

---

# Micro Focus Enterprise Developer チュートリアル

---

## メインフレーム COBOL 開発 : IMS

### Visual Studio 2019 編

#### 4. 目的

本チュートリアルでは、Visual Studio 2019 を使用したメインフレーム COBOL プロジェクトの作成、コンパイル、IMS を使用したトランザクションの実行、デバッグまでを行い、その手順の習得を目的としています。

#### 5. 前提

- 本チュートリアルで使用したマシン OS : Windows 10 Enterprise
- 使用マシンに Microsoft Visual Studio 2019 がインストールされていること
- 使用マシンに Micro Focus Enterprise Developer 6.0 for Visual Studio 2019 がインストールされていること
- 使用マシンに TN3270 エミュレータがインストールされており、稼働実績があること

#### 6. チュートリアル手順の概要

1. チュートリアルの準備
2. Visual Studio の起動
3. メインフレーム COBOL プロジェクトの作成
4. プロジェクトプロパティの設定
5. IMS データベースのロードおよびトランザクションリスト設定
6. 環境変数の設定
7. ビルドの実行
8. Enterprise Server インスタンスの設定
9. Enterprise Server インスタンスの開始と確認
10. IMS リソースの確認
11. IMS トランザクションの実行
12. プログラムの動的デバッグ
13. Enterprise Server インスタンスの停止

### 3.1 チュートリアル準備

例題プログラムに関連する資源を用意します。

- 1) 使用する例題プログラムは、キットに添付されている IMStutorial.zip に圧縮されています。これを C:¥ 直下に解凍します。



- 2) Visual Studio のソリューションを保存する VS フォルダを C:¥ 直下に作成します。

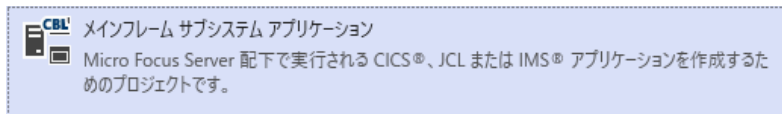
### 3.2 Visual Studio の起動

- 1) Visual Studio 2019 を起動します。



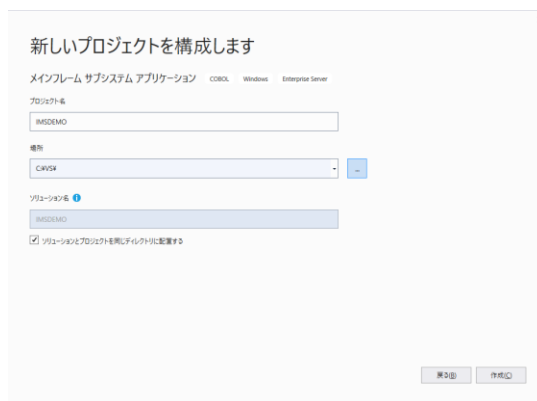
### 3.3 メインフレーム サブシステム アプリケーション プロジェクトの作成

- 1) 新しいソリューションとプロジェクトを作成します。[ファイル] プルダウンメニューから [新規作成] > [プロジェクト] を選択して [新しいプロジェクト] ウィンドウを表示し、[メインフレーム サブシステム アプリケーション] を選択後、[次へ] ボタンをクリックします。

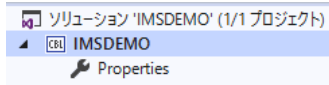


- 2) 下記項目を入力後、[作成] ボタンをクリックします。

項目名	説明
プロジェクト名	任意ですが、ここでは IMSDEMO を入力します。
場所	前項で作成した C:¥VS を指定します。
ソリューションとプロジェクトを同じディレクトリに配置する	ここではチェックをオンにします。

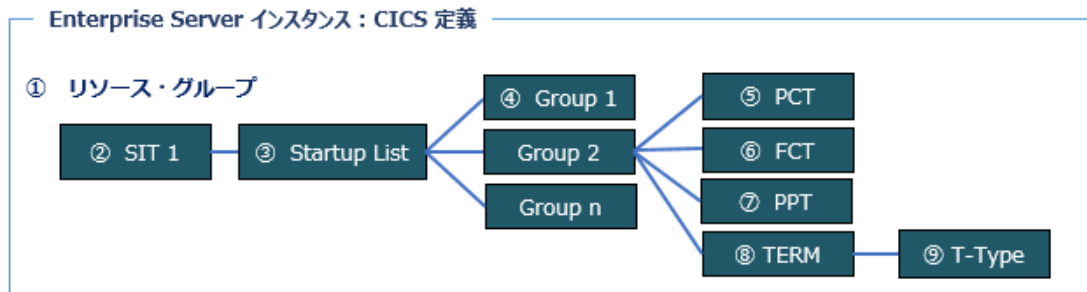


- 3) [ソリューション エクスプローラー] へ作成したプロジェクトが表示されます。

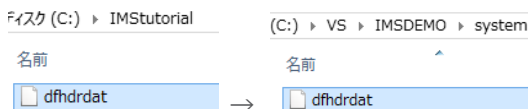


- 4) プロジェクトを作成したことにより C:\¥VS¥IMSDEMO フォルダが作成されていますので、このフォルダ直下に C:\¥VS¥IMSDEMO¥system フォルダを作成します。

Enterprise Server インスタンスの Mainframe Subsystem Support (MSS) はリソースと呼ぶ CICS アプリケーションに使用するソフトウェアやハードウェアの項目を定義、制御、および監視するための機能を備えています。IMS アプリケーションに必要なリソースはこの機能に含まれるよう拡張された経緯から、IMS においても CICS と同様にリソース定義ファイルを配置します。リソース構成の詳細については [CICS SIT 構築チュートリアル] をご参照ください。

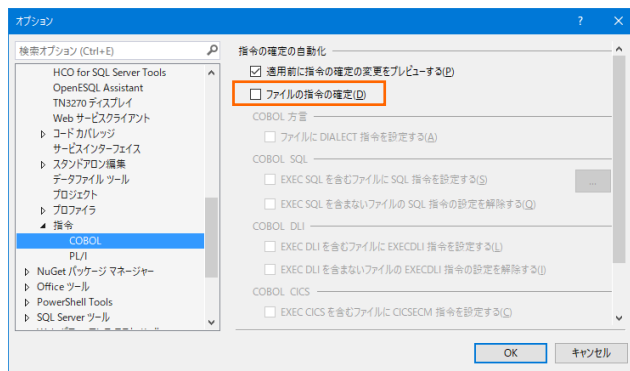


前項で作成した C:\¥IMStutorial フォルダにある dfhdrdat ファイルを C:\¥VS¥ IMSDEMO¥system へコピーします。このファイルには IMS で使用するリソース定義が含まれています。

