

---

# Multi LABELIST Web Component

リファレンスマニュアル  
(カラープリンタ編)

Ver.5.9.11.0

株式会社サトー

2024年2月28日

---

## はじめに

この度は、「Multi LABELIST V5 Web Engine」（以下、MLV5 Web Engine）をご購入頂きまして誠にありがとうございます。

MLV5 Web Engine は、弊社製汎用ラベル・タグ発行ソフトウェア「Multi LABELIST V5」(以下 MLV5) の資産を利用し、お客様のシステムにラベル/タグの発行・プレビュー機能を追加する開発支援ツールです。発行機能として「Multi LABELIST Web Component」（以下、MLWebComponent）、プレビュー機能として「Multi LABELIST Preview Component」（以下、MLPreviewComponent）を提供致します。

本マニュアルでは MLWebComponent でカラープリンタレイアウトを利用する際に必要なプロパティ・メソッドについてのみご説明しております。サトープリンタレイアウトを利用する場合は、機能別の利用方法「MLWebComponent リファレンスマニュアル」をご参照ください。

尚、カラープリンタレイアウトの利用には、「MLV5 WebEngine カラープリンタ発行機能追加」をご購入頂き、サトープリンタ・カラープリンタ両対応のライセンスファイルが必要です。

## ご注意

- 本マニュアルの一部または全部を弊社の許可なく複写・複製することは、その形態を問わず禁じます。
- 本マニュアルの内容は、訂正・改善のため予告なく変更することがあります。
- 本マニュアルを運用した結果の影響については責任を負いかねますのでご了承下さい。
- 本マニュアルの内容については万全を期しておりますが、万一ご不審な点やお気づきの点がございましたら、弊社までご連絡ください。
  
- SATO、Multi LABELIST は、サトーホールディングス株式会社の登録商標または商標です。
- Microsoft、Windows は、米国マイクロソフト社の登録商標です。
- その他記載されている会社名、製品名は各社の登録商標または商標です。

# 目次

はじめに.....	2
ご注意.....	2
<b>第1章 製品情報.....</b>	<b>6</b>
1-1.必要なハードウェア・ソフトウェア.....	7
■サーバ環境.....	7
■クライアント環境.....	7
■サポートプリンタ（カラープリンタレイアウト）.....	7
1-2.MLWebComponent 機能一覧.....	8
■プロパティ一覧.....	8
■メソッド一覧.....	8
<b>第2章 プロパティ.....</b>	<b>10</b>
2-1.[通信]Setting.....	11
2-2.[基本]LayoutFile.....	12
2-3.[基本]PrnData.....	13
2-4.[基本]PrnDataType.....	15
2-5.[動作]Offset.....	16
2-6.[動作]HeaderTailSetting.....	17
2-7.[動作]HeaderFile.....	18
2-8.[動作]TailFile.....	19
2-9.[特殊]LayoutNameCaption.....	20
2-10.[特殊]TotalQtyCaption.....	21
2-11.[特殊]TaxRate.....	22
2-12.[情報]Version.....	23
<b>第3章 メソッド・例外.....</b>	<b>24</b>
3-1.[通信]OpenPort.....	25
3-2.[通信]ClosePort.....	27
3-3.[発行]Output.....	28
3-4.[発行]OutputHeader.....	29
3-5.[発行]OutputTail.....	30
3-6.[ダウンロード]GetFile.....	31
3-7.[ダウンロード]GetFileCredential.....	33
3-8.[ダウンロード]SetProxyCredential.....	35
3-9.[ダウンロード]SetClientCertificate.....	36
3-10.[ダウンロード]GetClientCertCount.....	38
3-11.[ダウンロード]GetClientCertInfo.....	39
3-12.[情報]GetPrinter.....	40

3-13.[情報]GetLayoutFileProperty .....	41
3-14.[情報]IsColorPrinter .....	42
3-15.[データ]GetInputCount .....	43
3-16.[データ]GetInputFields .....	44
3-17.[データ]SetPrnDataField .....	46
3-18.[データ]GetPrnDataRowCount .....	48
3-19.[データ]GetPrnDataRow .....	49
3-20.[データ]SetPrnDataRow .....	50
3-21.[ドライバ]GetDriverNameList .....	52
3-22.[ドライバ]ClearPrintJob .....	53
3-23.[認証]Authenticate .....	54
3-24.[エラー]発行エラー .....	56
■例外エラーについて .....	60
<b>第4章 ワンポイントテクニック .....</b>	<b>62</b>
4-1.MLWebComponent を利用する .....	63
■object タグで配置する .....	63
■スクリプト上で宣言する .....	64
■ライセンス認証を行う .....	64
■サーバ環境を構築する .....	64
■クライアント環境を構築する .....	65
4-2.ライセンス認証を行う .....	66
■カラープリンタ利用ライセンス .....	66
■ライセンスファイルと認証キー .....	66
■ライセンス認証を行う .....	67
4-3.バージョンを確認する .....	68
■Version プロパティで取得する .....	68
■バージョン確認を行ってバージョンアップを通知する .....	68
■ファイルのプロパティで確認する .....	68
4-4.バージョンアップを行う .....	69
■サーバの WebComponent を更新する .....	69
■クライアントの WebComponent を更新する .....	69
4-5.データを一括で入力する .....	71
■入力順位 .....	71
■データ形式 .....	72
4-6.データを変数名で指定して入力する .....	73
■変数名 .....	73
4-7.動作設定ファイルを利用する .....	75
■ログファイルを出力する .....	77
4-8.同一ラベルの発行速度を改善する .....	79
■印刷部数機能を利用する .....	79

4-9. 発行時に文字や枠線などオブジェクトの色を指定する.....	81
■発行時にオブジェクトの色を指定する.....	81
■色の指定方法.....	84
<b>第5章 注意事項</b> .....	<b>85</b>
5-1.実行環境について.....	86
■ライセンスファイルと認証キーについて.....	86
■ライセンス認証について.....	86
■IEの保護モードについて.....	86
■IE10以降での利用.....	86
■Windows 7での利用.....	86
■.NET Frameworkのバージョンについて.....	86
■サーバサイドアプリケーションでの利用.....	87
■プリンタドライバの設定.....	87
■初回起動時・レイアウトファイルロード時の遅延.....	87
5-2.レイアウト情報について.....	88
■入力定義.....	88
■連番変数.....	88
■文字の使用について.....	88

## 第1章

# 製品情報

## 1-1

## 必要なハードウェア・ソフトウェア

■ハードウェア ■ソフトウェア

MLWebComponent をご使用いただくには、次のハードウェアとソフトウェアが必要です。

## ■サーバ環境

- ハードウェア Web サーバが動作する環境  
サポートプリンタ（ページ下部に記載）
- ソフトウェア 対応 OS 限定されません。Windows、Linux など。  
Web サーバ 限定されません。IIS、Apache など。

## ■クライアント環境

- ハードウェア 下記対応 OS が動作する環境  
5MB 以上の空き容量を持つハードディスク
- ソフトウェア 対応 OS Windows 11、Windows 10、Windows 8.1  
実行環境 .NET Framework\*1 4.5/4.6/4.7/4.8  
ブラウザ Microsoft Edge（IE モード）、Internet Explorer 11

\*1 実行環境に存在しない場合は、別途.NET Framework のインストールが必要です。（Ver.5.7.0.0 以降は必須）

<https://docs.microsoft.com/ja-jp/dotnet/framework/install/guide-for-developers>

古いバージョンの.NET Framework を利用すると、MLWebComponent が正常に動作しない場合があります。  
Microsoft でサポートされているバージョンをご利用ください。

<https://docs.microsoft.com/ja-JP/lifecycle/fag/dotnet-framework>

## ■サポートプリンタ（カラープリンタレイアウト）

サトー	GHS612R
ネットアンドプリント	LP-812W、NP-ODL8Ⅱ
ダイオーポスタルケミカル	NPC-122W、NPC-122X
フェニックス	PX509 $\alpha$ 、PX509
キヤノンマーケティングジャパン	LX-D5500、LX-P5500、LX-760、LX-760RF、LX-P1500
エプソン販売	CW-C4020 シリーズ、CW-C6500 シリーズ、 CW-C6000 シリーズ、GP-C831、 TM-C7500、TM-C3500、TM-C3400
カシオ計算機	GE5000-YPOS、N6100、V2500 RPCS
日本エレクトロニクス工業	JP621-LC、JP635-LC、JP600-LC
沖電気工業	Pro1040、Pro1050

## 1-2

## MLWebComponent 機能一覧

■プロパティ ■メソッド

MLWebComponent で利用できるプロパティ・メソッド・例外の一覧を記載します。  
カラープリンタレイアウトで使用する機能のみに限定しております。

## ■プロパティ一覧

プロパティ名	概要	初期値
<b>通信設定</b>		
Setting	通信パラメータの設定	LAN:127.0.0.1
<b>基本設定</b>		
LayoutFile	レイアウトファイルの設定	Default.mllayx
PrnData	印字データの設定	""（空文字）
PrnDataType	印字データタイプの設定	Tsv
<b>プリンタ動作設定</b>		
Offset	印字補正值の設定	0.0000,0.0000
HeaderTailSetting	ヘッダ・テール札の設定	False（印字しない）
HeaderFile	ヘッダ札の取得	""（空文字）
TailFile	テール札の取得	""（空文字）
<b>特殊設定</b>		
LayoutNameCaption	「レイアウト名」の設定	""（空文字）
TotalQtyCaption	「総発行枚数」の設定	0
TaxRate	税率の設定	""（空文字）
<b>バージョン情報</b>		
Version	バージョン情報の取得	—

## ■メソッド一覧

メソッド名	概要
<b>通信</b>	
OpenPort	通信ポートのオープン
ClosePort	通信ポートのクローズ
<b>発行</b>	
Output	ラベル発行
OutputHeader	ヘッダ札発行
OutputTail	テール札発行
<b>ダウンロード</b>	
GetFile	ファイルのダウンロード
GetFileCredential	ファイルのダウンロード（BASIC 認証）
SetProxyCredential	プロキシサーバの BASIC 認証設定
SetClientCertificate	クライアント証明書の検索・設定



メソッド名	概要
GetClientCertCount	クライアント証明書数の取得
GetClientCertInfo	クライアント証明書情報の取得
<b>レイアウト情報取得</b>	
GetPrinter	プリンタ情報の取得
GetLayoutFileProperty	レイアウト情報の取得
IsColorPrinter	プリンタタイプの取得
<b>印字データ指定</b>	
GetInputCount	入力項目数の取得
GetInputFields	入力情報の取得
SetPrnDataField	入力項目の印字データ設定
GetPrnDataRowCount	複数データの行数取得
GetPrnDataRow	複数データの取得
SetPrnDataRow	複数データの 1 行分のデータ設定
<b>プリンタドライバ制御</b>	
GetDriverNameList	プリンタドライバ名の取得
ClearPrintJob	プリンタドライバのジョブ削除
<b>ライセンス認証</b>	
Authenticate	ライセンス認証の実行

## 第2章

# プロパティ

## 2-1

## [通信]Setting プロパティ

通信パラメータを設定、取得します。

## ■書式

MLWebComponent.Setting [As String](#)

## ■設定値

出力先	Setting
	説明
初期値	LAN:127.0.0.1
	カラープリンタレイアウトでは利用できません。
カラープリンタドライバ	DRVX:Drivername
	<i>Drivername</i> カラープリンタドライバ名称

## ■解説

- カラープリンタレイアウトを使用する場合は、「DRVX:」（カラープリンタドライバ）のみ使用可能です。
- サトープリンタレイアウトで「DRVX:」（カラープリンタドライバ）を指定した場合は、Output、OutputHeader、OutputTail メソッドの戻り値が「151」となります。
- PC にインストールされているカラープリンタドライバであっても、サポート機種以外の場合は、OpenPort メソッドの戻り値が「4」となります。
- 発行時にデザイン時と異なる機種のプリンタドライバで発行すると、用紙サイズが保持されない場合があります。デザイン時と発行時は同一のプリンタ機種でご利用ください。

## ■例外

- JavaScript での例外処理は try…catch で実行してください。Error オブジェクトの message プロパティにメッセージが格納されます。

例外	説明 (message)
InvalidOperationException	条件：ポートオープン中にプロパティを変更 The communication parameter cannot be changed when the port is opened.

## ■使用例

プリンタドライバ名称「SATO GHS612R」の場合

```
MLWebComponent.Setting = "DRVX:SATO GHS612R";
```

## ■関連項目

メソッド [OpenPort](#)、[Output](#)

## 2-2

## [基本]LayoutFile プロパティ

MLV5 で作成したレイアウトファイルを設定・取得します。

## ■書式

MLWebComponent.LayoutFile [As String](#)

## ■設定値

LayoutFile	説明
Default.mllayx	初期値
任意文字列	レイアウトファイルのローカルファイルパス

## ■解説

- レイアウトファイル (\*.mllayx) のみ使用可能です。
- 値を設定すると、PrnData プロパティ、SetPrnDataField メソッド、SetPrnDataRow メソッドで指定した入力データはすべて初期化されます。

## ■例外

- JavaScript での例外処理は try…catch で実行してください。Error オブジェクトの message プロパティにメッセージが格納されます。

例外	説明 (message)
ArgumentException	条件：不正な設定値 The value of LayoutFile property is invalid.

