 文出力] を実行してください。

このボタンをクリックすると、以下の確認メッセージが表示されます。



以下のいずれかのボタンをクリックしてください

ボタン	機能
はい	すべてのテーブル定義情報から、CREATE 文を一括出力します。
いいえ	処理を行いません

## [データ作成用シート作成]

テーブル仕様書の情報を元にデータ作成用シート(\*.xlsm)を出力するための機能です。  
 データ作成用シートとは、以下の例で示すように、テーブルのカラム定義を横方向に展開し、レコード情報を縦方向に入力できるようにしたものです。  
 また[INSERT 文作成]ボタンを押下すると、シート内に INSERT 文表示用の計算式を出力することも可能です。

【注意】 データ作成用シートに[INSERT 文作成]ボタンを表示するためには、Excel マクロ有効テンプレートファイル“データ作成用テンプレート.xlsm”が必要です。このファイルがない場合は[INSERT 文作成]ボタンなしの状態でデータ作成用シート(\*.xlsx)を出力します。

## &lt;テーブル仕様書&gt;

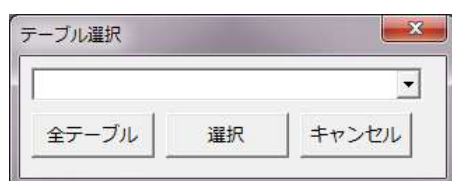
スキーマ名		schema_name				TABLESPACE(DATA)	
テーブル名(日本語)		テスト				TABLESPACE(INDEX)	
テーブル名(英語)		TEST				ログテーブル出力	
テーブル備考							
No.	項目名(日本語)	項目名(英語)	データ型	桁数	位取	デフォルト値	
1	カラム1	CLM1	VARCHAR	10			
2	カラム2	CLM2	VARCHAR	10			
3	カラム3	CLM3	VARCHAR	10			
4	カラム4	CLM4	DATE				
5	カラム5	CLM5	VARCHAR	10			
6							

## &lt;データ作成用シート&gt; 上記のテーブル TEST から作成

	A	B	C	D	E	F	G	H
1	"TEST"	カラム1	カラム2	カラム3	カラム4	カラム5	INSERT 文作成	
2	列名	"CLM1"	"CLM2"	"CLM3"	"CLM4"	"CLM5"		
3	型	VARCHAR(10)	VARCHAR(10)	VARCHAR(10)	DATE	VARCHAR(10)		
4	文字列							
5								
6								
7								

## ・操作方法

[データ作成用シート作成]ボタンをクリックすると、以下の確認メッセージが表示されます。



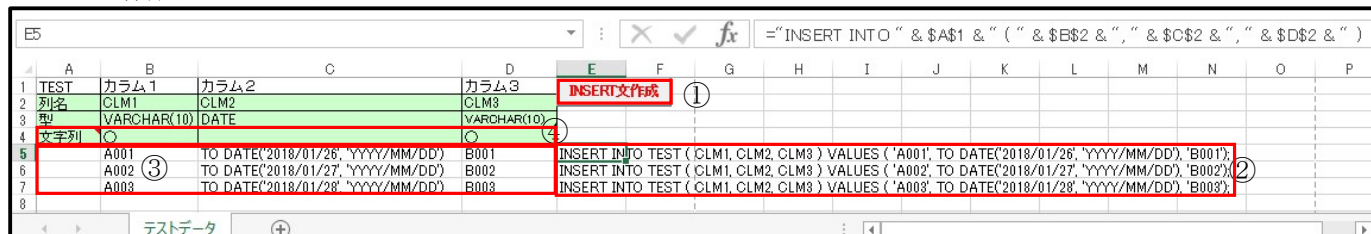
以下のいずれかのボタンをクリックしてください

ボタン	機能
全テーブル	テーブル一覧に登録されているすべてのテーブル仕様書について、指定したフォルダにデータ作成用シートを出力します。
選択	プルダウンリストで選択したテーブル仕様のデータ作成用シートをテーブル仕様書を格納しているフォルダに出力します。
キャンセル	処理を行いません

処理が終了すると、以下のメッセージが表示されます。



### ・データ作成用シートの使用法



- |                 |                                         |
|-----------------|-----------------------------------------|
| ① INSERT 文作成ボタン | 押下すると、入力したデータ行数分 INSERT 文を出力します         |
| ② “文字列”指定行      | ○を入力した項目は、INSERT 文のデータを”(シングルクォート)で囲みます |
| ③ データ入力エリア      | 各項目に対応したデータを複数行入力します                    |
| ④ INSERT 文表示エリア | INSERT 文表示用の計算式を出力します                   |

### <作成された INSERT 文の例>

```
INSERT INTO "TEST" ( "CLM1", "CLM2", "CLM3", "CLM4", "CLM5" ) VALUES ( 'A001', 'B001', 'C001', TO_DATE('2018/01/26', 'YYYY/MM/DD'), 'D001');
```

```
INSERT INTO "TEST" ( "CLM1", "CLM2", "CLM3", "CLM4", "CLM5" ) VALUES ( 'A002', 'B002', 'C002', TO_DATE('2018/01/27', 'YYYY/MM/DD'), 'D002');
```

```
INSERT INTO "TEST" ( "CLM1", "CLM2", "CLM3", "CLM4", "CLM5" ) VALUES ( 'A003', 'B003', 'C003', TO_DATE('2018/01/28', 'YYYY/MM/DD'), 'D003');
```

※”列名”行の内容は、環境設定の「Create 文のテーブル名、項目名」で ”引用符で囲む” が選択されている場合は、”(ダブルクォート)で囲んで出力されます。

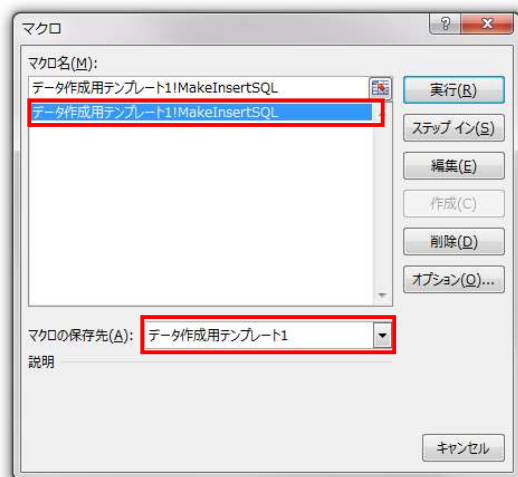


・[INSERT 文作成]ボタンなしのデータ作成用シートで INSERT 文を作成する方法

データ作成用シートから、Excel マクロ有効テンプレートファイル “データ作成用テンプレート.xltm”内のマクロを実行することにより、INSERT 文を作成することが可能です。(ただしテーブル仕様書 Ver1.8.3 以前で出力したデータ作成用シートには対応していません)

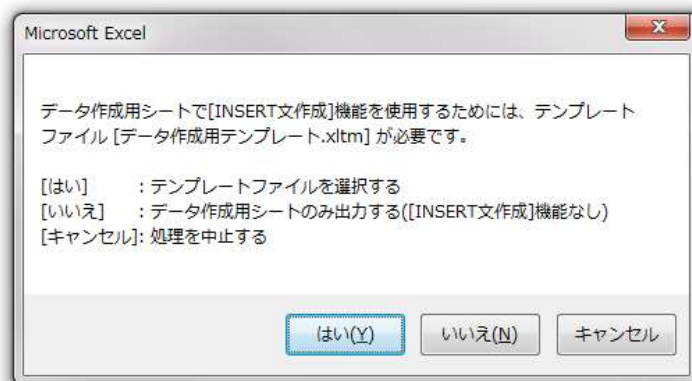
以下の手順で操作してください。

1. “データ作成用テンプレート.xltm”を開く
2. データ作成用シート(\*.xlsx)を開き、表示メニュー/マクロを選択
3. マクロの保存先で、“データ作成用テンプレート 1”を選択
4. マクロ名で、“データ作成用テンプレート 1!MakeInsertSQL”を選択して[実行]ボタンを押下



※このとき、他のブックに登録されているマクロを実行しないようにご注意ください。

・テーブル仕様書と同じフォルダに”データ作成用テンプレート.xltm”が存在しない場合は、以下の確認ダイアログが表示されますので、メッセージの内容に従って操作してください。



ボタン	機能
はい	ファイル選択ダイアログで”データ作成用テンプレート.xltm”の格納場所を選択してください。 ファイル選択後、INSERT 文作成ボタンありで、データ作成用シート(*.xlsm)を出力します。
いいえ	INSERT 文作成ボタンなしで、データ作成用シート(*.xlsx)を出力します。
キャンセル	データ作成用シートを出力しません



## [テーブル仕様シート作成]

テーブル一覧シートに入力された情報を元にデータ仕様シートを一括出力するための機能です。  
[テーブル追加]ボタンではテーブルを1つずつ登録するのに対し、この機能ではテーブル一覧シートに入力されたテーブル名を参照してテーブル仕様シートを一括作成します。すでにテーブル一覧シートにデータが入力された場合などに使用してください。

## ・操作方法

テーブル一覧シートにデータを入力した後、[データ仕様シート作成]ボタンをクリックすると、以下の確認メッセージが表示されます。

テーブル仕様シート作成

テーブル一覧で[処理対象]に○が入力されているテーブルについてテーブル仕様シートを一括作成します。よろしいですか？  
(既にテーブル仕様シートが存在する場合は何もありません)

以下のいずれかのボタンをクリックしてください

ボタン	機能
はい	テーブル仕様シートを一括作成します。テーブル一覧の処理対象に○が入力されているテーブルについて、テーブル仕様シートを一括作成します。
いいえ	処理を行いません

## &lt;操作例&gt;

## 実行前

各テーブル仕様へ移動
CREATE文出力
データ作成用シートの作成
DBインポート  
(テーブル一覧)
DBインポート  
(テーブル仕様)
テーブル仕様シート作成
テーブル仕様書マイグレーション

環境設定
テーブル追加
テーブル削除
テーブル順序入替(上)
テーブル順序入替(下)
テーブル一覧更新
項目名変換(日→英)
項目名変換(英→日)

No.	テーブル名(日本語)	テーブル名(英語)	備考	処理対象
1	テーブル1	Table1	備考1	○
2	テーブル2	Table2	備考2	○
3	テーブル3	Table3	備考3	○

表紙 改訂履歴 テーブル一覧

## 実行後

各テーブル仕様へ移動		CREATE文出力	データ作成用シートの作成	DBインポート (テーブル一覧)	DBインポート (テーブル仕様)	テーブル仕様シート作成	テーブル仕様書マイグレーション
環境設定	テーブル追加	テーブル削除	テーブル順序入替(上)	テーブル順序入替(下)	テーブル一覧更新	項目名変換(日→英)	項目名変換(英→日)

No.	テーブル名(日本語)	テーブル名(英語)	備考	処理対象
1	テーブル1	Table1	備考1	<input type="radio"/>
2	テーブル2	Table2	備考2	<input type="radio"/>
3	テーブル3	Table3	備考3	<input type="radio"/>

#Table 表紙 改訂履歴 テーブル一覧 テーブル1 テーブル2 テーブル3 +

## テーブル一覧データ

各項目の内容は以下のようになっています

No.

