

# TileControl for WinForms

2018.04.11 更新

グレースィティ株式会社

## 目次

<a href="#">はじめに</a>	4
<a href="#">コンポーネントをプロジェクトに組み込む方法</a>	4-5
<a href="#">コンポーネントのランタイムファイル</a>	5
<a href="#">主な特長</a>	6
<a href="#">クイックスタート</a>	7
<a href="#">手順 1: C1Tile アプリケーションの作成</a>	7
<a href="#">手順 2: 要素を含むテンプレートの作成</a>	7-9
<a href="#">手順 3: 指定タイルへのテンプレートの適用</a>	9-10
<a href="#">デザイン時のサポート</a>	11
<a href="#">C1TileControl コンテキストメニュー</a>	11
<a href="#">C1TileControl スマートタグ</a>	12-13
<a href="#">グループタスク</a>	13-14
<a href="#">タイルタスク</a>	14-16
<a href="#">C1TileControl のコレクションエディタ</a>	16
<a href="#">C1TileControl.Templates コレクションエディタ</a>	16-17
<a href="#">Template.Elements コレクションエディタ</a>	17-18
<a href="#">C1TileControl.Groups コレクションエディタ</a>	18-19
<a href="#">Group.Tiles コレクションエディタ</a>	19-20
<a href="#">C1TileControl PropertyMapping コレクションエディタ</a>	20-21
<a href="#">TileControl レイアウト</a>	22-23
<a href="#">TileControl の動作</a>	24
<a href="#">TileControl のスクロール</a>	24-25
<a href="#">TileControl のナビゲーション</a>	25-26
<a href="#">TileControl タッチスクリーンサポート</a>	26
<a href="#">TileControl のテンプレート</a>	27
<a href="#">TileControl のグループ</a>	28
<a href="#">TileControl のタイルと要素</a>	29-30
<a href="#">画像要素</a>	30-31
<a href="#">パネル要素</a>	31
<a href="#">テキスト要素</a>	31
<a href="#">データ連結の概要</a>	32

<a href="#">プロパティマッピング</a>	32-33
<a href="#">データ連結チュートリアル - 単純なデータ連結 TileControl の作成</a>	33
<a href="#">手順 1: 新しいデータソースへの接続</a>	33-34
<a href="#">手順 2: データ接続とデータアダプタの構成</a>	34
<a href="#">手順 3: C1TileControl への Products テーブルの連結</a>	34-35
<a href="#">手順 4: TileControl のプロパティマッピングの設定</a>	35
<a href="#">データ連結チュートリアル - 複雑なデータ連結 TileControl の作成</a>	35-36
<a href="#">手順 1: C1TileControl のデータソースの作成</a>	36
<a href="#">手順 2: データ接続とデータアダプタの構成</a>	36-37
<a href="#">手順 3: データセットの生成</a>	37-38
<a href="#">手順 4: データスキーマの準備</a>	38-40
<a href="#">手順 5: C1TileControl への Products テーブルの連結</a>	40-41
<a href="#">手順 6: TileControl のプロパティマッピングの設定</a>	41-42
<a href="#">手順 7: C1TileControl のデフォルトテンプレートの変更</a>	42-44
<a href="#">手順 8: FormatValue イベントでのタイルの書式設定</a>	44-46
<a href="#">タスク別ヘルプ</a>	47
<a href="#">C1TileControl へのテンプレートの追加</a>	47
<a href="#">指定したタイルへのテンプレートの割り当て</a>	47-48
<a href="#">テンプレートへの要素の追加</a>	48
<a href="#">C1TileControl のルックアップテーブルの作成</a>	48-49
<a href="#">テンプレートの背景色の変更</a>	49-50
<a href="#">特定のテンプレートの削除</a>	50
<a href="#">タイマーによるテキストビューの変更</a>	50-53
<a href="#">XML ファイルの TileControl の保存／ロード</a>	53
<a href="#">XML ファイルからの TileControl のロード</a>	53-54
<a href="#">XML ファイルとしての TileControl の保存</a>	54-55
<a href="#">TileControl とグループのテキストの設定</a>	55-56
<a href="#">C1TileControl へのグループの追加</a>	56
<a href="#">C1TileControl からのグループの削除</a>	56-57
<a href="#">グループのフォントプロパティの変更</a>	57
<a href="#">TileControl の背景色の設定</a>	57-58
<a href="#">特定グループへのタイルの追加</a>	58-59
<a href="#">タイルの CheckMark の作成</a>	59-60

<a href="#">特定タイルのサイズの拡大</a>	60
<a href="#">タイルへの画像要素の追加</a>	60
<a href="#">タイルへのシンボルの追加</a>	60-61
<a href="#">実行時の画像の描画</a>	61-63

## はじめに

**TileControl for WinForms** は、情報をグラフィカルに表示する直感的かつ魅力的な方法を提供する線形および円形ゲージをサポートします。**TileControl for WinForms** は、きわめて目を引く、プロフェッショナルな、考えられる限りのゲージを作成するために数多くの高度な機能を提供すると共に、きちんと性能を発揮し、役割を果たす単純で実地的なゲージを作成する柔軟性も備えています。

## コンポーネントをプロジェクトに組み込む方法

### コンポーネントの組み込み

Visual Studio では、ツールボックスにコンポーネントを追加しただけでは、プロジェクトにコンポーネントを追加したことにはなりません。プロジェクトの参照設定へ追加された時点でコンポーネントが組み込まれます。

以下のいずれかの操作を行うとプロジェクトへコンポーネントが組み込まれます。

1. フォームにコンポーネントを配置する
2. ソリューションエクスプローラ上で参照の追加を行う

プロジェクトに組み込まれているコンポーネントの一覧は、ソリューションエクスプローラで確認できます。また、各コンポーネントが使用している DLL もソリューションエクスプローラに登録される場合があります。詳細については、Visual Studio の製品ヘルプを参照してください。

本製品で使用しているコンポーネントの一覧を以下に示します。

ファイル	内容
C1.C1TileControl.2.dll	本体アセンブリ
C1.C1TileControl.4.dll	本体アセンブリ(※)
C1.C1TileControl.4.Design.dll	デザイナアセンブリ(※)

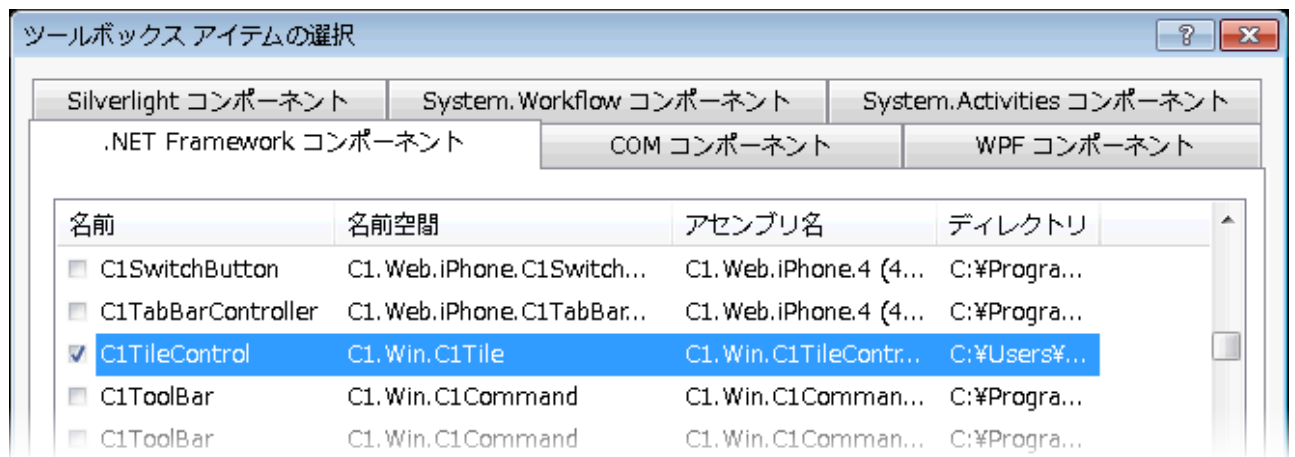
※ .NET Framework 4 以上でご利用いただけます。

### フォームにコンポーネントを配置する方法

以下に、**C1TileControl** コントロールをツールボックスに追加し、フォームに配置する方法を示します。これにより、コンポーネントがプロジェクトに組み込まれます。

#### 配置手順

1. [ツール]メニューから[ツールボックス アイテムの選択]を選択します。
2. [ツールボックス アイテムの選択]ダイアログの「.NET Framework コンポーネント」タブを選択します。
3. 使用するコンポーネントのチェックボックスを ON にして〈OK〉ボタンをクリックしてください。ツールボックスに指定したコントロールのアイコンが表示されます。



 C1TileControl


4. ツールボックスから指定したコントロールのアイコンを選択してフォームに配置します。ソリューションエクスプローラの参照設定に指定したコントロールの名前空間が追加されます。

## コンポーネントのランタイムファイル

TileControl for WinForms のランタイムファイルは、C:\Program Files\ComponentOne\WinForms\bin\ フォルダにインストールされる次のファイルです。

ファイル	内容
C1.C1TileControl.4.dll	本体アセンブリ(※)

※ .NET Framework 4 以上でご利用いただけます。

 **注意:**本製品に含まれているファイルのうち、上記以外のファイルは配布できません。

## 主な特長

役立ちそうな **C1TileControl** の主な機能として、次の事項が挙げられます。

### カスタムのタイルレイアウト

**TileControl** は、グループ内のタイルの自動または手動レイアウトを使用します。タイルは任意のサイズを設定できます。大きいサイズと小さいサイズだけに限定されません。タイル(正確にはタイルテンプレート)の内部レイアウトは非常に柔軟です。ドッキングして積み重ねたパネル、ネストしたパネル、テキスト要素、および画像を使用できます。また、レイアウトを XML ファイルに保存し、いつでも XML ファイルの読み込みできます。

### 2つの表示方向

タイルグループは垂直または水平方向に積み重ねることができます。

### 画像

コントロールの背景画像は、Windows 8 のスタート画面と同様に、タイルと一緒にスクロールできます。ユーザー画像に加えて、タイルに表示できるさまざまなサイズの標準シンボルセットが用意されています。また、「バッジ番号」または「5つ星」格付け画像をテンプレートの一部として簡単に表示できます。これらの要素はタイルのプロパティで指定された整数値にバインドできます。

### テンプレート

各タイルを個別に設計する必要はありません。代わりに、1つまたはいくつかのタイルテンプレートを作成した後、これらのテンプレートをタイルに関連付けることができます。タイルは、テンプレートに文字列、色、および画像などのデータを提供できます。1つのテンプレートを複数のテンプレートに関連付けて、1つのタイルで複数のテンプレートを切り替えることができます。たとえば、タイマーでテキストや画像ビューを切り替えることができます。これを行う方法の例については、[を参照してください](#)。

### タッチサポート

TileControl は、Windows 7 または Windows 8 マシン上のタッチ入力ハードウェアを使用して、(スワイプジェスチャによる)タイルのパン、タップ、およびチェックをサポートします。ユーザーが指先でタイルを押したままにすると、タイルにツールチップが表示されます。

### ナビゲーション

キーボードを使用してタイル間を簡単に移動できます。

### データ連結

C1TileControl は、BindingList や DataSet などの任意の .NET データソースに広く連結できます。C1TileControl を使用すると、コードをほとんどまたはまったく必要とせず、完全なナビゲーション機能を持つデータベースブラウザを短時間で作成できます。詳細については、「[データ連結の概要](#)」を参照してください。

## クイックスタート

### クイックスタート

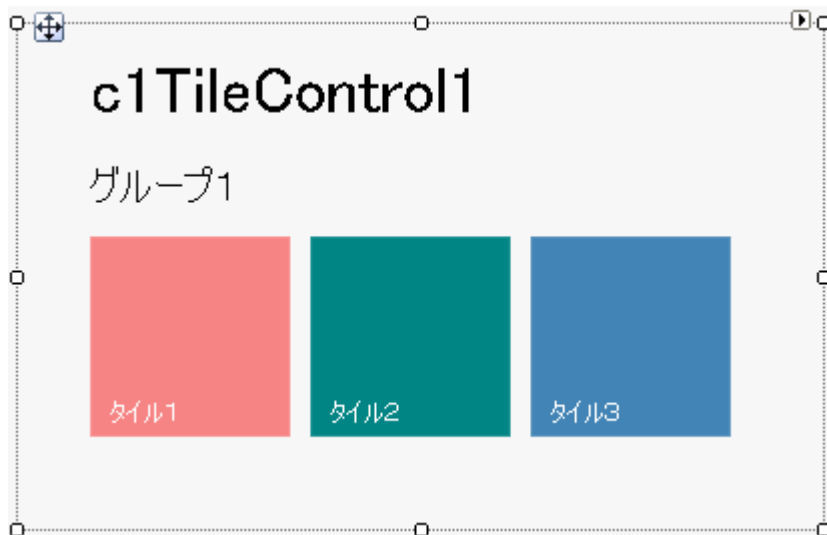
このクイックスタートガイドの目的は、**TileControl for WinForms** に親しんでいただくことです。クイックスタートガイドの最初の手順では、**C1TileControl** を WinForms プロジェクトに追加します。また、このクイックスタートガイドでは、**TileControl for WinForms** によるアプリケーションへの C1Tile コントロールの追加、C1Tile コントロールに表示されるコンテンツの追加、および可能な実行時のインタラクティブ操作のいくつかの観察を行う方法も説明します。

## 手順 1: C1Tile アプリケーションの作成

### 手順 1: C1Tile アプリケーションの作成

この手順では、**TileControl for WinForms** を使用し、.NET プロジェクトを作成します。C1Tile コントロールをアプリケーションに追加すると、コンテンツを表示できるインターフェースが使用可能になります。プロジェクトを設定し、C1Tile コントロールをアプリケーションに追加するには、以下の手順を実行します。

1. まず、新しい Windows フォームアプリケーションを作成します。この例では、アプリケーションに「QuickStart」という名前を付けます。プロジェクトに別の名前を付ける場合は、後の手順で、「QuickStart」への参照を各自のプロジェクトの名前に変更する必要があります。
2. ソリューションエクスプローラで、プロジェクト名を右クリックし、**[参照の追加]**を選択します。「参照」タブを選択し、**C1.Win.C1TileControl.2.dll** を指定します。**[参照の追加]**ダイアログボックスで、C1.Win.C1TileControl.2.dll を選択し、〈OK〉をクリックして、参照をプロジェクトに追加します。
3. デザインビューで、Visual Studio のツールボックスに移動し、**C1TileControl** をダブルクリックしてフォームに追加します。そこがない場合は、ツールボックス内で右クリックし、**[タブの追加]**を選択します。タブに名前(C1TileControl など)を付けます。**C1TileControl** の下で右クリックし、**[アイテムの選択]**を選択します。**[ツールボックスアイテムの選択]**が表示されます。**C1.Win.C1Tile** アセンブリを参照し、〈OK〉をクリックします。
4. **C1TileControl** コントロールが表示されます。



## 手順 2: 要素を含むテンプレートの作成

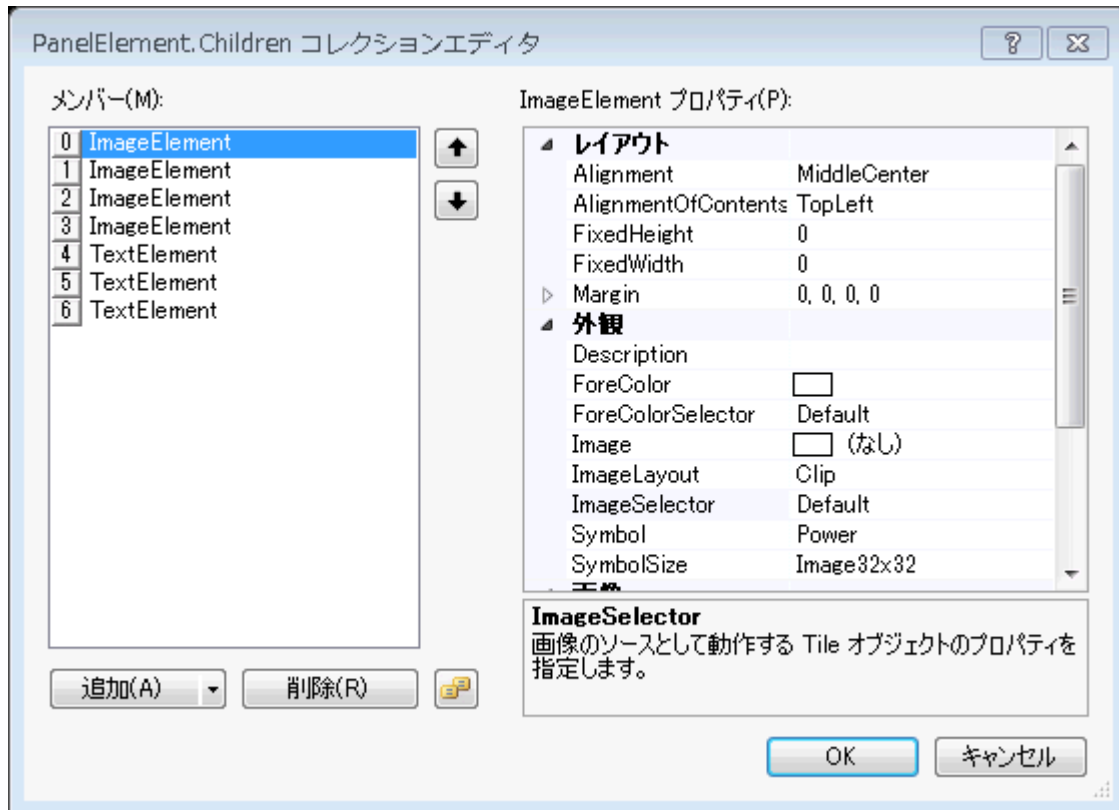
前の手順では、WinForms アプリケーションを作成し、**C1TileControl** をプロジェクトに追加しました。

画像とテキスト要素を含むパネルをタイルに追加するには、以下の手順を実行します。

### 設計時に要素を含むテンプレートをタイルに追加するには



1. デフォルトの**タイル1**を右クリックし、**[テンプレートの編集]**を選択します。**[C1TileControl.Templates コレクションエディタ]**が表示されます。
2. **<追加>**をクリックし、新しいテンプレートを Tile1 に追加します。
3. **Elements** の横にある**<...>**ボタンをクリックし、**[Template.Elements コレクションエディタ]**を開きます。**[追加]**を選択し、そのドロップダウン矢印をクリックして、**PanelElement**、**TextElement** の順に選択します。これにより、要素が **TemplateCollection** に追加されます。
4. **[メンバ]**リストから **PanelElement** を選択し、**ChildSpacing** プロパティの横に**9**を入力します。
5. **Children** の横にある**<...>**ボタンをクリックし、**[PanelElement.Children コレクションエディタ]**を表示します。**[追加]**を選択し、そのドロップダウン矢印をクリックして、**ImageElement** を選択します。4つの **ImageElement** と3つの **TextElement** を追加します。これにより、要素が **TemplateCollection** に追加されます。



6. **[メンバ]**から、1番目の **[0]ImageElement**を選択し、各プロパティを次のように設定します。
  - **ImageSelector** プロパティを **UnboundSymbol** に設定。これにより、選択したシンボルが指定したタイルの画像になります。
  - **Symbol** プロパティを **LeftToRight** に設定。これにより、LeftToRight 画像が指定したタイルに表示されます。
  - **SymbolSize** プロパティを **Image64x64** に設定。これにより、デフォルトのシンボルサイズが **32x32** から **64x64** に変更されます。
  - **FixedHeight** を **70** に設定。これにより、パネルのコンテンツの高さが 70 ピクセルに設定されます。
7. **[メンバ]**リストから、2番目の **[1]ImageElement** を選択し、各プロパティを次のように設定します。
  - **ForeColor** プロパティを **255, 192, 255** に設定。
  - **ForeColorSelector** プロパティを **Unbound** に設定。
  - **ImageSelector** プロパティを **UnboundSymbol** に設定。
  - **Symbol** プロパティを **CircleWithPlus** に設定。
8. **[メンバ]**リストから、3番目の **[2]ImageElement** を選択し、各プロパティを次のように設定します。
  - **ForeColor** プロパティを **255, 192, 255** に設定。
  - **ForeColorSelector** プロパティを **Unbound** に設定。
  - **ImageSelector** プロパティを **UnboundSymbol** に設定。
  - **Symbol** プロパティを **CircleWithMinus** に設定。
  - **Alignment** プロパティを **TopCenter** に設定。これにより、シンボルがパネルの TopCenter に揃えられます。
9. **[メンバ]**リストから、4番目の **[3]TextElement** を選択し、各プロパティを次のように設定します。
  - **ForeColor** プロパティを **255, 224, 192** に設定。
  - **ForeColorSelector** プロパティを **Unbound** に設定。

