

ネイティブ モバイル アプリケーション開発の概要 (スマートデバイス)

GeneXus™

これまで、旅行代理店が使用する Web アプリケーションについて、代理店の求める要件をすべて組み込んできました。

ここでは、代理店が、エンドユーザー向けの小規模なスマートデバイス対応アプリケーションの提供を考えていると想定します。

目的は、代理店が旅行を提供しているすべての都市、利用可能なツアー、主要な観光名所などについて、スマートデバイスから誰でも調べられるようにすることです。

トランザクション設計

Country トランザクション

名前	タイプ
Country	Country
CountryId	Id
CountryName	Name
CountryISO	VarChar(2)
City	City
CityId	Id
CityName	Name

Airport トランザクション

名前	タイプ
Airport	Airport
AirportId	Id
AirportName	Name
CountryId	Id
CountryName	Name
CityId	Id
CityName	Name

CountryId と
CityId は外部キー

Attraction トランザクション

名前	タイプ
Attraction	Attraction
AttractionId	Id
AttractionName	Name
AttractionFurigana	Name
CountryId	Id
CountryName	Name
CityId	Id
CityName	Name
CategoryId	Id
CategoryName	Name
AttractionPhoto	Image

Trip トランザクション

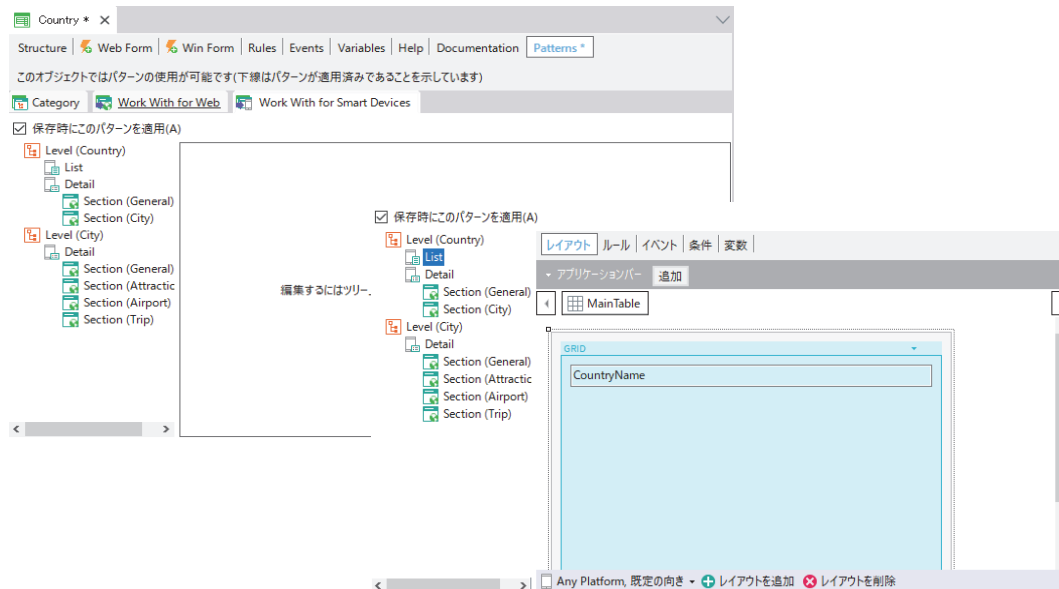
名前	タイプ
Trip	Trip
TripId	Id
TripDate	Date
TripDescription	VarChar(40)
CountryId	Id
CountryName	Name
CityId	Id
CityName	Name
Attraction	Attraction
AttractionId	Id
AttractionName	Name

今回の例題の前提となるトランザクション設計を確認してみましょう。

- 各国の都市を持つ Country トランザクションがあります。
- Attraction トランザクションは CountryId および CityId 項目属性を外部キーとして持ちます。
- Airport トランザクションも CountryId および CityId を外部キーとして持ちます。
- そして、Trip トランザクションも CountryId および CityId を外部キーとして持ちます。