DB インポート時	CREATE 文出力時
データベースのテーブル一覧から、テーブル情報を取得した順に 1 から始まる連続する数字を表示します。	No.に入力されているデータと関係なく、一番上の行から順に CREATE 文を出力します。

テーブル一覧の、[テーブル追加], [テーブル削除], [テーブル順序(上)], [テーブル順序(下)], [テーブル更新] ボタンを使用した場合は、No.は自動で 1 から始まる数値を割り付けます。

注意

手作業でテーブル一覧データを追加、削除した場合は、No.の表示は自動更新されません。  
また No.欄には空白を入力しないでください。

### テーブル名(日本語)

DB インポート時	CREATE 文出力時
データベースのテーブル一覧から、テーブルコメントの 1 番目の要素(※)を取得して表示します。	入力された値を、テーブル定義のテーブルコメントの 1 番目の要素(※)にセットします。

### テーブル名(英語)

DB インポート時	CREATE 文出力時
データベースのテーブル一覧から、テーブル名を取得して表示します。	入力された値を、テーブル定義のテーブル名にセットします。

### 備考

DB インポート時	CREATE 文出力時
データベースのテーブル一覧から、テーブルコメントの 2 番目の要素(※)を取得して表示します。	入力された値を、テーブル定義のテーブルコメントの 2 番目の要素(※)にセットします。

※テーブルコメントの要素

テーブル仕様書では、データベースのテーブルコメントを、1 番目の要素(日本語テーブル名)と 2 番目の要素(テーブルの備考)を、スペースで区切って格納するというルールを設定し、それに従って DB インポート、CREATE 文出力を行います。

### 処理対象

DB インポート時	CREATE 文出力時
○が入力されたテーブルの情報を DB から取得し、テーブル仕様シートを作成します。	○が入力されたテーブルの CREATE 文を一括出力します。

### シートの印刷

- ・印刷範囲はテーブル一覧データが入力されている領域のみに限定しています。
- ・ヘッダ部に、シート名(テーブル一覧)、印刷日時、ヘッダ部にページ番号/総ページ数、Copyright を表示します。

## 6) テーブル仕様シート

テーブル仕様シートは、作成したテーブル仕様の数に応じて複数作成されますが、テンプレートシート (Table Template) からコピーして作成されるので、各ボタンや表示機能はすべて共通となります。テーブル仕様シートは以下のような形式になっています。

### <Ver1.3～Ver1.9.0>の画面

テーブル一覧へ移動		項目追加	項目削除	項目名変換(日→英)	項目名変換(英→日)	SQL文出力	DBインポート(テーブル仕様)	CREATE文出力			
スキーマ名	schema_name			TABLESPACE(DATA)							
テーブル名(日本語)	テスト			TABLESPACE(INDEX)							
テーブル名(英語)	TEST			ログテーブル出力							
テーブル備考											
No.	項目名(日本語)	項目名(英語)	データ型	指数	位取	デフォルト値	NOTNULL	UNIQUE	PRIMARY	備考	INDEX
1											
2											
3											
4											
5											
6											
7											
8											
9											
10											
11											
12											
13											
14											
15											
16											
17											
18											
19											
20											
21											
22											
23											
24											
25											
26											
27											
28											
29											
30											

### <Ver1.9.1以降>の画面

テーブル一覧へ移動		項目追加	項目削除	項目名変換(日→英)	項目名変換(英→日)	SQL文出力	DBインポート(テーブル仕様)	CREATE文出力			
スキーマ名	schema_name			TABLESPACE(DATA)		CREATE文出力オプション					
テーブル名(日本語)	テスト			TABLESPACE(INDEX)		<input checked="" type="checkbox"/> テーブルロックアップ <input checked="" type="checkbox"/> テーブルのコメント <input checked="" type="checkbox"/> インサート、アップデートリガー <input type="checkbox"/> ログテーブル出力					
テーブル名(英語)	TEST			別名(シノニム)		<input checked="" type="checkbox"/> テーブルリストア <input type="checkbox"/> ログテーブル出力					
テーブル備考	<div>全項目 ON</div> <div>全項目 OFF</div>										
No.	項目名(日本語)	項目名(英語)	データ型	指数	位取	デフォルト値	NOTNULL	UNIQUE	PRIMARY	備考	INDEX
1											
2											
3											
4											
5											
6											
7											
8											
9											
10											
11											
12											
13											
14											
15											
16											
17											
18											
19											
20											
21											
22											
23											
24											
25											
26											
27											
28											
29											
30											
31											
32											
33											
34											
35											
...											

## 各ボタンの機能

### [テーブル一覧へ移動]

テーブル一覧シートへ移動します。

### [項目追加]

選択した行の上にデータ項目を追加します。複数行を選択することも可能です。

### [項目削除]

選択した行のデータ項目を削除します。複数行を選択することも可能です。

### [項目名変換(日→英)]

項目名(日本語)に入力されたデータを元に、日本語-英語変換辞書を参照して項目名(英語)を自動入力することができます(※)。

### [項目名変換(英→日)]

項目名(英語)に入力されたデータを元に、日本語-英語変換辞書を参照して項目名(日本語)を自動入力することができます(※)。

※あらかじめ辞書データを作成しておく必要があります。詳細は『[5.7 日本語-英語変換機能](#)』を参照してください。

### [SQL 文出力]

項目名(英語)、項目名(日本語)に入力されたデータを元に、SQL 文(SELECT, INSERT, UPDATE, DELETE)のテンプレートを出力します。SQL 文作成時の支援機能としてご使用ください。

※実際の出力例は「SQL 文テンプレート出力例」を参照してください。

[SQL 文出力]ボタンをクリックすると、下記の”SQL 文出力設定”画面が表示されるので、SQL 文種別と SQL 文出力先を選択した後、SQL 文出力ボタンを押下します。

ボタン	機能
SQL 文種別	出力する SQL 文の種別を選択します。 ・ SELECT ・ INSERT

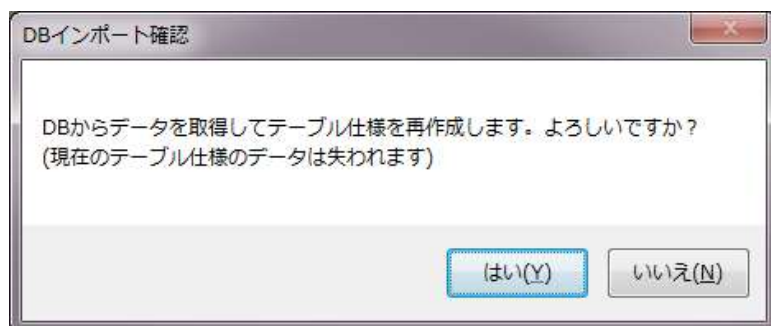
	<ul style="list-style-type: none"> <li>・ UPDATE</li> <li>・ DELETE</li> <li>・ 全種類</li> </ul> <p>“全種類”を選択した場合は、SELECT, INSERT, UPDATE, DELETE 文をすべて出力します。</p>												
SQL 文出力先	<p>SQL 文の出力先を選択します。</p> <ul style="list-style-type: none"> <li>・ クリップボード</li> <li>・ ファイル</li> </ul> <p>“クリップボード”を選択した場合は、SQL 文をクリップボードに出力しますので、テキストエディタ等にペーストしてください。</p> <p>“ファイル”を選択した場合は、SQL 文ファイルを出力するフォルダを入力してください。[参照]ボタンを押下するとフォルダ選択画面から任意のフォルダを指定できます。</p> <p>出力ファイル名は SQL 文種別ごとに以下の名称となります。</p> <table border="1"> <thead> <tr> <th>SQL 文種別</th><th>出力ファイル名</th></tr> </thead> <tbody> <tr> <td>SELECT</td><td>SELECT_&lt;テーブル名(英語)&gt;.sql</td></tr> <tr> <td>INSERT</td><td>INSERT_&lt;テーブル名(英語)&gt;.sql</td></tr> <tr> <td>UPDATE</td><td>UPDATE_&lt;テーブル名(英語)&gt;.sql</td></tr> <tr> <td>DELETE</td><td>DELTE_&lt;テーブル名(英語)&gt;.sql</td></tr> <tr> <td>全種類</td><td>SIUD_&lt;テーブル名(英語)&gt;.sql</td></tr> </tbody> </table>	SQL 文種別	出力ファイル名	SELECT	SELECT_<テーブル名(英語)>.sql	INSERT	INSERT_<テーブル名(英語)>.sql	UPDATE	UPDATE_<テーブル名(英語)>.sql	DELETE	DELTE_<テーブル名(英語)>.sql	全種類	SIUD_<テーブル名(英語)>.sql
SQL 文種別	出力ファイル名												
SELECT	SELECT_<テーブル名(英語)>.sql												
INSERT	INSERT_<テーブル名(英語)>.sql												
UPDATE	UPDATE_<テーブル名(英語)>.sql												
DELETE	DELTE_<テーブル名(英語)>.sql												
全種類	SIUD_<テーブル名(英語)>.sql												
キャンセル	何も処理を行わずに画面を閉じます。												
SQL 文出力	<p>SQL 文出力処理を行った後、画面を閉じます。</p> <p>選択/入力された設定情報は保存されます。</p> <p>※環境設定の CREATE 文出力設定(スキーマ名、テーブル名、項目名)をの内容を反映します。</p>												