# TileControl for WinForms

- **ImageSelector** プロパティを **UnboundSymbol** に設定。
  - **Symbol** プロパティを **CircleWithMultiply** に設定。
  - **Alignment** プロパティを **BottomCenter** に設定。
10. [メンバ]リストから、5番目の [4]TextElement を選択し、各プロパティを次のように設定します。
    - **ForeColor** プロパティを **192, 192, 255** に設定。
    - **ForeColorSelector** プロパティを **Unbound** に設定。
    - **Text** プロパティを **Top** に設定。
    - **TextSelector** を **Unbound** に設定。
    - **Alignment** を **TopCenter** に設定。
  11. [メンバ]リストから、6番目の [5]TextElement を選択し、各プロパティを次のように設定します。
    - **BackColorSelector** プロパティを **Unbound** に設定。
    - **ForeColor** プロパティを **255, 224, 192** に設定。
    - **ForeColorSelector** プロパティを **Unbound** に設定。
    - **Text** プロパティを **Bottom** に設定。
    - **TextSelector** プロパティを **Unbound** に設定。これにより、デフォルトのテキストではなく、新しいテキスト「Bottom」が表示されます。
    - **Alignment** プロパティを **BottomCenter** に設定。
    - **DirectionVertical** プロパティを **True** に設定。
  12. [メンバ]リストから、7番目の [6]TextElement を選択し、各プロパティを次のように設定します。
    - **BackColorSelector** プロパティを **Unbound** に設定。
    - **ForeColor** プロパティを **255, 192, 255** に設定。
    - **ForeColorSelector** プロパティを **Unbound** に設定。
    - **Text** プロパティを **Middle** に設定。
    - **TextSelector** プロパティを **Unbound** に設定。
  13. <OK>をクリックし、[PanelElement.Children コレクションエディタ]を保存して閉じます。
  14. [Template.Elements コレクションエディタ]で、[メンバ]リストから [1] TextElement を選択します。
  15. [メンバ]リストから、2番目の [1]TextElement を選択し、各プロパティを次のように設定します。
    - **Alignment** プロパティを **BottomCenter** に設定。
    - **Margin** プロパティを **0, 0, 0, 5** に設定。
  16. <OK>をクリックし、[Template.Elements コレクションエディタ]を保存して閉じます。
  17. <OK>をクリックし、[C1TileControl.Templates コレクションエディタ]を保存して閉じます。

## このトピックの作業結果

作成したテンプレート Template1 がタイトルの **Template** プロパティに適用されていないため、Tile1 は変わらないまま表示されます。

次の手順では、[C1TileControl タスク]メニューを使用し、新しいテンプレートを指定したタイトルに割り当てて、タイトルのいくつかのプロパティを変更する方法を学びます。

## 手順 3: 指定タイトルへのテンプレートの適用

前の手順では、テンプレートを作成し、パネル、画像、およびテキストなどのテンプレート要素を追加しました。また、テンプレートの各要素の配置とレイアウトプロパティも設定しました。この手順では、テンプレートを指定したタイトルに適用し、いくつかのタイトルプロパティ(BackColor、Template、HorizontalSize、および Text プロパティなど)を設定します。

テンプレートを **C1TileControl** の1番目のタイトルに適用し、1番目のタイトルのいくつかのプロパティを変更するには、以下の手順を実行します。

1. **タイトル1**を選択し、[C1TileControl タスク]メニューを開きます。
2. [C1TileControl タスク]メニューで、デフォルトテキスト「Tile1」を削除し、**Template** プロパティを **Template** に設定し、**HorizontalSize** プロパティを**3**に設定します。

C1TileControl タスク	
tile2:	
タイルのテキスト	<input type="text" value="タイル2"/>
ツールチップのテキスト	<input type="text"/>
タイルのグループ	<input type="text" value="group1"/> ▼
タイルのインデックス	<input type="text" value="1"/>
水平方向のサイズ	<input type="text" value="1"/>
垂直方向のサイズ	<input type="text" value="1"/>
IntValue	<input type="text" value="0"/>
<input type="checkbox"/> チェック済み	
テンプレート	<input type="text" value="(既定)"/> ▼
<a href="#">タイルテンプレートの編集</a>	
<a href="#">グループの選択</a>	
<a href="#">C1TileControl の選択</a>	

- 変更した1番目のタイルを右クリックし、[グループの編集]を選択します。[C1TileControl.Groups コレクションエディタ]が表示されます。
- Tiles の横にある<...>ボタンをクリックし、[Group.Tiles コレクションエディタ]を開き、tile1[] を選択します。
- tile1[] のBackColor プロパティを DimGrey に設定します。
- <OK>をクリックし、[Group.Tiles コレクションエディタ]を保存して閉じた後、<OK>をクリックし、[C1TileControl.Groups コレクションエディタ]を保存して閉じます。

Template1 とタイル設定が1番目のタイルに適用されています。



## ✔ このトピックの作業結果

おめでとうございます！ C1TileControl クイックスタートが正常に完了しました。このトピックでは、C1TileControl をウィンドウフォームに追加し、指定のタイルのテンプレートを作成した後、タイルのいくつかのプロパティを設定しました。

## デザイン時のサポート

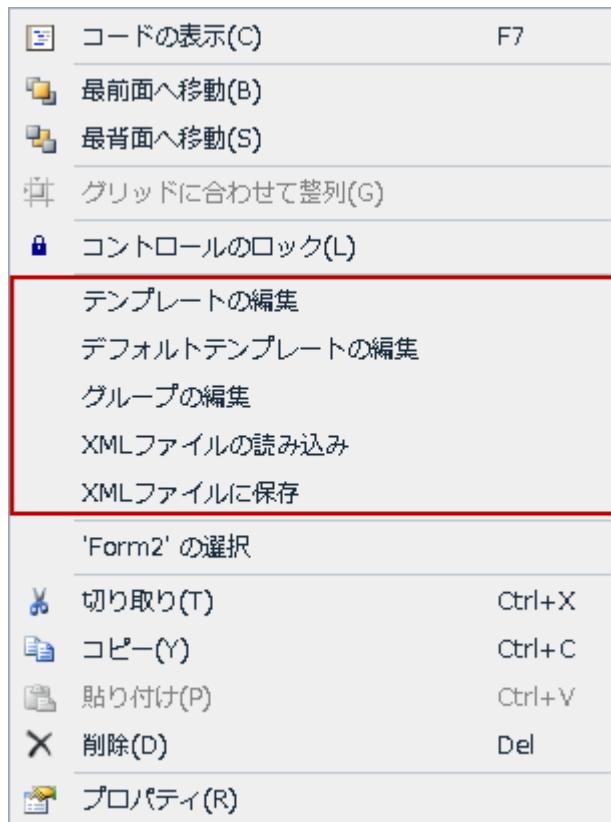
デザイン時のサポート  
デザイン時のサポート **C1TileControl** は、カスタマイズされたコンテキストメニュー、スマートタグ、および充実した設計時サポートを提供するデザイナーを備えており、オブジェクトモデルの操作が簡素化されています。

以下のトピックでは、**C1TileControl** の設計時環境を使用して **C1TileControl** を設定する方法について説明します。

## C1TileControl コンテキストメニュー

**C1TileControl** コントロールは、設計時に使用する追加的な機能のコンテキストメニューを備えています。

C1TileControl のコンテキストメニューにアクセスするには、**C1TileControl** コントロールを右クリックします。コンテキストメニューが次のように表示されます。



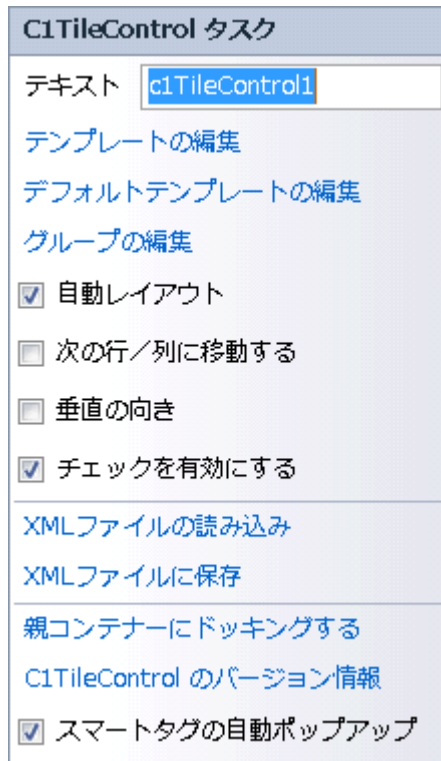
**C1TileControl** のコンテキストメニューは次のように動作します。

- **テンプレートの編集**  
[テンプレートの編集]を選択すると、[C1TileControl.Templates コレクションエディタ]が開き、テンプレートを追加、削除、または変更できます。
- **デフォルトテンプレートの編集**  
[デフォルトテンプレートの編集]項目を選択すると、[Template.Elements コレクションエディタ]が開き、テキスト、画像、およびパネルを C1TileControl のテンプレートに追加できます。
- **グループの編集**  
[グループの編集]項目を選択すると、[C1TileControl.Groups コレクションエディタ]が開き、C1Tile コントロールのグループを追加、削除、または変更できます。
- **XML ファイルの読み込み**  
[XML ファイルの読み込み]を選択すると、[XML ファイルの読み込み]ダイアログボックスが開き、ロードする .xml ファイルを参照できます。
- **Xml ファイルに保存**  
[Xml ファイルに保存]を選択すると、[Xml ファイルに保存]ダイアログボックスが開き、保存する .xml ファイルを参照できます。

## C1TileControl スマートタグ

Visual Studio では、**TileControl for WinForms** の各コンポーネントにスマートタグが備わっています。スマートタグは、各コントロールで最もよく使用されるプロパティを提供するショートカットのタスクメニューを表します。

[**C1TileControl タスク**]メニューにアクセスするには、**C1TileControl** コントロールの右上隅にあるスマートタグ (📌) をクリックします。[**C1TileControl タスク**]メニューが開きます。



[**C1TileControl タスク**]メニューは次のように動作します。

### タイルのチェックを許可

タイルのチェックはデフォルトで有効になっています。チェックボックスの選択を解除すると、タイルのチェックが無効になります。

### タイルの並べ替えを許可

タイルの並べ替えはデフォルトで無効になっています。チェックボックスを選択すると、タイルを並べ替えることができます。

### Text

[**テキスト**]項目の横にあるテキストボックスをクリックすると、TileControl の上部に表示されるテキストが作成されます。

### テンプレートの編集

[**テンプレートの編集**]を選択すると、[**C1TileControl.Templates コレクションエディタ**]が開き、テンプレートを追加、削除、または変更できます。

### [デフォルトテンプレートの編集]

[**デフォルトテンプレートの編集**]項目を選択すると、[**Template.Elements コレクションエディタ**]が開き、テキスト、画像、およびパネルを **C1TileControl** のテンプレートに追加できます。

# TileControl for WinForms

## [グループの編集]

[グループの編集]項目を選択すると、[C1TileControl.Groups コレクションエディタ]が開き、C1Tile コントロールのグループを追加、削除、または変更できます。

## [垂直の向き]

[垂直の向き]チェックボックスを選択すると、C1TileControl が垂直に配置されます。

## [自動レイアウト]

選択すると、自動レイアウトが有効になります。

## [次の行／列に移動する]

選択すると、次の行／列に移動するが有効になります。

## [チェックを有効にする]

選択すると、ON/OFF が有効になります。

## [XML ファイルの読み込み]

[XML ファイルの読み込み]を選択すると、[XML ファイルの読み込み]ダイアログボックスが開き、ロードする .xml ファイルを参照できます。

## [XML ファイルに保存]

[XML ファイルに保存]を選択すると、[XML ファイルに保存]ダイアログボックスが開き、保存する .xml ファイルを参照できます。

## [親コンテナにドッキングする]

[親コンテナにドッキングする]をクリックすると、C1TileControl は親コンテナにドッキングします。

## [C1TileControl のバージョン情報]

[C1TileControl のバージョン情報]をクリックすると、[ComponentOne について]ダイアログボックスが表示されます。このダイアログボックスには、C1TileControl 製品のバージョン番号やライセンス情報が表示されます。

## [スマートタグの自動ポップアップ]

[スマートタグの自動ポップアップ]チェックボックスを選択解除すると、グループまたは各タイル／パネルのクリック時にポップアップスマートタグが無効になります。

## グループタスク

[C1TileControl タスク]メニューにアクセスするには、C1TileControl コントロール内のグループのいずれかをクリックします。[C1TileControl タスク]メニューが開きます。

C1TileControl タスク	
group1:	
グループのテキスト	<input type="text" value="グループ1"/>
グループのインデックス	<input type="text" value="0"/>
タイルの編集	
C1TileControl の選択	

「C1TileControl タスク」メニューは次のように動作します。

#### [グループのテキスト]

[グループのテキスト]項目の横にあるテキストボックスをクリックすると、C1TileControl の**グループ**の上部に表示されるテキストが作成されます。

#### [グループのインデックス]

グループ内のタイルの位置を指定します。

#### [タイルの編集]

[タイルの編集]項目を選択すると、[Group.Tiles コレクションエディタ]が開き、グループ内のタイルを追加、削除、または変更できます。

#### [C1TileControl の選択]

[C1TileControl の選択]項目をクリックすると C1TileControl が選択されます。

## タイルタスク

[C1TileControl タスク]メニューにアクセスするには、C1TileControl コントロール内のタイルのいずれかをクリックします。

[C1TileControl タスク]メニューが開きます。

# TileControl for WinForms

C1TileControl タスク	
tile2:	
タイルのテキスト	<input type="text" value="タイル2"/>
ツールチップのテキスト	<input type="text"/>
タイルのグループ	<input type="text" value="group1"/> ▼
タイルのインデックス	<input type="text" value="1"/>
水平方向のサイズ	<input type="text" value="1"/>
垂直方向のサイズ	<input type="text" value="1"/>
IntValue	<input type="text" value="0"/>
<input type="checkbox"/> チェック済み	
テンプレート	<input type="text" value="(既定)"/> ▼
<a href="#">タイルテンプレートの編集</a>	
<a href="#">グループの選択</a>	
<a href="#">C1TileControl の選択</a>	

[C1TileControl タスク]メニューは次のように動作します。

## [タイルのテキスト]

[タイルのテキスト]項目の横にあるテキストボックスをクリックすると、C1TileControl 内のタイルの上部に表示されるテキストが作成されます。

## [ツールチップのテキスト]

[ツールチップのテキスト]項目の横にあるテキストボックスをクリックすると、C1TileControl 内のタイルのテキストにホバーすると表示されるテキストが作成されます。

## [タイルのグループ]

ドロップダウン矢印をクリックすると、既存のグループをリストするメニューが表示されます。タイルを入れるグループを選択します。

## [タイルのインデックス]

グループ内のタイルの位置を指定します。

## [水平方向のサイズ]

タイルの水平サイズを指定します。

## [垂直方向のサイズ]

タイルの垂直サイズを指定します。

## [IntValue]

タイルの整数値を指定します。



### [チェック済み]

チェックボックスをクリックすると、チェックマークが有効になり、タイルの上に次のようなチェックマークが表示されます。



### [タイルテンプレートの編集]

[**タイルテンプレートの編集**]項目をクリックすると、[**Template.Elements コレクションエディタ**]が開き、テキスト、画像、およびパネルを **Tiles** のテンプレートに追加できます。

### [グループの選択]

[**グループの選択**]項目をクリックすると、タイルが入っているグループが選択されます。

### [C1TileControl の選択]

[**C1TileControl の選択**]項目をクリックすると C1TileControl が選択されます。

## C1TileControl のコレクションエディタ

**C1TileControl** は、次の各コレクションエディタを提供し、これらを使って設計時にプロパティを C1TileControl 要素に適用できます。

- **C1TileControl.Templates** コレクションエディタ
- **Template.Elements** コレクションエディタ
- **C1TileControl.Groups** コレクションエディタ
- **Group.Tiles** コレクションエディタ
- **C1TileControl.PropertyMappings** コレクションエディタ

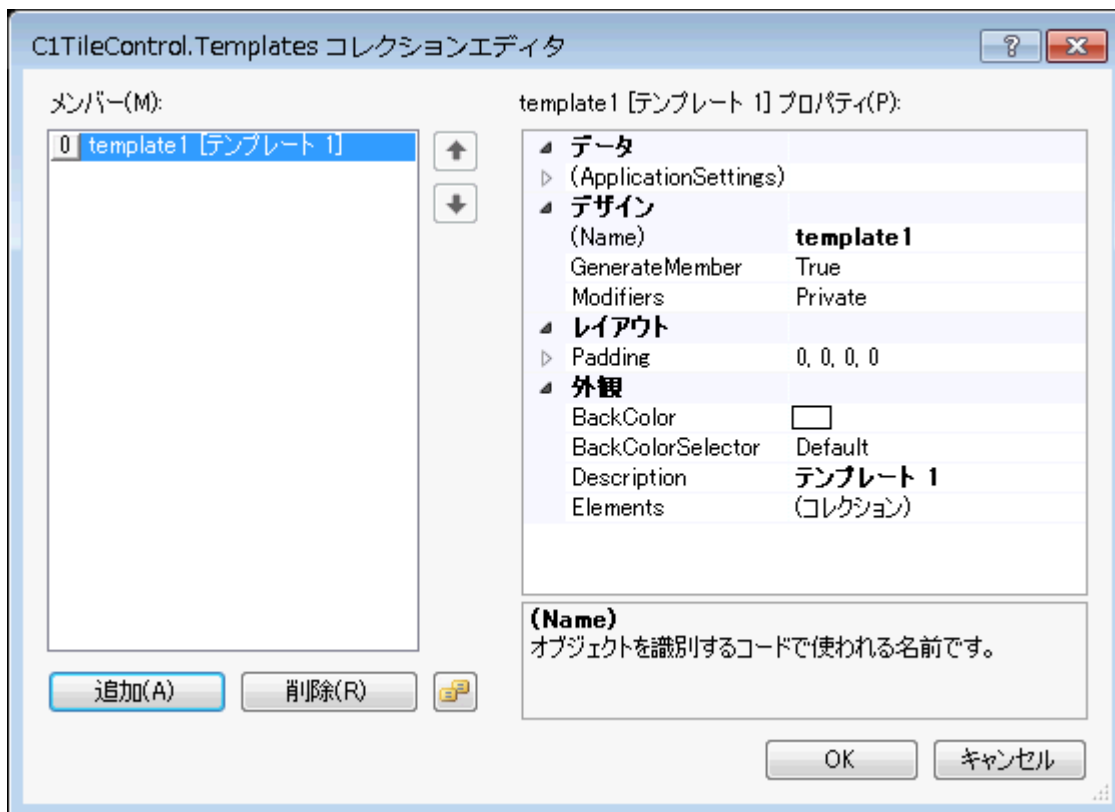
以下のトピックでは、各 C1TileControl コレクションエディタの概要を提供し、それぞれにアクセスする方法を示します。

## C1TileControl.Templates コレクションエディタ

**C1TileControl.Templates** コレクションエディタは、テンプレートを **C1TileControl** に追加するために使用されます。テンプレートは、テキスト、画像、およびパネルなどの要素を保持できます。これらの要素は、設計時に **Template.Elements** コレクションエディタを使用するか、プログラムによって各テンプレートに追加できます。各テキスト、画像、またはパネル要素は、子要素(テキスト、画像、および/またはパネル)を保持できます。たとえば、パネル要素は複数のテキストと画像要素を含むことができます。

### C1TileControl.Templates コレクションエディタにアクセスするには

**C1TileControl** を右クリックし、コンテキストメニューから[**テンプレートの編集**]を選択します。1つのメンバがコレクションに追加されているとき、次のような **C1TileControl.Templates** コレクションエディタが表示されます。

