

Primeros pasos con GeneXus™

Cree su primera aplicación sin saber codificar.

© GeneXus. Todos los derechos reservados. Este documento no puede ser reproducido por ningún medio sin la autorización expresa de GeneXus™. La información contenida en este documento es para uso personal únicamente.

Marca registrada:

GeneXus es una marca comercial o una marca registrada de GeneXus S.A. Todas las demás marcas comerciales mencionadas en este documento son propiedad de sus respectivos dueños.

ÍNDICE

INTRODUCCIÓN	3
COMENZAR A UTILIZAR GENEXUS	3
CREAR UNA NUEVA APLICACIÓN CON GENEXUS	4
DEFINIR LOS PRIMEROS OBJETOS	5
GENERAR Y EJECUTAR LA APLICACIÓN POR PRIMERA VEZ	18
LA APLICACIÓN CRECE	28
AGREGAR REGLAS DE NEGOCIOS	32
DEFINIR CÁLCULOS: FÓRMULAS	34
USO DE PATRONES (PARA WEB Y PARA DISPOSITIVOS MÓVILES)	37
¿QUÉ SUCEDE SI QUIERE GENERAR LO QUE HA DEFINIDO HASTA AHORA EN OTRO LENGUAJE O PARA OTRA BASE DE DATOS?	61
¿QUÉ MÁS OFRECE GENEXUS?	63
PRÓXIMOS PASOS	66

INTRODUCCIÓN

GeneXus es una suite de desarrollo de código de bajo nivel que permite la generación rápida de aplicaciones de software en múltiples lenguajes y plataformas. GeneXus ofrece varias ventajas: es fácil de aprender, altamente productivo, multiplataforma y preparado para el futuro, de forma que protege sus activos digitales y simplifica la adopción de nuevas tecnologías.

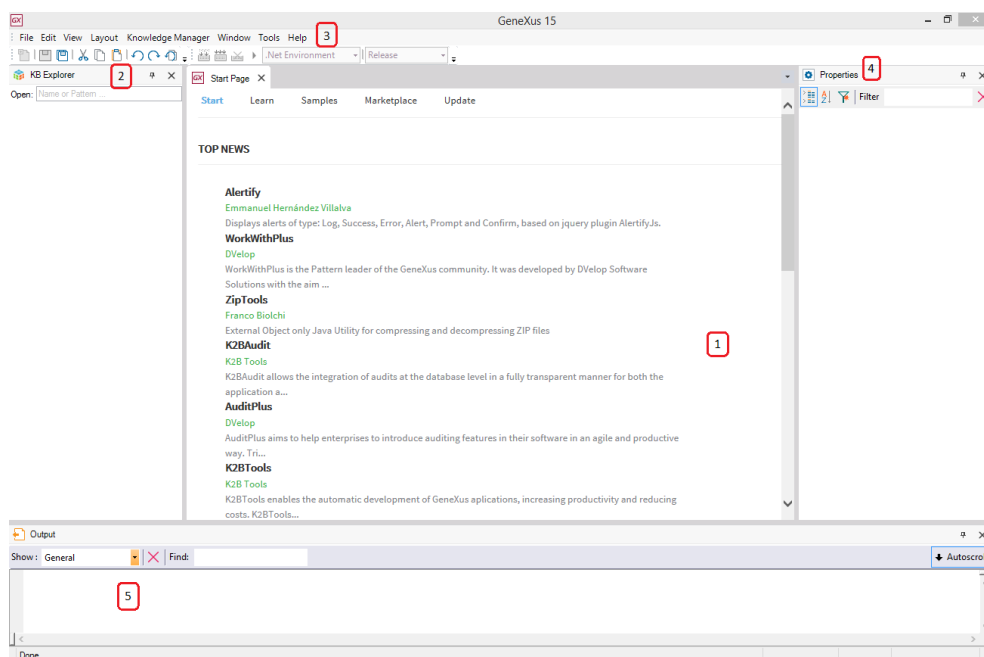
Por ejemplo, GeneXus genera aplicaciones para la Web y / o dispositivos móviles e inteligentes (desde un reloj pulsera, un celular, una tableta o una TV), para la plataforma de destino seleccionada por el desarrollador (cierto lenguaje, base de datos, entorno, plataforma, con diseño web responsivo, etc.).

Este documento es una guía para principiantes para desarrollar aplicaciones con GeneXus.

COMENZAR A UTILIZAR GENEXUS

Al abrir GeneXus, verá una interfaz similar a la siguiente que se conoce como IDE (Entorno de desarrollo integrado). Esta interfaz es fácil de usar y puede ser parametrizada por cada desarrollador.

Se compone de diferentes ventanas:

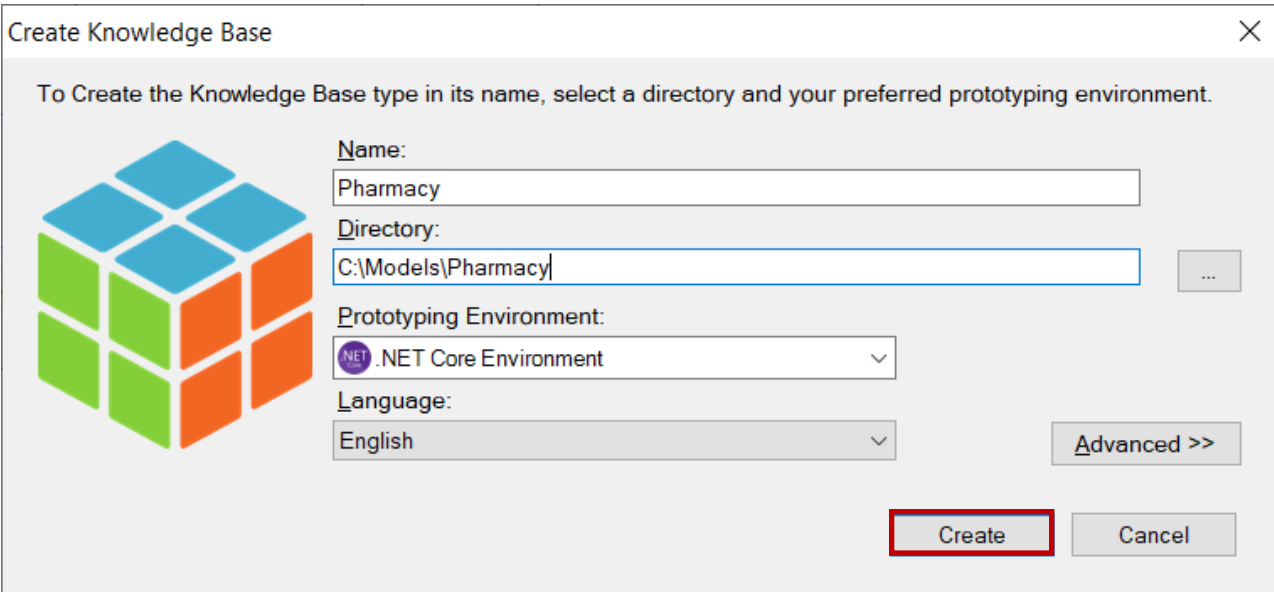


1. **Ventana principal (Página de inicio):** muestra dinámicamente información técnica sobre la herramienta y la Comunidad GeneXus (noticias y soluciones publicadas por otros desarrolladores). También muestra los proyectos que se abrieron recientemente y ofrece la posibilidad de crear un nuevo proyecto.
2. **KB Explorer:** muestra objetos y configuraciones del proyecto abierto actualmente.
3. **Barra de herramientas:** muestra una interfaz fácil de usar para las funciones más utilizadas en GeneXus.
4. **Ventana de propiedades:** muestra las propiedades asociadas con el contexto en el cual desarrollador se encuentra posicionado (como un objeto seleccionado, atributo, variable, control, etc.).
5. **Salida:** muestra la salida de las acciones realizadas.

CREAR UNA NUEVA APLICACIÓN CON GENEXUS

Para comenzar a desarrollar una nueva aplicación con GeneXus, debe crear una nueva Base de Conocimiento (una Base de Conocimiento es un proyecto GeneXus).

Al seleccionar **File > New > Knowledge Base** en la barra de herramientas, se mostrará el siguiente cuadro de diálogo:



Create Knowledge Base

To Create the Knowledge Base type in its name, select a directory and your preferred prototyping environment.

Name:
Pharmacy

Directory:
C:\Models\Pharmacy

Prototyping Environment:
.NET Core Environment

Language:
English

Advanced >>

Create Cancel

La aplicación que se definirá a lo largo de este documento es una aplicación real pero simplificada para una farmacia. Por lo tanto, tiene sentido que la Base de Conocimiento se llame

"Farmacia" (o "Pharmacy" o "PharmacySystem", entre otras opciones). A continuación, debe ingresar la ruta donde desea crear esa Base de Conocimiento.

También debe seleccionar uno de los lenguajes de programación disponibles en el combo **Prototyping Environment**. GeneXus utilizará el lenguaje seleccionado para generar la aplicación, así como los programas necesarios para crear y mantener la base de datos. El lenguaje seleccionado en este ejemplo es **.NET Core**. Más adelante, habrá que ingresar la información relativa a la base de datos.

El combo Language le permite seleccionar el idioma en el que desea que GeneXus genere los títulos de botones que se crean automáticamente, etiquetas, mensajes para los usuarios, etc. El idioma predeterminado es inglés.

Al presionar el botón Create, GeneXus inicia el proceso de creación de la Base de Conocimiento.

DEFINICIÓN DE LOS PRIMEROS OBJETOS

Una vez que haya creado una nueva Base de Conocimiento, deberá describir las visiones de los usuarios. Para esto, es necesario identificar objetos de la vida real (recomendamos prestar atención a los sustantivos que los usuarios mencionan en sus descripciones, tales como: productos, facturas, clientes, etc.) y comenzar a definirlos utilizando **objetos** GeneXus.

Los desarrolladores GeneXus no trabajan en tareas de bajo nivel, como definir tablas, normalizar, diseñar programas, programación, y similares. En cambio, su trabajo es una actividad de alto nivel que implica describir la realidad de los usuarios. Después de eso, GeneXus analiza los objetos definidos y pasa a diseñar la base de datos y los programas de aplicación para la plataforma seleccionada de forma **totalmente automática**.

Suponga que la farmacia que solicita la aplicación desea poder registrar los Productos que tiene a la venta.

Para describir cada objeto de la vida real identificado, debe crear un objeto GeneXus de tipo Transacción (no está relacionado con el concepto que se refiere a transacciones de la base de datos). Para crear un objeto Transacción, seleccione **File > New > Object** en la barra de herramientas. A continuación, se le mostrará el siguiente cuadro de diálogo que le permite seleccionar el tipo de objeto que desea crear (en este caso: Transacción) e ingresar un nombre para él (en este caso: Product).

New Object

Select a Category:

- Data Management
- User Interface
- BPM
- Chatbots
- Resources
- Documentation
- Extensibility
- Deploy
- Reporting
- Test
- ALL

Select a Type:

- API
- Data Provider
- Data Selector
- Data View
- Domain
- Procedure
- Structured Data Type
- Subtype Group
- Transaction** ←

Describes an object or actor of reality, defining the structure of the database, business rules, and the UI for data manipulation.

Name:

Description:

Module/Folder:

Al hacer clic en el botón Create, se crea la Transacción Product y se mantiene abierta para comenzar a definir su estructura:

Name	Type	Description	Formula	Nullable
Product	Product	Product		No

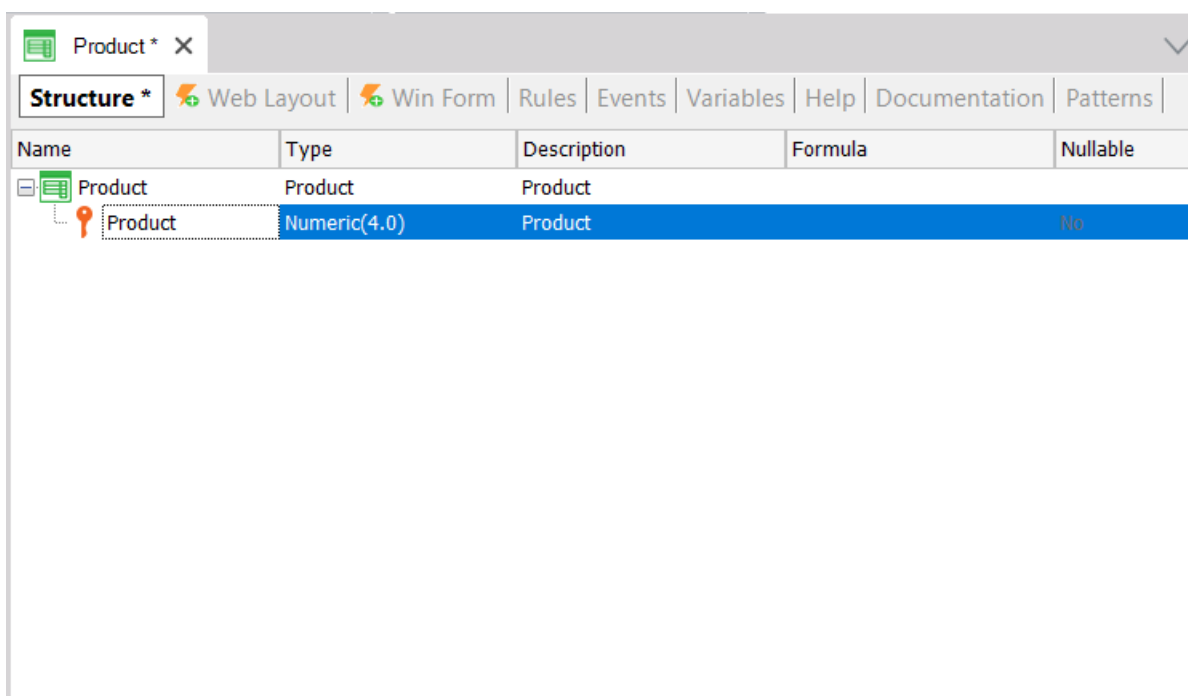
Cada Transacción tiene algunas secciones que se explicarán gradualmente. Concretamente, la estructura (Structure) de una Transacción permite definir los atributos o campos que describen un objeto de la vida real.

Suponga que en la farmacia le transmitieron que para cada producto necesitan registrar su código, nombre, precio de venta, stock y tipo (medicamento, cosmético, etc.). Por lo tanto, estos datos que deben ser registrados para cada producto, coinciden con los atributos que deben crearse para esta Transacción.

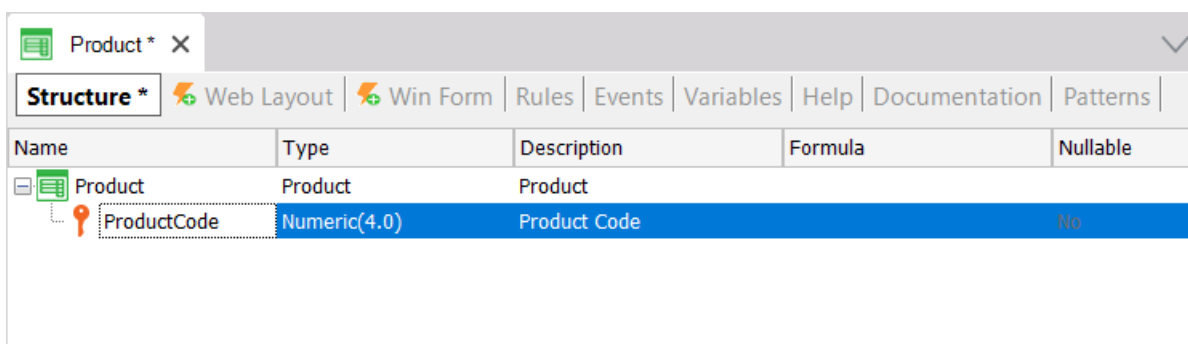
Observe en la imagen de arriba, que la primera línea en la estructura de la Transacción automáticamente queda lista para que defina el primer atributo. Además, observe que un ícono con la imagen de una llave está asociado con esta línea. Esto se debe a que, en cada Transacción, un atributo, o conjunto de atributos, debe configurarse con la función de identificador único o llave primaria.

El concepto de identificador único o llave primaria tiene como objetivo identificar de forma única a cada producto (o cualquier objeto de la realidad). En otras palabras, los usuarios finales de la aplicación no podrán ingresar dos productos con el mismo valor identificador. Claramente, el atributo llave primaria de la Transacción Product es el código del producto. A continuación, se muestra cómo definirlo.

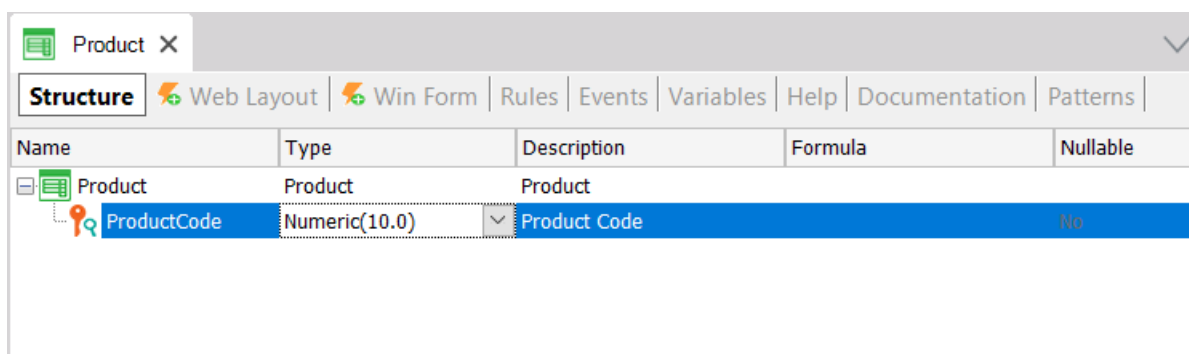
Si presiona la tecla de punto en el teclado, GeneXus mostrará automáticamente el nombre de la Transacción como prefijo en el nombre del atributo:



y solo le resta escribir Code después del prefijo Product:

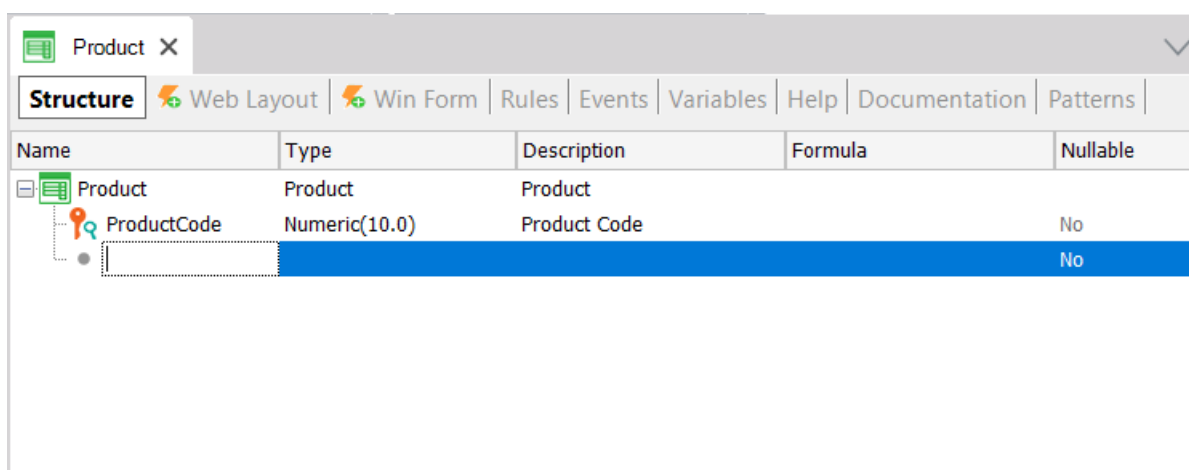


Luego, al presionar la tecla Tab, puede elegir el tipo de datos que se almacenará para este atributo. El tipo de datos predeterminado es: Numérico de 4 dígitos sin decimales. Sin embargo, la farmacia solicitó que el código del producto siempre debe tener un valor numérico de hasta 10 dígitos, por lo que hay que cambiar su longitud a 10:



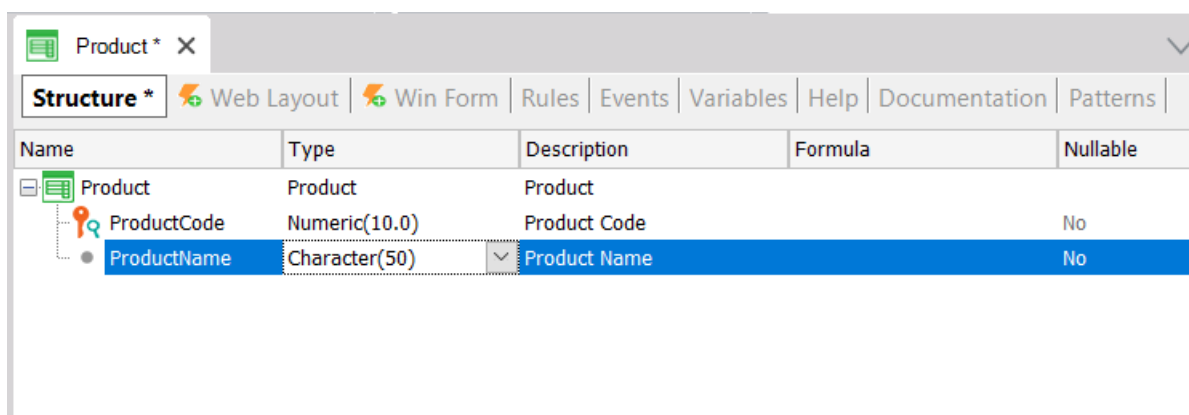
Name	Type	Description	Formula	Nullable
Product	Product	Product		
ProductCode	Numeric(10.0)	Product Code		No

Al presionar Enter, se abre una nueva línea donde puede comenzar a definir el segundo atributo:



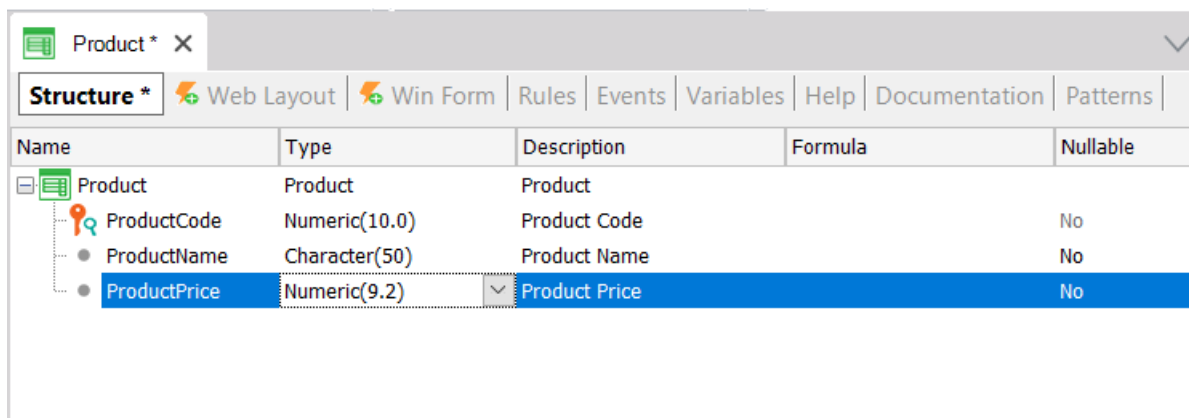
Name	Type	Description	Formula	Nullable
Product	Product	Product		
ProductCode	Numeric(10.0)	Product Code		No
				No

