

オブジェクト指向と ゲームプログラミング

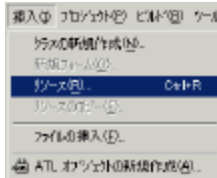
基礎編 - 第22回 アイコンの作成

アイコンの作成と登録

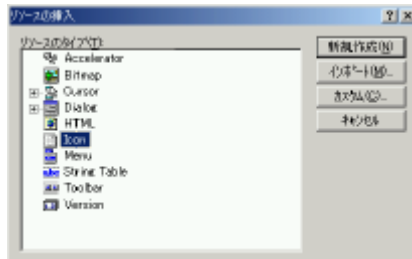
(1) アイコンの作成

アイコンはアプリケーション内の「リソース」という場所に保存されます。リソースとは、アプリケーションが使用するアイコン、カーソル、ビットマップといった絵や音などの資源(リソース)をまとめて管理、保存しておく場所です。リソースに登録したものは、アプリケーションに内蔵され、ファイルがなくても読み込むことができます。

アプリケーション内蔵のリソースを作成するには、メニューから「挿入(I) リソース(R)」を選択します。



「リソースの挿入」ダイアログが表示されるので、「Icon」をクリックし、「新規作成(N)」を選択します。



アイコンのエディタ画面になります。ここで  をクリックし、「32 x 32, 256色」を選択します。



アイコンを作成します。ペイントやPhotoshopなどのグラフィックエディタで作成したものを貼り付けることもできます。また、フリーソフトやシェアウェアにはアイコン専用のエディタがあります。

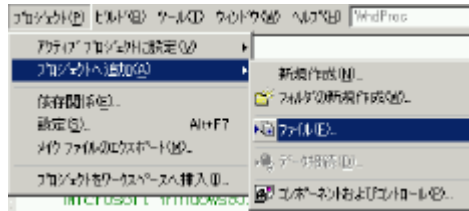


完成したら保存します。メニューから「ファイル(F) 名前を付けて保存(A)」を選択して"Script.rc"のように、拡張子が".rc"となるように名前を付けて保存します。



(2) リソースの登録

次にリソースをプロジェクトに登録します。メニューから「プロジェクト(P) プロジェクトへ追加(A) ファイル(F)」を選択します。



「プロジェクトへファイルを追加」ダイアログが表示されるので、(1)で保存したリソーススクリプトファイル(Script.rc)を選択します。



これで(1)で作成したリソースがプロジェクトに登録されます。この状態でビルドすると、実行ファイルのアイコンが変更されるので、確認しましょう。

(3) システムメニューアイコンの変更

ウィンドウ左上に表示されるシステムメニューのアイコンは、ウィンドウクラスを定義するとき使用するWNDCLASSEX構造体のhIconメンバで指定します。このメンバに、LoadIcon関数で表示したいアイコンを読み込み、そのハンドルを指定します。

LoadIcon関数の1つ目の引数は、アプリケーションのインスタンスハンドルです。システム標準のアイコンを使用する場合はNULL、オリジナルアイコンの場合は、WinMain関数がプログラム起動時に受け取るインスタンスハンドル(hInstance)を渡します。

2つ目の引数は、読み込むアイコンの識別子です。識別子には文字列と定数の2種類あります。(1)で作成したアイコンは、プロパティを変更していないなら、デフォルトで「IDI_ICON1」という定数が割り当てられます。このような定数はMAKEINTRESOURCEマクロで正式な識別子を作成しないと使用できません。なお、文字列の識別子そのまま使用できます。

アイコンの識別子が文字列の場合：LoadIcon(hInstance, "ICON1")

アイコンの識別子が定数の場合：LoadIcon(hInstance, MAKEINTRESOURCE(IDI_ICON1))

IDI_ICON1などの識別子は、リソース作成時に自動的に生成されるヘッダ"resource.h"で定義されています。識別子を使用する場合は、このヘッダをインクルードしないとエラーになります。このヘッダはプライベートヘッダとして扱われるので、" (ダブルクォーテーション)で囲んでインクルードします。

```
#include "resource.h"
```

そのほかのリソース

カーソルもアイコンと同様に作成することができます。アイコンやカーソルは複数登録しておき、状況に合わせて変更することもできます(すでにリソースが存在する場合は、前述(1),(3)のみの手順で作成できます。この場合、(1)の最後の「リソースに名前を付けて保存」は必要ありません)。

また、バージョン情報を作成しておく、実行ファイルを右クリックしたときに、バージョンや会社名などの情報を表示することができます。

練習問題

オリジナルのアイコンを作成し、実行ファイルとシステムメニューのアイコンを変更しましょう。また、カーソルとバージョン情報も作成してみましょう。