

オブジェクト指向と ゲームプログラミング

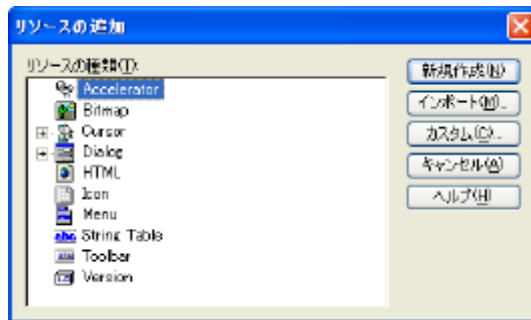
フレームワーク編 - 第7回 リソースの作成

課題

アイコンとバージョン情報を作成しましょう。

アイコンやカーソル、バージョン情報は、実行ファイルのリソースという特別な場所に保存することができます。リソースに保存しておけば、.icoや.curのような外部ファイルがなくても、いつでも読み込むことができます。

(1)メニューから「プロジェクト(P) リソースの追加(R)」を選択し、リソースの追加ダイアログを開きましょう。



(2)アイコンを作成します。「リソースの種類(T)」の"Icon"を選択した状態で、新規作成する場合は、「新規作成(N)」ボタンを、すでに作成済みで読み込む場合は「インポート(M)」ボタンをクリックしましょう。

(3)新規作成の場合は、アイコンのエディタが起動します。インポートする場合は、ファイル選択画面になるので、読み込むファイルを選択します。

アイコンエディタでは、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高機能なアイコン専用のエディタがあります。

(4)アイコンが完成したら保存します。プロジェクトにリソーススクリプトファイル(プロジェクト名.rc)が追加されているか確認しましょう。

(5)この状態でプロジェクトをビルドすると、実行ファイルが登録したアイコンに変更されるので、確認しましょう。

(6)システムメニューのアイコンを変更します。

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、インスタンスのハンドルです。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、インスタンスのハンドルを渡します。

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。(3)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。プログラムでリソースの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、以下のように""(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"
```

(7)バージョン情報を作成しましょう。

バージョン情報を作成しておくことで、実行ファイルのプロパティを表示したときに、バージョン番号や会社名などの情報が表示されます。