





2011年7月

ジェネクサス・ジャパン株式会社 〒141-0022 東京都品川区東五反田5丁目10-18、TK五反田ビル3F 電話番号:03-5793-5481 FAX:03-5793-5484 http://www.genexus.jp



#### Copyright · Artech Consultores S. R. L. 1988-2008.

All rights reserved. This document may not be reproduced by any means without the express consent of Artech Consultores S.R.L. The information contained herein is intended for personal use only.

#### **Registered Trademarks**

Artech and GeneXus are trademarks or registered trademarks of Artech Consultores S.R.L. All other trademarks mentioned herein are the property of their respective owners.



# 目次

序論	3
GeneXus トライアルバージョン	3
インストールとオーソライズ	4
はじめに:ステップバイステップ チュートリアル	8
パート1:GeneXusの紹介	9
パート2:最初のアプリケーション	.10
パート3:保守	.35
パート4:機能の追加	.38
パート5:インタラクティブなダイアログ	.40
パート6:アプリケーションの改良	.45
パート7:プロシージャ	.48
パート8:マルチプラットフォーム	.55
要約	.56
FAQ	.57
コンタクトとリソース	.60



#### 序論

この文書は、GeneXus が提唱するナレッジベース開発方法論の潜在能力を、その主要な機能を経験しながらあなたが発見するのを援助することが目的です。

- 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 トライアルバージョン

#### 機能制限

GeneXusトライアルバージョンは全機能を網羅しています。利用できるジェネレータ(.Net+SQL Server)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)

しかしながら、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は制限が掛かっています。

- 項目属性の最大数は90個
- オブジェクトの最大数は 140 個

#### ライセンス制限

GeneXus トライアルバージョンは、評価とトレーニングの目的のみに使用することが許可されています。ライセンスは、 シングルユーザのローカルインストールのみとなります。

#### ライセンス 同意

GeneXus X トライアル版ライセンス(知的財産)の利用規約

1. この合意は、GeneXus X トライアル版と知識ベースに基づき自動的に設計、生成、保守されるデータベースやアプリケー ションを対象とします。

2. Artech Consultores S.R.L. (これ以降は Artech) は以下のことを宣言しユーザは同意します。GeneXusの知的な生成物、GeneXus, GXflow, GXplorer, GXportal, GXquery そして Artechの名前やロゴ、その他GeneXusに関係して発表するものは、それらが登録されているいないに係わらず Artechの資産です。この合意は、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、ユーザに本件のライセンスを譲渡する権利も与えません。

3.ユーザは、GeneXus X トライアル版を使うことで得るGeneXusがベースとなったアイデアや技術は公開せず、自分の利益のためだけに使うことに同意します。特に、ユーザはそのコードを解釈するためにリバースエンジニアリング技術を使用しない、また他の人にもそうさせないことに同意します。

4.ユーザはライセンスを複写したり、GeneXusの認可されない使用に対するプロテクションの仕組みを無効にしたりすることを決して行ってはなりません。Artechは、これらのメカニズムを修正したりいつでも新しいものを加える権利を保有します。
5. Artechは以下の状況の下で全ての条件に従うことを前提に、ツール(GeneXus Xトライアル版)の使用の許可をユーザに与えます:

a) 各々のライセンスは、1 台のパソコンで使用します。ユーザはどのライセンスも複数のパソコンで同時に利用しないこと に同意します。



b) このライセンスは排他的ではありませんが、移転することはできません。GeneXus X トライアル版で生成したプログラムを、営利目的のために使うことはできません。

c)ユーザは個人的な利用のため、または評価目的のためだけに、GeneXusプログラムとドキュメンテーションを使用します。 プログラムとドキュメンテーションの何れもコピーを第三者に提供しないことに同意します。

d) Artechは暗黙的にも明示的にも決して、GeneXus X トライアル版の使用に関して直接的、あるいは間接的な付帯的な損害賠償に対しても責任を負いません。

e) ユーザは、教材で示される指示と仕様に従ってライセンスを使用することに同意します。

f) GeneXus X トライアル版(全体または一部の)のどんな商用利用も禁じられています。この種のアプリケーションを開発するためには、ユーザは製品版を購入しなければなりません。

g) GeneXus X トライアル版は、60 日間の期間限定で.NETジェネレータとSQLデータベースのサポートだけが提供されま す。また最大で、90 の項目属性と 140 のオブジェクトまで作成することができます。

6.このライセンスが必要なソフト製品をインストールすることは、あなたがすべての契約条件を受け入れたことを意味します。

#### テクニカルサポート

GeneXusに関するテクニカルサポートについては、以下のメニューが用意されています。 必要に応じてサービスを選択し、ご利用ください。

- 1. GeneXusに関する総合的な技術情報については、以下のWikiをご利用ください。 http://wiki.genexus.jp/hwiki.aspx?Wiki+Home,
- Wikiを利用し、それでも問題が解決しない場合は、以下のサポートフォーラムをご利用ください。 http://www.genexus.jp/gxpsites/hgxpp001?6,7,94,O,J,0,MNU;E;19;5;MNU
- 3. トレーニングサービス及び資料に関しての情報は、以下のサイトをご確認ください。 http://www.genexus.com/community-and-support-jp/training-313?ja http://www.genexus.com/40/-317?ja

さらなる情報が必要な場合は、お近くの代理店にお問い合わせください。

http://www.genexus.com/company-jp/sub-distributors-in-japan?ja

# インストールとオーソライズ

#### インストール要件

GeneXusトライアルバージョン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。それは、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及び保守することができます。

#### • GeneXus ジェネレータ

GeneXusは、市場の主要なプラットホ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一覧に関しては、



http://www.genexus.com/technologies をご覧ください。

GeneXusトライアルバージョンに提供されるジェネレータは.NETのみになります。

GeneXus及び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とソフトウェア要件は以下のとおりです。

ハードウェア要件	プロセッサー:最低1GHz (マルチコア対応プロセッサをお勧めします)
	メモリー :最低1 GB (2 GB 以上の容量をお勧めします)
	ハードディスク : インストールするために最小限 300 MBの空き容量が必要です。
	GeneXusアプリケーションを作成するためには、知識ベース及び生成するコードのためにさらなる空
	き容量あるいは共有ディスクが必要です。
ソフトウェア要件	Microsoft Windows XP Professional SP2 以降の SP 適用済み
	Microsoft Windows Vista 各工ディション(※1)
	Microsoft Windows Windows7 各工ディション(※1)
	Microsoft .NET Framework 3.5 SP1 もしくは以降のバージョン
	Microsoft Internet Information Service 5.1 もしくは以降のバージョン(※2)
	Microsoft SQL Server 2005 あるいは 2008(Express, Standard あるいは他のエディション) (※3)
	Adobe Acrobat Reader10 もしくは以降のバージョン

(※1)OS 要件につき、IIS 機能がインストールできるエディションに限ります。

(※2)IIS7.0以降のバージョンについては、「IIS6と互換性のある管理」機能を有効化して下さい。

(※3)Microsoft SQL Server Express Edition は無料であり、 再配布可能なバージョンです。

アプリケーションデータベースを作成して、生成されたアプリケーションを実行するために、下記の DBMS を必要とします:

• Microsoft SQL Server 2005 あるいは 2008(Express, Standard あるいは他のエディション)。

GeneXus トライアルバージョンをインストールする端末に Microsoft SQL Server がインストールされていない場合は、 GeneXus トライアルバージョンのインストーラが自動的に Microsoft SQL Server 2005 をインストールし、Microsoft SQL Server のシステムアドミニストレータユーザー「sa」のパスワードは「genexustrial」で設定されます。



1. GeneXus トライアルバージョンのセットアップファイル(genexusxtrial.exe)を実行してください。Windows スタ ートメニューから、あるいはトライアルバージョンの CD で対応するオプションを選択してください。

GeneXus X Trial Version - InstallAware Wizard	Build 14555 Setup	
Welcome to the InstallAware Wizard for GeneXus X Trial Version. This wizard guides you through installing this program.	GeneXus X	
Click Next to continue.	7	
InstallAware	< Previous Next >	Cancel

図:GeneXus X トライアルバージョンのインストールウィザード

1. 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を行ってください。

2. GeneXus トライアルバージョンを初めて使用する際には、実行するためにこの製品をオーソライズしなければなりません。オーソライズは、オンラインもしくは電子メールで行うことが可能です。インターネットアクセスがあるならば、オンラインでのオーソライズをお勧めします。

Authorization Required
GeneXus X Evolution 1 Trial をインストールして頂いて誠にありがとうございます。オーソライズキーの請 求は オンライン または メール・ボタンから行ってください。
インターネット接続がある場合はオンライオーソライズを推奨します。オンラインオーソライズができない場合ま たはインターネット接続が無い場合はメールでオーソライズを行ってください。
Click Cancel to close GeneXus without authorizing.
Back         オンライン(0)         メール(B)         Cancel

図:GeneXus トライアルバージョンを登録するためのダイアログボックス

- 3. 必須項目を入力してください:会社名と電話番号は任意ですが、名前、電子メールと国名は必須です。
- 一度オンラインでのオーサライズを選び全ての必須のデータを完備したならば、GeneXus トライアルバージョンは即座 に 60 日の期間中に利用することができます。
- もし電子メールオプションを選ぶならば、名前と国名を入力しなければなりません。
   会社名と電話番号の項目に関しては、任意です。
   入力したデータを確認し登録したデータが正しければ、"CONTINUE"を押してください。入力したデータを変更するためには、"BACK"を押してください。
   このオーソライズ方法は Outlook がデフォルトの電子メールクライアントであることを想定しています。メールを送付していただくと、ライセンスマネージャーが生成するサイトコードを記載した電子メールを生成し、あなたの電子メールアドレスにサイトキーが送られます。



Outlook がデフォルト電子メールクライアントでない場合は、**手動**オーソライズで、どんな電子メールアドレスからでもライ センスマネージャーが生成したデータを送ることができます。

サイトキーは、送付していただいた際と同じ電子メールアドレスに送られます。Authorization Required 画面のサイトキー 項目に**サイトキー**をコピー&ペーストしてから**続行**をクリックします。

これで GeneXus トライアルバージョンを使い始めることができます!

GeneXusトライアルバージョンを起動させるために支援が必要な場合、サポートフォーラムをご利用ください。

http://www.genexus.jp/gxpsites/hgxpp001?6,7,94,O,J,0,MNU;E;19;5;MNU



# 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、GeneXus の初体験をしていただきながら GeneXus 方法論の基礎を学んでいただくことです。 やがてあなたが考えているよりも早く、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演習において、単純なビジネスアプリケーションを作成し、GeneXus トライアルバージョンが サポートするプラットホーム(.Net+SQL Server)上でそれを実装します。

GeneXus がサポートしているその他のプラットホーム一覧に関しては、http://www.genexus.com/technologies/ をご覧 ください。

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

- ナレッジベースアプリケーション設計
- 知的なデータベース生成
- 自動的なコード生成
- 十分に機能的なアプリケーション
- インクリメンタル開発及びアプリケーションの自動的な保守
- ノンインタラクティブなプロセス設計(プロシージャー)
- マルチプラットフォームに向けた開発

使用されるアイコン

Ĩ	このアイコンは重要な概念であることを意味します。
Ň	このアイコンはヒントを意味します。このフレームの内にあるテキストを読むことをお勧めします。
Ö	このアイコンは実習に必要な目安となる時間を意味します。
Ü	このアイコンは、注釈を意味します。



# パート1:GeneXusの紹介



推定された時間:5分

#### GeneXus とは何か?

簡潔にまとめると、GeneXus はプログラムを作成するプログラムです。

これはユーザ要件から始まるツールであり、我々が**ナレッジベース(知識ベース)**と呼ぶ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及びプログラムを設計、作成そして保守を行います。





### パート2:最初のアプリケーション

推定された時間:1時間40分

#### 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、ユーザの要件に基づいて GeneXus ナレッジベースを作成及びアプリケーションを設計することを学び ます。今回のケースにおいては、請求書を最初に設計して、その後セールス管理システムで要求されている他の構成要素を設 計します。