**[DB インポート(テーブル仕様)]**

データベースに接続します。

選択したシートに対応するテーブル定義情報をデータベースから取得し、テーブル仕様シートの内容を更新します。

このボタンをクリックすると、以下の確認メッセージが表示されます。



以下のいずれかのボタンをクリックしてください

ボタン	機能
はい	データベースからテーブル定義情報を取得し、テーブル仕様書データを再作成します。
いいえ	処理を行いません。

**[CREATE 文出力]**

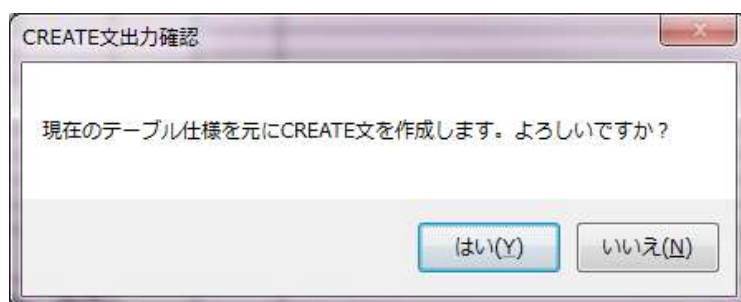
テーブル仕様シートに入力されたデータから、CREATE 文を出力します。

CREATE 文出力オプション を指定することにより、出力内容を制御することが可能です。詳細は本章の CREATE 文出力オプション を参照してください。

出力されるファイル名は、"CT\_<テーブル名(英語)>.sql" です。

ファイル出力先は、環境設定の CREATE 文出力フォルダで指定した位置です。

このボタンをクリックすると、以下の確認メッセージが表示されます。



以下のいずれかのボタンをクリックしてください

ボタン	機能
はい	現在のテーブル定義情報から CREATE 文を作成します。
いいえ	処理を行いません。

上記以外に、テーブル仕様に入力されたデータの整合性をチェックしますが、エラーメッセージの内容は、テーブル仕様データの各項目で説明します。

行メニュー / ◆選択行を辞書に登録

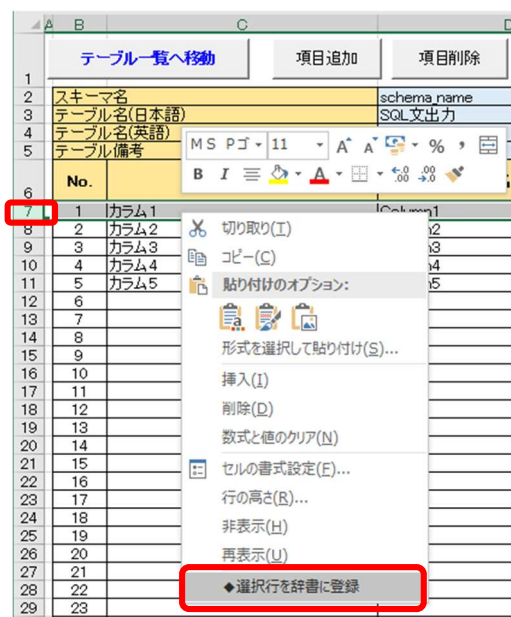
セルメニュー / ◆選択行を辞書に登録

※「改ページプレビュー」モードでは本メニューは表示されません。

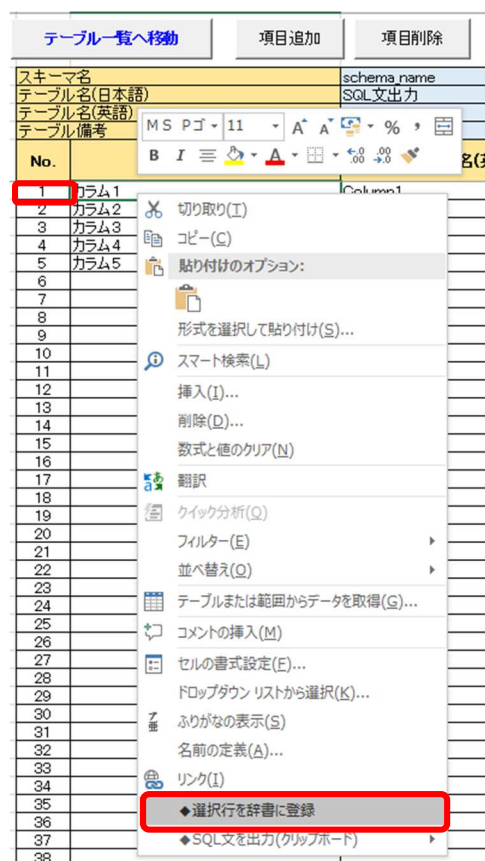
選択した行に入力されたデータを、辞書データ(#DIC シート)に登録します。

テーブル仕様シート内で、辞書データに登録したいデータ行、またはデータ行に含まれる任意のセルを選択後、右クリックして表示されるメニューから”◆選択行を辞書に登録”を選択します。

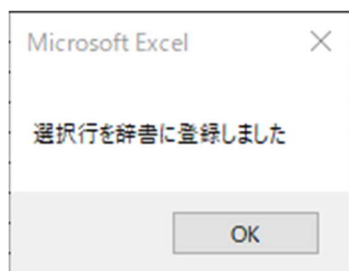
行メニューから選択した場合



セルメニューから選択した場合



選択した行の情報を元に、辞書データ(#DIC シート)の最下行にデータを追加します。  
データが追加されると下記のメッセージを表示します。



※選択した行の”項目名(英語)”がすでに辞書に存在する場合は登録できません。



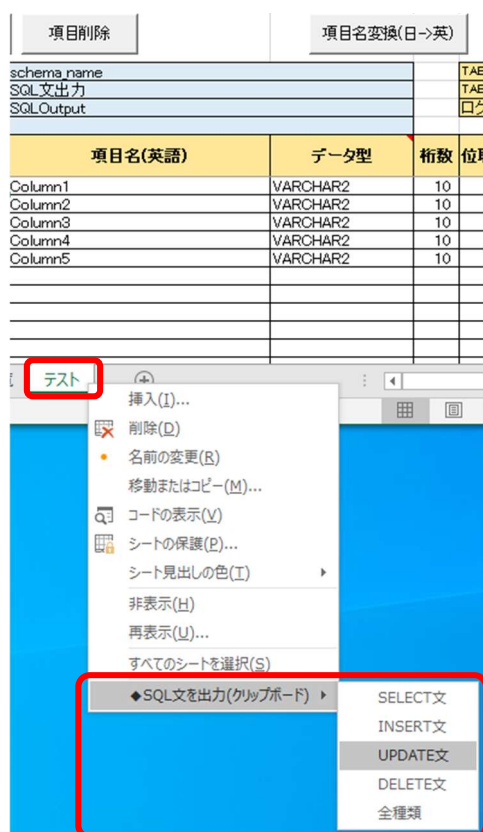
## シートメニュー / ◆SQL 文を出力(クリップボード)

※「改ページプレビュー」モードでは本メニューは表示されません。

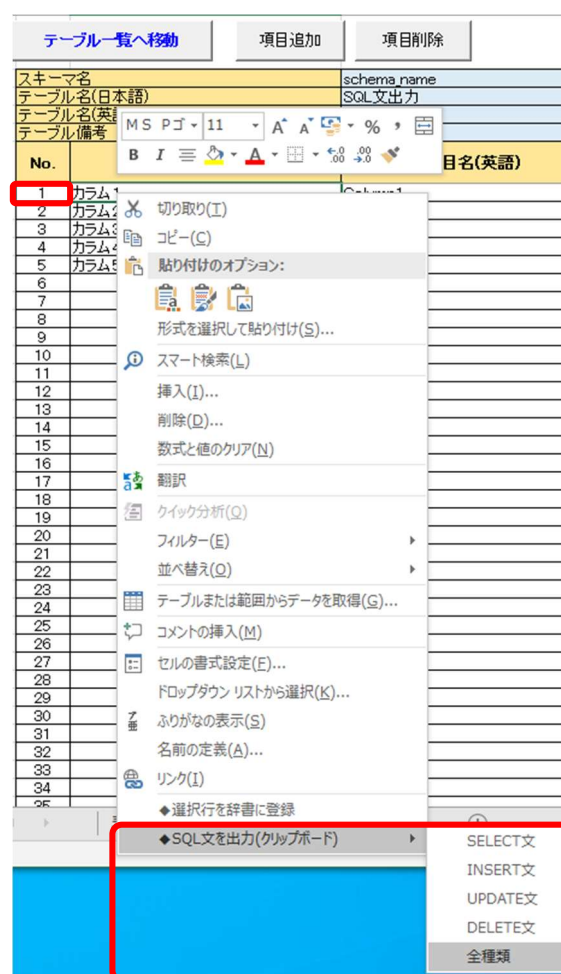
テーブル仕様シートに入力された“項目名(日本語)”、“項目名(英語)” データを元に、SQL 文テンプレートを出力します。

シート見出し、または任意のセルを右クリックして表示されるメニューから”◆SQL 文を出力(クリップボード)”を選択後、サブメニューの”SELECT 文, INSERT 文, UPDATE 文, DELETE 文, 全種類” からいずれか 1 つを選択します。

