Code engineering set
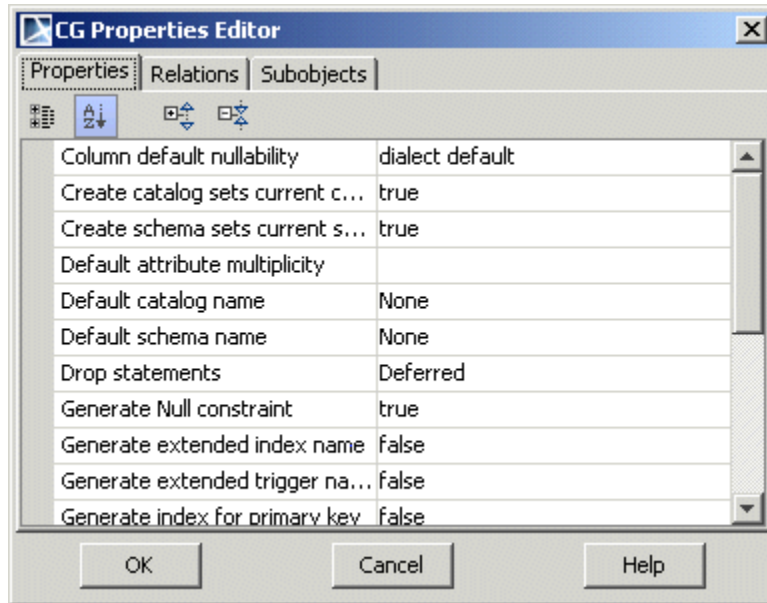
On this page

You can create code engineering sets for database scripts in the same manner as CE sets for other code types (see Code engineering set in MagicDraw User Guide). Right-click the Code engineering Sets, New, DDL, and then the appropriate database flavor. When the CE set is created, you can add database model elements to it, after which DDL script file(s) can be generated OR you can add the script files to the CE set and reverse-engineer them into the database models. In addition to reversing from files, there is Reverse from the DB radio button. Once it is switched, the options for JDBC connection configuring appear, allowing you to set up a connection to the live database.

Box name	Description
Recently Used	Contains the list of the recently used reverse templates. Choose the one you need and click **Apply**.
DB Connection URL	The connection URL for the selected profile.
Driver Files	Contains .jar and .zip files or directories with JDBC driver's classes. To choose the files or directories you want to add or remove, click the **...** button. The **Select Files and/or Directories** dialog appears. ⚠️ **Note** If the driver file is empty, the Driver Class is searched from the classpath.
Driver Class	Contains the connection driver class. Click the **...** button to display the list of available driver classes available in the selected driver files. ⚠️ **Note** The system searches for driver classes only in the files selected in the **Driver Files** list.
Username	Type the username to connect to the database.
Password	Type the password to connect to the database.
Catalog	Contains the name of the selected Catalog. To retrieve the list of available Catalogs from the database, click the **...** button and select the catalog. The catalog name appears in the **Catalog** textbox. ⚠️ **Note** The list of catalogs can be retrieved only when all other properties in this dialog box are correctly defined.
Schema	Contains a name of the selected Schema. To retrieve the list of available Schemas from the database, click the **...** button and select the schema. The schema name appears in the **Schema** textbox. ⚠️ **Note** The list of schemas can be retrieved only when all other properties in this dialog box are correctly defined.
Property Name	The name of the JDBC driver property. ⚠️ **Note** If using Oracle drivers, while retrieving db info from Oracle db • To retrieve comments on table and column, set property as remarks=true. • To connect to a db as sysdba, set property as internal_logon=sysdba.
Debug JDBC Driver	If selected, all output from a JDBC driver will be directed to Message Window.
Reload Driver	The Reload Driver check box is selected by default. If you do not want that driver to be reloaded, clear the check box.

Properties of Code Engineering Set for DDL

Two separate properties sets are stored as the properties of code engineering set for DDL:

- Properties for DDL script generation
- Properties for DDL script reverse engineering

DDL properties in CG Properties Editor dialog.

