

Micro Focus メインフレームソリューション

スターターズキット

7. Enterprise Developer for Visual Studio 2012 on Windows8 : データベース連携

7.1 目的

当チュートリアルではデータベース接続を行うための下記手順の習得を目的としています。

- 1) プリコンパイルについて
- 2) プロジェクトの作成と既存プログラムの取り込み方法
- 3) プロジェクトと個別ファイルプロパティの設定方法
- 4) XA スイッチモジュールの作成方法
- 5) プログラムのコンパイル方法
- 6) Enterprise Server 設定方法
- 7) XA リソース正常ロードの確認方法
- 8) JCL の実行と確認方法

7.2 チュートリアル的前提と準備

本チュートリアル実施に伴う前提条件と、例題プログラムに関連する資源を用意します。

- 1) 「5. Enterprise Developer JCL チュートリアル」を終了していることを前提とします。
- 2) 対象データベースクライアントがインストール済で接続確認済であることを前提とします。
- 3) キットに添付されている DBtutorial.zip を C:¥ 直下へ解凍し、C:¥DBtutorial を作成します。

本書作成時に使用したデータベース環境は下記の通りです。

Oracle

Oracle クライアント 11g 11.2.0.4.0 64bit

DB: リモート環境 (Oracle 10g 10.2.0.1.0 64bit)

(Oracle Pro*COBOL プリコンパイラ使用)

DB2

DB2 クライアント 9.7.900.250 64bit

DB: ローカル環境 (DB2 9.7.900.250 64bit)

(DB2 PREP プリコンパイラ使用)

SQL Server

ODBC ドライバー: SQL Server Native Client 11.0

DB: ローカル環境 (SQL Server 2012 R1 64bit)

(ODBC 使用)

7.3 プリコンパイルについて

埋め込み SQL 付き COBOL ソースは下記 2 通りのプリコンパイル方法があります。

- 1) 予め Micro Focus 形式の COBOL ソースを作成してプリコンパイルする。
- 2) 製品機能のプリプロセッサ機能からプリコンパイラを呼び出し内部的にプリコンパイルする。

ここでは、オリジナルソースイメージでデバッグが可能かつ管理が容易な 2) を紹介します。

JCL バッチからの使用方法として紹介していますが、CICS や IMS プログラムで Oracle, DB2, SQL Server などとのデータベース連携を XA リソース方式で行う場合も同様の設定となります。

