

Enterprise Developer チュートリアル メインフレーム COBOL 開発:スプール プリンター出力 Eclipse 編

1. 目的

本チュートリアルでは JOB 実行後に出力されるスプール内容を直接プリンターへ出力する手順と、SYSOUT リソース制御 方法の習得を目的としています。

2. 前提

- 本チュートリアルで使用したマシン OS : Windows 11 Pro
- 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
- JCL チュートリアルを完了していること
 補足) 完了していない場合は JCL チュートリアルを先に実施してください。

3. チュートリアル手順の概要

- 1. デフォルトプリンターへの出力
- 2. プリンター出口モジュールの作成
- 3. Enterprise Server インスタンスの設定
- 4. JCL の実行
- 5. 出力フォントの変更
- 6. プリンター出力指示
- 7. Enterprise Server インスタンスの停止

3.1 デフォルトプリンターへの出力

実施するマシンに接続されているプリンターへスプールの内容を出力します。

注意)Web 画面からユーザーを指定せずに Enterprise Server インスタンスを開始した場合のユーザーはデフォルト 値である SYSTEM となっています。 JCL が正常に実行されてスプールが印刷済ステータスになっても SYSTEM ユー ザーがプリンター出力権限を持たない場合は出力されません。

- 1) JES 機能を持つ Enterprise Server インスタンスを開始します。ここでは JCL チュートリアルで使用した JCLDEMO インスタンスを使用します。
- 2) ESCWA へ移動して、JCLDEMO の [編集] アイコンをクリックします。

	名前	タイプ	ステータス	<u>64</u> ビット	MSS有効	セキュリティ PAC	
=	JCLDEMO	Region	Stopped	\checkmark	\checkmark	デフォルト	Ø 🖗 🗓
							編集

3) 画面上部の [JES] プルダウンメニューから [構成] を選択し、[プリンター] の [新規作成] ボタンをクリックします。

プリンター	С	* 新規作成
-------	---	--------

4) 下記項目を指定して [保存] をクリックします。

項目名	設定値
名前	出力先を指定します。
クラス	出力したいクラスを指定します。
説明	任意で入力します。

JESプリンター	
名前 • ♀	
Microsoft XPS Document Writer	
クラス Q A	
出口モジュール 🛛	
Printer Exit	
✓ 有効 ② 説明 ②	
・入力必須の項目です	/i
保存	戻る

- 5) JCLDEMO インスタンスを起動します。
- 6) コンソールログを表示して、追加したプリンター出力が正常に設定されていることを確認します。

CASBJ0021I	1	Batch printer started for print output classes "A"
CASSI1600I	1	SEP initialization completed successfully
CASBJ0005I	1	Batch initiator started for job classes "ABC"
JES0000291	1	Locating batch printer: Microsoft XPS Document Writer
JES0000291	1	Available batch printer: OneNote for Windows 10
JES0000291	1	Available batch printer: tok-printer2
JES0000241	T. Contraction	Batch Printer: Microsoft XPS Document Writer

- 7) この設定により、クラス A で実行された JOB 結果は全て指定プリンターへ出力されることになります。
 //COPY1 JOB MSGCLASS=A //S1 EXEC PGM=IEBGENER //SYSPRINT DD SYSOUT=* //SYSUI1 DD *
- 8) A クラスで実行される JCL をサブミットすると、設定通り XPS を保存するために下記画面が表示されます。

 回 印刷結果を名前を付け 	て保存	×
\leftrightarrow \rightarrow \land \uparrow	PC > ۲キュメント > نال کار	の検索
整理 ▼ 新しいフォル	Ø-	== • ?
 PC 3D オブジェクト ダウンロード デスクトップ ドキュメント ビクチャ 	◆ 名前 ヘ 更新日時 ■ ISCKPTESS 2014/19/25 25000 Micro Focus 2021/11/26 16:32 ○ Micro Focus User 2018/09/13 16:47 ○ My Web Sites 2014/03/25 9:06 ○ SQL Server Management Studio 2021/08/23 17:30	種類 アパロン フォルター ファイル フォルダー ファイル フォルダー ファイル フォルダー ファイル フォルダー ファイル フォルダー メ
ファイル名(N): ファイルの種類(I): 0 ヘ フォルダーの非表示	penXPS ドキュメント (*.oxps) 保存(5)	~ ~ キャンセル

