

# Web 画面

グリッドのない Web パネルの  
ベーステーブルとナビゲーション

*GeneXus*<sup>™</sup>

## グリッドのない Web パネル

グリッドのない Web パネルから始めます。

Web Form \* **Rules \*** Events Conditions Variables

```
1 parm(in:&AttractionId);
```

Web Form \* Rules \* Events **Conditions** Variables

```
1 AttractionId = &AttractionId;
```

- フォーム内の項目属性 (表示または非表示)
- 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)

```
Event Load
    &trips = Count(TripDate)
Endevent
```

```
Event CountryName.Click
    ViewCountryInfoSimple(CountryId)
Endevent
```

```
Event Start
    CountryId.Visible = False
    AttractionName.ForeColor = RGB(147,4,55) //DarkBase
    AttractionName.FontBold = True
Endevent
```

フォーム内に項目属性がある場合、この事実だけでその値をデータベース内で検索する必要があることが GeneXus にわかります。これは、グリッドがある Web パネルおよびグリッドがない Web パネルの両方に当てはまります。

それでは、フォーム内に項目属性がない場合はどうでしょうか。それだけでベーステーブルを確立しない理由として十分でしょうか。そのようなことはありません。

GeneXus は項目属性をフォーム内 (表示および非表示) およびすべてのイベント内で探します。ただし、イベント内では「コンテキストがない」項目属性のみが考慮されます。つまり、それを表す具体的なナビゲーションコンテキストがない項目属性です。たとえば For Each コマンドや Aggregate インライン式の中の項目属性の場合、ナビゲーションは、それぞれ For Each コマンドまたは式によって決まります。

ここに示す例では、3 つのイベントがプログラミングされています。最初は Load イベントです。これは、GeneXus が Web パネルのベーステーブルを見つけるかどうかに応じて、1 回のみ、または n 回実行されます。プログラミングを見ると、TripDate という項目属性がありますが、これは Aggregate インライン式のコンテキストに含まれるため、パネルのベーステーブルの決定には使われず、式で参照するテーブルの決定にのみ使われます。パネルにベーステーブルがなかった場合、Load イベントは 1 回のみ実行され、その際に式で Trips のすべてのレコードが数えられます (TripDate は Trip トランザクションの従属項目属性)。

しかし、残り 2 つのイベントは状況異なります。コンテキストなしで記述されている項目属性があり、コンテキストとなる**ベーステーブルが存在**すると想定されています。これらの項目属性は、インライン式や For Each コマンドの中、コンテキストを確立するその他の構造の中に入っていません。これらは、Web パネルのベーステーブルの拡張テーブルの項目属性になります。

このため、フォームで項目属性を指定していなくても、これらの項目属性がここに存在するだけで、GeneXus ではベーステーブル (および拡張テーブル) が推論されます。

先に進む前に、parm ルール内の項目属性や Web パネルの条件で指定する項目属性は関与しないことを確認しておきます。条件に AttractionId があることでパネルにベーステーブルがあると決まるわけではありません。この点については後で詳しく説明します。

Web Form \* Rules \* Events Conditions Variables

1 parm(in:&amp;AttractionId);

Web Form \* Rules \* Events Conditions Variables

1 AttractionId = &amp;AttractionId;

- フォーム内の項目属性 (表示または非表示)
- 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)

Event Load

For each

```

    &AttractionName = AttractionName
    &CountryId = CountryId
    &CountryName = CountryName
    &CityName = CityName
    &CategoryName = CategoryName
    &AttractionPhoto = AttractionPhoto
    &trips = Count(TripDate)

```

endfor

Endevent

Event &amp;CountryName.Click

ViewCountryInfoSimple(CountryId)

Endevent

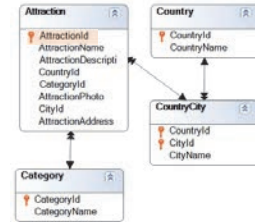
Event Start

```

    &CountryId.Visible = False
    &AttractionName.ForeColor = RGB(147,4,55) //DarkBase
    &AttractionName.FontBold = True

