

ビジネスコンポーネントの使用例

GeneXus™

この資料では、ビジネスコンポーネントの具体的な利用ケースを例を利用し、説明します。

ユーザーからの新規要望

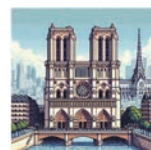
パリ



~~モニュメント~~



~~美術館~~



~~有名なランドマーク~~

観光地

~~遺跡~~

~~遺跡~~

北京



本資料では、ユーザーによる次の要望を想定し、説明を行います。

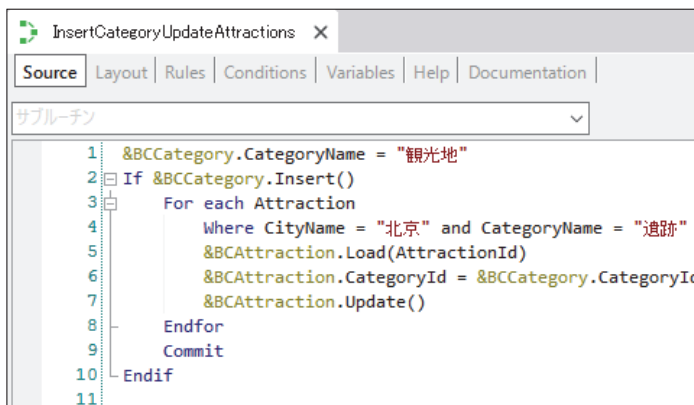
“指定された都市に記録された観光名所は、登録時のカテゴリに関係なく、新しい「観光地」という特殊なカテゴリとなるように変更したい”

ただし、まずは北京を対象にし、意図した結果になるかをテストする機能として実装し、確認後、データを元に戻せる実装が要望されました。

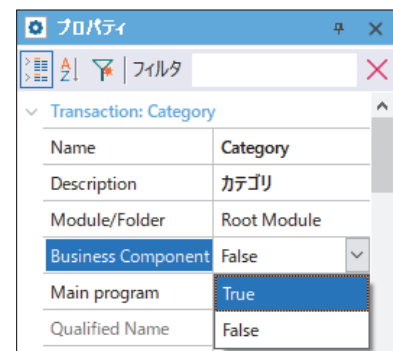
この要望は、現在実装済みのアプリケーション上で、1件ずつ操作を行うのであれば、対応可能です。

しかし、件数が多ければ、多くの工数を必要とし、見落とす可能性も考えられるため、この要望を満たす画面を追加することが要望されています。

プロシージャオブジェクトの実装



名前	タイプ	Is Collection	デスクリプション
Variables			
Standard Variables			
• BCCategory	Category	<input type="checkbox"/>	BCCategory
• BCAttraction	Attraction	<input type="checkbox"/>	BCAttraction



ビジネスコンポーネントの利用を説明した資料と同様に、プロシージャオブジェクトで実装を行います。

特別なカテゴリについても、この処理の中で登録したいという要望があったため、Category トランザクションもビジネスコンポーネントとして利用できるように変更しています。

カテゴリの登録に成功した場合、For each コマンドを利用し、ATTRACTION テーブルの内容を参照します。

この時、条件として都市名が北京と一致する観光名所を対象にします。

該当したレコードの AttractionId 項目属性で取得できる値を利用し、ビジネスコンポーネントの更新対象となるレコードの読み込みを行います。

CategoryId は、自動採番が有効になっているため、ビジネスコンポーネントの新規登録処理で、決定した値は、BCCategory 変数の CategoryId メンバーを参照することで、利用できます。

Update メソッドで更新を行い、同じ処理を該当するレコードの件数分繰り返します。

この時、次の更新の前に New 演算子の実行は必要ありません。

Update メソッドを実行した時点で、変数内の値は、テーブルの更新に利用され、変数内に維持する必要はなく、繰り返し処理における次の Load メソッドによって、変数内の値すべては、次のレコードの内容で、上書きされるためです。

Web パネルオブジェクトの実装

The screenshot displays the GeneXus IDE interface for implementing a Web Panel Object. On the left, the 'Web Layout' tab for 'CategoryAndAttractions' shows a 'MainTable' with a text block 'カテゴリ「観光地」を登録し、北京の観光名所すべてに適用します' and a button 'カテゴリ変更'. On the right, the 'Events' tab for the 'Do' event shows a call to 'InsertCategoryUpdateAttractions()'. Below that, the 'Source' tab shows the code for 'InsertCategoryUpdateAttractions()'.

```

1 &BCCategory.CategoryName = "観光地"
2 If &BCCategory.Insert()
3   For each Attraction
4     Where CityName = "北京" and CategoryName = "遺跡"
5     &BCAttraction.Load(AttractionId)
6     &BCAttraction.CategoryId = &BCCategory.CategoryId
7     &BCAttraction.Update()
8   Endfor
9   Commit
10 Endif
11

