
Multi LABELIST Component 練習マニュアル - Visual Studio 2010 -

株式会社サトー

2015年4月7日

はじめに

この度は、「Multi LABELIST Component」(以下 MLComponent)をご利用いただき誠にありがとうございます。MLComponent 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、お客様のアプリケーションにラベル・タグ発行機能を追加するために開発した.NET コンポーネントです。

MLV5 で作成したレイアウトファイルをもとに、自由度の高いラベル／タグ発行システムを構築していただくために MLV5 の一部の機能は省かせていただきましたが、USB、LAN、COM (シリアルポート)、および弊社製プリンタドライバと、出力デバイスを問わない設計が可能です。ステータス監視機能をサポートしており、プリンタがどのような状態になっているか取得することができます。

本マニュアルでは MLV5 を利用したラベルデザイン (レイアウトファイル) の作成から、MLComponent を利用した発行アプリケーションの開発まで簡単な手順でご説明しております。プロパティ・メソッド毎の詳細な説明は、「[MLComponent リファレンスマニュアル](#)」、用途別の利用方法については、「[MLComponent テクニックマニュアル](#)」も合わせてご参照ください。

◆本マニュアルは以下の環境で作成しています。

ソフトウェア	Visual Studio 2010 Premium (Version 10.0.40219.1 SP1Rel)
	Multi LABELIST V5 (Ver.5.0.2.0)
	Multi LABELIST Component (Ver.5.0.2.0)
OS	Windows 7 Professional Service Pack 1

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ください。
- ・ 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
第 1 章 レイアウト作成	4
1.MLV5 でレイアウトを作成する	5
■ML デザインの起動	5
■レイアウトの基本設定	8
2.オブジェクトでデザインを作成する	10
■基本操作	10
■文字オブジェクトの作成	11
■バーコードオブジェクトの作成	13
■2 次元コードオブジェクトの作成	15
■表オブジェクトの作成	16
3.変数でデータを入力する	19
■変数の作成	19
■変数とオブジェクトの紐付け	23
■入力順序の確認	25
第 2 章 アプリケーション開発	27
1.MLComponent を利用する	28
■発行画面の作成	28
■MLComponent の参照設定	29
2.発行方法を定める	30
■発行方法	30
3.ラベルを発行する	31
■発行処理（プリンタドライバ出力）	31
■発行処理（インターフェース出力）	32
■ラベル発行	33
4.アプリケーションを配布する	34
■インストーラを作成する	34

第1章

レイアウト作成

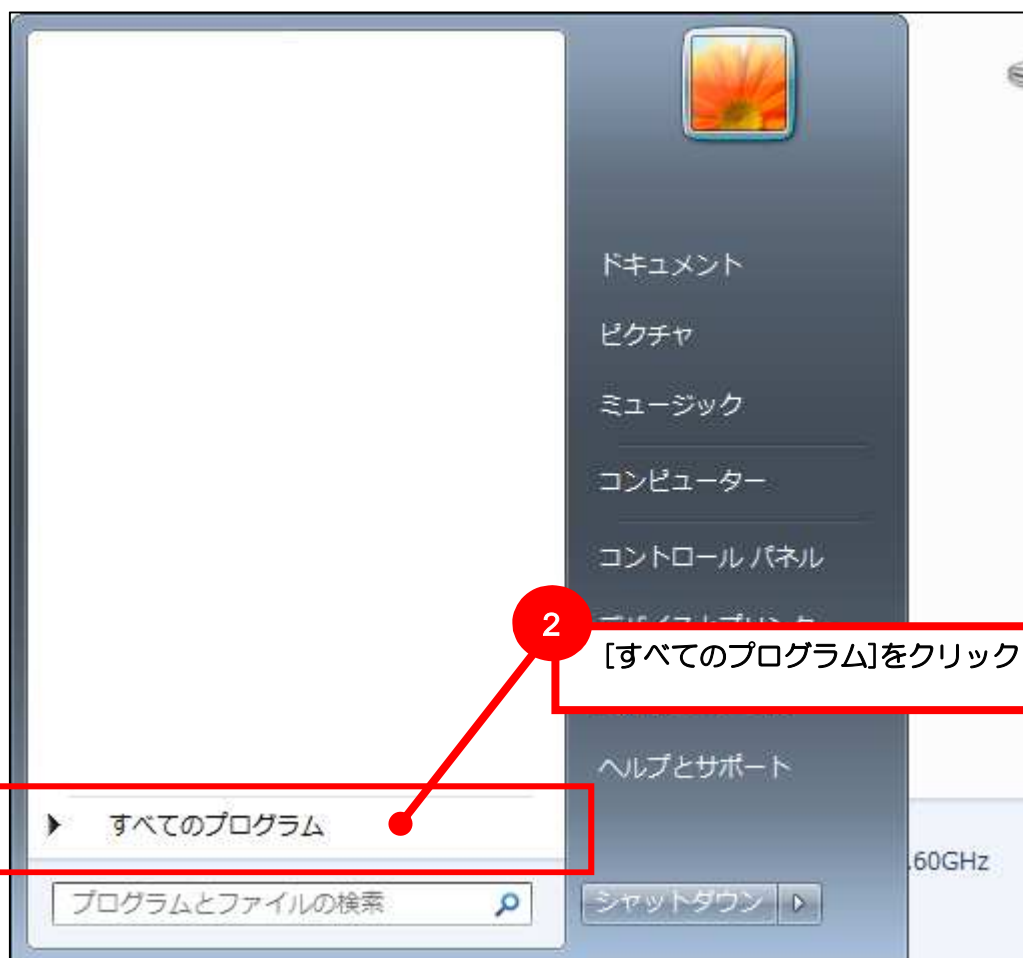
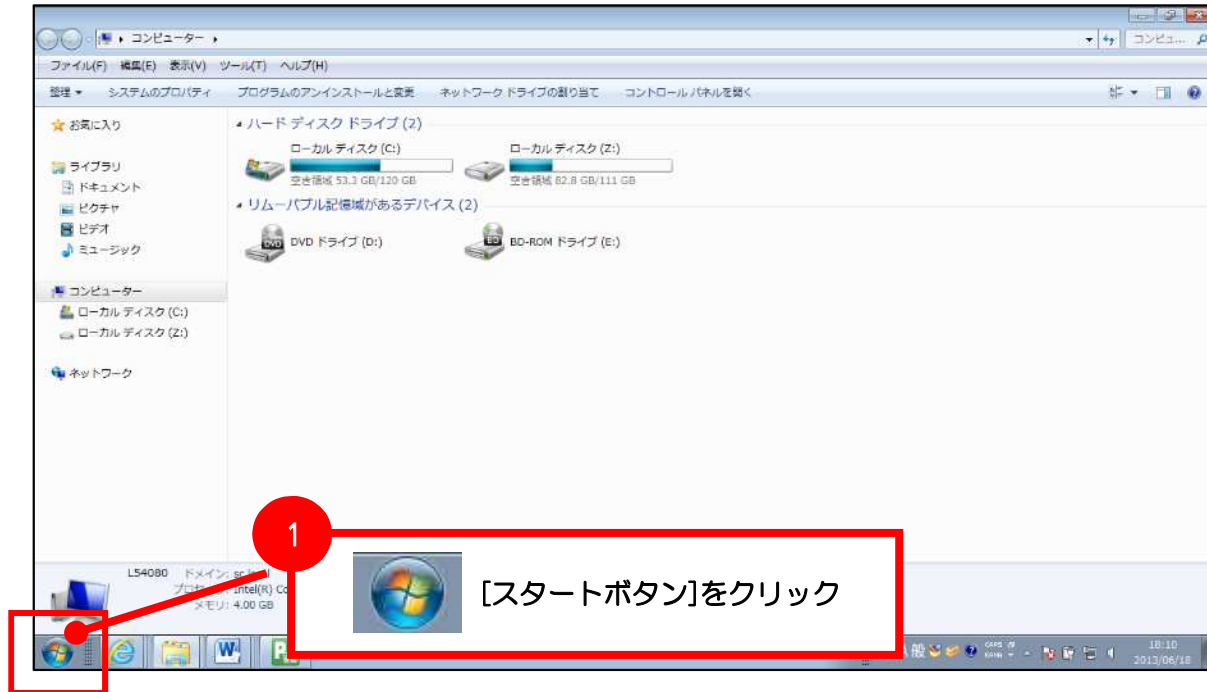
1

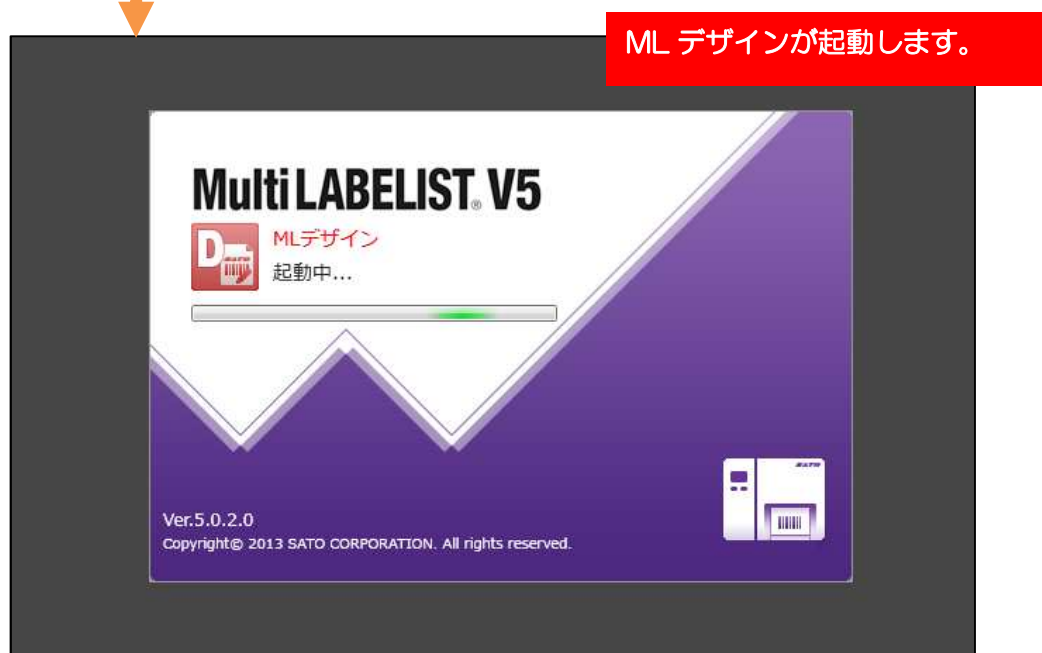
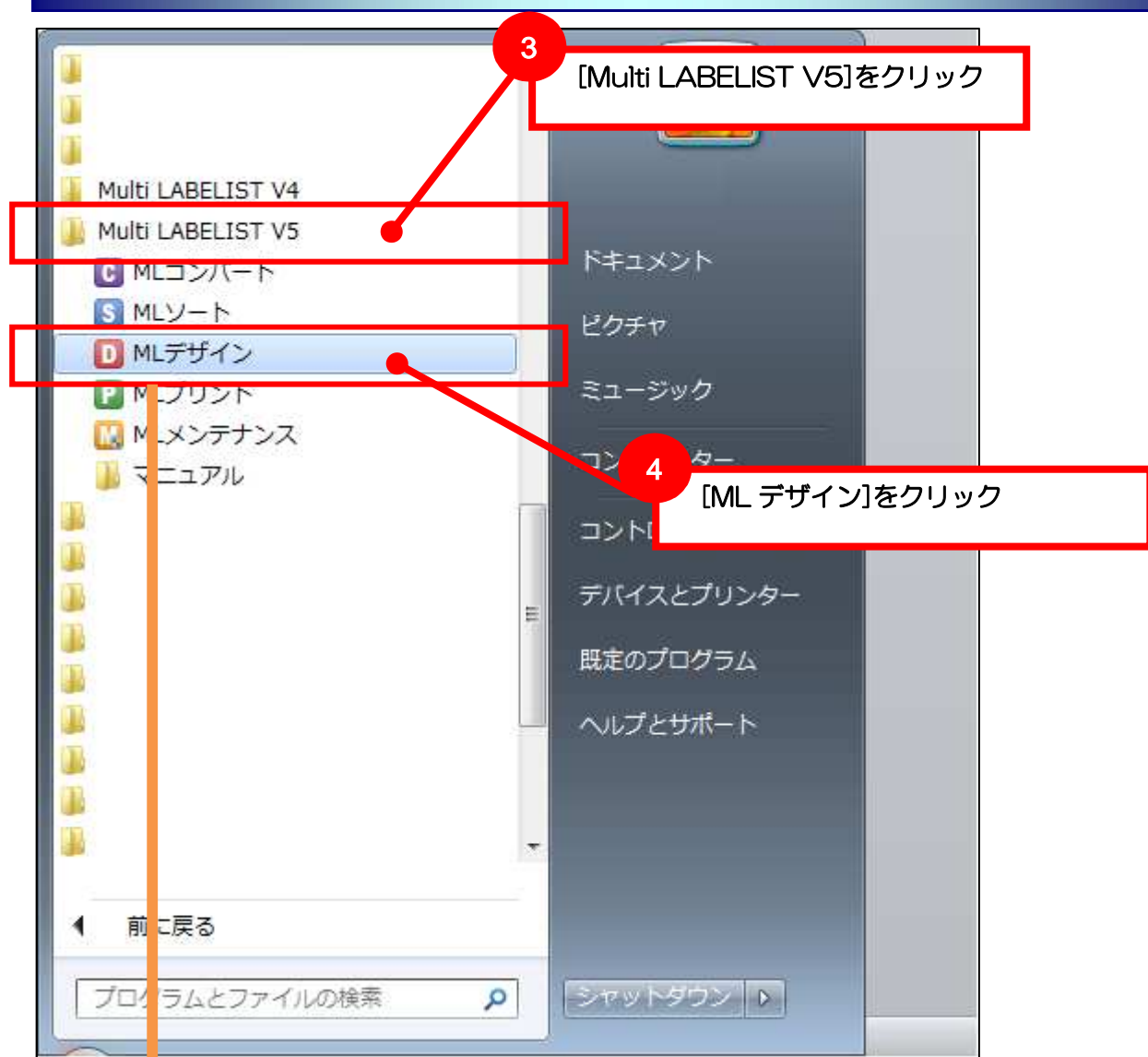
MLV5 でレイアウトを作成する