```

Endevent



前の例のパネルをベーステーブルのないものに変換するとします。そのためには最初にパネルを別名で保存します。その後、フォーム内のすべての項目属性を変数に変更します。Start イベントで非表示、色、太字の設定を行うのは、これらの変数になります。今回はベーステーブルなしにしたいので、Load イベント内に For Each コマンドを記述します。すべてが想定どおりに動作すれば、実行は 1 回のみで、そのときに変数の値をロードする必要があるからです。この例では、グリッドに行を追加しないため、Load コマンドを記述する必要はありません。変数は、ロードしたときの値で画面上にロードされます。

For Each コマンドにベーストランザクションを指定していないため、GeneXus ではすべての項目属性が抽出され、**そのすべてが含まれる最小拡張テーブル**として Attraction テーブルが特定されます。このテーブル内の項目属性を含む条件があるため、それが適用されます。すべてが正常に機能します。

Load イベントでは、For Each コマンドにより、AttractionId に該当する観光名所のレコードが 1 つ取得されます。

Web Form \* **Rules \*** Events Conditions Variables

1 parm(in:&AttractionId);

Web Form \* Rules \* Events **Conditions** Variables

1 AttractionId = &AttractionId;

Name &AttractionName

Country Id &CountryId

Country Name &CountryName

City Name &CityName

Category Name &CategoryName

Trips &trips

**Web Panel ViewAttractionFromScratch2 Navigation Report**
**Name:** ViewAttractionFromScratch2  
**Description:** View Attraction From Scratch2

**Environment:** Default (C#)  
**Spec. Version:** 17\_0\_0-144971  
**Form Class:** HTML  
**Program Name:** ViewAttractionFromScratch2  
**Parameters:** in: &AttractionId

**Event Load**

For First Attraction (Line: 2)

Order: AttractionId  
Index: IATTRACTION  
Navigation Start from: AttractionId = &AttractionId  
filters: Loop while: AttractionId = &AttractionId  
Join location: Server

Attraction (AttractionId)
Country (CountryId)
CountryCity (CountryId, CityId)
Category (CategoryId)
count(TripDate) navigation
TripAttraction (AttractionId)
Trip (TripId)

ベーステーブルのない Web パネルに想定されるナビゲーション表示はこのようになります。Load イベント内で、Attraction テーブルに対して For Each コマンドが指定されています (また、パラメーターとして受け取る変数の値と一致する AttractionId で適切にフィルタリングされていることもわかります)。

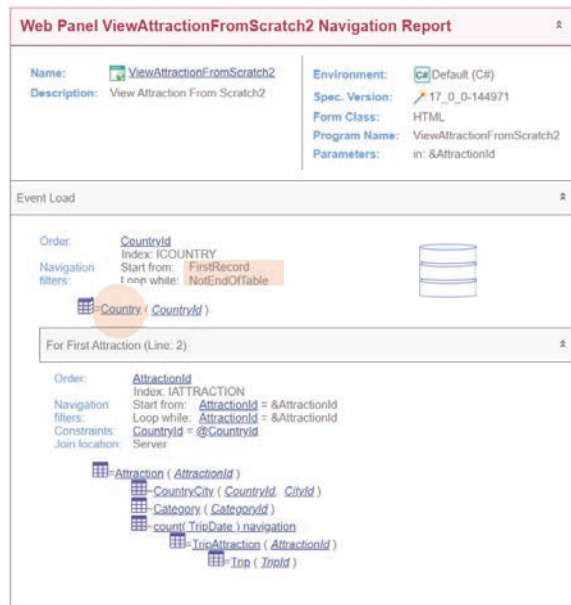
これがテーブルの主キーであるため、For Each コマンドでは 1 つのレコードが取得されます。順序はこの項目属性を基準とするため、これが最初 (かつ唯一) のレコードとなります。この理由から、ここでは **For Each** ではなく、**For First** と表記されています。

Web Form \* **Rules \*** Events Conditions Variables

1 parm(in:&AttractionId);

Web Form \* Rules \* Events **Conditions** Variables

1 AttractionId = &AttractionId;



ところが、GeneXus で実際のナビゲーションを確認すると、レポートはこうになっており、明らかにベーステーブルが見つかっています。Load イベントを見るとわかります。Country テーブルです。

このテーブル全体が参照され、国ごとに Load イベントが 1 回ずつ実行されます。そして、その都度、For Each コマンドが実行されます。

画面表示はどうなるでしょうか。実行時に問題に気づくことはありません。画面変数への割り当ては、**観光名所の国**の Load が実行される時にのみ実行されるからです。これが、For Each の本体が実行される唯一の場合です。

