

オブジェクト指向と ゲームプログラミング

DirectX Audio編 - 第1回 DirectX Audio

DirectX Audio

DirectX Audioは、MIDIデータとWaveデータの管理や制御を行うインタフェースを提供するDirectMusicと、サウンドカードの管理や制御を行うインタフェースを提供するDirectSoundの総称で、主に以下の機能を提供します。

- MIDIデータの再生
- Waveデータの再生と録音
- 複数のサウンドの同時再生
- ファイルやリソースからのオーディオファイルの読み込み
- ピッチの変更やリバーブなどのエフェクトを加えた再生
- Waveデータが格納されているメモリ領域へ直接的なアクセス
- そのほか、オーディオスクリプトによる再生制御、DLSによる環境に依存しない再生、3Dサウンド、インタラクティブ再生などを提供

DirectX Audioを使ったソースファイルをビルドする場合は、ヘッダファイル<dmusic.h>とGUIDが格納された静的リンクライブラリ"dxguid.lib"が必要になります。

課 題

DirectX Audio(バージョン8)をカプセル化するクラスCDXAUDIO8を作成しましょう。

(1) CDXAUDIO8クラスの定義を行います。クラスのヘッダファイル(DXAUDIO8.hpp)を以下のように作成しましょう。

- DXAUDIO8.hpp -

```
/*
=====
                          オブジェクト指向ゲームプログラミング
    Programmed by Hibikino software. Copyright (c) 2005 Hibikino software. All rights reserved.
=====
【対象OS】
    Microsoft Windows2000/XP

【コンパイラ】
    Microsoft Visual C++ 2005

【プログラム】
    DXAUDIO8.hpp
    DirectX Audioクラスヘッダ

【履歴】
    * Version    1.00    2005/03/dd  hh:mm:ss
=====
*/

#pragma once

/*****
/*
                          インクルードファイル
*****/

/*****
/*
                          DirectX Audio8クラス定義
*****/
class CDXAUDIO8 {
public:
```

```
private:  
};
```

(2) 「インクルードファイル」にDirectX Audioで必要となるファイルを追加しましょう。

(3) CDXAUDIO8クラスのソースファイル(DXAUDIO8.cpp)を以下のように作成しましょう。

- DXAUDIO8.cpp -

```
/*  
===== オブジェクト指向ゲームプログラミング =====  
Programmed by Hibikino software. Copyright (c) 2005 Hibikino software. All rights reserved.  
=====
```

【対象OS】
Microsoft Windows2000/XP

【コンパイラ】
Microsoft Visual C++ 2005

【プログラム】
DXAUDIO8.hpp
DirectX Audioクラスヘッダ

【履歴】
* Version 1.00 2005/03/dd hh:mm:ss

```
=====
```

*/

```
/*  
***** インクルードファイル *****  
*/
```

```
/*  
***** 静的リンクライブラリ *****  
*/
```

(4) 「インクルードファイル」に、このソースファイルをコンパイルするのに必要となるファイルを追加しましょう。

(5) 「静的リンクライブラリ」に、このソースファイルをリンクするときに必要なファイルを追加しましょう。

(6) 以下のプログラムをprivateに追加しましょう。これにより、オブジェクトのコピーができなくなります。

```
// コピーできないようにする  
CDXAUDIO8(const CDXAUDIO8&); // コピーコンストラクタ  
CDXAUDIO8& operator=(const CDXAUDIO8&); // 代入演算子
```

コピーコンストラクタと代入演算子の関数本体が定義されていないので、CDXAUDIO8クラスのオブジェクトを=演算子などでコピーしようとすると、コンパイル時にエラーになります。

(7) コンストラクタとデストラクタを作成します。以下のプロトタイプをpublicに追加しましょう。

```
CDXAUDIO8(); // コンストラクタ  
~CDXAUDIO8(); // デストラクタ
```

(8)コンストラクタとデストラクタを実装します。以下のプログラムをソースファイルに追加しましょう。

```
/*
 *
 *          コンストラクタ
 *
 */
CDXAudio8::CDXAudio8()
{
}

/*
 *
 *          デストラクタ
 *
 */
CDXAudio8::~CDXAudio8()
{
}
```

(9)CDXAudio8クラスが継承されることも考えられるので、デストラクタを仮想デストラクタに変更しましょう。

(10)CDXAudio8クラスをシングルトン化しましょう。

CDXAudio8クラスのインスタンスは、1つのアプリケーションにつき1つあればよく、複数のインスタンスが存在すると資源の奪い合いなどで不具合が起こればと考えられます。以下の手順に従ってシングルトン化し、インスタンスの生成を制限しましょう。

コンストラクタをpublicからprivate(継承する場合はprotected)に変更します。

唯一のインスタンスを取得するGetInstance関数を以下のように作成し、publicに追加します。

```
// 唯一のインスタンスの取得
static CDXAudio8& GetInstance()
{
    static CDXAudio8 theDXAudio;
    return theDXAudio;
}
```

短い名前でもインスタンスを取得できるように、以下のインライン関数をヘッダファイルに追加します。

```
/*
 *
 *          インライン関数
 *
 */
inline CDXAudio8& DXAudio() { return CDXAudio8::GetInstance(); }
```

インライン関数DXAudio()を呼び出すと、CDXAudio8クラスの外部でも、各メンバにアクセスすることができます。

(11)DirectXをカプセル化したクラスのヘッダファイルをまとめてインクルードする"DirectX.h"を以下のように作成しましょう。

- DirectX.h -

```
/*
=====
                        オブジェクト指向ゲームプログラミング
=====
Programmed by Hibikino software. Copyright (c) 2005 Hibikino software. All rights reserved.
=====
【対象OS】
Microsoft Windows2000/XP

【コンパイラ】
Microsoft Visual C++ 2005

【プログラム】
DirectX.h
DirectXヘッダ
*/
```

【履歴】

* Version 1.00 2005/03/dd hh:mm:ss

*/

#pragma once

/*
/*
/* インクルードファイル */
/* */

#include "DXAudio8.hpp"