2014.7.15

1. 在WndProc里获取hInstance

将Lparam强制转换成Createstruct

hInstance = ((LPCREATESTRUCT)lParam)->hInstance;

1. HBITMAP LoadBitmap(

HINSTANCE hInstance, // handle to application instance

LPCTSTR lpBitmapName // name of bitmap resource

);

返回一个位图的句柄

如果资源在资源视图的，则用makeintresource(idb\_id);

hBitmap = LoadBitmap(hInstance, MAKEINTRESOURCE(IDB\_block));

1. int GetObject(

HGDIOBJ hgdiobj, // handle to graphics object

int cbBuffer, // size of buffer for object information

LPVOID lpvObject // buffer for object information

);

获得指定图像信息，根据图像，把信息放入缓冲区

GetObject(hBitmap, sizeof(BITMAP), &bitmap);

1. hdcMem = CreateCompatibleDC(hdc); //创建兼容DC

hdc：现有设备上下文环境的句柄，如果该句柄为NULL，该函数创建一个与应用程序的当前显示器兼容的内存设备上下文环境。

返回值：如果成功，则返回内存设备上下文环境的句柄；如果失败，则返回值为NULL。

1. HGDIOBJ SelectObject(

HDC hdc, // handle to DC

HGDIOBJ hgdiobj // handle to object

);

选择一个对象替换原对象到DC中

比如：bitmap,brush,font,pen,region

返回值为原对象

SelectObject(hdcMem, hBitmap);

1. BOOL BitBlt(

HDC hdcDest, // handle to destination DC

int nXDest, // x-coord of destination upper-left corner

int nYDest, // y-coord of destination upper-left corner

int nWidth, // width of destination rectangle

int nHeight, // height of destination rectangle

HDC hdcSrc, // handle to source DC

int nXSrc, // x-coordinate of source upper-left corner

int nYSrc, // y-coordinate of source upper-left corner

DWORD dwRop // raster operation code

);

*dwRop*

[in] Specifies a raster-operation code. These codes define how the color data for the source rectangle is to be combined with the color data for the destination rectangle to achieve the final color.

从源DC画到目标DC

BitBlt(hdc, x, y, cxSource, cySource / 2, hdcMem, 0, cySource / 2, SRCCOPY);

1. BOOL DeleteObject(

HGDIOBJ hObject // handle to graphic object

);

The **DeleteObject** function deletes a logical pen, brush, font, bitmap, region, or palette, freeing all system resources associated with the object. After the object is deleted, the specified handle is no longer valid.

1. RECT rc = { 0, 0, 48, 48 };

POINT pt;

GetCursorPos(&pt); //获取坐标点

ScreenToClient(hWnd, &pt); //转换成客户区坐标

char buf[30];

sprintf\_s(buf, "%d %d", pt.x, pt.y); //格式化输出字符串

InvalidateRect(hWnd, &rc, TRUE); //发送部分窗体重绘消息，true则背景重绘