Work With for Smart Devices

セクションの確認



これまでの章で「Web 用」の Work With パターンについて学習し、いくつかのトランザクションに適用したことを思い出してください。

ここでは、Country トランザクションに移動して [Patterns] エLEMENTを選択します。

Web 用の Work With パターンの隣に、スマートデバイス用の Work With パターンがあります。

[Work With for Smart Devices] を選択します。

左側にはツリーのノードが表示され、ノードの内容が右側に表示されます。

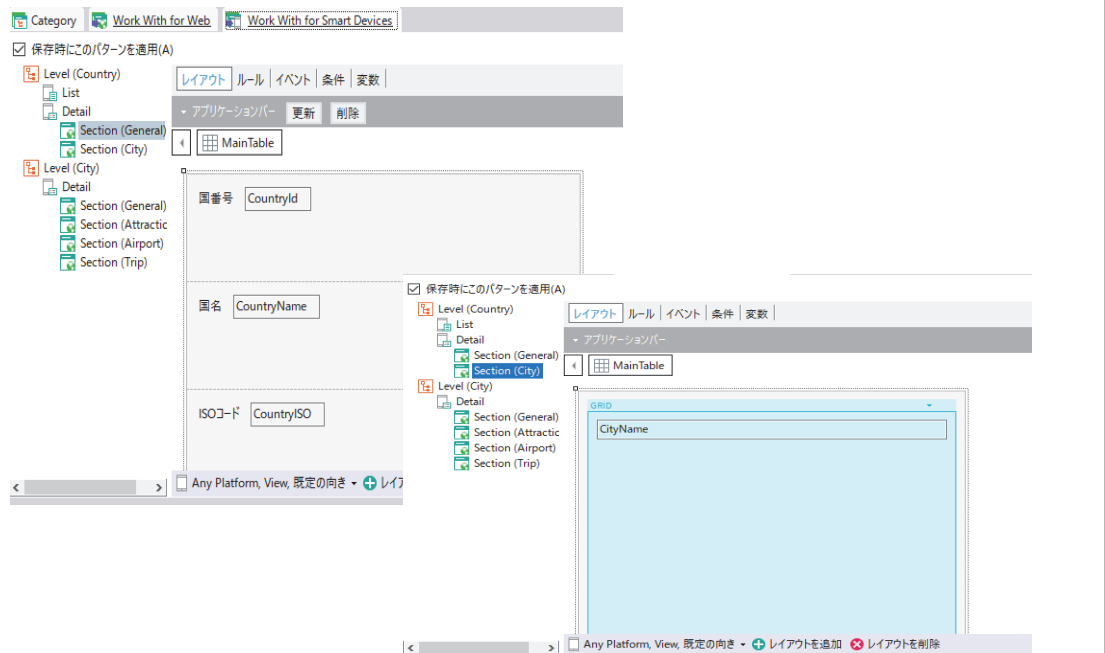
Country レベルに対応するノードの下に [List] ノードがあります。これをクリックすると、CountryName 項目属性が挿入されたグリッドに対応していることが分かります。

「List」ノードは、各レコードのデータ表示に対応したグリッドを持つパネルです。呼び出されるパネルのグリッドに、すべての国の名前がリストで表示されます。

このようにパターンでは [List] ノードの [Layout] は、既に項目属性をリストで表示するためにグリッドが含まれています。

Work With for Smart Devices

セクションの確認



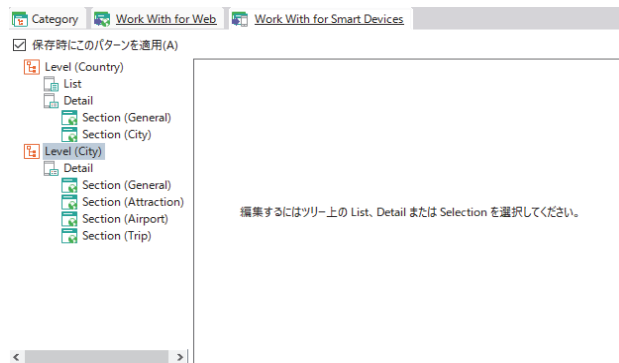
[Detail] ノードを確認します。「Detail」という用語は Work With for Web パターンの「View」と関連付けて考えることが可能です。これはリスト内の特定のレコードの詳細を表示することができるためです。