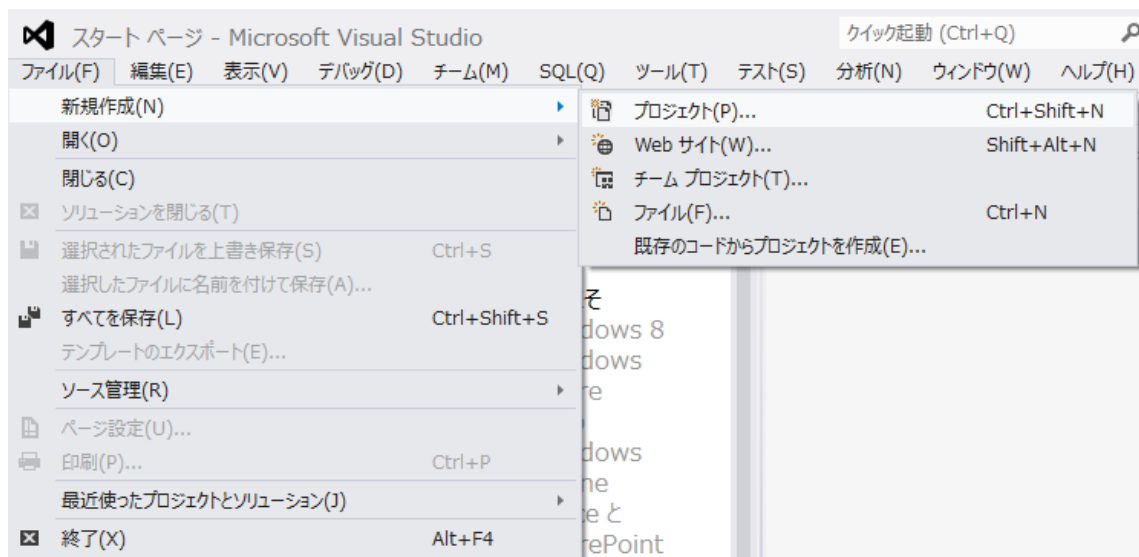
7.4 プロジェクトの作成とプログラム取り込み

- 1) スタートアイコンから Enterprise Developer for Visual Studio 2012 を起動します。



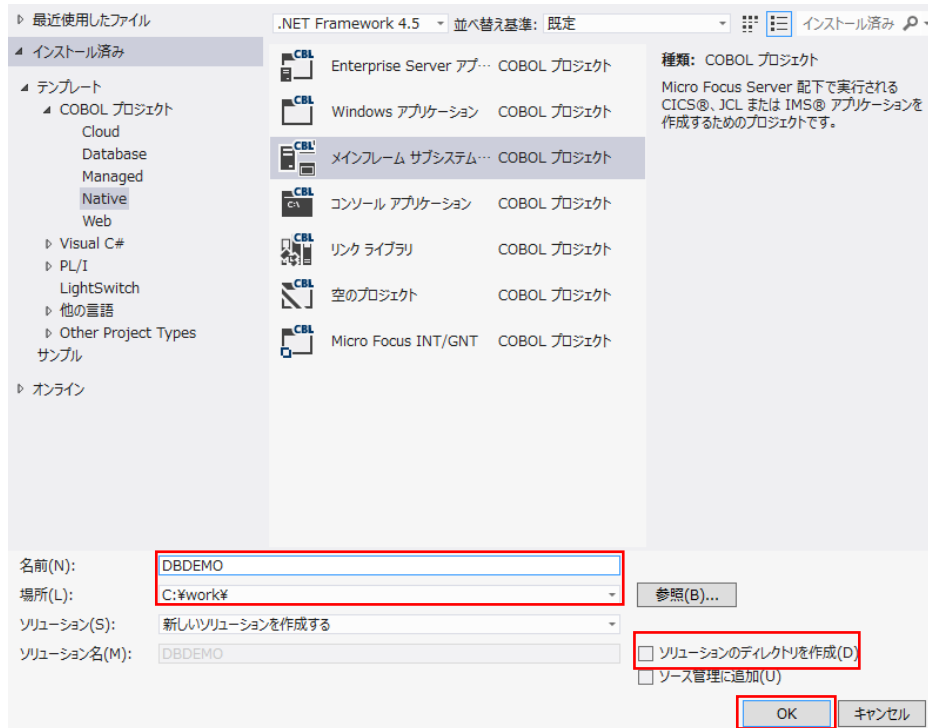
- 2) 新しいプロジェクトを作成します。

プロダクションメニューの [ファイル]> [新規作成]> [プロジェクト] を選択します。



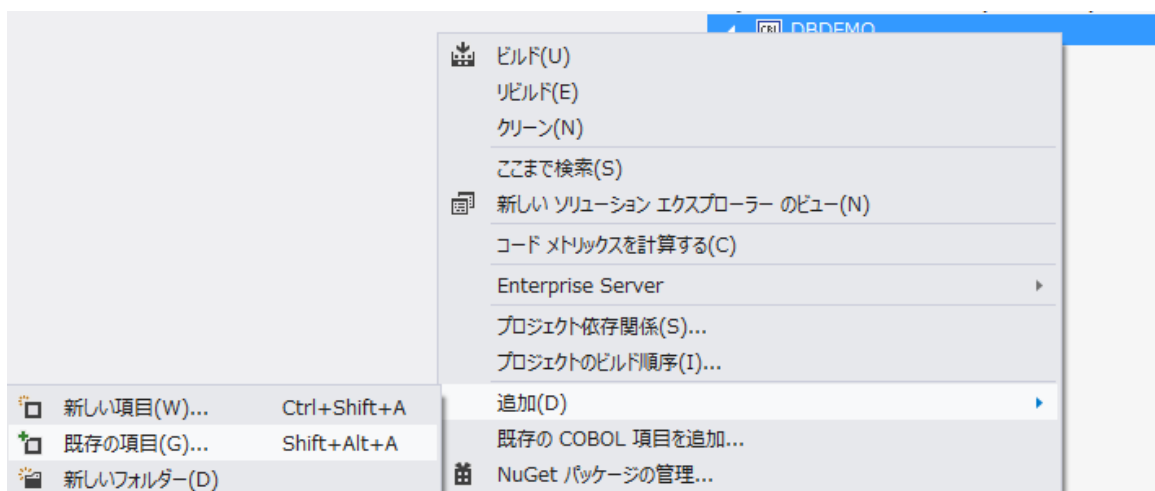
3) プロジェクトの種類と名前、場所を指定します。

1. ペインツリーの [テンプレート]> [COBOL プロジェクト]> [Native] 内から “メインフレーム サブシステム アプリケーション”を選択します。
2. [名前] は “DBDEMO” を入力します。
3. [場所] は 既存 C:\work フォルダを指定します。
4. [ソリューションのディレクトリを作成] のチェックを外して [OK] をクリックします。

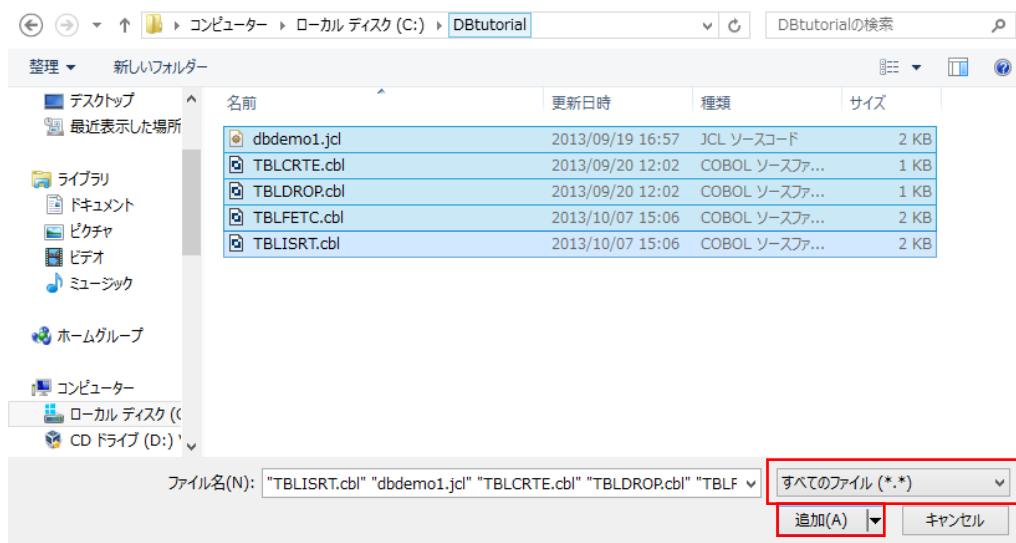


4) プロジェクトへ例題プログラムを追加します。

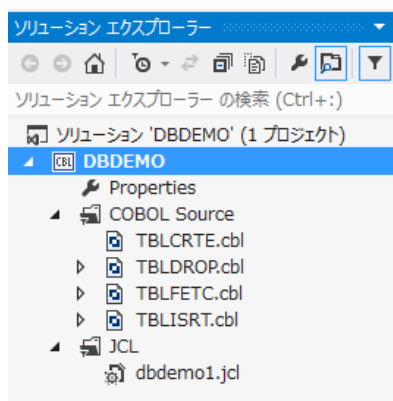
ソリューションエクスプローラでプロジェクトを右クリックし、[追加]> [既存の項目] を選択すると、ファイル選択のためのエクスプローラが表示されます。



- 5) [すべてのファイル]を選択して、展開した C:DBtutorial 直下に存在する全ファイルを選択し [追加] をクリックします。



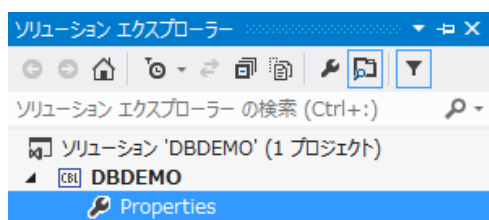
- 6) ソリューションエクスプローラへ追加されたファイルが表示されます。



7.5 プロジェクトと個別ファイルプロパティの設定

- 1) プロジェクトのプロパティを設定します。

- ① ソリューション エクスプローラ内のプロジェクト [Properties] をダブルクリックし、設定ウィンドウを開きます。



- ② ペインツリーの [アプリケーション]> [出力の種類] は [INT/GNT] オブジェクトタイプを選択します。

アプリケーション

構成(C): 該当なし プラットフォーム(M): 該当なし

出力の名前: MSSDEMO

出力の種類: INT/GNT

エンリポイント: (設定なし)

環境(E)...

補足情報
[リンクライブラリ] を指定すると“.DLL”が作成されます。

- ③ ペインツリーの [COBOL] を選択し、以下の項目を設定します。

1. [全般]> [プラットフォームターゲット] は “x64” を選択します。
2. [全般]> [COBOL 方言] は “Enterprise COBOL for z/OS” を選択します。
3. [全般]> [文字セット] は “ASCII” を選択します。
4. [全般]> [.GNT にコンパイル] ヘチェックをして有効にします。
5. [全般]> [デバッグ用にコンパイル] ヘチェックをして有効にします。

アプリケーション

構成(C): アクティブ (Debug) プラットフォーム(M): アクティブ (x86)

全般

プラットフォームターゲット: x64

COBOL 方言: Enterprise COBOL for z/C 文字セット: ASCII

ソースフォーマット: 固定

デバッグ用にコンパイル .GNT にコンパイル

ビルド イベント...

