GeneXus™

Grow thru knowledge

GeneXus 試用版:

チュートリアル

GeneXus™
www.genexus.com

MONTEVIDEO - URUGUAY Av. Italia 6201- Edif. Los Pinos, P1 (598) 2601 2082
CHICAGO - USA 1143 W Rundell PL, Suite 200 (1 312) 836 9152
SÃO PAULO - BRASIL Rua Samuel Morse 120 Conj. 141 (5511) 55026722
CIUDAD DE MÉXICO - MÉXICO Hegel N° 221, Piso 2 (5255) 52554733
TOKYO - JAPAN 2 27 3 Gotanda Front (813) 6303 9381
Nishi Gotanda, Shinagawa ku (813) 6303 9980

Copyright © Artech Consultores S. R. L. 1988-2013.

All rights reserved. 本書は、Artech Consultores S.R.L. の明示的同意なしには如何なるメディアにも複写することはできません。本書の内容は個人的使用のみを目的として提供するものです。

登録商標

Artech および GeneXus は Artech Consultores S.R.L. の商標または登録商標です。本書において取り上げているその他の商標はすべて、それぞれの所有財産です。

GeneXus Japan Inc.
Copyright © 2013 GeneXus Japan Inc. All rights reserved.

本書には正確な情報を記載するように努めました。ただし、誤植や制作上の誤記がないことを保証するものではありません。なお、本書に記載されている画面はソフトウェアの更新などにより予告なく変更される場合があります。ご了承ください。
本書に記載のその他の製品名および会社名は、各社の商標または登録商標です。GeneXus Japan Inc. は他社製品の性能または使用につきましては一切の責任を負いません。

ジェネクサス・ジャパン株式会社
〒141-0031　東京都品川区西五反田 2 丁目 27 番 3 号 五反田フロント
電話番号：03-6303-9381　　FAX：03-6303-9980
http://www.genexus.jp

目次

序論 **3**
GeneXus 試用版 **3**
機能制限 3
ライセンス制限 3
プロトタイプサーバーの制限 3
ライセンス使用許諾 4
技術サポート 5
インストールとオーソライズ **5**
インストール要件 5
.NET ジェネレーター要件 6
スマート デバイス ジェネレーター要件 6
オーソライズ **7**
はじめに：ステップバイステップ チュートリアル **10**
使用されるアイコン 10
パート 1：GeneXus の紹介 **11**
GeneXus とは何か？ 11
パート 2：最初のアプリケーション **11**
知識に基づくアプリケーションの設計 11
ステップ 0：ケーススタディ 12
ステップ 1：開発環境 13
ステップ 2：ナレッジベースの作成および環境の定義 14
ステップ 3：トランザクションオブジェクトの作成 16
ステップ 4：トランザクション構造の定義 17
ステップ 5：計算された項目の定義 ⇨ 式 20
ステップ 6：GeneXus が推論したデータモデルを確認する 21
ステップ 7：トランザクションオブジェクトのフォームを確認する 23
ステップ 8：アプリケーションの実行 25
ステップ 9：アプリケーションのテスト 27
ステップ 10：ビジネスルールの追加⇨（ルール） 28
ステップ 11：Customer トランザクションオブジェクトの作成 30
ステップ 12：データモデルに加えられた変更を確認する 33
ステップ 13：分析レポートを確認する 35
パート 3：開発 パターンからアプリケーションを生成する **36**
開発の一部にパターンを使用する 36
ステップ 14：Web アプリケーションのパターン 36
ステップ 15：スマートデバイスアプリケーション用のパターン 38
要約 **45**
連絡先とリソース **46**
GeneXus コミュニティ 46
サポート 46
購入方法 46
参考 **46**

序論

この文書は、GeneXus が提唱する知識（ナレッジ）ベース開発方法論の潜在能力を、その主要な機能を体験しながら、見つけ出すための手助けをすることを目的としています。

- データモデルの自動設計
- コードの自動生成
- データベースとコードの自動的な保守
- マルチプラットフォーム開発と実装

GENEXUS 試用版

機能制限

GeneXus 試用版は全機能を網羅しています。利用できるジェネレーター(.Net および SmartDevice)はシングルサイトキーでオーソライズすることができます。キーは、起動してから 60 日後に期限切れになりますが。延長は可能です(再オーソライズ)。ただし、ナレッジベースに作成できる GeneXus オブジェクトと項目属性の最大数に次のような制限があります：

- 項目属性： 最大 90 個
- オブジェクト： 最大 140 個

ライセンス制限

GeneXus 試用版は、評価とトレーニングの目的にのみ使用が許可されています。ライセンスは、シングルユーザーのローカルインストールのみとなります。

プロトタイプサーバーの制限

生成されたアプリケーションには、このバージョンのサーバー側のコンポーネントが含まれ、GeneXus International によって提供される Web サーバー（クラウドコンピューティング）上でのみ実行することができます。それらをローカル、または企業プラットフォーム（業務）で実行するには、フルバージョンが必要です。GeneXus 試用版の試用期限は、オーソライズ後 60 日です。これはユーザーおよびプロトタイプサーバーによって生成されるアプリケーションおよびデータベースにも同様に適用されます。

ライセンス使用許諾

GeneXus X Evolution 2 試用版使用権許諾書

1. 1.本許諾書は、システム開発の設計・生成・保守に関して業務知識に基づきデータベースとアプリケーションを自動的に設計し、生成し、維持し保守する創造的なツールであるGeneXus X Evolution 2 試用版を使用する場合を対象とします。

2. GeneXusと同様、お客様はGeneXusに関連して紹介されたGeneXusの知的創造物である関連製品（**GXサーバー、ジェネクサス、GXフロー、GXプローラー、GXポータル、GXクエリー、K2B**）及び**Artech Consultores S.R.L.(以下「Artech社」と称する)社**の名前とロゴやその他の商標について商標登録されているか否かにかかわらず著作権、特許権、実用新案権、商標権その他の知的財産権は全てArtech社に帰属し、直接的／間接的を問わず、いかなる所有権の移動を意味しませんし、ユーザーに本件のライセンスを譲渡する権利を与えないことを、お客さまも認識しているものとします。

3. お客様はGeneXus X Evolution 2 試用版を使用することに同意をしたとき、GeneXusの動作原理に関するアイディアや技術を自らの利益の為に公開したり使用したりしないことを約束するものとします。特にお客様はリバースエンジニアリングによってそのコードの解読や、または、第三者の解読を支援しない義務を負います。

4. GeneXus（ディスク、キー、制御機構など）の不正使用を防ぐための保護メカニズムを無効とする行為や、キーを複製する行為が認められることは絶対にありません。Artech社はこれらの保護メカニズムをいつでも変更し、新たなメカニズムを加える権利を保留します。

5. 以上の一般的な条件のもとで、お客様はGeneXus X Evolution 2 試用版ライセンスを利用するにあたり以下の項目を必ず遵守しなければなりません。：

a) 分離し独立したコンピュータ上で使用権を稼働させるには、各々別の使用権が必要です。お客様は、2台以上のコンピュータで同時に一つの使用権を使用しないことを約束するものとします。

b) GeneXusのライセンスはお客様専用に作られた排他的な製品ではなく、使用権の他への移転はできません。GeneXus X Evolution 2 試用版を使って生成したプログラムは商用を目的とした使用はしないこととします。

c) お客様はGeneXusのプログラムとマニュアル類を個人的に使用、または評価目的にのみ使用するものとします。そして、お客様はそれらの原本や複製物を第三者に渡さないことを約束します。

d) GeneXus X Evolution 2 試用版の使用に関連し何らかの損害をお客様が直接的または間接的に被った場合、Artech社は全てのケースにおいて責務は負いません。

e) お客様は付随する取扱説明書や仕様を遵守することを約束したうえでGeneXusのライセンスを使用するものとします。

f) 全体的もしくは部分的使用に関わらずGeneXus X Evolution 2 試用版の商用使用は禁じられています。アプリケーション開発をする場合お客様は商用ライセンスを購入するものとします。

g) GeneXus X Evolution 2 試用版の使用は60日間有効です。その内容は.NETジェネレーターとSQLデータベースサポート、およびスマートデバイスジェネレーターのみであり、それを使って最大90の項目属性と140のオブジェクトを作成することが可能です。

6 . お客様が本許諾書に定める使用方法に違反しその他本許諾書記載の条項に違反して、ARTech社、ジェネクサス・ジャパン株式会社、他の使用者に損害が発生した場合は、お客様に対し損害賠償請求を行うことができるものとします。

7. ライセンス付与されたソフトウェア製品をインストールすることによってお客様は本許諾書に定める全ての項目について同意するものとします。

技術サポート

試用版のインストールおよびライセンスについてご不明な点がございましたら、次のアドレスまでお問い合わせください：
info@genexus.jp

GeneXus のさまざまな技術的なサポート、トレーニングサービス、およびリソースの詳細については、以下を参照してください。
http://www.genexus.com/community-and-support-jp/community?ja および
http://www.genexus.com/community-and-support-jp/support?ja

追加情報については、お近くの販売店 http://www.genexus.com/company-jp/sub-distributors-in-japan?ja または toiawase@genexus.jp にお問い合わせください。

インストールとオーソライズ

インストール要件

GeneXus 試用版は以下の製品を含みます：

- **GeneXus モデラー**

  Integrated Development Environment（IDE）は、「統合開発環境」です。これにより、アプリケーションが使用する稼働プラットフォームに関係なく、ビジネスアプリケーションを開発および保守することができます。

- **GeneXus ジェネレーター**

  GeneXus は、市場の主なプラットフォーム向けにネイティブコードを生成します。GeneXus が対応しているジェネレーターは、C#、Java、.Net Mobile、Ruby、および SmartDevice になります。GeneXus 試用版で提供されるジェネレーターは、.NET ジェネレーターおよびスマートデバイスジェネレーター(iOS、Android、Blackberry)です。

GeneXus および GeneXus で作成されたアプリケーションを実行するために必要なハードウェアおよびソフトウェア要件は次のとおりです。

ハードウェア要件	プロセッサ：1 GHz 以上（マルチコアを推奨）
	メモリ：1 GB 以上の RAM（2 GB を推奨）
	ハードディスク： 300 MB 以上のディスク領域（インストール用）。 GeneXus アプリケーションを作成するには、ナレッジベースおよび生成するコードのために追加領域または共有ディスクを使用する必要があります。

ソフトウェア要件	Microsoft Windows　2000、XP SP2　以上
	Microsoft .NET Framework 3.5 SP1 以上
	Microsoft Internet Explorer 6.0 SP1 以上
	Microsoft SQL Server 2005 以上（Express、Standard、その他任意のエディション）[1]

[1] GeneXus 試用版をインストールすると、Microsoft .NET Framework 3.5 SP1 および Microsoft Internet Explorer 8.0 が自動的にインストールされます。

[2] Microsoft SQL Server Express Edition は、無料であり、再配布可能なバージョンです。Microsoft SQL Server がインストールされていない場合、GeneXus 試用版をインストールする際に、Microsoft SQL Server がインストールされます。GeneXus 試用版からインストールされた Microsoft SQL Server の管理者ユーザーは「sa」で、パスワードは「genexusxtrial」です。

.NET ジェネレーター要件

Microsoft .NET Framework 3.5 SP1 以上

スマート デバイス ジェネレーター要件

Android

- Java JDK version 6
- Android SDK (Google API 7 および Google API 17)

詳細情報：Android 前提条件

Blackberry

- Java JDK version 6
- Blackberry JDE 5.0
- Blackberry シミュレーター

詳細情報：Blackberry 前提条件

Apple (iOS)

- Knowledge Base Navigator（iOS デバイス）：

  生成したアプリケーションをプロトタイプするための Mac がない場合、Apple Store で入手可能な Knowledge Base Navigator を使って、直接デバイス（iPhone、iPad、iPod）にプロトタイプすることができます。

