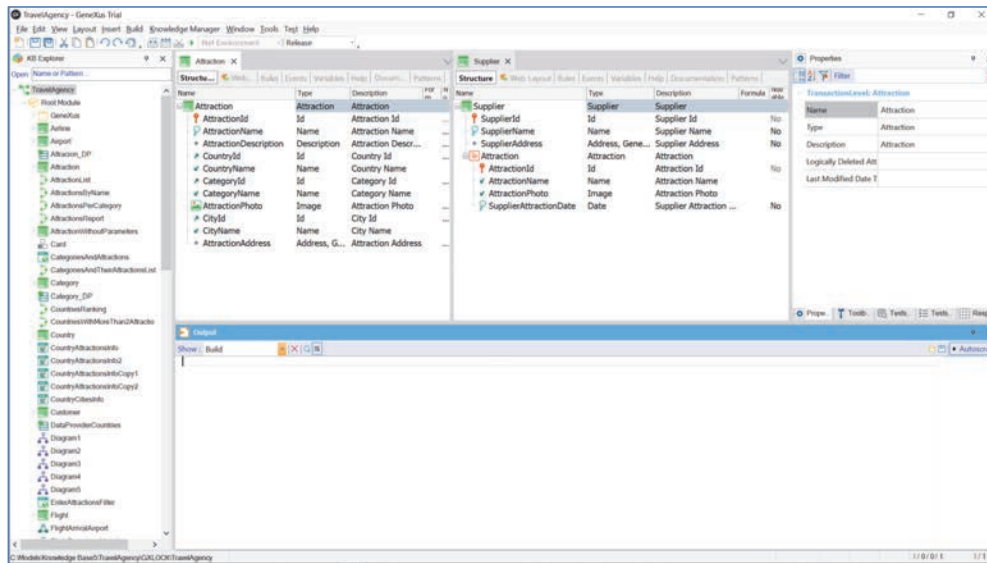


ベーステーブルなしでグリッドに ロードする場合の考慮事項

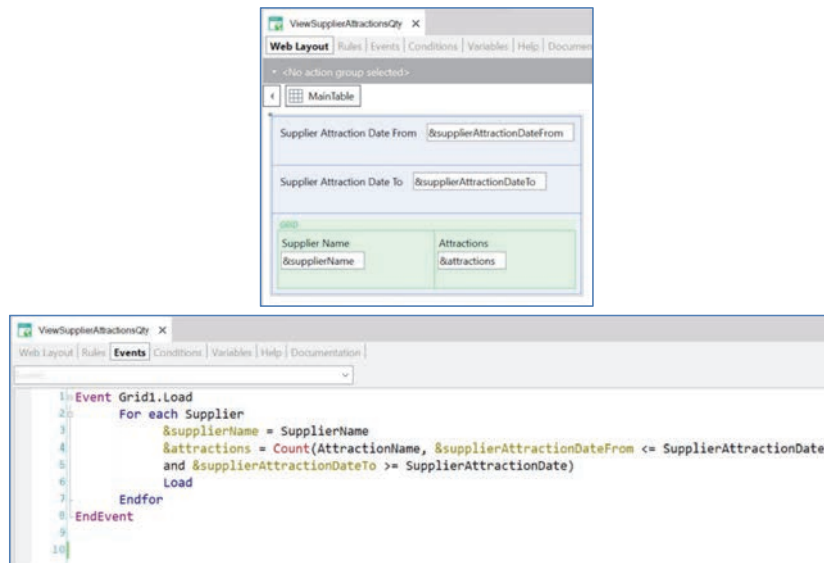
GeneXus[™]



この章で確認する例を実装するには、これまでの章で使用した旅行代理店向けアプリケーションを使用します。

このアプリケーションでは、ほかのものに加えて Attraction トランザクションおよび Supplier トランザクションを使用しています。Supplier トランザクションには第 2 レベルがあり、その項目属性は、Attraction トランザクションの項目属性と、新しい項目属性である SupplierAttractionDate です。この SupplierAttractionDate は、サプライヤーが観光名所を提供する日付を記録するために使用されます。

そのため、各サプライヤーに N 個の観光名所があることになります。



次のような Web パネルを考えます。

固定部分に、SupplierAttractionDateFrom と SupplierAttractionDateTo という Date タイプの 2 つの変数があります。