- 9) Enterprise Server インスタンスを停止するまでの一時的な使用としてプリンターを設定することも可能です。
 - ESCWA 画面上部の [JES] プルダウンメニューから [コントロール] を選択すると、[JES イニシエータとプリ ンターの開始] 項目が表示されます。

sイニシエータとブリンターの開始		
クラス 🖇	出口モジュール/ MPR名 🕅	タイプ Q イニシエータ v 開始
プリンター名:		
OneNote for Windows 10		
tok-printer2		
Microsoft XPS Document Writer		
Microsoft Print to PDF		
Fax		



② [クラス] ヘプリンター出力を行いたいジョブクラスを、[タイプ] には [プリンター] を選択、[プリンター名] には出力先を指定して、[開始] ボタンをクリックすることにより即時に開始されます。この設定は JCLDEMO インスタンスが終了するまで継続されます。

クラス Q B	出口モジュール/MPR名 🛛	タイプ 🛿 プリンター	×	開始
プリンター名:				
OneNote for Windows 10				
tok-printer2				
Microsoft XPS Document Writer				
Microsoft Print to PDF				
Fax				

- 10) Enterprise Server インスタンスが稼働中でも、開始されている JES イニシエータ や プリンターを停止することも可能です。
 - ① ESCWA の [モニター] プルダウンメニューから [SEP] を選択して [JES プロセス] 項目を表示します。

<u></u> 9.	イプ〜 プロ	コセスルマン ジョフ	ブ番号∨ JESク	フラスマ ジョ	ブ数〜					
	タイプ	プロセスル	ジョブ番号	JES クラス	ジョブ数	開始時刻	存続時間	デバイス	EXIT	
Ş	JES Initiator	6156	26	ABC	2					
7	JES Printer	5540	28	A	26			Microsoft X	((none)	

② 停止したいプロセス行の [停止] アイコンをクリックします。

	タイプ	プロセスル	ジョブ番号	JES クラス	ジョブ数	開始時刻	存続時間	デバイス	EXIT	
Ç	JES Initiator	6156	26	ABC	2					
Ç	JES Printer	5540	28	А	26			Microsoft X	(none)	⊚ ×
										停止

③ プロセスは停止し、一覧から削除されます。

3.2 プリンター出口モジュールの作成

プリンター出口モジュール sampprnx.cbl を利用すると、JCL に含まれている OUTPUT 文の情報を他のサードパー ティまたは外部の印刷サブシステムで使用するために連携することができます。

1) 製品にはあらかじめプリンター出口モジュールが用意されていますので、これを作業可能な場所へコピーします。

製品パス例)C:¥Program Files (x86)¥Micro Focus¥Enterprise Developer¥src¥enterpriseserver¥exits

→ « src → enterpriseserver → exits 名前 図 sampprnx.cbl



2) 下記のようなサンプル JCL で OUTPUT 指定を確認します。

//COPY1 JOB MSGCLASS=B	n	
7/ST EXEC PGM=TEBGENER	π.	
//SYSPRINT DD SYSOUT=*		
//SYSUT1 DD *		
00001Soseki Natsume	1-1,Koishikawa,Bunkyo-ku,Tokyo-to	188
00002Ryotaro Shiba	2-3.Sonezaki.Kita-ku.Osaka-shi.Osaka-fu	190
00003Hideyo Noguchi	5-1.Inawashiro.Aizu-shi.Fukushima-ken	191
000040samu Dazai	2-6.Tsugaru.Tsugaru-gun.Aomori-ken	191
00005Eiii Yoshikawa	9-3.Miyamotomura.Mimasaka-gun.Okayama-ken	192
00006 lirocho Shimizu	6-6. liro-cho.Shimizu-shi.Shizuoka-ken	180
000070gai Mori	3-1.Rintaro-cho.Tsuwano-shi.Shimane-ken	188
000088yoma Sakamoto	1-1 Harimayahachi Kochi-chi Kochi-kan	182
0000000 oma bakamoto	5-5 Dage Opeen Meteuvene-shi Ehime-ken	102
00010Vuliahi Euluranna	0-0,00g0 Onsen,Matsuyama-shi,Enime-ken	107
UUUTUTUKIChi rukuzawa	o-o,Kelo-cho,Nakatsu-shi,Ulta-ken	103
/*		
VOULT OUTPUT DEFAULT=YE	S,DEST=COMPLEX1,FORMS=BILLING,COPIES=2	
//SYSUT2 DD SYSOUT=A,OUTI	PUT=*.OUT1	
//SYSIN DD DUMMY		

3) JCL からの設定値を確認するため sampprnx.cbl ヘコードを追加します。

【指定した COPIES が格納されている変数:mvsdout.cpy → idadoutp.cpy】 7 ()-copies-str.← 9 ()-copies pic x(1) comp-x occurs 9.