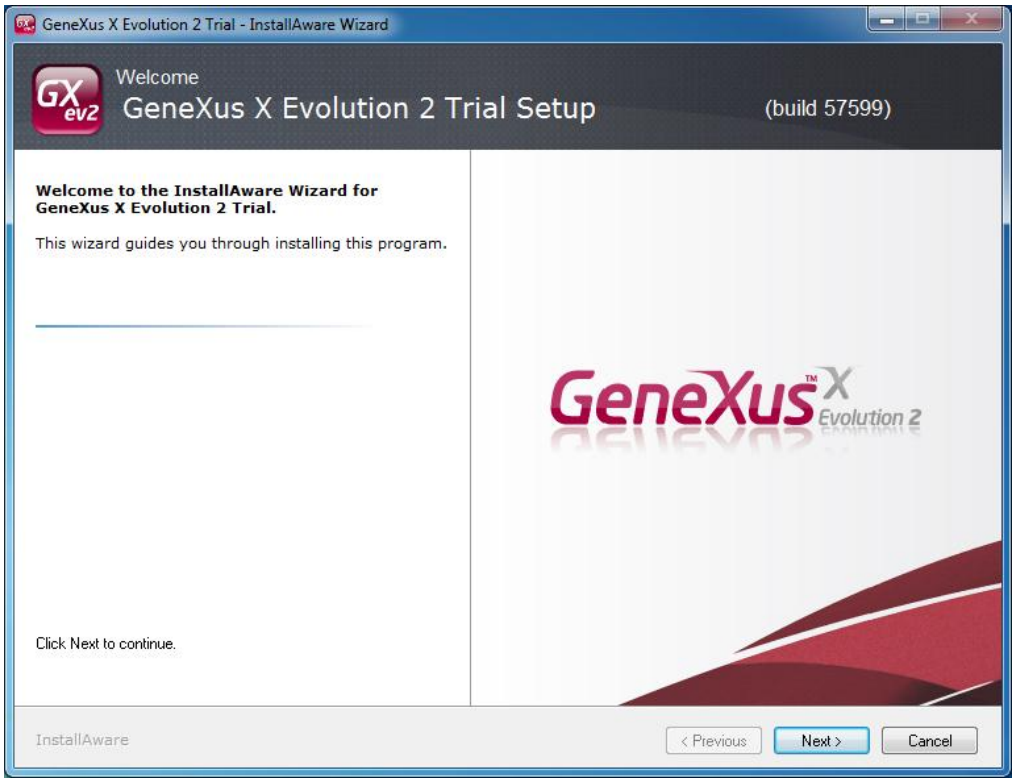
- Xcode (Mac)：

  Mac がある場合、iPhone / iPad のシミュレーターまたは直接デバイス上で、生成されたアプリケーションをプロトタイプすることができます。

詳細情報：iOS 前提条件

オーソライズ

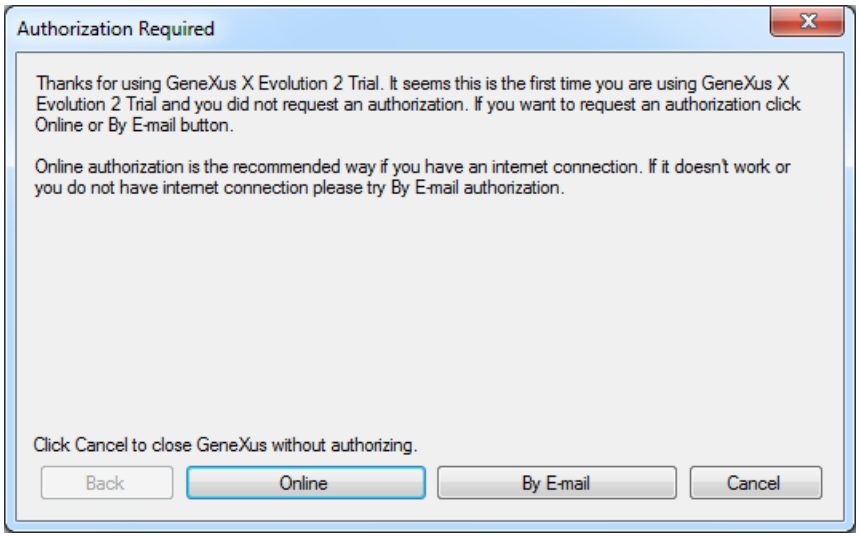
1. GeneXus 試用版のセットアップファイル(GeneXusTrial.exe)を実行します。



図：GeneXus 試用版インストールウィザード

2. インストールウィザードに表示される手順に従ってインストールします。

3. GeneXus 試用版をはじめて使用する場合、起動するためにこの製品をオーソライズする必要があります。オーソライズはオンラインまたはメールで行うことができます。インターネットにアクセスできる環境であれば、オンラインでのオーソライズをお勧めします。

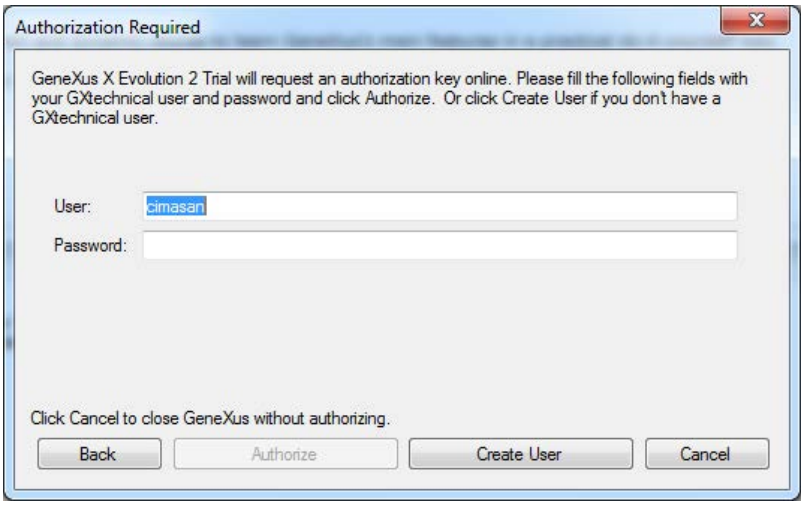


図：GeneXus 試用版の登録

4. オーソライズを行うには、GXtechnical のユーザーアカウントを持っている必要があります。[Create User(ユーザーの作成)] ボタンをクリックすると、ユーザー登録することができます。

参考：上記の「Create User」ボタンをクリックすると、英語のユーザー登録ページが表示されます。GeneXus Japan Web サイト にある日本語のユーザー登録ページで事前に登録することもできます：

http://www.genexus.com/trial-jp/UserRegistration?ja



図：ユーザー名とパスワードの入力

5. すべての必要なデータの設定が完了したら、すぐに 60 日間の GeneXus 試用版が有効になります。
6. [By E-mail(E メール)]オプションを選択した場合、オーソライズを行うための、サイトコードを記載したメールが作成されます。**サイトキー**が記載されたメールを受信します。これを実行するには、GXtechnical のアカウントと連携するメールアドレスを使用する必要があります。
7. アクティベーション用のサイトキーが同じメールアドレスに送信されます。「オーソライズ請求」ウィンドウのサイトキー入力用のフィールドにメールに記載されている**サイトキー**をコピー＆ペーストしてから[**Authorize(オーソライズ)**]をクリックします。
8. これで GeneXus 試用版を使用する準備ができました。
9. GeneXus 試用版を有効にする方法についてご不明な点がある場合は、info@genexus.jp までお問い合わせください。

重要：GeneXus 試用版で生成したアプリケーションは Artech 社が用意したクラウドサービスにデプロイされます。プロキシサーバー経由でクラウドに接続する場合には、以下の例外をプロキシサーバーの設定に追加する必要があります：

- <server-name> tcp/80
- <server-name> tcp/443
- <server-name> tcp/3306 (mysql)
- <server-name> tcp/1433 (ms sqlserver)

GeneXus 試用版の場合、<server-name>は [trialapps2.genexus.com]となります。

はじめに：ステップバイステップ チュートリアル

このチュートリアルの目的は、はじめての GeneXus を体験しながら GeneXus 方法論の基礎を学んでいくことです。最終的には、想像よりも早く、複雑なビジネスアプリケーションを作成することができるようになります。

このステップバイステップの実習では、単純なビジネスアプリケーションを作成し、GeneXus 試用版がサポートするプラットホーム（.NET）上でそれを実装します。さらに、Android、iOS、Blackberry などのスマート デバイス プラットフォームのアプリケーションを生成することができるようになります。GeneXus が対応しているプラットフォームについて詳しくは以下を参照してください：http://www.genexus.com/products-japan/genexus?ja

このチュートリアルに取り組みながら、GeneXus の主な機能を体験していきます：

- ナレッジベースアプリケーションの設計
- 知的なデータベース生成
- コードの自動生成
- 完全に機能的なアプリケーション
- インクレメンタル開発およびアプリケーションの自動保守
- マルチプラットフォーム開発

使用されるアイコン

!

このこのアイコンは重要な概念を意味します。

このアイコンは注記を意味します。

パート 1：GENEXUS の紹介

GENEXUS とは何か？

基本的には、GeneXus はプログラムを生成するプログラムです。

これはユーザー要件から始まるツールであり、ナレッジベースと呼ばれるものに知識をカプセル化します。GeneXus はその知識を体系化し、その後自動的にデータベースおよびプログラムを設計、作成そして保守します。

! 要約すると、GeneXus は複数のプラットフォームでクリティカルなアプリケーションの迅速な開発を成し遂げるために、プログラムとデータベースの自動的な**設計、作成**、および**保守**を実現します。

パート 2：最初のアプリケーション

知識に基づくアプリケーションの設計

このセクションでは、ユーザー要件に基づいて GeneXus ナレッジベースを作成し、アプリケーションを設計する方法を学習します。今回のケースでは、請求書を最初に設計して、次にセールス管理システムで要求されているその他の構成要素を設計します。

! ビジネスアナリスト　対　開発者

エンドユーザーのニーズを理解することは、ソフトウェア開発において、自動化することができない作業の 1 つと言えます。そのため、GeneXus 開発者はプログラマー、コーダーあるいは単なる開発者ではなく、**ビジネスアナリスト**と呼ばれています。

! アプリケーションのプログラミング 対 アプリケーションの宣言

GeneXus 方法論は、アプリケーションが取り扱うエンドユーザーの実体（エンティティ）を**記述**することに基づいています(有形および無形両方の現実オブジェクト)。これは、エンティティにエンドユーザー要件を高い抽出概念レベルで**記述**するこによって行われます。この理由により、GeneXus では**宣言プログラミング**を使用します。このように、**ビジネスアナリスト**は現実を記述し、GeneXus は要求された機能を満たすためにデータモデルを特定のデータベースに生成し、アプリケーションプログラムを生成します。この現実が変わる場合、ビジネスアナリストは変更を記述するだけで、GeneXus がその変更を自動的にデータモデルとプログラムに反映してくれます。

ステップ 0：ケーススタディ

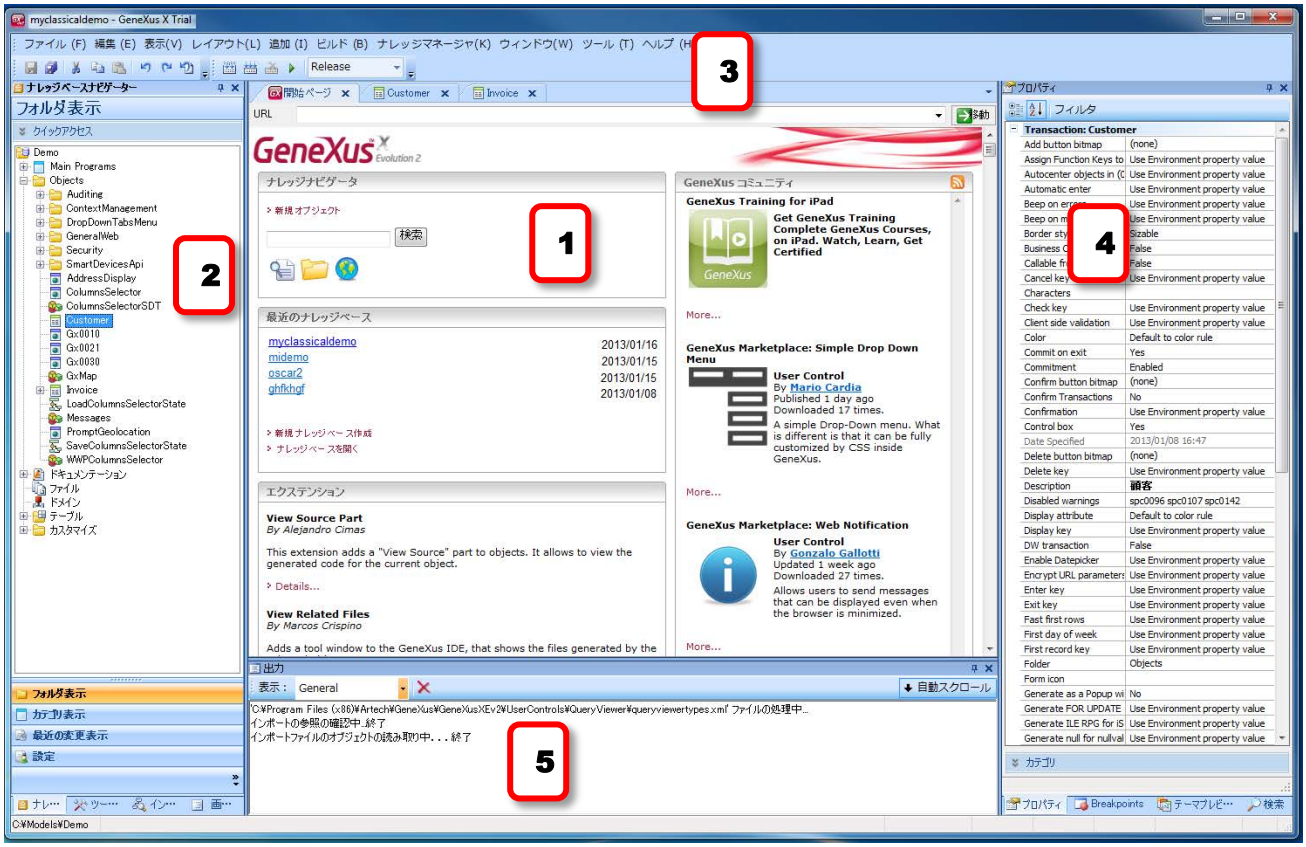
ある会社が請求書システムを管理するアプリケーションを必要としているとしましょう。現在、彼らはそのアプリケーションを持っていないので、製品コードが記載された大きなカタログを参照しながら請求書を手作業で処理しています。このチュートリアルは GeneXus 方法論の基礎を学習しながら、請求書（Invoice）、顧客（Customer）、および製品（Product）を処理していきます。

