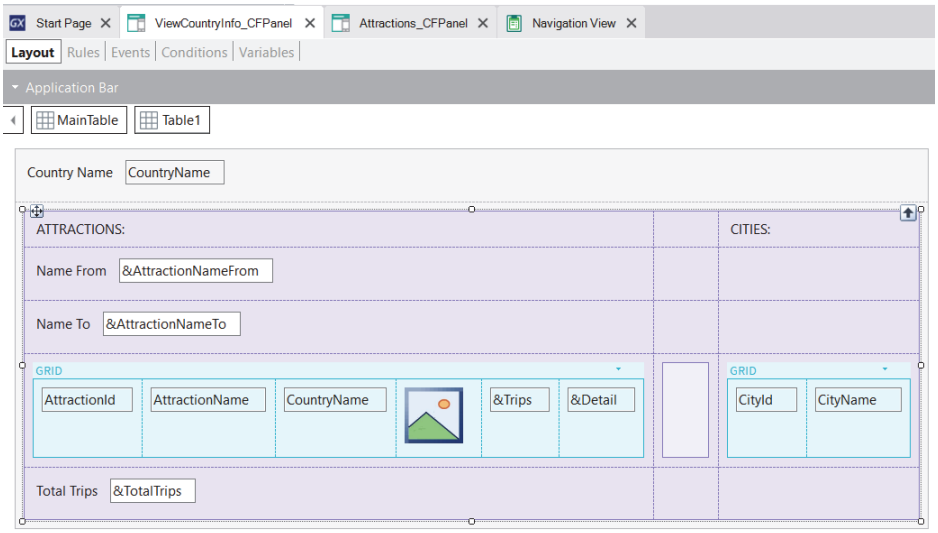
顧客向けに重点を置いた Web 画面

複数のグリッドの使用

GeneXus™

これまで、Panel オブジェクトでグリッドを使用する方法を見てきました。ここでは、パネルに複数のグリッドを追加する場合の考慮事項について説明します。

複数のグリッドを持つ Panel オブジェクト



パラメーターで受け取った国の観光名所と都市を表示する Panel オブジェクトを構築します。

そのためには、Attractions_CFPanel パネルから [名前を付けて保存] を選択し、ViewCountryInfo_CFPanel という名前を付けます。このパネルは、観光名所のリストのパネルで国名がクリックされたときに呼び出されるようにするため、メインオブジェクトの設定は削除します。

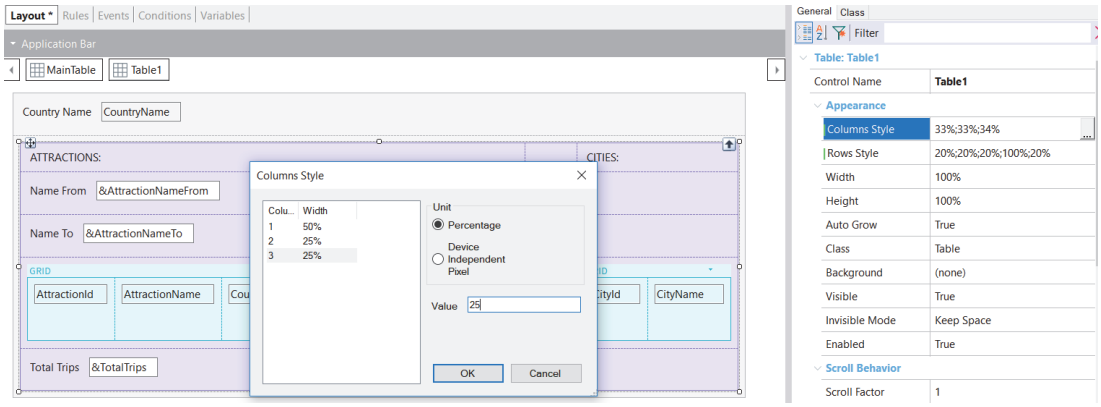
[Rules] エレメントで、CountryId 項目属性を入力パラメーターとした Parm ルールを追加します。

フォームに CountryName 項目属性を追加して、ダイナミック コンボ ボックス タイプの &CountryId 変数を削除します。これは、国を選択するのではなくパラメーターで受け取るためです。

その国の観光名所のデータを左側にまとめ、都市のデータを右側にまとめるには、ツールバーからテーブルコントロールを挿入し、変数 &AttractionNameFrom と &AttractionNameTo、グリッド、および変数 &TotalTrips をテーブル内に配置します。次に、右側に別のテーブルをセパレーターとして挿入し、その右側にグリッドを挿入して、項目属性 CityId および CityName を使用して国の都市が表示されるようにします。

