

スクール COBOL2002

操作入門書

－(株)日立製作所－

(2008年3月作成)

目 次

1. COBOL2002起動から実行まで	-----	3
2. テストデバッガの使用方法	-----	36
3. 関連資料	-----	69
(a) ファイルの入出力処理	-----	70
(b) テストデータの作成方法	-----	80
(c) 用紙の節約方法	-----	90
(d) 印刷書式の設定方法	-----	96
(e) エディタ設定方法	---	101
(f) コンパイルリストの入手方法	---	114
(g) オンラインマニュアルの使用方法	---	124
(h) 登録集原文の指定方法	---	132
(i) サブプログラムの追加方法	---	140
(j) 索引ファイルを新規に作成する方法	---	144
(k) 既存のプロジェクトマスタファイルの開き方		147

1. COBOL2002起動から実行まで

－ 初めてCOBOL2002を使う方のために －

－ 目次 －

1. はじめに
2. COBOL2002の起動
3. プロジェクトマスタファイルの作成
4. プロジェクトの作成
5. COBOLソースプログラムの編集
6. コンパイル(実行用ファイルの生成)
7. 実行
8. プロジェクトの追加
9. 終わりに

1. はじめに

これから、COBOL2002の使い方を順を追って説明していきます。

スクールCOBOL2002には次の2種類の製品があります。

- ・ Windows Vista® 対応版スクールCOBOL2002
- ・ Windows XP/2000 対応版スクールCOBOL2002

どちらの製品をお使いになっても操作手順は同じです。COBOL2002のメニュー画面の形式も変わりません。スタートボタンやウインドウのデザインが異なる等の違いはありますが、手順そのものは同じです。

本資料では、スタートボタンからCOBOL2002を起動するまでの流れだけ、Windows Vista® 対応版とWindows XP/2000対応版に分けて説明します。起動後の操作につきましては、Windows XP/2000の画面表示例を使って説明します。

なお、Windows Vista® 対応版には次の制約がありますので、ご注意ください。

《Windows Vista® 対応版の制約事項》

- ・ プログラム名として日本語を使用することはできません。
(プログラム名段落のプログラム名, CALL文で呼び出すプログラム名等)

操作手順の説明に入る前に、まず次の注意事項をお読みください。

《注意事項》

COBOLのソースプログラムや実行形式ファイルは特定のフォルダの下に作ります。後で削除等しやすいように、練習用のフォルダを作成してください。本説明書では、Cドライブの下にtempという名称のフォルダを作成し、その下にsample01というフォルダを作成するものとして、説明を進めて行きます。

2. COBOL2002の起動

COBOL2002を起動するには、スタートボタンから行う方法と、COBOL2002の各種ツールをアイコン化しておいて、そのアイコンを(ダブル)クリックする方法があります。ここでは、スタートボタンから行う方法を説明します。

(1) Windows Vista® 対応版スクールCOBOL2002をお使いの場合

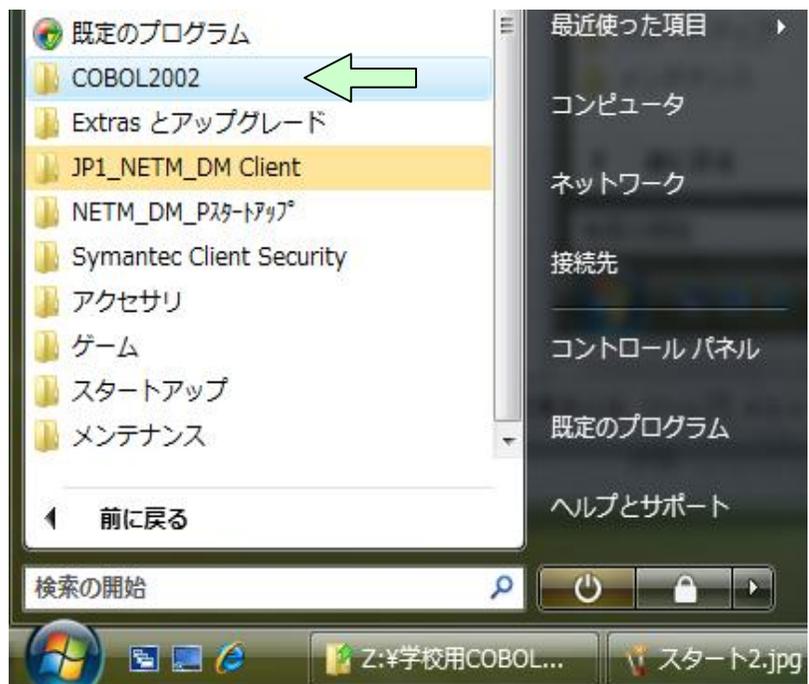
[手順 1]

スタートボタンを押し、「すべてのプログラム」の所にマウスポインタを移動します。
すると起動できるプログラムの一覧が出てきます。

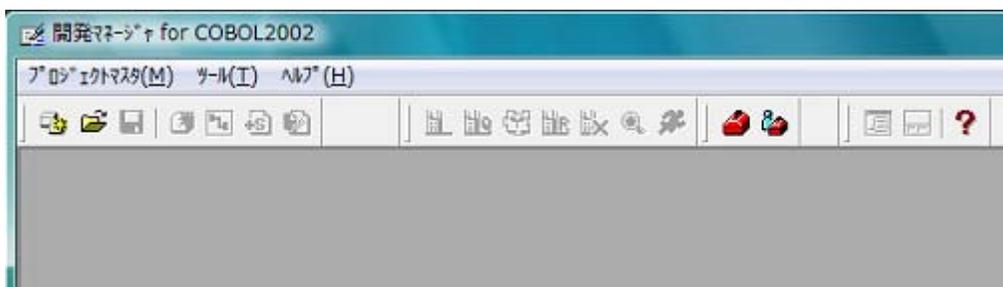
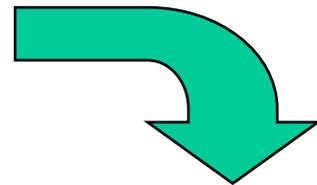


[手順 2]

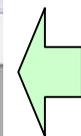
プログラムの一覧の中からCOBOL2002の所にマウスポインタを移動します。
するとCOBOL2002の使用できるツール一覧が表示されます。



[手順3] COBOL2002の使用できるツールの中から、COBOL2002開発マネージャを選んでクリックします。すると、COBOL2002開発マネージャが起動されます。



すぐにこの画面が表示されます。

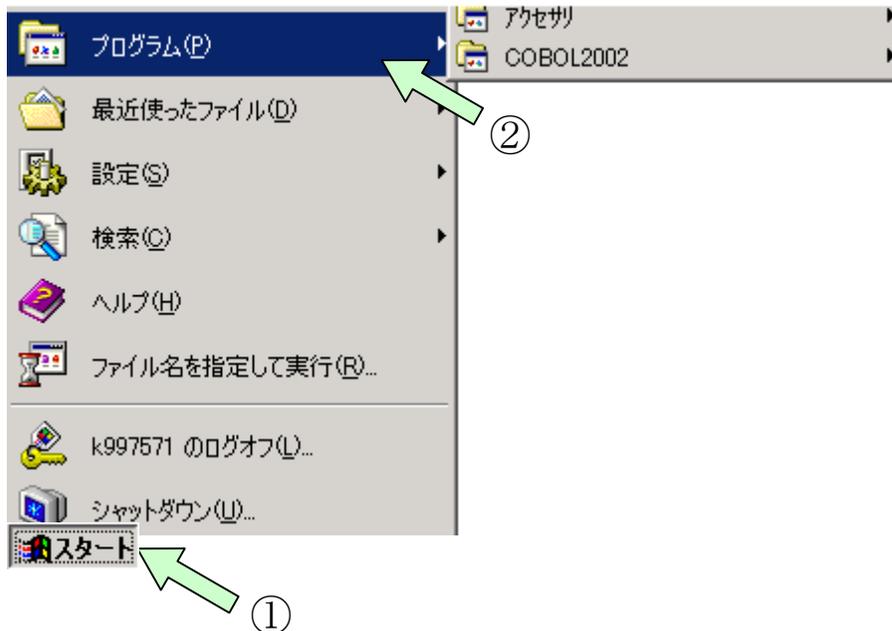


[用語解説] COBOL2002開発マネージャ

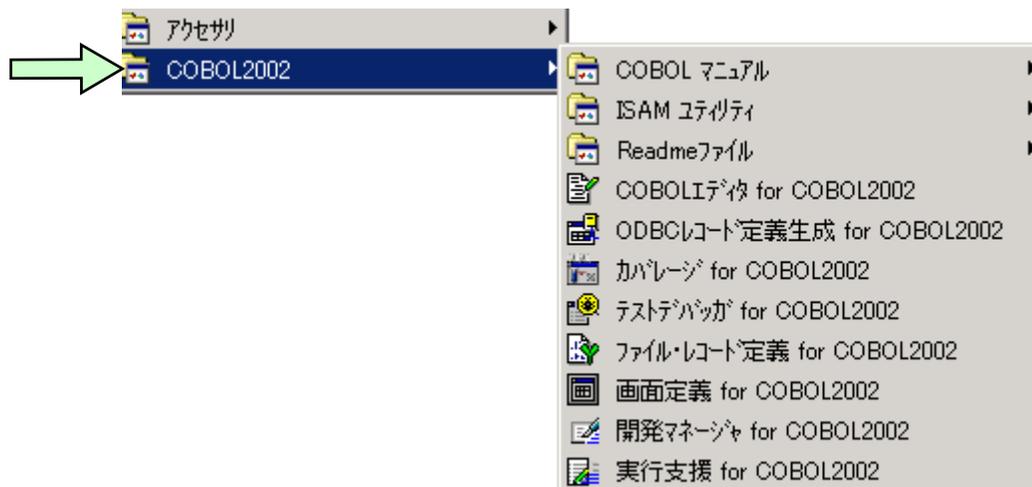
日立COBOL2002を用いてコーディングからコンパイル、テスト、実行等の操作をコントロールするツールです。基本操作ではこのツールを使用します。

(2) Windows XP/2000 対応版スクールCOBOL2002をお使いの場合

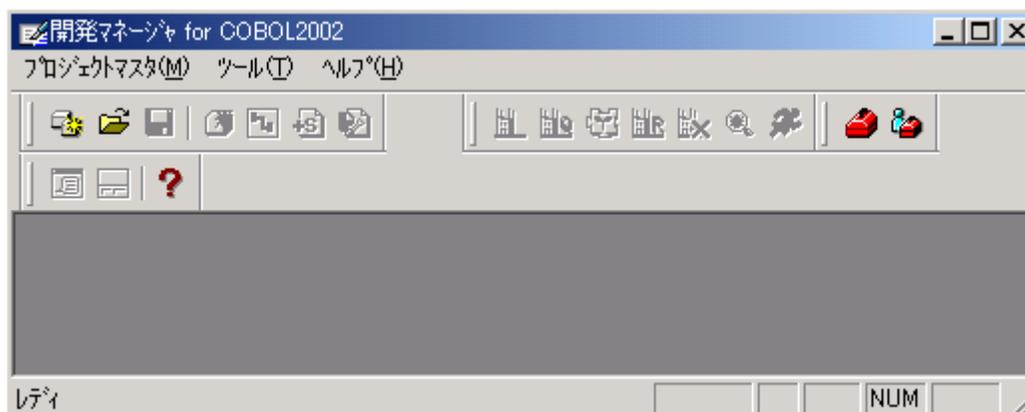
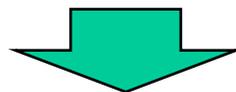
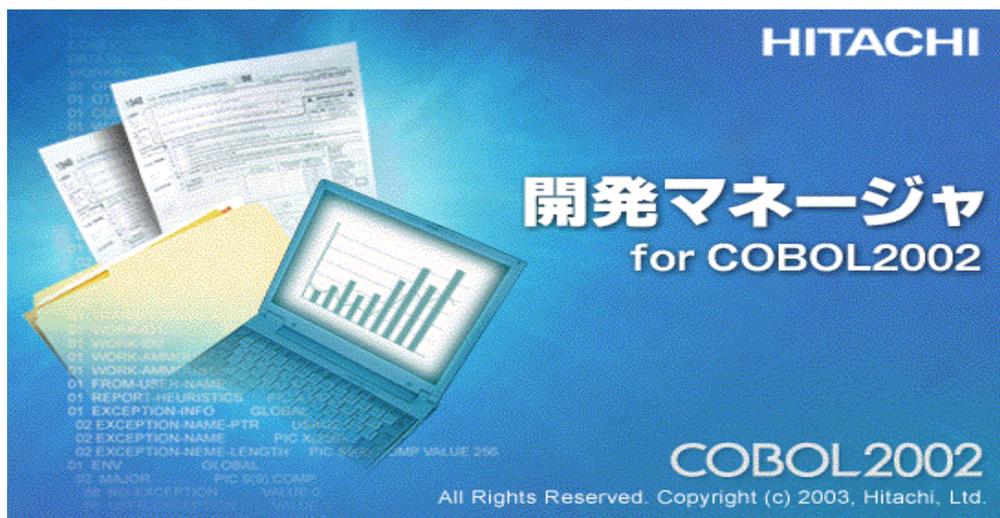
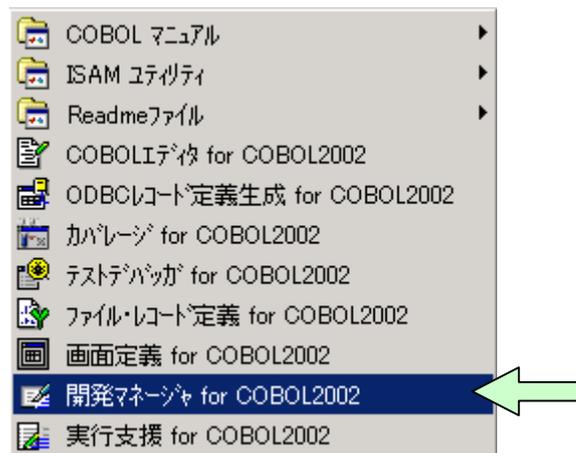
[手順 1] スタートボタンを押し、「プログラム(P)」の所にマウスポインタを移動します。すると起動できるプログラムの一覧が出てきます。



