

---

# Multi LABELIST Component

## 練習マニュアル

株式会社サトー

2022年11月25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 V6」(以下 MLV6)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。

MLV6 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 MLV6 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 MLV6 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 2022 Professional (Version 17.3.6) Multi LABELIST V6 (Ver.6.0.0.0) Multi LABELIST Component (Ver.6.0.0.0)
OS	Windows 10 Enterprise 20H2

## 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 レイアウト作成 .....</b>	<b>4</b>
1.MLV6 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 .....	6
2.オブジェクトでデザインを作成する.....	8
■基本操作 .....	8
■文字オブジェクトの作成 .....	9
■バーコードオブジェクトの作成 .....	11
■2 次元コードオブジェクトの作成.....	12
■表オブジェクトの作成 .....	13
3.変数でデータを入力する .....	16
■変数の作成 .....	16
■変数とオブジェクトの紐付け .....	20
■入力順序の確認 .....	22
<b>第 2 章 アプリケーション開発 .....</b>	<b>24</b>
1.MLComponent を利用する .....	25
■プロジェクトの作成.....	25
■発行画面の作成 .....	26
■MLComponent の参照設定 .....	27
2.発行方法を決める.....	29
■発行方法 .....	29
3.ラベルを発行する.....	30
■発行処理（プリンタドライバ出力） .....	30
■発行処理（インターフェース出力） .....	31
■ラベル発行 .....	32
4.アプリケーションを配布する.....	33
■Visual Studio 2022 でインストーラを作成する.....	33

# 第1章

# レイアウト作成

## 1

## MLV6 でレイアウトを作成する

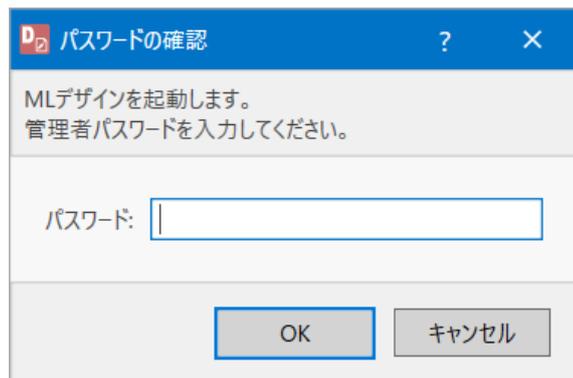
■ML デザイン ■基本設定

MLV6 を使ってレイアウトを作成します。MLComponent は MLV5 で作成したレイアウトファイルも利用可能です。

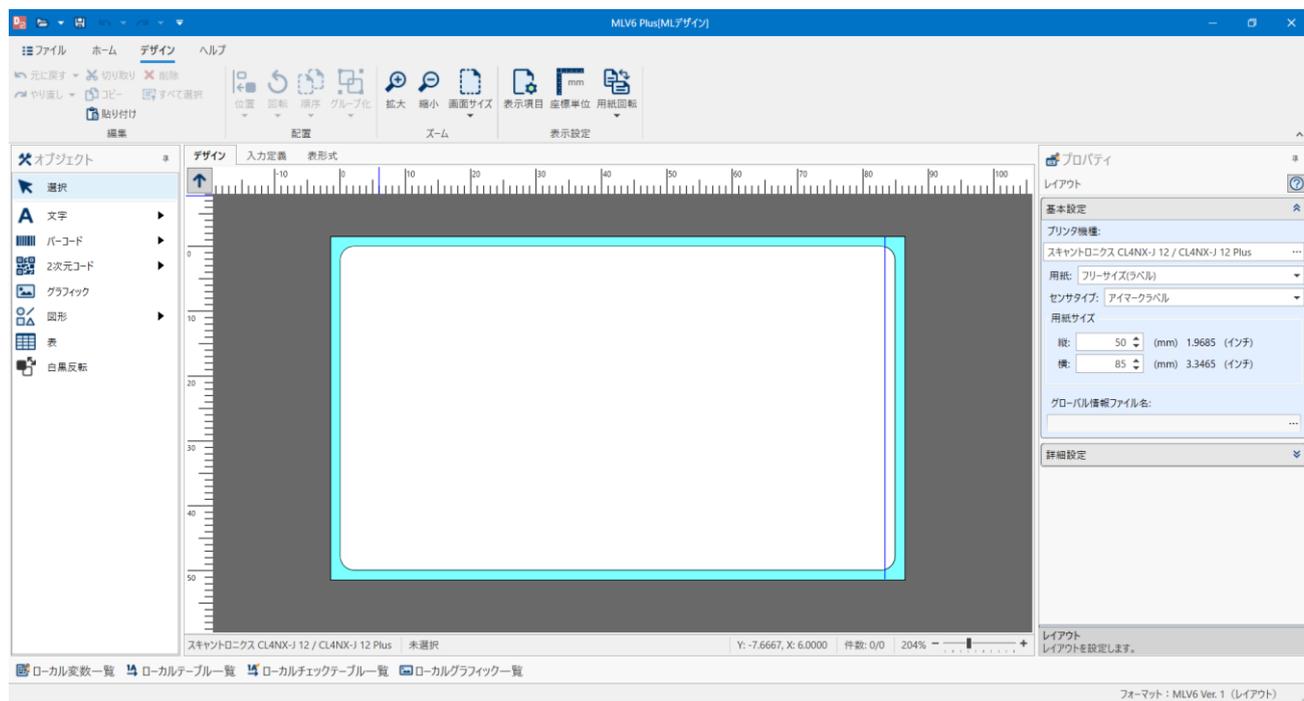
## ■ML デザインの起動

スタートメニュー>「Multi LABELIST V6」>「ML デザイン」を選択します。

パスワードの確認画面では「user」と入力し、「OK」を選択します。



ML デザイン画面が表示されます。

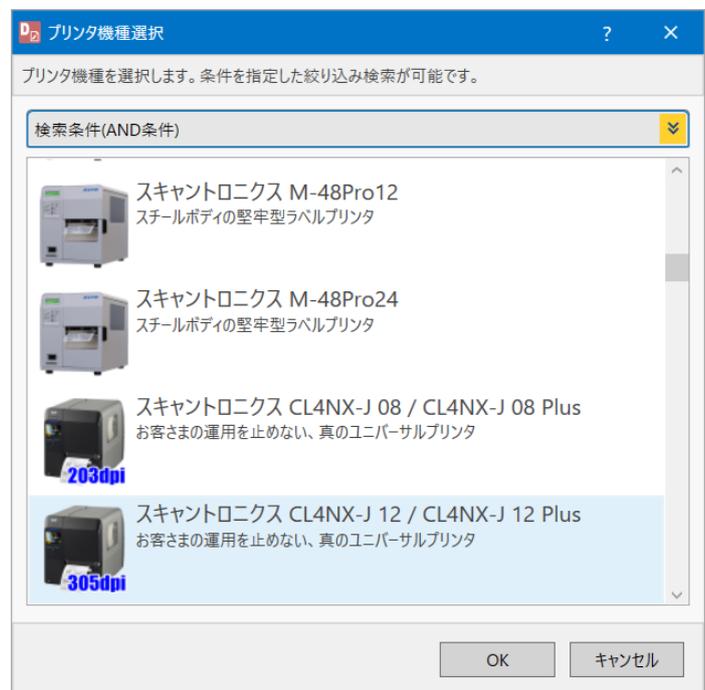
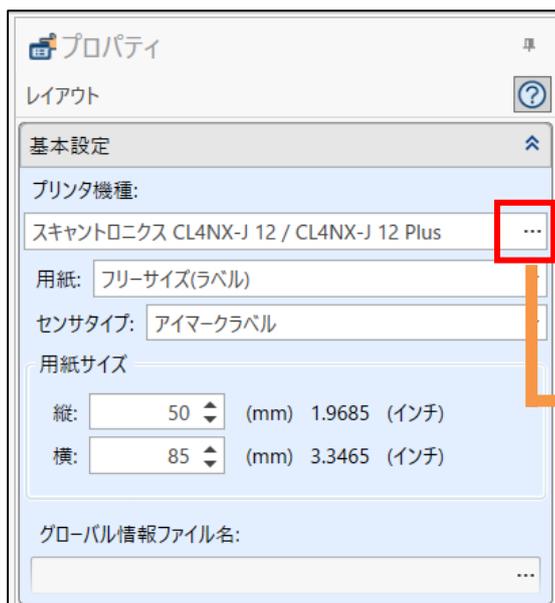
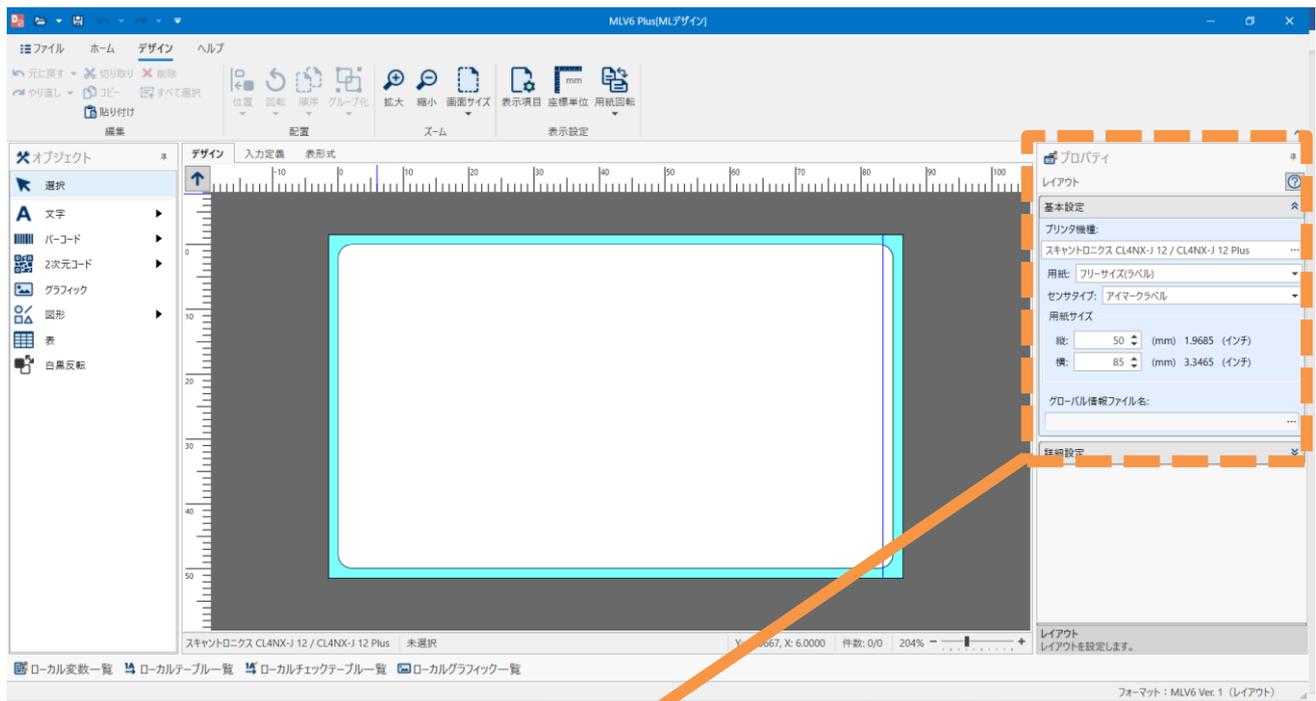


**[注意]** 事前にライセンス認証を行ってください。

ML デザインの利用にはライセンス認証が必要です。スタートメニュー>「Multi LABELIST V6」>「ML メンテナンス」を起動し、ライセンス認証を行ってください。

## ■レイアウトの基本設定

ラベルを発行するプリンタ機種を設定します。ML デザインの「基本設定」>「プリンタ機種」の「…」を選択します。プリンタ機種選択画面が表示されたら、使用するプリンタを選択します。