観光名所セクションの上に「ATTRACTIONS」、都市セクションの上に「CITIES」というテキストブロックを配置して、各セクションにタイトルを追加します。クラスを TextBlockTitle に変更します。

テーブルの行と列のスタイルの変更



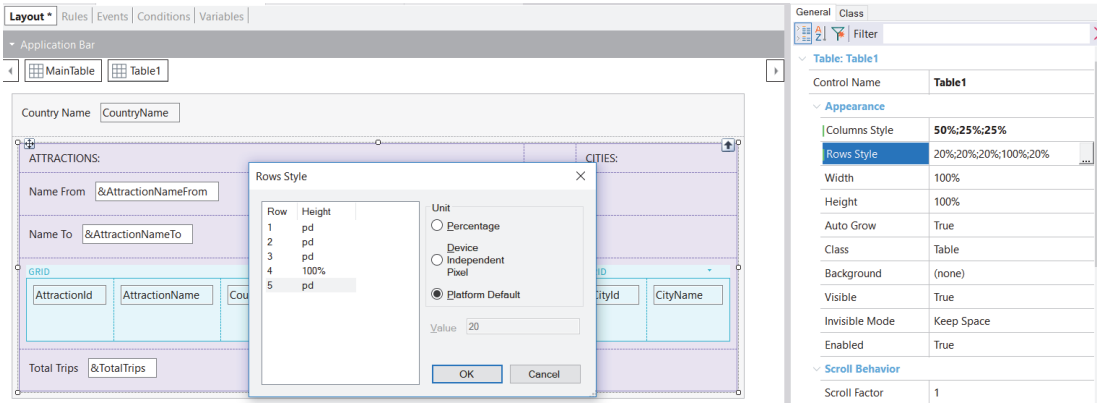
さきほど、観光名所と都市のコンテンツを分けるためのテーブルを挿入しました。テーブルコンテンツの表示方法を定義するには、テーブルの行または列のサイズを変更する必要があります。たとえば、観光名所のコンテンツは都市のコンテンツよりも多いことから、観光名所のコンテンツを表示するテーブルの列にはより多くのスペースを割り当て、対応する行に十分なスペースを確保して正しく表示されるようにします。

そのために、[Columns Style] プロパティと [Rows Style] プロパティが用意されています。まず、3 つの列、つまり観光名所のデータを含む列、分離用のテーブルを含む列、および都市のデータを含む列を変更します。
[Table1] を選択して [Column Style] プロパティをクリックすると、使用可能なスペースのうち、各列が占める既定の割合がそれぞれ 33% であることが分かります。

列の幅に割り当てる値は、テーブルスペース全体に対する割合 (%) またはデバイスに依存しないピクセル (DIP) で指定できます。この方法では、プラットフォームに依存しない抽象ピクセル単位を割り当てることができ、割り当てた値は、後でアプリケーションの実行時に実際のピクセルに変換されます。各 DIP のピクセル数は、画面のサイズによって異なります。この方法で、標準サイズを使用して多様なサイズの画面に対応することができます。

1 列目には使用可能なスペースの 50% を割り当て、2 列目と 3 列目にはそれぞれ 25% を割り当てるとします。このスペースは、DIP で指定された列の分を割り当てた後に残ったスペースです。言い換えると、割合 (%) のスペースは、合計幅からまず定数 (DIP 単位) を差し引いた上で (今回の例では DIP 単位で指定している列はないので差し引かれる列はありません)、残りのスペースに対し、各割合に応じて各列にスペースが配分されると言えます。