【working-storage section へ追加するコード】

01 TEST-WORK PIC 9(01).

【process-output-statements section へ追加するコード】

*に囲まれているコードを追加します。



4) 出口モジュールをコンパイルするため、コマンドプロンプトを管理者として実行します。

注意)Enterprise Server インスタンスのビット数と一致するビット数のコマンドプロンプトを使用する必要があります。

0:5.	Enterprise Developer コマンドプロンプト (32-bit)		······································
C:1.	Enterprise Developer コマンドプロンプト (64-bit)	[🕲	管理者として実行(A)

5) コピー先の出口モジュールが存在するパスへ移動します。

例)cd C:¥work¥printexit

- 6) 下記コマンドで sampprnx.cbl ファイルを gnt 実行形式へコンパイルします。
 - 例) cobol sampprnx dialect(MF) gnt;



7) sampprnx.gnt が作成されました。

sampprnx.cbl
sampprnx.gnt

8) sampprnx.gnt 実行形式ファイルをインスタンスの JES タブに指定している JES プログラムパス へ配置する か、

製品フォルダ配下の bin64 フォルダヘコピーして配置します。

注意) 32 ビットの場合は bin フォルダヘコピーします。

JESの構成 C					
			« Enter	prise Develo	per → bin64
JES プログラムパス 🛛			^	名前	~
\$proj\New_Configu	ration.bin	- OR		amp 🖉	prnx.gnt

3.3 Enterprise Server インスタンスの設定

前項で追加した Enterprise Server インスタンスの JES の プリンターヘプリンター出口モジュールを設定します。

- 1) JCLDEMO インスタンスを停止します。
- 2) [出口モジュール] へ 前項で設置した sampprnx を入力して [保存] ボタンをクリックします。



3) JCLDEMO インスタンスを開始します。コンソールログに出口モジュールが正常にロードされていることを確認しま す。

JES000024I	1	Batch Printer: Microsoft XPS Document Writer
JES000026I	1	Exit Module: sampprnx
CASKC0025I	1	EXIT - Printer Initializing



3.4 JCL の実行

前項で作成した JCL を Enterprise Server インスタンスのコントロール画面からサブミットし、結果を確認します。

- 1) ESCWA の [JES] プルダウンメニューから [コントロール] を選択します。
- 2) 実行する JCL を選択して [サブミット] ボタンをクリックします。

ESにJCLをサブミット	
Please enter a JCL file name to upload.	
	クリックまたはドラッグ&ドロップしてファイルをアップロード
copy1.jcl ×	
〇 VSE	

3) 前項で出力先に [Microsoft XPS Document Writer] を指定しているため下記ウィンドウが表示されます。格 納したい場所を指定してください。

🧿 印刷結果を名前を付けて	て保存		×
$\leftarrow \rightarrow \checkmark \uparrow$	PC > ドキュメント > ~ ご	P 1/42X740)検索
整理 ▼ 新しいフォルタ	<i>9–</i>		
PC ·	▲ 名前 ▲	更新日時	種類 ^
🧊 3D オブジェクト	DB2LOG	2018/07/25 16:14	ファイル フォルダー
🕹 ダウンロード	FavBinEdit	2021/12/07 17:13	ファイル フォルダー
デスクトップ	IISExpress	2014/03/25 9:06	ファイル フォルダー
🔮 ドキュメント	Micro Focus	2021/11/26 16:32	ファイルフォルダー 🧹
■ ピクチャ	v <		>
ファイル名(<u>N</u>):			~
ファイルの種類(<u>T</u>): Op	eenXPS ドキュメント (*.oxps)		~
▲ フォルダーの非表示		保存(<u>S</u>)	キャンセル

