

Enterprise Developer チュートリアル メインフレーム COBOL 開発: JCL Eclipse 編

1 目的

本チュートリアルでは、Eclipse 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、JCL 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。

2 前提

- 2.1 本チュートリアルで使用したマシン OS: Windows 11 Pro
- 2.2 使用マシンに Enterprise Developer 10J for Eclipse がインストールされていること

3 チュートリアル手順の概要

- 3.1 チュートリアルの準備
- 3.2 Eclipse の起動
- 3.3 メインフレーム COBOL プロジェクトの作成
- 3.4 テキストファイルのエンコード指定
- 3.5 プロジェクトプロパティの設定
- 3.6 ビルドの実行
- 3.7 文字エンコーディングの設定
- 3.8 Enterprise Server インスタンスの設定
- 3.9 Enterprise Server インスタンスの開始と確認
- 3.10 JCL の実行
- 3.11 プロシージャライブラリの作成
- 3.12 COBOL バッチプログラムの実行
- 3.13 COBOL バッチプログラムのデバッグ
- 3.14 Enterprise Server インスタンスの停止



3.1 チュートリアルの準備

例題プログラムに関連する資源を用意します。

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



3.1.2 Eclipse のワークスペースで使用する「work」フォルダを C:¥ 直下に作成します。

3.2 Eclipse の起動

3.2.1 メニューから [Enterprise Developer for Eclipse] を起動します。



3.2.2 前項で作成した C:¥work をワークスペースへ指定して、[起動] ボタンをクリックします。



3.2.3 [ようこそ] タブでは [Open COBOL Perspective] をクリックして、COBOL パースペクティブを開きます。





3.2.4 パースペクティブ表示後、[プロジェクト] プルダウンメニューの [自動的にビルド] を選択して、これをオフにします。



3.2.5 COBOL パースペクティブが開いていない場合は Eclipse 右上の [パースペクティブを開く] アイコンをクリッ クして表示後、[COBOL] を選択して [開く] ボタンをクリックします。



3.3 メインフレーム COBOL プロジェクトの作成

3.3.1 用意した例題ソースをインポートします。[ファイル] プルダウンメニューから [新規] > [メインフレーム COBOL プロジェクト] を選択します。

411 cm	1000	the second second	1.1242.1.144	10.00	-			and the second	de la contrata de la contra	
/アイル(F)	漏枼(E)	リノアクタリンク	ナビケート(N)	検索	70	ジェク	F(P)	実行(R)	ワイントワ(W)	1
新規	(N)		Alt+	・シフト+I	N >	ピ	COB	OL JVM J	ロジェクト	
ファイ	ルを開く(.)					(B)	メイン	יסט א-ערי	BOL プロジェク	1

3.3.2 [プロジェクト名] は任意ですが、ここでは JCLDEMO を入力し、テンプレートは 64 ビットを選択して [終了] ボタンをクリックします。

IIII メインフレーム COBOL プロジェクトの新規作成 - ロ X
メインフレーム coBol プロジェクト CICS®、JCL または IMS® アプリケーションを作成するためのプロジェクトです。
プロジェクト名(D) JCLDEMO プロジェクトテンプレートを選択
(월 Micro Focus 카가/나~ト (의 선가) (월 Micro Focus 카가/나~ト (의 선가)
テンプレートの設定を構成
は所: アデバルシステムを選択: default ~
✓ デフォルト・ロケーションの使用(D)
ロケーション(): CwookWCLDEMO 単質(B) ファイル・ウステルを装置(C): 778かト
⑦ 終7(D) キャンセル

3.3.3 COBOL エクスプローラーに作成したプロジェクトが表示されます。

🔓 COBOL エクスプローラ 🛙

JCLDEMO



- 3.3.4 プロジェクトを作成したことにより C:¥work¥JCLDEMO フォルダが作成されています。このフォルダ配下に JES 機能で使用するフォルダを Windows エクスプローラーを使用してあらかじめ用意しておきます。
 - カタログファイルやスプールファイルを配置するため DATAFILE フォルダを C:¥work¥JCLDEMO 配下 へ作成します。
 - プロシージャファイルを配置するため、プロシージャライブラリとして SYS1.PROCLIB フォルダを C:¥work¥JCLDEMO¥DATAFILE 配下へ作成します。

ク(C:)	\rightarrow work \rightarrow JCLDEMO \rightarrow DATAFILE \rightarrow
^	名前
	SYS1.PROCLIB

・ 実行時に使用する system フォルダを C:¥work¥JCLDEMO 配下へ作成します。

7 (C:) > work > JCLDEMO > 名前

	.settings
	DATAFILE
а.	system

3.3.5 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。 [ウィンドウ] プルダウンメニューの [設定] > [Micro Focus] > [COBOL] > [指令 の確定] > [指令の自動確定を実行] で [常になし] を選択し、[適用して閉じる] ボタンをクリックします。



3.3.6 用意した例題プログラム類をインポートします。JCLDEMO プロジェクトを右クリックして [インポート] > [イン ポート] を選択し、インポートウィンドウにて [一般] > [ファイル・システム] を選択後 [次へ] ボタンをクリックし ます。

◎ インポート		
選択 ローカル・ファイル・システムから既存のプロジェクトへリソースをインボートします。	Ľ	5
インボート・ウィザードの選択(<u>S</u>):		
7イルタ入力		
 ◆ (⇒ 一般 ● アーカイブ・ファイル ● ファイル・ジスラム ● フォルダーまたはアーカイブ由末のプロジェクト ● 営業フリジェクトをワークスペースへ ■ 設定 		^
(?) < 戻る() 次へ(い) > 純了()	キャンセ	JL



3.3.7 C:¥Tutorials を [次のディレクトリーから] へ指定すると内容が表示されますので、最後から 4 ファイルにチェックをして [終了] 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へ例題プログラムが配置されます。

◎ インポート		-	
ファイル・システム ローカル・ファイル・システムからリソースをインボートします。			
次のディレクトリーから(Y): C.¥Tutorials¥Tutorials		~	参照(<u>R</u>)
> ■ > Tutorials タイプをフィルター(D)) すべて選択(S) 選択をすべて解除(D) インボート先フォルダ(L): [JCLDEMO オプション □ 警告を出さずに既存リソースを上巻き(Q) □ トップ・レベルのフォルダーを作成(C) 林準 5-201	ACCISET.cpy CopyLjcl GeosyNT2.cbl SORTD.prc i SORTD.prc i somwrt2.icl		◆ ◎ ◆ 照(<u>₩</u>)
0	< 戻る(B) 次へ(N) > 終了	(E)	キャンセル

3.3.8 COBOL エクスプローラー内に表示されている JCLDEMO プロジェクトにインポートしたファイルが表示されて いることを確認後、作成した SYS1.PROCLIB フォルダへ SORTD.prc ファイルをドラッグ&ドロップして移 動します。

ಕ್ಷಿ COBOL I⊅スプローラ ⋈ 📒	≌ COBOL ΙΆΖΊ□−Ϝ ⋈ 📃
 ▲ 2 JCLDEMO ▶ ○ COBOL プログラム ▶ ○ JCL ファイル ▲ ▷ DATAFILE ▷ SYS1.PROCLIB ■ SORTD.prc → 	
 	パたフォルダが表示されない場合があります。その際は COBOL エクスプローラー > [フィルタとカスタマイズ] を選択後 [カテゴリ外の空のフォルダ] のチェックをオ ^{アとカスタマイズ(F)} → □ カテコリ外の空のフォルダ

