

# Web 画面

グリッドが 1 つの Web パネルの  
ベーステーブルとナビゲーション

*GeneXus*<sup>™</sup>

## グリッドが 1 つの Web パネル

グリッドが 1 つの Web パネルの場合を見ていきます。

## グリッドが 1 つの場合

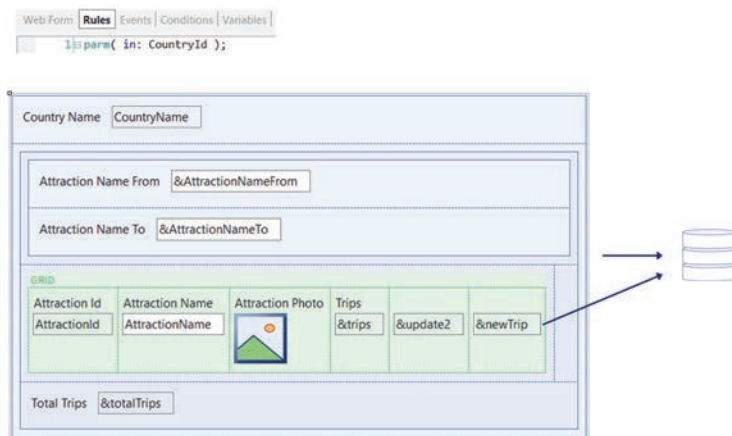
The screenshot displays the GeneXus IDE on the left and a web application on the right. The IDE shows a 'Rules' tab with a parameter `parm( in: CountryId );` and a form layout for 'Country Name' and 'Attraction Name'. A grid is defined with columns: 'Attraction Id', 'Attraction Name', 'Attraction Photo', 'Trips', and '&update'. The web application, titled 'Travel Agency', shows the result for 'France'. It includes input fields for 'Attraction Name From' and 'Attraction Name To', and a table of attractions.

Attraction...	Attraction...	Trips		
Eiffel Tower		5	UPDATE	NEW TRIP
Louvre Museum		1	UPDATE	NEW TRIP
Matisse Museum		2	UPDATE	NEW TRIP

Total Trips: 8

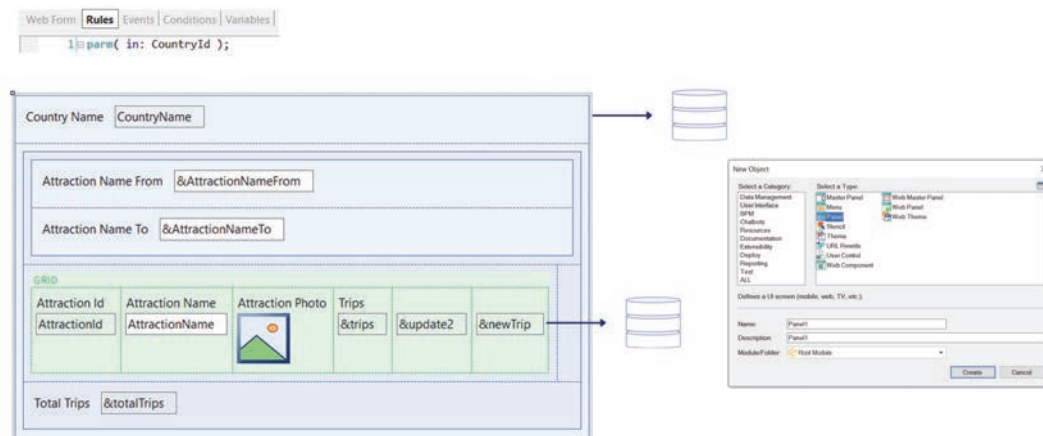
この例では、パラメーターとして国の識別子を受け取っています。また、観光名所はすべてグリッドに表示し、国の名前は固定部分に表示しています。