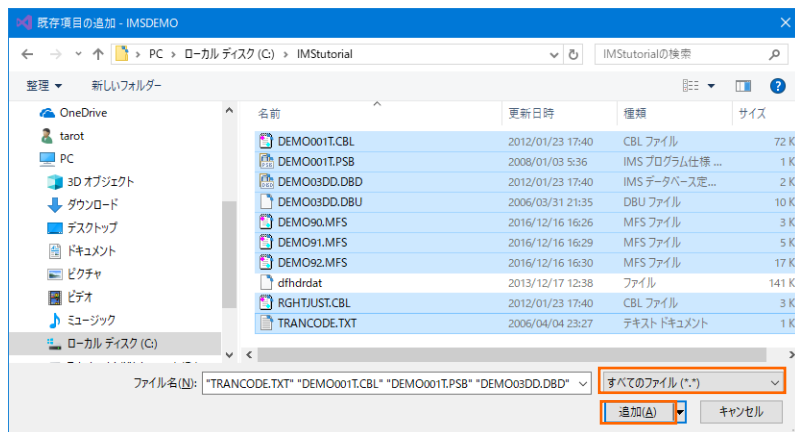


- 5) 既存ファイルのインポート時、自動的にコンパイル指令が指定される機能が用意されていますが、本チュートリアルではこれを解除します。[ツール] プロダクションメニューの [オプション] を選択してオプションウィンドウを表示します。

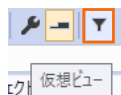
左側ツリービューの [Micro Focus Tools] > [指令] > [COBOL] > [ファイルの指令の確定] チェックボックスをオフにして [OK] ボタンをクリックします。



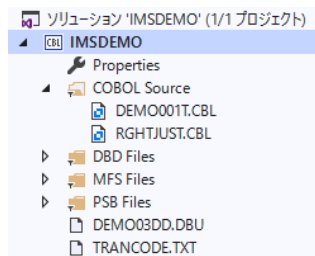
- 6) 用意した例題プログラム類をインポートします。IMSDEMO プロジェクトを右クリックして [追加] > [既存の項目] を選択し、既存項目の追加ウィンドウにて C:\¥IMStutorial を指定すると内容が表示されますので、[すべてのファイル] を対象にして、dfhdrdat ファイル以外を選択後 [追加] ボタンをクリックします。この実行により、プロジェクトフォルダーへ例題プログラムが配置されます。



- 7) 種類別に表示するため、[ソリューション エクスプローラー] 内の [仮想ビュー] アイコンをクリックします。



- 8) [ソリューション エクスプローラー] 内に表示されている IMSDEMO プロジェクトにインポートしたファイルが表示されていることを確認します。

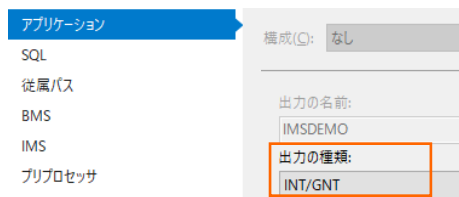


### 3.4 プロジェクトプロパティの設定

この例題には COBOL プログラム, DBD , MFS , PSB ファイルが含まれており、プログラム内容に沿った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 エクスプローラー] 内の [Properties] をダブルクリックしてプロパティウィンドウを表示します。

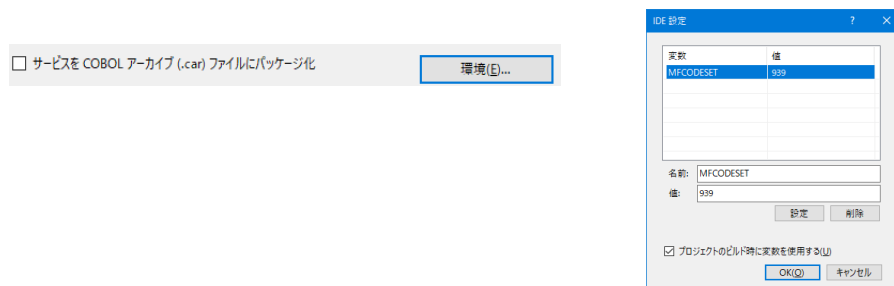
- 2) 左側ツリービューの [アプリケーション] を選択して、生成する実行ファイルを GNT にするため [出力の種類] へ [INT/GNT] を選択します。



**情報**

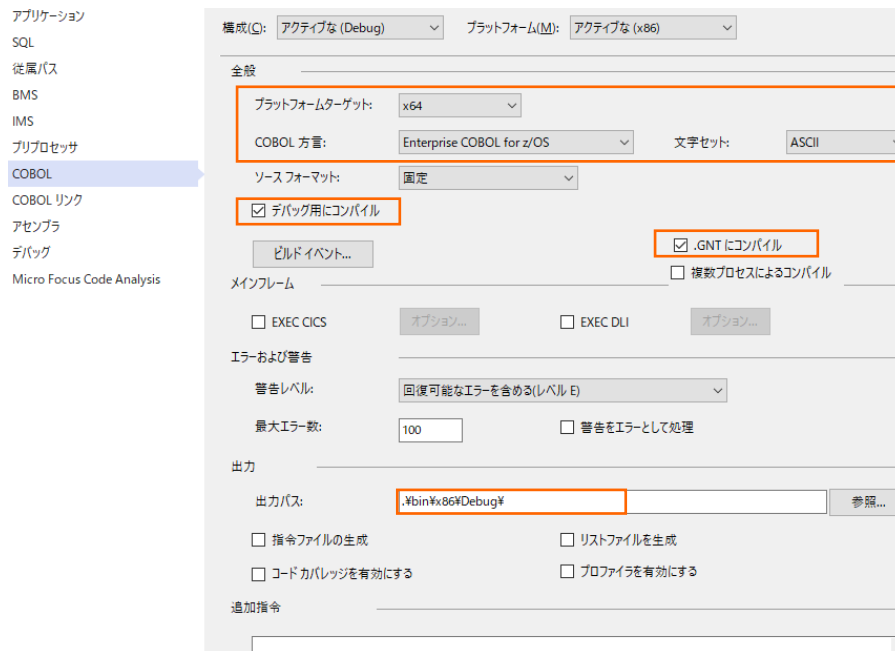
GNT は Micro Focus 独自のオブジェクトで、Micro Focus COBOL ランタイム環境下で実行可能となります。