3.4 テキストファイルのエンコード指定

Eclipse ではワークスペースの設定として、テキストファイルのエンコードを指定できます。ソースファイル類のエンコードに沿って適切なエンコードを指定してください。この例題では Windows-31j を使用します。

- 3.4.1 Eclipse の [ウィンドウ] プルダウンメニューから [設定] を選択し、設定ウィンドウを表示します。
- 3.4.2 左側ペインで [一般] > [ワークスペース] を選択し、右側ペインの [テキスト・ファイル・エンコード] に MS932 または Windows-31j が指定されていることを確認します。「デフォルト」 がこの値ではない場合は、「その他」を 選択して、これらを指定してください。

テキスト・ファイル・エンコード(T)						
O デフォルト(U) (windows-31j)						
〇 その他(O):	windows-31j \sim					



3.4.3 指定後は [適用して閉じる] ボタンをクリックします。

3.5 プロジェクトプロパティの設定

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。

- 3.5.1 COBOL エクスプローラー内の JCLDEMO プロジェクトを右クリックして [プロパティ] を選択します。
- 3.5.2 左側ツリービューの [Micro Focus] > [ビルド構成] > [リンク] を選択して、下記項目を指定します。指定後は [適用] ボタンをクリックしてください。

項目名		説明		
ターゲットの種類		実行ファイル形式を指定します。ここでは [全て INT/GNT ファイル] を 選択します。		
プラットフォーム ターグ	ブット	稼働ビット数を指定	します。ここでは	は [64 ビット] を指定します。
 Micro Focus ビルダー ビルド構成 > EMS > COBOL 	フィルタテキストを 設定 > 一般	入力 🗐	値	
> アセンブラリンカ アセンブラリンカ イベント	<u>出力の名</u> 出力パス		JCLDEMO New Configuration.bin	
ディプロイ ビルド環境	エノトリホイント ターゲットの種類 ドット数		すべて INT/GNT ファイル 64 ピット]
> リンク ✓ プロジェクト設定 > BMS ◇ COBOL		ッケージ化 COBOL アーカイブ (.car) ファイルにパッケージ化	いいえ	
		ッド	1261	

3.5.3 左側ツリービューの [Micro Focus] > [プロジェクト設定] > [COBOL] を選択して、下記項目を指定します。 指定後は [適用して閉じる] ボタンをクリックしてください。

項目名	説明
文字集合	EBCDIC または ASCII を指定します。ここでは [ASCII] を選択します。
ソースエンコーディング	Eclipse に指定したエンコードと一致するエンコーディングを指定します。 MS932 または Windows-31j と一致するように ANSI を指定します。
言語方言	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。
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チェックをオンに指定します。
.GNT にコンパイル	実行ファイル形式を GNT に指定します。
追加指令	ここでは指定しません。



✓ Micro Focus	フィルタテキストを入力	
ビルダー		12
ビルド パス	設定	1旦
> ビルド構成	~ 一般	
∨ プロジェクト設定	文字セット	ASCII
> BMS	ソース エンコーディング	ANSI
> COBOL	COBOL 方言	Enterprise COBOL for z/OS
Enterprise Server	ソース フォーマット	固定
> IMS	メインフレームのコピー処理	COPY
> アセンブラ コンパイラ	デバッグ用にコンパイル	はい
アセンブラ リンカ	EXIT PROGRAM を GOBACK として	て処理 ANSI
コンテナー	詳細	いいえ
ビルド環境	.GNT にコンパイル	はい
指令の確定	✓ 出力	
> 実行時構成	指令ファイルを生成する	いいえ
タスク・タグ	リストファイルを生成	いいえ
ビルター	コード カバレッジを有効にする	false
プロジェクト・ネーチャー	プロファイラを有効にする	false
プロジェクト・ファセット	✓ エラ-/警告	
プロジェクト参照	警告レベル	回復可能なエラーを含める(レベル E)
リファクタリング履歴	最大エラー数	100
> 検証	✓ 追加指令	
実行/デバッグ設定	追加指令	
	<mark>ソース エンコーディング</mark> SOURCE-ENCODING はソース プログラムの	カエンコーディングをコンパイラに渡します。 そ
	COBOL コンパイル設定:	
	CHARSET"ASCII" SOURCE-ENCODING" SOURCEFORMAT"fixed" NOPANVALET EXITPROGRAM"ANSI" NOTESTCOVER ERROR"100"	"ANSI" DIALECT"ENTCOBOL" " NOLIBRARIAN NOLIST anim NOPROFILE generate WARNING"1" MAX-
		デフォルトの復元① 適用①
(?)		適用して閉じる キャンヤル

3.6 ビルドの実行

3.6.1 COBOL エクスプローラー内のプロジェクトを右クリックして [プロジェクトのビルド] を選択するとビルドが実行 されます。

🔓 COBOL エクスプローラー 🕱

✓ ピ JCLDEMO プロジェクトのビルド(B)

3.6.2 コンソールタブで成功を確認します。

🖳 コンソール 🛛 🖹 問題 🧔 タスク 🔲 プロパ	ティー 🚹
Micro Focus ピルド: JCLDEMO аертоу.ств.неw_conтiguration:	
combinedbuild.cfg.New_Configuration	:
BUILD SUCCESSFUL Build finished with no errors.	
Total time: 0 seconds ―――ビルド完了――	_

- 3.6.3 COBOL エクスプローラーのプロジェクト内に存在する New_Configuration.bin フォルダ配下に実行ファイル(.gnt ファイル)が作成されていることを確認してください。
 - ジ JCLDEMO
 ジ COBOL ブログラム
 ジ JCL ファイル
 ジ DATAFILE
 ジ New_Configuration.bin
 ※ KSDSWRT2.gnt
 ※ KSDSWRT2.gnt.1.tlog
 【 KSDSWRT2.idy

Rocket software

3.7 文字エンコーディングの設定

Enterprise Server インスタンスを運用、管理する Enterprise Server Common Web Administration(以降 ESCWA)では、スプールやデータ内容などに含まれる日本語を正しく表示させるために、事前に文字セットを所定のフォルダ へ展開します。製品マニュアルの「リファレンス > コードセットの変換 > CCSID 変換テーブルのインストール > CCSID 変換テーブルをインストールするには」を参照しながら進めてください。

- 3.7.1 CCSID 変換テーブルをインストールします。
 製品マニュアルにリンクされている下記の IBM CCSID 変換テーブルを、Web ブラウザから任意のフォルダへ ダウンロードします。アドレスは変更される可能性がありますので、製品マニュアルにてご確認ください。
 http://www.microfocus.com/docs/links.asp?vc=cdctables
- 3.7.2 製品インストールフォルダ配下の etc フォルダに CCSID フォルダがない場合はこれを作成します。
 例)C:¥Program Files (x86)¥Micro Focus¥Enterprise Developer¥etc¥CCSID
- 3.7.3 ダウンロードファイルに含まれている Package2.zip を展開します。
- 3.7.4 展開した Package2 フォルダに含まれる IBM-932.zip を展開します。
- 3.7.5 展開した IBM-932 フォルダを切り取り、作成した CCSID フォルダ配下へ貼り付け、14 ファイルが含まれて いることを確認します。