シートメニューから選択した場合



セルメニューから選択した場合



この機能は、[SQL 文出力] ボタン押下時の処理と同じですが、SQL 文の出力先が”クリップボード”のみとなります。

SQL 文をファイルに出力する場合は、[SQL 文出力]ボタンを使用してください。

## SQL 文テンプレート出力例

以下のようにテーブル仕様が定義されている場合に出力される SQL 文テンプレートの出力例を示します。出力されたコードを元に必要な箇所を編集してください。

テーブル一覧へ移動		項目追加	項目削除
スキーマ名		schema name	
テーブル名(日本語)		SQL文出力	
テーブル名(英語)		SQLOutput	
テーブル備考			
No.	項目名(日本語)	項目名(英語)	
1	カラム1	Column1	
2	カラム2	Column2	
3	カラム3	Column3	
4			
5			

SQL 文種別	SQL 文テンプレート出力例
SELECT	— SELECT 文 SQL 文出力 SELECT Column1 — カラム 1 , Column2 — カラム 2 , Column3 — カラム 3 FROM SQLOutput — SQL 文出力 WHERE Column1 = <value> — カラム 1 AND Column2 = <value> — カラム 2 AND Column3 = <value> — カラム 3 ORDER BY Column1 — カラム 1 , Column2 — カラム 2 , Column3 — カラム 3 ;
INSERT	— INSERT 文 SQL 文出力 INSERT INTO SQLOutput ( — SQL 文出力 Column1 — カラム 1 , Column2 — カラム 2 , Column3 — カラム 3 ) VALUES ( <value> — カラム 1 , <value> — カラム 2 , <value> — カラム 3 ) ;
UPDATE	— UPDATE 文 SQL 文出力 UPDATE SQLOutput — SQL 文出力 SET Column1 = <value> — カラム 1 , Column2 = <value> — カラム 2 , Column3 = <value> — カラム 3 WHERE Column1 = <value> — カラム 1 AND Column2 = <value> — カラム 2 AND Column3 = <value> — カラム 3 ;
DELETE	— DELETE 文 SQL 文出力 DELETE FROM SQLOutput — SQL 文出力 WHERE Column1 = <value> — カラム 1 AND Column2 = <value> — カラム 2 AND Column3 = <value> — カラム 3 ;

## テーブル仕様データ

### <テーブル仕様シートの設定領域>

スキーマ名	schema_name
テーブル名(日本語)	テスト
テーブル名(英語)	TEST
テーブル備考	テスト用情報

#### スキーマ名

DB インポート時	CREATE 文出力時
データベースからは取得しません。 環境設定のスキーマ名に設定されたデータを表示します。	環境設定で”CREATE 文でスキーマ名を使用する”設定になっている場合、CREATE 文出力時に各テーブル仕様シート内に入力されたスキーマ名を使用します。

#### テーブル名(日本語)

DB インポート時	CREATE 文出力時
データベースのテーブルコメントの 1 番目の要素(※1)を取得して表示します。	CREATE 文でテーブルコメントの 1 番目の要素(※1)として出力します。

#### テーブル名(英語)

DB インポート時	CREATE 文出力時
データベースのテーブル名を取得して表示します。	CREATE 文でテーブル名として出力します。

#### 備考

DB インポート時	CREATE 文出力時
データベースのテーブルコメントの 2 番目の要素(※1)を取得して表示します。	CREATE 文でテーブルコメントの 2 番目の要素(※1)として出力します。

#### ※1 テーブルコメントの要素

テーブル仕様書では、データベースのテーブルコメントを、1 番目の要素(日本語テーブル名)と 2 番目の要素(テーブルの備考)を、スペースで区切って格納するというルールを設定し、それに従って DB インポート、CREATE 文出力を行います。

## &lt;テーブル仕様シートの設定領域&gt;

TABLESPACE(DATA)	
TABLESPACE(INDEX)	

**TABLESPACE(DATA)**

DB インポート時	CREATE 文出力時
データベースのデータ格納用のテーブルスペース名を取得して表示します。	入力されたデータを、テーブル定義のデータ格納用テーブルスペース名として出力します。

**TABLESPACE(INDEX)**

DB インポート時	CREATE 文出力時
データベースのインデックス格納用のテーブルスペース名を取得して表示します。	入力されたデータを、テーブル定義のインデックス格納用テーブルスペース名として出力します。

以下の項目はバージョンにより設定面が異なります。

<Ver1.3～Ver1.9.0>の画面  
<テーブル仕様シートの設定領域>

ログテーブル出力	<input type="radio"/>
----------	-----------------------

ログテーブル出力

DB インポート時	CREATE 文出力時
参照されません。	以下の設定により、ログテーブル出力用の SQL 文(ログテーブル生成, ログ出力トリガー)を出力します。  “ログテーブル出力”欄に○を入力する

※Ver1.9.1 以降、ログテーブル出力の指定は、CREATE 文出力オプションの”ログテーブル出力”にチェックを付ける方法に変更

## &lt;Ver1.9.1 以降&gt;の画面

## &lt;テーブル仕様シートの設定領域&gt;

別名(シノニム)	name1, name2, *p_name
----------	-----------------------

## 別名(シノニム)

DB インポート時	CREATE 文出力時
参照されません。	<p>テーブルの別名を入力します。 空白の場合はシノニムを設定しません。 “,”(カンマ)区切りで複数の別名を設定可能です。 別名の先頭文字に”*”(アスタリスク)を付けると、別名が public 指定となります。(Oracle のみ)</p> <p>※Ver1.9.1 以降でサポート ※シノニムを使用するユーザに対して、あらかじめ適切な権限が付与されている必要があります。 ※対応 DB 種別は以下</p> <ul style="list-style-type: none"> <li>• Oracle</li> <li>• SQLServer</li> </ul>

## CREATE 文出力オプション (チェックボックス)

CREATE文出力オプション				全項目 ON
<input checked="" type="checkbox"/> テーブルバックアップ	<input checked="" type="checkbox"/> テーブルのコメント	<input checked="" type="checkbox"/> インサート,アップデートトリガー		
<input checked="" type="checkbox"/> テーブルリストア	<input type="checkbox"/> ログテーブル出力			全項目 OFF

DB インポート時	CREATE 文出力時
参照されません。	CREATE 文出力時に、上記チェックボックスの設定を参照し、各項目ごとの出力制御を行います。

## 操作説明

チェックボックス, ボタン	機能
テーブルバックアップ	テーブルバックアップ用処理の出力制御を行います (既定値=ON)
テーブルリストア	テーブルリストア用の処理の出力制御を行います (既定値=ON)
テーブルのコメント	テーブルのコメント用の処理の出力制御を行います (既定値=ON)
インサート,アップデートトリガー	インサート,アップデートトリガー用の処理の出力制御を行います (既定値=ON)
ログテーブル出力	ログテーブル出力用の処理の出力制御を行います (既定値=OFF)
全項目 ON	すべてのチェックボックスにチェックを付けます
全項目 OFF	すべてのチェックボックスのチェックをはずします

以下の項目はすべてのバージョンで共通です。

No.

DB インポート時	CREATE 文出力時
データベースのテーブル定義から、カラム情報を取得した順に 1 から始まる連続する数字を表示します。	No.に入力されているデータと関係なく、1 番上の行から順に CREATE 文のカラム定義を出力します。

項目名(日本語)

DB インポート時	CREATE 文出力時
データベースのテーブル定義から、カラムコメントの 1 番目の要素(※2)を取得して表示します。	カラムコメントの 1 番目の要素(※2)として出力します。

項目名(英語)

DB インポート時	CREATE 文出力時
データベースのテーブル定義から、カラム名を取得して表示します。	テーブル定義のカラム名として出力します。

## データ型

データベース	DB インポート時	CREATE 文出力時																																
Oracle	データベースのテーブル定義から、データ型を取得して表示します。 表示されるデータ型は Create 文で設定した内容と一致します。	データ型定義として出力します。 以下のデータ型は、データ型と桁数,位取りの内容をチェックします。  <b>VARCHAR2, NVARCHAR2, CHAR, NCHAR, LONG, CLOB, NUMBER, BINARY_FLOAT, BINARY_DOUBLE, DATE, RAW, BLOB, BFILE, ROWID, UROWID, FLAT</b>  上記以外のデータ型はチェックを行いませんので、ワークシートに入力したデータがそのままデータ型として出力されます。																																
PostgreSQL	データベースのテーブル定義から、データ型を取得して表示します。 ただし下記のデータ型は、Create 文で設定した内容と異なる名称で表示されます。 <table><tr><th>Create 文</th><th>データ型</th></tr><tr><td>bigint</td><td>int8</td></tr><tr><td>bigserial</td><td>int8 (注)</td></tr><tr><td>bit varying</td><td>varbit</td></tr><tr><td>boolean</td><td>bool</td></tr><tr><td>character varying</td><td>varchar</td></tr><tr><td>character</td><td>char</td></tr><tr><td>double precision</td><td>float8</td></tr><tr><td>integer</td><td>int4</td></tr><tr><td>decimal</td><td>numeric</td></tr><tr><td>real</td><td>float4</td></tr><tr><td>smallint</td><td>int2</td></tr><tr><td>smallserial</td><td>int2 (注)</td></tr><tr><td>serial</td><td>int4 (注)</td></tr><tr><td>time with time zone</td><td>timetz</td></tr><tr><td>Timestamp with time zone</td><td>timestamptz</td></tr></table>  (注) Create 文で指定したデータ型が自動増分整数のため、デフォルト値の欄に nextval('テーブル名_カラム名_seq'::regclass) NOTNULL の欄に ○が表示されます。	Create 文	データ型	bigint	int8	bigserial	int8 (注)	bit varying	varbit	boolean	bool	character varying	varchar	character	char	double precision	float8	integer	int4	decimal	numeric	real	float4	smallint	int2	smallserial	int2 (注)	serial	int4 (注)	time with time zone	timetz	Timestamp with time zone	timestamptz	データ型定義として出力します。 以下のデータ型は、データ型と桁数,位取りの内容をチェックします。  <b>varchar, character varying, char, character, bpchar, serial, serial4, serial8, bigserial, int, int4, int8, integer, bigint, decimal, numeric, real, float4, time, timestamp</b>  上記以外のデータ型はチェックを行いませんので、ワークシートに入力したデータがそのままデータ型として出力されます。
Create 文	データ型																																	
bigint	int8																																	
bigserial	int8 (注)																																	
bit varying	varbit																																	
boolean	bool																																	
character varying	varchar																																	
character	char																																	
double precision	float8																																	
integer	int4																																	
decimal	numeric																																	
real	float4																																	
smallint	int2																																	
smallserial	int2 (注)																																	
serial	int4 (注)																																	
time with time zone	timetz																																	
Timestamp with time zone	timestamptz																																	