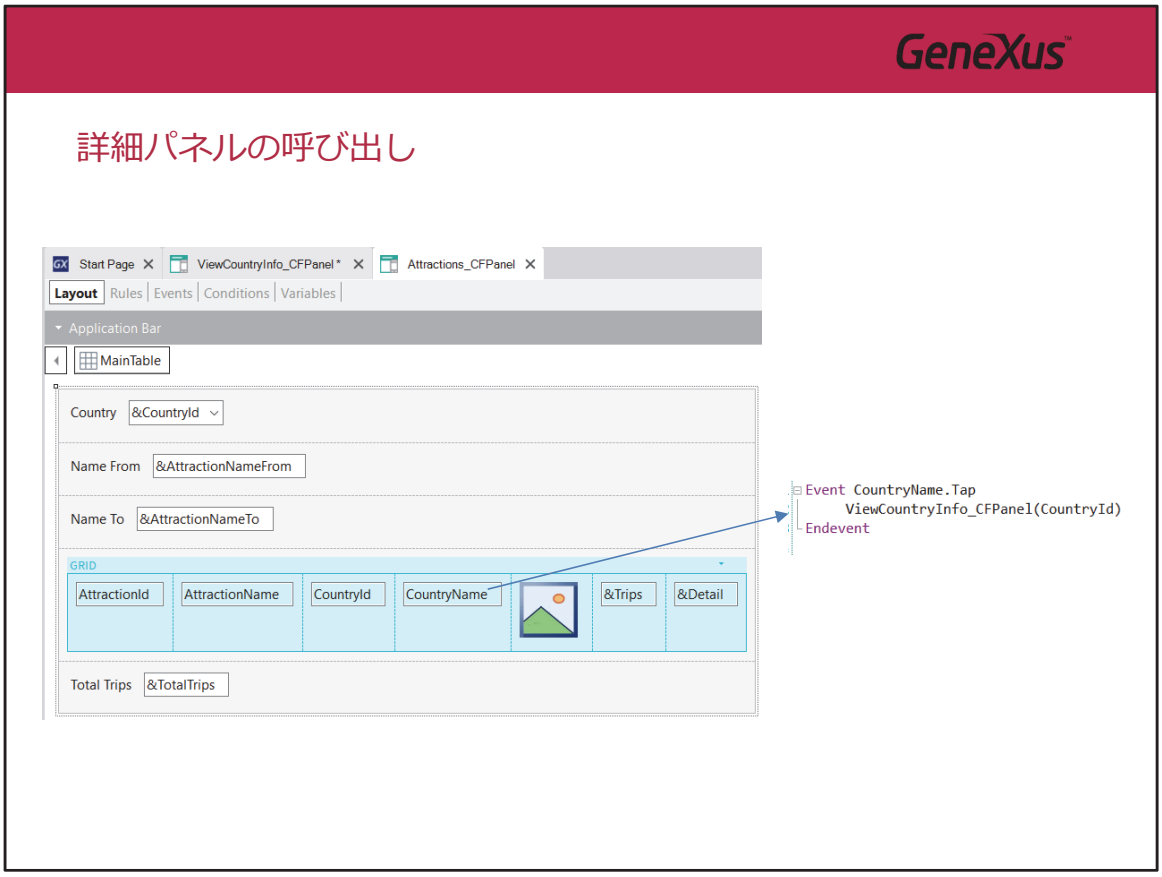
テーブルの行と列のスタイルの変更



次に、行の高さを変更します。[Rows Style] プロパティをクリックすると、グリッドが配置されている 4 行目を除いて、既定ですべての行に 20% が割り当てられていることが分かります。

ここでは、割合 (%)、DIP、およびプラットフォームの既定値を使用して値を割り当てることができます。この最後の値は、「プラットフォームとコンテキストに応じて最適な値を使用する」というものです。つまり、プラットフォームによって異なり、同じプラットフォームであっても、セルのコンテンツによって異なります。

グリッドがある 4 行目を除くすべての行にこの値を割り当てます。4 行目については、使用可能なスペースの 100% を占めるように設定します。



構築したオブジェクトを右クリックしてナビゲーションを表示すると、[出力] ウィンドウにエラーが表示され、ナビゲーション表示でエラーを確認すると、複数のグリッドがある場合は Load イベントをプログラムできない旨が示されます。
[Events] に移動すると、元のオブジェクトと同じように Load イベントがプログラムされていることが分かります。ここでは、この状況を回避するために、前に言及した内容を考慮せずに、汎用イベントではなくグリッドの Load イベントを使用する方がよいと考えられます。

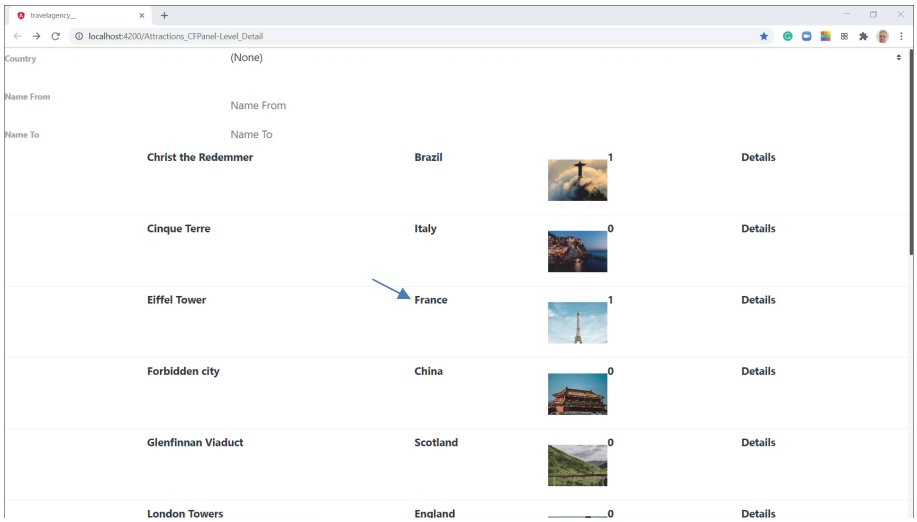
そこで、Load に Grid1 を追加します。[ナビゲーションを表示] を選択すると、問題が解決されたことが分かります。