#### ビジネスアナリスト 対 開発者

エンドユーザの要求を理解することは、自動化することができないソフトウェア開発における作業の一つです。 そのため、我々は GeneXus 開発者をプログラマー、コーダーあるいは単なる開発者と呼ぶのではなく、ビジネス アナリストと呼んでいます。

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の実体を記述することに基づきます。(有形と無形両方の現実オブジェクト)

エンティティにエンドユーザ要件を高い抽象概念レベルで記述することを行います。

この理由により、我々は**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。

この現実が変わる際に、ビジネスアナリストはその変更を記述します。そして、GeneXus は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するとしましょう。

現在彼らはそのアプリケーションを保有しておらず、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、請求書は手作 業で作成しています。

このチュートリアルでは GeneXus 方法論の基礎を学習しながら、請求書(Invoice)、顧客(Customers)と製品(Products)を処理していきます。

このチュートリアルで GeneXus に現状の実務を記述することで. NET ジェネレータと SQLServer データベースを使用する ウェブ請求書システムが生成されることを学びます。

#### ステップ1:開発環境

GeneXus を起動すると、下記の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感的で使いやすく、そして独自設定が できる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割されています:



#### 図:開発環境(IDE)

1	メインウインドウ: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を必要とします。 もしパソコンがインターネットに接続されていない場合は GeneXus コミュニティに発表されるニュースを表示す ることができません。
2	ツールウインドウ
3	ツールバー

#### 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、作業環境を定義することです。

それは、複数のプラットホ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、 <b>現実の記述</b> を保管する場所といえます。	



	環境
	アプリケーションを特定のソフトウェアプラットホームで生成及び保守を行うためには実行プラットホームに関連する情報を全て <b>環境</b> に定義します(例:プラットホームのジェネレータ、データベース接続情報、DBMS、ユー ザインタフェース、その他のプラットホームのプロパティ)。
•	このためには、環境ごとに DBMS、対象言語及びパラメータを設定します。
	GeneXus は、選択されたプラットホーム上でデータベーススキーマと全てのプログラムを生成及び保守します。
	従って、GeneXus アナリストは対象プラットホームの深い知識を必要としません。

もし初めて GeneXus トライアル版を開いた場合、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。

初めてトライアル版を起動すると、ナレッジベースは SQL+.NET 環境において自動的に作成されます。

新しいナレッジベースと新しい環境を作成する場合は、下記のステップのとおりに行ってください:

- メインウィンドウに表示されている開 始ページ内にある、「新規ナレッジベ ース作成」をクリックしてください。
- 2. ナレッジベースの名前に"Demo"と入 力してください。
- ナレッジベースを作成するディレクト リを選択してください。
- 4. 生成する言語を選択してください。
- 5. 作成を完了するには「作成」をクリッ クしてください。

	名前	
	Demo	
	ディレクトリ	
	D:¥GeneXysXKB	
	> フロトタイフ環境	
	C# Environment	-
	言語:	
	Japanese	• 詳細 >>
蓋ベースの作成	8 <b>7</b> 1	

図:ナレッジベースを作成するためのダイアログボックス

#### 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 オブジュエ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
I	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(第三正規形)を推論するのに用いられま
-	
	GeneXus では、エンドユーサか物理テータベースに対してインタラクティブにテータの新規登録、更新、削除を 行うためのアプリケーションプログラムについて、トランザクションオブジェクトを用いることで実現します。



ー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。その ためには、下記のステップの通りに行ってください。

🐼 Demo GeneXus X

1. **Objects を**右クリックし

て「新規」→「オブジェク ト」を選択するか、もしくは ツールバーの「ファイル」→ 「新規」→「オブジェクト」 を選択してください。また CTRL+N ショートカットキー 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。

タイプの選択から、
 Transaction オブジェクトタ
 イプを選択してください:

名前に"Invoice"と入力してください。

デスクリプションに請求
 書と入力します。

5. 「作成」をクリック してください。



図:新しいオブジェクトの作成

カテゴリの選択		タイプの選択		
Web Web Workflow Documentation		Business Process Diagram         Data Provider         Data Selector         Data View         Diagram         Document         External Object         Image         Language	Master Page Frocedure Subtype Group Theme Transaction Web Component Web Panel	
名前:	Invoice			
デスクリプショ	諸求書			
フォルダ:	Objects		-	

図:新しい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

次のステップでは、請求書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定義します。

	トランザクションオブジェクト構造(Structure)
	トランザクションオブジェクト構造は、現実のオブジェクトが必要とするデータ(項目属性)を表すものです。
	トランザクション構造では、(エンドユーザが操作するデータ)項目属性とそれらの関連性を定義します。
	この構造に基づいて、GeneXus は第三正規形に対応するデータベース(テーブル、キー、インデックス、整合性
	制約、その他)を自動的に設計及び保守します。
	各項目属性は、どこで使用されても常に同じ名前でなければなりません。 
	更に、2 つの異なる項目属性は同じ名前を持つことはできません。
-	トランザクション構造を定義する主な要素は、以下の通りです:
	<ul> <li>名前、タイプ、デスクリプション:各項目属性は名前、データタイプと説明によって定義されます。</li> </ul>
	レベル:項目属性は一つ以上のレベルでグループ化されます。     これらのレベルは、入れ子もしくは並
	列に定義できます(複数の入れ子レベルを定義することも可能です)。
	例:請求書の明細行は、ルートレベルから1つ入れ子になったレベルを表します。
	請求書の明細行レベルは、請求書が多くの行を持っていることを意味します。つまり、それは請求書と
	請求書の明細行間の1対多数の関係を定義しています。
	● 主キー項目属性(PK):
	各レベルにおいて、一つ以上の項目属性が主キーとして定義されなければなりません。
	o 主キーは、各レベルのインスタンスの識別子です。
	○ 主キーの値は唯一で、一旦主キーが入力されると更新することができません。
	○ もし自然にオブジェクトの主キー(識別子になる項目)が決まらない場合は、明示的に例えば
	CustomerId の様な主キーを作られなければなりません;

生産性ヒント:

• • 4	新しい項目属性の名前を入力し始める時にピリオドキーを押すことにより、トランザクションの名前が
Ň	項目属性名に書きこまれます。
•	項目属性のデスクリプションが自動的に推論されて意味をなすために、項目属性名にキャメルケースを
	使用してください(日本語の場合は項目属性名から意味を推論することはできません)。
	例:「InvoiceTotal」項目属性の場合は、デスクリプションが「Invoice Total」と推論されます。
	データタイプ(型)をコンボボックスから選択するには、指定するデータタイプの最初の文字を入力し
	てください。
	また、 Character 型や LongVarChar 型、Numeric 型あるいは VarChar 型の場合は、桁数を設定して
	ください。



下のテーブルで示される通り、Invoice トランザクションの Structure エレメントに項目属性名、データタイプとデスクリプ ション(説明)を入力してください。

項目属性名、データタイプとデスクリプションの間でカーソルを移動するためには、TAB キーを使用してください。新 しい項目属性を追加するには、ENTER キーを使用してください。

名前	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、請求書のヘッダ項目	目を入力しました。これか	らは明細行を入力します。
明細行を入力するために、デ-	- タ構造に新しいレベルを	加えます。CTRL +右矢印を押してください。
ProductID	Numeric(4.0)	製品番号
ProductName	Character(20)	製品名
ProductPrice	Numeric(8.2)	単価
LineQuantity	Numeric(4.0)	数量
LineTotal	Numeric(8.2)	行金額
ENTER そして CTRL +左矢印	-  を押して、ヘッダレベルに	こ戻ります。フッタを入力してください。
InvoiceSubtotal	Numeric(8.2)	小計
InvoiceTax	Numeric(8.2)	税
InvoiceTotal	Numeric(8.2)	合計

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ます。項目属性を右クリックして「設定・解除キー」又は(CTRL + K)オプションを選ぶ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( ?)で識別されます。

この例では、InvoiceIDはレベル1の識別子です、そして、ProductID項目属性はレベル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。



1. 項目属性 InvoiceID を選択後、画面の右側にマウスを動かす ことにより、選択した項目属性のプロパティメニューが表示 されます。

Vame	Type	Description	Formula	Filter		1
Invoice	Invoice	Invoice	and the second sec	Attribute: Invoice	eID	10
- @ InvoiceID	Numeric(4.0)			Name	InvoiceID	
-  InvoiceDate	Date	Invoice Date		Description	Invoice ID	
- • CustomerID	Numeric(4.0)	Customer ID		Title	Invoice ID	
-  CustomerName	Character(20)	Customer Name		Column title	Invoice ID	
Product	Product	Product		ContextualTitle	1D	
ProductID	Numeric(4.0)	Product ID		Formula		
<ul> <li>ProductName</li> </ul>	Character(20)	Product Name		Empty as null	Yes	
<ul> <li>ProductPrice</li> </ul>	Numeric(8,2)	Product Price		Type Definition		
- • LineQuantity	Numeric(4.0)	Line Quantity		Supertype		
- fr LineTotal	Numeric(0,2)	Line Total	ProductPrice * LineTotal	Based on	(nane)	
C. Investor Subject al	Normanic (S. 2)	Investora Subtritial	e m(lineTotal)	Data Type	Numeric	
f. InvoiceTax	Numeric(0.2)	Invoice Tay	InvolceSubTotal * 0.11	Length	4	
f- InvoiceTotal	formatic(0.2)	Invoice Total	InvoiceSubtotal + Invoice1	Decimals	0	
Ja sirocorosa	(Manufactoric)	a moleci roca	2110102320104 1 2170001	Signed	False	
				Autonumber	False	
				Walidation		
				Value range		
				Validation Failed N	Nost	
				* Reorganization		
				- Picture		
				Left fill	Blank.	4
				<ol> <li>Cateoprist</li> </ol>		

#### 図:項目属性の定義画面



図:「Autonumber」プロパティ



図:Invoice トランザクションの Structure エレ メント

 「Type Definition」オプションを展開して、InvoiceIDの 「Autonumber」プロパティを True に設定してください。

生成されるプログラムで InvoiceID (請求書番号) は、自動 採番されます。

3. 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てください。

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てください。保存をするもう一つの方法は メニューの「ファイル>>保存」オプションから保存ができます。

これでレベルが二つあるトランザクションが定義されました。

- レベル1(Invoice)に請求書ヘッダに必要とされる情報の定義。
- 入れ子レベルに請求書の明細行で繰り返される情報の定義。

請求書の明細行では行番号を主キーとして使っていないことに注目してください。

これは、意図的な設計です。

請求書の明細行に Product ID を主キーとして使用することで、同じ製品が同じ請 求書に 2 回登録されることがありません。



	普遍的な関係仮定(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、全て同じ名前でなければなりません。
Ĩ	<ul> <li>異なる概念を持つものは、同じ名前を持ってはいけません。</li> <li>これにより、同じ項目属性を他の GeneXus オブジェクト(トランザクション、プロシージャ、その他)で参照す</li> </ul>
	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。
	GIK(Genexus インクレメンタル知識) 命名規則
	GIK は、Artech 社が作った命名規則です。
	項目属性の概念を出来るだけはっきり伝えられるように名前が付けられています。
	この命名規則では、項目属性の名前は四つの構成要素からできています。そして、そのうちの幾つかは任意で
	<b>र्व</b> :
	● オブジェクト+カテゴリ[+ 特性] [+ 補足]
-	オブジェクト : 実際のエンティティ名。項目属性の名前に表すことで項目属性の管理がしやくなりま す。
!	通常は、項目属性が定義されたトランザクションの名前になります。項目属性は定義されたトランザク
	ションと関連したテーブルに保管されますので、項目属性の名前には常にトランザクション名をつけな くても構いません。
	(これは、特定のトランザクションのみで推論されるものではありません)。
	例:Invoice、Customer、 Product、Line 。
	<ul> <li>● カテゴリ:項目属性の意味を表すカテゴリです。項目属性がオブジェクト間及びトランザクション環境で果た す役割を育味します。</li> </ul>
	例: ID、Date、 Invoice、 Description など。
	● 特性及び補足は任意な部分です。例:Start、End、Minimum、 Maximum、など。