このチュートリアルでは、GeneXus に現状の実務を記述することで .NET ジェネレーターと SQL Server データベースを使用する Web 請求書システムが生成されることを学習します。さらに、顧客の詳細情報を問い合わせたり、変更したりするための、対応するスマート デバイス アプリケーションが生成されます。

ステップ 1：開発環境

GeneXus を開くと、下図のようなインターフェースが表示されます。このインターフェースは IDE（統合開発環境）と呼ばれます。それは各開発者にとって直観的で使いやすく、そして独自設定が可能なインターフェースです。

このインターフェースは、複数のウィンドウに分離されています：



図：開発環境（IDE）

1. **メインウィンドウ（開始ページ）**：RSS を使ってダイナミックに、ツールに関する技術情報やユーザーコミュニティを表示する開始ページです。この情報を得るためにはインターネット接続が必要です。パソコンがインターネットに接続されていない場合、GeneXus コミュニティで発表されるニュースを表示することができません。

2. **ナレッジベースナビゲーター**

3. **ツールバー**

4. **[プロパティ] ウィンドウ**

5. **出力**

ステップ 2：ナレッジベースの作成および環境の定義

GeneXus でアプリケーションを作成するための第一歩は、ナレッジベースを作成して作業環境を定義することです。

!

ナレッジベース（KB）

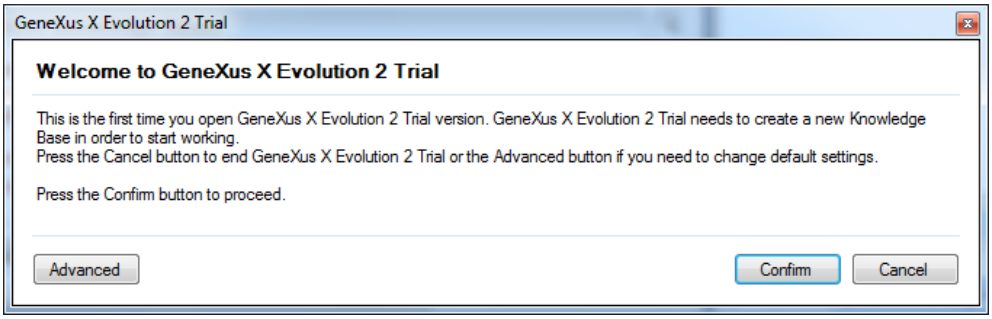
複数のプラットフォームで、アプリケーションを生成するために必要な情報を保管する場所です。言い換えると、**現実の記述**を保管する場所のことです。

!

環境

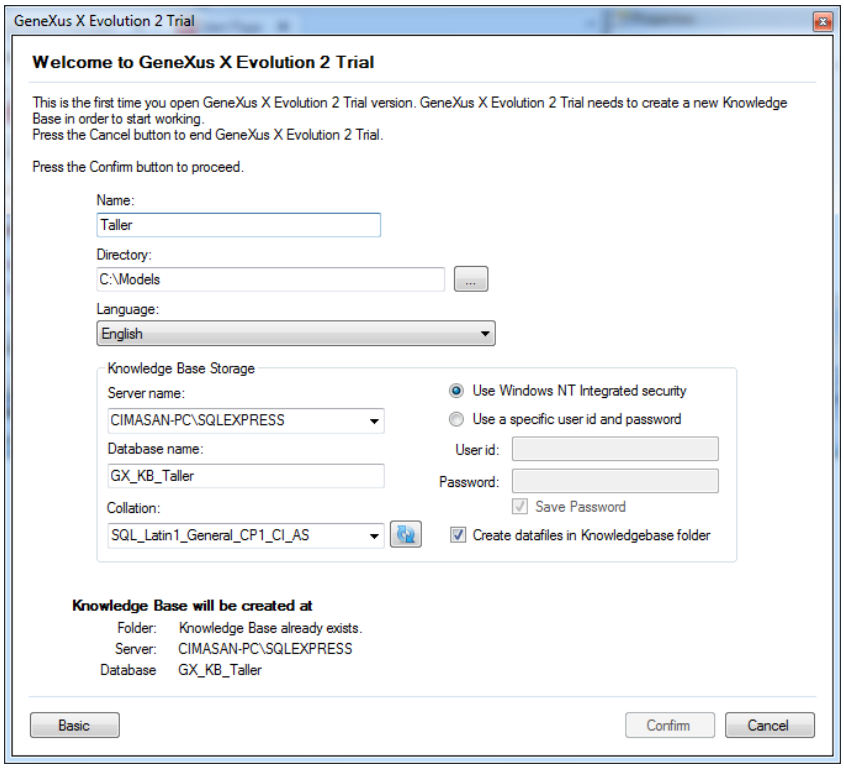
特定のソフトウェアプラットフォームで、アプリケーションの生成および保守を行うためには、実行プラットフォームに関連する情報をすべて**環境**に定義します（例：プラットフォームのジェネレーター、データベース接続情報、DBMS、ユーザーインターフェース、およびその他のプラットフォームのプロパティ）。このためには、環境ごとに DBMS、対象言語およびパラメーターを設定します。GeneXus は、選択されたプラットフォーム上でデータベーススキーマとすべてのプログラムを生成および保守します。従って、GeneXus アナリストは対象プラットフォームの深い知識を必要としません。

GeneXus 試用版をはじめて開いた場合、GeneXus がナレッジベースを自動的に作成するメッセージが表示されます。はじめて試用版を開くと、ナレッジベースは SQL ＋ .NET 環境で自動的に作成されます。



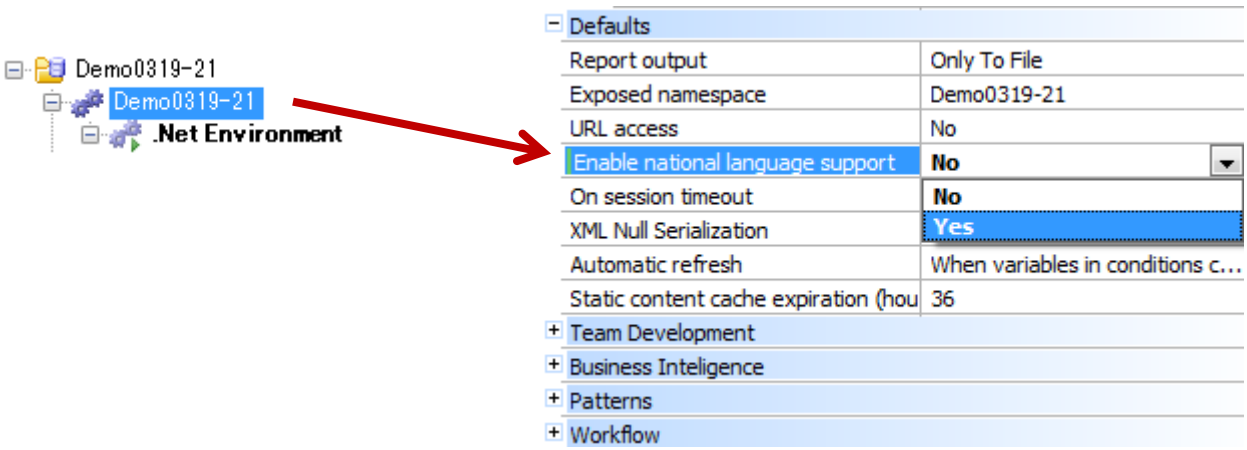
図：GeneXus X Evolution 2 試用版の初期画面

[**Advanced**（詳細設定）]ボタンをクリックすると、ナレッジベース名とディレクトリを指定することができます。



図：新規ナレッジベース作成 - 詳細設定

重要：生成した Web アプリケーションでマルチバイト文字を使用する場合、[Preferences] の [Version] ノードを右クリックして、[Properties（プロパティ）]を選択し、プロパティの一覧に表示される[Defaults]にある[Enable national language support]プロパティを[Yes]に設定する必要があります。



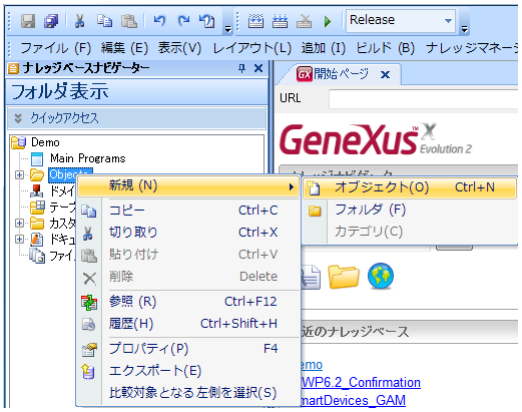
ステップ 3：トランザクションオブジェクトの作成

ユーザー要件を GeneXus オブジェクトに定義することが目的です。

> **トランザクションオブジェクト**
>
> これは、アプリケーションが取り扱う現実のオブジェクトを表現します。定義済みのトランザクションは、アプリケーションのデータモデル（第 3 正規形）を推論するのに用いられます。GeneXus は、エンドユーザーが物理的データベースにインタラクティブにデータの新規登録、更新、および削除を行うための、アプリケーションプログラムを生成するために、トランザクションオブジェクトを使用します。

一旦ナレッジベースが作成されると、次のステップは請求書を表す最初のトランザクションを作成することになります。これを実行するには、次の手順に従います：

1. **Objects を右クリックして[新規]→[オブジェクト]** と選択するか、[ファイル]メニューから[新規]→[オブジェクト]と選択します。また、**Ctrl＋N** のショートカットを使用して、新しいオブジェクトを作成することもできます。



図：新しいオブジェクトの作成

2. 次のオブジェクトのタイプを選択します：**Transaction (トランザクション)**

3. 次のオブジェクト名を入力します：*Invoice*

4. [**作成**]をクリックします。

図：新規オブジェクト ダイアログ ボックス

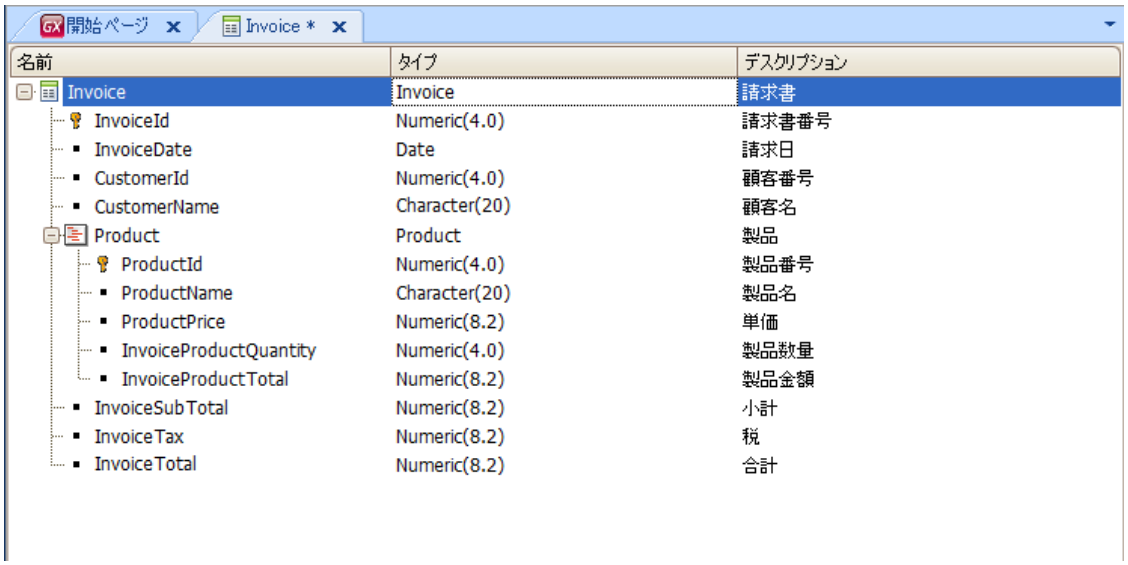
ステップ 4：トランザクション構造の定義

次の手順では、Invoice オブジェクトに属している項目属性の定義とそれらがどのように関連しているのかを記述します。

1. 以下の表で示されているとおりに、Invoice トランザクションの[Structure]タブ内に項目属性名、データタイプ、およびデスクリプションを入力します。項目属性名、データタイプ、およびデスクリプション間を移動するには、Tab キーを使用します。新しい項目属性を追加するには Enter キーを使用します。

