

Visual Studio のメニュー「ファイル | 新規作成 | プロジェクト」を選び、プロジェクトの種類として CLR、テンプレートとして CLR コンソールアプリケーションを選ぶと、次のプログラムのひな型が表示される。

リスト 1 表示されたプログラムのひな型。

```
// sample00.cpp : メインプロジェクトファイルです。
#include "stdafx.h"
using namespace System;
int main(array<System::String ^> ^args)
{
    Console::WriteLine(L"Hello World");
    return 0;
}
```

このプログラムを F5 キーを押してビルド・実行すると、文字列「Hello World」のコンソール画面への出力後、すぐに次の「return 0」が実行されてプログラムは終了し、コンソール画面は消える。

リスト 2 Console::ReadLine()による Enter キー待ち。

```
// sample00a.cpp : メインプロジェクトファイルです。
#include "stdafx.h"
using namespace System;
int main(array<System::String ^> ^args)
{
    Console::WriteLine(L"Hello World");
    Console::ReadLine();
    return 0;
}
```

リスト 2 のプログラムは、return 文の前に「Console::ReadLine()」を入れたものである。この ReadLine() は Enter キーが押されるまで待つものであるため、コンソール画面は Enter キーが押されるまで表示され続ける。Enter キーを押すと ReadLine() の実行終了となり、次の「return 0;」が実行されて、プログラムは終了する。

リスト 1 あるいはリスト 2 で用いられている Console クラスは Visual C++/CLR における標準入出力用のものであるが、C++ の標準入出力ストリーム cout や cin とともに用いることもできる。リスト 3 は cout と Console の両方を用いたものである。cout を用いるときはヘッダーファイル iostream を #include することと、cin や cout の namespace が std で

あることに注意する。

リスト 3 cout と Console の使用。

```
// sample00b.cpp : メインプロジェクトファイルです。

#include "stdafx.h"
#include <iostream>

using namespace System;
using namespace std;

int main(array<System::String ^> ^args)
{
    cout << "こんにちは!" << endl;
    cout << "Enterキーを押して終了。" << endl;
    Console::ReadLine();
    return 0;
}
```

リスト 3 を実行すると図 1 のコンソール画面が表示される。

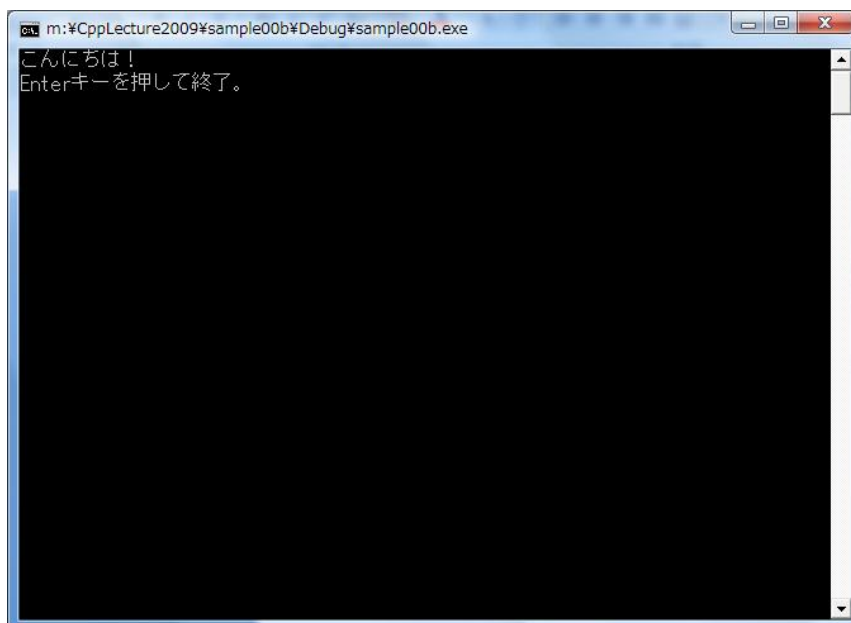


図 1 リスト 3 の実行。

この場合も、Enter キーを押すと「Console::ReadLine();」の実行終了となり、次の return 文が実行されてプログラムが終了する。