### 【確認】プリンタ機種

使用するプリンタとレイアウトのプリンタ機種は合わせてください。プリンタ機種名はプリンタ本体の「MODEL」で確認できます。

「センサタイプ」より使用するラベルのセンサタイプを選びます。

#### [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です。ギャップは台紙部分にマークがないタイプです。使用するラベルに合わせてください。

「用紙サイズ」に使用するラベルのサイズを入力します。

#### [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

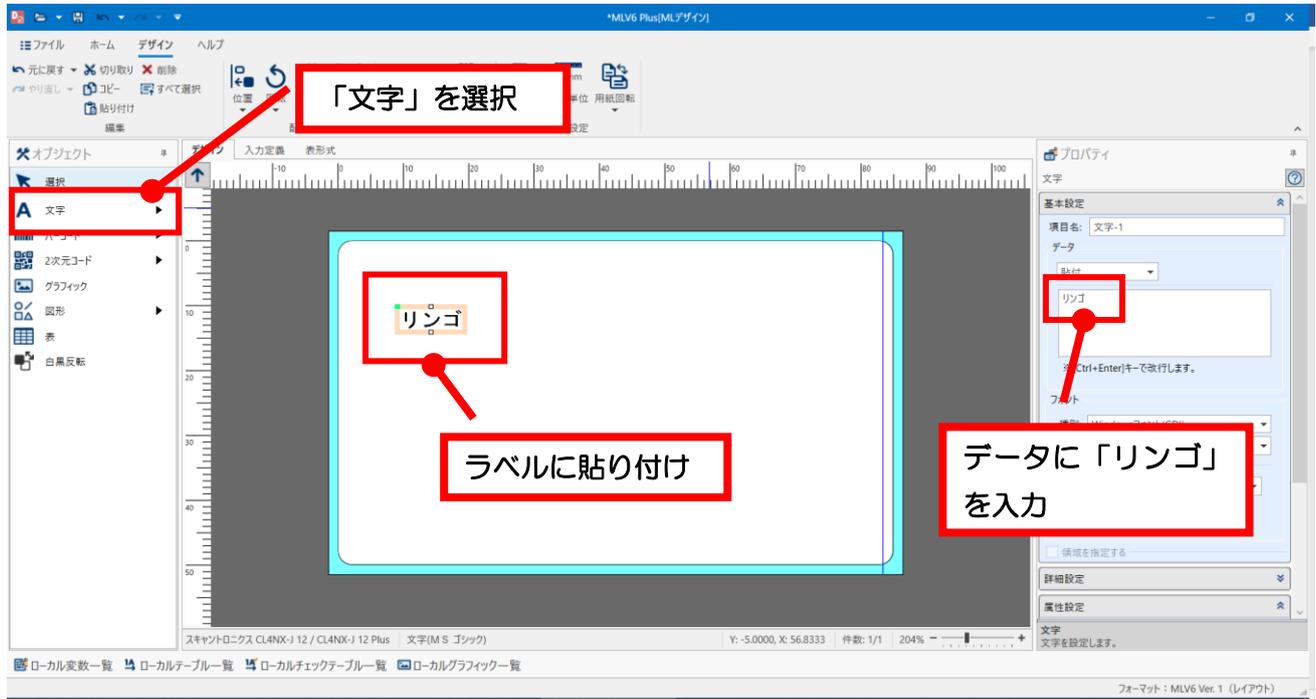
## オブジェクトでデザインを作成する

■基本操作 ■文字 ■バーコード ■2次元コード ■表

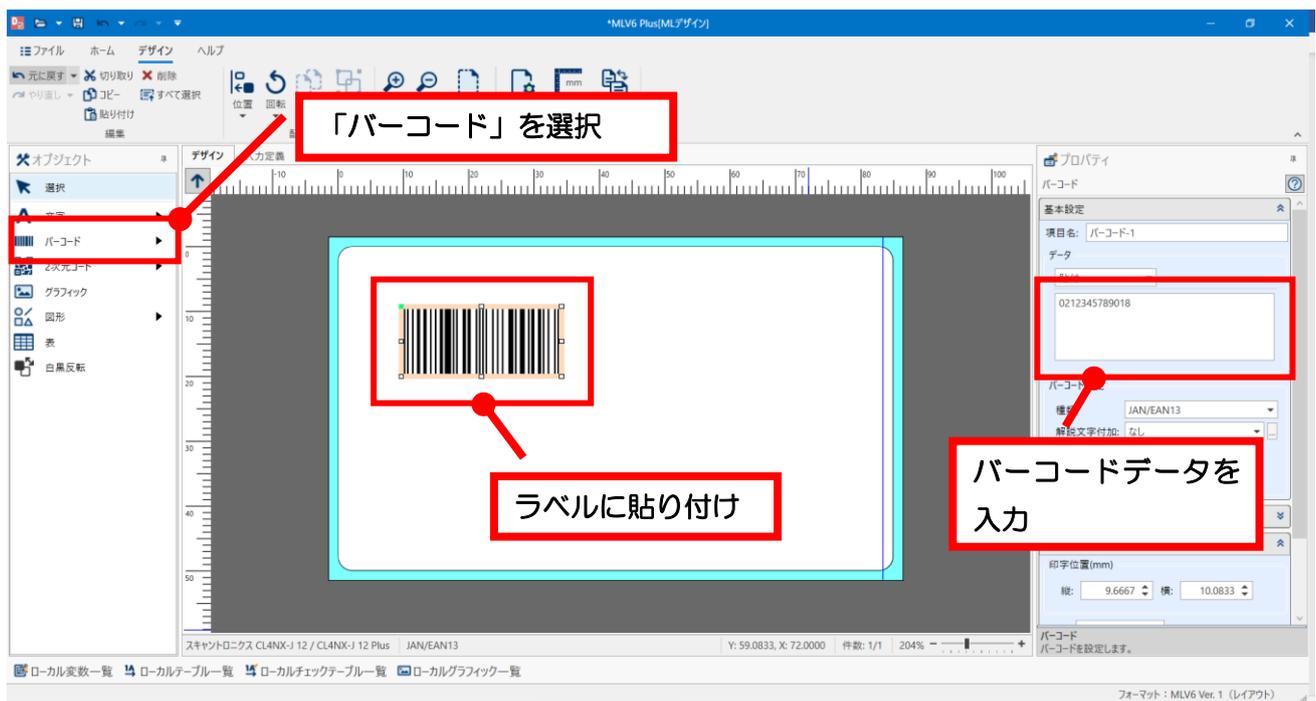
## ■基本操作

「オブジェクト」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。

例) 文字を貼り付ける



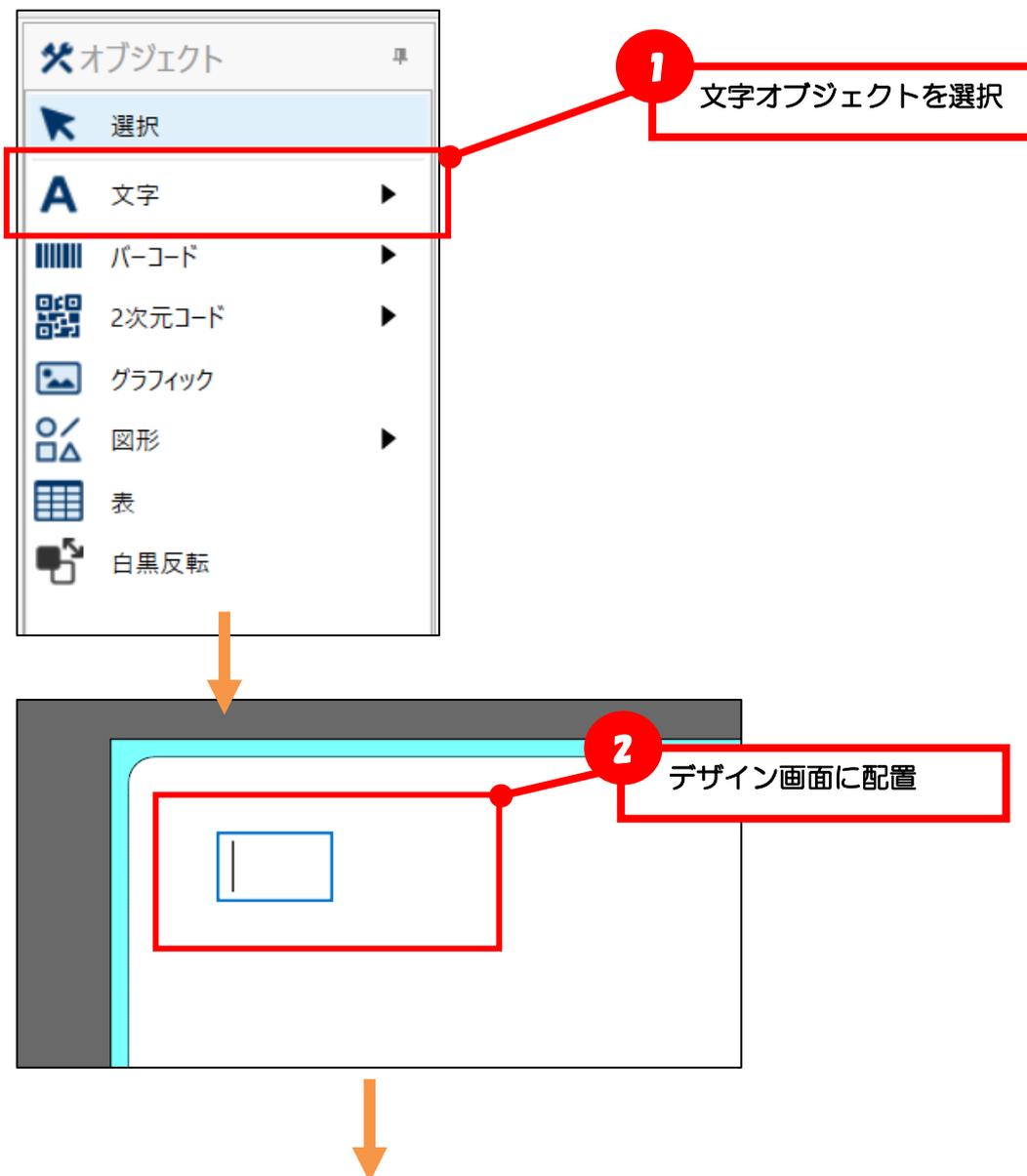
例) バーコードを貼り付ける

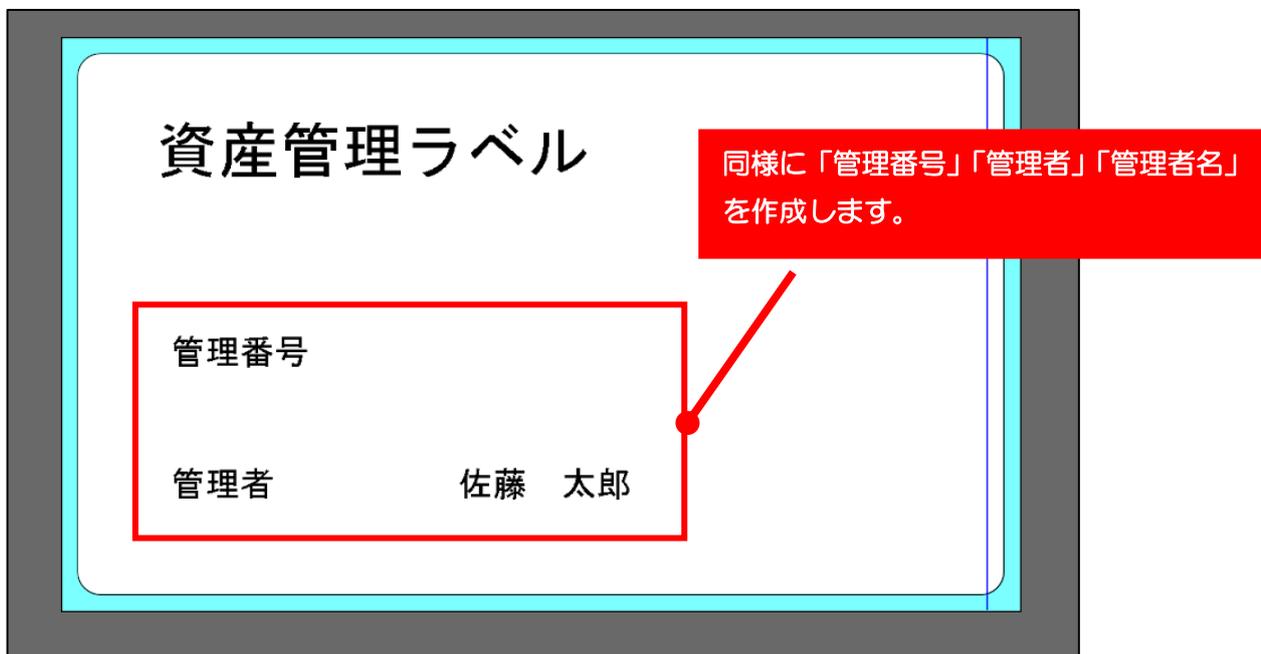
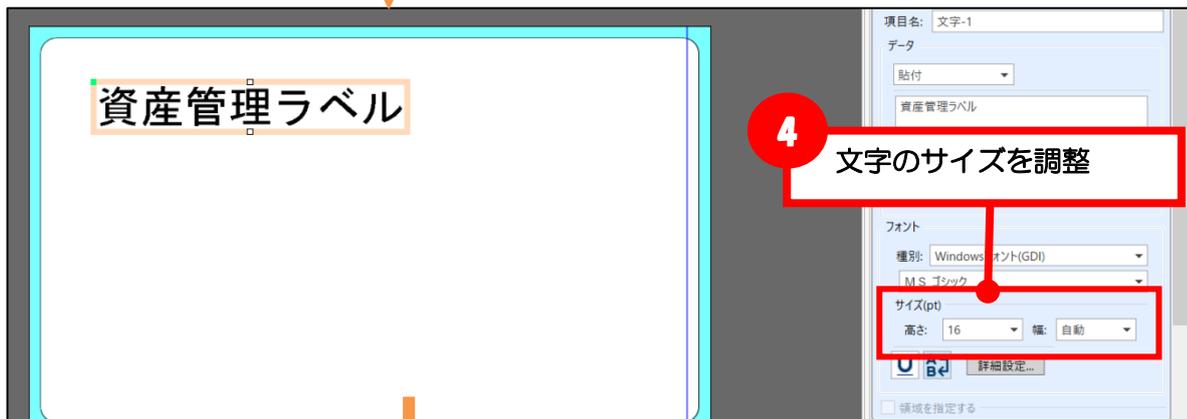


