Micro Focus Visual COBOL チュートリアル

RESTful Web サービスによる COBOL 資産の再利用 Visual Studio 編

1. 目的

Micro Focus Visual COBOL に付属する COBOL 専用のアプリケーションサーバー「Enterprise Server」は、ネイティブにコンパイル した COBOL のビジネスロジックを REST API を利用し Web サービスとして呼び出す機能を提供しています。RESTful の Web サービスと して呼び出しを行う場合、JSON 形式でやり取りが可能であれば呼び出し側のプログラムに依存することなく連携できるようになります。

このドキュメントでは COBOL のソースコードに一切手を加えることなくビジネスロジックとして Enterprise Server にディプロイし、それを Visual COBOL のクライアント生成機能を使って動作確認用のクライアントを作成し連携する方法を説明します。

2. 前提条件

本チュートリアルは、下記の環境を前提に作成されています。

> 開発クライアント ソフトウェア

OS	Windows Server 2016 Standard Edition(64bit)
COBOL 開発環境製品	Micro Focus Visual COBOL 4.0J for Visual Studio (PU5 適用版)

チュートリアル用サンプルプログラム 下記のリンクから事前にチュートリアル用のサンプルファイルをダウンロードして、任意のフォルダに解凍しておいてください。 サンプルプログラムのダウンロード

3. チュートリアル手順の概要

- 1. Windows クライアントでの開発準備作業
- 2. RESTful Web サービスの開発作業
- 3. コンパイルした COBOL アプリケーションを Enterprise Server ヘディプロイ
- 4. RESTful Web サービスのテスト
- 5. インスタンスの停止

3.1 Windows クライアントでの開発準備作業

- 1) Visual COBOL for Visual Studio を起動
 - ① [スタート] メニュー > [Visual Studio XXXX]を選択します。(XXXX はバージョン番号)



- 2) [Enterprise Server アプリケーション] プロジェクトの作成
 - ① [ファイル]メニュー > [新規作成] > [プロジェクト]を選択します。
 - ② [COBOL プロジェクト] > [Native] > [Enterprise Server アプリケーション] を選択します。

新しいプロジェクト	
▶ 最近使用したファイル	並べ替え: 既定 🔹 🔡 듣
▲ インストール済み	Enterprise Server アプリケーション
▲ COBOL プロジェクト	
Managed Database	Windows アプリケーション
Cloud	CBL コンソール アプリケーション
Web Native	リンク ライブラリ

 ③ [名前(N)]フィールドに"RESTfulCOBOL"を入力して、[OK] ボタンをクリックします。[場所(L)]は任意のフォルダを 指定します。

名前(N):	RESTfulCOBOL
場所(L):	C:¥work¥RESTful¥
ソリューション名(M):	RESTfulCOBOL

- 3) コンパイラオプションの指定とソースコードのインポート
 - ① 作成されたプロジェクトの Properties をダブルクリックします。





[COBOL]をクリックし、画面を下にスクロールして、[追加指令]に "ASSIGN(EXTERNAL)" を指定し、画面を閉じます。

			¥
構成(C): アクティブ (Debug)	 ブラットフォーム(M): アクティブ (x86) 		
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		
🗌 コードカパレッジを有効の	こする □ プロファイラを有効にする		
追加指令 —			
ASSIGN(EXTERNAL)		^	
		~	
		高度	

- ③ エクスプローラを起動し、サンプルのソースコードを解凍したフォルダから BOOK-INFO.cpy"と"BOOK.cbl"をプロジェ クトフォルダーにドラッグアンドドロップします。
- ④ ソリューションエクスプローラーから、2つのファイルが正常にロードされていることを確認します。



- 4) Enterprise Server の起動
 - [サーバーエクスプローラー] ペインを選択、[Micro Focus Server] > [localhost] > [ESDEMO]と展開しま す。[ESDEMO]の上で右クリックし、コンテクストメニューから[開始]を選択します。



② Visual Studio の[出力]に起動メッセージが表示されます。





③ 正常に開始されると [サーバーエクスプローラー] 上の ESDEMO アイコンが起動されたことを示す緑色のアイコンに 切り替わります。



3.2 Enterprise Server の設定変更

- 1) ディプロイ用フォルダーを作成します。
 - ① 「RESTfulCOBOL」プロジェクト上で右クリックし、コンテクストメニューから[追加]→[新しいフォルダー]を選択します。
 - ② フォルダー名に"deploy"を指定します。
- 2) 「.mfdeploy」ファイルをインポートします。
 - ① 作成した「deploy」フォルダ上で右クリックし、コンテクストメニューから[追加]→[既存の項目]を選択します。
 - ② Visual COBOL インストールフォルダ¥deploy 配下にある[.mfdeploy]ファイルを指定します。
- 3) サーバーエクスプローラーを表示し、「Micro Focus Server」上で右クリックし、コンテクストメニューから[管理] を選択しま す。