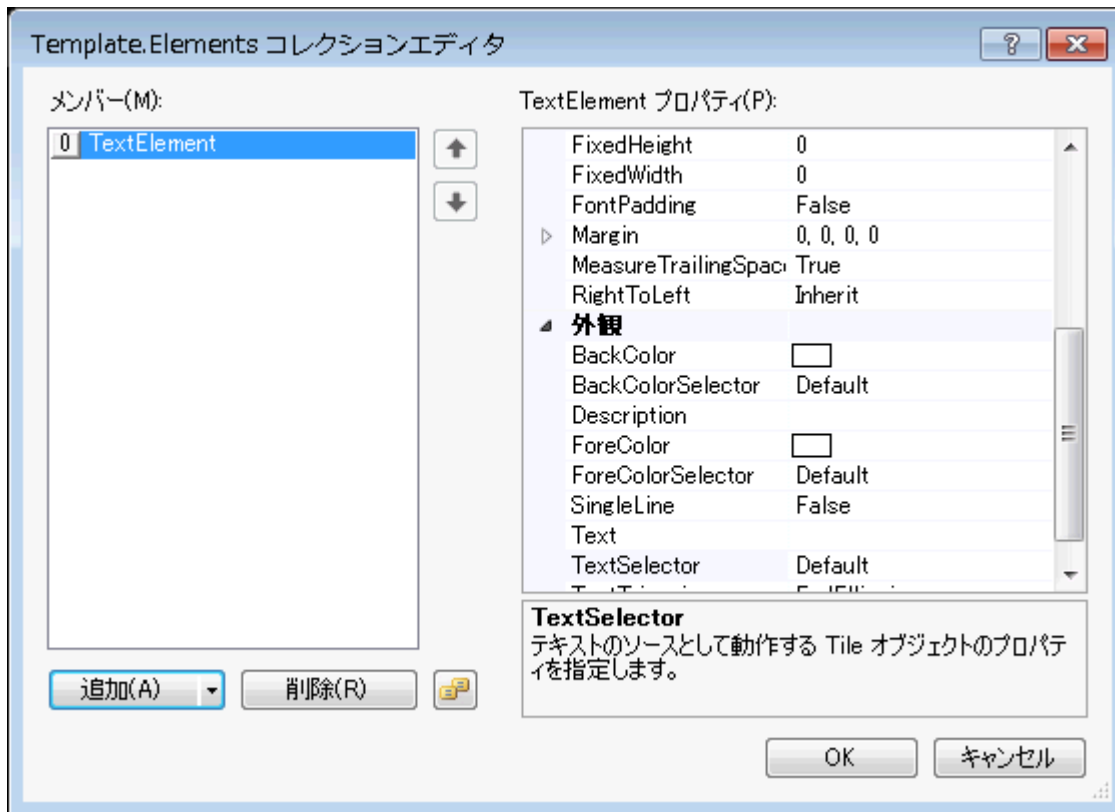


## Template.Elements コレクションエディタ

**Template.Elements コレクションエディタ**は、設計時にパネル要素などを追加、削除、または変更するために使用されます。

**Template.Elements コレクションエディタ**にアクセスするには

**C1TileControl** を右クリックし、コンテキストメニューから[**テンプレートの編集**]を選択します。[**C1TileControl.Templates コレクションエディタ**]が表示されます。<追加>をクリックし、テンプレート項目をコレクションに追加します。**Elements** プロパティにある<...>ボタンをクリックします。[**Template.Elements コレクションエディタ**]が表示されます。ドロップダウンリストボックスをクリックし、TextElement などのメンバを選択し、**TextElement** のプロパティを変更します。

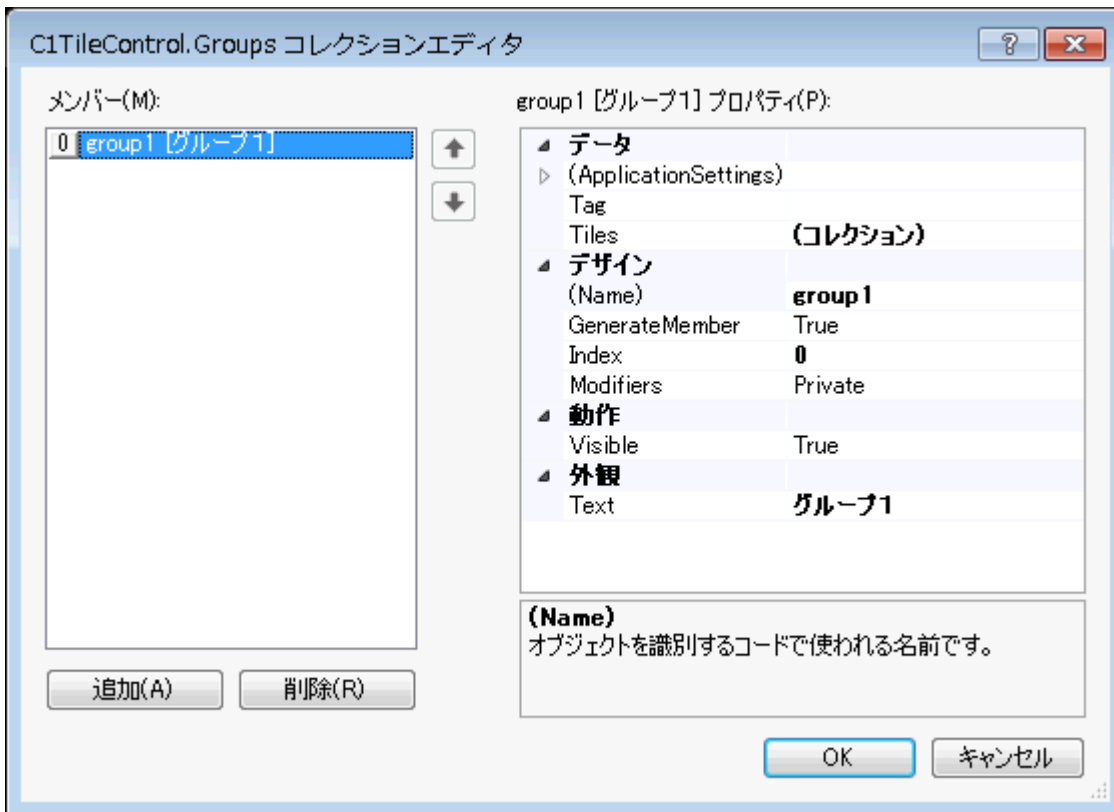


## C1TileControl.Groups コレクションエディタ

**C1TileControl.Groups コレクションエディタ**は、設計時に C1TileControl 内のグループを追加、削除、または変更するために使用されます。

### C1TileControl.Groups コレクションエディタにアクセスするには

**C1TileControl** を右クリックし、コンテキストメニューから[**グループの編集**]を選択します。1つのメンバがコレクションに追加されているとき、次のような **C1TileControl.Groups コレクションエディタ**が表示されます。

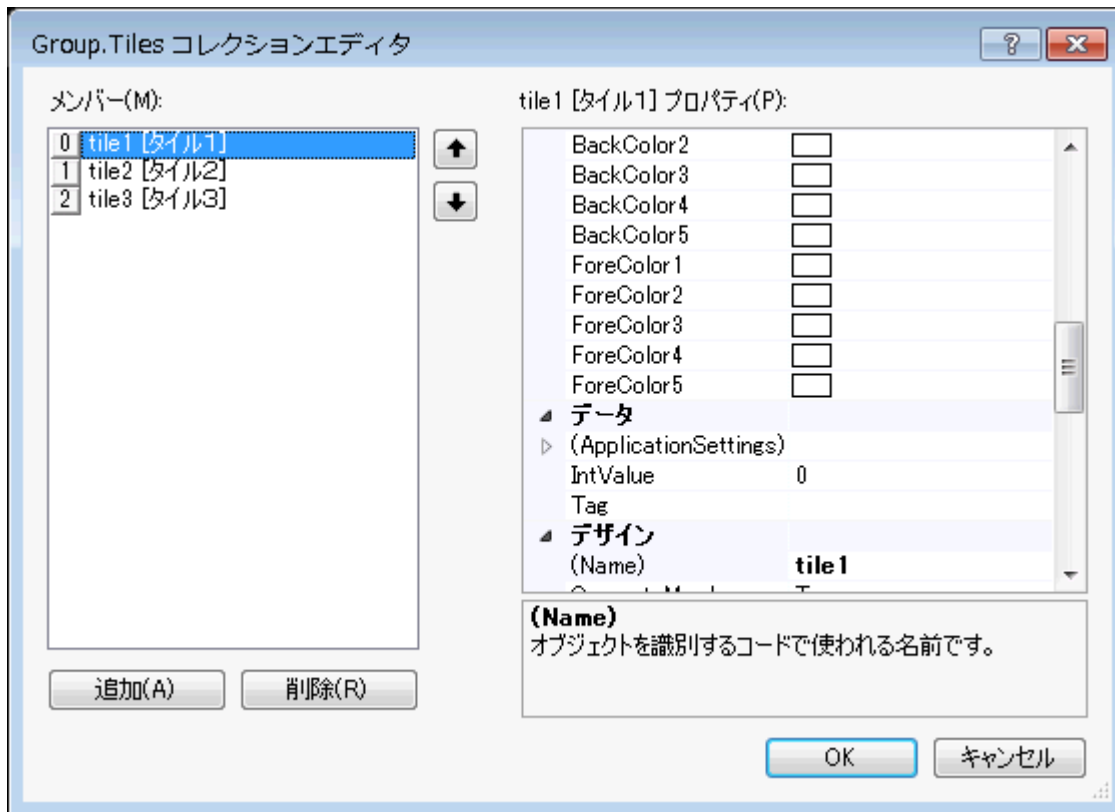


## Group.Tiles コレクションエディタ

**Group.Tiles コレクションエディタ**は、**C1TileControl**のグループ内のタイルを追加、削除、または変更するために使用されます。

### Group.Tiles コレクションエディタにアクセスするには

**C1TileControl** の任意のグループをクリックします。[**C1TileControl-Group** タスク]メニューから[**タイルの編集**]を選択します。次のような **Group.Tiles コレクションエディタ**が表示されます。



## C1TileControl PropertyMapping コレクションエディタ

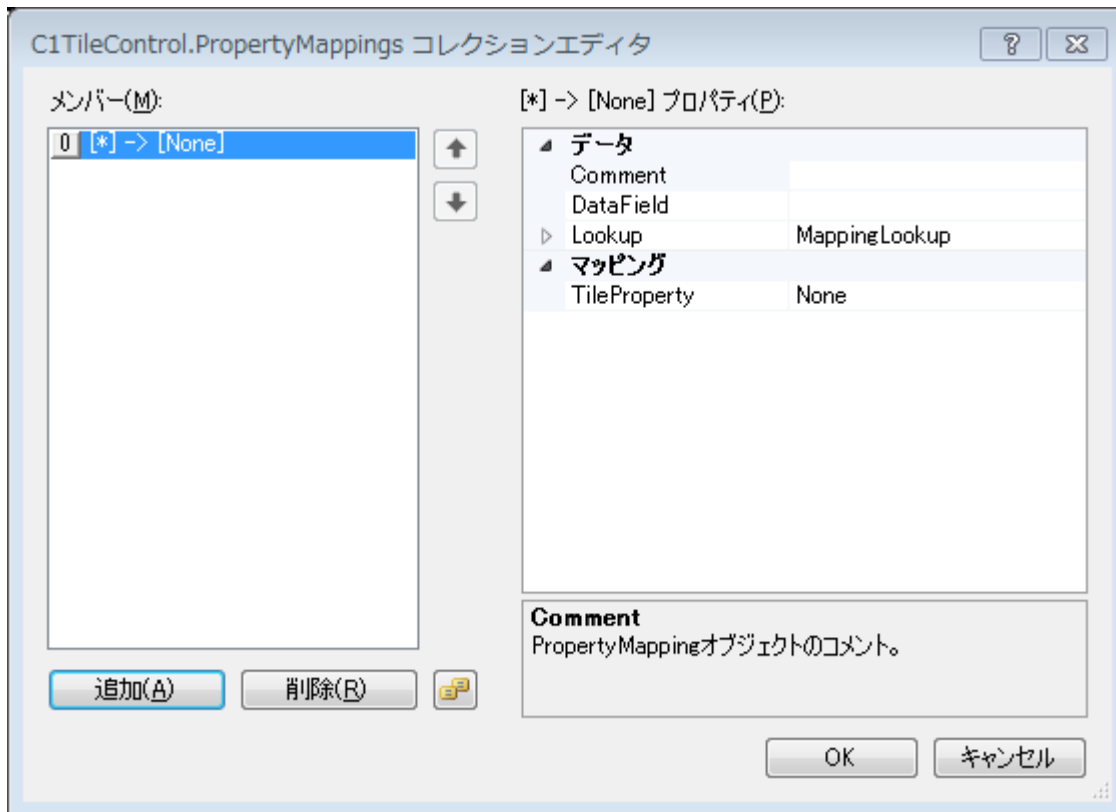
PropertyMapping コレクションエディタは、C1TileControl 内でデータマッピングオブジェクトを追加、削除、変更するために使用されます。

### C1TileControl PropertyMapping コレクションエディタにアクセスする

1. **C1TileControl** を右クリックし、[プロパティ]を選択します。
2. プロパティウィンドウで、**PropertyMappings** の横にある省略符ボタンをクリックします。

**C1TileControl.PropertyMappings** コレクションエディタが表示されます。

# TileControl for WinForms



[メンバ]リストボックスにはメンバが含まれます。メンバを追加するには、[追加]ボタンをクリックします。メンバが追加されると、そのメンバにプロパティを割り当てることができます。

C1TileControl.PropertyMappings コレクションエディタには、次のプロパティが表示されます。

プロパティ	説明
<b>PropertyMapping.Comment</b>	C1.Win.C1Tile.PropertyMapping オブジェクトのコメントを取得または設定します。
<b>PropertyMapping.DataField</b>	タイルプロパティにマップされるデータソースのフィールドを取得または設定します。
<b>MappingLookup.DataSource</b>	ルックアップ対象のデータソースオブジェクトを指定します。
<b>MappingLookup.DisplayMember</b>	意味のある値のソースとして使用するフィールド。
<b>MappingLookup.ValueMember</b>	DataField の外部キーと一致する値(主キー)のソースとして使用するフィールド。
<b>PropertyMapping.TileProperty</b>	マッピングの対象である Tile オブジェクトのプロパティを指定します。

## TileControl レイアウト

**TileControl** は、グループ内のタイルの自動または手動レイアウトを使用します。**AutomaticLayout** プロパティは、タイルを自動または手動で配列するかどうかを決定します。手動レイアウトを使用すると、タイルをフォーム上の任意の場所にドラッグ & ドロップできます。タイルは任意のサイズを設定できます。大きいサイズと小さいサイズだけに限定されません。タイル(正確にはタイルテンプレート)の内部レイアウトは非常に柔軟です。ドッキングして積み重ねたパネル、ネストしたパネル、テキスト要素、および画像を使用できます。また、レイアウトを XML ファイルに保存し、いつでも XML ファイルの読み込みできます。

**TileControl** のタイルのレイアウトはレイアウトプロパティから完全にカスタマイズできます。デフォルトでは、タイルは各グループ内で水平に配列されますが、**Orientation** プロパティから垂直に変更できます。水平レイアウトモードを使用すると、Tile コントロールは Windows 8 UI のような表示になります。垂直レイアウトを使用すると、Tile コントロールは高度なリストボックスのような表示になります。

下表は、C1TileControl のすべてのタイルに共通な表面/レイアウトのプロパティです。

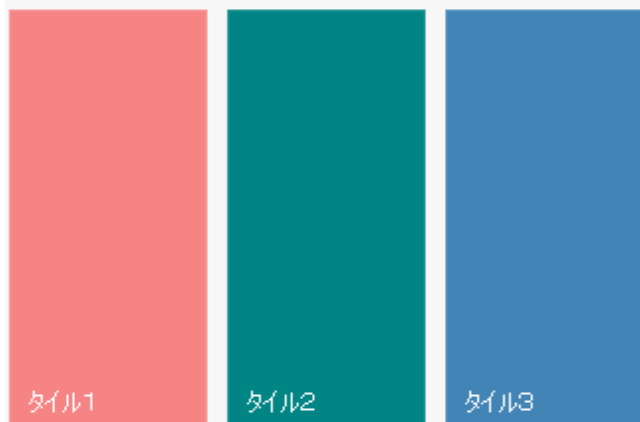
プロパティ	説明
<b>AutomaticLayout</b>	タイルを自動または手動で配列するかどうかを示します。
<b>CellHeight</b>	1つのタイルセルの高さを指定します。
<b>CellSpacing</b>	グループ内のタイルセル間の間隔を指定します。
<b>CellWidth</b>	1つのタイルセルの幅を指定します。
<b>MaximumRowsOrColumns</b>	自動レイアウトモードのセルの行または列の最大数を指定します。
<b>Orientation</b>	タイルグループの配列方法を指定します。
<b>ScrollBarStyle</b>	デフォルトまたはシステムスクロールバーを表示するかどうかを指定します。
<b>ScrollOffset</b>	スクロール可能領域の負またはゼロオフセットを指定します。
<b>SurfaceContentAlignment</b>	スクロール可能表面上のグループの配置を指定します。
<b>SurfacePadding</b>	スクロール可能表面の内部スペース。

設定を先行の Surface プロパティに適用すると、追加された任意の新しいタイルにもそれらの同じ設定が適用されます。タイルに異なるレイアウトを設定する場合は、独自のテンプレートをタイルに適用する必要があります。

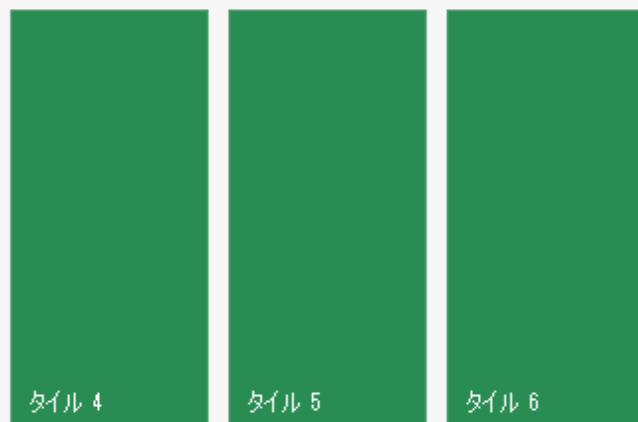
新しいグループが追加されると、タイルはデフォルトの Orientation 設定によって水平に表示されます。下図は、タイルの Horizontal 表示方向を例示します。

## c1TileControl1

グループ1



グループ 2





## TileControl の動作

以下のセクションでは、TileControl 内のタイルの動作の制御に使用される動作プロパティを詳しく説明します。

## TileControl のスクロール

デフォルトでは TileControl のタイルはスクロール可能です。

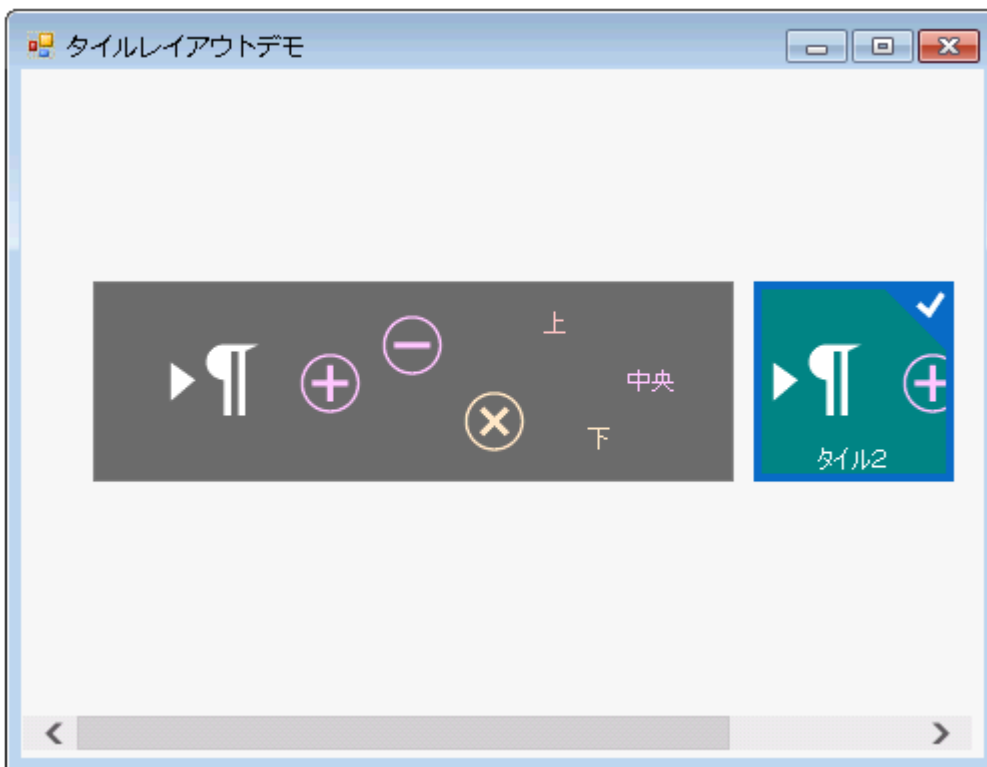
C1TileControl のスクロールバーの外観は、**ScrollBarStyle** ScrollBarStyle プロパティで決定できます。

下表は、ScrollBarStyle プロパティを設定するときいずれかを選択する2つの異なるスクロールバースタイルを表します

値	説明
デフォルト	C1TileControl にデフォルトスクロールバーを指定します。
System	C1TileControl にシステムスクロールバーを指定します。