## ■使用例

レイアウト情報ファイルパスを設定します。

```
MLWebComponent.LayoutFile = "C:¥¥SATO¥¥ABC.mllayx";
```

現在のレイアウト情報ファイルパスを取得します。

```
LayoutFile = MLWebComponent.LayoutFile;
```

## ■関連項目

プロパティ [PrnData](#)、[HeaderTailSetting](#)、[HeaderFile](#)、[TailFile](#)  
メソッド [Output](#)、[GetInputFields](#)、[SetPrnDataField](#)、[SetPrnDataRow](#)、[GetPrinter](#)、[IsColorPrinter](#)

## 2-3

## [基本]PrnData プロパティ

印字データを設定、取得します。

## ■書式

MLWebComponent.PrnData [As String](#)

## ■設定値

<i>PrnData</i>	説明
空文字	初期値
任意文字列	PrnDataType プロパティのデータ形式で指定します。文字フォーマットは MLV5 準拠で Unicode(UTF-16)となります。

## ■解説

- バーコードにチェックデジットを含めたデータを指定する際は、チェックデジットが正しいかお客様のアプリケーションで必ずチェックした上で指定してください。チェックが出来ない場合は、チェックデジットを含まないデータに変更するか、変数の設定でチェックデジット部分を取り除くようにレイアウトファイルを変更してください。
- データの項目数は、レイアウトファイルの入力項目数に合わせて指定してください。
- LayoutFile プロパティを設定した時点で PrnData プロパティは初期化されます。
- 値を設定すると、SetPrnDataRow メソッドで設定した入力データは初期化されます。
- SetPrnDataField メソッドと併用する場合は、混乱を避けるため PrnData プロパティを先に指定してください。
- PrnDataType プロパティで指定された形式の区切り文字はデータとして設定しないでください。例として、PrnDataType プロパティが「タブ区切り」の場合に、データ内にタブ (0x09) を入力すると、入力項目の入力順がずれて不正な入力データとなります。
- QR コード等のバーコードデータに制御コードを含める場合は、レイアウトの入力変数種類を「16進文字コード」に設定し、上位アプリからは 16 進文字列のデータをセットしてください。（例：12345<TAB>の場合、「313233343509」をセット）

## ■使用例

印字データを設定します。

```
MLWebComponent.PrnDataType = 0;
MLWebComponent.PrnData = "1000" + "¥" + "2000" + "¥" + "1";
```

## ■関連項目

プロパティ

[LayoutFile](#)、[PrnDataType](#)

メソッド

[Output](#)、[GetInputFields](#)、[SetPrnDataField](#)、[GetPrnDataRow](#)、[SetPrnDataRow](#)

■サポート情報

ワンポイントテクニック

[データを一括で入力する](#)

## 2-4

## [基本]PrnDataType プロパティ

印字データのタイプ(Tsv/Csv/Prn)を設定、取得します。

## ■書式

MLWebComponent.PrnDataType [As Integer](#) / [SATO.MLWebComponent.PrnDataTypes](#)

## ■設定値

<i>PrnDataType</i>	説明
0 - PrnDataTypes.Tsv	タブ区切り指定(初期値)
1 - PrnDataTypes.Csv	カンマ区切り指定 (CSV 形式)
2 - PrnDataTypes.Prn	スペース区切り

## ■解説

- QR コード等のバーコードデータに制御コードを含める場合は、レイアウトの入力変数種類を「16 進文字コード」に設定し、上位アプリからは 16 進文字列のデータをセットしてください。(例：12345<TAB>の場合、「313233343509」をセット)

## ■例外

- JavaScript での例外処理は try…catch で実行してください。Error オブジェクトの message プロパティにメッセージが格納されます。

例外	説明 (message)
ArgumentOutOfRangeException	条件：範囲外の設定値 The value of PrnDataType property is invalid.

## ■使用例

印字データのタイプに TSV を設定します。

```
MLWebComponent.PrnDataType = 0;
```

## ■関連項目

プロパティ  
メソッド

[PrnData](#)  
[SetPrnDataField](#)、[GetPrnDataRow](#)、[SetPrnDataRow](#)

## 2-5

## [動作]Offset プロパティ

印字補正を設定・取得します。

## ■書式

MLWebComponent.Offset [As String](#)

## ■設定値

Offset	説明
0,0	補正なし（初期値）
S,S	レイアウト情報の印字補正を使用
横補正值,縦補正值	横・縦の補正値をカンマ区切りで mm 単位指定（-99.9~99.9） 小数点以下 4 桁まで有効

## ■解説

- 「S,S」でレイアウトに設定がない場合、プリンタに登録された設定値が有効になります。
- 印字補正を行うと、プリンタの印字領域をオーバーする、ラベルから文字がはみ出るなどの可能性があります。Output メソッドで以下のエラーが発生する可能性があります。注意が必要です。
  - 610 用紙に印字できない項目が存在します。
  - 611 計算できない無効な項目が存在します。

## ■例外

- JavaScript での例外処理は try…catch で実行してください。Error オブジェクトの message プロパティにメッセージが格納されます。

例外	説明 (message)
ArgumentOutOfRangeException	条件：範囲外の設定値 The value of Offset property is invalid. This property must be within the range from -99.9 to 99.9.

## ■使用例

印字位置を横に 3mm、縦に 2mm 補正して発行します。

```
MLWebComponent.Offset = "3.0,2.0";
```

レイアウト情報の印字補正値を使用します。

```
MLWebComponent.Offset = "S,S";
```

## ■関連項目

メソッド

[Output](#)



## 2-6

## [動作]HeaderTailSetting プロパティ

ヘッダ・テール札の印字有無を設定・取得します。

## ■書式

MLWebComponent.HeaderTailSetting [As Boolean](#)

## ■設定値

<i>HeaderTailSetting</i>	説明
False（初期値）	ヘッダ・テール札を発行しない
True	レイアウトの設定に従いヘッダ・テール札を発行する

## ■解説

- 「True」を指定し、レイアウトの設定が「ページごとに出力する」に設定されている場合は、発行指示ごとにヘッダ・テール札を出力します。

## ■使用例

レイアウトの設定に従いヘッダ・テール札を発行します。

```
MLWebComponent.LayoutFile = "C:¥¥SATO¥¥ABC.mllayx";
MLWebComponent.SetPrnDataRow(0, "商品 A" + "¥t" + "1");
MLWebComponent.SetPrnDataRow(1, "商品 B" + "¥t" + "2");
MLWebComponent.SetPrnDataRow(2, "商品 C" + "¥t" + "1");
MLWebComponent.SetPrnDataRow(3, "商品 D" + "¥t" + "1");
MLWebComponent.HeaderTailSetting = True;
Result = MLWebComponent.Output();
if ( Result != 0 ){
    //発行エラー
}
```

## ■関連項目

プロパティ            [LayoutFile](#)、 [PrnData](#)  
メソッド            [SetPrnDataRow](#)、 [Output](#)

**2-7****[動作]HeaderFile プロパティ**

ヘッダ札ファイルを取得します。設定はできません。

**■書式**

MLWebComponent.HeaderFile [As String](#)

**■設定値**

<i>HeaderFile</i>	説明
文字列（パス名）	ヘッダ札ファイルのローカルファイルパス名
空文字	取得失敗 レイアウトファイルにヘッダ札が設定されていない

**■解説**

- LayoutFile プロパティで設定されたレイアウトファイルから取得します。

**■使用例**

ヘッダ札を取得します。

```
HeaderFile = MLWebComponent.HeaderFile;
if ( HeaderFile != "" ){
    window.alert( "HeaderFile = " + HeaderFile );
} else {
    window.alert( "HeaderFile Error !! " );
}
```

**■関連項目**

プロパティ            [LayoutFile](#)、 [HeaderTailSetting](#)、 [TailFile](#)  
メソッド             [OutputHeader](#)

**2-8****[動作]TailFile プロパティ**

テール札ファイルを取得します。設定はできません。

**■書式**

MLWebComponent.TailFile [As String](#)

**■設定値**

<i>TailFile</i>	説明
文字列（パス名）	テール札ファイルのローカルファイルパス名
空文字	取得失敗 レイアウトファイルにテール札が設定されていない

**■解説**

- LayoutFile プロパティで設定されたレイアウトファイルから取得します。

**■使用例**

テール札を取得します。

```
TailFile = MLWebComponent.TailFile;  
if ( TailFile != "" ){  
    window.alert( "TailFile = " + TailFile );  
}else{  
    window.alert( "TailFile Error !! " );  
}
```

**■関連項目**

プロパティ            [LayoutFile](#)、 [HeaderTailSetting](#)、 [HeaderFile](#)  
メソッド            [OutputTail](#)

## 2-9

## [特殊]LayoutNameCaption プロパティ

システム変数「レイアウト名」を設定・取得します。

## ■書式

MLWebComponent.LayoutNameCaption [As String](#)

## ■設定値

<i>LayoutNameCaption</i>	説明
空文字	LayoutFile プロパティに設定されているレイアウト名を使用します。(初期値)
任意文字列	システム変数「レイアウト名」に設定する任意文字列か、ローカルファイルパスを指定します。

## ■解説

- MLV5 のシステム変数「レイアウト名」に値が使用されます。
- 「ローカル ファイル パス」指定の場合、ファイル名のみが「レイアウト名」になります。

## ■使用例

「レイアウト名」を設定します。

```
MLWebComponent.LayoutNameCaption = "C:¥¥SATO¥¥ABC.mllayx";
```

「レイアウト名」を取得します。

```
LayoutNameCaption = MLWebComponent.LayoutNameCaption;
```

## ■関連項目

プロパティ            [LayoutFile](#)、 [TotalQtyCaption](#)  
メソッド            [Output](#)

## 2-10

## [特殊]TotalQtyCaption プロパティ

システム変数「総発行枚数」を設定・取得します。

## ■書式

MLWebComponent.TotalQtyCaption [As Integer](#)

## ■設定値

<i>TotalQtyCaption</i>	説明
0	Output メソッド実行時に、総発行枚数を自動計算します。（初期値）
任意数値（0 以上）	システム変数「総発行枚数」を指定します。

## ■解説

- MLV5 のシステム変数「総発行枚数」に値が使用されます。

## ■例外

- JavaScript での例外処理は try…catch で実行してください。Error オブジェクトの message プロパティにメッセージが格納されます。

例外	説明 (message)
ArgumentOutOfRangeException	条件：範囲外の設定値 The value of TotalQtyCaption property is invalid.

## ■使用例

「総発行枚数」を設定します。

```
MLWebComponent.TotalQtyCaption = 13;
```

「総発行枚数」を取得します。

```
TotalQtyCaption = MLWebComponent.TotalQtyCaption;
```

## ■関連項目

プロパティ      [LayoutFile](#)  
メソッド        [Output](#)

**2-11****[特殊]TaxRate プロパティ**

税率を設定・取得します。

**■書式**MLWebComponent.TaxRate [As String](#)**■設定値**

<i>TaxRate</i>	説明
空文字	税率を設定しない（初期値）
任意文字列	税率 1～20 をカンマ区切りで指定（範囲:0～99.9）

**■解説**

- 条件印字のデータ種類を「税率」に設定した場合に設定値が使用されます。
- レイアウトファイルで変数の編集パラメータで設定した税編集で「固定値を使用する」が有効な場合は使用されません。
- 値が設定されていない場合は、Output メソッドの戻り値が「413」となります。

**■例外**

- JavaScript での例外処理は try…catch で実行してください。Error オブジェクトの message プロパティにメッセージが格納されます。

例外	説明 (message)
ArgumentOutOfRangeException	条件：範囲外の設定値(税率が 20 以上) The value of TaxRate property is invalid.
	条件：範囲外の設定値(0～99.9 の範囲外) The value of TaxRate property is invalid. This property must be within the range from 0 to 99.9.

**■使用例**

税率 No.1 に「5.0」、税率 No.2 に「8.0」を設定します。

```
MLWebComponent.TaxRate =" 5.0,8.0";
```

現在の税率を取得します。

```
TaxRate = MLWebComponent.TaxRate;
```

**■関連項目**

プロパティ            [LayoutFile](#)  
メソッド             [Output](#)

**2-12****[情報]Version プロパティ**

バージョン情報を取得します。設定はできません。

**■書式**

MLWebComponent.Version *As String*

**■設定値**

<i>Version</i>	説明
Multi LABELIST Web Component, Version x.x.x.x (x : バージョン番号)	MLWebComponent のバージョン情報を取得します。

**■使用例**

バージョン情報を取得します。

```
Version=MLWebComponent.Version;
```

取得結果 : "Multi LABELIST Web Component, Version 5.2.3.2"

## 第3章

# メソッド・例外



## 3-1

## [通信]OpenPort メソッド

通信ポートをオープンします。

## ■書式

MLWebComponent.OpenPort (ByVal *SyncMode* As Integer) As Integer

## ■パラメータ

パラメータ	設定値	説明
<i>SyncMode</i>	1（固定）	同期通信モード

## ■戻り値

戻り値	説明
0	正常終了です。
1	Setting プロパティの値が不正です。
3	既にオープンされています。
4	ポートオープン時にエラーが発生しました。
12	通信設定と通信プロトコルの組み合わせが未サポートです。
31	ライセンス認証が実行されていません。
36	OpenPort実行時の日付が、認証ファイルの有効期限を超えています。
39	ライセンスファイルがカラープリンタレイアウトに対応していません。

## ■解説

- WebComponent を利用するには、はじめに Authenticate メソッドで **ライセンス認証が必要**です。正しくライセンスが認証されていないと戻り値「31」、「36」となります。
- Setting プロパティで設定されたデバイスをオープンします。
- 戻り値「4」は、Setting プロパティで存在しないプリンタドライバ名や、サポートされていない機種種のプリンタドライバ名が指定されている場合などに発生します。Setting プロパティの解説もご参照ください。

## ■使用例

通信ポートのオープンからクローズまでのプログラム例を示します。

```
//前提としてライセンス認証が実行されていること
MLWebComponent.Setting = "DRVX:SATO GHS612R";
Result = MLWebComponent.OpenPort(1);
if ( Result = 0 ){
    //一連の出力処理
    MLWebComponent.ClosePort();
}
```

■関連項目

- プロパティ [Setting](#)
- メソッド [Output](#)

**3-2****[通信]ClosePort メソッド**

通信ポートをクローズします。

**■書式**MLWebComponent.ClosePort () [As Integer](#)**■戻り値**

戻り値	説明
0	正常終了です。
5	ポートがオープンされていません。
6	ポートクローズ時にエラーが発生しました。

**■使用例**

通信ポートのオープンからクローズまでのプログラム例を示します。

```
//前提としてライセンス認証が実行されていること
MLWebComponent.Setting = "DRVX:SATO GHS612R";
Result = MLWebComponent.OpenPort(1);
if ( Result = 0 ){
    //一連の出力処理
    MLWebComponent.ClosePort();
}
```

**■関連項目**

プロパティ

[Setting](#)

メソッド

[OpenPort](#)

**3-3****[発行]Output メソッド**

ラベル・タグを発行します。

**■書式**MLWebComponent.Output () [As Integer](#)**■戻り値**

戻り値	説明
0	正常終了です。
5	ポートがオープンされていません。
7	プリンタドライバにデータ出力が出来ません。
上記以外	「 <a href="#">発行エラー</a> 」をご参照ください。

**■使用例**

連番印字を行います。

```
MLWebComponent.PrnData = "10";  
Result = MLWebComponent.Output();  
if ( Result != 0 ){  
    //エラー処理  
}
```

**■関連項目**プロパティ [Setting](#)、[LayoutFile](#)、[PrnData](#)**■サポート情報**メソッド・例外 [発行エラー](#)

**3-4****[発行]OutputHeader メソッド**

ヘッダ札を発行します。

**■書式**MLWebComponent.OutputHeader () [As Integer](#)**■戻り値**

戻り値	説明
0	正常終了です。
5	ポートがオープンされていません。
7	プリンタドライバにデータ出力が出来ません。
上記以外	「 <a href="#">発行エラー</a> 」をご参照ください。

**■解説**

- ・ レイアウトファイルに設定されているヘッダ札を発行枚数の指定に関わらず 1 枚のみ発行します。

**■使用例**

ヘッダ札を発行します。

```
MLWebComponent.PrnData = "5";
Result = MLWebComponent.OutputHeader();
if ( Result != 0 ){
    //エラー処理
}
```

**■参照項目**プロパティ [Setting](#)、[HeaderTailSetting](#)、[HeaderFile](#)、[TailFile](#)**■サポート情報**メソッド・例外 [発行エラー](#)

**3-5****[発行]OutputTail メソッド**

テール札を発行します。

**■書式**MLWebComponent.OutputTail() [As Integer](#)**■戻り値**

戻り値	説明
0	正常終了です。
5	ポートがオープンされていません。
7	プリンタドライバにデータ出力が出来ません。
上記以外	「 <a href="#">発行エラー</a> 」をご参照ください。