メインフレーム

EXEC CICS オプション... EXEC DLI オプション...

エラーおよび警告

警告レベル: 情報メッセージを含める(レベル I)

最大エラー数: 100 警告をエラーとして処理

出力

出力パス: .%bin%x86%Debug% 参照...

指令ファイルの生成 リストファイルを生成

追加指令

次項参照

高度...

ビルド設定

```
dialect"ENTCOBOL" anim cobidy".%bin%x86%Debug%" warnings"3" max-error"100" charset"ASCII"
```



重要事項

「プラットフォーム ターゲット = XA リソース = Enterprise Server の動作モード = DB」 のビット数が一致することが正常に稼働するために必要な条件です。

6. [追加指令] へは使用するデータベース製品に合わせ、埋め込み SQL 対応のプリプロセッサの設定を指定します。

<Oracle の例>

COBSQL プリプロセッサを使用しますので、以下の指令を追加します。

入力値) P(COBSQL) ENDP

追加指令:

P(COBSQL) ENDP

<DB2 の例>

DB2 ECM プリプロセッサを使用しますので、以下の指令を追加します。“DB=”に続く名称は DB 2 エイリアス名です。

入力値) DB2(DB=SAMPLE,VALIDATE=RUN)

追加指令:

DB2(DB=SAMPLE,VALIDATE=RUN)

<SQL Server の例>

OPENESQL プリプロセッサを使用しますので、以下の指令を追加します。

ODBC を指定するため、使用する ODBC データソースの追加が必要です。

入力値) SQL(DBMAN=ODBC,BEHAVIOR=JCL,TARGETDB=MSSQLSERVER)

追加指令:

SQL(DBMAN=ODBC,BEHAVIOR=JCL,TARGETDB=MSSQLSERVER)

- ④ プロパティ設定ファイルを上書き保存します。



4) Oracle を使用する場合

- ① XA スイッチモジュールを build コマンドで作成します。

“build < ora11 または ora12 >” を実行します。ここでは 11g を使用するため下記コマンドを実行します。
コマンド) build ora11

```
管理者: Enterprise Developer コマンドプロンプト (64-bit)
C:\Users\tarot\Documents>cd c:\%xa
c:\%xa>build ora11
Building 64-bit switch module...
Micro Focus COBOL
Version 2.2.02170 Copyright (C) Micro Focus 1984-2014. All rights reserved.

* Cobsq1 Integrated Preprocessor
* CSQL-I-018: Oracle プリコンパイラトランスレータを起動します。
* CSQL-I-020: Oracle プリコンパイラの出力を処理中。
* CSQL-I-001: COBSQL: チェッカへの引き渡しを完了しました。
* チェック終了: エラーはありません- コード生成を開始します
* Generating ESORAXA
* Data:      8704      Code:      34336      Literals:      2224
Micro Focus COBOL - CBLLINK utility
Version 2.2.2.44 Copyright (C) Micro Focus 1984-2014. All rights reserved.

Microsoft (R) Incremental Linker Version 10.00.40219.01
Copyright (C) Microsoft Corporation. All rights reserved.

ESORAXA.obj
cbll1ds00001130.obj
  Creating library ESORA11XA.lib and object ESORA11XA.exp
Microsoft (R) Manifest Tool version 5.2.3790.2076
Copyright (c) Microsoft Corporation 2005.
All rights reserved.

c:\%xa>
```

5) DB2 を使用する場合

- ① XA スイッチモジュールを下記 build コマンドで作成します。

コマンド) build db2



```
管理者: Enterprise Developer 2012 コマンドプロンプト (64-bit)
C:¥Users¥tarot¥Documents>cd c:¥xa
c:¥xa>build db2
Building 64-bit switch module...
Micro Focus COBOL - CBLLINK utility
Version 2.2.0.93 Copyright (C) Micro Focus 1984-2013. All rights reserved.

Micro Focus COBOL
Version 2.2.00151 Copyright (C) Micro Focus 1984-2013. All rights reserved.
* チェック終了: エラーはありません- コード生成を開始します
* Generating ESDB2XA
* Data:      3784      Code:      7424      Literals:      992
Microsoft (R) Incremental Linker Version 11.00.50727.1
Copyright (C) Microsoft Corporation. All rights reserved.

ESDB2XA.obj
cbllids000011B4.obj
Creating library ESDB2XA.lib and object ESDB2XA.exp
Microsoft (R) Manifest Tool version 6.2.9200.16384
Copyright (c) Microsoft Corporation 2012.
All rights reserved.

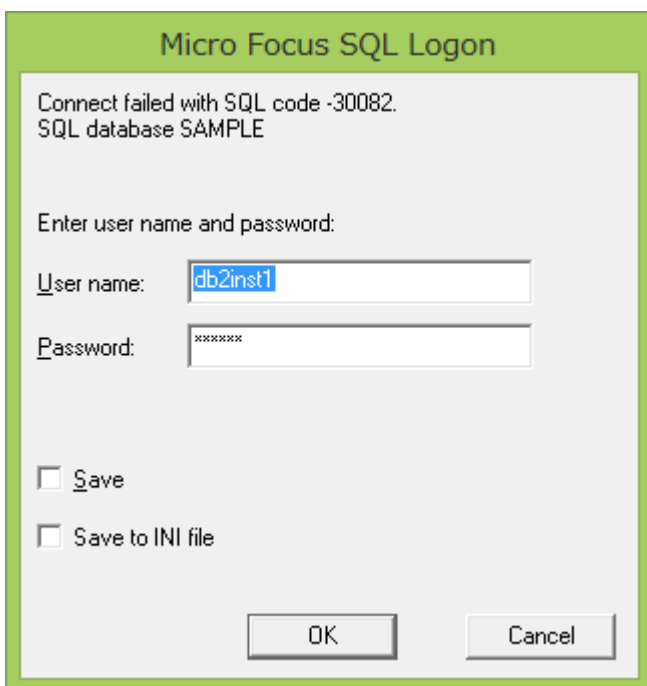
c:¥xa>
```

- ② build db21pc < DB のエイリアス > を実行します。ここでは下記コマンドを実行します。

コマンド) build db21pc sample

DB2 へアクセスするためのログイン画面が表示されますので、ユーザ ID とパスワードを入力します。

“build db21pc sample ユーザ ID パスワード”を実行するとログイン画面は出力されません。



```
管理者: Enterprise Developer 2012 コマンドプロンプト (64-bit)
C:\>build db21pc sample
Building 64-bit switch module...
Micro Focus COBOL
Version 2.2.02166 Copyright (C) Micro Focus 1984-2014. All rights reserved.
* チェック終了: エラーはありません- コード生成を開始します
* Generating ESDB20PC
* Data:          5992      Code:          11078      Literals:          976
Micro Focus COBOL - CBLLINK utility
Version 2.2.2.39 Copyright (C) Micro Focus 1984-2014. All rights reserved.

Microsoft (R) Incremental Linker Version 11.00.50727.1
Copyright (C) Microsoft Corporation. All rights reserved.

ESDB20PC.obj
cbllids00017C70.obj
  Creating library ESDB20PC.lib and object ESDB20PC.exp
Microsoft (R) Manifest Tool version 6.2.9200.16384
Copyright (c) Microsoft Corporation 2012.
All rights reserved.

If you have upgraded your installation from an earlier version of Net
Express or Studio, then you may need to additionally rebuild the main
switch module, that is, execute :

    build db2

Unable to locate the COBOL bin64 directory.

If you intend to execute JES-initiated transactions under Enterprise Server
you must copy ESDB20PC.DLL to a directory on your PATH, such as your COBOL
bin64 directory.

If you do not do so, then such transactions will not be able to communicate
with the database server.

C:\>
```