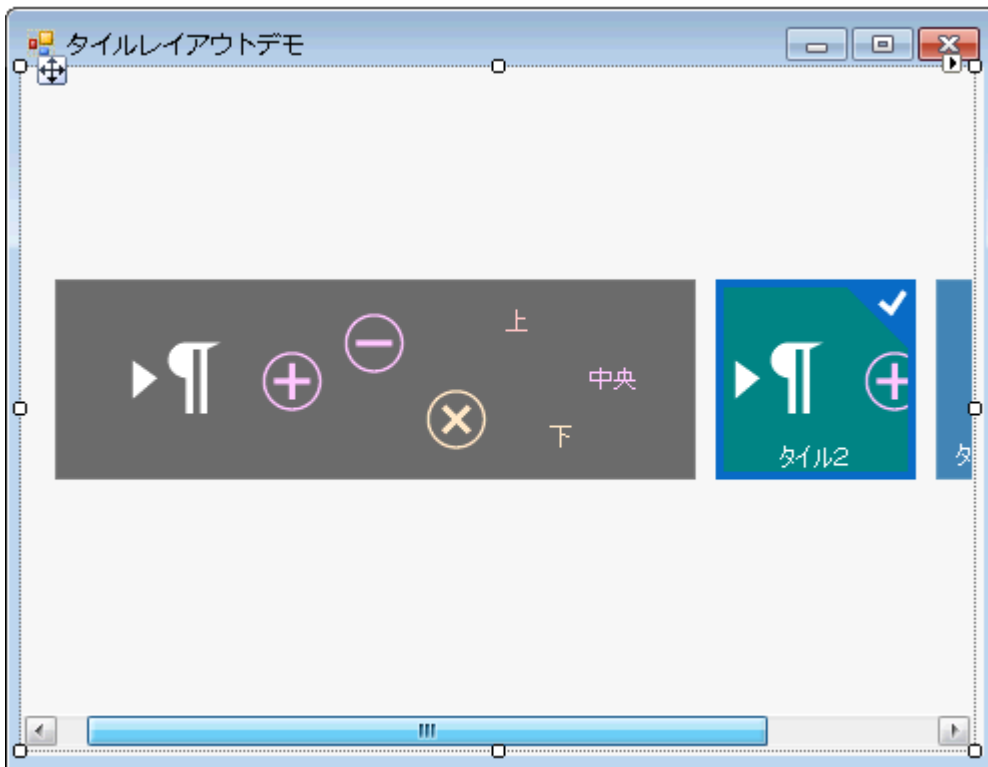
### デフォルトスクロールバー

デフォルトスクロールバー上にホバーすると、バーの色が変わります。デフォルトスクロールバーは次のように表示されます。



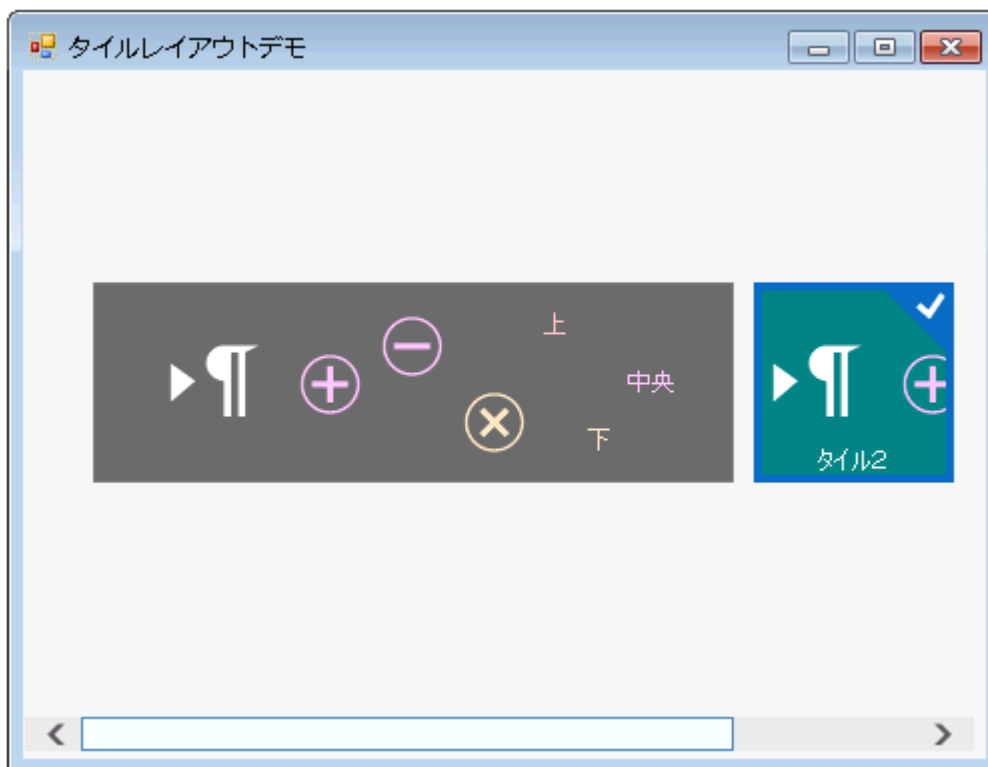
### システムスクロールバー

システムスクロールバー上にホバーすると、バーの色が変わります。システムスクロールバーは次のように表示されます。



スクロールバーのつまみの境界線と内部の色は、それぞれ **SBThumbBorderColor** と **SBThumbInnerColor** プロパティで指定できます。スクロール可能領域の負またはゼロオフセットは、**ScrollOffset**プロパティを使用して決定されます。

下図は、SBThumbBorderColor と SBThumbInnerColor プロパティの効果を示します。SBThumbBorderColor は **SteelBlue** に、および SBThumbInnerColor は **AliceBlue** に設定されています。



## TileControl のナビゲーション

**PassthroughNavigation** プロパティは、垂直および／または水平レイアウトで現在の行／列の末尾タイルにフォーカスを移

動した後、キーボードが次の行／列に移動するかどうかを取得または設定します。

## TileControl タッチスクリーンサポート

TileControl は、Windows 7 または Windows 8 マシン上のタッチ入力ハードウェアを使用して、スワイプジェスチャによるタイルのパン、タップ、および ON/OFF をサポートします。C1TileControl のパン可能領域の最後に到達すると視覚的な合図が表示されます。視覚的な合図を無効にするには、**AllowPanningFeedback** を **False** に設定します。

**AllowChecking** プロパティが有効の場合、スワイプジェスチャを使用するか、マウスを右クリックし、タイルを ON/OFF できます。

**ShowTooltips** プロパティはデフォルトで有効になり、指先でタッチスクリーン上のタイルを押したままにすると、ツールチップが表示されます。**ToolTipInitialDelay** プロパティを使用し、ツールチップが表示されるまでに経過する時間(ミリ秒)を指定できます。

## TileControl のテンプレート

タイルとテンプレートは、**C1TileControl** の最も重要な構成要素です。タイルはデータを提供し、テンプレートは視覚化パターンを提供します。同じタイルで複数のテンプレートを切り替えることができます。たとえば、1番目のテンプレートはタイル画像を表示し、2番目は詳細なタイルテキストを表示できます。また、同じテンプレートを複数のタイルに適用することもできます。

テンプレートは、パネル、テキスト、および画像の3つの可能な要素で構成できます。**PanelElement** クラス型のパネル要素は、ネストしたパネルを含む子要素を格納できます。子要素は、設計時にデザイナを使用するか、プログラムで **Children** プロパティから追加できます。**TextElement** 型のテキスト要素は、設計時にデザイナを使用するか、プログラムで **TextElement** クラスから追加できます。**ImageElement** 型の画像要素は、設計時にデザイナを使用するか、プログラムで **ImageElement** クラスから追加できます。多くのレイアウト設定が用意されており、テンプレート要素を自由自在にレイアウトできます。



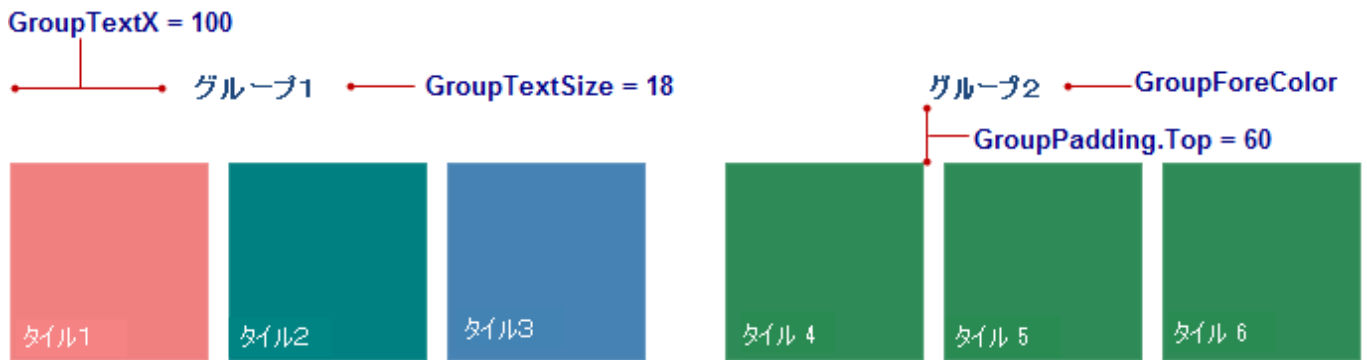
## TileControl のグループ

**TileControl** のグループは、**Group** クラス型に属します。TileControl は1つ以上のグループを持つことができます。各グループは、**Tile** クラス型の1つ以上のタイルを持つことができます。各グループは、グループの名前を表すキャプションを含むことができます。グループのキャプションは **Text** プロパティで指定されます。グループキャプションのフォント、前景色、パディング、テキストサイズ、および位置を変更できます。グループのキャプションを太字に設定することもできます。

グループは、各項目を **Orientation** プロパティに基づいて配列します。グループ間のスペースは、**GroupSpacing** プロパティから指定できるため、さまざまなグループを他のグループから分離して表示できます。

設計時に **C1TileControl.Groups コレクションエディタ** を使用するか、プログラムで **Groups** を使用し、グループを **C1TileControl** に追加できます。グループをデザイナーに追加すると、空のグループが C1TileControl 上に表示されます。タイルをグループに追加する必要があります。Tiles プロパティの横にある(...)ボタンをクリックして、タイルを追加できます。タイルが追加されると、デフォルトのグループキャプションに表示されます。グループを C1TileControl に追加する方法を表示するには、「[C1TileControl へのグループの追加](#)」を参照してください。

下図は、グループの外観プロパティの効果を例示しています。



下表は、C1TileControl のグループの外観とレイアウトプロパティです。

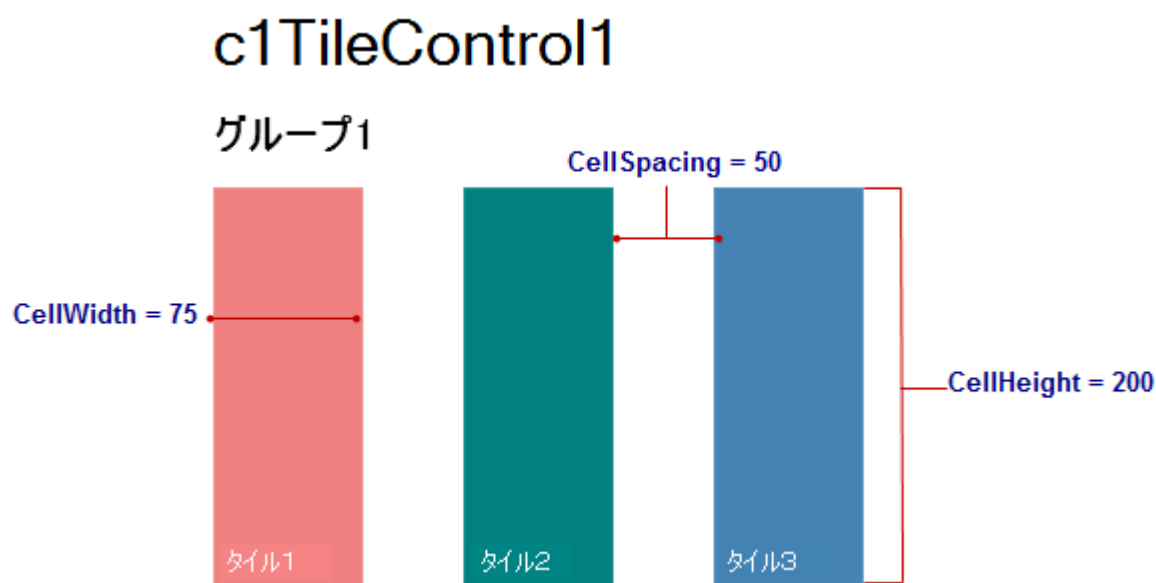
プロパティ	説明
<b>GroupFont</b>	グループキャプションのフォントを指定します。
<b>GroupForeColor</b>	グループキャプションの前景色を指定します。
<b>GroupPadding</b>	タイルグループの内部スペース。
<b>GroupTextBold</b>	グループキャプションのフォントが太字かどうかを示します。このプロパティは GroupFont プロパティよりも優先されます。
<b>GroupTextSize</b>	グループキャプションのフォントサイズを指定します。このプロパティは GroupFont プロパティよりも優先されます。デフォルトのテキストサイズは 15 ピクセルです。
<b>GroupTextX</b>	グループキャプションの水平オフセットを指定します。デフォルトの水平オフセットサイズは 20 ピクセルです。
<b>GroupTextY</b>	グループキャプションの垂直オフセットを指定します。デフォルトの垂直オフセットサイズは 5 ピクセルです。

## TileControl のタイルと要素

**Tile** クラスは、画像 (ImageElement)、テキスト (TextElement)、およびパネル (PanelElement) などの複数の種類の要素を保持できる1つのタイルを表します。タイルは、TileControl の最も重要な構成要素の1つで、データに関与します。TileControl の最も単純なフォームのタイルは、ボタンコントロールのような表示になります。画像、テキスト、およびパネルを各プロパティから簡単に書式設定できます。

設計時に **Group.Tiles コレクションエディタ** を使用するか、プログラムで **Tiles** を使用し、タイルを **C1TileControl** に追加できます。タイルをデザイナーに追加すると、C1TileControl の指定したグループに追加されます。デフォルトで1つのグループが表示されるため、グループへのタイルの追加を簡単に開始できます。設計時に **Tiles** プロパティの横にある〈...〉ボタンをクリックした後、〈追加〉ボタンをクリックし、タイルを追加できます。C1TileControl でタイルを特定グループに追加する方法を表示するには、「[特定グループへのタイルの追加](#)」を参照してください。

下図は、タイルのいくつかの外観プロパティの効果を例示しています。



下表は、C1TileControl のグループの外観とレイアウトプロパティです。

プロパティ	説明
<b>BackColor</b>	タイルの背景色を取得または設定します。
<b>BackColor1</b>	タイルの第1追加背景色を取得または設定します。
<b>BackColor2</b>	タイルの第2追加背景色を取得または設定します。
<b>BackColor3</b>	タイルの第3追加背景色を取得または設定します。
<b>BackColor4</b>	タイルの第4追加背景色を取得または設定します。
<b>BackColor5</b>	タイルの第5追加背景色を取得または設定します。
<b>Checked</b>	タイルが ON かどうかを取得または設定します。
<b>ForeColor</b>	タイルの前景色を取得または設定します。
<b>ForeColor1</b>	タイルの第1追加前景色を取得または設定します。
<b>ForeColor2</b>	タイルの第2追加前景色を取得または設定します。
<b>ForeColor3</b>	タイルの第3追加前景色を取得または設定します。

<b>ForeColor4</b>	タイルの第4追加前景色を取得または設定します。
<b>ForeColor5</b>	タイルの第5追加前景色を取得または設定します。
<b>Height</b>	タイルの高さをピクセル単位で取得または設定します。
<b>HorizontalSize</b>	タイルの幅をピクセル単位で取得または設定します。
<b>Image</b>	タイルに表示される画像を取得または設定します。
<b>Image1</b>	タイルに表示される第1追加画像を取得または設定します。
<b>Image2</b>	タイルに表示される第2追加画像を取得または設定します。
<b>Image3</b>	タイルに表示される第3追加画像を取得または設定します。
<b>Image4</b>	タイルに表示される第4追加画像を取得または設定します。
<b>Image5</b>	タイルに表示される第5追加画像を取得または設定します。
<b>Symbol</b>	タイルに関連付けられたシンボルを取得または設定します。
<b>Template</b>	タイルテンプレートを取得または設定します。
<b>Text</b>	タイル上のテキストを取得または設定します。
<b>Text1</b>	タイルの第1追加テキスト文字列を取得または設定します。
<b>Text2</b>	タイルの第2追加テキスト文字列を取得または設定します。
<b>Text3</b>	タイルの第3追加テキスト文字列を取得または設定します。
<b>Text4</b>	タイルの第4追加テキスト文字列を取得または設定します。
<b>Text5</b>	タイルの第5追加テキスト文字列を取得または設定します。
<b>Text6</b>	タイルの第6追加テキスト文字列を取得または設定します。
<b>Text7</b>	タイルの第7追加テキスト文字列を取得または設定します。
<b>Text8</b>	タイルの第8追加テキスト文字列を取得または設定します。
<b>Text9</b>	タイルの第9追加テキスト文字列を取得または設定します。
<b>ToolTipText</b>	タイルのツールチップテキストを取得または設定します。
<b>VerticalSize</b>	タイルの高さをピクセル単位で取得または設定します。
<b>Width</b>	タイルの幅をピクセル単位で取得します

TileControl のタイルは、次の各要素を含むことができます。

[画像]- 画像は **ImageElement** クラスで表されます。

[パネル]- パネルは **PanelElement** クラスで表されます。

[テキスト]- テキストは **TextElement** クラスで表されます。

## 画像要素

タイルは1つまたは複数の画像を表示できます。its **Image**、**ImageKey**、または **Symbol** プロパティを使用し、タイルの画像を指定できます。また、画像はテンプレートの **ImageElement** の一部として指定するか、**C1TileControl** に属する **CommonImage** オブジェクトの1つに格納できます。

画像を処理する場合にいくつかの方法があります。たとえば、幅が N 個の画像、および高さが M 個の画像で構成される1つの大きい画像を作成できます。その場合、**ImageColumns** プロパティを N に設定し、**ImageRows** プロパティを M に設定する必要があります。また、**ColumnIndex** と **RowIndex** プロパティを使用し、大きな行列から小さい画像を選択することも

# TileControl for WinForms

できます。

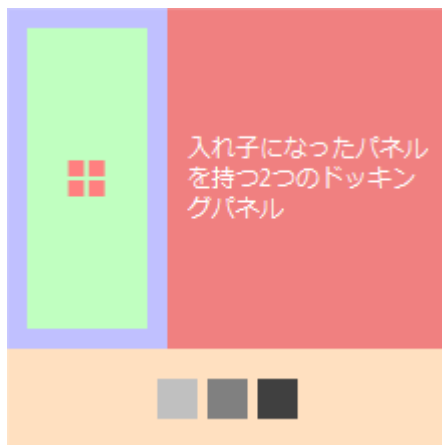
**ImageIndex** プロパティを使用するには、まず **ImageList** コレクションを指定する必要があります。

詳細については、「[タイルへの画像要素の追加](#)」を参照してください。

## パネル要素

タイルは1つまたは複数のパネルを表示できます。パネルはネストしたパネルを含むことができます。**PanelElement** クラス型のパネル要素は、ネストしたパネルを含む子要素を格納できます。子要素は、設計時に **PanelElement.Children** コレクションエディタを使用するか、プログラムで **Children** プロパティから追加できます。

下図では、ネストしたパネルを含む2つのドッキングしたパネルがタイルに表示されています。



## テキスト要素

タイルは1つ以上のテキスト要素を表示できます。テキスト要素は **TextElement** クラスで表されます。子要素は、設計時にデザイナを使用するか、プログラムで **Text**、**Text1** ~ **Text6** プロパティから追加できます。

下図では、いくつかのテキスト要素を含む1つのタイルと、IntValue がゼロより大きい場合に1つのバッジが表示されます。





## データ連結の概要

C1TileControl は、任意の .NET データソースに広く連結できます。C1TileControl を使用すると、コードをほとんどまたはまったく必要とせず、完全なナビゲーション機能を持つデータベースブラウザを短時間で作成できます。

**TileControl for WinForms** は、DataTable、DataView、DataSet などの ADO.NET オブジェクトへの複雑なデータ連結を完全にサポートします。

**C1TileControl** を ADO.NET または **ComponentOne DataObjects for WinForms** データソースに関連付けるには、**TileControl の DataSource** プロパティを同じフォームにある DataSet に設定します。DataSet に複数のテーブルが含まれる場合は、DataMember プロパティコンボボックスでテーブル名を選択します。


**DataSource** プロパティと **DataMember** プロパティは、コードでも Visual Studio のプロパティウィンドウでも設定できます。これだけで、**TileControl for WinForms** はアプリケーション内のデータベースまたは DataTable を完全に認識します。

このようなリンクを一度設定すれば、**TileControl for WinForms** と DataSet は、自分側の操作をすべて自動的に通知し、相手側の操作に応答します。このように、アプリケーションの開発は容易です。

C1TileControl の Tile プロパティをマップするには、設計時に **C1TileControl.PropertyMappings コレクションエディタ** を使用するか、プログラムで **PropertyMapping** クラスを使用します。

