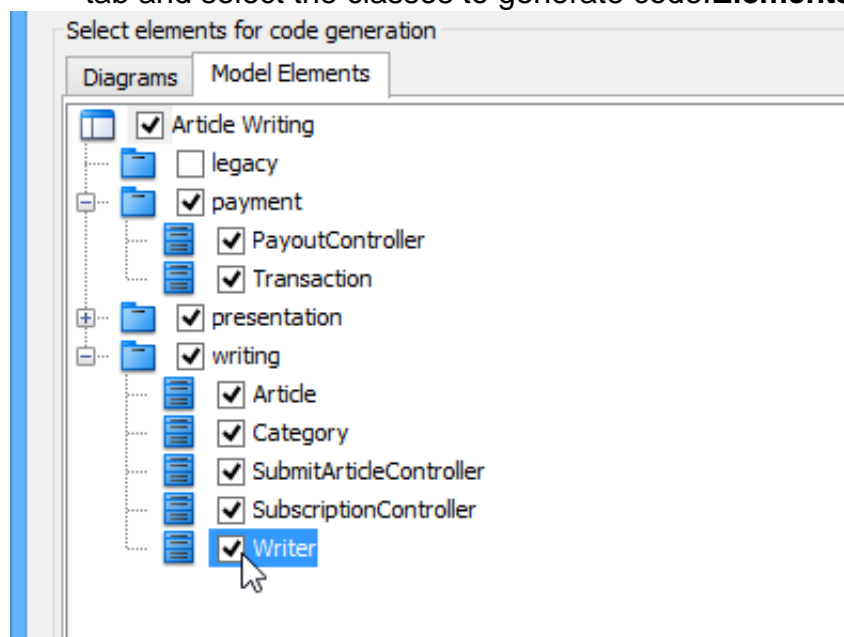

How to generate C# from UML

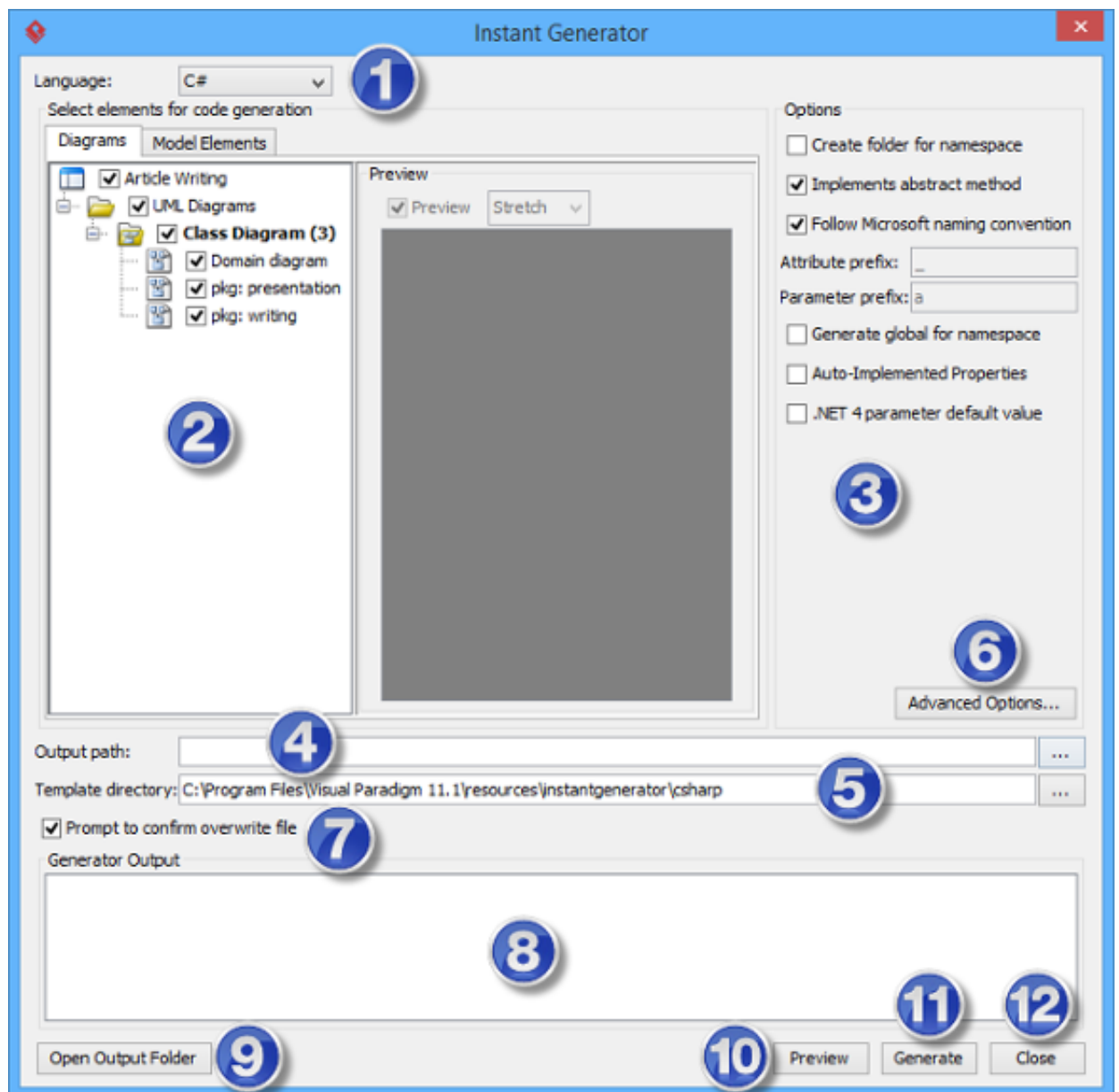
. Designers or [UML class model](#) is the process of producing source code from [Instant Generator](#) software architects can build a high level domain class model, then pass to programmer to perform more lower-level system or application modeling and eventually generate source code from implementation model. This chain makes building software faster and cheaper. In this chapter, we will go through the instant generation of C# source code. To generate code by instant generator:

from the toolbar. **Tools > Code > Instant Generator** Select .1
. **Language** as the **C#** window, select **Instant Generator** In the .2
, which is the directory where you want the code to generate to. **Output Path** Fill in the .3
tab, you can select the diagrams **Diagrams** Select the classes to generate code. In the .4
Model to generate code for classes in the selected diagrams. Alternatively, open the
tab and select the classes to generate code. **Elements**



Optionally configure the generator options. Read the section below for a description of .5
options.
to generate code. **Generate** Click .6

Overview of Instant Generator

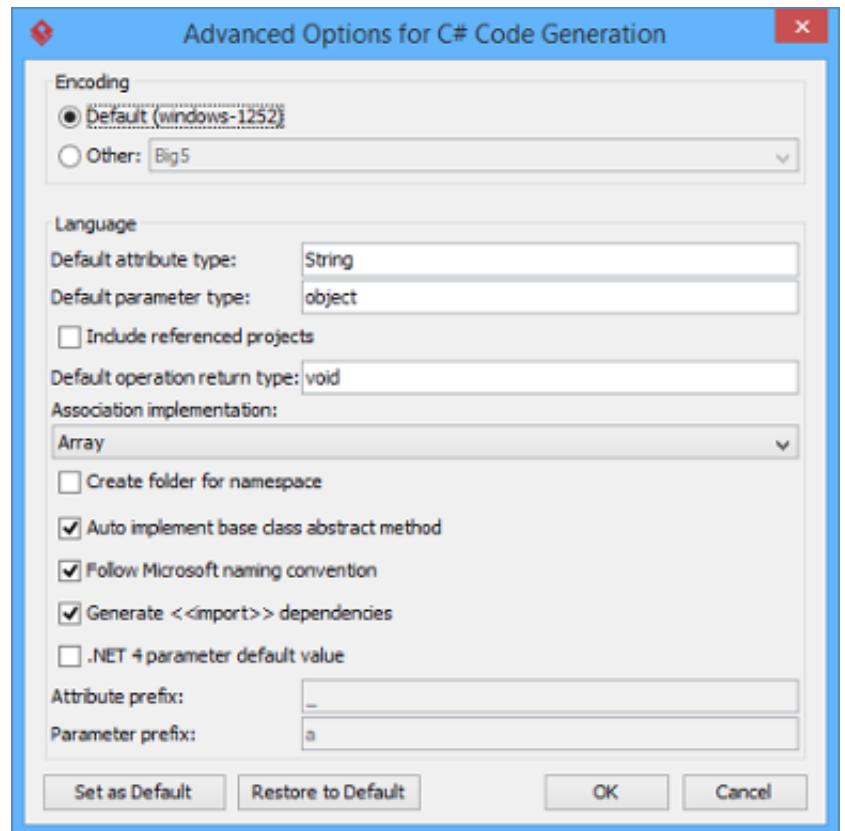


	Description	Name	No.
The programming language to generate.	Language		1
A list of packages and classes that can be selected for code generation. You must select classes for code generation.	Model element tree		2
Some of the common configurable options are shown here. You can configure them in advanced options.	General options		3
The folder where you want the code files to be generated.	Output path		4
Template governs how code will be generated from model to code. You can customize	Template directory		5

	Description	Name	No.
the template to suit your needs, such as to print company specific headers to each code file. If you want to use your own template, provide the template directory here. If you want to keep using the build in template, leave this option unchanged to generate Visual Paradigmlet with build in template. To learn more about customization, read the final chapter of this part.			
Click this button to configure any options related to code generation in a new window.	Advanced options		6
If a code file instant generator going to generate is already exist, by checking this option you will be asked whether to overwrite that file or not. If you uncheck this option, it will help you to overwrite the existing file automatically.	Prompt to confirm overwrite file		7
Any warning, error or progress about generation will be printed here.	Output pane		8
Open the output path with system browser.	Open output folder		9
Click to preview the code content. It is just a preview and code will not be generated to the output path by previewing.	Preview		10
Click to start generation.	Generate		11
Click to close the instant generator.	Close		12

Generator options

window you can configure some of the common code options at the **Instant Generator**On the right of window. You can also configure the advanced options for more detailed settings by button.**Advanced Options...**clicking the



Below is a description of available options.

	Description	Option
The encoding of source file.		Encoding
Attribute type that will be used when attribute has no type specified.		Default attribute type
Parameter type that will be used when parameter has no type specified.		Default parameter type
Check to generate also classes in referenced project.		Allow From Linked Project
Operation return type that will be used when operation has no return type specified.		Default operation return type
The type of collection to be used for association.		Association implementation
Create a directory in system for namespace		Create folder for namespace
Whether or not to generate operations for implementing abstract operations defined in super class.		Auto implement base class abstract method
Make the code convention follow Microsoft		Follow Microsoft naming convention
The text to append to attribute name as prefix.		Attribute prefix
The text to append to parameter name as prefix.		Parameter prefix