実行する前に、Attraction_CFPanel パネルに移動し、グリッドで国名をクリックしたときに、国の情報パネルに移動するために必要な CountryId 項目属性を追加します。
次に、CountryIName 項目属性に Tap イベントを追加します。ここで、ViewCountryInfo_CFPanel パネルの呼び出しを記述し、パラメーターとして CountryId を渡します。

実行して結果を確認します。

GeneXus

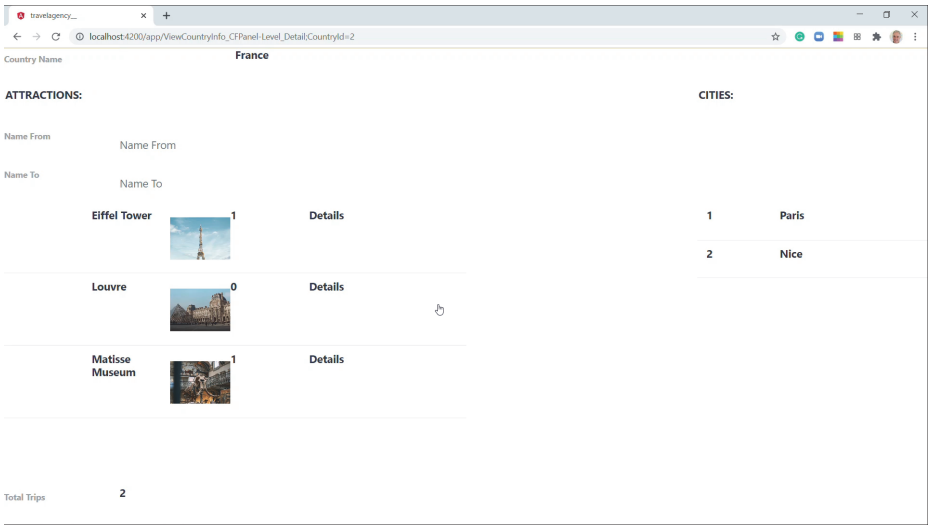
実行時の例



フランスをクリックした場合の結果を確認します。

GeneXus

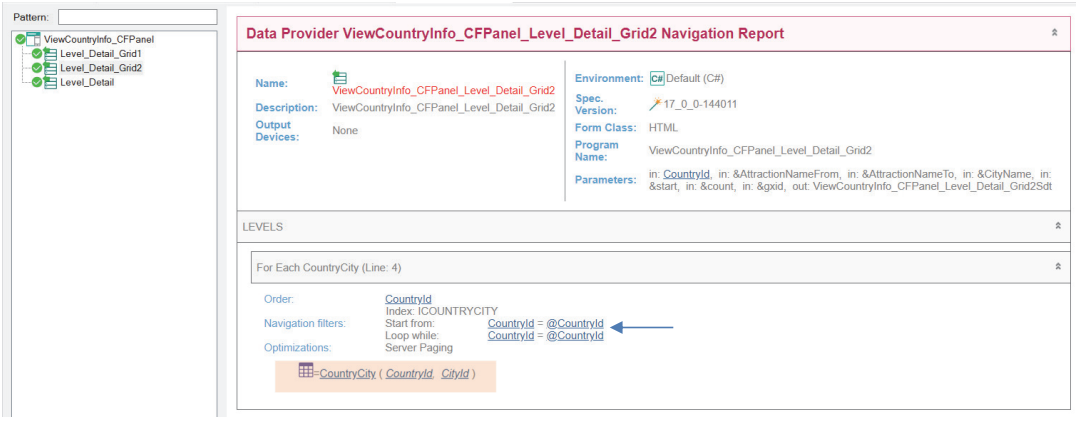
実行時の例



フランスの情報が開き、フランスの観光名所、それらの観光名所を含むツアーの合計、およびフランスへのツアーの合計数が表示されます。右側にはその国の都市が表示されます。