また、次の 2 つの変数を持つグリッドもあります: supplierName および attractions。supplierName は Supplier テーブルの SupplierName 項目属性に基づきます。attractions は Numeric タイプの変数です。

この Web パネルで次の内容を実装します:

ユーザーは開始日と終了日という 2 つの日付を選択できます。すべてのサプライヤー (Supplier) の名前が、対応する観光名所の数と一緒に画面上に表示されます。各サプライヤーについて、ここでは、ユーザーが入力した日付の範囲に含まれる観光名所の数だけを使用します。

[Events] エLEMENTで、実装方法を確認しましょう。

グリッドの Load イベントの For each で、Supplier テーブルを参照します。

見つかった各レコードについて、SupplierName 項目属性の値が &supplierName 変数に割り当てられます。

&attractions 変数について、インライン式が定義されています。ここでは、サプライヤーの観光名所の日付 (SupplierAttractionDate) が、ユーザーが入力した日付の範囲に含まれる、そのサプライヤーのすべての観光名所をカウントします。

フォームにもグリッドのプロパティにも項目属性が宣言されていないため、これはベーステーブルのない Web パネルです。また、イベントのコンテキスト外にも項目属性はありません。この例では For each 内に項目属性があります。この場合、ベーストランザクションが記載されていないければ、For each が参照するテーブルの決定に役立ちますが、グリッドのベーステーブルの決定には影響しません。また、ほかの項目属性も、Count 式のコンテキスト内にあります。それらの項目属性は、この式がナビゲートするテーブルの決定にのみ役立ちます。

Load コマンドを入力して、見つかった各レコードをグリッドにロードする必要があるのはそのためです。
これをプログラムしているときには、パネルにボタンを追加して、日付が入力された際にクリックしてグリッドを更新できるようにする必要があると思うかもしれません。

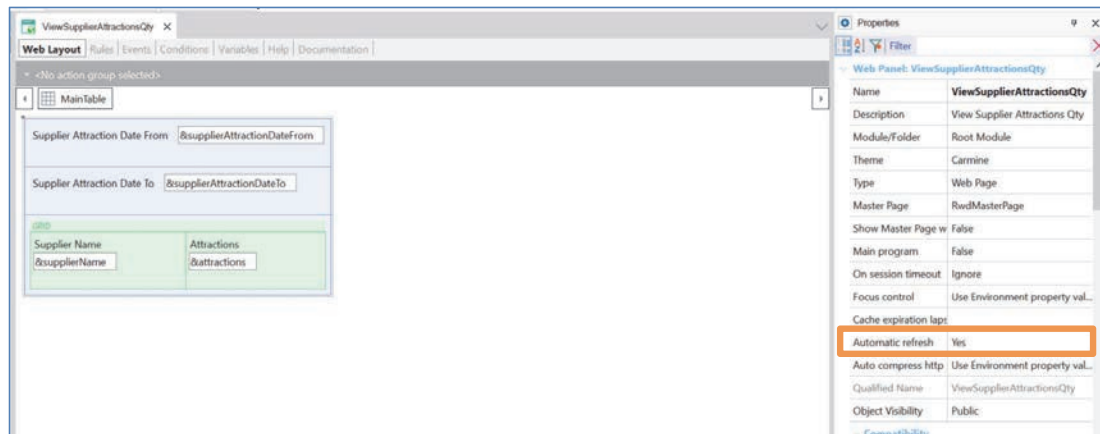
実行時

Supplier Name	Attractions
Transport Travel	5
TipGroup Operator	6
Trafalgar	2
Intrepid Travel	3