## グリッドが 1 つの場合



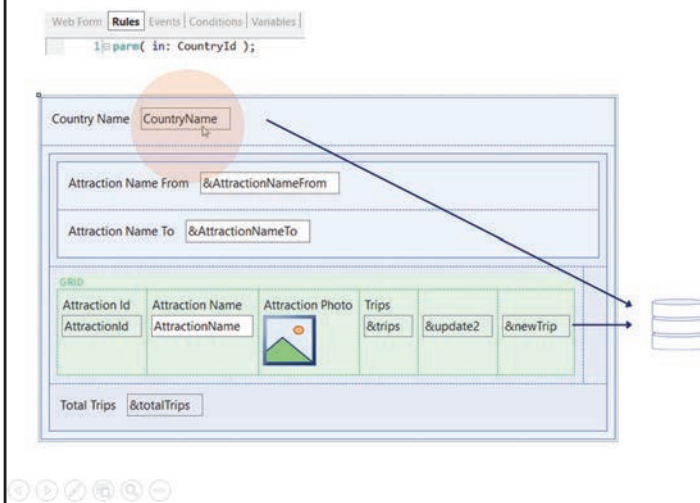
Web パネル内のグリッド数が最大で 1 つの場合、ベーステーブルが存在するのであれば、Web パネルのベーステーブルへの言及と、グリッドのベーステーブルへの言及に違いはないと言えます。なぜでしょうか。

## グリッドが 1 つの場合



それは、Angular や Android、Apple 向けに開発された普通のパネルのように、固定部分用に 1 つのナビゲーションが存在し、グリッド用に別のものが存在することは不可能だからです。

## グリッドが 1 つの場合



グリッドが 1 つの Web パネルでは、ベーステーブルがある場合、それは固有のものとなります。正しく実装された場合には、固定部分の項目属性は、グリッドにロードされるベーステーブルのすべてのレコードで同じ値になります。同じでない場合、その項目属性は固定部分ではなくグリッド内に置くべきです。

## グリッドが 1 つの場合

Web Form | **Rules** | Events | Conditions | Variables

1) perm( in: CountryId );

The screenshot shows the GeneXus Rules editor interface. On the left, a web form is displayed with a 'Country Name' field containing 'CountryName'. Below it, there are two input fields for 'Attraction Name From' and 'Attraction Name To', both containing '&AttractionNameFrom' and '&AttractionNameTo' respectively. A grid is shown below these fields, with columns for 'Attraction Id', 'Attraction Name', 'Attraction Photo', 'Trips', '&update2', and '&newTrip'. The 'Attraction Photo' column contains a small image icon. Below the grid, there is a 'Total Trips' field containing '&totalTrips'. On the right, a table for 'Grid: Grid1' is shown, detailing its properties and conditions.

Grid: Grid1	
Control Name	Grid1
Collection	
Base Trn	Attraction
Order	CountryId, AttractionName
Conditions	AttractionName >= &AttractionNameFrom...
Unique	
Save State	False
Data Selector	(none)

Grid1's Conditions

```
AttractionName >= &AttractionNameFrom
when not &AttractionNameFrom.IsEmpty();

AttractionName <= &AttractionNameTo
when not &AttractionNameTo.IsEmpty();
```

ここでは、条件のフィルタを満たし、パラメーターとして受け取る国にある観光名所がすべてグリッドにロードされます。CountryName は、すべての観光名所で同じになるため、固定部分にあります。

パラメーターとして受け取る CountryId 項目属性は、ベーステーブルの決定には関係ありませんが、前の例で見たように、GeneXus が決定するナビゲーションに対するフィルタとして機能します。

## グリッドが 1 つの場合


Web Form **Rules** Events Conditions Variables

1) perm( in: CountryId );

Country Name CountryName


Attraction Name From &AttractionNameFrom

Attraction Name To &AttractionNameTo

Attraction Id	Attraction Name	Attraction Photo	Trips	&update2	&newTrip
AttractionId	AttractionName		&trips	&update2	&newTrip