- 3) 同じ画面にある [環境] ボタンをクリックして環境変数設定ウィンドウを表示します。MFCODESET 環境変数に日本語ラテン拡張を指定する 939 を設定、[プロジェクトのビルド時に変数を使用する] のチェックをオンにして [OK] ボタンをクリックします。

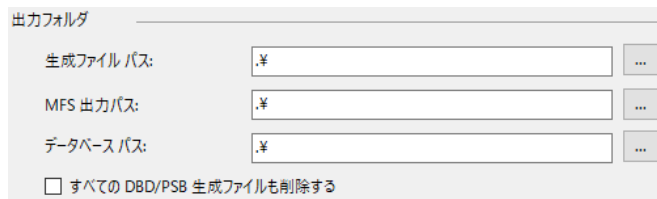


- 4) 左側ツリービューの [COBOL] を選択して、下記項目を入力します。

項目名	説明
プラットフォーム ターゲット	稼働ビット数を指定します。ここでは [x64] を指定します。
COBOL 方言	COBOL 言語方言を指定します。 例題プログラムは IBM Enterprise COBOL の方言を使用しているため、ここでは [Enterprise COBOL for z/OS] を指定します。
文字セット	EBCDIC または ASCII を指定します。ここでは [ASCII] を選択します。
デバッグ用にコンパイル	デバッグ実行時に使用するファイルを生成するように指定します。
.GNT にコンパイル	実行ファイル形式を GNT に指定するためにチェックをオンにします。
出力パス	実行ファイルが出力されるパスを指します。任意に指定可能です。
追加指令	ここでは何も指定しません。



- 5) 左側ツリービューの [IMS] を選択して、IMS 資産をビルドした際に生成されるファイルの出力パスを確認します。デフォルトはソリューションが存在するパスになっていますが、任意に変更することも可能です。



- 6) 左側ツリービューの [IMS] を選択して表示されている [DBD ビルド設定] の [DBD のマップを生成] チェックをオンにします。



- 7) 左側ツリービューの [IMS] を選択して表示されている [MFS ビルド設定] の [追加指令] へ CBEGCS を入力します。



これは MFS 定義へ EATTR=EGCS と EATTR=MIX が指定してある場合に追加する指令です。

MFS 定義のダブルバイト文字指定は下記の点に注意してください。

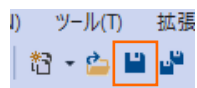
- ① EGCS = 固定文字の場合は 'G'あいう' と指定し、前後のシフトイン、シフトアウト文字は排除します。
- ② MIX = 固定文字の場合は '<A>DD -追加' と指定し、ダブルバイト文字前後のシフトイン、シフトアウト文字は排除します。

詳細に関しては製品ヘルプをご参照ください。

- 8) 左側ツリービューの [デバッグ] を選択して、[アクティブ設定] へ [IMS] を選択します。



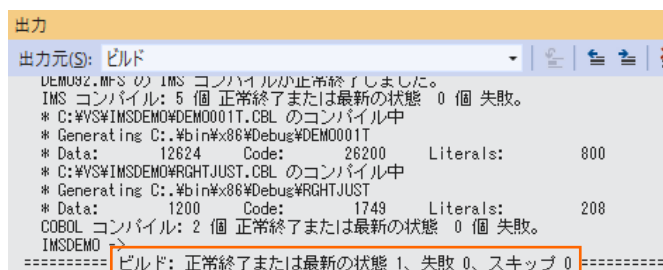
- 9) プロパティファイルを上書き保存します。



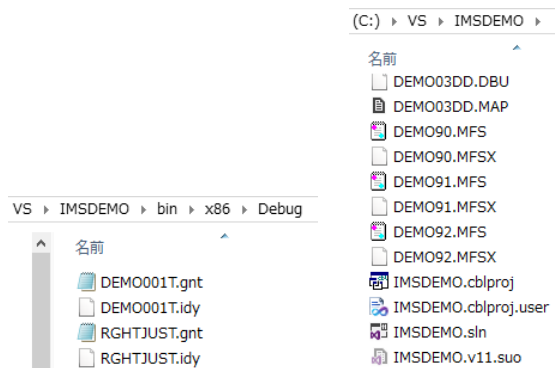
### 3.5 ビルドの実行

- 1) [ソリューション エクスプローラー] の IMSDEMO ソリューションを右クリックして [ソリューションのビルド] を選択すると、コンパイル指定に沿ったビルドが実行されます。

- 2) [出力] ウィンドウで成功を確認します。



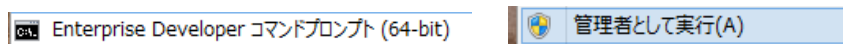
- 3) 前項で確認した出力パスへ実行ファイルに指定した実行ファイル (.gnt ファイル) が作成され、ACB, DIF, DOF, MAP, MFSX, MID, MOD や DBDGEN2.DAT, DBDGEN2F.DAT, PSBGEN3.DAT などのマクロからシステム生成された IMS 関連生成ファイルはソリューションと同じフォルダーへ出力されています。



### 3.6 IMS データベースのロードおよびトランザクションリスト設定

MFIMS コマンドを使用して、IMS データベースのロードと IMS トランザクションリストの設定を行います。

- 1) Windows のプログラムメニューから [Micro Focus Enterprise Developer] > [ツール] > [Enterprise Developer コマンドプロンプト (64-bit)] を右クリックして [管理者として実行] を選択します。



- 2) 作成した COBOL プロジェクトのパスへ移動します。

```
C:\Users\tarot\Documents>cd C:\%VS%\IMSDEMO
C:\%VS%\IMSDEMO>
```

- 3) 例題に含まれている IMS データベースをロードするために、下記のコマンドを実行して成功を確認します。




コマンド) MFIMS IMSDBU LOAD DEMO03DD NOCLS

```
C:\%VS%\IMSDEMO>MFIMS IMSDBU LOAD DEMO03DD NOCLS
```

↓

```
03:44 DEMO03DD LOAD started
03:44 DEMO03DD LOAD completed successfully
```

コマンドにより下記ファイルが作成されます。

-  DEMO03DD.DAT
-  DEMO03DD.LST
-  IMSDBU.LOG

コマンドの詳細に関しては製品ヘルプをご参照ください。



- 例題に含まれている TRANCODE.TXT ファイルは、メインフレームから抽出した Stage1 ファイルを Enterprise Server インスタンスの IMS 機能がサポートするファイル形式に stage1ext コマンドを使用して変換したものです。