プロパティマッピングを使用する方法の例については、「[手順 6: TileControl のプロパティマッピングの設定](#)」を参照してください。

プロパティマッピングの概念については、「[プロパティマッピング](#)」を参照してください。

 **注意:** C1TileControl をデータソースに連結する例については、「[データ連結サンプル](#)」を参照してください。このサンプルは **PropertyMappings** を使用してタイルをデータに連結する方法、および DataGridView オブジェクトを使用して C1TileControl をデータに連結する方法を示します。

## プロパティマッピング

**C1.Win.C1Tile.Tile** コンポーネントは PropertyMapping クラスを使用して、データソースの項目のプロパティと、それに対応する **C1.Win.C1Tile.Tile** コンポーネントのプロパティを関連付けます。

**PropertyMapping** クラスには、以下のメンバが含まれています。

プロパティ	説明
<b>Comment</b>	PropertyMapping オブジェクトのコメントを指定します。
<b>DataField</b>	タイルプロパティにマップされるデータソースのフィールド。
<b>DataSource</b>	グループ内のタイルセル間のギャップを指定します。
<b>Lookup</b>	ルックアッププロパティをカプセル化します。
<b>TileControl</b>	C1.Win.C1Tile.C1TileControl の所有者を取得します。
<b>TileProperty</b>	マッピングの対象である C1.Win.C1Tile.Tile オブジェクトのプロパティを取得または設定します。

**TileProperty** は、TileProperty 列挙値のいずれかに設定できます。

C1TileControl は、あるテーブルの外部キーフィールドの値に基づいて別のテーブルの情報を表示するために、**MappingLookup** クラスを使用します。

たとえば、売上データベースに **Products** というテーブルがあるとします。**Products** テーブルの各レコードには、その製品が属するカテゴリを示す **CategoryID** があります。**CategoryID** は、Categories テーブルのカテゴリレコードを指す外部キーです。(Products テーブルから) 製品のリストを表示するときに、**ProductsID** ではなく、各製品の実際のカテゴリ名を表示することを考えます。カテゴリ名はカテゴリテーブルにあるため、Products テーブルのデータを表示する際には、ルックアップテーブルとなる **MappingLookup** クラスを作成する必要があります。このクラスは、Products レコードの CategoryID 値を受け取り、その値を使用してリレーションシップをたどり、わかりやすいカテゴリ名を返します。この概念をルックアップテーブルと呼び

# TileControl for WinForms

まず、ルックアップテーブルを作成する例については、「[手順 6: TileControl のプロパティマッピングの設定](#)」を参照してください。

MappingLookup クラスには、以下のメンバが含まれています。

プロパティ	説明
<b>DataSource</b>	ルックアップ対象のデータソースオブジェクトを取得または設定します。
<b>DisplayMember</b>	タイルプロパティにマップされるデータソースのフィールド。
<b>ValueMember</b>	C1.Win.C1Tile.PropertyMapping.DataField の外部キーと一致する値(主キー)のソースとして使用するフィールドを取得または設定します。

## データ連結チュートリアル - 単純なデータ連結 TileControl の作成

このセクションでは、タイルコントロールをデータベースに連結する手順を説明します。**C1TileControl** を製品データ項目のコレクションに連結して、各タイルで製品名を作成します。

プロジェクトを実行すると、下のスクリーンショットのような **C1TileControl** が表示されます。



### 手順 1: 新しいデータソースへの接続

この手順では、TileControl のデータ連結プロパティを使用して、後で C1TileControl を連結するためのデータソースを作成します。

1. フォームに **C1TileControl** を追加します。
2. **データ**メニューから、**[新しいデータソースの追加]**を選択します。**[データソース構成ウィザード]**ダイアログボックスが表示されます。
3. **[データベース]**を選択し、**[次へ]**をクリックします。
4. **[データセット]**を選択し、**[次へ]**をクリックします。
5. **[新しい接続]**をクリックします。
6. **[接続の追加]**ダイアログボックスで、**[参照]**をクリックします。
7. **[Microsoft Access データベースファイルの選択]**ダイアログボックスで、**Nwind.mdb**(デフォルトの場所は

C:\Users\UserName\Documents\ComponentOne Samples\Common)を見つけ、[開く]をクリックし、[OK]をクリックします。

8. [次へ] ボタンをクリックして続行します。データファイルをプロジェクトに追加し、接続文字列を修正するかどうかを確認するダイアログボックスが表示されます。プロジェクトにデータベースをコピーする必要はないため、[いいえ]をクリックします。
9. [次の名前で接続を保存する] チェックボックスがオンになっていることを確認し、[次へ] をクリックして続行します。
10. 接続文字列が `NwindConnectionString` という名前で保存されます。
11. データソース構成ウィザードで、[テーブル] ノードを選択します。
12. [完了] を選択します。

## 手順 2: データ接続とデータアダプタの構成

この手順では、products テーブルのデータアダプタを追加します。

1. ツールボックスで、**OleDbDataAdapter** コンポーネントをダブルクリックします。このコンポーネントがツールボックスにない場合は、右クリックして[アイテムの選択]を選択します。ダイアログボックスの[.NET Framework コンポーネント] タブで、**OleDbDataAdapter** を選択します。  
コンポーネントトレイに **OleDbDataAdapter** が表示され、データアダプタ構成ウィザードが表示されます。
2. データアダプタ構成ウィザードで、データアダプタとして使用する接続をドロップダウンリストボックスから選択し(この場合は `C:\Users\UserName\Documents\ComponentOne Samples\Common\C1Nwind.mdb`)、[次へ] をクリックします。
3. [SQL ステートメントを使用する] がデフォルトで選択されています。[次へ] をクリックします。
4. データアダプタ構成ウィザードのテキストボックスに次の SQL ステートメントをコピーして貼り付けます。

```
SELECT    Products.*
FROM      Products
```

5. [次へ] をクリックします。
6. [完了] をクリックします。

## 手順 3: C1TileControl への Products テーブルの連結

この手順では、products データテーブルを TileControl に連結し、データセットを設定します。

1. **C1TileControl** を選択し、C1TileControl のプロパティウィンドウで Data ノードを展開します。
2. DataSource プロパティの横にあるドロップダウン矢印をクリックし、[その他のデータソース]の[Form1 リストインスタンス]を展開し、c1NwindDataSet を選択します。
3. **DataMember** を Products に設定します。

この2つのプロパティを設定すると、**c1NwindDataSet** の products データテーブルが **TileControl** に連結されます。

TileControl は c1NwindDataSet に連結されますが、自動的にデータが挿入されません。データセットにデータを設定するには、フォームをダブルクリックし、次のように Form1\_Load でデータアダプタメソッドを呼び出します。

### Visual Basic コードの書き方

```
oleDataAdapter1.Fill(c1NwindDataSet)
```

### C# コードの書き方

```
oleDbDataAdapter1.Fill(c1NwindDataSet);
```

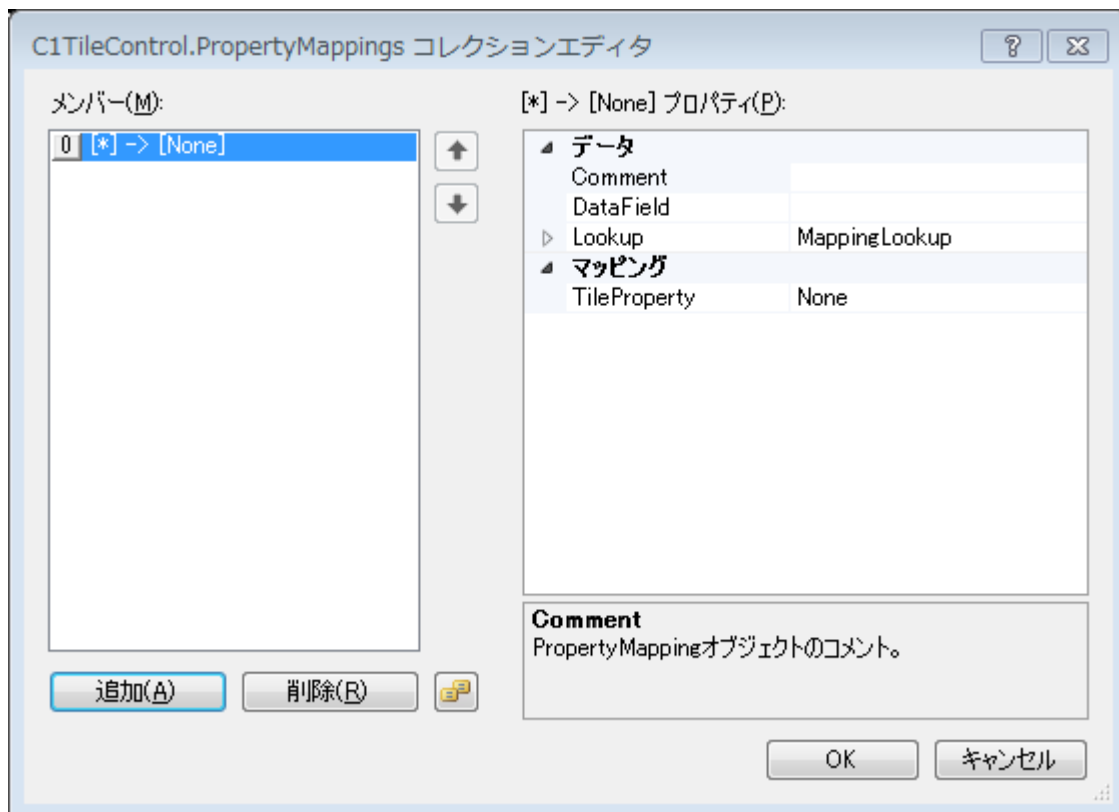
# TileControl for WinForms

プロジェクトを実行すると、製品テーブル内の製品名ごとにタイルがあることがわかりますが、タイルに製品名は表示されていません。この場合は、**ProductName** と **Text** を **C1TileControl** にマップする必要があります。

## 手順 4: TileControl のプロパティマッピングの設定

この手順では、PropertyMapping メンバを作成して、ProductName と ProductName のテキストを C1TileControl に接続します。

1. **C1TileControl** を選択し、C1TileControl のプロパティウィンドウで **Data** ノードを展開します。
2. C1TileControl のプロパティウィンドウで、**PropertyMappings** プロパティの横にある省略符ボタンをクリックします。PropertyMapping コレクションエディタが表示されます。
3. [**追加**] をクリックします。**C1.Win.C1Tile.PropertyMapping** 項目は、次のようになります。



4. [**追加**] をクリックし、[0] [\*] -> [None] のプロパティを次のように設定します。
  - **PropertyMapping.DataField** を **ProductName** に設定  
これで、製品名が各タイルに連結されます。
  - **PropertyMapping.TileProperty** を **Text** に設定

これで、**ProductName** が **C1TileControl** の **Text** プロパティに連結されます。

### プロジェクトの実行と動作の確認

[F4]を押してプロジェクトを実行し、ProductName の **Text** が **C1TileControl** の各タイルに連結されていることを確認します。

## データ連結チュートリアル - 複雑なデータ連結 TileControl の作成

このセクションでは、タイルコントロールをデータベースに連結する手順を説明します。C1TileControl をカテゴリデータ項目のコレクションに連結して、カテゴリごとにタイルを作成します。各タイルグループのヘッダーには、カテゴリの名前が表示されま

す。

プロジェクトを実行すると、下のスクリーンショットのような C1TileControl が表示されます。

### Beverages



## 手順 1: C1TileControl のデータソースの作成

この手順では、TileControl のデータ連結プロパティを使用して、後で C1TileControl を連結するためのデータソースを作成します。

### 新しいデータソースへの接続

1. フォームに **C1TileControl** を追加します。
2. プロジェクトツールバーで、[データ]メニューから、[新しいデータソースの追加]を選択します。データソース構成ウィザードダイアログボックスが表示されます。
3. [データベース]を選択し、[次へ]をクリックします。
4. [データセット]を選択し、[次へ]をクリックします。
5. [新しい接続]をクリックします。
6. [接続の追加]ダイアログボックスで、[参照]をクリックします。
7. [Microsoft Access データベースファイルの選択]ダイアログボックスで、Nwind.mdb(デフォルトの場所は **C:\Users\UserName\Documents\ComponentOne Samples\Common**)を見つけ、[開く]をクリックし、[OK]をクリックします。
8. [次へ]ボタンをクリックして続行します。データファイルをプロジェクトに追加し、接続文字列を修正するかどうかを確認するダイアログボックスが表示されます。プロジェクトにデータベースをコピーする必要はないため、[いいえ]をクリックします。
9. [次の名前で接続を保存する]チェックボックスがオンになっていることを確認し、[次へ]をクリックして続行します。
10. 接続文字列が **NwindConnectionString** という名前で作成されます。
11. データソース構成ウィザードで、[テーブル]ノードを選択します。
12. [完了]を選択します。

## 手順 2: データ接続とデータアダプタの構成

この手順では、products、categories、suppliers の各テーブルに3つの異なるデータアダプタを追加します。使用するデータセットにはテーブルごとに異なるデータアダプタが含まれるため、各アダプタの **Fill** メソッドを個別に呼び出す必要があります。

1. ツールボックスで、**OleDbDataAdapter** コンポーネントをダブルクリックします。このコンポーネントがツールボックスにない場合は、右クリックして[**アイテムの選択**]を選択します。ダイアログボックスの[.NET Framework コンポーネント]タブで、**OleDbDataAdapter** を選択します。コンポーネントトレイに OleDbDataAdapter が表示され、**データアダプタ構成ウィザード**が表示されます。
2. データアダプタ構成ウィザードで、データアダプタとして使用する接続をドロップダウンリストボックスから選択し(この場合は **C:\Users\UserName\Documents\ComponentOne Samples\Common\Nwind.mdb**)、[**次へ**]をクリックします。
3. [**SQL ステートメントを使用する**]がデフォルトで選択されています。[**次へ**]をクリックします。
4. **データアダプタ構成ウィザード**のテキストボックスに次の SQL ステートメントをコピーして貼り付けます。

```
SELECT    Products.*
FROM      Products
```

5. [**次へ**]をクリックし、[**はい**]をクリックしてクエリーに主キー列を追加します。
6. [**完了**]をクリックします。
7. **OldDataAdaper1** の **Name** プロパティを **productsDataAdapter** に、**OleDbConnection1** の **Name** プロパティを **productsConnection** に設定します。
8. 別の **OleDbDataAdapter** を追加します。
9. **データアダプタ構成ウィザード**で、データアダプタとして使用する接続をドロップダウンリストボックスから選択し(この場合は **C:\Users\UserName\Documents\ComponentOne Samples\Common\Nwind.mdb**)、[**次へ**]をクリックします。
10. [**SQL ステートメントを使用する**]がデフォルトで選択されています。[**次へ**]をクリックします。
11. **データアダプタ構成ウィザード**のテキストボックスに次の SQL ステートメントをコピーして貼り付けます。

```
SELECT    Categories.*
FROM      Categories
```

12. [**次へ**]をクリックし、[**はい**]をクリックしてクエリーに主キー列を追加します。
13. [**完了**]をクリックします。
14. **OldDataAdaper2** の **Name** プロパティを **categoriesDataAdapter** に、**OleDbConnection2** の **Name** プロパティを **categoriesConnection** に設定します。
15. 別の **OleDbDataAdapter** を追加します。
16. **データアダプタ構成ウィザード**で、データアダプタとして使用する接続をドロップダウンリストボックスから選択し(この場合は **C:\Users\UserName\Documents\ComponentOne Samples\Common\Nwind.mdb**)、[**次へ**]をクリックします。
17. [**SQL ステートメントを使用する**]がデフォルトで選択されています。[**次へ**]をクリックします。
18. **データアダプタ構成ウィザード**のテキストボックスに次の SQL ステートメントをコピーして貼り付けます。

```
SELECT    Suppliers.*
FROM      Suppliers
```

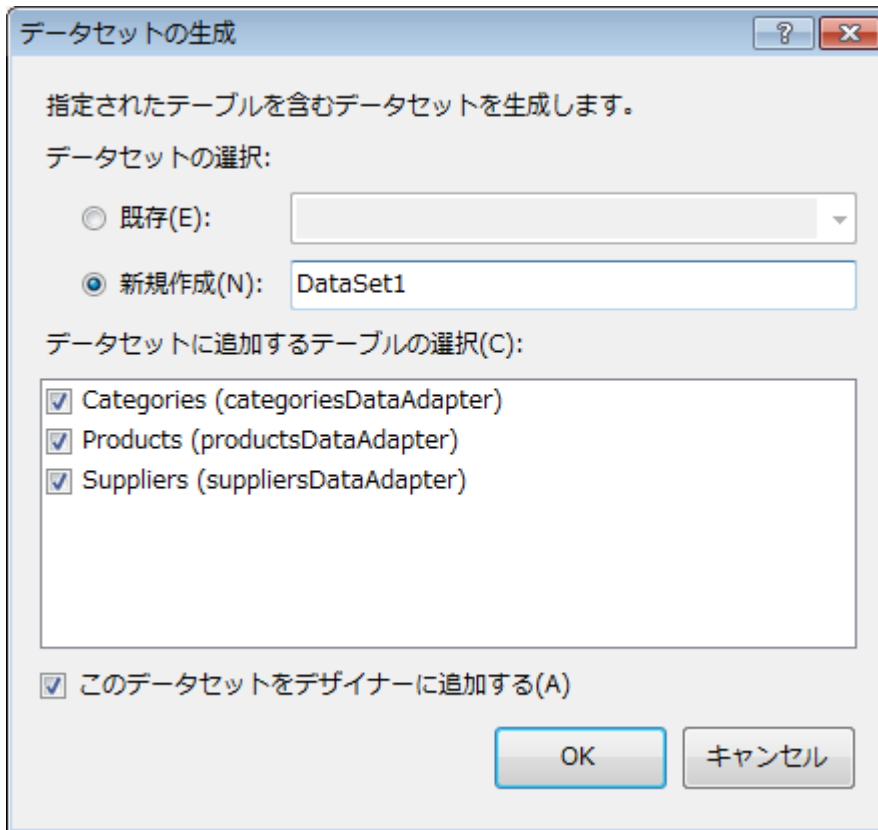
19. [**次へ**]をクリックし、[**はい**]をクリックしてクエリーに主キー列を追加します。
20. [**完了**]をクリックします。
21. **OldDataAdaper3** の **Name** プロパティを **SuppliersDataAdapter** に、**OleDbConnection3** の **Name** プロパティを **SuppliersConnection** に設定します。

## 手順 3: データセットの生成

この手順では、データアダプタに指定したクエリーに基づいてデータセットを作成します。このデータセットは、クラスの要素

(テーブル、列、制約)が記述された XML Schema(.xsd)ファイルに対応する DataSet クラスのインスタンスです。

1. [データ]メニューから、[データセットの生成]を選択します。  
[データセットの生成]ダイアログボックスが表示されます。



2. [新規]オプションを選択し、データセットに DataSet1 と名前を付けます。
3. [データセットに追加するテーブルの選択]の下でのリストで、**Categories**、**Products**、**Suppliers** の各テーブルを選択します。
4. [このデータセットをデザイナーに追加する]がオンになっていることを確認し、[OK]をクリックします。

新しいデータセットクラス **DataSet1** のインスタンスがフォームに追加されます。

## 手順 4: データスキーマの準備

この手順では、**DataSet1** の **Categories Product** リレーションと **Suppliers Product** リレーションを作成します。

1. マウスで Categories テーブルの **CategoryID** フィールドをクリックし、それを Products テーブルの **CategoryID** フィールドにドラッグしてドロップします。次のウィンドウのように、2つのテーブルの間に新しいリレーションが作成されます。

リレーションシップ

名前(N):  
Categories\_Products

データセットでテーブルに関連付けるキーを指定してください。

親テーブル(P): Categories      子テーブル(C): Products

列(O):

キー列	外部キー列
CategoryID	SupplierID

作成する制約の選択

リレーションシップと外部キー制約の両方(B)

外部キー制約のみ(F)

リレーションシップのみ(R)

UpdateRule の設定(U): Cascade

DeleteRule の設定(D): Cascade

AcceptRejectRule の設定(A): None

入れ子になったリレーションシップ(E)

OK      キャンセル

- マウスで Suppliers テーブルの **SupplierID** フィールドをクリックし、それを Products テーブルの **SupplierID** フィールドにドラッグしてドロップします。

次のウィンドウのように、2つのテーブルの間に新しいリレーションが作成されます。



## 手順 5: C1TileControl への Products テーブルの連結

この手順では、products データテーブルを TileControl に連結し、products、suppliers、categories にデータセットを設定します。

### TileControl のデータプロパティの設定

1. **C1TileControl** を選択し、C1TileControl のプロパティウィンドウで Data ノードを展開します。
2. **C1TileControl.DataSource** プロパティを **dataSet11** に設定します。それには、[その他のデータソース]ノードを展開する必要があります。
3. **DataMember** を **Products** に設定します。

この2つのプロパティを設定すると、**dataSet11** の products データテーブルが TileControl に連結されます。

### TileControl へのデータの挿入

TileControl は **dataSet11** に連結されますが、自動的にデータが挿入されません。データセットにデータを設定するには、次のようにデータアダプタメソッドを呼び出します。