[手順 2] プログラムの一覧の中からCOBOL2002の所にマウスポインタを移動します。するとCOBOL2002の使用できるツール一覧が表示されます。



[手順3] COBOL2002の使用できるツールの中から、COBOL2002開発マネージャを選んでクリックします。すると、COBOL2002開発マネージャが起動されます。



すぐにこの画面が表示されます。



3. プロジェクトマスタファイルの作成

COBOL2002を起動したあと、まず、ソースプログラムの作成からコンパイル・実行までの一連の作業を行うための環境を整えます。

ソースプログラムの作成から実行までを行うために必要な各種リソースやコンパイラオプション等の管理を行うファイルをプロジェクトマスタファイルと言います。最初に「デフォルトオプションの設定」をしてからプロジェクトマスタファイルを作成します。

[デフォルトオプションについて]

コンパイラに対するオプション情報をコンパイラオプションと言います。コンパイラオプションは、「プロジェクトの設定」メニューから設定することができますが、プログラムを作成するたびに設定しなければなりません。全てのプログラムに共通のコンパイラオプションは、デフォルトオプションで設定しておくことです。デフォルトオプションを設定しておくこと、新規にプログラムを作成するとき、コンパイラオプションの初期値(設定済の扱い)として有効になります。

[手順1] デフォルトオプションの設定

基本操作では、次のコンパイラオプションを設定します。

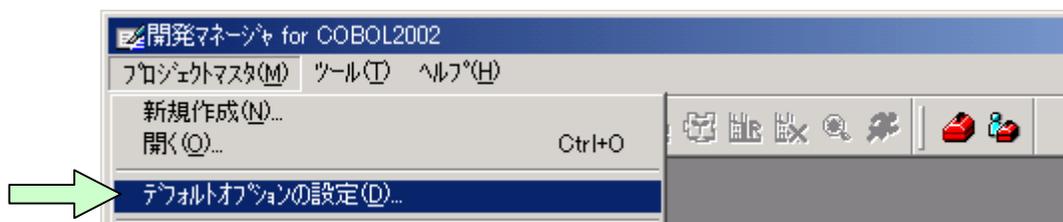
①コンパイルリストの出力

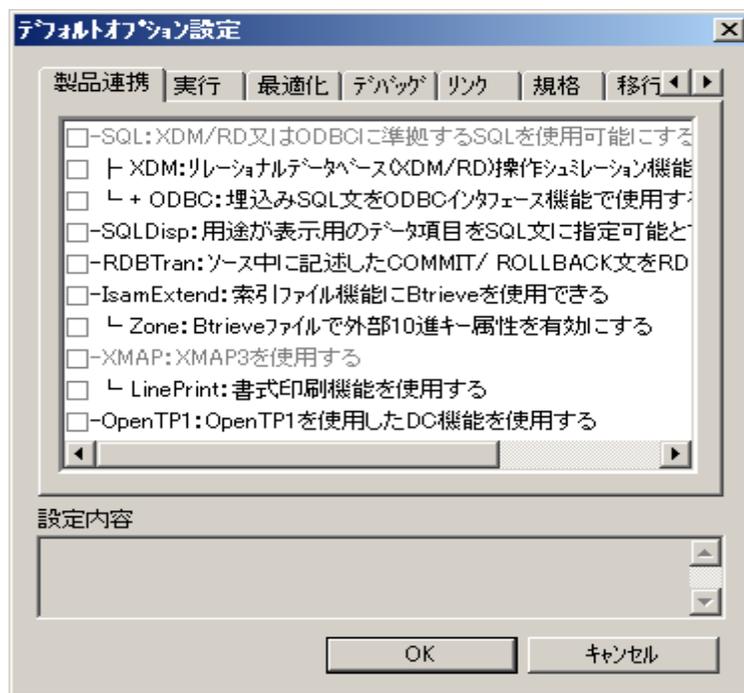
コンパイルした結果のリストが出力されます。

②デバッグ情報の出力

テストデバッガを使用してデバッグするために必要な情報が出力されます。

開発マネージャの画面から、プロジェクトマスタファイルを開いていない状態で設定します。メニューバーの「プロジェクトマスタ(M)」-「デフォルトオプションの設定(D)」をクリックしてください。「デフォルトオプションの設定」画面が表示されます。





【手順2】 該当するコンパイラオプションの設定

コンパイラオプションは、カテゴリ別に分かれて表示されます。該当するカテゴリのタブをクリックしてからコンパイラオプションをチェックします。

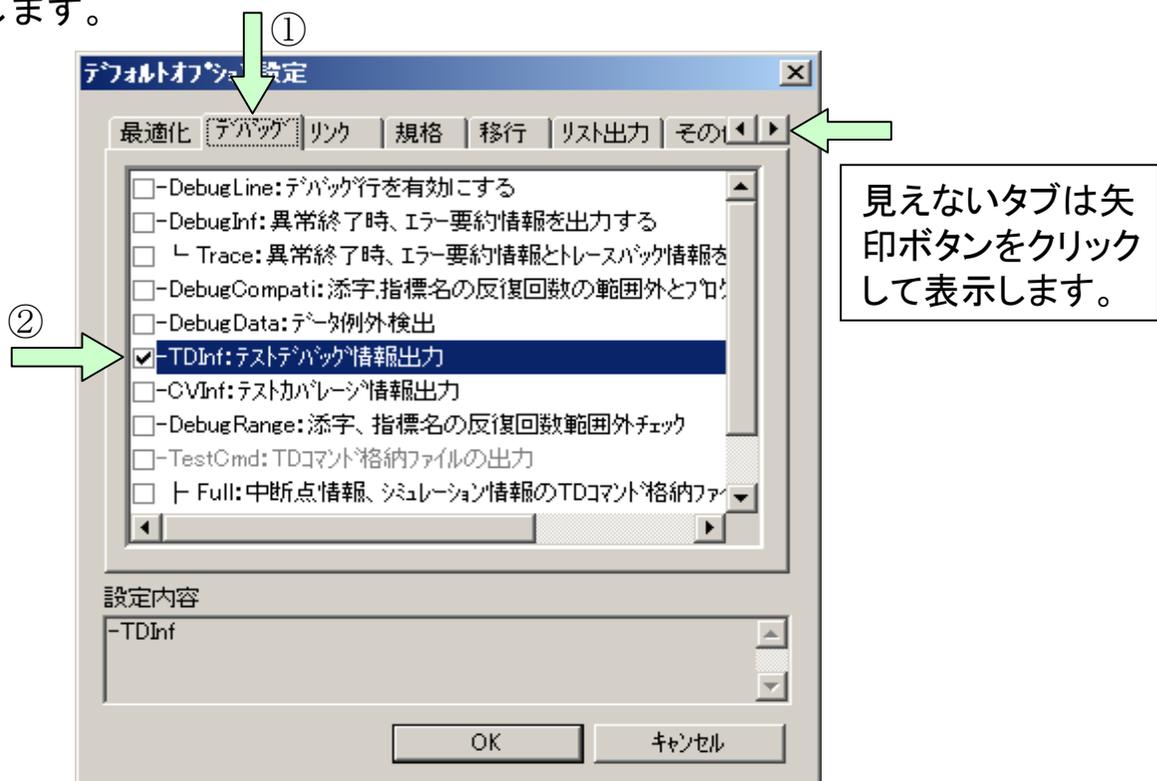
①「デバッグ」タブをクリック（下記画面参照）し、次のオプションをチェックする。

「デバッグ」タブ： -TDInf: テストデバッグ 情報出力

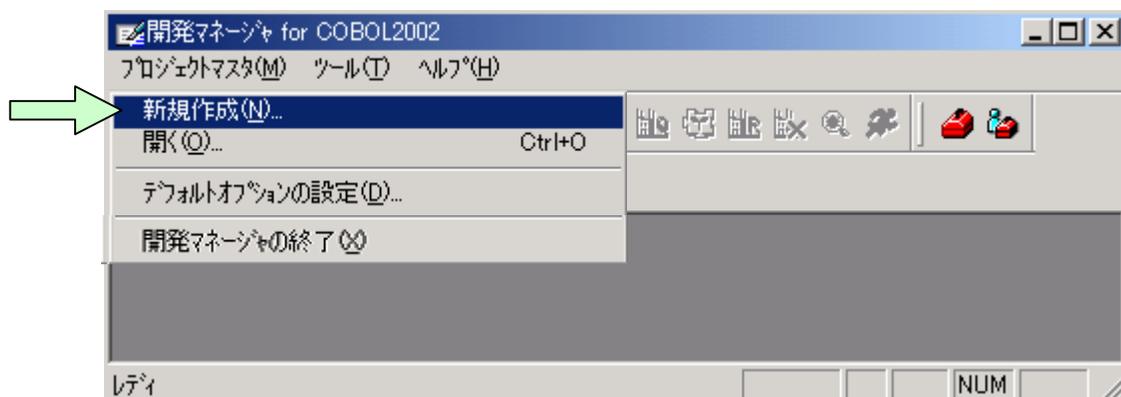
②右矢印（▶）をクリックして「リスト出力」タブを表示し、次のオプションをチェックする。

「リスト出力」タブ： - OutputAll: 全てのソースをリストに展開する

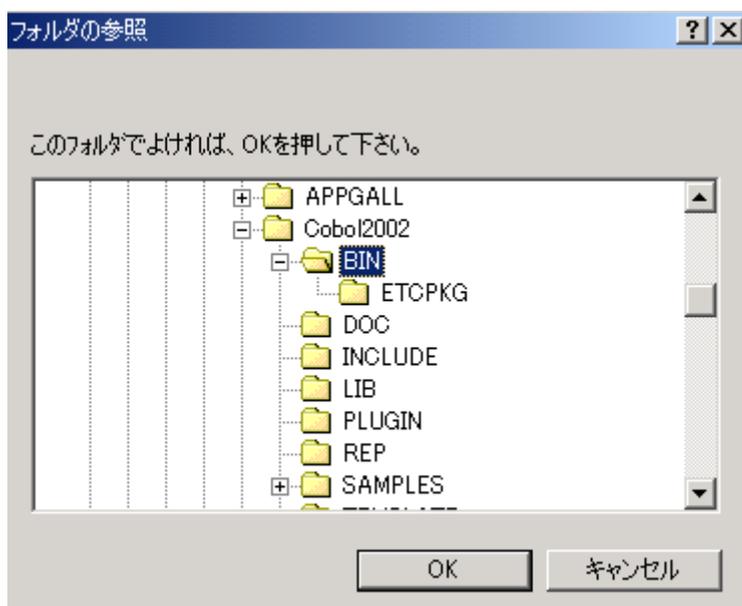
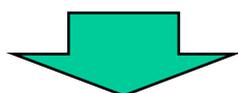
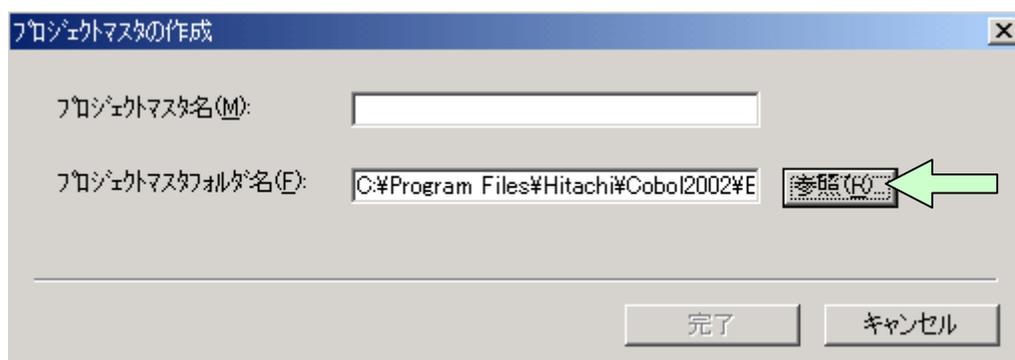
二つのオプションのチェックが完了したら、「Enter」キーを押すか「OK」ボタンをクリックします。