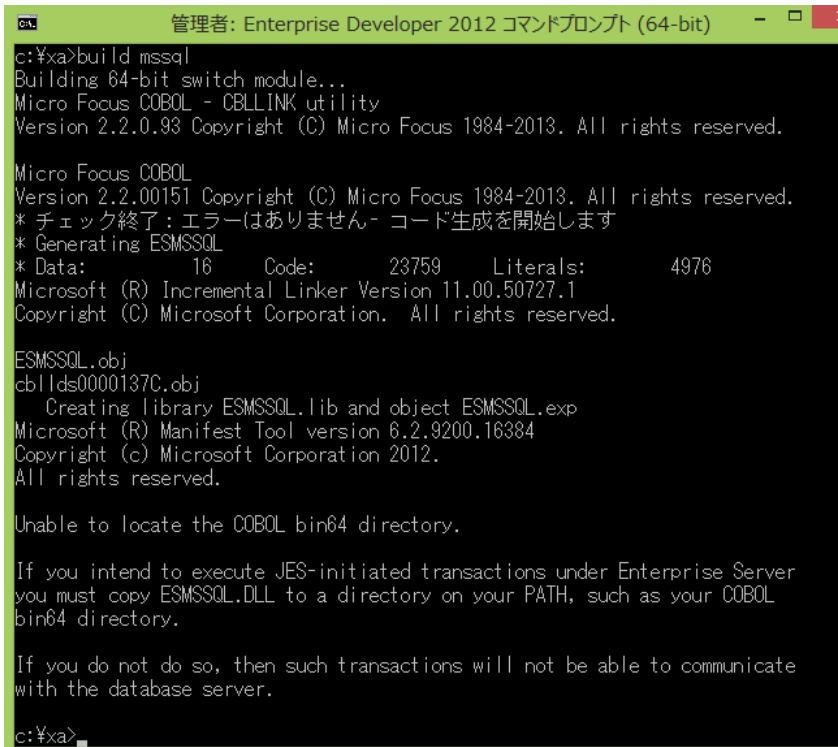
正常に作成されました。メッセージにあるように “ESDB20PC.DLL” ファイルを製品の bin64 フォルダへコピーします。32 ビットの場合は bin フォルダへコピーします。

例) C:\Program Files (x86)\Micro Focus\Enterprise Developer\bin64

6) SQL SERVER を使用する場合

- ① XA スイッチモジュールを下記 build コマンドで作成します。

コマンド) build mssql



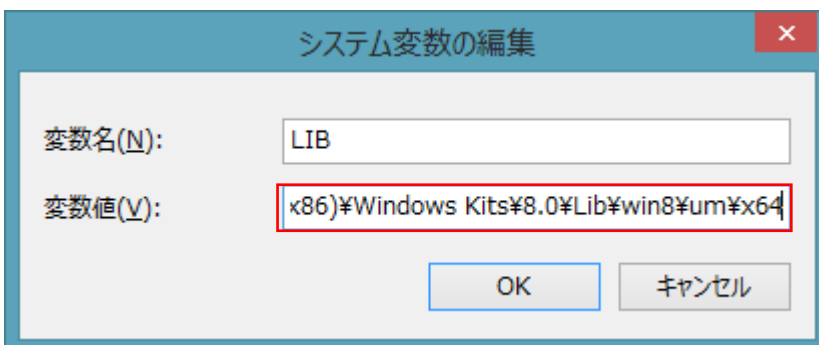
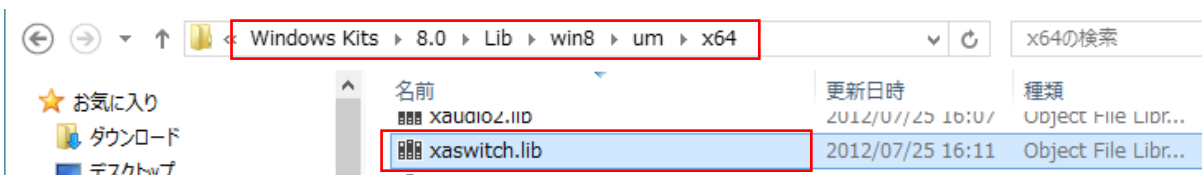
正常に作成されました。メッセージにあるように “ESMSSQL.DLL” ファイルを製品の bin64 フォルダへコピーします。32 ビットの場合は bin フォルダへコピーします。

例) C:\Program Files (x86)\Micro Focus\Enterprise Developer\bin64

xaswitch.lib の読み込み失敗でリンクエラーとなる場合は、環境変数「LIB」へ左記ファイルパスを追加し、コマンドプロンプトを再起動後に再ビルド成してください。

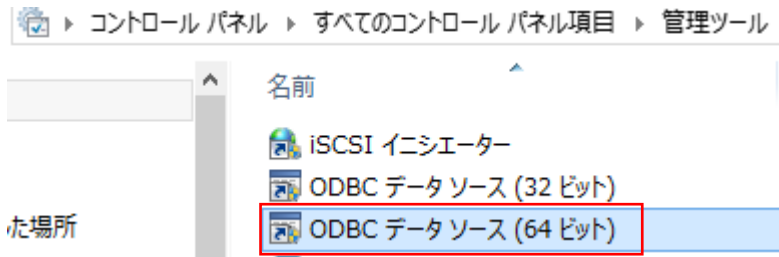
32 ビット例 : C:\Program Files (x86)\Windows Kits\8.0\Lib\win8\um\x86

64 ビット例 : C:\Program Files (x86)\Windows Kits\8.0\Lib\win8\um\x64

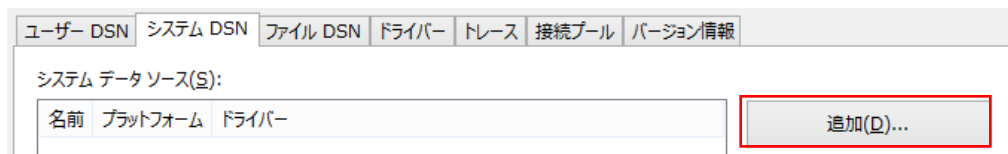


② 使用するビット数に合わせた ODBC データソースを追加します。

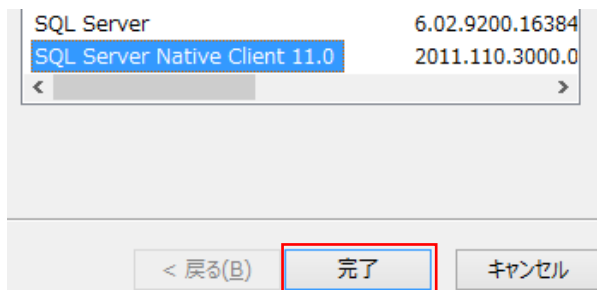
1. [コントロールパネル]>[管理ツール]>[ODBC データソース(64 ビット)]を選択します。