[Detail] ノードは 2 つのセクションで構成されています: General と City です。

Work With for Web パターンで実装された機能と同様に、[General] セクションには国に関連付けられたデータが表示され、[City] セクションにはすべての都市が表示されます。

Work With for Smart Devices

セクションの確認



パターンの適用



今度は次のノードを確認します: Level (City)

先ほどは Country トランザクションの 2 つのレベルのうち、1 つ目のレベルに対応するノードを確認しました。ここで確認するノードは 2 つ目のレベルに対応するものです。

Country の場合と同じように、このレベルの下にも General セクションがあります。そして、都市が持つ 1 対 N の関係ごとにセクションが 1 つ存在します。言い換えると、各都市には N 個の観光名所、N 個の空港、N 個のツアーがあります。

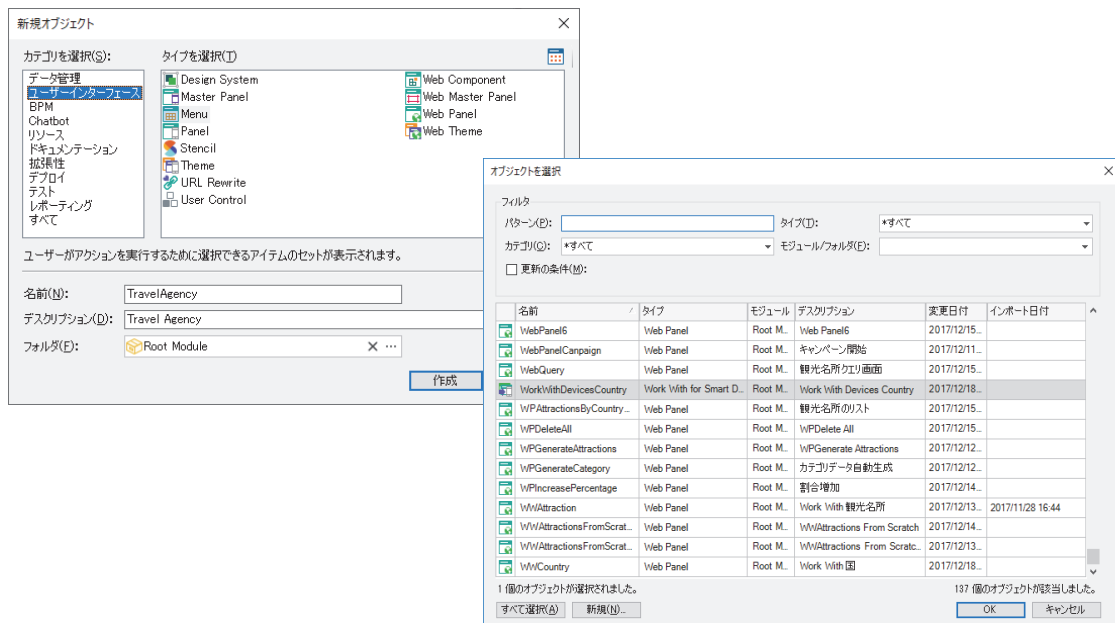
このパターンを適用して、何が生成され、提供されるかを確認します。[保存時にこのパターンを適用] ボックスをクリックします。

これで、WorkWithDevicesCountry オブジェクトが表示されるようになります。

Web 用のパターンに適用するときとは対照的に、生成されたオブジェクトのセットは表示されません。WorkWithDevicesCountry オブジェクトのみが表示されます。

Menu

アクセスメニュー



次に、パターンによって生成された機能呼び出すための、デバイスのアクセスメニューを作成します。これを行うために、Menu オブジェクトを作成します。

Menu オブジェクトでは、スマートデバイス用のアプリケーションに対応したアクセスメニューを作成できます。

ここで、メニューに含まれるオプションが追加されます。

[Items] ノードを右クリックし、[追加] → [Action] を選択します。ボックス内で WorkWithDevicesCountry オブジェクトを選択します。