練習として「資産管理ラベル」を作成します。



### ■文字オブジェクトの作成

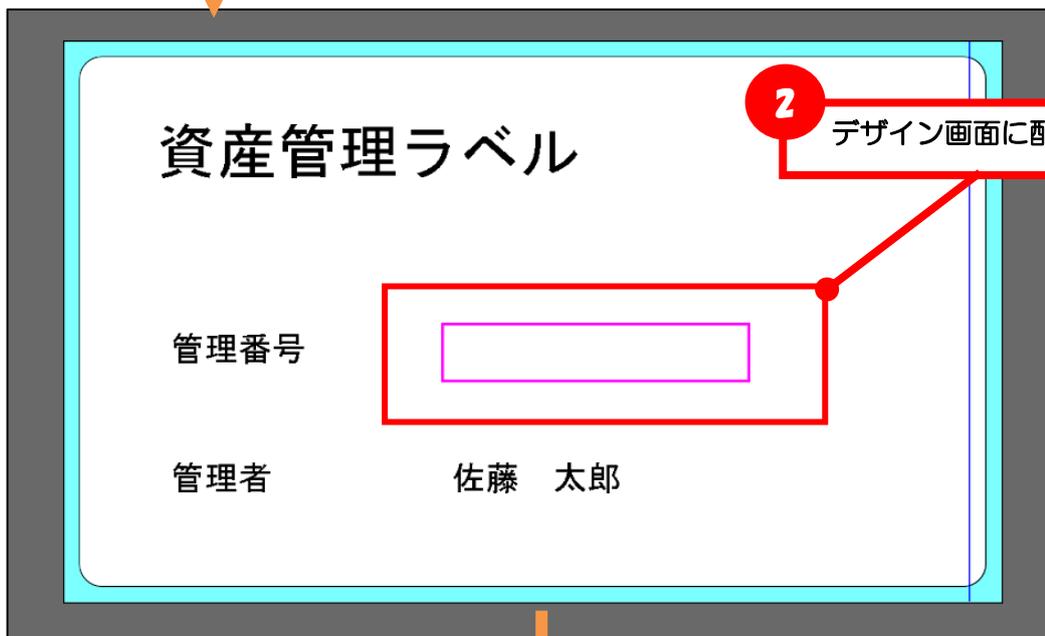




■バーコードオブジェクトの作成



1 バーコードオブジェクトを選択



2 デザイン画面に配置

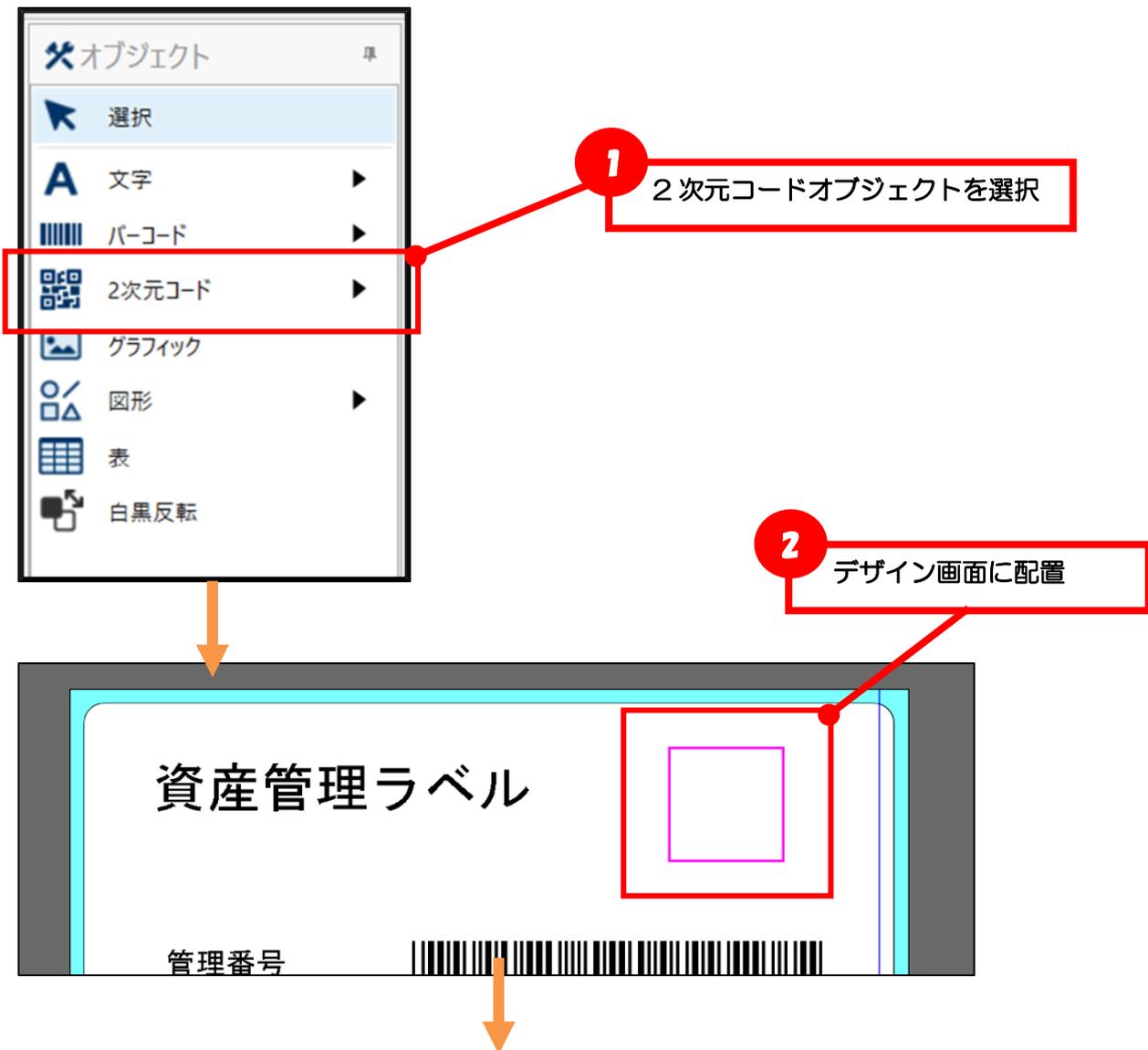


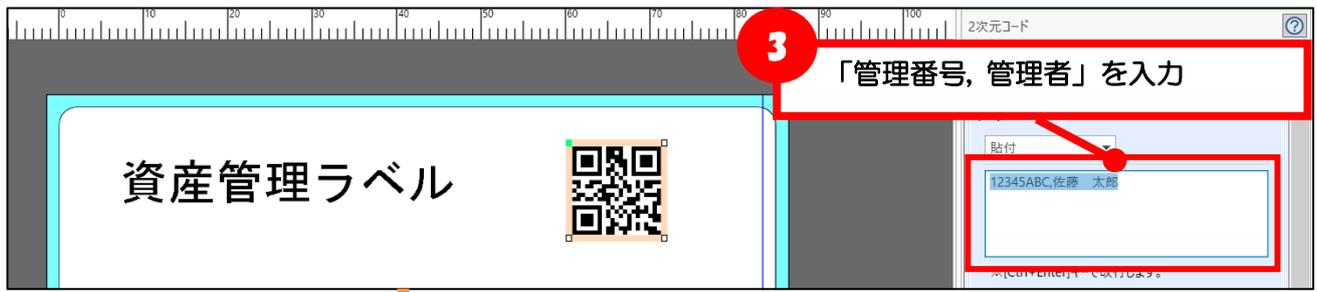
3 「管理番号」を入力

4 「CODE39」を選択

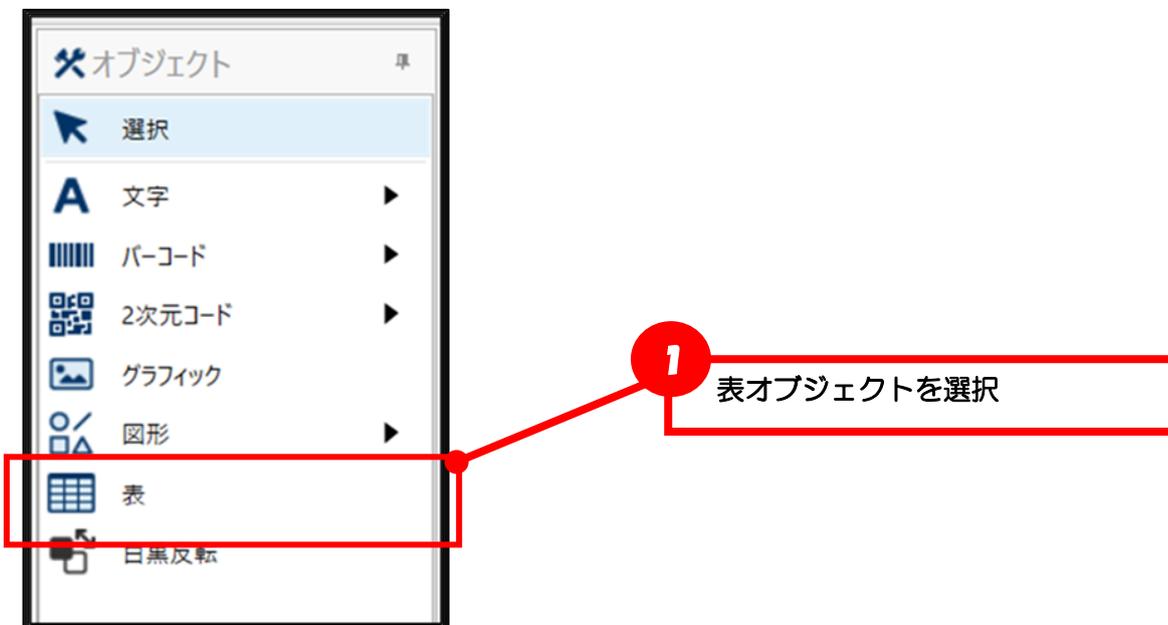


### ■2次元コードオブジェクトの作成





### ■表オブジェクトの作成



**2** デザイン画面に配置

資産管理ラベル

管理番号

管理者 佐藤 太郎

**3** 行数を「2」に変更

基本設定  
項目名: 表-1  
行  
行数: 2 横線の縦位置を指定する  
横線の太さ: 0.1667 (mm)  
列  
列数: 2 縦線の横位置を指定する  
縦線の太さ: 0.1667 (mm)  
カド丸半径: 0 (mm)  
サイズ(mm)  
高さ: 24.9167 幅: 77.3333  
セルを結合するには、[Ctrl]+[Shift]キーを押しながらマウスをクリックして2つ以上のセルを選択し、マウスを右クリックして表示されたメニューから[セルを結合]を選択します。  
属性設定  
印字位置(mm)