Nuevamente, debe presionar la tecla de punto en el teclado y completar el nombre del atributo con Name, es decir, *ProductName* (del tipo Character y longitud de 50):



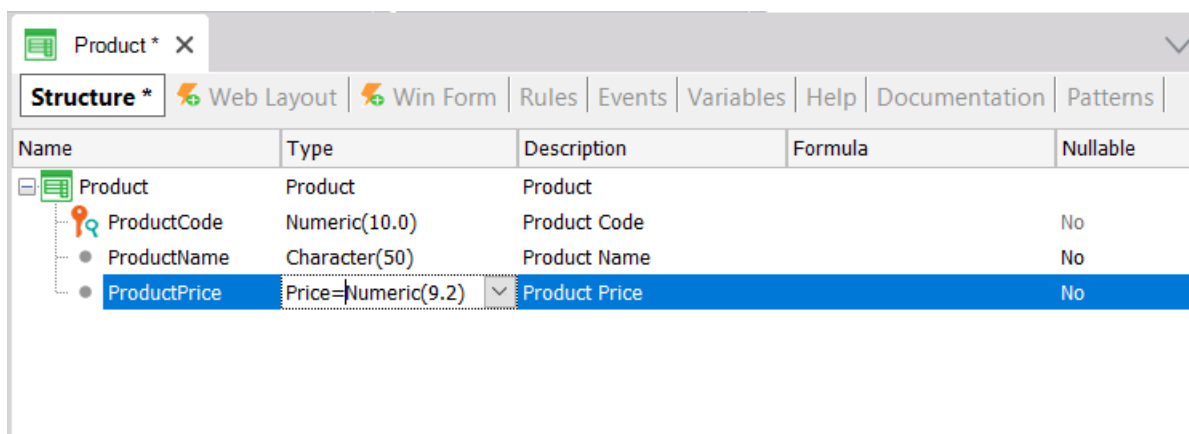
Name	Type	Description	Formula	Nullable
Product	Product	Product		
ProductCode	Numeric(10.0)	Product Code		No
ProductName	Character(50)	Product Name		No

Ahora debe agregar el atributo *ProductPrice* (de tipo Numeric, con 9 dígitos y 2 decimales):



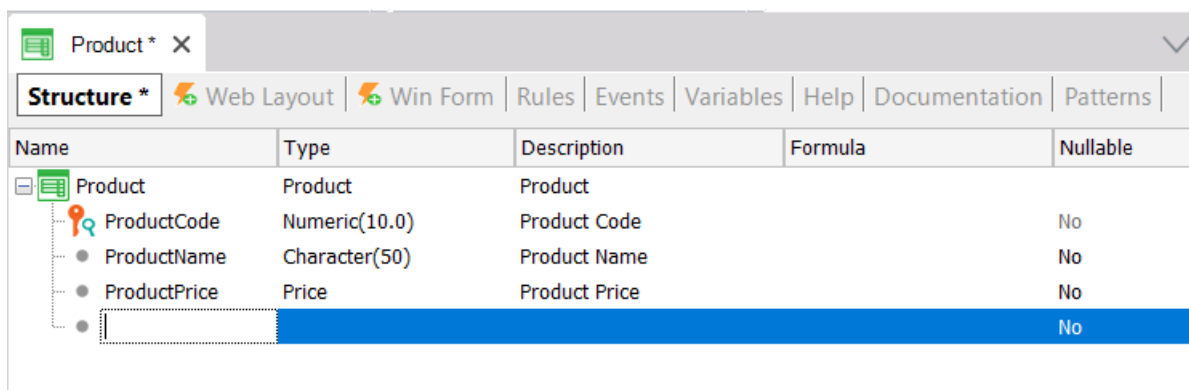
Name	Type	Description	Formula	Nullable
Product	Product	Product		
ProductCode	Numeric(10.0)	Product Code		No
ProductName	Character(50)	Product Name		No
ProductPrice	Numeric(9.2)	Product Price		No

Como probablemente necesitará crear más atributos para definir precios o cantidades (es decir, cuando la farmacia compra o vende productos), podría ser una buena idea crear un tipo de definición genérica para todos los precios. Para hacer esto, en la columna Tipo, solo tiene que escribir: "Price =", antes del tipo que tiene seleccionado:



Name	Type	Description	Formula	Nullable
Product	Product	Product		
ProductCode	Numeric(10.0)	Product Code		No
ProductName	Character(50)	Product Name		No
ProductPrice	Price=Numeric(9.2)	Product Price		No

Luego presione Enter, y verá que el atributo *ProductPrice* se ha establecido como de tipo *Precio* (Price):



Name	Type	Description	Formula	Nullable
Product	Product	Product		
ProductCode	Numeric(10.0)	Product Code		No
ProductName	Character(50)	Product Name		No
ProductPrice	Price	Product Price		No
				No

La definición de *Precio* con tipo numérico (9 dígitos con 2 decimales) se llama **Dominio**.

Los dominios permiten hacer definiciones genéricas. Una de las ventajas que proporcionan los dominios es que, si más adelante necesita que los precios sean numéricos de una longitud diferente, solo deberá cambiar la definición del dominio para actualizar todos los atributos basados en ese dominio en un solo paso.

Para ver los dominios creados en la Base de Conocimiento, seleccione **View > Domains** en la Barra de herramientas:

The screenshot shows the GeneXus interface with the 'Domains' window open. The 'Domains' window displays a list of domains with columns for Name, Type, Module, and Description. The 'Price' domain is selected, showing a Type of 'Numeric(9.2)' and a Module of 'Root Module'. The 'Properties' window on the right shows the configuration for the 'Price' domain, including its Name, Description, Empty as null status, Class, Module, Qualified Name, Object Visibility, and Type Definition (Based on: none, Data Type: Numeric, Length: 9, Decimals: 2, Signed: False, Enum Values, Collection: False, Autonumber: False, Dimensions: Scalar, Initial value).

Name	Type	Module	Description
CryptoSignAlgorithm	Character(40)	GeneXus	Crypto Algorithm Sign
TrnMode	Character(3)	GeneXus	TrnMode
Address	VarChar(1K)	GeneXus	Address
Component	Url, GeneXus	GeneXus	Component
Email	VarChar(100)	GeneXus	Email
Geolocation	Character(50)	GeneXus	Geolocation
Html	LongVarChar(2M)	GeneXus	Html
Phone	Character(20)	GeneXus	Phone
APIAuthorizationStatus	Numeric(1.0)	GeneXus	APIAuthorization Status
MessageTypes	Numeric(2.0)	GeneXus	Message Types
ProgressIndicatorType	Numeric(1.0)	GeneXus	Progress Indicator Type
RecentLinksOptions	Numeric(4.0)	GeneXus	Recent Links Options
ObjectName	VarChar(256)	GeneXus	Object Name
CallTargetSize	Character(10)	GeneXus	Call Target Size
EventExecution	Numeric(1.0)	GeneXus	Event Execution
PushNotificationPriority	Character(20)	GeneXus	Push Notification Priority
SmartDeviceType	Numeric(1.0)	GeneXus	Smart Device Type
CameraAPIQuality	Numeric(1.0)	GeneXus	Camera API Quality
AudioAPISessionType	Numeric(1.0)	GeneXus	Audio APISession Type
MediaDuration	Numeric(12.0)	GeneXus	Media Duration
PlaybackState	Numeric(4.0)	GeneXus	Playback State
NetworkAPIConnectionType	Numeric(1.0)	GeneXus	Network API Connection Type
EventAction	Numeric(4.0)	GeneXus	Event Action
EventStatus	Numeric(4.0)	GeneXus	Event Status
EventData	LongVarChar(2M)	GeneXus	Event Data
EventErrors	LongVarChar(2M)	GeneXus	Event Errors
ApplicationState	Numeric(1.0)	GeneXus	Application State
SynchronizationReceiveResult	Numeric(4.0)	GeneXus	Synchronization Receive Result
RegionState	Numeric(1.0)	GeneXus	Region State
BeaconProximity	Numeric(1.0)	GeneXus	Beacon Proximity
MediaFinishReason	Numeric(4.0)	GeneXus	Media Finish Reason
HttpMethod	Character(7)	GeneXus	Http Method
HttpAuthenticationType	Numeric(4.0)	GeneXus	Http Authentication Type
CommonCallTarget	Character(20)	GeneXus	Common Call Target
Price	Numeric(9.2)	Root Module	Price

Name	Price
Description	Price
Empty as null	Yes
Class	Attribute
Module	Root Module
Qualified Name	Price
Object Visibility	Public

Type Definition	
Based on	(none)
Data Type	Numeric
Length	9
Decimals	2
Signed	False
Enum Values	
Collection	False
Autonumber	False
Dimensions	Scalar
Initial value	

Como puede observar en la imagen, GeneXus crea algunos dominios automáticamente. Al hacer clic en un determinado dominio, la ventana de propiedades se actualiza mostrando las propiedades del dominio seleccionado. Tenga en cuenta que, además de configurar el tipo de datos para un dominio, también puede definir otras propiedades interesantes.

La ventana de Propiedades se actualizará cada vez que seleccione otro atributo, dominio, objeto, etc., con las correspondientes propiedades disponibles para su configuración en cada caso. Si no está visible, puede presionar F4 para abrirlo.

Regrese a la Transacción del *Producto*, donde el siguiente atributo que debe definir es *ProductStock* de tipo numérico, y longitud 4:

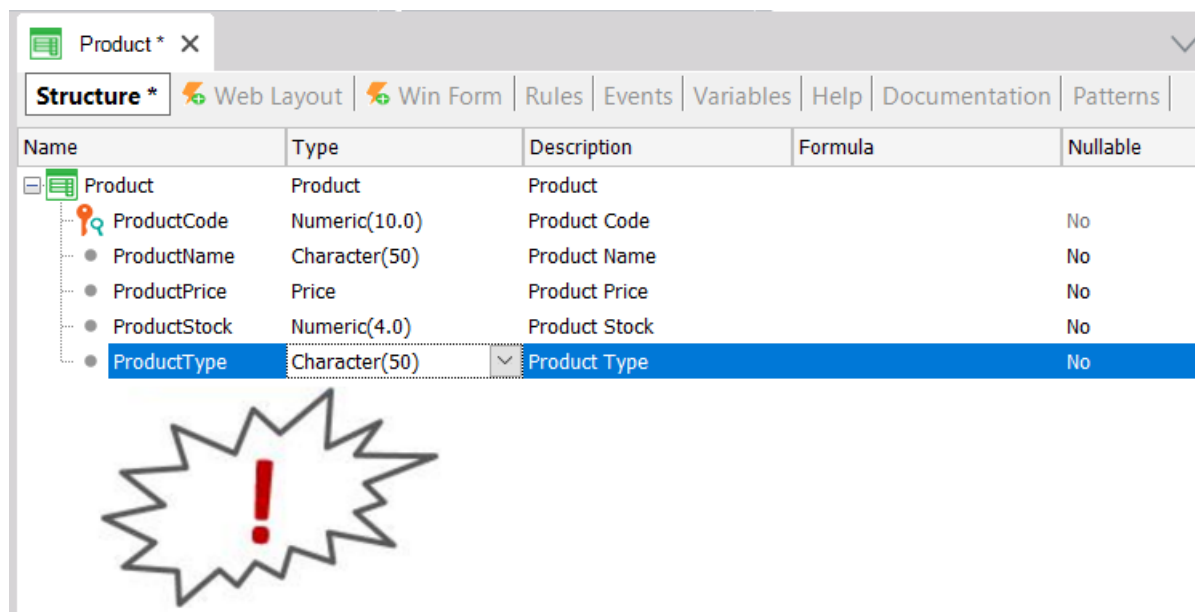
Name	Type	Description	Formula	Nullable
Product	Product	Product		
ProductCode	Numeric(10.0)	Product Code		No
ProductName	Character(50)	Product Name		No
ProductPrice	Price	Product Price		No
ProductStock	Numeric(4.0)	Product Stock		No

Ahora es necesario registrar el tipo de producto. Podría crear un atributo llamado *ProductType* como *Character(50)* ...

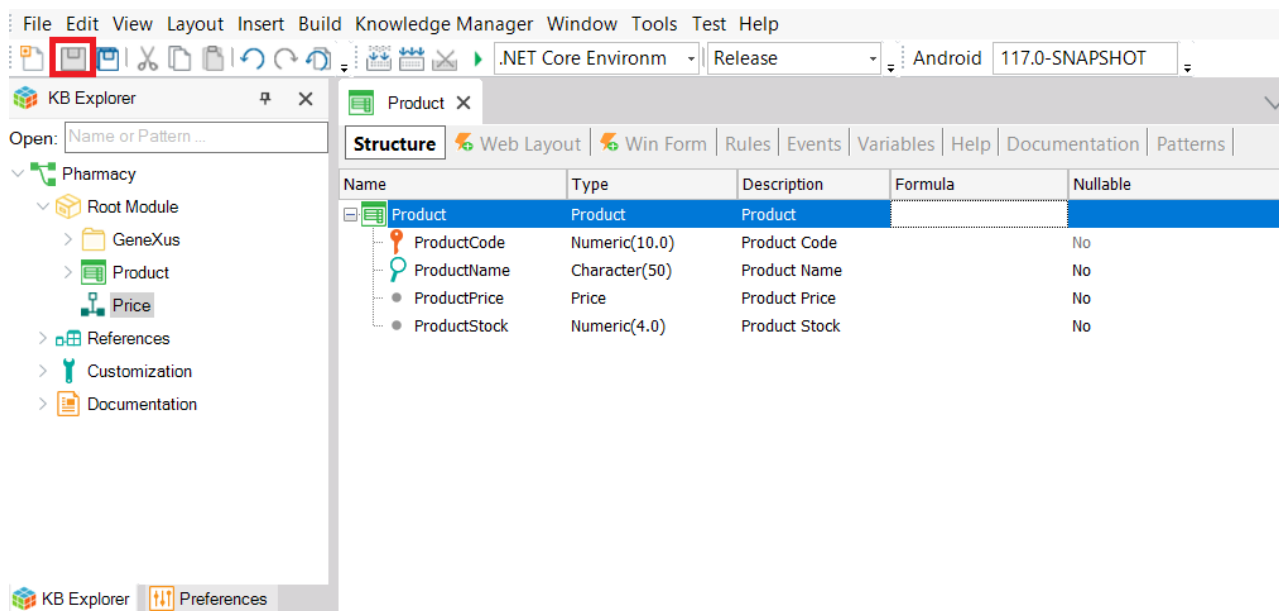
Name	Type	Description	Formula	Nullable
Product	Product	Product		
ProductCode	Numeric(10.0)	Product Code		No
ProductName	Character(50)	Product Name		No
ProductPrice	Price	Product Price		No
ProductStock	Numeric(4.0)	Product Stock		No
ProductType	Character(50)	Product Type		No

Pero, ¿qué pasa si los usuarios quieren ingresar dos productos del mismo tipo? Tendrían que ingresar el mismo nombre de tipo dos veces, ¡teniendo cuidado de escribirlo exactamente igual!

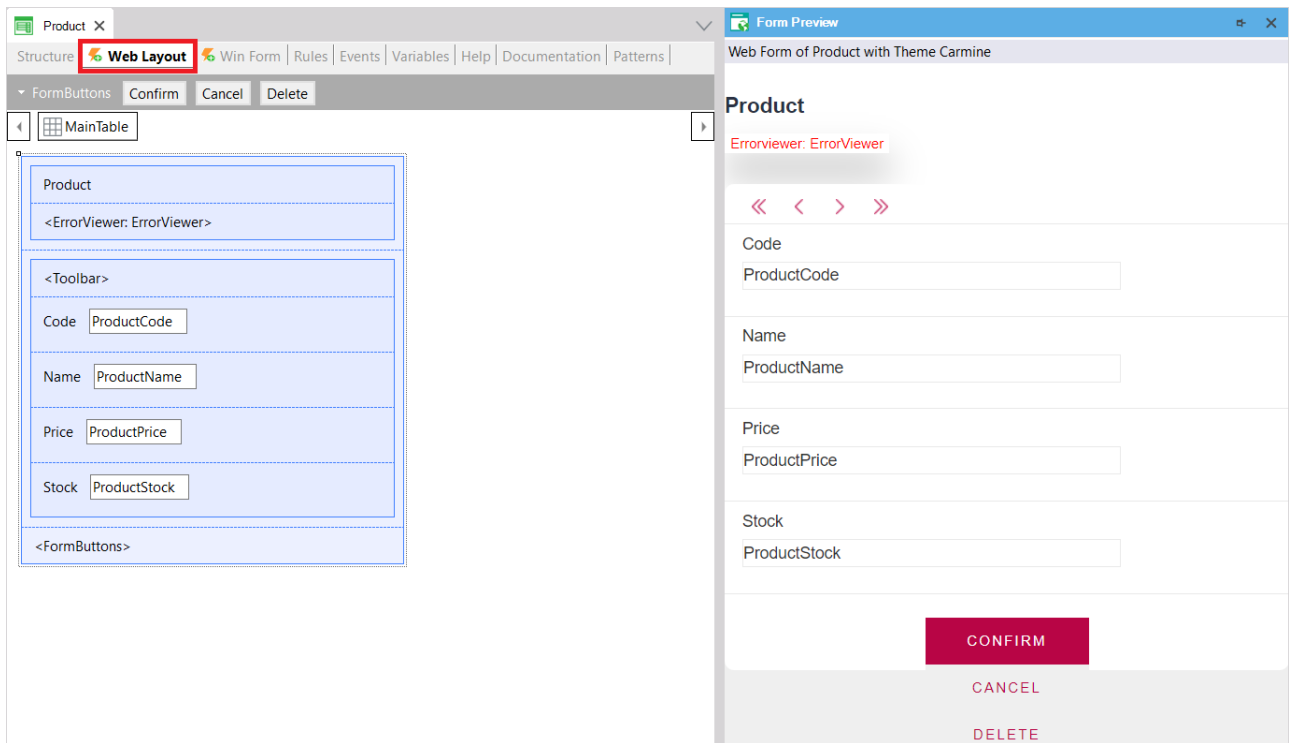
Más adelante, podrían necesitar buscar todos los productos de un cierto tipo y, para obtenerlos, el tipo debe haber sido escrito exactamente igual.



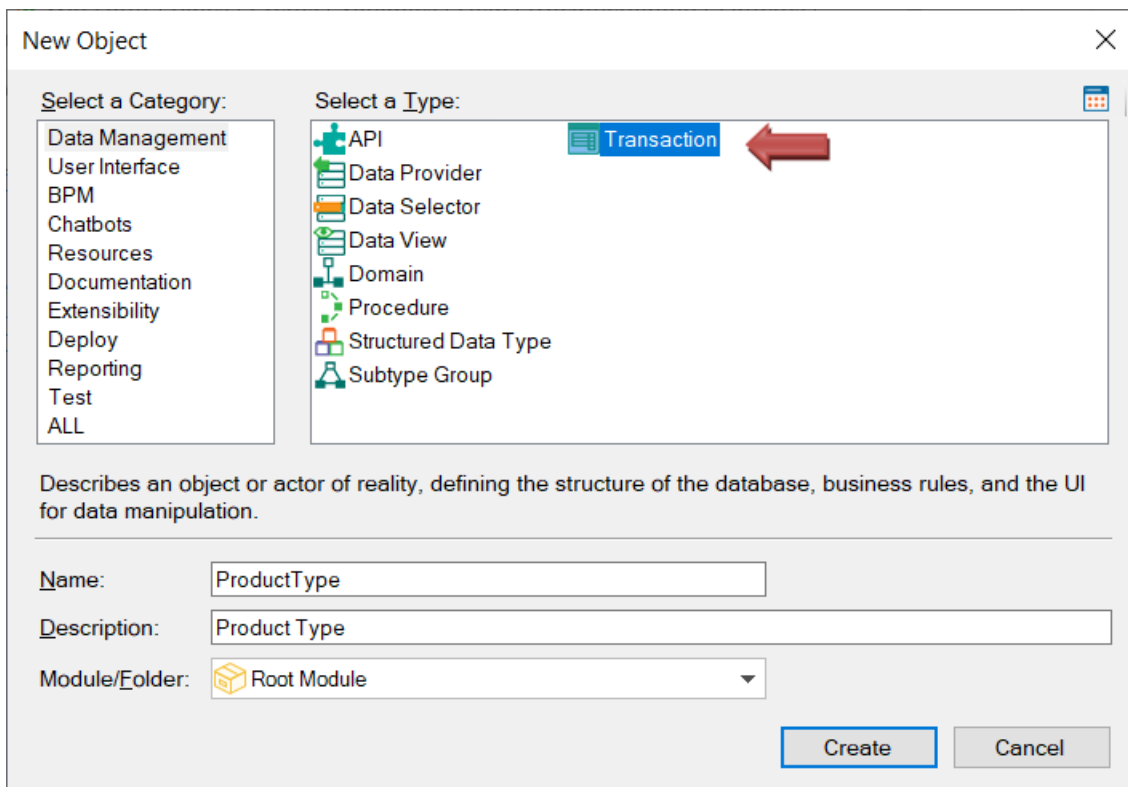
Parece más razonable ingresar el tipo una sola vez, en un solo lugar, y de ahí en adelante para cada producto, asignar el tipo de producto correspondiente. Por lo tanto, hay que eliminar el atributo ProductType de la estructura y guardar la Transacción Product como se muestra:



Para cada Transacción, GeneXus diseña automáticamente una interfaz Web (Web Layout) a partir de la estructura (Structure) definida. Esta interfaz Web permitirá a los usuarios de la aplicación en tiempo de ejecución agregar, actualizar y eliminar productos:



Ahora proceda a crear otra Transacción para registrar los tipos de productos:



Name	Type	Description	Formula	Nullable
ProductType	ProductType	Product Type		No

Para cada tipo de producto hay que registrar su código y nombre de tipo de producto, así que debe crear los dos atributos que se muestran:

Name	Type	Description	Formula	Nullable
ProductType	ProductType	Product Type		No
ProductTypeCode	Numeric(4.0)	Product Type Code		No
ProductTypeName	Character(50)	Product Type Name		No

Recuerde la recomendación de presionar la tecla de punto en el teclado al definir atributos, para que GeneXus escriba automáticamente el nombre de la Transacción como prefijo, y solo haya que completar el final de los nombres de los atributos.

Nombrar atributos con el nombre de transacción como prefijo no solo permite definir atributos de manera más fácil y rápida, sino que también es una convención de la comunidad GeneXus para facilitar la comprensión al leer un nombre de atributo donde sea que esté, para saber a qué objeto se está describiendo.