運がよかったようです。しかし、Web パネルのプログラミングは間違っていました。なぜベーステーブルが見つかったのでしょうか。

Web Form \* **Rules \*** Events Conditions Variables

```
1 parm(in:&AttractionId);
```

Web Form \* Rules \* Events **Conditions** Variables

```
1 AttractionId = &AttractionId;
```

- フォーム内の項目属性 (表示または非表示)
- 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)

```
Event Load
  For each
    &AttractionName = AttractionName
    &CountryId = CountryId
    &CountryName = CountryName
    &CityName = CityName
    &CategoryName = CategoryName
    &AttractionPhoto = AttractionPhoto
    &trips = Count(TripDate)
  endfor
Endevent

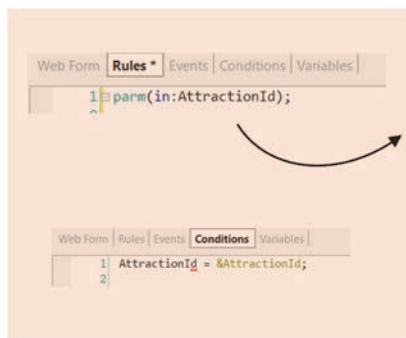
Event &CountryName.Click
  ViewCountryInfoSimple(CountryId)
Endevent

Event Start
  &CountryId.Visible = False
  &AttractionName.ForeColor = RGB(147,4,55) //DarkBase
  &AttractionName.FontBold = True
Endevent
```

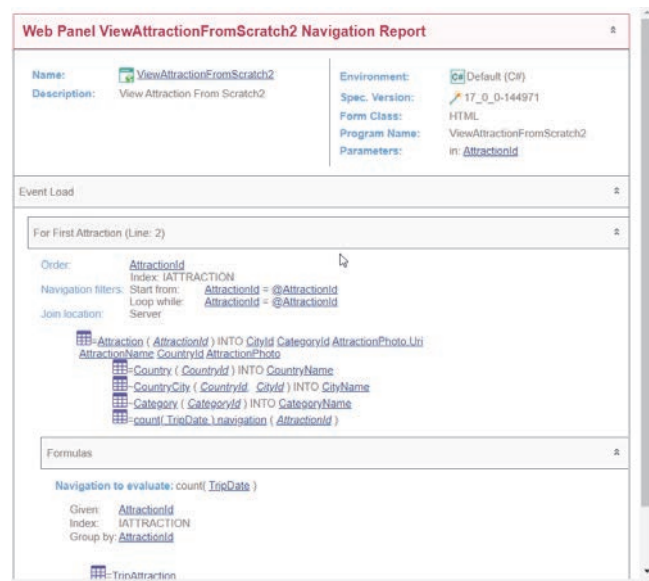
それは、この項目属性を見落としていたからです。ここには &CountryId 変数を入れる必要があります。

具体的には、GeneXus で Web パネルのナビゲーションが決定されるとき、イベント内でこの項目属性がコンテキストなしで見つかり、Web パネルのベーステーブルというコンテキストが暗示的に伝えられていると理解されます。そしてこれが唯一の項目属性であるため、ベーステーブルを決定する際にこの項目属性のみが考慮されることになります。この理由から、Country テーブルが選択されます。





ベーステーブルの決定後



前述のプログラミングに従ったナビゲーション表示を示します。これから問題を修正します。項目属性を変数 &CountryId に置き換えます。

その後、保存します。ここで新しいナビゲーション表示を見ると、想定どおりになっています。つまり、Load イベントにベーステーブルがなく、その中で For Each コマンドが Attraction テーブル全体に対して実行されています。

これで、ベーステーブルのない Web パネルが完成しました。条件内の項目属性が Web パネルのベーステーブルの決定に関与しないことも確認できました。そうでなければ、この項目属性からベーステーブルが決定されたはずですが、そうはならなかったからです。

同様に、変数ではなく、項目属性で受け取った場合、ナビゲーションはどうなるかを確認します。この場合も、ベーステーブルがないパネルのナビゲーションになります。

条件は、parm ルールで項目属性として暗示的に受け取る場合も、パネルの条件で明示的に指定している場合も、ナビゲーション対象のテーブルと、アクセスする拡張テーブルが決定された後、データのフィルタリング時に考慮されます。フィルタは、必要に応じてこのとき初めて適用されます。