Visual Basic コードの書き方

```
categoriesDataAdapter.Fill(dataSet11)
suppliersDataAdapter.Fill(dataSet11)
productsDataAdapter.Fill(dataSet11)
```

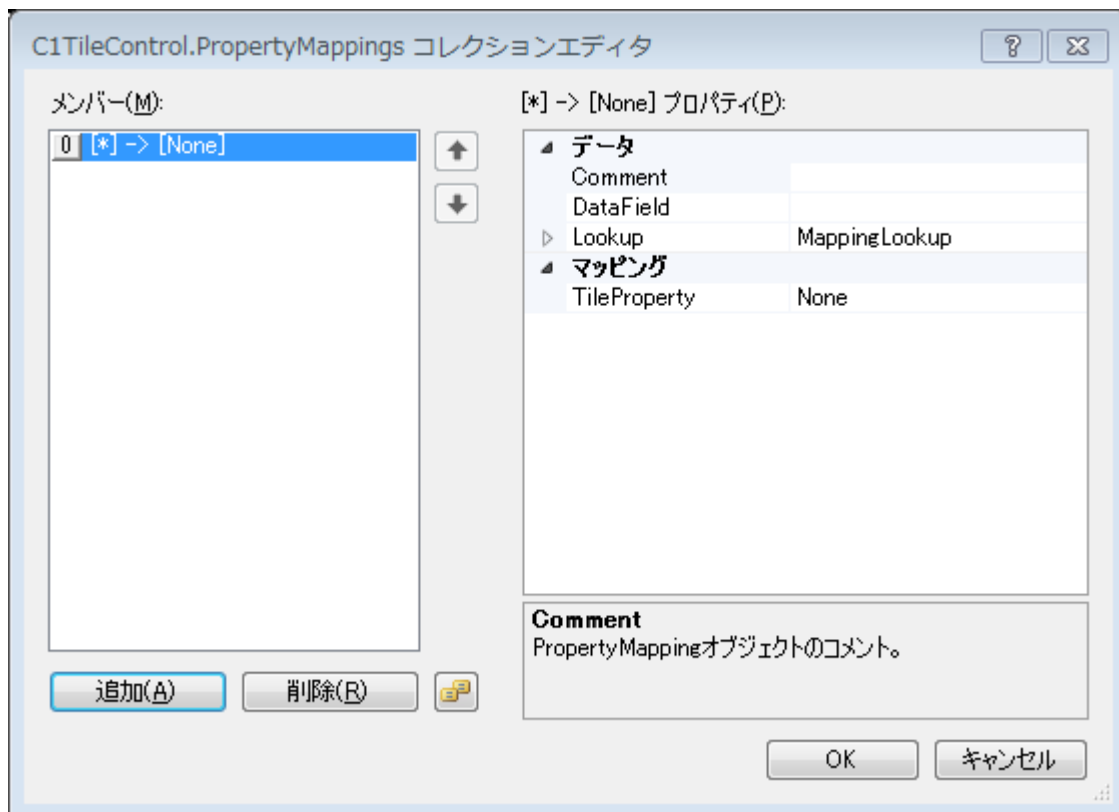
## C# コードの書き方

```
categoriesDataAdapter.Fill(dataSet11);
suppliersDataAdapter.Fill(dataSet11);
productsDataAdapter.Fill(dataSet11);
```

## 手順 6: TileControl のプロパティマッピングの設定

この手順では、設計時に **C1TileControl.PropertyMappings** コレクションエディタを使用して、PropertyMapping クラスにプロパティマッピングを追加します。

1. フォーム内の C1TileControl を選択します。C1TileControl のプロパティウィンドウで、**PropertyMappings** プロパティの横にある省略符ボタンをクリックします。**C1TileControl.PropertyMappings コレクションエディタ**が表示されます。
2. **[追加]**をクリックします。C1.Win.C1Tile.PropertyMapping 項目は、次のようになります。



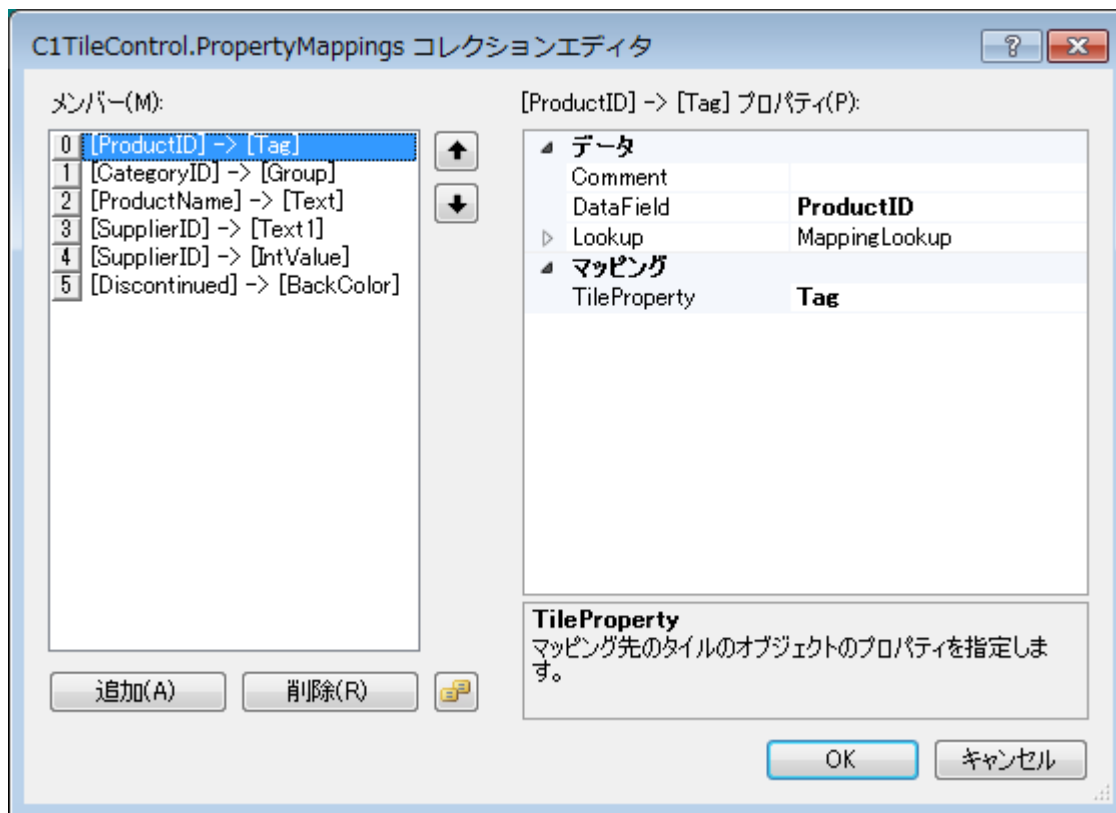
3. [\*]->[None] メンバの **PropertyMapping.DataField** を ProductID に、PropertyMapping.TileProperty を Tag に設定します。
4. **[追加]**をクリックし、[\*]->[None] [1] のプロパティを次のように設定します。
  - DataField を CategoryID に設定
  - PropertyMapping.Lookup.DataSource を dataSet11 に設定
  - PropertyMapping.Lookup.DisplayMember を Categories.CategoryName に設定
  - PropertyMapping.Lookup.ValueMember を Categories.CategoryID に設定
  - CategoryID は、Categories テーブルのカテゴリレコードを指す外部キーです。ここで、ルックアップテーブルを作成しています。このルックアップテーブルは、Products レコードの CategoryID 値を受け取り、その値を使用してリレーションシップをたどり、わかりやすいカテゴリ名を返します。

- TileProperty を Group に設定

これで、各タイルグループにカテゴリ名が表示されます。たとえば、Beverages、Condiments、Dairy Products などです。

5. [追加]をクリックし、[\*]->[None] [2] のプロパティを次のように設定します。
6. DataField を ProductName に設定
7. TileProperty を Text に設定
8. これで、各タイルに製品の名前が表示されます。たとえば、Chai、Chang などです。
9. [追加]をクリックし、[\*]->[None] [3] のプロパティを次のように設定します。
  - DataField を SupplierID に設定
  - PropertyMapping.Lookup.DataSource を dataSet11 に設定
  - PropertyMapping.Lookup.DisplayMember を Suppliers.Country に設定
  - PropertyMapping.Lookup.ValueMember を Suppliers.SupplierID に設定
  - TileProperty を Text1 に設定
10. これで、国名が各タイルに割り当てられます。
11. [追加]をクリックし、[\*]->[None] [4] のプロパティを次のように設定します。
  - DataField を SupplierID に設定
  - PropertyMapping.Lookup.DataSource を dataSet11 に設定
  - PropertyMapping.Lookup.DataMember を Suppliers.Country に設定
  - PropertyMapping.Lookup.ValueMember を Suppliers.SupplierID に設定
  - TileProperty を IntValue に設定
12. [追加]をクリックし、[\*]->[None] [5] のプロパティを次のように設定します。
13. DataField を Discontinued に設定
  - TileProperty を BackColor に設定

プロパティマッピングがメンバリストに次のように表示されます。

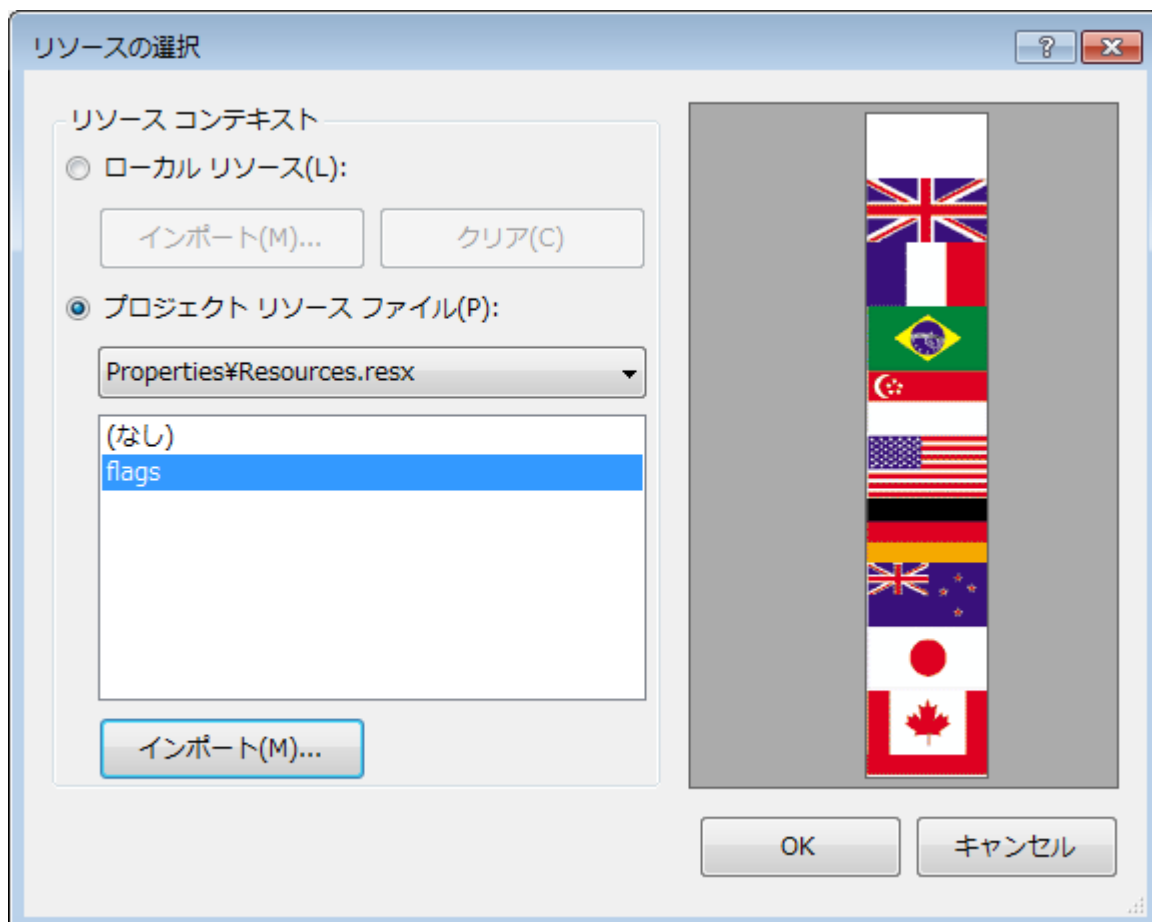


## 手順 7: C1TileControl のデフォルトテンプレートの変更

この手順では、C1TileControl のデフォルトテンプレートを変更します。

# TileControl for WinForms

1. ソリューションエクスプローラで、プロジェクト名を右クリックし、[追加]→[新規フォルダ]を選択します。
2. C:\Users\username\Documents\ComponentOne Samples\WinForms\C1TileControl\Samples\Resourcesにある Databound サンプルプロジェクトの flags.png をローカルプロジェクトにコピーします。
3. このフォルダの名前を **Resources** に変更し、このフォルダから flags.png イメージを追加します。
4. ソリューションエクスプローラで、[すべてのファイルを表示]ボタンをクリックします。
5. **Resources** フォルダを右クリックし、イメージを選択し、[プロジェクトに含める]を選択します。
6. **C1TileControl** のプロパティウィンドウで、**DefaultTemplate** ノードを展開し、**Templates** プロパティの横にある省略符ボタンをクリックします。
7. **Elements** プロパティの横の省略符ボタンをクリックします。**Template.Elements コレクションエディタ**が表示されます。
8. **Children** プロパティの横の省略符ボタンをクリックします。
9. **Members** リストから **ImageElement** を削除します。
10. Members リストの **TextElement** を選択し、その **FontBold** を **True** に設定します。
11. [OK]をクリックして保存し、**PanelElement.Children コレクションエディタ**を閉じます。
12. [追加]項目の横にあるドロップダウン矢印をクリックし、**TextElement** を選択します。
13. **TextElement** を選択し、**ForeColorSelector** を **Unbound** に設定します。
14. **ForeColor** を **255, 192, 192** に設定します。
15. **TextSelector** を **Text1** に、**Alignment** を **BottomLeft** に設定します。
16. これで、各国名のテキストが各タイトルの左下に表示されるように設定されます。
17. **Template.Elements コレクションエディタ**で、[追加]ボタンの横にあるドロップダウン矢印をクリックし、**ImageElement** を選択します。
18. Image の横の省略符ボタンをクリックします。[リソースの選択]ダイアログボックスが表示されます。
19. [インポート]ボタンをクリックし、プロジェクトの Resources フォルダを参照し、flags.png を選択します。



20. [OK]をクリックして保存し、[リソースの選択]ダイアログボックスを閉じます。
21. **Template.Elements コレクションエディタ**で、**ImageSelector** を **Unbound** に設定します。
22. **ImageRows** を **17** に、**Alignment** を **BottomRight** に設定します。

23. [OK]をクリックして保存し、**Template.Elements** コレクションエディタを閉じます。

## 手順 8: FormatValue イベントでのタイルの書式設定

この手順では、**FormatValue** イベントを使用して、タイルの背景、タイルの画像、タイルの国名の書式を設定します。

1. C1TileControl のイベントプロパティウィンドウで、FormatValue 項目をダブルクリックしてイベントハンドラを作成します。
2. コードビューで、**C1TileControl1\_FormatValue** イベントハンドラに次のコードを追加します。

### Visual Basic コードの書き方

#### Visual Basic

```
Private Sub c1TileControl1_FormatValue(sender As Object, e As FormatValueEventArgs)
    If e.TileProperty = TileProperty.BackColor Then
        If TypeOf e.Value Is Boolean AndAlso CBool(e.Value) Then
            e.Value = Color.Firebrick
        Else
            e.Value = Color.DimGray
        End If
    ElseIf e.TileProperty = TileProperty.IntValue Then
        Dim result As Integer = 0
        Select Case TryCast(e.Value, String)
            Case "UK"
                result = 1
                Exit Select
            Case "France"
                result = 2
                Exit Select
            Case "Brazil"
                result = 3
                Exit Select
            Case "Singapore"
                result = 4
                Exit Select
            Case "USA"
                result = 5
                Exit Select
            Case "Germany"
                result = 6
                Exit Select
            Case "Australia"
                result = 7
                Exit Select
            Case "Japan"
                result = 8
                Exit Select
            Case "Canada"
                result = 9
                Exit Select
            Case "Netherlands"
                result = 10
        End Select
    End If
End Sub
```

```
        Exit Select
    Case "Finland"
        result = 11
        Exit Select
    Case "Norway"
        result = 12
        Exit Select
    Case "Italy"
        result = 13
        Exit Select
    Case "Spain"
        result = 14
        Exit Select
    Case "Sweden"
        result = 15
        Exit Select
    Case "Denmark"
        result = 16
        Exit Select
    End Select
    e.Value = result
End If
End Sub
```

## C# コードの書き方

```
C#
{
    if (e.TileProperty == TileProperty.BackColor)
    {
        if (e.Value is bool && (bool)e.Value)
            e.Value = Color.Firebrick;
        else
            e.Value = Color.DimGray;
    }
    else if (e.TileProperty == TileProperty.IntValue)
    {
        int result = 0;
        switch (e.Value as string)
        {
            case "UK":
                result = 1; break;
            case "France":
                result = 2; break;
            case "Brazil":
                result = 3; break;
            case "Singapore":
                result = 4; break;
            case "USA":
                result = 5; break;
            case "Germany":
                result = 6; break;
        }
    }
}
```

```
        case "Australia":
            result = 7; break;
        case "Japan":
            result = 8; break;
        case "Canada":
            result = 9; break;
        case "Netherlands":
            result = 10; break;
        case "Finland":
            result = 11; break;
        case "Norway":
            result = 12; break;
        case "Italy":
            result = 13; break;
        case "Spain":
            result = 14; break;
        case "Sweden":
            result = 15; break;
        case "Denmark":
            result = 16; break;
    }
    e.Value = result;
}
}
```

## タスク別ヘルプ

タスク別ヘルプのセクションは、Visual Studio 環境でのプログラミングに精通し、**C1TileControl** コントロールの一般的な使用方法を理解しているユーザーを対象としています。

各トピックでは、**C1TileControl** コントロールを使用した特定のタスクのソリューションを提供します。各トピックで概説されている手順に従うことによって、さまざまな **C1TileControl** 機能を使用したプロジェクトを作成できます。

## C1TileControl へのテンプレートの追加

テンプレートは、設計時かプログラムによって作成できます。設計時、テンプレートは **C1TileControl.Templates コレクションエディタ** を使用して作成できます。また、テンプレートはプログラムで **TemplateCollection** `TemplateCollection` クラスから作成することもできます。テンプレートは、**InsertItem** メソッドを使用し、`TemplateCollection` の指定したインデックスに挿入できます。

### 設計時の場合

設計時にテンプレートを追加するには、以下を実行します。

1. **C1TileControl** を選択し、スマートタグをクリックして、**[C1TileControl タスク]** メニューを開きます。
2. **[タイルの編集]** を選択します。Select **Edit Tiles**.  
**[C1TileControl.Templates コレクションエディタ]** エディタが表示されます。
3. **<追加>** をクリックし、テンプレートを**[メンバ]** リストに追加します。

## 指定したタイルへのテンプレートの割り当て

設計時に **C1TileControl.Templates コレクションエディタ** を使用するか、プログラムで `TemplateCollection` クラスからテンプレートを作成した後、それを指定したタイルに割り当てることができます。

### 設計時の場合

テンプレートを指定したタイルに割り当てするには、以下を実行します。

1. テンプレートを追加するタイルを選択します。選択したタイルの**[C1TileControl タスク]** メニューが表示されます。
2. 指定したタイルに割り当てる**テンプレート**のドロップダウンリストボックスからテンプレートを選択します。たとえば、**template1** を **tile1** に割り当てます。



C1TileControl タスク	
tile2:	
タイルのテキスト	タイル2
ツールチップのテキスト	
タイルのグループ	group1
タイルのインデックス	1
水平方向のサイズ	1
垂直方向のサイズ	1
IntValue	0
<input checked="" type="checkbox"/> チェック済み	
テンプレート	template1
タイルテンプレートの編集	(既定)
グループの選択	template1
C1TileControl の選択	template2

## テンプレートへの要素の追加

テンプレート要素は、設計時かプログラムによって作成できます。設計時、テンプレートは **C1TileControl.Templates コレクションエディタ** を使用して作成できます。また、テンプレート要素はプログラムで **TemplateCollection** クラスから作成することもできます。テンプレートは、**InsertItem** メソッドを使用し、TemplateCollection の指定したインデックスに挿入できます。

### デザイン時

設計時に要素をテンプレートに追加するには、以下を実行します。

1. **C1TileControl** を選択し、スマートタグをクリックして、[**C1TileControl タスク**]メニューを開きます。
2. [**タイルの編集**]を選択します。  
  