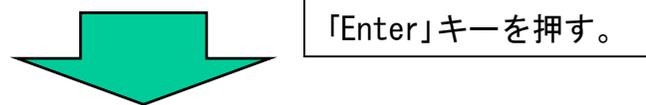
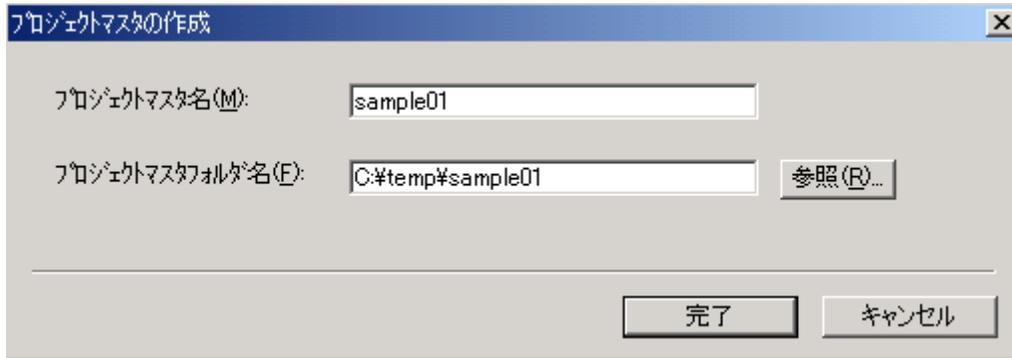
[手順 3] プロジェクトマスタファイルを新規作成します。
 開発マネージャのメニューバーの「プロジェクトマスタ(M)」-「新規作成(N)」を
 クリックしてください。すると新規作成画面が表示されます。



[手順 4] 新規作成画面の「参照(R)」ボタンを押します。
 すると、「フォルダの参照」画面が表示されます。



[手順7] プロジェクトマスタ名に「sample01」と入力し、「Enter」キーを押すかまたは「完了」ボタンをクリックしてください。



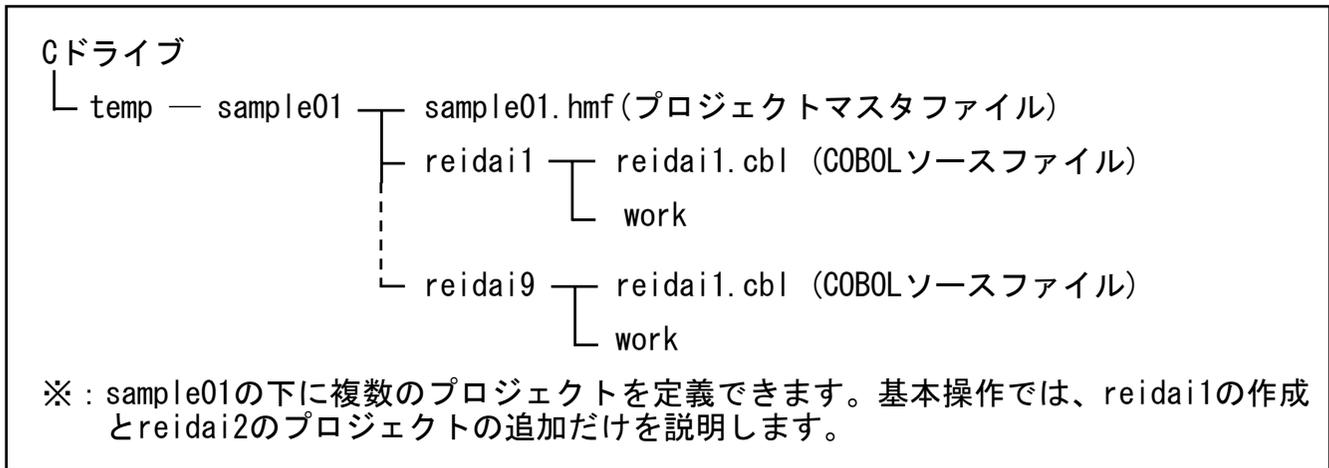
[手順8] 次の画面が表示されるので、「Enter」キーを押すかまたは「OK」ボタンをクリックしてください。



「Enter」キーを押す。

4. プロジェクトの作成

プロジェクトマスタは、プロジェクトという単位でリソースを管理します。基本操作では、Cドライブのtempフォルダの下にsample01というプロジェクトマスタを作成し、その中にreidai1というプロジェクト(プロジェクト名はソースファイル名と同じにします)を作成します。このときのフォルダ構成は次のようになります。



[概要] プロジェクトの作成

プロジェクトは、次の操作で作成します。

- ・ 入力要求には「プログラム名」を入力し、あとは全て「Enter」キーを押すだけで作成できます。

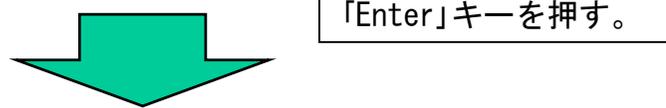
詳細の手順を、次に示します。

- [手順 1] プロジェクトマスタファイルの設定に続いて次の画面が表示されます。プロジェクト名「reidai1」を入力して「Enter」キーを押してください。(「OK」ボタンをクリックしてもよい。)

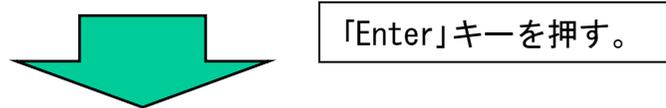


[ワンポイントアドバイス]
 プログラム名は3回入力します。毎回 入力するとスペルミスをすることがあるので、カット&ペーストで貼り付けるとよいでしょう。

[手順 2] 次の画面が順に出ますので、続けて「Enter」キーを押してください。
 (「OK」ボタンをクリックしてもよい。)

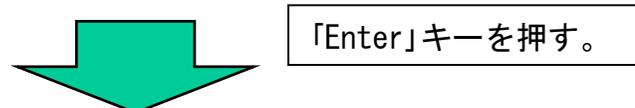
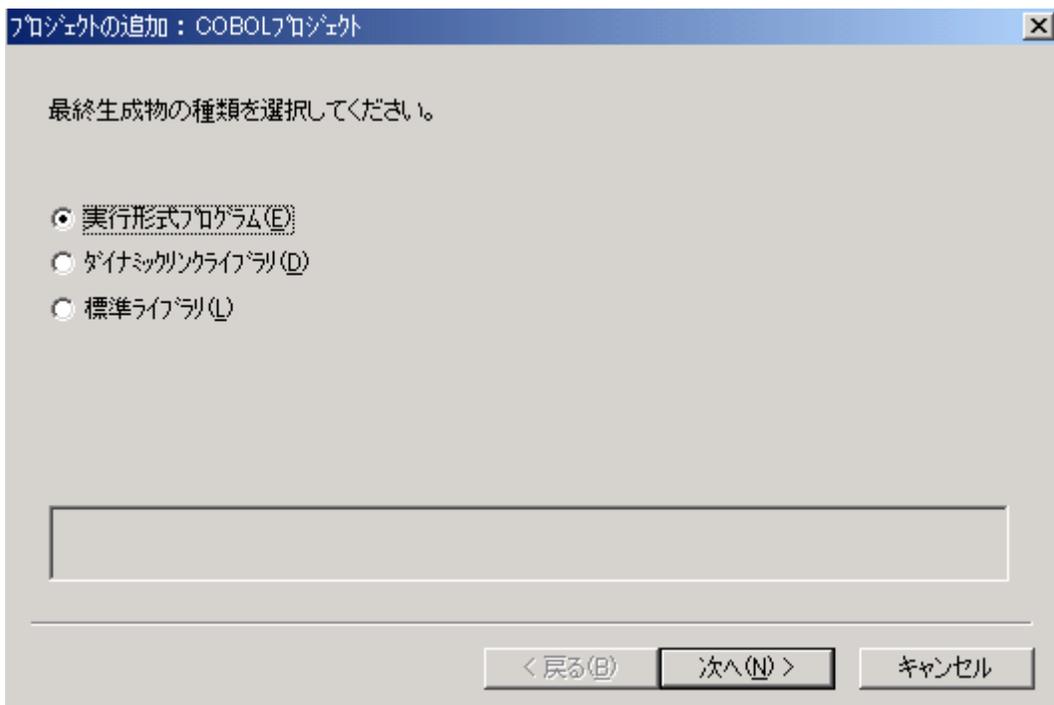


「Enter」キーを押す。



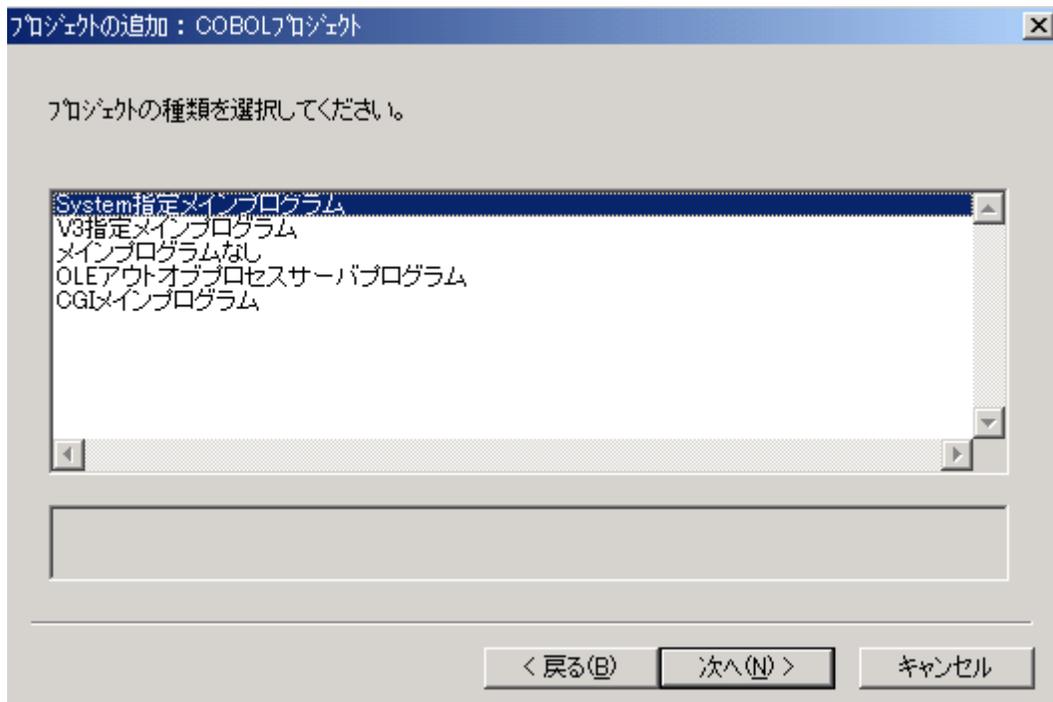
「Enter」キーを押す。

[手順 3] プロジェクトの追加画面が表示されます。デフォルトの「実行形式プログラム(E)」を指定すればよいので、この画面も「Enter」キーを押してください。(「次へ(N)」ボタンをクリックしてもよい。)



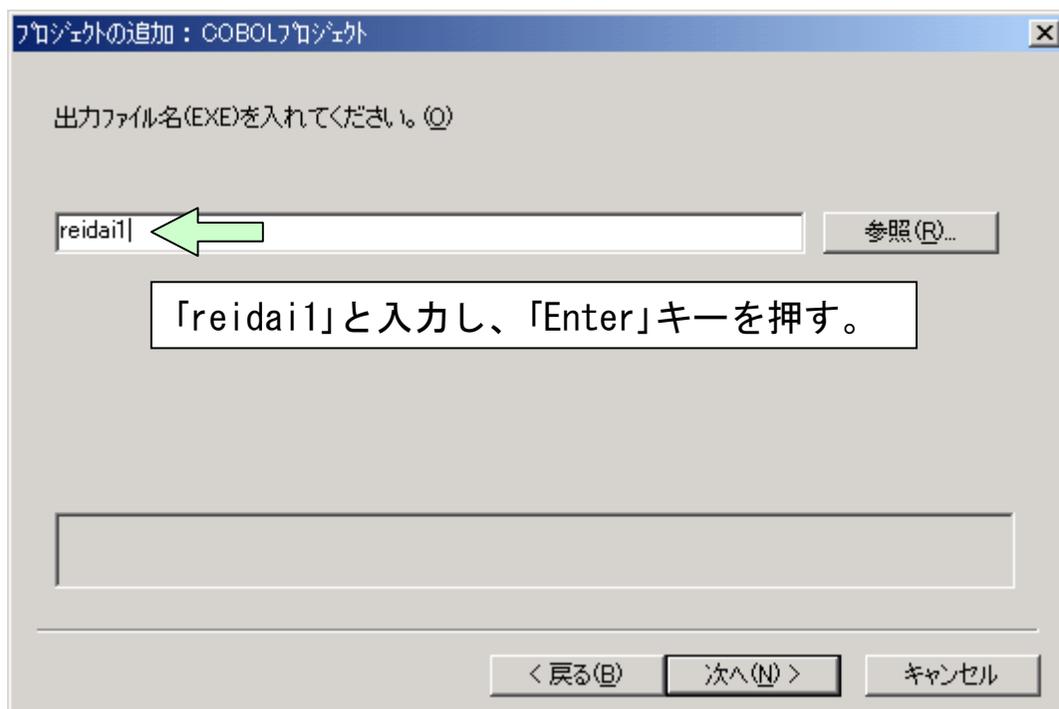
「Enter」キーを押す。

- [手順4] プロジェクトの種類もデフォルトの「System指定メインプログラム」でよいので、「Enter」キーを押してください。
(「次へ(N)」ボタンを押してもよい。)