Observe las propiedades del atributo *ProductTypeCode*:

The screenshot shows the GeneXus IDE interface. On the left, the 'Structure' window displays a table of entities:

Name	Type	Description	Formula	Nullable
ProductType	ProductType	Product Type		
ProductTypeCode	Numeric(4,0)	Product Type Code		No
ProductTypeName	Character(50)	Product Type Name		No

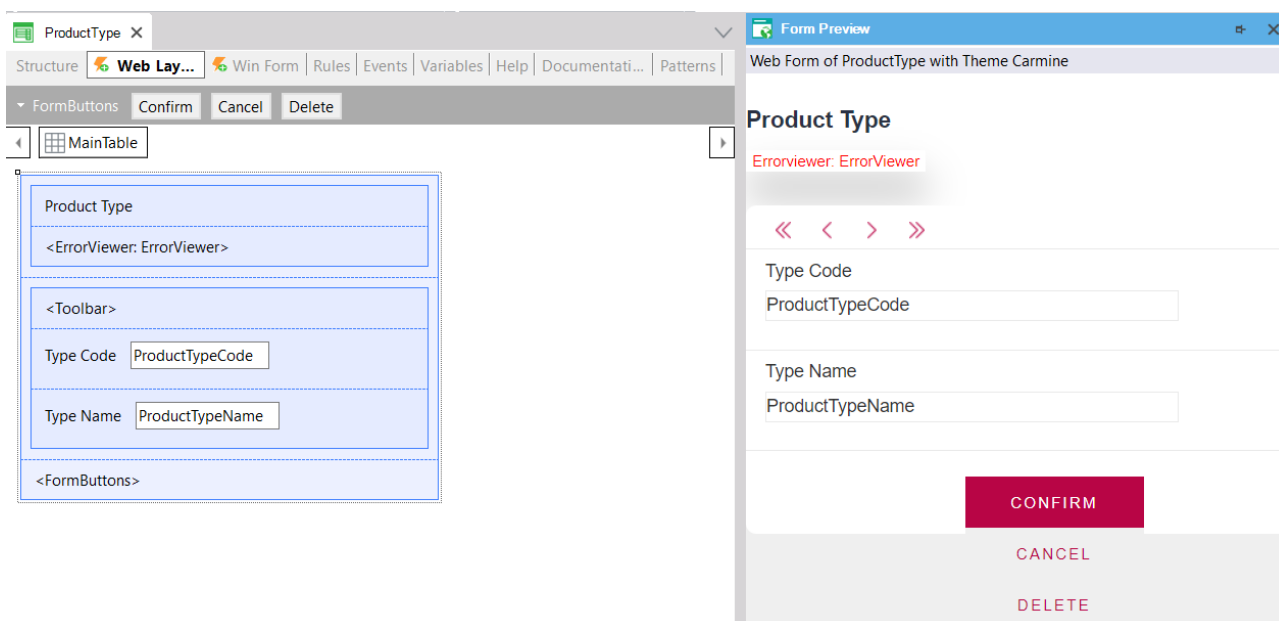
On the right, the 'Properties' window shows the configuration for the 'Attribute: ProductTypeCode'.

Name	ProductTypeCode
Description	Product Type Code
Title	Product Type Code
Column title	Product Type Code
Contextual Title	Type Code
Formula	
Nulls in Forms	Empty as Null
Class	Attribute
Qualified Name	ProductTypeCode
Type Definition	
Supertype	
Based on	(none)
Data Type	Numeric
Length	4
Decimals	0
Signed	False
Autonumber	False
Initial value	

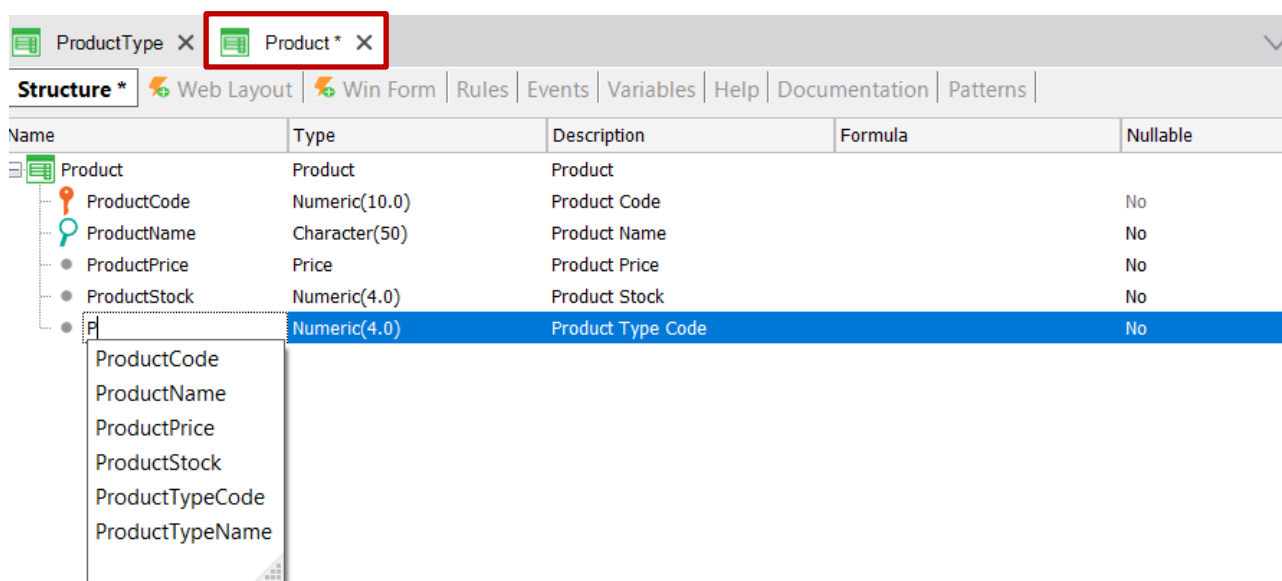
Red arrows point to the 'Nullable' column in the Structure window and the 'Autonumber' property in the Properties window.

Observe la propiedad Autonumber. Está configurada con valor False por defecto, y al cambiarla a True, todos los nuevos tipos de productos ingresados por el usuario final se numerarán automáticamente en forma consecutiva. Por lo tanto, configure la propiedad Autonumber en True para este atributo identificador y guarde la transacción *ProductType*.

Como se explicó anteriormente, cada Transacción tiene una interfaz Web (Web Layout) diseñada automáticamente por GeneXus según su estructura. La siguiente imagen muestra la interfaz Web de *ProductType*:



Ahora, asigne un tipo de producto a cada producto. Para ello, vuelva a la transacción *Product*, agregue una nueva línea en su estructura y escriba la letra "P". Se le mostrará la lista de atributos existente en la Base de Conocimiento que comienza con esa letra:



Seleccione *ProductTypeCode* y se le mostrará la definición completa del atributo.

En esta Transacción también incluya el atributo *ProductTypeName*, porque cuando los usuarios ejecuten esta Transacción y seleccionen un código de tipo de producto, querrán ver el correspondiente nombre del tipo de producto. Concentrémonos en estos dos atributos que se encuentran incluidos en más de una Transacción:

Name	Type	Description	Formula	Nullable
Product	Product	Product		
ProductCode	Numeric(10.0)	Product Code		No
ProductName	Character(50)	Product Name		No
ProductPrice	Price	Product Price		No
ProductStock	Numeric(4.0)	Product Stock		No
ProductTypeCode	Numeric(4.0)	Product Type Code		No
ProductTypeName	Character(50)	Product Type Name		No

ProductTypeCode es el atributo identificador en la Transacción *ProductType* (más específicamente, es la llave primaria de esa Transacción). Entonces, cuando se incluye una llave primaria en otra Transacción, GeneXus entiende que allí el atributo tiene el papel de llave foránea.

Incluir un atributo que es una llave primaria de una Transacción en otra Transacción permite relacionar ambas Transacciones.

GeneXus establece relaciones a través de nombres de atributos, por lo que cuando encuentra atributos con el mismo nombre en diferentes transacciones, asume que se refieren al mismo concepto.

El atributo *ProductTypeName* también está presente en ambas Transacciones. Sin embargo, no está marcado como identificador de ninguna de las Transacciones definidas. Por lo tanto, GeneXus entenderá que es un atributo secundario. GeneXus incluirá *ProductTypeName* en la tabla física *ProductType* que creará en la base de datos y no en la tabla física *Product*.

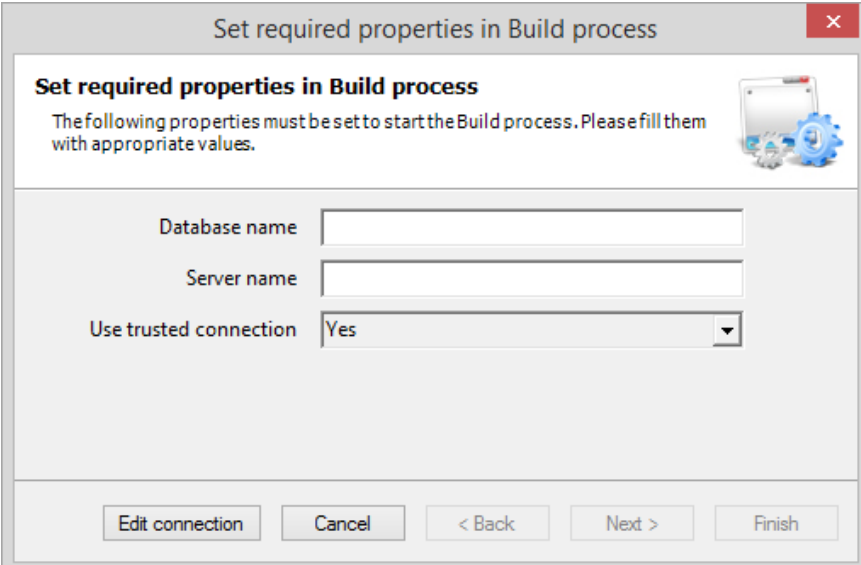
El concepto de Transacción y el concepto de tabla física no son lo mismo. Recuerde que la Transacción es el objeto GeneXus que usted crea en la Base de Conocimiento para representar un objeto de la realidad. Al considerar su estructura y las demás estructuras de Transacciones definidas en la Base de Conocimiento (y también teniendo en cuenta algunas propiedades), GeneXus determinará las tablas físicas que debe crear en la base de datos, así como los atributos que debe almacenar en cada tabla.

En tiempo de ejecución, cuando el usuario ejecute la interfaz Web de la Transacción *Product*, deberá ingresar para el atributo *ProductTypeCode* (que allí es un atributo llave foránea), un valor que haya sido previamente registrado a través de la Transacción *ProductType*. De lo contrario, se

mostrará un error. Al ingresar un *ProductTypeCode* válido, su valor *ProductTypeName* se obtendrá y se mostrará en la pantalla.

GENERAR Y EJECUTAR LA APLICACIÓN POR PRIMERA VEZ

Si desea generar y ejecutar la aplicación por primera vez, solo tiene que presionar F5 y se mostrará el siguiente cuadro de diálogo:



Set required properties in Build process

The following properties must be set to start the Build process. Please fill them with appropriate values.

Database name

Server name

Use trusted connection

GeneXus ofrece la posibilidad de crear un prototipo de la aplicación localmente o en la nube. Existe una propiedad para configurar su elección. De forma predeterminada, la propiedad está configurada para prototipar localmente. Entonces, debe ingresar el nombre que desea para la base de datos de su aplicación y el nombre del servidor de base de datos que tiene instalado:

Set required properties in Build process

Set required properties in Build process
The following properties must be set to start the Build process. Please fill them with appropriate values.

Database name: PharmacyDB

Server name: MYPC\SQLEXPRESS

Use trusted connection: Yes

Al hacer clic en **Finish**, GeneXus evaluará el impacto causado por las nuevas definiciones en la Base de Conocimiento, y mostrará un informe de nombre **Análisis de Impacto (Impact Analysis)**:

Product X ProductType X **Impact Analysis X**

The Database tables will be created.

This report describes how the Database tables will be created.
Please select Create to proceed or Cancel.

Pattern:

Product
 ProductType

ProductType is new

Table Structure

Attribute	Definition	Previous values	Takes value from
ProductTypeCode	Numeric (4), Not null, Autonumber		
ProductTypeName	Character (50), Not null, NLS		

Indexes

Name	Definition	Composition
IPRODUCTTYPE	primary key Clustered	ProductTypeCode

Statements

```
CREATE TABLE [ProductType] (
  [ProductTypeCode] SMALLINT NOT NULL IDENTITY ( 1 , 1 ),
  [ProductTypeName] NCHAR(50) NOT NULL,
  PRIMARY KEY ( [ProductTypeCode] ))
```

0 Errors 0 Warnings 2 Success

Este informe muestra qué cambios estructurales deben realizarse en la base de datos. Observe que, en este caso, el título principal informa que se crearán las tablas de la base de datos (“The Database tables will be created”).

Al hacer clic en cada tabla (Product y ProductType en la ventana izquierda), verá en la ventana de la derecha los atributos que se incluirán en ellos:

The screenshot shows the GeneXus Impact Analysis tool interface. The title bar includes tabs for 'Product', 'ProductType', and 'Impact Analysis'. The main window displays a message: 'The Database tables will be created. This report describes how the Database tables will be created. Please select Create to proceed or Cancel.' Below this message are 'Create' and 'Cancel' buttons. A red arrow points to the 'Product' table in the left pane, which is selected. The right pane shows the 'Table Product specification' for the 'Product' table. The table structure is as follows:

Attribute	Definition	Previous values	Table Name
ProductCode	Numeric (10), Not null, Autonumber		
ProductName	Character (50), Not null, NLS		
ProductPrices	Numeric (9.2), Not null		
ProductStock	Numeric (4), Not null		
ProductTypeCode	Numeric (4), Not null		
ProductPhoto	Image, Not null		
ProductPhoto_GXI	VarChar (2048), Not null		

Below the table structure, there is an 'Indexes' section with a table header: Name, Definition, Composition.

The status bar at the bottom indicates: 0 Errors, 0 Warnings, 2 Success.

The Database tables will be created.

This report describes how the Database tables will be created.
Please select Create to proceed or Cancel.

Create Cancel

Pattern:

Product
 ProductType

Table ProductType specification

Table name: [ProductType](#)

ProductType is new

Table Structure

Attribute	Definition	Previous values	Takes value from
ProductTypeCode	Numeric (4), Not null, Autonumber		
ProductTypeName	Character (50), Not null, NLS		

Indexes

Name	Definition	Composition
IPRODUCTTYPE	primary key Clustered	ProductTypeCode

Statements

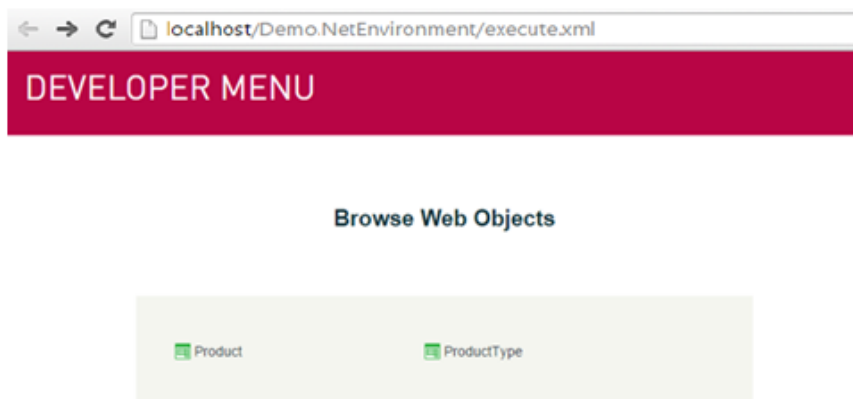
```
CREATE TABLE [ProductType] (
```

0 Errors 0 Warnings 2 Success

Observe que, como se explicó, el atributo *ProductTypeName* no se incluye en la tabla física *Product* que se creará, a pesar de que la incluyó en la estructura de la Transacción *Product* (con el fin de mostrar su valor en la interfaz Web de la Transacción).

Si está de acuerdo con el Análisis de Impacto, puede hacer clic en el botón *Create*, y GeneXus comenzará a crear los programas necesarios para crear la base de datos (aún inexistente), así como las tablas con sus estructuras en esa base de datos. Seguidamente, GeneXus ejecuta esos programas y después de crear la base de datos y las tablas, generará todas las líneas de código necesarias, en el lenguaje de programación seleccionado, para obtener la aplicación que permitirá a los usuarios insertar, actualizar y eliminar tipos de productos y productos.

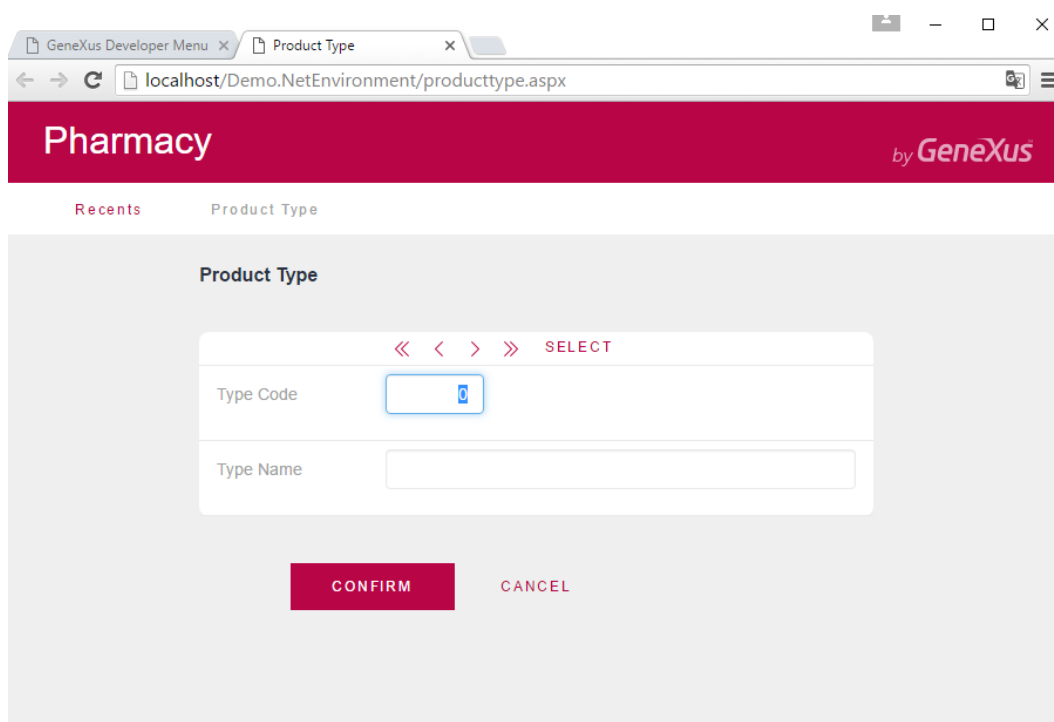
A continuación, se le informará si el resultado fue exitoso o si hubo algún error o advertencia, y verá la aplicación ejecutándose:



El navegador Web se abre por defecto, mostrando una página simple que ofrece una manera rápida de ejecutar los objetos definidos.

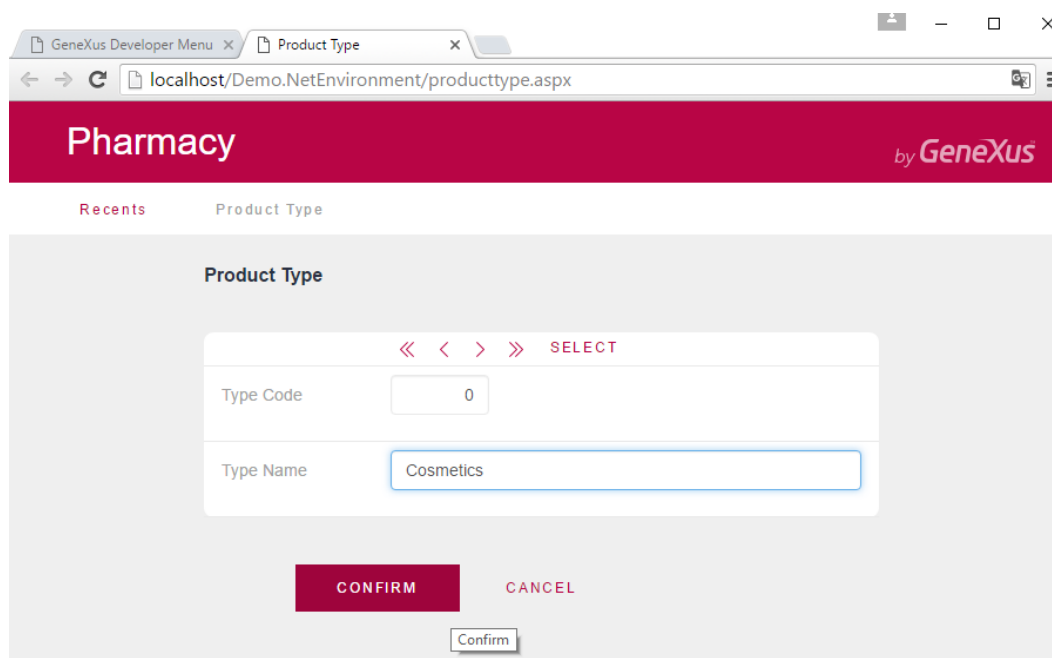
Esta página simple, llamada Developer Menu (Menú del desarrollador), es para desarrolladores, como lo indica su nombre. Obviamente, no es lo que los usuarios verán.

Haga clic con el botón derecho del mouse sobre el enlace *ProductType* y ábralo en una nueva pestaña:



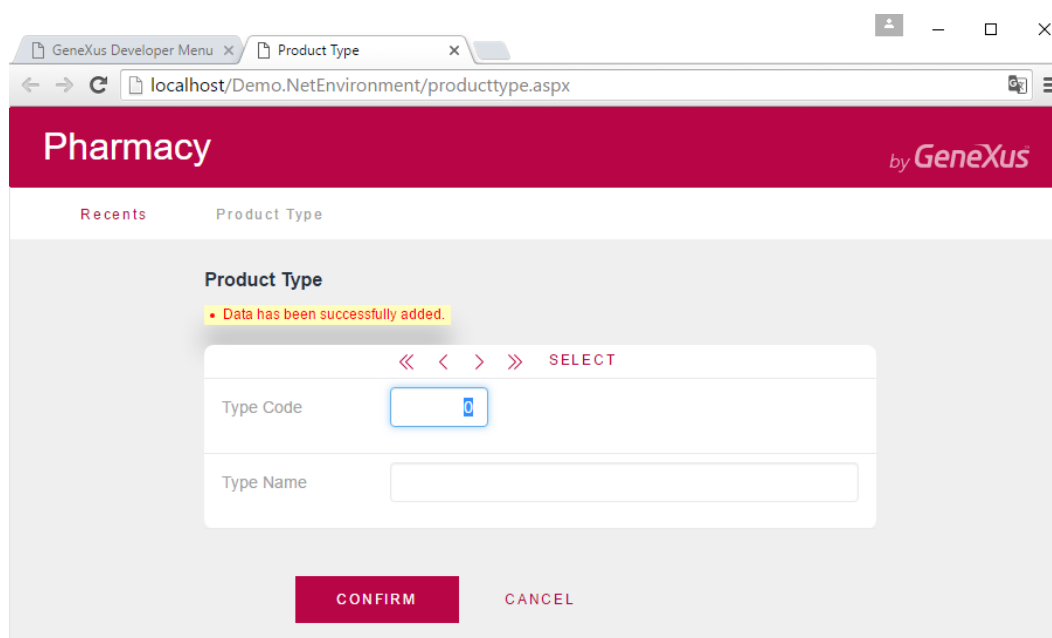
La página que se muestra arriba permite al usuario agregar, actualizar y eliminar tipos de productos. Ingrese el primer *tipo de producto*.