項目属性	**タイプ**	**デスクリプション**
InvoiceId	Numeric(4.0)	請求書番号
InvoiceDate	Date	請求日
CustomerId	Numeric(4.0)	顧客番号
CustomerName	Character(20)	顧客名
ここまでで、請求書のヘッダー項目を入力しました。次に新しい明細行を入力します。これを実行するには、Ctrl + 右矢印キーをクリックしてデータ構造に新しいレベルを追加します。		
ProductId	Numeric(4.0)	製品番号
ProductName	Character(20)	製品名
ProductPrice	Numeric(8.2)	単価
InvoiceProductQuantity	Numeric(4.0)	製品数量
InvoiceProductTotal	Numeric(8.2)	製品金額
Enter キーおよび　Ctrl + 左矢印キーをクリックすると、ヘッダーレベルに戻ります。次にフッターを入力します。		
InvoiceSubtotal	Numeric(8.2)	小計
InvoiceTax	Numeric(8.2)	税
InvoiceTotal	Numeric(8.2)	合計

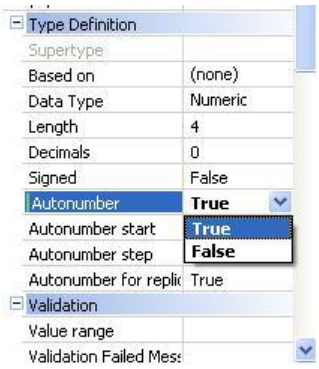
既定では、各レベルの最初の項目属性はそのレベルの主キーとして定義されています。項目属性を右クリックして「主キーの設定・解除」（Ctrl ＋ K）を選択することによって、変更することができます。主キー項目属性は、名前の左にある小さい鍵（ ）のアイコンで示されています。この例では、InvoiceId はレベル 1 の識別子です。ProductId 属性は、レベル 2 の識別子です。ある請求書番号（InvoiceId）に対して ProductId 項目属性の値が異なる行で繰り返されないことを意味します。



図：Invoice（請求書）構造

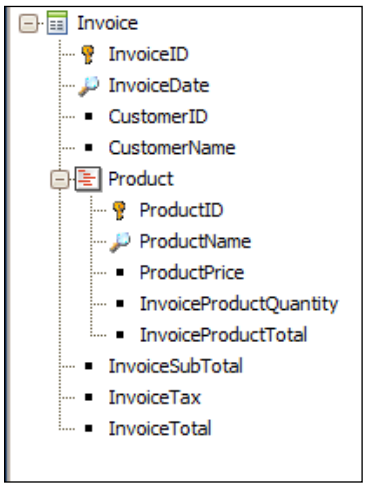
2. [InvoiceId]を選択してから画面の右側にマウスを移動させると、[プロパティ]メニューが表示されます。

3. [*Type Definition*] オプションを展開して、[**Autonumber**] プロパティを[**True**]に設定します。生成されるプログラムは、請求書番号が自動採番されます。



図：[Autonumber] プロパティ

4. 編集ツールバーにある**保存**（ ）ボタンを押して、トランザクション構造を保存します。このツールバーが見当たらない場合は、メニューバーを右クリックして「編集」オプションにチェックを入れます。



これで 2 つのレベルから構成されるトランザクションが定義されました：

- レベル 1（Invoice）には、請求書ヘッダーに必要なすべての情報を指定。
- ネストされたレベルには、請求書の明細行で繰り返される情報を指定。

!

普遍的な関係仮定（UNIVERSAL RELATIONSHIP ASSUMPTION = URA）

GeneXus 方法論の主な要素は、同じ名前を持っている項目属性が同じ項目属性であるという仮定です。これは普遍的な関係仮定（URA）と呼ばれています。

- 概念的に等しいものは、すべて同じ名前でなければなりません。
- 異なる概念を持つものは、同じ名前であっては**いけません**。

これにより、同じ項目属性をほかの GeneXus オブジェクト（トランザクション、プロシージャーなど）で参照するだけで使用できます。GeneXus は、項目属性の名前に基づいてデータモデルのテーブル間の関係（外部キー）を確立します。

ステップ 5：計算された項目の定義 ⇨ 式

!

式

式は、ほかの項目属性の値から推論することができる項目属性です。式項目属性は、通常の項目属性と同様に定義します。例えば、名前、データタイプ、およびデスクリプションを定義し、さらに計算式を定義します。

- 特に明記していない限り、式が定義される項目属性はデータベースに保管されません（**仮想項目属性**になります）。
- 式は、ナレッジベースに対して**グローバル**です。それらは定義されたトランザクションの範囲内だけで有効ではなく、ナレッジベース全体の中で有効です。項目属性がトランザクションから、あるいはほかの GeneXus オブジェクト（レポート、Web パネルなど）から参照される度に式が実行されることを意味しています。
- **ユーザーが定義する変数です。**これらの変数は式に記述することができません。なぜなら、変数はローカルであるため定義されたオブジェクト内でしか使用できません。

次の式項目属性を定義してみましょう：

InvoiceProductTotal = ProductPrice * InvoiceProductQuantity

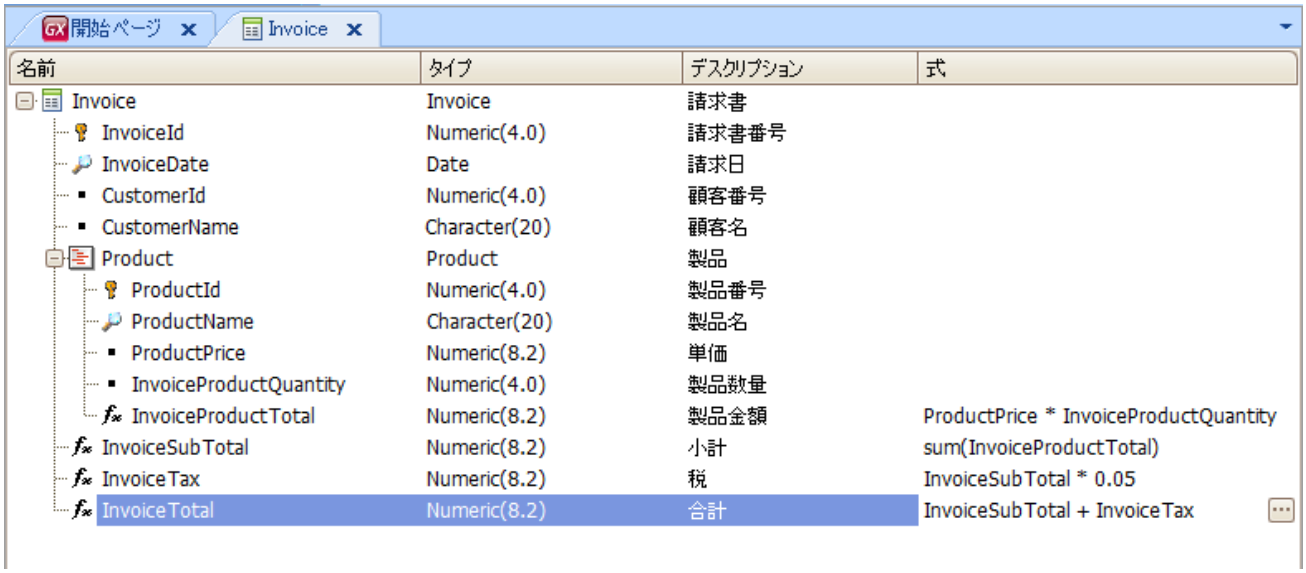
InvoiceSubTotal = SUM(InvoiceProductTotal)

InvoiceTax= InvoiceSubTotal * 0.05

InvoiceTotal = InvoiceSubtotal + InvoiceTax

1. （項目属性の説明の右側にある）InvoiceProductTotal 項目属性の**式**の列のボタンをクリックします。
2. 次の式を入力します：「*ProductPrice * InvoiceProductQuantity*」。式エディターを開くには、項目属性のプロパティ一覧にある[Formula]プロパティから開くこともできます。
3. このセクションのはじめにリストされている残りの式については、ステップ 1 および 2 を繰り返して定義してください。
4. 新しい式を保存するには[**保存**]をクリックしてください。

式の定義が完了したら、トランザクションは次の図のようになります：



図：式を含む Invoice（請求書）構造

ステップ 6：GeneXus が推論したデータモデルを確認する

GeneXus が推論したデータモデルを見て、それを修正することもできます。

!

知的なデータモデル生成

トランザクションを保存する度に GeneXus はエンドユーザーのエンティティに対応する最適なデータモデル（第 3 正規形において冗長なし）をトランザクションオブジェクトから推論してくれます。このデータモデルに基づいて、モデルの DBMS が設定されると、物理テーブルが生成されます。

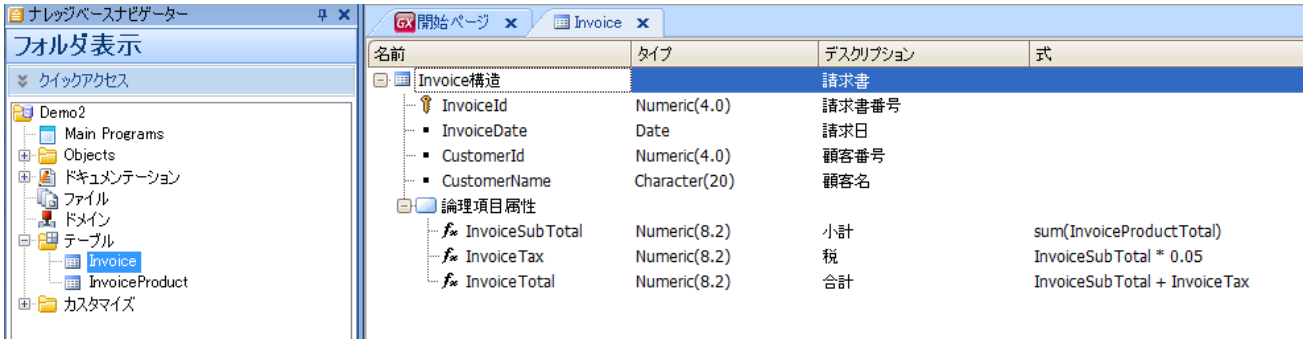
!

知的なデータモデル生成

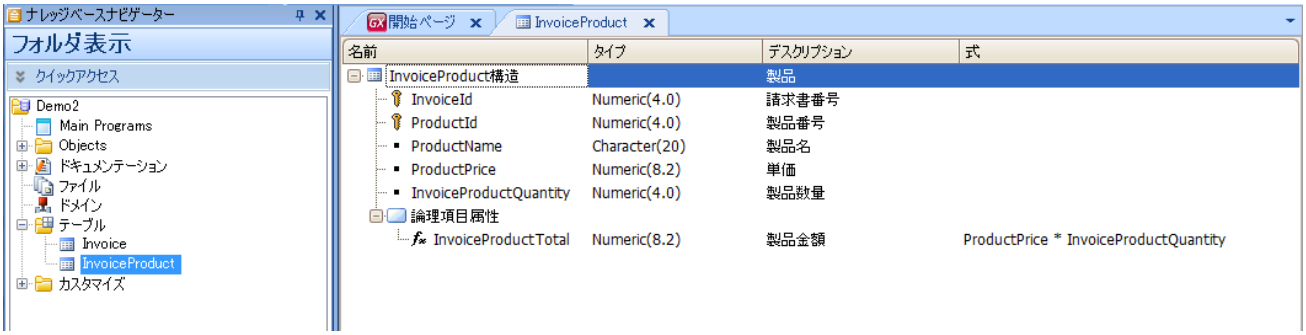
トランザクションオブジェクトの構造は、作成されるテーブルとインデックスを確定します：

- テーブルとインデックスの名前は、GeneXus により自動的にトランザクション名が割り当てられます。しかし、必要応じて任意の名前に変更することができます。
- GeneXus は第 3 正規形において冗長なしのデータモデルを推論します。ただし、GeneXus が自動的に管理する冗長項目属性を定義することができます。
- N レベルトランザクションに対応しているテーブルの主キーは、前のネストされた N-1 レベルの識別子と N レベルの識別子を連結して得られます。

1. 左側のメニューにある**テーブル**フォルダを選択します。
2. Invoice と InvoiceProduct テーブルをダブルクリックして開きます。GeneXus は、トランザクションから推論したデータ構造を表示します。



図：Invoice テーブル



図：InvoiceProduct テーブル

上記のデータベース一覧で、GeneXus が正規化したデータモデルを自動的に推論したことが分かります。推論した結果、Invoice トランザクションオブジェクトに対応する以下の 2 つのテーブルが作成されました：Invoice（請求書のヘッダ）と InvoiceProduct（請求書の明細行）：

Invoice	InvoiceProduct
InvoiceId	InvoiceId
InvoiceDate	ProductId
CustomerId	ProductName
CustomerName	ProductPrice
	InvoiceProductQuantity