この状態で実行してみましょう。

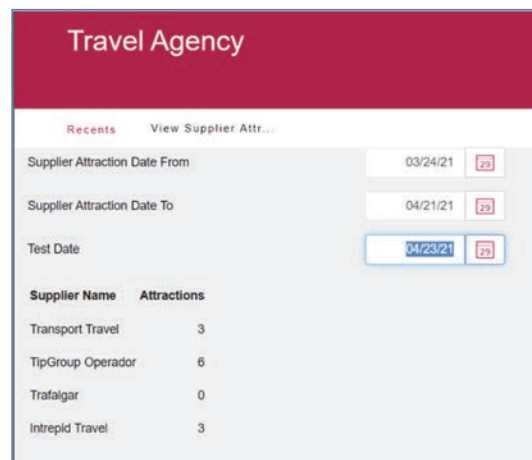
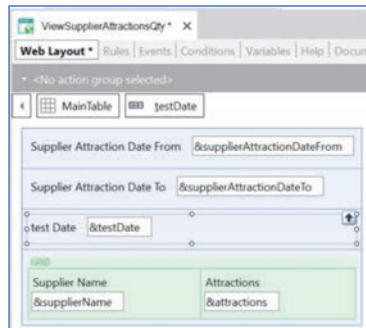
すべてのサプライヤーのリストがロードされ、観光名所の数が 0 と表示されます。これは、観光名所の数をカウントする条件となる日付がまだ入力されていないためです。

開始日および終了日の 2 つの日付を入力します。先ほど確認したように、これらの日付は、各サプライヤーの観光名所をカウントするための条件となります。入力された日付を条件として、各サプライヤーの観光名所の数が自動で更新されます。日付を変更した際にグリッドを更新するよう指示しなくても、自動で更新されました。このような動作になるのはどうしてでしょうか。



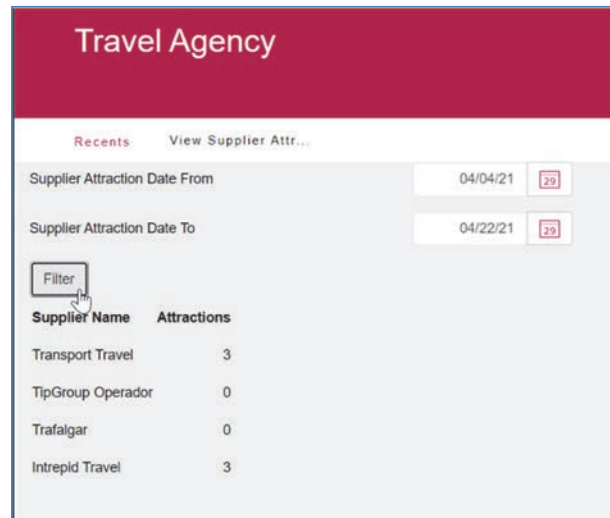
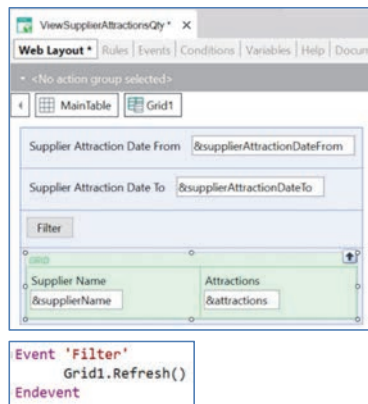
Web パネルに、[Automatic refresh] というプロパティがあります。これは既定で Yes に設定されています。このプロパティによって、[Web Layout] エLEMENTで定義されている変数に対する変更が検出された場合に、グリッドが更新されます。変更された値は、グリッドの Refresh イベントおよび Load イベント、グリッドまたは Web パネルの条件で使用されます。

ここでは、変数 SupplierAttractionDateFrom および SupplierAttractionDateTo が入力され、表示される観光名所の数がフィルタされます。その後、それらの変数はグリッドの Load イベント内で定義されます。より正確には、先ほど示した Count 式内で定義されます。これらの変数はこのイベント内にあるため、表示するデータに影響することがわかります。そのため、一方の変数が変更されるたびにグリッドが更新されます。つまり、グリッドの Refresh イベントがトリガーされます。



Web パネルの固定部分に新しい変数を追加したらどうなるか確認しましょう。ただし、定義する場所は、Load イベント、Refresh イベント、条件のいずれでもありません。

この変数の値を変更する際に、グリッドの Refresh イベントはトリガーされず、グリッドは更新されません。既に説明したように、この場合は [Automatic refresh] プロパティが適用されないためです。