2. [システム DSN]タブを選択して[追加]ボタンをクリックします。

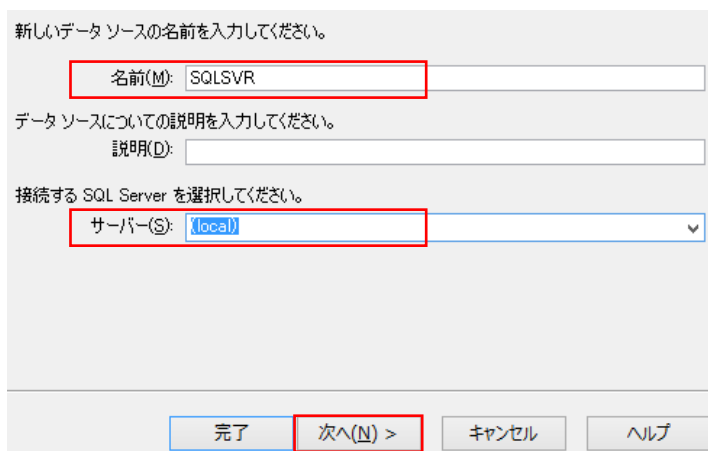


3. [SQL Server Native Client 11.0]を選択して[完了]ボタンをクリックします。



4. [名前]、[サーバー]を入力して[次へ]ボタンをクリックします。

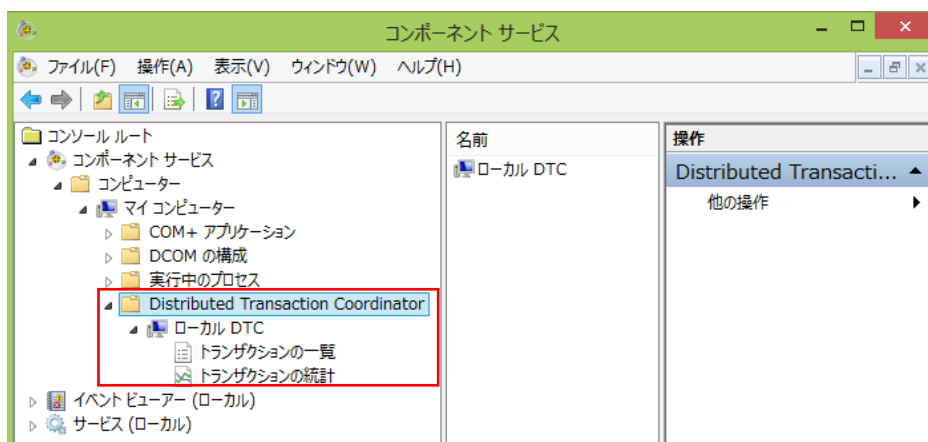
- ・名前 : Enterprise Server の XA リソース定義で使用する DSN 名称です。
- ・サーバー : DB が存在する場所を選択します。このチュートリアルではローカルに DB が存在しています。



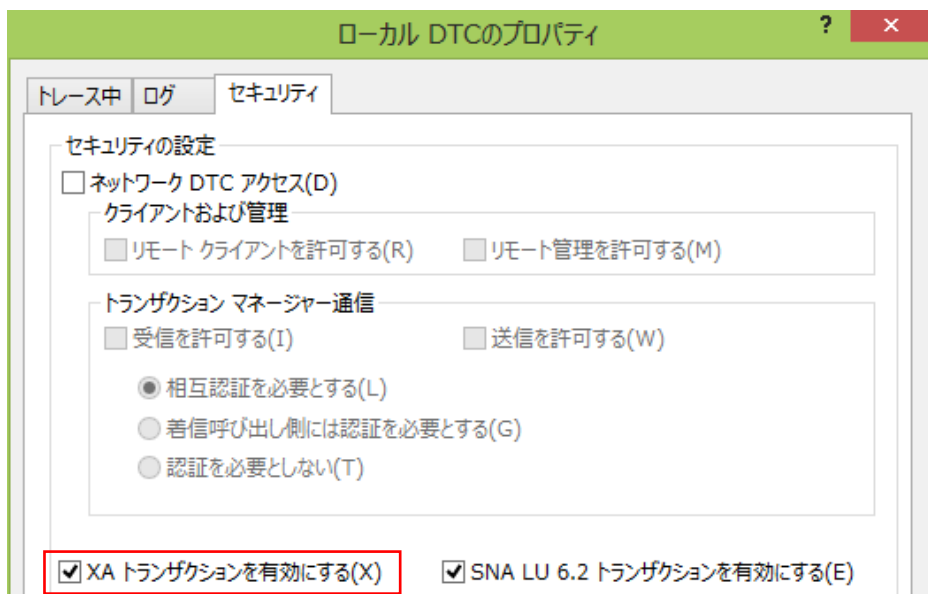
5. [認証方法]、[対象データベース]などを入力後、最終ウィンドウで[データソースのテスト]ボタンをクリックして接続を確認してください。

③ XA スイッチモジュールを有効にするため下記設定を確認します。

1. [コントロールパネル]>[管理ツール]>[コンポーネントサービス]>[コンピューター]>[マイコンピュータ]>[Distributed Transaction Coordinator]>[ローカル DTC]まで展開します。

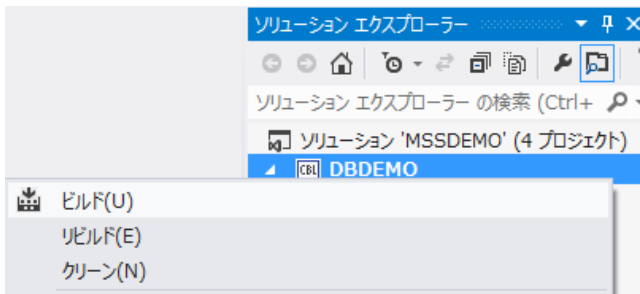


2. [ローカル DTC]を右クリックして[プロパティ]を選択し、[セキュリティ]タブへ移動します。
[XA トランザクションを有効にする]にチェックされていることを確認、もしくはチェックして有効にします。