[**C1TileControl.Templates コレクションエディタ**]エディタが表示されます。
3. <追加>をクリックし、テンプレートを[メンバ]リストに追加します。
4. **Elements** プロパティの横にある<...>ボタンを選択します。[**Template.Elements コレクションエディタ**]が表示されます。
5. ドロップダウン矢印をクリックし、**PanelElement** を2回選択します。2つのパネルがテンプレートに追加されます。

## C1TileControl のルックアップテーブルの作成

**PropertyMapping コレクションエディタ** を使用してルックアップテーブルを作成するには、次の手順に従います。

1. 新しいデータソースを作成します。詳細については、「**新しいデータソースへの接続**」を参照してください。
2. データソース構成ウィザードで、[テーブル]を展開し、Categories、Products、Suppliers を選択します。
3. [完了]を選択します。
4. **C1TileControl** を選択し、**C1TileControl** のプロパティウィンドウで **Data** ノードを展開します。
5. **C1TileControl.DataSource** プロパティを **c1NwindDataSet** に設定します。
6. **DataMember** を **Products** に設定します。

この2つのプロパティを設定すると、c1NwindDataSet の products データテーブルが TileControl に連結されます。

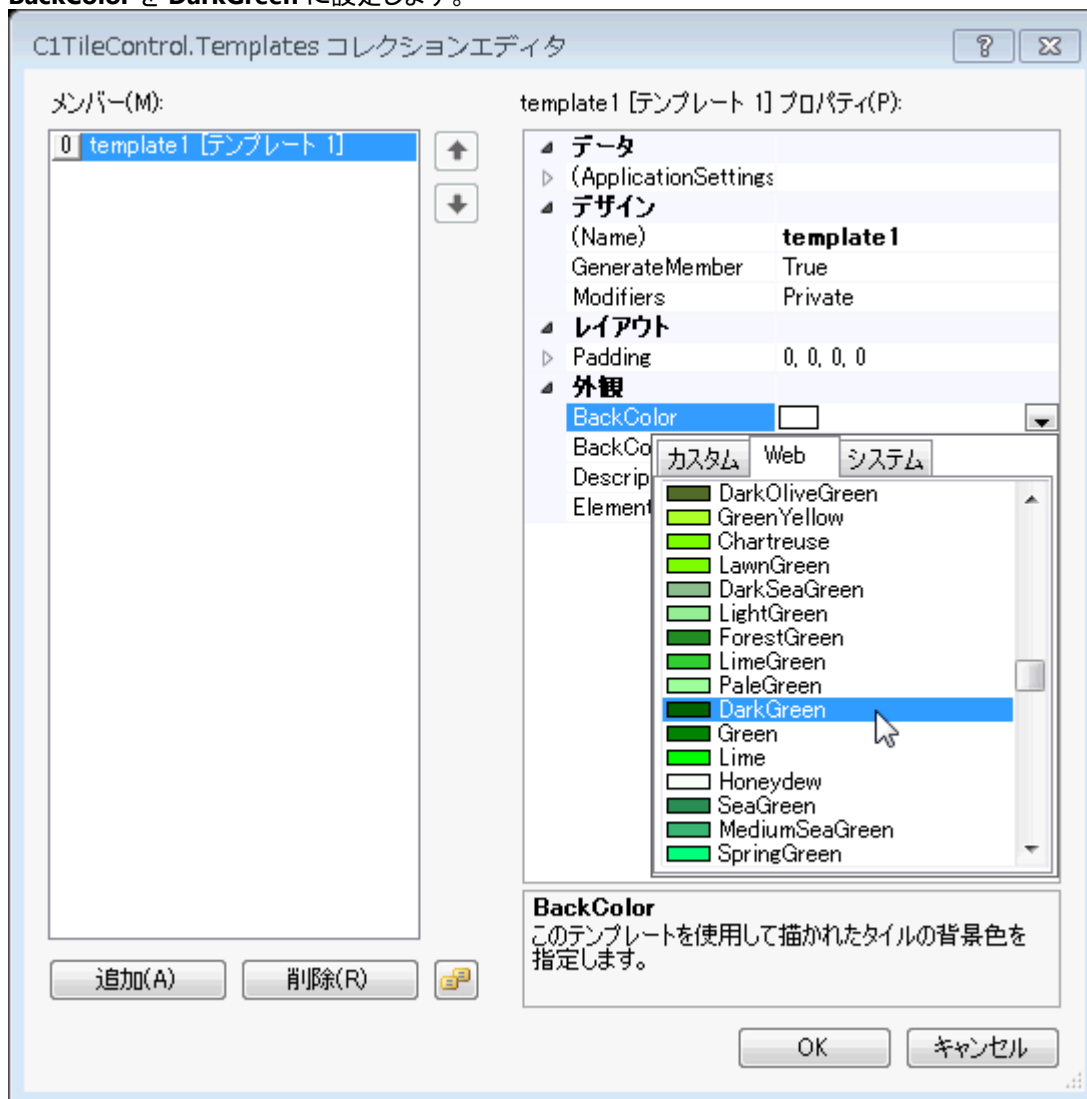
## テンプレートの背景色の変更

設計時にテンプレートの背景色を変更するには、以下を実行します。

1. **C1TileControl** を選択し、スマートタグをクリックして、**[C1TileControl タスク]**メニューを開きます。
2. **[タイルの編集]**を選択します。

**[C1TileControl.Templates コレクションエディタ]**エディタが表示されます。

3. **<追加>**をクリックし、テンプレートを**[メンバ]**リストに追加します。
4. **BackColor** を **DarkGreen** に設定します。



5. **BackColorSelector** を **Unbound** に設定します。

これにより、デフォルトの背景色でなく、DarkGreen の背景色がこのテンプレートに割り当てられます。テンプレートが指定したタイルに割り当てられると、新しい背景色が表示されます。

6. 1番目のタイルを選択し、Template プロパティを Template1 に設定します。1番目のタイルに新しい背景色を持つテンプレートが更新されます。

### 🟢 このピックの作業結果

テンプレートの新しい背景色が1番目のタイルに表示されます。



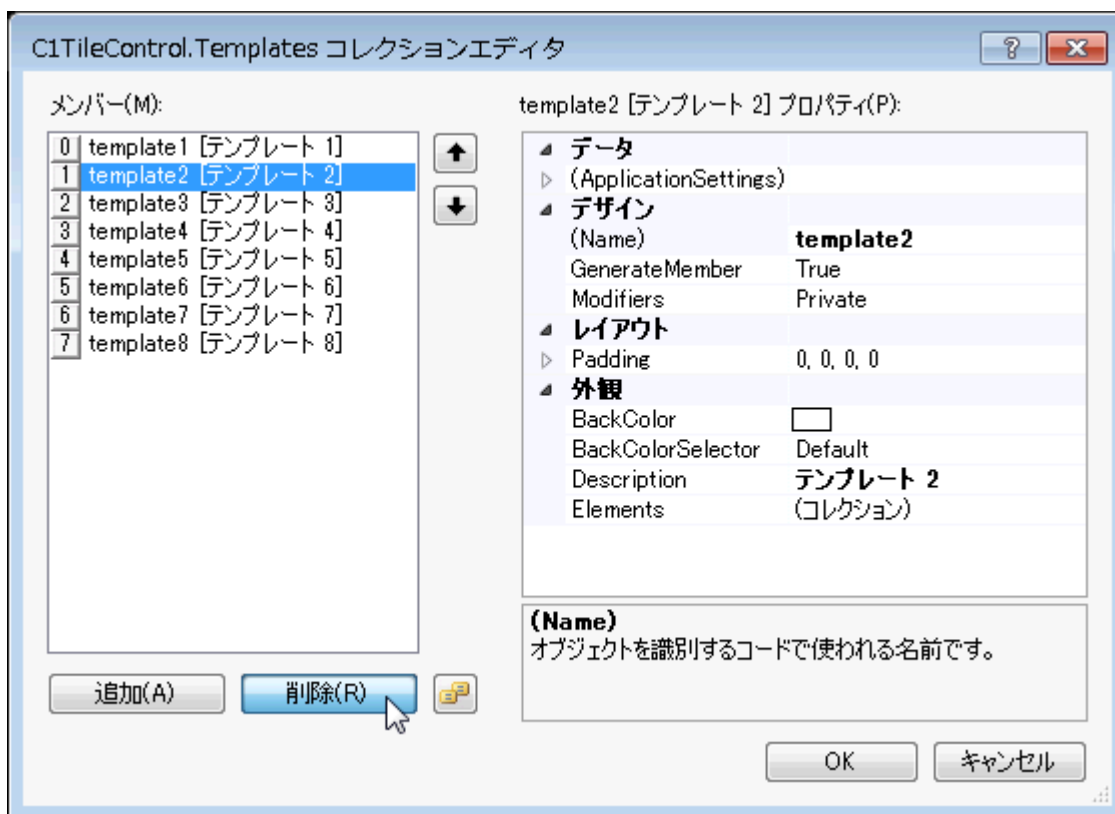
## 特定のテンプレートの削除

TileControl のテンプレートは、プログラムか設計時に削除できます。

### 設計時の場合

設計時に特定のテンプレートを TileControl から削除するには、以下を実行します。

1. C1TileControl を右クリックし、[テンプレートの編集]を選択します。[C1TileControl.Templates コレクションエディタ]が表示されます。
2. [メンバ]リストからテンプレートを選択し、<削除>をクリックします。



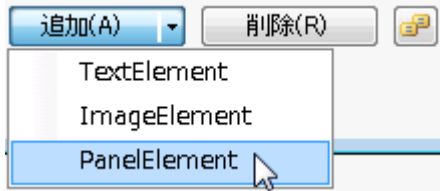
## タイマーによるテキストビューの変更

タイマーでテキストビューを変更するには、以下を実行します

### 1番目のテンプレートを追加する

# TileControl for WinForms

1. Tile コントロールを右クリックし、[テンプレートの編集]を選択します。[C1TileControl.Templates コレクションエディタ]が表示されます。
2. <追加>を2回クリックし、2つのテンプレートを C1TileControl に追加します。
3. **template1** を選択し、**Elements コレクション**の横にある<...>ボタンをクリックします。[Template.Elements コレクションエディタ]が表示されます。
4. [追加]のドロップダウンリストボックスから **PanelElement** を選択します。



5. PanelElement の各プロパティを次のように設定します。
  - Alignment プロパティを **TopLeft** に設定。
  - ChildSpacing プロパティを **0** に設定。これにより、子要素間のデフォルトのスペースが5ピクセルから0ピクセルに減ります。
  - Orientation プロパティを **Vertical** に設定。
6. **Children** プロパティの横の<...>ボタンをクリックします。
7. 2つの **TextElements** を **PanelElement** に追加します。
8. 2番目のテキスト要素 [1] **TextElement** を選択し、**TextSelector** プロパティを **Text1** に設定します。これにより、Text1 プロパティの値がこのテンプレートに割り当てられます。
9. <OK>をクリックし、[PanelElement.Children コレクションエディタ]を保存して閉じた後、<OK>をクリックし、[Template.Elements コレクションエディタ]を保存して閉じます。

## 2番目のテンプレートを追加する

1. [C1TileControl.Templates コレクションエディタ]で、**template2** を選択します。
2. **Elements コレクション**の横にある<...>ボタンをクリックします。[Template.Elements コレクションエディタ]が表示されます。
3. <追加>ボタンの横にあるドロップダウン矢印をクリックし、**PanelElement** を追加します。
4. [0]Panel Element の各プロパティを次のように設定します。
  - Alignment プロパティを **TopLeft** に設定。
  - ChildSpacing プロパティを **0** に設定。
  - Orientation プロパティを **Vertical** に設定。
5. **Children (コレクション)**プロパティの横にある<...>ボタンをクリックし、2つの **TextElement** を追加します。
6. 1番目のテキスト要素 [0] **TextElement** を選択し、**TextSelector** プロパティを **Text1** に設定します。
7. 2番目のテキスト要素 [1] **TextElement** を選択し、**TextSelector** プロパティを **Text2** に設定します。
8. <OK>をクリックし、[PanelElement.Children コレクションエディタ]を保存して閉じた後、<OK>をクリックし、[Template.Elements コレクションエディタ]を保存して閉じます。
9. Tile1 を右クリックし、[グループの編集]を選択します。[C1TileControl.Groups コレクションエディタ]が表示されます。
10. **Tiles コレクション**の横にある<...>ボタンをクリックします。
11. tile1 を選択し、各プロパティを次のように設定します。
  - Template プロパティを **template1** に設定。template1 の設定が Tile1 に適用されます。
  - Text1 プロパティを **タイルの詳細説明** に設定。
  - Text2 プロパティを **タイルの動作の追加情報と説明** に設定。
12. <OK>をクリックし、[Group.Tiles コレクションエディタ]を保存して閉じます。

Tile1 は次のような表示になるはずですが。

## タイマーを追加して Tile1 のテンプレートビューを変更する

1. WindowsForm の **Timer** コントロールをダブルクリックし、コンポーネントトレイに追加します。
2. timer1 の **Interval** プロパティを **3000** に設定して、**Enabled** プロパティを **True** に設定します。
3. TileControl フォームを右クリックし、[コードの表示]を選択します。
4. 次のコードをプロジェクトに追加し、各テンプレートのテキストビューを変更するアニメーションを作成します。

### Visual Basic コードの書き方

#### Visual Basic

```
Public Partial Class Form1
    Inherits Form
    Private _tile1Flipped As Boolean

    Public Sub New()
        InitializeComponent()
    End Sub

    Private Sub timer1_Tick(sender As Object, e As EventArgs)
        Dim a As Boolean = _tile1Flipped
        tile1.Template = If(a, template1, template2)
        _tile1Flipped = Not a
    End Sub
End Class
```

### C# コードの書き方

#### C#

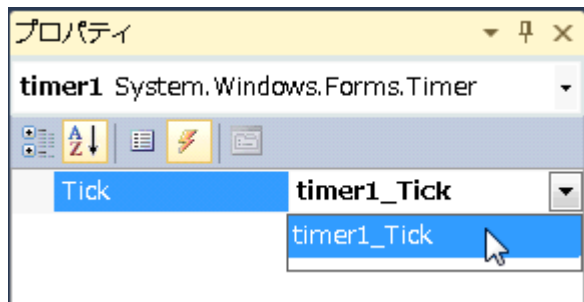
```
public partial class Form1 : Form
{
    bool _tile1Flipped;

    public Form1()
    {
        InitializeComponent();
    }

    private void timer1_Tick(object sender, EventArgs e)
    {
        bool a = _tile1Flipped;
        tile1.Template = a ? template1 : template2;
        _tile1Flipped = !a;
    }
}
```

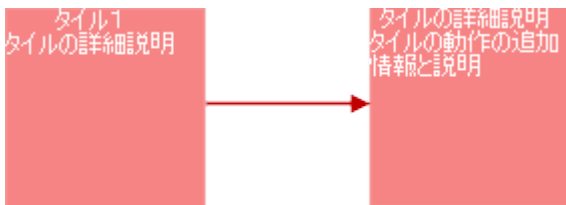
5. デザインビューで、**timer\_Tick** イベントハンドラを **timer1** に割り当てます。

# TileControl for WinForms



## ✔このトピックの作業結果

タイルはタイマーに基づいてテンプレートを変更します。タイルの1番目のテンプレートが数秒間表示された後、2番目のテンプレートが1番目の代わりに表示されます。



## XML ファイルの TileControl の保存／ロード

このトピックでは、**C1TileControl** を XML ファイルとして保存する方法と既存の **C1TileControl** を xml ファイルからロードする方法を説明します。

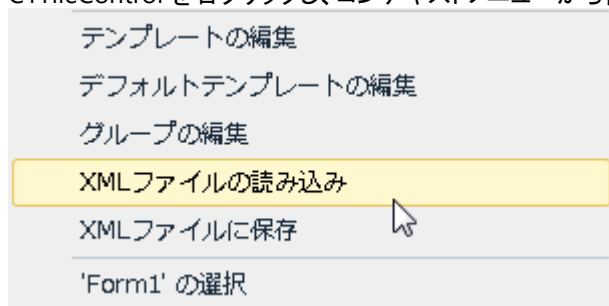
## XML ファイルからの TileControl のロード

このタスクでは、実行時とコードで C1TileControl を XML ファイルとしてロードする方法を示します。

### 実行時に C1TileControl を XML ファイルとしてロードする

実行時に C1TileControl を XML ファイルとしてロードするには、以下を実行します。

1. C1TileControl を右クリックし、コンテキストメニューから **[XML ファイルからロード]** 項目を選択します。



[Xml ファイルからロード] ダイアログボックスが表示されます。

2. xml ファイルのロード元を参照します。=
3. [Xml ファイルからロード] ダイアログボックスで **〈開く〉** をクリックします。

### コードで C1TileControl を XML ファイルからロード

コードで template1 を XML ファイルからロードするには、以下を実行します。

### Visual Basic コードの書き方

#### Visual Basic

```
Private Sub btnLoadXml_Click(sender As Object, e As EventArgs)
    Using dlg As New OpenFileDialog()
        dlg.DefaultExt = ".xml"
        dlg.Filter = "XML files|*.xml|All files|*.*"
        dlg.Title = "Load From Xml File"
        If dlg.ShowDialog() = DialogResult.OK Then
            Try
                template1.LoadXml(dlg.FileName)
            Catch
                MessageBox.Show("Bad tilecontrol XML.", dlg.Title)
            End Try
        End If
    End Using
End Sub
```

### C# コードの書き方

#### C#

```
private void btnLoadXml_Click(object sender, EventArgs e)
{
    using (OpenFileDialog dlg = new OpenFileDialog())
    {
        dlg.DefaultExt = ".xml";
        dlg.Filter = "XML files|*.xml|All files|*.*";
        dlg.Title = "Load From Xml File";
        if (dlg.ShowDialog() == DialogResult.OK)
        {
            try
            {
                Tilecontrol.LoadXml(dlg.FileName);
            }
            catch
            {
                MessageBox.Show("Bad tilecontrol XML.", dlg.Title);
            }
        }
    }
}
```

## XML ファイルとしての TileControl の保存

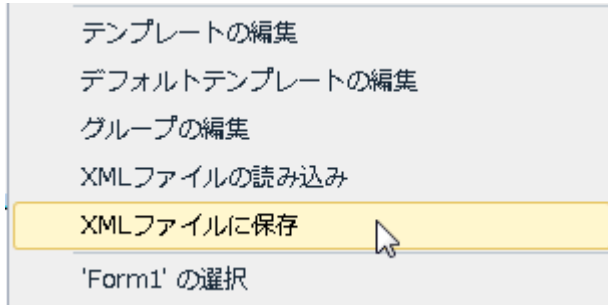
このタスクでは、設計時とコードで C1TileControl を XML ファイルとして保存する方法を示します。

### 設計時に C1TileControl を XML ファイルとして保存する

設計時に C1TileControl を XML ファイルとして保存するには、以下を実行します。

# TileControl for WinForms

1. C1TileControl を右クリックし、コンテキストメニューから[Xml ファイルに保存]項目を選択します。



[Xml ファイルに保存]ダイアログボックスが表示されます。

2. xml ファイルの保存先を参照します。
3. [Xml ファイルに保存]ダイアログボックスで<保存>をクリックします。

## コードで C1TileControl を XML ファイルとして保存する

コードで C1TileControl を XML ファイルとして保存するには、以下を実行します。

### Visual Basic コードの書き方

#### Visual Basic

```
Private Sub menuItemSaveXml_Click(sender As Object, e As EventArgs)
    Using dlg As New SaveFileDialog()
        dlg.DefaultExt = ".xml"
        dlg.FileName = "tilecontrol"
        dlg.Filter = "XML files|*.xml|All files|*.*"
        dlg.Title = "Save As Xml File"
        If dlg.ShowDialog() = DialogResult.OK Then
            TileControl.SaveXml(dlg.FileName)
        End If
    End Using
End Sub
```

### C# コードの書き方

#### C#

```
private void menuItemSaveXml_Click(object sender, EventArgs e)
{
    using (SaveFileDialog dlg = new SaveFileDialog())
    {
        dlg.DefaultExt = ".xml";
        dlg.FileName = "tilecontrol";
        dlg.Filter = "XML files|*.xml|All files|*.*";
        dlg.Title = "Save As Xml File";
        if (dlg.ShowDialog() == DialogResult.OK)
        {
            TileControl.SaveXml(dlg.FileName);
        }
    }
}
```