注記：

- InvoceProduct テーブルの主キーは 2 つの項目属性により構成されました：InvoiceId と ProductId（レベル 1 の識別子（InvoiceId）とレベル 2 の識別子（ProductId））。
- GeneXus は式として定義された項目属性を自動的にテーブルから削除して、グローバル式に変換しました。グローバル式に変換されたことにより、その項目属性はナレッジベース内のどこからでもアクセスできます。
- Invoice テーブルには：
  - 同じ InvoiceId を持つ請求書は存在しません。
  - InvoiceId ごとに、InvoiceDate、CustomerId、および CustomerName は **1 つ**の値しか存在しません。
- InvoiceProduct テーブルには：
  - 同じ InvoiceId と ProductId を持つ請求書の明細行は存在しません。
  - InvoiceId と ProductId の各ペアにとって、ProductName、ProductPrice、および InvoiceProductQuantity は **1 つ**の値しか存在しません。

ステップ 7：トランザクションオブジェクトのフォームを確認する

作成したトランザクションオブジェクトに対して GeneXus が自動的に生成した既定の Web フォームを見てみましょう（まず、Invoice オブジェクトを開いてください）。

> **!** WEB フォーム
> トランザクションオブジェクトの構造を定義する際に GeneXus はエンドユーザーがアプリケーションでデータにアクセスする既定の Web 画面を作成してくれます。これらの Web フォームは、GeneXus アナリストによるカスタマイズが可能です。

Web フォームを見るには、次の手順に従います：

1. Invoice トランザクションの[**Web Form**]タブを選択します。

図：Invoice Web フォーム

"Error Viewer: ctlError"は、エラーメッセージが表示される既定のコントロールです。フォームのどこにでも配置してプロパティの設定をすることができます。可能性のあるメッセージは、Msg ルール、Error ルールと GeneXus の自動検証が出力するものです（例えば、参照整合性、データタイプエラーなど）。

これらのフォームでエンドユーザーは対応するテーブルに新しい請求書を新しいレコードとして登録することができます。さらに、ユーザーは既存の請求書を更新および削除することができます。

新規登録・更新・削除の機能はトランザクションのロジックに内在されているため GeneXus アナリストは何も記述する必要はありません。GeneXus は選択された言語のネイティブコードを自動的に生成してくれます。

GeneXus のトランザクションを定義する際は、次のことを意識してください：

- 明示的：データの表示および入力用のユーザーインターフェース定義。
- 暗黙的：アプリケーションのデータモデル設計(テーブル、インデックスなど）。

ステップ 8：アプリケーションの実行

!

第 3 正規形のデータベースの作成

- 推論したデータモデルに基づいて、GeneXus は 選ばれた DBMS でデータベースを作成する実行プログラムを生成します。
- データ構造を**更新**した際には、GeneXus は、データベースを再編成する実行プログラムを生成します。つまり、それはデータベース構造を変更し、データも新しい構造に移行します。
- これらの場合には、GeneXus が何をするかを示す**データベース作成レポート**または**影響分析レポート**が表示されます。

このステップでは、DBMS として SQL Server を使用し .NET でアプリケーションを実行します。

!

クラウドでのプロトタイピング（クラウドへのデプロイ）

- GeneXus では、いつでも任意のデバイスから自動的にアクセスできるように、クラウド環境のサーバーにアプリケーションをプロトタイプすることができます。
  詳細情報については次を参照してください：Easy prototyping:Deploy to cloud

アプリケーションサーバー（IIS）およびデータベース・サーバー（SQL サーバー）は、GeneXus 試用版に割り当てられたクラウド環境のプロトタイピングサーバーのアプリケーションサーバーに配置されます。アプリケーションの実行中に、ローカルで実行するために必要なプログラムが生成され、自動的にクラウドにアップロードされます。**製品版の GeneXus では、GeneXus でサポートされているすべての DBMS（データベース管理システム）を使用することができ、独自のアプリケーションサーバーおよびデータベースサーバーでアプリケーションをプロトタイプすることができます。**
GeneXus がサポートするジェネレーターおよび DBMS については次を参照してください：
http://www.genexus.com/products-japan/genexus?ja

!

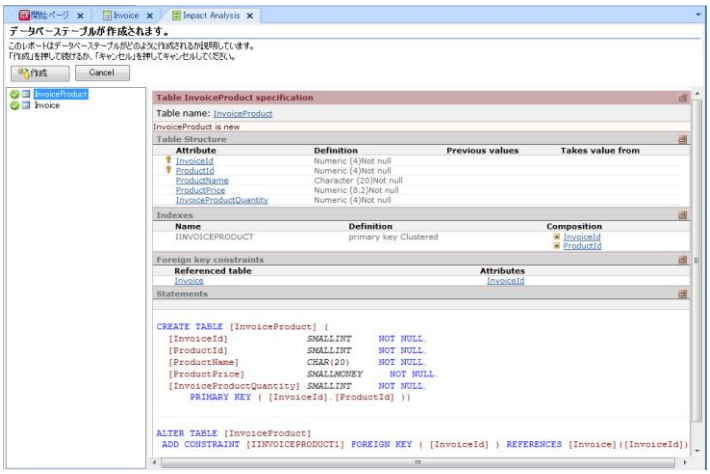
データベース作成レポート：これは、選択された DBMS に GeneXus が生成するデータベース構造を説明するレポートです。これは、推論したデータモデルを基に生成する推奨するデータベース構造のすべての情報を示します。各テーブルに関する情報は、5 つのセクションに分離されます：

- **ヘッダー：**テーブルの名前、テーブルに対するアクション、警告、およびエラーが含まれています。データモデルにエラーが含まれている場合は、再編成ボタンが無効になります。
- **テーブル構造：**テーブルの項目属性、それらの関連および項目属性に対してのアクションを示します。
- **インデックス：**GeneXus がデータベースの参照整合性を管理するためのインデックスと能率的にテーブルにアクセスするためのインデックスを示します。
- **外部キー制約：**テーブルの整合性制約を示します。
- **ステートメント：**実行される順番を示します。

1. F5 キーを押すか、メニューの[ビルド]→[開発者メニューを実行]オプションを選択します。
2. GeneXus 試用版のオーソライズに使用したのと同じユーザー名とパスワードを入力します。

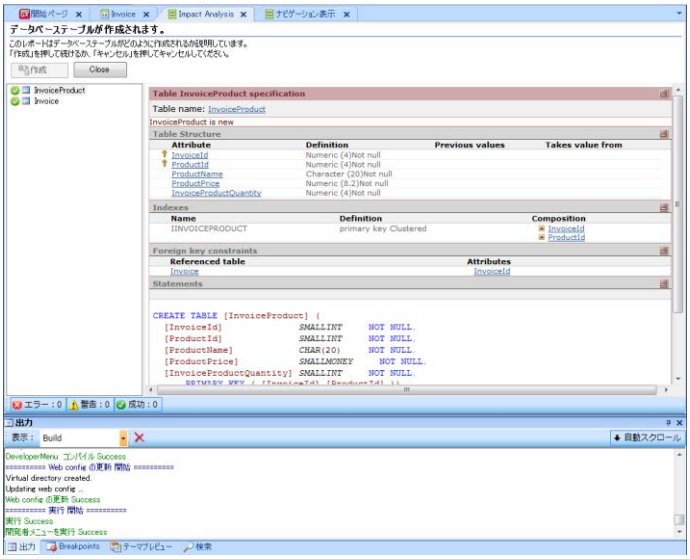
図：ユーザー名/パスワード

3. GeneXus はデータベース作成レポートを表示します。



図：データベース作成レポート

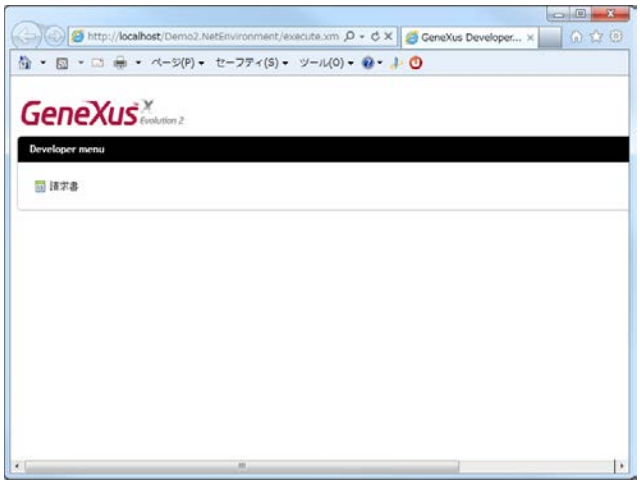
4. [**作成**]をクリックします。GeneXus はテーブルを作成するためのコードを選択された言語で生成し、データベースにアクセスします。



図：データベースの作成

ステップ 9：アプリケーションのテスト

1. 開発者メニューは、実行可能なオブジェクトすべてを表す XML ファイルです。これは、アプリケーションをプロトタイピングするための補助メニューです。[**請求書**]オプションをクリックします。



図：開発者メニュー

2. いくつかの請求書を登録します。AJAX 技術のおかげで、全ページを再ロードせずに式は自動的に計算されます。

3. 完了したらブラウザを**閉じて**ください。

図：Invoice トランザクションのテスト

ステップ 10：ビジネスルールの追加⇨（ルール）

いくつかの基本的なビジネスロジックをアプリケーションに追加しましょう。

GENEXUS ルール

GeneXus **ルール**は、各オブジェクトのビジネスロジックを定義するための場所です。ルールは宣言的に書かれており、GeneXus はどのルールをいつ適用すべきかを知的に決定します。

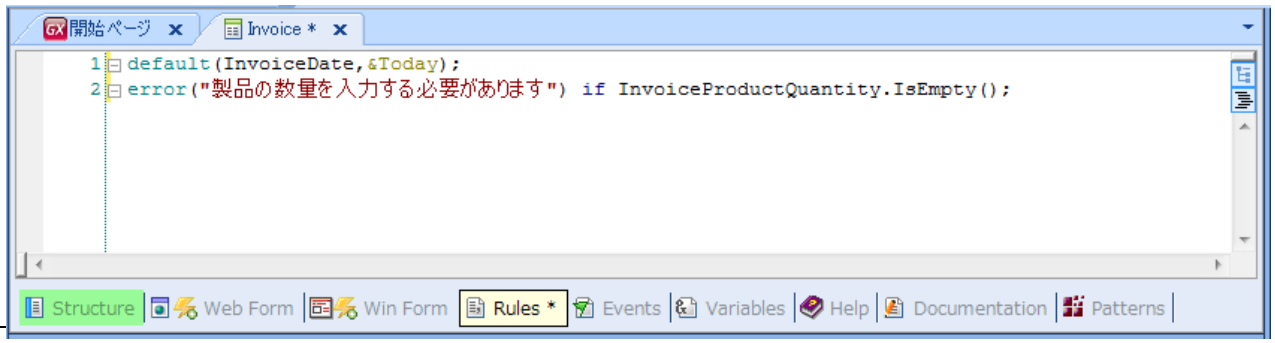
これらの**ルール**は、トランザクションオブジェクトで非常に重要な役割を担っています。それは、トランザクションの動作を定義することです（例えば：デフォルト値の設定、データ規制の定義、その他）。

- ルールには、トランザクション構造で定義されている項目属性、変数および関数を含める（書く）ことができます。
- ルールは、**宣言的な手法**で定義されています。ルールが書かれる順番が必ずしも実行される順番である必要ではないことを意味します。実行の順番は、GeneXus が自動的に決定します。

ルールの有効範囲は、定義されたトランザクション内のみです。なぜならルールは**ローカル**であるからです。

請求日の初期値を当日の日付に設定するルールを記述しましょう：

1. Invoice トランザクションの[**Rules**]タブを選択します。
2. 項目属性および変数の既定値を設定する **Default** ルールを使用します。
3. 次の通りにルールを記述します：Default(InvoiceDate, &today); これは、請求日の初期値に当日の日付を設定することを示しています。 [1]
4. それでは、入力された製品の購入数量が未入力の時に表示されるエラーメッセージのルールを見てみましょう。次のルールを追加します：Error("製品の数量を入力する必要があります") if InvoiceProductQuantity.IsEmpty();
5. [**保存**]ボタンをクリックします。



図：Invoice のルール

6. 入力したルールの動作を確認するために、アプリケーションを実行します（F5 キーを押すか、メニューの[ビルド]→[開発者メニューをビルド]オプションを選択します）。

図：Invoice のルールのテスト

ステップ 11：CUSTOMER トランザクションオブジェクトの作成

顧客は請求書とは別に独立したエンティティに対応する概念を持っています。したがって、顧客用のトランザクションを定義する必要があります。

> **セマンティックドメイン**
>
> - GeneXus は、トランザクション内の項目属性に動作を追加するためのセマンティックドメイン(電話、E メール、アドレスなど）を提供します。例：E メールの項目属性を作成すると、オブジェクト内でこの項目属性を使用するたびに E メールとして動作し、スマートデバイスまたは Web オブジェクト上でタップすることで E メールメッセージを送信することができます。