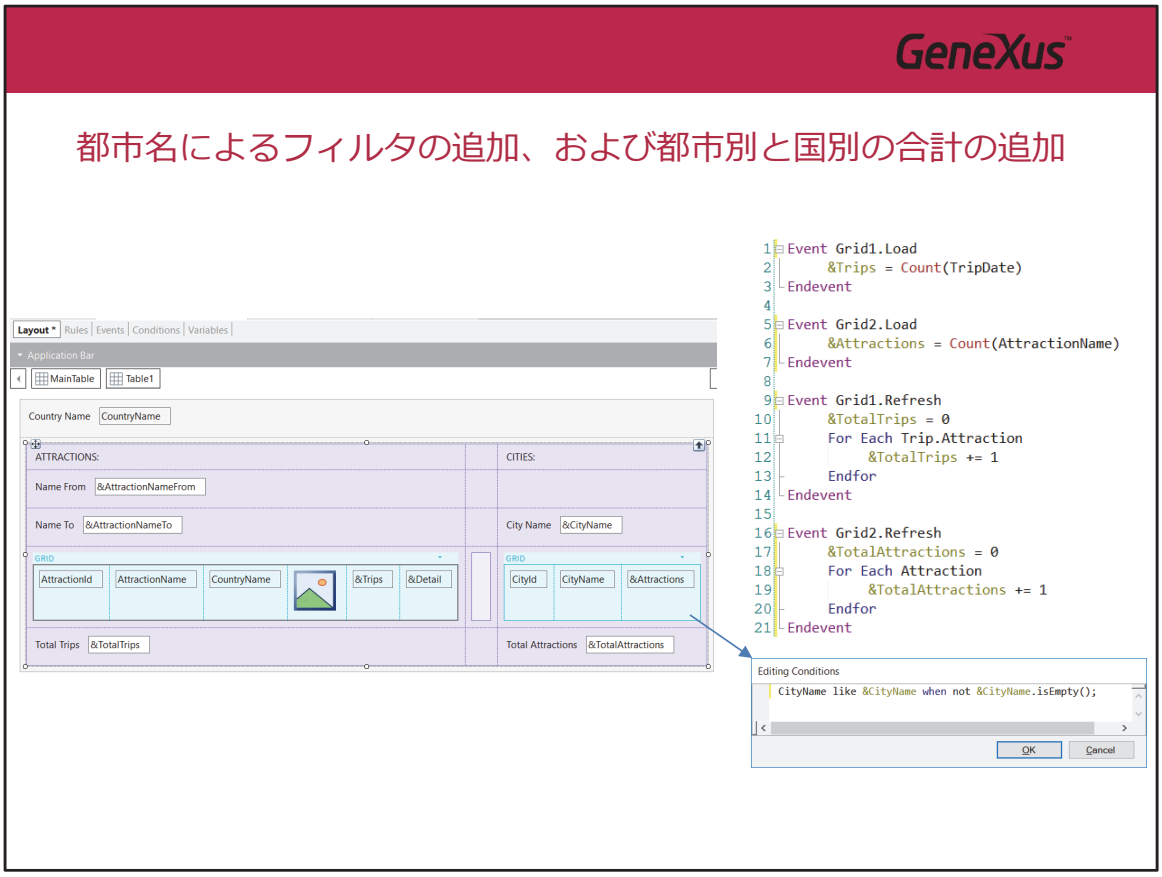
GeneXus

新しいパネルのナビゲーション表示



ナビゲーション表示に移動すると、Grid1 と Grid2 にそれぞれ個別の Level_Detail エントリーがあることが分かります。
Grid1 については、Attractions テーブルへのアクセス、および Attractions_CFPanel パネルを実装したときに取り上げた式のナビゲーションが表示されます。

Grid2 に対応するノードを選択すると、グリッドに表示される都市は CountryCity テーブルにアクセスして取得していること、および Parm ルールで受け取る CountryId 項目属性をフィルタとして使用していることが分かります。



観光名所の合計と名前によるフィルタを表示する場合と同様の方法で、各都市の観光名所の数と国内の観光名所の合計を表示し、都市名によるフィルタを追加します。

[Variables] エレメントで、&Attractions 変数、&TotalAttractions 変数、そして &CityName という変数を作成します。
次に、[Label Position] プロパティを None に設定して、&Attractions 変数をグリッドに追加します。その後、グリッドの下に &TotalAttractions を追加し、都市のグリッドの上にフィルタ変数 &CityName を追加します。フィルタに必要な条件をグリッドに追加しましょう。