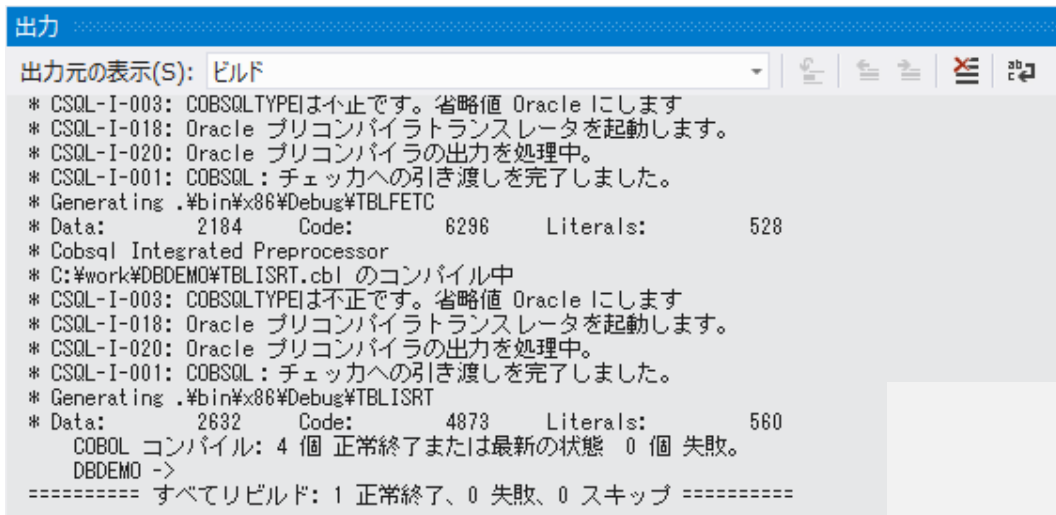


7.7 プログラムのコンパイル

- 1) ソリューションエクスプローラでプロジェクトを選択後、右クリックで[ビルド]を選択します。



- 2) 出力ウィンドウでビルドが正常に完了したことを確認します。



7.8 Enterprise Server の設定

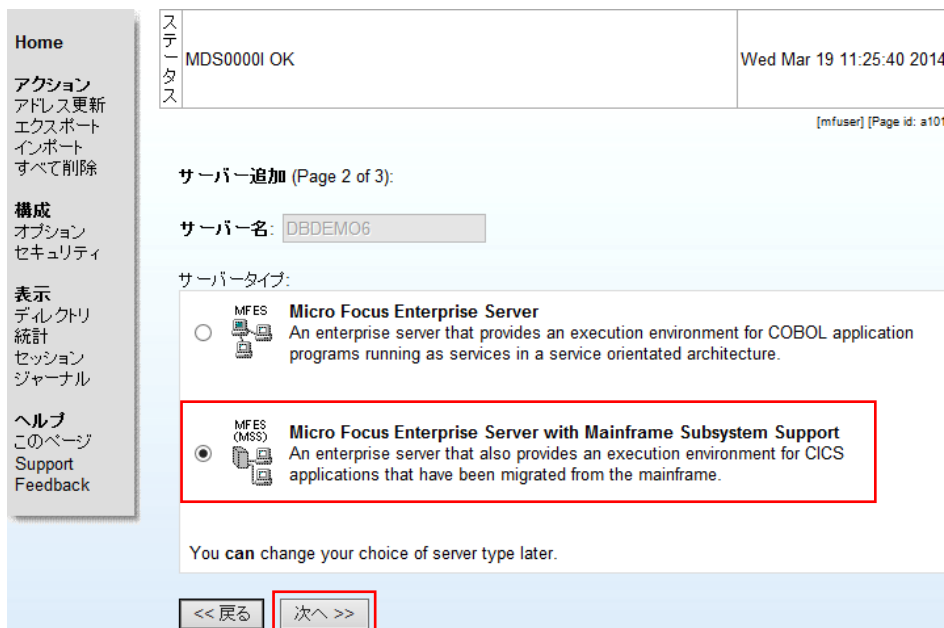
本チュートリアルでは例題 JCL を Enterprise Server にサブミットして実行します。64 ビットオブジェクトを作成しているため、64 ビットの Server を準備します。

- 1) Enterprise Server Administration ウィンドウで [追加] ボタンをクリックし下記を設定します。(1/3 ページ)
 - ① [サーバー名] には “DBDEMO64” を入力します。
 - ② [動作モード] では “64-bit” を選択します。
 - ③ [次へ] ボタンをクリックします。



The screenshot shows the 'Enterprise Server Administration' interface. The top navigation bar includes the Micro Focus logo, the title 'Enterprise Server Administration > サーバー追加', and the version 'Version 1.16.42 Enterprise Server (MSS LICENSED)'. The user is logged in as 'mfuser' with ID 'a001'. The main content area is titled 'サーバー追加 (Page 1 of 3)'. It features a form with the following elements: a 'サーバー名' field containing 'DBDEMO64', a '動作モード' section with radio buttons for '32-bit' and '64-bit' (the latter is selected), and a warning message: 'You cannot change your choice of working mode once a server is created, although you can change it when copying or importing a server.' At the bottom, there are 'キャンセル' and '次へ >>' buttons. A sidebar on the left contains navigation links for Home, Actions, Configuration, Display, and Help.

- 2) [サーバータイプ] では [Micro Focus Enterprise Server with Mainframe Subsystem Support] を選択し、[次へ] ボタンをクリックします。(2/3 ページ)



The screenshot shows the 'Enterprise Server Administration' interface on Page 2 of 3. The top navigation bar is identical to the previous page. The main content area is titled 'サーバー追加 (Page 2 of 3)'. It features a form with the following elements: a 'サーバー名' field containing 'DBDEMO6', a 'サーバータイプ' section with two radio button options: 'Micro Focus Enterprise Server' (unselected) and 'Micro Focus Enterprise Server with Mainframe Subsystem Support' (selected). Below the options is a warning message: 'You can change your choice of server type later.' At the bottom, there are '<< 戻る' and '次へ >>' buttons. The sidebar on the left is identical to the previous page.

- 3) [64-Bit Working Mode] がチェックされていることを確認し、[TN3270 リスナーの作成] のチェックをはずします。[追加] ボタンをクリックします。(3/3 ページ)

| | | | |
|------------------------------|-------------------------------------|-----------------------------|-------------------------------------|
| 共有メモリページ数: | <input type="text" value="512"/> | サービス実行プロセス: | <input type="text" value="2"/> |
| 共有メモリクッション: | <input type="text" value="32"/> | トレーステーブルサイズ: | <input type="text" value="341"/> |
| ローカルトレースサイズ: | <input type="text" value="341"/> | 診断ファイル最大サイズ: | <input type="text" value="0"/> |
| 要求ライセンス: | <input type="text" value="10"/> | | |
| ワールドスタート診断ファイル: | <input checked="" type="checkbox"/> | システムアベンド時ダンプ: | <input checked="" type="checkbox"/> |
| 補助トレースアクティブ: | <input type="checkbox"/> | ローカルコンソールを表示: | <input type="checkbox"/> |
| Mainframe Subsystem Support: | <input checked="" type="checkbox"/> | 64-Bit Working Mode: | <input checked="" type="checkbox"/> |