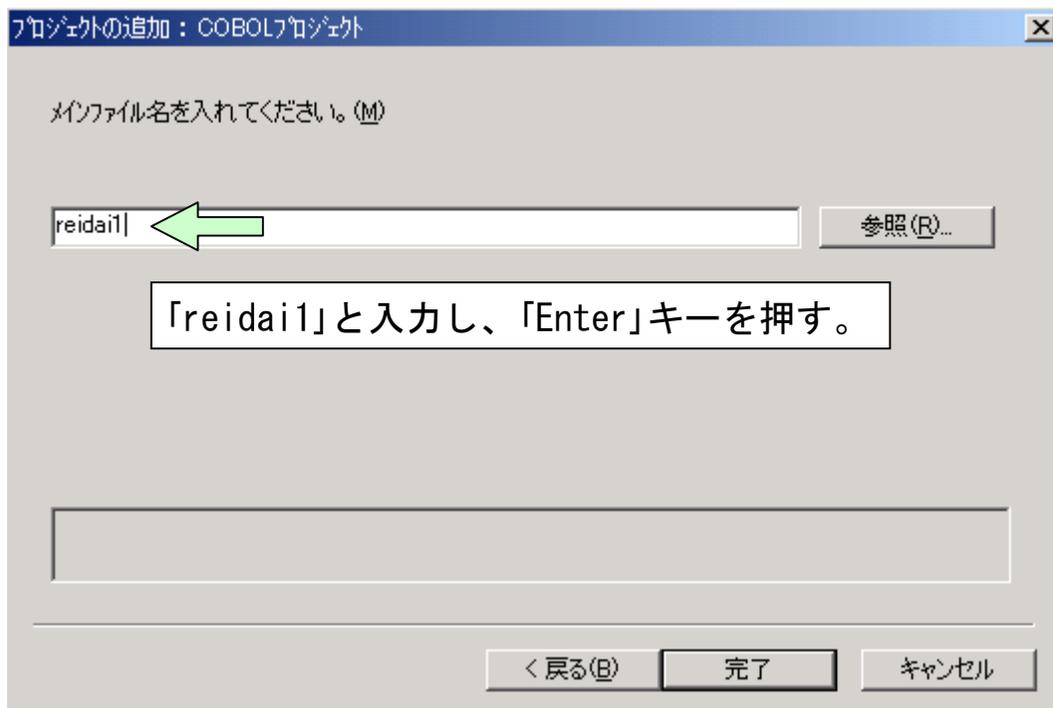


「Enter」キーを押す。

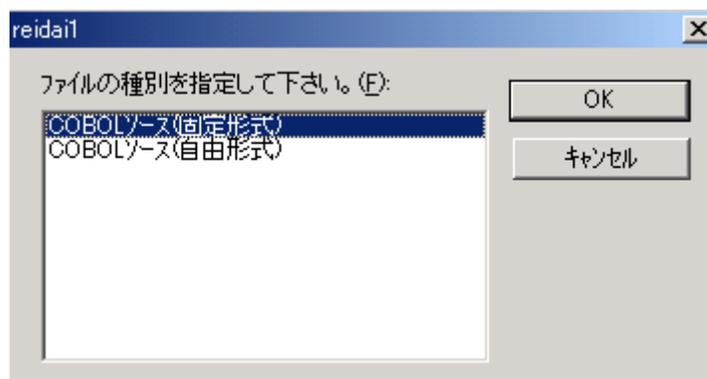
- [手順5] 出力ファイル名に「プログラム名」(プロジェクト名と同じ名称)を入力して「Enter」キーを押してください。(「次へ(N)」ボタンを押してもよい。)



- [手順6] メインファイル名も「プログラム名」(プロジェクト名と同じ名称)を入力します。続いて「Enter」キーを押してください。
(「完了」ボタンをクリックしてもよい。)



- [手順7] ファイルの種別もデフォルトの「COBOLソース(固定形式)」でよいので、「Enter」キーを押してください。
(「OK」ボタンをクリックしてもよい。)



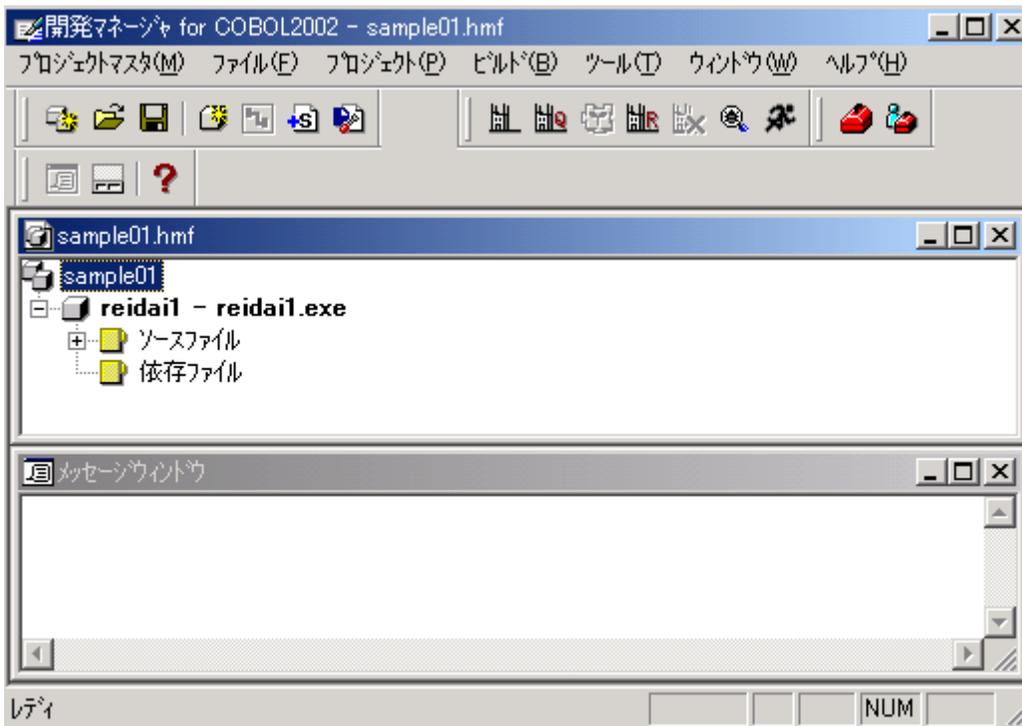
「Enter」キーを押す。

これでプロジェクトの設定が完了です。プログラム名の入力が3回、あとは「Enter」キーを押していく(8回)だけでできました。

[手順 8] 開発マネージャの画面に戻り、実行ファイルとソースファイルフォルダ、依存ファイルフォルダ、メッセージウィンドウが表示されます。

[ワンポイントアドバイス]

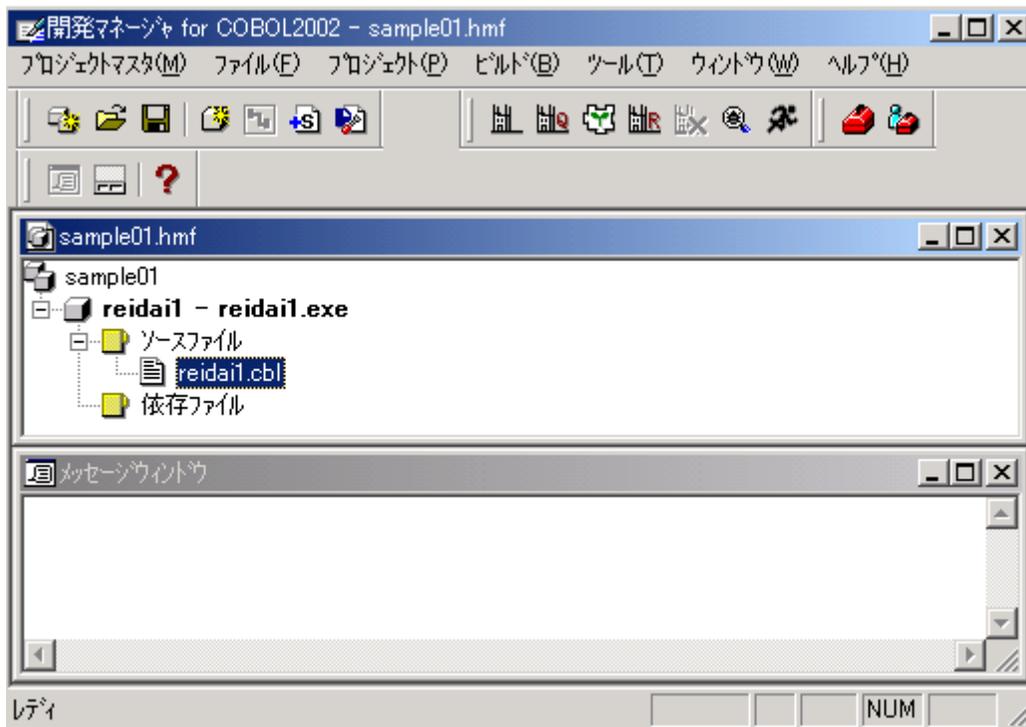
- ①ここで、一度保存しておくといいでしょう。作業終了時に開発マネージャを閉じるとき、保存するか否かを聞いてきますが、うっかり「いいえ」をクリックしてしまうとせっかく設定してきた内容が失われます。作業終了時にも保存する必要がありますが、ここで保存しておけば全てを失うことは避けられます。
- ②メッセージウィンドウが表示されない場合は、開発マネージャのメニューバーの「ウィンドウ(W)」をクリックし、プルダウンメニューの中の「並べて表示(T)」をクリックしてください。実行ファイルとメッセージウィンドウが並べて表示されます。他に「重ねて表示(C)」もできます。



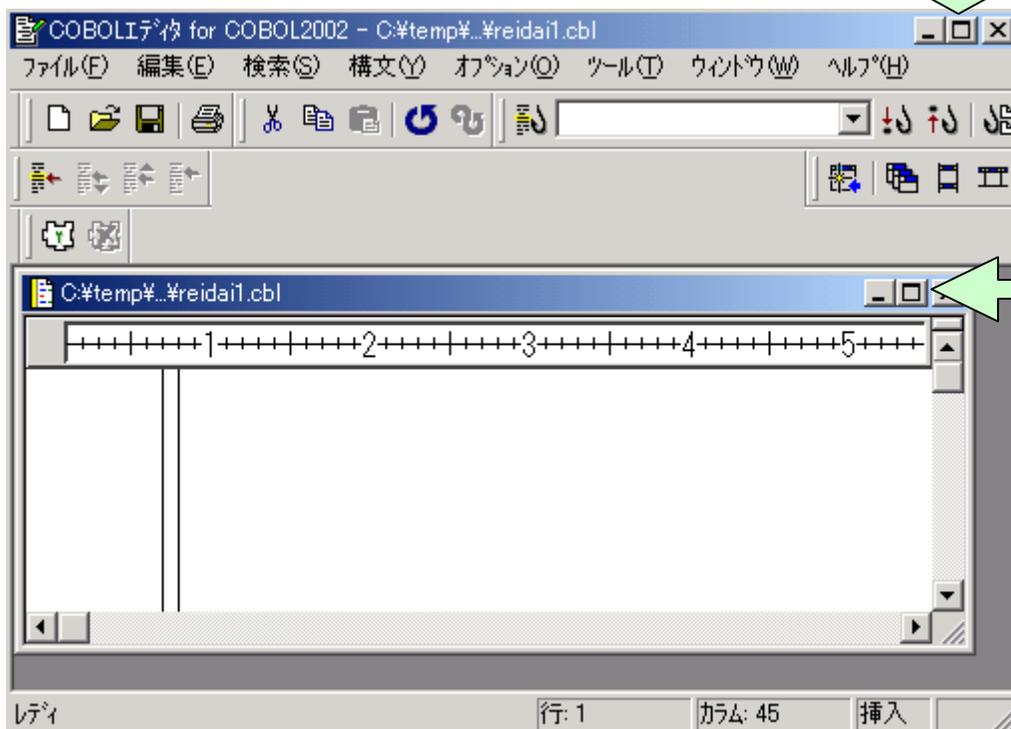
5. COBOLソースプログラムの編集

開発マネージャで、環境を作りました。この後は実際にCOBOLソースプログラムをコーディングしていきます。

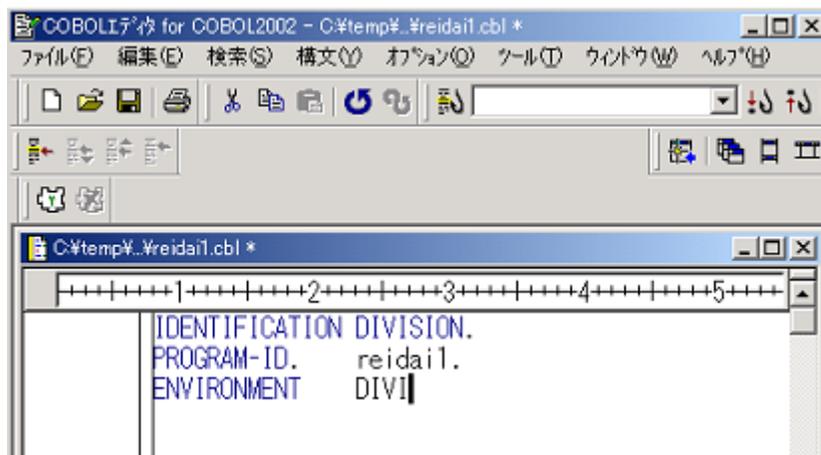
[手順1] 開発マネージャに表示されている、ソースファイルフォルダ配下にある.cblファイルをダブルクリックしてください。すると、自動的にCOBOL2002の専用エディタが起動されます。