TRANCODE.TXT ファイルから IMS トランザクションのリストを設定するために、次のコマンドを実行します。

コマンド) MFIMS STAGE1IMP TRANCODE.TXT

```
MPP MFDEMO DEM0001TDEM0001T0000100000000000NYSSR001NPU
MPP TESTMENUTEST001TTEST001T0000101000000000NYSSR001NPU
MPP TESTMAINTEST002TTEST002T0000101000000000NYSSR001NPU
```

```
C:\#VSY#IMSDEMO>MFIMS STAGE1IMP TRANCODE.TXT
Importing transactions
Number of definitions inserted:
BMP : 00000.
QBMP : 00000.
MPP : 00003.
NRMP : 00000.
LTERM: 00000.
Classes used:
Class: 1 used by: 3 transactions
```

コマンドにより下記ファイルが作成されます。

 IMSGEN2.DAT  
 STG1IMPLST

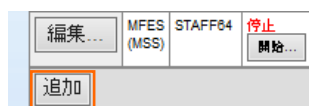
### 3.7 Enterprise Server インスタンスの設定

Enterprise Server インスタンスには CICS や IMS をエミュレーションする機能が搭載されており、この開発用インスタンスを使用してメインフレームアプリケーションのテスト実行やデバッグを行います。本番環境には実行環境製品である Enterprise Server をインストールし、本番用インスタンス上でアプリケーションを稼働させます。

- Enterprise Server インスタンスを作成します。[サーバー エクスプローラー] タブの [Micro Focus Server] を右クリックして [管理] を選択します。



- Enterprise Server Administration では Enterprise Server インスタンス一覧が表示されますので、画面の左下にある [追加] ボタンをクリックします。



- 3) サーバー名には IMSDEMO を入力、動作モードは 64-bit を指定して [次へ] ボタンをクリックします。

サーバーの追加 (ページ 1 / 3):

サーバー名:

作業モード:

32ビット  64ビット

サーバーの作成後に、その作業モードの変更は可能です。



### 重要

実行ファイル生成に指定した稼働ビット数 = Enterprise Server インスタンス稼働ビット数である必要があります。

- 4) 画面の Page 2/3 では、CICS や JCL を実行可能な機能を持つ [Micro Focus Enterprise Server メインフレーム サブシステム サポート対応] が選択されていることを確認後、[次へ] ボタンをクリックします。

サーバータイプ:

**Micro Focus Enterprise Server**  
サービス指向アーキテクチャーのサービスとして実行される COBOL アプリケーション プログラムの実行環境として動作するエンタープライズ サーバーです。

**Micro Focus Enterprise Server - メインフレーム サブシステム サポート対応**  
メインフレーム 技術を使用する、移行されたアプリケーションの実行環境としても動作するエンタープライズ サーバーです。

- 5) Page 3/3 では [TN3270 リスナーの作成] のチェックがオンであることを確認して、[使用ポート] へ 5039 を指定します。[追加] ボタンをクリックすると、IMSDEMO という名前の 64 ビットアプリケーション稼働用 Enterprise Server インスタンスが追加されます。

生成オプション:

TN3270リスナーの作成  使用ポート:

→  MFES (MSS) IMSDEMO

64



### 情報

ポート番号には 1025 より小さい番号を指定しないでください。  
指定したポート番号で TN3270 リスナーが作成されます。

- 6) 左にある [編集] ボタンをクリックします。

MFES (MSS) IMSDEMO

64

7) [サーバー] > [プロパティ] > [一般] タブ内の下記項目を設定します。

- ① [動的デバッグを許可] チェックボックスをオンにします。この指定により、Visual Studio からの動的デバッグが可能になります。

開始オプション:

共有メモリーブページ数:	512	サービス実行プロセス:	2
共有メモリックッション:	32	要求ライセンス:	10
ローカルコンソールを表示:	<input type="checkbox"/>	動的デバッグを許可:	<input checked="" type="checkbox"/>
Start on System Start:	<input type="checkbox"/>	64-Bit Working Mode:	<input checked="" type="checkbox"/>
以前のログを削除:	<input type="checkbox"/>	コンソールログサイズ (K):	0

- ② [構成情報] 欄にパス変数と、国別コードを有効にするため下記内容を入力します。

[ES-Environment]

IMSPROJ=C:\¥VS¥IMSDEMO

MFCODESET=939

構成情報

[ES-Environment]  
IMSPROJ=C:\¥VS¥IMSDEMO  
MFCODESET=939

**重要**

入力値は全て半角英数字で指定してください。

**注意**

MFCODESET ヘビルド時に指定した環境変数と同じ値を指定します。

- ③ [適用] ボタンをクリックします。

8) [サーバー] > [プロパティ] > [MSS] > [CICS] タブで表示される画面の各項目を設定します。入力後は [Apply] ボタンをクリックします。

項目名	説明
メインフレーム サブシステム サポート有効	[MSS] タブ配下の設定をオン、オフ指定します。ここではオンを指定します。
システム初期化テーブル (SIT)	IMS リージョン設定の詳細が提供されるシステム初期化テーブルを指定します。ここでは例題に含まれている IMSSIT を指定します。
トランザクションパス	実行プログラムの探索パスを指定します。ここでは指定しません。
File Path	データセットのデフォルトパスを指定します。ここでは指定しません。
マップパス	コンパイル済み BMS マップセットのパスを指定します。ここでは指定しません。
リソース定義ファイルパス	リソース定義ファイルが存在するパスを指定します。ここでは前項で作成した \$IMSPROJ¥system フォルダを指定します。

メインフレーム サブシステム サポート有効:

CICS (✓) JES... IMS... PL/I

CICS 有効:

システム初期化テーブル (SIT):

トランザクションパス:

File Path:

マップパス:

リソース定義ファイルパス:

構成情報

**重要**

入力値は全て半角英数字で指定してください。  
 これらのフィールドでは改行を入れないように注意してください。

- 9) [サーバー] > [プロパティ] > [MSS] > [IMS] > [一般] タブで表示される画面の各項目を設定します。入力後は [Apply] ボタンをクリックします。

項目名	説明
IMS 有効	[IMS] タブ配下の設定をオン、オフ指定します。ここではオンにします。
ACB ファイル ディレクトリ	ACB ファイルが存在するパスを指定します。ここでは \$IMSPROJ を指定します。
Gen ファイル ディレクトリ	GEN ファイルが存在するパスを指定します。ここでは \$IMSPROJ を指定します。

一般 XAリソース (0) MSS... (✓) MC

メインフレーム サブシステム サポート有効:

CICS (✓) JES... IMS... PL/I

一般 DB TM...