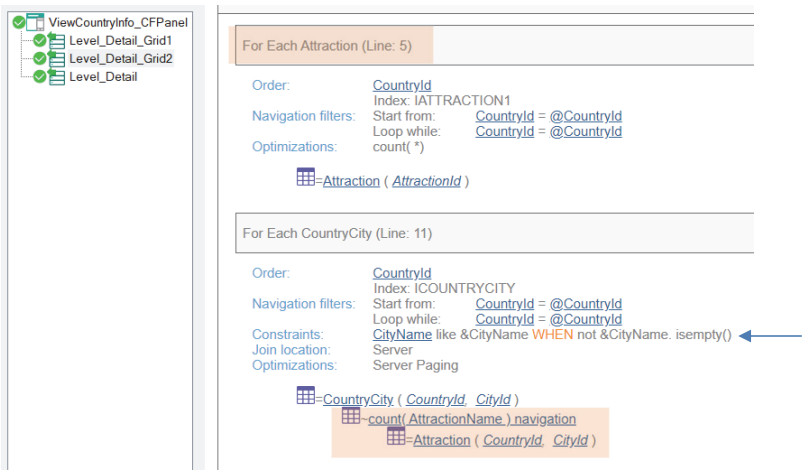
その前に、Country.City の [Base Transaction] プロパティを割り当てます。
[Conditions] エレメントをクリックし、Like 演算子を使用してフィルタ条件を記述します。この演算子を使用するのは、名前の範囲ではなく、フィルタに入力した名前に似た名前でフィルタリングするためです。

次に、Grid2.Load イベントを追加します。このイベントでは、&Attractions 変数をロードし、AttractionName 項目属性を渡して Count 式の計算を行います。
グリッドのベーステーブルは CountryCity であるため、この式では、各行に対応する都市の観光名所の数のみが計算されます。

観光名所の合計を計算するために、ツアーの合計を計算したときと同じ理由で、Grid2.Refresh イベントを記述しました。このイベントでは、観光名所の数を計算するために、Attractions テーブルに対して For Each コマンドをプログラムし、パラメーターで受け取った国識別子の項目属性でフィルタリングされるようにしました。

この計算は、明細行がロードされるたび、つまり Grid2 の Load イベント内で行います。

新しい変更を反映したパネルのナビゲーション表示

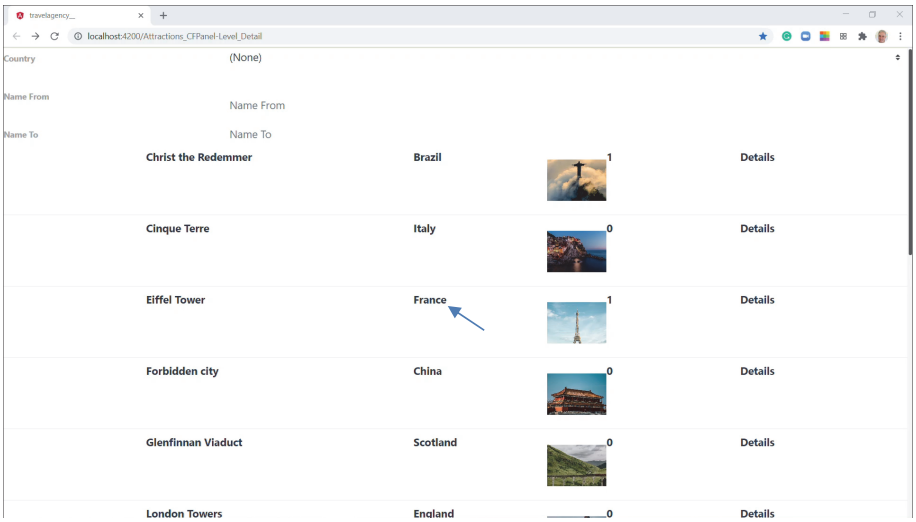


パネルのナビゲーション表示を見ると、ノード Level_Detail_Grid2 に Attraction テーブルに対する For Each コマンドのナビゲーションが表示され、その下に Attraction テーブルに対する Count 式のナビゲーションが表示されます。

