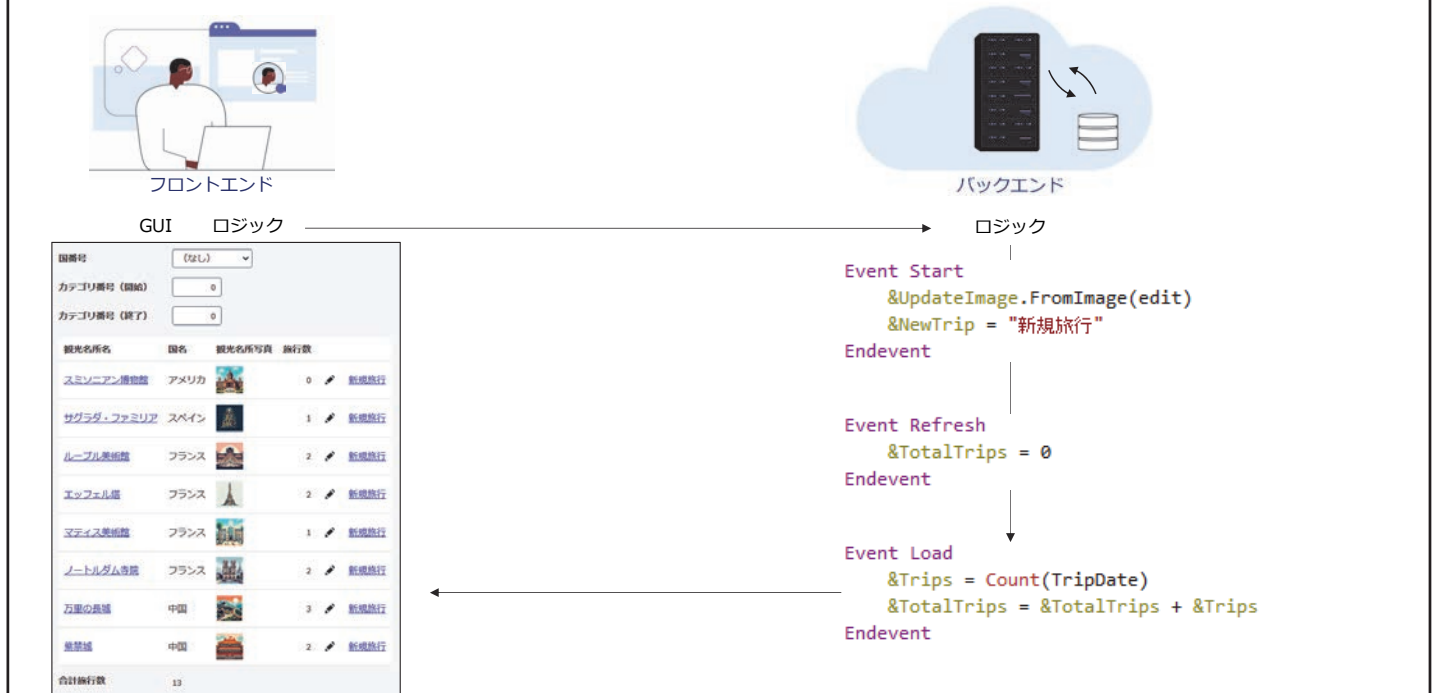


Web Panel オブジェクトのイベント実行スキーム

GeneXus™

Web パネルの初回表示時のロジック（ベーステーブルありのグリッド）



このコースで、ここまでに説明してきた相互作用するフロントエンド、バックエンドのロジックについて改めて確認を行います。

初めて画面を開いた場合、フロントエンドのロジックレイヤーから、バックエンドのWeb パネルロジックとして、Start イベントが実行され、Refresh イベントが続けて実行されます。

その後、画面にグリッドコントロールが配置され、ベーステーブルがある場合、データの取得処理が行われます。

この処理は、For each コマンドが定義されている場合に似ています。

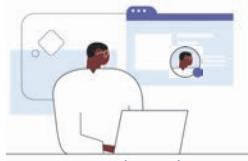
また、データの取得処理が行われる中で、Load イベントが定義されている場合、グリッドとして描画するデータに加え、実行され、結果がグリッドに表示される1行分のデータとなります。

この場合、対象となるデータ分、処理が繰り返し実行され、データの取得が行われます。ベーステーブルがあるグリッドコントロールとなるため、Load コマンドは不要です。

これらの内部の処理が完了すると、レスポンスがフロントエンドに送信され、画面の描画が行われます。

描画完了時点で、ユーザーは使い始めることができます。

Web パネルの初回表示時のロジック（ベーステーブルなしのグリッド）



フロントエンド



バックエンド

GUI ロジック

ロジック

国番号

(なし)