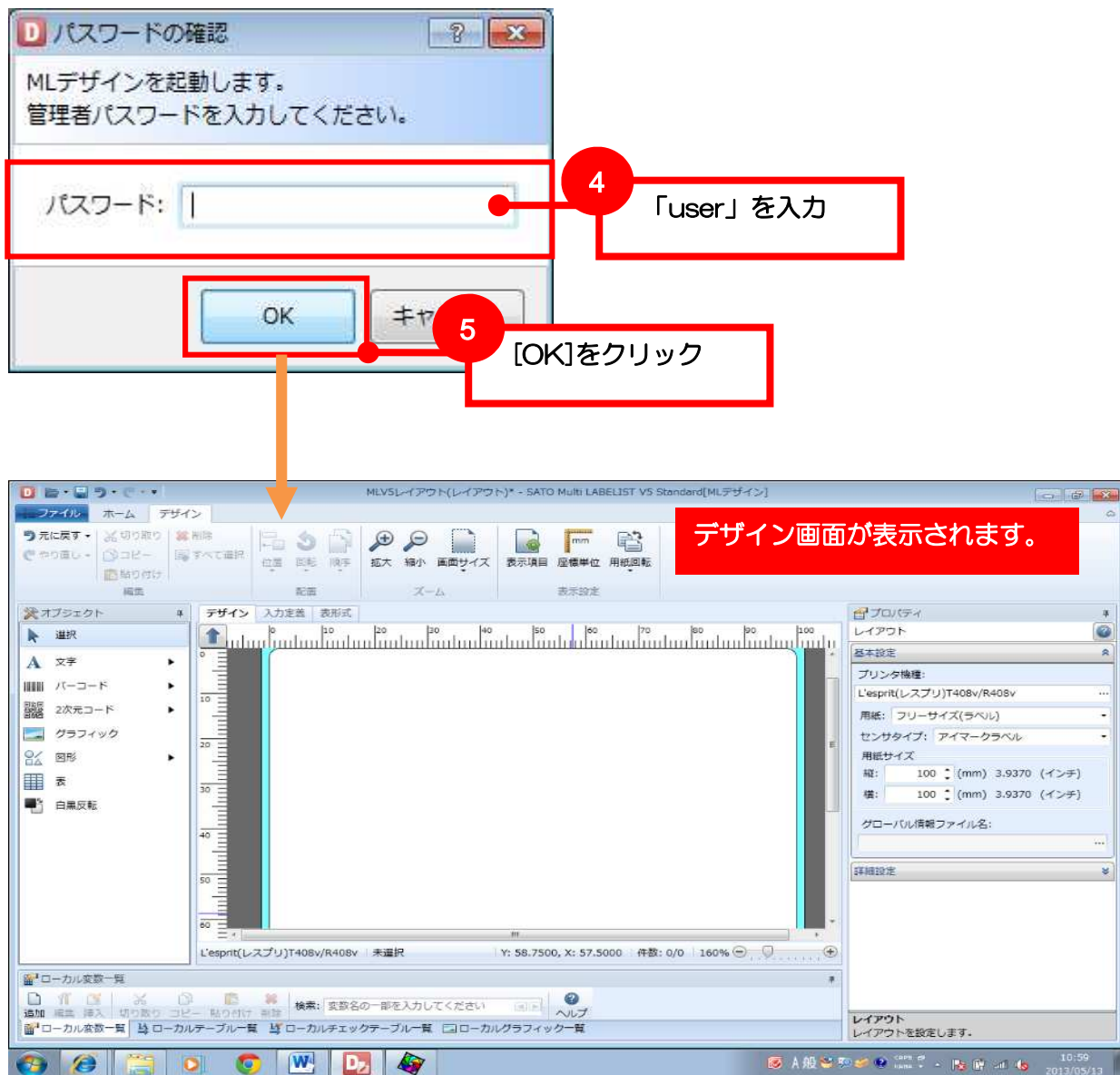
■ML デザイン ■基本設定

MLV5を使ってレイアウトを作成します。

■ML デザインの起動



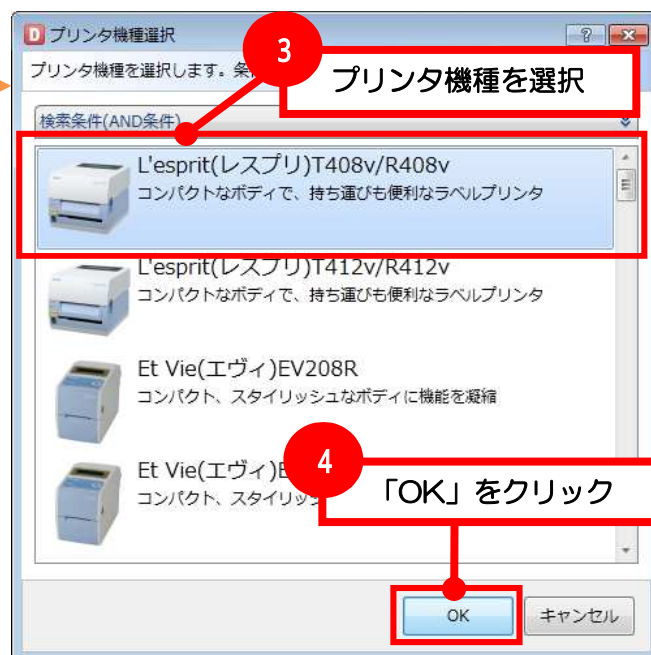
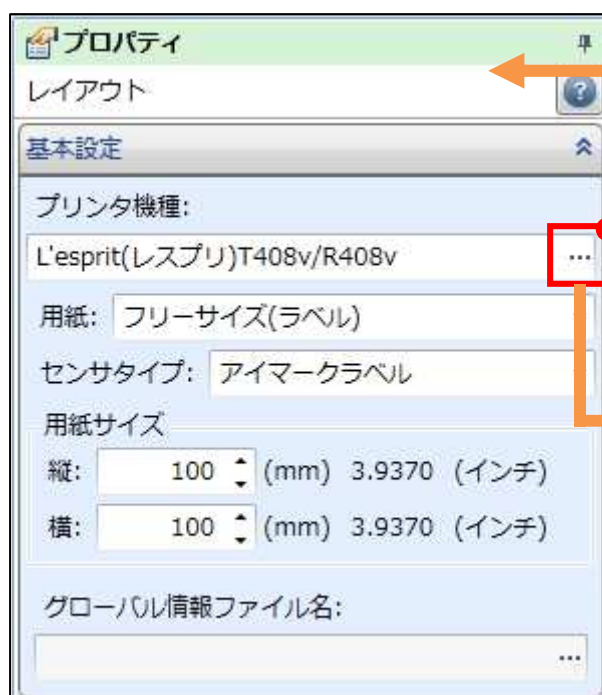
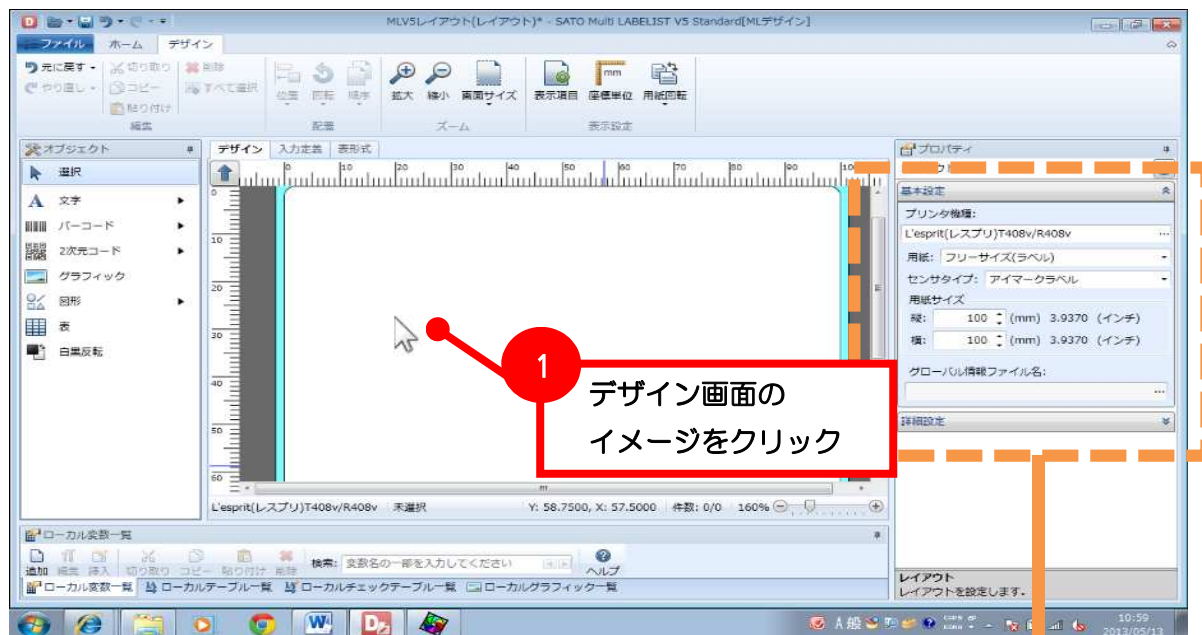




[確認] ライセンスキー (HASP)

MLV5 用のライセンスキー (HASP) が接続されていないと、「パスワードの確認」画面の後に「Trial 版で動作します」と表示され、一部動作が制限されます。必ずライセンスキー (HASP) を接続してご利用ください。

■レイアウトの基本設定



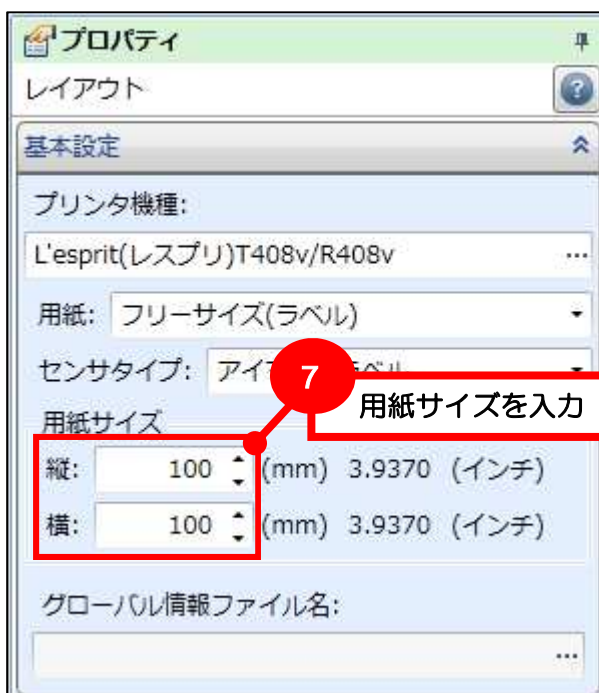
[確認] プリンタ機種

ご利用のプリンタを選択してください。プリンタ機種名はプリンタの背面に張られている銘版ラベルのMODELで確認できます。



[確認] センサタイプ

アイマークとは台紙部分に黒いマークがあるタイプのことを示します。ギャップは台紙部分にマークが何も入っていないタイプのことで、台紙部分（ギャップ）を検地して用紙サイズを判断します。



[確認] 用紙サイズ

ラベルの場合、ラベルの部分のみで台紙は含みません。タグの場合、台紙はありませんのでタグ全体のサイズを指定します。

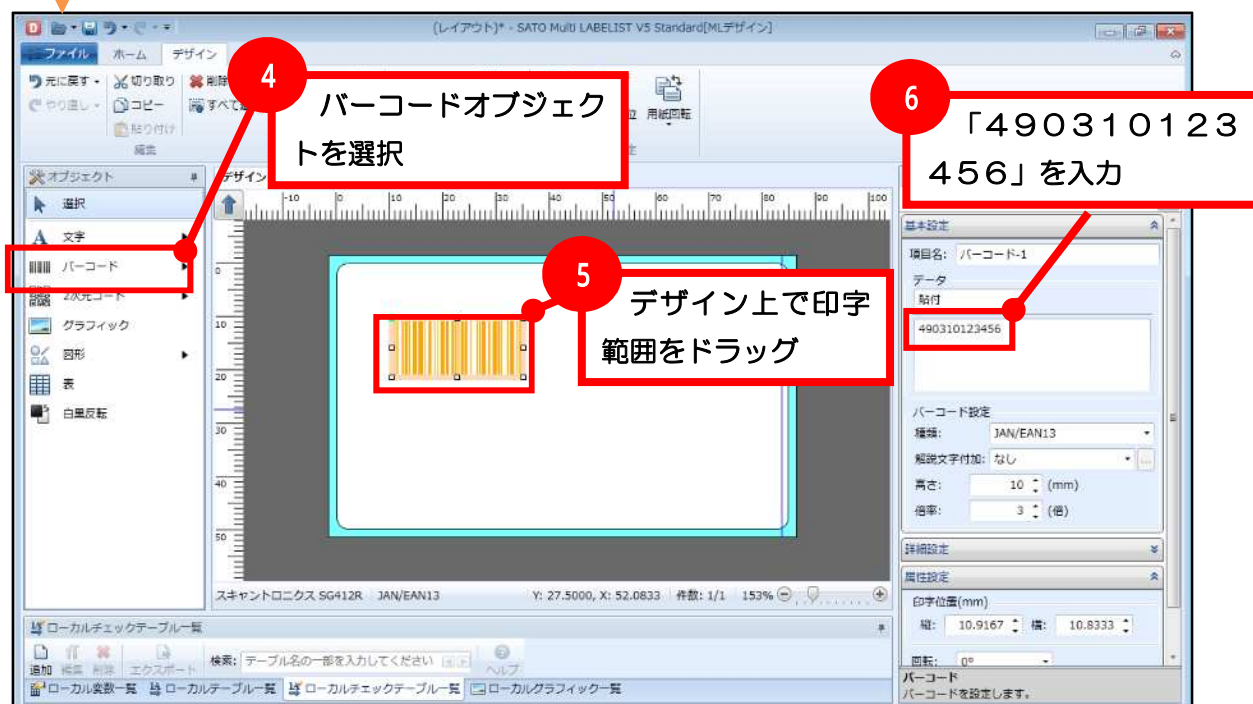
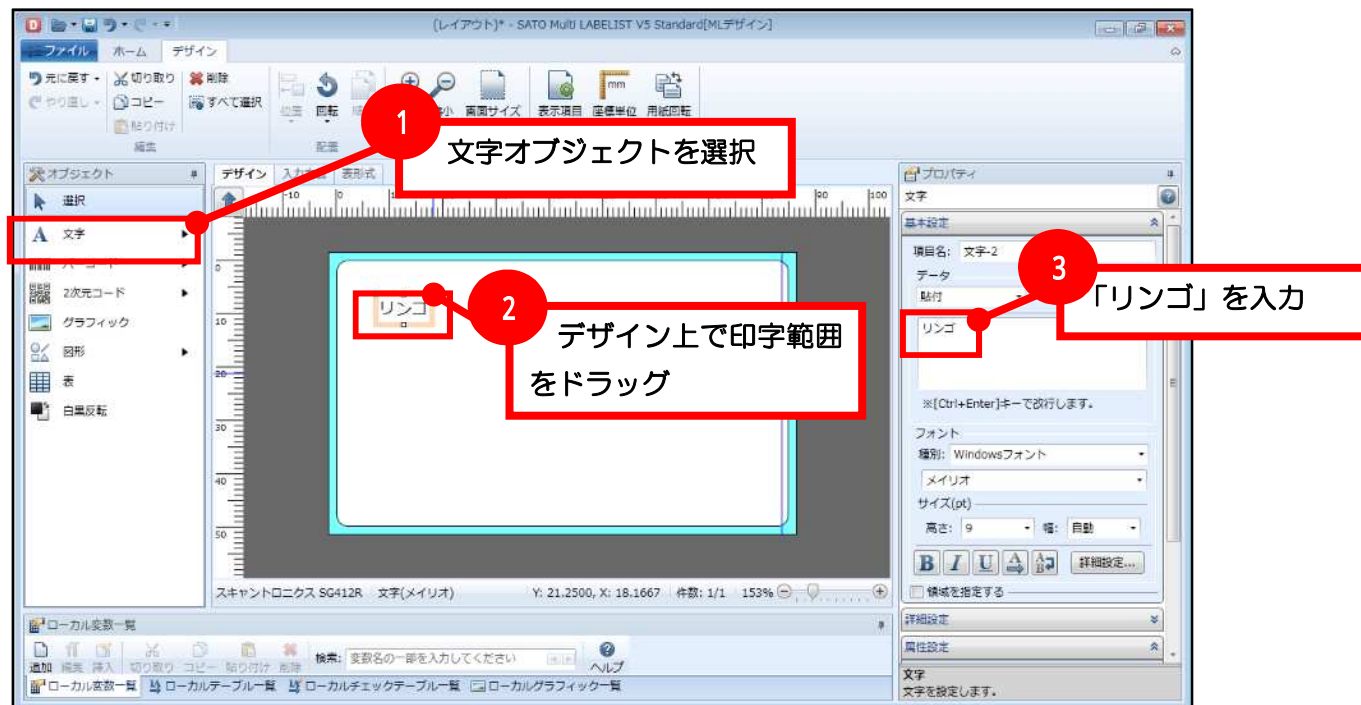
2