#### ステップ5:計算された項目の定義⇒式

式
式は、他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、データ型と説明を定義し、さらに計算式を
定義します。
<ul> <li>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</li> </ul>
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、あるいは他の GeneXus オブジェクト (レポート、ウェブパネル、
   その他) 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。

ユーザが定義する変数
 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。

以下の式項目属性を定義しましょう:

LineTotal=ProductPrice \* LineQuantity InvoiceSubTotal = SUM(LineTotal) InvoiceTaxes= InvoiceSubTotal \* 0.05 InvoiceTotal = InvoiceSubtotal + InvoiceTax

- 1. (項目属性のデスクリプションの右側にある)LineTotal 項目属性の式の列をダブルクリックしてください。
- 次の式を入力してください:「ProductPrice \* LineQuantity」。式のエディターを開くには、項目属性のプロパティー覧にある 「Formula」プロパティから右クリックで開くこともできます。
- 3. このセクションの始めにリストされた残りの式は、ステップ1と2を繰り返して定義してください。

新しい式を保存するためには「**保存**」をクリックしてください。

名前	タイプ	デスクリプション		ヌル許容
🖃 🧮 Invoice	Invoice	請求書		
	Numeric(4.0)	請求書番号		No
- 🔑 InvoiceDate	Date	請求書日		No
<ul> <li>CustomerID</li> </ul>	Numeric(4.0)	顧客番号		No
<ul> <li>CustomerName</li> </ul>	Character(20)	顧客名		No
🖨 🔚 Product	Product	Product		
💡 ProductID	Numeric(4.0)	製品番号		No
🔑 ProductName	Character(20)	製品名		No
··· • ProductPrice	Numeric(8.2)	単価		No
LineQuantity	Numeric(4.0)	数量		No
<sup>∟</sup> <b>f</b> ≈ LineTotal	Numeric(8.2)	行金額	ProductPrice * LineQuantity	
− <b>f</b> ∗ InvoiceSubTotal	Numeric(8.2)	小計	sum(LineTotal)	
− f <sub>*</sub> InvoiceTax	Numeric(8.2)	税	InvoiceSubtotal * 0.05	
	Numeric(8.2)	合計	InvoiceSubtotal + InvoiceTax	

図1: 式が定義された Invoice(請求書)トランザクション



#### ステップ6:GeneXusが推論したデータモデルを見ます

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 はエンドユーザのエンティティに対応する最適なデータモデル(第3
ļ	このテータモデルに基づいて、DBMS が設定されますと物理テーブルが主成されます。 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、必要であればアナリストが指定した名前に変更することが可能です。
	・GeneXus は冗長無しで第三正規形においたデータモデルを推論します。しかしながら、必要であれば
	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、前の入れ子にされた 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てください。
- 2. Invoice と InvoiceProduct テーブルをダブルクリックして開いていください。GeneXus は、トランザクションオブジェクトから推論 したデータ構造を表示します。

📔 ナレッジベースナビゲータ 🎾 🔀	✓ 図開始ページ / ■ Invoice			<b>▼</b> X
フォルダビュー	名前	タイプ	デスクリプション	定
⇒ クイックアクセス	🖃 💷 Invoice構造		諸求書	
Pil QuickStart	- 🕆 🔋 InvoiceId	Numeric(4.0)	諸求書番号	
Main Programs	<ul> <li>InvoiceDate</li> </ul>	Date	諸求日	
🖨 🛅 Objects	• CustomerId	Numeric(4.0)	顧客番号	
⊕ 🔁 GeneralWeb	- CustomerName	Numeric(4.0)	顧客名	
Gx0010	■ 20 ロジカルな項目属性			
	<b>f</b> ∗ InvoiceSubTotal	Numeric(4.0)	小計	sum(Line Total)
🎧 Messages	<i>… f</i> ∗ InvoiceTax	Numeric(4.0)	税	InvoiceSubTotal*0.05
■ ▲ ドキュメンテーション	🖙 🖍 InvoiceTotal	Numeric(4.0)	合計	InvoiceSubTotal*InvoiceTax
□ □ 15/1 ノ □ □ 15/1 フ				
Invoice				
🛄 InvoiceProduct				
□ ● ● テーブル ■ 「Invoice ■ InvoiceProduct ■ ● ● カスタマイズ				

#### 図: Invoice(請求書)テーブル



図: InvoiceProduct (Product) テーブル



上のデータベース一覧において、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確認できます。推論した結果、Invoice(請求書)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:<u>Invoice</u>(請求書のヘッダ)とInvoiceProduct(請求書の明細行):

Invoice	InvoceProduct
InvoiceID	InvoiceID
InvoiceDate	ProductID
CustomerID	ProductName
CustomerName	ProductPrice
	LineQuantity

注釈:

- 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 テーブルにおいて
  - 。 同じ InvoiceID を持つ請求書は存在しません。
  - 。 InvoiceID ごとに InvoiceDate、CustomerID と CustomerName には **1つ**の値しか存在しません。
- InvoceProduct テーブルにおいて
  - 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 と LineQuantity には 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見ます

作成したトランザクションオブジェクトに GeneXus は自動的に既定のウェブフォームを生成しますので、それを確認しましょう。まず Invoice(請求書)トランザクションオブジェクトを開いてください。

	ウェブフォーム
<u>!</u>	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 はエンドユーザがアプリケーションでデータにア
	クセスする既定のウェブ画面を作成してくれます。
	これらのウェブフォームは、GeneXus 設計者によるカスタマイズが可能です。

ウェブフォームを見るためには、下記のステップを行ってください:

1. Invoice(請求書)トランザクションオブジェクトの Web Form エレメントを選択してください。



/	Grow t 家園始ページ		nvoice <b>x</b>				+
	<ul> <li>71</li> </ul>	-4					 Þ
		) @ 🤉 🗜	<b>.</b>				 <u> </u>
	請求書 • Erro	rviewer: ctlE	rror				
	諸求書番号	InvoiceId					
	請求書日	InvoiceDate	е				
	顧客番号	Custome					
	顧客名	CustomerN	lame				
	Prod 製品番号		製品名	単価	教量	行金額	Ξ
	Prod 製品番号 Productid		<mark>親品名</mark> ProductName	単值 ProductPrice	<b>鼓量</b> LineQuantity	行金額 Line Total	Ξ
	Prod 288 # 7 Productid		<mark>製品名</mark> ProductName	単価 ProductPrice	<mark>教业</mark> LineQuantity	<mark>行金額</mark> Line Total	H
	Productid		<mark>製品名</mark> ProductName	単価 ProductPrice	教皇 LineQuantity	行全額 LineTotal	Ш
	Productid	luct	<mark>製品名</mark> ProductName	単值 ProductPrice	教量 LineQuantity	行全額 Line Total	Η
	Productid 小計	Iuct	製品名 ProductName	単価 ProductPrice	教皇 LineQuantity	行全額 Line Total	E
HTML	Prod Productid 小計 商	InvoiceSub InvoiceTax InvoiceTota	製品名 ProductName	単価 ProductPrice	教皇 LineQuantity	行金額 Line Total	Ħ
ign HTML	Productid Productid	Iuct InvoiceSub InvoiceTax InvoiceTota	<mark>製品名</mark> ProductName	単信 ProductPrice	教皇 LineQuantity	行金額 Line Total	111
Design HTML	Productid	InvoiceSub InvoiceTax InvoiceTota	<mark>製品名</mark> ProductName	単作 ProductPrice	<b>致业</b> LineQuantity	行全額 Line Total	jiii ▼

図: Invoice トランザクションにおける既定のウェブ画面

"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することができ、また、プロパティの設定を変更することができます。

実行時に表示されるメッセージは、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、データ型エラー、その他)。

このフォームでエンドユーザ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、ユーザは既存の請求書を更新及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 設計者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オブジェクトを定義する際には、以下のことを意識してください:

- 明示的に:データの表示及び入力用のユーザインタフェース定義。
- 暗黙的に: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、その他)



#### 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
	<ul> <li>推論したデータモデルに基づいて、GeneXus は選択された DBMS でデータベースを作成するための実行プログラムを生成します。</li> </ul>
ī	• 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ための実行プログラムを生成します。
	<ul> <li>再編成を行う際には、データベース構造の変更を行いデータも新しい構造に移行します。</li> </ul>
	<ul> <li>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あるいは影響分析レポートが表示されます。</li> </ul>

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。 正規バージョンでは、GeneXus がサポートしている DBMS のどれでも使用することができます。 GeneXus がサポートしている DMBS の一覧を見るには以下のリンクをクリックしてください: http://www.genexus.com/technologies

データベース作成レポート:
 これは、選択された DBMS において 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全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:テーブルの名前、テーブルに対してのアクション、警告及びエラーを含みます。
 データモデルにエラーがある場合は、再編成ボタンは使用不能になります。
 デーブル構造:テーブルの項目属性、それらの関連及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の管理を行うためのインデックスと能率的にデーブルを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。

F5 を押すか、もしくはツールバーの「ビルド>>開発者メニューを実行」オプションを選択してください。



- DBの設定について、**ウィ ザード**に従い、以下の通り
   に設定してください:
  - データベース名:
     Demo
  - サーバー名:
     localhost
  - 信頼認証接続を使用 します:Yes

続行するために、Finish ボ タンをクリックしてくださ い。

 GeneXus はデータベース 作成レポート(Impact Analysis)を出力します。

ne rollowing properties must ith appropriate values.	be set to start the Build process.	
Database name	Demo	
Server name	localhost	
Use trusted connection	Yes 💌	

図:モデル設定の詳細



図:データベース作成レポート画面

 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 タベースをアクセスしま す。

> また、各オブジェクトの生 成結果は、ナビゲーション ビューに表示されます。

🔮 🧕 AppMasterPage	Web Panel AppMasterPage Navigation Report			
G G X0010	Name	AppMasterPage	Environment	
Contraction Contraction	Description	Application Master Page	Snec Version	
PromotMasterPage			Form Class	
Recent links			Program Name	
U TOCORCENICS			Parameters	
表示: Build 🔹 🗙				
Processing Searching for new or changed tables				
Processing Searching for new or changed tables Analyzing Invoice Removing specification files Specification files removed. Generators messages for Japanese Generators messages generated. Analyzing InvoiceProduct Searching for deleted tables Searching redundancy in Invoice				
rrocessne Searching for new or changed tables Analyzing Invoice Removing specification files Specification files removed. Generators messages for Japanese Jaenerators messages generated. Analyzing InvoiceProduct Searching for deleted tables Searching for deleted tables Searching redundancy in Invoice Updating redundancy in Invoice				
Processne				
*rocessne *rocessne Searching for new or changed tables Analyzing Invoice Specification files removed. Specification files removed. Benerating messages for Japanese Benerators messages generated. Analyzing InvoiceProduct Searching rodundancy in Invoice Jpdating redundancy in InvoiceProduct Jpdating redundancy in InvoiceProduct Disartime redundancy in InvoiceProduct Specificing VIED				
Processing - Searching for new or changed tables - Analyzing Invoice - Removing specification files - Specification files removed. Jenerators messages for Japanese - Jenerators messages generated. Analyzing InvoiceProduct - Searching for deleted tables - Searching redundancy in Invoice Updating redundancy in InvoiceProduct Specifying GXLRED - Loadne table and attribute properties				
Processing Searching for new or changed tables Analyzing Invoice of the searching specification files Specification files removed. Benerating messages for Japanese Amatzing InvoiceProduct Searching for deleted tables Searching redundancy in Invoice Updating redundancy in Invoice Dipdating redundancy in Invoice Searching redundancy in InvoiceProduct Jpdating redundancy in InvoiceProduct Specifying GXLRED 				

図:出力ウィンドウのメッセージ



開発者メニュー

 (DeveloperMenu)は、
 実行可能なオブジェクト全
 てを表示する 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。請求書のリンクを
 クリックしてください。

GeneXus Developer Menu - Windows Internet Explore	er				
🖉 🗢 🖉 http://localhost/Demo.NetEnvironment/execute.>	xml	<b>×</b>	Google 🖌		۶.
🍘 🍘 🍘 GeneXus Developer Menu				💧 🖬 🔻	S - 🖶 -
			$\geq$		
Developer Menu					