Menu オブジェクトの [Items] ノードに追加されたアクションの [Description] プロパティに、次のような説明を入力します: TravelAgency

アイコンとして使う画像の選択も可能です。用途に合う画像がナレッジベースの既存のものに含まれない場合、ファイルから画像を選択してインポートすることができます。

ナレッジベースでスマートデバイス用のオブジェクトを作成し、F5 キーを押すと、Android 用のエミュレーターが自動的に実行されます。この Web アプリケーションには、開発者メニューからもアクセスできます。

F5 キーを押します。

実行時

QR コード付きの開発者メニュー

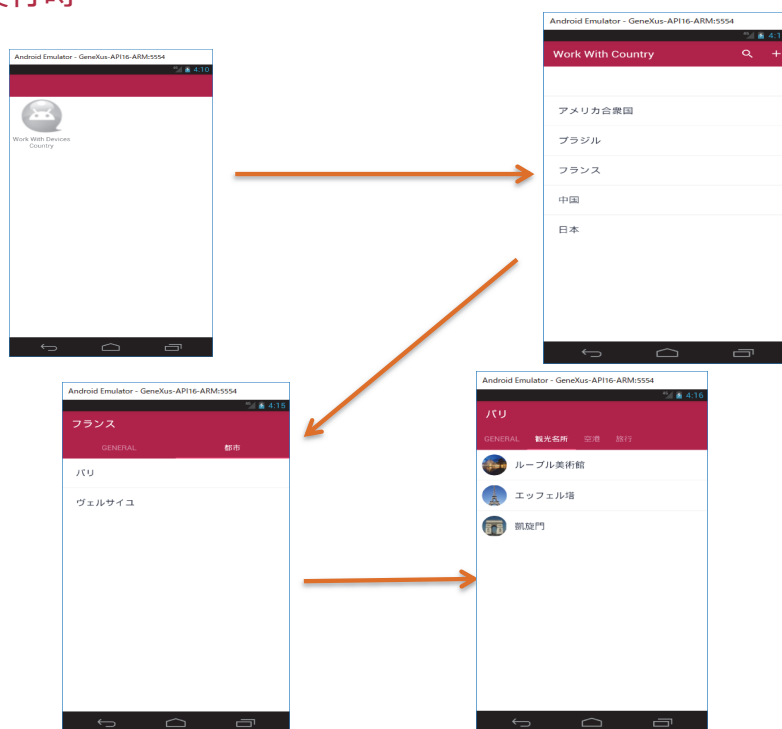


Android エミュレーターの自動実行

Web アプリケーションにアクセスするためのオプションとは別に、スマートデバイスにアプリケーションをダウンロードするための QR コードが開発者メニューに表示されます。