オブジェクトでデザインを作成する

■基本操作 ■文字 ■バーコード ■2次元コード ■表

■基本操作

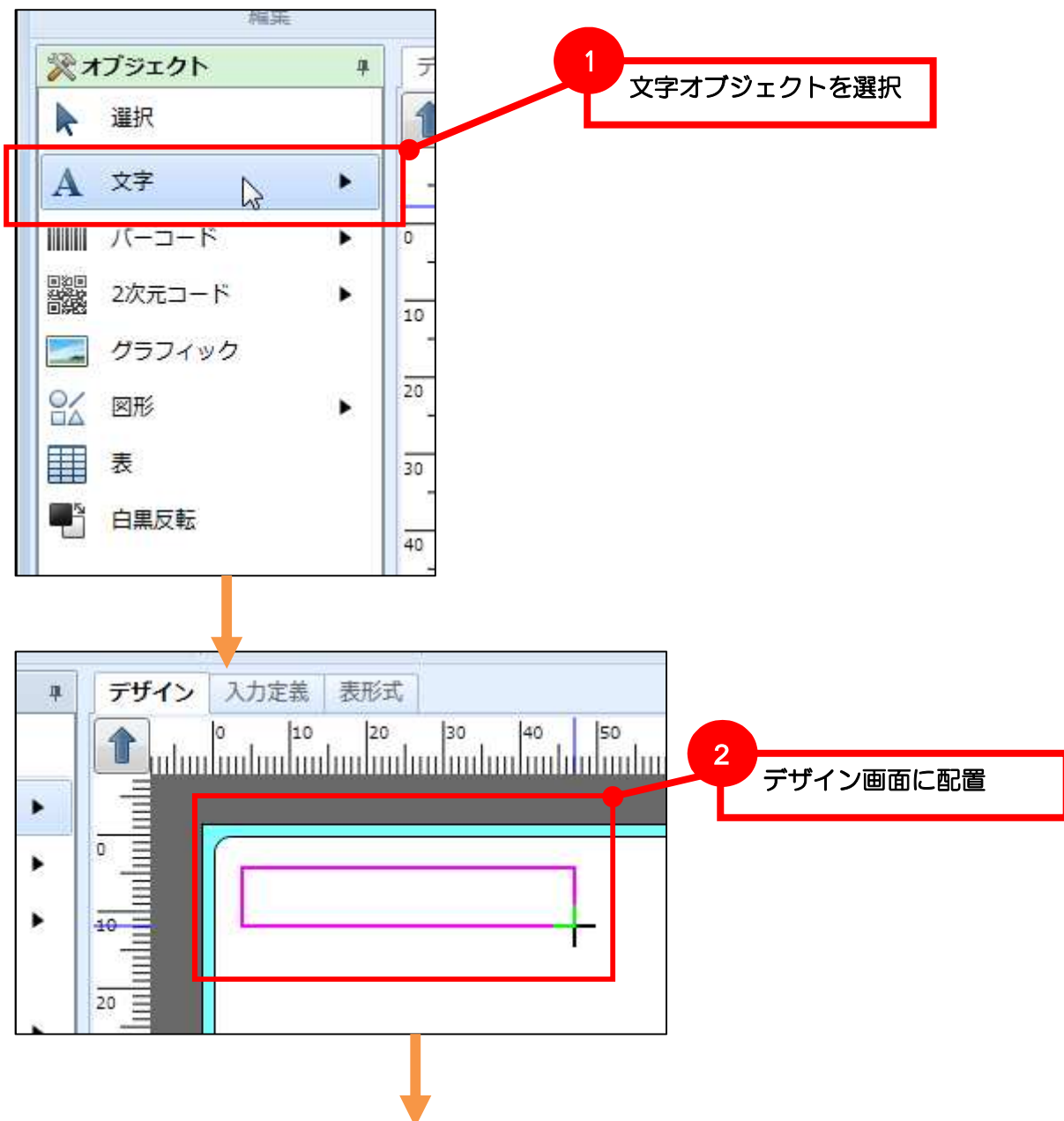
オブジェクトバーからデザインオブジェクトを選択して、デザイン画面上に配置します。

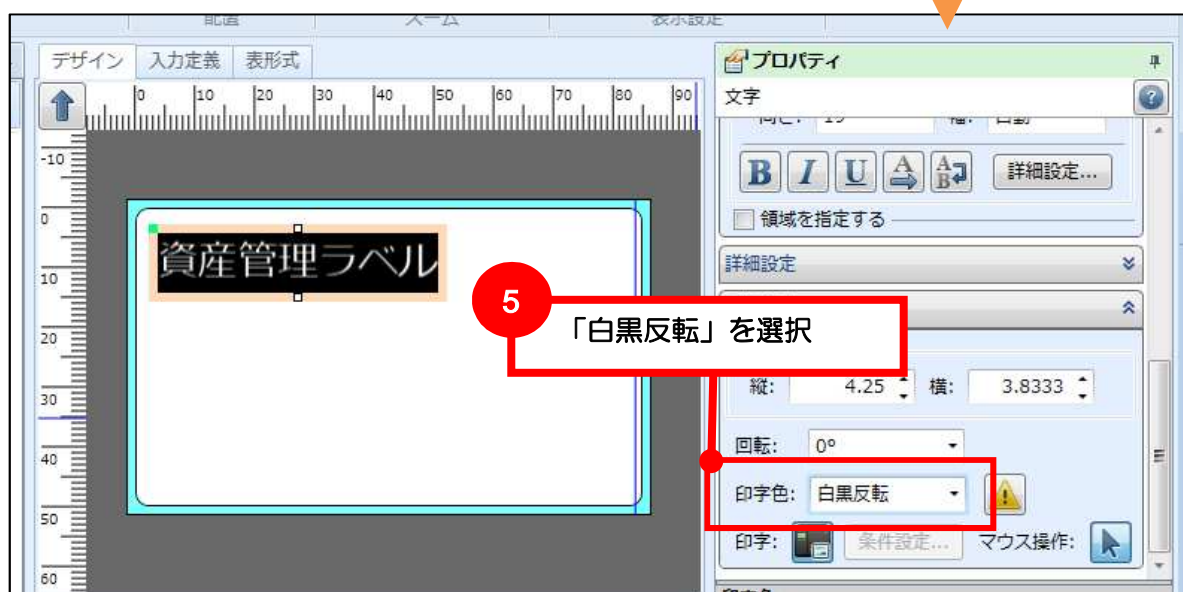
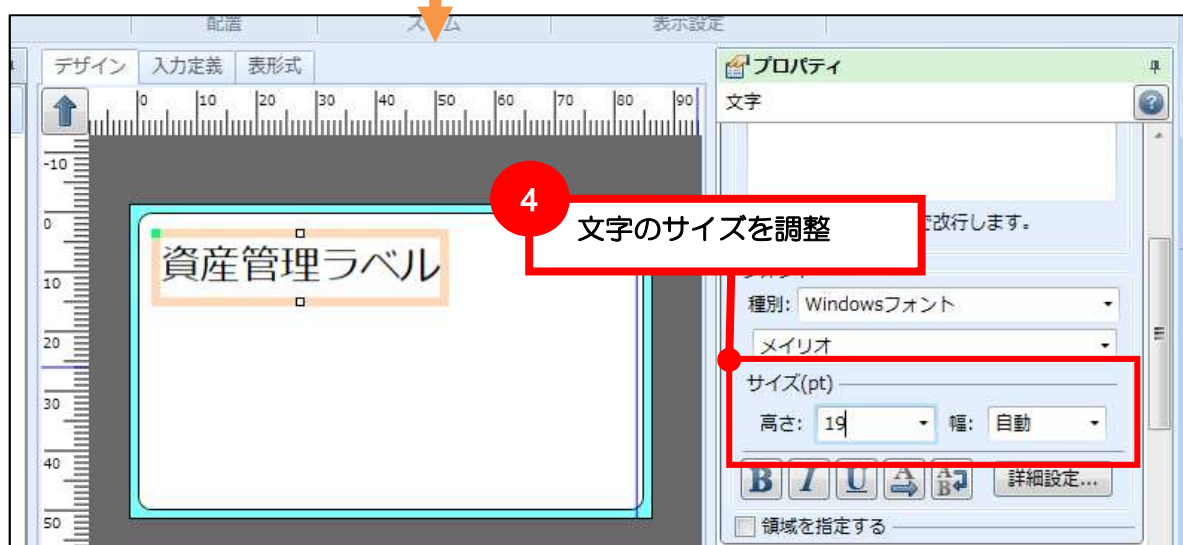
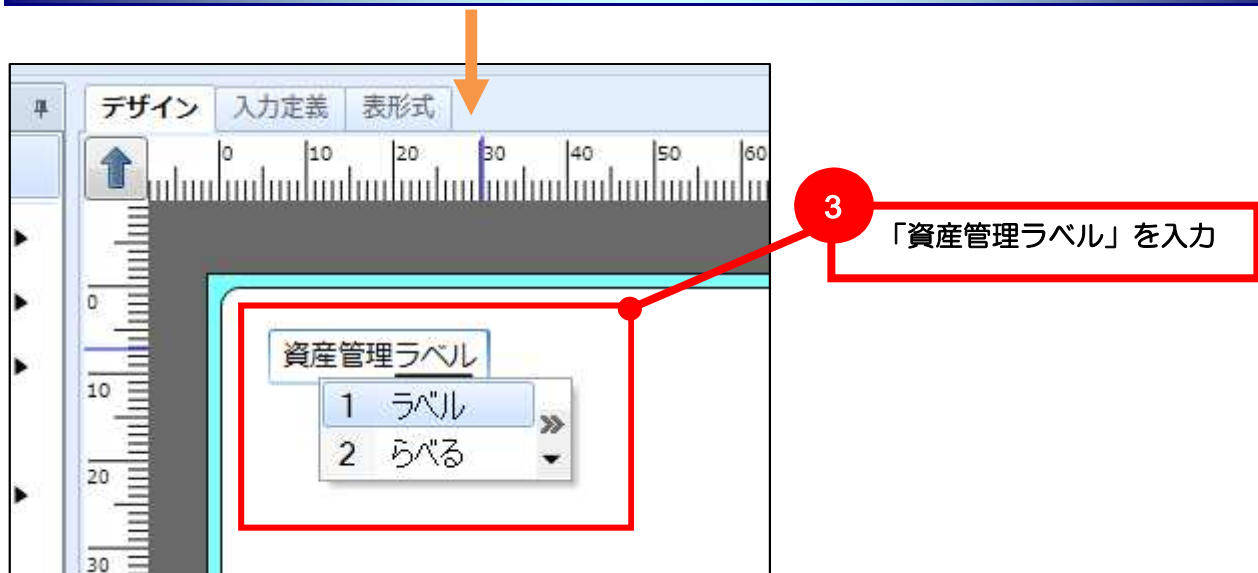