Total Trips &totalTrips

- フォーム内の項目属性 (表示または非表示)
- グリッドの [Base Trm] プロパティ
- グリッドの [Order] プロパティ
- グリッドの [Conditions] プロパティ
- グリッドの [Unique] プロパティ
- グリッドの [Data Selector] プロパティ
- イベント内の項目属性 (For each コマンド、Aggregate インライン式などのコンテキストがないもの)

Grid: Grid1		拡張 テーブル
Control Name	Grid1	
Collection		
Base Trm	Attraction	
Order	CountryId, AttractionName	
Conditions	AttractionName >= &AttractionNameFrom...	
Unique		
Save State	False	Grid1's Conditions
Data Selector	(none)	

AttractionName >= &AttractionNameFrom  
when not &AttractionNameFrom.IsEmpty();

AttractionName <= &AttractionNameTo  
when not &AttractionNameTo.IsEmpty();

ベーステーブルを決定する際には、Web パネルのどの部分が考慮されるのでしょうか。グリッドがある場合、前の例で見た部分に、グリッドに関連するものが加わります。

したがって、項目属性がグリッド内または [Order]、[Conditions]、[Unique]、[Data Selector] の各プロパティにあるか、ベーストランザクションがプロパティで指定されている場合、グリッド (そして Web パネル) には暗示的なベーステーブルが存在することになります。

ベーストランザクションがある場合、フォーム内の項目属性や、イベント内にある「コンテキストがない」項目属性が、その拡張テーブルに含まれている必要があります。含まれていなかった場合、「到達可能」ではないため値なしになる項目属性について、GeneXus から警告されます。

ベーストランザクションの指定がなかった場合、グリッドのない Web パネルの場合と同様に、GeneXus は、定義した場所にあるすべての項目属性が含まれる**最小拡張テーブル**を検索します。そのベーステーブルがベーステーブルとして選択されます。



## グリッドが 1 つの場合

Web Form | **Rules** | Events | Conditions | Variables


1) perm( in: CountryId );

Country Name: CountryName

Attraction Name From: &AttractionNameFrom

Attraction Name To: &AttractionNameTo

GRID

Attraction Id	Attraction Name	Attraction Photo	Trips	&update2	&newTrip
AttractionId	AttractionName		&trips	&update2	&newTrip

Total Trips: &totalTrips

Entity Diagram:

```

graph TD
    Attraction[Attraction] <--> Country[Country]
    Attraction <--> CountryCity[CountryCity]
    Country <--> CountryCity
    Category[Category] <--> Attraction
  
```

Grid: Grid1

Control Name	Grid1
Collection	
Base Trm	Attraction
Order	CountryId, AttractionName
Conditions	AttractionName >= &AttractionNameFrom...
Unique	Grid1's Conditions
Save State	False
Data Selector	(none)

Grid1's Conditions

```

AttractionName >= &AttractionNameFrom
when not &AttractionNameFrom.IsEmpty();

AttractionName <= &AttractionNameTo
when not &AttractionNameTo.IsEmpty();
  
```

イベントを確認する前に、ベーステーブルは Attraction となることがわかります。なぜなら、ベーストランザクションがあるからです。そして、ほかの項目属性がすべてその拡張テーブルに含まれることを確認できます。このため、イベントが 1 つもプログラミングされていなくても、Attraction テーブルが参照されることがわかります。順序は CountryId および AttractionName を基準とします。観光名所のフィルタとしては、AttractionName が 2 つの条件を満たし、パラメーターとして受け取る国と一致している必要があります。

繰り返しになりますが、Web パネルに**グリッドが 1 つある場合**、固定部分の項目属性は、そのグリッドのナビゲーションに対応します。グリッドが複数ある場合は、そのうち 1 つのナビゲーションに対応します。

CountryId 項目属性でデータを受け取らないとしたら、グリッドにはさまざまな国の観光名所がロードされます。画面上で CountryName 項目属性の値はどうなるでしょうか。グリッドにロードされる最初か最後のレコードになると容易に推測できます。

試してみましょう。

## グリッドが 1 つの場合

**Web Panel ViewCountryInfoSimple Navigation Report**

Name: ViewCountryInfoSimple  
 Description: View Country Info Simple  
 Environment: Default (C#)  
 Spec. Version: 17\_0\_0-144971  
 Form Class: HTML  
 Program Name: ViewCountryInfoSimple  
 Parameters: in: CountryId

Warnings:

SPS0038 There is no index for order CountryId, AttractionName; poor performance may be noticed in grid 'Grid1'.