4) Enterprise Server Administration コンソール画面にて [通信プロセス] 列中の [リスナー] の横にある [詳細] ボ タンをクリックします。

	サーノ	(- 1-	1フィル	夕後: 1 合計: 1			10	ページ	内表示
フィルタ	9	イプ : []	すべて	✓ 名前: *			ステータス: すべて ▼ クリア		
リポジトリ: fi	le:///C:\Pr	ogramData	Micro Foci	IS/VISUAI COBOL/MFDS/	5		9-70	:1 開始:1 ∉	<u>5,1</u> ∷0 ⊼:
	タイプ	名前 ▼	ステータス	通信 プロセス	イセンス	セキュ リティ	ステータス ログ	オブジェクト	説明
編集	MFES	ESDEMO	開始 詳細	1 tep:10.18.12.104*9000 (1 10:12.109)* 3 リスナー 詳細		Default	Server: CP 1: CP 1	7 サー ビス 3 バン ドラ パッ 1 ケー 詳細 ジ	Sample



- 5) リスナーの構成を変更します。
 - ① リスナーの構成情報が表示されるので[名前]列中の[Web]の[編集]ボタンをクリックします。
 - ② 構成情報フィールドを見るとデフォルトは「uploads=<ES>/deploy」となっています。この場合、Visual COBOLインストールディレクトリ配下の deploy フォルダがディプロイ用フォルダとして使用されます。通常、Program Files (x86)等のフォルダは管理者権限を持つユーザーでないと書き込みできないので変更を行います。
 - ③ 下記の例のように設定を変更し、[適用]ボタンをクリックします。

例:uploads=c:/tutorials/NativeCOBOL/ RESTfulCOBOL/deploy

3.3 RESTful Web サービスの開発作業

- 1) RESTful Web サービスのプロファイル作成
 - ① 「RESTfulCOBOL」プロジェクトを右クリックし、コンテクストメニューから[追加]> [新しい項目]を選択します。



② [COBOL] > [Native] アイテムを選択し、 [Service Interface] を指定します。

新しい項目の追加 - RESTfulCOBOL			
▲ インストール済み	並べ替え	2. 既定 🔹 👯 📘	
▲ COBOL Native	٥	COBOL プログラム	COBOL
▶ オンライン	••••• •••••	Service Interface	COBOL
	D	コピーブック	COBOL
	D	リソース ファイル	COBOL
	Ð	アプリケーション マニフェスト	COBOL

- ③ [名前]は、"RESTfulBOOK.svi"を指定し、[追加(A)] ボタンをクリックします。
- ④ [サービスインターフェイス]画面が表示されるので、インターフェイスの種類はデフォルトの「Web サービス」を選択したま ま[OK]ボタンをクリックします。

Focus

サービスインターフェイス	×
名前: RESTfulBOOK インターフェイスの種類:	
Web サービス	~
転送形式 SOAP JSON ソースの種類: COBOL プログラム OK キ	~ ヤンセル

- 2) 書籍検索用のオペレーションを作成
 - 下図のように Visual Studio 内で「RESTfulBOOK.svi」がアクティブな状態で[オペレーション] メニュー > [新 規作成]を選択します。



 ② [オペレーションプロパティ] ダイアログが表示されるので[名前] に"SeachBOOK"を入力し、[プログラム/コピーブック を選択] では、「BOOK」を選択、[Select entry point]も「BOOK」を選択し、[OK]ボタンをクリックします。

オペレーションプロパティ		:	×
全般 HTTP ユーザー出口			
名前: SearchBOOK			
プログラム/コピーブックを選択:			
BOOK			
Select entry point (parameters show	n at right):		
BOOK	Name	Picture	
	LNK-FUNCTION	X	
		X(2)	

3) COBOLとRESTful Web サービス間の変数型変換マッピングを定義(書籍情報検索用オペレーション)