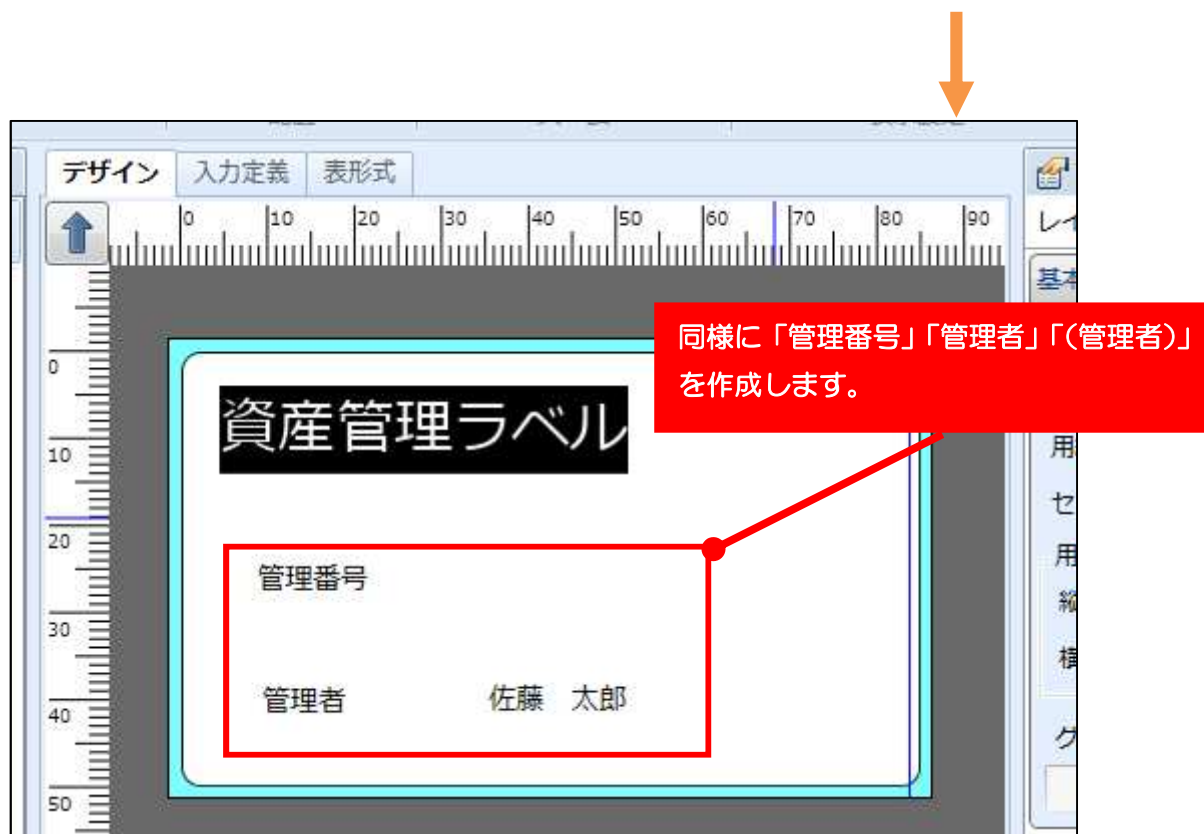
練習として「資産管理ラベル」を作成します。



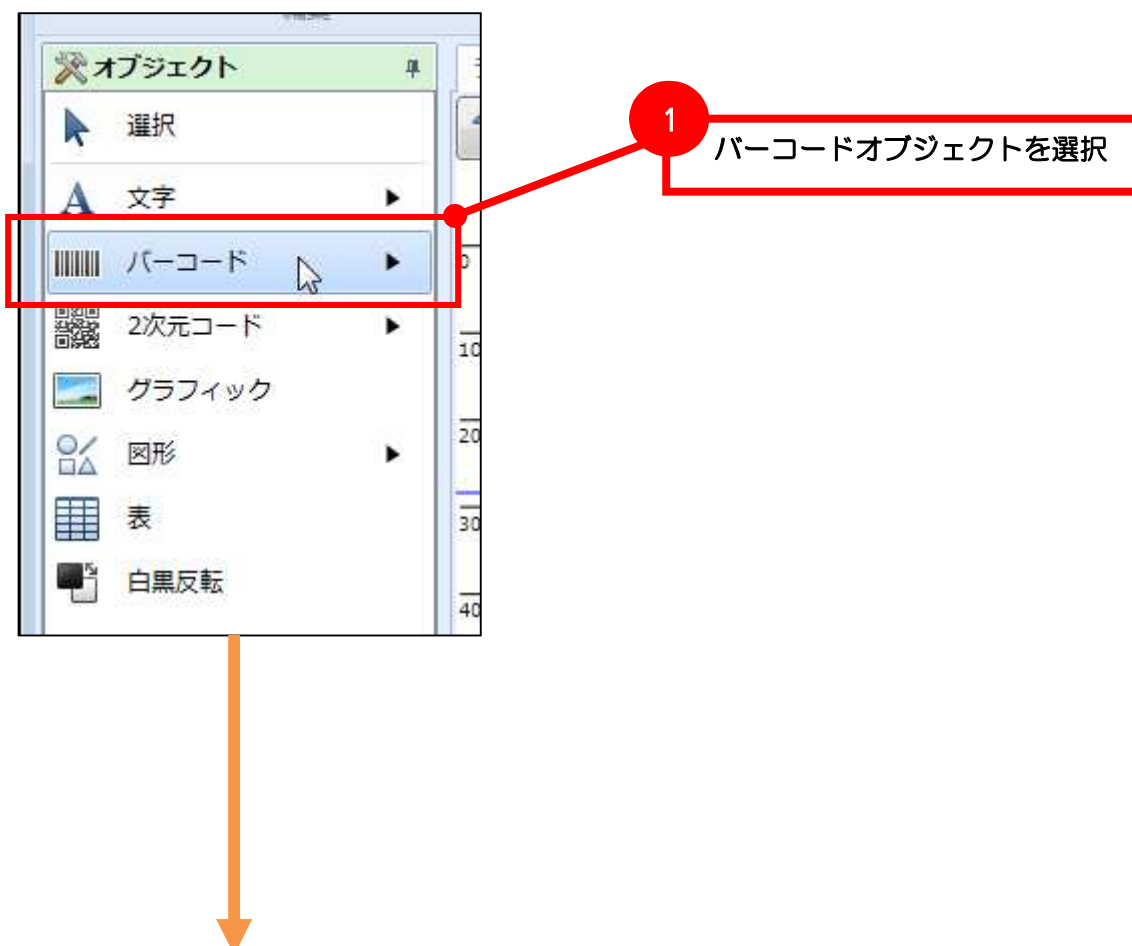
■文字オブジェクトの作成

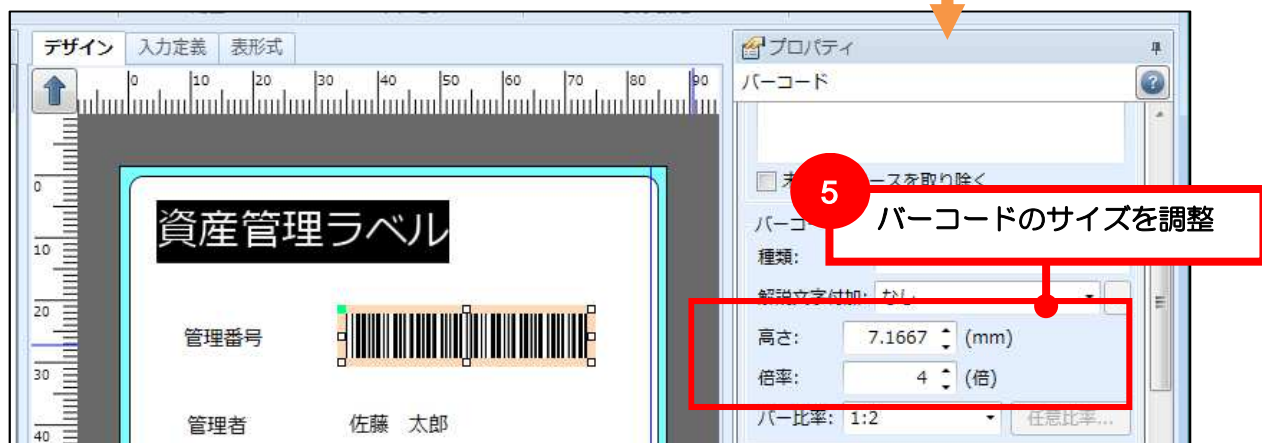
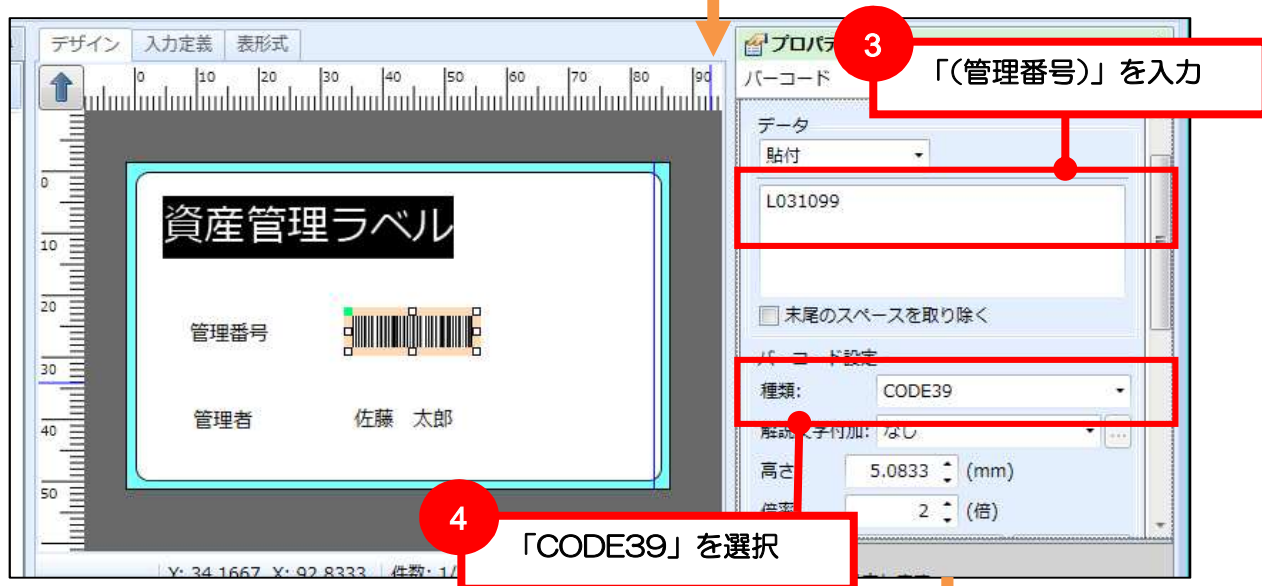
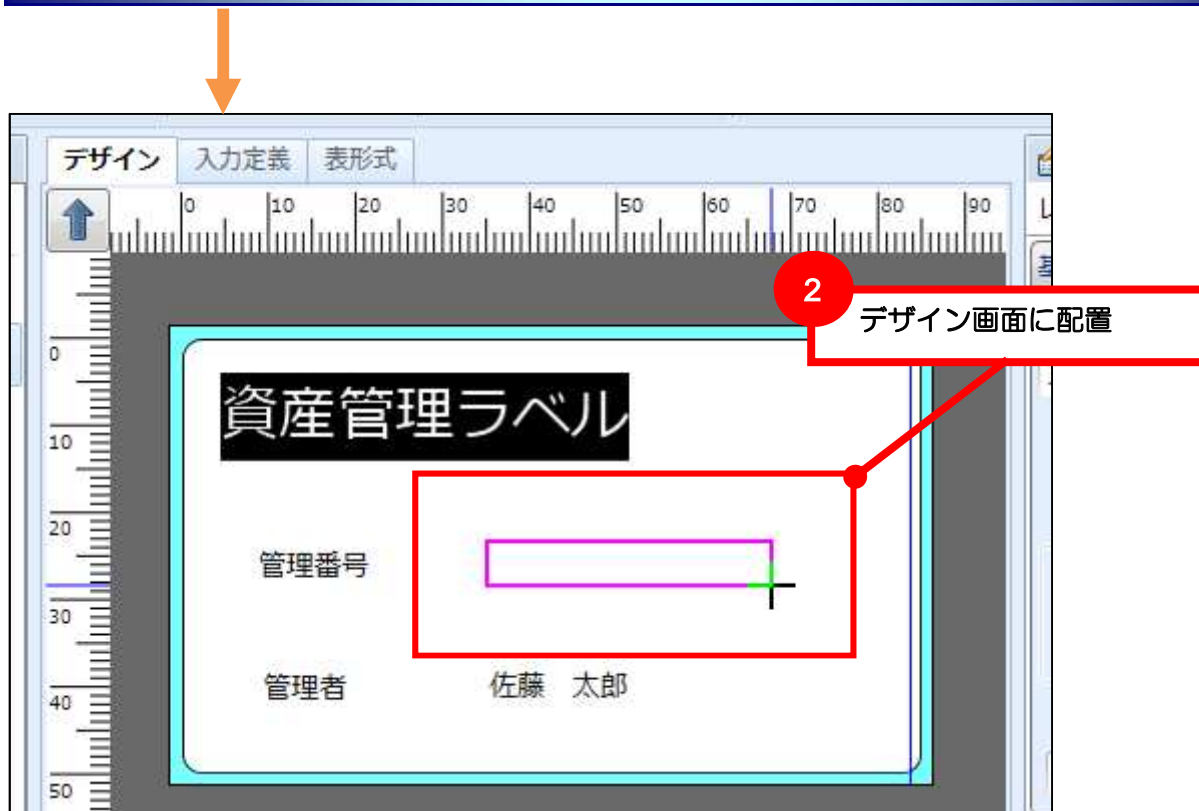