### [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「」の状態ドラッグすると変更できます。

次章で変数との紐付けを行うため、表オブジェクトを右クリックして「最背面へ移動」を選択します。



作業が終わったらレイアウトファイルを保存します。

保存場所は任意ですが、ここでは「C:\¥SATO」フォルダに保存します。

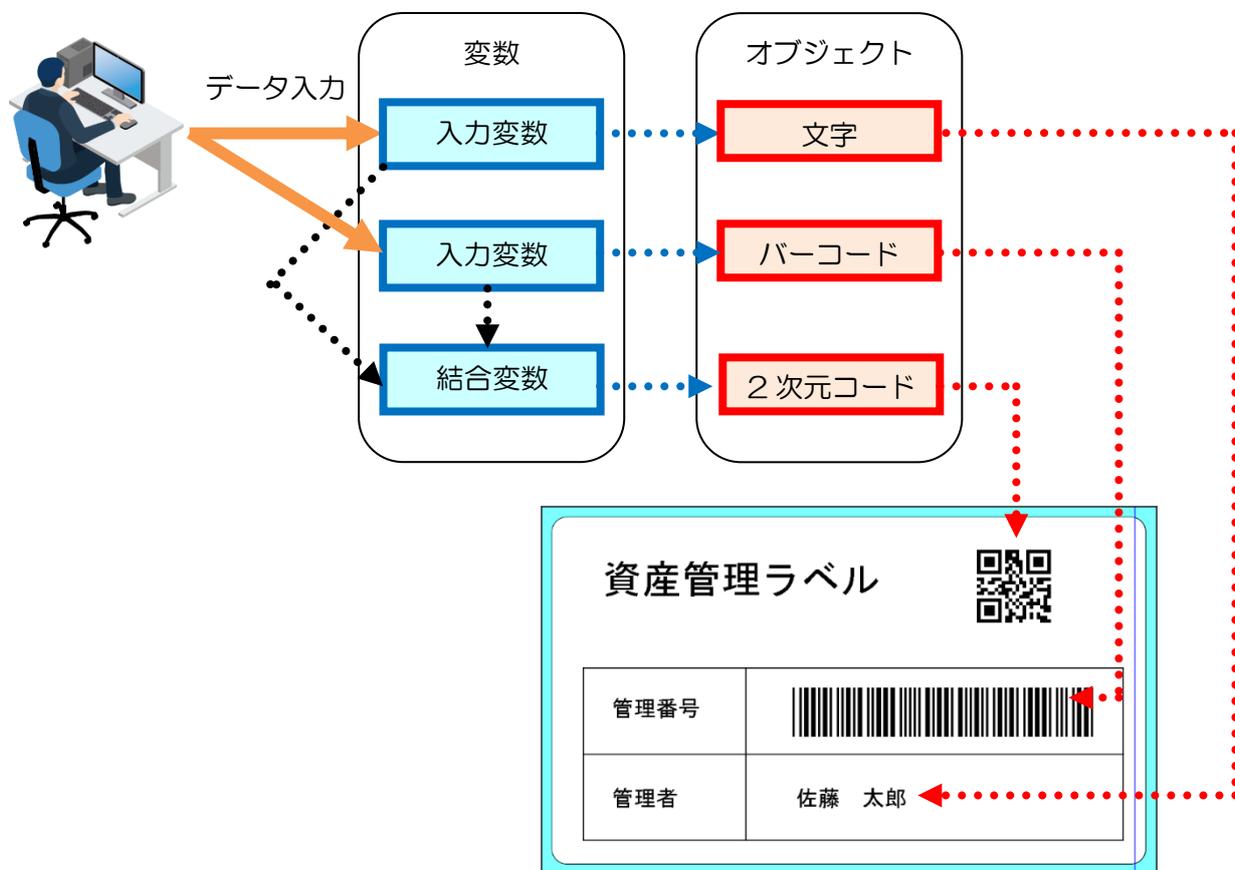
以上でデザイン作成は完了です。

## 3

## 変数でデータを入力する

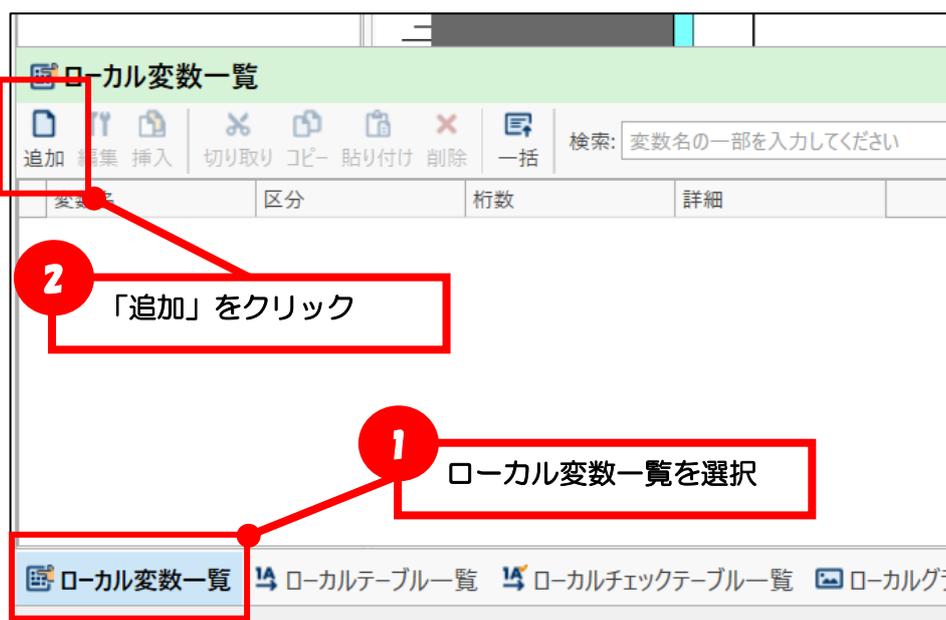
■変数の作成 ■オブジェクトの紐付け ■入力順序

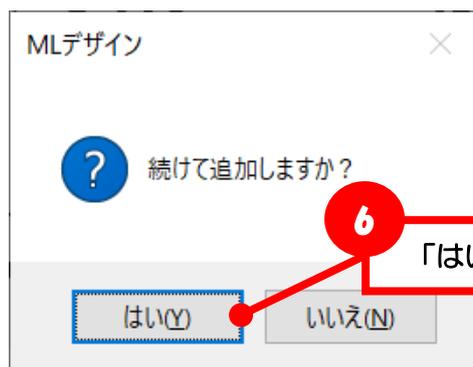
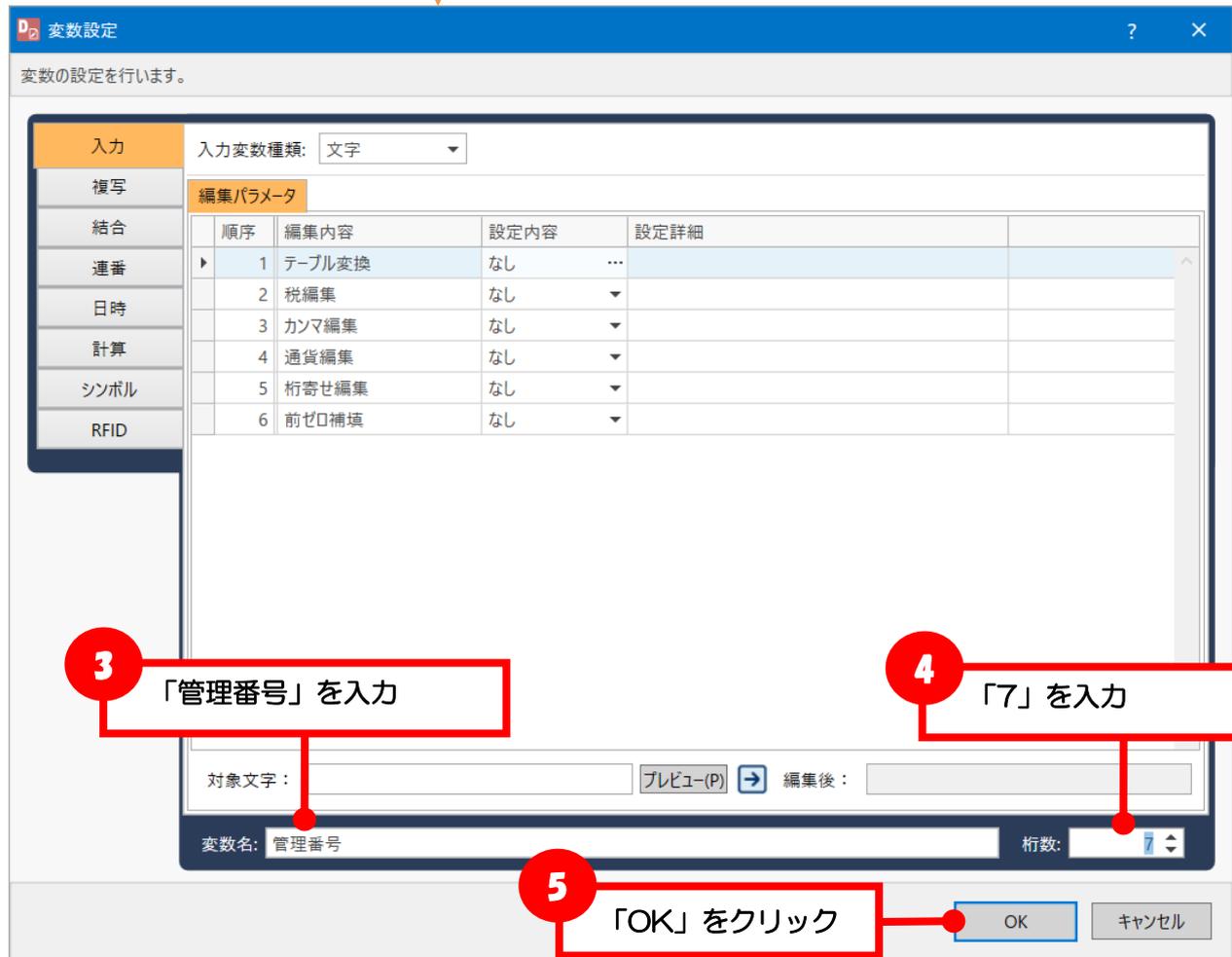
発行時に入力したデータをラベルに印字するために、データの箱となる「変数」を作成して、オブジェクトと紐付けを行います。

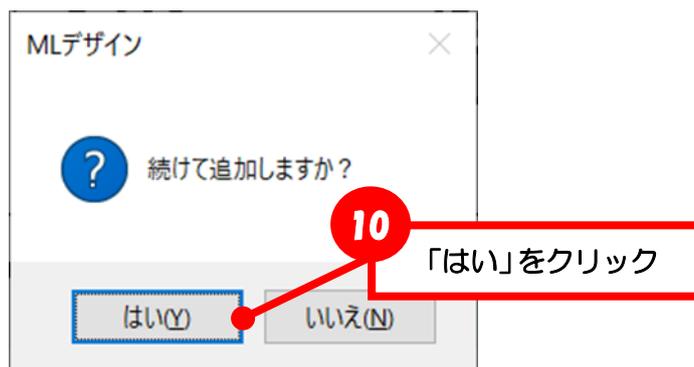
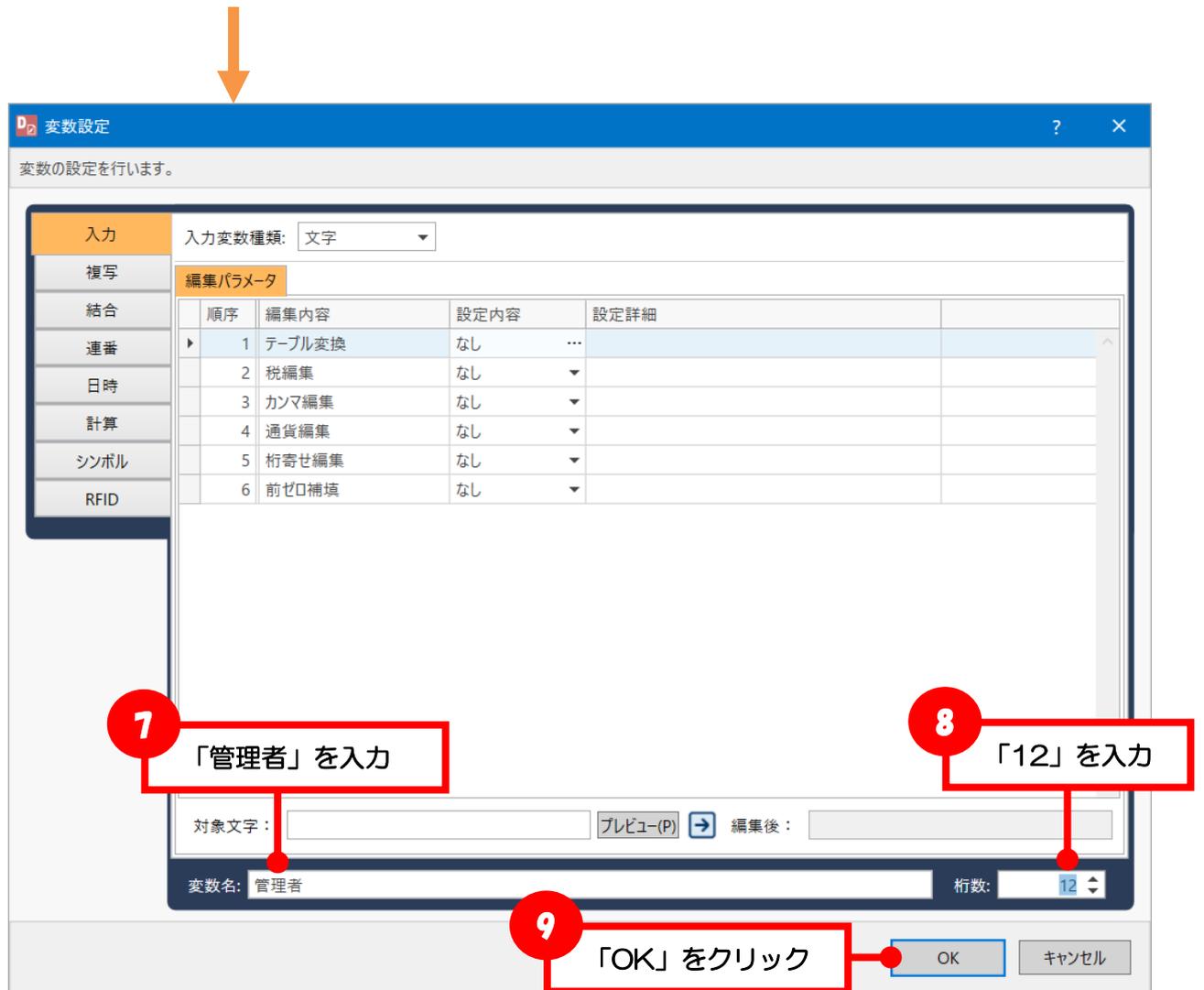