Event Grid1 Load

```

Order: CountryId, AttractionName
      No index!
Navigation filters: Start from: CountryId = @CountryId
                  Loop while: CountryId = @CountryId
Constraints: AttractionName = &AttractionNameFrom WHEN not &AttractionNameFrom, isempty()
            AttractionName = &AttractionNameTo WHEN not &AttractionNameTo, isempty()
Join location: Server
            Attraction (AttractionId)
            Country (CountryId)
            count (TripDate) navigation
            TripAttraction (AttractionId)
            Trip (TripId)
          
```

The screenshot shows a web application titled "Travel Agency" with a red header. Below the header, there's a navigation bar with links: "Recent", "WebAttractions From...", "View Attraction From...", "View Country Info...". The main content area shows a form for "COUNTRY NAME" with "France" selected. Below the form, there's a table with columns "Attraction Name", "Attraction Photo", and "Trips". The table lists three attractions: "Eiffel Tower", "Louvre Museum", and "Museum of Modern Art". Each row has "UPDATE" and "NEW TRIP" buttons. At the bottom, there's a "Total Trips" row showing a value of 8.

正しくプログラミングされた Web パネルを示します。最初にナビゲーション表示を確認します。

ベーステーブルである Attraction があります。グリッドの Load イベントで明確に特定されていることがわかります。

パラメーターを変更する前に実行してみます。この Web パネルを基に、今回のパネルを作成します。

## グリッドが 1 つの場合

**Web Panel ViewCountryInfoSimple Navigation Report**

Name: ViewCountryInfoSimple  
Description: View Country Info Simple

Environment: Default (C#)  
Spec. Version: 17\_0\_0-144371  
Form Class: HTML  
Program Name: ViewCountryInfoSimple  
Parameters: in: &CountryId

Warnings

▲ spc0938 There is no index for order CountryId, AttractionName: poor performance may be noticed in grid 'Grid1'.

Event Grid1 Load

```

Order:
  CountryId AttractionName
  No index!
Navigation filters:
  Start from: FirstRecord
  Loop while: NotEndOfTable
Constraints:
  AttractionName >= &AttractionNameFrom WHEN not &AttractionNameFrom, isempty()
  AttractionName <= &AttractionNameTo WHEN not &AttractionNameTo, isempty()
Join location:
  Server
  Attraction (AttractionId)
    Country (CountryId)
      country.TripDate Navigation
        TripAttraction (AttractionId)
          Trip (TripId)

```

Travel Agency

Country Name: Brazil

Attraction Name From:

Attraction Name To:

Attraction Name	Attraction Photo	Trips	UPDATE	NEW TRIP
Christ the Redeemer		1	UPDATE	NEW TRIP
Eifel Tower		5	UPDATE	NEW TRIP
Louvre Museum		1	UPDATE	NEW TRIP
Musee Museum		2	UPDATE	NEW TRIP
Forbidden city		1	UPDATE	NEW TRIP

パネルに変更を加え、データを CountryId 項目属性ではなく、&CountryId 変数で受け取るようにします。これで自動フィルタはなくなります。

ナビゲーション表示を見てみましょう。フィルタがないことがわかります。

実行環境で確認してみましょう。最後のレコードではなく、最初のレコードの国の名前がロードされています。

## グリッドが 1 つの場合


Web Form **Rules** Events Conditions Variables

1) perm( in: CountryId );