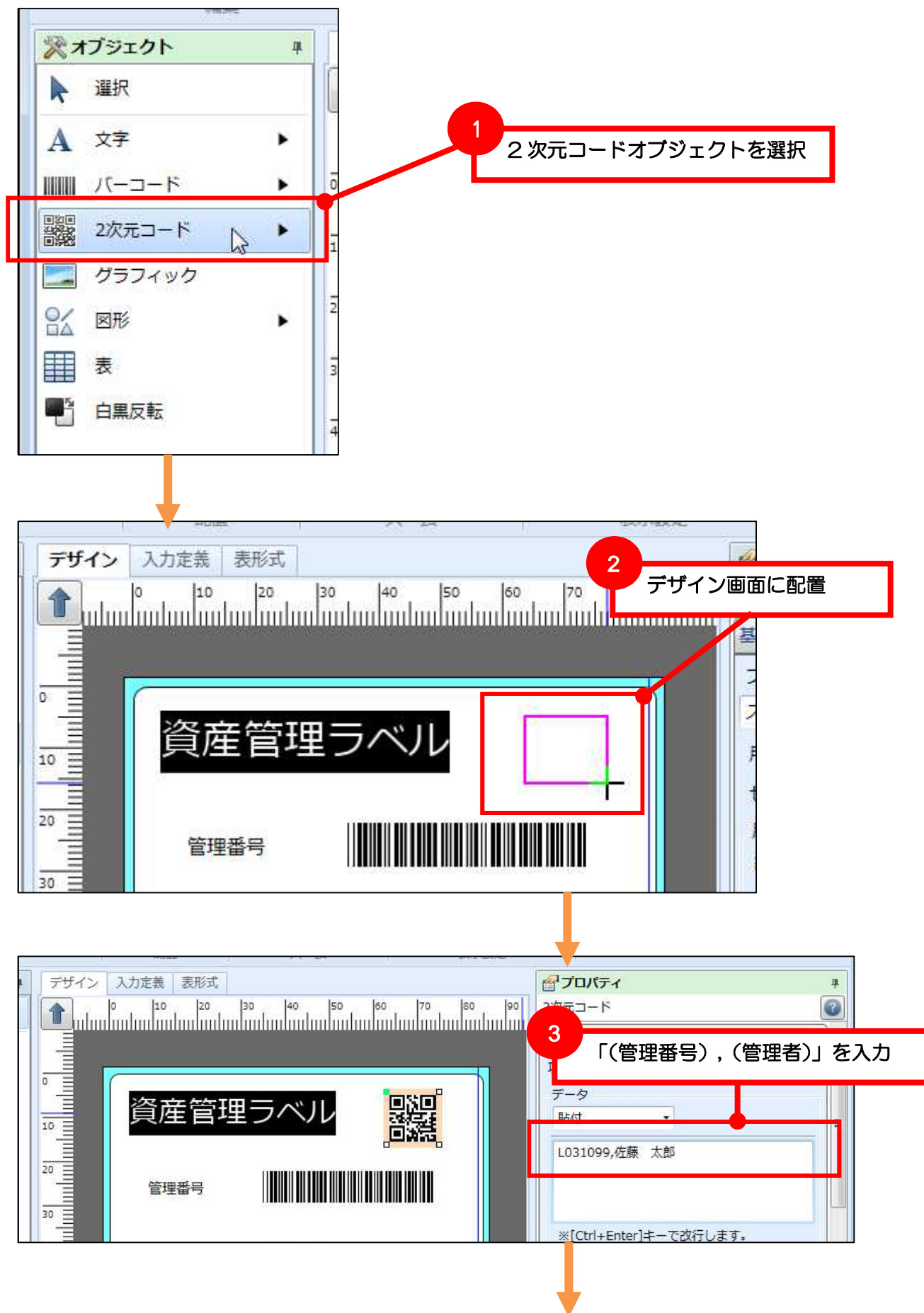


■バーコードオブジェクトの作成



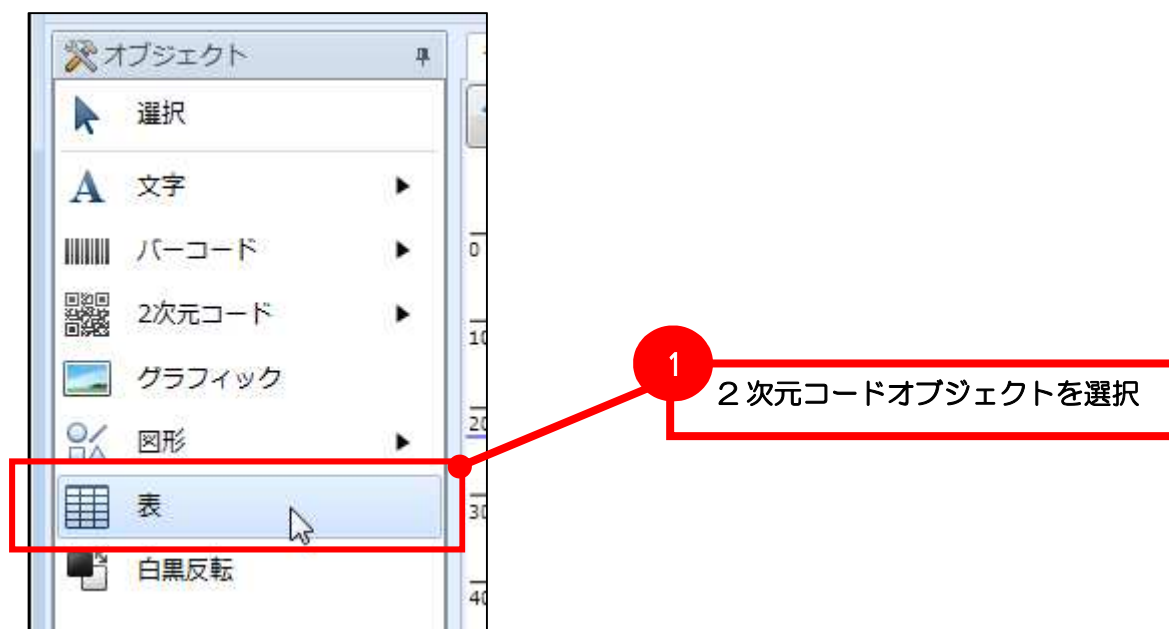


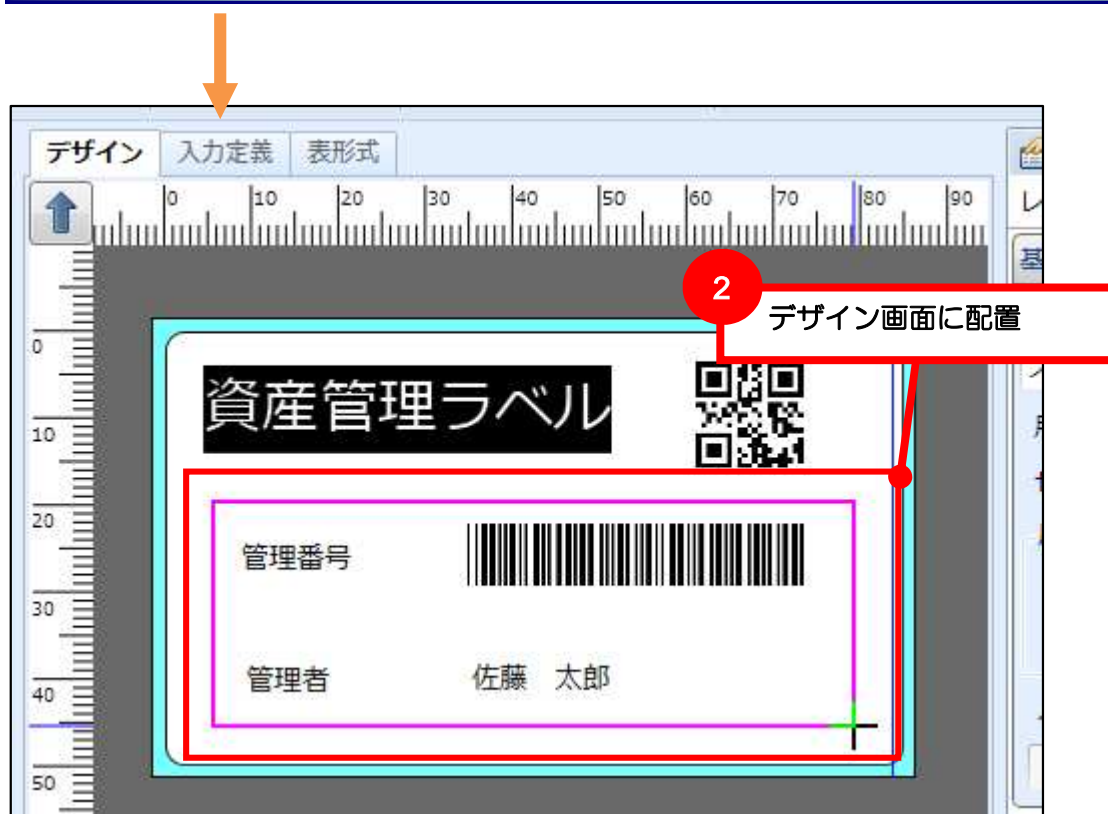
■2次元コードオブジェクトの作成







■表オブジェクトの作成







[ワンポイント]表サイズ

表の全体的なサイズは枠線の四角をドラッグして変更できます。縦線・横線の位置は線上でポインタが「」や「」の状態でドラッグすると変更できます。

以上、デザイン作成は完了です。

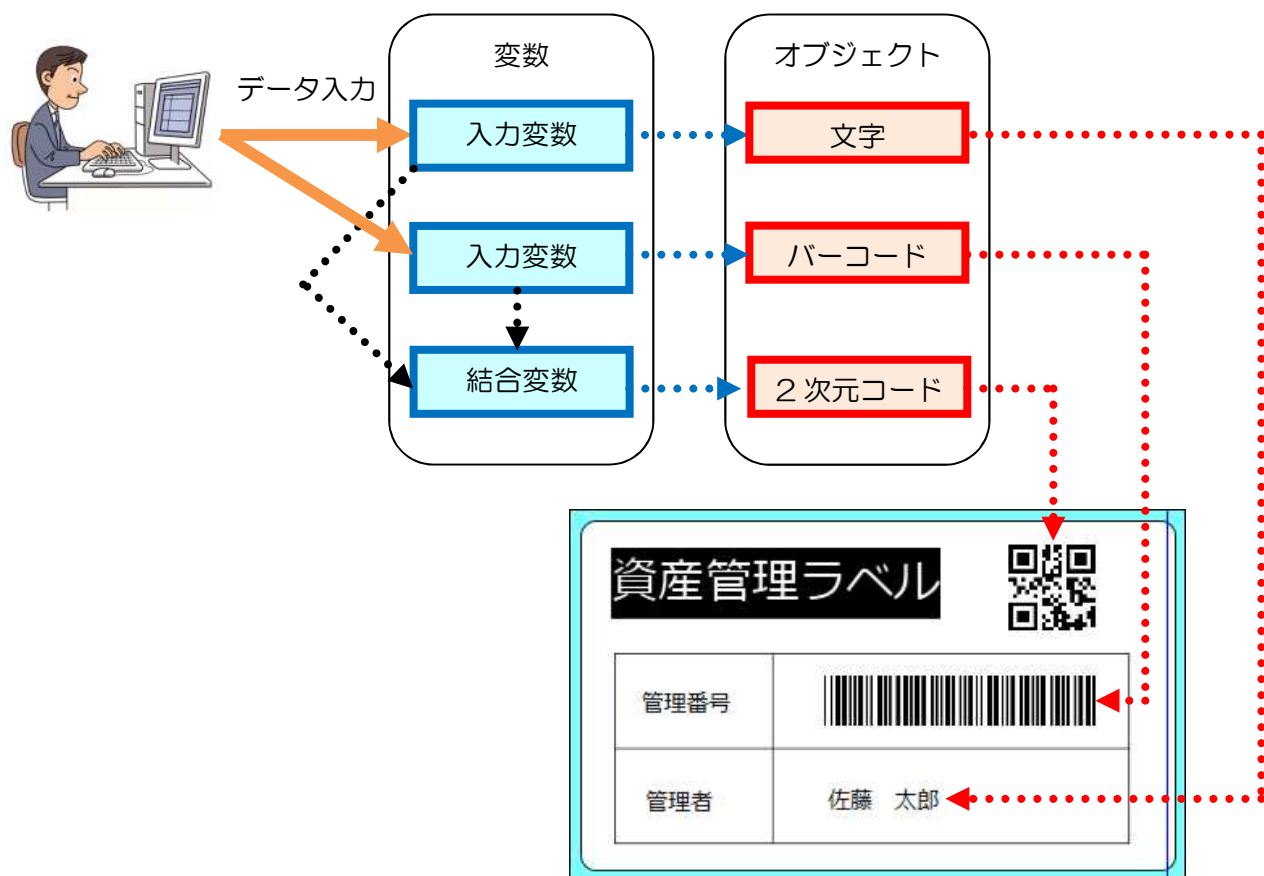


3

変数でデータを入力する

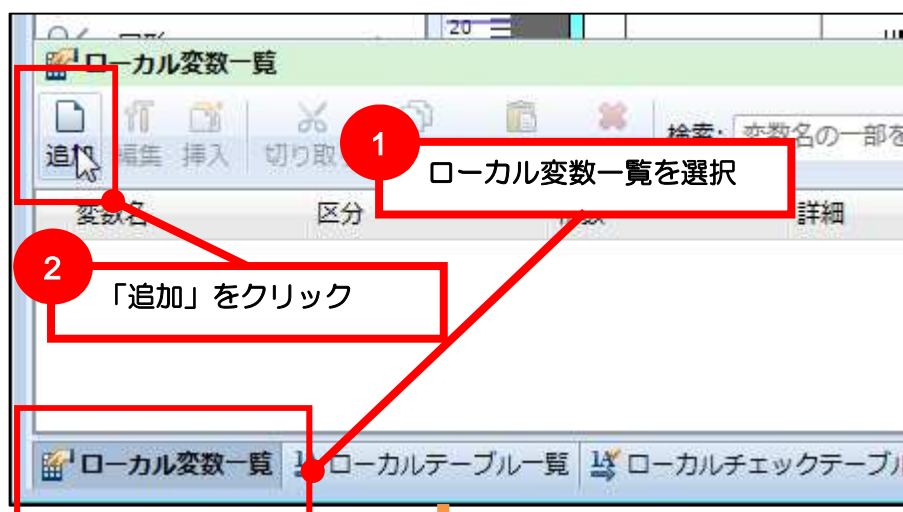