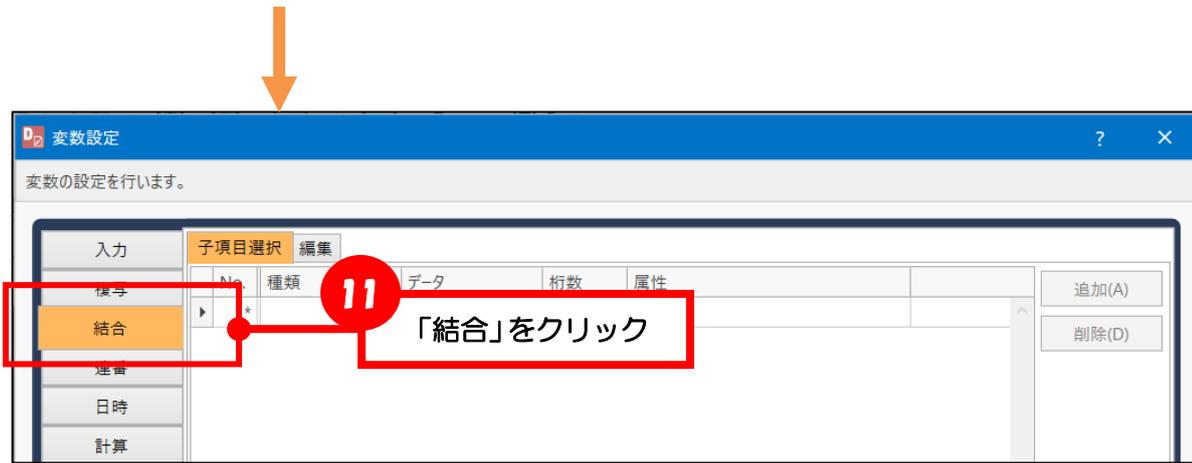
## ■変数の作成

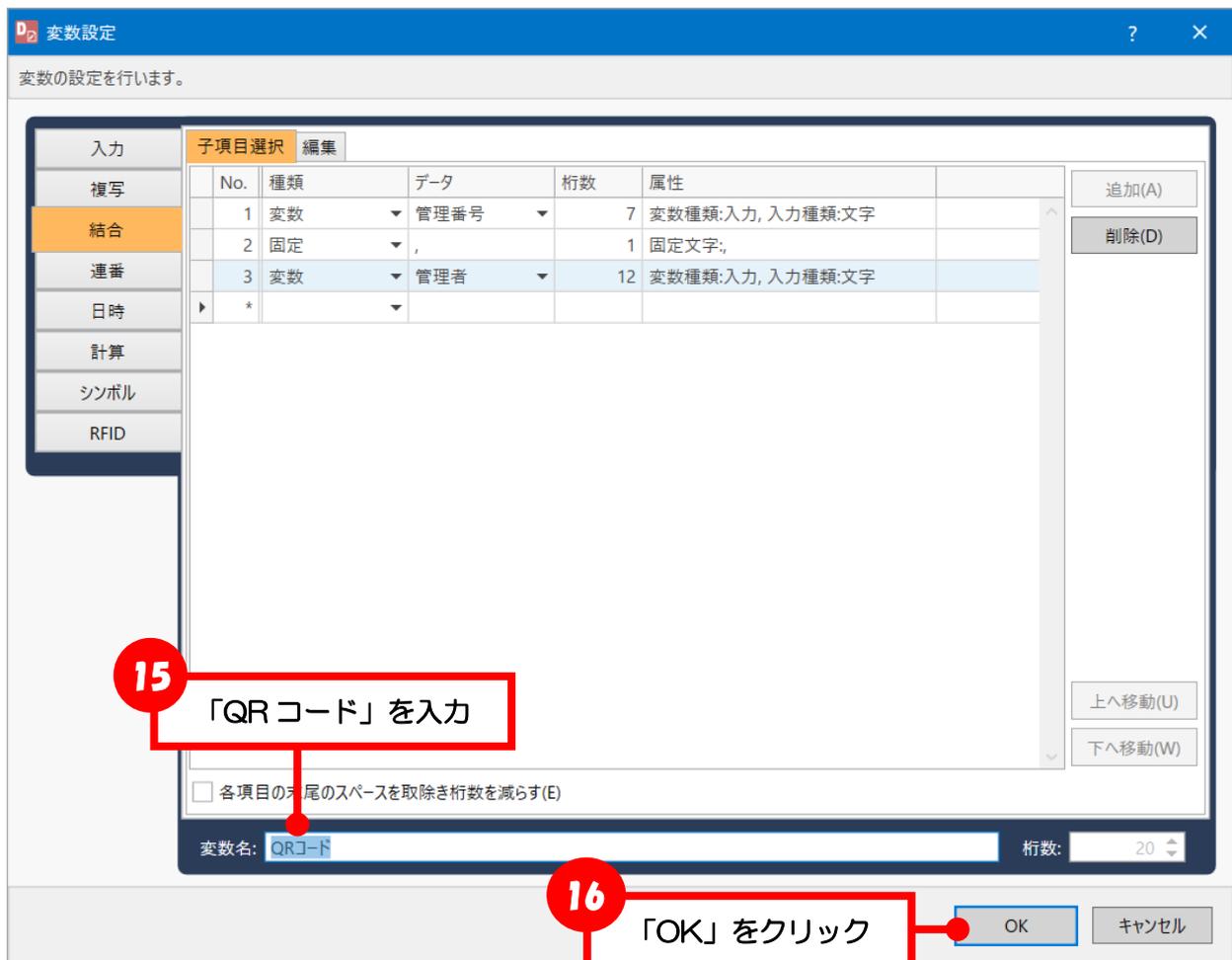
データを入力する「入力変数」と、変数や固定値を結合する「結合変数」を作成します。





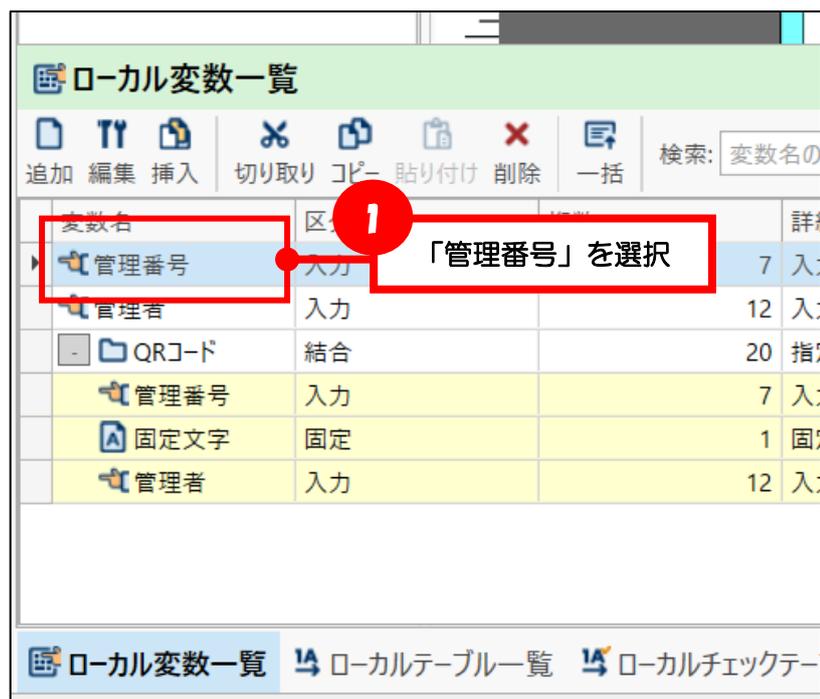


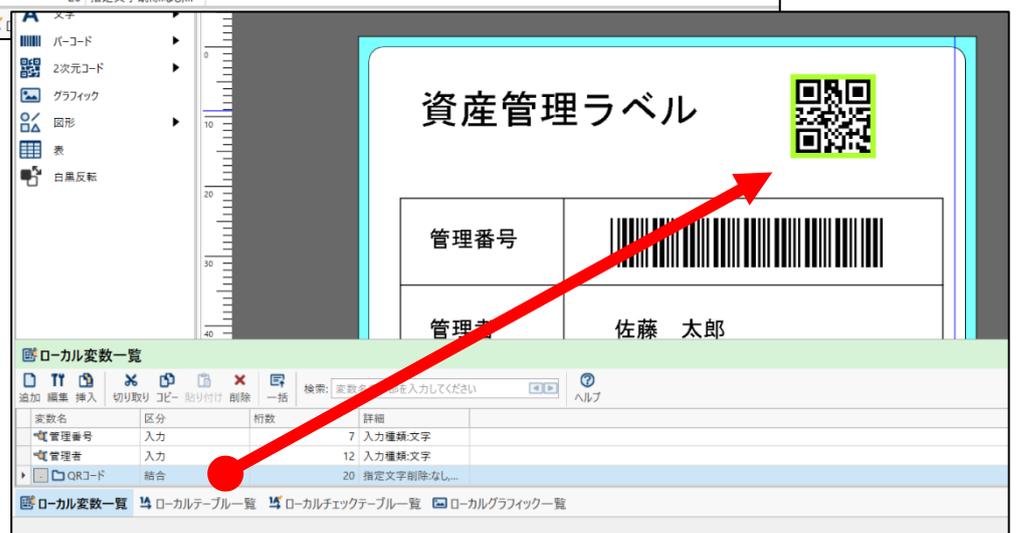
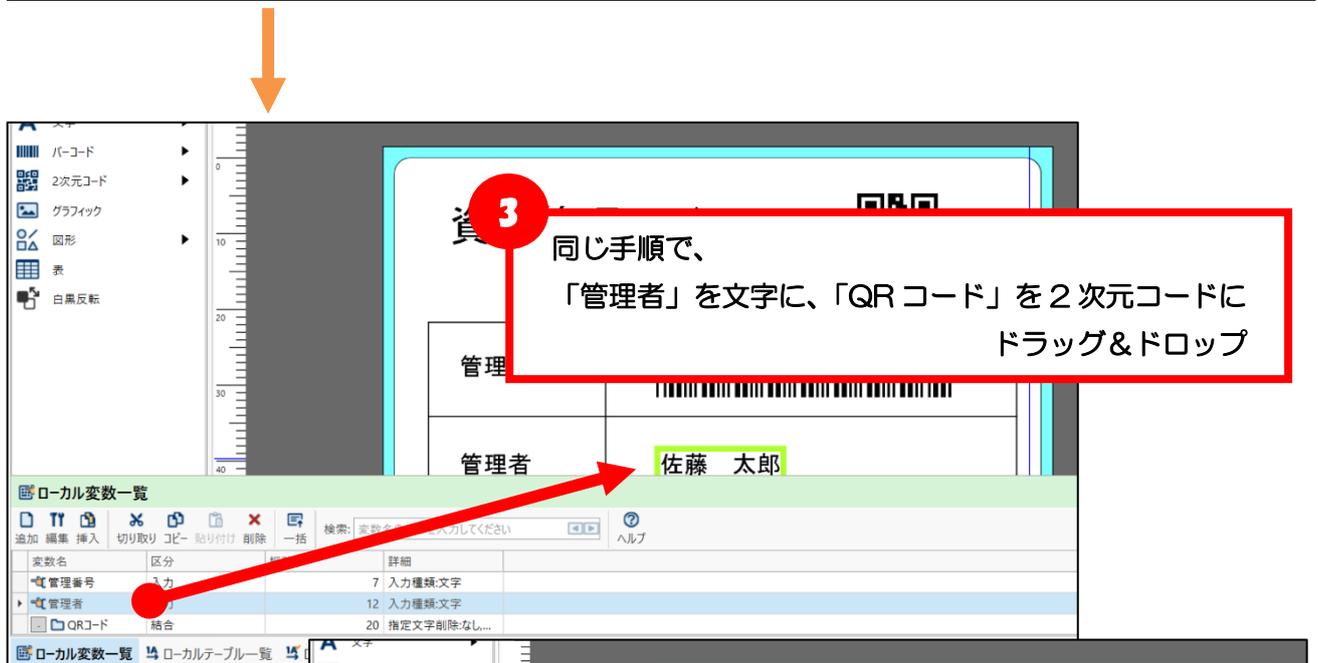
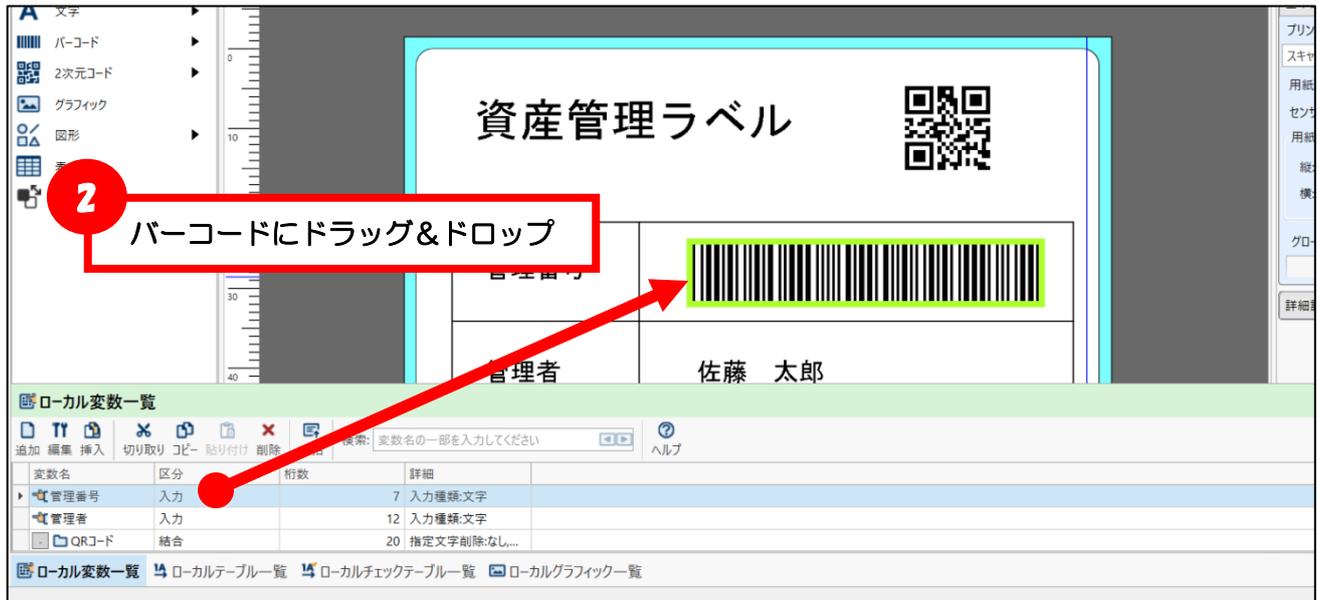




