

```

WIN.H
/*****
/*                               W I N . H                               */
/**-----**/
/* task       : Include file for Window routines                        */
/**-----**/
/* author      : Michael Tischer / Bruno Jennrich                      */
/* developed on : 5/3/1994                                              */
/* last update  : 4/06/1995                                            */
/**-----**/
/* COMPILER    : Borland C++ 3.1, Microsoft Visual C++ 1.5          */
/*****
#ifndef _INC_WIN_H
#define _INC_WIN_H

#include "types.h"

#define VM_COLOR 1
#define VM_MONO 2

#define VIOINT      0x10
#define VIOPAGE     0x00
#define VIOSETCURSOR 0x02
#define VIOGETCURSOR 0x03
    .1 .4 cm 118 Raster
#define VIOSETPAGE   0x05
#define VIOGETMODE   0x0F

#define WIN_CRLF      0x0001
#define WIN_SCROLL    0x0002
#define WIN_HASCURSOR 0x0004
#define WIN_ACTIVE    0x0008

```

```

typedef struct tagWINDOW
{
    INT    iX, iY;           /* position to the higher Window/screen */
    INT    iW, iH;           /* width and height */
    INT    iCX, iCY;        /* current (actual) print out position */
}
/*
    BYTE    iAttr;           /* actual attribute */
    BYTE    iHiAttr;        /* active attribute */
    BYTE    iLoAttr;        /* passive attribute */
    UINT    uFlags;         /* s. WIN_??? */
} WINDOW;
typedef WINDOW *PWINDOW;

#define OT_INT 1             /* OBJECTTYPE - INT */
#define OT_BOOL 2

typedef struct tagINTDATA
{
    INT    iMin, iMax;
    PINT    pValue;
    PCHAR    pText;
} INTDATA;
typedef INTDATA *PINTDATA;

#define DT_TRUEFALSE 0
#define DT_ONOFF 1
#define DT_YESNO 2

typedef struct tagBOOLDATA
{
    INT    iDisplay;
    PINT    pValue;
    PCHAR    pText;
}

```

```

} BOOLDATA;
typedef BOOLDATA *PBOOLDATA;

#define MSG_LOSTFOCUS 0
#define MSG_GOTFOCUS 1
#define MSG_KEY 2
#define MSG_CHANGED 3
#define MSG_PRECHANGE 4
typedef struct tagOBJECT
{
    INT    X, Y, W, H;
    INT    iType;
    PVOID  pData;
} OBJECT;
typedef OBJECT *POBJECT;

```

```
/* prototypes */
```

```

VOID _FP win_Init( PWINDOW pWin, INT iX, INT iY,
                  INT iW, INT iH, BYTE iHA, BYTE LA, UINT uFlags );
VOID _FP win_GetVIOSMEM( VOID );
WORD _FP win_GetMode( VOID );
VOID _FP win_GotoXY( PWINDOW pWin, INT iX, INT iY );
VOID _FP win_Clr( PWINDOW pWin );
VOID _FP win_Frame( PWINDOW pWin, PCHAR pFrame );
PBYTE _FP win_Save( PWINDOW pWin );
VOID _FP win_Restore( PBYTE pMem, BYTE bFree );
VOID _FP win_GetScreenSettings( PWINDOW pWin );
VOID _FP win_ScrollUp( PWINDOW pWin, INT NumCols );
VOID _FP _win_Print( PWINDOW pWin, PCHAR pText, INT iCnt );
VOID _FP win_Print( PWINDOW pWin, PCHAR pText );
VOID _FP win_PrintAt( PWINDOW pWin, INT iX, INT iY, PCHAR pText );

```

```
VOID _FP win_printf( PWINDOW pWin, PCHAR pFormat, ... );
VOID _FP win_Beep( VOID );
VOID _FP win_HiVideo( PWINDOW pWin );
VOID _FP win_LoVideo( PWINDOW pWin );
                                /* Oh, how I love OOP! */
VOID _FP win_InitINTObject( POBJECT pObject, PINTDATA pData,
                           INT x, INT y, INT w, INT h,
                           PCHAR pText, INT iMin, INT iMax, PINT pValue );

#endif
```