This is part of **Win16 API** which allow to create versions of program from one source code to run under OS/2 and Win16. Under OS/2 program can be running under Win-OS/2 if program is Windows NE executable, and with help on Windows Libraries for OS/2, if it is OS/2 NE executable. Here is a WLO to OS/2 API mapping draft

2021/09/01 04:23 · prokushev · 0 Comments

# LocalInit

## Brief

Initializes a local heap within a specified memory segment.

## Syntax

```
BOOL LocalInit(
  WORD  wSegment,
  WORD  wOffset,
  WORD  wHeapSize
);
```

## Parameters

- **wSegment** - The selector of the segment in which to initialize the local heap.
- **wOffset** - The offset within the segment at which the heap should start.
- **wHeapSize** - The size, in bytes, of the heap to initialize.

## Return Code

- Returns **non-zero** if the heap is successfully initialized.
- Returns **0** if initialization fails.

## Notes

- `LocalInit`

  is specific to the 16-bit Windows environment and is not supported in Win32 or later.

- The function sets up a local heap manager within the provided segment, enabling local memory allocation functions to operate within that segment.

- Applications typically call

  `LocalInit`

  during initialization of a data segment or a dynamically allocated memory block intended for use as a local heap.

- The heap size (

  `wHeapSize`

  ) must be large enough to accommodate heap management overhead.

- For compatibility and performance reasons, modern Windows applications should use the standard heap functions (e.g.,

  `HeapCreate`

  ,

  `HeapAlloc`

  ) instead.

# Example Code

## C Binding

```
#include <windows.h>

BOOL InitializeLocalHeap(WORD seg, WORD offset, WORD size) {
    return LocalInit(seg, offset, size);
}
```

## MASM Binding

```
; Assume AX = segment, DX = offset, CX = heap size
push cx          ; wHeapSize
push dx          ; wOffset
push ax          ; wSegment
call LocalInit   ; Returns AX = non-zero if success
```

# See also

- LocalAlloc - Allocates memory from the local heap.
- LocalFree - Frees memory allocated from the local heap.
- LocalSize - Retrieves the size of a local memory block.
- LocalReAlloc - Reallocates a local memory block.

| Group | Functions |
|---|---|
| **Module manager** | GETVERSION GETMODULEHANDLE GETMODULEUSAGE GETMODULEFILENAME GETPROCADDRESS MAKEPROCINSTANCE FREEPROCINSTANCE GETINSTANCEDATA CATCH THROW GETCODEHANDLE LOADLIBRARY |
| **Memory Manager** | GlobalAlloc GlobalCompact GlobalDiscard GlobalFree GlobalLock GlobalReAlloc GlobalSize GlobalUnlock GlobalFlags LocalInit LocalAlloc LocalCompact LocalDiscard LocalFree LocalLock LocalFreeze LocalMelt LocalReAlloc LocalSize LocalUnlock LocalHandleDelta LockData UnlockData LocalFlags |
| **Task Scheduler** | GetCurrentTask Yield SetPriority |
| **Resource Manager** | AddFontResource RemoveFontResource LoadBitmap LoadCursor LoadIcon LoadMenu LoadString LoadAccelerators FindResource LoadResource AllocResource LockResource FreeResource AccessResource SizeofResource SetResourceHandler |
| **String Translation** | AnsiUpper AnsiLower AnsiNext AnsiPrev |
| **Atom Manager** | InitAtomTable AddAtom DeleteAtom FindAtom GetAtomName |
| **Windows Initialization File** | GetProfileInt GetProfileString WriteProfileString |
| **Debugging** | FatalExit |
| **File I/O** | _lopen _lcreat _llseek _lread _lwrite _lclose OpenFile GetTempFileName GetTempDrive |
| **Registry** | RegOpenKey RegCreateKey RegCloseKey RegDeleteKey RegSetValue RegQueryValue RegEnumKey |

2022/11/17 15:22 · prokushev · 0 Comments