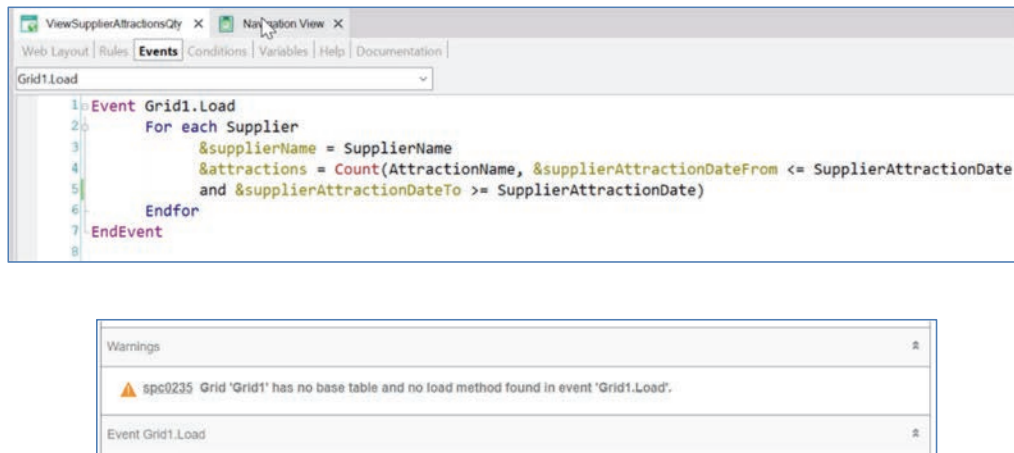
この変数を削除しましょう。

ここでは、[Automatic refresh] プロパティによる動作は不要であるとしてます。一方の変数に変更されるたびにグリッドが更新されるのではなく、希望するタイミングで更新されるようにするためです。

[Automatic refresh] プロパティを No に設定して、パネルの固定部分にボタンを追加することで、グリッドの Refresh イベントをトリガーするタイミングを制御できます。

その場合、ボタンのイベントで、グリッドの Refresh メソッドを定義するだけで対応できます。

この状況では、変数の値を変更してもグリッドは更新されません。ボタンをクリックするとグリッドの Refresh イベントがトリガーされ、グリッドが更新されます。



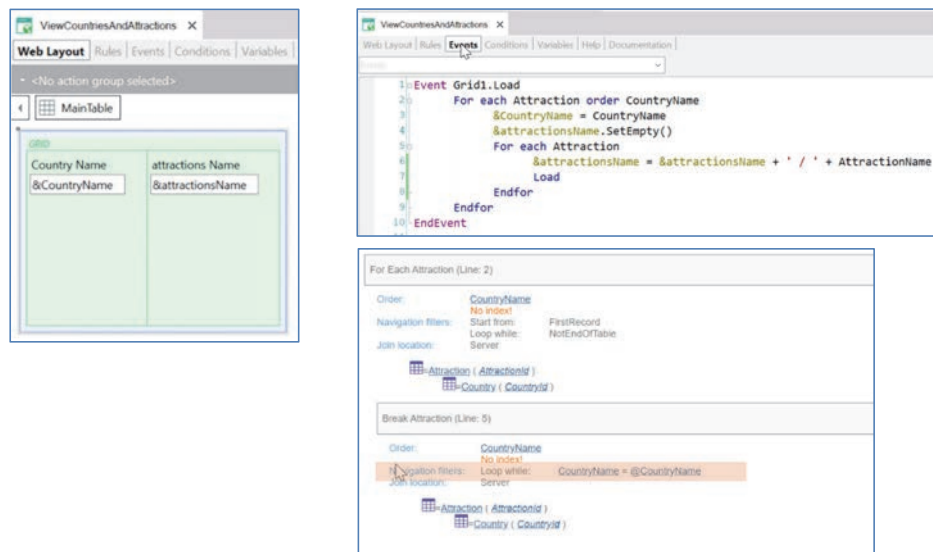
イベントおよびボタンを削除し、このプロパティを再度 Yes に設定しましょう。

ここで、イベント内にこの Load コマンドを宣言することを忘れたとします。どうなるでしょうか。どのような動作になるでしょうか。
この変更を実装して保存しましょう。

ナビゲーション表示を確認します。

グリッドにベーステーブルがなく、Load イベント内に Load コマンドが宣言されていない旨の警告が表示されます。この警告は出力にも表示されます。

これを実行環境でテストすると、グリッドにロードされるレコードはありません。Load コマンドがない場合は、どのレコードもロードされないことが想定されるためです。この実装はあまり意味がないため、警告がトリガーされます。