**■解説**

- レイアウトファイルに設定されているテール札を発行枚数の指定に関わらず 1 枚のみ発行します。

**■使用例**

テール札を発行します。

```
MLWebComponent.PrnData = "5";
Result = MLWebComponent.OutputTail();
if ( Result != 0 ){
    //エラー処理
}
```

**■参照項目**プロパティ [Setting](#)、[HeaderTailSetting](#)、[HeaderFile](#)、[TailFile](#)**■サポート情報**メソッド・例外 [発行エラー](#)

## 3-6

## [ダウンロード]GetFile メソッド

サーバから HTTP 通信でファイルをダウンロードします。

## ■書式

MLWebComponent.GetFile(

ByVal *URL* As String,

ByVal *LocalPath* As String,

ByVal *Param* As Integer) As Integer

## ■パラメータ

パラメータ	設定値	説明
<i>URL</i>	任意文字列	ファイルを公開している URL を指定します。
<i>LocalPath</i>	任意文字列	ダウンロード先のローカルフォルダパスを指定します。
<i>Param</i>	0	形式を変えずにダウンロードします。
	1	CAB 形式ファイルの場合、ダウンロード後にファイルを展開します。
	2	サーバにあるファイルの更新日付が、ローカルの同一ファイルより新しい場合にダウンロードします。
	3	サーバにあるファイルの更新日付が、ローカルの同一ファイルより新しい場合にダウンロードします。CAB 形式ファイルの場合、ファイルを展開します。

## ■戻り値

戻り値	説明
0	正常終了です。
-1	レイアウトファイル名が不正です。
-2	プロトコルが HTTP および HTTPS ではありません。
-3	指定した URL の記述が不正です。（URL の書式ではありません。）
-5	ダウンロード実行中に予期せぬエラーが発生しました。 ①HTTP のステータスを取得することができません。 ②HTTPS でサーバ証明書が不正です。（認証エラー） ③指定した URL にアクセスできません。 ④ファイルの最終更新日時を示すヘッダ情報「Last-Modified」が取得できません。 ⑤.NET Framework を 4.5 以降にバージョンアップしてください。 ⑥HTTP から HTTPS にリダイレクトされました。URL を HTTPS にしてください。
-6	受信ファイルがオープンできません。 <a href="#">セキュリティ設定を確認してください。</a>
300 以上	HTTP 要求エラー（HTTP ステータスコード）。

## ■解説

- 接続する URL がプロキシ経由(インターネット)の場合、インターネットオプションで設定を行ってください。設定がない場合(IE での閲覧ができない場合)は、接続できませんのでご注意ください。□

ーカル(イントラネット)の場合は必要ありません。

- WebサイトにBASIC認証が設定されている場合は、GetFileCredential メソッドをご利用ください。GetFile メソッドでは戻り値が「401」となります。
- Webサイトにクライアント証明書認証が設定されている場合は、GetFile メソッドの前に SetClientCertificate メソッドを実行してください。GetFile メソッドでは戻り値が「403」となります。
- クライアント側のプロキシにBASIC認証を設定している場合は、GetFile メソッドの前に SetProxyCredential メソッドを実行してください。GetFile メソッドでは戻り値が「407」となります。
- Ver.5.9.1.0以降は、TLS 1.1、TLS 1.2への対応により、実行環境に.NET Framework 4.5以降が必要です。.NET Framework 4.0の環境では戻り値が「-5」となります。

### ■使用例

レイアウトファイルをダウンロードします。

```
Result = MLWebComponent.GetFile("http:// example.com /label.mllayx", "C:¥¥TEMP¥¥SATO¥¥", 3);
if ( Result != 0 ){
    //ダウンロードエラー
}
```

### ■関連項目

プロパティ  
メソッド

[LayoutFile](#)  
[Output](#)、[GetFileCredential](#)、[SetProxyCredential](#)、[SetClientCertificate](#)、[GetClientCertCount](#)、[GetClientCertInfo](#)



## 3-7

## [ダウンロード]GetFileCredential メソッド

BASIC 認証が必要な Web サイトから HTTP 通信でファイルをダウンロードします。

## ■書式

MLWebComponent.GetFileCredential(

ByVal *URL* As String,  
 ByVal *UserName* As String,  
 ByVal *Password* As String,  
 ByVal *Domain* As String,  
 ByVal *LocalPath* As String,  
 ByVal *Param* As Integer) As Integer

## ■パラメータ

パラメータ	設定値	説明
<i>URL</i>	任意文字列	ファイルを公開している URL を指定します。
<i>UserName</i>	任意文字列	Web サイトの資格情報のユーザー名を指定します。
<i>Password</i>	任意文字列	Web サイトの資格情報のパスワードを指定します。
<i>Domain</i>	任意文字列	Web サイトの資格情報のドメイン名を指定します。
<i>LocalPath</i>	任意文字列	ダウンロード先のローカルフォルダパスを指定します。
<i>Param</i>	0	形式を変えずにダウンロードします。
	1	CAB 形式ファイルの場合、ダウンロード後にファイルを展開します。
	2	サーバにあるファイルの更新日付が、ローカルの同一ファイルより新しい場合にダウンロードします。
	3	サーバにあるファイルの更新日付が、ローカルの同一ファイルより新しい場合にダウンロードします。CAB 形式ファイルの場合、ファイルを展開します。

## ■戻り値

戻り値	説明
0	正常終了です。
-1	レイアウトファイル名が不正です。
-2	プロトコルが HTTP および HTTPS ではありません。
-3	指定した URL の記述が不正です。（URL の書式ではありません。）
-5	①HTTP のステータスを取得することができません。 ②HTTPS でサーバ証明書が不正です。（認証エラー） ③指定した URL にアクセスできません。 ④ファイルの最終更新日時を示すヘッダ情報「Last-Modified」が取得できません。 ⑤.NET Framework を 4.5 以降にバージョンアップしてください。
-6	受信ファイルがオープンできません。 <a href="#">セキュリティ設定を確認してください。</a>
300 以上	HTTP 要求エラー（HTTP ステータスコード）。

## ■解説

- 接続する URL がプロキシ経由(インターネット)の場合、インターネットオプションで設定を行ってください。設定がない場合(IE での閲覧ができない場合)は、接続できませんのでご注意ください。ローカル(イントラネット)の場合は必要ありません。
- Web サイトに BASIC 認証を設定している場合に使用します。NTLM 認証、ネゴシエート認証、および Kerberos ベースの認証では Windows 資格情報を使用するため GetFile メソッドをご使用ください。
- Web サイトにクライアント証明書認証が設定されている場合は、GetFileCredential メソッドの前に SetClientCertificate メソッドを実行してください。GetFileCredential メソッドでは戻り値が「403」となります。
- クライアント側のプロキシに BASIC 認証を設定している場合は、GetFileCredential メソッドの前に SetProxyCredential メソッドを実行してください。GetFileCredential メソッドでは戻り値が「407」となります。
- Ver.5.9.1.0 以降は、TLS 1.1、TLS 1.2 への対応により、実行環境に .NET Framework 4.5 以降が必要です。 .NET Framework 4.0 の環境では戻り値が「-5」となります。

## ■使用例

レイアウトファイルをダウンロードします。

```
Result = MLWebComponent.GetFileCredential("http://example.com/label.mllayx", "user1234",  
"pw5678", "", "C:¥¥TEMP¥¥SATO¥¥", 3);  
if ( Result != 0 ){  
    //ダウンロードエラー  
}
```

## ■関連項目

プロパティ [LayoutFile](#)  
メソッド [Output](#)、[GetFile](#)、[SetProxyCredential](#)、[SetClientCertificate](#)、[GetClientCertCount](#)、[GetClientCertInfo](#)

## 3-8

## [ダウンロード]SetProxyCredential メソッド

クライアント側でプロキシ認証が必要な場合にあらかじめ資格情報を設定します。

## ■書式

```
MLWebComponent.SetProxyCredential(
    ByVal UserName As String,
    ByVal Password As String,
    ByVal Domain As String) As Integer
```

## ■パラメータ

パラメータ	設定値	説明
<i>UserName</i>	任意文字列	Web サイトの資格情報のユーザー名を指定します。
<i>Password</i>	任意文字列	Web サイトの資格情報のパスワードを指定します。
<i>Domain</i>	任意文字列	Web サイトの資格情報のドメイン名を指定します。

## ■戻り値

戻り値	説明
0	正常終了です。
その他	設定値が不正です。

## ■解説

- クライアント側のプロキシサーバに BASIC 認証を設定している場合に、GetFile メソッドまたは GetFileCredential メソッドよりも先に実行してください。
- 設定値は MLWebComponent のインスタンスごとに有効で、インスタンスが破棄されると設定値も破棄されます。

## ■使用例

プロキシ認証の資格情報を設定します。

```
Result = MLWebComponent.SetProxyCredential("user0123", "pw4567", "");
if ( Result != 0 ){
    //資格情報設定エラー
}
```

## ■関連項目

プロパティ [LayoutFile](#)  
 メソッド [Output](#)、[GetFile](#)、[GetFileCredential](#)

## 3-9

## [ダウンロード]SetClientCertificate メソッド

ダウンロード時に使用するクライアント証明書を設定します。

## ■書式

```
MLWebComponent.SetClientCertificate(
    ByVal SubjectName As String,
    ByVal IssuerName As String,
    ByVal IsMultiErr As Boolean) As Integer
```

## ■パラメータ

パラメータ	設定値	説明
<i>SubjectName</i>	任意文字列	検索条件の発行先を指定します
<i>IssuerName</i>	任意文字列	検索条件の発行者を指定します。
<i>IsMultiErr</i>	true	該当する証明書が複数ある場合にエラーとします。
	false	該当する証明書が複数ある場合は、有効期限が最も遅い証明書を使用します。

## ■戻り値

戻り値	説明
0	正常終了です。
80	該当する証明書が存在しません。
81	該当する証明書が複数あります。（IsMultiErr が true の場合）

## ■解説

- Web サイトにクライアント証明書認証が設定されている場合に、GetFile メソッドまたは GetFileCredential メソッドよりも先に実行してください。
- 設定値は MLWebComponent のインスタンスごとに有効で、インスタンスが破棄されると設定値も破棄されます。
- 現在のログインユーザで個人証明書として有効な証明書から、検索条件で指定した発行先、発行者に部分一致する証明書を検索して設定します。検索した結果、複数の証明書が該当する場合は、パラメータ IsMultiErr の設定値に従ってエラーまたは設定完了とします。
- Windows 標準の証明書マネージャー ツール(Certmgr.exe)で[個人]-[証明書]から参照できる証明書と同じ情報から検索します。有効期限が切れている証明書は除外します。

## ■使用例

クライアント証明書を取得します。

```
// クライアント証明書設定
Result = MLWebComponent.SetClientCertificate("subject", "issuer", false);
if (Result != 0 ){
    //設定エラー
}
```

## ■関連項目

メソッド [GetFile](#)、[GetFileCredential](#)、[GetClientCertCount](#)、[GetClientCertInfo](#)

## 3-10

## [ダウンロード]GetClientCertCount メソッド

現在のログインユーザで有効なクライアント証明書の数を取得します。

## ■書式

MLWebComponent.GetClientCertCount() *As Integer*

## ■戻り値

戻り値	説明
1 以上の整数	有効なクライアント証明書の数返します。
0	有効なクライアント証明書が見つかりません。

## ■解説

- 現在のログインユーザで個人証明書として有効な証明書の数を取得します。有効期限が切れている証明書は除外します。
- Windows 標準の証明書マネージャー ツール(Certmgr.exe)で[個人]-[証明書]から参照できる証明書と同じ情報が取得できます。
- GetClientCertInfo メソッドと組み合わせて使用します。

## ■使用例

クライアント証明書の情報を取得します。

```

CertCount = MLWebComponent.GetClientCertCount;           //クライアント証明書数取得
for(var i = 0; i < CertCount - 1; i++){                  //クライアント証明書数ループ
    CertInfo = MLWebComponent.GetClientCertInfo(n);      //クライアント証明書情報の取得
    //取得したクライアント証明書情報を使用する
}

```

## ■関連項目

メソッド

[GetFile](#)、[GetFileCredential](#)、[SetClientCertificate](#)、[GetClientCertInfo](#)

## 3-11

## [ダウンロード]GetClientCertInfo メソッド

現在のログインユーザで有効なクライアント証明書を取得します。

## ■書式

MLWebComponent.GetClientCertInfo(ByVal Index As Integer) As String

## ■パラメータ

パラメータ	設定値	説明
<i>Index</i>	任意数値	取得するクライアント証明書のインデックス

## ■戻り値

戻り値	説明
(入力情報)	指定したインデックスのクライアント証明書情報（カンマ区切り）
空文字	取得に失敗

- 入力情報のフォーマット

- ①発行先                   文字列
- ②発行者                   文字列
- ③有効期限                文字列
- ④フレンドリ名           文字列
- ⑤秘密鍵の有無          true（あり） / false（なし）

## ■解説

- 現在のログインユーザで個人証明書として有効な証明書の情報を取得します。有効期限が切れている証明書は除外します。
- Windows 標準の証明書マネージャー ツール(Certmgr.exe)で[個人]-[証明書]から参照できる証明書と同じ情報が取得できます。
- GetClientCertCount メソッドと組み合わせて使用します。

## ■使用例

クライアント証明書の情報を取得します。

```

CertCount = MLWebComponent.GetClientCertCount;           //クライアント証明書数取得
for(var i = 0; i < CertCount - 1; i++){                   //クライアント証明書数ループ
    CertInfo = MLWebComponent.GetClientCertInfo(n);       //クライアント証明書情報の取得
    //取得したクライアント証明書情報を使用する
}

```

## ■関連項目

メソッド

[GetFile](#)、[GetFileCredential](#)、[SetClientCertificate](#)、[GetClientCertCount](#)

**3-12****[情報]GetPrinter メソッド**

レイアウトファイルからプリンタ情報を取得します。

**■書式**MLWebComponent.GetPrinter() [As String](#)**■戻り値**

戻り値	説明
文字列	レイアウトファイル作成時に設定したプリンタドライバのモデル名
空文字	取得に失敗

**■解説**

- LayoutFile プロパティで設定されたレイアウトファイルから取得します。

**■使用例**

レイアウトファイル” C:¥SATO¥ABC.mlmax” のプリンタ機種を取得します。

```
MLWebComponent.LayoutFile = "C:¥SATO¥ABC.mllayx";
PrinterModel = MLWebComponent.GetPrinter;
if ( PrinterModel != "" ){
    //正常に取得
} else {
    //取得エラー
}
```

