Linker Options

Visual Studio 2015

The new home for Visual Studio documentation is Visual Studio 2017 Documentation on docs.microsoft.com.

The latest version of this topic can be found at Linker Options.

LINK.exe links Common Object File Format (COFF) object files and libraries to create an executable (.exe) file or a dynamic-link library (DLL).

The following table lists options for LINK.exe. For more information about LINK, see:

- Compiler-Controlled LINK Options
- LINK Input Files
- LINK Output
- Reserved Words

On the command line, linker options are not case-sensitive—for example, /base and /BASE mean the same thing. For details on how to specify each option on the command line or in Visual Studio, see the documentation for that option.

You can use the comment pragma to specify some linker options.

Option	Purpose
@	Specifies a response file.
/ALIGN	Specifies the alignment of each section.
/ALLOWBIND	Specifies that a DLL cannot be bound.
/ALLOWISOLATION	Specifies behavior for manifest lookup.
/APPCONTAINER	Specifies whether the app must run within an appcontainer process environment.
/ASSEMBLYDEBUG	Adds the DebuggableAttribute to a managed image.
/ASSEMBLYLINKRESOURCE	Creates a link to a managed resource.
/ASSEMBLYMODULE	Specifies that a Microsoft intermediate language (MSIL) module should be imported into the assembly.

Option	Purpose
/ASSEMBLYRESOURCE	Embeds a managed resource file in an assembly.
/BASE	Sets a base address for the program.
/CGTHREADS	Sets number of cl.exe threads to use for optimization and code generation when link-time code generation is specified.
/CLRIMAGETYPE	Sets the type (IJW, pure, or safe) of a CLR image.
/CLRSUPPORTLASTERROR	Preserves the last error code of functions that are called through the P/Invoke mechanism.
/CLRTHREADATTRIBUTE	Specifies the threading attribute to apply to the entry point of your CLR program.
/CLRUNMANAGEDCODECHECK	Specifies whether the linker will apply the SuppressUnmanagedCodeSecurity attribute to linker-generated PInvoke stubs that call from managed code into native DLLs.
/DEBUG	Creates debugging information.
/DEBUGTYPE	Specifies which data to include in debugging information.
/DEF	Passes a module-definition (.def) file to the linker.
/DEFAULTLIB	Searches the specified library when external references are resolved.
/DELAY	Controls the delayed loading of DLLs.
/DELAYLOAD	Causes the delayed loading of the specified DLL.
/DELAYSIGN	Partially signs an assembly.
/DLL	Builds a DLL.
/DRIVER	Creates a kernel mode driver.
/DYNAMICBASE	Specifies whether to generate an executable image that can be randomly rebased at load time by using the address space layout randomization (ASLR) feature.
/ENTRY	Sets the starting address.
/errorReport	Reports internal linker errors to Microsoft.
/EXPORT	Exports a function.
/FIXED	Creates a program that can be loaded only at its preferred base address.
/FORCE	Forces a link to complete even with unresolved symbols or symbols defined more than once.

Option	Purpose
/FUNCTIONPADMIN	Creates an image that can be hot patched.
/GENPROFILE, /FASTGENPROFILE	Both of these options specify generation of a .pgd file by the linker to support profile-guided optimization (PGO). /GENPROFILE and /FASTGENPROFILE use different default parameters.
/GUARD	Enables Control Flow Guard protection.
/HEAP	Sets the size of the heap, in bytes.
/HIGHENTROPYVA	Specifies support for high-entropy 64-bit address space layout randomization (ASLR).
/IDLOUT	Specifies the name of the .idl file and other MIDL output files.
/IGNORE	Suppresses output of specified linker warnings.
/IGNOREIDL	Prevents the processing of attribute information into an .idl file.
/IMPLIB	Overrides the default import library name.
/INCLUDE	Forces symbol references.
/INCREMENTAL	Controls incremental linking.
/INTEGRITYCHECK	Specifies that the module requires a signature check at load time.
/KEYCONTAINER	Specifies a key container to sign an assembly.
/KEYFILE	Specifies a key or key pair to sign an assembly.
/LARGEADDRESSAWARE	Tells the compiler that the application supports addresses larger than two gigabytes
/LIBPATH	Specifies a path to search before the environmental library path.
/LTCG	Specifies link-time code generation.
/MACHINE	Specifies the target platform.
/MANIFEST	Creates a side-by-side manifest file and optionally embeds it in the binary.
/MANIFESTDEPENDENCY	Specifies a <dependentAssembly> section in the manifest file.
/MANIFESTFILE	Changes the default name of the manifest file.

Option	Purpose
/MANIFESTINPUT	Specifies a manifest input file for the linker to process and embed in the binary. You can use this option multiple times to specify more than one manifest input file.
/MANIFESTUAC	Specifies whether User Account Control (UAC) information is embedded in the program manifest.
/MAP	Creates a mapfile.
/MAPINFO	Includes the specified information in the mapfile.
/MERGE	Combines sections.
/MIDL	Specifies MIDL command-line options.
/NOASSEMBLY	Suppresses the creation of a .NET Framework assembly.
/NODEFAULTLIB	Ignores all (or the specified) default libraries when external references are resolved.
/NOENTRY	Creates a resource-only DLL.
/NOLOGO	Suppresses the startup banner.
/NXCOMPAT	Marks an executable as verified to be compatible with the Windows Data Execution Prevention feature.
/OPT	Controls LINK optimizations.
/ORDER	Places COMDATs into the image in a predetermined order.
/OUT	Specifies the output file name.
/PDB	Creates a program database (PDB) file.
/PDBALTPATH	Uses an alternate location to save a PDB file.
/PDBSTRIPPED	Creates a program database (PDB) file that has no private symbols.
/PGD	Specifies a .pgd file for profile-guided optimizations.
/PROFILE	Produces an output file that can be used with the Performance Tools profiler.
/RELEASE	Sets the Checksum in the .exe header.
/SAFESEH	Specifies that the image will contain a table of safe exception handlers.
/SECTION	Overrides the attributes of a section.
/STACK	Sets the size of the stack in bytes.
/STUB	Attaches an MS-DOS stub program to a Win32 program.

Option	Purpose
/SUBSYSTEM	Tells the operating system how to run the .exe file.
/SWAPRUN	Tells the operating system to copy the linker output to a swap file before it is run.
/TLBID	Specifies the resource ID of the linker-generated type library.
/TLBOUT	Specifies the name of the .tlb file and other MIDL output files.
/TSAWARE	Creates an application that is designed specifically to run under Terminal Server.
/VERBOSE	Prints linker progress messages.
/VERSION	Assigns a version number.
/WHOLEARCHIVE	Includes every object file from specified static libraries.
/WINMD	Enables generation of a Windows Runtime Metadata file.
/WINMDFILE	Specifies the file name for the Windows Runtime Metadata (winmd) output file that's generated by the /WINMD linker option.
/WINMDKEYFILE	Specifies a key or key pair to sign a Windows Runtime Metadata file.
/WINMDKEYCONTAINER	Specifies a key container to sign a Windows Metadata file.
/WINMDDELAYSIGN	Partially signs a Windows Runtime Metadata (.winmd) file by placing the public key in the winmd file.
/WX	Treats linker warnings as errors.

For more information, see Compiler-Controlled LINK Options.

See Also

C/C++ Building Reference
Setting Linker Options
Frequently Asked Questions on Building