ここでは、このアプリケーションで、観光名所が 1 つ以上あるすべての国を表示し、すべての観光名所をバーで区切って並べて表示する Web パネルを作成するとします。

Attraction トランザクションには Country トランザクションの CountryId 項目属性および CountryName 項目属性があります。1 つの観光名所は 1 つの国に属します。

それを踏まえ、[Web Layout] エlementには 2 つの変数を持つグリッドがあります。一方は CountryName 項目属性に基づき国名を表示する変数で、もう一方はすべての観光名所を表示する LongVarChar タイプの変数です。

[Events] Elementで、2 つのネストされた For each コマンド内にグリッドの Load イベントが定義されています。

外側の For each コマンドでは、観光名所のテーブルを参照し、国名でレコードを並べ替えています。また、CountryName 項目属性の値を名前用の変数に割り当てています。

その次に、同じ Attraction テーブルをナビゲートするネストされた For each があります。この定義があることで、見つかったすべての観光名所が単一の attractionName 変数内で連結されます。

内側の For each では外側の For each と同じテーブルを参照し、順序は外側の For each で定義されているため、コントロールブレイクが作成されます。ブレイク基準は、外側の For each の order 節で宣言されている項目属性によって正確に決定されます。つまり、CountryName でグループ化されます。

これはナビゲーション表示ではっきり確認できます。

実行時

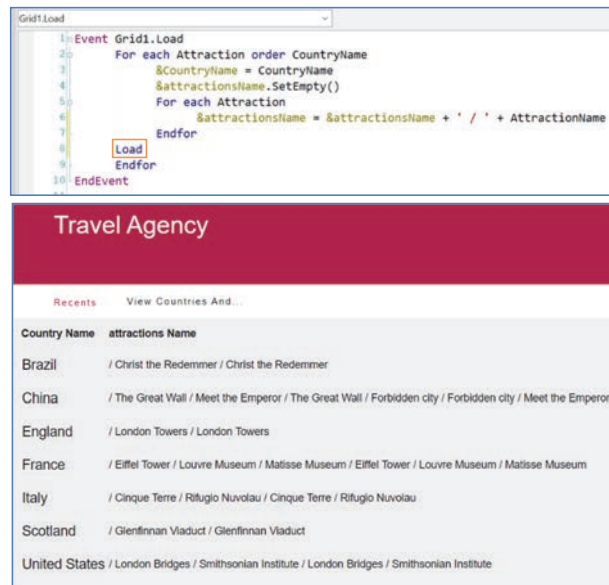
Travel Agency	
Recents	View Countries And...
Country Name	attractions Name
Brazil	/ Christ the Redemmer
Brazil	/ Christ the Redemmer / Christ the Redemmer
China	/ The Great Wall
China	/ The Great Wall / Meet the Emperor
China	/ The Great Wall / Meet the Emperor / The Great Wall
China	/ The Great Wall / Meet the Emperor / The Great Wall / Forbidden city
China	/ The Great Wall / Meet the Emperor / The Great Wall / Forbidden city / Forbidden city
China	/ The Great Wall / Meet the Emperor / The Great Wall / Forbidden city / Forbidden city / Meet the Emperor
England	/ London Towers
England	/ London Towers / London Towers
France	/ Eiffel Tower
France	/ Eiffel Tower / Louvre Museum
France	/ Eiffel Tower / Louvre Museum / Matisse Museum

試してみましょう。

想定どおりの結果は得られません。

たとえば、中国は最初の観光名所の横に出力されます。次の行でも再度出力されますが、最初の 2 つの観光名所が結合されて表示されています。その次の行では、最初の 3 つの観光名所が結合されて表示されています。その後も同様です。すべての国について同様の動作が発生しています。

何が起きているのでしょうか。定義のどこが間違っていたのでしょうか。



ここでは、Load コマンドを宣言する部分が間違っています。この処理は、ネストされた For each コマンド内で行われていました。そのため、観光名所が見つかって変数と結合されるたびに、画面に出力されます。ただし、これは想定する動作ではありません。実際には、その国のすべての観光名所が連結されたら、この情報を表示するようにします。

そのため、この For each コマンドの外側に Load コマンドを定義する必要があります。

試してみましょう。
再度実行します。

想定どおりに動作することを確認できます。
この章で扱ったトピックの詳細については、Wiki を参照してください。