**■関連項目**プロパティ [LayoutFile](#)



## 3-13

## 【情報】GetLayoutFileProperty メソッド

レイアウトファイルからプリンタ情報を取得します。

## ■書式

MLWebComponent.GetLayoutFileProperty() *As String*

## ■戻り値

戻り値	説明
(レイアウト情報)	カンマ区切りのレイアウト情報
空文字	取得に失敗

- レイアウト情報のフォーマット

## ①センサタイプ

「0」アイマークラベル、「1」ギャップラベル、  
「2」センターホールタグ、「3」横ホールタグ、  
「4」アイマークタグ、「5」角Rタグ、  
「-1」その他（カラープリンタなど）

## ②用紙サイズ

横サイズ(タブ)縦サイズ 単位:mm(小数点以下 1 桁固定)

## ③1 シートの枚数

多面取り設定の枚数

## ④グローバル情報ファイルパス

フルパスのローカルファイルパス

## ■解説

- LayoutFile プロパティで設定されたレイアウトファイルから取得します。

## ■使用例

レイアウトファイルの情報を取得します。

```
MLWebComponent.LayoutFile = "C:¥¥SATO¥¥ABC.mllayx";
LayoutInfo = MLWebComponent. GetLayoutFileProperty();
if (LayoutInfo != "" ){
    //正常に取得
} else {
    //取得エラー
}
```

## ■関連項目

プロパティ

[LayoutFile](#)

**3-14****[情報] IsColorPrinter メソッド**

レイアウトファイルのプリンタタイプ(サトープリンタ/カラープリンタ)を取得します。

**■書式**MLWebComponent.IsColorPrinter () [As Boolean](#)**■戻り値**

戻り値	説明
True	カラープリンタに設定されています。
False	サトープリンタに設定されている、または取得エラーです。

**■関連項目**プロパティ [LayoutFile](#)

**3-15****[データ]GetInputCount メソッド**

レイアウトファイルの入力項目の数を取得します。

**■書式**MLWebComponent.GetInputCount() [As Integer](#)**■戻り値**

戻り値	説明
1 以上の整数	入力項目の数
0	取得に失敗

**■解説**

- LayoutFile プロパティで設定されたレイアウトファイルから取得します。
- GetInputFields メソッドと組み合わせて使用します。

**■使用例**

入力情報を取得します。

```
InputCount = MLWebComponent.GetInputCount();
for( n = 0 ; InputCount - 1 ; i++){
    InputStatus = MLWebComponent.GetInputFields(n);
    //取得した入力情報を使用する。
}
```

**■関連項目**

プロパティ            [LayoutFile](#)  
メソッド             [GetInputFields](#)

## 3-16

## [データ]GetInputFields メソッド

レイアウトファイルの入力情報を取得します。

## ■書式

MLWebComponent.GetInputFields(ByVal *Index* As Integer) As String

## ■パラメータ

パラメータ	設定値	説明
<i>Index</i>	任意数値	取得する入力情報のインデックス

## ■戻り値

戻り値	説明
(入力情報)	指定したインデックスのカンマ区切りの入力情報
空文字	取得に失敗

## ・ 入力情報のフォーマット

- ①項目名                    入力項目名称
- ②ヘッダ・行              0（ヘッダ） / 1（行）
- ③表示位置                Y(縦)X(横)H(高さ)W(幅) 単位:ピクセル  
※行項目の場合、YおよびXは0になります。
- ④桁数                    入力桁数
- ⑤入力許可                0（入力可） / 1（入力不可） / 2（非表示）
- ⑥発行後削除              0（OFF） / 1（ON）
- ⑦漢字入力                0（OFF） / 1（ON）
- ⑧仕分・カット            「0」OFF / 「1」ON
- ⑨入力チェック            0（なし） / 1（数値） / 2（英字） / 3（英数字） / 4（半角）  
5（全角） / 10（日付） / 11（16進） / 12（CODE39） / 13（NW-7）
- ⑩入力画面文字種        P(ポイント数)N(フォント名)
- ⑪その他チェック        0（なし） / 1（CD） / 2（テーブル） / 3（丸め） / 4（日付範囲）  
5（特殊） / 6（条件）
- ⑫初期値                 初期値文字列

## ■解説

- ・ LayoutFile プロパティで設定されたレイアウトファイルから取得します。
- ・ パラメータ InputStatus で取得できる入力情報の順番がレイアウトファイルの入力順です。PrnData で指定する項目順になります。
- ・ GetInputCount メソッドと組み合わせて使用します。
- ・ 入力画面文字種は、ML デザインの入力定義で設定した入力項目の表示用文字種です。ラベル上の文字フォントではありません。

## ■使用例

入力情報を取得します。

```
InputCount = MLWebComponent.GetInputCount();
for( n = 0 ; InputCount - 1 ; i++){
    InputStatus = MLWebComponent.GetInputFields(n);
    //取得した入力情報を使用する。
}
```

### • 出力結果

品番,0,Y100X150H10W70,7,0,0,0,0,1, P12NMS Pゴシック,0,12345,  
商品コード,1,YOX0H10W80,8,0,0,0,0,1,,P12NMS Pゴシック, 0,12345678  
発行枚数,1,YOX0H10W50,6,0,0,0,0,0,,P12NMS Pゴシック,0,1

## ■関連項目

プロパティ            [LayoutFile](#)、 [PrnData](#)  
メソッド             [GetInputCount](#)、 [SetPrnDataField](#)

## 3-17

## [データ]SetPrnDataField メソッド

指定した入力項目の印字データを設定します。

## ■書式

MLWebComponent.SetPrnDataField

(ByVal Name As String, ByVal Data As String) As Integer

## ■パラメータ

パラメータ	設定値	説明
Name	任意文字列	入力項目名称
Data	任意文字列	設定するデータ 文字フォーマットは MLV5 準拠で Unicode(UTF-16)

## ■戻り値

戻り値	説明
0	設定完了しました。
1	レイアウト情報ファイルのロード時にエラーが発生しました。 <ul style="list-style-type: none"> <li>指定されたレイアウトファイルが存在しません。</li> <li>指定されたレイアウトファイルが開けません。</li> <li>該当する入力項目が存在しません。</li> </ul>
61	レイアウトファイルが MLWebComponent より新しいバージョンの MLV5 で作成されています。MLWebComponent のバージョンを更新してください。

## ■解説

- 同一名称が複数存在する場合は、入力順で最初の入力項目のみに印字データを設定します。
- LayoutFile プロパティを設定した時点で、設定したデータはすべてクリアされます。
- バーコードにチェックデジットを含めたデータを指定する際は、チェックデジットが正しいかお客様のアプリケーションで必ずチェックした上で指定してください。チェックが出来ない場合は、チェックデジットを含まないデータに変更するか、変数の設定でチェックデジット部分を取り除くようにレイアウトファイルを変更してください。
- PrnData プロパティと併用する場合は、混乱を避けるため PrnData プロパティを先に指定してください。
- PrnDataType プロパティで指定された形式の区切り文字はデータとして設定しないでください。例として、PrnDataType プロパティが「タブ区切り」の場合に、データ内にタブ (0x09) を入力すると、入力項目の入力順がずれて不正な入力データとなります。
- QR コード等のバーコードデータに制御コードを含める場合は、レイアウトの入力変数種類を「16 進文字コード」に設定し、上位アプリからは 16 進文字列のデータをセットしてください。（例：12345<TAB>の場合、「313233343509」をセット）

## ■使用例

各入力名の位置に、印字データをセットします。

```
Result = MLWebComponent.SetPrnDataField("品番", "12-34");           //品番セット
if ( Result != 0 ){
//設定エラー
}
Result = MLWebComponent.SetPrnDataField("商品 No", "012345");       //商品 No セット
if ( Result != 0 ){
//設定エラー
}
Result = MLWebComponent.SetPrnDataField("取引先名", "サトー");     //取引先名セット
if ( Result != 0 ){
//設定エラー
}
Result = MLWebComponent.SetPrnDataField("発行枚数", "1");          //発行枚数セット
if ( Result != 0 ){
//設定エラー
}
```

## ■関連項目

プロパティ            [LayoutFile](#)、[PrnData](#)、[PrnDataType](#)  
メソッド             [Output](#)、[GetInputFields](#)

**3-18****[データ]GetPrnDataRowCount メソッド**

設定されている印字データの行数を取得します。

**■書式**MLWebComponent.GetPrnDataRowCount() [As Integer](#)**■戻り値**

戻り値	説明
0以上の整数	設定されている印字データの行数

**■使用例**

データの行数を取得します。

```
RowCount = MLWebComponent.GetPrnDataRowCount();
```

**■関連項目**

プロパティ

[PrnData](#)、[SetPrnDataField](#)

メソッド

[GetPrnDataRow](#)、[SetPrnDataRow](#)



**3-19****[データ]GetPrnDataRow メソッド**

指定された行に設定されている印字データを取得します。

**■書式**

MLWebComponent.GetPrnDataRow(ByVal *Index* As Integer) As String

**■パラメータ**

パラメータ	設定値	説明
<i>Index</i>	任意数値	データの行番号

**■戻り値**

戻り値	説明
(印字データ)	指定された行に設定されている印字データ
空文字	データが存在しません。

**■解説**

- LayoutFile プロパティを設定した時点で、入力データはすべてクリアされます。

**■使用例**

印字データを取得します。

```
InputData = MLWebComponent.GetPrnDataRow(2);
```

**■関連項目**

プロパティ [LayoutFile](#)、[PrnData](#)、[PrnDataType](#)  
メソッド [SetPrnDataRow](#)、[GetPrnDataRowCount](#)、[SetPrnDataRow](#)

## 3-20

## [データ]SetPrnDataRow メソッド

1 行分のデータを設定します。

## ■書式

MLWebComponent.SetPrnDataRow(ByRef *Data* As String()) As Integer

## ■パラメータ

パラメータ	設定値	説明
<i>Data</i>	任意文字列	各項目を PrnDataType プロパティで指定に対応した文字の区切りで指定します。文字フォーマットは MLV5 準拠で Unicode(UTF-16)となります。

## ■戻り値

戻り値	説明
0 以上の数値	印字データを設定した行番号です。
-1	設定失敗しました。

## ■解説

- LayoutFile プロパティを設定した時点で、設定したデータは初期化されます。
- 値を設定すると PrnData プロパティ、SetPrnDataField メソッドで設定したデータは初期化されます。
- バーコードにチェックデジットを含めたデータを指定する際は、チェックデジットが正しいかお客様のアプリケーションで必ずチェックした上で指定してください。チェックが出来ない場合は、チェックデジットを含まないデータに変更するか、変数の設定でチェックデジット部分を取り除くようにレイアウトファイルを変更してください。
- PrnDataType プロパティで指定された形式の区切り文字はデータとして設定しないでください。例として、PrnDataType プロパティが「タブ区切り」の場合に、データ内にタブ (0x09) を入力すると、入力項目の入力順がずれて不正な入力データとなります。
- QR コード等のバーコードデータに制御コードを含める場合は、レイアウトの入力変数種類を「16 進文字コード」に設定し、上位アプリからは 16 進文字列のデータをセットしてください。（例：12345<TAB>の場合、「313233343509」をセット）

## ■使用例

複数データをセットします。

```
Result = MLWebComponent.SetPrnDataRow( "商品 A" & Chr$(9) & "1");
if ( Result == -1 ){
    //設定エラー
}
Result = MLWebComponent.SetPrnDataRow( "商品 B" & Chr$(9) & "2");
if ( Result == -1 ){
    //設定エラー}
Result = MLWebComponent.SetPrnDataRow( "商品 C" & Chr$(9) & "1");
if ( Result == -1 ){
    //設定エラー
}
Result = MLWebComponent.Output();
if ( Result != 0 ){
    //発行エラー
}
```

## ■関連項目

プロパティ  
メソッド

[LayoutFile](#)、[PrnData](#)、[PrnDataType](#)  
[Output](#)、[SetPrnDataField](#)、[GetPrnDataRowCount](#)、[GetPrnDataRow](#)

**3-21****[データ]GetDriverNameList メソッド**

インストールされているプリンタドライバ名を取得します。

**■書式**

MLWebComponent.GetDriverNameList(ByVal *Param* As Integer) As String

**■パラメータ**

パラメータ	設定値	説明
<i>Param</i>	0	サトープリンタのみ
	1	すべてのプリンタ

**■戻り値**

戻り値	説明
(プリンタドライバ名)	カンマ区切りのプリンタドライバ名
空文字	取得に失敗

**■解説**

- *Param*が「0」（サトープリンタ）の場合、プリンタドライバのモデル名が「SATO」で始まる機種のみ取得します。

**■使用例**

プリンタドライバ名を取得します。

```
DriverName = MLWebComponent.GetDriverNameList(0);
```

**■関連項目**

プロパティ            [Setting](#)  
メソッド            [OpenPort](#)、[ClearPrintJob](#)

**3-22****[データ]ClearPrintJob メソッド**

プリンタドライバの印刷ジョブを削除します。

**■書式**MLWebComponent.ClearPrintJob( ByVal *DriverName* As String ) As Integer**■パラメータ**

パラメータ	設定値	説明
<i>DriverName</i>	任意文字列	印刷ジョブを削除するプリンタドライバ名

**■戻り値**

戻り値	説明
0	印刷ジョブを削除しました。
21	プリンタドライバが指定されていません。 指定されたプリンタドライバがありません。
22	印刷ジョブの削除に失敗しました。 プリンタドライバを操作する権限がありません。 ユーザ権限で実行されている場合、管理者権限の印刷ジョブは削除できません。

**■使用例**

プリンタドライバの印刷ジョブを削除します。

```
Result = MLWebComponent.ClearPrintJob("SATO GHS612R");
if ( Result != 0 ){
    //削除エラー
}
```

**■関連項目**

メソッド

[GetDriverNameList](#)

## 3-23

## [認証]Authenticate メソッド

ライセンス認証を実行します。

## ■書式

MLWebComponent.Authenticate(  
     ByVal Key As String , ByVal LocalPath As String) As Integer

## ■パラメータ

パラメータ	設定値	説明
Key	任意文字列	認証ファイルに対応した認証キーを指定します。
LocalPath	任意文字列	ライセンスファイルのローカルファイルパス（フルパス）を指定します。
	空文字	MLWebComponent と同じディレクトリを参照します。

## ■戻り値

戻り値	説明
0	認証成功
32	認証ファイルが存在しません。
33	認証ファイルにアクセスできません。
34	認証ファイルのデータが読み取れません。
35	認証ファイルが改ざんされています。
36	認証ファイルの有効期限を超えています。
37	認証キーが認証ファイルと一致しません。
38	認証ファイルの対象アプリケーションが一致しません。

## ■解説

- MLWebComponent を使用する際に必ず指定してください。ライセンス認証がされていない場合は、OpenPort の戻り値が「31」となります。
- ライセンスファイルと認証キーの入手方法は[ワンポイントテクニック「ライセンス認証を行う」](#)をご参照ください。

## ■使用例

ライセンス認証を行います。

```
Key = "d6aa8272-16f8-4c83-9b14-cf4b8fd35f1c";
Result = MLWebComponent.Authenticate(Key, "") ;
if ( Result != 0 ){
    //認証エラー
}
```

## ■関連項目

プロパティ [Setting](#)