📕 I 🕑	🔄 👻 IBM-932					- 🗆	×
ファイル	ホーム 共有 表示						~ 🕐
$\leftarrow \ \ \rightarrow$	 	→ Micro Focus → Enterpri	ise Developer >	etc > CCSID > IBM-932	~ ∂	IBM-932の検索	P
^	名前	2種類 へ	サイズ	更新日時			^
	03A434B0.MU-R-A2	MU-R-A2 ファイル	28 KB	1999/10/14 10:04			
	03A434B0.MU-R-A3	MU-R-A3 ファイル	28 KB	2002/01/15 11:58			
	03A434B0.MU-R-D	MU-R-D ファイル	28 KB	1999/10/14 10:04			
	03A434B0.PACKAGE	PACKAGE ファイル	5 KB	2002/01/16 10:37			
	34B003A4.RPMAP12A	RPMAP12A ファイル	336 KB	1999/10/14 10:03			
	34B003A4.RPMAP120	RPMAP120 ファイル	336 KB	1999/10/14 10:03			
	03A434B0.TPMAP11A	TPMAP11A ファイル	329 KB	1999/10/14 10:02			
	03A434B0.TPMAP12A	TPMAP12A ファイル	329 KB	2002/01/15 11:58			
	03A434B0.TPMAP110	TPMAP110 ファイル	329 KB	1999/10/14 10:02			
d,	34B003A4.UM-E-A21	UM-E-A21 ファイル	54 KB	1999/10/14 10:04			
2	34B003A4.UM-E-D12	UM-E-D12 ファイル	54 KB	1999/10/14 10:04			
1	03A434B0.UPMAP12A	UPMAP12A ファイル	446 KB	1999/10/14 10:01			
	03A434B0.UPMAP13A	UPMAP13A ファイル	490 KB	2002/01/15 13:42			
~	03A434B0.UPMAP120	UPMAP120 ファイル	447 KB	1999/10/14 10:01			~
14 個の項							

詳細については、製品マニュアルの「ディプロイ > 構成および管理 > Enterprise Server の構成および管理 > Enterprise Server Common Web Administration > [Native] > [Directory Servers] > リージョンとサーバー > リージョン > エンタープライズ サーバー リージョンの文字エンコーディングのサポート」を ご参照ください。

3.7.6 Windows サービスとして起動している Micro Focus[™] Enterprise Server Common Web Administration を再起動し、インストールした CCSID をロードさせます。

Q #=82				-		×
ファイル(E) 操作(A)	表示(V) ヘルプ(日)					
🗢 🔿 📅 🖾 🤅	à 📴 🖬 🖬 🕨 🔳 🕪 👘					
③ サービス (ローカル)	〇、サービス (ローカル)					
	Micro Focus Enterprise Server	·	説明	状態	スタート7	P: ^
	Common Web Administration	California Ces daemon	Micro Focus CES daemon	実行中	自動	
	サービスの使い	S Micro Focus Directory Server	Micro Focus Directory Server	実行中	自動	
	サービスの再起動	Micro Focus EA Integration Service Host	Micro Focus EA Integration Service Host	実行中	自動	
		Micro Focus Enterprise Analyzer Web Service	Micro Focus Enterprise Analyzer Web Service	実行中	自動	
	** 99.	🙀 Micro Focus Enterprise Server Common Web Administration	Micro Focus Enterprise Server Common Web	実行中	自動	
	Micro Focus Enternise Server	🖏 Micro Focus Event Monitor Service Shutdown Coordinator	Micro Focus Event Monitor Service Shutdow	実行中	自動	
	Common Web Administration	Micro Focus SEE Administration Server	Micro Focus SEE Administration Server		手動	
		🖏 Micro Focus SEE Listener	Micro Focus SEE Listener		手動	
		Mirro Forur SEE Monitor	Micro Focur SEE Monitor	*行由	A 25	, ×
	14118 (1819) (`				<u> </u>
	\ <u>\$\$\$</u> /(#*)					



3.8 Enterprise Server インスタンスの設定

Enterprise Developer は JES のエミュレーション機能を搭載している開発用 Enterprise Server インスタンスを内 包しており、各開発者がこのインスタンスを占有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うことができ ます。本番環境にはコンパイラーなどを含まない実行環境製品 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。

3.8.1 Enterprise Server インスタンスを運用、管理する Enterprise Server Common Web Administration (以降 ESCWA)は、製品が提供する VSAM 外部セキュリティマネージャー(ESM)モジュールによるセキュリティ がデフォルトで有効になっており、すべての処理において実行ユーザーの認証が行われます。 コマンドによりこの設定を無効にすることもできます。

ユーザー権限によっては実行を拒否されることがあり、この場合は ESCWA にログオン後、ESCWA に設定され ているセキュリティ設定を確認、または関連するリソースをメンテナンスしてください。

ES 自理 タッシュホー セキュリティ ナビゲーション	ト ネイティン ^ ユーサ	ゲーオプショ)-ム ン	セキュリティ * 新規作成				ES 空理 ダッシュボード ヤキュリティナドグーション ※ セキュリティマネージャ	*	1717 UV-7	メインフレーム 1 	2#1974 052.0%	inni.	
 ● セキュリティ マネージャ ◆ B ESCWAの構成 ◆ B VSAM ESM 9. コーザー 	V	アカウントロ		▽ 名前	V 109		V	 グロンボンボム う VIAN LOM 合 ユーザー 8 グルーブ 8 グルーズ 		2018 V	Web Administrat	ion 💌	1 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2	Uter I
怒 グループ	С С В В	アカニ CICSUSER IMSUSER	名前	デフォルト グループ ALLUSER ALLUSER	說明 Default CICS Default IMS	70	ション 団 向	多。ロール ② 構成レポート セキュリティ リソースナビゲーショ ン	ک	עש–ג	.IJスト <u>* リ</u> ソ	208.865		
() 構成レホート	8 8 8	JESUSER PLTPISUR SAFU		ALLUSER OPERATOR DEVGROUP	Default JCL CICS User fo Test security	000		> 高 ACIOSPOT > 高 AdmirAPI > 高 COICSOMD > 高 CIMS	Î	7 :	अ) इ.स.	₽ (68)	10 M	ACL
	8	SAFUIMS SYSAD		IVPGRP SYSADM	ES IMS User Administrato	0	前	Common Web Administration Communications Server Log Server Log	1	0	Communications Server	Log	Allow acce.	ALLOW #0
								Control Control ESCWA Configuration Kits Configuration		0	Configuration		Allow centr.	ALLOW-NO
詳細は製品	マニコ	アルを	<u>-</u> ت	参照くださし	<i>د</i> ارم			Logan MPUS Configuration Managed Access Native Access		0	ESCWA Configuration KBs Configuration		Allow acce.	ALLOW:*se ALLOW:*se

詳細は製品アニュアルをご参照ください。

デフォルトユーザーと初期パスワードを取得するために、スタートメニューから [Enterprise Developer] を選 択し、[Enterprise Developer コマンドプロンプト] を起動します。

Enterprise Developer コマンドプロンプト (64-bit)

(1)Enterprise Developer コマンドプロンプトから次のコマンドを実行して、 デフォルトユーザーと初期パスワードを取得します。 mfsecretsadmin read microfocus/temp/admin

:¥Users¥tarot¥Documents>mfsecretsadmin read microfocus/temp/admin "mfUser":"SYSAD", "mfPassword":"CKfa+x\t"}

上記例の場合、SYSAD がデフォルトユーザー、CKfa+xWt が初期パスワードです。 この情報をメモ帳などに記録してください。

セキュリティを無効にする場合 2 Enterprise Developer コマンドプロンプトから次のコマンドを実行します。 DisableESDefaultSecurity.cmd

C:¥Users¥tarot¥Documents>DisableESDefaultSecurity.cmd Micro Focus DisableESDefaultSecurity utility: Version 1.0.0