IMS 有効:

デフォルト コードセット: ASCII

ACBファイル ディレクトリ:

Genファイル ディレクトリ:

(C:) > VS > IMSDEMO > (C:) > VS > IMSDEMO >

名前	名前
<input type="checkbox"/> DEMO001T.ACB	<input type="checkbox"/> DBDGEN2.DAT
<input type="checkbox"/> DEMO03DD.ACB	<input type="checkbox"/> DBDGEN2F.DAT

- 10) [サーバー] > [プロパティ] > [MSS] > [IMS] > [DB] タブで表示される画面の各項目を設定します。入力後は [Apply] ボタンをクリックします。

項目名	説明
データベース パス	IMS データベースが存在するパスを指定します。ここでは \$IMSPROJ を指定します。

CICS (✓) JES... **IMS... (✓)**

一般 **DB** TM...

一般 圧縮 ロック 復旧

データベース パス:  
\$IMSPROJ

- 11) [サーバー] > [プロパティ] > [MSS] > [IMS] > [TM] タブで表示される画面の各項目を設定します。入力後は [Apply] ボタンをクリックします。

項目名	説明
MFS パス	MFS ファイルが存在するパスを指定します。ここでは \$IMSPROJ を指定します。
アプリケーションパス	ビルドした実行ファイルが存在するパスを指定します。 ここでは \$IMSPROJ¥bin¥x86¥Debug を指定します。

一般 MPR (0)

メッセージキュー:  
デフォルト キュー名を使用:   
名前: \_\_\_\_\_  
最大ブロック数:   
キューバッファ数:  コールドスタート: \_\_\_\_\_

MFS パス:  
\$IMSPROJ

MFS 属性定義:

MFS null 文字:  
 文字:   16進: 0x

キーポイント生成頻度:  末尾スレ  
トランザクションしきい値:  (秒)

アプリケーションパス:  
\$IMSPROJ¥bin¥x86¥Debug

- 12) [サーバー] > [プロパティ] > [MSS] > [IMS] > [TM] > [MPR] タブへ表示されている [追加] ボタンをクリックして、メッセージ処理領域 (MPR) を作成します。IMS アプリケーションを実行するためには最少 1 つは必要です。

追加画面にて下記項目を入力後、[追加] ボタンをクリックします。

項目名	説明
名前	任意ですが、ここでは MPR01 を指定します。
トランザクションクラス	実行するトランザクションクラスを指定します。ここでは 001 を指定します。
説明	任意ですが、ここでは MPR for class1 を指定します。

一般 **MPR (0)**

▲ メッセージ処理領域の追加...

名前: MPR01

トランザクション クラス: 001

説明:  
MRP for class1

キャンセル 追加

一般 **MPRs (1)**

	名前	クラス	説明
編集...	MPR01	001	MPR for class1

**情報**

MPR とは :

IMS TM アプリケーションを実行する際に必要となる、特殊なサービス実行プロセス (SEP) を指します。

- 13) 画面左上の [Home] をクリックして一覧画面に戻ります。



### 3.8 Enterprise Server インスタンスの開始と確認

- 1) [サーバー エクスプローラー] 内に IMSDEMO インスタンスが表示されていることを確認します。表示されていない場合は [Micro Focus Servers] を右クリックし、[最新の情報に更新] を選択してリフレッシュしてください。

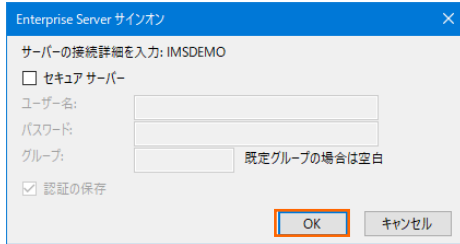
- 2) [サーバー エクスプローラー] 内の IMSDEMO インスタンスを右クリックし、[プロジェクトと関連付ける] > [IMSDEMO] を選択します。これにより IMSDEMO プロジェクトから実行されるアプリケーションは IMSDEMO インスタンスで処理されることになります。



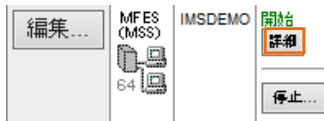
- 3) IMSDEMO インスタンスを右クリックして [開始] を選択します。



- 4) 下記ウィンドウが表示された場合は、ここではユーザーによる制限を行わないため [OK] ボタンをクリックします。

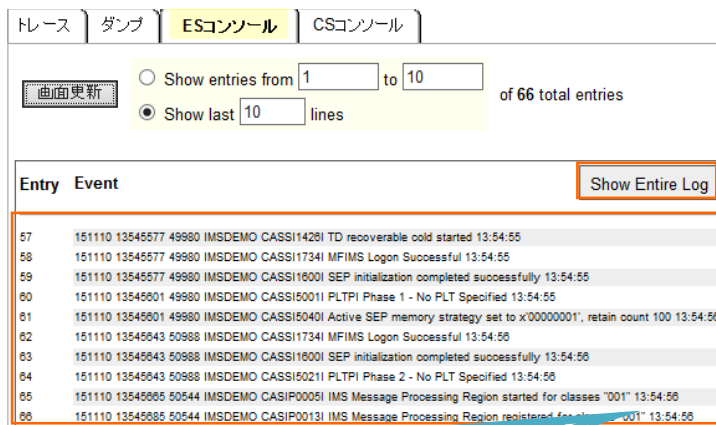


- 5) Enterprise Server Administration 画面へ移動して開始状態であることを確認後、[詳細] ボタンをクリックします。



- 6) [サーバー] > [診断] > [ES コンソール] で IMSDEMO インスタンスのコンソールログをリアルタイムにチェックすることができます。また [Show Entire Log] をクリックしてログ全体を表示させることも可能です。

正常に開始されたことを確認します。



【IMS 機能の正常開始ログ抜粋】

```

CASTM5001I IMS TM Control Process initialization started 11:07:20
CASCD1059I IMS MPR Process created for server IMSDEMO, process-id = 5628 11:07:21
CASCS5100I Communications Process instance 01 is ready to accept requests 11:07:21
CASIP0023I IMS Message Processing Region MPRO1 started: class(es) "001" 11:07:21
CASDB5001I IMS DB Control Process initialization started 11:07:21
CASTS0002I ES TSC Service Process initialization complete 11:07:21
CASIP0006I IMS Message Processing Region initialization started 11:07:21
IMSRTO099I IMS DB Control Build Tag ED_4_0_0_2_20180711 11:07:22
CASCD1071I Administration SEP created for Server IMSDEMO, process-id = 3656 11:07:22
  