メソッド

[OpenPort](#)

## 3-24

## [エラー]発行エラー

発行エラーについて説明します。

## ■発行エラー発生メソッド

[Output](#)、[OutputHeader](#)、[OutputTail](#)

## ■発行エラー一覧

発行エラーはレイアウトファイルなど発行に必要な情報がない場合に発生します。エラー発生時はエラーの内容に応じた対処を行ってください。

戻り値	内容 詳細
ファイルロードエラー：必要なファイルを用意してください。	
51	レイアウト情報ファイルのロード時にエラーが発生しました。
	レイアウトファイル（.mllayx）が存在しないか、読取りが出来ません。リクエストで指定したファイルパスをご確認ください。他のアプリケーションで利用されている場合、アクセス権限がない場合は読取りできません。 Windows アップデートの不具合により PC のフォント情報が読み取れません。更新プログラムを適用してください。 <a href="https://support.microsoft.com/ja-jp/help/4074906/">https://support.microsoft.com/ja-jp/help/4074906/</a>
52	プリンタ情報のロード時にエラーが発生しました。
	レイアウトファイルで使用されているプリンタ機種が現在のバージョンでサポートされていません。MLWebComponent をバージョンアップしてください。
57	印字データが無効です。
	PrnDataType プロパティに設定したデータタイプと異なるデータが設定されています。指定しているデータをご確認ください。
61	レイアウトファイルのロード時にエラーが発生しました。
	レイアウトファイルが MLWebComponent より新しいバージョンの MLV5 で作成されています。MLWebComponent をバージョンアップしてください。
62	グローバル情報ファイルのロード時にエラーが発生しました。
	グローバル情報ファイルが MLWebComponent より新しいバージョンの MLV5 で作成されています。MLWebComponent をバージョンアップしてください。
63	ヘッダ札ファイルのロード時にエラーが発生しました。
	ヘッダ札ファイルが MLWebComponent より新しいバージョンの MLV5 で作成されています。MLWebComponent をバージョンアップしてください。
64	テール札ファイルのロード時にエラーが発生しました。
	テール札ファイルが MLWebComponent より新しいバージョンの MLV5 で作成されています。MLWebComponent をバージョンアップしてください。
65	グローバルテーブルファイルのロード時にエラーが発生しました。
	グローバルテーブルファイルが MLWebComponent より新しいバージョンの MLV5 で作成されています。MLWebComponent をバージョンアップしてください。



66	グローバルチェックテーブルファイルのロード時にエラーが発生しました。
	グローバルチェックテーブルファイルが MLWebComponent より新しいバージョンの MLV5 で作成されています。MLWebComponent をバージョンアップしてください。
100	グローバル情報ファイルのロード時にエラーが発生しました。
	レイアウトに設定されているグローバル情報ファイル (.mlglex) が存在しないか、読取りが出来ません。レイアウトに設定されているファイルパスをご確認ください。他のアプリケーションで利用されている場合、アクセス権限がない場合は読取りできません。
101	ヘッダ札ファイルのロード時にエラーが発生しました。
	レイアウトに設定されているヘッダ札レイアウトファイル (.mlhtlx) が存在しないか、読取りが出来ません。レイアウトに設定されているファイルパスをご確認ください。他のアプリケーションで利用されている場合、アクセス権限がない場合は読取りできません。
102	テール札ファイルのロード時にエラーが発生しました。
	レイアウトに設定されているテール札レイアウトファイル (.mlhtlx) が存在しないか、読取りが出来ません。レイアウトに設定されているファイルパスをご確認ください。他のアプリケーションで利用されている場合、アクセス権限がない場合は読取りできません。
103	グローバル情報ファイルの設定がヘッダ・テール札と一致しません。
	レイアウトとヘッダ・テール札レイアウトに設定されているグローバル情報ファイル (.mlgle) の設定をご確認ください。
110	レイアウトファイルの保存時にエラーが発生しました。
	レイアウトファイルの保存ができません。書き込みが禁止されていないかご確認ください。
111	グローバル情報ファイルの保存時にエラーが発生しました。
	グローバル情報ファイルの保存ができません。書き込みが禁止されていないかご確認ください。
<b>カラープリンタ設定エラー：動作環境や設定を変更してください。</b>	
150	バーコード描画モジュールが読み込めません。
	①MLWebComponent をアンインストールして、再インストールしてください。 ②BCD32.dll を Ver.5.9.2.1 に更新してご確認ください。
151	レイアウトのプリンタタイプと Setting プロパティの値が不整合です。
	カラープリンタレイアウトを使用する場合、Setting プロパティは「DRVX:」（カラープリンタドライバ）を使用してください。
152	ヘッダ札のプリンタタイプが一致していません。
	レイアウトとヘッダ札ファイル (.mlhtlx) のプリンタタイプ（サトープリンタ/カラープリンタ）を合せてください。
153	テール札のプリンタタイプが一致していません。
	レイアウトとテール札ファイル (.mlhtlx) のプリンタタイプ（サトープリンタ/カラープリンタ）を合せてください。
154	カラープリンタドライバでこのレイアウトを使用することはできません。
	レイアウトのプリンタタイプがサトープリンタです。カラープリンタを使用する場合は、プリンタタイプを変更してください。
<b>例外エラー</b>	

201	プリンタコマンド生成以外の処理で予期しないエラーが発生しました。
	<a href="#">例外エラーについてをご参照ください。</a>
202	プリンタコマンド生成処理中に予期しないエラーが発生しました。
	<a href="#">例外エラーについてをご参照ください。</a>
<b>対象変数エラー：対象となる変数を再度設定し直してください。</b>	
300	日時変数の対象変数検索時にエラーが発生しました。
	日時変数の経過値に設定した変数がレイアウトに存在しません。
301	編集前複写変数の対象変数検索時にエラーが発生しました。
	編集前複写変数の複写対象に設定した変数がレイアウトに存在しません。
302	入力日時変数の対象変数検索時にエラーが発生しました。
	入力日時変数の経過値か位上対象に設定した変数がレイアウトに存在しません。
303	テーブル変換の対象変数検索時にエラーが発生しました。
	テーブル変換の変換対象キーに設定した変数がレイアウトに存在しません。
304	プライス丸め変数の対象変数検索時にエラーが発生しました。
	計算変数のプライス丸めで丸め対象に設定した変数がレイアウトに存在しません。
306	プライスチェックCD変数の対象変数検索時にエラーが発生しました。
	計算変数のプライスチェックCDで計算対象に設定した変数がレイアウトに存在しません。
307	クロスチェック変数の対象変数検索時にエラーが発生しました。
	分割変数のクロスチェックで比較対象に設定した変数がレイアウトに存在しません。
308	編集後複写変数の対象変数検索時にエラーが発生しました。
	編集後複写変数の複写対象に設定した変数がレイアウトに存在しません。
309	結合変数の結合子変数検索時にエラーが発生しました。
	結合変数の子項目に設定した変数がレイアウトに存在しません。
310	結合項目の親変数検索時にエラーが発生しました。
	結合変数の親項目となる変数がレイアウトに存在しません。
320	レイアウトに割り付いたローカル変数検索時にエラーが発生しました。
	デザインオブジェクトで設定したローカル変数がレイアウトに存在しません。
321	レイアウトに割り付いたグローバル変数検索時にエラーが発生しました。
	デザインオブジェクトで設定したグローバル変数がレイアウトに存在しません。
<b>編集エラー：発行時動作設定のエラー処理で「グラフィック変換エラー」が有効な場合に発生します。</b>	
330	貼付グラフィックで該当するグラフィックファイルがありません。
	指定されたパスにグラフィックがありません。設定したパスをご確認ください。
332	呼出グラフィックで該当する登録グラフィックがありません。
	指定した No.のグラフィックがグラフィックテーブルに登録されているかご確認ください。
333	変数グラフィックで指定したグラフィックがありません。
	指定先したパスやグラフィック No.に該当するグラフィックがあるかご確認ください。
<b>編集エラー：発行時動作設定のエラー処理で括弧内に記載した設定を有効にした場合に発生します。</b>	
400	日時変数編集時にエラーが発生しました。（日時項目編集エラー）
	日時の計算が出来ません。経過値を確認してください。
403	テーブル変換時にエラーが発生しました。（テーブル変換エラー）

	テーブル一覧に入力値に合致する値がありません。
404	ODBCテーブル変換時にエラーが発生しました。（ODBC テーブル変換エラー）
	テーブル一覧（ODBC）に入力値に合致する値がありません。
405	プライス丸め計算時にエラーが発生しました。（プライス丸め変換エラー）
	プライス丸め計算が出来ません。丸め対象変数の入力値を確認してください。
406	四則演算計算時にエラーが発生しました。（数式計算エラー）
	四則演算が出来ません。入力値を確認してください。
407	プライスチェックCD計算時にエラーが発生しました。（プライスC/D 計算エラー）
	プライスチェックCD 計算が出来ません。対象変数の入力値を確認してください。
408	¥・カンマ編集時にエラーが発生しました。（通貨カンマ編集エラー）
	テーブル変換や計算などで¥・カンマを付加するための桁数が足りません。入力値や変数の桁数を確認してください。
409	1文字補填編集時にエラーが発生しました。（1文字補填編集エラー）
	テーブル変換や計算などで1文字補填するための桁数が足りません。入力値や変数の桁数を確認してください。
410	CD計算時にエラーが発生しました。（C/D 計算エラー）
	モジュラス11でCD置換をエラー判定にした場合、置換できないCD値の場合にエラーが発生します。入力値を確認してください。
411	特殊編集時にエラーが発生しました。
	特殊編集で利用するDLL や関数が存在しないなど編集情報が足りません。再度設定し直してください。
413	税編集時にエラーが発生しました。（税編集エラー）
	消費税率が設定されていません。TaxRate プロパティを設定してください。
430	条件判定でエラーが発生しました。（条件判定エラー）
	条件変数の条件式で判定結果が計算できません。条件式や対象の変数に正しい値が設定されているかご確認ください。
431	条件判定結果で使用する変数が見つかりません。（条件判定エラー）
	条件変数の判定結果に指定した変数が存在するかご確認ください。
400	日時変数編集時にエラーが発生しました。（日時項目編集エラー）
	日時の計算が出来ません。経過値を確認してください。
610	用紙に印字できない項目が存在します。（用紙外に指定した項目が存在する場合）
	オブジェクトが用紙外に配置されています。ML デザインでエラーのオブジェクトがないかご確認ください。印字に必要なオブジェクトは属性設定の印字で「印字しない」に設定してください。
<b>描画エラー：発行時動作設定のその他で「画像描画をチェックして黒塗りの場合はエラーメッセージを表示する」を有効にした場合に発生します。</b>	
130	画像描画で黒塗りが発生しました。
	<ul style="list-style-type: none"> <li>Windows Update の不具合が発生している可能性があります。最新の更新プログラムを適用して改善されるかご確認ください。</li> <li>発行中に「ユーザーの切り替え」やリモートデスクトップ接続により操作する PC が切り</li> </ul>

	替わる場合に発生している可能性があります。Windows の制限事項になるため、発行中はログインユーザーや操作 PC が切り替わるような運用は避けてください。
<b>描画エラー：オブジェクトの描画に必要な情報がありません。</b>	
450	条件印字の判定でエラーが発生しました。
	条件印字で指定している入力チェックテーブルのファイルが存在しないか、ファイル内に該当のテーブルが見つかりません。
470	文字の書式タグが不正です。
	開始・終了タグの数が一致しているかなど書式タグの記述方法を確認してください。
471	文字の書式タグとして利用できない変数が設定されています。
	シンボル変数で書式タグ指定は利用できません。文字オブジェクトの変数を変更してください。
472	文字の書式タグが有効の場合は C/D 自動付加は利用できません。
	C/D 自動付加を無効にしてください。
473	文字の書式タグが有効の場合はフォントサイズの幅は任意に設定できません。
	フォントサイズの幅を自動にしてください。
600	レイアウトで使用している Windows フォントがありません。
	発行環境に該当する Windows フォントをインストールしてください。開発環境で一時的に発行テストを行う場合は、動作設定ファイルをご利用ください。動作設定ファイルの詳細は「第 4 章ワンポイントテクニック」をご参照ください。
601	レイアウトで使用しているバーコードの情報がありません。
	MLWebComponent をバージョンアップしてください。
<b>サポート外エラー：プロパティが利用できない組み合わせで設定されています。</b>	
802	発行枚数が指定されていません。
	PrnData プロパティ、SetPrnDataField メソッド、SetPrnDataRow メソッドで入力するデータを確認してください。
806	多面取りカラープリンタレイアウトの用紙サイズエラーです。
	多面取りのカラープリンタレイアウトで、作成時の用紙サイズが PrinterCaption で指定したプリンタドライバで使用できません。
820	レイアウトにヘッダ札が設定されていません。
	ヘッダ札を出力する場合は、発行時動作設定のヘッダ・テール札出力を設定してください。
821	レイアウトにテール札が設定されていません。
	テール札を出力する場合は、発行時動作設定のヘッダ・テール札出力を設定してください。
上記以外	レイアウトファイルがカラープリンタレイアウトであるかご確認ください。サトープリンタレイアウトを利用する場合は、「MLWebComponent リファレンスマニュアル」をご参照ください。

### ■例外エラーについて

MLWebComponent が発行処理中に予期せぬタイミングで OS から例外をキャッチした場合に発生します。例外の詳細は、エラー発生後にイベントビューアに出力されるイベント「SATO MLV5 MLWebComponent」でご確認ください。

- IE の再起動や発行間隔を空けることでエラーが復旧する場合  
メモリ不足が発生している可能性があります。[動作設定ファイルで「メモリ自動解放」](#)を有効にして、メモリ使用量が安定するかご確認ください。メモリの自動解放で改善されない場合は、タスクマネージャーでアプリケーションのハンドル、GDI オブジェクトなどメモリ使用量を確認して、アプリケーションのロジックを改善してください。

## 第4章

# ワンポイント テクニック

## 4-1

## MLWebComponent を利用する

■宣言 ■サーバ ■クライアント

## ■object タグで配置する

Web ページで利用するには、body 内に object タグを記述します。

属性	説明	変更の可否
id	ソース上で宣言する際に利用する名前	可能
classid	コントロールのクラス ID	不可
codebase	MLWebComponent インストーラの相対パス	可能
type	ブラウザに認識させる MIME の種類	不可
style="display:none"	Web ページ上にコントロールを表示させないスタイル	可能

（補足）object タグ内に文字列など記述すると、MLPreviewComponent の読み込みに失敗した際に表示させることができます。

```
<body>
  <object
    id="objMLWebComponent"
    classid="clsid:C137E319-41FE-4F0F-BD1F-190424FD7E2B"
    codebase="WebComponent-Installer-ja.exe"
    style="display:none"
  >WebComponent が使用できません。 </object>
  <object
    id="objFileAccessComponent"
    type="application/x-oleobject"
    classid="clsid:A3F14F83-0717-444B-9DE5-BFC3AF5C32E8"
    style="display:none"
  ></object>

  //ページの内容

</body>
```