Dado que el atributo *ProductTypeCode* tiene la propiedad *Autonumber* configurada en *True*, no será necesario ingresar un valor para el identificador porque se numerará automáticamente. Por lo tanto, ingrese el nombre del tipo de producto “Cosmetics”:



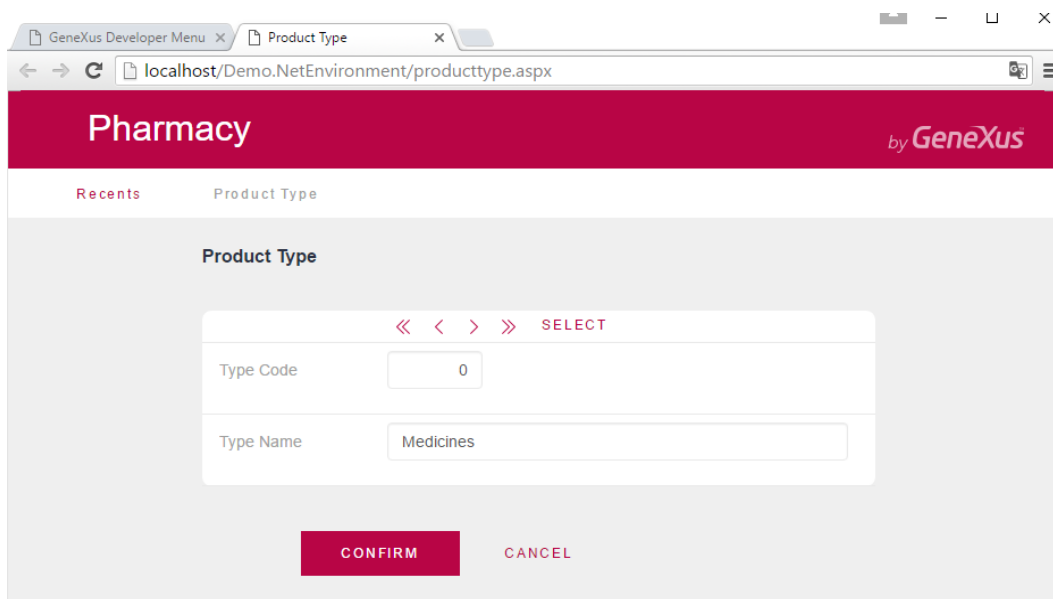
The screenshot shows a web browser window with the URL `localhost/Demo.NetEnvironment/producttype.aspx`. The page has a red header with the text "Pharmacy" and "by GeneXus". Below the header, there are two tabs: "Recents" and "Product Type". The main content area is titled "Product Type" and contains a form with two input fields: "Type Code" (containing "0") and "Type Name" (containing "Cosmetics"). Above the "Type Code" field are navigation arrows and a "SELECT" button. Below the form are two buttons: "CONFIRM" (highlighted in red) and "CANCEL". At the bottom of the form area is a "Confirm" button.

Después de ingresar el nombre del tipo de producto y hacer clic en el botón Confirm, aparecerá un mensaje informando que los datos se agregaron correctamente. A continuación, la página quedará vacía y listo para ingresar otro tipo de producto:



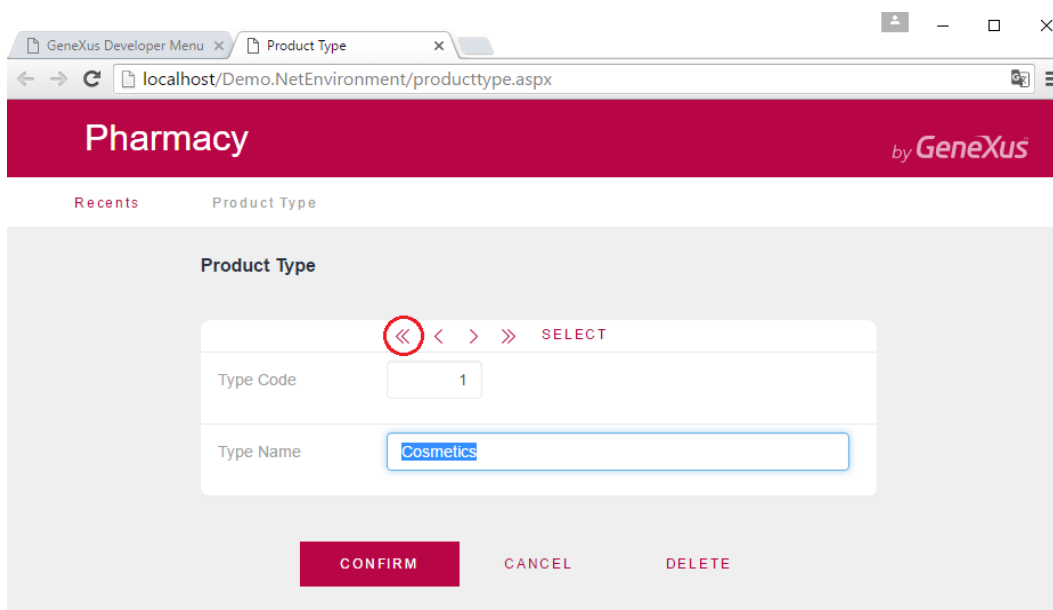
The screenshot shows the same web browser window as the previous one. The page content is now empty except for a yellow message box that says "Data has been successfully added." above the "Product Type" form. The "Type Code" field is empty and the "Type Name" field is empty. The "CONFIRM" button is highlighted in red. The "Recents" and "Product Type" tabs are still visible at the top.

Ingrese el segundo tipo de producto (por ejemplo “Medicines”):

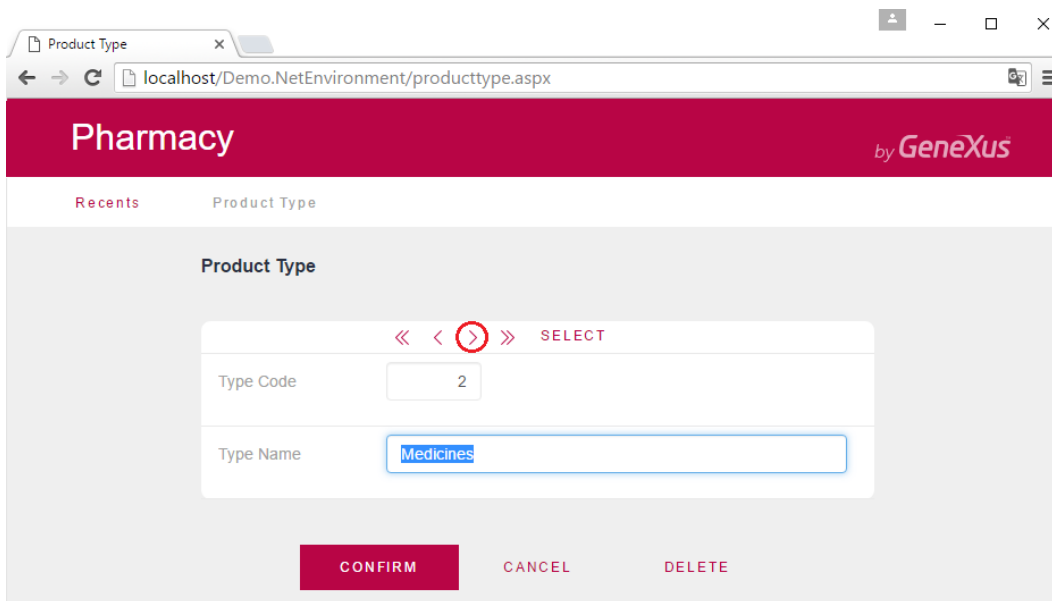


The screenshot shows a web browser window with the URL localhost/Demo.NetEnvironment/producttype.aspx. The page has a red header with "Pharmacy" and "by GeneXus". Below the header, there are two tabs: "Recents" and "Product Type". The "Product Type" form is displayed, featuring a "SELECT" button with navigation arrows. The "Type Code" field contains the value "0" and the "Type Name" field contains "Medicines". At the bottom of the form, there are two buttons: "CONFIRM" and "CANCEL".

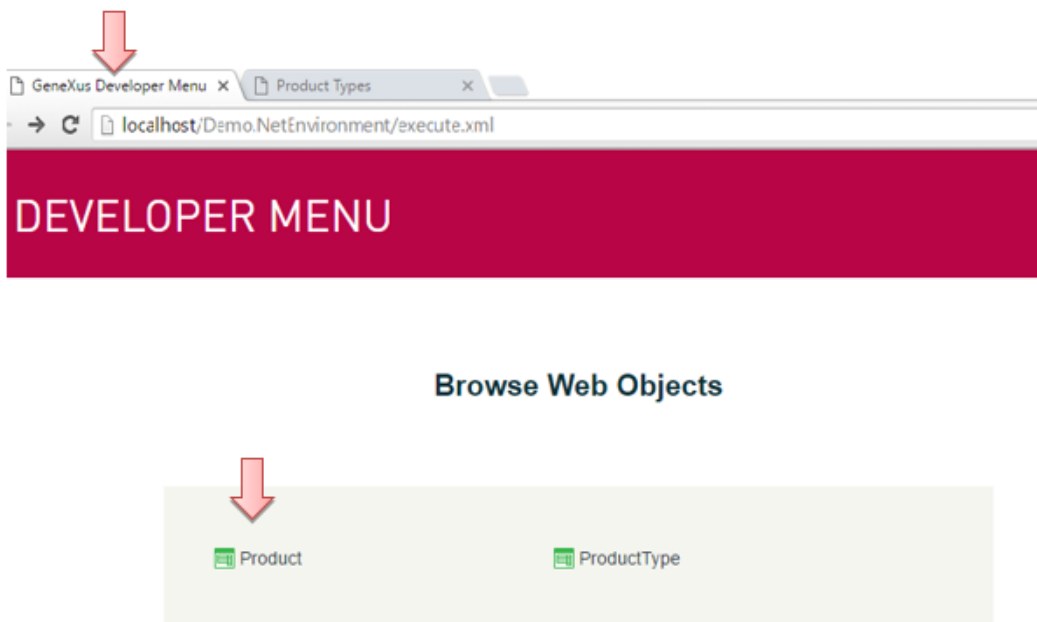
Haga clic en el botón Confirm. Puede navegar y verificar los datos para confirmar que están numerados:



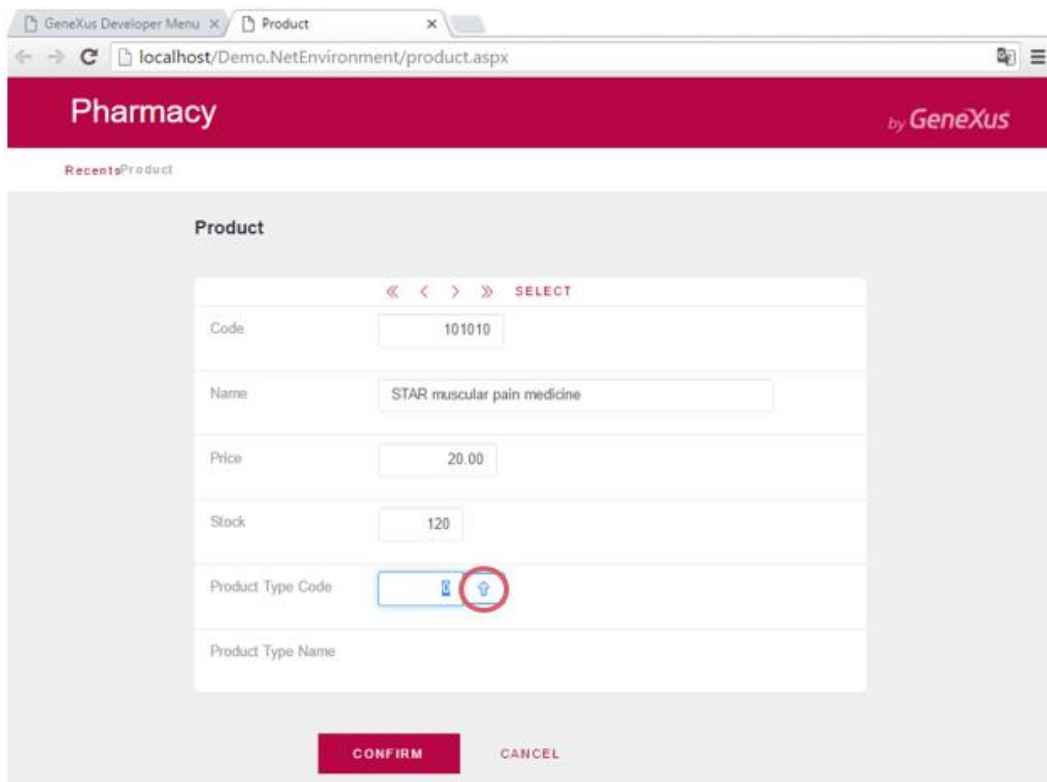
The screenshot shows the same web browser window as the previous one, but the "Product Type" form now displays "1" in the "Type Code" field and "Cosmetics" in the "Type Name" field. The "SELECT" button is circled in red, indicating it is the focus of the instruction. The "CONFIRM" button is highlighted in red, and a "DELETE" button has appeared next to it. The "CANCEL" button remains visible.



Ahora, ejecute la Transacción *Product*. Para hacerlo, seleccione la pestaña del navegador donde se muestra la página del Developer Menu y luego ejecute la Transacción *Product*:



Agregue el primer producto:

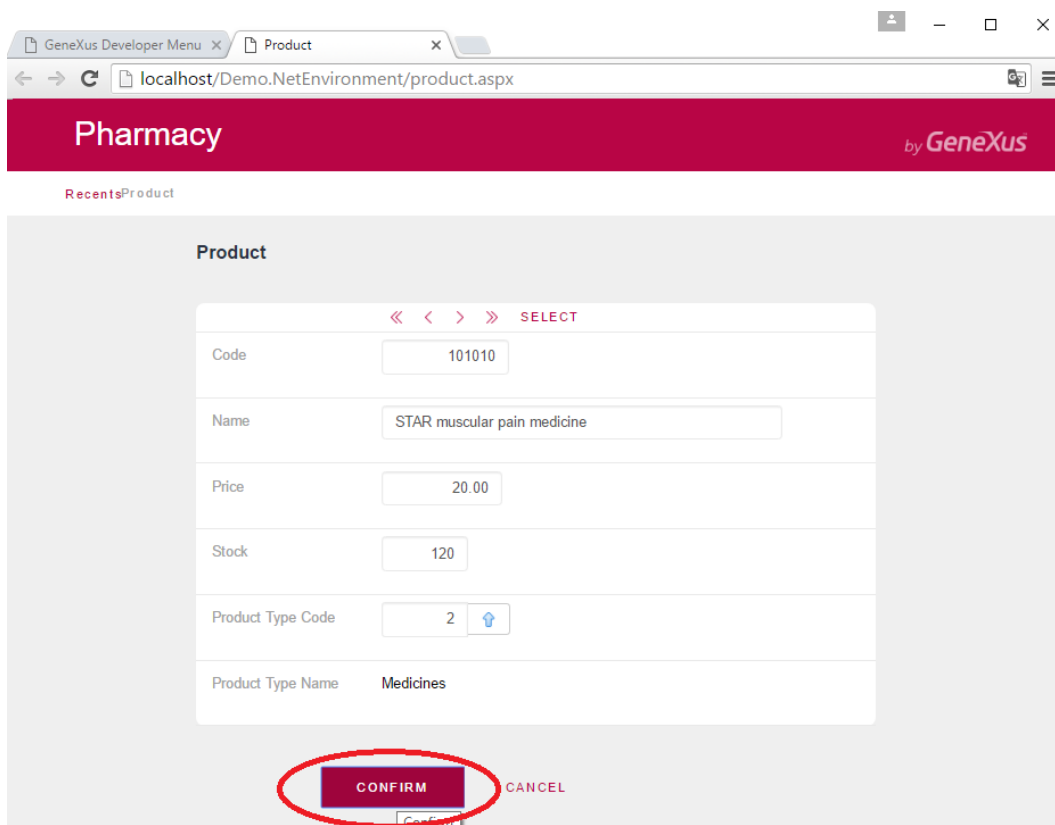


The screenshot shows a web browser window with the URL `localhost/Demo.NetEnvironment/product.aspx`. The page title is "Pharmacy" and it includes the GeneXus logo. The main content area is titled "Product" and contains a form with the following fields:

Field	Value
Code	101010
Name	STAR muscular pain medicine
Price	20.00
Stock	120
Product Type Code	[Dropdown arrow]
Product Type Name	

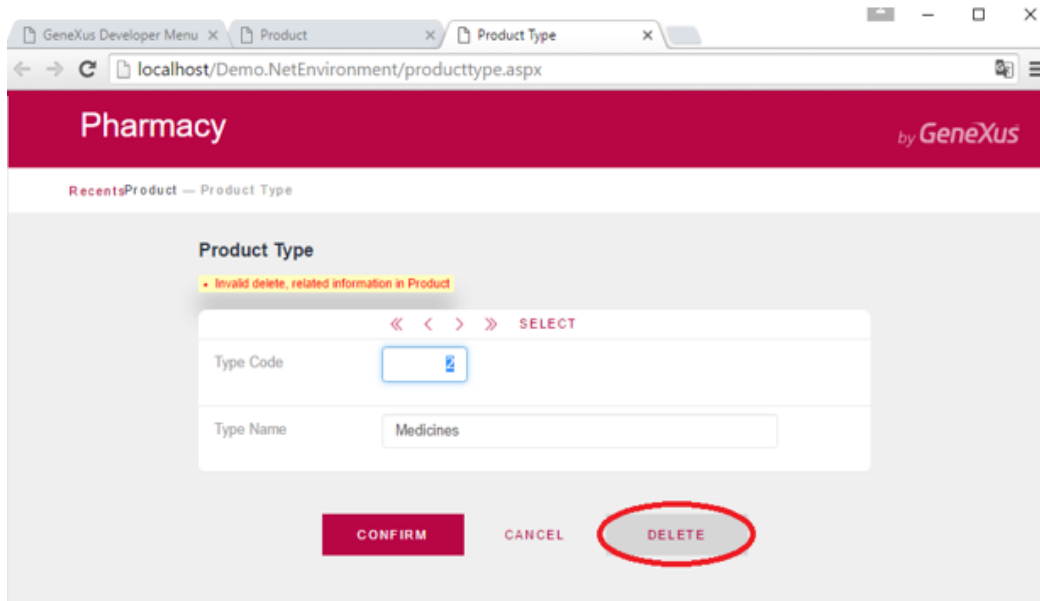
At the bottom of the form, there are two buttons: "CONFIRM" and "CANCEL". The dropdown arrow for "Product Type Code" is circled in red.

Debe indicar el tipo de producto. Si recuerda el código del tipo de producto, puede ingresarlo. Otra opción es seleccionarlo de una lista haciendo clic en la flecha señalizada.



The screenshot shows the same web browser window as the previous one. The "Product Type Code" field now contains the value "2" and has a small dropdown arrow icon next to it. The "Product Type Name" field now contains the text "Medicines". The "CONFIRM" button at the bottom is circled in red.

Ahora, intente eliminar el tipo de producto “Medicines”:



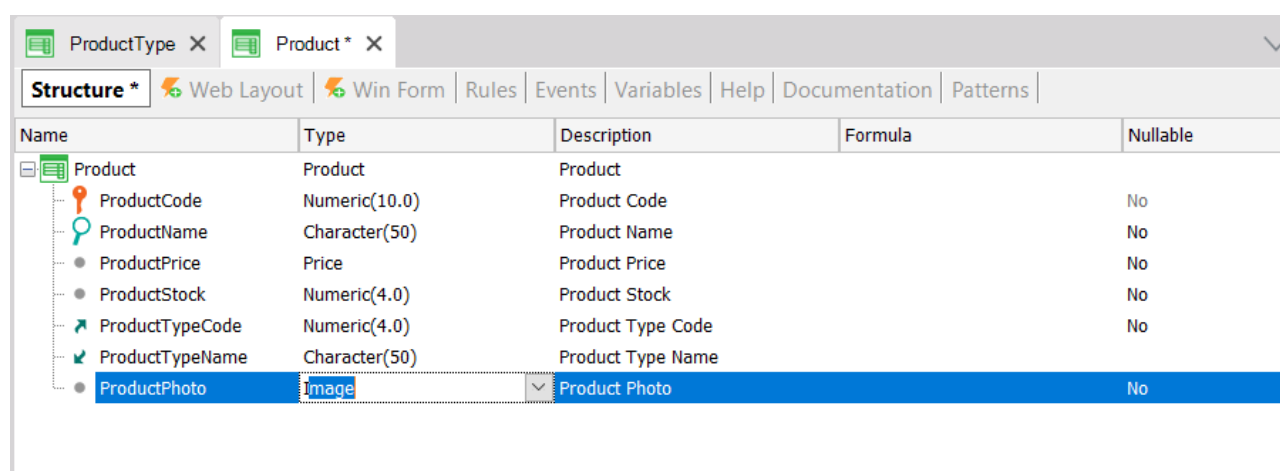
Un mensaje le informa que la eliminación no se puede realizar porque existen datos relacionados en Product (el medicamento para el dolor muscular STAR es un producto que pertenece a este tipo de producto).

LA APLICACIÓN CRECE

Probablemente haya notado todo lo que GeneXus generó automáticamente a partir de los dos objetos que usted definió.

Ahora suponga que en la farmacia le dicen que, para cada producto, quieren registrar una imagen.

Para hacerlo, vuelva a GeneXus y, en la Transacción *Product*, solo tiene que ingresar un atributo llamado *ProductPhoto*:



El tipo Image le permite almacenar imágenes.

La interfaz Web (Web Layout) se actualiza automáticamente, incluyendo el atributo *ProductPhoto*.

Presione F5 y GeneXus evaluará el impacto causado por las nuevas definiciones en la Base de Conocimiento. A continuación, se mostrará el Informe de **Análisis de Impacto (Impact Analysis Report)**:

Database needs to be reorganized.

This report describes Database changes and how they will be handled by reorganization programs.
Please select Reorganize to proceed or Cancel.

Reorganize Cancel

Table Product specification

Table name: Product

Product needs conversion

Warnings

⚠ gqz007 Attribute ProductPhoto does not allow nulls and does not have an Inibal Value. An empty default value will be used.

Table Structure

Attribute	Definition	Previous values	Takes value from
ProductCode	Numeric (10)Not null		Product- ProductCode
ProductName	Character (50)Not null		Product- ProductName
ProductPrice	Numeric (9,2)Not null		Product- ProductPrice
ProductStock	Numeric (4)Not null		Product- ProductStock
ProductTypeCode	Numeric (4)Not null		Product- ProductTypeCode
New ProductPhoto	Image Not null		nullvalue(ProductPhoto)
New ProductPhoto_GXI	Varchar (2048)Not null		

Indexes

Name	Definition	Composition
IPRODUCT	primary key Clustered	ProductCode
IPRODUCT1	duplicate	ProductTypeCode

Foreign key constraints

Referenced table: ProductType

Attributes: ProductTypeCode

Statements

```
ALTER TABLE [Product]
ADD [ProductPhoto_GXI] VARCHAR(2048) NOT NULL CONSTRAINT ProductPhoto_GXIProduct_DEFAULT DEFAULT '',
[ProductPhoto] VARBINARY(MAX) NOT NULL CONSTRAINT ProductPhotoProduct_DEFAULT DEFAULT Convert(VARBINARY(1),'')

ALTER TABLE [Product]
DROP CONSTRAINT ProductPhoto_GXIProduct_DEFAULT

ALTER TABLE [Product]
DROP CONSTRAINT ProductPhotoProduct_DEFAULT
```

Recuerde que el Informe de Análisis de Impacto indica los cambios estructurales que se necesitan realizar en la base de datos.