1. Customer トランザクションオブジェクトを作成します。

2. Customer トランザクションの構造に以下の項目属性を定義します：

項目属性	タイプ	デスクリプション
CustomerId	---------	----------
CustomerName	---------	----------
CustomerAddress	Address	顧客住所
CustomerEmail	Email	顧客 E メール

   CustomerId と CustomerName 項目属性の入力を始めると、GeneXus によって完全な名前、タイプ、およびデスクリプションが自動的に表示されます。これらの項目属性は既にナレッジベースに定義されているため、この動作が起こります。

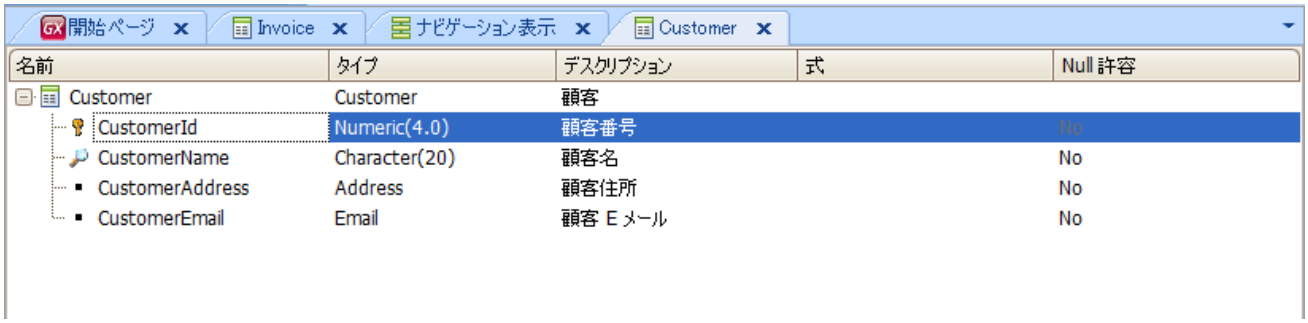
3. CustomerId 項目属性を右クリックして**プロパティ**を選択します。

4. CustomerId の Type Definition グループ内にある **Autonumber** プロパティを **True** に設定します。この定義によって、CustomerId が自動採番されます。

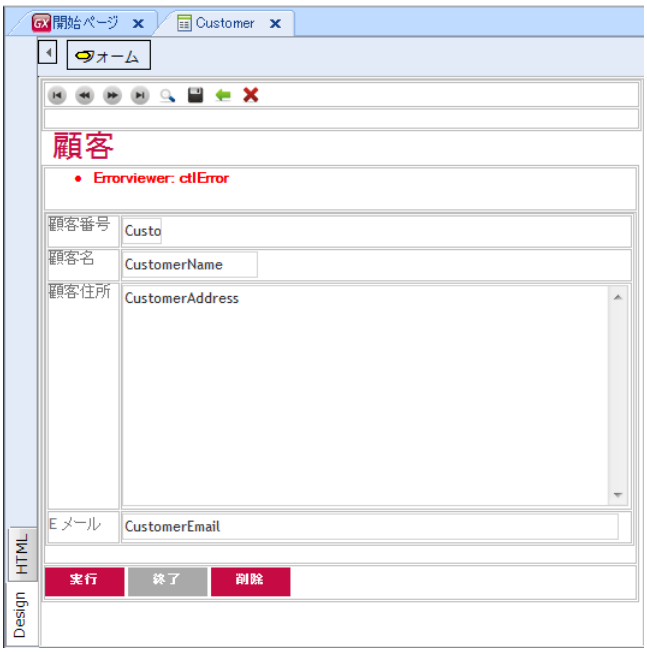
5. CustomerId のプロパティを開いて、サジェスト機能を有効化します。まずは InputType プロパティを Descriptions に設定します。そして、ItemDescriptions に次の項目属性を設定します：CustomerName

   この手法では、顧客を特定するために、顧客番号を入力する代わりに顧客名を入力することで、顧客番号が自動的に推論されます。Suggest プロパティは、入力された名前と一致する顧客名を補完する形で表示（サジェスト）します。これらのプロパティは、GeneXus が自動的に実装する **AJAX** の一部です。

以下の 2 つ目の図は Customer トランザクションの構造を示しています。



図：Customer トランザクションの構造



図：Customer トランザクションの Web Form

CustomerId 項目属性のプロパティ変更が Invoice の Web Form にまで反映されていることに注目してください。顧客名のところが CustomerId 項目属性になっています。すぐにお分かりいただけるとおり、これによって、アプリケーションのユーザビリティが大いに強化されます！

図：Invoice トランザクションの Web Form

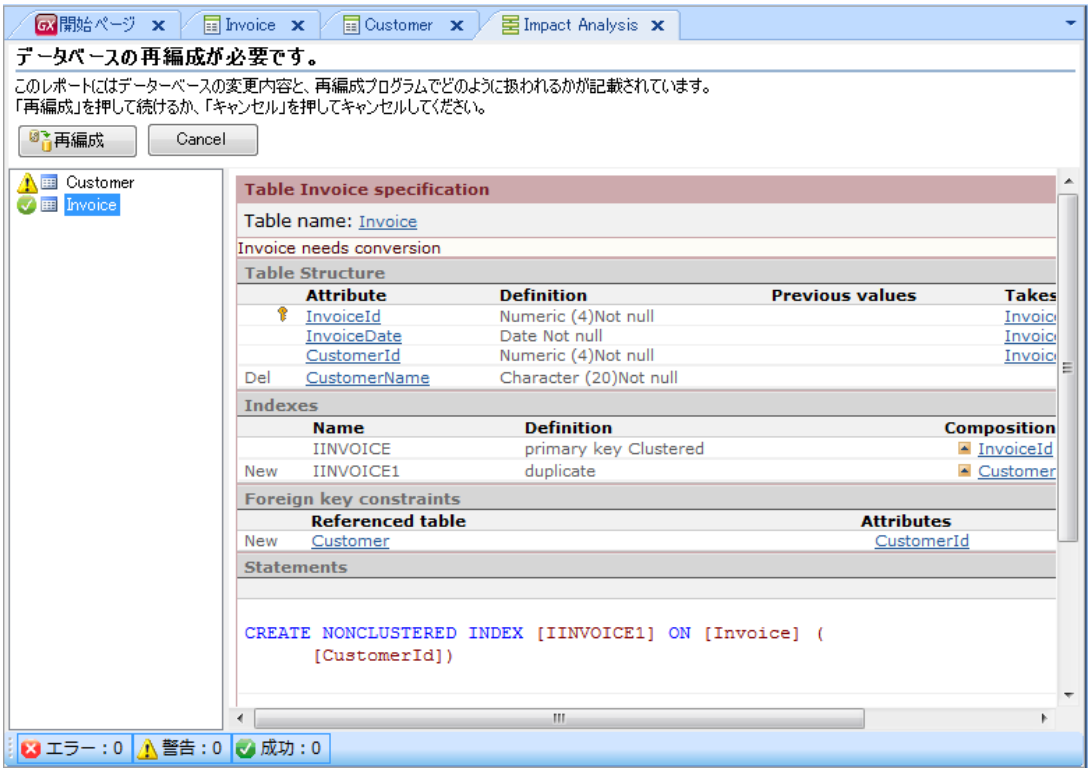
ステップ 12：データモデルに加えられた変更を確認する

F5 キーを押します。これにより、影響分析レポートが生成されます。

> **!** 影響分析
> GeneXus が推論したデータモデルの変更を物理データベースに反映するための更新・差分処理を示します。

新しいトランザクションの定義後に GeneXus がデータモデルを自動的に正規化したことが分かります。

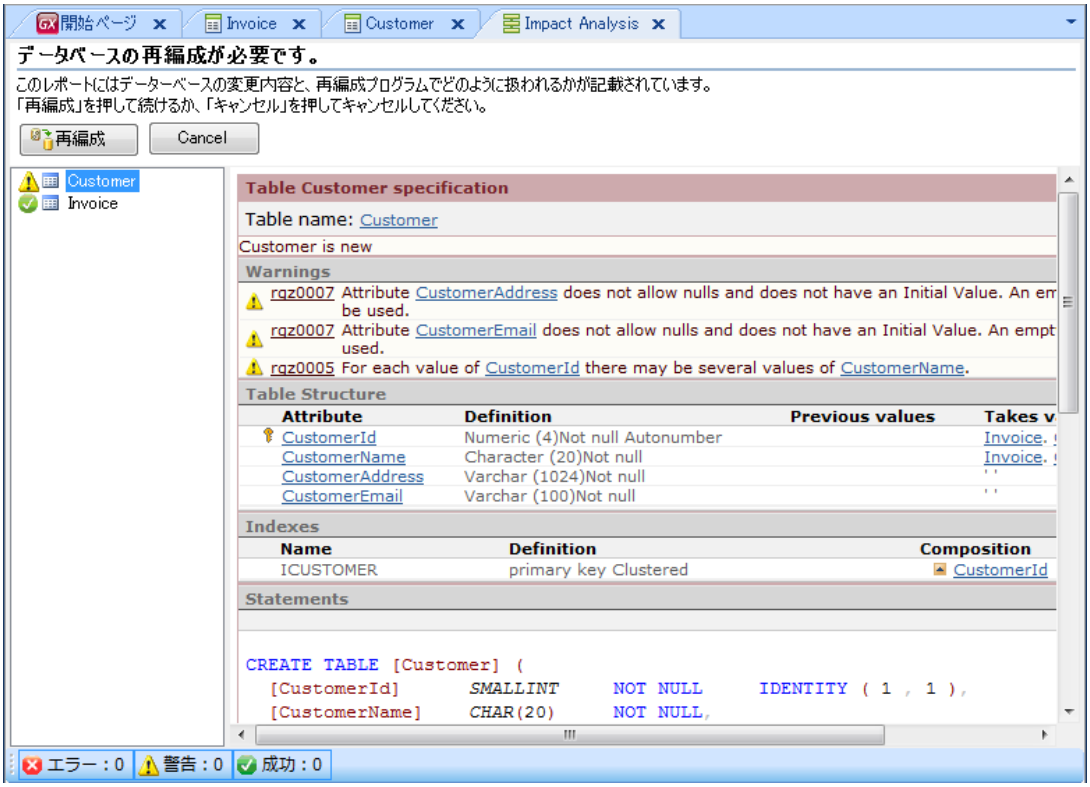
Invoice テーブルが影響を受け、CustomerName 項目属性が CustomerId を通して参照することができるため、Invoice テーブルから CustomerName 項目属性を削除します。



図：データベースの影響分析レポート

Customer の新しいテーブルが作成されます：

> **注記** GeneXus の Inpact Analysis（影響分析）で、Invoice テーブルに顧客の重複した値がある可能性から、正規化する際に影響を受けることを警告で示します。もう 1 つの警告には、テーブルの新しい項目属性は、NULL で作成されることを示します。



図：データベースの影響分析レポート

注記：

- GeneXus は新しいテーブルを追加しました：Customer テーブル（Customer トランザクションに関連します）。
- データベースが正規化されることで CustomerName 項目属性は Invoice テーブルから Customer テーブルに移動されます。
- Customer テーブルでは：
  - 複数の顧客が同じ CustomerId を持つことはできません。
  - 各顧客に対して **1 つ**の CustomerName、CustomerAddress、および CustomerEmail しか存在しません。
- Invoice テーブルでは：
  - 同じ InvoiceId を持つ請求書は存在しません。
  - Invoice テーブルの CustomerId 項目属性は、Customer テーブルから取得された外部キーです。このように、Customer と Invoice 間は 1 対 N で関連しています：
    - 各 InvoiceId には CustomerId が **1 つ**しかありません。
    - 各 CustomerId は**複数の**請求書（InvoiceId）を持つことができます。

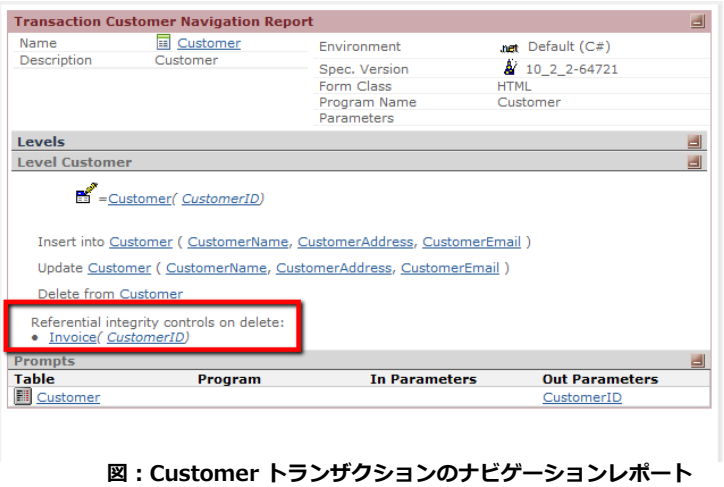
ステップ 13：分析レポートを確認する

データベースの再編成を行う準備ができています。「再編成」オプションを選択します。再編成プログラムは、影響分析で捉えた変更を物理データベースに反映してデータも移行してくれます。