※<object>タグ内の改行は必要ありません。説明のために改行しています。





③ Web サーバに MIME タイプを登録します。

ファイル名の拡張子	MIME の種類	設定の要否
.lic	application/octet-stream	必須
.mllayx	application/octet-stream	必須
*.mlglex	application/octet-stream	任意
*.mltblx	application/octet-stream	任意
*.mlchkx	application/octet-stream	任意

## ■クライアント環境を構築する

① Internet Explorer のセキュリティを設定します。

下記のいずれかの方法で設定してください。

### 1. 信頼済みサイトに登録する（推奨）

MLWebComponent を使用したページを信頼済みサイトに登録してください。

- ① 「インターネット オプション」を開く。
- ② セキュリティタブで「信頼済みサイト」を選択して、「サイト」をクリック。
- ③ 「この Web サイトをゾーンに追加する」に、ページの URL を入力し、「追加」をクリック。
- ④ 「閉じる」をクリック

### 2. 個別に変更する

信頼済みサイトに登録せずに、セキュリティを個別に設定する場合は、下記の項目を有効に設定してください。設定項目の名称は IE のバージョンにより若干異なります。

#### ○ActiveX コントロールとプラグイン

- ActiveX コントロールとプラグインの実行
- スクリプトを実行しても安全だとマークされている ActiveX コントロールの実行
- 署名された ActiveX コントロールのダウンロード

#### ○スクリプト

- アクティブ スクリプト

\*IE10 以降の場合、詳細設定の拡張保護モードが有効になっていると、ファイルのダウンロード（GetFile）に失敗します。拡張保護モードを無効にするか、セキュリティ設定の保護モードを無効にしてご利用ください。

② .NET Framework をインストールする。（Windows 7 の場合）

Windows 7 で .NET Framework 4.5.2 以降がインストールされていない場合は、別途 .NET Framework 4.5.2 以降のインストールをお願いいたします。JavaScript を利用して自動的にチェックしてインストールを案内することも可能です。利用方法はサンプルプログラムをご参照ください。

<https://docs.microsoft.com/ja-jp/dotnet/framework/install/guide-for-developers>

## 4-2

## ライセンス認証を行う

■ライセンスファイル ■認証する

## ■カラープリンタ利用ライセンス

MLWebComponent の「カラープリンタ発行機能追加」で、カラープリンタレイアウトが利用可能なクライアント（PC）は10台までです。10台を超える場合はカラープリンタクライアントライセンスを追加でご購入ください。

## ■ライセンスファイルと認証キー

MLWebComponent を利用するにはライセンス認証が必要です。認証に利用する「ライセンスファイル」「認証キー」は、保守ライセンスユーザサイトでダウンロードしてください。ログインに必要な「ライセンスID」と「パスワード」は、製品に同梱されている「ライセンス登録シート」を弊社にお送りいただく事で発行致します。（ご購入から1ヶ月間限定で仮ライセンスIDでのダウンロードが可能です。）「カラープリンタ発行機能追加」を追加でご購入頂いた場合は、ライセンスファイルを再度保守ユーザサイトよりダウンロードしてください。

- ① 保守ライセンスを登録（更新）する。
- ② 保守ユーザサイトにログインする。

<http://dcs.sato.co.jp/mlv5web/support/index.asp>

- ③ ライセンスファイルをダウンロードする。

## ■ライセンス認証を行う

ライセンスファイルをサーバに配置し、GetFile メソッドでクライアントにダウンロード、Authenticate メソッドで認証キーとパスを指定し、ライセンス認証を行います。

ライセンスファイルは、お客様毎に個別のシリアル番号で管理しております。認証キーおよびライセンスファイルは第三者に悪用されないように管理し、セキュリティ対策を行ってください。ライセンスファイルの流用が発覚した場合は、WebComponent の使用を停止させて頂く場合もございます。

### セキュリティ対策

- ライセンスファイルのファイル名、拡張子を変更する（ライセンスファイルの流用防止）
- ソースコードを HTML と分離する（ソースの直接参照防止） …など

```
var result;
var LicUrl = "(ライセンスファイルの URL)";
var LicDir = "(ライセンスファイルの保存先ディレクトリ)";

//ライセンスファイルをダウンロードする
result = MLWebComponent.GetFile( LicUrl , LicDir , 2 );
if (result != 0){
    window.alert("ライセンスファイルダウンロードエラー No." + result);
    return;
}
// ライセンス認証
result = MLWebComponent.Authenticate("(認証キー)" , LicDir + "¥¥MLWebComponent.lic");
if (result != 0) {
    window.alert("ライセンス認証エラー No." + result);
    return;
}
```

## 4-3

## バージョンを確認する

■プロパティ ■ファイルバージョン

## ■Version プロパティで取得する

Version プロパティで取得できる情報からバージョン番号を確認できます。

Multi LABELIST Web Component, Version **5.x.x.x**

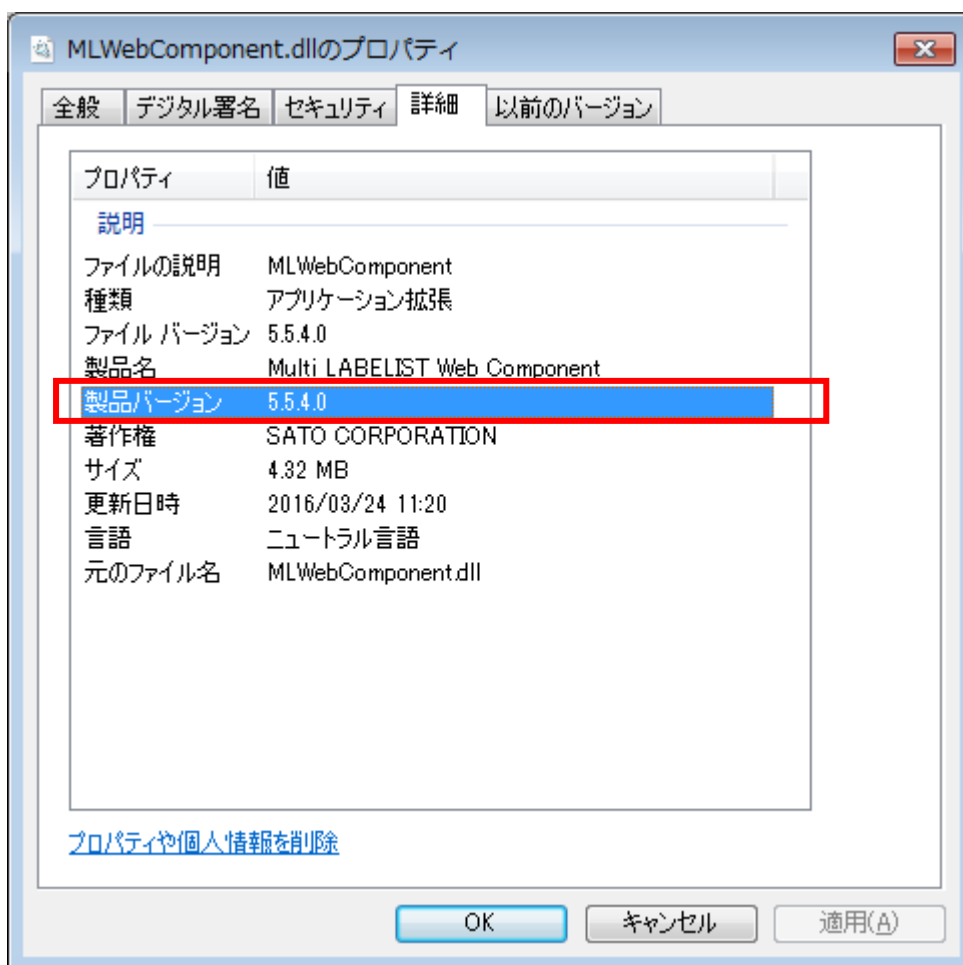
Multi LABELIST FileAccessComponent, Version **5.x.x.x**

## ■バージョン確認を行ってバージョンアップを通知する

クライアントにインストールされているバージョンを Version プロパティで確認することで、指定したバージョンより古い場合にバージョンアップを通知することができます。利用方法はサンプルプログラムをご参照ください。

## ■ファイルのプロパティで確認する

WebComponentがインストールされている環境で、インストール先にある「MLWebComponent.dll」のファイルプロパティで確認できます。



## 4-4

## バージョンアップを行う

■サーバ環境 ■クライアント環境

## ■サーバの WebComponent を更新する

- ①MLWebComponent のセットアップ「WebComponent-Installer-ja.exe」を入替える
- ②[Javascript でバージョン確認を行っている場合](#)、バージョン番号を書換える

## ※Ver.5.6.1.0 以前のバージョンから Ver.5.7.0.0 以降へ入替える場合

Ver.5.7.0.0 からインストーラに包含していた.NET Framework 4.0 のインストーラを削除しました。WebComponent を既にご利用のお客様には影響ございませんが、**Windows 7 で.NET Framework 4.5.2 以降がインストールされていない環境**に新規導入する場合には、別途.NET Framework 4.5.2 以降のインストールをお願いいたします。JavaScript を利用して自動的にチェックしてインストールを案内することも可能です。利用方法はサンプルプログラムをご参照ください。

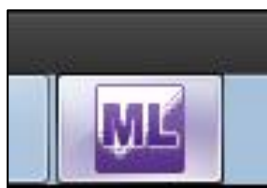
## ■クライアントの WebComponent を更新する

- ①MLWebComponent を利用したページを開く  
(既にページを開いている場合は、更新ボタンまたは F5 でページを更新します)

- ②「インストール」をクリック



- ③タスクバーの ML アイコンをクリック



④ 「インストール」 をクリック



⑤ 「完了」 をクリック



⑫ ページが表示されれば完了です。

## 4-5

## データを一括で入力する

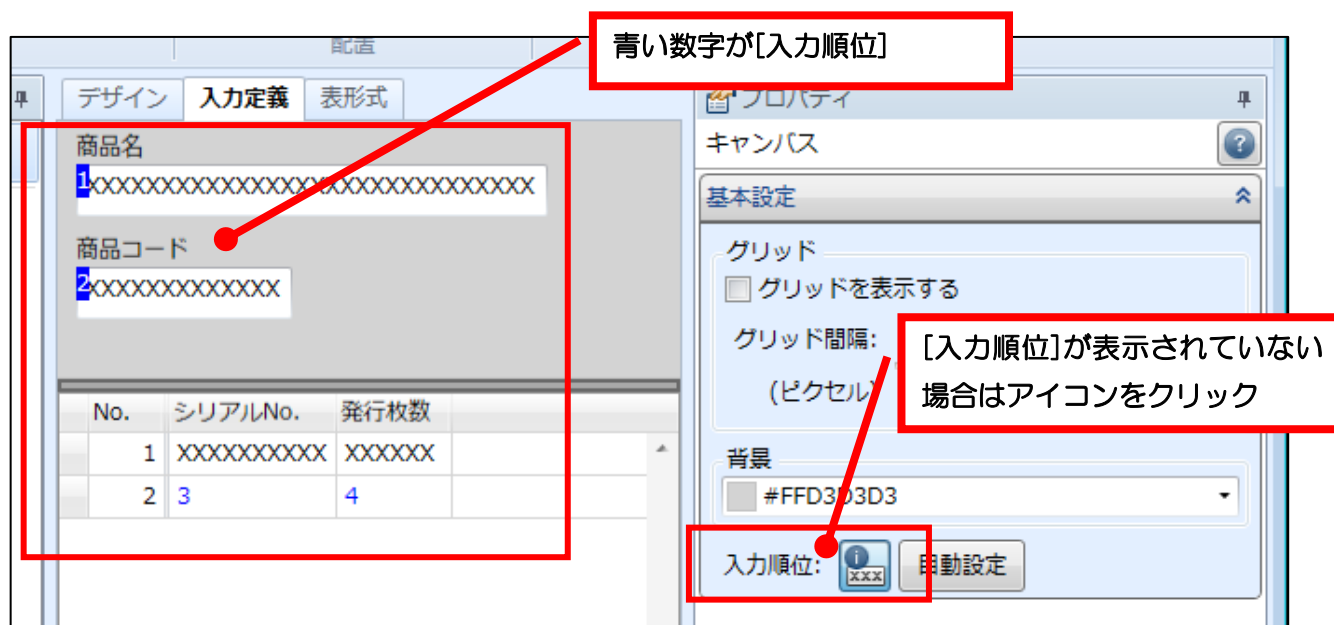
■入力順位 ■データ形式

指定したデータ形式に従って、一括でデータを入力します。プリンタドライバを利用する場合は、1つのデータだけでなく、複数のデータを一括して入力することもできます。

順番を意識せず変数名で入力したい場合はワンポイントテクニック「[データを変数名で指定して入力する](#)」をご確認ください。

## ■入力順位

まずデータを入力する順番を、MLデザインの入力定義で[入力順位]を表示させて確認します。



## • 使用例（TSV形式）

```
//入力データ指定
```

```
//商品名、商品コード、シリアル No.、発行枚数を順番に入力
```

```
MLWebComponent.PrnData =
```

```
    "SG412R-ex" + "¥t" + "490310999999" + "¥t" + "RX00007802" + "¥t" + "3";
```

## ■データ形式

デフォルトではタブ区切り（TSV形式）でデータを指定しますが、カンマ区切りのCSV形式やスペース区切りのPRN形式でデータを指定することもできます。

### • 使用例（CSV 形式）

```
//データ形式指定(CSV 形式)
MLWebComponent.PrnDataType = 1;

//入力データ指定
MLWebComponent.PrnData = "SG412R-ex,490310999999,RX00007802,3";

//括弧文字""（ダブルクォーテーション）を使って改行コードも入力可能
MLWebComponent.PrnData =
    ""SG412R-ex は、" + "¥n" + "高速発行が可能な 4 インチ堅牢型プリンタです。"";1"
```

### • 使用例（PRN 形式）

```
//データ形式指定(PRN 形式)
MLWebComponent.PrnDataType = 2;

//入力データ指定
MLWebComponent.PrnData = "SG412R-ex 490310999999 RX00007802 3";
```



