

Data Types in Win32 API

The data types supported by the Microsoft Win32 API are used to define function return values, function and message parameters, and structure members.

They define the size and meaning of these elements.

The following table contains the following types: character, integer, Boolean, pointer, and handle. The character, integer, and Boolean types are common to most C compilers.

Most of the pointer-type names begin with a prefix of P or LP. Handles refer to a resource that has been loaded into memory.

Type	Definition
ATOM	Atom (a reference to a character string in an atom table).
BOOL	Boolean variable (should be TRUE or FALSE).
BOOLEAN	Boolean variable (should be TRUE or FALSE).
BYTE	Byte (8 bits).
CALLBACK	Calling convention for callback functions.
CHAR	8-bit Windows (ANSI) character. For more information, see Character Sets Used By Fonts .
COLORREF	Red, green, blue (RGB) color value (32 bits). See COLORREF for information on this type.
CONST	Variable whose value is to remain constant during execution.
CRITICAL_SECTION	Critical-section object.
CTRYID	Country identifier.
DWORD	32-bit unsigned integer.
DWORD_PTR	Unsigned long type for pointer precision. Use when casting a pointer to a long type to perform pointer arithmetic.
DWORD32	32-bit unsigned integer.
DWORD64	64-bit unsigned integer.

FLOAT	Floating-point variable.
FILE_SEGMENT_ELEMENT	64-bit pointer to a buffer.
HACCEL	Handle to an accelerator table.
HANDLE	Handle to an object.
HBITMAP	Handle to a bitmap.
HBRUSH	Handle to a brush.
HCOLORSPACE	Handle to a logical color space.
HCONV	Handle to a dynamic data exchange (DDE) conversation.
HCONVLIST	Handle to a DDE conversation list.
HCURSOR	Handle to a cursor.
HDC	Handle to a device context (DC).
HDDEDATA	Handle to DDE data.
HDESK	Handle to a desktop.
HDROP	Handle to an internal drop structure.
HDWP	Handle to a deferred window position structure.
HENHMETAFILE	Handle to an enhanced metafile.
HFILE	Handle to a file opened by OpenFile , not CreateFile .
HFONT	Handle to a font.
HGDIOBJ	Handle to a GDI object.
HGLOBAL	Handle to a global memory block.
HHOOK	Handle to a hook.
HICON	Handle to an icon.
HIMAGELIST	Handle to an image list.
HIMC	Handle to input context.

HINSTANCE	Handle to an instance.
HKEY	Handle to a registry key.
HKL	Input locale identifier.
HLOCAL	Handle to a local memory block.
HMENU	Handle to a menu.
HMETAFILE	Handle to a metafile.
HMODULE	Handle to a module.
HMONITOR	Handle to a display monitor.
HPALETTE	Handle to a palette.
HPEN	Handle to a pen.
HRGN	Handle to a region.
HRSRC	Handle to a resource.
HSZ	Handle to a DDE string.
HWINSTA	Handle to a window station.
HWND	Handle to a window.
INT	32-bit signed integer.
INT_PTR	Signed integral type for pointer precision. Use when casting a pointer to an integer to perform pointer arithmetic.
INT32	32-bit signed integer.
INT64	64-bit signed integer.
IPADDR	IP address. To convert this value to the " <i>a.b.c.d</i> " string form of an IP address, map the high-order byte to <i>a</i> , the low-order byte to <i>d</i> , and so on.
IPMASK	Subnet mask. Uses the same format as IPADDR .
LANGID	Language identifier. For more information, see Locales .
LCID	Locale identifier. For more information, see Locales .

LCSCSTYPE	Color space type.
LCSGAMUTMATCH	Gamut-matching method.
LCTYPE	Locale information type. For a list, see Locale and Language Information
LONG	32-bit signed integer.
LONG_PTR	Signed long type for pointer precision. Use when casting a pointer to a long to perform pointer arithmetic.
LONG32	32-bit signed integer.
LONG64	64-bit signed integer.
ONGLONG	64-bit signed integer.
LPARAM	Message parameter.
LPBOOL	Pointer to a BOOL .
LPBYTE	Pointer to a BYTE .
LPCOLORREF	Pointer to a COLORREF value.
LPCRITICAL_SECTION	Pointer to a CRITICAL_SECTION .
LPCSTR	Pointer to a constant null-terminated string of 8-bit Windows (ANSI) characters. For more information, see Character Sets Used By Fonts .
LPCTSTR	An LPCWSTR if UNICODE is defined, an LPCSTR otherwise.
LPCVOID	Pointer to a constant of any type.
LPCWSTR	Pointer to a constant null-terminated string of 16-bit Unicode characters. For more information, see Character Sets Used By Fonts .
LPDWORD	Pointer to a DWORD .
LPHANDLE	Pointer to a HANDLE .
LPINT	Pointer to an INT .
LPLONG	Pointer to a LONG .
LPSTR	Pointer to a null-terminated string of 8-bit Windows (ANSI) characters. For more information, see Character Sets Used By Fonts .