4) 格納した oxps ファイルを表示します。JCL で設定した値が連携されていることを確認してください。

OUTPUT statement - CLASS: A DEST NODE: CONPLEX1 DEST USERID: FORM: BILLING FORMDEF: COPIES: 2 C:\WORK3\JCLDEMO\DATAFILE\MFE2016.S1011.S170948.J01200.D00003.SYSUT2.DAT 00001Soseki Natsume 1-1,Koishikawa,Bunkvo-ku,Tokvo-to 1886 00002Ryotaro Shiba 2-3, Sonezaki, Kita-ku, Osaka-shi, Osaka-fu 1900 00003Hideyo Noguchi 5-1, Inawashiro, Aizu-shi, Fukushima-ken 1911 000040samu 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 000070gai Mori 3-1, Rintaro-cho, Tsuwano-shi, Shimane-ken 1886 00008Ryoma Sakamoto 00009Shiki Masaoka 1-1,Harimayabashi,Kochi-shi Kochi-ken 1820 5-5,Dogo Onsen,Matsuyama-shi,Ehime-ken 8-8,Keio-cho,Nakatsu-shi,Oita-ken 1870 00010Yukichi Fukuzawa 1835

プリンター出口モジュール の詳細は製品ヘルプをご参照ください。

3.5 出力フォントの変更

プリンター出口モジュール sampprnx.cbl にコーディングされている出力フォントを変更することも可能です。

1) プログラムの固定値を変更します。

デフォルト値はコメント行にしてあります。有効な行が変更後の値です。

```
01 ws-font-family.
     03 value 11
                                pic x(2) comp-5.
     03 value 'Courier New'.
    03 value 7
                                pic x(2) comp-5.
    03
       value 'Calibri'.
```

- 2) 前項と同様にモジュールをコンパイルし、実行ファイルを配備します。
- 3) 新しいモジュールをロードさせるため、JCLDEMO インスタンスを再起動します。
- 4) 前項と同じ方法で同じ JCL を実行し、プリンター出力結果を比較してみます

```
【フォント変更前】
OUTPUT statement - CLASS: A DEST NODE: CONPLEX1 DEST USERID:
                        FORM: BILLING FORMDEF:
                        COPIES: 2
C:\WORK3\JCLDEMO\DATAFILE\MFE2016.S1011.S170948.J01200.D00003.SYSUT2.DAT
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
000040samu Dazai
                     2-6, Tsugaru, Tsugaru-gun, Aomori-ken
                                                         1911
00005Eiji Yoshikawa
                     9-3, Miyamotomura, Mimasaka-gun, Okayama-ken 1920
```

```
6-6, Jiro-cho, Shimizu-shi, Shizuoka-ken
00006Jirocho Shimizu
000070gai Mori
                          3-1, Rintaro-cho, Tsuwano-shi, Shimane-ken
                                                                      1886
00008Ryoma Sakamoto
                          1-1,Harimayabashi,Kochi-shi.Kochi-ken
                                                                      1820
00009Shiki Masaoka
                                                                      1870
                          5-5, Dogo Onsen, Matsuyama-shi, Ehime-ken
00010Yukichi Fukuzawa
                         8-8, Keio-cho, Nakatsu-shi, Oita-ken
                                                                      1835
```

【フォント変更後】

****************** OUTPUT STATEMENTS ****************** OUTPUT statement - CLASS: A DEST NODE: CONPLEX1 DEST USERID: FORM: BILLING FORMDEF: COPIES: 2

C:\WORK3\JCLDEMO\DATAFILE\MFE2016.S1011.S173315.J01202.D00003.SYSUT2.DAT

00001Soseki Natsume 1-1,Koishikawa,Bunkyo-ku,Tokyo-to 1886 00002Rvotaro 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, Miyamotomura, Mimasaka-gun, Okayama-ken 1920 00006Jirocho Shimizu 6-6, Jiro-cho, Shimizu-shi, Shizuoka-ken 1800 000070gai 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

1800

5) 出口モジュールで指定したフォントへ変更されていることが確認できました。

3.6 プリンター出力指示

プリンター出口モジュール sampprnx.cbl に任意のプリンターへの出力をコーディングします。

【シナリオ】

JES プリンターへ制御が渡った際に起動するプリンター出口モジュールで、DEST に CONP が含まれる SYSUT2 ファ イルを 任意のプリンターへ出力する。その際にはフォント名とフォントサイズを変更し、COPIES 数分プリントを行う。