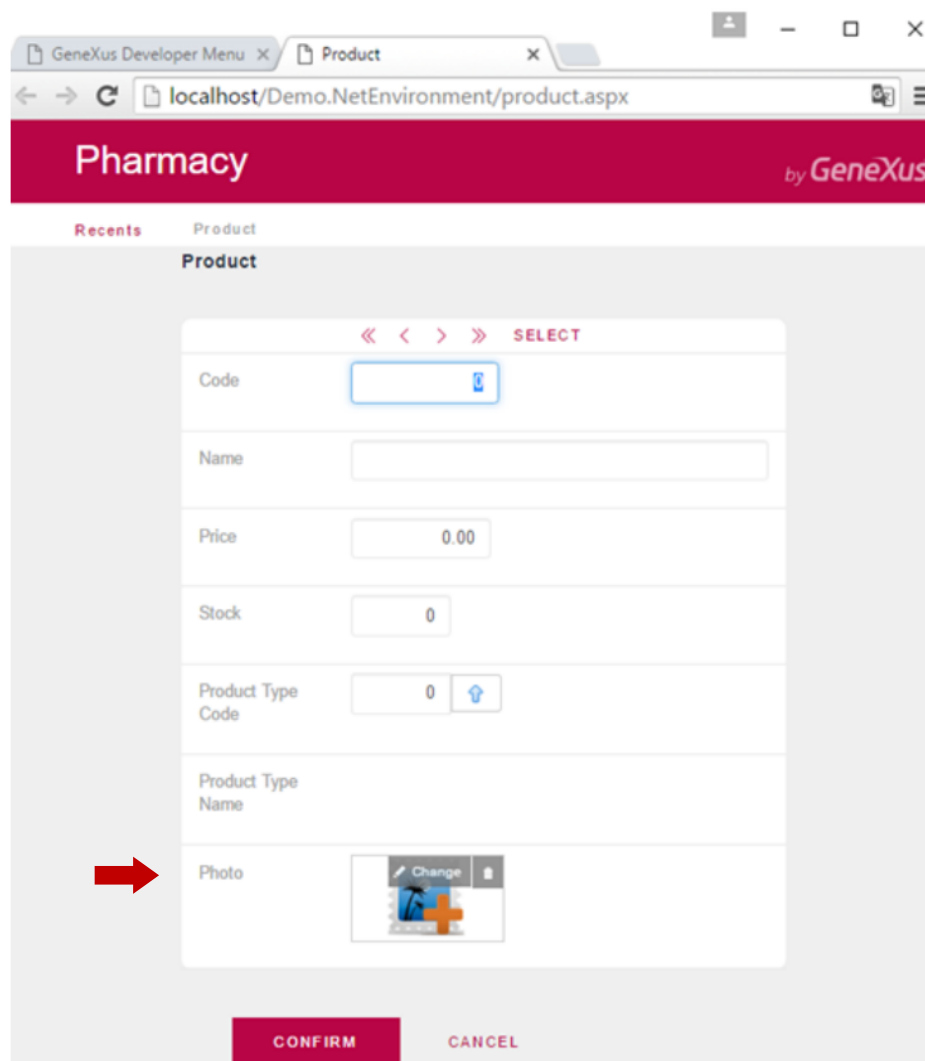
Al leer el informe, verá que el título principal, en este caso, informa que **se necesita una reorganización de la base de datos**. El término “Reorganizar” se refiere a la tarea de realizar cambios en la base de datos. En este caso en particular, el informe indica que la tabla Product debe actualizarse.

Al hacer clic en el botón Reorganizar, GeneXus creará y ejecutará los programas que harán cambios en la base de datos. Luego, generará los programas requeridos que correspondan a la aplicación.

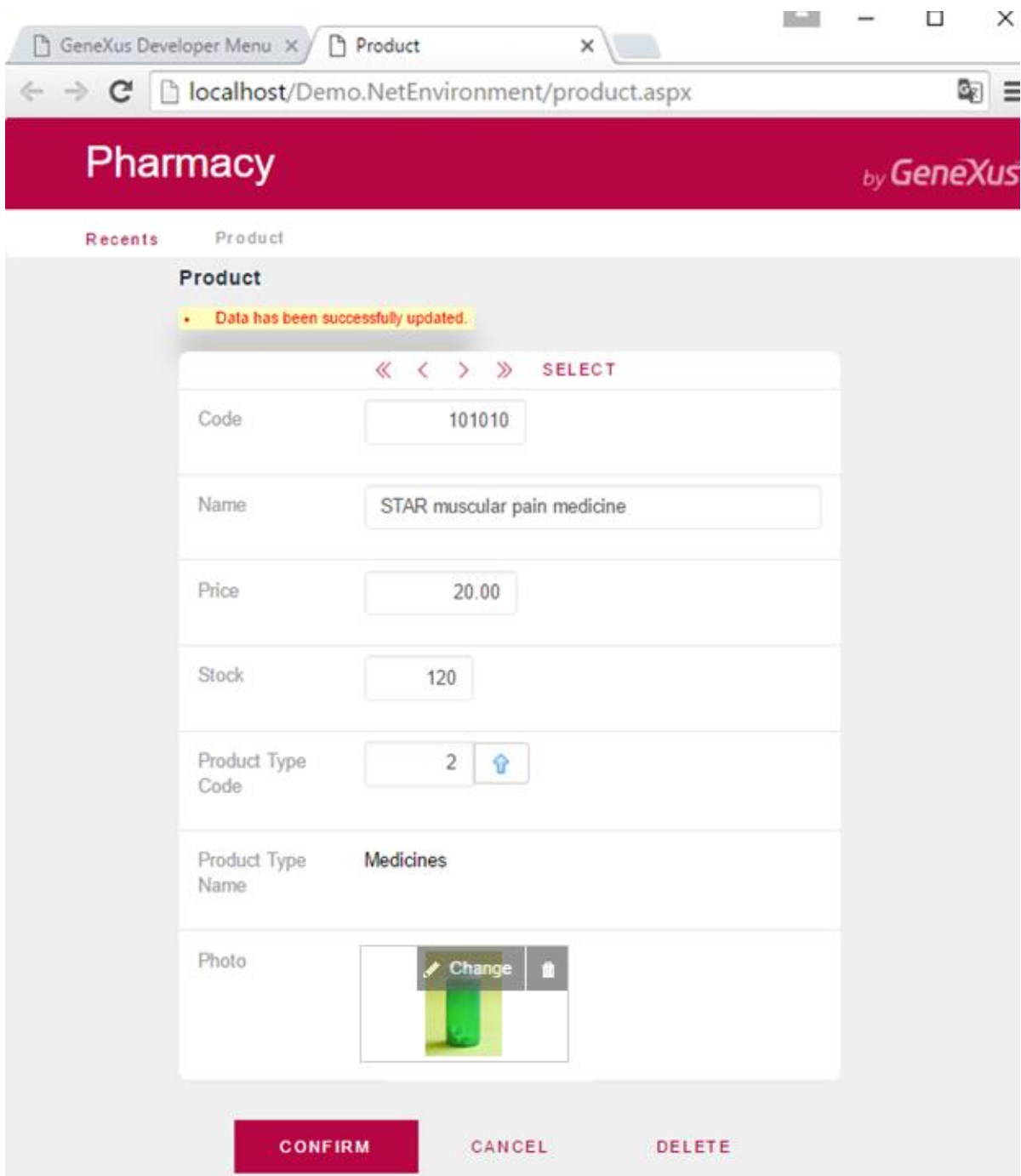
Observe que inmediatamente tendrá la aplicación ejecutándose nuevamente en el navegador web, con la nueva definición incluida:







Browse Web Objects



Si recuerda el código del producto, puede ingresarlo. Otra opción es seleccionarlo de una lista haciendo clic en el botón SELECT. Desde allí, puede recuperar el producto “STAR muscular pain medicine” y subir su foto:



The screenshot shows a web browser window with the URL `localhost/Demo.NetEnvironment/product.aspx`. The page has a red header with the text "Pharmacy" and "by GeneXus". Below the header, there are two tabs: "Recents" and "Product". The main content area is titled "Product" and contains a message: "Data has been successfully updated." Below this message is a form with the following fields:

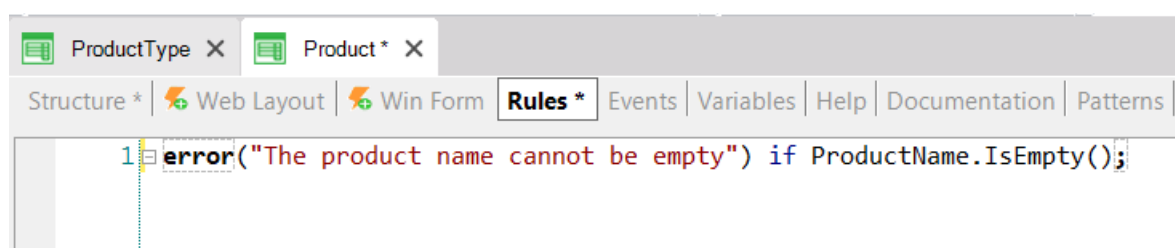
Product	
Code	101010
Name	STAR muscular pain medicine
Price	20.00
Stock	120
Product Type Code	2 
Product Type Name	Medicines
Photo	  

At the bottom of the form, there are three buttons: "CONFIRM", "CANCEL", and "DELETE".

AGREGAR REGLAS DE NEGOCIOS

Además de todos los controles automáticos incluidos por GeneXus en las aplicaciones que genera, a veces quienes solicitan la aplicación piden realizar algunos controles específicos. En las Transacciones, las reglas que deben cumplirse, o los controles que se deben realizar, se definen en la sección **Rules**.

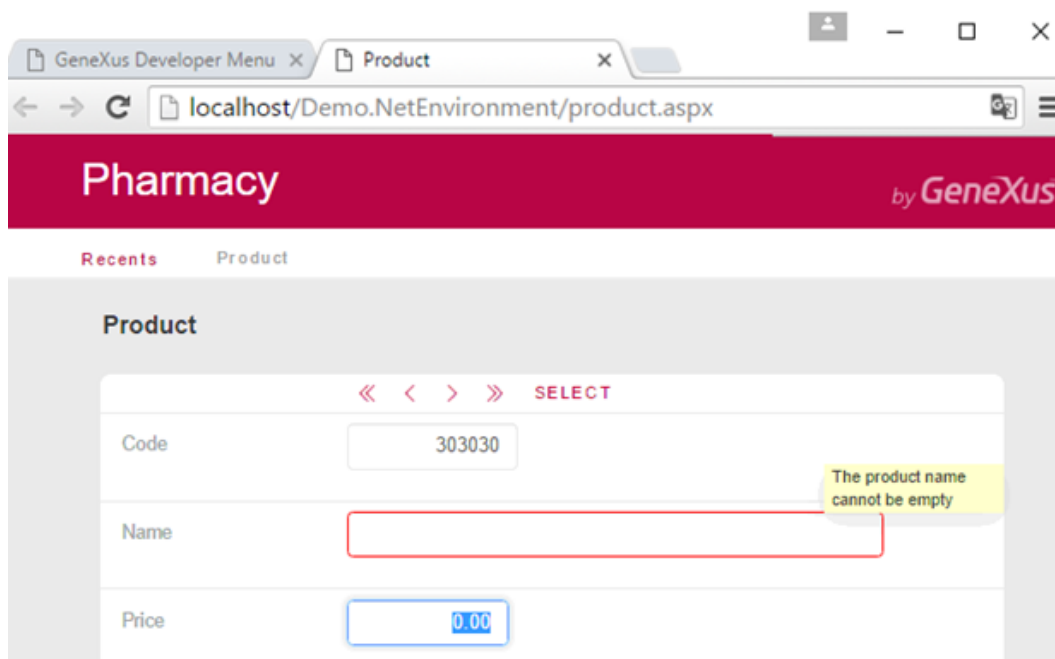
Si, por ejemplo, un requisito es que no se permita guardar productos sin un nombre, GeneXus ofrece una regla llamada **Error** que le permitirá evitar eso:



```
1 error("The product name cannot be empty") if ProductName.IsEmpty();
```

Presione F5 y GeneXus guardará y evaluará las nuevas definiciones incluidas en la Base de Conocimiento. En este caso, deducirá que no es necesario modificar la base de datos, por lo que no mostrará un Informe de Análisis de Impacto. GeneXus generará el código necesario y luego ejecutará la aplicación actualizada con las nuevas definiciones.

Ejecute la Transacción *Product*. Tenga en cuenta que, si el nombre del producto se deja en blanco, la regla que definió se ejecutará:



Hay otra regla cuya sintaxis es muy similar a la regla de error. Se llama **Msg** y la única diferencia entre ellas es que si se cumple la condición, en este caso el mensaje se muestra como un aviso o advertencia, y el usuario puede seguir trabajando.

Si, por ejemplo, desea informar que el precio del producto se ha dejado en blanco sin obligar al usuario a que lo ingrese, puede agregar la siguiente regla en la Transacción *Product*:

```

ProductType X Product * X
Structure | Web Layout | Win Form | Rules * | Events | Variables | Help | Documentation | Patterns |
1 error("The product name cannot be empty")
2   L   if ProductName.IsEmpty();
3 msg("The product price is empty")
4   L   if ProductPrice.IsEmpty();

```

Este conjunto de reglas podría escribirse en cualquier otro orden y el resultado en tiempo de ejecución sería exactamente igual, porque GeneXus analiza y determina cuándo debe dispararse cada una de las reglas definidas (cuando el usuario abandona cada campo, si la condición da verdadero, etc.).

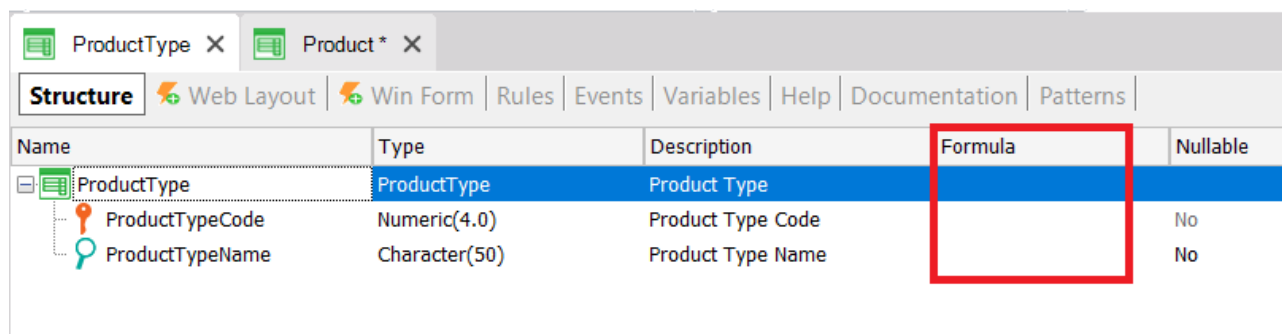
Por supuesto, hay más reglas disponibles en GeneXus para definir diferentes tipos de validaciones y acciones. Cada Transacción debe tener sus propias reglas de comportamiento definidas.

DEFINIR CÁLCULOS: FÓRMULAS

A menudo se necesita que las aplicaciones realicen cálculos que involucren los valores de ciertos atributos, constantes y/o funciones específicas. Para estos casos, GeneXus ofrece definir **Fórmulas**.

Existen diferentes formas posibles de definir fórmulas.

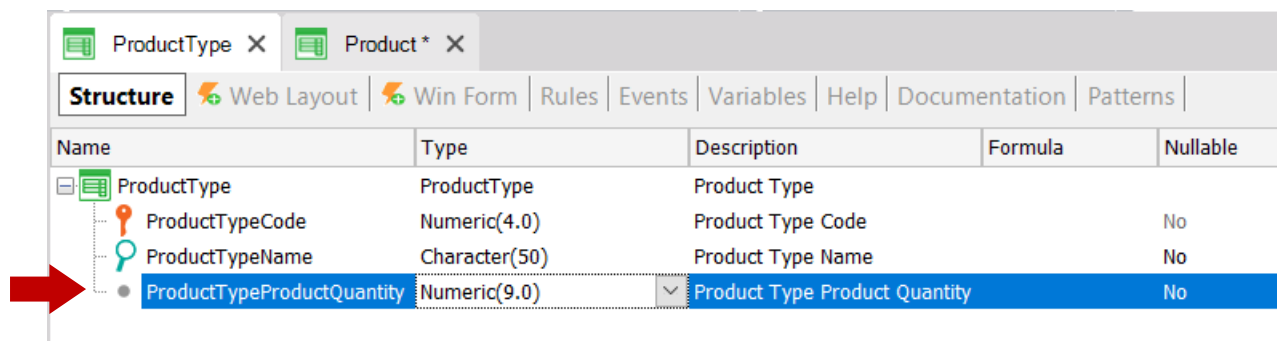
Empecemos por aprender qué es una **Global Formula**. Una fórmula global es un cálculo que el desarrollador define asociado a un atributo. Observe que las estructuras de las Transacciones contienen una columna llamada **Fórmula**:



Name	Type	Description	Formula	Nullable
ProductType	ProductType	Product Type		
ProductTypeCode	Numeric(4.0)	Product Type Code		No
ProductTypeName	Character(50)	Product Type Name		No

Cuando se define un cálculo en esta columna para un atributo, significa que el atributo es virtual. En otras palabras, no se creará físicamente como un campo en una tabla porque el valor del atributo se obtendrá cada vez que sea necesario haciendo el cálculo.

Veamos esto con un ejemplo. Suponga que la farmacia necesita saber en todo momento cuántos productos registrados hay de cada tipo de producto. Para ello, defina un nuevo atributo en la Transacción ProductType con el fin de definirlo como una fórmula global:



Name	Type	Description	Formula	Nullable
ProductType	ProductType	Product Type		
ProductTypeCode	Numeric(4.0)	Product Type Code		No
ProductTypeName	Character(50)	Product Type Name		No
ProductTypeProductQuantity	Numeric(9.0)	Product Type Product Quantity		No

Ahora, defina el cálculo asociado al atributo *ProductTypeProductQuantity*.

GeneXus ofrece una fórmula llamada Count para realizar la cuenta que necesita la farmacia (hay muchas otras, como Sum, Average, etc.).

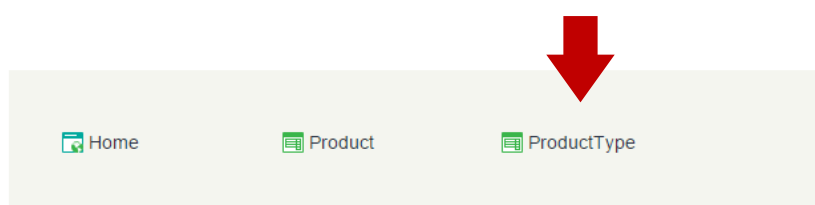
Name	Type	Description	Formula	Nullable
ProductType	ProductType	Product Type		
ProductTypeCode	Numeric(4.0)	Product Type Code		No
ProductTypeName	Character(50)	Product Type Name		No
ProductTypeProductQuantity	Numeric(9.0)	Product Type Product Quantity	count(ProductName)	...

El atributo que se incluye dentro del paréntesis de la fórmula le brinda a GeneXus la información de la tabla a navegar para hacer el cálculo (en la definición anterior, ProductName se encuentra dentro del paréntesis; por lo tanto GeneXus sabe que tiene que contar en la tabla Product). Luego, si GeneXus detecta una relación entre la tabla que navegará (Product) y el contexto donde está definido el atributo fórmula (ProductType), solo considerará los registros relacionados al realizar el cálculo. En este ejemplo, el atributo *ProductTypeId* está presente en ambos contextos: donde la fórmula se encuentra definida y en la tabla que debe navegarse para realizar el cálculo. Por lo tanto, para cada tipo de producto solamente se contarán los productos de dicho tipo de producto (en vez de contar todos los productos registrados en la tabla navegada). Si no se encuentra ninguna relación, GeneXus hará el cálculo considerando todos los registros de la tabla navegada.

Presione F5. Puede ver que no se realizarán cambios físicos en la base de datos. GeneXus solo generará algunos programas y el Developer Menu se ejecutará nuevamente:



Browse Web Objects



Ejecute la Transacción *ProductType* para ver, para cada tipo de producto, cómo la cantidad de productos de dicho tipo siempre se calcula en el momento:

The screenshot shows a web browser window with the URL `localhost/Demo.NetEnvironment/producttype.aspx`. The page has a red header with the text "Pharmacy" and "by GeneXus". Below the header, there are navigation links: "Recents" and "Product — Product Type". The main content area is titled "Product Type" and contains a form with the following fields:

Product Type	
Type Code	1
Type Name	Cosmetics
Product Quantity	0

The "Product Quantity" field is circled in red. Below the form, there are three buttons: "CONFIRM", "CANCEL", and "DELETE".

The screenshot shows a web browser window with the URL `localhost/Demo.NetEnvironment/producttype.aspx`. The page title is 'Pharmacy by GeneXus'. Below the header, there are tabs for 'Recents' and 'Product Type'. The main content area is titled 'Product Type' and contains a form with the following fields:

- Navigation: « < > » SELECT
- Type Code: 2
- Type Name: Medicines
- Product: 1
- Quantity: (empty)

A red oval highlights the 'Product' and 'Quantity' fields. At the bottom of the form, there are three buttons: CONFIRM, CANCEL, and DELETE.

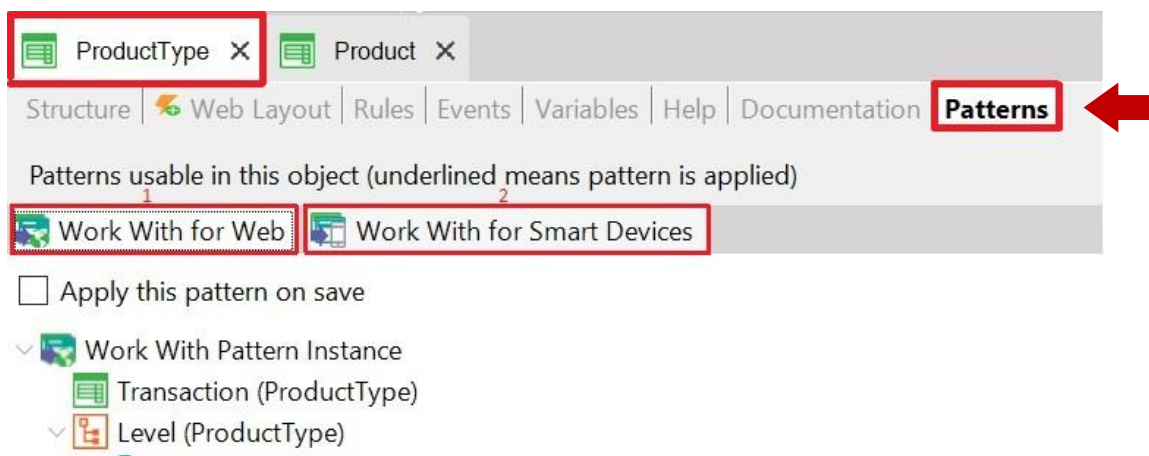
Puede agregar más productos con el fin de verificar, para cada tipo de producto, que la cantidad de productos de dicho tipo siempre se calcula en el momento.

USO DE PATRONES (PARA WEB Y PARA DISPOSITIVOS MÓVILES)

Los patrones le permiten potenciar aún más sus aplicaciones, de forma automática.

Aplicar un patrón es muy fácil, e inmediatamente GeneXus crea objetos, códigos y configuraciones para proporcionar comportamientos interesantes sin necesidad de programarlos.