① 「LNK-FUNCTION」を [インターフェイスフィールド]ペインにドラッグアンドドロップします。

В	00 K.cbl RESTfulB C	00K.svi * 🌸 🗙 RESTfulCo	OBOL		
	COBOL エントリポイ	1 - F [BOOK] of I	Prograi	Operation SearchBOOK	
1	2 <mark>前</mark>	PICTURE		× 1/2 - 7 - 1/7 - 1 - 1 - 1	
	LNK-FUNCTION	X			
	E LNK-B-DETAILS			名前	「方
	🗉 🗃 LNK-B-TEXT-D				
	📥 LNK-B-TITLE	X(50)			
	📥 LNK-B-TYPE	X(20)			
	📥 LNK-B-AUT	X(50)			
	📥 LNK-B-STOCK	X(4)			
	📥 LNK-B-RETAIL	9 (5) co mp-3			
	📥 LNK-B-ONHAND	9 (5) co mp-3			
	📥 LNK-B-SOLD	9 (5) co mp-3			
	📥 LNK-FILE-STATUS	X(2)			

- ② 次に「LNK-B-STOCKNO」も [インターフェイスフィールド] ペインにドラッグアンドドロップします。
- ③ 次に「LNK-B-DETAILS」及び「LNK-FILE-STATUS」も [インターフェイスフィールド] ペインにドラッグアンドド ロップします
- ④ 次に「LNK-B-DETAILS」をダブルクリックします。

Operation SearchBOOK			~
インターフェイスフィーJ	۲		
名前	方向	型	0
LNK_FUNCTION	入力	string	
LNK B STOCKNO	スカ	string	
E CNK_B_DETAILS	入力		
LNK_FILE_STATUS	入力	string	

⑤ グループプロパティが表示されるので方向を「入力」から「出力」に変更します。

グループプロノ	रित		×
名前: OCCURS 方向:	LNK_B_DETAILS		
		ОК	キャンセル

⑥ 同様に「LNK-FILE-STATUS」の方向も「入力」から「出力」に変更します。



- 4) 書籍データ追加機能のオペレーションを追加
 - ① 2)の手順で「AddBOOK」オペレーションを作成します。
 - ② 次に[HTTP] タブを選択します。HTTP メソッドが [POST] になっていることを確認し、[OK] ボタンをクリックしま す。
- 5) 書籍データ登録機能オペレーションのインターフェイスマッピングを定義
 - 3)の手順と同様に「AddBOOK」オペレーションの変換マッピングを定義します。最終的には下図のようになるよう にしてください。

Operation	AddBOOK				~
インターフ	ェイスフィー	ールド			
名前			方向	型	0
📥 LNK_FUN	CTION		- እ. ታ	string	
	ETAILS _STATUS		入力 出力	string	
I			LNK_FUN LNK_B_D LNK_FILE をドラッグアン LNK_FILE カへ変更し	CTION、 ETAILS、 ミ_STATUS ンドドロップして、 ミ_STATUS のみ方 ます。	向を出

- 6) プロジェクトと Enterprise Server 「ESDEMO」を関連付ける
 - サーバーエクスプローラーにて、[Micro Focus Servers] > [localhost] > [ESDEMO] を右クリックし、コンテク ストメニューから[プロジェクトと関連付ける] > [RESTfulCOBOL] を選択します。

サー	バー エクスプロ	1-5-		RESTfulBO	OK.svi	+ ×		ļ
Ċ	× 🕛	*=	ē-	СОВО	LΙ	ントリポイン	► [BOOK]	
4	Micro	Focus alhost	Server	名前			PICTURE	
⊳	Azure		開始(S)			CTION TAILS	Х	
⊳	■ サーバー	- ±	停止 再起動(R)			-TEXT-DETAILS		
	-) - / 1	×	判定的(N)			K-B-TITLE K-B-TYPE	X(50) X(20)	
			プロジェクトと関う	重付ける	•	RESTfulCOB	OL	
			認証情報をクリ	ア(D)		-STOCKNO	X(4)	
			コンソールログを剥	長示(L)		-RETAIL	9(5) comp-3	
					LINK-E	-ONHAND	9(5) comp-3	

3.4 コンパイルした COBOL アプリケーションを Enterprise Server ヘディプロイ

1) ディプロイする COBOL プログラムの指定



- ① ソリューションエクスプローラーにて「RESTfulBOOK.svi」を右クリックし、コンテクストメニューから[プロパティ]を選択 します。
- ② プロパティペインの[ディプロイするアプリケーションファイル] 右横にある[...]ボタンをクリックします。