Country Name

Attraction Name From

Attraction Name To

Attraction Id	Attraction Name	Attraction Photo	Trips	&update2	&newTrip
					

Total Trips

```

Event Grid1.Load
    &trips = Count(TripDate,
    &totalTrips = &totalTrips + &trips
Endevent

Event Grid1.Refresh
    &totalTrips = 0
Endevent

Event Start
    &newTrip = "New trip"
    &update2 = "UPDATE"
    CountryName.ForeColor = RGB(147,4,55) //DarkBase
    CountryName.FontBold = True
Endevent

Event &newTrip.Click
    &trips = NewTrip(AttractionId)
    Refresh
endevent

Event AttractionName.Click
    ViewAttractionFromScratch(AttractionId)
Endevent

```

パラメーターを元に戻します。

イベントを確認すると、Load イベントは、グリッドにベーステーブルがあることを前提にプログラミングされています。GeneXus 上で、ベーステーブルの決定に関与する項目属性を確認すると、すべてが「コンテキストがない」ことがわかります。

TripDate が関与しないのは明らかなです。TripDate は、Count 式がナビゲートするテーブルの決定にのみ使われます。Web パネル全般に言えることですが、グリッドにベーステーブルがある場合、この Count 式に暗示的な条件が適用される可能性があります。この例では、これからロードされる観光名所のツアー数が数えられます。

## グリッドが 1 つの場合


Web Form **Rules** Events Conditions Variables

1 parm( in: CountryId );

Country Name &CountryName

Attraction Name From &AttractionNameFrom

Attraction Name To &AttractionNameTo

Attraction Id	Attraction Name	Attraction Photo	Trips	&update2	&newTrip
&AttractionId	&AttractionName		&trips	&update2	&newTrip

Total Trips &totalTrips

```
Event Grid1.Load
  For each
    &CountryName = CountryName
    &AttractionId = AttractionId
    &AttractionName = AttractionName
    &AttractionPhoto = AttractionPhoto
    &trips = Count(TripDate)
    Load
    &totalTrips = &totalTrips + &trips
  endfor
Endevent

Event Grid1.Refresh
  &totalTrips = 0
Endevent

Event Start
  &newTrip = "New trip"
  &update2 = "UPDATE"
  &CountryName.ForeColor = RGB(147,4,55) //DarkBase
  &CountryName.FontBold = True
Endevent

Event &newTrip.Click
  &trips = NewTrip(&AttractionId)
  Refresh
Endevent

Event &AttractionName.Click
  ViewAttractionFromScratch(&AttractionId)
Endevent
```

前に説明したように、表示するすべての項目属性を変数に置き換える際には、1 か所でも忘れていたところがあると不十分になります。たとえば、ここでは、前のパネルをベーステーブルなしで実装するために別名で保存しました。

フォーム内の項目属性をすべて変数に置き換え、またイベント内でも同じ項目属性を置き換えました。

グリッドの Load イベントは、ベーステーブルのない Web パネルのプログラミングを想定し、For each コマンドで Attraction テーブルを参照し、グリッド内のすべての変数 (および固定部分の &CountryName) をロードするようプログラミングしました。

For each にベーストランザクションを指定していませんが、指定することも可能でした。CountryId をパラメーターとして受け取っているため、CountryId でフィルタリングされることはわかっています。また、parm ルールのこの項目属性がベーステーブルの決定に関与しないこともわかっています。

それでは、観光名所名によるフィルタはどこで指定しているのでしょうか。そのことは忘れていました。グリッドのプロパティを変更することも忘れていました。そして、グリッドのプロパティから**ベーステーブルが決定**されます。

## グリッドが 1 つの場合


Web Form **Rules** Events Conditions Variables

1 param( in: CountryId );

Country Name &CountryName

Attraction Name From &AttractionNameFrom

Attraction Name To &AttractionNameTo

Attraction Id	Attraction Name	Attraction Photo	Trips	&update2	&newTrip
&AttractionId	&AttractionName		&trips	&update2	&newTrip

Total Trips &totalTrips

```
Event Grid1.Load
For each
    &CountryName = CountryName
    &AttractionId = AttractionId
    &AttractionName = AttractionName
    &AttractionPhoto = AttractionPhoto
    &trips = Count(TripDate)
    Load
    &totalTrips = &totalTrips + &trips
endfor
Endevent
```