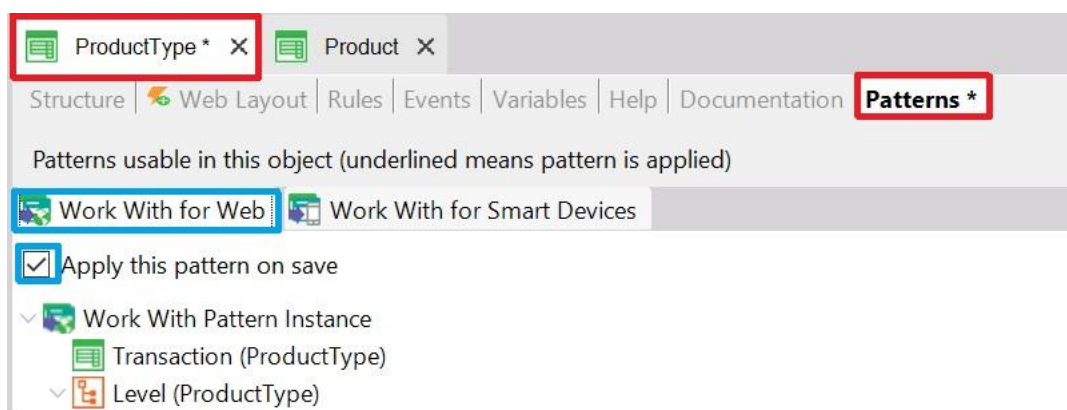
Observe la sección Patterns de una Transacción. Por ejemplo, en la Transacción *ProductType*, seleccione la sección Patterns:



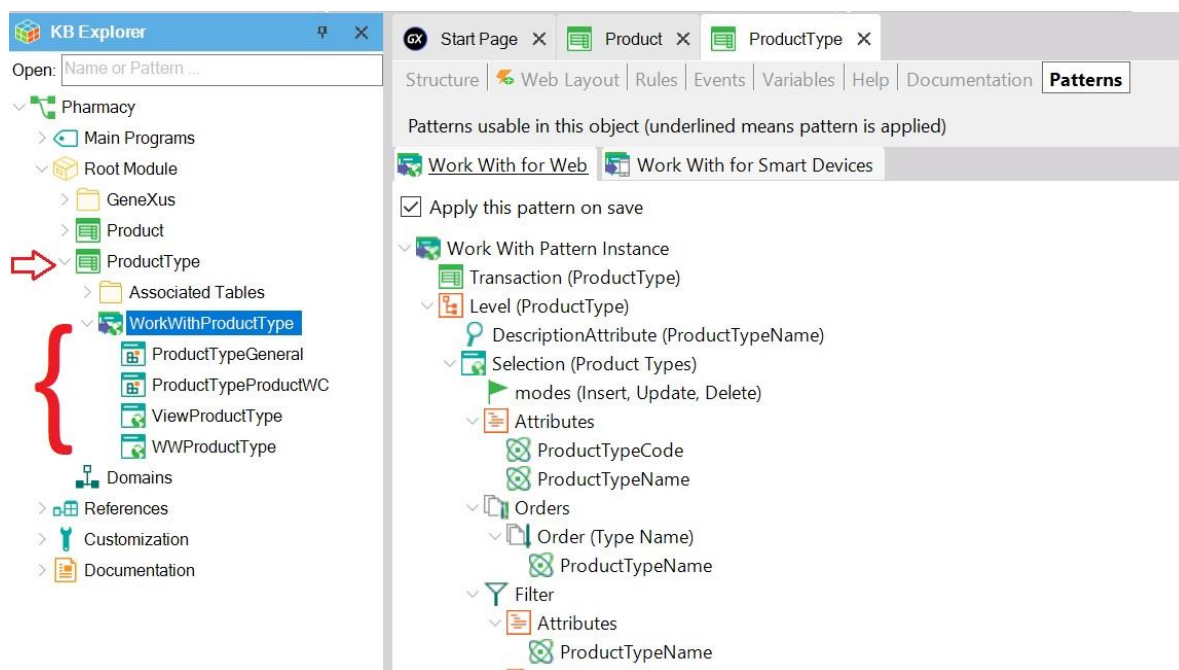
Note que hay dos pestañas disponibles, y cada una ofrece un patrón diferente para aplicarlo a la misma Transacción.

En primer lugar, seleccione la pestaña *Work With for Web*.

Para aplicar el patrón *Work With for Web* solamente tiene que hacer clic en la casilla **Apply this pattern on save** y guardar (📁):

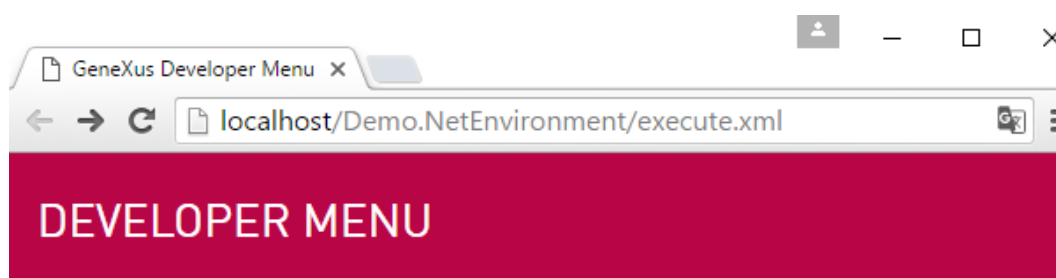


A continuación, si busca la Transacción ProductType en el **KB Explorer**, puede ver que hay varios objetos nuevos debajo de la Transacción:

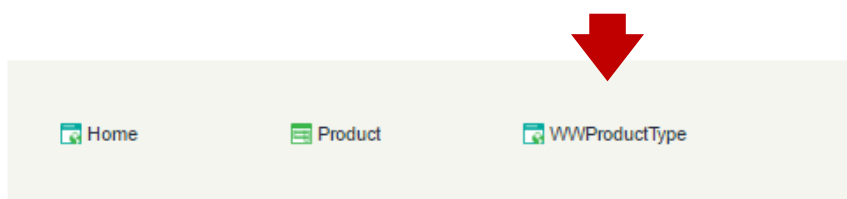


Fueron creados por GeneXus por haber aplicado el patrón *Work With for Web*.

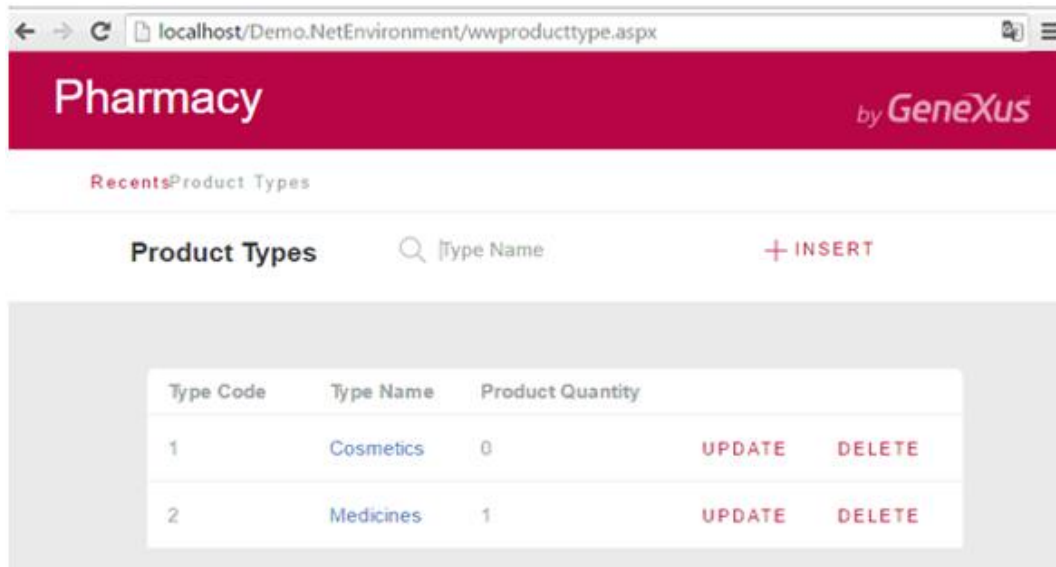
Ahora presione F5 para ver los resultados en tiempo de ejecución:



Browse Web Objects

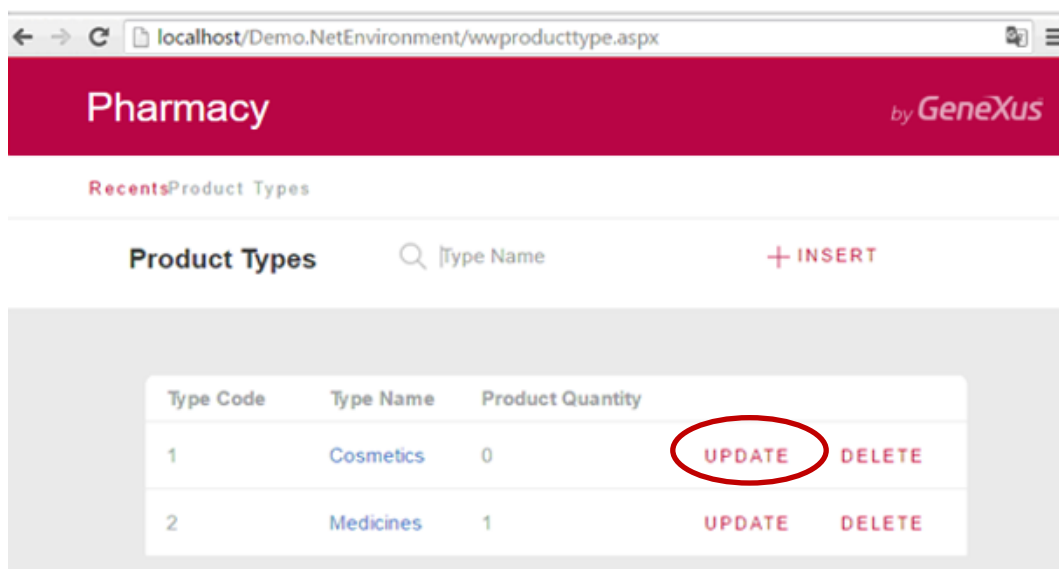


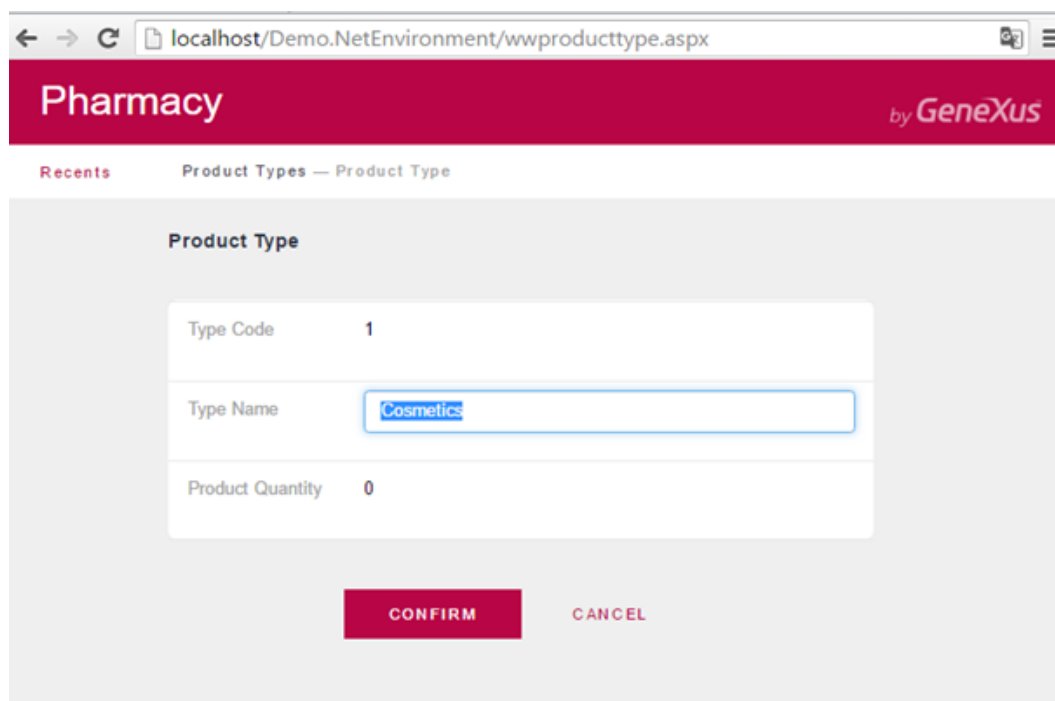
Observe el último enlace que se muestra (WWProductType). Se le ofrece “trabajar con tipos de productos” (y desde allí hay invocaciones a la Transacción *ProductType*). Haga clic en ese enlace.



Puede ver que se abre una página que muestra todos los tipos de productos almacenados. Esta página permite que los usuarios trabajen con los tipos de productos con una variedad más amplia de funcionalidades.

Por ejemplo, haga clic en UPDATE para la primera línea:





localhost/Demo.NetEnvironment/wwproducttype.aspx

Pharmacy

by GeneXus

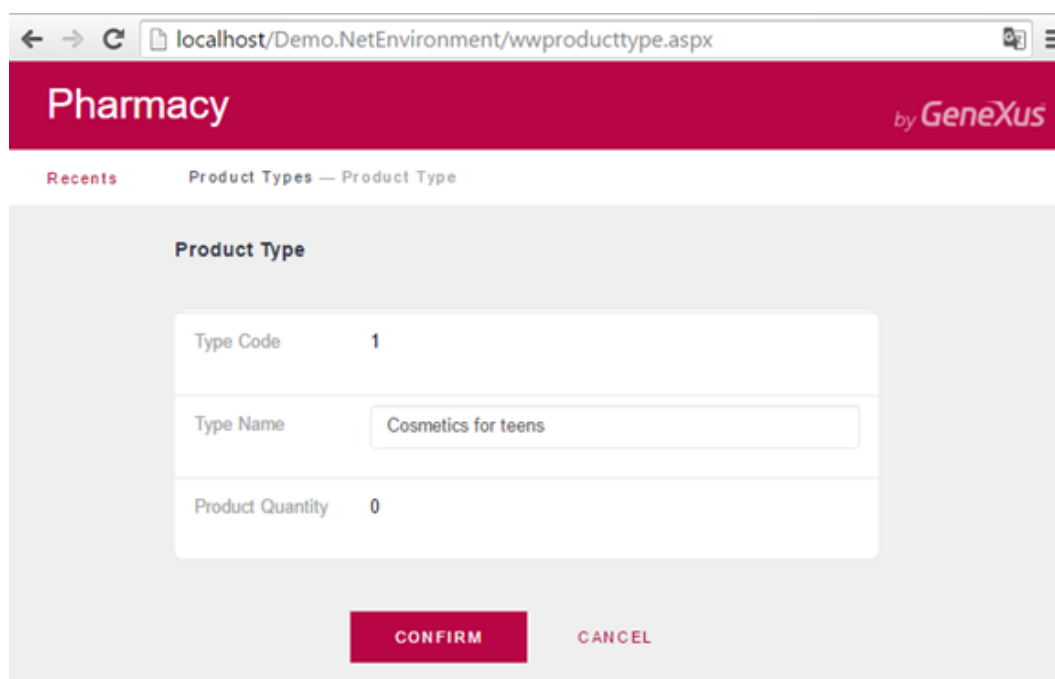
Recents Product Types — Product Type

Product Type

Type Code	1
Type Name	<input type="text" value="Cosmetics"/>
Product Quantity	0

CONFIRM CANCEL

Puede ver que se abre la Transacción *ProductType* para editar los datos del tipo de producto que se encuentra en esa línea. Cambie el nombre del tipo y confirme:



localhost/Demo.NetEnvironment/wwproducttype.aspx

Pharmacy

by GeneXus

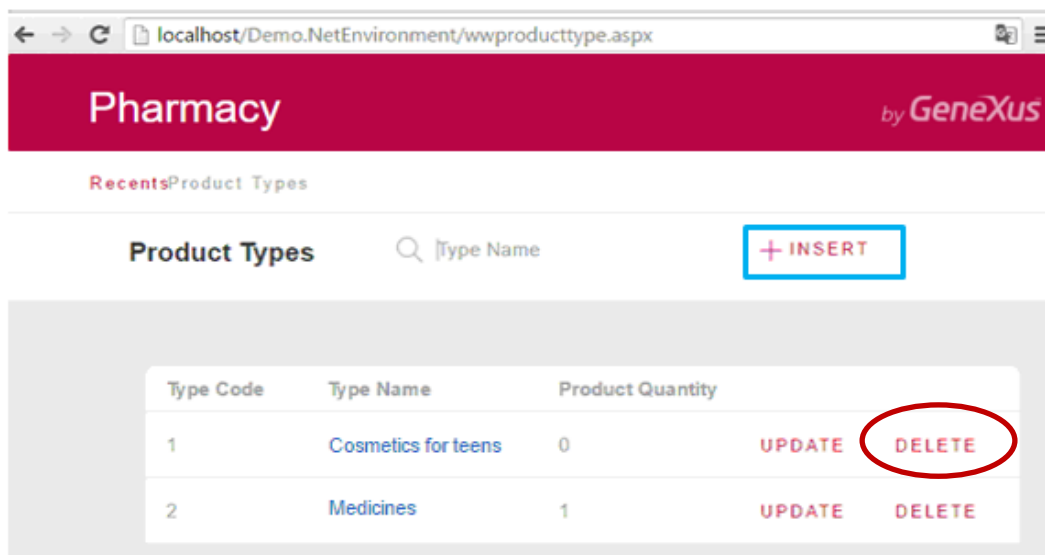
Recents Product Types — Product Type

Product Type

Type Code	1
Type Name	<input type="text" value="Cosmetics for teens"/>
Product Quantity	0

CONFIRM CANCEL

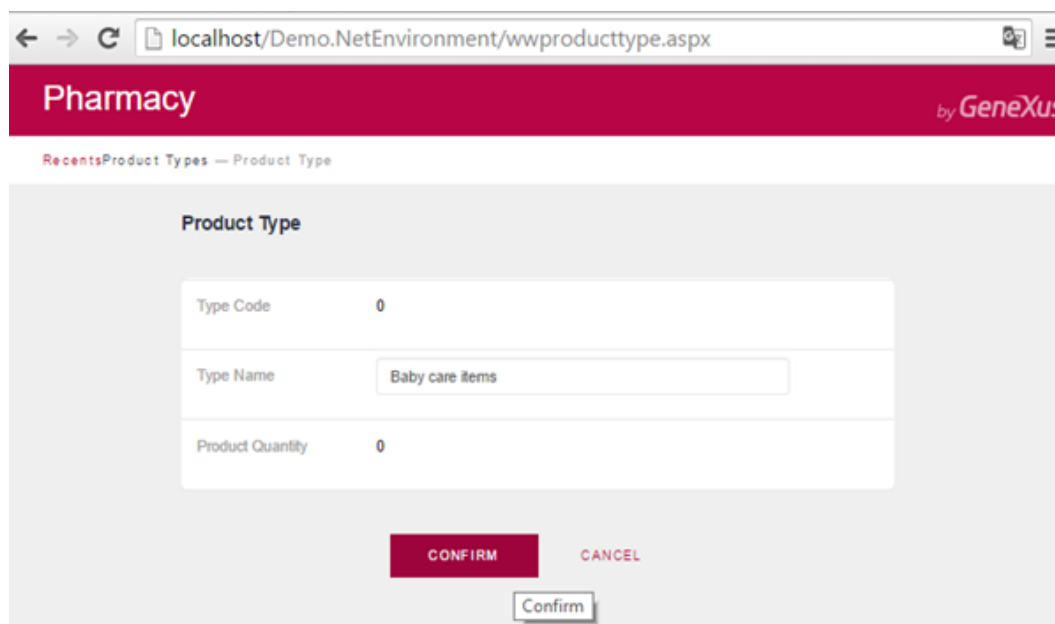
Después de la edición y la confirmación, la aplicación vuelve a la página *Work With Product Types* (Trabajar con tipos de productos):



El enlace DELETE ofrece eliminar el tipo de producto de la línea.

Además, el botón INSERT ubicado fuera de la grilla, permite agregar nuevos tipos de productos.

Al hacer clic en el botón INSERT, se abre la Transacción *ProductType*, lista para agregar un nuevo tipo de producto. Presiónelo para ingresar un nuevo tipo de producto (recuerde que en este caso solo es necesario ingresar el nombre del tipo de producto porque ha configurado la propiedad Autonumber del atributo clave con valor True):



Una vez más, después de la inserción, la aplicación vuelve a la página *Work With Product Types*:

Pharmacy by GeneXus

RecentsProduct Type — Product Types

Product Types + INSERT

Type Code	Type Name	Product Quantity		
3	Baby care items	0	UPDATE	DELETE
1	Cosmetics for teens	0	UPDATE	DELETE
2	Medicines	1	UPDATE	DELETE

Ahora observe que cada nombre de tipo de producto tiene un enlace. Haga clic en el enlace: “Medicines”.

Pharmacy by GeneXus

RecentsProduct Type — Product Types

Product Types + INSERT

Type Code	Type Name	Product Quantity		
3	Baby care items	0	UPDATE	DELETE
1	Cosmetics for teens	0	UPDATE	DELETE
2	Medicines	1	UPDATE	DELETE

Se abre la siguiente página:

Product Type Information ← PRODUCT TYPES

Type Name Medicines

General Product

Type Code 2

Type Name Medicines


UPDATE DELETE

Como puede observar, en la primera pestaña se muestran todos los detalles del tipo de producto seleccionado, y otra pestaña muestra la lista de productos que pertenecen a ese tipo de producto:

Product Type Information ← PRODUCT TYPES

Type Name Medicines

General Product

Code	Name	Price	Stock	Photo
101010	STAR muscular pain medicine	20.00	120	

La *pestaña Product* se generó automáticamente porque **cada tipo de producto tiene varios productos relacionados**. Si cada tipo de producto también tuviera varios datos relacionados de otro tipo, se habrían generado más pestañas para mostrar cada lista de datos relacionados con el tipo de producto.

Ahora, vuelva a la página *Work With Product Types*, haciendo clic en el enlace correspondiente en la sección **Recientes (Recents)** que aparece en la parte superior izquierda de la página.

Observe que es posible buscar por nombre. Esto significa que si, por ejemplo, el usuario escribe "C", solo se mostrarán los tipos de productos que comienzan con esta letra:

Pharmacy

by GeneXus

RecentsProduct Type — Medicines — Product Types

Product Types

d

+ INSERT

Type Code	Type Name	Product Quantity		
1	Cosmetics for teens	0	UPDATE	DELETE

Vuelva a GeneXus. Hasta ahora, solo ha seleccionado **Apply this pattern on save** en la pestaña *Work With for Web* de la Transacción *ProductType*, y después de guardar, ha visto todas las funciones que se generan automáticamente.