カテゴリ番号 (開始)

0

カテゴリ番号 (終了)

0

観光名所名	国名	観光名所写真	旅行数	
スミソニアン博物館	アメリカ		0	新規旅行
サグラダ・ファミリア	スペイン		1	新規旅行
ルーブル美術館	フランス		2	新規旅行
エッフェル塔	フランス		2	新規旅行
マティス美術館	フランス		1	新規旅行
ノートルダム寺院	フランス		2	新規旅行
万里の長城	中国		3	新規旅行
紫禁城	中国		2	新規旅行
合計旅行数	13			

```

Event Start
  &UpdateImage.FromImage(edit)
  &NewTrip = "新規旅行"
Endevent

Event Refresh
  &TotalTrips = 0
Endevent

Event Load
  For each Attraction
    Order CountryName
    Where CountryId = &CountryId when not &CountryId.IsEmpty()
    Where CategoryId >= &CategoryIdFrom when not &CategoryIdFrom.IsEmpty()
    Where CategoryId <= &CategoryIdTo when not &CategoryIdTo.IsEmpty()
    &AttractionId = AttractionId
    &AttractionName = AttractionName
    &CountryName = CountryName
    &AttractionPhoto = AttractionPhoto
    &Trips = Count(TripDate)
    Load
    &TotalTrips = &TotalTrips + &Trips
  Endfor
Endevent
  
```

もし、グリッドにベーステーブルがない実装だった場合、Load イベントの実装が異なります。

ベーステーブルがない場合、Load イベントは、1 回だけ実行されます。

そのため、グリッド上にデータを読み込むための処理を開発者がすべて実装する必要があります。

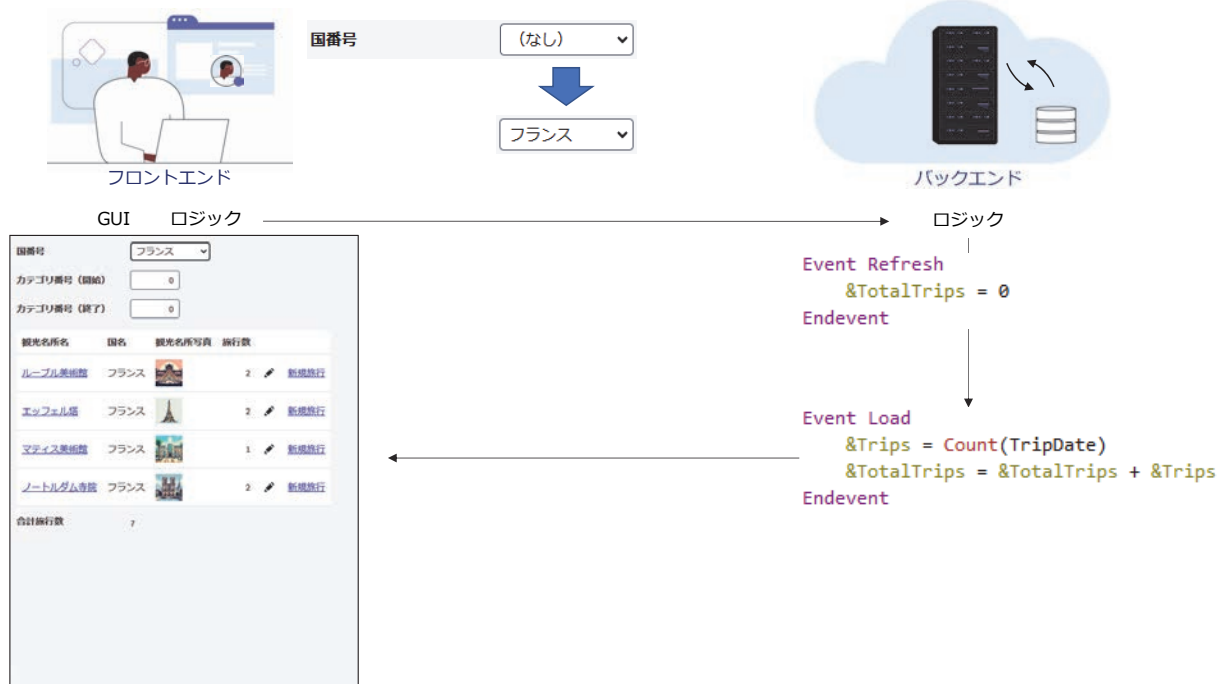
データベースにアクセスし、表示する場合には、For each コマンドを利用します。