■変数の作成 ■オブジェクトの紐付け ■入力順序

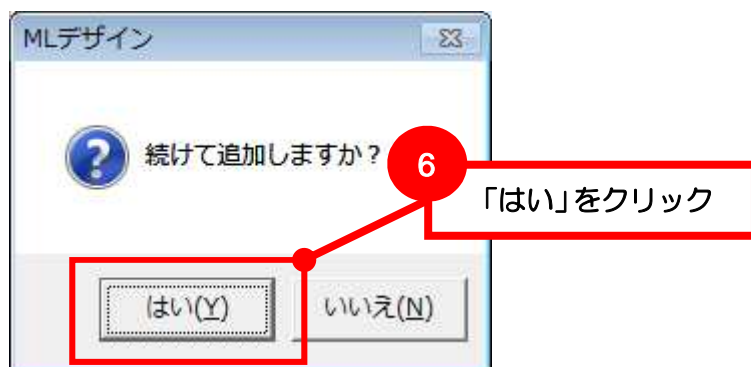
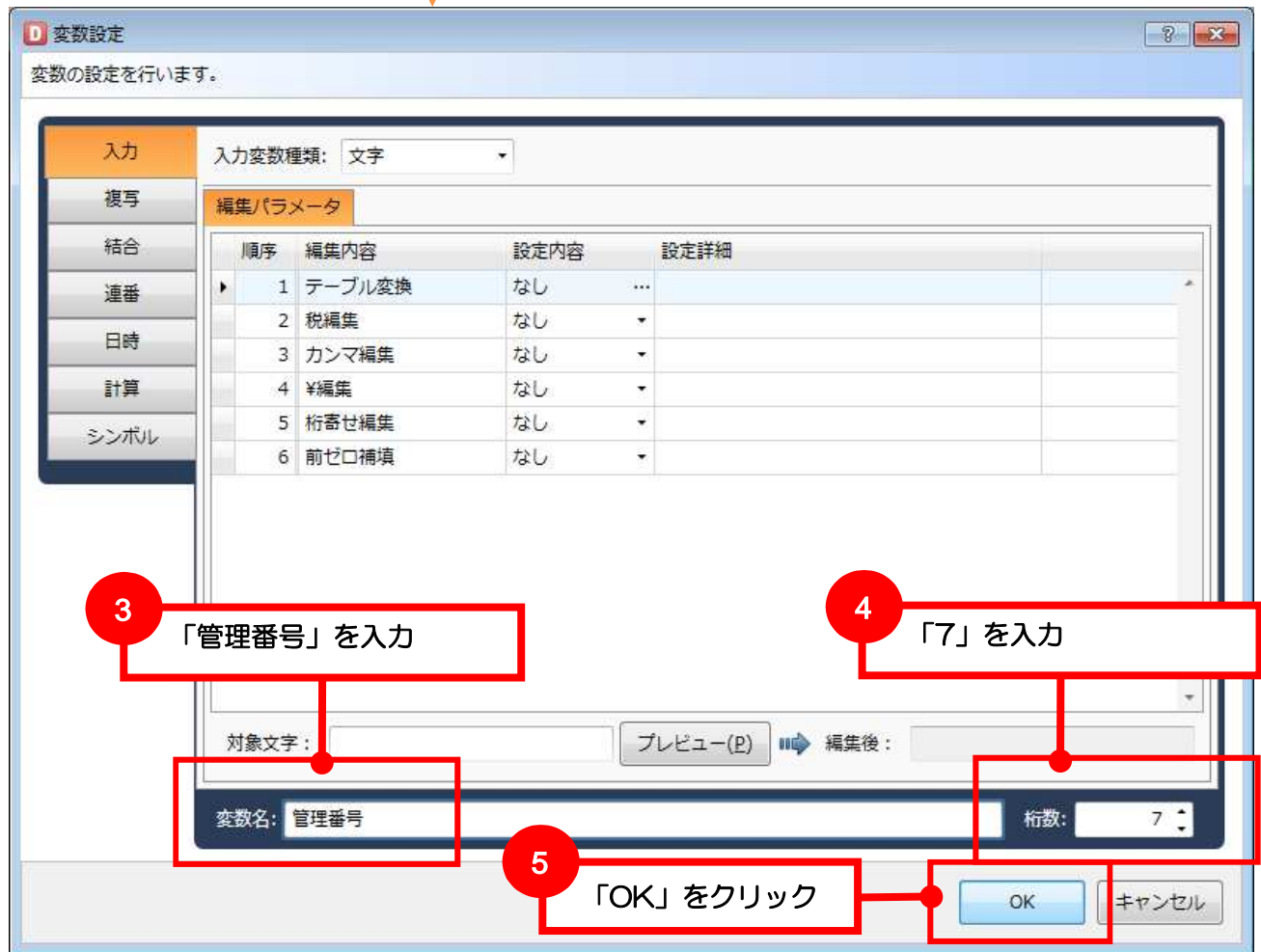
発行時に入力したデータをラベルに印字するために、データの箱となる「変数」を作成して、オブジェクトと紐付けを行います。

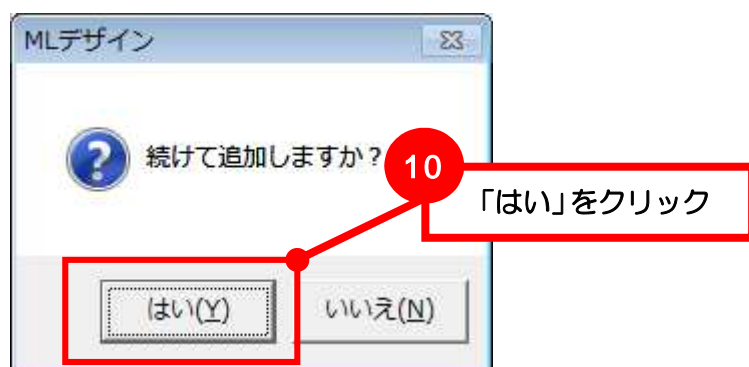
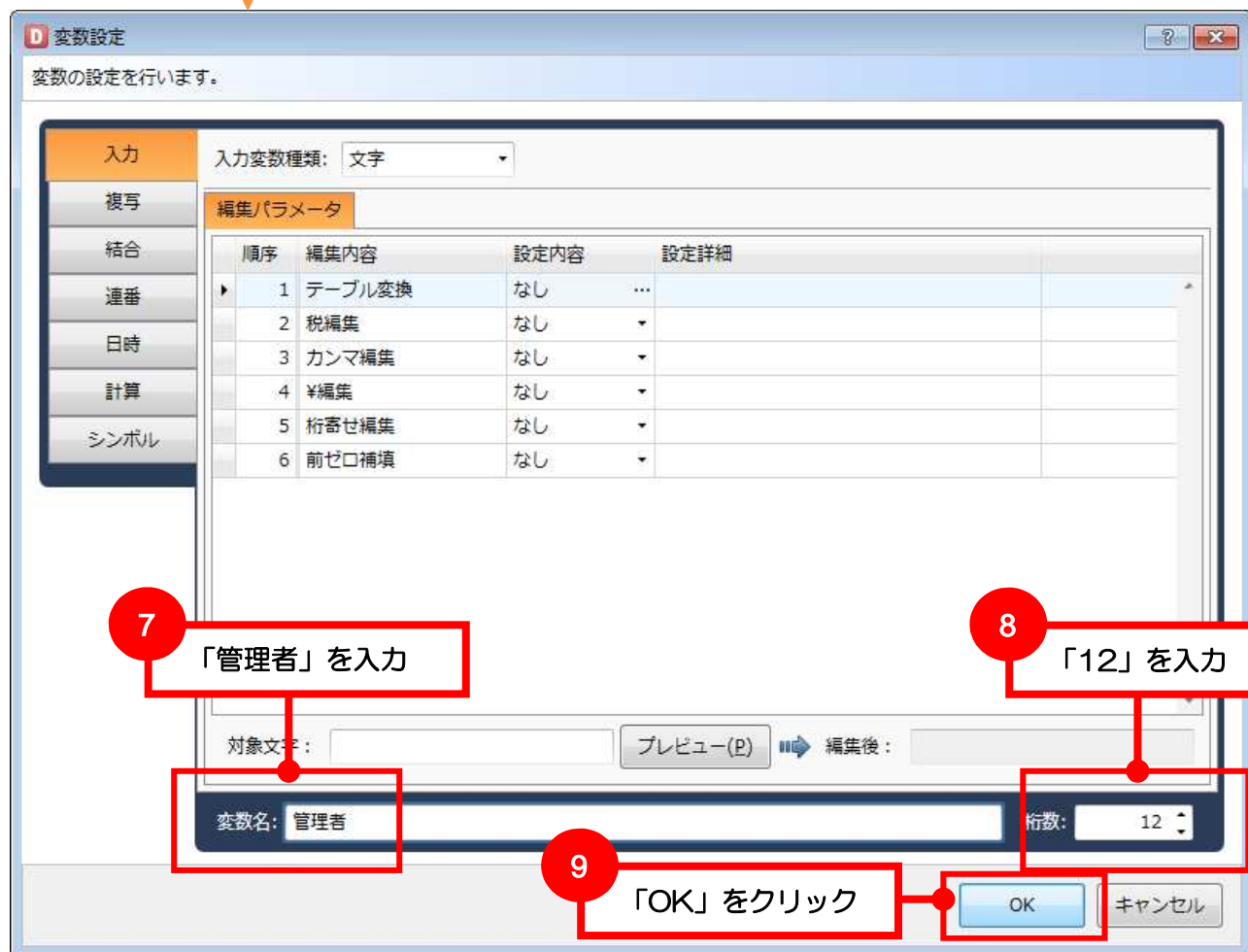


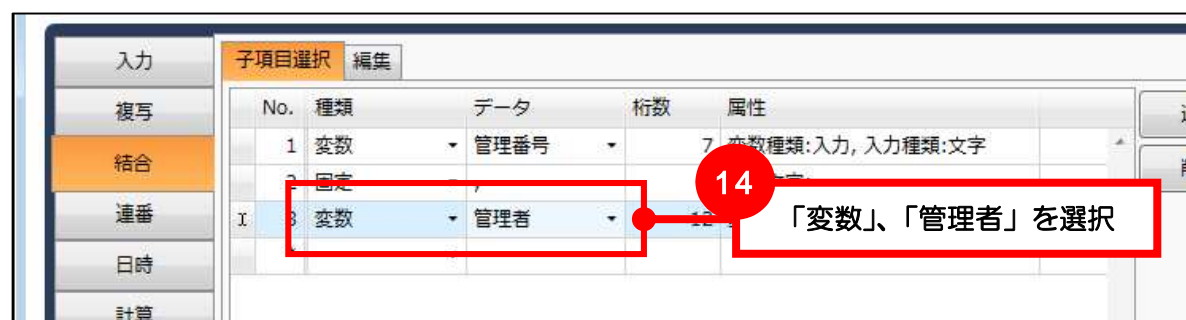
■変数の作成

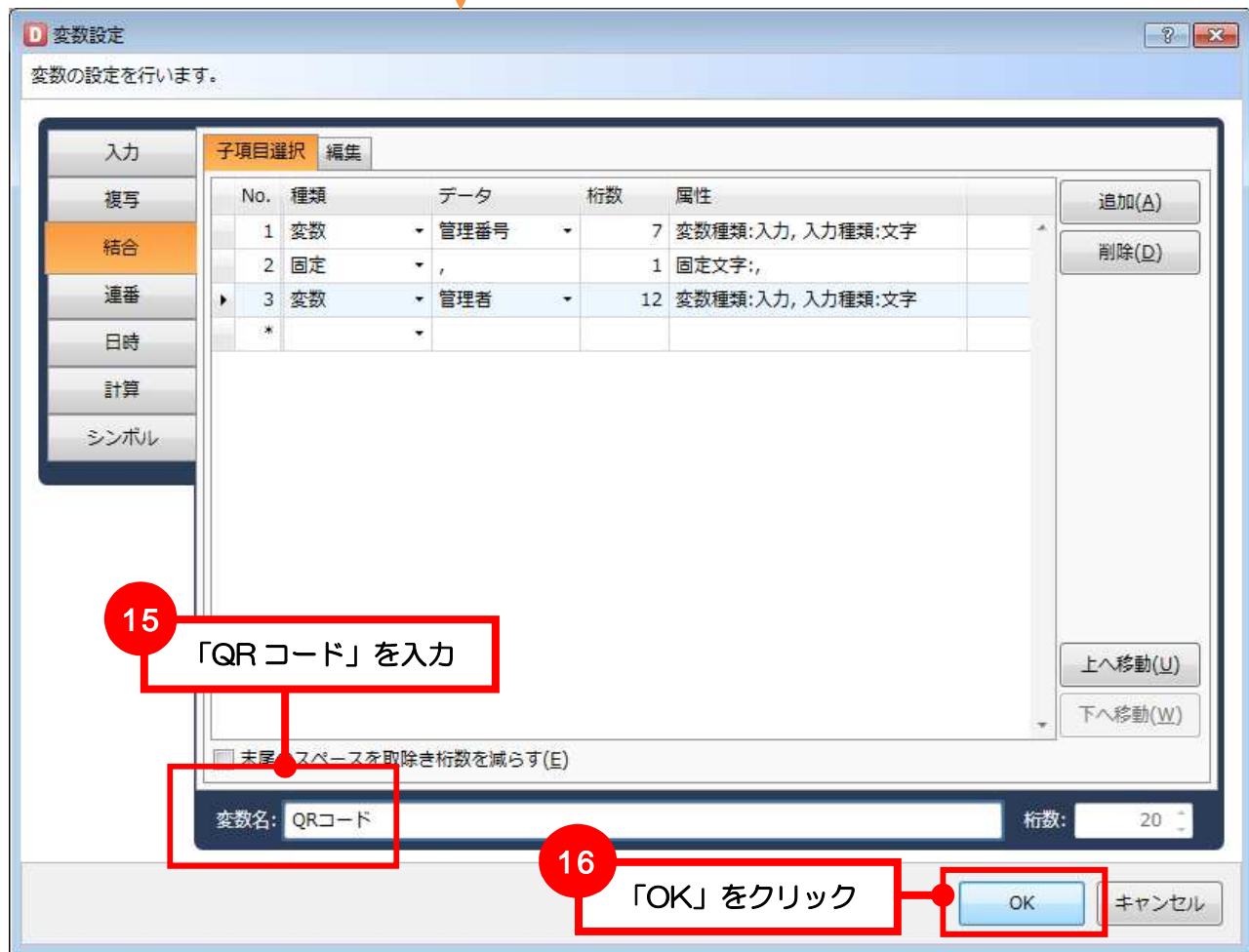
データを入力する「入力変数」と、変数や固定値を結合する「結合変数」を作成します。