```

要望として、画面の追加が含まれていたため、Web パネルオブジェクトを作成し、画面を実装します。

このオブジェクトについては、改めて詳細に説明を行うため、この資料では、必要な範囲のみ補足しています。

[Web Layout] エlementでは、テキストブロックコントロールを配置し、この画面の役割を記述しています。

その下にボタンコントロールを配置し、イベントは、Do という名前にしました。ボタンコントロールを右クリックし、「イベントへ移動」をクリックし、イベントの定義を行います。

Do イベントでは、先ほど定義したプロシージャオブジェクトを呼び出すため、オブジェクト名を記述し、最後に括弧を追記します。

これで、ボタンをクリックすることで、カテゴリの追加と、一部の観光名所のカテゴリが変更されます。

現状の確認と更新の実行

観光名所s

追加

Q 観光名所名

国名 中国 ^

中国

観光名所番号	観光名所名	国番号	国名	都市番号	都市名	カテゴリ番号	カテゴリ名	観光名所写真	
14	紫禁城	3	中国	1	北京	2	遺跡		更新 削除
13	万里の長城	3	中国	1	北京	2	遺跡		更新 削除

選択リスト カテゴリ

カテゴリ番号

0

カテゴリ名

カテゴリ番号

1

美術館

カテゴリ番号

2

遺跡

カテゴリ番号

3

モニュメント

カテゴリ番号

4

有名なランドマーク

終了

カテゴリ「観光地」を登録し、北京の観光名所すべてに適用します

カテゴリ変更

観光名所の一覧で、国名のフィルタを利用し、中国で絞り込むと、都市が北京となっている 2 件のみが表示されることが確認できます。
また、カテゴリも登録時の遺跡のままであることが確認できます。

同様に、カテゴリの選択画面から登録済みのカテゴリー一覧を確認すると、「観光地」というカテゴリは登録されていないことが確認できます。

では、実際に新規に作成した Web パネルを実行し、表示さる「カテゴリ変更」ボタンをクリックしてみます。

実行結果

観光名所s

追加

Q 観光名所名

国名 中国 ^

中国

観光名所番号	観光名所名	国番号	国名	都市番号	都市名	カテゴリ番号	カテゴリ名	観光名所写真	
14	紫禁城	3	中国	1	北京	5	観光地		更新 削除
13	万里の長城	3	中国	1	北京	5	観光地		更新 削除

選択リスト カテゴリ

カテゴリ番号

0

カテゴリ名

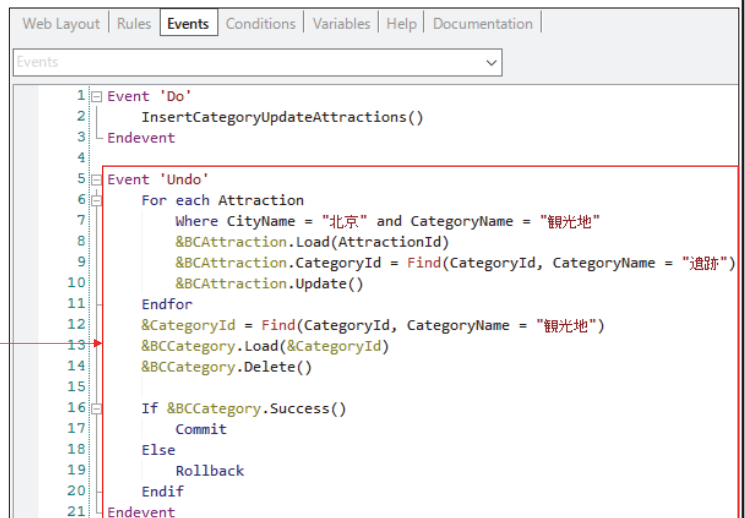
カテゴリ番号	カテゴリ名
✓ 1	美術館
✓ 2	遺跡
✓ 3	モニュメント
✓ 4	有名なランドマーク
✓ 5	観光地

終了

観光名所一覧画面を再読み込みし、国名のフィルタで中国と絞り込むと、登録されている2件の観光名所が表示され、カテゴリ番号、カテゴリ名が更新されたことを確認できます。

もちろん、カテゴリの登録画面で、選択リンクをクリックして、表示されるダイアログの内容にも新しいカテゴリが追加されていることが確認できました。

Undo イベントの実装



今回の要望では、テストを行い、確認ができた場合には、データを元に戻す機能が必要となっていました。

そのため、Undo というイベントを紐づけたボタンを配置し、このイベントで、変更した内容を元に戻す実装を行います。

この場合もビジネスコンポーネントを利用しますが、プロシージャオブジェクトに処理は追記しません。

ビジネスコンポーネントについての説明をした別の資料で、ビジネスコンポーネントを利用可能なオブジェクトは、プロシージャオブジェクトだけでないことを記載していました。

そして、この Web パネルオブジェクトもビジネスコンポーネントを利用可能なオブジェクトの 1 つであるため、直接データ操作の実装を行います。

元に戻す必要のある変更点は 2 つあります。

1 つは、変更した観光名所のカテゴリ、もう 1 つは追加したカテゴリです。

まずは、カテゴリの変更を元に戻すため、For each コマンドを記述し、都市が「北京」で、カテゴリ名が「観光地」となっているレコードが対象となるように実装を行います。