そしてここに、追加した CountryName によるフィルタが表示されます。

GeneXus

都市別および国別の合計の表示

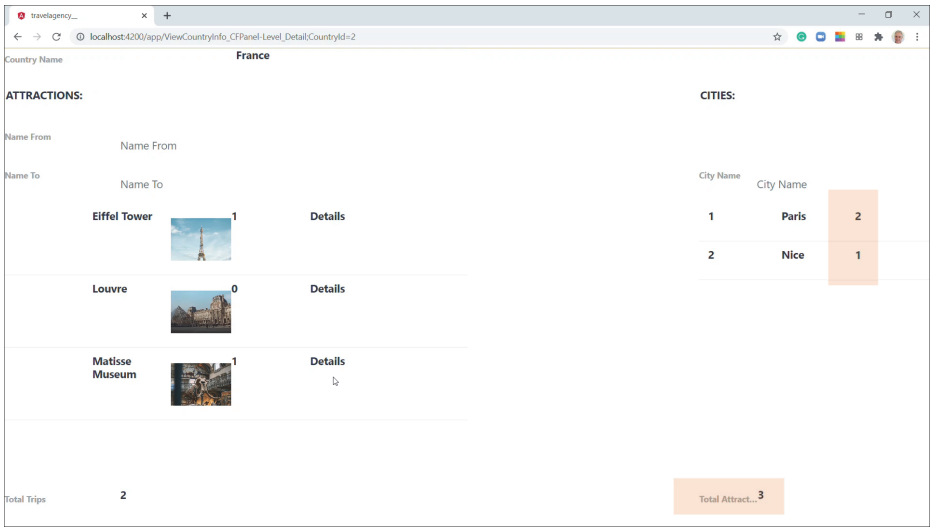


メインオブジェクトである Attractions_CFPanel を実行して結果を確認しましょう。

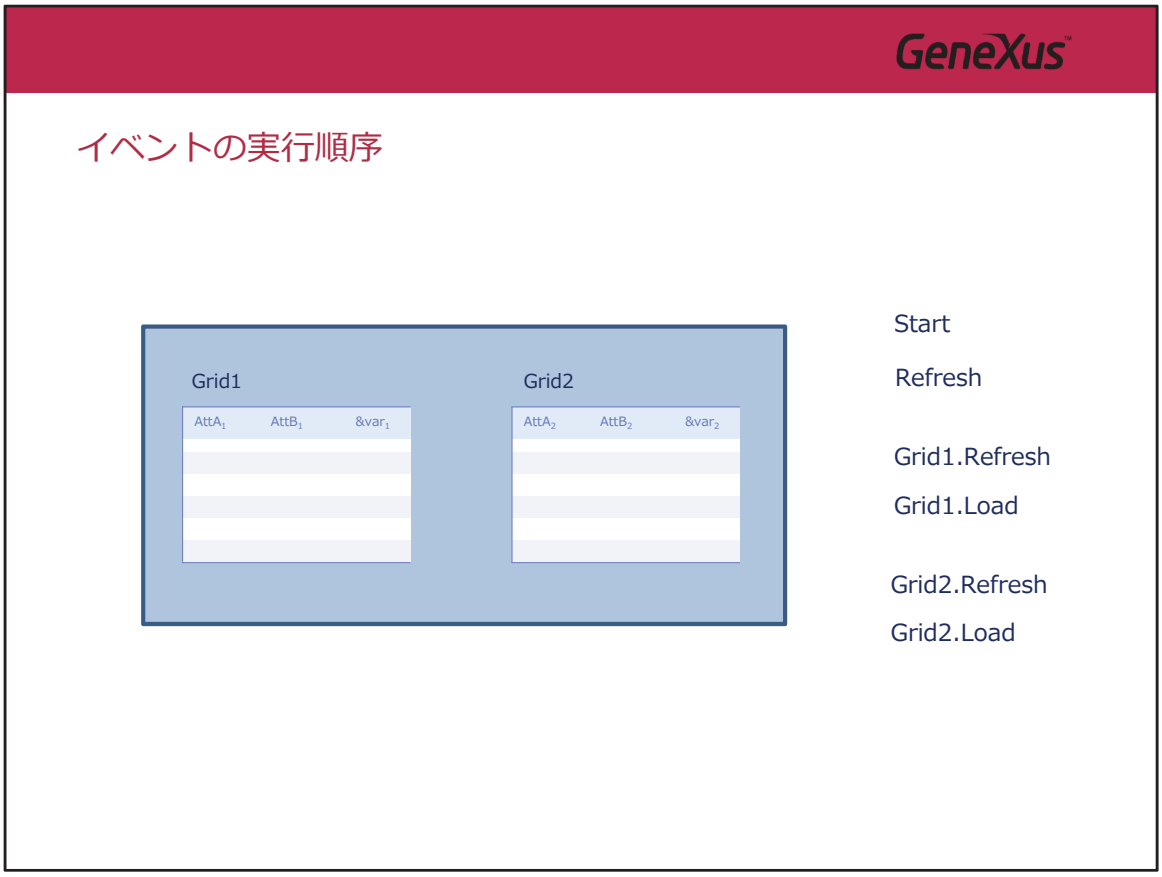
もう一度フランスをクリックします。

GeneXus

都市別および国別の合計の表示



今回は、フランスの各都市の観光名所の合計とフランス国内の観光名所の合計が表示されました。



パネルに複数のグリッドがある場合、各グリッドの Refresh および Load イベントを使用する必要がありました。

では、これらのイベントを追加すると、Panel オブジェクトのイベントとの関連でこれらのイベントはどのような順序でトリガーされるのでしょうか。

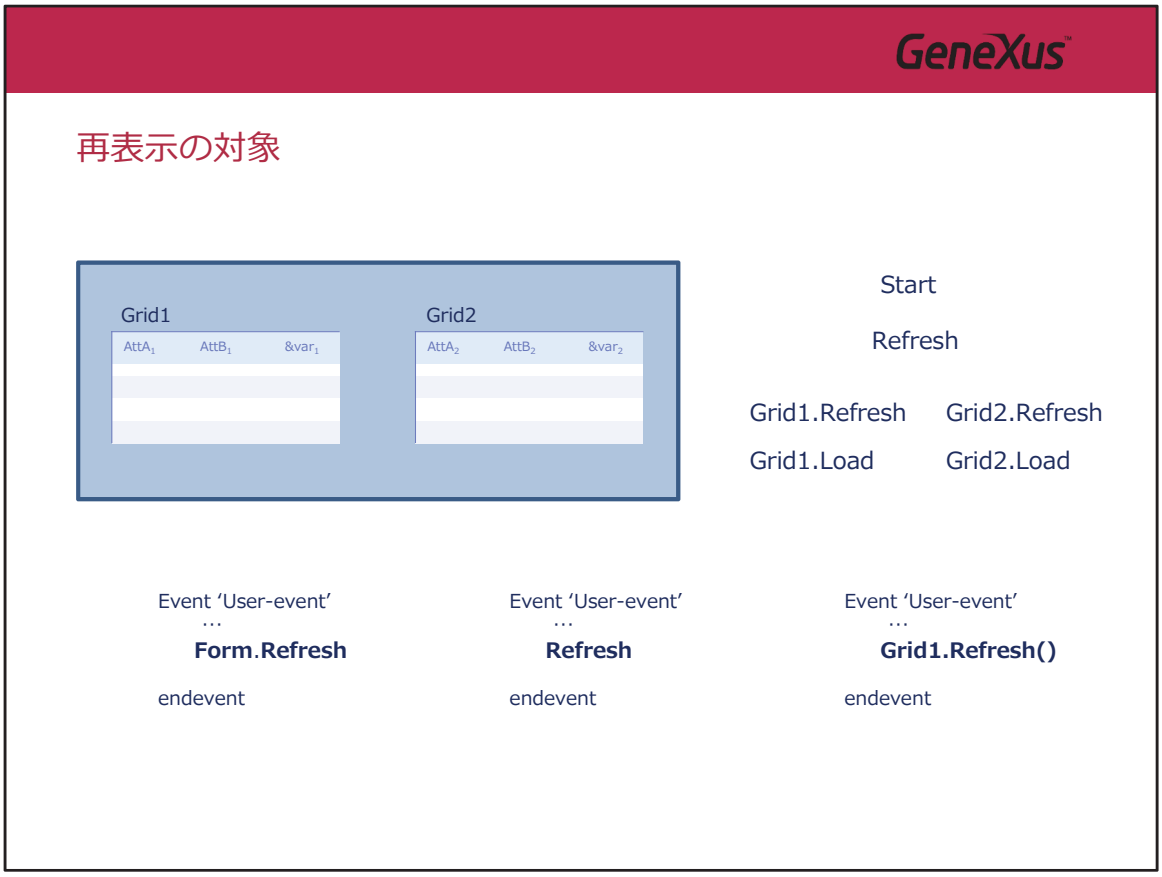
Panel オブジェクトが初めて実行されると、イベント実行は次の順序でトリガーされます:

まず、Start イベントが実行されます (1 回のみ)。

次に、汎用 Refresh イベント、つまりパネル自体の Refresh イベントが実行されます。