Lo que quizás no haya notado es este árbol configurable:

Product × ProductType ×

Structure | Web Layout | Rules | Events | Variables | Help | Documentation | **Patterns**

Patterns usable in this object (underlined means pattern is applied)

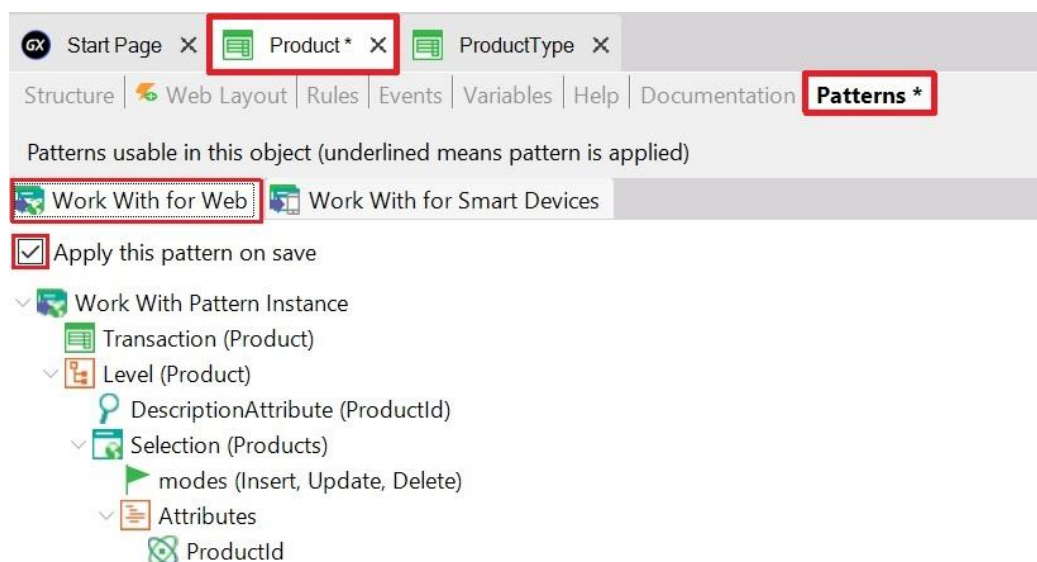
Work With for Web | Work With for Smart Devices

Apply this pattern on save

- Transaction (ProductType)
 - Level (ProductType)
 - DescriptionAttribute (ProductType)
 - Selection (Product Types)
 - modes (Insert, Update, Delete)
 - Attributes
 - ProductType
 - Orders
 - Order (Type Id)
 - ProductType
 - Filter
 - Attributes
 - ProductType
 - Conditions
 - ProductType >= &ProductType when not &ProductType.IsEmpty()
 - View (Product Type Information)
 - Parameters
 - ProductType
 - Fixed Data
 - Attributes
 - ProductType
 - Tabs
 - Tab (General)

Tiene nodos, subnodos y elementos configurables, de modo que es posible personalizar los comportamientos a generar (es decir, cambiar o agregar criterios de búsqueda, etc.).

Aplice también el patrón *Work With for Web* a la Transacción Product. Como se explicó antes, solo tiene que abrir la Transacción Product y en su sección Patterns, debe seleccionar la pestaña *Work With for Web*; entonces, tiene que marcar la opción **Apply this pattern on save** y guardar:



Presione F5. GeneXus genera los programas necesarios y ejecuta la aplicación con los cambios. Luego, ejecute la página *Work With Product*:



Puede ver las mismas funciones de consulta que ya vio en la página *Work With Product Types*:

Pharmacy


by GeneXus

Recents Products

Products

🔍 Name

+ INSERT

Code	Name	Price	Stock	Product Type Code	Product Type Name	Photo		
101010	STAR muscular pain medicine	20.00	120	2	Medicines		UPDATE	DELETE



Inserte un nuevo producto:

Pharmacy

by GeneXus

Recents Products — Product

Product

Code	<input type="text" value="202020"/>
Name	<input type="text" value="X Eyeshadow Palette"/>
Price	<input type="text" value="25.00"/>
Stock	<input type="text" value="30"/>
Product Type Code	<input type="text" value="1"/> 
Product Type Name	Cosmetics for teens
Photo	

CONFIRM

CANCEL

Después de la confirmación, la aplicación regresa a la página *Work With Product*:

Pharmacy



by GeneXus

Recents Products

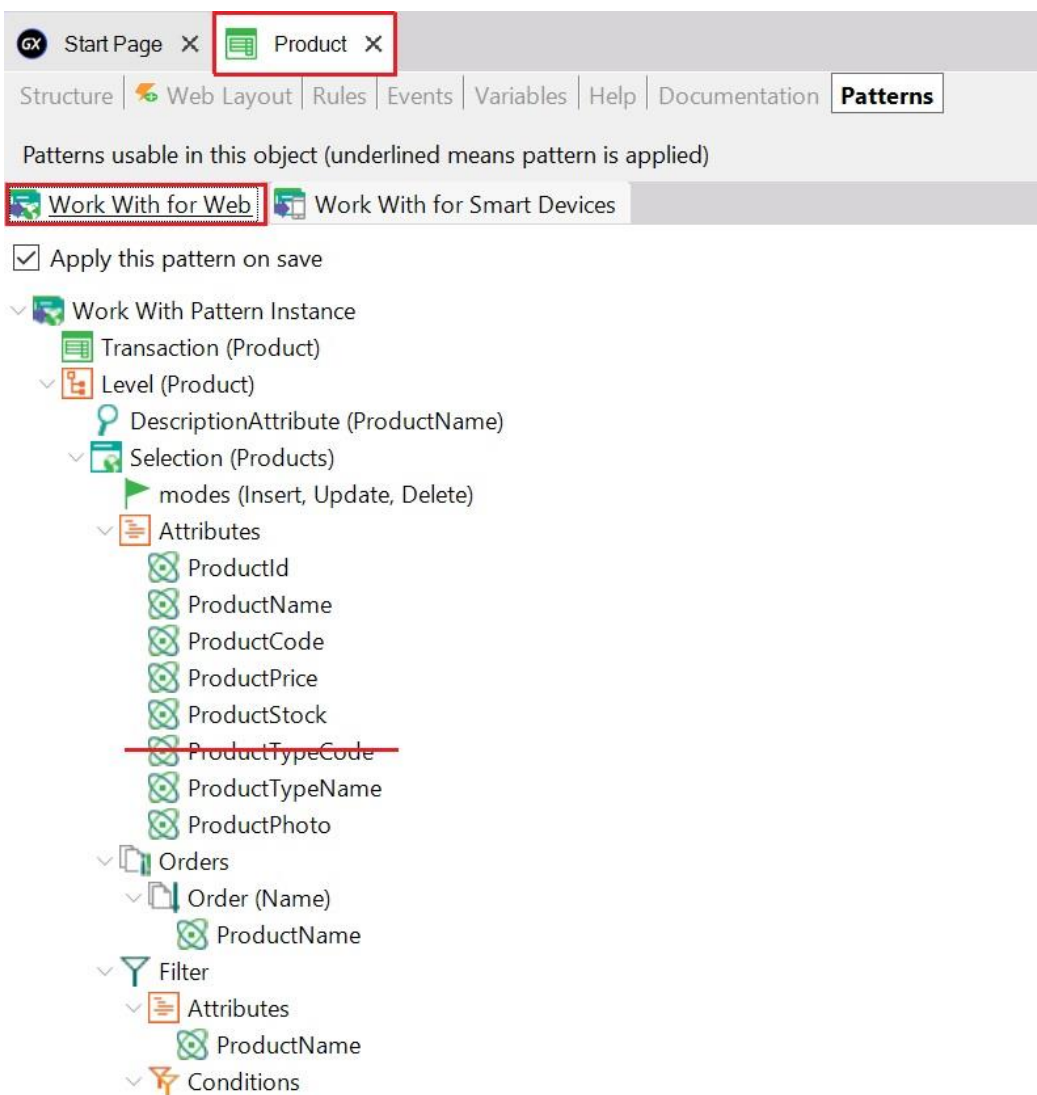
Products

Name

+ INSERT

Code	Name	Price	Stock	Product Type Code	Product Type Name	Photo		
101010	STAR muscular pain medicine	20.00	120	2	Medicines		UPDATE	DELETE
202020	X Eyeshadow Palette	25.00	30	1	Cosmetics for teens		UPDATE	DELETE

Dado que es irrelevante mostrar el código de tipo de producto en la grilla, elimínalo del árbol configurable que se tiene en cuenta para generar este componente *Work With for Web*:



Start Page X Product X

Structure | Web Layout | Rules | Events | Variables | Help | Documentation | **Patterns**

Patterns usable in this object (underlined means pattern is applied)

Work With for Web Work With for Smart Devices

Apply this pattern on save



- Work With Pattern Instance
 - Transaction (Product)
 - Level (Product)
 - DescriptionAttribute (ProductName)
 - Selection (Products)
 - modes (Insert, Update, Delete)
 - Attributes
 - ProductId
 - ProductName
 - ProductCode
 - ProductPrice
 - ProductStock
 - ~~ProductTypeCode~~
 - ProductTypeName
 - ProductPhoto
 - Orders
 - Order (Name)
 - ProductName
 - Filter
 - Attributes
 - ProductName
 - Conditions

Presione F5 y GeneXus guardará lo realizado, generando solo los programas necesarios y ejecutando la aplicación con los cambios:

Pharmacy by GeneXus

Recents Products

Products
🔍 Name
+ INSERT

Code	Name	Price	Stock	Product Type Name	Photo		
101010	STAR muscular pain medicine	20.00	120	Medicines		UPDATE	DELETE
202020	X Eyeshadow Palette	25.00	30	Cosmetics for teens		UPDATE	DELETE








Agregue algunos productos de la misma manera consecutiva que se mostró anteriormente (presionando el botón INSERT que invoca la Transacción *Product*).

A continuación, se muestra el cuadro de diálogo Work With Products que enumera todos los productos que se han agregado:

Pharmacy by GeneXus

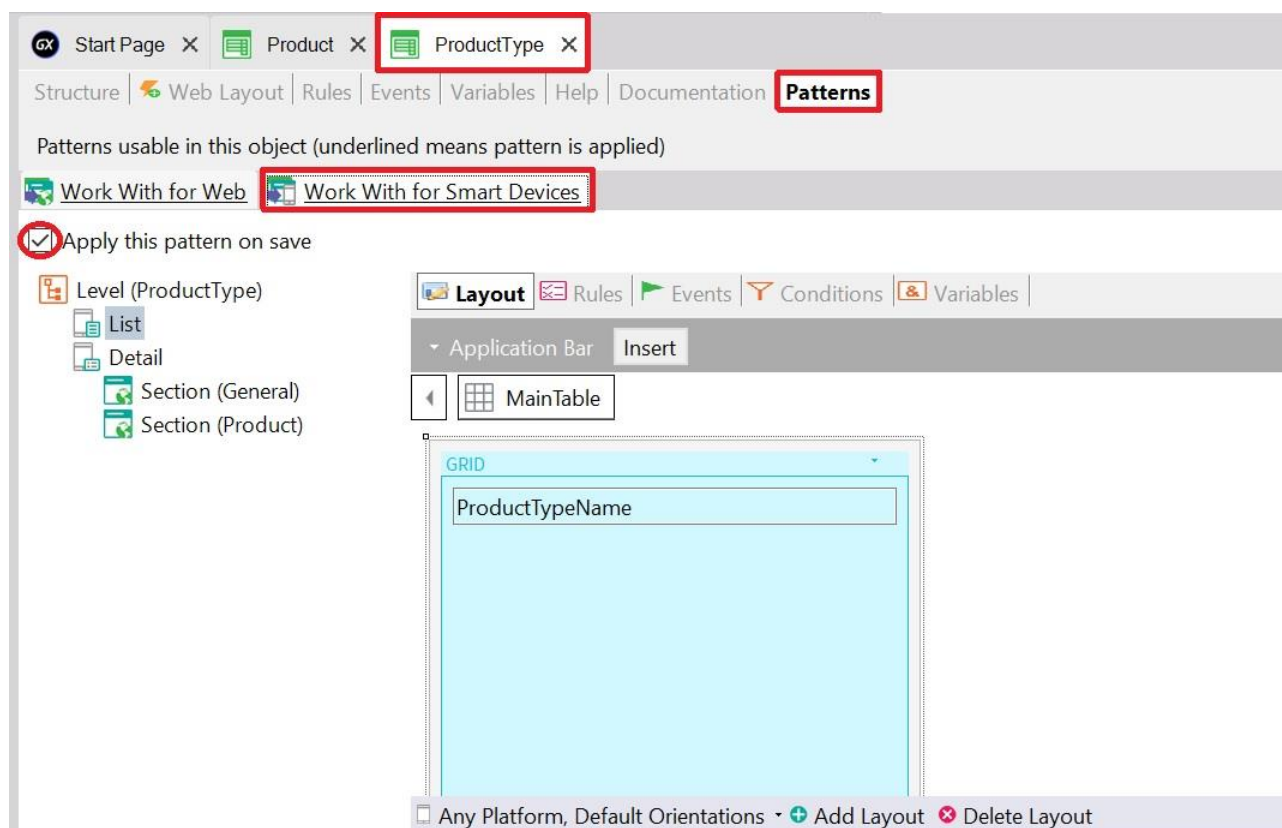
Recents LOVE Lipstick #18 — Asterix Headache M... — XXX Antiseptic cream — WONDER facial cream — Product — Products

Products
🔍 Name
+ INSERT

Code	Name	Price	Stock	Product Type Name	Photo		
707070	ABC feeding bottle	20.00	15	Baby care items		UPDATE	DELETE
404040	ASTERIX Headache Medicine	20.00	100	Medicines		UPDATE	DELETE
303030	LOVE Lipstick #18	8.00	60	Cosmetics for teens		UPDATE	DELETE
505050	MAGIC Anti-inflammatory painkillers	30.00	10	Medicines		UPDATE	DELETE
101010	STAR muscular pain medicine	20.00	120	Medicines		UPDATE	DELETE
606060	WONDER facial cream	90.00	20	Cosmetics for teens		UPDATE	DELETE
202020	X Eyeshadow Palette	25.00	30	Cosmetics for teens		UPDATE	DELETE

Ahora preste atención a la pestaña *Work With for Smart Devices* que se ofrece para cada Transacción.

Aplíquelo a la Transacción *ProductType*:



Observe que bajo el nodo principal **Level (ProductType)**, hay un nodo **List**. Si hace clic en él, en la ventana de la derecha verá una grilla que contiene el atributo *ProductTypeName*.

A diferencia del patrón *Work With for Web*, en este caso, ya se muestra el diseño (Layout) con una grilla en vez de una lista de atributos debajo del nodo que se incluirán en la grilla.

Ahora, observe el nodo **Detail**, que permite ver los detalles de una línea determinada de la grilla.

El nodo **Detail** está compuesto por dos secciones: **General** y **Product**.

Al igual que la funcionalidad implementada por el patrón *Work With for Web*, la sección **General** muestra los datos asociados con el tipo de producto seleccionado y la sección **Product** muestra dentro de una grilla todos los productos que pertenecen al tipo de producto.

Start Page × Product × **ProductType ×**

Structure | Web Layout | Rules | Events | Variables | Help | Documentation | **Patterns**

Patterns usable in this object (underlined means pattern is applied)

Work With for Web | Work With for Smart Devices

Apply this pattern on save

Level (ProductType)

- List
- Detail
- Section (General)
- Section (Product)**

Layout | Rules | Events | Conditions | Variables

Application Bar | Update | Delete

MainTable

Type Code | ProductTypeCode

Type Name | ProductTypeName

Any Platform, View, Default Orientations | + Add Layout | × Delete Layout

Product × **ProductType ×**

Structure | Web Layout | Rules | Events | Variables | Help | Documentation | **Patterns**

Patterns usable in this object (underlined means pattern is applied)

Work With for Web | Work With for Smart Devices

Apply this pattern on save

Level (ProductType)

- List
- Detail
- Section (General)
- Section (Product)**

Layout | Rules | Events | Conditions | Variables

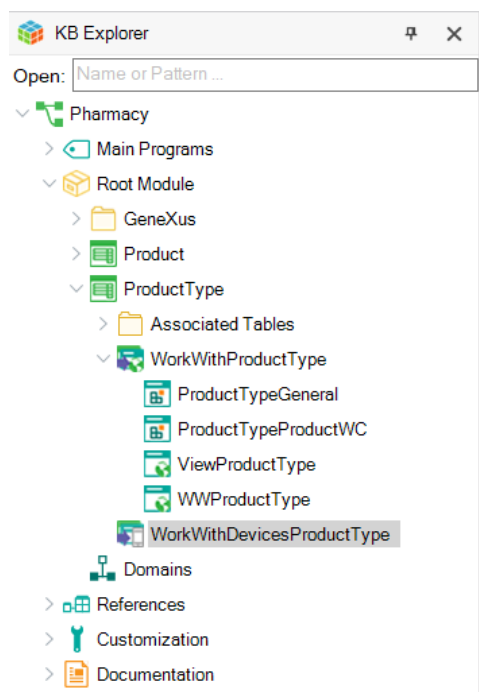
Application Bar

MainTable

GRID

ProductName

Después de aplicar este patrón y guardar, si observa la Transacción *ProductType* en el KB Explorer, ahora puede ver un nuevo objeto llamado *WorkWithDevicesProductType* debajo de la Transacción *ProductType*:



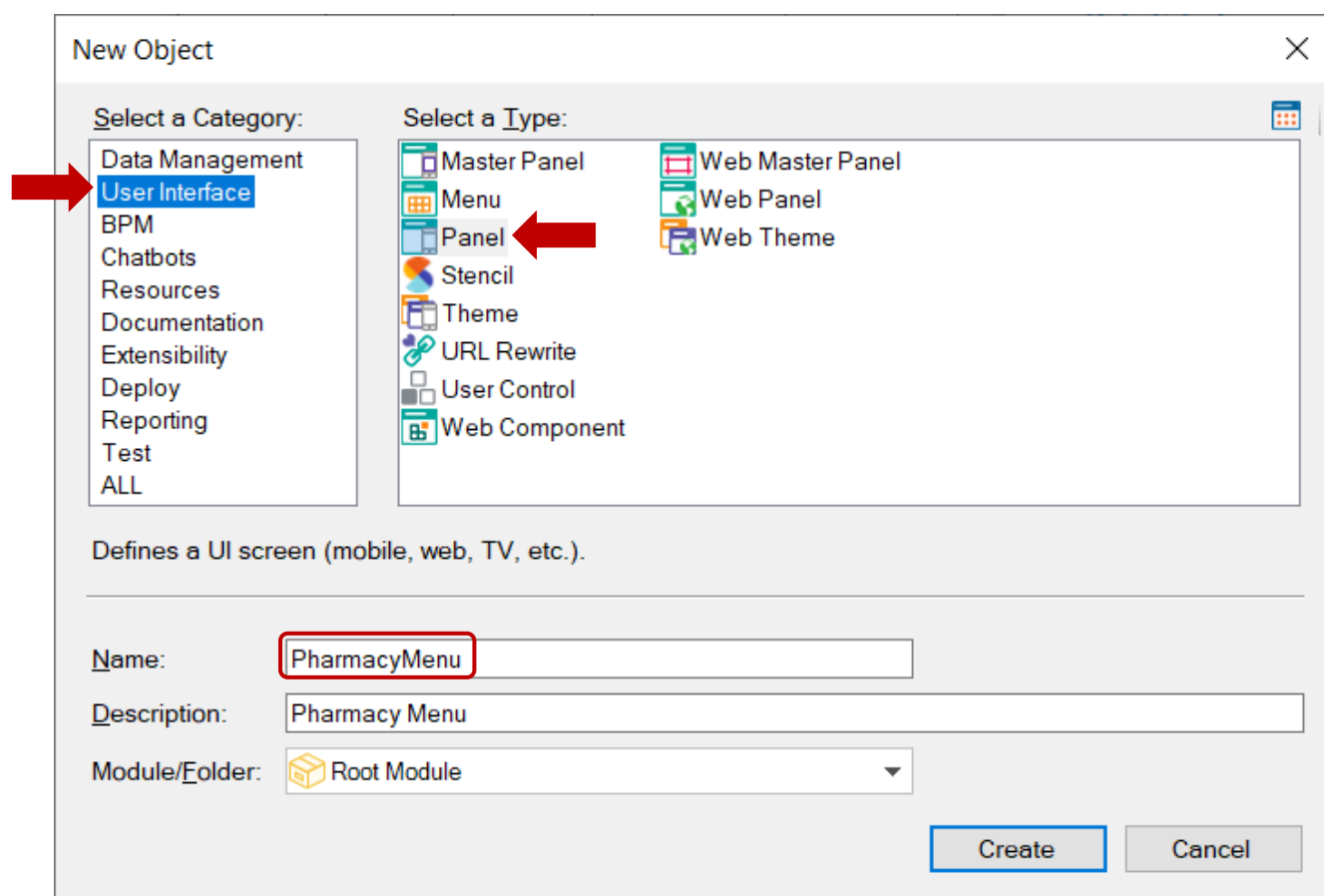
Si observa los dos objetos generados en la Transacción *ProductType*, puede observar que *WorkWithDevicesProductType* no tiene otros objetos debajo (porque incluye diferentes secciones para definir la implementación completa en su interior).

En cambio, *WorkWithProductType*, como ya ha visto, es una instancia configurable; y GeneXus genera otros objetos debajo de él para proporcionar los comportamientos útiles que se han mencionado.

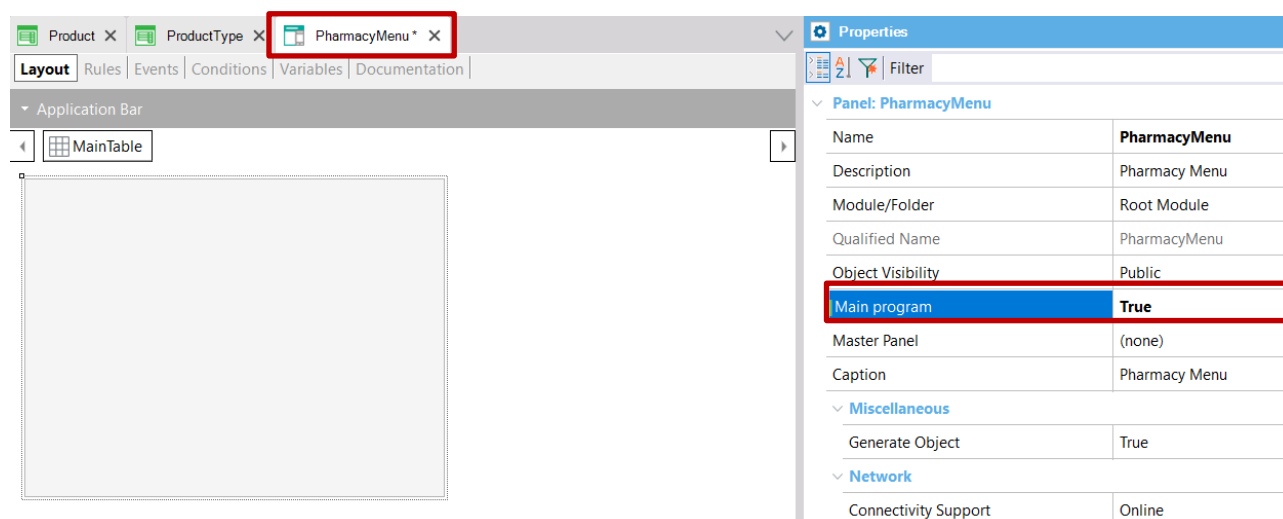
Pronto, verá el componente *WorkWithDevicesProductType* en acción.

La propuesta ahora es crear un objeto **Panel**, para que sea el primer objeto de la aplicación que se ejecute. El mismo mostrará una imagen, para que cuando el usuario la pulse, se ejecute el objeto *WorkWithDevicesProductType*.