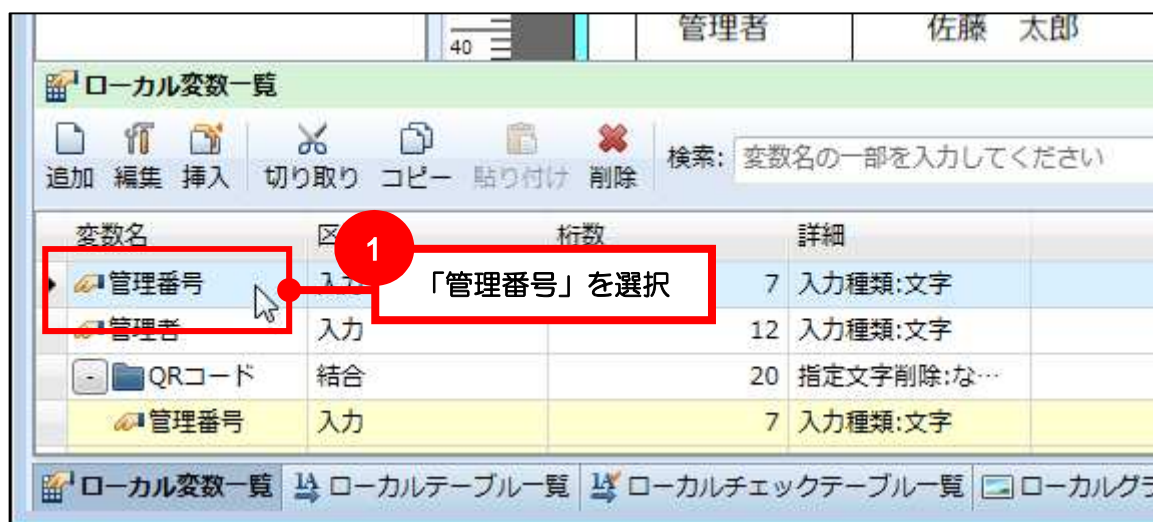


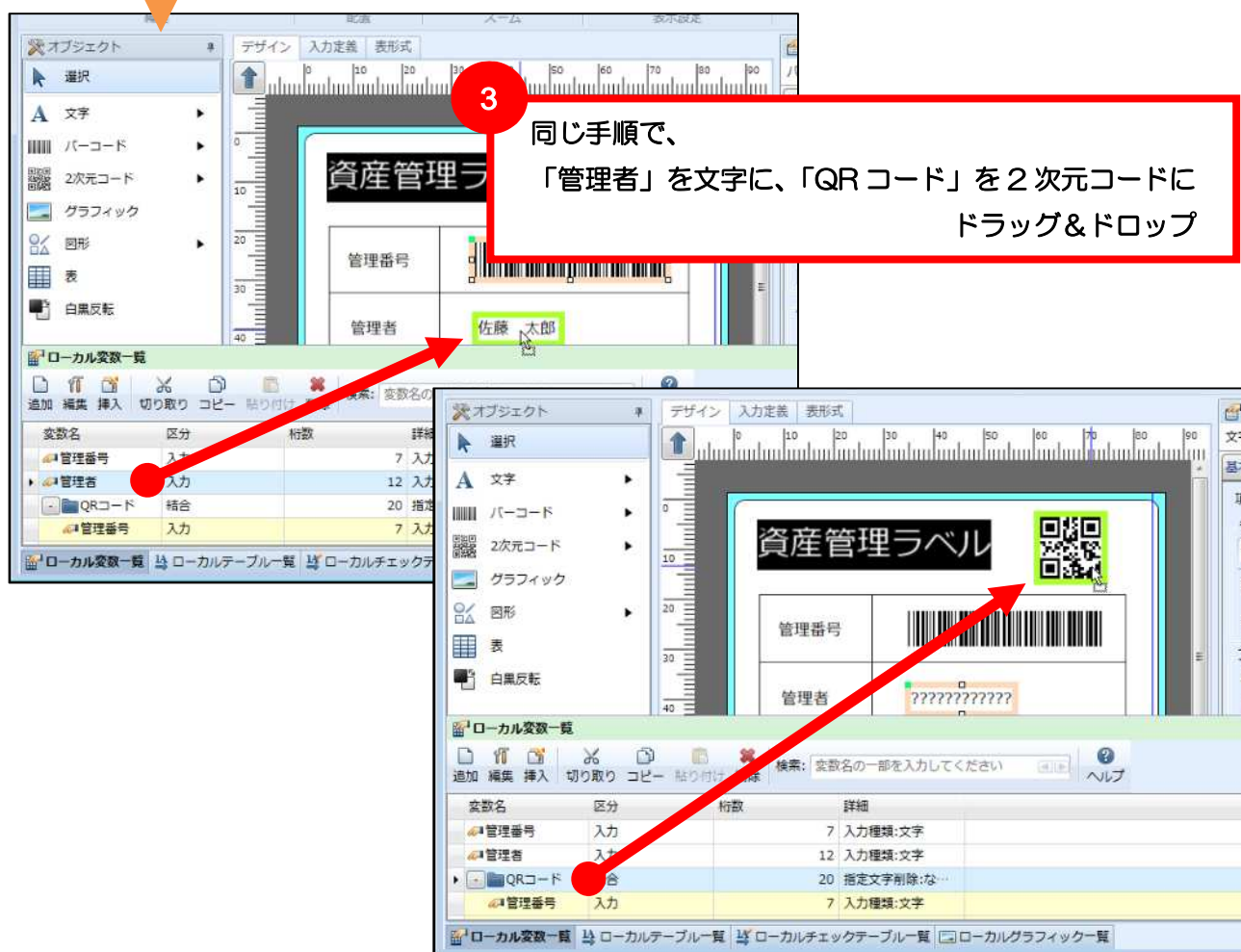
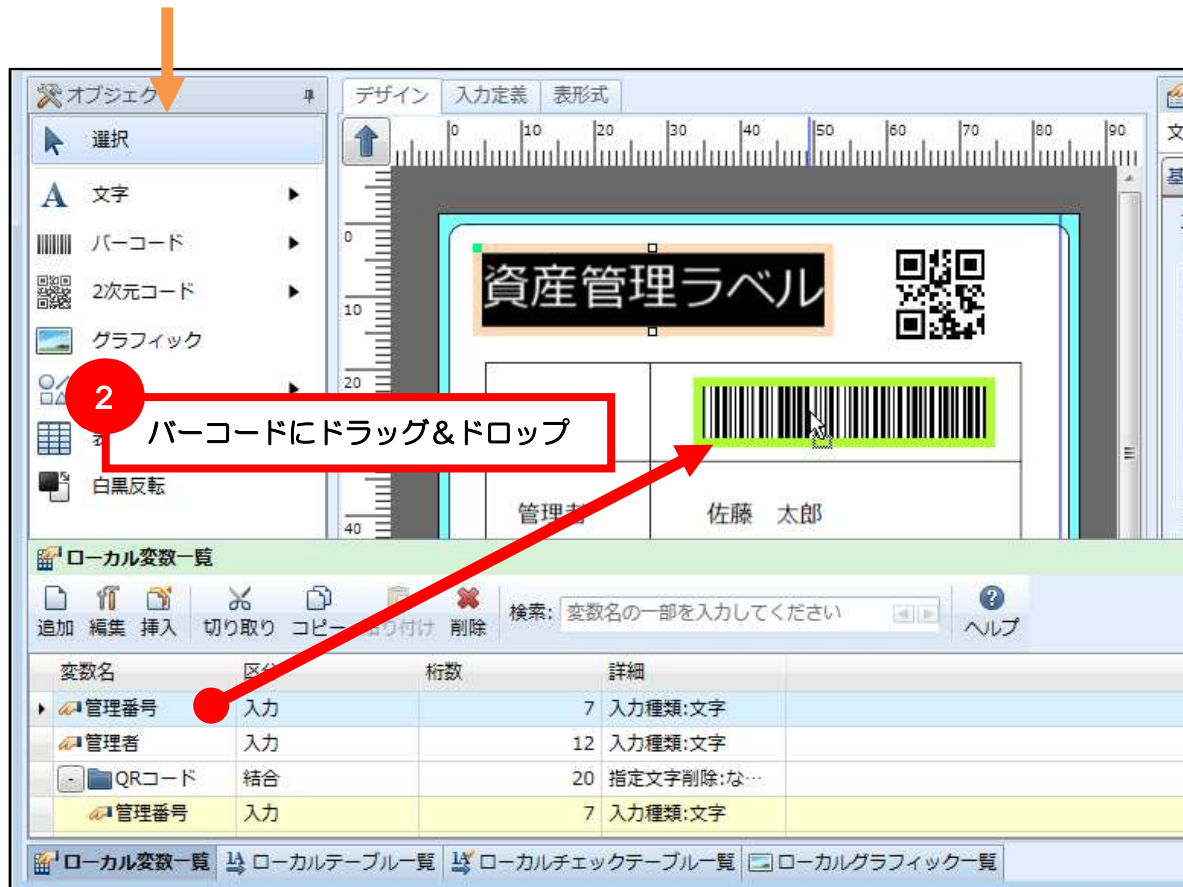




■変数とオブジェクトの紐付け

変数のデータをオブジェクトに反映させるために紐付けを行います。





以上、変数の作成からオブジェクトとの紐付けは完了です。



■入力順序の確認

アプリケーションからのデータ入力は、入力変数の入力順位に従って行います。

入力順位は「第2章 アプリケーション開発」で利用するため、メモするなどして後で確認できるようにしてください。

アプリケーションの記述例（VB2010の場合）

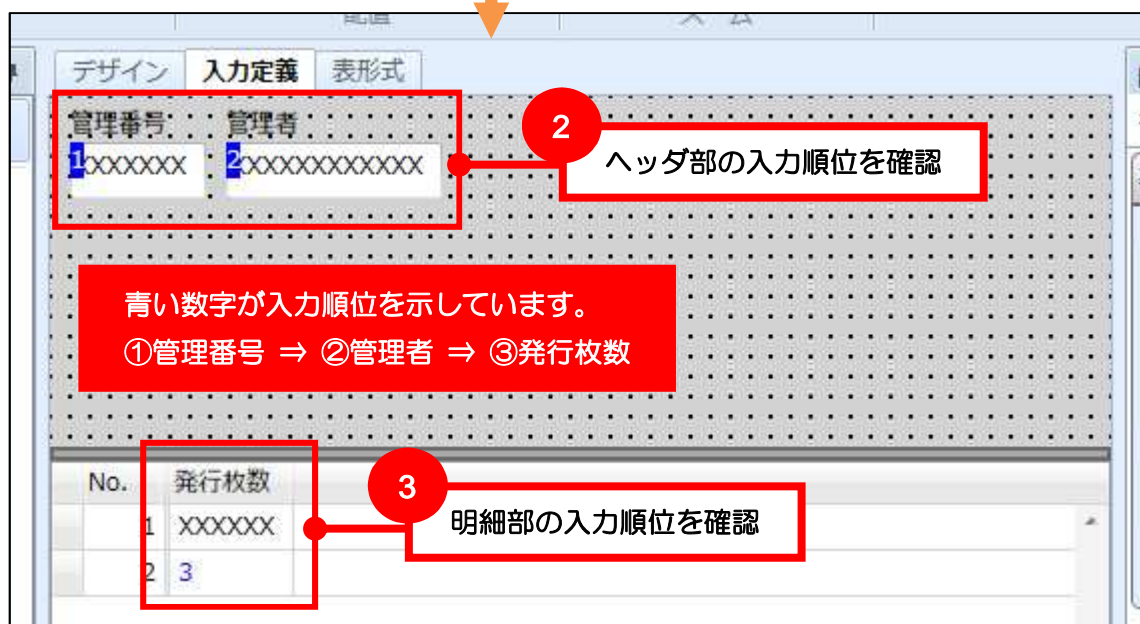
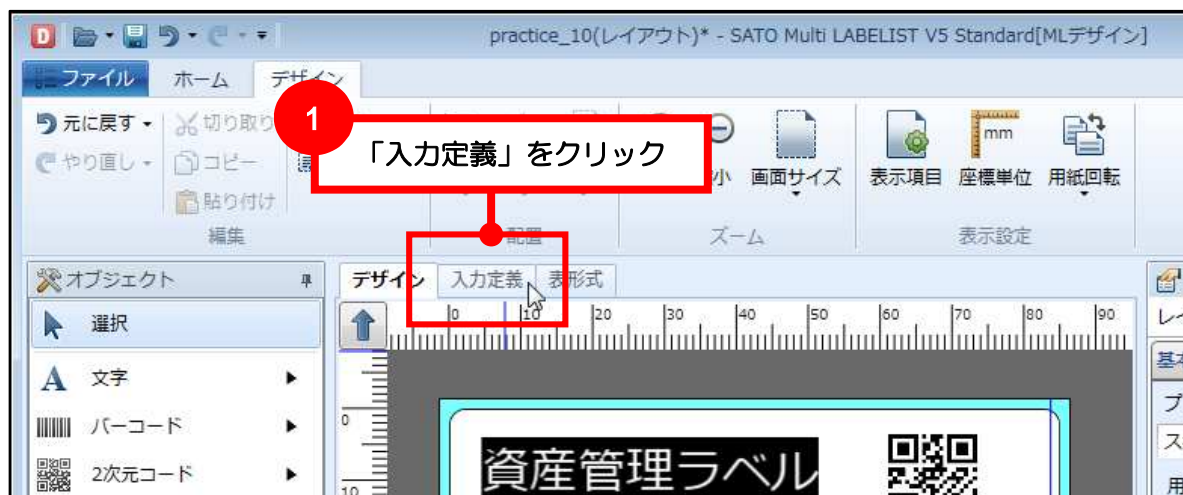
```
MLComponent.LayoutFile = "C:¥sato¥label.mllayx"
```

```
MLComponent.PrnData = "L123431" & vbTab & "佐藤 花子" & vbTab & "1"
```

変数の入力順

1. 管理番号
2. 管理者
3. 発行枚数





第2章

アプリケーション 開発

1

MLComponent を利用する

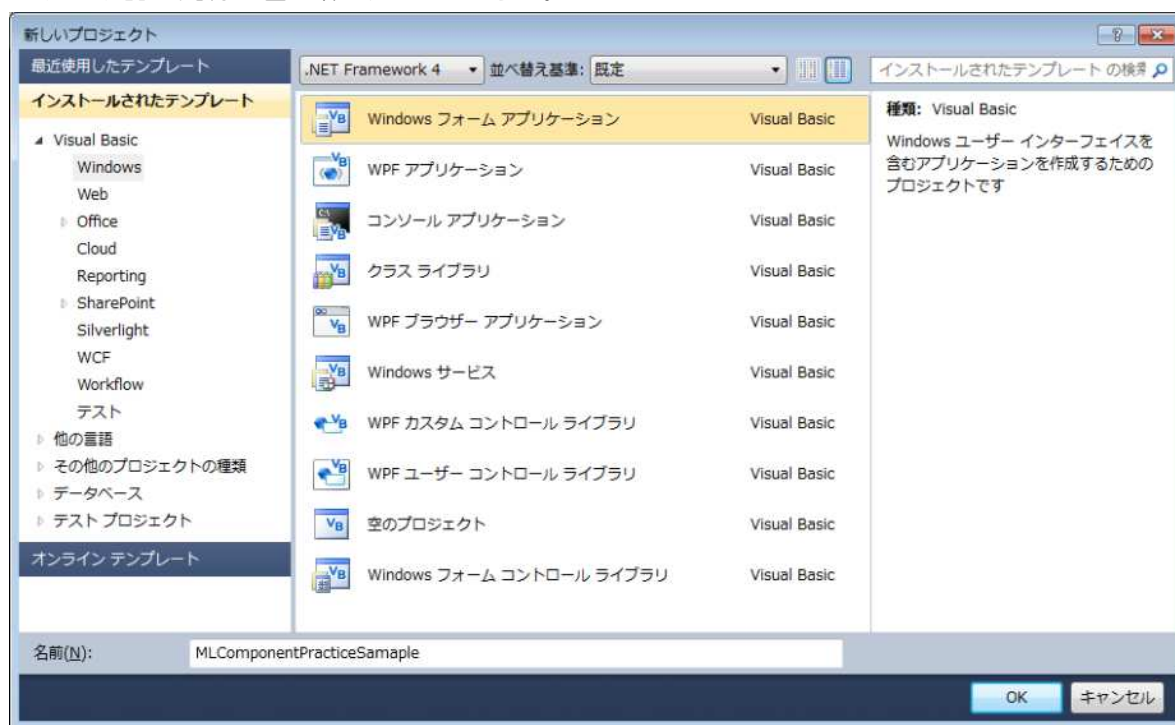
■変数の作成 ■オブジェクトの紐付け ■入力順序

Visual Studio 2010 で MLComponent を利用した Windows アプリケーションを開発します。

Visual Studio 2010 の使い方に関しては説明を省略しております。Visual Studio 2010 の基本的な使い方はオンラインヘルプや参考書などをご参照ください。