## 4-6

## データを変数名で指定して入力する

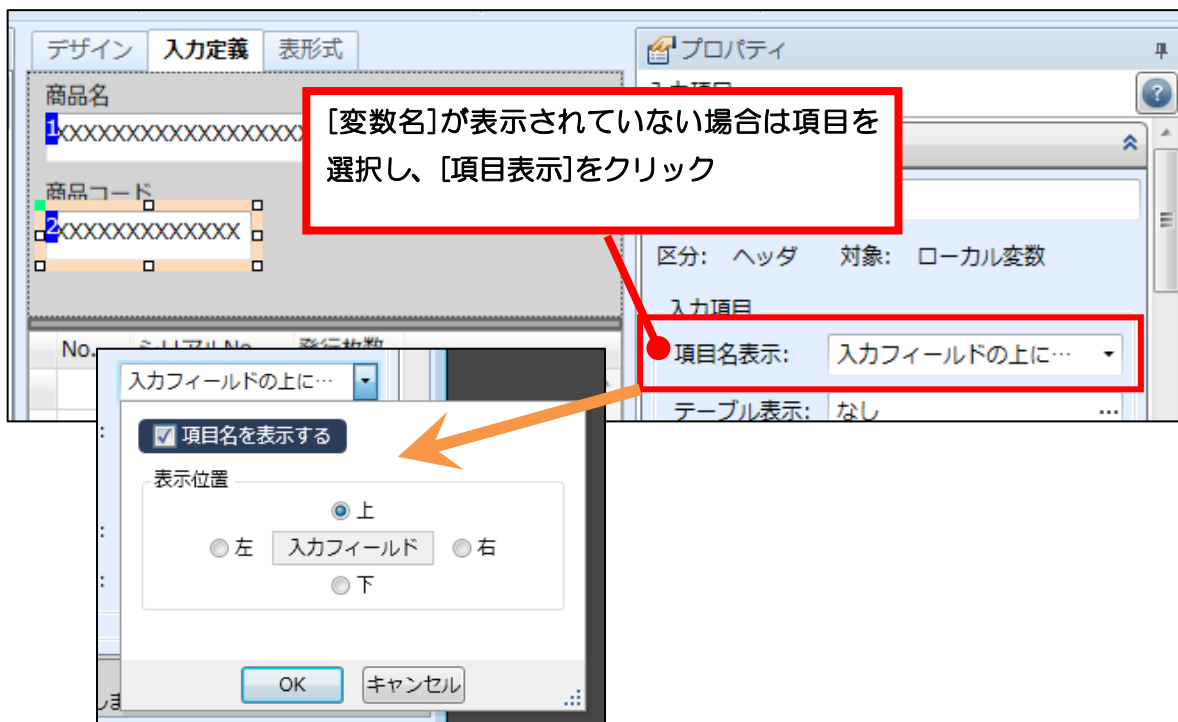
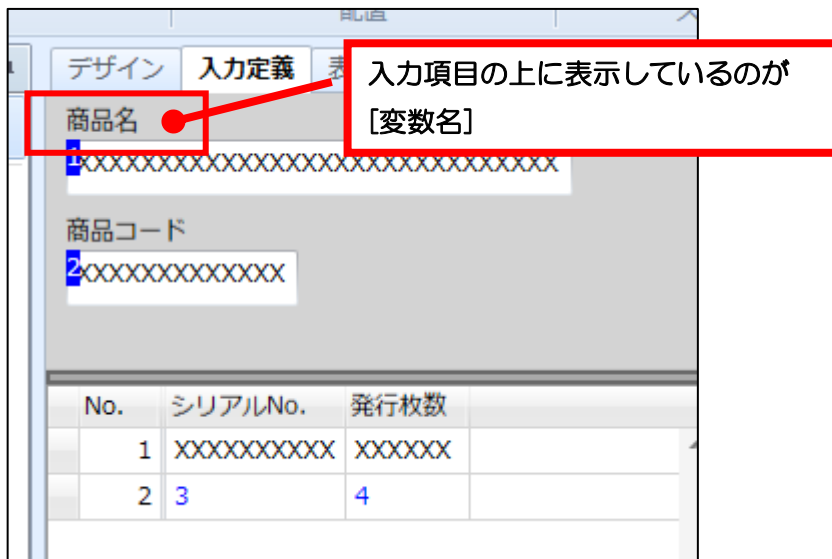
## ■変数名

変数名を指定してデータを入力します。変数名が共通しているレイアウトファイルを複数使う場合や入力順番が変更される可能性がある場合など、レイアウトファイルの入力順位を意識せずにデータを入力できます。

変数名を意識せず入力順位で簡単に入力したい場合はワンポイントテクニック「[データを一括で入力する](#)」をご確認ください。

## ■変数名

まずデータ指定に必要な[変数名]を、MLデザインの入力定義で確認します。



• 使用例

// 「商品名」を入力

```
MLWebComponent.SetPrnDataField("商品名", "SG412R-ex");
```

// 「商品コード」を入力

```
MLWebComponent.SetPrnDataField("商品コード", "490310999999");
```

// 「シリアル No.」を入力

```
MLWebComponent.SetPrnDataField("シリアル No.", "RX00007802");
```

// 「発行枚数」を入力

```
MLWebComponent.SetPrnDataField("発行枚数", "3");
```

## 4-7

## 動作設定ファイルを利用する

## ■動作設定ファイル

MLWebComponentと同じフォルダに動作設定ファイルを配置することで、プロパティでは設定できない拡張設定が利用できます。設定値（XMLタグ）がない場合は、初期値が利用されます。

## ■ファイル名

MLWebComponentSettings.xml

## ■格納先

MLWebComponent.dllと同一フォルダ

（通常は[プログラムフォルダ] – SATO – ML Web Component）

## ■文字エンコーディング

Unicode (UTF-8)

## ■書式（記述例）

```
<?xml version="1.0"?>
<MLWebComponentSettings
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xmlns:xsd="http://www.w3.org/2001/XMLSchema">
  <IsLog>false</IsLog>
  <LogFolder>D:¥Logs</LogFolder>
  <IsSheetCountError>false</IsSheetCountError>
  <TaxRate>5.0,8.0,・・・</TaxRate>
  <AlternativeFont>false</AlternativeFont>
  <DesignDefaultWindowsFontName>MS ゴシック</DesignDefaultWindowsFontName>
  <DesignDefaultWindowsFontSize>9</DesignDefaultWindowsFontSize>
  <IsFileForwardCompatibleError>true</IsFileForwardCompatibleError>
  <IsEnableGcCollectForced>false</IsEnableGcCollectForced>
  <GcCollectForcedThreshold>0</GcCollectForcedThreshold>
</MLWebComponentSettings>
```

## IsLog（ログ出力の有効・無効）

ログファイルを出力するかを設定します。

true	ログファイルを出力します。
false(初期値)	ログファイルを出力しません。

## LogFolder（ログ出力先）

ログファイルの出力先フォルダをフルパスで指定します。

## IsSheetCountError（発行枚数エラーの有効・無効）

発行枚数が指定されていない場合に、エラーとするかを設定します。**MLV4 WebEngine互換設定のため、通**

常は設定を変更せず、印字データで発行枚数を指定してください。

true(初期値)	PrnData プロパティまたはPrnDataArrayメソッドで発行枚数が指定されていない場合に、Output メソッドの戻り値が「802」となります。
false	発行枚数が未設定でも発行エラーとせず、発行枚数ゼロでプリンタコマンドを送信します。カット動作が機能しないなど問題が発生する可能性があります。通常は使用しないでください。

#### TaxRate（税率の設定）

税編集で使用する税率を設定します。設定値はTaxRateプロパティをご参照ください。

緊急対応用として用意している設定値です。通常はTaxRateプロパティで指定してください。

#### AlternativeFont（代替フォント使用の有効・無効）

レイアウトで使用しているWindowsフォントがない場合に、代替フォントを使用するか設定します。

true	代替フォントが利用されます。 デザイン時に指定したフォントと異なるフォントで印字されるため、文字の形状やサイズ、自動改行位置が異なります。通常は発行環境に該当フォントをインストールするか、フォントを変更するなどの対応を行ってください。
false(初期値)	Outputメソッドの戻り値が「600」となります。

#### DesignDefaultWindowsFontName（デフォルトフォントのフォント設定）

代替フォントに使用されるデフォルトフォントのフォントをWindowsフォント名で設定します。

本設定がない場合は、OSのデフォルトフォントが利用されます。

#### DesignDefaultWindowsFontSize（デフォルトフォントのフォントサイズ設定）

代替フォントに使用されるデフォルトフォントのフォントサイズをポイント数で設定します。

本設定がない場合は、OSのデフォルトフォントサイズが利用されます。

#### IsFileForwardCompatibleError（ファイルバージョンチェックの有効・無効）

レイアウトのファイルバージョンがMLWebComponentより新しい場合に発行エラーとするか設定します。

true(初期値)	発行エラー（No.61～66）となります。
false	エラーが発生しません。 レイアウトによっては、正常に印字されない危険性があります。ファイルバージョンチェックを無効にする場合は、レイアウトの新規追加・編集に充分注意し、変更後は運用前に必ず正常に印字されるかテストを行ってください。

#### IsEnableGcCollectForced（メモリ自動解放の有効・無効）

IEの使用メモリが閾値に達した場合にガベージコレクションによるメモリ解放（GC.Correct）を実行します。

閾値はGcCollectForcedThresholdで指定します。

true	自動解放を行います。
------	------------



Output	LayoutFile、Darkness、Speed、Offset、MultiCut、SortMark、
OutputHeader	HeaderTailSetting、Formoverlay、TaxRate、PrnDataType、PrnData
OutputTail	(PrnDataArray)
GetStatus	プリンタから受信したステータス文字列
Cut	なし
SendCancel	なし

## 4-8

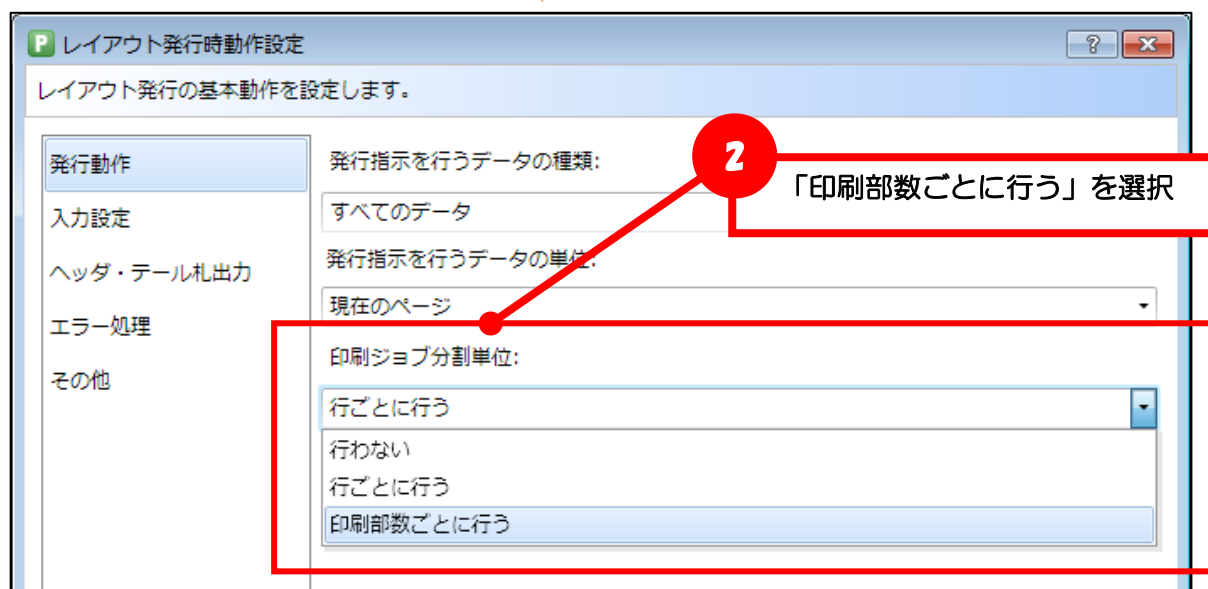
## 同一ラベルの発行速度を改善する

## ■印刷部数

ご利用プリンタ機種のプリンタドライバが印刷部数機能に対応している場合は、連番を利用しない同一ラベルの発行を速くすることが可能です。

## ■印刷部数機能を利用する

ML デザインまたは ML プリントで発行時動作を設定します。



発行枚数に関係なく常に 1 ページのデータとして出力されるため、発行速度が改善されます。プリンタドライバの印刷部数機能により、発行枚数と同じ枚数で発行されます。

The image displays two screenshots of the SATO GHS612R printer control panel. The top screenshot shows a document named 'C:¥sato¥GHS-LABEL.mllayx' with 10 pages and a size of 33.0 MB. The bottom screenshot shows the same document with 1 page and a size of 3.73 MB. A red callout box points to the page count and size information in the bottom screenshot, stating: '発行枚数によらず常に 1 ページで出力されますが、プリンタドライバの印刷部数機能で指定した枚数で発行されます。'

ドキュメント名	状態	ページ数	サイズ	受付時刻
C:¥sato¥GHS-LABEL.mllayx		10	33.0 MB	16:56:10 2017/0
C:¥sato¥GHS-LABEL.mllayx		1	3.73 MB	17:14:43 2017/0



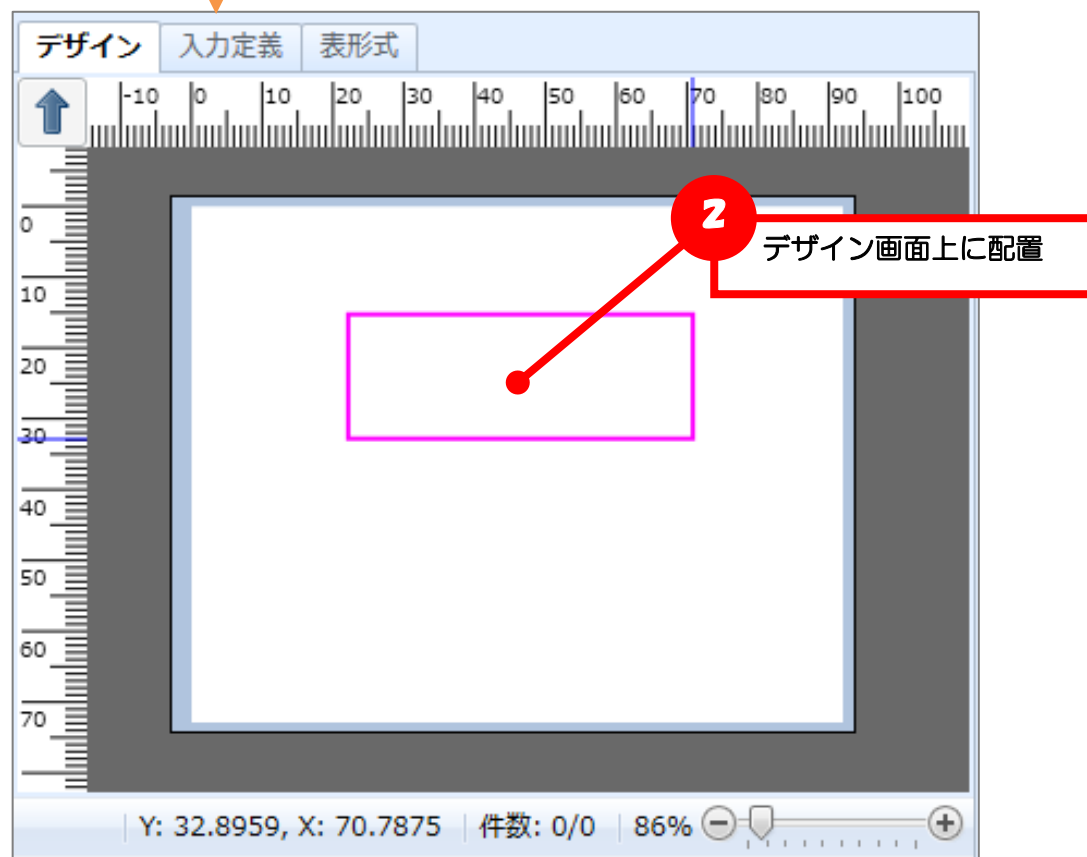
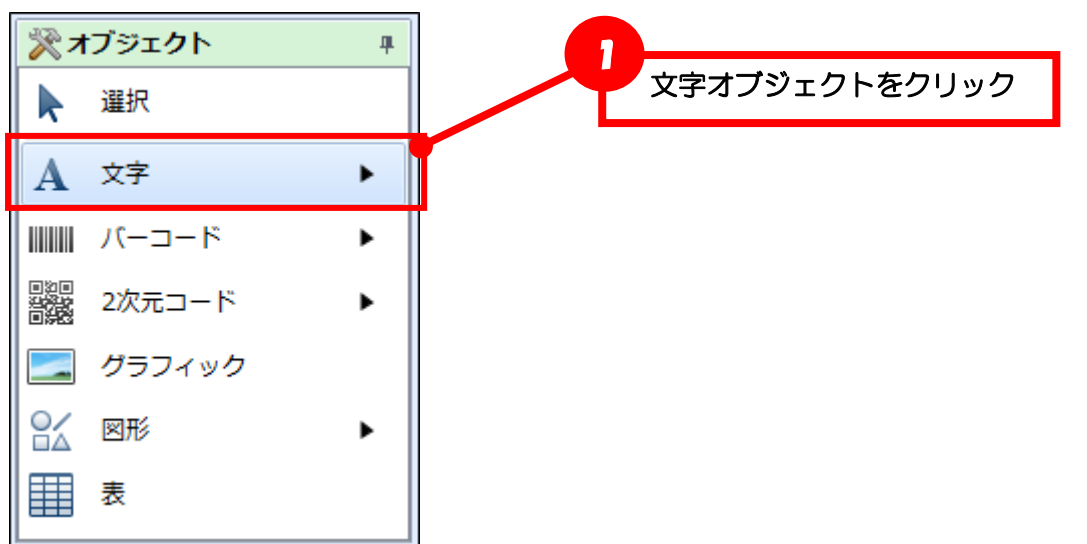
## 4-9