この処理の中では、ビジネスコンポーネントを利用し、データの更新を行います。

幸い、この 2 つの観光名所は、もともとカテゴリが「遺跡」だったため、Find 関数を利用し、「遺跡」のカテゴリ番号を割り当て、更新します。

続いて、追加されたカテゴリの削除を行います。

はじめに、&CategoryId という変数に、Find 関数を利用し、追加された「観光地」のカテゴリ番号を取得します。

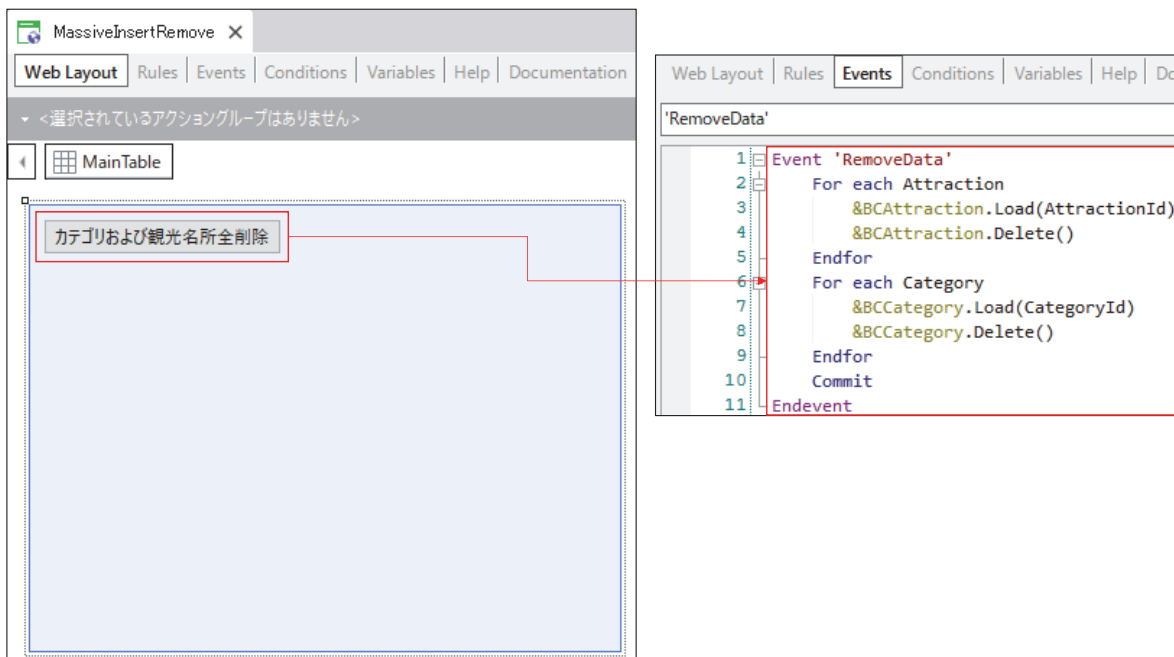
そして、ビジネスコンポーネントを利用した削除を行うため、このカテゴリ番号でデータを読み込み、Delete メソッドを実行します。

処理の実行順は、変更を行った際と逆の順序で実行しました。

これは、参照整合性のエラーを発生させないためでした。

もし、先にカテゴリを削除した場合、そのカテゴリを参照している観光名所があるため、操作は失敗に終わります。

データの削除



ユーザーから追加で要望があった想定で、もう 1 つ画面を作成します。

要望としては、以下の通りです。

“登録済みのカテゴリおよび観光名所を全件削除する指示が出せる画面を利用したい”

この要望を満たすためには、ここまでの要望同様に、ビジネスコンポーネントを利用し、削除する画面を実装します。

新しい Web パネルオブジェクトを作成し、[Web Layout] エLEMENT で、ボタンコントロールを配置し、RemoveData という名前のイベントを紐づけます。

このイベント内には、並行する 2 つの For each コマンドを記述し、1 つは、ATTRACTION テーブルをベーステーブルとし、もう 1 つは、CATEGORY テーブルをベーステーブルとして定義します。

各 For each コマンド内では、ベーステーブルに対応したビジネスコンポーネントを利用し、データを削除するため、Load メソッドで、データを読み込み、Delete メソッドで、削除の指示を行っています。

1 ページ前の元に戻す場合と同様に、観光名所を先に削除し、そのあとにカテゴリを削除する必要があります。

逆の順番で実施した場合、参照整合性エラーが発生します。

また、今回のアプリケーションでは、観光名所を参照するテーブルはないため、問題ありませんが、もし観光名所を参照するテーブルがある場合、参照整合性エラーとなり、一部のデータは削除できません。

*GeneXus*TM

training.genexus.com
wiki.genexus.com