この場合は、以降のユーザー認証にかかわる手順は省いてください。

3.8.2 実行する開発用 Enterprise Server インスタンスを作成します。

Eclipse の [サーバー エクスプローラー] タブの [ローカル] > [Default] を右クリックして [管理ページを 開く] を選択します。Default に登録されているインスタンスが表示エラーになる場合は、Windows の Micro Focus[™] Directory Server サービスが開始されているか確認し、停止している場合は開始してください。



COBOL 1クスプローラー ■ サーバー エクスプローラー ※ Micro Focus Directory Server 年前 ^ CoBOL 1クスプローラー ※ サーバー エクスプローラー ※ サービン回答法 マロン 回答法 マロン 回答法 マロン 回答法 Supersonal Sectory Server 名前 ^ CoBOL 1クスプローラー ※ Micro Focus Directory Server 名前 ^ CoBOL 1クスプローラー ※ Micro Focus Directory Server 名前 ^ CoBOL 1クスプローラー ※ Micro Focus Directory Server 名前 ^ CoBOL 1クスプローラー ※ Micro Focus Directory Server 名前 ^ CoBOL 1クスプローラー ※ Micro Focus Directory Server 名前 ^ CoBOL 1クスプローラー ※ Micro Focus Directory Server 名前 ^ CoBOL 1クスプローラー ※ Micro Focus Directory Server 名前 ^ CoBOL 1クスプローラー ※ Micro Focus Directory Server 3 Micro Focus Directory Server 3 Micro Focus Officerup Server 3 Micro		(ローカル)	
等理パージを見/	℃ COBOL Iクスプローラー ■ サーバー Iクスプローラー ※ ◆ ③ ローカル [localhost:10086] > ■ Default [127.0.0.1:86] > ■ 新規作成(N) >	Unco Focus Directory Server <u>サービスの得た</u> 型売 型売 Micro Focus Directory Server	6.# ^ @MMADUS @MMADUS @MMES @MMES @MMCrofocu AutoPas Daemon @MMCrofocu CLS demon @MMCrofocu CLS demon @MMCrofocu CLS demon @MMCrofocu CLS demon

3.8.3 ブラウザが立ち上がり、ESCWA が表示され、ユーザー認証を求められます。

セキュリティ確保のために初期パスワードを変更してログオンします。 [パスワード変更] をクリックし、前項で確認したデフォルトユーザーと初期パスワード、新しいパスワードを入力し て [サブミット] をクリックします。

変更後のパスワードはご自身の責任で管理してください。

パスワード変更後、画面がタイムアウトした場合は新しいパスワードを使用してログオンしてください。

ES Enterprise Server Common Web Administration	ES Enterprise Server Common Web Administration
▲ Meto Focus Enterprise Serverでは、インストール後に基本 的なセキュリティ制能がデフォルトで何効になっていま す。 ↓ 目目清朝	ユーザーを SYSAD
1-9-8	ノスワード
/(スワード	新しんリイスワード・ ノスフードの細胞・
パスワード変更 ログオン	キャンセルサブミット
認証情報は、次のセキュリティ マネージャを使用 して検証されます: VSAM ESM	認証情報は、次のゼギュリティ マネージャを使用 して検証されます: VSAM ESM

3.8.4 画面の中央にある [新規作成] ボタンをクリックします。

ES 管理		ダッシュボード ネイティブ	ES .NET メインフ		
◆ ⑦グループ → 論理	⑦ リージョンおよびサーバー	プロパティー〜	セキュリティ 🗸 🗸	ジャーナル	& × バージョン: 1
> PAC	△ 岳 このDirectory ServerホストではTLSが	有効ではありませんが、ループバック トラコ	フィックの受信のみに制限されてい	\ इ व	
✓ I Directory Server ✓ I G Default C	リージョンおよびサーバー !	リスト C *新規作成	査 すべて削除 上工	クスポート ① インボー	-ト 國コピー ゆ 更新
ESDEMO					
ESDEMO64 > ∲ SOR	▼ 名前	▽ 説明	PAC	▼ エンドポイント	タイプ マステータス マ

3.8.5 [リージョンの新規作成] 項目の [名前]、[説明] は任意ですが、ここでは名前に JCLDEMO、説明に JCL チュ ートリアル用と入力します。 Eclipse の実行可能ファイルは 64 ビットを指定してコンパイルしたため、稼働させ る Enterprise Server インスタンスも同様に [64 ビット作業モード] ヘチェックを入れます。これにより警告 が表示されますが無視して先に進んでください。[MSS 有効] にチェックが入っていることを確認し、 [TN3270 リスナーの作成] のチェックを外して [保存] ボタンをクリックします。

リージョンの新規作成 ※示・ メCLEHO 当時 メCLデュートリアル用 ▲ Descon Second 10 - 20 × 20 月本に、	
やうーロレスない。 プラキンスイーム なっては、数時に次考約が増生するの 取びめるります。 の ムビンドが展モード ■ MSSTのリスナーの作成 ■ MSSTのリスナーの作成 ■ MSSTのリスナーの作成 ■ MSSTのリスナーの作文 ・ 入力必須の項目です	全要 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。
保存 戻る	



3.8.6 64 ビットアプリケーション稼働用の JCLDEMO インスタンスが作成され、一覧に表示されます。

ージョ	ョンおよう	びサーバー	リスト	C * ###	見作成 💼	すべて削除	业 エクスポート	☆インオ	K- F
名前	(JCLDEMO)	╱ 説明〜	PAC	エンドポイン	・ト〜 タイプ	▽ ステータ	ス〜 64ピット〜	MSS有効 ~	×
	名前	タイプ	ステータス	64ビット	MSS有効	セキュリ	71		
Ξ	JCLDEMO	Region	Stopped	~	~	デフォル	<i>.</i> ۲		

3.8.7 JCLDEMO インスタンスにカーソルを合わせ、[編集] アイコンをクリックします。

E	JCLDEMO	Region	Stopped	\checkmark	~	デフォルト	🧷 🔅 🗇
							編集

3.8.8 JCLDEMO インスタンスのログなどが出力される [システムディレクトリ] には前項で作成した system フォル ダを指定して、[リージョンの機能] の [JES 有効] をチェックします。指定後は [適用] ボタンをクリックします。

一般的なプロパティ ご 🥦 💼 🗰	リージョンの機能	
開始オプション 名前・Q システムディレクトリ Q JCLDEMO C\work\JCLDEMO\system	✓ MSS有効 ♀ ✓ JES有効 ♀ ✓ MQ有効 ♀	

3.8.9 表示画面の下にある [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、Eclipse からの 動的デバッグが可能になります。

🗌 ローカル コンソールを表示 💡	✔ 動的デバッグを許可 💡
□ システム起動時に開始する 🎗	☑ 64ビット作業モード 💡
□ 以前のログを削除 9	

3.8.10 [追加設定] の [構成情報] 欄に、文字エンコーディングを指定する MFACCCGI_CHARSET 環境変数に IBM-932 を認識させるための値である Shift_JIS と、プロジェクトのパスを指定する環境変数を設定し、最 後に [適用] ボタンをクリックします。

入力値)

[ES-Environment]

追加設定

proj=C:¥work¥JCLDEMO MFACCCGI_CHARSET=Shift_JIS

構成情報 😡 [ES-Environment] <u>proj</u>=C:\work\JCLDEMO MFACCCGI_CHARSET=<u>Shift_JIS</u>

3.8.11 画面上部の [JES] プルダウンメニューから [構成] を選択し、表示される画面の各項目を設定します。構成情報 に指定した proj 環境変数を使用して値を入力後、 [適用] ボタンをクリックします。