トレースフラグ:

| | | | | | |
|--------------|--------------------------|----------|--------------------------|------------|--------------------------|
| タスク管理 | <input type="checkbox"/> | ストレージ管理 | <input type="checkbox"/> | テーブル管理 | <input type="checkbox"/> |
| アプリケーションコンテナ | <input type="checkbox"/> | 要求ハンドラ | <input type="checkbox"/> | RMインターフェース | <input type="checkbox"/> |
| 通信 | <input type="checkbox"/> | アプリケーション | <input type="checkbox"/> | 終了 | <input type="checkbox"/> |

生成オプション:

TN3270リスナーの作成 using port

- 4) 管理ウィンドウから [編集] ボタンをクリックし、[XA リソース] タブを開き、[追加] ボタンをクリックします。

The screenshot shows a management window with the following elements:

- Buttons:** 編集... (Edit), 開始... (Start), 停止 (Stop), 詳細 (Details), 追加... (Add).
- Resource List:** MFES (MSS), DBDEMO64, 1 top:*.*, 2 リスナー (Listener).
- Server Configuration:** Server: [], CP 1: [], OK button.
- Summary:** サーバー... (Servers), リスナー (2) (Listeners), サービス (4) (Services), ハンドラ (3) (Handlers), パッケージ (0) (Packages).
- Properties:** プロパティ... (Properties), 構成 (Configuration), 診断... (Diagnosis), 過去の統計 (Past Statistics).
- General Tab:** 一般 (General), **XAリソース (0)** (XA Resources), MSS... (✓), MQ..., スクリプト (Scripts), アクセス権 (Access Rights), セキュリティ (Security).
- Action:** **追加...** (Add) button is highlighted with a red box.

5) 使用するデータベース製品に合わせた XA リソースの下記項目を設定します。

- ① [ID]: IKJEFT ユーティリティに渡す DSN TSO コマンドの SYSTEM パラメタで渡す DB2 サブシステム名を指定します。(このチュートリアルでは XADB とします)
- ② [名前]: Oracle の場合は "ORACLE_XA" を指定し、他はリソース名として任意の名前を指定します。
- ③ [モジュール]: 前項で作成した XA スイッチメインモジュールのパスと DLL ファイル名を指定します。(xxx1pc.dll は指定無し)

<<DB に依存した当チュートリアルで指定する値の例>>

- **Oracle**
"C:¥xa¥ESORA11XA.DLL"
- **DB2**
"C:¥xa¥ESDB2XA.DLL"
- **SQL Server**
"C:¥xa¥ESMSSQL.DLL"

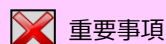
- ④ [OPEN 文字列]: DB オープンのために必要な値を指定します。(設定済みの DB 環境設定に依存)

1. Oracle の例

例えば以下のように設定し、[追加] ボタンをクリックします。名前と OPEN 文字の先頭は必ず "ORACLE_XA" を指定してください。

| | | | | | |
|----|-------------------|------------|-------|-------|-------|
| 一般 | XAリソース (3) | MSS... (✓) | MQ... | スクリプト | アク... |
|----|-------------------|------------|-------|-------|-------|

| |
|---|
| ID: XADB |
| 名前: ORACLE_XA |
| モジュール: C:\xa\ESORA11XA.DLL |
| OPEN文字列: ORACLE_XA+SesTm=100+SqlNet=tok-par+Acc=P/scott/tiger |
| CLOSE文字列: |
| 説明: |
| 有効: <input checked="" type="checkbox"/> |



重要事項

パスは全て半角英数字で指定してください。
以降も同様です。

追加

キャンセル

2. DB2 の例

例えば以下のように設定し、[追加] ボタンをクリックします。

| | | | | | |
|----|-------------------|------------|-------|-------|----|
| 一般 | XAリソース (0) | MSS... (✓) | MQ... | スクリプト | アク |
|----|-------------------|------------|-------|-------|----|

ID:
XADB

名前:
DB2_XA

モジュール:
C:\xa\ESDB2XA.DLL

OPEN文字列:
DB=SAMPLE,uid=db2inst1,pwd=ibmdb2,AXLIB=casaxlib

CLOSE文字列:

説明:

有効:

追加 **キャンセル**

3. SQL Server の例

例えば以下のように設定し、[追加] ボタンをクリックします。

前項で作成した ODBC データソース名称を OPEN 文字列に指定します。

- DSN=SQLSVR

| | | | | | |
|----|-------------------|------------|-------|-------|----|
| 一般 | XAリソース (3) | MSS... (✓) | MQ... | スクリプト | アク |
|----|-------------------|------------|-------|-------|----|

ID:
XADB

名前:
SQLSVR_XA

モジュール:
C:\xa\ESMSSQL.DLL

OPEN文字列:
DSN=SQLSVR

CLOSE文字列:

説明:

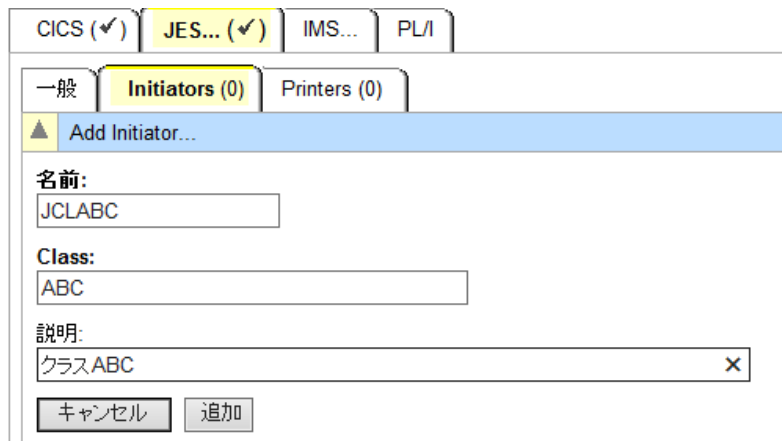
有効:

追加 **キャンセル**

- 6) JCL チュートリアルで実行したように、JCL を実行するために [JES] の設定を行い [Apply] ボタンをクリックします。



- 7) [JES]>[Initiators] を作成し、実行する JCL のクラスが含まれるよう入力後、[追加] ボタンをクリックします。



- 8) サーバーにプロジェクトを関連付けます。
 9) サーバー エクスプローラから [DBDEMO64] を開始します。



7.9 XA リソース正常ロードの確認

XA スイッチモジュールが Enterprise Server に正しくロードできると、Enterprise Server Administration のコンソールログには以下のメッセージが出力されます。下記は SQL Server の例です。

Server DBDEMO64 [開始 ✓]

サーバー... リスナー (2) サービス (4) ハンドラ (3) パッケージ (0)