図:開発者メニュー

# **Application Header**

- いくつかの請求書を登録し てください。AJAX 技術を 利用しており、ページを再 ロードせずとも、式の値を 自動的に算出します。
- 完了したらブラウザーを
   閉じてください。

Recents: 請求書	
②       ③       ③       ③       ☑       ☑       ☑       ✗         請求書       ①       □       □       □       □       □         請求書日       ○       ○       □       □       □       □         請求書日       ○       ○       □       □       □       □         顧客番号       ○       □       □       □       □       □         顧客名       ○       □       □       □       □       □	
Product 횛品춃당 혫品名	単価 数量 行金額

0	0.00	0	0.00
0	0.00	0	0.00
0	0.00	0	0.00
0	0.00	0	0.00
0	0.00	0	0.00
	[行追加]		

小計	0.00
税	0.00
合計	0.00
<b>実行</b>	\$ <b>7</b> 副除

図:請求書トランザクションのインスタンス



#### 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。
```

	<b>GeneXus ルール</b> GeneXus <b>ルール</b> 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。
	ルールは宣言的に記述し、GeneXus はどのルールをいつ適用すべきか(トリガーオーダー)を知的に決定しま す。
ļ	これらの <b>ルール</b> 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持っています。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)。
	<ul> <li>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、変数及び関数を含める(記述する)ことができます。</li> <li>ルールは、<b>宣言的な</b>手法で定義します。ルールが記述される順序が、必ずしも実行される順序どおりである必要はありません。</li> <li>ルールの実行順序は、GeneXus が自動的に決定します。</li> </ul>
	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。ルールは <b>ローカル</b> 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(請求書)トランザクションの Rules エレメントを選択してください。
- 2. 項目属性及び変数の既定値を設定する Default ルールを使用しましょう。
- 3. 次の通りにルールを記述してください: 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。

4. 保存ボタンをクリックしてください。



図:トランザクション ルール



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:

- 次のルールを記述してください:
   Error("購入数量を入力してください") if LineQuantity.IsEmpty();
- 6. **保存**ボタンをクリックしてください。



7. 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しょう。(F5 を押下、もしくはツールバーより管 理者メニューのビルドを実行する)

# **Application Header**

Recents: 請求:	·····································			
<ul> <li>① ① ① ③</li> <li>請求書</li> </ul>	🔍 🛃 🔊 🗙			
諸求書番号	1			
諸求書日	11/02/01			
顧客番号	1			
顧客名 顧客	601			
Draduct				
Floudet				
	製品番号 製品名	単価	購入数量を入力してください 蕃	i.
×	1 製品01	10.00	0.00	)
	0	0.00	0 0.00	)
	0	0.00	0 0.00	)
	0	0.00	0 0.00	) )
	0	[行追加]	0.00	
小計 0.00	)			
税 0.00	)			
合計 0.00				
<b>実行)(株7)</b>				
		図:アプリケーションの実行		



#### ステップ11: Customer(顧客)トランザクションオブジェクトの作成

顧客は請求書とはまた別のエンティティなので、顧客用のトランザクションを定義しなければなりません。

- 1. 「名前: Customer; デスクリプション=顧客」トランザクションオブジェクトを作成してください
- 2. Customer (顧客) トランザクションの構造に以下の項目属性を定義してください:

名前	タイプ	デスクリプション
CustomerID		
CustomerName		
CustomerAddress	Character(50)	顧客住所
CustomerEmail	Character(50)	顧客電子メール

「CustomerID」と「CustomerName」項目属性の入力を始めると、GeneXus は全体の名前、タイプとデスクリプションを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(サジェスト)が起こります。

- CustomerID(顧客番号)項目属性を右クリックしてプロパテ イを選択してください。
- CustomerIDのType Definition グループ内にある
   Autonumber プロパティを True に設定してください。
   この定義によって、CustomerID が自動採番されます。
- CustomerID のプロパティを開いて、サジェスト機能を有効化 しましょう。まずは InputType プロパティを Description に 設定します。そして、ItemDescription に CustomerName 項 目属性(顧客名)を設定します。

この手法は顧客 ID を特定するために、ID を入力する代わりに 顧客名を入力することにより顧客番号は自動的に推論されま す。 サジェストプロパティは、入力されたものと一致する顧客名を サジェストしてくれます。 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。

E Z♥ Z♥ //////	<b>`</b>
Name	CustomerID
Description	頭安番号
Title	顧客番号
Column title	顧客番号
ContextualTitle	顧客番号
Formula	
Empty as null	Yes
<ul> <li>Type Definition</li> </ul>	
Supertype	
Based on	(none)
Data Type	Numeric
Length	4
Decimals (	0
Signed	False
Autonumber	True
Autonumber start	1
Autonumber step	1
Autonumber for replic	True
Initial value	
+ Validation	
+ Picture	
<ul> <li>Control Info</li> </ul>	
ControlType 8	Edit
InputType I	Descriptions
ItemValues (	CustomerID
ItemDescriptions	CustomerName
Suggest 1	Incremental
FilterOperator	Begins with
SuggestMaxItems	5
SortDescriptions	True
CaseSensitive	False
CaseSensiuve	
Conditions	
Conditions InstantiatedAttribute	

図:プロパティ画面



下記の図は、顧客トランザクションのウェブフォームを示しています。

前	タイプ	デスクリプション	式	ヌル許容
Customer	Customer	顧客	dia.	ð til
- 💡 CustomerId	Numeric(4.0)	顧客番号		
- 🔑 CustomerName	Character(20)	顧客名		No
- CustomerAddress	Character(50)	顧客住所		No
CustomerEmail	Character(50)	顧客電子メール		No

#### 図:Customer(顧客)トランザクションの Structure エレメント

Frrory	iewer: ctlFrror	
顧客番号	Custome	
顧客名	CustomerName	
顧客住所	CustomerAddress	
顧客電子メール	U CustomerEmail	

図: Customer (顧客) トランザクションの Web Form エレメント



Invoice(請求書)トランザクションが既に開かれている場合は、閉じて開きなおします。CustomerID(顧客番号)項目属性 のプロパティ変更が請求書のウェブフォームにまで反映されたことにお気づきでしょうか? 顧客名のところには CustomerName(顧客名)項目属性から、CustomerID(顧客番号)項目属性に変更されていることが分かります。

ご確認いただけたとおり、これは大いにアプリケーションのユーザビリティを強化します!

J <b>○</b> フォーム					
	<b>N N N</b>				
請氷書 ● Errorviewer	ctlError				
<sup>唐求書番号</sup> Invoic	eld				
唐求書日 Invoic	eDate				
顧客名 Custo	merld				
휧 <b>品</b> 퓹号 Reducted	製品名 BraduetNama	単価	教量	行金額	
製品番号 ProductId	製品名 ProductName	単值 ProductPrice	教量 LineQuantity	行金額 Line Total	
製品番号 ProductId	製品名 ProductName	単価 ProductPrice	教量 LineQuantity	行金額 LineTotal	
휓品番号 Productid	製品名 ProductName	単価 ProductPrice	<u>教</u> 量 LineQuantity	行金額 LineTotal	
製品番号 ProductId	製品名 ProductName	単価 ProductPrice	教业 LineQuantity	行金額 LineTotal	
製品番号 ProductId	製品名 ProductName	単価 ProductPrice	款量 LineQuantity	行金額 Line Total	
製品番号 Productid 小計 Invoic 短 Invoic	製品名 ProductName eSubtc eTax eTotal	単価 ProductPrice	教量 LineQuantity	行金額 Line Total	
製品番号 ProductId 「計」 Invoic 現 Invoic	製品名 ProductName aSubtc eTax eTotal	単価 ProductPrice	教业 LineQuantity	行金額 Line Total	
製品番号 ProductId 小計 Invoic 范 Invoic 合計 Invoic	<b>製品名</b> ProductName ■ ■ ■ ■ ■ ■ ■ ■ ■ ■ ■ ■ ■ ■ ■ ■ ■ ■ ■	単価 ProductPrice	教皇 LineQuantity	行金額 Line Total	
製品番号 ProductId	¥品名 ProductName eSubtc eTax eTotal	単価 ProductPrice	教量 LineQuantity	行金額 Line Total	

図:請求書トランザクションの Web Form エレメント

#### ステップ12:データモデルに対する影響・変更内容の確認

F5 キーを押して、影響分析レポートを生成してください。

I	影響分析
_	GeneXus が推論したデータモデルの変化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(顧客名)項目属性が CustomerID(顧客番号)項目属性を通して参照す ることができるため、Invoice テーブルから CustomerName(顧客名)項目属性を削除します。



	e			
Customer	Table Invoice speci	fication		
	Table name: Invoice			
	Invoice needs conversi	on		
	Table Structure			
	Attribute	Definition	Previous values	Та
	InvoiceId	Numeric (4)Not null Autonumber		Inv
	InvoiceDate	Date Not null		Inv
	Del Customeria	Character (20)Not null		10
	Indexes			
	Indexes	Definition	Compo	sitic
	TINVOICE	primary key Clustered	🔺 Inv	voiceI
	New IINVOICE1	duplicate	🛽 Cu	stom
	Foreign key constra	aints		
	Referenced t	able	Attributes	
	Now Customer		CustomerId	

Customerの新しいテーブルが作成されます:

[***]	注	GeneXus の分析表で請求書テーブルに複製の顧客がある可能性から正規化する際に影響を受けることを
	釈	警告で示します。もう一つの警告は、テーブルの新しい項目属性はヌルで作成されることを示します。

が、 人工の変更し、 続行の場合は再編成を、	もしくはキャンセルを選択して下さい。 Close	75010004 CODA - NOBRO16499					
▲ I Customer	Table Customer spec	cification					
Man Invoice	Table name: Custome	Table name: Customer					
	Customer is new						
	Warnings	Warnings					
	▲ rgz0007 Attribute C value will b ▲ rgz0007 Attribute C value will b ▲ rgz0005 For each va	<u>SustomerAddress</u> does not allow nu le used. <u>SustomerEmail</u> does not allow nulls le used. alue of <u>CustomerId</u> there may be s	lis and has not a Initia and has not a Initial \ several values of <u>Cus</u>	al Value. A /alue. An stomerNar			
	Table Structure						
	Attribute	Definition	Previous values	Takes			
	CustomerId	Numeric (4)Not null Autonumber		Invoice.			
		Character (20)Not null		Invoice.			
	CustomerName	Character (Lo)not nan					
	CustomerName CustomerAddress	Character (50)Not null		21.1 2			
	CustomerName CustomerAddress CustomerEmail	Character (50)Not null Character (50)Not null		12			





注釈:

- GeneXus は新しいテーブルを追加しました: Customer テーブル (Customer (顧客) 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 テーブルに関して:
  - 。 複数の顧客が同じ CustomerID を持つことはできません。
  - 。 各顧客に対して一つの名前、住所と電子メールしか持てません。
- 請求書テーブルに関して:
  - 。 複数の請求書が同じ InvoiceID を持つことはできません。
  - Invoice テーブルの Customer ID 項目属性は Customer テーブルからの外部キーです。このように、
     Customers と Invoices 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:分析表を見ます

データベースの再編成を行う準備ができました。

再編成オプションを選択してください。

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各プログラムの解析表を含む分析表を表示してくれます

Table	Program	In Paramete	ers Out Parameter	15
Prompts				-
Referential in <u>Invoice(</u>	tegrity controls on del <u>ustomerId</u> )	ete:		
Delete from	Customer			
Update Cust	comer ( <u>CustomerNam</u>	e, <u>CustomerAddress</u> , <u>Cu</u>	stomerEmail.)	
Under a		C. de martida en c	-to	
Insert into (	Customer ( CustomerN	lame CustomerAddress	CustomerEmail )	
=	Customer( <u>CustomerId</u>	)		
Level Custom	er			
Levels				
		Parameters	Castomer	
		Form Class Program Name	HTML Customer	
Description	110-	Spec. Version	A 10_0_3-18367	
Name	III <u>Customer</u>	Environment	.net C#	
Transaction C	ustomer Navigation	Report		

図:顧客トランザクションの解析表

「削除する際の参照整合性の制御」とは、顧客トランザクションから顧客を削除する時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。



Transaction Invoice Navigation Report				
Name II Invoice	Environment	.net Default (C#)		
Description	請氺書	Spec. Version	▲ 10_1_3-34783	
		Form Class	HTML	
		Program Name	Invoice	
		Parameters		
Levels				
Level Invoice				
EI= <u>Invo</u> E	ice( <u>InvoiceId</u> ) == <u>Customer( Custom</u> =~ <u>InvoiceSubtotal na</u> = <u>InvoiceProd</u>	<u>nerId)</u> vigation luct( <u>InvoiceId</u> )		
Insert into <u>Inv</u>	<u>oice</u> ( <u>InvoiceDate</u> , <u>C</u>	<u>ustomerId</u> )		
Update <u>Invoic</u>	e ( <u>InvoiceDate</u> , <u>Custo</u>	omerId )		
Delete from <u>Ir</u>	ivoice			

#### 図: 請求書トランザクション解析表

請求書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時に、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、全ての有効な値をその中から選ぶことができます。

#### ステップ14:見栄えの良いフォームの作成(テーマ)

このセクションでは、あらかじめ定義されたテンプレートに基づいて新しいテーマを作成し、アプリケーションのテーマとし て指定します。

	テーマオブジェクト
	GeneXus テーマオブジェクトでビジネスアナリストの仕事とウェブデザイナーの仕事を分けてウェブアプ リケーションの開発と保守が行えます。
:	<ul> <li>ウェブフォームは、既定のテーマオブジェクトに基づいて作成されます。</li> <li>新しいテーマを作成してナレッジベース全体に適用、あるいは特定のオブジェクトに適用することができます。</li> </ul>



- 左のフォルダビューの中にあるカスタマ イズノードを展開してテーマノードをさ らに展開します。そうすると、いくつか のテーマオブジェクトが確認できます。
- 2. GeneXusX テーマオブジェクトをダブ ルクリックして開いてください。
- Form を選択して右にあるプロパティ編 集画面から「Defaults for From Control Properties」の配下にある 「BackGroundColor」プロパティを変 更し背景色を変更してください。
- 4. 保存ボタン(<sup>1</sup>→)を押して、変更を保存 してください。
- 再び送り状トランザクションのウェブフ オームを開くと、背景色が反映されたこ とが見られます。
   注釈:既にトランザクションが開いてい て行った変更が見られない場合は、一度 トランザクションを閉じて再度開きなお

してください。

Demo GeneXus X	PROFESSION AND ADDRESS OF ADDRESS	
ファイル (E) 編集 (E) 表示 (Y)	レイアウト 挿入(1) ビルド(日) ナレッジマネージャ(K) ウィンドウ	<ul><li>(W) ツール (I) ヘルプ(H)</li></ul>
088   8 <b>8</b> × 8	$[ \underbrace{ \mathfrak{A}} ] $ $[ \underbrace{ \mathfrak{G}} ] $ Verdana, Arial $\underline{ = } $ $\mathbb{S} $ $\underline{ = } $ $\mathbf{B} $ $\underline{ r} $ $\underline{ \mathfrak{U} } $ $\mathbf{T}_{\mathbf{S}} \cdot 0_{\mathbf{S}} \cdot \mathbf{ \equiv } $ $[ \underbrace{ \overset{ }{ \mathbf{ I} } } ]$	□□□ ○ 中京る 小田口 ● 第2 ○ ○ Normal 三 三 日 日 日 日 44 × 5
ナレッジベース ・検索	8.8.4 可以引回国家 日田子	⊕ 柳 朔 朔 貞 善 ff of of _ Japanese -
ナレッジベースナビゲータ		Navigation View SGeneXusX + x 3770/77
フォルダビュー	🗟 🛅 GeneXusX	HTML J905 HTMLY-2
ま クイックアクセス 型 Demo 一 画 Main Programs	Classes     Main Attribute     W Button     ErrorViewer	GX BadgroundPoston BadgroundRepeat - Box - Georfraton
● しのほどで ● ドメイン ● □ テーブル ● □ カスタマイズ	e E PresStyleGrid	Theme Deplay institution
Classic	e Croup e Hyperlink Group	Editor
With Gene Veek Modern Offic Orange	Messages     Section     Table	Titles is the page are defined using (H1) and (H2) tags.
※ 🏄 ドキュメンテーション	A TextBlock     B    B    HTMLNodes	About this
		page
	Styles 🛃 Images 📓 Documentation	
→7##9E1-	出版力 表示:Build ・X	

図:GeneXus テーマエディター

<b>000</b>						
<ul> <li>Errorviewer:</li> </ul>	ctlError					
求書番号 Invoic	InvoiceId					
求書日 Invoic	eDate					
客名 Custe	merld					
退品番号 Industid	製品名 ReductName	単価	鼓量	行金額		
성品番号 roductId	<b>왕品名</b> ProductName	単価 ProductPrice	教量 LineQuantity	行金額 LineTotal		
战品 都 号 <sup>9</sup> roductId	製品名 ProductName	単価 ProductPrice	放量 LineQuantity	行金額 LineTotal		
製品番号 ProductId	製品名 ProductName	単価 ProductPrice	数量 LineQuantity	行金額 LineTotal		
設品番号 ProductId	왕읍名 ProductName	単価 ProductPrice	数量 LineQuantity	行金額 LineTotal		
設品 番 号 Productid	製品名 ProductName	ProductPrice	鉄业 LineQuantity	行全額 Line Total		
왕문 16 동 Productid 타 Invoic 타 Invoic	RBA ProductName Subtc Tax PTotal	192 (m) Product Price	教皇 LineQuentty	行全額 LineTotal		
設品番号 ProductId 타 Invoic 타 Invoic	製品名 ProductName Subtc eTax eTotal	Product Price	秋日 LineQuentty	行全知 LineTotal		
設品番号 Productid 計 Invoic 計 Invoic 計 Invoic	R	Product Price	85 92 LineQuantity	行金90 LineTotal		
設品番号 Productid 計 Invoic 計 Invoic 計 (株子) 番	ReductName ProductName Subtr PTax PTotal	Product Price	85 92	<b>行金額</b> LineTotal		

図:テーマ変更により、背景色が変化した

Invoice(請求書)トランザクションウェブフォーム



#### パート3:保守



推定された時間:15分

#### インクリメンタル開発とアプリケーション保守

これまではいくつかのトランザクションオブジェクトとビジネスルールに基づいてアプリケーションを作成してきました。 今度は GeneXus アプリケーションの保守について、単にオブジェクトを編集もしくは新しいオブジェクトを追加し、その後 でデータベースを更新して自動的にアプリケーションプログラムが再生成される事を確認してみましょう。

	GeneXus によるインクリメンタル開発
	実務が変わることに従って、システム要件も変化します。そしてその変化は GeneXus オブジェクトの更
	新あるいは新しいオブジェクトの追加により対応します。
	新しいオブジェクトに基づいて GeneXus は自動的にデータベース(システム変更をデータベース構造に
_	反映してデータの移行もしてくれます)を更新して変更の影響を受けたアプリケーションプログラムを再
<u>!</u>	生成します。
	GeneXus 方法論がインクリメンタル方法論であると言うことができます。なぜなら、その方法論は繰り
	返し作業で連続した近似に基づいたシステム開発であるからです。
	GeneXus がデータベース及びアプリケーションのプログラムの構造を自動的に保守することで、インク
	リメンタルな開発が可能になります。

#### ステップ15:新しいオブジェクトの追加-製品トランザクション

「名前:Product; デスクリプション=製品」トランザクションオブジェクトを作成してください。 アプリケーションをテストした際に、各請求書の明細行に製品番号、製品説明及び製品価格を入力しなければならないことに 気がついたかと思います。これは、販売管理システムに求められることではありません。 このサンプルアプリケーションのような単純なシステムでもそのようなアプリケーション構造は好まれません! 別のシステム画面から製品を追加、更新そして削除する機能が必要とされます。 そのために、製品トランザクションをナレッジベースに追加しましょう:

1. パート2のステップ 3: "トランザクションオブジェクトの作成"及びパート2のステップ4: "トランザクション構造の 定義"を基に製品トランザクションを作成してください。

以下の項目属性を製品トランザクション構造に挿入してください:



名前	タイプ	デスクリプション
ProductID		
ProductName		
ProductPrice		

これらの項目属性の名前を入力し始めるとすぐに GeneXus がフルネームでサジェストを表示してくれることに注目してください。

項目属性が既にナレッジベースに定義されているためそのサジェストされた値が表示されるのです。

製品トランザクションの構造とそのウェブフォームは以下のような形になります。

名前	タイプ	デスクリプション	元	ヌル許容
🖃 🔲 Product	Product	製品		
ProductId	Numeric(4.0)	製品番号		No
ProductName	Character(20)	製品名		
ProductPrice	Numeric(8.2)	単価		

#### 図: Product(製品)トランザクションの構造

	<ul> <li>フォーム</li> </ul>	4
		<u>^</u>
	製品 • Errorviewer: ctlError	
	製品番号 Productic	
	製品名 ProductName	
HTML		
Design		
	- Structure 🖥 🐔 Web Form 🔤 🧏 Win Form 🗟 Rul	s 🗑 Events 🕄 Variables  Help 🖺 Documentati 🎬 Patterns

図: Product(製品)トランザクションの Web Form

#### ステップ16:データベースの分析と再編成

製品トランザクションオブジェクトを保存する時に、GeneXus は再びデータモデルを正規化します。

データモデルチェックで 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。これまで InvoiceProduct テーブルにて構成されていた項目属性(ProductId、ProductName、ProductPrice)が、新しい Product というテーブルに移動されています。



and the state of t	-							
🖉 🗐 Invoice 🏳 🍓 Japane	se 🔚 Cust	tomer 🔄 Navi	gation View 🛛 🛅 Gene	XusX  🔚 Product 🖉 🔀 Impa	ct Analysis 📃 🔻 🗧			
テータベースは再編成を必	要とします。							
データベース上の変更と、それが 続行の場合は再編成を、もしく!	再編成プログラ はキャンセルをえ	らムでどのように処理: 醒択して下さい。	されるか、このレポートに記載	はされます。				
📲 👔 Cance								
InvoiceProduct	Table	InvoiceProdu	ct specification					
	Table	Table name: InvoiceProduct						
	Invoice	InvoiceProduct needs conversion						
	Table	Structure						
		Attribute	Definition	Previous values	Takes value fro			
	8	InvoiceId	Numeric (4)Not null		InvoiceProduct.In			