項目名	説明
JES プログラム パス	COBOL アプリケーションの実行ファイルが存在するパスを指定します。
システムカタログ	カタログファイルを出力するパスと、そのファイル名称を指定します。
データセットの省略時ロケーション	ジョブ実行時に生成されるスプールデータやカタログされるデータセットの デフォルトパスを指定します。
システムプロシージャライブラリ	プロシージャライブラリの名前を指定します。 ここでは SYS1.PROCLIB を入力します。



šの構成 C 適用		
JES プログラムバス Q	システム カタログ 🛛 🖓	
<pre>\$proj\New_Configuration.bin</pre>	\$proj\DATAFILE\catalog.dat	
	データセットの省略時ロケーション ♀	
	<pre>\$proj\DATAFILE</pre>	
システム プロシージャ ライブラリ 🎧	Fileshare 構成ロケーション 〇	
SYS1.PROCLIB		

3.8.12 [イニシエータ] の [新規作成] ボタンをクリックします。

イニシエータ	С	* 新規作成

3.8.13 下記画面のように入力して [保存] ボタンをクリックします。この指定により JCLDEMO インスタンスが開始時 にイニシエータが稼働し、ジョブクラス A,B,C のジョブが実行可能になります。

JESイニシエータ	
名前・♀	
INITABC	
クラス 🛿	
ABC	
説明Q	
A,B,C 97X01 = 91 ~ 3	/
・入力必須の項目です	
保存	戻る

3.8.14 セキュリティ観点から、Web リスナーのデフォルトステータスは [Disabled] になっています。安全を確認したう えで、 [一般] プルダウンメニューから [リスナー] を選択し、表示された Web リスナーのステータスを [Stopped] へ変更後、 [適用] ボタンをクリックします。

TLS設定			
ステータス	ステータスの設定 🛛		実際のアドレス
Disabled	Stopped	~	tcp:0.0.0.0:0

3.8.15 画面左側ペインの [Default] をクリックして一覧画面に戻ります。

~	🖽 Directory Server	
Г	🗸 🔢 🔂 Default	
	CICSDEMO	
3	重要	

詳しくは製品マニュアルをご参照ください。

バージョン 7.0 から、JES 関連ファイルである SPLJOB.DAT のフォーマットが改善されています。 そのため、旧バージョンのファイルを 7.0 以降で利用する場合は mfsplcnv コマンドを使用して新フォ ーマットにコンバートする必要があります。コンバートを実行すると、古いフォーマットのファイルは SPLJOB.bak として保存されます。 対象ファイルの特定には MFSYSCAT 環境変数を利用して、カタログファイルを指定します。 例) set MFSYSCAT=C:¥work¥JCLDEMO¥DATAFILE¥catalog.dat mfsplcnv -2

SPLJOB.dat



3.9 Enterprise Server インスタンスの開始と確認

- 3.9.1 Eclipse に戻り、サーバーエクスプローラー内に JCLDEMO 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 [Default] を右クリックし、[更新] を選択してリフレッシュしてください。
- 3.9.2 サーバーエクスプローラー内の JCLDEMO インスタンスを右クリックし、[プロジェクトに関連付ける] > [JCLDEMO] を選択します。これにより JCLDEMO プロジェクトから実行されるアプリケーションは JCLDEMO インスタンスで処理されることになります。

プロジェクトに関連付ける	 JCLDEMO

3.9.3 JCLDEMO インスタンスを右クリックして [開始] を選択します。

>	1	JCLDEMO
>	1	新規作成(N)
>		管理ページを開く
- -		開始

3.9.4 ESMAC サインオンウィンドウが表示されます。前述のセキュリティが有効の場合は [サーバーに資格情報が必要] にチェックし、SYSAD ユーザーと変更したパスワードを指定します。無効の場合はそのまま [OK] ボタンを クリックします。

SMAC サインオン		×
サーバー JCLDEMO の ESMAC	認証情報を入力してください	
□[ワーハーに貧格情報か必要]		
ユーザー名:		
パスワード:		
グループ:	デフォルト グループは空白	
☑ 資格情報の保存		
		ОК + +у/т/l

3.9.5 ESCWA へ移動して開始状態であることを確認後、[編集] アイコンをクリックします。

Ξ	JCLDEMO	Region	Started	~	~	デフォルト	D 🕸 🗇
							編集

3.9.6 画面上部の [モニター] プルダウンメニューから [ログ] > [コンソールログ] を選択し、正常に開始されたことを 確認します。

ログレベルが I はインフォメーション、S や E の場合はエラー表示されます。

メッセージID	∇ ×	(ッセージ	ブロセスID 〜	ログレベル~	Job Entry Subsystem (JES) services initialized
フロセスID	リージョン	メッセージID	ロクレベル	メッセージ	JES 5 digit iob numbering support enabled
	10105140				Communications interface 01 initialization started
11912	JCLDEMO	CASTS51151	1	ES ISC Log facility file: C:\work\JCLDEMO\system	IEC Initiates excepted for Secure ICI DEMO, excepted id = 0304
11912	JCLDEMO	CASTS5113I		ES TSC Log facility cold started	JES Initiator created for Server JCLDEWO, process-id = 9364
11912	JCLDEMO	CASTS5116		ES TSC Log facility file size: 4 blocks (316,23	Communications interface 01 initialization complete
11912	JCLDEMO	CASTS5117I	1	ES TSC Log facility is running with 4 buffers	Batch initiator INITABC: class(es) "ABC"
8184	JCLDEMO	JES000012I	I	Batch Spool files have been "cold-started"	
7648	JCLDEMO	CASTS1002I	1	ES TRC Service Process initialization complete	
11912	JCLDEMO	CASTS0002I	1	ES TSC Service Process initialization complete	
8184	JCLDEMO	JES000051I	I	Job Entry Subsystem (JES) services initialized	
8184	JCLDEMO	JES0000591	1	JES 5 digit job numbering support enabled	
4852	JCLDEMO	CASCS5001I	1	Communications interface 01 initialization started	
		CASCD1060I	1	JES Initiator created for Server JCLDEMO, process	-id = 9364
4852	JCLDEMO	CASCS5003I	1	Communications interface 01 initialization complete	
8184	JCLDEMO	CASBJ0023I	1	Batch initiator INITABC: class(es) "ABC"	
-					
Ξ.					



3.10 JCL の実行

現在 JCLDEMO 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。まずは簡単な JCL を実行してみます。

3.10.1 COBOL エクスプローラー内にある JCLDEMO プロジェクト配下の copy1.jcl をダブルクリックし、エディタ で内容を確認します。この JCL は IEBGENER ユーティリティを使用して、JCL 内に書かれたインラインデータ を SYSOUT に書き出しています。

😤 COB 🙁 📃 サーバ 🖳 🗖	ザーパー: localhost	
Image: Cost of the system I	copyljcl //copyl JoB MSGCLASS=A //copyl JOB MSGCLASS=A //copyl JOB MSGCLASS=A //sySPRINT DD SYSOUT=* //sySPRINT DD SYSOUT=* //sySPRINT DD SYSOUT=* //sySPRINT DD SYSOUT=* //sySUT1 DD * 00002Ryotaro Shiba 2-3,Sonezaki,Kita-ku,Osaka-shi,Osaka-fu 00002Ryotaro Shiba 2-6,TSugaru,Tsugaru-gun,Aomori-ken 00006Jirocho Shimizu 2-6,Jiro-cho,Shimizu-shi,Shizuoka-ken 00006Jirocho Shimizu 00009Shiki Masaoka 00009Shiki Masaoka 00009Shiki Fukuzawa 2-5,DOg Onsen,Matsuyama-shi,Ehime-ken 8-8,Keio-cho,Nakatsu-shi,Oita-ken //* //s*	1886 1900 1911 1911 1920 1800 1886 1820 1870 1835