Android 用エミュレーターが起動します。

実行時



Menu が実行され、設定された画像が表示されます。

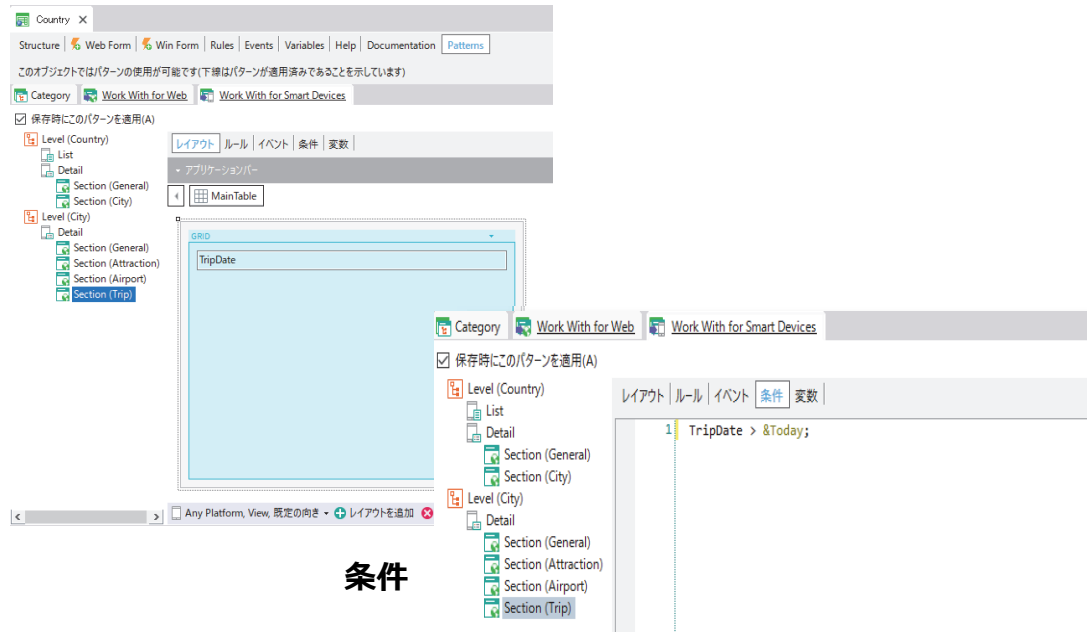
表示された画像を選択すると、登録されているすべての国がリストで表示されます。

たとえば、[フランス] を選択すると、フランスの国情報や都市が表示されます。

[パリ] を選択すると、パリの情報と関連情報 (観光名所、空港、ツアー) が表示されます。

Work With for Smart Devices のカスタマイズ

フィルタの条件



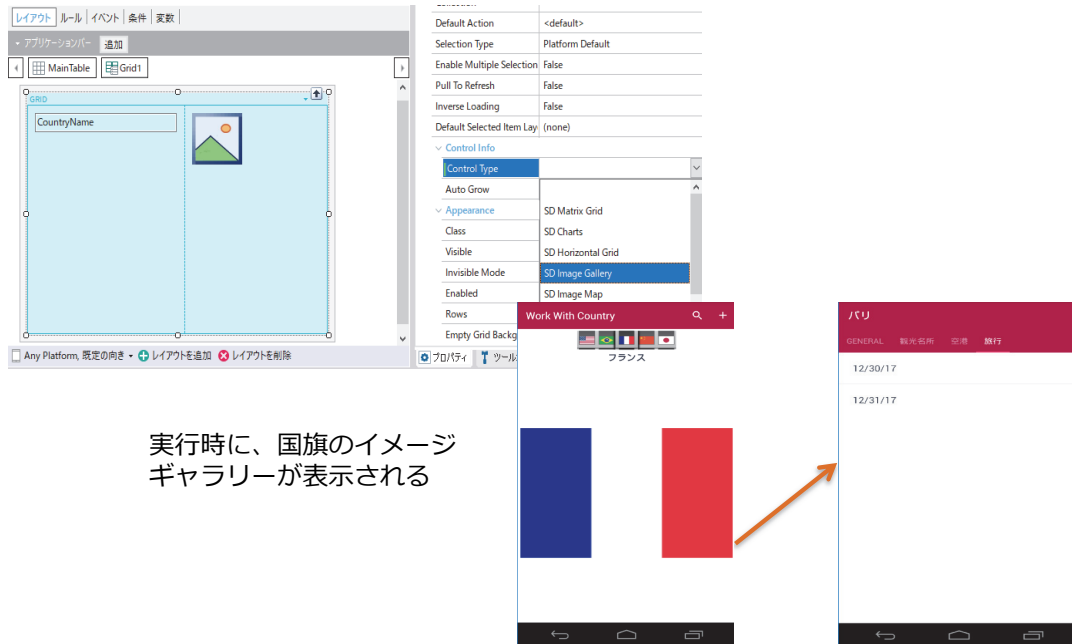
GeneXus に戻り、パターンインスタンスを少しカスタマイズします。

たとえば、現在の日付よりも後の日付のツアーのみを、つまり、今後のツアーのみを表示する場合を考えます。

[Conditions] に次のように入力します: `TripDate > &Today;`

Work With for Smart Devices のカスタマイズ

イメージギャラリータイプのグリッド: SD Image Gallery



国のリストを、単なる名前のリストではなく、より魅力的な形で表示したい場合があります。たとえば、各国の国旗を、国名とともに表示することができます。国旗が選択されると、関連情報にアクセスします。