	8	ProductId	Numeric (4)Not null		InvoiceProduct.Pr			
		LineQuantity	Numeric (4)Not null		InvoiceProduct.Li			
	Del	ProductName ProductPrice	Character (20)Not nu Numeric (8.2)Not nul					
	Index	ces						
		Name		Definition	Composi			
		IINVOICEPRO	DUCT	primary key Clustered	<ul> <li>Invoi</li> <li>Prodi</li> </ul>			
	New	IINVOICEPRO	DUCT2	duplicate	🔺 Prodi			
	Forei	gn key constra	ints					

図: InvoiceProduct と Product テーブルの変化

- 1. **再編成ボタン**を押してください。
- GeneXus は、データベース再編成を実行し、またアプリケーションプログラムに必要な変更を行います。



図: 再編成と分析の実行結果を表示するダイアログボックス



 これでアプリケーションを実行することが できます!

#### Application Header Recents: 請求書 🔞 🚯 🚯 🔄 🔄 🖉 🗶 請求書 請求書番号 📃 0 請求書日 11/02/01 💬 顧客名 Product 製品番号 製品名 単価 费量 行金額 0 1 0.00 0 0.00 0 🗘 0.00 0.00 0 0 🗘 0.00 0.00 0 0 🕜 0.00 0 0.00 0 î 0.00 0.00 0 [行追加] 小計 0.00 0.00 税 合計 0.00 実行 算了 副院

図:アプリケーションの実行画面

#### パート4:機能の追加



推定された時間:10分

#### 開発の一部にパターンを使用します。

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ー して並び替えて一覧表示ができるとさらに便利になります。

この機能を迅速かつ容易な形で実装するために、パターンを使います。

Ī	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したアクション(アナリストの操作)です。
	パターンにより特定の機能を成し遂げるために必要である全てのオブジェクトを自動作成します。これにより
	手動による実装を避けることができます。

#### ステップ17:パターンの適用

「ワークウィズ」パターンを作成したトランザクションに適用しましょう。



 「表示」メニューから「オブジェクトー 覧」オプションを選択してください。
 Customer(顧客)トランザクション、
 Invoice(請求書)トランザクション及び
 Product(製品)トランザクションを選択してください。

10 M		タイプ	Transact	ion 👻	戻る <<			-
食索(キー	ワードかフレーズ):					57	クの選択	2
コテゴリ:	*すべて	<b>▼</b> 7 <del>8</del>	ルダ: *すべて	•		D.	開( (0)	
更新さ	n. <del>.</del>					×	肖·IF余	
The second second	Ture.					2	参照 (R)	
	名前	/ <u>\$1</u>	(7	デスクリプション	/ Ef	すの愛 📄	履歴 (H)	
		Transaction	顧客		2009/	D1/11 <b>\$</b>	ナビゲーション表示	
			清沢	·*	2009/		Robel McContractor	-
		Iransaction	<b>設</b> 面			JI/11 J-3	いのわっト	
						$\rightarrow$	1	
						> < <		
						>		
						> < <		
						> < <		
							-+>/520+520	\$

図 2:トランザクションを選択するダイアログ

- 3. 選択したトランザクションを右クリックして、Apply Pattern(パターンを適用します)、Work With オプションを選択してください。
- 4.変更を保存してください。
- 5. F5 を押して、アプリケーションを実行し てください。

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。

顧客、請求書と製品それぞれのパターン "Work With"の実行ができます。

パターン:	タイプ: 📊 Transaction	• ▼ 戻る <<
検索(キーワードかフレーズ): カテゴリ: *すべて 回更新された:	新規(№) ・ う 開<( <u>0</u> ) F12 開< <u>P</u> art ・	v
名前	× 削除 Del	デスクリプション 日付の愛
E Customer	名前を付けて保存 ( <u>A</u> )	2009/01/11
Product	<ul> <li>参照(R) Ctrl+F12</li> <li>履歴(<u>H</u>) Ctrl+Shift+H</li> <li>プロパティ(P) F4</li> </ul>	2009/01/11
	フォルダビューに置く ビルド リビルド 実行	
∢  出力 表示: Build	これだけを実行 これだけをビルド 量 ナビゲーション表示 対 エクスポート	
mpiling invoicesuccess	Apply Pattern	Category

図:トランザクションの選択



6. 請求書の Work With を実行してください。

パターンは、以下の機能を追加しました:

- アプリケーションを再表示せずに日 付でデータが抽出できます。
- 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開くこ とができます。
- アクセスしたリンクの履歴が保存されます。

Decester		. (	/				
Recents:	VVORK VVItn 計算3K·	뢉S					
Work	With 請求書s						
諸求書日 [	11						
							Ŧ
	請求書番号	請求書日	顧客名	小計	税	승計	
🔀 🗙	1	<u>11/02/01</u>	<u>顧客01</u>	100.00	5.00	105.00	
			Fool	er Info			
					<u>] n_tu</u>	イントニナッ	. k.
					3 0 - 777	/1ンドフホン	/r

🤌 Work With 請求書s - Windows Internet Explorer

🚖 🏟 🏾 🌈 Work With 請求書s

Application

😋 🕤 👻 http://localhost/Demo.NetEnvironmer 🗹 🐓 🗙 🛛 Google

🟠 • 🔊

# パート5:インタラクティブなダイアログ

推定された時間:15分

インタラクティブなダイアログとクエリーの設計(ウェブパネル)

# ウェブパネル ウェブパネルは、ウェブ環境でデータベースへとインタラクティブなクエリー画面を作成するために使用されます。 ウェブパネルは、様々な形のクエリーを実行してデータを多様な方法で見ることを可能にします。ウェブパネルは、単純なイベントドリブン言語で記述されます。

#### ステップ18:ウェブパネルの作成:顧客ごとの請求書

各顧客の請求書を見るためにウェブパネルを作成しましょう。 ウェブパネルを作成するためには下記のステップを行ってください:



Grow thru knowledge

- オブジェクトフォルダを右クリック して「新規」→「オブジェクト」を 選択してください。
- 作成する「Web Panel」型オブジ ェクトを選択してください。
- 名前には"InvoicesPerCustomer" を入力してください。そしてデスク リプションに、 "顧客毎の請求書" と入力してください。
- 4. 作成をクリックしてください



#### 図:オブジェクトの新規作成画面

5. オブジェクト作成後、ウェブパネル は空のフォームを表示します。

> ツールバーにある「**表示**」メニュー から「**他のツールウィンドウ**」の 「**ツールボックス**」オプションを選 択してください。

「**フリースタイルグリッド**」コント ロールをドラッグしてください。

- 「フリースタイルグリッド」コント
   ロールを拡大してください。
- ブリッド内に「項目属性/変数」コントロールを挿入してください。
- CustomerName (顧客名)の項目 属性を選択し、「OK」をクリック してください。



#### 図:フリースタイルグリッドを配置した画面



#### 図:ウェブパネルの Web Form



 各顧客に対応する請求書一覧を表示 させるためには、グリッドを挿入し て表示される次の項目属性を選択し てください。

> 「InvoiceID(請求書番号)」 「InvoiceDate(請求日)」 「InvoiceTotal(合計)」

GeneXus は顧客ごとの請求書一覧 を表示するためのテーブル間の関係 を見つけます。

- 「列の調整」画面を閉じてください。
- 11. 見を押して変更を保存してください。

12. F5 を押して、実行してください。

GeneXusX 🗐 P	Insert Grid	and and Public of	TT installed		<b>▼</b> ×	シッツールボックス
1 077-4	コノルね				Þ.	
	2.1762					4メージ
	パターン:				*	A テキストブロック
	オブジェクトタイン		*			・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・
CustomerName	オブジェクト:		*			🛄 Table
	and a second second second					□ セクション
	Create New Grid	Control				□ ボタン
	7		- 81			➡ 横野線
	その外に化って方類の	O STOR MELLER	フックしてください。			🎬 グループ
	名前	タイプ	デスクリプション	<b>^</b>		∞ 項目属性/変数
	Att CustomerAddress	Character(50)	顧客住所			目 /Juws
	Att CustomerEmail	Character(50)	顧客電子メール			
	Att CustomerId	Numeric(4.0)	顧客番号			E フリースタイルグリッド
	Att CustomerName	Character(20)	顧客名			Web コンポーネント
	Att InvoiceDate	Date	諸求日			60 71 21 71 491 21
Ê	Att Invoiceld	Numeric(4.0)	請求書番号			
-	Att InvoiceSubtotal	Numeric(8.2)	小計			
6	Att InvoiceTax	Numeric(8.2)	税			□ フ ーサfフントロール
8	<b>⊿</b> # InvoiceTotal			E		The check
	Att LineQuantity	Numeric(4.0)	数量		T	iki Chart
Woh Form * Et P	Att LineTotal	Numeric(8.2)	行の合計金額			ImageGallery
	Att Productid	Numeric(4.0)	製品番号			] 🖆 ブロバー・ 🔆 ツール・・・ 🖧 インミ
司出力	Att ProductName	Character(20)	製品名			(
表示: Build	Att ProductPrice	Numeric(8.2)	単価	Ţ.		× ▲ 自動スクロー
Barris Duild						
Compiling home_success "C:¥GeneXusXKB¥Demo¥C: Updating web config	新い変数	Edit	OK ++2	den .		

#### 図:グリッド挿入画面

stomentame		
InvoiceId	InvoiceDate	InvoiceTotal
		E
		E

図:グリッドを持ったウェブパネル画面

Appl	ication Header	e
Recents:	顧客毎の請求書一覧	
丽安(1)		

請求書番号	請求日	合計
1	2009/02/02	12500.00
2	2009/02/02	50000.00

顧客02		
請求書番号	請求日	승計
3	2009/02/02	12500.00

図:アプリケーションの実行



#### ステップ19:パターンインスタンスを変更してウェブパネルを再び実行

顧客トランザクションの「Work With 顧客」パターンインスタンスを変更して、「InvoicesPerCustomer」ウェブパネルを 呼び出せるようにしましょう。

 Customer(顧客)ト ランザクションを開い て、Patternsエレメン トを選択し、パターン インスタンスを開いて ください。

/ 宮アビケーションビュー × / 画オフシェクト一覧 × / 画 Customer × / 画 Invoice × / 画 Product × / 画 InvoiceSPerCustomer ×	· ·
これらのオブジェクトではパターンの使用が可能です(下線が引かれているのはパターン適用済みです)	
Category 🔽 Work With	
☑ 保存時にこのパターンの適用(▲)	
🖃 👿 Work With Pattern Instance	*
Transaction (Customer)	
🛓 📗 Level (Customer)	
– عن DescriptionAttribute (CustomerName)	E
🔤 🗟 Selection (Work With 顧客s)	
- 😭 Ins: default, Upd: default, Del: default, Dis: default	
e 🖪 Attributes	
- Att. CustomerID	
- Att. CustomerName	
- <i>Att</i> . CustomerAddress	
Latt CustomerEmail	
⊕ <u>G</u> Orders	
□ <del>2</del> ↓ Order (顧客名)	
<i>A</i> t: CustomerName	
Filter	
□ □ I Attributes	<b>T</b>
🔋 Structure 🛯 🚝 Web Form 📴 🚝 Win Form 🗟 Rules 🕱 Events 🚯 Variables 🛷 Help 📓 Documentation 📲 Patterns	

#### 図: Customer(顧客)トランザクション

- このパターンのインス タンスにアクションを 追加し、グリッドの外 にボタンを配置してア クションを結びつけま す。アクションは 「Invoices per Customer (顧客ごと の請求書)」ウェブパ ネルを呼び出しましょ う。
- 3. 「Selection(Work With 顧客 s)」オプシ ョンを右クリックし て、「Add」→ 「Actions」オプショ ンを選択してくださ い。



図: Work with 顧客 s



- Action を右クリックし 「Add」→「Action」 オプションを選択し、 新しいアクション追加 してください。
- F4 を押して、プロパ ティを編集してくださ い。

#### Name に

「ViewInvoicesPer Customer」を入力して、 Caption には「顧客毎の請 求書」を入力してくださ い。

GXObject では InvoicesPerCustomer ウェ ブパネルを指定します。

- 変更を保存してください。
- F5を押して、アプリ ケーションを実行して ください。



#### 図:パターンインスタンスの変更

. 136.2.						
(ターン:	-		タイプ:	*ALL		-
ヮテゴリ:	・ *すべて ・		フォルダ	フォルダ: *すべて		
」更新され トプジェク	<sup>れた</sup> トをマッチングして	ています		_		
そのタリに1 	ルマ ( )が 類 9 る /cl 名前	のに、列をここにドラツクし( 1 タイプ	くにさい。	デスクリプション	日付の変更	
GetTab	Image	Procedure	Ge	t Tab Image Name	2007/06/29 6:58	
Gx0010		Web Panel	Sel	ection List 請求書	2009/01/11 17:	ſ
		Web Panel	Sel	ection List Product	2009/01/11 23:	
Gx0021						
Gx0021 Gx0030		Web Panel	Sel	ection List 顧客	2009/01/11 17:	1