3.10.2 COBOL エクスプローラー内の copy1.jcl を右クリックして [Enterprise Server へのサブミット] を選択す ると、この JCL が実行されます。



3.10.3 ESCWA からスプールを確認します。JCLDEMO インスタンスを選択後、[JES] プルダウンメニューから [スプ ール] を選択します。



スプ-

3.10.4 フィルタ機能で [すべて] が指定されていることを確認後、[リスト] ボタンをクリックして一覧を表示します。

フィルタ	
* 入力必須の項目です	
*	
¥* 8	
*	
ジョブ番号* 💡	
*	
クラス* 8	
*	
COND* 8	
*	
出力タイプ	キュー
✓ すべて	w th O
	🗖 ШЛІ А
✓ 入力の休留 ¥ ✓ ディスパッチ Q	✓ 出力の保留 🖓
■ アクティブ Q	
☑ 完了 🖓	M Frank, A



3.10.5 実行した JOB 番号のスプールをダブルクリックして内容を表示します。

COPY1	J0001007 A mfuser 0	0000	DD名~	ステップ〜 『	PROCステップ〜	状態∨	クラス〜	
			状態	クラス	DD名	ステップ	ステップ番	PROCステ レコード数
			🗅 Hold	А	JESYSMSG		0	31
			🗋 Ready	А	SYSPRINT	S1	1	4
			🗅 Ready	А	SYSUT2	S1	1	10

3.10.6 先頭の [JESYSMEG] をダブルクリックしてジョブログを確認すると、正常に終了していることが確認できます。

---> 14:25:30 JCLCM0191I STEP ENDED 51 - COND CODE 0000

---> 14:25:30 JCLCM0182I JOB ENDED - COND CODE 0000

3.10.7 右上にある [戻る] ボタンをクリックしてスプール一覧に戻り、[SYSPRINT] をダブルクリックすると、 IEBGENER ユーティリティの実行ログが記録されていることが確認できます。

Micro Focus MFJGENER Utility Version ED10.0_001 Copyright (C) Micro Focus 1997-2023. All rights reserved.

JCLGN01101(00) - 0000000010 RECORDS COPIED FROM SYSUT1 TO SYSUT2 ***END OF FILE***

3.10.8 右上にある [戻る] ボタンをクリックしてスプール一覧に戻り、 [SYSUT2] をダブルクリックすると、出力された スプールの内容が確認できます。

00001Soseki Natsume 1-1,Koishikawa,Bunkyo-ku,Tokyo-to 1886 2-3, Sonezaki, Kita-ku, Osaka-shi, Osaka-fu 00002Ryotaro Shiba 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, Miyamotomura, Mimasaka-gun, Okayama-ken 1920 00006Jirocho Shimizu 6-6.Jiro-cho.Shimizu-shi.Shizuoka-ken 1800 3-1,Rintaro-cho,Tsuwano-shi,Shimane-ken 1886 000070gai Mori 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

3.11 プロシージャライブラリの作成

プロシージャを使用する JCL を実行するために、プロシージャライブラリを作成します。Enterprise Server インスタンス ではプロシージャを区分データセットのメンバーとして配置します。前項で作成したプロシージャライブラリとなるフォルダをカ タログします。

DATAFILE
 SYS1.PROCLIB
 SORTD.prc

3.11.1 ESCWA へ移動して JCLDEMO インスタンスの画面上部にある [JES] プルダウンメニューの [カタログ] を 選択し、[リスト] ボタンをクリックすると、現時点では何も登録されていないことがわかります。新しくカタログす るために [新規作成] ボタンをクリックします。

カ	9ログ Q IJ スト	▼ フィルタ 🗴 前除	* 新規作成		
~	く 〉 ページ 1 1	00 個のエントリ			
	V LRECL	▼ ファイルサイズ	♥ BLKSIZE	RECFM ~	
	□ ア DS編成	DS名	LRECL	RECFM	ファイル サイズ



3.11.2 カタログエントリの入力画面が表示されますので、以下のように入力し [保存] ボタンをクリックします。

項目名	説明
DS 名	SYS1.PROCLIB を入力します。
物理ファイル	物理パスを指定します。ここでは前項で作成したフォルダパスを入力します。
DS 編成	区分データセットである PO を選択します。
RECFM	行順編成ファイルである LSEQ を選択します。
動的 PDS	プロシージャファイルをフォルダ配下に保持する動的 PDS の場合にオンにします。 ここではオンを指定します。
PDS 拡張	プロシージャファイル拡張子を指定します。ここでは PRC を入力します。

カタログ エントリを追加	保存
--------------	----

* 入力必須の項目です DS名* ♀								
SYS1.PROCLIB						カタログ式 Q		
物理ファイル* ♀								
C:\work\JCLDEMO\DATA	FILE\SYS1.PR	OCLIB						
DS編成 🛛		コードセット 8		LRECL* Ø	BLKSIZE* 9			
PO	~	ASCII	~	0	0	<u>्</u> री/रि		
RECFM 8		作成日時 🛛		最終参照日時 🛛	有効期限・♀			
LSEQ	~				0000/000			
MGMTCLASS 🖗		ファイルサイズ 🛛						
		0バイト						
				PDS拡張子				
✔ 動的PDS				PRC				

3.11.3 カタログ一覧に戻り [リスト] ボタンをクリックすると作成した PO が表示されます。[SYS1.PROCLIB] をク リックすると配置されているメンバーが確認できます。

_		DO	CVC1 PDOCUP		
Ш	8]	PO	SYST.PROCLIB		O 23 II
					表示
					344
	B	DO	SVS1 DDOCLIP		S B B
	61	FU	STSL-ROCLID		
	8]	PDSM	SYS1.PROCLIB(SC	(RTD)	
					表示
[S	YS1.	PROC	LIB(SORTD)]	の [表示] アイコンを	クリックすると内容が確認できます。
表	示	« <)	▶ ページ 1 10000	行 コードセット ASCII ~	
	//SORTD	PROC	OCM-FORT		
	//SVS00	T=*	UN=SUKT		
	//SORTW	KØ1 DD	UNTT=SYSDA SPACE=(CVI	(10, 10))	
	//sontu		UNIT-SYSDA SDACE-(CYL	(10,10))	
1	// DEND	NO2 DD	UNIT-STODA, SPACE=(CTL)	(10,10))	
	// PEND	05 5715**	*		
	END	OF FILE**			



3.12 COBOL バッチプログラムの実行

Eclipse に戻り、COBOL プログラムを含む JCL を実行してみます。

3.12.1 COBOL エクスプローラー内にある JCLDEMO プロジェクト配下の vsamwrt2.jcl をダブルクリックし、エディ タで内容を確認します。



ステップ 1:DEFVSAM1

IDCAMS を使用して VSAM データセット JINJI.KSDS を削除し、クラスターを持つ KSDS として再作成します。

② ステップ 2:SORTSTEP

前項で登録した SORTD.prc を使用して、JINJI.KSDS ファイルへの書き込み用データをソートします。

③ ステップ 3:APPL1

アプリケーション KSDSWRT2 を呼び出しステップ2でソートされたデータをステップ1で定義した VSAM クラスターに書き込みを行います。同時に書き込まれたデータを DD=PRINTER へ出力します。

- ④ ステップ 4:VERIFY1
 出力内容確認のため、IDCAMS の REPRO で内容を出力します。
- 3.12.2 COBOL エクスプローラー内の vsamwrt2.jcl を右クリックして [Enterprise Server へのサブミット] を選択し、この JCL を実行します。
- 3.12.3 ESCWA で JCLDEMO インスタンスを選択後、[JES] プルダウンメニューから [スプール] を選択します。