■発行画面の作成

はじめに、Visual Studio 2010 の Visual Basic で Windows フォームアプリケーションのプロジェクトで下記の発行画面を作成してください。



管理番号

管理者

発行枚数

TextBox で Name を「txtNumber」で作成

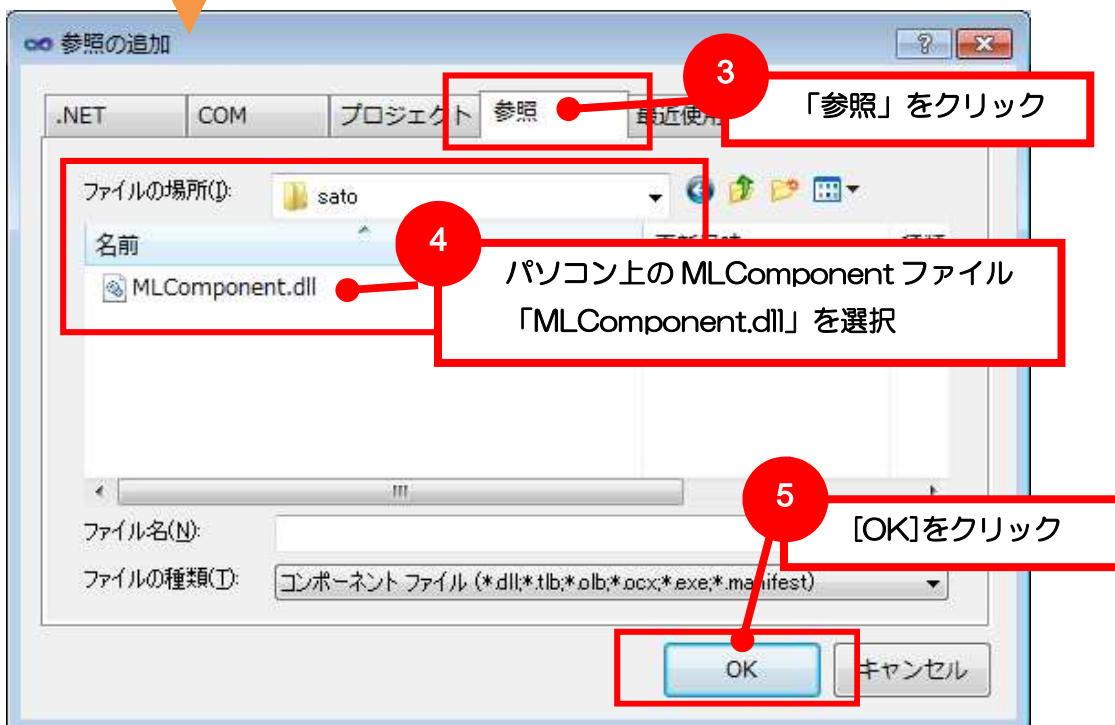
TextBox で Name を「txtName」で作成

TextBox で Name を「txtQty」で作成

Button で Name を「btnPrintOut」で作成

■MLComponent の参照設定

MLComponent を利用するには、プロジェクトに参照設定を追加します。



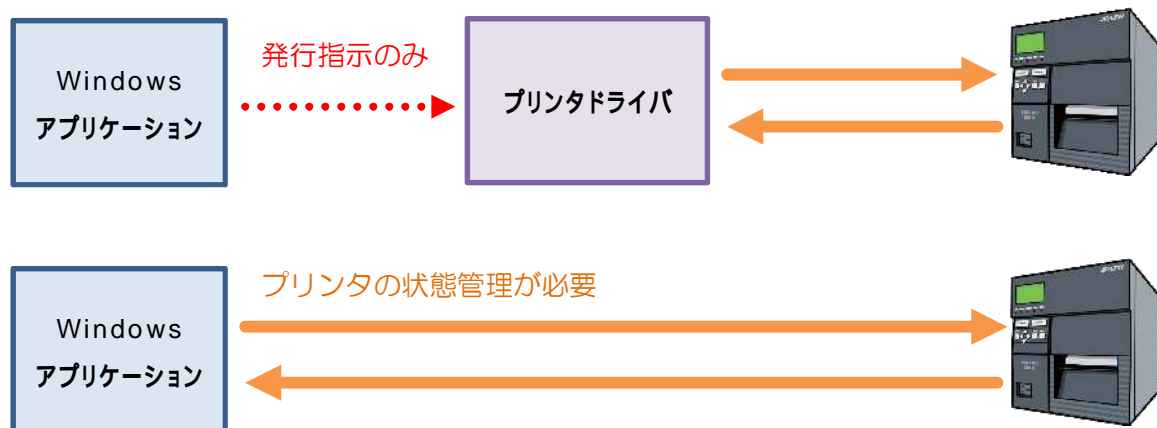
2

発行方法を決める

■発行方法

発行処理を作成する前に、アプリケーションからどのようにプリンタを操作するのか、発行方法を選択してください。

ラベル発行は、パソコンとプリンタをケーブルで接続し、プリンタに発行データを送信して行います。発行方法には、大きく分けて「プリンタドライバ出力」と「インターフェース出力」の2つの方法があります。



■発行方法

	プリンタドライバ出力	インターフェース出力
利用可能な インターフェース	USB LAN（無線 LAN） RS-232C IEEE1284	USB LAN（無線 LAN） RS-232C
開発レベル	初級	中級～上級
プリンタの制御	× （プリンタドライバに全て依存）	○
メリット	通信制御をプリンタドライバに任せるため、アプリケーションの開発が簡単。	プリンタの状態を確認できる。 プリンタドライバのインストール作業が いらない。
デメリット	発行する PC 毎に、プリンタドライバを インストールする必要がある。 プリンタの状態が確認できない。プリン タでエラーが発生していることをアプリ ケーションで通知できない、	プリンタとの通信制御をするために、通 信に関する技術やプリンタの仕様を理解し た上で、プリンタのエラーや通信障害な ど様々な状況に応じた処理を作り込む必 要がある。

3

ラベルを発行する

■発行処理

ボタン押下時のイベントに発行処理を作成します。

■発行処理（プリンタドライバ出力）

```
Private Sub btnPrintOut_Click(sender As System.Object, e As System.EventArgs) Handles btnPrintOut.Click

    'MLComponentの宣言
    Dim MLComponent As SATO.MLComponent.MLComponent = New SATO.MLComponent.MLComponent

    'レイアウトファイルの設定(第1章で作成したレイアウトファイルを指定します。)
    MLComponent.LayoutFile = "C:¥sato¥label.mllayx"

    'データの設定(データをタブ区切りで指定します。)
    MLComponent.PrnData = _
        txtNumber.Text & vbTab & txtName.Text & vbTab & txtQty.Text

    '出力先の設定(プリンタドライバ名を指定します。)
    MLComponent.Setting = "DRV:SATO SG412R"

    '接続
    Dim Result As String
    Result = MLComponent.OpenPort(1)
    If Result <> 0 Then
        MessageBox.Show("接続エラー No." & Result)
        Exit Sub
    End If

    '発行
    Result = MLComponent.Output
    If Result <> 0 Then
        MessageBox.Show("発行エラー No." & Result)
        Exit Sub
    End If

    '切断
    Result = MLComponent.ClosePort
    If Result <> 0 Then
        MessageBox.Show("切断エラー No." & Result)
        Exit Sub
    End If

    '完了メッセージ表示
    MessageBox.Show("発行完了")

End Sub
```


■発行処理（インターフェース出力）

```
Private Sub btnPrintOut_Click(sender As System.Object, e As System.EventArgs) Handles btnPrintOut.Click

    'MLComponentの宣言
    Dim MLComponent As SATO.MLComponent.MLComponent = New SATO.MLComponent.MLComponent

    'レイアウトファイルの設定(第1章で作成したレイアウトファイルを指定します。 )
    MLComponent.LayoutFile = "C:¥sato¥label.mllayx"

    'データの設定(データをタブ区切りで指定します。 )
    MLComponent.PrnData = _
        txtNumber.Text & vbTab & txtName.Text & vbTab & txtQty.Text

    '出力先の設定(例としてLAN接続の指定します。 )
    MLComponent.Setting = "LAN:192.168.1.1"

    '接続
    Dim Result As String
    Result = MLComponent.OpenPort(1)
    If Result <> 0 Then
        MessageBox.Show("接続エラー No." & Result)
        Exit Sub
    End If

    'プリンタ状態確認
    Dim Status As String = ""
    Result = MLComponent.GetStatus(Status)
    If Result <> 0 Then
        MessageBox.Show("プリンタ状態確認[発行前]エラー No." & Result)
        Exit Sub
    End If

    'プリンタ状態判定(送信可能な状態のステータスが判定します。 )
    Dim SendStatus As String() = New String() { "0", "1", "A", "B", "G", "H", "M", "N", "S", "T" }
    If 0 > Array.IndexOf(SendStatus, Status.Substring(2, 1)) Then
        MessageBox.Show("プリンタの準備ができていません。 Status = " & Status)
        Exit Sub
    End If

    '発行
    Result = MLComponent.Output
    If Result <> 0 Then
        MessageBox.Show("発行エラー No." & Result)
        Exit Sub
    End If

    'プリンタ状態確認
    Dim Status As String = ""
    Result = MLComponent.GetStatus(Status)
    If Result <> 0 Then
        MessageBox.Show("プリンタ状態確認[発行後]エラー No." & Result)
        Exit Sub
    End If

    'プリンタ状態判定(エラーステータスが判定します。 )
    Dim SendStatus As String() = New String() { "a", "b", "c", "d", "e", "f", "g", "h", "i", "j", "k", "l", "m", "o" }
    If 0 <= Array.IndexOf(SendStatus, Status.Substring(2, 1)) Then
```



```

    MessageBox.Show("プリンタでエラーが発生しています。Status = " & Status)
End If

'切断
Result = MLComponent.ClosePort
If Result <> 0 Then
    MessageBox.Show("切断エラー No." & Result)
Exit Sub
End If

'完了メッセージ表示
MessageBox.Show("発行完了")

End Sub

```

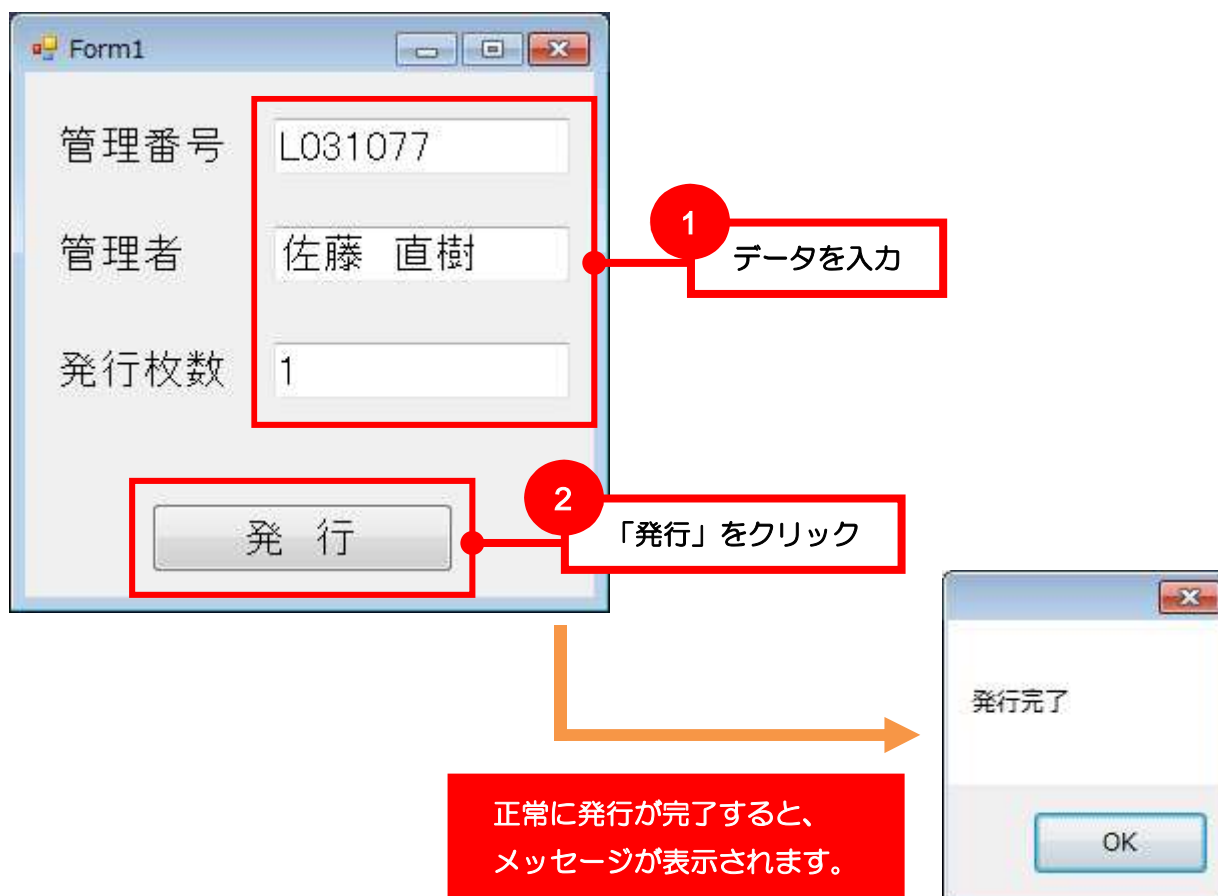
重要ポイント

インターフェース出力では必ずプリンタの状態を確認した上で、発行指示を行ってください。プリンタの準備が整う前に発行指示を行なうと、ラベルが実際に発行されないトラブルが発生する可能性があります。

取得したプリンタのステータスは、リファレンスマニュアルの「付録：ステータス一覧」で「送信の可否」を参考に判定してください。

■ラベル発行

発行処理を作成したら、アプリケーションをビルドして実行します。



4

アプリケーションを配布する

■インストーラ

最後に、セットアップ プロジェクトでインストーラを作成して配布する方法を説明します。

インストーラを利用しない場合は、「MLComponent.dll」をアプリケーションの実行ファイル (*.exe) と同じ場所にコピーしてご利用ください。

■インストーラ

