NAME
swig – Simplified Wrapper and Interface Generator

SYNOPSIS
swig [*options*] *file*

DESCRIPTION
The swig command is used to create wrapper code to connect C and C++ code to scripting languages like Perl, Python, etc. from the definition of the interface. For detailed information on writing those interface definitions please refer to /usr/share/doc/swig-doc/Manual/index.html from the swig-doc package. This manpage concentrates on explaining the invocation of the swig command.

OPTIONS
Target Language Options:

–chicken
  Generate CHICKEN wrappers

–csharp
  Generate C# wrappers

–guile Generate Guile wrappers

–java Generate Java wrappers

–mzscheme
  Generate Mzscheme wrappers

–ocaml
  Generate Ocaml wrappers

–perl Generate Perl wrappers.

–php Generate PHP wrappers

–pike Generate Pike wrappers

–python
  Generate Python wrappers

–ruby Generate Ruby wrappers

–sexp Generate Lisp S-Expressions wrappers

–tcl Generate Tcl wrappers

–xml Generate XML wrappers.

General Options

–c++ Enable C++ processing

–co Check a file out of the SWIG library

–dirprot
  Turn on wrapping of protected members for director classes

–D_symbol_
  Define a symbol *symbol* (for conditional compilation)

–E Preprocess only, does not generate wrapper code

–fcompact
  Compile in compact mode

−fvirtual
> Compile in virtual elimination mode

−Fstandard
> Display error/warning messages in commonly used format

−Fmicrosoft
> Display error/warning messages in Microsoft format

−help   Show a summary of supported options and especially of supported options for the selected language.

−I*dir*   Look for SWIG files in *dir*

−ignoremissing
> Ignore missing include files

−importall
> Follow all #include statements as imports

−includeall
> Follow all #include statements

−l*ifile*   Include SWIG library file *ifile*

−M   List all dependencies

−MM   List dependencies, but omit files in SWIG library

−makedefault
> Create default constructors/destructors (the default)

−module *name*
> Set module name to *name*

−nocontract
> Turn off contract checking

−nodefault
> Do not generate constructors/destructors

−nodirprot
> Do not wrap director protected members

−noexcept
> Do not wrap exception specifiers

−noextern
> Do not generate extern declarations

−noruntime
> Do not include SWIG runtime code

−o *outfile*
> Set name of the output file to *outfile*

−outdir *dir*
> Set language specific files output directory

−runtime
> Make the runtime support code globally visible.

−small   Compile in virtual elimination & compact mode

−swiglib
> Report location of SWIG library and exit

−v   Run in verbose mode

–version
> Print SWIG version number

–Wall   Enable all warning messages

–Wallkw
> Enable warning messages for all the languages keywords

–Werror
> Force to treat the warnings as errors

–w *n*   Suppress warning number *n*

AUTHOR

SWIG was originally created by David Beazley. For up-to-date information about authors and contributors please check http://www.swig.org/guilty.html. This manual page was written by Torsten Landschoff <torsten@debian.org> for the Debian project (but may be used by others).