```



注意

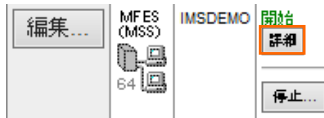
いくつかのサービス開始が失敗してもインスタンスは開始されますので、ログ内容を必ず確認してください。

7) 画面左上の [Home] をクリックして一覧画面に戻ります。

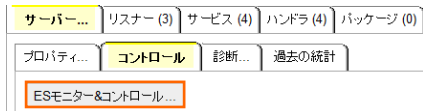
### 3.9 IMS リソースの確認

Enterprise Server インスタンスでは IMS トランザクションを参照・更新・追加・削除することが可能です。また IMS コントロール画面ではコマンドによる状態の確認が可能です。

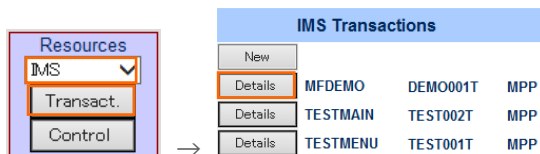
1) Enterprise Server Administration 画面の IMSDEMO インスタンスの開始ステータス直下にある [詳細] ボタンをクリックします。



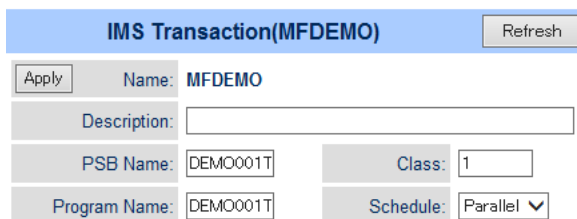
2) [ES モニター & コントロール] ボタンをクリックします。



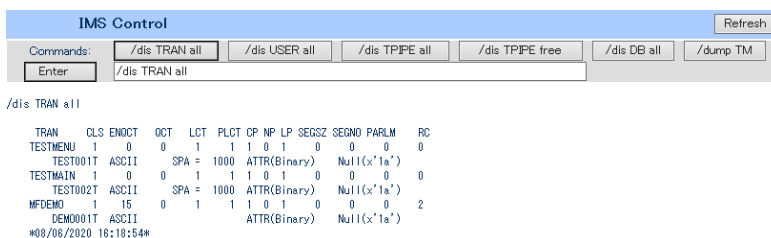
3) 画面左側中央にある [Resources] カテゴリ内のコンボボックスで [IMS] を選択して、[Transact] ボタンをクリックすると一覧が表示されますので、[MFDEMO] の左にある [Details] ボタンをクリックします。



4) [MFDEMO] トランザクションの詳細を表示すると、呼び出すプログラムや関連する PSB ファイルが確認できます。



5) 次に [IMS] の [Control] ボタンをクリックすると IMS コントロール画面が表示されます。ボタンをクリックまたはコマンド入力でトランザクションの状態などが確認できます。





### 3.10 IMS トランザクションの実行

現在 IMSDEMO インスタンスが稼働していますので、例題プログラムを実行することができます。

ご使用の TN3270 エミュレータを、前項で作成した TN3270 リスナーポート (localhost:5039) へ接続します。

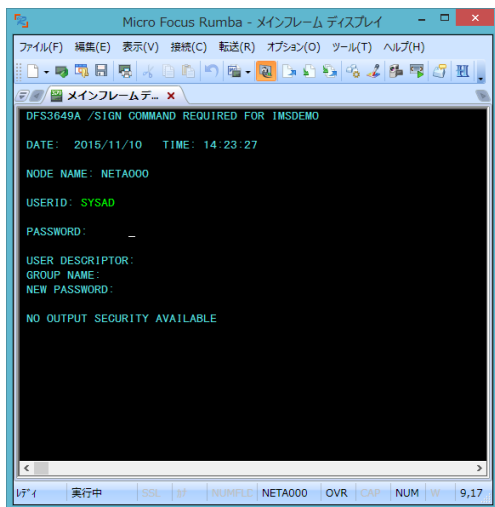
補足) TN3270 エミュレータで、使用しているキーボード設定をご確認ください。

Rumba の例)



- 1) 下記は Micro Focus Rumba を使用した画面です。例題の SIT では初期トランザクションに /IMS が指定されているため、接続後 IMS サインオン画面が表示されます。[USERID] と [PASSWORD] へ SYSAD を入力して実行キーを押します。

CICS SIT - IMSSIT		Refresh
Apply	Name: IMSSIT	
Description:	ES/IMS example SIT	
Startup List:	IMSSTRT	Work Area: 512 (CV)
SysID:	EIMS	Initial Tran ID: /IMS



- 2) 正常にサインオンできましたら、クリアキーで画面をクリアします。
- 3) IMS トランザクションを開始する下記コマンドを入力後、実行キーを押します。

コマンド) /STA TRAN MFDEMO

**/STA TRAN MFDEMO**

- 4) 1) 正常に開始されると下記メッセージが表示されます。

```
DFS058I 13:02:03 START COMMAND COMPLETED
```