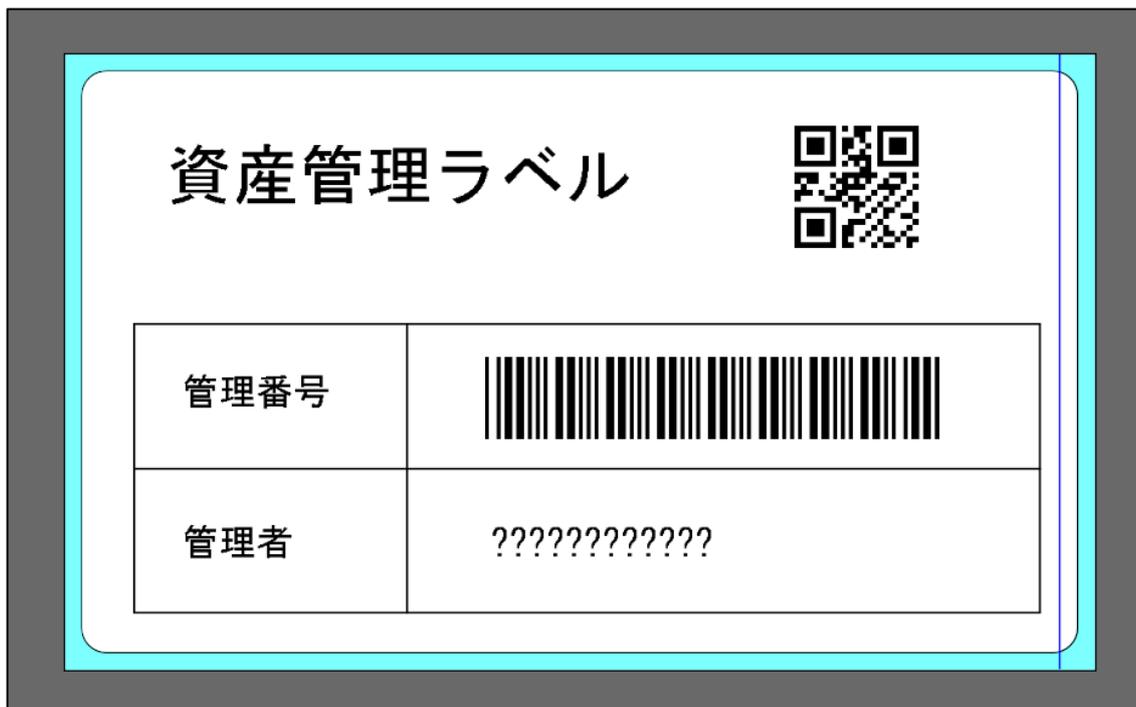
### ■変数とオブジェクトの紐付け

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。

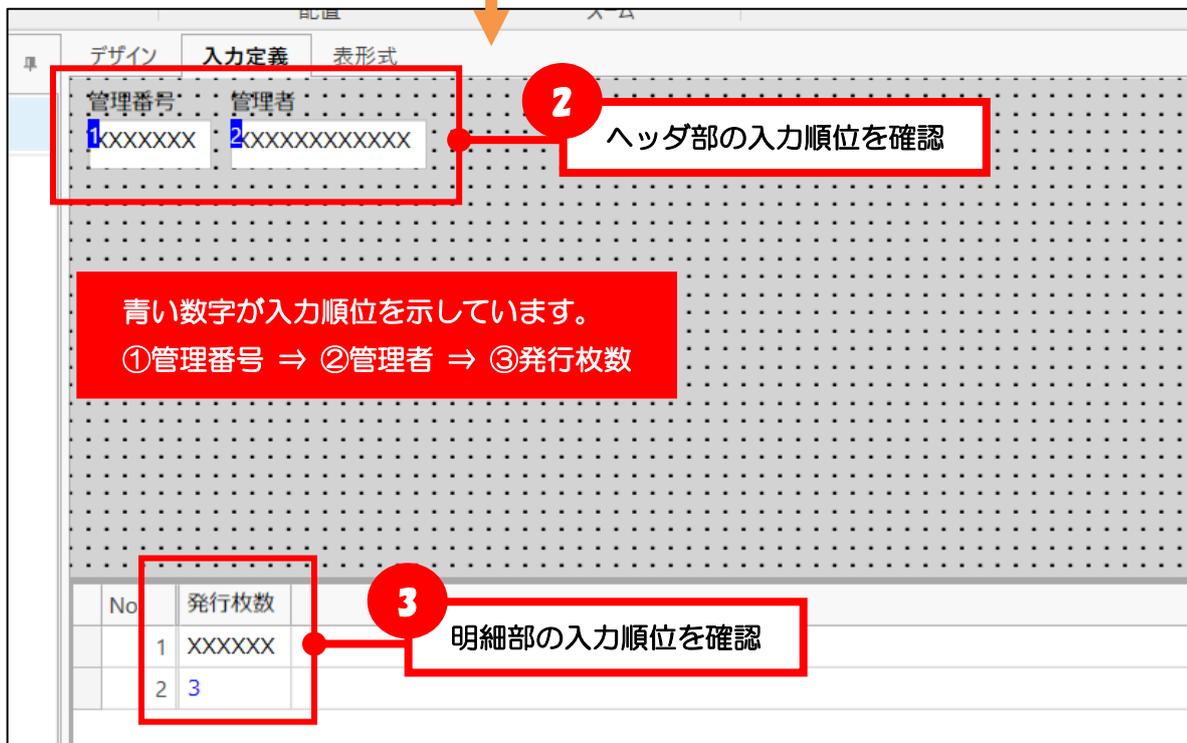
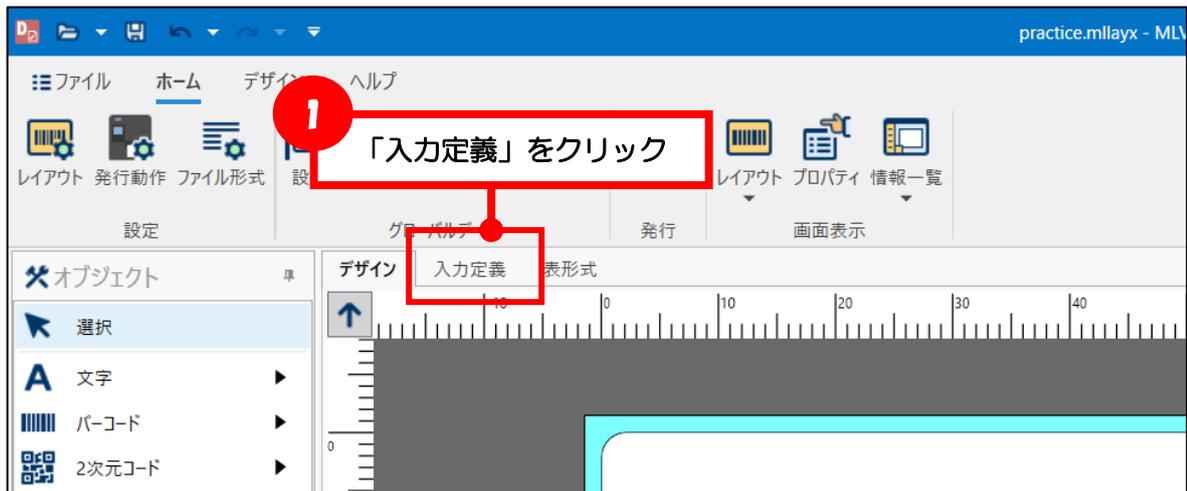
アプリケーションの記述例 (VB)

```
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

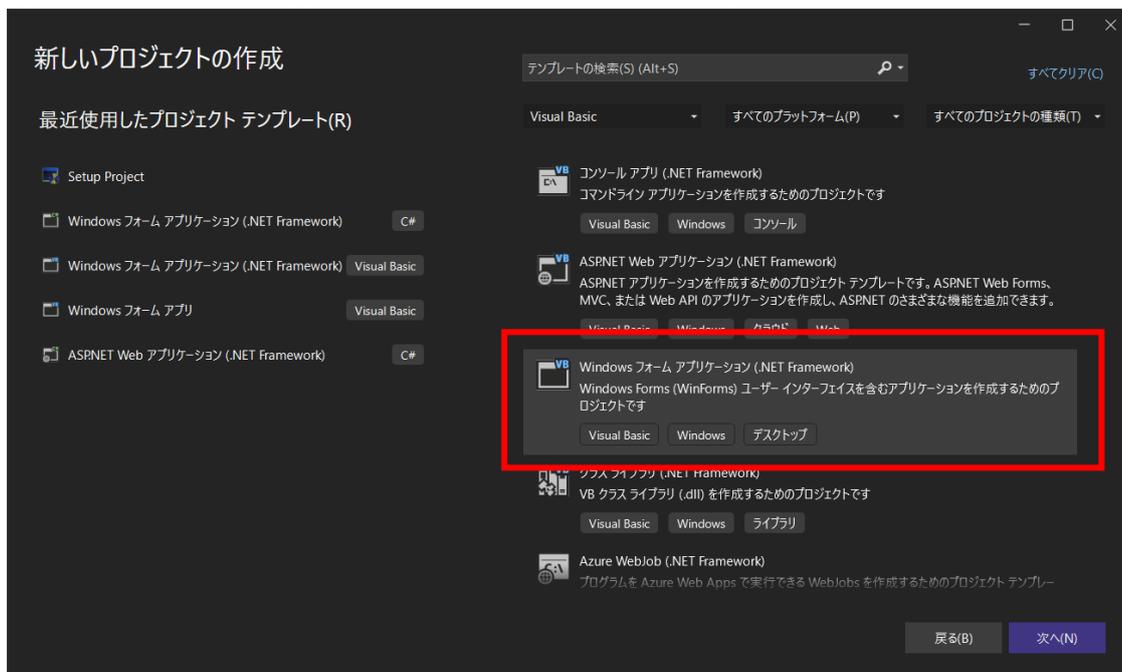
## MLComponent を利用する

■変数の作成 ■オブジェクトの紐付け ■入力順序

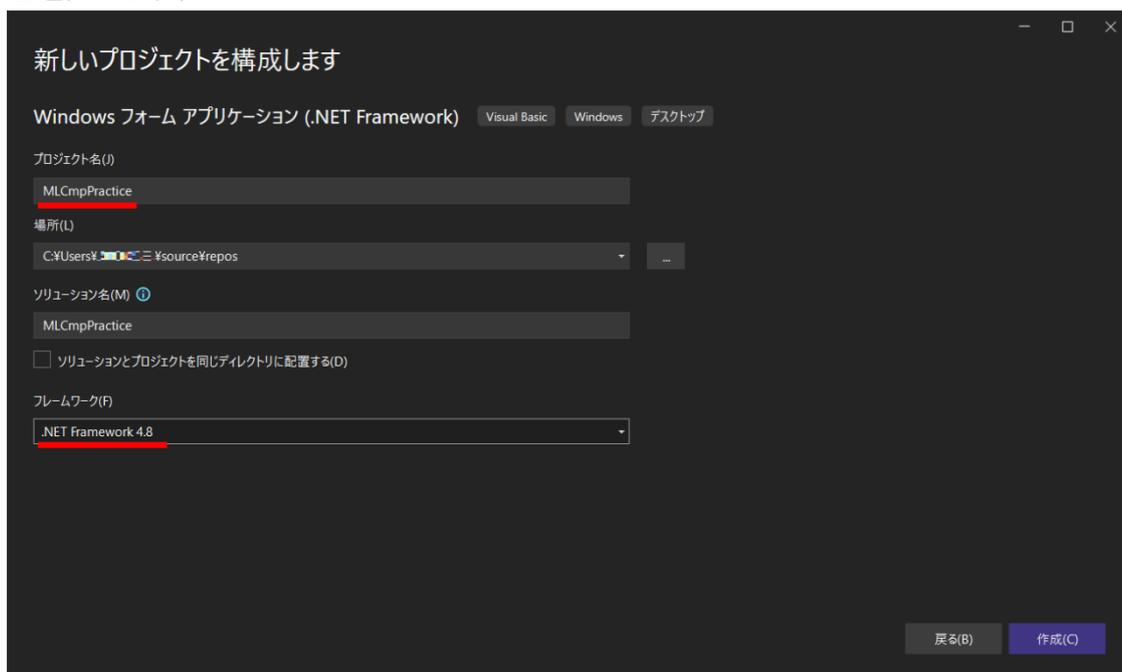
Visual Studio 2022 で MLComponent を利用した Windows フォームアプリケーションを開発します。本書では開発言語を Visual Basic で説明します。なお、Visual Studio 2022 の利用方法については説明を省略しています。別途オンラインヘルプなどをご参照ください。

## ■プロジェクトの作成

新しいプロジェクトの作成で「Windows フォームアプリケーション(.NET Framework)」を選びます。

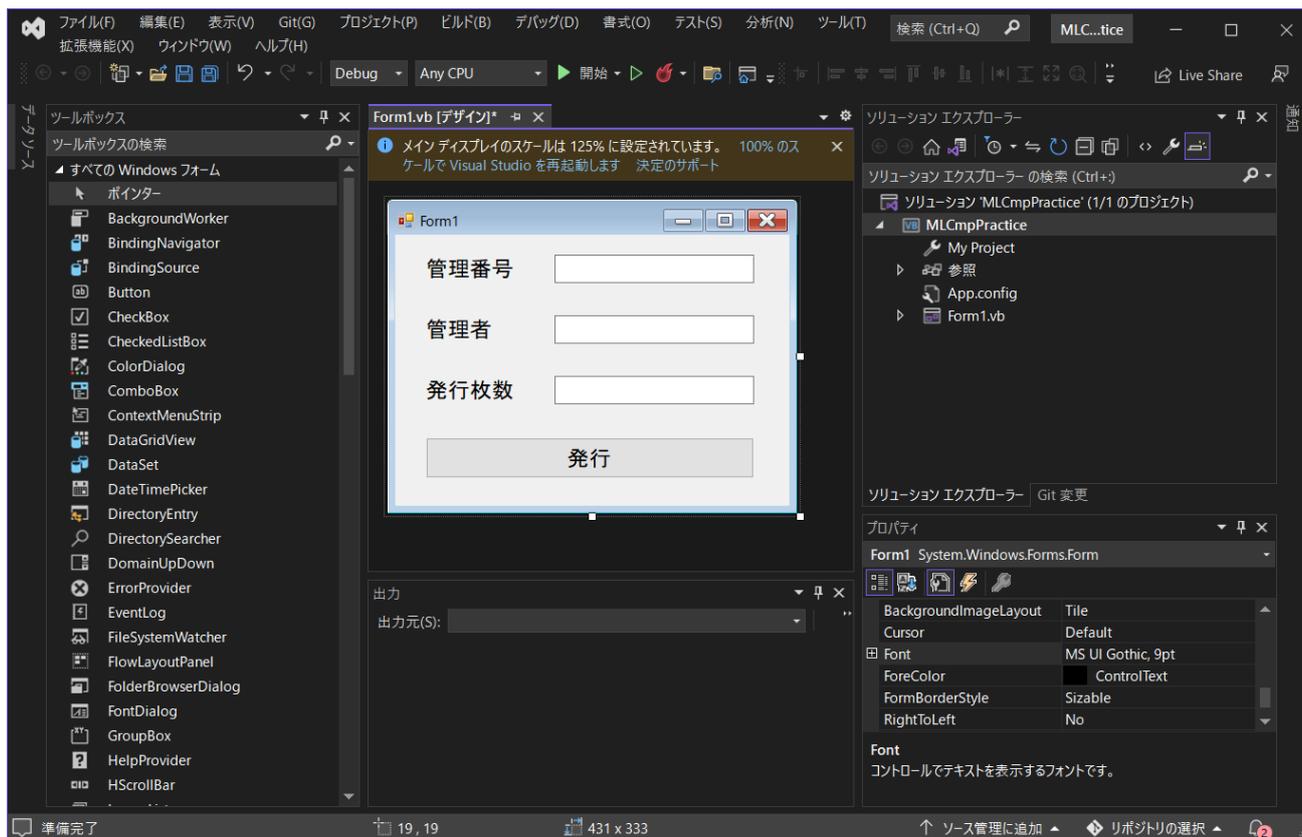


プロジェクト名に「MLCmpPractice」、フレームワークに「.NET Framework 4.8」を選び、「作成」を選択します。

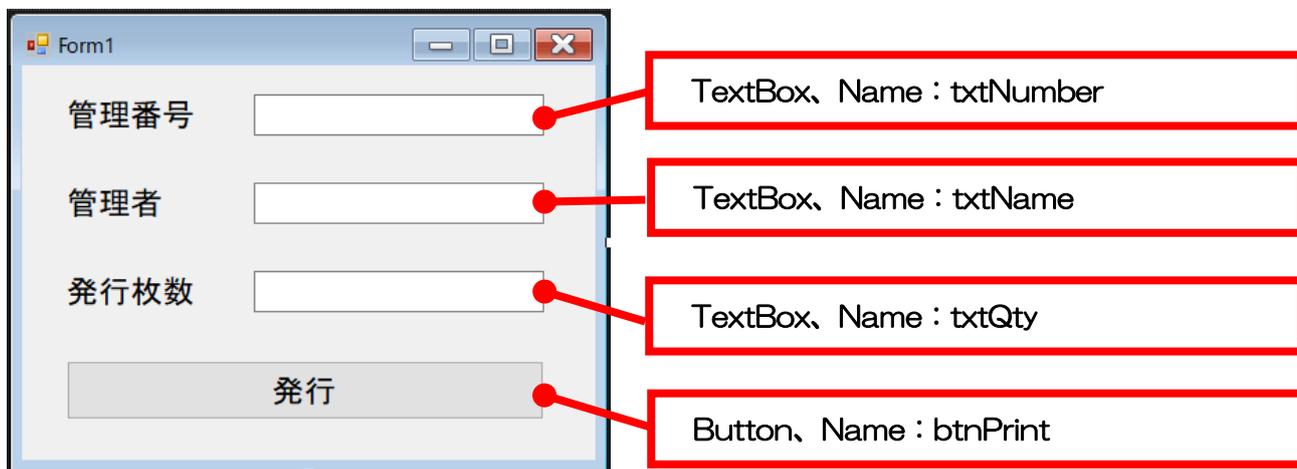


## ■発行画面の作成

発行画面を作成します。



本書では、各コントロールの名称を以下の例で説明します。

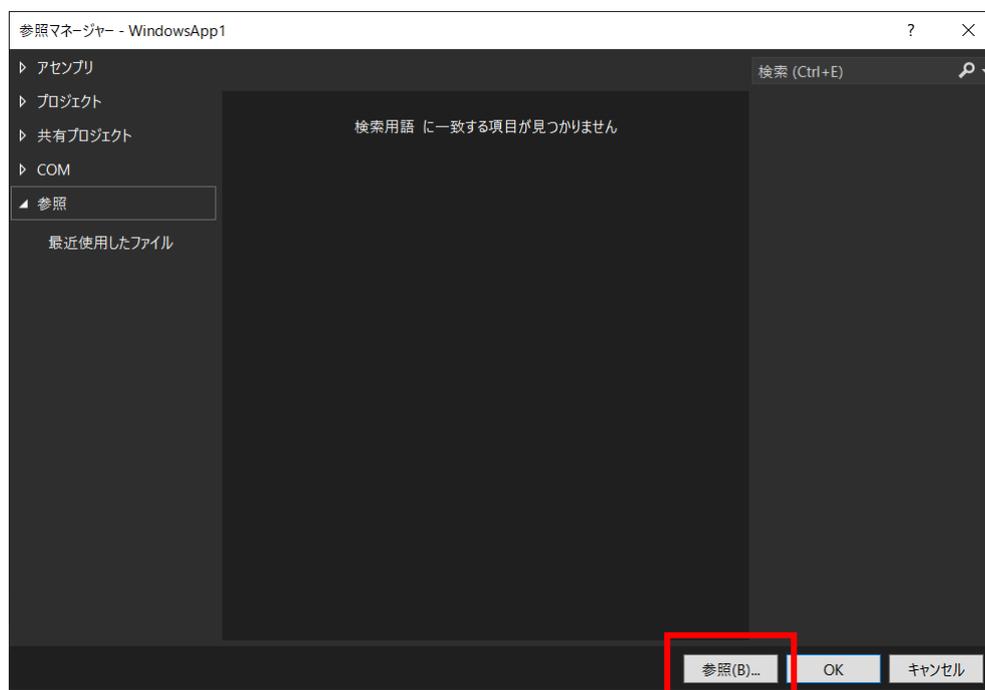


## ■MLComponentの参照設定

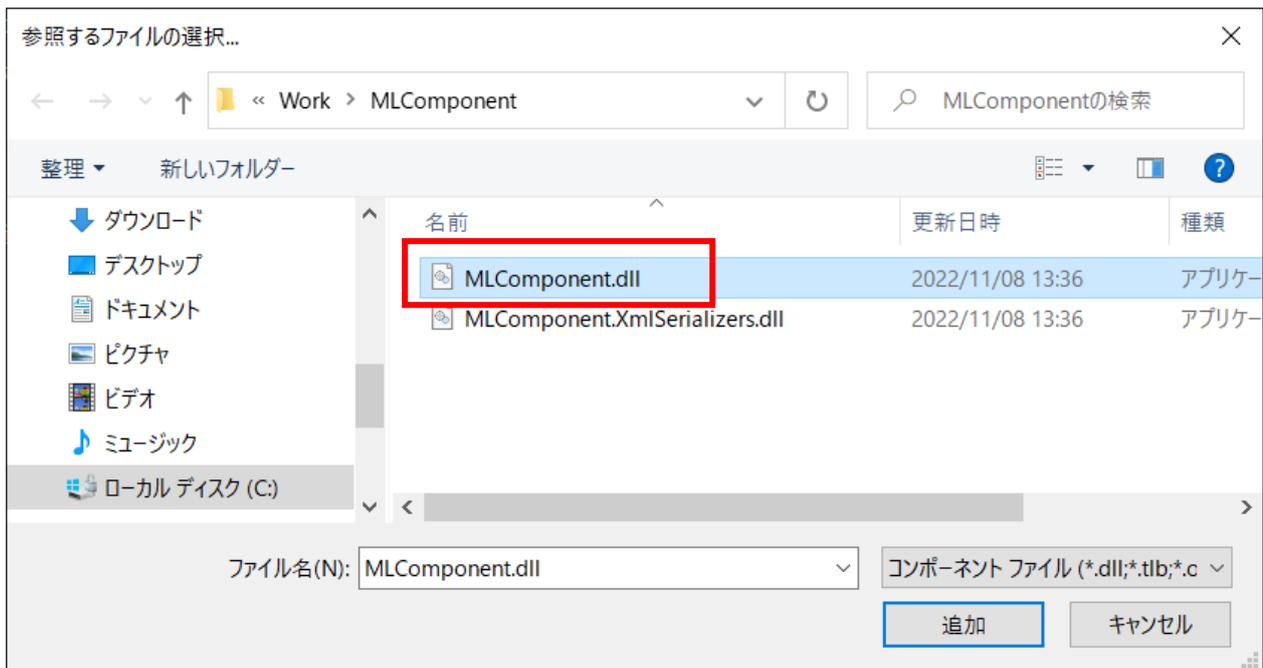
「プロジェクト」>「参照の追加」を選択します。



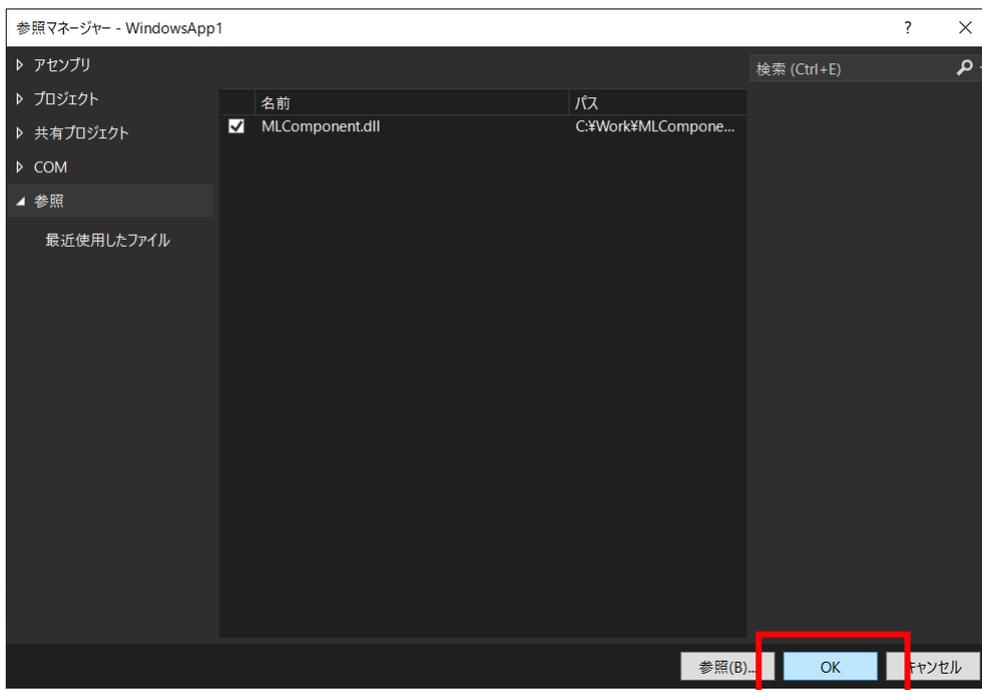
「参照」を選択します。



「MLComponent.dll」を選択して「追加」を選択します。



「参照」に「MLComponent.dll」が追加されたことを確認したら、「OK」を選択します。



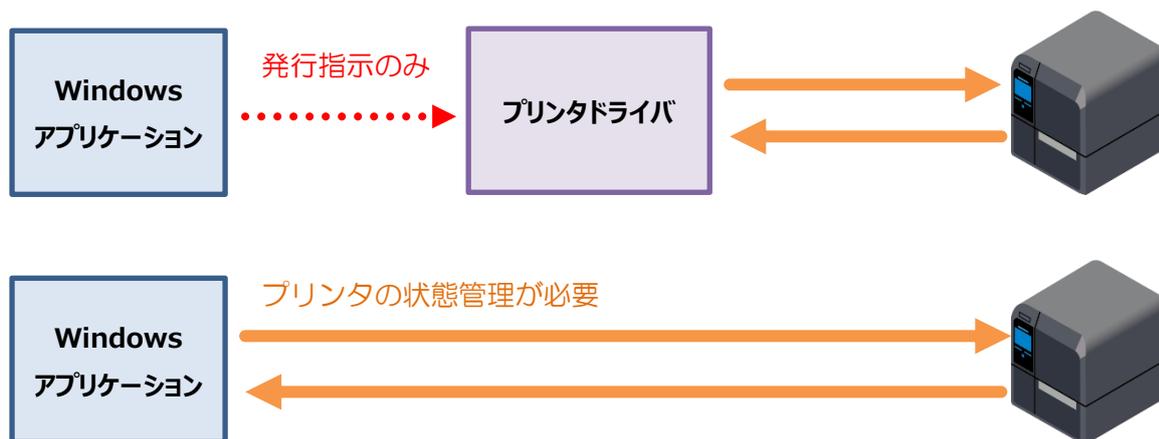
## 2

## 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	LAN（無線 LAN） USB RS-232C	LAN（無線 LAN） USB RS-232C Bluetooth
開発レベル	<b>初級</b>	<b>中級～上級</b>
プリンタの制御	<b>不要</b>  (プリンタドライバにおまかせ)	<b>必要</b>
メリット	通信制御はプリンタドライバが行うので、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不要。
デメリット	各 PC にプリンタドライバのインストールが必要。 アプリからプリンタの状態は確認できない。	プリンタとの通信制御の実装が必要。通信に関わ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 btnPrint_Click(sender As System.Object, e As System.EventArgs) Handles btnPrint.Click