そこで、Country トランザクションに新しい項目属性を定義しました: Image タイプの CountryFlag です。

Work With for Smart Devices の Country パターンに戻り、国旗のギャラリーとして表示される国のリストを定義します。

グリッドの [Control Type] プロパティで、[SD Image Gallery] を選択します。

[Data Attribute] プロパティで、表示する画像が含まれる項目属性、この例では [CountryFlag] を指定します。そして [Title Attribute] プロパティで、値が画像のタイトルになる項目属性を指定することができます。[CountryName] を選択します。

そして、アプリケーションを実行し、これらすべての変更を確認します。

再度 F5 キーを押します。エミュレーターが実行されます。ダッシュボードでアイコンを選択し、国のリスト画面を表示します。国のリストは国旗のギャラリー形式で表示されるようになりました。

[フランス] を選択すると国の情報が表示されます。関連情報から [パリ] を選択し、パリのツアーのリストを表示します。また、表示されるツアーは、現在の日付より後の日付のものです。

地理情報 (Geolocation)

セマンティックドメイン

ドメイン ×

フィルタ(F): モジュール(M): *ALL

名前	タイプ
Timezones	Character(60)
Effect	Character(20)
CallType	Character(20)
CryptoEncryptAlgorithm	Character(40)
CryptoHashAlgorithm	Character(40)
CryptoSignAlgorithm	Character(40)
TmMode	Character(3)
Address	VarChar(1K)
Component	Url, GeneXus
Email	VarChar(100)
Geolocation	Character(50)
Html	LongVarChar(2M)
Phone	Character(20)
APIAuthorizationStatus	Numeric(1.0)

Airport ×

Structure Web Form Win Form Rules Events Va

名前	タイプ
Airport	Airport
AirportId	Id
AirportName	Name
CountryId	Id
CountryName	Name
CityId	Id
CityName	Name
AirportGeolocation	Geolocation, GeneXus

Layout Rules Events Conditions Variables

Application Bar

MainTable

GRID

AirportName

Image

Geolocation

AirportGeolocation

Any Platform, View, Default Orientations Add Layout Delete Layout

[Section(Airport)] セクションの項目属性

次に、空港を見てみましょう。名前のみが表示されていますが、空港の位置を地図でも見ることができると、アプリケーションのユーザーの役に立ちます。

Airport トランザクションに、必要な情報を格納できる新しい項目属性を定義します。この情報は、デバイスで空港の位置を地図で表示するために必要な地理情報を格納できるデータタイプに基づいている必要があります。

GeneXus は、この目的のために Geolocation ドメインを提供しています。これは、経度と緯度を単位として地理情報を管理できるデータタイプです。項目属性は AirportGeolocation という名前にし、再編成を行い、各空港の緯度と経度を登録します。デバイスは、データベース内のこの情報を使用して対応する位置の地図を表示します。

さらに、アプリケーションをより魅力的にするために、各空港の画像を追加します。

そのため、AirportImage 項目属性も作成し、空港の写真を登録しました。

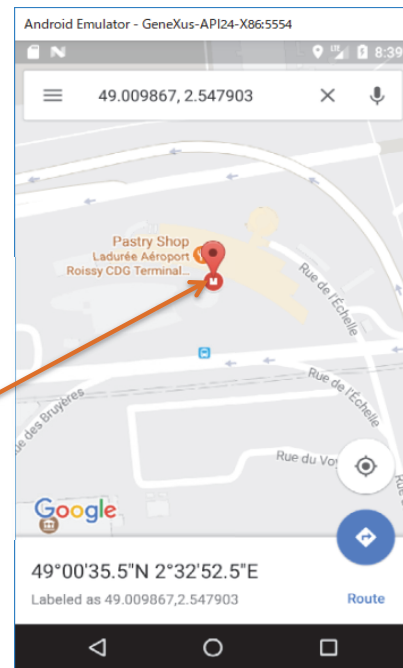
アプリケーションを実行する前に、[Work With for Smart Devices] ですべての情報が意図どおりに表示されるように変更します。