データベース	DB インポート時	CREATE 文出力時
SQLServer	データベースのテーブル定義から、データ型を取得して表示します。 表示されるデータ型は Create 文で設定した内容と一致します。	<p>データ型定義として出力します。 以下のデータ型は、データ型と桁数,位取りの内容をチェックします。</p> <p><b>bigint, int, smallint, tinyint, bit, decimal, numeric, money, smallmoney, float, real, datetime, smalldatetime, char, varchar, text, nchar, nvarchar, ntext, binary, varbinary, image, sql_variant, timestamp, uniqueidentifier</b></p> <p>上記以外のデータ型はチェックを行いませんので、ワークシートに入力したデータがそのままデータ型として出力されます。</p>
Dr.Sum	※DB インポート機能はサポートしていません	<p>データ型定義として出力します。 以下のデータ型は、データ型と桁数,位取りの内容をチェックします。</p> <p><b>VARCHAR, CHAR, VARYING CHARACTER, INTEGER, INT, SMALLINT, TINYINT, NUMERIC, DECIMAL, REAL, FLOAT, DOUBLE, DATE, TIME, TIMESTAMP, INTERVAL</b></p> <p>上記以外のデータ型はチェックを行いませんので、ワークシートに入力したデータがそのままデータ型として出力されます</p>

## 桁数 および 位取

## Oracle データベースの場合

DB インポート時			CREATE 文出力時			
データベースのテーブル定義から取得したデータ型に応じて、以下のように表示します。			入力されたデータにより、カラム定義のデータ型を以下のように出力します。			
データ型	桁数	位取	データ型	桁数	位取	CREATE 文
VARCHAR2	char_length	空白	VARCHAR2	n		VARCHAR2(n)
NVARCHAR2	char_length	空白	NVARCHAR2	n		NVARCHAR2(n)
CHAR	char_length	空白	CHAR			CHAR
NCHAR	char_length	空白	CHAR	n		CHAR(n)
CLOB	data_length	空白	NCHAR			NCHAR
NUMBER(x, y)	data_precision	data_scale	NCHAR	n		NCHAR(n)
NUMBER(x)	data_precision	空白	CLOB			CLOB
NUMBER	空白	空白	NUMBER			NUMBER
FLOAT(x)	data_precision	空白	NUMBER	p		NUMBER(p)
FLOAT	空白	空白	NUMBER	p	s	NUMBER(p, s)
BINARY_FLOAT	空白	空白	FLOAT			FLOAT
BINARY_DOUBLE	空白	空白	FLOAT	n		FLOAT(n)
DATE	空白	空白	BINARY_FLOAT			BINARY_FLOAT
TIMESTAMP	空白	空白	BINARY_DOUBLE			BINARY_DOUBLE
INTERVAL YEAR TO MONTH	空白	空白	DATE			DATE
INTERVAL DAY TO SECOND	空白	空白	ROWID			ROWID
ROWID	空白	空白	RAW			RAW
RAW	data_length	空白	RAW	n		RAW(n)
UROWID	data_length	空白	UROWID			UROWID
BLOB	空白	空白	UROWID	n		UROWID(n)
BLOB	空白	空白	BLOB			BLOB
BFILE	空白	空白	BFILE			BFILE
LONG	空白	空白	LONG			LONG

上記以外のデータ型が入力された場合は、データ型、桁数、位取りいずれも内容をチェックしません。

## PostgreSQL データベースの場合

DB インポート時			CREATE 文出力時			
データベースのテーブル定義から取得したデータ型に応じて、以下のように表示します。			入力されたデータにより、カラム定義のデータ型を以下のように出力します。			
データ型	桁数	位取	データ型	桁数	位取	CREATE 文
int8 (bigint), int4 (integer), int2 (smallint)	空白	空白	bigint, integer, smallint			bigint, integer, smallint
float4 (real)	空白	空白	real, float4			real, float4
serial8 (bigserial), serial4 (serial), serial2 (smallserial)	空白	空白	bigserial, serial, smallserial			bigserial, serial, smallserial
varchar (character varying), bpchar (character, char)	char_length	空白	character varying, varchar			character varying, varchar
			character varying, varchar	n		character varying(n), varchar(n)
			character, char			character, char
			character, char	n		character(n), char(n)
numeric (decimal)	data_precision	data_scale	numeric, decimal			numeric, decimal
date	空白	空白	numeric, decimal	p		numeric(p), decimal(p)
time	data_precision	空白	numeric, decimal	p	s	numeric(p,s), decimal(p,s)
timetz (time with time zone)	data_precision	空白	date			date
timestamp	data_precision	空白	time, time with time zone			time, time with time zone
timestamp (timestamp with time zone)	data_precision	空白	timestamp, timestamp with time zone			timestamp, timestamp with time zone
			上記以外のデータ型が入力された場合は、データ型、桁数、位取りいずれも内容をチェックしません。			

## SQLServer データベースの場合

DB インポート時			CREATE 文出力時			
データベースのテーブル定義から取得したデータ型に応じて、以下のように表示します。			入力されたデータにより、カラム定義のデータ型を以下のように出力します。			
データ型	桁数	位取	データ型	桁数	位取	CREATE 文
bigint integer smallint tinyint bit	空白	空白	bigint, integer, smallint, tinyint, bit			bigint, integer, smallint, tinyint, bit
numeric decimal	data_precision	data_scale	real, float			real, float
real	空白	空白	float	n		float(n)
float	data_precision	空白	money, smallmoney			money, smallmoney
money smallmoney	空白	空白	char, varchar nchar, nvarchar, text, ntext			char, varchar nchar, nvarchar, text, ntext
date datetime smalldatetime	空白	空白	char, varchar nchar, nvarchar	n		char(n), varchar(n), nchar(n), nvarchar(n)
time datetime2 datetimeoffset	data_precision	空白	numeric, decimal			numeric, decimal
char varchar	char_length	空白	numeric, decimal	p		numeric(p), decimal(p)
text	空白	空白	numeric, decimal	p	s	numeric(p,s), decimal(p,s)
nchar nvarchar	char_length	空白	datetime, smalldatetime			datetime, smalldatetime
ntext	空白	空白	binary, varbinary			binary, varbinary
binary varbinary	char_length	空白	binary, varbinary	n		binary(n), varbinary(n)
image	空白	空白	image, sql_variant, timestamp, uniqueidentifier			image, sql_variant, timestamp, uniqueidentifier
			上記以外のデータ型が入力された場合は、データ型、桁数、位取りいずれも内容をチェックしません。			

## Dr.Sum データベースの場合

DB インポート時	CREATE 文出力時																																															
※DB インポート機能はサポートしていません	入力されたデータにより、カラム定義のデータ型を以下のように出力します。																																															
	<table><tr><th>データ型</th><th>桁数</th><th>位取</th><th>CREATE 文</th></tr><tr><td>VARCHAR, CHAR, VARYING CHARACTER</td><td></td><td></td><td>VARCHAR, CHAR, VARYING CHARACTER</td></tr><tr><td>VARCHAR, CHAR, VARYING CHARACTER</td><td>n</td><td></td><td>VARCHAR(n), CHAR(n), VARYING CHARACTER(n)</td></tr><tr><td>INTEGER, INT, SMALLINT, TINYINT</td><td></td><td></td><td>INTEGER, INT, SMALLINT, TINYINT  ※Dr.Sum 内部データ 型はすべて NUMERIC(33, 0)</td></tr><tr><td>NUMERIC, DECIMAL</td><td></td><td></td><td>NUMERIC, DECIMAL</td></tr><tr><td>NUMERIC, DECIMAL</td><td>n</td><td></td><td>NUMERIC(n), DECIMAL(n)</td></tr><tr><td>NUMERIC, DECIMAL</td><td>n</td><td>s</td><td>NUMERIC(n, s) DECIMAL(n, s)</td></tr><tr><td>REAL, FLOAT, DOUBLE</td><td></td><td></td><td>REAL, FLOAT, DOUBLE  ※Dr.Sum 内部データ 型はすべて REAL</td></tr><tr><td>DATE</td><td></td><td></td><td>DATE</td></tr><tr><td>TIME, TIMESTAMP, INTERVAL</td><td></td><td></td><td>TIME, TIMESTAMP, INTERVAL</td></tr><tr><td>TIME, TIMESTAMP, INTERVAL</td><td>n</td><td></td><td>TIME(n), TIMESTAMP(n), INTERVAL(n)</td></tr></table>				データ型	桁数	位取	CREATE 文	VARCHAR, CHAR, VARYING CHARACTER			VARCHAR, CHAR, VARYING CHARACTER	VARCHAR, CHAR, VARYING CHARACTER	n		VARCHAR(n), CHAR(n), VARYING CHARACTER(n)	INTEGER, INT, SMALLINT, TINYINT			INTEGER, INT, SMALLINT, TINYINT  ※Dr.Sum 内部データ 型はすべて NUMERIC(33, 0)	NUMERIC, DECIMAL			NUMERIC, DECIMAL	NUMERIC, DECIMAL	n		NUMERIC(n), DECIMAL(n)	NUMERIC, DECIMAL	n	s	NUMERIC(n, s) DECIMAL(n, s)	REAL, FLOAT, DOUBLE			REAL, FLOAT, DOUBLE  ※Dr.Sum 内部データ 型はすべて REAL	DATE			DATE	TIME, TIMESTAMP, INTERVAL			TIME, TIMESTAMP, INTERVAL	TIME, TIMESTAMP, INTERVAL	n		TIME(n), TIMESTAMP(n), INTERVAL(n)
データ型	桁数	位取	CREATE 文																																													
VARCHAR, CHAR, VARYING CHARACTER			VARCHAR, CHAR, VARYING CHARACTER																																													
VARCHAR, CHAR, VARYING CHARACTER	n		VARCHAR(n), CHAR(n), VARYING CHARACTER(n)																																													
INTEGER, INT, SMALLINT, TINYINT			INTEGER, INT, SMALLINT, TINYINT  ※Dr.Sum 内部データ 型はすべて NUMERIC(33, 0)																																													
NUMERIC, DECIMAL			NUMERIC, DECIMAL																																													
NUMERIC, DECIMAL	n		NUMERIC(n), DECIMAL(n)																																													
NUMERIC, DECIMAL	n	s	NUMERIC(n, s) DECIMAL(n, s)																																													
REAL, FLOAT, DOUBLE			REAL, FLOAT, DOUBLE  ※Dr.Sum 内部データ 型はすべて REAL																																													
DATE			DATE																																													
TIME, TIMESTAMP, INTERVAL			TIME, TIMESTAMP, INTERVAL																																													
TIME, TIMESTAMP, INTERVAL	n		TIME(n), TIMESTAMP(n), INTERVAL(n)																																													
	上記以外のデータ型が入力された場合は、データ型, 桁数,位取りいずれも内容をチェックしません。																																															