Gx0021 Gx0030 Gx0040		Web Panel Web Panel	Sel Sel	ection List 顧客 ection List 製品	2009/01/11 17: 2009/01/11 23:	
Gx0021 Gx0030 Gx0040 Home		Web Panel Web Panel Web Panel	Sel Sel Hor	ection List 顧客 ection List 製品 me	2009/01/11 17: 2009/01/11 23: 2007/06/29 6:58	
Gx0021 Gx0030 Gx0040 Home Invoice		Web Panel Web Panel Web Panel Transaction	Sel Sel Hor 諸习	ection List 顧客 ection List 製品 me 求書	2009/01/11 17: 2009/01/11 23: 2007/06/29 6:58 2009/01/11 23:	
Gx0021 Gx0030 Gx0040 Home Invoice	General	Web Panel Web Panel Web Panel Transaction Web Component	Sel Sel Hor 諸3 Inv	ection List 顧客 ection List 製品 me 校書 pice General	2009/01/11 17: 2009/01/11 23: 2007/06/29 6:58 2009/01/11 23: 2009/01/11 23:	
Gx0021 Gx0030 Gx0040 Home Invoice	General PerCustomer	Web Panel Web Panel Web Panel Transaction Web Component Web Panel	Sel Sel Hor 諸功 Invi	ection List 顧客 ection List 製品 me 校書 oice General 冬毎の請求書	2009/01/11 17: 2009/01/11 23: 2007/06/29 6:58 2009/01/11 23: 2009/01/11 23: 2009/01/12 0:19	
Gx0021 Gx0030 Gx0040 Home Invoice Invoice	General PerCustomer ProductWC	Veb Panel Veb Panel Veb Panel Transaction Veb Component Veb Panel Veb Component	Sel Sel Hor 諸3 Invo 107	ection List 顧客 ection List 製品 me 校書 oice General 答冊 の請求書 oice Product WC	2009/01/11 17: 2009/01/11 23: 2007/06/29 6:58 2009/01/11 23: 2009/01/11 23: 2009/01/12 0:19 2009/01/11 23:	
Gx0021 Gx0030 Gx0040 Home Invoice Invoice Invoice Invoice	General PerCustomer ProductWC rized	Veb Panel Veb Panel Veb Panel Transaction Veb Component Veb Panel Veb Component Procedure	Sel Sel Hor Inv Inv Inv Is A	ection List 顧客 ection List 製品 me 校書 oice General 客冊 の請求書 oice Product WC Authorized	2009/01/11 17: 2009/01/11 23: 2007/06/29 6:58 2009/01/11 23: 2009/01/11 23: 2009/01/12 0:19 2009/01/11 23: 2007/06/29 6:58	

図:オブジェクトの選択



 実行されると、作成さ れたウェブパネルを呼 び出すボタンが追加さ れた Work With 顧客 を選択してください。

🥭 Work Wi	ith 顧客s - Windows	Internet Explorer			_ 🗆 🔀
<b>GO</b> -	🙋 http://localhost/De	mo.NetEnvironment/wwc.	Google		<b>P</b> -
😭 🏟	🏉 Work With 顧客s			i 🟠 •	5 · 🖶 · *
Applie	cation Head	der			×
Recents:	Work With 顧客s				
Work 顧客名	With 顧客s				
			(+)		
	顧客名 顧客住所	顧客電子メール			
🔀 🗙	<u>顧客01</u> 東京	support@genexus.jp			
🔀 🗙	<u>顧客02</u> 千葉	genexus@genexus.jp			
顧客毎の諸家	「「「」「」「」「」「」「」「」」				
		Footer Info	D		
					~
ページが表示さ	\$れました		📃 😔 อหาวมหาว	ントラネット	🔍 100% 🔻 💡

図:アプリケーションの実行

パート6:アプリケーションの改良

推定された時間:15分

ユーザコントロール

製品の売上チャートを表示させましょう。

この要件を満たすために、GxChart というユーザーコントロールを使用しましょう。



#### ステップ20:ユーザーコントロールの定義

1. SalesxProd と呼ばれる Web Panel 型のオブジェクトを新規 作成してください。デスクリプ ションには「製品売り上げ」と 入力してください。

カテゴリの選択		タイプの選択				
All Busines Web Workflo	esIntellige w entation	Business Process Diagram Data Provider Data Selector Data View Diagram Document External Object Mage	Language     Master Page     Mester Page     Procedure     Query     Subtype Group     Theme     Transaction	🔡 Web Co 💽 Web Pa		
< III.	•	< [				
名前:	SalesxP	rod				
デスクリプション:	製品売り	FQ				
	·	ts 🔹				

#### 図:SalesxProd ウェブパネル

 Navigation Wew 
 ③ GeneXuxX 
 □ Product 
 ③オガジュカトの管理 
 □ InvoicePerCustome 
 ③ SalessProd 
 ・ x 
 SUP-16/522
 □ D+D-A
 ③ A-D
 ③ SalessProd 
 ・ x
 SUP-16/522
 □ D+D-A
 ③ A-D
 ④ A-D
 ④ A-D
 ④ A-D
 ⑤
 □ D+D
 ④
 □ A-D
 ③
 □ A-D
 □ D+D
 □
 □ A-D
 □ A-D
 □ D+D
 □
 □ A-D
 □ A-D
 □ D+D
 □
 □ A-D
 □ A-D
 □ A-D
 □ D+D
 □
 □ A-D
 2. 「表示」メニューから「他のツ ールウィンドウ」のツールボッ クスを表示させてください。 花気気 Category 1 Categ Title on X HIME

Design

3. ウェブパネルに「GxChart」ユ ーザコントロールをドラッグし てください。

図:GxChart コントロールを配置した WebForm エレメント

🕼 Web Form 🔹 🗟 Rules 👻 Events 💙 Conditions 🕼 Variables 🥔 Help 👔 Documentation

株野48
 グループ

グルーフ
 項目器性皮数
 別ッド
 フリースか(ルグリッド
 Web エンボーネンド

■ 組み込みページ E 13-ビューア

- 2-92/10-# Take Salley

4. Events エレメント内に自動生 成されたコードは削除して、次 のコードを記述してください。



図: Events エレメント



#### Event Start

GxChartControl1.Title = 'ProductSales'

For Each

&GxChartData.Categories.Add(ProductName)

&GxChartSerie.Values.Add(InvoiceSubTotal)

EndFor

&GxChartData.Series.Add(&GxChartSerie)

EndEvent

- 5. ウェブパネルを保存してくださ い。
- F5 を押して、アプリケーションを実行してください。



図:アプリケーションの実行

#### おめでとうございます!

GeneXus で最初のアプリケーションの作成に成功しました。 次の段階では、GeneXus のもう一層深い機能を紹介していきます。



# パート7:プロシージャー



推定された時間:15分

#### インタラクティブではないプロセスの設計(プロシージャー)

	プロシージャー
	これまでに、データベースにデータの追加、更新及び削除するためにユーザーのアクションを必要とするトラ
	ンザクションオブジェクトを作成してきました。
	しかしながら、ユーザーのアクションに頼らずに自動的に処理を実行するケースが必ずあります。その場合に
	は、別の GeneXus オブジェクトを使用します:
	• <b>プロシージャー:</b> プロシージャーは、インタラクティブではないデータベースの読み取り及び更新を行
-	う処理を定義するオブジェクトです。 プロシージャーは、関数及びサブルーチンを定義するために使わ
	れます。
	GeneXus プロシージャーの主な機能定義は、トランザクションオブジェクトを定義した時と同様にナレッジ
	ベースに基づく方法論が使われます。
	プロシージャーオブジェクトの定差け、頂日尾性の値が保友される物理データベースのテーブルレイアウトに
	基づいているのではなく、ナレッシベースに登録された項目属性の名前に基づきます。
	したがって、プロシージャー内に記述される項目属性がナレッジベースにある限りプロシージャーの定義は有
	効であり続けます。

高レベルのプロセス:プロシージャーは、非常に高度な手法で記述されます。実行順序は、設計者が話		プロシージャーの特徴
<ul> <li>述するコントロール、印刷、テータへのアクセス及び他のコマンドを含む単純なプロセス言語によって 決定されます。</li> <li>ナレッジベースに基づく:プロシージャーのソースに書く項目属性は、物理データベースのテーブルを 直接参照するのではなくナレッジベース内に定義されている項目属性名を参照します:         <ul> <li>項目属性の物理的な場所は GeneXus が推論します。</li> <li>GeneXus は、物理データベースのテーブル間の関連を保持しています。</li> <li>式項目属性は、GeneXus により自動的に推論・計算されます。</li> </ul> </li> </ul>	ļ	<ul> <li>高レベルのプロセス:プロシージャーは、非常に高度な手法で記述されます。実行順序は、設計者が記述するコントロール、印刷、データへのアクセス及び他のコマンドを含む単純なプロセス言語によって決定されます。</li> <li>ナレッジベースに基づく:プロシージャーのソースに書く項目属性は、物理データベースのテーブルを 直接参照するのではなくナレッジベース内に定義されている項目属性名を参照します:         <ul> <li>項目属性の物理的な場所は GeneXus が推論します。</li> <li>GeneXus は、物理データベースのテーブル間の関連を保持しています。</li> <li>式項目属性は、GeneXus により自動的に推論・計算されます。</li> </ul> </li> </ul>

合計と共に全ての顧客一覧を印刷するプロシージャーと製品を10%値下げするプロシージャーを作成しましょう。



#### ステップ21:印刷するプロシージャーの定義

- オブジェクトフォルダを右クリック して「新規」→「オブジェクト」を 選択してください。
- 作成する「Procedure」型オブジ エクトを選択してください。
- 名前には"Customer\_Report"を入 カしてください。そしてデスクリプ ションに、 "顧客一覧"と入力して ください。
- 4. 作成をクリックしてください



- 図: Customer\_Report プロシージャー
- 1. 以下のソースを Source エレメントに記述 してください:

#### For each

&Total=SUM(InvoiceTotal) Print printBlock1 Endfor

 Layout エレメントへ移動して、ツールバ ーのメニューから「挿入」→「項目属 性」から CustomerName(顧客名)の項 目属性を挿入してください。

1 🗆 Fo	or each
2	<pre>&amp;Total=SUM(InvoiceTotal)</pre>
3	Print printBlock1
4 – Er	ndfor
5	

#### 図: Customer\_Report プロシージャーの Source エレメント

e ≡ printBlock // ∄ ∄	J-Juby           by-Jr           JJJ1Jh4/J           JJJ1Jh		*
	項日周1日を選択 その列に従って分類す	るために、列をここにドラ	ッグしてください。
	名前	タイプ	デスクリプション
40	CustomerAddress	Character(50)	顧客住所
40	CustomerEmail	Character(50)	顧客電子メール
At	CustomerId	Numeric(4.0)	顧客番号
An	CustomerName	Character(20)	顧客名
At	InvoiceDate	Date	請求日
At	Invoiceld	Numeric(4.0)	請求書番号
At	InvoiceSubtotal	Numeric(8.2)	小計
At	InvoiceTax	Numeric(8.2)	税
At	InvoiceTotal	Numeric(8.2)	合計
At	LineQuantity	Numeric(4.0)	数量
	LineTotal	Numeric(8.2)	行の合計金額
At	ProductId	Numeric(4.0)	製品番号
力 At	ProductName	Character(20)	製品名
E Build	ProductPrice	Numeric(8.2)	単価
		ana baana ahaa ahaa ahaa ahaa ahaa ahaa	

図:プロシージャのレイアウト



ツールバーのメニューから「挿入」→「変数」を選択し、変数の挿入画面を呼び出した後、画面左下にある「新規変数」ボタンを押下してください。

新規変数ウィンドウが起動しますの で、以下のプロパティを設定してくだ さい。

Name : Total Length : 15 Decimals : 2

プロパティ設定後、「OK」ボタンを押下 し、変数を定義してください。

4. ツールボックスを使用してレポートを設計

の挿入)。

することができます。

述してください:

を上方へ移動してください。

することができます(ラベル、ライン等

例えば、PrintBlock1 を右クリックし「プ

リントブロックを挿入」で、もう一つの

PrintBlock を挿入して「顧客名」及び 「合計」の見出しをツールボックスの TextBlock コントロールを使用して挿入

それから、マウスで右クリックして表示 されるオプションを使用して PrintBlock2

右記のコードを Source エレメントに記

新規変数			×
🧱 🛃 フィルタ			
- 変数: &Total			~
Name	Total		
Description	Total		
Column title	Total		
· · · · · · · · · · · · · · · · · · ·			Help
<ul> <li>Type Definition</li> </ul>			
Based on	(none)		
Data Type	Numeric		=
Length	15		
Decimals	2		
Signed	False		
Collection	False		
Dimensions	Scalar		
Initial value			
+ Validation			
+ Picture			
<ul> <li>Control Info</li> </ul>			