プロパティ	- ₽	×
RESTfulBOOK.svi ファイル プロパティ		•
サービス エンドポイント URL	http://localhost:9003	٠
サービス バージョン	1.0	
サービス ベース パス	/temppath/	
サービス名	RESTfulBOOK	
ディプロイされたアプリケーション パス	C:¥work¥RESTful¥RESTfulCOBOL¥RE	
ディプロイするアプリケーションファイル		
トランザクション管理	Application	-

③ 「項目の追加/削除] ウィンドウが表示されるので「ファイル追加]ボタンを押します。

項目の追加/削除	×
項目の追加/削除するにはボタンを使い、再配置	言するには項目をドラッグします。
	ファイル追加
	肖山除
	OK キャンセル

- ④ エクスプローラから「VisualStudio プロジェクトフォルダ¥RESTfulCOBOL¥bin¥x86¥Debug」まで移動し、 「BOOK.dll」及び「BOOK.idy」を指定します。
- ⑤ [項目の追加/削除] ウィンドウにファイルがセットされるので[OK]ボタンをクリックします。

項目の追加/削除	×
項目の追加/削除するにはボタンを使い、再配置するには項 OBOL¥RESTfulCOBOL¥bin¥x86¥Debug¥BOOK.dll OBOL¥RESTfulCOBOL¥bin¥x86¥Debug¥BOOK.idy	目をドラッグします。
	ファイル追加 削除
< >	
ОК	キャンセル

- 2) RESTful Web サービスのコンポーネント一式を Enterprise Server ヘディプロイする
 - ① 「RESTfulBOOK.svi」を右クリックし、コンテクストメニューから[ディプロイ] を選択します。

			1.2
	前回の実行を表示		fulBOOK.svi
9	開く(O)		
	ファイルを開くアプリケーションの選択(N)		
	ここまでスコープ指定する(S)		
Ē	新しいソリューション エクスプローラーのビュー(N)		
	プロジェクトから除外(J)		
Ж	切り取り(T)	Ctrl+X	
Ð	⊐ピ–(Y)	Ctrl+C	
X	削除(D)	Del	
χ	名前の変更(M)		
	実行時環境を構成(C)		
	検査(V)		
	รัสวีนิส(D)		

② ディプロイが完了すると下図のようなメッセージが出力されます(ここでは警告は無視して構いません)。

出力			
出力元(S): Genera	ition	- 🖆 🖆 🞽	22
0020 (2019/02/22 0021 (2019/02/22 0030 (2019/02/22 0030 (2019/02/22 0002 (2019/02/22 ディブロイメント ディプロイメント	14:19:57): Adding service and package d 14:19:58): Using directory at mrpi://12 14:19:58): ES server "ESDEMO" notified 14:19:58): ES server "ESDEMO" notified 14:19:58): Installation of package "RES が完了 完了	objects to directory 27.0.0.1:86 service "/temppath/RESTfi service "/temppath/RESTfi STfulBOOK.car" finished w	ulBOOK/1.0#SearchBOOK″ is available ulBOOK/1.0#AddBOOK″ is available ith 3 warnings

- 3) Enterprise Server Administration コンソール画面よりディプロイされたことを確認
 - サーバーエクスプローラーを表示し、「Micro Focus Server」上で右クリックし、コンテクストメニューから[管理]を選択します。

きサ	-/(- I	クスプロー	5 -			▼ -12	×
	5 ×	te te	8	1			
<u>کړ را</u>	⊿ 📑	Micro Fo	cue S	on/or			
Ч		local		管理(A)			
Ϋ́		, 📲 E		MFDS 構成((M)		
4		Azure サーバー	Ç	最新の情報	こ更新		\$
- Juli	กิ	データ接続	ŧ				

② Enterprise Server Administration コンソール画面にて [オブジェクト] 列中の [サービス] の横にある [詳細] ボタンをクリックします。