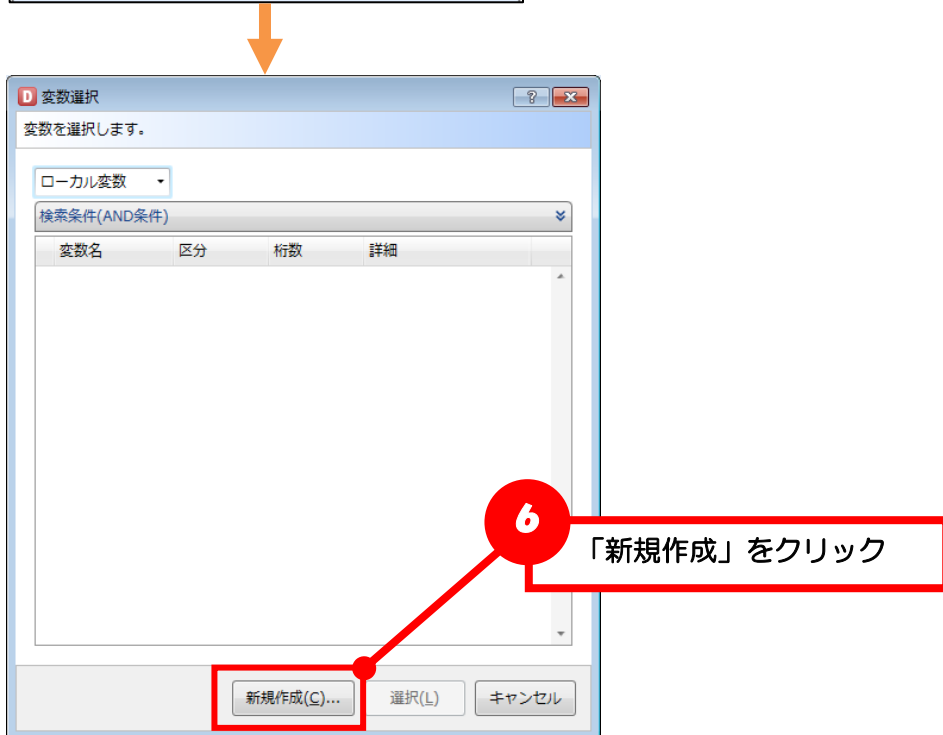
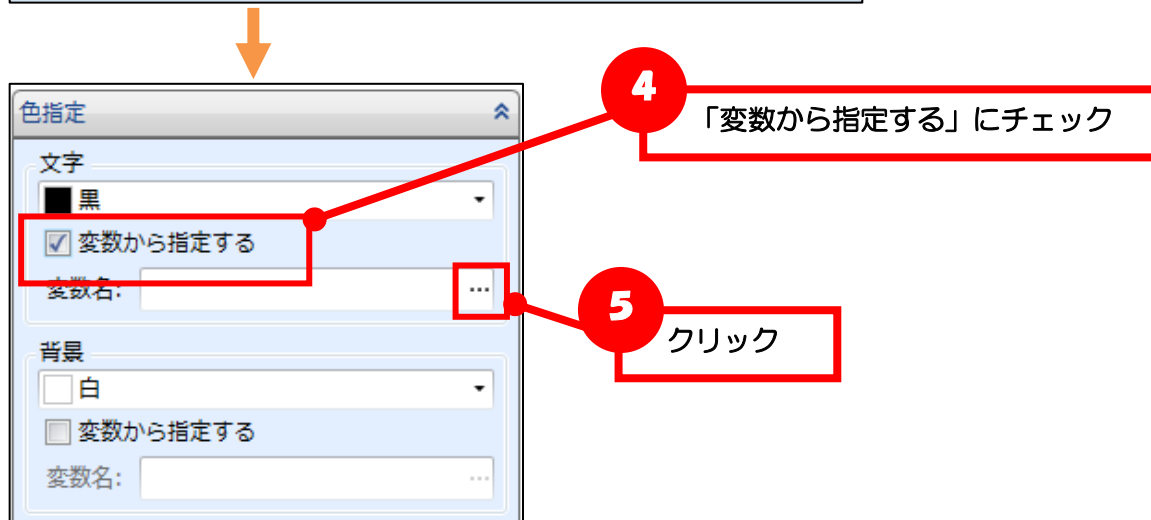
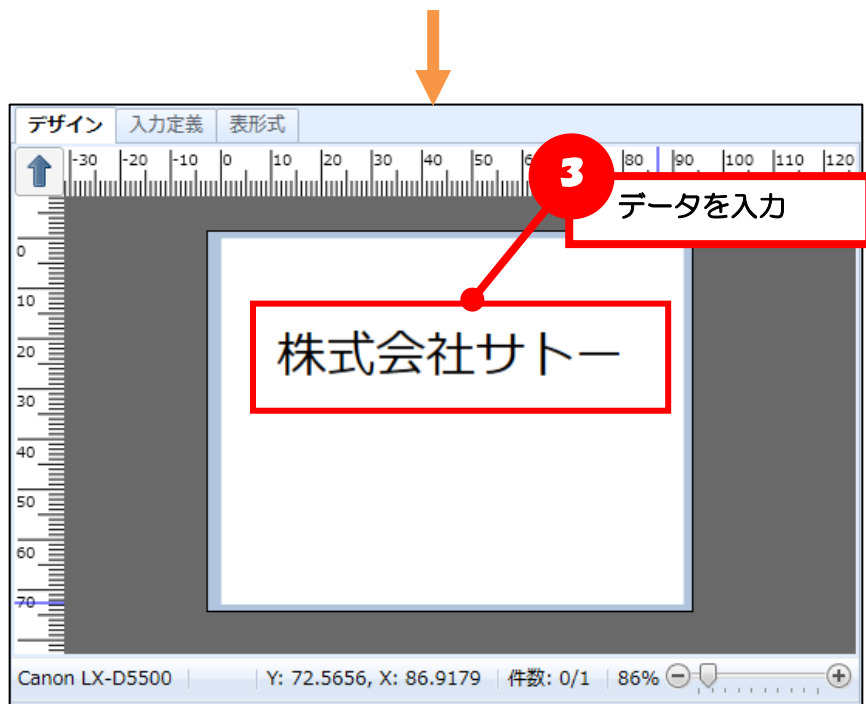
## 発行時に文字や枠線などオブジェクトの色を指定する

■オブジェクトの色指定 ■色の指定方法

オブジェクトの色指定を変数に設定することで、発行時に文字や枠線を任意の色に変更できます。

## ■発行時にオブジェクトの色を指定する





変数設定

変数の設定を行います。

入力変数種類: 文字

順序	編集内容	設定内容	設定詳細
1	テーブル変換	なし	...
2	税編集	なし	▼
3	カンマ編集	なし	▼
4	通貨編集	なし	▼
5	桁寄せ編集	なし	▼
6	前ゼロ補填	なし	▼

対象文字:  プレビュー(B) 編集後:

変数名: カラー 桁数: 6

OK キャンセル

7 変数名に「カラー」を入力

8 桁数に「6」を入力

9 「OK」をクリック

10 「テスト発行」をクリック

11 カラーに「0000FF」を入力  
MLWebComponent では PrnData や SetPrnDataField で指定

文字の色が青に変わります。

株式会社サトー

## ■色の指定方法

発行時に入力する色は 16 進数カラーコード（ARGB 値）またはカラーネームで指定します。

16 進数カラーコードは以下の書式で入力します。





A R G B （A：透明度 R：赤 G：緑 B：青）

入力桁	書式	入力例	色
3 桁	rgb	00F (⇒0000FF)	
4 桁	argb	5F00 (⇒55FF0000)	
6 桁	rrggbb	569012	
8 桁	aarrggbb	80569012	

カラーネームは、Windows Presentation Foundation で定義されている色名が使用可能です。

<https://docs.microsoft.com/ja-jp/dotnet/api/system.windows.media.colors>

### 色の指定例

カラーコード		カラーネーム	イメージ
6 桁	3 桁		
FF0000	FOO	red	
FF8000	—	—	
FFFF00	FFO	yellow	
009900	O9O	—	
0000CC	OOC	—	
6600CC	6OC	—	

## 第5章

# 注意事項

## 5-1

## 実行環境について

## ■ライセンスファイル ■IE

## ■ライセンスファイルと認証キーについて

ライセンスファイルは、お客様毎に個別のシリアル番号で管理しております。認証キーおよびライセンスファイルは第三者に悪用されないように管理し、セキュリティ対策を行ってください。ライセンスファイルの流用が発覚した場合は、MLWebComponent の使用を停止させて頂く場合もございます。

## セキュリティ対策

- ・ライセンスファイルのファイル名、拡張子を変更する（ライセンスファイルの流用防止）
- ・ソースコードを HTML と分離する（ソースの直接参照防止） …など

## ■ライセンス認証について

カラープリンタレイアウトを使用するには、カラープリンタ発行機能が有効なライセンスファイルで「ライセンス認証」が必要です。使用方法はワンポイントテクニック「[ライセンス認証を行う](#)」をご確認ください。

Setting プロパティで「DRVX:」（カラープリンタドライバ）を指定した OpenPort メソッドの戻り値は、ライセンス認証がされていない場合は「31」となり、カラープリンタ発行機能が無効の場合は「39」となります。

## ■IE の保護モードについて

IE の保護モードはセキュリティが高く、MLWebComponent の一部機能が使用できません。MLWebComponent を利用する場合は、IE のメニューで[ツール]-[インターネットオプション]を選択し、[セキュリティ]タブで[信頼済みサイト]に開発した Web サイトを登録してご利用ください。

## ■IE10 以降での利用

IE10 以降は HTML5 に準拠して HTML の記述方法が厳密にチェックされます。MLWebComponent の動作に問題ありませんが、MLWebComponent を宣言する object タグは head タグに記述できないため、body タグなど別の場所へ記述してください。その他、正常に動作しない場合は、[ツール]メニューから[開発ツール]を表示させ、[ブラウザモード]や[ドキュメントモード]を変更するか、[コンソール]タブに表示されるメッセージに従って HTML の修正を行ってください。

## ■Windows 7 での利用

Windows 7 で .NET Framework 4.5.2 以降がインストールされていない場合は、別途 .NET Framework 4.5.2 以降をインストールしてください。

<https://docs.microsoft.com/ja-jp/dotnet/framework/install/guide-for-developers>

## ■.NET Framework のバージョンについて

古いバージョンの .NET Framework を利用すると、MLWebComponent が正常に動作しない場合があります。Microsoft でサポートされているバージョンをご利用ください。

<https://docs.microsoft.com/ja-JP/lifecycle/faq/dotnet-framework>

### ■サーバサイドアプリケーションでの利用

MLWebComponent はクライアントで動作させることを前提に開発・提供しており、マニュアル・サンプルは全てクライアントでの利用方法で記載しております。ASP.NET などサーバサイドアプリケーションでの利用は技術的には可能ですが、動作確認は行っておりません。ご利用の際は事前に十分な検証をお願いいたします。

### ■プリンタドライバの設定

プリンタドライバの[詳細設定]で[プリンタに直接印刷データを送る]に設定している場合、OpenPort に失敗し正常に動作しません。[印刷ドキュメントをスプールし、プログラムの印刷処理を高速に行う]に設定を変更してください。

### ■初回起動時・レイアウトファイルロード時の遅延

MLWebComponent で使用している.NET Framework のライブラリがロードされるため、アプリケーション起動後に初めて MLWebComponent を呼び出す時、初めてレイアウト情報を読み込む時（Output メソッドや GetPriter メソッドなど）に処理が遅延することがあります。

運用上で遅延が障害となる場合は、アプリケーションの起動時に MLWebComponent で仮のレイアウトを呼び出すなどの方法で回避をお願い致します。

**5-2****レイアウト情報について**

レイアウト情報に関する注意事項を説明します。

**■入力定義**

レイアウトファイルの入力定義（初期値や桁数制限、入力チェックなど）は、MLプリントの発行画面で使用される設定項目です。お客様のアプリケーションで入力定義の設定を利用する場合は、GetInputFieldsで取得してください。

**■連番変数**

レイアウトファイルに保存されている連番値の取得および変更はできません。MLV5で確認・変更を行ってください。お客様のアプリケーションで連番値の管理が必要な場合は、レイアウトファイルの連番保存方法を「なし」でご利用ください。

**■文字の使用について**

カラープリンタレイアウトのラベルデザイン上に文字を配置し、MLWebComponent、MLPreviewComponent で利用する場合、発行またはプレビュー実行中に、お客様のアプリケーションで.NET Framework(WPF)の文字描画（DrawingContext.DrawTextなどの文字描画API）を同時に実行しないでください。Windows内部で発生する現象により、まれに入力した文字と異なる文字が印字される可能性があります。