LPTSTR	An LPWSTR if UNICODE is defined, an LPSTR otherwise.
LPVOID	Pointer to any type.
LPWORD	Pointer to a WORD .
LPWSTR	Pointer to a null-terminated string of 16-bit Unicode characters. For more information, see Character Sets Used By Fonts .
LRESULT	Signed result of message processing.
LUID	Locally unique identifier.
PBOOL	Pointer to a BOOL .
PBOOLEAN	Pointer to a BOOL .
PBYTE	Pointer to a BYTE .
PCHAR	Pointer to a CHAR .
PCRITICAL_SECTION	Pointer to a CRITICAL_SECTION .
PCSTR	Pointer to a constant null-terminated string of 8-bit Windows (ANSI) characters. For more information, see Character Sets Used By Fonts .
PCTSTR	PCWSTR if UNICODE is defined, a PCSTR otherwise.
PCWCH	Pointer to a constant WCHAR .
PCWSTR	Pointer to a constant null-terminated string of 16-bit Unicode characters. For more information, see Character Sets Used By Fonts .
PDWORD	Pointer to a DWORD .
PFLOAT	Pointer to a FLOAT .
PHANDLE	Pointer to a HANDLE .
PHKEY	Pointer to an HKEY .
PINT	Pointer to an INT .
PLCID	Pointer to an LCID .
PLONG	Pointer to a LONG .
PLUID	Pointer to a LUID .

POINTER_32	32-bit pointer. On a 32-bit system, this is a native pointer. On a 64-bit system, this is a truncated 64-bit pointer.
POINTER_64	64-bit pointer. On a 64-bit system, this is a native pointer. On a 32-bit system, this is a sign-extended 32-bit pointer.
PSHORT	Pointer to a SHORT .
PSTR	Pointer to a null-terminated string of 8-bit Windows (ANSI) characters. For more information, see Character Sets Used By Fonts .
PTBYTE	Pointer to a TBYTE .
PTCHAR	Pointer to a TCHAR .
PTSTR	PWSTR if UNICODE is defined, a PSTR otherwise.
PUCHAR	Pointer to a UCHAR .
PUINT	Pointer to a UINT .
PULONG	Pointer to a ULONG .
PUSHORT	Pointer to a USHORT .
PVOID	Pointer to any type.
PWCHAR	Pointer to a WCHAR .
PWORD	Pointer to a WORD .
PWSTR	Pointer to a null-terminated string of 16-bit Unicode characters. For more information, see Character Sets Used By Fonts .
REGSAM	Security access mask for registry key.
SC_HANDLE	Handle to a service control manager database.
SC_LOCK	Handle to a service control manager database lock.
SERVICE_STATUS_HANDLE	Handle to a service status value.
SHORT	Short integer (16 bits).
SIZE_T	The maximum number of bytes to which a pointer can point. Use for a count that must span the full range of a pointer.
SSIZE_T	Signed SIZE_T .

TBYTE	WCHAR if UNICODE is defined, a CHAR otherwise.
TCHAR	WCHAR if UNICODE is defined, a CHAR otherwise.
UCHAR	Unsigned CHAR.
UINT	Unsigned INT.
UINT_PTR	Unsigned INT_PTR.
UINT32	Unsigned INT32.
UINT64	Unsigned INT64.
ULONG	Unsigned LONG.
ULONG_PTR	Unsigned LONG_PTR.
ULONG32	Unsigned LONG32.
ULONG64	Unsigned LONG64.
ULONGLONG	64-bit unsigned integer.
UNSIGNED	Unsigned attribute.
USHORT	Unsigned SHORT.
VOID	Any type.
WCHAR	16-bit Unicode character.
WINAPI	Calling convention for the Win32 API.
WORD	16-bit unsigned integer.
WPARAM	Message parameter.

Built on Thursday, October 12, 2000