3.12.4 実行した JOB 番号のスプールをダブルクリックして内容を表示します。

_ i)	VSAMWRT2	J0001006 A	mfuser	0000		DD	å∨ .	ステップ〜 PRC	cステップ〜	状態~	ラスマ		
						1	状態	クラス	DD名	ステップ	ステップ番	PROCステ	レコード数
					I		Hold	A	JESYSMSG		0		76
					[Ready	A	SYSPRINT	DEFVSAM1	1		18
					(D 1	Ready	А	SYSOUT	SORTSTEP	2	SORT1	12
					[D 1	Ready	А	SYSOUT	APPL1	3		1
					[D 1	Ready	A	PRINTER	APPL1	3		10
					[Ready	А	SYSPRINT	VERIFY1	4		41

初回はステップ1で削除するファイルが見つからないために [COND CODE] へ [0008] が返却されますが、次回以降は [0000] が返却されます。

3.12.5 [JESYSMSG] の内容を確認すると、各ステップの COND CODE が確認できます。ジョブが異常終了した場合 にはここでエラーの原因を調査することができます。

---> 13:23:50 JCLCM0191I STEP ENDED STEP04 - COND CODE 0000

---> 13:23:50 JCLCM0182I JOB ENDED - COND CODE 0000

3.12.6 右上にある [戻る] ボタンをクリックしてスプール一覧に戻り、DEFVSAM1 ステップの [SYSPRINT] をダブ ルクリックして内容を確認します。

JCLAM0114I(00) - ENTRYNAME DELETED [JINJI.KSDS]

SET LASTCC=0 JCLAM0140I(00) - LASTCC set to 0.

DEFINE CLUSTER (NAME(JINJI.KSDS)) -DATA -(RECORDS(10) FREESPACE(20 10) KEYS(5 0) -RECORDSIZE(71 71)) -INDEX (RECORDS(56 50)) JCLAM0113I(00) - ENTRYNAME DEFINED [JINJI.KSDS]

3.12.7 右上にある [戻る] ボタンをクリックしてスプール一覧に戻り、SORTSTEP の [SYSOUT] をクリックしてソート内容を確認します。

SORT FIELDS=(1,5,CH,A)

SORT2041:	********************** ソート結果	*****
SORT2051:	INPUT ファイル 'SORTIN'	
	入力レコード	10 件
	使用レコード	10 件
SORT206I:	OUTPUT ファイル 'SORTOUT'	
	使用レコード	10 件
	出力レコード	10 件
SORT399I:	Micro Focus MFJSORT ユーティリテ	ィ終了
END OF	FILE	

3.12.8 右上にある [戻る] ボタンをクリックしてスプール一覧に戻り、他ステップに関しても確認してみてください。

3.12.9 前項と同様の手順で、この JOB によってカタログされた情報を確認します。カタログ一覧で [リスト] ボタンをク リックすると、VSAM ファイルの JINJI.KSDS が新たにカタログされていることが確認できます。



このカタログ情報にカーソルを合わせ、[DCB] アイコンをクリックすると登録情報が表示されますので、内容を確認します。

	JINJI.KSDS	
	SYST PROCEIR	DCB
また、一覧の[表示 ます。] アイコンもしくは DCB 情報	の [表示] ボタンをクリックすると、ファイルの内容が表示され
JINJI.KSDS I 連用 コピー	- リネーム (1) 削除	
・入力必須の項目です DS名 JINJI.KSDS 物理ファイル・♀ C:WWORKVJCLDEMOVDATAFILE	JINJI.KSDS.DAT	🛛 ಗಳದಲೆಸ 🖗
DS編成 Q VSAM ~	□-ドセット Ø LRECL* Ø ASCII ~ 71	CITTE
表示 《 《 》 《->	1 10000 行 コードセット ASCII ~	□ 詳細 □ ページ方向を反転
00001Soseki Natsume 00002Ryotaro Shiba 00003Hideyo Noguchi 00004Osmu Dazai 00005Eiji Yoshikawa 00005Eiji Yoshikawa 00005Eiji Yoshikawa 00005Gai Mori 00008Ryoma Sakamoto 00009Shiki Masaoka 00010Yukichi Fukuzawa ***END OF FILE***	1-1,Koishikawa,Bunkyo-ku,Tokyo-to 2-3,Sonezaki,Kita-ku,Osaka-shi,Osaka-fu 5-1,Inawashiro,Aizu-shi,Fukushima-ken 2-6,Tsugaru,Tsugaru-gun,Aomori-ken 9-3,Miyamotomura,Mimasaka-gun,Okayama-ken 6-6,Jiro-cho,Shimizu-shi,Shizuoka-ken 3-1,Rintaro-cho,Tsuwano-shi,Shimane-ken 1-1,Harimayabashi,Kochi-shi,Kochi-ken 5-5,Dogo Onsen,Matsuyama-shi,Ehime-ken 8-8,Keio-cho,Nakatsu-shi,Oita-ken	1886 1900 1911 1911 1920 1800 1820 1820 1870

3.13 COBOL バッチプログラムのデバッグ

Eclipse に戻り、JCL から実行される COBOL プログラムをデバッグします。

3.13.1 [実行] プルダウンメニューの [デバッグの構成] を選択します。

<u>بر</u>	ース	ナビゲート(N)	検索	プロジェクト(P)	実行(R)
R	実行	テ点をリセット			
Q,	実行	, (R)		Ctr	1+F11
₩ь	デバ	ッグ(D)			F11
	実行	亍履歴(T)			+
	実行	, (S)			+
	実行	亍構成(N)			
	デバ	ッグ履歴(H)			+
	デバ	「ッグ(G)			+
	デハ	ッグの構成(B)			

3.13.2 左側のメニューから [COBOL Enterprise Server] を選択して、左上の [新規の起動構成] アイコンをクリックします。





3.13.3 [COBOL プロジェクト] ヘ対象となる JCLDEMO プロジェクトを入力し、[Enterprise Server] ヘ実行させ る JCLDEMO インスタンスを指定します。

[デバッグの種類]は「JCL」を選択した状態で [デバッグ] ボタンをクリックします。

名前(N): 新規構成	
▶ 一般 与 ソース □ 共通 与 デバッグシンボル 🥒 コンテナー	
 COBOL プロジェクス」 シーバス (シーバス) (シー	参照
サーバー Iクスプローラーの設定 ESCWA: LOCAL MFDS: WIN11-SVR リージョン JCLDEMO	
 ▼デパッグの種類 タイブ: JCL 	参照
	A 町(N): 新規構成 パー般 リーノス □ 共通 リーブ(ワグシンボル ● コンテナー Enterprise Server と でデバッグセンションを開始して、COBOL プログラムの起動を待機します。 ・ COBOL プロジェクト(P) J/CLDEMO ・ Enterprise Server 接続: サーバー エクスプローラー サーバー エクスプローラーの設定 ESCWA: LOCAL MFDS: WIN11-SVR リージョン バンローラー アデバッグの種類 タイブ: J/CL JCL 設定 (空白の場合はすべての JCL ジョブをデバッグ)

3.13.4 パースペクティブの切り替え確認ウィンドウが表示されますが、COBOL エクスプローラーから JCL を実行する ため、ここでは [いいえ] ボタンをクリックします。