1) JCL を変更します。SYSUT2 をプリンター出力するため DCB 情報を付加します。

//COPY1 JOB MSGCLASS=B		
//S1 EXEC PGM=IEBGENER	R	
//SYSPRINT DD SYSOUT=*		
//SYSUT1 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, Miyamotomura, Mimasaka-gun, Okayama-ken	1920
00006Jirocho Shimizu	6-6,Jiro-cho,Shimizu-shi,Shizuoka-ken	1800
000070gai 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
/*		
//OUT1 OUTPUT DEFAULT=YES	S, DEST=CONPLEX1, FORMS=BILLING, COPIES=2	
//SYSUT2 DD SYSOUT=A,O	JTPUT=*.OUT1,	
// DSN=TO.DA	TASET.NAME,DISP=(NEW,KEEP),	
// DCB=(DSOR	5=PS,RECFM=LSEQ,LRECL=80,BLKSIZE=8000)	
//SYSIN DD DUMMY		

- 2) プリンター出口モジュール sampprnx.cbl を変更します。
 - ① working-storage section へ使用する下記の変数と定数を定義します。

```
01 MyDocumentInfo.
  03 filename.
    05 len
                       pic x(2) comp-5.
    05 body
                     pic x(128).
  03 document.
   05 len
                     pic x(2) comp-5.
    05 body
                      pic x(128).
  03 document-flags pic x(4) comp-5.
  03 window-hwnd
                       pic x(4) comp-5.
78 USE-PROGRESS-DIALOG value 16.
01 <del>default-info</del>.
  03 def-option
                         pic x(4) comp-5.
  03 def-ourprinter.
    05 def-len
                           pic x(2) comp-5.
    05 def-body
                           pic x(128).
78 SET-DEFAULT-PRINTER value h"0001".
01 printer0-name
                           pic x(29)
             value "Microsoft XPS Document Writer".
                  pic 9(2) value 29.
01 printer0-len
                           pic x(12) value "tok-printer2".
01 printer2-name
01 printer2-len
                           pic 9(2) value 12.
01 printer-status-code pic 9(4) comp-5.
01 printer-num
                           pic 9(10).
01 wk-str1
                          pic x(70).
                  pic x(70).
pic 9(02) value zero.
pic 9(02) value zero.
01 inspect-count
01 inspect-count2
                          pic 9(02).
01 set-copy
01 copy-count
                           pic 9(02).
```



② procedure division へ下記コードを追加します。

```
OUTPUT 情報のアドレスを取得して、DEST に CONP が含まれているか、DD NAME に SYSUT2 が含
まれているかを検査します。
       set address of 1k-output to 1k-prn-output-record-ptr.
       move zero to inspect-count.
       INSPECT lk-output-dest-node
          TALLYING inspect-count FOR ALL "CONP".
       move zero to inspect-count2.
       INSPECT 1k-DD-NAME
          TALLYING inspect-count2 FOR ALL "SYSUT2".
両方に文字列が存在する場合はデフォルトプリンターへ任意のプリンターを設定します。
       if inspect-count > 0 and
          inspect-count2 > 0 then
           move printer2-name to def-body
           move printer2-len to def-len
           move SET-DEFAULT-PRINTER to def-option
           call "PC_PRINTER_SET_DEFAULT" using
                by value def-option
                by reference def-ourprinter
                returning printer-status-code
           end-call
デフォルトプリンターへフォントとサイズを設定します。
           call "PC PRINTER DEFAULT FONT" using
                by reference ws-font-family
                by value 12 *> font-size
                            0 *> font-style
                by value
           end-call
COPIES の数だけ任意のプリンターへ出力します。
           move lk-output-copies(1) to set-copy
           move 1 to copy-count
           perform until copy-count > set-copy
              move lk-prn-sysout-file-name to body of filename
              set len of filename to length of body of filename
              move lk-prn-document-title to body of document
              move lk-prn-document-len to len of document
              move USE-PROGRESS-DIALOG to document-flags
              move zero to window-hwnd
              call "PC PRINT FILE" using by reference filename
                                by reference document
                                by value document-flags
                                by value window-hwnd
                                returning printer-status-code
              end-call
              add 1 to copy-count
           end-perform
```



出力処理終了後、デフォルトプリンターを変更前の値へ設定します。

move printer0-name to def-body move printer0-len to def-len move SET-DEFAULT-PRINTER to def-option call "PC_PRINTER_SET_DEFAULT" using by value def-option by reference def-ourprinter returning printer-status-code end-call