最後に、1 つ目のグリッドの Refresh が実行されます。ベーステーブルが存在する場合、グリッドの Load イベントはデータベースから取得されるレコードの数と同じ回数だけ実行され、対応するレコードがフィルタリングされます。ベーステーブルが存在しない場合、グリッドの Load イベントは 1 回だけ実行されます。SDT に基づくグリッドの場合、Load イベントは実行されません。

その後、2 つ目のグリッドの Refresh イベントと Load イベントが同じように実行されます。



複数のグリッドが存在することから、再表示するグリッドを指定するために、Refresh コマンドも特殊化する必要があります。

汎用 **Refresh** コマンド (前に Attractions_CFPanel パネルで使用したもの) を使用すると、汎用 Refresh、および各グリッドの Refresh イベントと Load イベント (つまり、Start イベント以外のすべて) が実行されます。

さらに、グリッドの Refresh メソッドがあり、このメソッドではグリッドのみが再表示されます。つまり、グリッドの Refresh イベントと Load イベントが (グリッドのベーステーブルの有無に応じて n 回または 1 回) 実行されます。ただし、コレクション SDT 変数のグリッドである場合、これらは実行されません。

この章では、Panel オブジェクトで複数のグリッド (このケースでは並列のグリッド) を扱う方法、および各グリッドのイベントを呼び出す際の考慮事項について説明しました。なお、このコースでは、Panel オブジェクト内のネストグリッドに関するトピックは取り上げません。