ControlType	Edit		
InputType	Values		
Notify Context Cha	ang False		
<ul> <li>Appearance</li> </ul>			-
		ОК	キャンセル

図:変数の挿入



図:プロシージャーの Layout エレメント

🚡 Custon	nerReport 🔚 Navigation View
1 =	Header
2	Print printBlock2
3	End

図:プロシージャーの Source エレメントへの追記

5. プロシージャーを保存してください。



6. 右に表示されるプロシージャー自体のプロ パティ内にある「Main Program」プロ パティを TRUE に設定し、「Call Protocol」を HTTP に設定してくださ い。

7. ファイル名と拡張子を指定するために、

し、保存してください。

Rules エレメントに以下のコードを記述

Output\_file('Customers.PDF','PDF');

19109X-3189-3
 19
 19109X-3189-3
 19
 19109X-3189-3
 19
 19109X
 19109X

#### 図:プロシージャーのプロパティ



8. 顧客ごとに合計を印刷するこのプロシージ ャーを呼び出すために、Invoice トラン ザクションにボタンを挿入してくださ い。

まず Invoice(請求書)トランザクション オブジェクトを開きます。

Web Form エレメントを選択し、ツール ボックスを開いてボタンコントロールを ドラッグしてください。

#### 図:プロシージャーの Rules エレメント

<ul> <li>フォーム</li> </ul>						Þ
0000	) 🖶 🔗 🗙					^
請求書						
Errorviewer:	ctlError					
諸求書番号 Invoice	d					
請求書日 Invoicel	Date					
Custon	hend					
Product						
製品番号	製品名	単価	数量	行金額		
ProductId	ProductName	ProductPrice	LineQuantity	LineTotal		
//計 Invoices	Subto					
税 Invoice	Гах					
Invoice	Total					
<b>*</b> f7 <b>*</b> 7 <b>*</b>	e an					
ubise						
- 	Farm III // Win Farm	B Dulas 🖨 Duasta 🖉	A Ussiahlar 🕼 Usla 🛙	Desumentation 1 💕 Dates	una I	¥

図:Invoice(請求書)トランザクション



9. 挿入されたボタンを右クリックして、プロ パティを編集してください。

「OnClickEvent」には「Print」と入力 し、Caption に「印刷」を入力してくだ さい。ボタンを右クリックして「イベン トへ行く」を選択してください。

ボタン: Button1	
ControlName	Button1
Class	SpecialButtons
OnClickEvent	'print'
Caption	印刷
Tooltiptext	ED剧
- Appearance	
Туре	Text
+ Font	Microsoft San
ForeColor	Window T 🗸
BackColor	ButtonFace
BorderWidth	2
BorderColor	ButtonHig
BorderStyle	Rounded

図:ボタンのプロパティー

Application Header					
Recents: <u>2011/02/02</u> Work	With	<u>請求書</u> 』	<b>请</b> 求書	а а 	
<ul> <li>□ </li> <li< th=""><th></th><th></th><th></th><th></th><th></th></li<></ul>					
劉品番号	3	製品名	単価	日侍	行金額
<b>x</b> 1	Û	製品01	100.00	2	200.00
<b>x</b> 2	2 🗘	製品02	200.00	4	800.00
0	Ŷ		0.00	0	0.00
0	Û		0.00	0	0.00
0	Ŷ		0.00	0	0.00
0	Ŷ		0.00	0	0.00
0	Ŷ		0.00	0	0.00
		[行)	追加]		
小計 1000.00 税 50.00 合計 1050.00					

図:トランザクションの実行

10. Events エレメントには以下のコードを 追加してください:

Event 'Print'

Customer\_Report.Call () EndEvent



11.	保存し、F5 を押下して実行してくださ	顧客名	合計
	い。	顧客01 顧客02	62500.00 12500.00
	3.カネわた新しい機能を試すために 新		

入力された新しい機能を試すために、新 しい請求書を登録してください。

図:プロシージャーの実行



#### ステップ22:アップデートプロシージャーの実施

製品の価格を10%値下げするプロシージャーを作成しましょう。

 「名前:Discount\_of\_prices; デスクリプション:価格値下げ」Procedure型のオブジェクトを 新規作成してください。デスクリプションには 「価格値下げ」と入力してください。



2. Source エレメントに以下のコードを記述してください:

#### For each

ProductPrice = ProductPrice-ProductPrice\*0.10

Endfor



図:プロシージャーの Source エレメント

- 3. プロシージャーを保存してください。
- このプロシージャー を呼びだすために、ウェブパネ ルを新規作成してください。

ウェブパネルの名前に「Control\_Panel」を入力し てください。デスクリプションには「制御画面」と 入力してください。

カテゴリの選択		タイプの選択		
All BusinessIntellige Web Workflow Documentation		Business Process Diagram     Language       Data Provider     Master Page       Data Selector     Moster Page       Data View     Moury       Diagram     Structured Data Type       Document     Subtype Group       External Object     Transaction		⊞ Web Co ∎ Web Par
< III	•	×		+
治則; 	Control_	Panel		
ナスクリノンヨン:	制御画面	0		
フォルダ:	Ohiects		-	

図: Control\_Panel ウェブパネル



5. Web Form エレメントでツールボックスからボタン を挿入してください。

SalessProd S. CustomerReport III Invoice S. Discount of prices	Control Panel * * X (-9-8.#952
I PTT=/ [CRetton]	□ = 32/HD+Jk
	- 3 (1-9
実行(確定させる)	An ラキストブロック
	<ul> <li>パイパーリンク</li> </ul>
	🛄 Table
	日 セジョン
	- (* <b>T</b> is
	回 項目開始/求款
	田 グリッド
	12 フリースタイルグリッド
	■ web 12/オーネント
ž.	福祉法3ページ
2	17-61-7
uli	0 2-9°2/10-#
	- 📾 Chart
	a all ImaneGallery
Web Form * 🔠 Rules 🕲 Events 🔻 Conditions 🚱 Variables 🧐 Heip	Documentation

#### 図:Web From エレメント

6. 挿入されたボタンを右クリックしてプロパティを編 集してください。

キャプションに「価格を 10%値下げ」と入力してく ださい。



 ボタンを右クリックして、「イベントへ行く」オプ ションでイベントへ移動してください。

プロシージャーをイベントから呼びましょう:

Event Enter

Discount\_of\_prices.call()

EndEvent

8. ウェブパネルを保存し、F5 を押して実行してくださ

い。



SalesxProd 👼 CustomerReport 🛅 Invoice 👼 Discount\_of\_prices 🗃 Control\_Panel \*

図:アプリケーションの実行

# パート8:マルチプラットフォーム



推定された時間:5分

Recents: 制御画面 価格を10%値下げ



GeneXus はミッションクリティカルなアプリケーションを、複数のプラットホームで生成することが可能です。

ー旦アプリケーションが一つの環境で稼働した後、別の環境で同じアプリケーションを稼働させるには、新しい環境を定義す ることにより、簡単に対応することが可能です。

注: GeneXus トライアルバージョンは、GeneXus .NET ジェネレータのみを提供します。 GeneXus の正規版は市場の主要なプラットホームをサポートしています。GeneXus がサポートする技術の一覧は次の URL でご覧ください http://www.genexus.com/technologies。

#### 要約

このチュートリアルで GeneXus の主な機能と利点を経験することができたのではないかと思います。

#### 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全てのトランザクションに対応する最適のデータモデル(第三正規形)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を設定して(ウェブ、プログラミング言語、DBMS、その他)、GeneXus は自動的に指定された DBMS サーバに物理の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は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が実感できたと思われます。

#### インタラクティブでないプロセスの設計

アプリケーションのデータベース構造を意識しないで良いプロシージャの全体的なイメージを提供できたと思います。

#### 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。



#### FAQ



はい、GeneXus は、アプリケーションのデータベースとソースコードを設計、生成及び保守します。

いいえ、GeneXus は選択されたプラットホームでネイティブコードを生成するため独自のランタイムは必要ありません。

質問	既存のデータベースを使用することは可能ですか?
----	-------------------------

はい、可能です。GeneXus は、既存のデータベース構造を読み取る Data Base Reverse Engineering Tool(DBRET) という ツールを持っています。

質問	生成されたソースコードを変更することは可能ですか?

はい、可能ですが推奨できません。理由は大きく分けて二つあります:再度 GeneXus でソースを生成すると、ソースは上書 きされてしまいます。また、独自のソースを書き込むことによりアプリケーションがその言語に依存してしまいます。

もし既存のコードを使用されるのならば、GeneXus の「アプリケーションの統合」オプションを使用して外部機能とのイン ターフェースをとります。生成されたコードに直接手を加える必要はありません。



はい、可能です。GeneXus は様々なコントロールの統合を可能にしています。

詳細は以下のリンクをご覧ください。

http://marketplace.genexus.com

http://wiki.genexus.jp/UC



# **質問** 拡張テーブルの概念は何ですか?

ベーステーブルがあれば、その拡張テーブルとは直接あるいは間接的にベーステーブルの(参照)キーを使って持ってくることが出来る項目属性のセットです:

- ベーステーブルに属している項目属性。
- ベーステーブルと「N対1」の関係で直接あるいは間接的に関連がある全てのテーブルに属している項目属性。

使用例:

- データベース構造が変更されても拡長テーブルの概念により GeneXus のレポートとプロシージャが有効であり続けることが可能です。
- 拡長テーブルの概念は、トランザクションの Structure から参照されているベーステーブルの拡張テーブルに属している項目属性の登録、更新及び削除をトランザクションオブジェクトが行うのにも使用されます。

あるテーブルの拡張テーブルを見るためには、ナレッジベースのデータモデルのバッハマン図を使いましょう:

- 1. オブジェクトの新規作成で「Diagram」型のオブジェクトを作成してください。
- 2. 表示されたテーブルを選択して、ダイアグラムをドラッグしてください。



#### 図3:データモデルのバッハマン図

この例において、プロジェクトで各テーブルの拡張テーブルを特定することができます:

ベーステーブル	拡張テーブル
Customer	Customer
Invoice	Invoice, Customer
InvoiceProduct	InvoiceProduct, Invoice, Customer



質問 "For Each"コマンドとは何ですか?	
--------------------------	--

"For Each"コマンドとは、GeneXus で帳票や処理を記述する時に使う手続き型言語の重要なコマンドです。

このコマンドを使用して、データを読み取ったり更新したりするオンライン処理ができます。

"For Each"コマンドでアクセスするデータを決めて、必要な項目属性を記述します。

GeneXus は、生成時に使用するテーブルを自動的に推論します。

データベース構造が変わっても「For Each」の定義は有効であり続けます。

"For Each"の基本的な構文は以下の通りです:

for each

print < something >

endfor

各「For Each」コマンドは、ナビゲートするデータモデルのテーブルを必ず持っています。

ナビゲートするテーブルは、**"For Each"のベーステーブル**及びその**拡張テーブル**(コマンドに含まれる項目属性を取得する ために必要とするテーブル)になります。

アクセスされるテーブルは、"For Each"コマンドの中に含まれる項目属性により決定されます。

• "For each"内の項目属性に対してGeneXusは最小限の拡張テーブルを推論します。

"For each"内の項目属性に対してのベーステーブルは"For Each"のベーステーブルであります。



# コンタクトとリソース

#### GeneXus コミュニティ

GeneXus コミュニティは質問に対する回答をしたり、問題に対する解決策を出したり、経験の共有をする場所を提供したり しています。 以下のサイトで提供されているコミュニティリソースの一覧を見ることができます: http://www.genexus.com/community/(英語)

#### 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、誰でもオンラインで利用できます。
   しかしながら、 アクセスするデータは GXtechnical のユーザロールによって異なります(2種類のロールがあります: 一般ユーザもしくは顧客)。
- 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又はサポートチームとのインタラクション(情報交換)ができます。

   <a href="http://www.genexus.com/support/">http://www.genexus.com/support/</a> (英語) をご覧ください。

#### 購入方法

GeneXus Technologies は、世界的な総代理店ネットワークを通して販売されています。

<u>http://www.genexus.com/distributors</u> で最も近い総代理店を検索してください。あるいは、sales@genexus.jp にお問い 合わせください。