

Managing code engineering sets

The [com.nomagic.magicdraw.ce.CodeEngineeringManager](#) class provides static API methods for creating, removing, reversing, generating, and adjusting code-engineering settings. The [CodeEngineeringManager](#) class is used for managing code engineering sets.

When creating the code engineering set, the following fields are required:

- Language and dialect IDs. All languages and dialects IDs are stored in the [com.nomagic.magicdraw.ce.CodeEngineeringConstants](#) class.
- The name of the [code engineering set](#).
- The project, where code engineering set will be added.
- The working package.
- The working directory.

New [com.nomagic.magicdraw.ce.CodeEngineeringSet](#) can be created directly via [CodeEngineeringManager.createCodeEngineeringSet\(java.lang.String, java.lang.String, java.lang.String, Project, Package, java.lang.String\)](#) method.

The language and dialect can be selected from the [CodeEngineeringConstants](#) class. The dialect is required for DDL and C++ languages.



- null *workingPackage* means, that the code engineering working package is *Model* by default
- null *workingDirectory* means, that the code engineering working directory is the modeling tool install root

To remove the instance of [CodeEngineeringSet](#) use the [CodeEngineeringManager.removeCodeEngineeringSet\(CodeEngineeringSet\)](#) method.

All Code Engineering Sets can be retrieved by the [CodeEngineeringManager.getAllCodeEngineeringSets\(Project\)](#) method.

To perform generation of the [CodeEngineeringSet](#) object use the [CodeEngineeringManager.generate\(Project, boolean\)](#) method.

To perform the reverse engineering use the [CodeEngineeringManager.reverse\(Project, boolean\)](#) method.

Language Specific Options

[com.nomagic.magicdraw.ce.JavaCodeEngineeringManager](#) provides methods to set Java specific options. These options apply to all code engineering sets in the project.

To add a classpaths for the java code engineering sets, use [JavaCodeEngineeringManager.setJavaClasspath\(Project, java.lang.String\[\]\)](#)



Setting a new classpath will override the old one.

To get applied classpaths for java code engineering sets, use [JavaCodeEngineeringManager.getJavaClasspath\(Project\)](#).

To resolve collection generics when reversing the java code, [CodeEngineeringManager.setResolveCollectionGenerics\(Project, boolean\)](#)