COBOLエディタのウインドウの大きさは右上の  ボタン等で調整してください。



[手順2] エディタを使用してCOBOLソースを編集(作成)します。練習用に以下のようなプログラムをコーディングしてみてください。



[コーディング例]

8コラム目から書いてください。

```

IDENTIFICATION DIVISION.
PROGRAM-ID. reidail.
*
ENVIRONMENT DIVISION.
*
DATA DIVISION.
WORKING-STORAGE SECTION.
01 DATA0.
02 DATA1 PIC X(10) VALUE ALL '*'.
02 DATA2 PIC X(20) VALUE SPACE.
02 DATA3 PIC X(10) VALUE ALL '*'.
01 YYMMDD.
02 年 PIC 9(2).
02 月 PIC 9(2).
02 日 PIC 9(2).
*
PROCEDURE DIVISION.
*
Mein-Sec SECTION.
PERFORM 初期処理.
PERFORM 比較処理.
PERFORM 出力処理.
STOP RUN.
*
初期処理 SECTION.
ACCEPT YYMMDD FROM DATE.
*
比較処理 SECTION.
IF 月 = 9
THEN
MOVE 'September!!' TO DATA2
ELSE
MOVE 'Not September!!' TO DATA2
END-IF.
*
出力処理 SECTION.
DISPLAY DATA0.

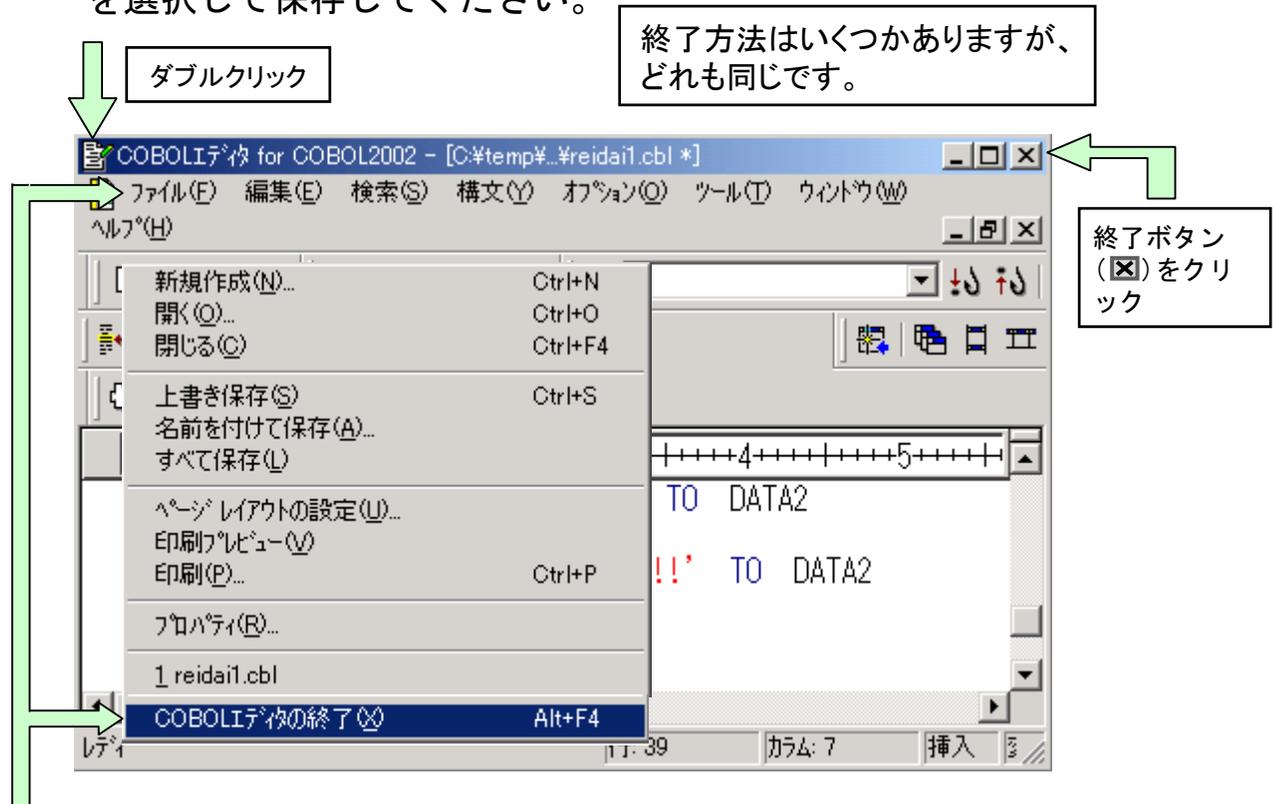
```

日本語のデータ名や手続き名が使えます。

小文字も使えます。

COBOL専用エディタは、予約語・定数等の色分け表示、キーワード補完、構文テンプレート、構文チェック等のCOBOLの文法に対応した各種機能を用意しています。

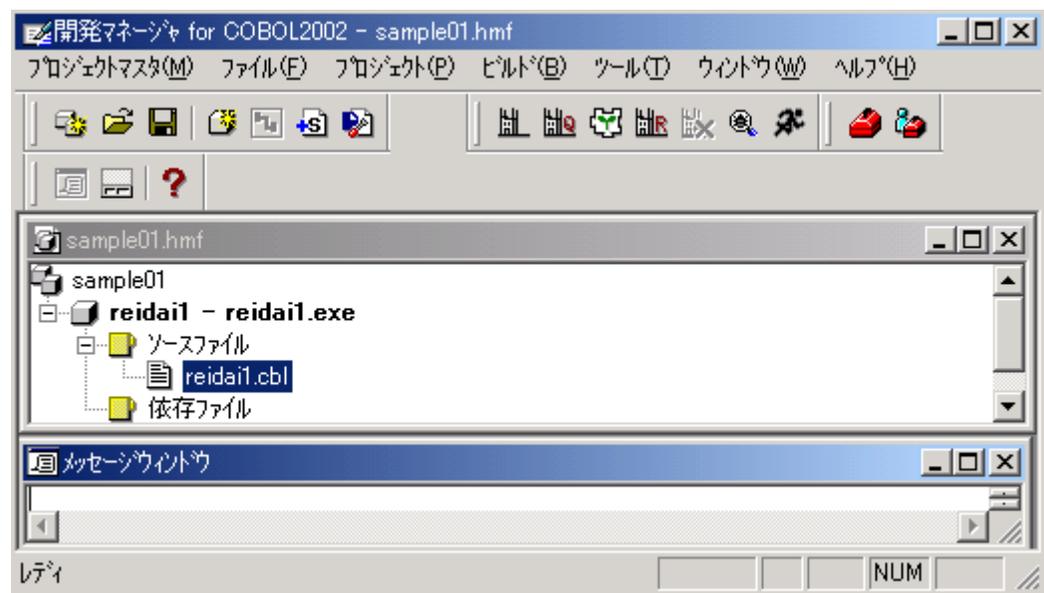
[手順3] コーディングが終了したら、終了ボタンを押してエディタを終了してください。すると、保存するかどうかの応答が返ってきますので、「はい」を選択して保存してください。



「ファイル(F)」をクリックして、プルダウンメニューから「COBOLエディタの終了」を選択



開発マネージャの画面に戻ります。



6. コンパイル(実行ファイルの生成)

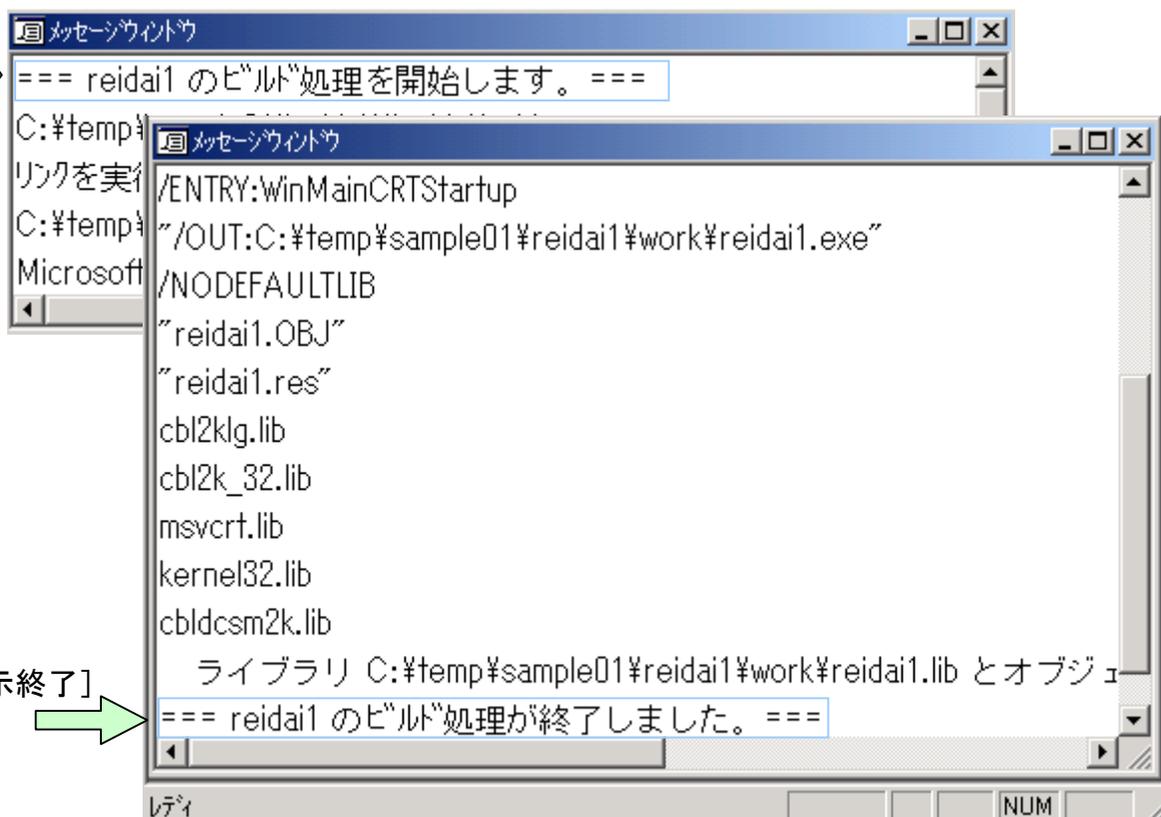
いよいよコンパイルです。コーディングしたプログラムにエラー(フラグ)がなかったら、そのままリンケージ(結合・編集)が行われ、実行ファイルが生成されます。エラーがある場合は、エラーメッセージが出力されます。

[手順 1] 開発マネージャの上の方にあるビルドボタン()をクリックします。



[表示開始]

コンパイルが開始されると、メッセージウインドウにメッセージが表示されます。

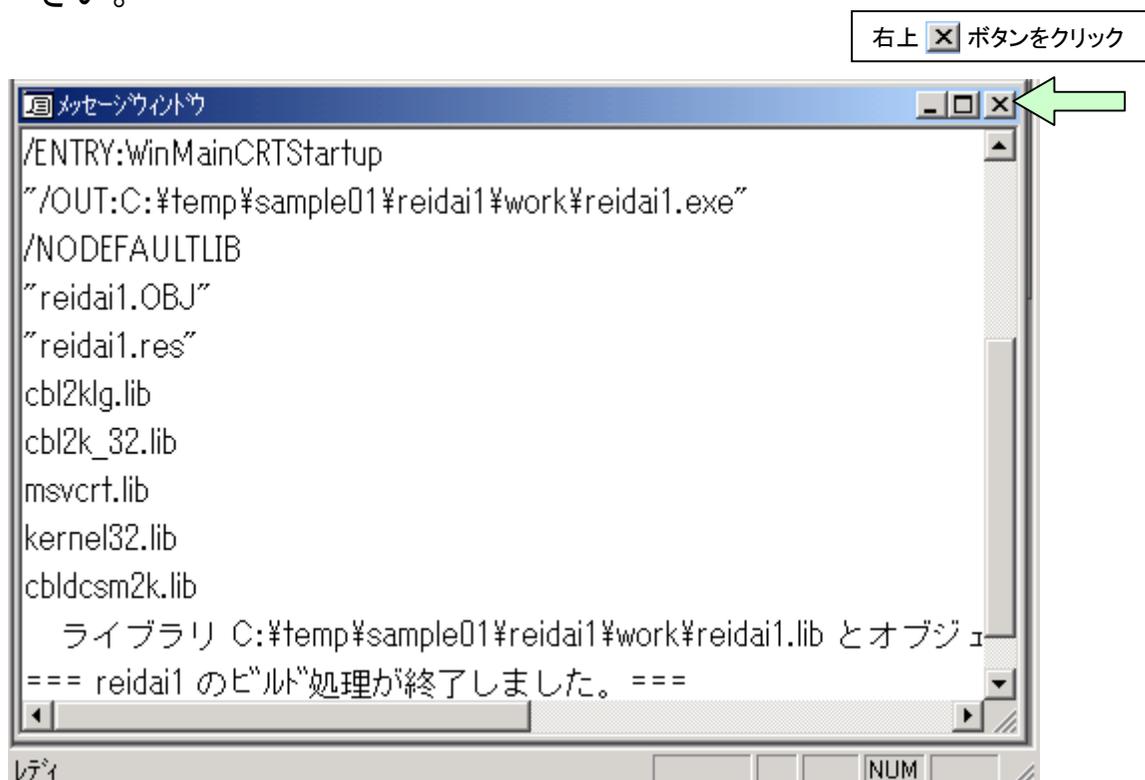


[表示終了]

