

WebServices - Axis

1. Axis-java2wsdl Task

Generates a WSDL description from a Java class.

1.1. Description

Generates a WSDL description from a Java class.

1.2. Parameters

Attribute	Description	Type
bindingname	The name to use use for the binding element. If not specified, the value of the servicePortName + "SoapBinding" is used.	java.lang.String
classname	the class name to import, eg. org.example.Foo. Required. The class must be on the classpath.	java.lang.String
exclude	Comma separated list of methods to exclude from the wsdl file.	java.lang.String
extraclasses	The setter for the "extraClasses" attribute	java.lang.String
implclass	Sometimes extra information is available in the implementation class file. Use this option to specify the implementation class.	java.lang.String
input	Optional attribute that indicates the name of the input wsdl file. The output wsdl file will contain everything from the input wsdl	java.io.File

	file plus the new constructs. If a new construct is already present in the input wsdl file, it is not added. This option is useful for constructing a wsdl file with multiple ports, bindings, or portTypes.	
location	The url of the location of the service. The name after the last slash or backslash is the name of the service port (unless overridden by the -s option). The service port address location attribute is assigned the specified value.	java.lang.String
locationimport	the location of the interface WSDL when generating an implementation WSDL Required when <code>outputImpl</code> is set	java.lang.String
methods	If this option is specified, only the indicated methods in your interface class will be exported into the WSDL file. The methods list must be comma separated. If not specified, all methods declared in the interface class will be exported into the WSDL file	java.lang.String
namespace	the target namespace. Required.	java.lang.String
namespaceimpl	Namespace of the implementation WSDL.	java.lang.String
output	The name of the output WSDL file. If not specified, a suitable default WSDL file is written into the current directory.	java.io.File
outputimpl	Use this option to indicate the name of the output implementation WSDL file. If specified, Java2WSDL will produce separate interface and	java.io.File

	implementation WSDL files. If not, a single WSDL file is generated	
porttypename	Indicates the name to use for the portType element. If not specified, the class-of-portType name is used.	java.lang.String
serviceelementname	the name of the service element. If not specified, the service element is the portTypeNameService.	java.lang.String
serviceportname	service port name (obtained from location if not specified)	java.lang.String
stopclasses	Comma separated list of classes which stop the Java2WSDL inheritance search.	java.lang.String
style	The style of the WSDL document: RPC, DOCUMENT or WRAPPED. If RPC, a rpc/encoded wsdl is generated. If DOCUMENT, a document/literal wsdl is generated. If WRAPPED, a document/literal wsdl is generated using the wrapped approach.	java.lang.String
typemappingversion	the default type mapping registry to use. Either 1.1 or 1.2. Default is 1.1	org.apache.axis.tools.ant.wsdl.TypeMappingVersion
use	Set the use option	java.lang.String
useinheritedmethods	should inherited methods be exported too? Default=false	boolean

1.3. Parameters as nested elements

mapping (org.apache.axis.tools.ant.wsdl.NamespaceMapping)

add a mapping of namespaces to packages

mappingset (org.apache.axis.tools.ant.wsdl.MappingSet)

add a mapping of namespaces to packages