Property name	Value list	Description
Properties for DDL generation		
Default attribute multiplicity	**0, 0..1**, any entered by user	If the attribute multiplicity is not specified, the value of this property is used.
Generate Null constraint	**True**, **false** (default)	If true, generates a NULL constraint for the column attribute with [0..1] multiplicity. If the DBMS you use supports NULL, you can enable this to generate NULL constraints. See also: GenerateNotNullConstraint, AttributeDefaultMultiplicity
Generate extended index name	**True**, **false** (default)	If true, generates an index name of the form: TableName_IndexName.
Generate extended trigger name	**True**, **false** (default)	If true, generates a trigger name of the form: TableName_TriggerName.
Generate index for primary key	**True** (default), **false**	If the DBMS you use requires explicit indexes for the primary key, you can enable explicit index creation using this flag. See also: GenerateIndexForUnique
Generate index for unique	**True** (default), **false**	If the DBMS you use requires explicit indexes for the primary key or unique columns, you can enable explicit index creation using this flag. See also: GenerateIndexForPK
Generate not Null constraint	**True** (default), **false**	If true, generates a NOT NULL constraint for the column attribute with [1] multiplicity. If you set a GenerateNullConstraint, you may not wish to generate the NOT NULL constraint. See also: GenerateNullConstraint, AttributeDefaultMultiplicity

Generate qualified names	**True** (default), **false**	If the value of the Generate Qualified Names check box is true, the package name is generated before the table or view name. For example: «Database» package "MQOnline" includes «Table» class "libraries". If the check box Generate Qualified Names is selected as true in the generated source, it would be written as CREATE TABLE MQOnline.libraries. If the check box Generate Qualified Names is selected as false, in the generated source it would be written as CREATE TABLE libraries.
Generate quoted identifiers	**True**, **false** (default)	Specifies whether DDL code generator should generate quoted names of identifiers.
Object creation mode	The Object Creation Mode combo box has the following options. - only CREATE statements - DROP & CREATE statements - CREATE OR REPLACE statements (only for Oracle dialect;default for this dialect) - DROP IF EXISTS & CREATE statements (only for MySQL dialect; default for this dialect).	
Properties for DDL script reverse engineering		
Column default nullability	**Dialect default** (default), **not specified**, **NULL**, **NOT NULL**	If column has no NULL or NOT NULL constraint specified, the value of this property is used.
Create catalog sets current catalog	**True** (default), **false**	Specifies whether create catalog statement changes current catalog name.
Create schema sets current schema	**True** (default), **false**	Specifies whether create schema statement changes current schema name.
Default catalog name	**DefaultCatalogNone** (default), **DefaultCatalogPackage**, any entered by the user	Specifies current database name. Used when DDL script does not specify database name explicitly.
Default schema name	**DefaultSchemaNone** (default), **DefaultSchemaPackage**, any entered by the user	Specifies current schema name. Used when DDL script does not specify schema name explicitly.
Drop statements	**Deferred** (default), **Immediate**, **Ignored**	Specifies whether execution of drop statements may be deferred, must be executed, or must be ignored. Deferred drop may be enabled if elements are recreated later. This will save existing views. Attribute stereotypes, multiplicity and default value are not always dropped immediately.
Map Null / not Null constraints to	**Stereotypes** (default), **Multiplicity**	When parsing DDLs, the null / not null constraints are modeled as either stereotype tag values or multiplicity.
Map foreign keys	**True** (default), **false**	An association with «FK» stereotype on the association end is created, to represent a Foreign Key.
Map indexes	**True** (default), **false**	A constraint with «Index» stereotype is added into the table, to represent the index.
Map triggers	**True** (default), **false**	An opaque behavior with «Trigger» stereotype is added into the table to represent a trigger.
Map views	**True** (default), **false**	A class with «ViewTable» stereotype is created to represent the view.

Related page

- Code engineering set