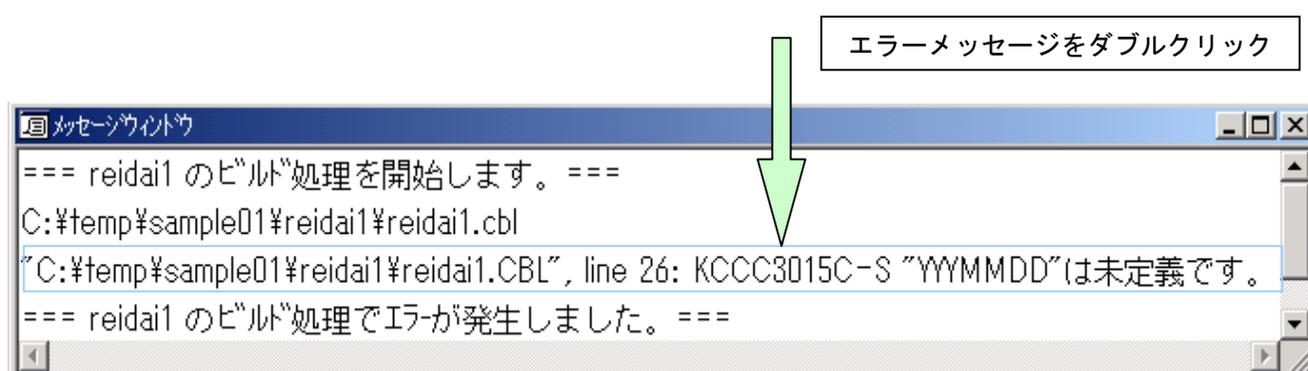
[用語解説] **ビルド**

コンパイルとリンケージを一括して行うことを、「ビルド」といいます。

- [手順2] ビルドが終了したら、メッセージウインドウを閉じてください。これで、コンパイルは終わりです。メッセージウインドウに「KCCCXXXX」のエラーメッセージが出力されたときは、手順3以降を参照してください。

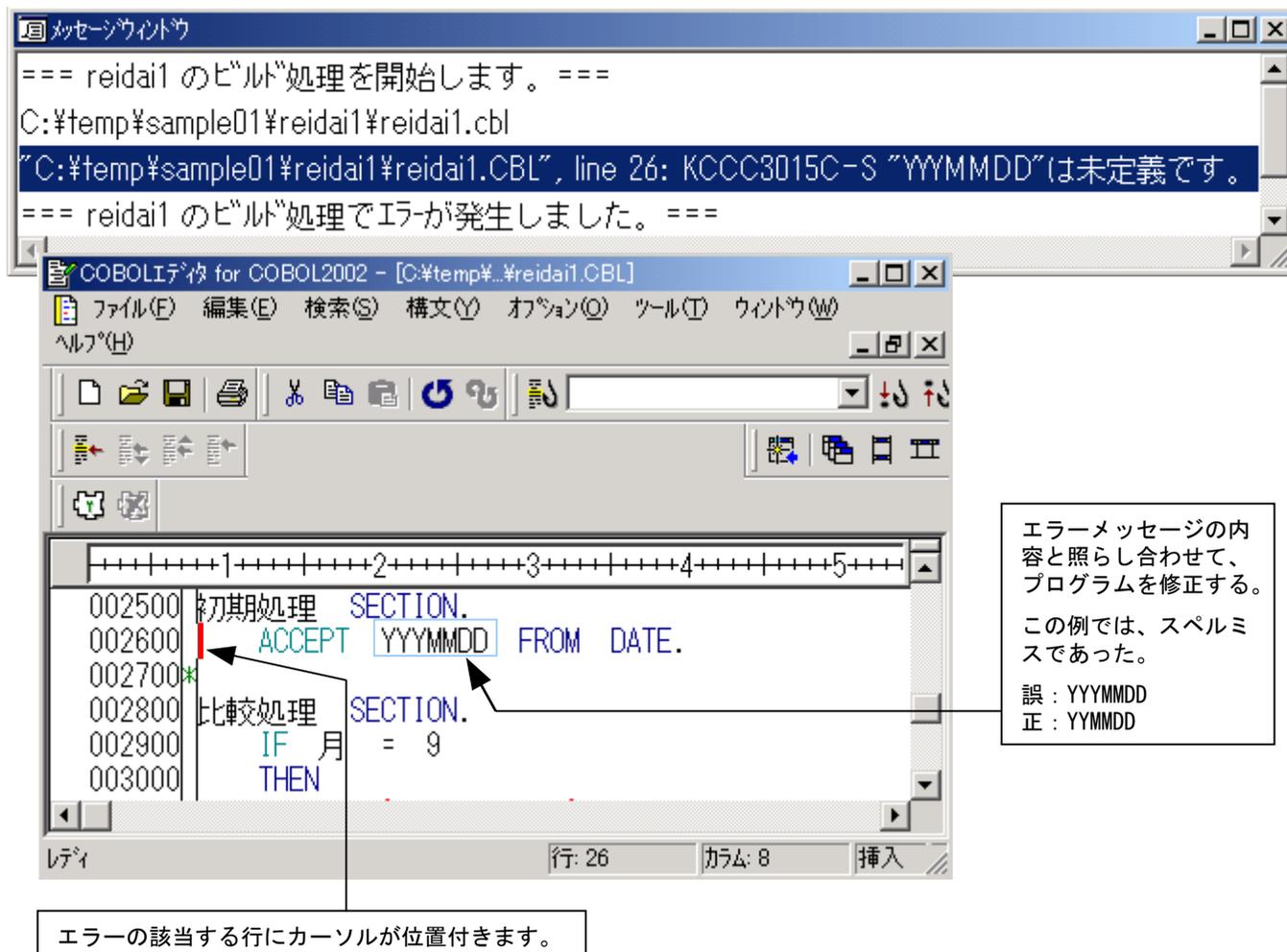


- [手順3] エラーがなくビルドが終了していたら、次の章の「実行」に進んでください。ここからは、コンパイルエラーが出たときのソース修正の方法を説明します。
メッセージウインドウ中に表示されているエラーメッセージをダブルクリックしてください。すると、エディタが自動的に起動されます。



エラーメッセージが見にくい場合は、スクロールバーを使ってスクロールするか、メッセージウインドウの大きさを調整することで、見やすいようにしてください。

[手順4] エディタが自動起動し、エラーに該当する行の先頭にカーソルが位置付きます。先のエラーメッセージの内容と照らし合わせて、エラーを修正してください。



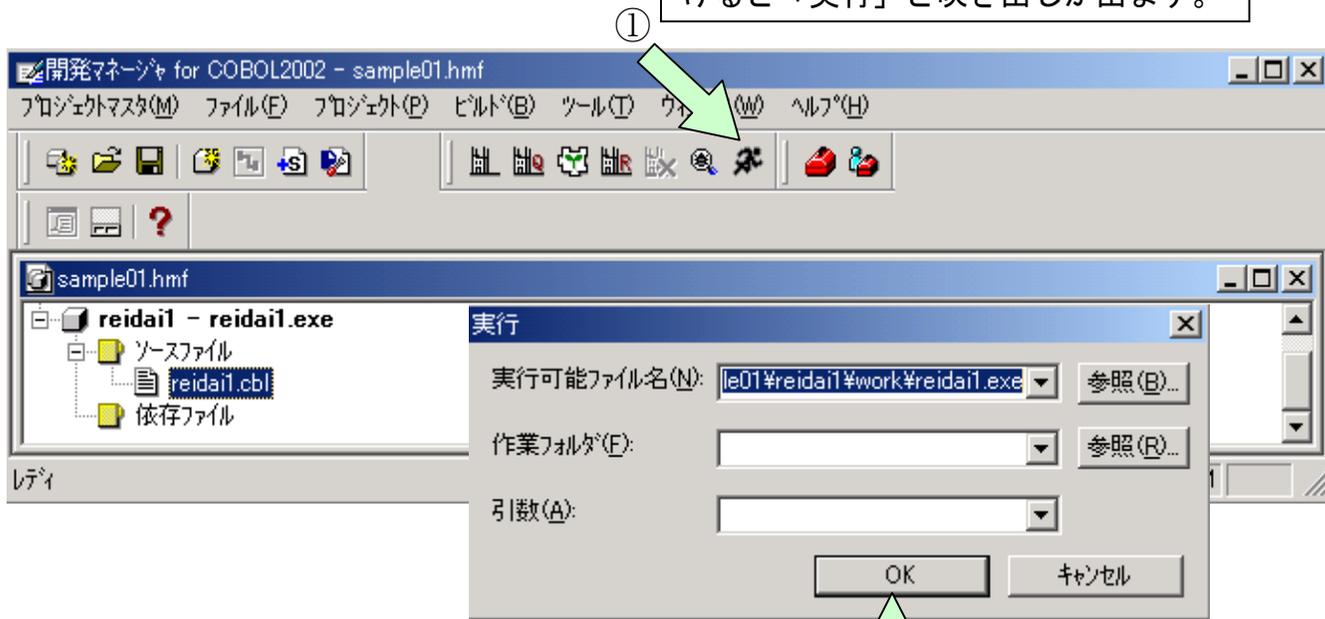
[手順5] エラーの修正が終わったら、エディタとメッセージウインドウを閉じて手順1に戻り、コンパイルからやり直してください。
なお、エラーが複数ある場合は、メッセージウインドウ上のエラーメッセージを次々にダブルクリックすれば、エディタの該当位置に位置付きます。また、一つのエラーのために複数のエラーが派生することや、一つのエラーに隠れて他のエラーが検知できない場合もありますので、ご注意ください。

7. 実行

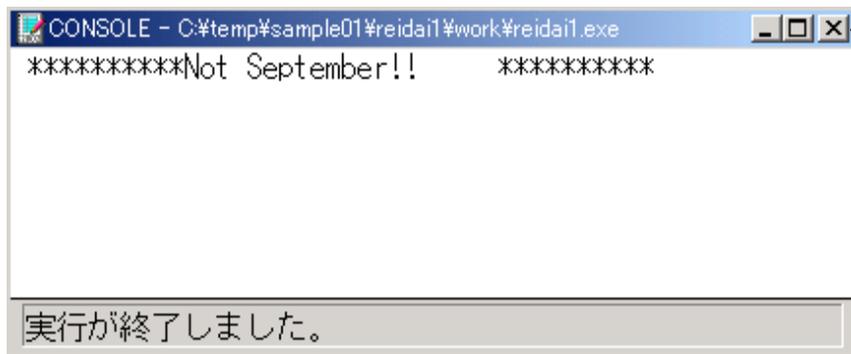
次は実行です。今回の例題では特に入出力ファイルを用いていないので、簡単に実行できます。入出力ファイルがある場合は、ファイルの割り当てを行う必要があります。ファイルの割り当てについての説明は、「後述の3. 関連資料」の「(a) ファイルの入出力処理」を参照してください。

[手順 1] 開発マネージャの上の方にある実行ボタン () をクリックします。このボタンを「実行ボタン」と呼びます。実行ボタンをクリックすると、実行確認画面が出ます。この画面の「OK」ボタンをクリックしてください。

このボタンにマウスポインタを位置づけると「実行」と吹き出しが出ます。



[CONSOLE画面]