    'MLComponentのインスタンス生成
    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 CL4NX-J 305dpi"

    '接続
    Dim Result As Integer
    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 btnPrint_Click(sender As System.Object, e As System.EventArgs) Handles btnPrint.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"
```

```
'出力先の設定(例としてLAN接続を指定します。)
```

```
MLComponent.Setting = "LAN:192.168.1.9,1024"
```

```
'通信プロトコル(ステータス4を指定します。)
```

```
MLComponent.Protocol = SATO.MLComponent.Protocols.Status4
```

```
'通信タイムアウト値(デフォルトは3秒です。)
```

```
MLComponent.Timeout = 3
```

```
'接続
```

```
Dim Result As Integer
```

```
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(発行前)
```

```
Dim Status As String = ""
```

```
Dim PrinterSendable As Boolean = False
```

```
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)
```

```
Else
```

```
    'プリンタ状態判定(送信可能な状態のステータスか判定します。)
```

```
    Select Case Status.Substring(2, 1)
```

```
        Case "A", "B"
```

```
            PrinterSendable = True '送信可能
```

```
        Case Else
```

```
            MessageBox.Show("プリンタの準備ができていません。 Status = " & Status)
```

```
    End Select
```

```
End If
```

```
'プリンタ状態確認失敗時
```

```
If PrinterSendable = False Then
```

```
    Result = MLComponent.ClosePort()
```

```
    If Result <> 0 Then
```

```
        MessageBox.Show("切断エラー No." & Result)
```

```
    End If
```

```
    Exit Sub
```

```
End If
```

```
'データの設定(データをタブ区切りで指定します。)
```