	サーノ	(- 1-	1フィル	夕後: 1 合計: 1			① 自動更新間隔(秒) 10	ページ	内表示
フィルタ	9	イプ: 🔤	すべて	▶ 名前: *			ステータス: すべて ソ クリア		
リポジトリ: fi	le:///C:\Pr	ogramData	Micro Focu	Is/Visual COBOL/MFDS/			グーゼ	:1 開始:1 傍	<u>▶止</u> :0 応
編集	917 MFES	名前 V ESDEMO	ステータ ス 開始 詳細 停止	通信 プロセス 1 top:10:18:12:104 ⁺ 9000 (10:18:12:104) ✔ 3 リスナー 詳 相	ライセンス -/10	セキュ リティ Default	ステータス ログ Server: CP 1: MDS38011 Server started successfully 16:19:08 02/21/19 22 hours 10 minutes 19 seconds in "MPE" state since 16:19:08 02/21/19 Started externally using ES ID "mf_mdsa" under system ID "Administrator, Server Manager PID 5612	オブジェクト 7 サー 詳細 3 バン 詳細 パッ 1 ケー 詳細 ジ	說明 Sample

③ 最下行にディプロイした RESTful Web サービスが追加されていることを確認します。



	ES	ES MF ES 編集	1	1	CP 1 Web Services and J2EE tcp:192.168.150.149*:9003 (192.168.150.149)			Available	ок	
削除	&	2 of 2 operations sho	wn							
		#SearchBOOK 編集	1	1	CP 1 Web Services and J2EE tcp:192.188.150.149*:9003 (192.168.150.149)	MFRHJSON	/temppath/RESTfulBOOK/1.0	Available	ок	(respo Allow-(
		#AddBOOK 編集…	1	1	CP 1 Web Services and J2EE top:192.168.150.149*:9003 (192.168.150.149)	MFRHJSON	/temppath/RESTfulBOOK/1.0	Available	ок	[respo Allow-(

 ④ 追加されていることを確認したら、Enterprise Server Administration コンソール画面のペインを×マークをクリック して閉じます。

 RESTfulCOBOL - Microsoft Visual Studio (管理者) 									
ファイル(F) 編集(E) 表示(V) プロ	ロジェクト(P) ビルド(B) デバッグ(D) チーム(M) ツール(T) テスト(
🖁 O - O 🏠 - 🚔 💾 🗳 🖇	○ - ペ - Debug - x86 - ▶ 開始 - 局場 -								
サーバー エクスプローラー 🔹 🕂 🗙	tok-atlas:86: Microerver Administration 🔸 🔀 RESTfulBOOK.svi								
🖒 🖂 🏣 🔚 📴	URL(U): http://127.0.0.1:86/ 問じる (Ctrl+E4)								
 Micro Focus Server 	MICRO 💒 Enterprise Server Aami								

3.5 RESTful Web サービスのテスト

- 1) RESTful Web サービステスト用のクライアント生成
 - ① 「RESTfulBOOK.svi」を右クリックし、コンテクストメニューから [クライアントの生成] を選択します。

٦		実行時環境を構成(C)	
		検査(V)	
		รั _้ าวึ่นิ1(D)	
	_	WSBIND の生成(G)	
		クライアントを生成(G)	
		プロジェクトのデフォルトを使用	
	۶	プロパティ(R)	Alt+Enter

- 2) Enterprise Server がアクセスするデータファイルを指定
 - ① 「サーバーエクスプローラー」に切り替えます。
 - ② [停止]ボタンをクリックし、Enterprise Server を停止します。
 - ③ [編集]ボタンを押して構成を変更します。

77119	9-	イプ: す	۲ ۲	✔ 名前: *			ステータス: すべて ▼ クリス				
リポジトリ: file	リポジトリ: file:///C:\ProgramData\Micro Focus\Visual COBOL\MFDS\ サーバー: 1 開始										
	タイプ	名前 🔻	ステータス	通信 プロセス	ライ セン ス	セキュ リティ	ステータス ログ				
編集	MFES	ESDEMO	<u>停止</u> 開始	1 tcp:192.168.150.149*-9000 (tok-svr2018.localdomain) 3 リスナー 詳細	-/10	Default	Server: CP 1: CP				

④ [サーバー] > [プロパティ] > [一般]タブの[構成情報]にデータファイルの場所を示す以下の値を設定します。ディレ クトリ名はデータファイルを解凍した任意のフォルダとなります。



[ES-Environment]

BOOKINFO=C:\u00e4work\u00e4DAT\u00e4BOOKINFO.DAT

構成情報	
[ES-Environment] BOOKINFO=C:\work\DAT\BOOKINFO.DAT	^
	\sim
説明	
Sample Micro Focus Enterprise Server	< >
キャンセル OK 適用 エクスポート コピー 削除 検証	