プロパティ... 構成 診断... 過去の統計

トレース ダンプ ESコンソール CSコンソール

画面更新 Show entries from 1 to 10 of 39 total entries
 Show last 10 lines

| Entry | Event |
|-------|--|
| 30 | 140319 13200222 17692 DBDEMO64 CASSI1426I TD non-recoverable cold started 13:20:02 |
| 31 | 140319 13200243 17692 DBDEMO64 CASSI1426I TD recoverable warm started 13:20:02 |
| 32 | 140319 13200264 17692 DBDEMO64 CASSI1600I SEP initialization completed successfully 13:20:02 |
| 33 | 140319 13200285 17692 DBDEMO64 CASX00020I XADB XA interface loaded. Name(Micro Focus SQL Server XA struct), Registration Mode(Static) 13:20:02 |
| 34 | 140319 13200488 17692 DBDEMO64 CASX00015I XADB XA interface initialized successfully 13:20:03 |
| 35 | 140319 13200509 17692 DBDEMO64 CASSI5001I PLTPI Phase 1 - No PLT Specified 13:20:04 |
| 36 | 140319 13200530 17692 DBDEMO64 CASSI5040I Active SEP memory strategy set to x'00000001', retain count 100 13:20:05 |
| 37 | 140319 13200571 5184 DBDEMO64 CASSI1600I SEP initialization completed successfully 13:20:05 |
| 38 | 140319 13200572 15116 DBDEMO64 CASSI1600I SEP initialization completed successfully 13:20:05 |
| 39 | 140319 13200875 5184 DBDEMO64 CA |

注意事項

ロードに失敗してもサーバーは開始されますので、ログ内容を確認してください。

7.10 JCL の実行と確認

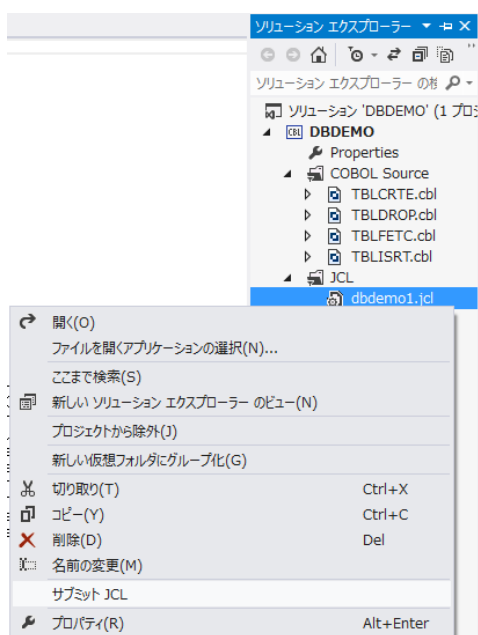
- 1) ソリューション エクスプローラから dbdemo1.jcl をダブルクリックしてエディタを開き、内容を確認します。

```
//DBDEMO1 JOB CLASS=A,MSGCLASS=A
*****
//STEP01 EXEC PGM=IKJEFT01
//SYSOUT DD SYSOUT=*
//SYSPRINT DD SYSOUT=*
//SYSTSIN DD *
DSN SYSTEM(XADB)
  RUN PROGRAM(TBLCRTE) PLAN(SAMPLES)
END
/*
//SYSTSPT DD SYSOUT=*
*****
//STEP02 EXEC PGM=IKJEFT01
//SYSOUT DD SYSOUT=*
//SYSPRINT DD SYSOUT=*
//SYSTSIN DD *
DSN SYSTEM(XADB)
  RUN PROGRAM(TBLISRT) PLAN(SAMPLES)
END
/*
//SYSDIN DD *
00001Soseki Natsume      1-1,Koishikawa,Bunkyo-ku,Tokyo-to      1886
00002Ryotaro Shiba      2-3,Sonezaki,Kita-ku,Osaka-shi,Osaka-fu  1900
00003Hideyo Noguchi     5-1,Inawashiro,Aizu-shi,Fukushima-ken  1911
00004Osamu Dazai        2-6,Tsugaru,Tsugaru-gun,Aomori-ken     1911
00005Eiji Yoshikawa     9-3,Miyatomomura,Mimasaka-gun,Okayama-ken  1920
00006Jirocho Shimizu    6-6,Jiro-cho,Shimizu-shi,Shizuoka-ken    1800
00007Ogai Mori          3-1,Rintaro-cho,Tsuwano-shi,Shimane-ken  1886
00008Ryoma Sakamoto     1-1,Harimayabashi,Kochi-shi,Kochi-ken   1820
00009Shiki Masaoka     5-5,Dogo Onsen,Matsuyama-shi,Ehime-ken  1870
00010Yukichi Fukuzawa  8-8,Keio-cho,Nakatsu-shi,Oita-ken      1835
/*
//SYSTSPT DD SYSOUT=*
*****
//STEP03 EXEC PGM=IKJEFT01
//SYSOUT DD SYSOUT=*
//SYSPRINT DD SYSOUT=*
//SYSTSIN DD *
DSN SYSTEM(XADB)
  RUN PROGRAM(TBLFETC) PLAN(SAMPLES)
END
/*
//SYSTSPT DD SYSOUT=*
*****
//STEP04 EXEC PGM=IKJEFT01
//SYSOUT DD SYSOUT=*
//SYSPRINT DD SYSOUT=*
//SYSTSIN DD *
DSN SYSTEM(XADB)
```

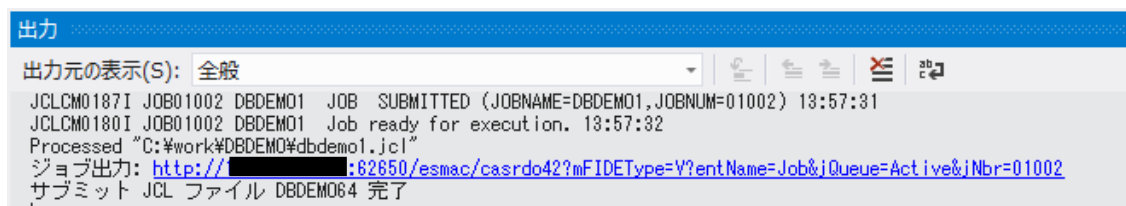
このジョブは4つのステップから構成され下記の様に連携されています。

- STEP01:
DB にテーブルを新規作成
- STEP02:
JCL に記述された SYSIN データをテーブルに追加
- STEP03:
テーブルからデータを全件読み出し SYSOUT へ出力
- STEP04:
テーブルを削除

- 2) ソリューション エクスプローラから dbdemo1.jcl を右クリックして [サブミット JCL] を選択します。



- 3) 以下のように DBDEMO1 ジョブが実行された旨のメッセージが出力ウィンドウへ表示されます。



- 4) ハイパーリンクをクリックすると自動的にスプールが表示されますので、以降は JCL チュートリアルを参考にして、スプールビューの Out Hold キューおよび Output キューを開き、実行された DBDEMO1 ジョブの結果を確認したり、バッチプログラムのデバッグも同様に実行してみてください。また、各 DB 付属の管理ツールを使用すると、各ステップにおける DB テーブルの状況が確認できます。

以上