Event Grid1.Refresh

Grid: Grid1	
Control Name	Grid1
Collection	
Base Trn	Attraction
Order	CountryId, AttractionName
Conditions	AttractionName >= &AttractionNameFrom...
Unique	Grid1's Conditions
Save State	False
Data Selector	(none)

AttractionName >= &AttractionNameFrom

when not &AttractionNameFrom.IsEmpty();

AttractionName <= &AttractionNameTo

when not &AttractionNameTo.IsEmpty();

endevent

```
Event &Attraction
ViewAttractionFromScratch(&AttractionId)
Endevent
```

忘れていたこの 3 つのプロパティがベーステーブルを確立しています。

## グリッドが 1 つの場合



```

Event Grid1.Load
  For each
    &CountryName = CountryName
    &AttractionId = AttractionId
    &AttractionName = AttractionName
    &AttractionPhoto = AttractionPhoto
    &trips = Count(TripDate)
    Load
    &totalTrips = &totalTrips + &trips
  endfor
Endevent

Event Grid1.Refresh
  &totalTrips = 0
Endevent

Event Start
  &newTrip = "New trip"
  &update2 = "UPDATE"
  &CountryName.ForeColor = RGB(147,4,55) //DarkBase
  &CountryName.FontBold = True
Endevent

Event &newTrip.Click
  &trips = NewTrip(&AttractionId)
  Refresh
endevent

Event &AttractionName.Click
  ViewAttractionFromScratch(&AttractionId)
Endevent

```

ここで、ナビゲーション表示を見てみましょう。Attraction ベーステーブルが見つかったので、Load が N 回実行され、パラメーターとして受け取る国でフィルタリングされ、またグリッドで指定した条件に従い、変数でフィルタリングされます。

そしてレコードごとに Load の For each が実行され、ベーステーブルが同じ Attraction であるため、コントロールブレイク処理が行われます。

グリッドのプロパティのチェックを忘れたため、このようになります。

## グリッドが 1 つの場合

The screenshot displays the GeneXus IDE interface for configuring a single grid. The top panel shows the 'Event Grid1.Load' code with a 'For each Attraction' loop. The middle panel shows the 'Properties' window for the 'Grid1' control. The bottom panel shows the 'Event Grid1.Load' configuration window with a 'For Each Attraction' loop and various filters and constraints.

```

Event Grid1.Load
  For each Attraction
    order CountryId, AttractionName
    where AttractionName >= &AttractionNameFrom when not &AttractionNameFrom.IsEmpty()
    where AttractionName <= &AttractionNameTo when not &AttractionNameTo.IsEmpty()
    &CountryName = CountryName
    &AttractionId = AttractionId
    &AttractionName = AttractionName
    &AttractionPhoto = AttractionPhoto
    &trips = Count(TripDate)
    Load
    &totalTrips = &totalTrips + &trips
  endfor
Endevent
  
```

Environment: Default (C#)  
 Spec. Version: 17\_0\_0-144971  
 Form Class: HTML  
 Program Name: ViewCountryInfoSimple2  
 Parameters: in: CountryId

Properties window for Grid1:

Control Name	Grid1
Collection	
Base Tm	
Order	
Conditions	
Unique	
Save State	False
Data Selector	(none)
Appearance	

Event Grid1.Load configuration:

```

For Each Attraction (Line: 2)
  Order: CountryId, AttractionName
  No index
  Navigation filters: Start from: CountryId = @CountryId
  Loop while: CountryId = @CountryId
  Constraints: AttractionName >= &AttractionNameFrom WHEN not &AttractionNameFrom.IsEmpty()
  AttractionName <= &AttractionNameTo WHEN not &AttractionNameTo.IsEmpty()
  Join location: Server
  -Attraction (AttractionId)
  -Country (CountryId)
  -Count(TripDate) navigation
  -TripAttraction (AttractionId)
  -Trip (TripId)
  