- 3) 前項と同様のコマンドで sampprnx.cbl をコンパイルし、同様の手順で実行ファイルをフォルダへ配置します。
- 4) 新しいモジュールをロードさせるため、JCLDEMO インスタンスを再起動します。
- 5) 前項と同様の手順で JCL を実行します。
- 6) 結果を確認します。

前項と同様に [Microsoft XPS Document Writer] への出力が行われます。

C:\WORK3\JCLDEMO\DATAFILE\MFE2016.S1109.S145028.J01276.D00003.SYSUT2.DAT

00001Soseki Natsume1-1,Koishikawa,Bunkyo-ku,Tokyo-to188600002Ryotaro Shiba2-3,Sonezaki,Kita-ku,Osaka-shi,Osaka-fu190000003Hideyo Noguchi5-1,Inawashiro,Aizu-shi,Fukushima-ken191100004Osamu Dazai2-6,Tsugaru,Tsugaru-gun,Aomori-ken191100005Eiji Yoshikawa9-3,Miyamotomura,Mimasaka-gun,Okayama-ken192000006Jirocho Shimizu6-6,Jiro-cho,Shimizu-shi,Shizuoka-ken180000007Ogai Mori3-1,Rintaro-cho,Tsuwano-shi,Shimane-ken188600008Ryoma Sakamoto1-1,Harimayabashi,Kochi-shi.Kochi-ken182000009Shiki Masaoka5-5,Dogo Onsen,Matsuyama-shi,Ehime-ken187000010Yukichi Fukuzawa8-8,Keio-cho,Nakatsu-shi,Oita-ken1835

続いて任意のプリンターへ SYSUT2 ファイルの内容が 2 部出力されます。

00001Soseki Natsume	1-1,Koishikawa,Bunkyo-ku,Tokyo-to	1886
00002Ryotaro Shiba	2-3,Sonezaki,Kita-ku,Osaka-shi,Osaka-fu	1900
OOOO3Hideyo 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
000070gai 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



指定したフォントとサイズで SYSUT2 ファイルの内容が 2 部出力されました。

00001Soseki Natsume 00002Ryotaro Shiba 00003Hideyo Noguchi 00004Osamu Dazai 00005Eiji Yoshikawa 00006Jirocho Shimizu 00007Ogai Mori 00008Ryoma Sakamoto 00008Ryoma Sakamoto 00009Shiki Masaoka 00010Yukichi Fukuzawa	1-1,Koishikawa,Bunkyo-ku,Tokyo-to 1886 2-3,Sonezaki,Kita-ku,Osaka-shi,Osaka-fu 1900 5-1,Inawashiro,Aizu-shi,Fukushima-ken 1911 2-6,Tsugaru,Tsugaru-gun,Aomori-ken 1911 3-3,Miyamotomura,Mimasaka-gun,Okayama-ken 1920 6-6,Jiro-cho,Shimizu-shi,Shizuoka-ken 1800 1,Rintaro-cho,Tsuwano-shi,Shimane-ken 1886 1-1,Harimayabashi,Kochi-shi.Kochi-ken 1820 5-5,Dogo Onsen,Matsuyama-shi,Ehime-ken 1870 8-8,Keio-cho,Nakatsu-shi,Oita-ken 1835
	00001Soseki Natsume1-1,Koishikawa,Bunkyo-ku,Tokyo-to188600002Ryotaro Shiba2-3,Sonezaki,Kita-ku,Osaka-shi,Osaka-fu190000003Hideyo Noguchi5-1,Inawashiro,Aizu-shi,Fukushima-ken191100004Osamu Dazai2-6,Tsugaru,Tsugaru-gun,Aomori-ken191100006Jirocho Shimizu6-6,Jiro-cho,Shimizu-shi,Shizuoka-ken180000006Jirocho Shimizu6-6,Jiro-cho,Shimizu-shi,Shizuoka-ken180000007Ogai Mori3-1,Rintaro-cho,Tsuwano-shi,Shimane-ken182000009Shiki Masaoka5-5,Dogo Onsen,Matsuyama-shi,Ehime-ken183000010Yukichi Fukuzawa8-8,Keio-cho,Nakatsu-shi,Oita-ken1835

このようにプリンター出口モジュール sampprnx.cbl を使用して、JCL へ指定した値を引き継ぎ、かつ目的に沿った出力が可能になります。

3.7 Enterprise Server インスタンスの停止

JCLDEMO インスタンスを停止します。

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。