今回の例題のDISPLAY文の実行結果は、CONSOLE画面に表示されます。
この画面を終了させるには、  ボタンをクリックします。

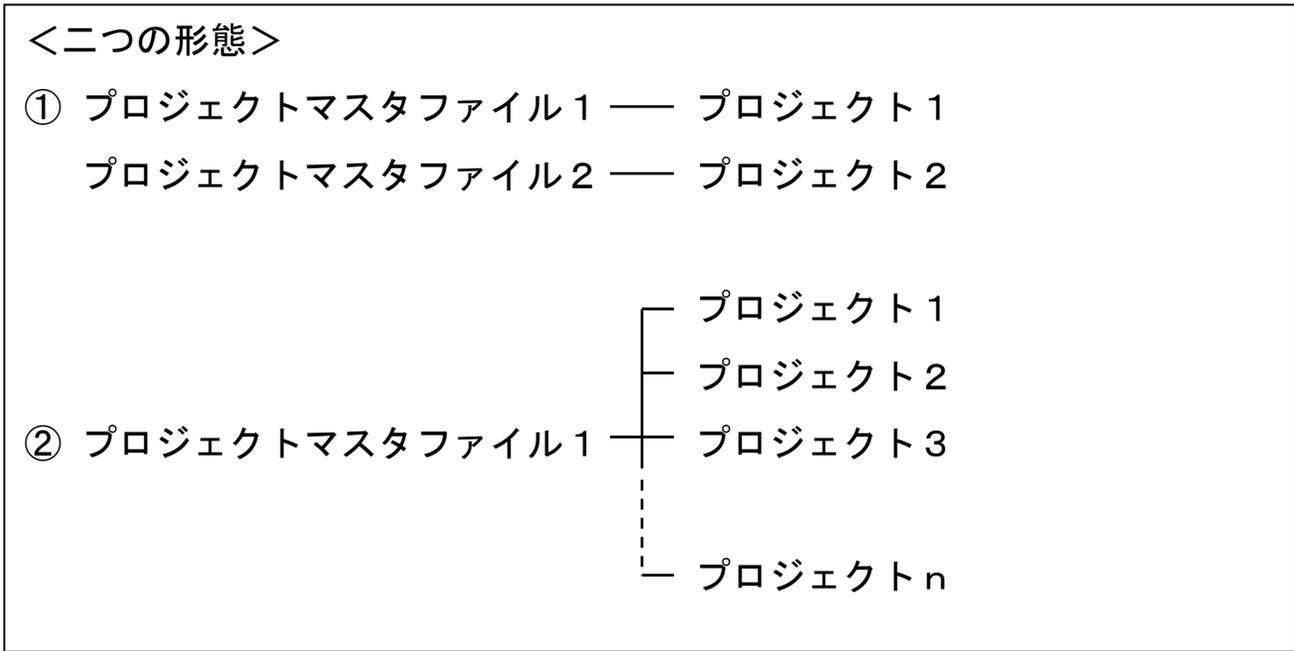
[ワンポイントアドバイス]
CONSOLE画面を閉じないうちは、実行ファイルは起動されたままの状態になっています。このままにしておくと、再度コンパイルしたときエラーになります。実行結果を確認したら、必ずCONSOLE画面を閉じるようにしましょう。

8. プロジェクトの追加

プログラムの作成から実行までの操作を一通り説明しました。ここでは、新たにプロジェクトを登録する方法と注意事項について説明します。

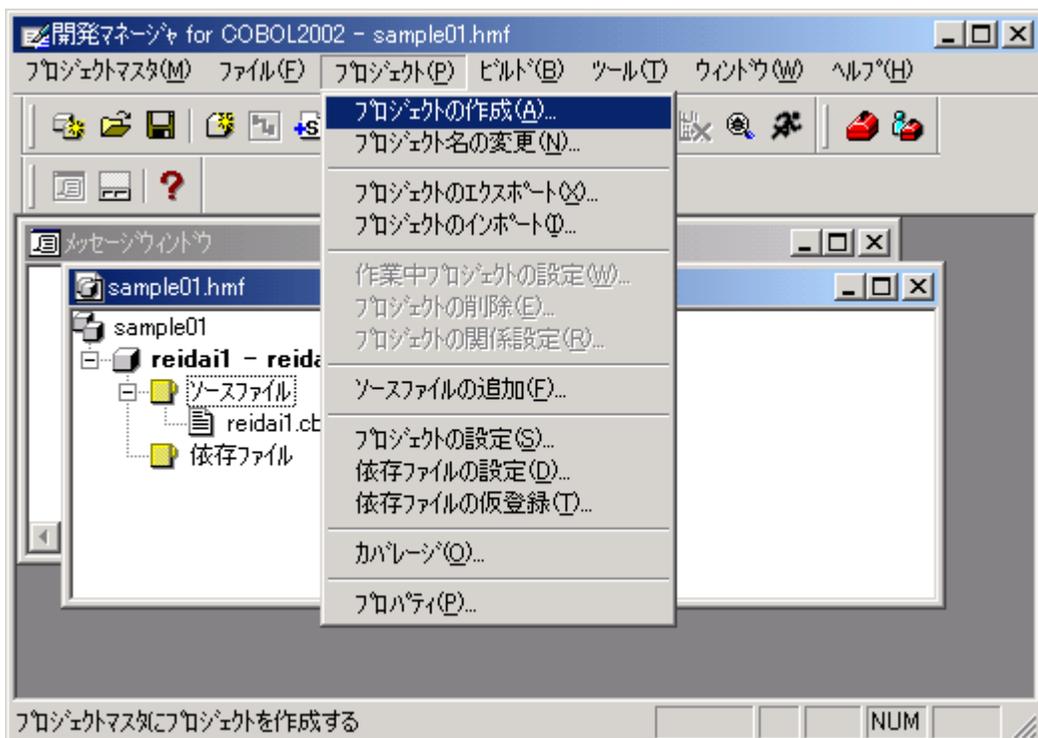
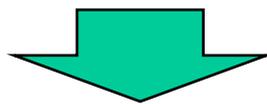
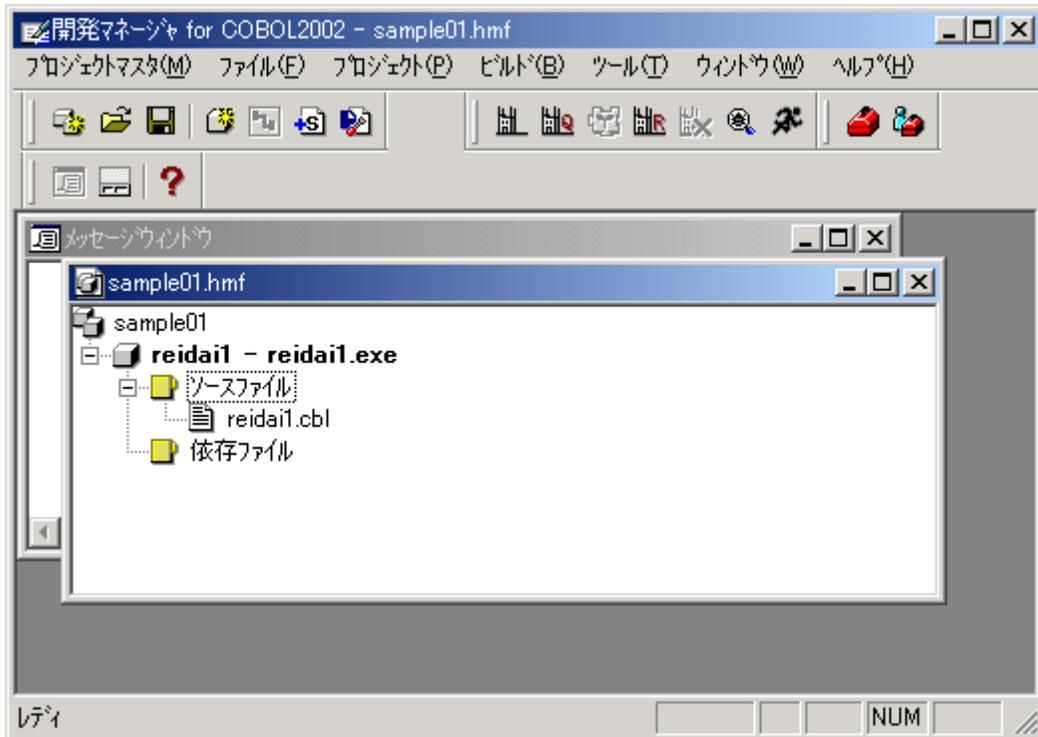
最初にも述べましたが、一つのプロジェクトマスタファイルの中に複数のプロジェクトを登録できます。もちろん、プロジェクト毎にプロジェクトマスタファイルを作成してもかまいません。この場合は、これまで説明した手順にしたがって作業してください。

プロジェクトを登録するときの二つの形態を以下の図に示します。



それでは、既にプロジェクトを作成したプロジェクトマスタファイルに新たなプロジェクトを追加する手順を示します。

[手順 1] 新たなプロジェクト「reidai2」を追加します。開発マネージャの画面から「プロジェクト(P)」-「プロジェクトの作成」の順にクリックします。



[手順2] 続いて、「プロジェクトの作成」画面が表示されます。
 これは、「4. プロジェクトの作成」の手順1の画面と同じです。
 もうおわかりですね。ここからは、これまで説明したプロジェクトの
 作成手順にしたがってください。おさらいの意味で画面の遷移を以下
 に示します。



プロジェクト名を入力する。



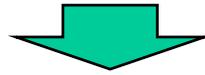
「Enter」キーを押す。



「Enter」キーを押す。



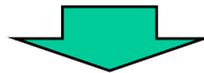
「Enter」キーを押す。



プロジェクトの追加 : COBOLプロジェクト

最終生成物の種類を選択してください。

実行形式プログラム(E)
 ダイナミックライブラリ(D)
 標準ライブラリ(L)



「Enter」キーを押す。

プロジェクトの追加 : COBOLプロジェクト

プロジェクトの種類を選択してください。

System指定メインプログラム
 V3指定メインプログラム
 メインプログラムなし
 OLEアウトオブプロセスサーバプログラム
 CGIメインプログラム
 GUIプログラム



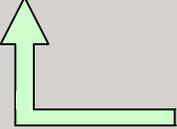
「Enter」キーを押す。

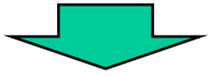


プロジェクトの追加：COBOLプロジェクト

出力ファイル名(EXE)を入れてください。(O)

reidai2

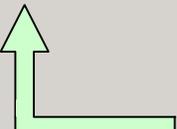
 「reidai2」と入力し、「Enter」キーを押す。

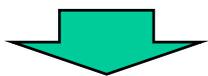


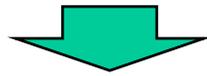
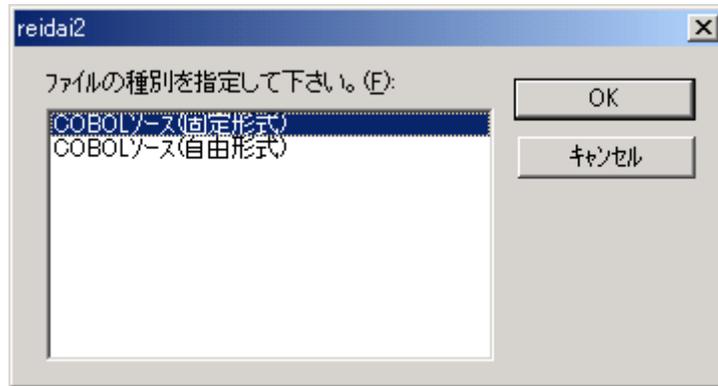
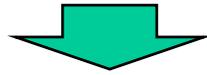
プロジェクトの追加：COBOLプロジェクト

メインファイル名を入れてください。(M)