```

そこで、これらのプロパティを For each コマンドに入れます。最初にベーストランザクションです。  
 次に順序です。  
 最後に条件として Where 節を加えます。  
 これを保存します。For each の構文でセミコロンは使用できないため、エラーが発生しました。もう一度、保存します。これで完了です。  
 ナビゲーション表示を確認します。  
 今度は、確実に Web パネルにベーステーブルがないようにすることができました。

グリッドのないパネルの例と同様に、Where 節をグリッドではなく Web パネルの全般的な条件として置いた場合は、条件の項目属性がベーステーブルの決定に関与しません。ここで、それがはっきりと見てとれます。ナビゲーション表示は前のものと変わりません。

全般的な条件は、その項目属性に関わる Web パネル全体のナビゲーションに対してフィルタとして適用されますが、問題はこのような処理を求めている場合もあることです。たとえば、グリッドが複数ある場合を考えてみてください。



## グリッドが 1 つの場合

```

Event Grid1.Load
  For each Attraction
    order CountryId, AttractionName
    &CountryName = CountryName
    &AttractionId = AttractionId
    &AttractionName = AttractionName
    &AttractionPhoto = AttractionPhoto
    &trips = Count(TripDate)
    Load
    &totalTrips = &totalTrips + &trips
  endfor
Endevent

```

Country Name: &CountryName

Attraction Name From: &AttractionNameFrom

Attraction Name To: &AttractionNameTo

Attraction Id	Attraction Name	Attraction Photo	Trips	&update2	&newTrip
&AttractionId	&AttractionName		&trips	&update2	&newTrip

Total Trips: &totalTrips

```

Event Grid1.Load
  &trips = Count(TripDate)
  &totalTrips = &totalTrips + &trips
Endevent

```

Country Name: CountryName

Attraction Name From: &AttractionNameFrom

Attraction Name To: &AttractionNameTo

Attraction Id	Attraction Name	Attraction Photo	Trips	&update2	&newTrip
AttractionId	AttractionName		&trips	&update2	&newTrip

Total Trips: &totalTrips

&trips > 1  
の場合にのみ  
観光名所を  
ロード?

一般に、Web パネルはベーステーブルを使って実装することをお勧めします。そのほうが作業が少なくなるからです。この例では、For each を定義し、項目属性の値を変数に割り当て、各行をロードする Load コマンドを指定する必要がありました。これに対して、ベーステーブルがあった場合は、すべてが大幅にシンプルです。

それでは、複数のツアーに含まれる観光名所のみをロードするには、どうすればいいでしょうか。

## グリッドが 1 つの場合

```
Event Grid1.Load
  For each Attraction
    order CountryId, AttractionName
    &CountryName = CountryName
    &AttractionId = AttractionId
    &AttractionName = AttractionName
    &AttractionPhoto = AttractionPhoto
    &trips = Count(TripDate)
    if &trips > 1
      Load
    endif
    &totalTrips = &totalTrips + &trips
  endfor
Endevent
```

&trips > 1  
の場合にのみ  
観光名所を  
ロード?

```
Event Grid1.Load
  &trips = Count(TripDate)
  If &trips > 1
    Load
  endif
  &totalTrips = &totalTrips + &trips
Endevent
```

Country Name: &CountryName

Attraction Name From: &AttractionNameFrom

Attraction Name To: &AttractionNameTo

Attraction Id	Attraction Name	Attraction Photo	Trips	&update2	&newTrip
&AttractionId	&AttractionName		&trips	&update2	&newTrip