また、各項目属性の [Label Position] プロパティで、ラベル位置に [Top] を指定します。

AirportGeolocation 項目属性を挿入し、ラベルのキャプションに Geolocation のみが表示されるようにします。

地理情報 (Geolocation)

実行時



ここで、再度アプリケーションを実行して確認します。

Panel

新しい要望：国別の観光名所リストを国別にグループ化する



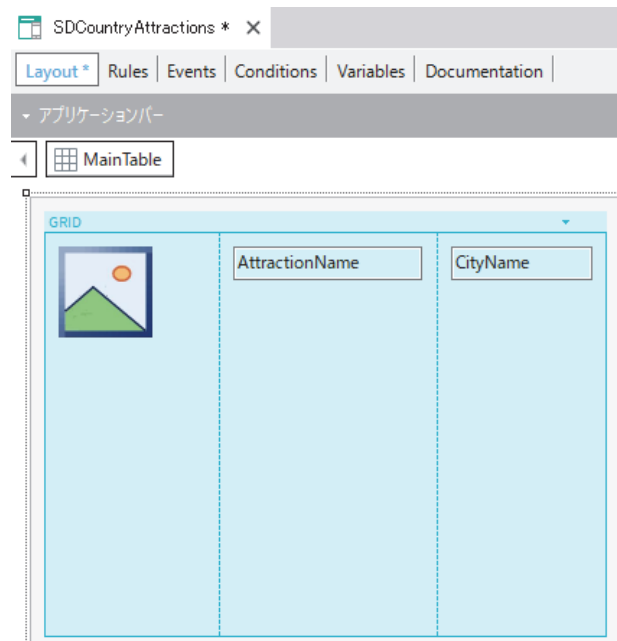
最後に、国のギャラリー画面でボタンを押すと、図のような画面が表示されるようにしたいと代理店から要望された場合について考えます。

この画面にはすべての国が一覧表示され、それぞれの国に登録されている観光名所が表示されます。

これによりエンドユーザーはそれぞれの国が持つ観光名所の数と種類を全体的に確認することができます。

Panel

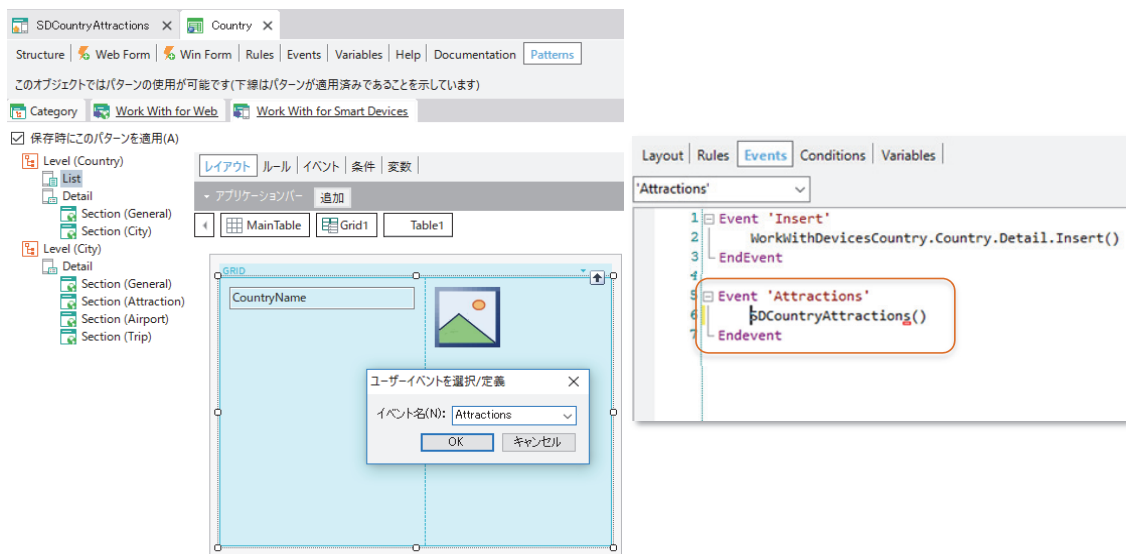
表示する項目属性を含むグリッド



まず、IDE より観光名所の一覧となる新しいオブジェクトを作成します。タイプは [Panel] で、名前は SDCountryAttractionsとします。
ツールボックスからグリッドをドラッグし、AttractionName、AttractionPhoto、CityName の項目属性を挿入して保存します。