● パーフ	ペクティブの切り替えの確認 ×
?	この種類の起動では、開始時にデバッグ パースペクティブを開くように構成します。
	このデバッグ・パースペクティブは、アプリケーションのデバッグをサポートするように設計されています。これには、デバッ グ・スタック、変数、およびプレークポイント管理を表示するビューが組み込まれています。
	今パースペクティブを開きますか?
口設知	26保存
	(สากา การสากา

- 3.13.5 デバッグタブで [アタッチ待機] 状態になったことを確認します。
 - ☆ デバッグ ☆
 ぷ 新規構成 [COBOL Enterprise Server]
 ② COBOL デバッガ: (アタッチ待機)
- 3.13.6 COBOL エクスプローラー内の vsamwrt2.jcl を右クリックして [Enterprise Server へのサブミット] を選択して、JCL を実行します。
- 3.13.7 再度、パースペクティブの切り替え確認ウィンドウが表示されますので、ここでは [切り替え] ボタンをクリックし、 デバッグ用のパースペクティブを開きます。

● パースペクティブ切り替えの確認							
?	This kind of launch is configured to open the ${ar au}/{ar u}{ar au}$ perspective when it suspends.						
	このデバッグ・パースペクティブは、アブリケーションのデバッグをサポートするように設計されています。これには、デバッ グ・スタック、変数、およびブレークボイント管理を表示するビューが組み込まれています。						
	Switch to this perspective?						
□ 常にこの設定を使用する(B)							
	切り替え(5) いいえ(11)						

3.13.8 少し待つとデバッグセッションが開始して、プログラムのステップ実行が可能になります。[F5] キーもしくは [実 行] 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、変数タブでは使用している 変数の値が確認できます。



ファイル(E) 編集(E) ソース リファクタリング ナビゲート(N) 検索	ガロジェクト(巴) 実行(B) ウインドウ(M) ヘルプ(Ð						
🗖 🕶 🔛 🐘 🐘 🗣 🕶 🔛 🕨 💷 🗱 🌼) + 3, 10, 10, -11, 10, -13, 20 🗾	📲 🗣 📴 🗉 🔍 🐐 ד 🔾 ד 💁 ד 🍋	• = 😂 😂 🛷	• 요 • 원 • 원 • 수	• => •			
					ック・アクセス 🔡 🛃	e 🛛 🕸		
巻デパッグ 😂 🍋 プロジェクト・エク 楽 サーバー 📍 🗖	サーバー: localhost 💽 KSDSWRT2.cbl 응용		- 8	00- 変数 23 代 式	🖄 🕫 E	~		
% p ⊽	R KEDEMETA			4前 ✓ ◆ INREC	10	^		
✓ 1 新規模式 [COBOL Enterprise Server]					00001Soseki N			
✓ 2 Micro Focus デパッガ: (一時停止)	01 FSTAT-K PIC XX.	01 PSTAT-K PIC XX. 01 PSTAT-I PIC XX. 01 PSTAT-PIC XX. 01 PSTAT-K. 05 VSIAN-RETURN-CODE PIC 9(2) COMP.	^	INAKEY	00001			
✓ ● アブリケーションスレッド: 6712 (一時停止)	01 FSTAT-I PIC XX.			INNAME1	Soseki Natsu			
C#work50#JCLDEMO#New_Configuration.bin	BI PSTAT-P PIC XX.			INADDR1	1-1,Koishikaw			
G#work50#JCLDEMO#New_Configuration.bin	05 VSAN-RETURN-CODE			INBKEY	1886	_		
> アノリケーション スレット: 5460 (一時停止)	05 VSAN-COMPONENT-COD	E PIC 9(1) COMP.		V SKREC	00001Soseki N			
) (アノリリーンヨンスレフド:4052(一時時止) の 2010年、ションスレード:4052(一時時止)	05 VSAN-REASON-CODE	PIC 9(3) COMP.		AKEY	00001			
> (m 1777)-242 X071:56/6(一時停止)	OPEN INPUT INDATA			NAME1	Soseki Natsu			
	OPEN OUTPUT INDEXFIL	OPEN OUTPUT INDEXFILE.		AUUK1	1-1,Koishikaw			
	OPEN OUTPUT PRTFILE.	OPEN OUTPUT PRTFILE.		BKEY	1886	~		
	IF "00" NOT = FSTAT-:	E OR FSTAT-K OR FSTAT-P		Soseki Natsume				
	CONTINUE:	THEN CONTINUE; ELSE		168:				
	ELSE			50/000240///00222222				
	PERFORM PROC1 TH	PERFORM PROC1 THRU PROCEND1; END-IF.						
	END-IF.			 ブレークボイント EE プログラムアウトライン ※ 				
	DISPLAY "** OPEN ERR	COSTAV" OPEL EROR STOP RR. 		v 🍕 🔍				
	STOP RUN.			Input-Output Section				
	PROC1.			✓ Sata Division				
	PERFORM UNTIL LOOP1			> 🚏 File Section				
	AT END			> Storage Section				
	SET LOOP1 TO T			v 🗞 Procedure Divis	tion			
	NOT AT END			PROCESS-EN	4D			
	NOVE INREC TO I	REC	~	PROC1				
	<		>	PROCEND1		~		
	🗳 32/9-1/2 23 🖹 開題 📋 Debug Shell 🍟 CICS F v 利ル 🕴 👻 👘 📑 🖤 😁 🗖							
	デバッグメッセージ コンソール: (新規編成) [プロセス ID: 6732]							
	aded: C:\Program Files (x86)\Micro	Focus\Enterprise Developer\bin64\MFFH.dll	\EXTSM- Not co	sepiled for debugging		^		
	aded: C:\Program Files (X86)\Hicro	Focus\Enterprise Developer\bin64\DSNRLL.d	.dll\casumdt	: Compiled for debugg	ing			
	aded: C:\Program Files (x86)\Micro	Focus\Enterprise Developer\bin64\mfjmvs.d	11\mfjxxit- No	t compiled for debug	ging			
	Loaded: C:\Program Files (x86)\Hicro Focus\Enterprise Developer\bin64\mfjmvs.dll\mfjxe81- Not compiled for debugging							
	Loaded: [User Module]C:\workS0/ZCLDEMO\Wew_Configuration.bin\KSDWRT2.grt- Symbols loaded from: C:\workS0/ZCLDEMO\Wew_Configuration.bin\KSD							
	aueu: c:\riogram riles (X86)\Hicro	rocus (circerprise beveloper\blnb4\le370.dl	TOULCEDUL- N	or comprised for debug	8108			
· · · · · · · · · · · · · · · · · · ·								

- 3.13.9 希望のステップの左端をダブル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。
 - OPEN OUTPUT PRTFILE.
 COBOL 行ブレークボイントNOT = FSTAT-I OR FSTAT-K OR FSTAT-P
- 3.13.10 先に進める場合は画面上部の再開アイコンをクリックします。



3.13.11 デバッグを終了させるため、画面上部の終了アイコンをクリックします。

`≈ ₽►		N	3	Q	R	R	R.		
終了(T) (Ctrl+F2 直接アクセス)									

3.14 Enterprise Server インスタンスの停止

3.14.1 COBOL パースペクティブへ戻り、JCLDEMO インスタンスを停止します。



3.14.2 ESCWA から JCLDEMO インスタンスの停止を確認後、Eclipse を終了します。

■ JCLDEMO Region Stopped ✓ ✓ デフォルト

4 免責事項

本チュートリアルの例題ソースコードは機能説明を目的としたサンプルであり、無謬性を保証するものではありません。例題ソ ースコードは弊社に断りなくご利用いただけますが、本チュートリアルに関わる全てを対象として、二次的著作物に引用する場 合は著作権法の精神に基づき適切な扱いを行ってください。

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。