## TileControl とグループのテキストの設定

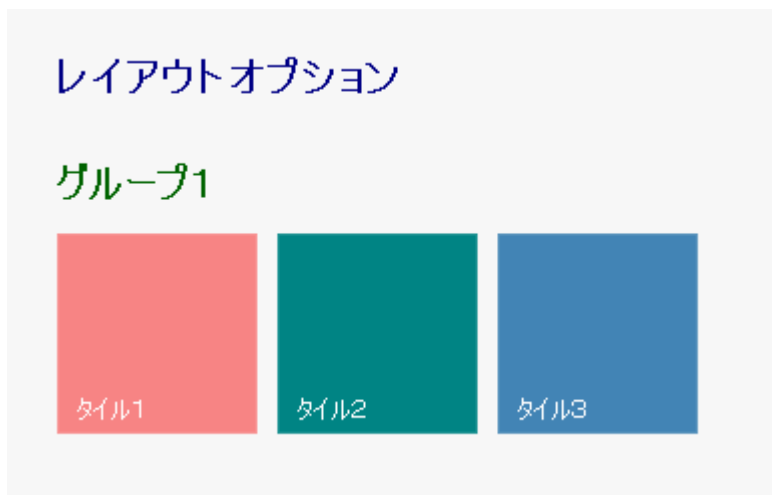
TileControl のテキスト、そのフォントサイズ、および色と、グループのテキスト、そのフォントサイズ、および色は、プログラムを使用するか設計時に TileControl とグループに適用できます。

TileControl とグループのテキストを変更するには、以下を実行します。

1. **C1TileControl** を右クリックし、[**プロパティ**]を選択します。C1TileControl のプロパティペインが表示されます。
2. [テキスト]テキストボックス内に、C1TileControl に表示されるテキスト(**Layout Options** など)を入力します。
3. **ForeColor** プロパティを **Navy** に設定します。これによりグループの前景色も設定されることに注意してください。グループの前景色を変更するには **GroupForeColor** プロパティで指定できます。
4. **TextSize** プロパティを **16** に設定します。これにより、**Font.Size** プロパティがオーバーライドされます。
5. C1TileControl のプロパティペインで、**Groups** ノードを展開します。
6. **GroupFont** を **True** に設定します。グループのフォントが太字で表示されます。
7. **GroupForeColor** を **DarkGreen** に設定します。

### ✔このトピックの作業結果

C1TileControl とグループのテキストが変更されました。



## C1TileControl へのグループの追加

グループは、設計時かプログラムによって作成できます。設計時、グループは **C1TileControl.Groups コレクションエディタ** を使用して作成できます。また、グループはプログラムで **Group** クラスから作成することもできます。**InsertItem** メソッドを使用し、グループを **GroupCollection** の指定したインデックスに挿入できます。

### 設計時の場合

設計時にテンプレートを追加するには、以下を実行します。

1. **C1TileControl** を右クリックし、[**グループの編集**]を選択して、[**C1TileControl.Groups コレクションエディタ**]を開きます。
2. <追加>ボタンをクリックし、新しいグループを C1TileControl に追加します。グループのテキストが空の状態が表示され、タイルをグループに追加するまでタイルは含まれていません。

## C1TileControl からのグループの削除

グループは、設計時かプログラムによって削除できます。設計時、グループは **C1TileControl.Groups コレクションエディタ** を

# TileControl for WinForms

使用して削除できます。また、グループはプログラムで **GroupCollection** クラスから削除することもできます。グループは、**RemoveItem** メソッドを使用し、GroupCollection の指定したインデックスから削除できます。

## 設計時の場合

グループを C1TileControl から削除するには、以下を実行します。

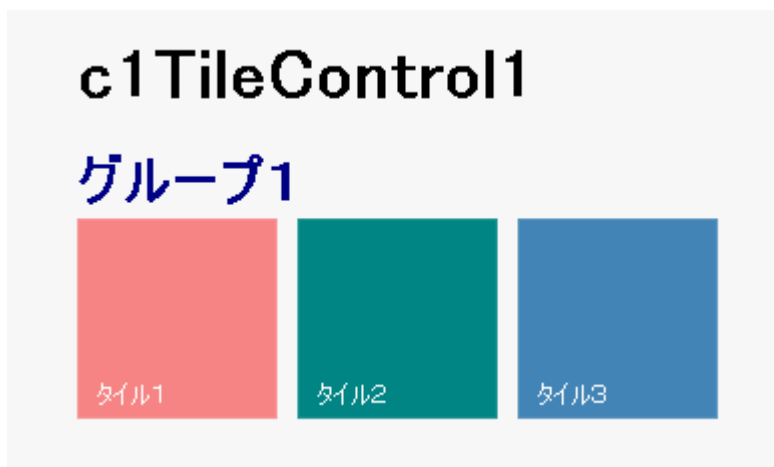
1. TileControl を Windows フォームに追加します。
2. **C1TileControl** を右クリックし、[**グループの編集**]を選択して、[**C1TileControl.Groups コレクションエディタ**]を開きます。
3. [メンバ]リストから、削除するグループを選択し、<削除>ボタンをクリックして、グループを C1TileControl から削除します。

## グループのフォントプロパティの変更

グループのフォントプロパティを変更するには、以下を実行します。

1. **C1TileControl** を右クリックし、[**プロパティ**]を選択します。
2. **Groups** ノードの下の **GroupFont** プロパティを指定し、<...>ボタンをクリックします。
3. Font を Calisto MT に、FontStyle を Bold、および Size を 20 に設定します。GroupTextBold と GroupTextSize がこれらの設定をオーバーライドすることに注意してください。そのため、FontStyle を Bold に設定しても、GroupTextBold が False に設定されている場合は、グループは Group.TextBold プロパティからの設定を継承します。
4. <OK>をクリックし、[フォント]ダイアログボックスを保存して閉じます。
5. GroupTextSize を 16 に設定します。このプロパティが[フォント]ダイアログボックスで適用されたフォントの Size プロパティよりも優先されることに注意してください。

### ✔ このトピックの作業結果

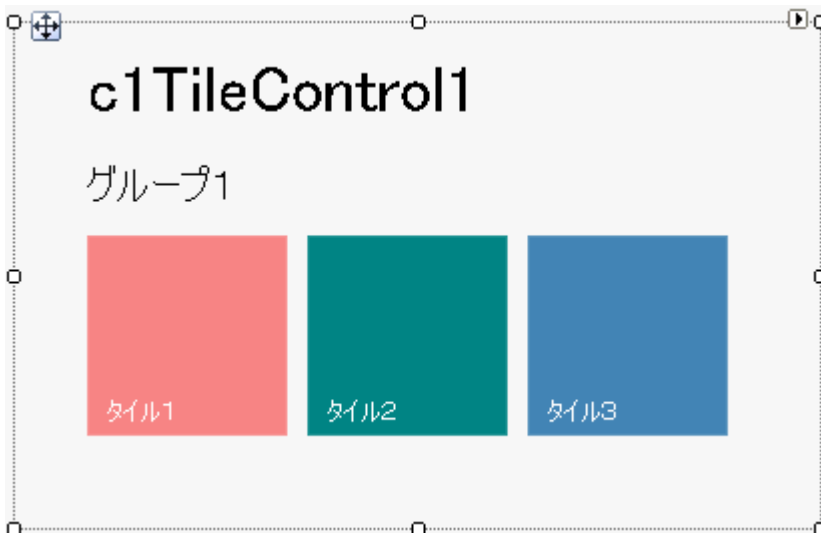


## TileControl の背景色の設定

TileControl の BackColor は、プログラムによるか設計時に TileControl に適用できます。プロパティウィンドウを使用し、TileControl の背景色を設定するには、以下を実行します。

1. **C1TileControl** を右クリックし、[**プロパティ**]を選択します。
2. **BackColor** プロパティを **LightYellow** に設定します。

### ✔ このトピックの作業結果



## 特定グループへのタイルの追加

タイルは、設計時かプログラムによって作成できます。設計時、タイルは **C1TileControl.Groups コレクションエディタ**と **Group.Tiles コレクションエディタ**を使用して作成できます。また、タイルはプログラムで **Tile** クラスから作成することもできます。タイルは、**InsertItem** メソッドを使用し、**TilesCollection** の指定したインデックスに挿入できます。その後、**Group.Tiles** プロパティを使用し、タイルを適切なグループに追加できます。

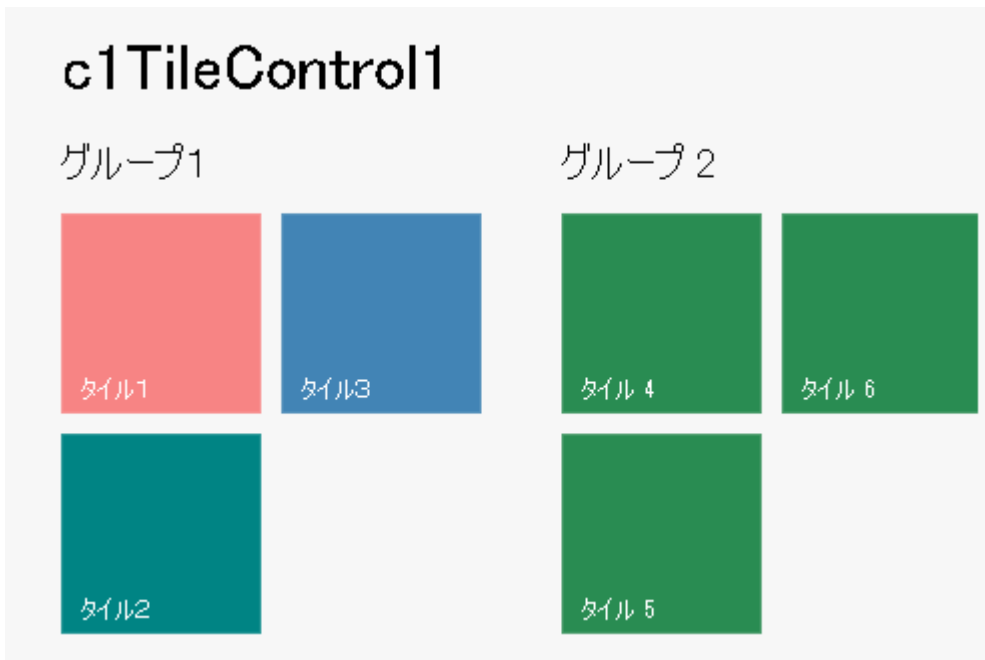
### 設計時の場合

設計時にテンプレートを追加するには、以下を実行します。

1. **C1TileControl** を右クリックし、[**グループの編集**]を選択して、[**C1TileControl.Groups コレクションエディタ**]を開きます。
2. [メンバ]リストから、タイルを追加するグループ(**Group2** など)を選択します。
3. Tiles コレクションの横にある<...>ボタンをクリックします。
4. 3回クリックして、3つの**タイル**を2番目のグループ(**group2**)に追加します。
5. <OK>をクリックし、[**Group.Tiles コレクションエディタ**]を保存して閉じた後、<OK>をクリックし、[**C1TileControl.Groups コレクションエディタ**]を保存して閉じます。

### ✔ このピックの作業結果

3つのタイルがデフォルトで水平に表示され、すべて同じデフォルトの濃い緑色になっています。



## タイルの CheckMark の作成

**Checked** プロパティを True に設定して、チェックマークをタイルの右上隅に表示できます。

### 設計時の場合

1. チェックマークを特定のタイルに適用するには、**tile2** を選択し、Checked プロパティの横にあるチェックボックスを ON にします。

C1TileControl タスク	
<b>tile1:</b>	
タイルのテキスト	<input type="text" value="タイル1"/>
ツールチップのテキスト	<input type="text"/>
タイルのグループ	<input type="text" value="group1"/> ▼
タイルのインデックス	<input type="text" value="0"/>
水平方向のサイズ	<input type="text" value="1"/>
垂直方向のサイズ	<input type="text" value="1"/>
IntValue	<input type="text" value="0"/>
<input checked="" type="checkbox"/> チェック済み	
テンプレート	<input type="text" value="(既定)"/> ▼
<a href="#">タイルテンプレートの編集</a>	
<a href="#">グループの選択</a>	
<a href="#">C1TileControl の選択</a>	

2. **C1TileControl** を右クリックし、[プロパティ]を選択します。
3. Appearance ノードの下で、**CheckMarkColor** プロパティを **Silver** に設定します。

### ✔このトピックの作業結果

チェックマークが tile2 の右上隅に表示されます。



## 特定タイルのサイズの拡大

タイルのサイズは、HorizontalSize と VerticalSize プロパティを使用して拡大できます。

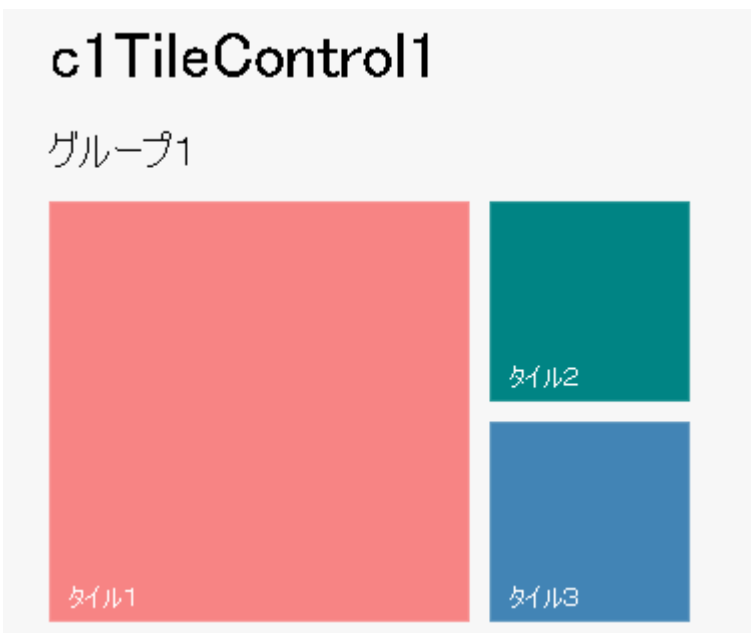
### 設計時の場合

設計時にタイルのサイズを拡大するには、以下を実行します。

1. グループを選択し、[C1TileControl – Group タスク]メニューから、[タイルの編集]をクリックします。
2. [Group.Tiles コレクションエディタ]で、tile1[Tile 1] を選択し、HorizontalSize を2 に設定して、VerticalSize を2に設定します。
3. <OK>をクリックし、[Group.Tiles コレクションエディタ]を保存して閉じます。

### ✔このトピックの作業結果

Tile 1 のサイズが拡大し、他の2つのタイルの2倍の大きさに表示されます。



## タイルへの画像要素の追加

以下のタスクでは、画像をタイルに追加するさまざまな方法を示します。

## タイルへのシンボルの追加

シンボルをタイルに追加するには、以下を実行します。

1. **C1TileControl** を右クリックし、[**テンプレートの編集**]を選択します。[**C1TileControl.Templates コレクションエディタ**]が表示されます。
2. <追加>をクリックし、テンプレートを追加します。
3. **Elements コレクション**プロパティの横にある<...>ボタンをクリックします。[**Template.Elements コレクションエディタ**]が表示されます。
4. [追加]のドロップダウンリストボックスから、**ImageElement** を選択します。
5. [**シンボル**]のドロップダウンリストボックスから、**Library** を選択します。
6. **ImageSelector** プロパティのドロップダウンリストボックスから、**Symbol** を選択します。これにより、Symbol プロパティの値が指定したタイルにバインドされます。
7. <OK>をクリックし、[**Template.Elements コレクションエディタ**]を保存して閉じます。
8. **C1TileControl** を右クリックし、[**グループの編集**]を選択します。
9. Tiles の横にある<...>ボタンをクリックします。[**Group.Tiles コレクションエディタ**]が表示されます。
10. [メンバ]リストから、1番目のタイルを選択します。
11. **Symbol** プロパティを **Home** に設定し、**Template** プロパティを **template1** に設定します。

### ✔ このピックの作業結果

ホームシンボルが1番目のタイルに適用されています。



## 実行時の画像の描画

実行時に **Paint** イベントを使用し、次のように、画像を描画できます。

1. **C1TileControl** を右クリックし、[**テンプレートの編集**]を選択します。
2. <追加>をクリックし、テンプレートを **C1TileControl** に追加します。
3. 次のコードをプロジェクトに追加し、実行時に画像を描画します。

### Visual Basic コードの書き方

#### Visual Basic

```
Private Sub template1_Paint(sender As Object, e As
C1.Win.C1Tile.TemplatePaintEventArgs)
    Dim g As Graphics = e.Graphics
    g.SmoothingMode = System.Drawing.Drawing2D.SmoothingMode.HighQuality
    Dim rect As Rectangle = e.ClipRectangle
    rect.X += (rect.Width - 28) \ 2
    rect.Y += (rect.Height - 28) \ 2
    rect.Width = 28
    rect.Height = 28
    Dim brush As Brush = New SolidBrush(e.Tile.GetBackColor())
    Dim pen As New Pen(e.Tile.GetForeColor())
    Select Case e.Tile.IntValue
```

```

        Case 1
            g.FillPie(brush, rect, 50F, 270F)
            g.DrawPie(pen, rect, 50F, 270F)
            Exit Select
        Case 2
            g.FillRectangle(brush, rect)
            g.DrawRectangle(pen, rect)
            Exit Select
        Case Else
            g.FillEllipse(brush, rect)
            g.DrawEllipse(pen, rect)
            Exit Select
    End Select
    brush.Dispose()
    pen.Dispose()
End Sub

```

### C# コードの書き方

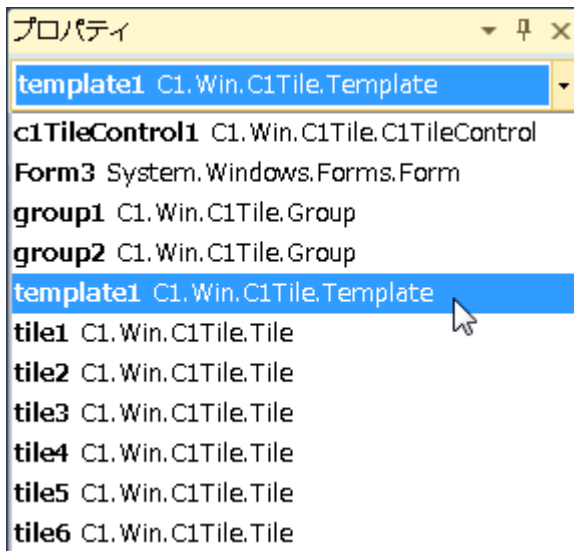
```

C#
private void template8_Paint(object sender, C1.Win.C1Tile.TemplatePaintEventArgs e)
{
    Graphics g = e.Graphics;
    g.SmoothingMode = System.Drawing.Drawing2D.SmoothingMode.HighQuality;
    Rectangle rect = e.ClipRectangle;
    rect.X += (rect.Width - 28) / 2;
    rect.Y += (rect.Height - 28) / 2;
    rect.Width = 28;
    rect.Height = 28;
    Brush brush = new SolidBrush(e.Tile.GetBackColor());
    Pen pen = new Pen(e.Tile.GetForeColor());
    switch (e.Tile.IntValue)
    {
        case 1:
            g.FillPie(brush, rect, 50f, 270f);
            g.DrawPie(pen, rect, 50f, 270f);
            break;
        case 2:
            g.FillRectangle(brush, rect);
            g.DrawRectangle(pen, rect);
            break;
        default:
            g.FillEllipse(brush, rect);
            g.DrawEllipse(pen, rect);
            break;
    }
    brush.Dispose();
    pen.Dispose();
}

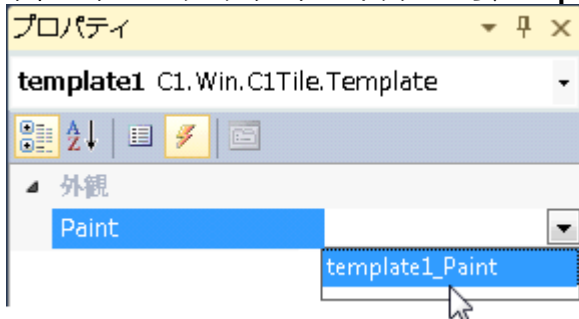
```

4. デザインビューで、**C1TileControl** を右クリックし、[プロパティ]を選択します。
5. ウィンドウのドロップダウンリストボックスから、**Template1** を選択します。

# TileControl for WinForms



6. ウィンドウのドロップダウンリストボックスから、**Template1** を選択します。



7. tile1 を選択し、その[C1TileControl **タスク**]メニューを表示して、各プロパティを次のように設定します。
- **Horizontal Size** を1に設定。
  - **Vertical Size** を1に設定。
  - **IntValue** を0に設定。これにより、1番目の描画が1番目のタイルに適用されます。
  - **Template** を **template1** に設定。
8. tile2 を選択し、[C1TileControl **タスク**]メニューを表示して、各プロパティを次のように設定します。
- **Horizontal Size** を1に設定。
  - **Vertical Size** を1に設定。
  - **IntValue** を1に設定。これにより、2番目の描画が2番目のタイルに適用されます。
  - **Template** を **template1** に設定。
9. tile2 を選択し、[C1TileControl **タスク**]メニューを表示して、各プロパティを次のように設定します。
- **Horizontal Size** を1に設定。
  - **Vertical Size** を1に設定。
  - **IntValue** を2に設定。これにより、3番目の描画が3番目のタイルに適用されます。
  - **Template** を **template1** に設定。
10. プロジェクトを実行し、タイル上に作成された描画を観察します。