## デフォルト値

DB インポート時	CREATE 文出力時
データベースのテーブル定義から、カラムの DATA DEFAULT を取得してその値を表示します。	データが入力されている場合、カラム定義に”DEFAULT” 情報を出力します。

## NOTNULL

DB インポート時	CREATE 文出力時	
データベースのテーブル定義から、カラムの NULLABLE を取得し、NULLABLE 指定されている場合は○を表示します。	Oracle ※1	○が入力されている場合、カラム定義に”NOT NULL ENABLE”を出力します。
	PostgreSQL SQLServer	○が入力されている場合、カラム定義に”NOT NULL”を出力します。
	Dr.Sum ※2	○が入力されている場合、カラム定義に”NOT NULL”を出力します。

※1 DB インポート時に、Oracle では PRIMARY KEY に指定されているカラムは NOTNULL に○が表示されません。

※2 Dr.Sum では DB インポート機能はサポートしていません。

## UNIQUE

DB インポート時	CREATE 文出力時	
データベースのテーブル定義から、カラムの NULLABLE を取得し、NULLABLE 指定されている場合は○を表示します。	Oracle PostgreSQL SQLServer	○が入力されている場合、カラム定義に”CONSTRAINT UNIQUE” 情報を出力します。
	Dr.Sum ※	○が入力されている場合、カラム定義に”UNIQUE”を出力します。

※ Dr.Sum では DB インポート機能はサポートしていません。

## PRIMARY

DB インポート時	CREATE 文出力時	
データベースのテーブル定義から、カラムの制限情報を取得し、PRIMARY 指定されている場合はそのポジション番号(1, 2, 3, ...)を表示します。	Oracle PostgreSQL SQLServer	1 から始まる連続する数値が入力されている場合、カラム定義に”CONSTRAINT PRIMARY” 情報を出力します。
	Dr.Sum ※	1 から始まる連続する数値が入力されている場合、カラム定義に”COMPOUND KEY” 情報を表制約「UNIQUE NOT NULL」で出力します。  ただし、主キーを 1 つの列だけに指定したい場合は、PRIMARY 列に数値を入力する方法ではなく、NOTNULL 列、および UNIQUE 列に○を入力し、INDEX 列に 1 を入力してください。

※ Dr.Sum では DB インポート機能はサポートしていません。

## 備考

DB インポート時	CREATE 文出力時
データベースのテーブル定義から、カラムコメントの 2 番目の要素(※2)を取得して表示します。	データが入力されている場合、カラムコメントの 2 番目の要素(※2)として”COMMENT” 情報を出力します。

## ※2 カラムコメントの要素

Wize テーブル仕様書では、データベースのカラムコメントを、1 番目の要素(日本語項目名)と 2 番目の要素(カラムの備考)を、スペースで区切って格納するというルールを設定し、それによって DB インポート、CREATE 文出力を行います。

## INDEX

DB インポート時	CREATE 文出力時	
データベースのテーブル定義から、INDEX 情報、および UNIQUE INDEX 情報を取得し、INDEX 情報はポジション番号(1, 2, 3, ...)を表示します。 UNIQUE INDEX 情報はポジション番号を(U1, U2, ...)のように表示します。	Oracle PostgreSQL SQLServer	1 から始まる連続する数値が入力されている場合、“INDEX”を出力します。 U1, U2, U3, ... のように U が付加された数値が入力されている場合、“UNIQUE INDEX” 情報を出力します。
	Dr.Sum ※	入力された値に応じて、カラム定義の COMPOUND KEY 情報 または CREATE INDEX 文出力を行います。 詳細は下記の表を参照してください。

※ Dr.Sum では DB インポート機能はサポートしていません。

表: Dr.Sum における INDEX 指定方法

INDEX 列の記述	CREATE 文のカラム定義	CREATE INDEX 文
1, 2, ... n	COMPOUND KEY 情報 表制約「なし」	出力しない
U1, U2, ... Un	COMPOUND KEY 情報 表制約「UNIQUE」	出力しない
UN1, UN2, ... UNn	COMPOUND KEY 情報 表制約「UNIQUE NOT NULL」	出力しない
N1, N2, ... Nn	COMPOUND KEY 情報 表制約「NOT NULL」	出力しない
1 (1 つの列のみ指定)	出力しない	出力する

## 出力される CREATE 文の構成

CREATE 文を生成する処理では、Create 文作成用テンプレート内の各キーワードを所定の文字列と置換えます。下記に示したテンプレートは Oracle 用のものですが、テンプレートはデータベース種別ごとに個別に用意されているので、内容をカスタマイズすることが可能です。

```
ALTER SESSION SET NLS_LENGTH_SEMANTICS=' CHAR' ;
```

```

— プロジェクト名: %project%
— データベース: Oracle
— テーブル名: %table_name_j%
— 説明: %table_remark%
—
— %copyright%
```

```
— %table_name_j% のバックアップを作成(%table_name_e%_SV)
```

```

DROP TABLE %schema_name%%table_name_e_sv%;
CREATE TABLE %schema_name%%table_name_e_sv% NOLOGGING PARALLEL AS SELECT * FROM %schema_name%%table_name_e%;
```

```
— %table_name_j% を作成(%table_name_e%)
```

```

DROP TABLE %schema_name%%table_name_e%;
%CREATE_TABLE%
```

```
— バックアップからデータを復帰 (%table_name_e%_SV -> %table_name_e%)
```

```

SET serveroutput ON
DECLARE
  varret NUMBER;
  varmsg VARCHAR2(255);
BEGIN
  — 文字のサイズが大→小となる場合はエラー
  varret := PKG_WIZEDBMASTER.COPYSAMECOLUMNMNDATA(' %schema_name%%table_name_e_sv%', ' %schema_name%%table_name_e%', 'D');
  SYS.DBMS_OUTPUT.PUT_LINE(' return = ' || varret);
  IF (varret = 0) THEN
    varmsg := ' テーブルのコピーが成功しました。';
  ELSE
    varmsg := ' テーブルのコピーが失敗しました。';
  END IF;
  SYS.DBMS_OUTPUT.PUT_LINE(varmsg);
END;
/
```

```
— コメント
```

```

COMMENT ON TABLE %schema_name%%table_name_e% IS '%table_comment%';
%COMMENT_COLUMN_LIST%
```



---

— インデックス

---

%INDEX\_COLUMN\_LIST%

#INSERT\_UPDATE\_TRG\_START

---

— インサート, アップデートトリガー

---

```
CREATE OR REPLACE TRIGGER %tgiu_table_name_e%
BEFORE INSERT OR UPDATE ON %schema_name%%table_name_e%
FOR EACH ROW
BEGIN
    IF (INSERTING) THEN
        %INSERT_TRIGGER_LIST%
    ELSIF (UPDATING) THEN
        %UPDATE_TRIGGER_LIST%
    END IF;
END;
/
#INSERT_UPDATE_TRG_END
```

#LOGTABLE\_OUT\_START

---

— ログ出力用テーブルを作成

---

```
DROP TABLE %schema_name%%log_table_name_e%;
CREATE TABLE %schema_name%%log_table_name_e% (
    %upddate% DATE
    , %module% VARCHAR2(100)
    , %host% VARCHAR2(100)
    , %terminal% VARCHAR2(100)
    , %ip_address% VARCHAR2(15)
    , %os_user% VARCHAR2(100)
    , %opefig% CHAR(1)
    , %LOG_CREATE_TABLE_CLM_LIST%
);
```

---

— ログ出力トリガー

---

```
CREATE OR REPLACE TRIGGER %tgi_table_name_e%
AFTER INSERT OR UPDATE OR DELETE ON %schema_name%%table_name_e%
FOR EACH ROW
BEGIN
    IF (INSERTING) THEN
        INSERT INTO %schema_name%%log_table_name_e% (
            %upddate%
            , %module%
            , %host%
            , %terminal%
            , %ip_address%
```

```

, %os_user%
, %opeflg%
, %LOG_CLM_NAME_LIST%
) VALUES (
  CURRENT_DATE
, SYS_CONTEXT('userenv', 'module')
, SYS_CONTEXT('userenv', 'host')
, SYS_CONTEXT('userenv', 'terminal')
, SYS_CONTEXT('userenv', 'ip_address')
, SYS_CONTEXT('userenv', 'os_user')
, 'I'
, %LOG_CLM_DATA_NULL_LIST%
, %LOG_NEW_CLM_DATA_LIST%
);
ELSIF (UPDATING) THEN
  INSERT INTO %schema_name%%log_table_name_e% (
    %update%
  , %module%
  , %host%
  , %terminal%
  , %ip_address%
  , %os_user%
  , %opeflg%
  , %LOG_CLM_NAME_LIST%
  ) VALUES (
    CURRENT_DATE
  , SYS_CONTEXT('userenv', 'module')
  , SYS_CONTEXT('userenv', 'host')
  , SYS_CONTEXT('userenv', 'terminal')
  , SYS_CONTEXT('userenv', 'ip_address')
  , SYS_CONTEXT('userenv', 'os_user')
  , 'U'
  , %LOG_OLD_CLM_DATA_LIST%
  , %LOG_NEW_CLM_DATA_LIST%
  );
ELSIF (DELETING) THEN
  INSERT INTO %schema_name%%log_table_name_e% (
    %update%
  , %module%
  , %host%
  , %terminal%
  , %ip_address%
  , %os_user%
  , %opeflg%
  , %LOG_CLM_NAME_LIST%
  ) VALUES (
    CURRENT_DATE
  , SYS_CONTEXT('userenv', 'module')
  , SYS_CONTEXT('userenv', 'host')
  , SYS_CONTEXT('userenv', 'terminal')
  , SYS_CONTEXT('userenv', 'ip_address')
  , SYS_CONTEXT('userenv', 'os_user')