- ⑤ [OK]ボタンをクリックし、画面を閉じたら[開始]ボタンをクリックし、Enterprise Server を起動します。
- 3) テスト用アプリケーションの実行準備
 - ① RESTfulBOOK-app.cbl をダブルクリックして開きます。
 - ② テスト用クライアントは実行を終了すると DOS 画面から消えてしまうため、158 行目にブレークポイントを設定します。



- ③ [ビルド]メニュー > [ソリューションのリビルド]を選択します。
- ④ 「RESTfulBOOKClientApp」プロジェクトを右クリックし コンテクストメニューから[スタートアッププロジェクトに設定] を選択します。

		RESTfulBOOKClientApp	
*	ี่ ¹	Properties	
	リビルド(E)	RESTfulBOOK-app.cbl	
-	クリーン(N)	RESTfulBOOK-proxy.cbl	
		STfulCOBOL	
_	ここよで入二一ノ損圧9~(3)	Properties	
	新しいソリューション エクスプローラーのビュー(N)	BOOK.cbl	
	ビルドの依存関係(B)	BOOK-INFO.cpy	
	追加(D)	► RESTRIBOOK SWI	
	既存の COBOL 項目を追加		
益	NuGet パッケージの管理(N)		
ф	スタートアップ プロジェクトに設定(A)		
	-" it then	<u></u>	

4) 生成したテスト用 COBOL クライアントの実行(登録処理)

ツールバーにて [開始] アイコンをクリックし、アプリケーションを起動します。 DOS プロンプトでアプリケーションが起動します。



- ② 「Service Address」は、デフォルトのままで Enter キーを押します。
- ③ 「Username」、「Password」は変更しないのでそのまま 2 回 Enter を押します。
- ④ 「Operation (1 = SEARCHBOOK, 2 = ADDBOOK」は、"2"を入力して Enter を押します。
- ⑤ 「LNK_FUNCTION」は、"2"を入力して Enter を押します。
- ⑥ 「LNK_B_TITLE」は"OLIVER TWIST"を入力して Enter を押します。
- ⑦ 「LNK_B_TYPE」は"FICTION"を入力して Enter を押します。
- ⑧ 「LNK_B_AUTHOR」は"DICKENS"を入力して Enter を押します。
- ⑨ 「LNK_B_STOCKNO」は、"5555"を入力して Enter を押します。
- ① 「LNK_B_RETAIL」は、"1000"、「LNK_B_ONHAND」は、"3000"、「LNK_B_SOLD」は、"2333"を入力して
 て Enter を押します。
- ⑪ RESTfulの Web サービスが実行されます。「Ink_FILE_STATUS」に"02"が返ってきたら登録成功です。
- ② ブレークポイントで止まっているのでツールバーから[続行]をクリックして処理を終了させます。
- 5) 生成したテスト用 COBOL クライアントの実行(検索処理)
 - ツールバーにて [開始] アイコンをクリックし、アプリケーションを起動します。DOS プロンプトでアプリケーションが起動します。
 - ② 「Service Address」は、デフォルトのままで Enter キーを押します。
 - ③ 「Username」、「Password」は変更しないのでそのまま 2 回 Enter を押します。
 - ④ 「Operation (1 = SEARCHBOOK, 2 = ADDBOOK」は、"1"を入力して Enter を押します。
 - ⑤ 「Ink_FUNCTION」は、"1"を入力して Enter を押します。
 - ⑥ 「Ink_FUNCTION」は、"2222"を入力して Enter を押します。
 - ⑦ RESTfulの Web サービスが実行され、データが返ってきます。「登録処理」で入力した内容が確認できます。





⑧ ブレークポイントで止まっているのでツールバーから[続行]をクリックして処理を終了させます。

3.6 インスタンスの停止

- 1) Enterpiser Server の停止
 - 「Native COBOL」ソリューションに戻り、サーバーエクスプローラーにて [Micro Focus Servers] > [localhost]
 > [ESDEMO] を右クリックし、コンテクストメニューから [停止] を選択します

刘 NativeCOBOL - Microsoft Visual Studio (管理者)				
ファイル(F) 編集(E) 表示(V) プロジェクト(P)	ビルド(B)			
🖉 🗢 🗢 🛛 🎌 🖕 🔛 🔐 🖉 - 🖓 -	Debug			
⁺ サーバー エクスプローラー	▼ -= X			
3 C × *e *e №				
A B localbost				
▶ \Lambda Azure 開始(S)				
ど ■ サーバ 停止				

WHAT'S NEXT

● 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。