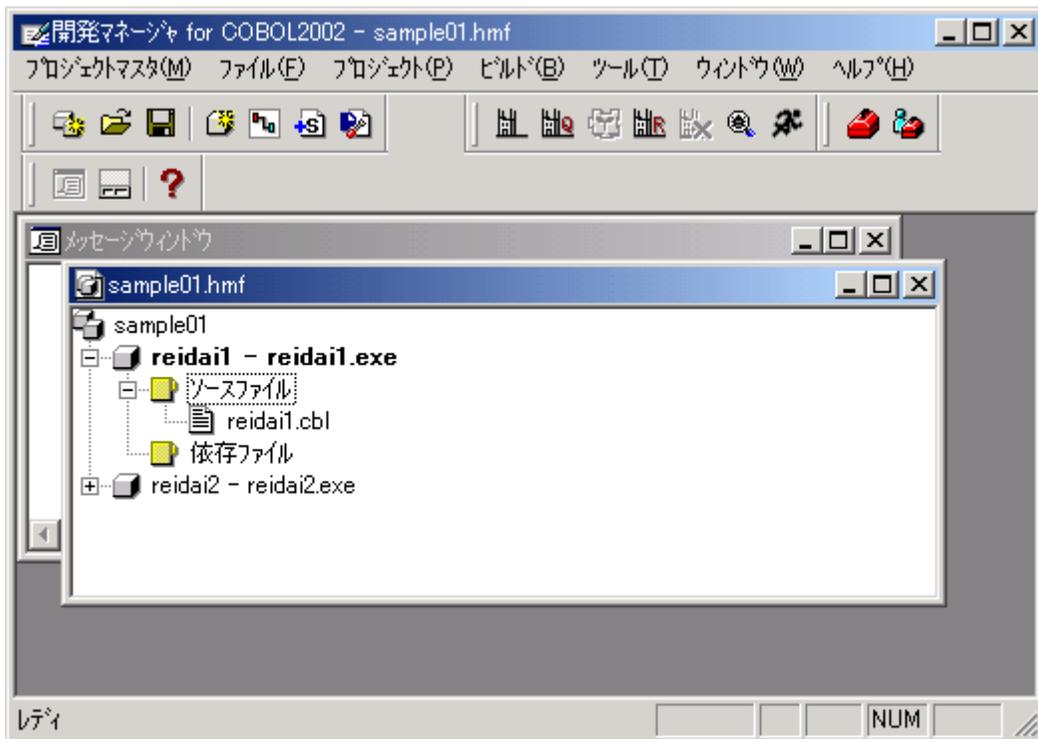
reidai2

 「reidai2」と入力し、「Enter」キーを押す。





「Enter」キーを押す。

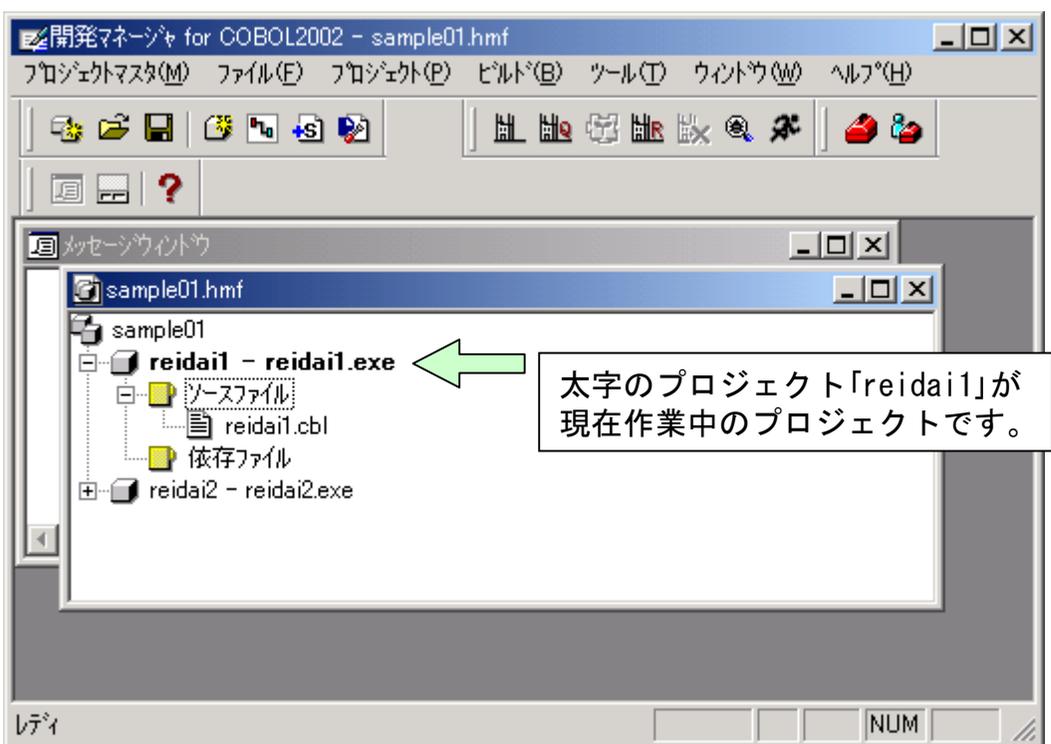


プロジェクトの追加が完了しました。

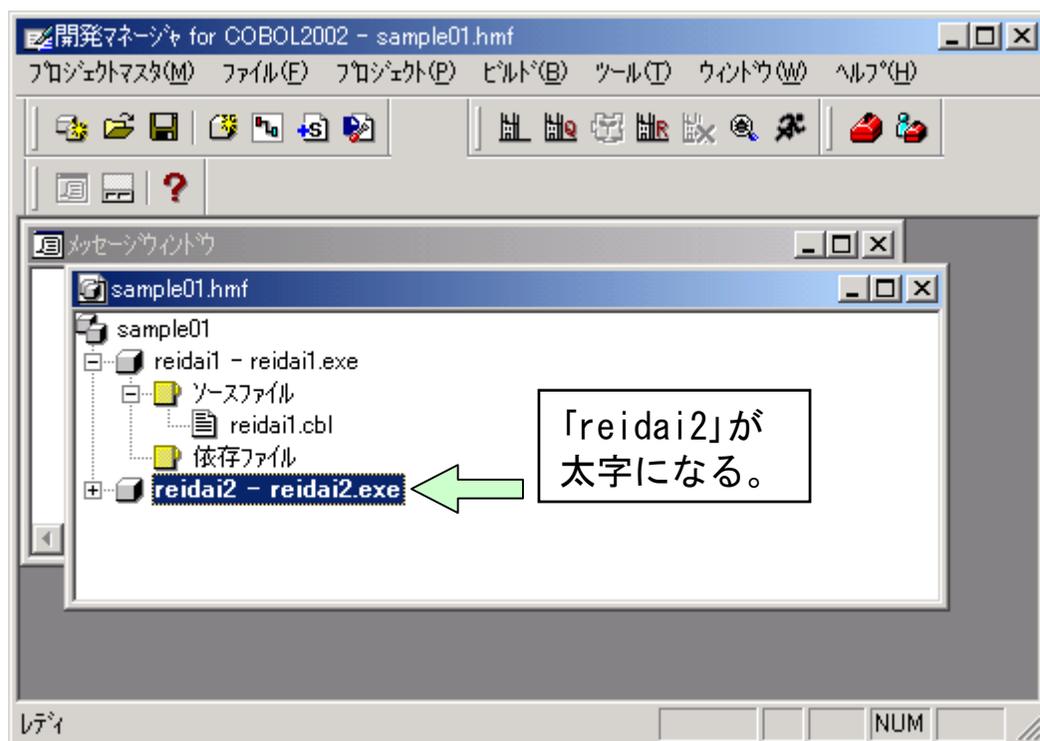
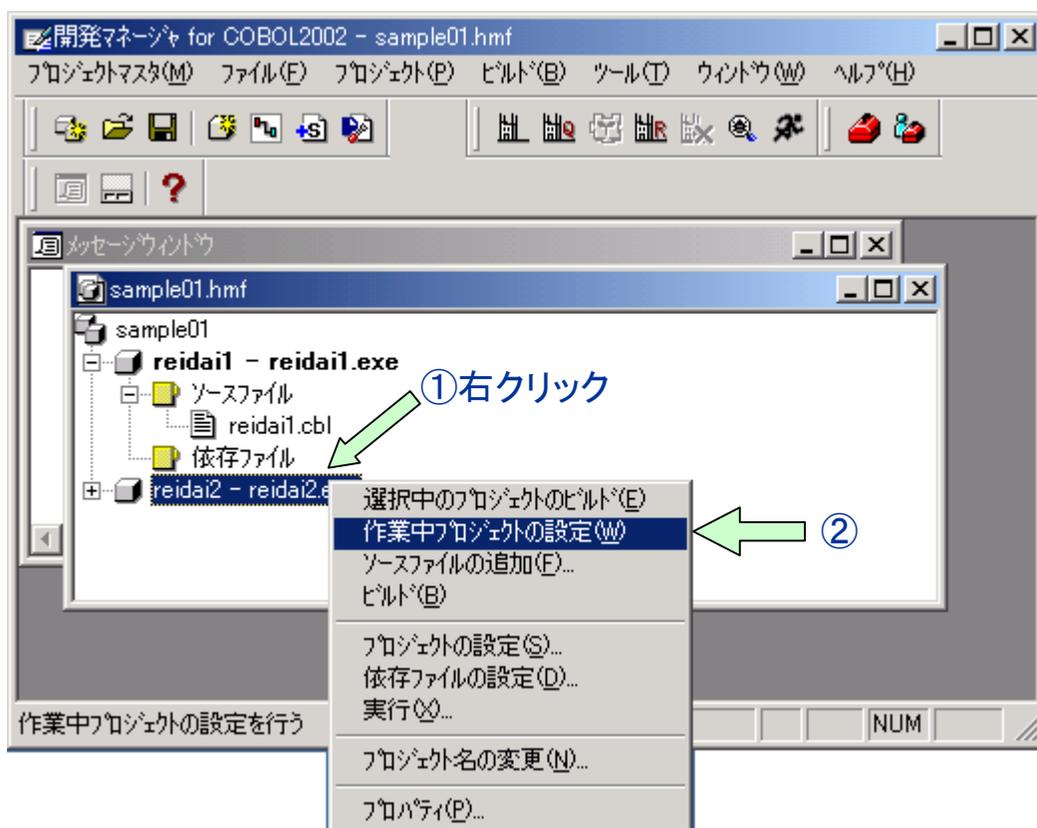
続いて、複数プロジェクト環境で作業をするときに必要な「作業対象プロジェクトの選択」と「プロジェクトの削除」について説明します。

作業対象プロジェクトの選択

プロジェクトが複数ある場合、作業を開始するときに、どのプロジェクトの作業をするかを明示的に指定する必要があります。
 開発マネージャの画面で、一つだけ太字で表示されているプロジェクトがあります。これが、現在作業中のプロジェクトです。



プロジェクト「reidai2」の作業をしたい場合、「reidai2」をクリックします。
 次に右クリックして表示されるプルダウンメニューの中から「作業中プロジェクトの設定 (W)」を選択します。

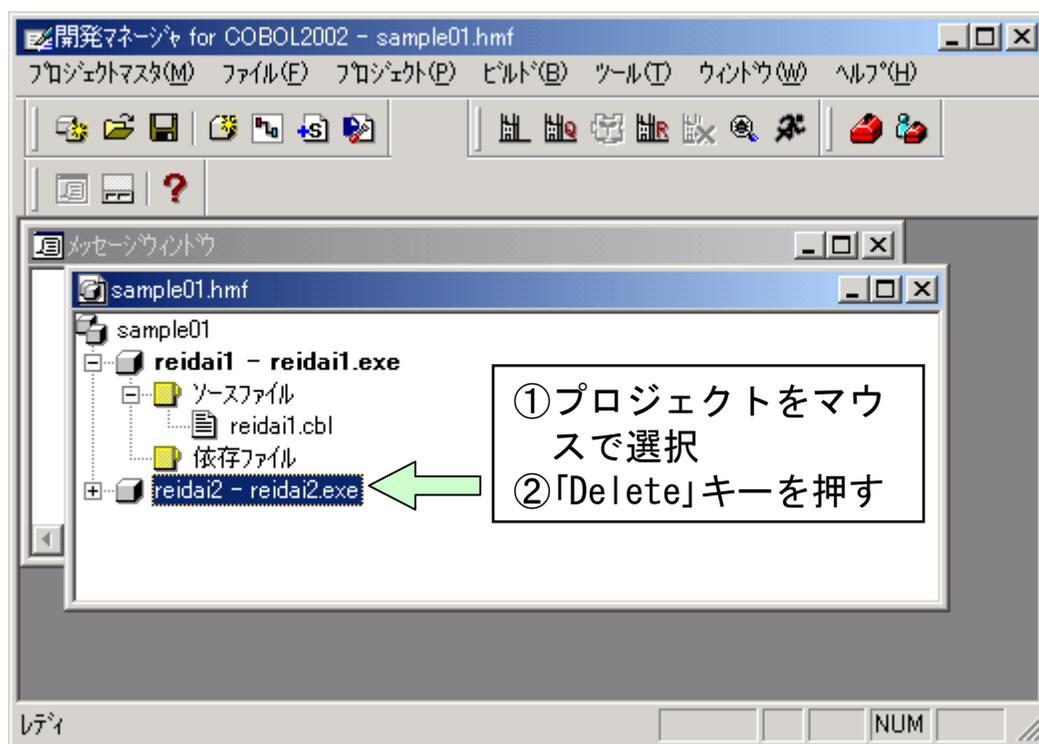


Reidai2に対して、ソース編集やビルド、実行を試してみてください。
 なお、ソースの編集は、該当ソースファイルをダブルクリックして編集するため、プロジェクトを選択しなくても編集作業が可能です。しかし、ビルド等をするときには、作業中のプロジェクトを設定しておかなければなりません。

プロジェクトの削除

プロジェクトを削除したい場合は、次のように操作します。

削除したいプロジェクトをマウスで選択(クリック)し、「Delete」キーを押します。



9. 終わりに

以上で、プロジェクトの作成、及びプログラムの作成から、コンパイル、エラー修正、実行までの一連の操作の説明は終わりです。

いろいろなプログラムを試してみてください。

テストデバッグ機能をお使いになりたい場合は後述の「2. テストデバッガの使用方法」をご覧ください。また、順ファイル等のファイルを使用するプログラムの実行については、後述の「3. 関連資料」の「(a) ファイルの入出力処理」を、エディタの細かい設定方法については後述の「3. 関連資料」の「(e) エディタ設定方法」をご覧ください。

最後に、reidai1のプログラムのコンパイルによって生成されたファイルを示しますので、エクスプローラーで参照してみてください。コンパイラの生成物は、「reidai1」フォルダの下の「work」フォルダに格納されています。

Reidai1.exe : 実行可能ファイルです。「7. 実行」で実行したのはこのファイルです。

Reidai1.lst : コンパイルリストです。

Reidai1.cbp : デバッグ情報を格納したファイルです。

その他 : 各種管理情報を持つファイルです。