```

```

, 'D'
, %LOG_OLD_CLM_DATA_LIST%
, %LOG_CLM_DATA_NULL_LIST%
);
END IF;
END;
/
#LOGTABLE_OUT_END

%EOF%

```

※テーブル作成時にバックアップからデータを復帰する処理は、現在 Oracle 版のみの対応となります。  
 またこの機能を使用するためには、データベース用パッケージ **WIZE\_PKG** をインストールしておく必要があります。  
 インストール方法は『[4.2 データベース用パッケージのインストール](#)』を参照してください。

キーワードと置換える文字の対応

SQL Template 内キーワード	置換える文字
%project%	環境設定の”プロジェクト名”
%table_name_j%	各テーブル仕様の”テーブル名(日本語)”
%table_remark%	各テーブル仕様の”テーブル備考”
%copyright%	コピーライト (内部固定)
%table_name_e%	各テーブル仕様の”テーブル名(英語)”
%CREATE_TABLE%	各テーブル仕様から生成した CREATE TABLE 句全体
%schema_name%	各テーブル仕様の”スキーマ名”
%table_comment%	各テーブルの”テーブル名(日本語)”と”テーブル備考”をスペースで連結したもの
%COMMENT_COLUMN_LIST%	各テーブル仕様の”項目名(日本語)”と”備考”をスペースで連結したもの
%INSERT_TRIGGER_LIST%	“トリガー設定”シートから生成したインサートトリガー設定文
%UPDATE_TRIGGER_LIST%	“トリガー設定”シートから生成したアップデートトリガー設定文
%LOG_CREATE_TABLE_CLM_LIST%	ログテーブル生成用の列定義
%LOG_CLM_NAME_LIST%	ログ出力トリガー用の列名
%LOG_CLM_DATA_NULL_LIST%	ログ出力トリガー用の NULL データ
%LOG_OLD_CLM_DATA_LIST%	ログ出力トリガー用の更新前データ(OLD)
%LOG_NEW_CLM_DATA_LIST%	ログ出力トリガー用の更新後データ(NEW)
%EOF%	置換えなし (ファイルの終りを判定するため)

## シートの印刷

- 印刷範囲はテーブル仕様データが入力されている領域のみに限定しています。  
 ただしテーブル仕様で INDEX 定義がひとつもない場合でも、INDEX 領域の 2 列分は印刷範囲に含まれます。
- ヘッダ部の表示  
 テーブル名(日本語), 印刷日時を表示します。
- ヘッダ部の表示  
 ページ番号/総ページ数、Copyright を表示します。

## 7) 日本語-英語変換機能

“テーブル一覧シート”のテーブル名(日本語)、テーブル名(英語)、および“テーブル仕様”シートの項目名(日本語)、項目名(英語)の入力作業を自動化するための機能です。

#DICシートに入力された変換用のデータを参照して、変換元に指定したデータを日本語→英語、または英語→日本語に一括変換することが可能です。

また、データ型、桁数、位取、デフォルト値、NOTNULL、UNIQUE、備考の値を定義しておくことにより、日本語-英語変換時に指定データが完全一致したときに、これらの設定値をテーブル仕様書にコピーすることが可能です。

### 1) テーブル一覧表

- ① [テーブル追加]ボタンを押下して表示されるテーブル追加ダイアログで、以下の操作が可能になります。
  - ・テーブル名(日本語)を入力し、[日本語→英語変換]ボタンを押下すると、テーブル名(英語)に対応する英語名が自動的に入力されます。
  - ・テーブル名(英語)を入力し、[英語→日本語変換]ボタンを押下すると、テーブル名(日本語)に対応する日本語名が自動的に入力されます。

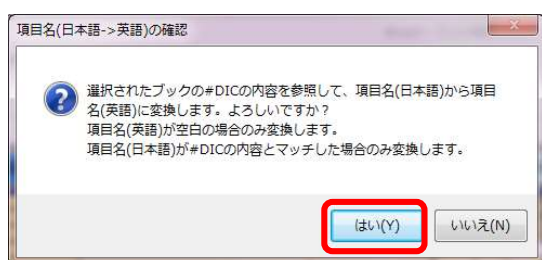
- ② [項目名変換(日→英)], [[項目名変換(英→日)]ボタンで、すべてのテーブル仕様シートの項目名について日本語-英語変換ができます。詳細は [2\) テーブル仕様書](#)を参照してください。

## 2) テーブル仕様書

テーブル仕様書シートで、以下の操作が可能になります。

- ・ 項目名(日本語)にデータを入力後、[項目名変換(日->英)]ボタンを押下すると、項目名(英語)に対応する英語名が自動的に入力されます。
- ・ 項目名(英語)にデータを入力後、[項目名変換(英->日)]ボタンを押下すると、項目名(日本語)に対応する日本語名が自動的に入力されます。

<例> 項目名変換(日->英)ボタンを押下し、No.4 の項目を自動入力



<辞書データの設定>



※本ページの画面表示例は Ver1.8.2 以前のものです。

### 3) 日本語-英語変換用辞書データの設定

日本語-英語変換機能を使用するためには、あらかじめ“#DIC”シートに変換用データを入力しておく必要があります。

#### (1) 日本語-英語変換用データ

項目名(日本語), 項目名(英語)の欄に変換用データを入力します。

項目名(日本語), 項目名(英語)の両方にデータが入力されている場合のみ、変換対象となります。

#### (2) データ型等の設定用データ

完全一致チェック欄に {(空白), ○, Oracle, PostgreSQL, SQLServer} のいずれかを入力することにより、日本語-英語変換と同時に、以下の処理を行うことが可能です。

#DIC シートの”完全一致チェック”	テーブル仕様書の”データ型”～”備考”
空白	変更しない
○	#DIC シートの”データ型”～”備考”をコピー
Oracle	環境設定のデータベース種別が”Oracle”の場合のみ、#DIC シートの”データ型”～”備考”をコピー
PostgreSQL	環境設定のデータベース種別が”PostgreSQL”の場合のみ、#DIC シートの”データ型”～”備考”をコピー
SQLServer	環境設定のデータベース種別が”SQLServer”の場合のみ、#DIC シートの”データ型”～”備考”をコピー
上記以外の文字	変更しない

ただし、「備考」については以下の処理を行います。

#DIC シートの”備考”	テーブル仕様書の”備考”
空白	変更しない
データが入力されている	#DIC シートの”備考”をコピー

※各テーブル仕様書の”備考”を個別に設定したい場合は、#DIC の”備考”は空白にしてください。

完全一致チェック欄に”○”以外を入力した場合は「データ型,桁数,位取,デフォ値,NOTNULL,UNIQUE,備考」のコピーは行いません。

#DIC シートの設定は以下の例を参照してください。

63 / 68



## 8) トリガー設定機能

CREATE 文出力時に、以下の 2 種類のトリガー設定文を出力する機能です。※Dr.Sum は未対応  
この機能を使用するためには、テーブル仕様シートの CREATE 文出力オプションで、“インサート,アップデートトリガー”にチェックを付けてください。(Ver1.9.1 以降)

CREATE文出力オプション	
<input type="checkbox"/> テーブルバックアップ	<input type="checkbox"/> テーブルのコメント
<input checked="" type="checkbox"/> インサート,アップデートトリガー	<input type="checkbox"/> ログテーブル出力
<input type="checkbox"/> テーブルリストア	

※Ver1.9.0 以前のバージョンでは、トリガー設定がされていれば、インサート,アップデートトリガーは常に出力します。

## 1) インサート/アップデートトリガー

あらかじめ“#トリガー設定”シートに必要なデータを入力しておくことにより、各テーブルに対して“作成”(INSERT)、“更新”(UPDATE) が実行されたときに、指定した列に対して自動的にデータをセットするためのトリガー設定文を出力します。