再編成を実行する際に、GeneXus は各プログラムの解析表を含む**分析レポート**を表示します。

> **!** 分析レポート
> ここには、プログラムがどのように実行されるか、どのテーブルに（どのように）アクセスするか、および実行する操作についての情報が記述されます。

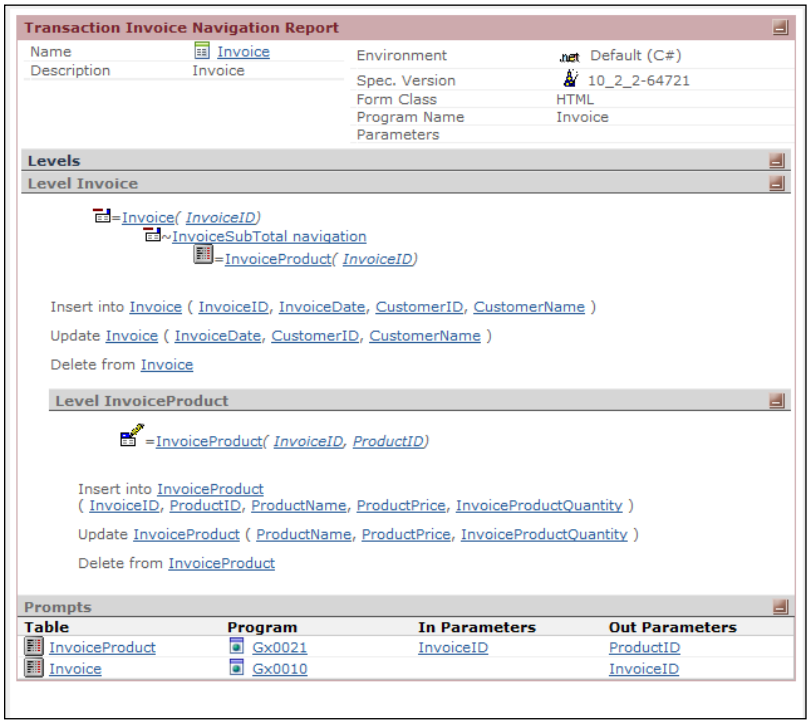
「削除する際の参照整合性の制御」とは、Customer トランザクションから顧客を削除する際にプログラムがその顧客に請求書がないことを確認することを意味します。この検索を効率よく実行するために Invoice テーブルの外部キーである CustomerId 項目属性のインデックスが使用されます。



図：Customer トランザクションのナビゲーションレポート

Invoice トランザクションの**参照整合性**：Invoice レベルの CustomerId 外部キーは、Invoice トランザクションを使って請求書を追加あるいは更新する際に、GeneXus が CustomerId 外部キーに入力される値が Customer テーブルに主キーとして存在することを自動的に確認することを意味します。この検索を効率よく実行するために、Customer テーブルの主キーである CustomerId 項目属性のインデックスが使用されます。

参照整合性のチェックは入力される項目属性が（例えば Invoice の CustomerId）有効であることを確認しますが、有効な値が何であるかという情報は提供しません。有効な値の検索を容易にするために、GeneXus は**検索一覧**オブジェクト（プロンプト）を作成して、すべての有効な値をその中から選ぶことができます。



図：INVOICE トランザクションのナビゲーションレポート

パート 3：開発 パターンからアプリケーションを生成する

開発の一部にパターンを使用する

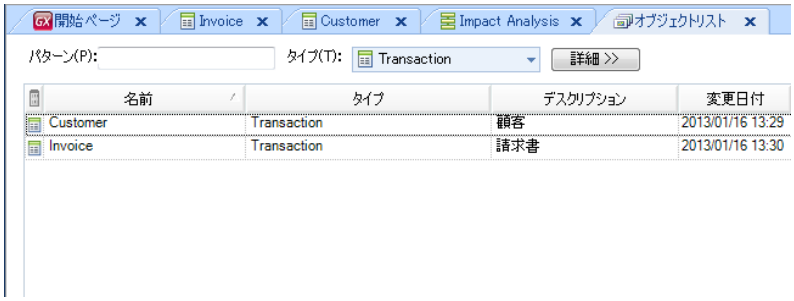
アプリケーションのテストをすると、トランザクションを通してデータの入力を行っていることに気づきます。しかし、顧客や請求書あるいは製品を入力、更新、削除するだけではなく、特定の基準に従ってそれらのデータをフィルタリングして並び替え、一覧表示ができるとさらに便利になります。

> **!** パターン
> パターンは、異なるエレメントに適用される類似のアクションです。パターンにより、特定の機能を満たすために必要なすべてのオブジェクトを自動的に生成することができます。これにより手動による方法を避けることができます。

ステップ 14：WEB アプリケーションのパターン

作成したトランザクションに　Work With パターンを適用します。

1. [**表示**]メニューから[**オブジェクトリスト**]オプションを選択します。



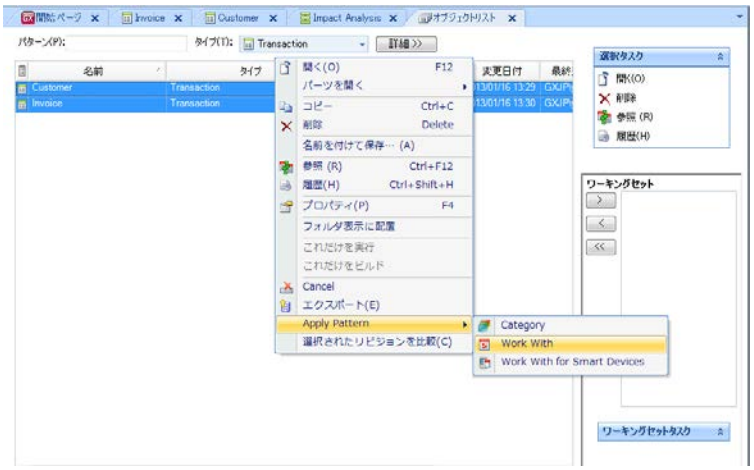
図：オブジェクトリスト

2. Work With Objects パネルの[タイプ]コンボボックスから[Transaction]を選択します。
3. Customer および Invoice トランザクションを選択します。

4. 選択したトランザクションを右クリックして、[Apply Pattern(パターンを適用)]→[Work With]と選択します。

5. 変更を保存します。

6. F5 キーを押して、アプリケーションを実行します。トランザクションを直接実行できないことが分かります。Customer および Invoice の "Work With" を実行することができます。

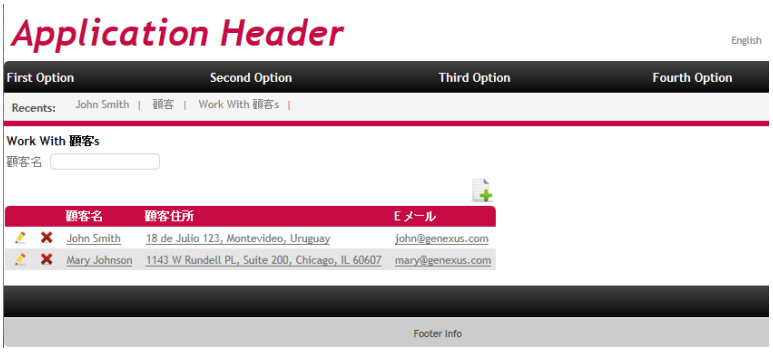


図：パターンの適用

7. Invoice の "Work With" を実行します。

パターンには次の機能が追加されます：

- 日付でデータが絞られます。アプリケーションのリフレッシュは不要です。
- 列の見出しをクリックしてデータを並べ換えることができます。
- 一覧表は 10 行単位でページングされています。
- リンクをクリックすることで、関連しているデータの照会画面が表示されます。
- アクセスしたリンクの履歴が保存されます。



図：アプリケーションの実行

!

セマンティックドメイン

- Customer WorkWith を開き、前の手順でセマンティックドメインとして設定した項目属性がアプリケーションでどのように動作するかを確認します。

ステップ 15：スマートデバイスアプリケーション用のパターン

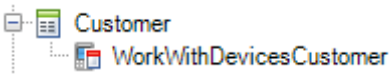
Customer トランザクションに"Work With for Smart Devices"パターンを適用します。

Work With for Smart Devices パターンを Customer トランザクションに適用します。次の操作を行います。

- トランザクションの [Patterns] セクションに移動します。
- [Work With for Smart Devices] タブをクリックし、[保存時にこのパターンを適用] を選択します。

- インスタンスの [List]、[Detail]、[Section (General)] の各ノードで、それぞれ表示される画面を確認します。

- 保存し、次のエレメントを確認します。
  - トランザクションのプロパティを次のように設定します。
    - [**Business Component**] = [True]
    - [**Expose as Web Service**] = [True]
    - [**Web Services Protocol**] = [ReST Protocol]
  - [フォルダ表示] → [Customer] の順に選択すると、サブノードが表示されます。

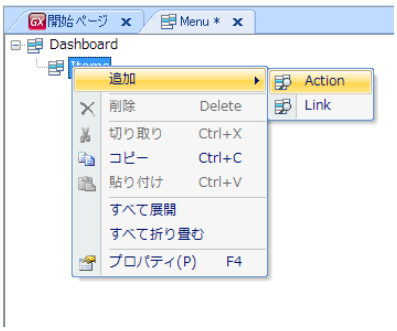


  - [設定] → [.Net Environment] → [ジェネレーター] の順に選択すると、SmartDevices ジェネレーターが 2 つ目のジェネレーターとして表示されます (プロパティ、特に [Generate Android] が [True] になっていることを確認します)。

- アプリケーションをテストするためにほかに何が必要でしょうか？アプリケーションのスマート デバイス セクションにはメインオブジェクトがないので、ここで F5 キーを押しても何も実行されません。したがって、F5 キーを押す前に次を実行します：「Menu」という名前の**ダッシュボード**([ファイル] メニュー→[新規] → [オブジェクト] → [Dashboard])を作成し、**Work With Customer** を呼び出すための項目を追加します。