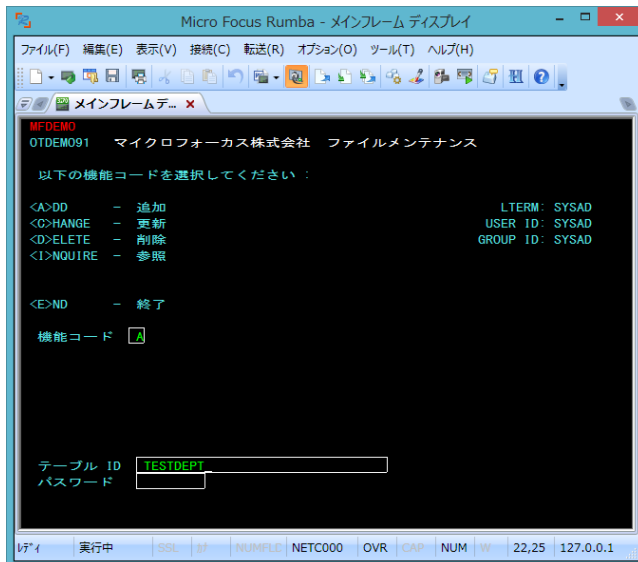
- 5) 再度、クリアキーで画面をクリアします。

- 6) MFDEMO (末尾ブランクあり)を入力して実行キーを押します。

```
MFDEMO _
```

- 7) 下記のように例題の初期画面が表示されます。ここでは TESTDEPT テーブルを追加します。

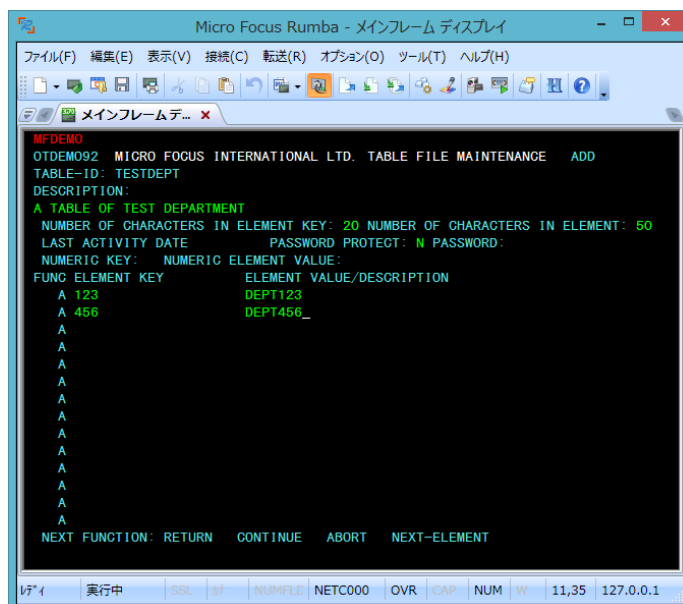
タブキーで項目間を移動して、[機能コード] へは A を [テーブル ID] へは TESTDEPT を入力後、実行キーを押します。



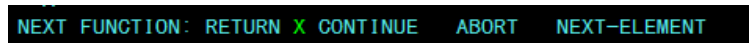
- 8) タブキーで項目間を移動して、TESTDEPT テーブルの要素を入力後、実行キーを押します。

なお、この例題は日本語入力に対応していません。

項目名	説明
DESCRIPTION	説明を入力します。ここでは A TABLE OF TEST DEPARTMENT を指定します。
NUMBER OF CHARACTERS IN ELEMENT KEY	ここでは 20 を指定します。
NUMBER OF CHARACTERS IN ELEMENT	ここでは 50 を指定します。
PASSWORD PROTECT	ここでは N を指定します。
FUNC	A を確認します。
ELEMENT KEY (1 行目)	ここでは 123 を指定します。
ELEMENT VALUE (1 行目)	ここでは DEPT123 を指定します。
ELEMENT KEY (2 行目)	ここでは 456 を指定します。
ELEMENT VALUE (2 行目)	ここでは DEPT456 を指定します。



- 9) 画面下部の [RETURN] までタブキーで移動し、X を入力します。ほかの選択肢に X が入っていないことを確認後、実行キーを押すとメニュー画面へ戻ります。

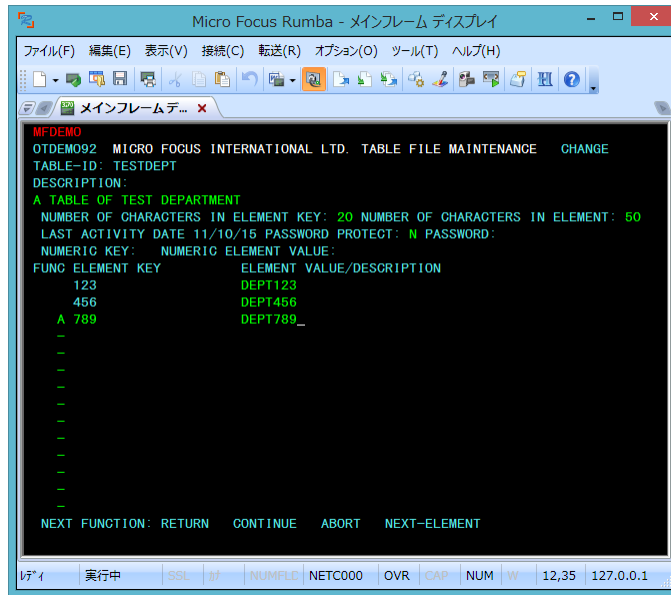


- 10) 作成した TESTDEPT テーブル要素を変更します。[機能コード] へは C を [テーブル ID] へは TESTDEPT を入力後、実行キーを押します。



11) テーブル要素の 3 行目に下記を追加して実行キーを押します。

項目名	説明
FUNC	ここでは A を指定します。
ELEMENT KEY (3 行目)	ここでは 789 を指定します。
ELEMENT VALUE (3 行目)	ここでは DEPT789 を指定します。



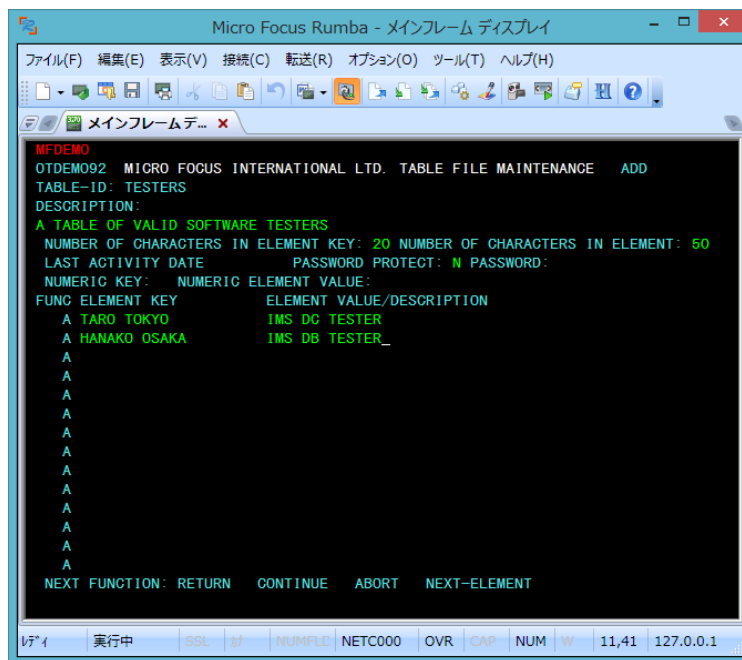
12) 画面下部の [RETURN] までタブキーで移動し、X を入力します。ほかの選択肢に X が入っていないことを確認後、実行キーを押すとメニュー画面へ戻ります。

13) 次に TESTERS テーブルを作成します。[機能コード] へは A を [テーブル ID] へは TESTERS を入力後、実行キーを押します。



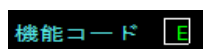
14) タブキーで項目間を移動して、TESTERS テーブルの要素を入力後、実行キーを押します。