“#トリガー設定”シートの設定例を以下に示します。

No.	名称	列名(英語)	定義型	桁数	位取	追加	更新	設定データ	DB種別
1	作成日時	InsertDate	DATE			○		NEW.InsertDate = SYSDATE	Oracle
2	作成モジュール	InsertModule	VARCHAR2	128		○		NEW.InsertModule = SYS_CONTEXT('userenv','module')	Oracle
3	作成ホスト	InsertHost	VARCHAR2	128		○		NEW.InsertHost = SYS_CONTEXT('userenv','host')	Oracle
4	作成ターミナル	InsertTerminal	VARCHAR2	128		○		NEW.InsertTerminal = SYS_CONTEXT('userenv','terminal')	Oracle
5	作成IPアドレス	InsertIPAddress	VARCHAR2	16		○		NEW.InsertIPAddress = SYS_CONTEXT('userenv','ip_address')	Oracle
6	作成作業者	InsertUserName	VARCHAR2	128		○		NEW.InsertUserName = SYS_CONTEXT('userenv','os_user')	Oracle
7	更新日時	UpdateDate	DATE				○	NEW.UpdateDate = SYSDATE	Oracle
8	更新モジュール	UpdateModule	VARCHAR2	128			○	NEW.UpdateModule = SYS_CONTEXT('userenv','module')	Oracle
9	更新ホスト	UpdateHost	VARCHAR2	128			○	NEW.UpdateHost = SYS_CONTEXT('userenv','host')	Oracle
10	更新ターミナル	UpdateTerminal	VARCHAR2	128			○	NEW.UpdateTerminal = SYS_CONTEXT('userenv','terminal')	Oracle
11	更新IPアドレス	UpdateIPAddress	VARCHAR2	16			○	NEW.UpdateIPAddress = SYS_CONTEXT('userenv','ip_address')	Oracle
12	更新作業者	UpdateUserName	VARCHAR2	128			○	NEW.UpdateUserName = SYS_CONTEXT('userenv','os_user')	Oracle
13	作成日時	InsertDate	DATE				○	new."INSERT DATE" = current_date	PostgreSQL
14	作成IPアドレス	InsertIPAddress	INET				○	new."INSERT IPADDRESS" = inet_client_addr()	PostgreSQL
15	作成作業者	InsertUserName	NAME				○	new."INSERT USERNAME" = current_user	PostgreSQL
16	更新日時	UpdateDate	DATE				○	new."UPDATE DATE" = current_date	PostgreSQL
17	更新IPアドレス	UpdateIPAddress	INET				○	new."UPDATE IPADDRESS" = inet_client_addr()	PostgreSQL
18	更新作業者	UpdateUserName	NAME				○	new."UPDATE USERNAME" = current_user	PostgreSQL
19	作成日時	InsertDate	DATETIME				○	InsertDate = GETDATE()	SQLServer
20	作成ホスト	InsertHost	NVARCHAR	128			○	InsertHost = HOST_NAME()	SQLServer
21	作成IPアドレス	InsertIPAddress	VARCHAR	48			○	InsertIPAddress = (SELECT client_net_address FROM sys.dm_exec_connections WHERE session_id = @@SPID)	SQLServer
22	作成作業者	InsertUserName	NVARCHAR	128			○	InsertUserName = SUSER_NAME()	SQLServer
23	更新日時	UpdateDate	DATETIME				○	UpdateDate = GETDATE()	SQLServer
24	更新ホスト	UpdateHost	NVARCHAR	128			○	UpdateHost = HOST_NAME()	SQLServer
25	更新IPアドレス	UpdateIPAddress	VARCHAR	48			○	UpdateIPAddress = (SELECT client_net_address FROM sys.dm_exec_connections WHERE session_id = @@SPID)	SQLServer
26	更新作業者	UpdateUserName	NVARCHAR	128			○	UpdateUserName = SUSER_NAME()	SQLServer

“#トリガー設定”シートの設定項目と機能の対応

列名	機能
No.	行番号(トリガー設定文に出力されないの、任意の値を入力してください)
名称	各行の名称(トリガー設定文に出力されないの、任意の値を入力してください)
列名(英語)	トリガー出力対象の列名を入力します。列名がテーブルに存在する場合にトリガー設定文を出力します。
定義型	列名(英語)のデータ型を入力します。
桁数	列名(英語)の桁数を入力します。
位取	列名(英語)の位取を入力します。
追加	○が入力された場合、INSERT トリガー設定文(TGIU_{テーブル名(英語)})を出力します。
更新	○が入力された場合、UPDATE トリガー設定文(TGIU_{テーブル名(英語)})を出力します。
設定データ	トリガー設定文で実行する処理を入力します。
DB 種別	データベース種別をドロップダウンリストから選択します。



“環境設定” のデータベース種別と一致する項目のみトリガー設定文を出力します。
-----------------------------------------

トリガー設定文の名称は INSERT, UPDATE 共通で、TGIU\_{テーブル名(英語)} となります。

※SQLServer の場合、プライマリーキーが存在するテーブルのみインサート/アップデートトリガー文を出力します。

#### ログ出力トリガー

対象テーブルが “作成” (INSERT), “更新” (UPDATE), “削除” (DELETE) が実行されたときに、対応するログテーブルを生成し、自動的にログデータを出力するためのトリガー設定文を出力します。

この機能を有効にするためには、以下の設定が必要です。

設定方法はバージョンにより異なります

テーブル仕様書のバージョン	ログ出力トリガー機能を有効にする
Ver1.3 ~ Ver1.9.0	ログテーブル出力欄に”○”を入力する
Ver1.9.1 以降	CREATE 文出力オプションで、“ログテーブル出力”にチェックを付ける

#### <Ver1.3～Ver1.9.0>の画面

ログテーブル出力	○
----------	---

#### <Ver1.9.1 以降>の画面

CREATE文出力オプション			全項目 ON
<input type="checkbox"/> テーブルバックアップ	<input type="checkbox"/> テーブルのコメント	<input type="checkbox"/> インサート, アップデートトリガー	全項目 OFF
<input type="checkbox"/> テーブルリストア	<input checked="" type="checkbox"/> ログテーブル出力		

ログテーブルは、処理対象テーブルに以下の項目を追加した構成になります。

#### <Oracle 用>

列名	データ型	設定されるデータ
UPDDATE	DATE	CURRENT_DATE
MODULE	VARCHAR(100)	SYS_CONTEXT('userenv', 'module')
HOST	VARCHAR(100)	SYS_CONTEXT('userenv', 'host')
TERMINAL	VARCHAR(100)	SYS_CONTEXT('userenv', 'terminal')
IP_ADDRESS	VARCHAR(15)	SYS_CONTEXT('userenv', 'ip_address')
OS_USER	VARCHAR(100)	SYS_CONTEXT('userenv', 'os_user')
OPEFLG	CHAR(1)	INSERT 時: 'I' UPDATE 時: 'U' DELETE 時: 'D'
BEF_{列名}	{列名}に対応	INSERT 時: NULL UPDATE 時, DELETE 時: :OLD.{列名}
AFT_{列名}	{列名}に対応	INSERT 時, UPDATE 時: :NEW.{列名} DELETE 時: NULL

## &lt;PostgreSQL 用&gt;

列名	データ型	設定されるデータ
UPDDATE	DATE	CURRENT_DATE
IP_ADDRESS	VARCHAR(15)	INET_CLIENT_ADDR0
DB_USER	VARCHAR(100)	CURRENT_USER
OPEFLG	CHAR(1)	INSERT 時: 'I' UPDATE 時: 'U' DELETE 時: 'D'
BEF_{列名}	{列名}に対応	INSERT 時: NULL UPDATE 時, DELETE 時: OLD.{列名}
AFT_{列名}	{列名}に対応	INSERT 時, UPDATE 時: NEW.{列名} DELETE 時: NULL

## &lt;SQLServer 用&gt;

列名	データ型	設定されるデータ
UPDDATE	DATETIME	GETDATE()
HOST	VARCHAR(100)	HOST_NAME()
IP_ADDRESS	VARCHAR(15)	(SELECT client_net_address FROM sys.dm_exec_connections WHERE session_id = @@SPID)
DB_USER	VARCHAR(100)	SUSER_NAME()
OPEFLG	CHAR(1)	INSERT 時: 'I' UPDATE 時: 'U' DELETE 時: 'D'
BEF_{列名}	{列名}に対応	INSERT 時: NULL UPDATE 時, DELETE 時: @DEL_{列名}
AFT_{列名}	{列名}に対応	INSERT 時, UPDATE 時: @INS_{列名} DELETE 時: NULL

※SQLServer の場合、INSERT 文、UPDATE 文を 1 回実行するごとに、ログテーブルにデータが 2 件ずつ記録されます。SQLServer では、インサート/アップデートトリガ処理内で対象テーブルに対して UPDATE 文を実行しているためです。

INSERT 時: UPDATE & INSERT、UPDATE 時: UPDATE & UPDATE

## 6. 処理実行時のチェック項目

- ・ Wize テーブル仕様書では、各処理実行時に以下のようなチェックを行います。
- ・ エラーが検出されると対応するメッセージを表示します。

### 1) テーブル一覧のチェック項目

#### [テーブル一覧更新] 実行時

- ・ テーブル仕様が 1 つ以上作成されていること。
- ・ テーブル仕様の “テーブル名(日本語)” がシート名と一致していること。

#### [DB インポート(テーブル仕様)] 実行時

- ・ テーブル一覧の “テーブル名(日本語)” が空白でないこと。
- ・ テーブル一覧の “テーブル名(日本語)” に、Excel シート名に使用できない文字が含まれていないこと。

Excel のシート名に使用できない文字は以下のとおりです(小文字, 大文字ともに)  
 “:”, “¥”, “/”, “?”, “\*”, “[”, “]”

- ・ テーブル一覧の “テーブル名(日本語)” が重複していないこと。

### 2) テーブル仕様のチェック項目

#### [SQL 文出力] 実行時

- ・ テーブル仕様の “テーブル名(英語)” が空白でないこと。  
 ※シートメニュー/◆SELECT 文を出力(クリップボード)～◆全種類の SQL 文を出力(クリップボード)  
 選択時も同様

#### セルメニュー/◆選択行を辞書に登録 実行時

- ・ テーブル仕様の “テーブル名(英語)” が辞書(#DIC シート)に登録されていないこと。

#### [DB インポート(テーブル仕様)] 実行時

- ・ テーブル仕様の “テーブル名(英語)” が空白でないこと。
- ・ テーブル仕様の “テーブル名(日本語)” が空白でないこと。

#### [CREATE 文出力] 実行時

- ・ テーブル仕様の “テーブル名(英語)” が空白でないこと。
- ・ テーブル仕様の “テーブル名(日本語)” が空白でないこと。
- ・ テーブル仕様の “項目名(英語)” が空白でないこと。
- ・ テーブル仕様の “データ型” が空白でないこと。

## 7. おわりに

本ソフトをダウンロードしていただきまして、ありがとうございます。

感想や要望、バグなどありましたらお手数ですが [info@wizejp.com](mailto:info@wizejp.com) までご連絡ください。