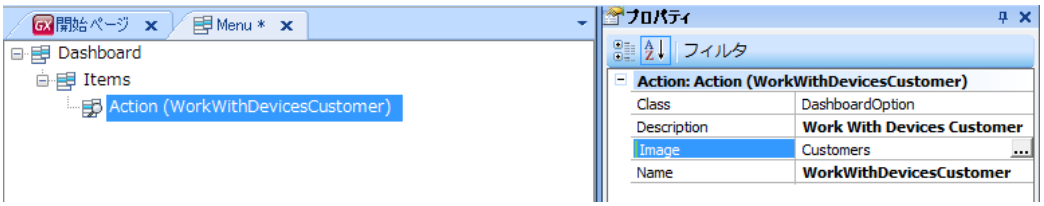
図：ダッシュボードの作成

- 作成したダッシュボードの [Item]ノードを右クリックして、[追加] → [Action]と選択します。



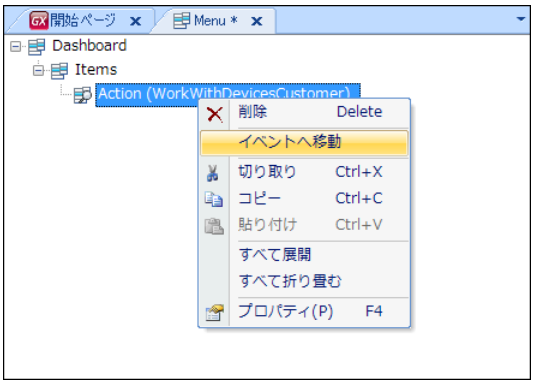
図：アクションの追加

- アクションに画像を関連付けます(この例では「Customers」という名前の画像を設定しています)。

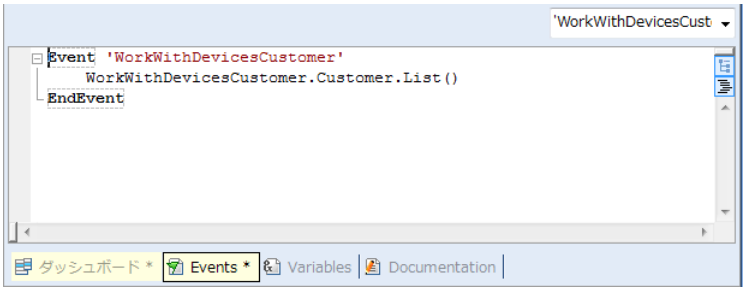


図：ダッシュボードへのアクションの追加

- （Action ノードを右クリックして）このアクションに関連付けられているイベントを確認します。



図：[イベントへ移動]操作

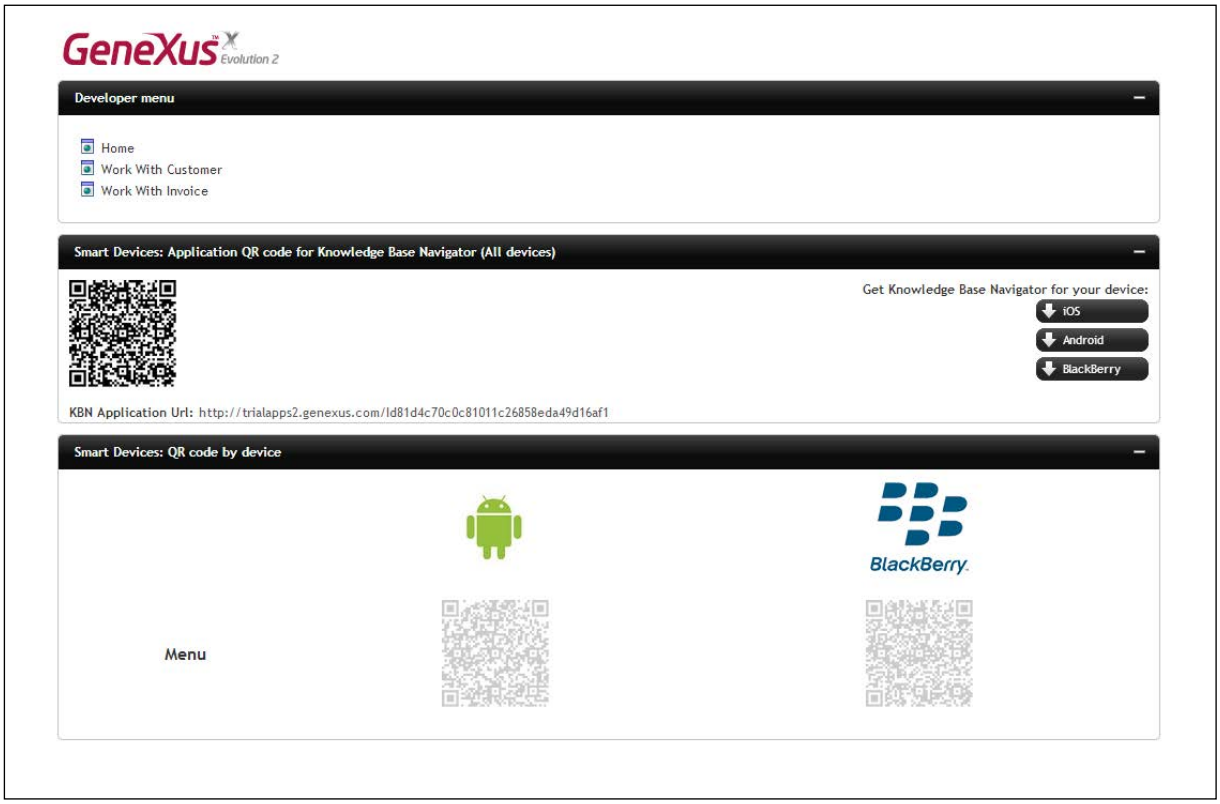


図：ダッシュボードのイベント

- 保存し、[**Main Program**] プロパティの値が [True] になっていることを確認します。次に **F5** キーを押し、アプリケーションを生成してエミュレーターで実行します。

  注記:

  - 開始オブジェクトを指定していないので、GeneXus は、定義済みブラウザーで Web の [Developer Menu] を開き、また Android 用エミュレーターも開きます。このため、Ruby の Web アプリケーションと、Android のスマートデバイス向けアプリケーションの両方を試すことができます。
  - [Developer Menu] には、Web アプリケーションのオブジェクトを実行するためのリンクに加えて、QR コードが表示されます。QR コードには、インストール先デバイスに応じた KBN（ナレッジベースナビゲーター）をダウンロードするための URL がカプセル化されています。また、Android と BlackBerry それぞれのスマートデバイス向けコンパイル済みアプリケーションが含まれます。コンパイル済みファイルは、作成されない場合もあります。これは、開始オブジェクトが設定されているかどうか、また使用する実行モード（F5 キー、[これだけを実行] など）によって異なります。



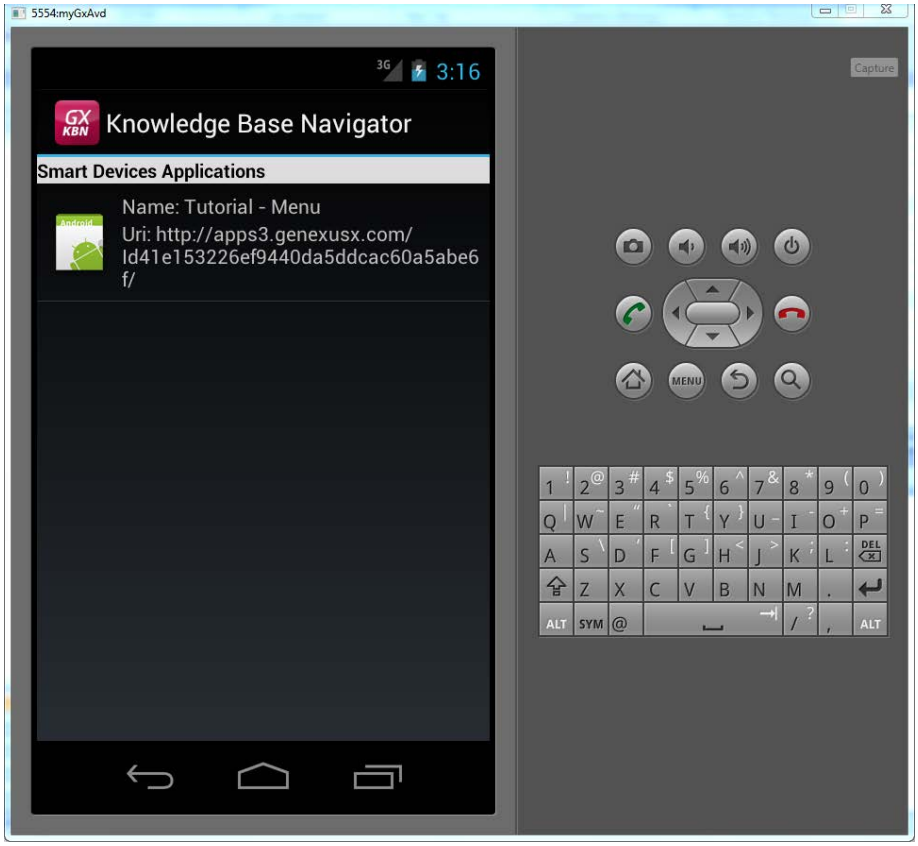
図：QR コード付きの開発者メニュー

注記:

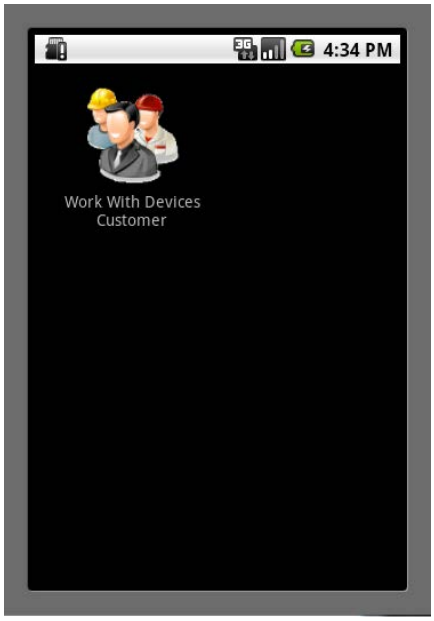
- 「開始オブジェクト」を指定していないので、GeneXus は、定義済みブラウザーで Web の [Developer Menu（開発者メニュー）]を開き、また Android 用エミュレーターも開きます。このため、.Net の Web アプリケーションと、Android のスマートデバイス向けアプリケーションの両方を試すことができます。
- この例の場合、Android エミュレーターで KBN(Knowledge Base Navigator) が開き、アプリケーションのダッシュボード(現時点で唯一のメインオブジェクト) を実行するための URL が表示されます。

- Web の開発者メニューには、Web アプリケーションのオブジェクトを実行するためのリンクに加えて、QR コードが表示されます。QR コードには、インストール先デバイスに応じた KBN（ナレッジベースナビゲーター）をダウンロードするための URL がカプセル化されています。また、Android と BlackBerry それぞれのスマートデバイス向けコンパイル済みアプリケーションが含まれます。また、すべてのスマートデバイスオブジェクトを「開始オブジェクト」として設定した場合、コンパイルされたファイルが作成され、関連する QR コードが Web の開発者メニュー上に表示されます。
- 詳細については Executing From QR Codes を参照してください（英語での情報提供となります）。

- URL をタップすると、ダッシュボードに対応するレイアウトが表示されます。

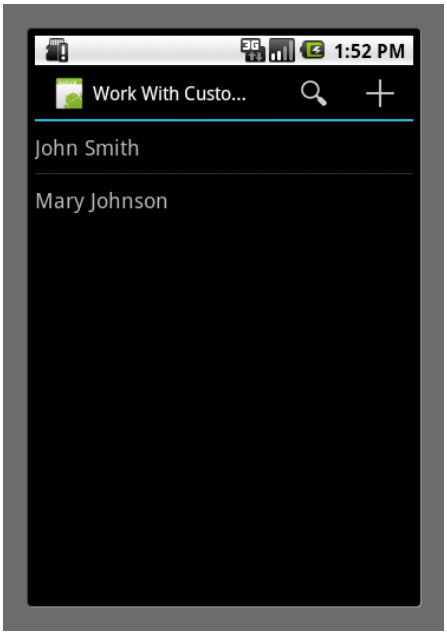


図：Android エミュレーター



図：ダッシュボード

- “**Work With Devices Customer**” をタップします。



図：Work With Device Customers

  - 顧客ごとに、表示されている項目属性を確認します。GeneXus に戻り、この**リスト (List)** のレイアウト (Layout) をパターンから探し、表示されている項目属性と一致していることを確認します。

- 詳細表示を確認するためにリストされている顧客のいずれかをタップします。

図：顧客の詳細

- [Address] フィールドに表示されるアイコンをタップするとどうなるか確認します (セマンティックドメインを思い出してください)。

> 注記
> エミュレータで E メールアカウントを設定するには、[Menu]を押して [System Setting]->[Account & Sync]に設定します。

- 任意の顧客を編集または削除するには、画面の上部にある関連するオプションを押します。

図：顧客の編集

- 新しい顧客を追加または検索するには、戻るボタンを押してから関連するオプションを押します。

図：検索

お疲れ様でした！

これで GeneXus による初めてのアプリケーションが作成されました。

要約

このチュートリアルで GeneXus の主な機能とメリットを体験して頂けたと思います。

知識を基にするアプリケーション設計

ナレッジベースを作成することから始まり、アプリケーションの要件をナレッジベースのトランザクションオブジェクトと呼ばれる一連のビジネス部分として記述しました。GeneXus は、記述された情報を基にすべてのトランザクションに対応する最適のデータモデル（第 3 正規形）を推論しました。追加したビジネスルールを基に、GeneXus はアプリケーションのビジネスロジックのコードを生成しました。

データベースの自動生成

特定の実行プラットホーム（Web、プログラミング言語、DBMS など）を設定して、GeneXus は自動的に指定された DBMS サーバーに物理的なデータモデルを作成しました。

コードの自動生成と最適なプロトタイプ

アプリケーションプログラムのソースコードを生成し、プロトタイプ環境でアプリケーションをテストすることができました。

アプリケーションの保守

GeneXus オブジェクトの修正・追加によるデータモデルの変更が簡単に物理データベースに反映され、影響を受けたプログラムが再度自動生成されることで、アプリケーションが完成することを実感できたと思います。

マルチプラットフォーム開発

最後には GeneXus アプリケーションの環境を簡単に移動できることを説明しました。

連絡先とリソース

GeneXus コミュニティ

GeneXus コミュニティでは、質問に対して回答したり、問題に対する解決策を提案したり、経験を共有する場所を提供したりしています。以下のサイトでは、提供されているコミュニティリソースの一覧を参照することができます：http://www.genexus.com/community-and-support-jp/community?ja

サポート

Artech 社は、以下の 2 種類のサポートリソースとサービスを提供しています：

- **オンライン　セルフサービ　スサポート**
  これらのリソースはどなたでもオンラインで利用することができます。ただし、アクセス可能なデータは GXtechnical のユーザー権限によって異なります（2 種類の権限があります：一般ユーザーまたは顧客）。
- **インタラクティブ・サポート サービス**
  コミュニティメンバーまたはサポートチームとの情報交換が可能です。

次を参照してください：http://www.genexus.com/community-and-support-jp/support?ja

購入方法

GeneXus の技術は、世界的な販売店ネットワークを通して販売されています。

最寄りの販売店については、次を参照してください：http://www.genexus.com/company-jp/sub-distributors-in-japan?ja

または info@genexus.jp にお問い合わせください。

参考

Web サイト：http://www.genexus.jp
教育：http://www.genexus.com/community-and-support-jp/training-313?ja
コミュニティ：http://www.genexus.com/community-and-support-jp/community?ja