また、グリッド上行として現在の情報を出力できるようにするためには、Load コマンドの実行が必要となります。

その後、ベーステーブルがある場合と同様に、レスポンスがフロントエンドに送信され、画面の描画が行われます。

描画完了時点で、ユーザーは使い始めることができます。

Web パネルの再表示時のロジック : 条件変更



では、画面上で操作が行われ、画面の再描画が必要となった場合についても記載します。

例えば、グリッドコントロールに表示されるデータの絞り込みが行われるように変数の変更が発生した場合があります。

この場合、データを絞り込み、対象のデータのみを表示するようにバックエンドで処理が必要となります。

変数を変更したタイミングで、フロントエンドからバックエンドのロジックが呼び出されます。

初回の処理でない場合、Start イベントは実行されません。

そのため、バックエンドでは、初めに Refresh イベントが実行されます。

その後、Load イベントが実行されます。

イベント内に記載が必要な内容に差異はありますが、ベーステーブルがあるグリッドを持つ Web パネルでも、持たない Web パネルでもこの挙動となります。

Web パネルの再表示時のロジック : クライアント側で完結するイベント



フロントエンド



バックエンド

GUI ロジック

ロジック

国番号	フランス		
カテゴリ番号 (開始)	0		
カテゴリ番号 (終了)	0		
観光名所名	国名	観光名所写真	旅行数
ルーブル美術館	フランス		2 新規旅行
エッフェル塔	フランス		2 新規旅行
マティス美術館	フランス		1 新規旅行
ノートルダム寺院	フランス		2 新規旅行
合計旅行数	7 フォントサイズ変更		

```
Event 'ChangeSize'
  &TotalTrips.FontSize = 18
Endevent
```

再表示が必要な場合でも、必ずバックエンドのロジックが実行されるものではありません。例えば、インターフェース上に配置された表示項目のフォントサイズや、太字への変更などの場合、バックエンドにアクセスする必要はなく、クライアント側で完結することができます。

行われた処理に基づき、対象となる変数などのコントロールの表示が更新されます。

Web パネルの再表示時のロジック : 画面再表示の必要性



フロントエンド



バックエンド

GUI ロジック

ロジック

国番号	フランス		
カテゴリ番号 (開始)	0		
カテゴリ番号 (終了)	0		
観光名所名	国名	観光名所写真	旅行数
ルーブル美術館	フランス		3 新規旅行
エッフェル塔	フランス		2 新規旅行
マティス美術館	フランス		1 新規旅行
ノートルダム寺院	フランス		2 新規旅行
合計旅行数	7		フォントサイズ変更

```

Event &NewTrip.Click
  &Trips = InsertTrip(AttractionId)
  Refresh
Endevent

Parm(IN:&AttractionId, OUT:&Trips);

New
  TripDate = &Today
  TripDescription = "観光名所一覧からの自動登録"
Endnew
&TripId = TripId

New
  TripId = &TripId
  AttractionId = &AttractionId
Endnew
&Trips = Count(TripDate, AttractionId = &AttractionId)
  
```

既に説明の通り、ユーザーイベントやコントロールイベントの場合、システムイベントは実行されません。

イベント内で変数への代入などが行われている場合、指定されたもののみ再表示が行えます。

もし、イベント内で変更した変数を別の変数で利用している場合、これだけでは不十分な実装となります。

では、別の変数の値を最新化するためのコードが、Refresh イベントや、Load イベントにある場合、開発者がシステムイベントの実行を指定することが可能です。これは、Refresh **コマンド**です。

Web パネルの再表示時のロジック : 任意の画面再表示 (Refresh コマンド)



フロントエンド

GUI ロジック

国番号	フランス		
カテゴリ番号 (開始)	0		
カテゴリ番号 (終了)	0		
観光名所名	国名	観光名所写真	旅行数
ルーブル美術館	フランス		3 新規旅行
エッフェル塔	フランス		2 新規旅行
マティス美術館	フランス		1 新規旅行
ノートルダム寺院	フランス		2 新規旅行
合計旅行数	8		フォントサイズ変更



バックエンド

ロジック

```

Event &NewTrip.Click
  &Trips = InsertTrip(AttractionId)
  Refresh
Endevent

Event Refresh
  &TotalTrips = 0
Endevent

Event Load
  &Trips = Count(TripDate)
  &TotalTrips = &TotalTrips + &Trips
Endevent
  
```

実行したイベントに Refresh コマンドが記述されていた場合、画面の再表示時に実行される場合と同様にシステムイベントが実行されます。

つまり、このイベントの処理を実行した後、Refresh イベントが実行され、Load イベントが実行されます。

そして、これらのイベントの実行が完了した後に、フロントエンドへレスポンスが送信され、結果が表示されます。

GeneXus[™]