項目名	説明
DESCRIPTION	説明を入力します。ここでは A TABLE OF VALID SOFTWARE TESTERS を指定します。
NUMBER OF CHARACTERS IN ELEMENT KEY	ここでは 20 を指定します。
NUMBER OF CHARACTERS IN ELEMENT	ここでは 50 を指定します。
PASSWORD PROTECT	ここでは N を指定します。
FUNC	A を確認します。
ELEMENT KEY (1 行目)	ここでは TARO TOKYO を指定します。
ELEMENT VALUE (1 行目)	ここでは IMS DC TESTER を指定します。
ELEMENT KEY (2 行目)	ここでは HANAKO OSAKA を指定します。
ELEMENT VALUE (2 行目)	ここでは IMS DB TESTER を指定します。

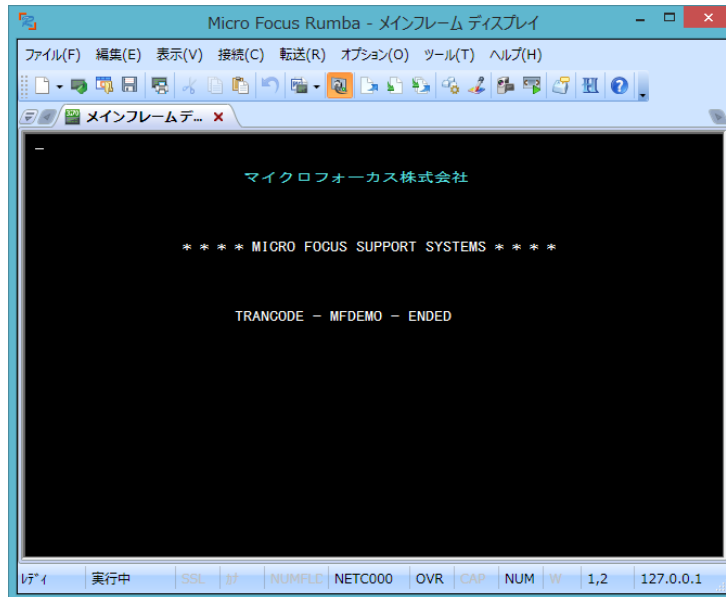


15) 前項と同じ要領でメニュー画面へ戻ります。

16) [機能コード] へ E を入力後、実行キーを押してトランザクションを終了します。



17) 例題プログラムの終了画面が表示されます。



18) TN3270 エミュレータを切断します。

### 3.11 プログラムの動的デバッグ

Visual Studio を使用して、例題プログラムのデバッグを行います。前項で実施しましたが、IMSDEMO インスタンスの [動的デバッグを許可] へのチェックと IMSDEMO インスタンスと Visual Studio プロジェクトの関連付けが必要です。

1) [デバッグ] プルダウンメニューの [ステップ イン] を選択します。

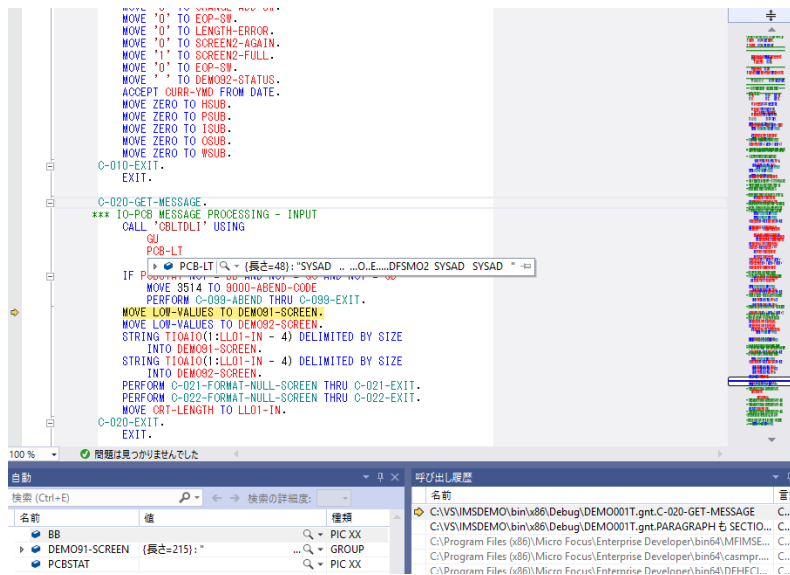
⇧	ステップイン(L)	F11
⇨	ステップオーバー(O)	F10

2) 画面左下がオレンジ色の [準備完了] となり、アタッチ待機状態になったことを確認します。



3) 前項と同様に TN3270 エミュレータから IMS トランザクションを実行すると、プログラムのステップ実行が可能になります。[F11] キーもしくは [デバッグ] プルダウンメニューから [ステップ イン] を選択してステップを進めることができます。

マウスオーバーやウォッチタブを利用して変数の値が確認できます。なお、この例題プログラムでは TN3270 エミュレータとの画面送受信がありますので、その都度、表示を切り替えてデバッグします。



- 4) 希望のステップの左端をクリックすることにより、ブレークポイントを設定することも可能です。



- 5) 先に進める場合は画面上部の [続行] アイコンをクリックします。

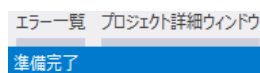


- 6) デバッグを終了させるため、画面上部の [デバッグ停止] アイコンをクリックします。

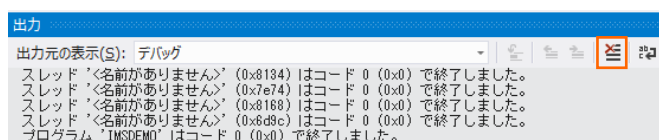


- 7) TN3270 エミュレータを切断します。

- 8) 画面左下が元の色の [準備完了] となり、アタッチ待機状態ではなくなったことを確認します。



- 9) 出力タブに表示されているデバッグ履歴を消去するには、[全てクリア] アイコンをクリックします。

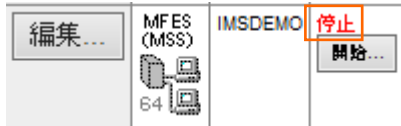


### 3.12 Enterprise Server の停止

- 1) IMSDEMO インスタンスを停止します。



- 2) IMSDEMO インスタンスの停止を確認後、Visual Studio を終了します。



### WHAT'S NEXT

- メインフレーム COBOL 開発 : JCL Visual Studio 2019 編
- メインフレーム COBOL 開発 : CICS Visual Studio 2019 編
- 本チュートリアルで学習した技術の詳細については製品マニュアルをご参照ください。