'※同一データのラベルを10枚発行する場合、「発行枚数=1」を複数送信するよりも「発行枚数=10」を1回送信する方が高速にラベル発行できます。

'※1枚ごとに異なるデータのラベルを発行する場合は、データの設定～発行～プリンタ状態確認の処理を1件ずつ行ってください。

```
MLComponent.PrnData = txtNumber.Text & vbTab & txtName.Text & vbTab & txtQty.Text
```

```

'発行
Result = MLComponent.Output()
If Result <> 0 Then
    MessageBox.Show("発行エラー No." & Result)
Else
    'プリンタ状態確認
    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)
    Else
        'プリンタ状態判定(プリンタでエラーが発生しているか判定します。)
        Select Case Status.Substring(2, 1)
            Case "0", "1", "A", "B", "G", "H", "M", "N", "S", "T"
                'エラーなし
            Case Else
                MessageBox.Show("プリンタでエラーが発生しています。Status = " & Status)
        End Select
    End If
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てください。プリンタの準備が整う前に発行指示を行うと、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。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」の「送信可否」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。

Form1

管理番号

管理者

発行枚数



発行完了

正常に発行が完了すると、  
メッセージが表示されます。

## 4

## アプリケーションを配布する

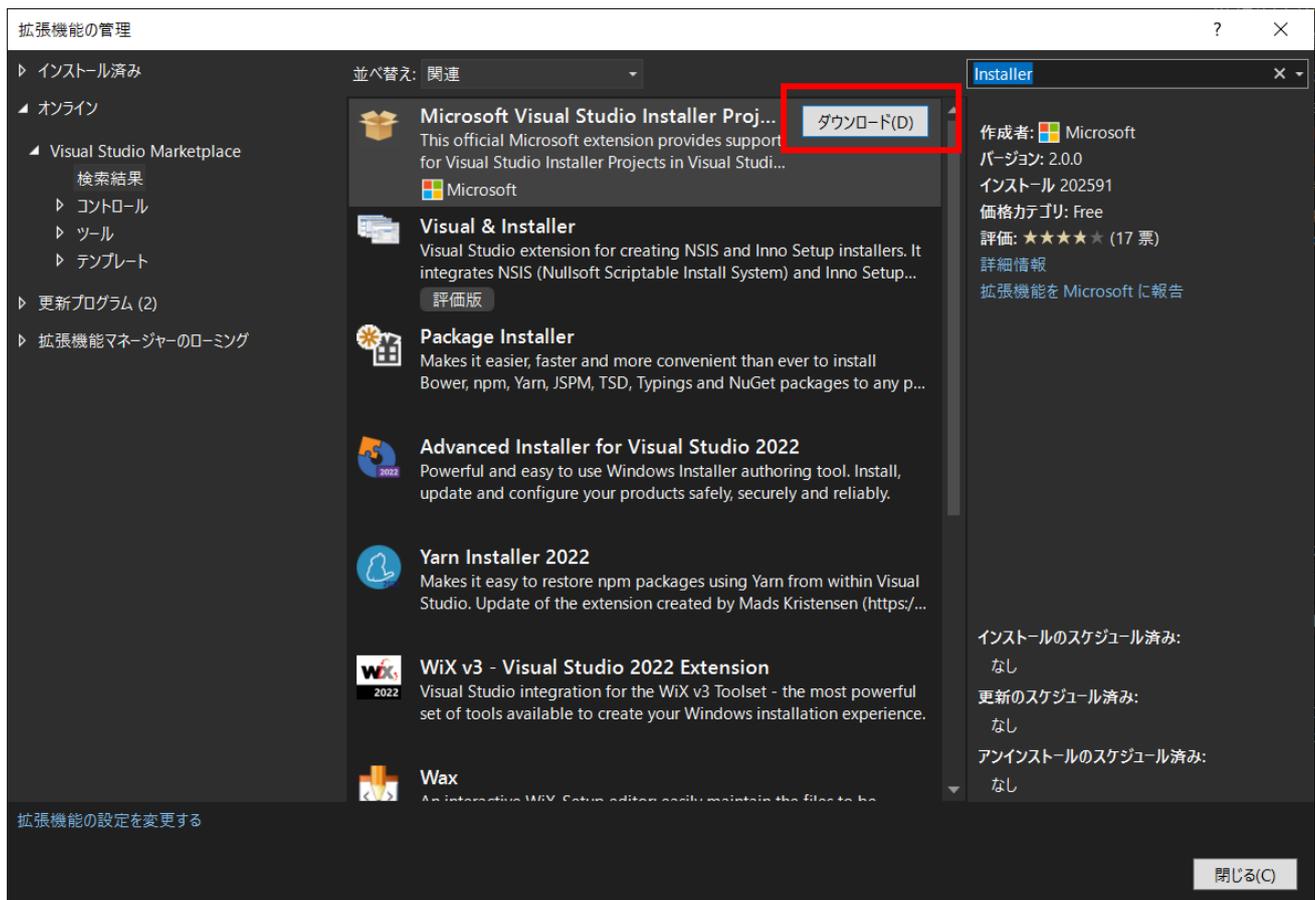
## ■インストーラ

MLComponent はアプリケーションと共に配布してください。実行ファイル (\*.exe) と同じ場所に「MLComponent.dll」と「MLComponent.XmlSerializers.dll」をコピーすることで動作します。ここでは、Visual Studio 2022 の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。

## ■Visual Studio 2022 でインストーラを作成する

「拡張機能」 > 「拡張機能の管理」を選択します。

検索ボックスに「Installer」と入力して、「Microsoft Visual Studio Installer Projects 2022」を検索し、「ダウンロード」を選択します。

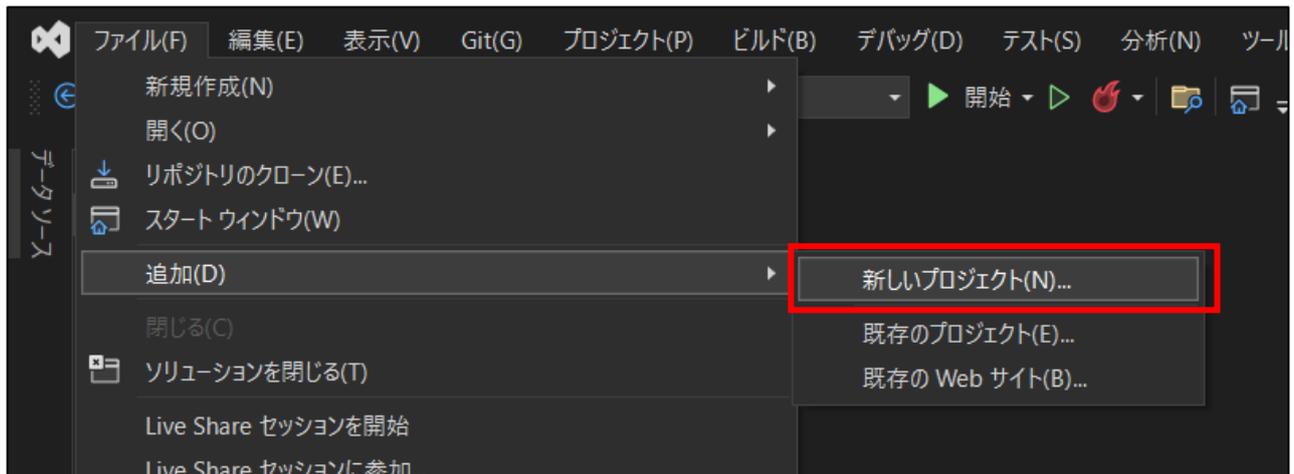


画面を閉じたあと、Visual Studio も終了します。

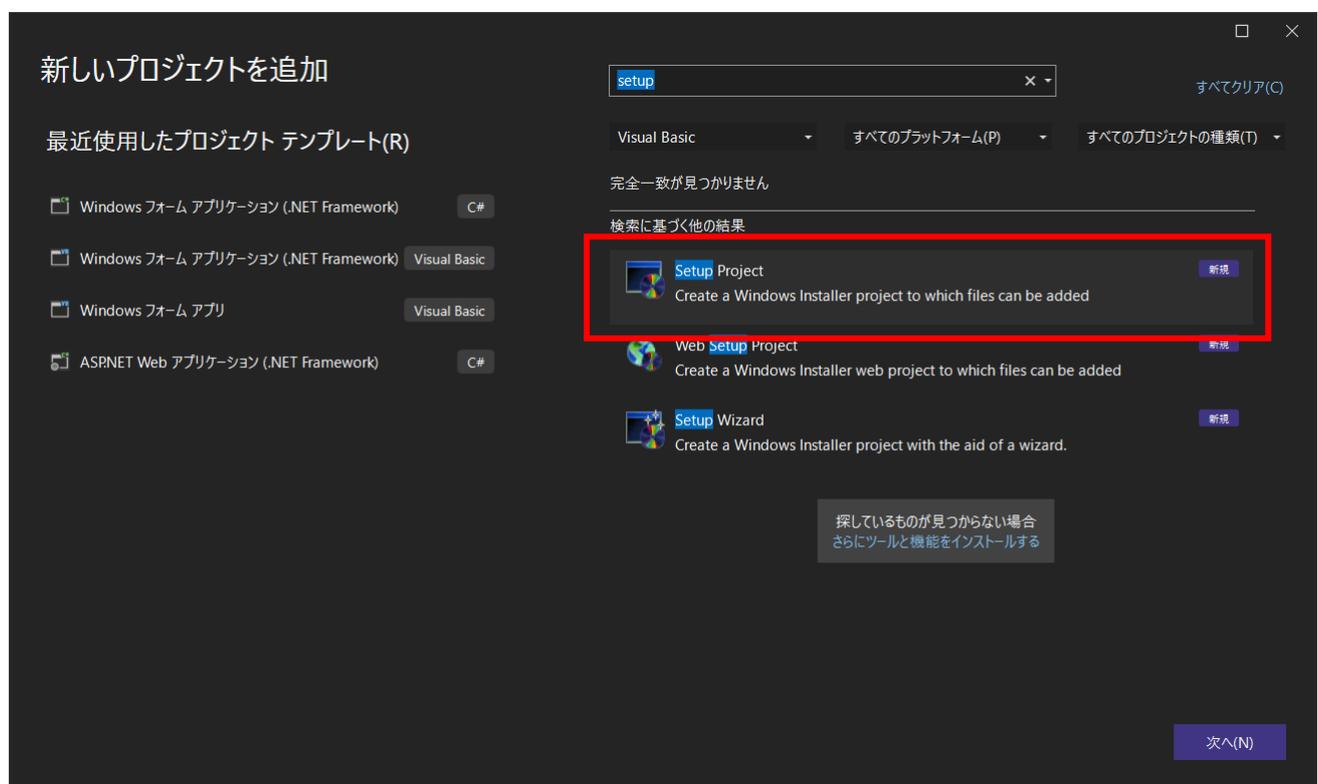
画面の指示に従って拡張機能をインストールします。

インストールが完了したら、再度プロジェクトを開きます。

「ファイル」 > 「追加」 > 「新しいプロジェクト」を選択します。

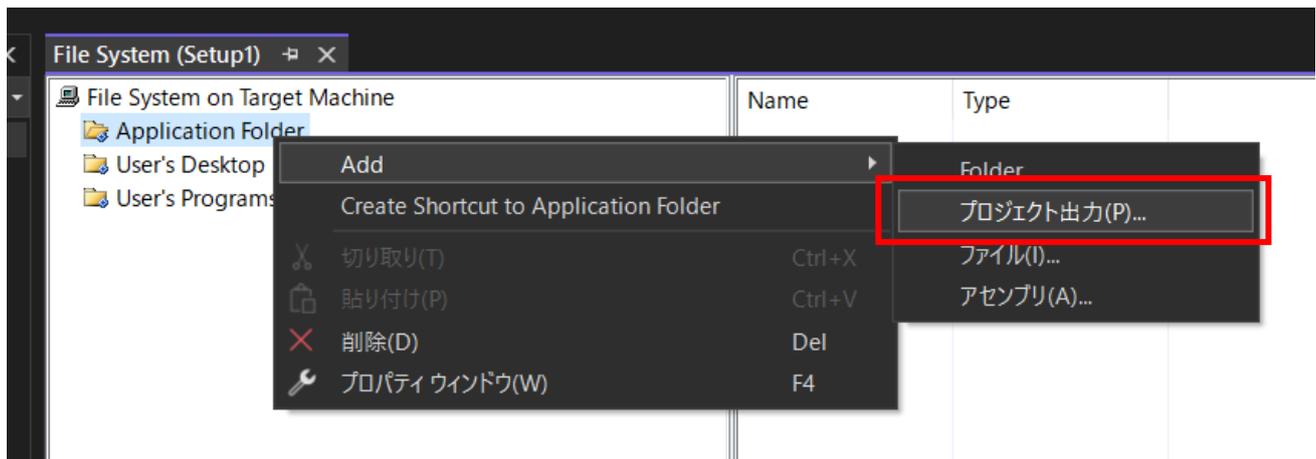


検索ボックスに「setup」と入力、「Setup Project」を選択して「次へ」を選択します。

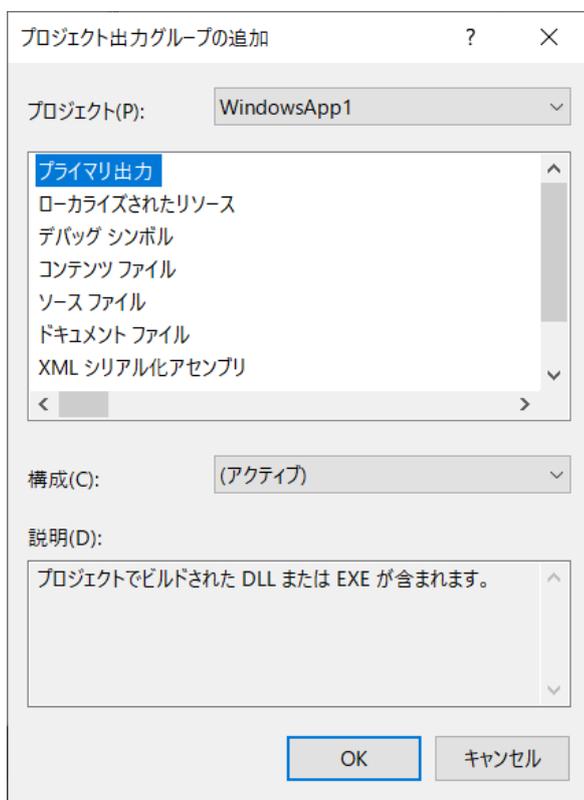


プロジェクト名を入力して「作成」を選択します。  
セットアッププロジェクトが作成されます。

「Application Folder」を右クリックし、「Add」>「プロジェクト出力」を選択します。



「OK」を選択します。



「MLComponent.dll」が追加されます。