Panel

国のリストから SD パネルを呼び出し



エンドユーザーがメイン画面から SDCountryAttractions にアクセスできるように、Work With for Smart Devices オブジェクトの国のリストにボタンを追加します。

ツールボックスからボタンをアプリケーションバーにドラッグします。

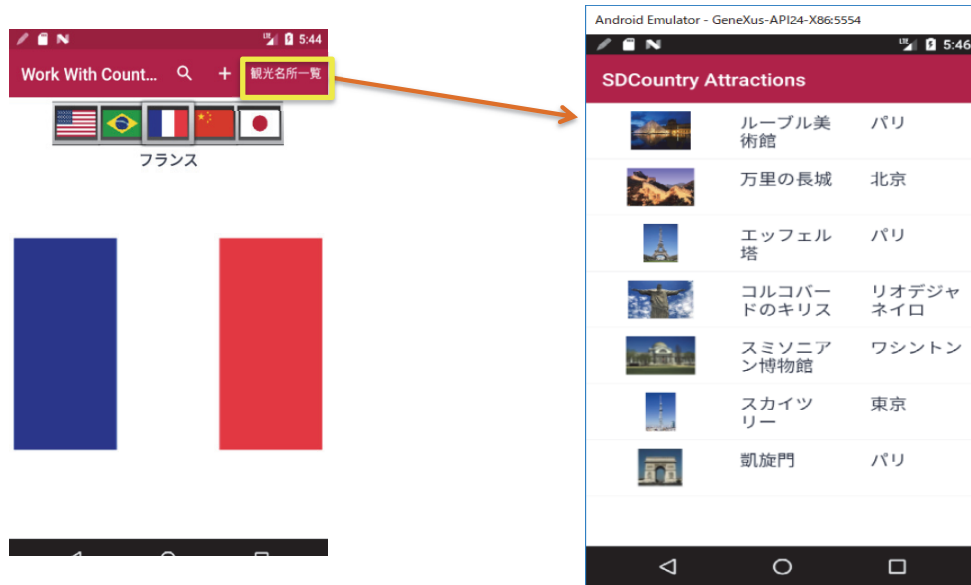
ボタンのイベント名に次のように入力します: Attractions

そして、ボタンの [Caption] プロパティに「観光名所一覧」と指定します。

ボタンをダブルクリックして [Events] エlementに移動します。
[KB エクスプローラー] から SDCountryAttractions オブジェクトをドラッグし、Web アプリケーションの場合と同様に 2 つの丸括弧を使用して呼び出しを完了します。

Panel

情報は表示されるが、グループ分けされていない



F5 キーを押します。

リスト表示でエミュレーターのメニューボタンを選択すると、新しいオプションが表示されます。観光名所一覧ボタンを選択すると、すべての観光名所が表示されます。

このリスト表示ではパリ、北京、パリと、都市が繰り返し表示されており、当初の国別にするという要件が考慮されていません。別の形式にするため、観光名所を国ごとにグループ分けします。

GeneXus に戻り、スマートデバイス向けパネルを編集します。

Panel

観光名所を CountryName 別にグループ分け

グリッドレベルの
[Orders] プロパティ

Orders

- Orders
- Order (Countries)
- CountryName
- Break By

order: Order (Countries)

Name	Countries
Break by	True
Enable Alpha Indexer	False
Break by up to	CountryName
Description Attribute	

breakBy: Break By

Description Attribute	CountryName

グリッドの [Orders] プロパティで Countries という名前の新しいオーダーを定義します。このオーダーの項目属性として CountryName 項目属性を追加します。また、図の通り、[Break by] プロパティの値を [True] に定義します。

これにより、観光名所が国ごとにグループ分けされます。

アプリケーションを再度実行して観光名所一覧ボタンを押すと、観光名所が国ごとにグループ分けされて表示されます。