Total Trips: &totalTrips

Country Name: CountryName

Attraction Name From: &AttractionNameFrom

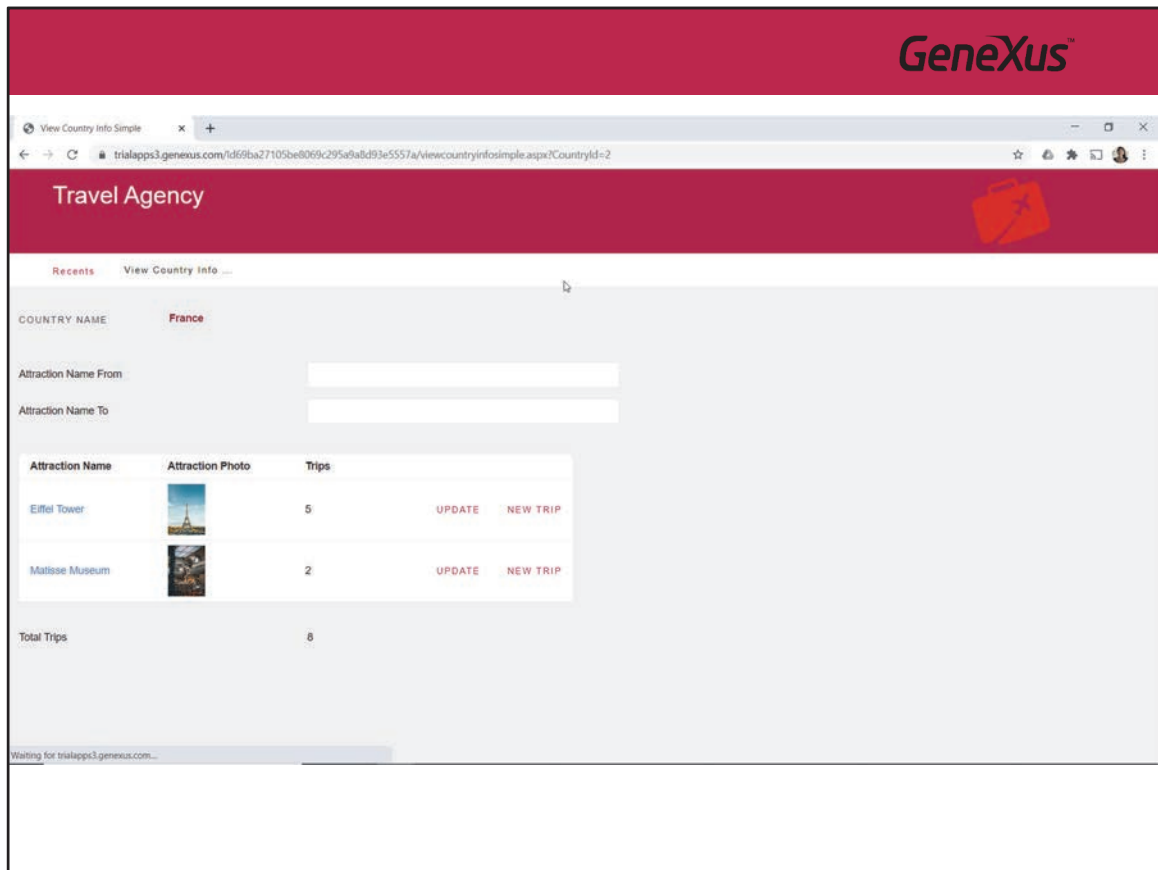
Attraction Name To: &AttractionNameTo

Attraction Id	Attraction Name	Attraction Photo	Trips	&update2	&newTrip
AttractionId	AttractionName		&trips	&update2	&newTrip

Total Trips: &totalTrips

ベーステーブルがない Web パネルでは非常に簡単です。&trips 変数の値を Load コマンドの条件にします。&trips 変数が 1 よりも大きい場合にのみグリッドに行をロードします。条件を満たさない場合はロードしません。

ベーステーブルがある Web パネルのほうはどうでしょうか。この場合、グリッド行のロード直前に Load が実行されるため、同じ方法は不可能に思われます。また、ベーステーブルがない Web パネルでは明示的だった Load コマンドが、ここでは暗示的であるため、ロードを止める方法がないようです。ロードは、開発者ではなく GeneXus によってソースコードに入れられています。しかし、不可能ではありません。Load を明示的にプログラミングすれば、GeneXus が Load を置くべきではないと理解します。この方法で、開発者側で処理ができるようになります。



試してみましょう。