How to generate C# from UML

. Designers or <u>UML class model</u> is the process of producing source code from <u>Instant Generator</u> 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 GeneratorSelect .1
- .Languageas the C# window, selectInstant GeneratorIn the .2
- , which is the directory where you want the code to generate to. Output PathFill 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 Select elements for code generation Model Elements Diagrams Article Writing legacy ė... payment PayoutController Transaction presentation <u>ب</u> writing Article Category SubmitArticleController SubscriptionController /riter

- 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

Instant Generator	×
Language: C# Select elements for code generation Diagrams Model Elements Article Writing UML Diagrams UML Diagrams O Class Diagram (3) Preview Stretch V Preview Stretch V pkg: writing pkg: writing	Options Create folder for namespace Implements abstract method Follow Microsoft naming convention Attribute prefix: Parameter prefix: Cenerate global for namespace Auto-Implemented Properties NET 4 parameter default value
Output path:	5
Open Output Folder	O Preview Generate Close

	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

	De	scription	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	Advanced options			6
any options related to code				
generation in a new window.				
If a code file instant generator	Prompt to confirm overwrite			7
going to generate is already	file			
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.	-			-
Any warning, error or progress	Output pane			8
about generation will be				
printed here.				-
Open the output path with	Open output folder			9
system browser.				
Click to preview the code	Preview			10
content. It is just a preview				
and code will not be generated				
to the output path by				
previewing.	2			
Click to start generation.	Generate			11
Click to close the instant	Close			12
generator.				

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

Advanced	Options for C# Code Generation
Default (windows-1252) Other: Big5	
	Y
Language	
Default attribute type:	String
Default parameter type:	object
Include referenced project	ts
Default operation return type:	void
Association implementation:	
Array	~
Create folder for namespare Auto implement base class Follow Microsoft naming of Complete class of the second sec	abstract method
✓ Generate < <import>> de</import>	pendencies
.NET 4 parameter default	value
Attribute prefix:	_
Parameter prefix:	a
Set as Default Resto	re to Default OK Cancel

Below is a description of available options.

	Description	Option
source file.		Encoding
en attribute	Default atti	ribute type
e specified.		
used when	Default parar	neter type
e specified.		
referenced	Allow From Link	ed Project
project.		
used when	Default operation r	eturn type
e specified.		
be used for	Association imple	mentation
ssociation.		
namespace	Create folder for na	•
erations for	Auto implement base class abstra	ct method
s defined in		
uper class.		
w Microsoft	Follow Microsoft naming o	
e as prefix.		oute prefix
er name as	Param	eter prefix
prefix.		

The encoding of source file. Attribute type that will be used when attribute has no type specified. Parameter type that will be used when parameter has no type specified. Check to generate also classes in referenced project.

Operation return type that will be used when operation has no return type specified. The type of collection to be used for association.

Create a directory in system for namespace Whether or not to generate operations for implementing abstract operations defined in super class.

Make the code convention follow Microsoft The text to append to attribute name as prefix. The text to append to parameter name as prefix.