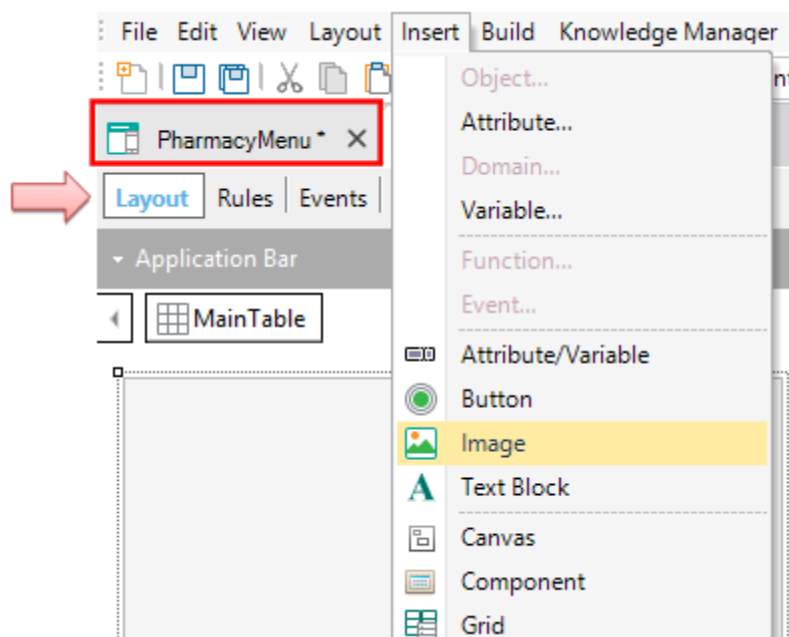
Como se explicó anteriormente, para crear un objeto solo tiene que seleccionar **File > New > Object** en la barra de herramientas:



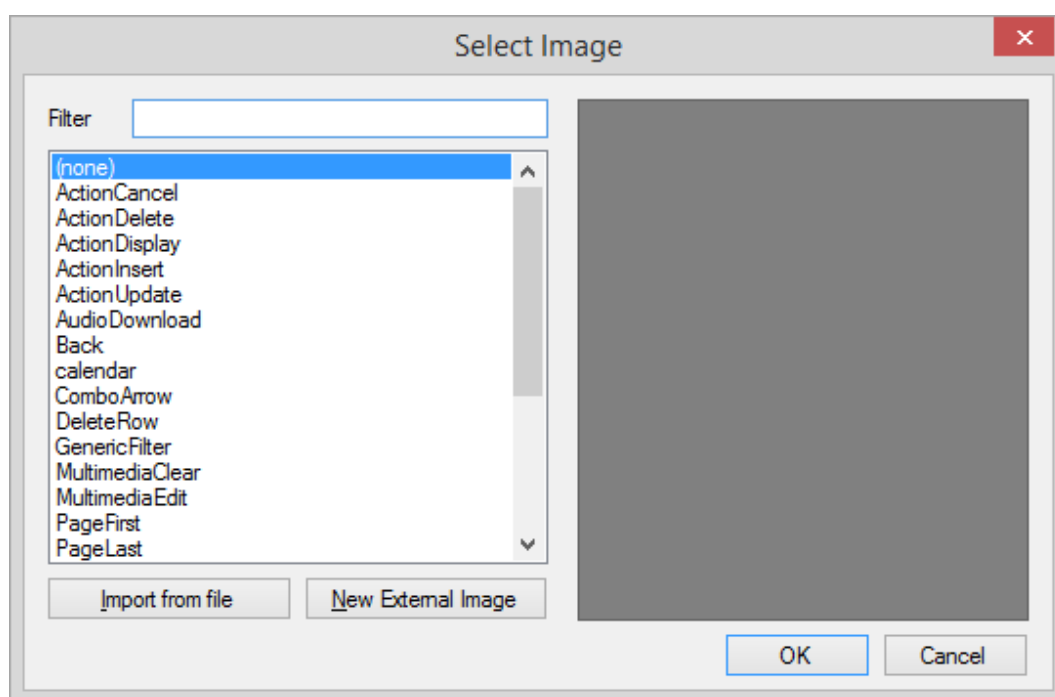
Una vez que haya creado el **Panel** (que en este ejemplo se le dio el nombre: PharmacyMenu), debe configurar su propiedad **Main program = True**, para que este objeto se convierta en un objeto ejecutable independiente del Developer Menu (es decir, es compilable y ejecutable por sí mismo).



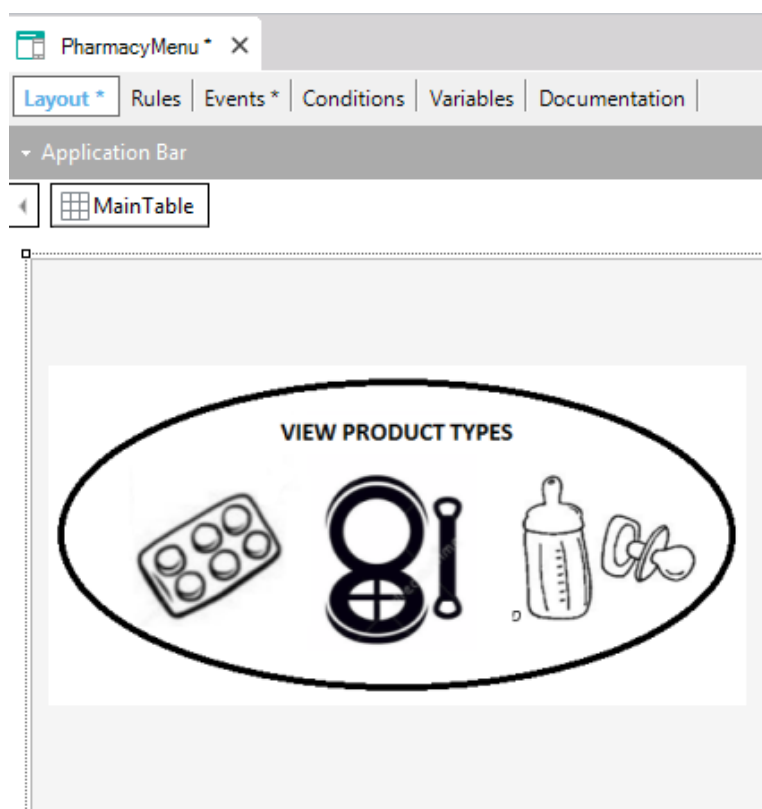
Ahora inserte un control de imagen dentro de su sección Layout:



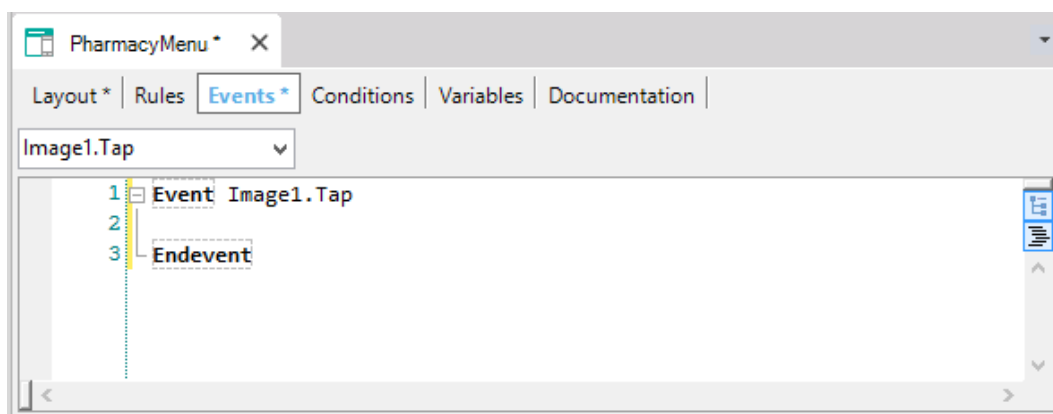
Cuando selecciona **Insert > Image**, se muestra la siguiente ventana. Si su foto está almacenada en un archivo, debe presionar el botón **Import from file** (Importar desde archivo).



Después de importar la imagen a la Base de Conocimiento e insertarla en el Layout del Panel **PharmacyMenu**, su diseño debería tener este aspecto:



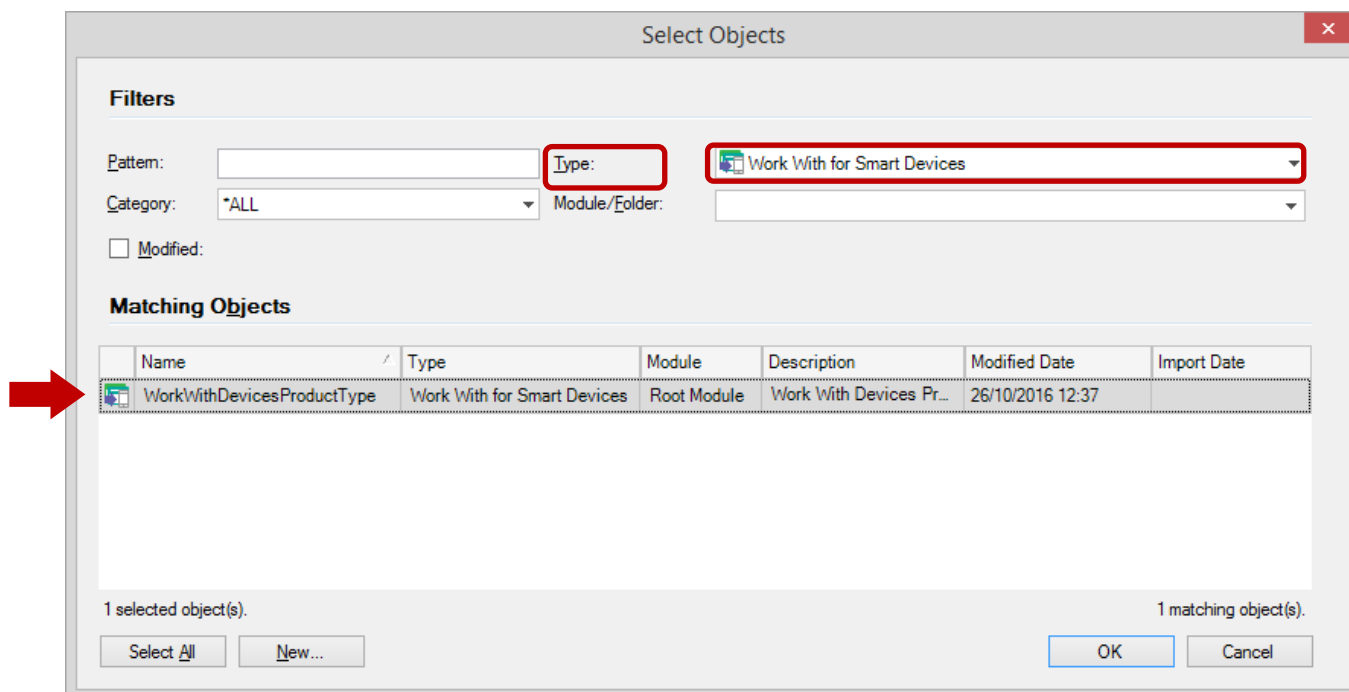
Al hacer doble clic en el control de imagen, se muestra su evento asociado por defecto:



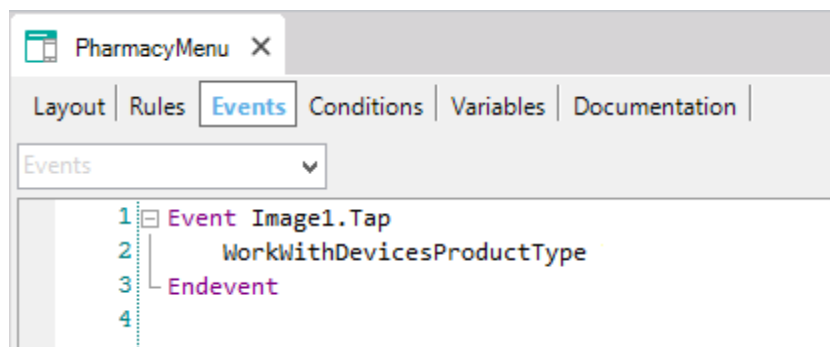
Dentro del evento Tap asociado a la imagen, se debe llamar al objeto `WorkWithDevicesProductType`; pero dicho objeto se compone de muchas partes (recuerde o mire nuevamente sus nodos). En este caso particular, se debe llamar al nodo **Lista** del objeto, ya que nuestro objetivo es mostrar la lista con todos los tipos de productos que ofrece la farmacia.

Para llevar a cabo esto, después de ubicar el cursor dentro del evento, seleccione **Insert > Object**.

Se muestra el siguiente diálogo:

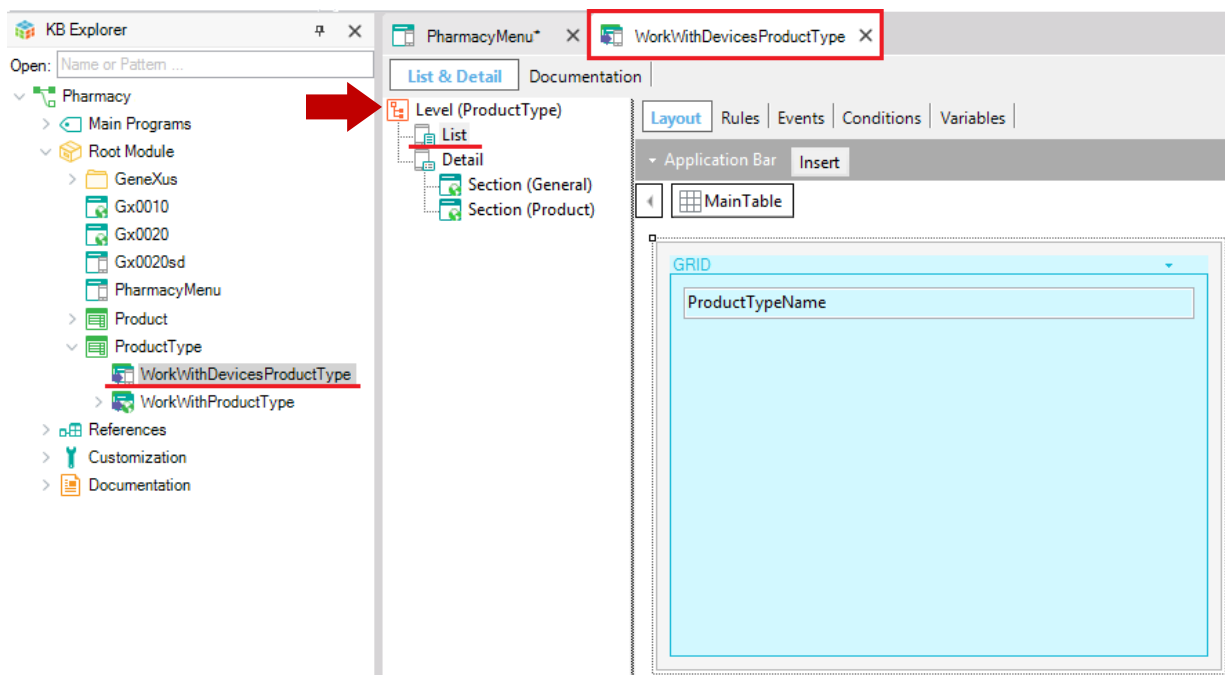


Filtre como se muestra arriba y seleccione el objeto *WorkWithDevicesProductType*:

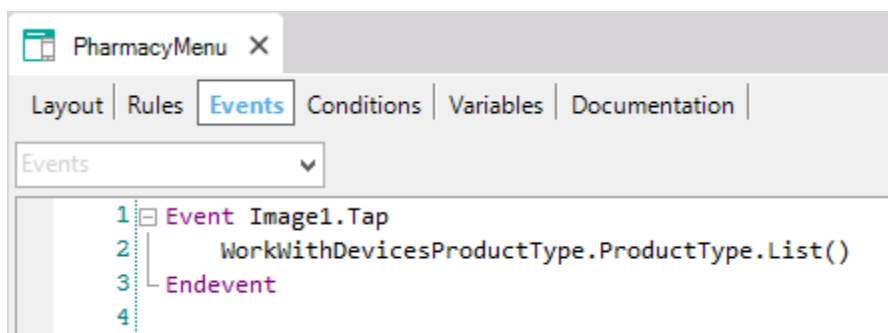


El nombre del objeto se ha insertado, por lo que ahora vamos a completar qué componente del objeto debe ejecutarse.

Para comprender lo que debe completar en la línea de código, revise el árbol de nodos que está dentro del objeto *WorkWithDevicesProductType*:

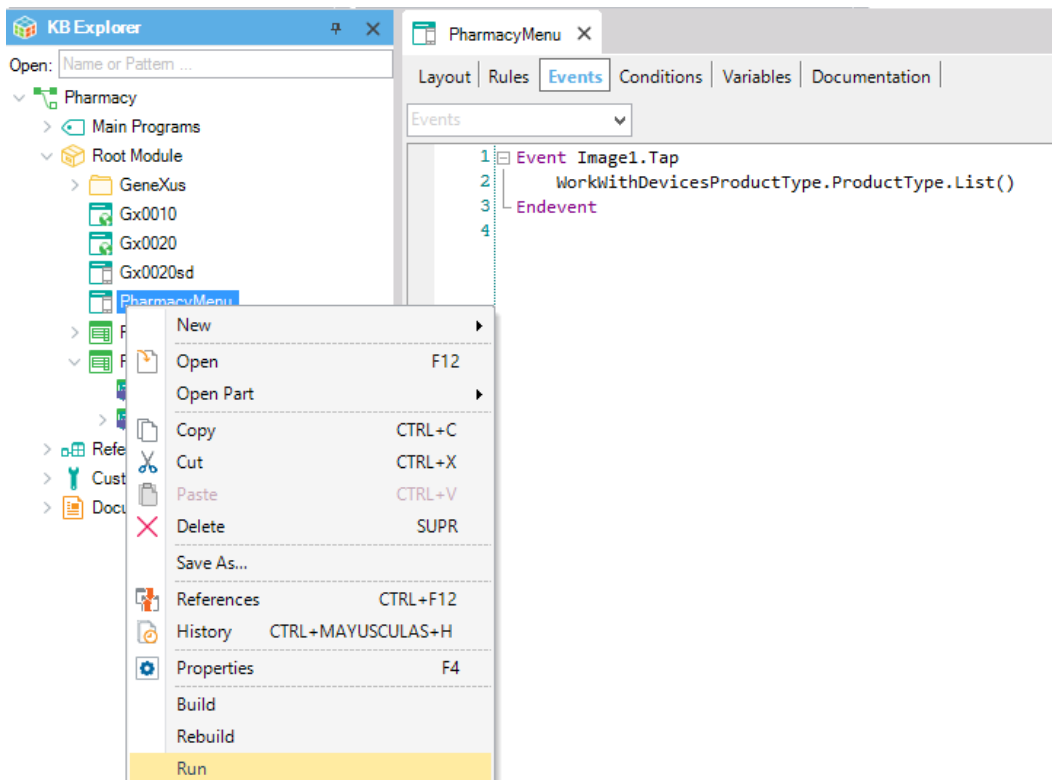


El objeto tiene un nodo principal, *ProductType*, y debajo de él puede encontrar los nodos List y Detail, respectivamente. Para llamar al nodo List, la sintaxis completa es:

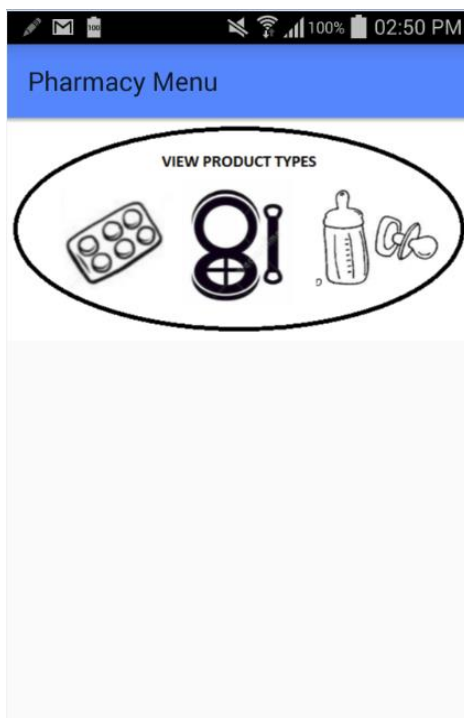


Ahora todo está definido y listo para ejecutar la aplicación para dispositivos móviles e inteligentes.

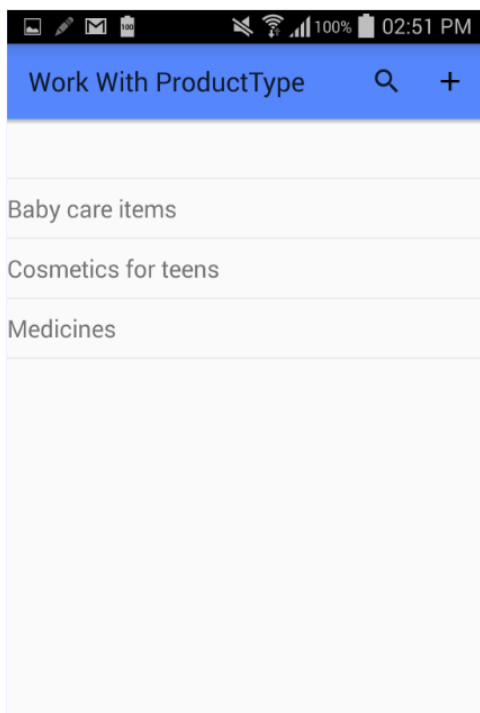
Dado que el objeto *PharmacyMenu* tiene su propiedad Main Program configurada en True, puede ejecutarlo independientemente del KB Explorer, haciendo clic con el botón derecho en el objeto y seleccionando **Run**:



Puede ejecutar en su computadora con un emulador de dispositivos móviles que se abrirá o directamente en su dispositivo inteligente si lo conectó a la computadora en la que está trabajando:

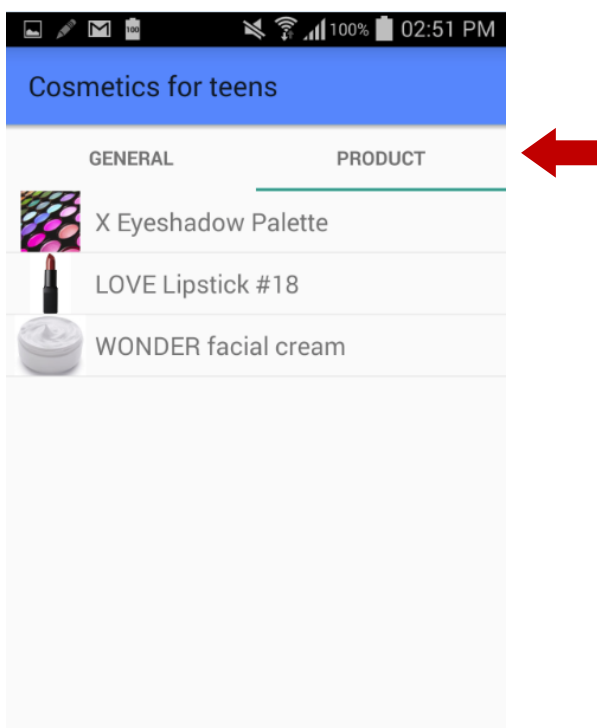


Cuando hace tap sobre la imagen, se muestra la lista de tipos de productos que ofrece la farmacia:



Tenga en cuenta que el botón Insert en la esquina superior derecha se puede eliminar fácilmente, ya que esta aplicación es para usuarios finales, y no queremos que puedan insertar nuevos productos; solo deben poder ver los diferentes tipos de productos, no editarlos.

Al tocar cada tipo de producto (por ejemplo: "Cosméticos para adolescentes"), se muestra el **Detalle**, junto con sus dos secciones GENERAL y PRODUCT:



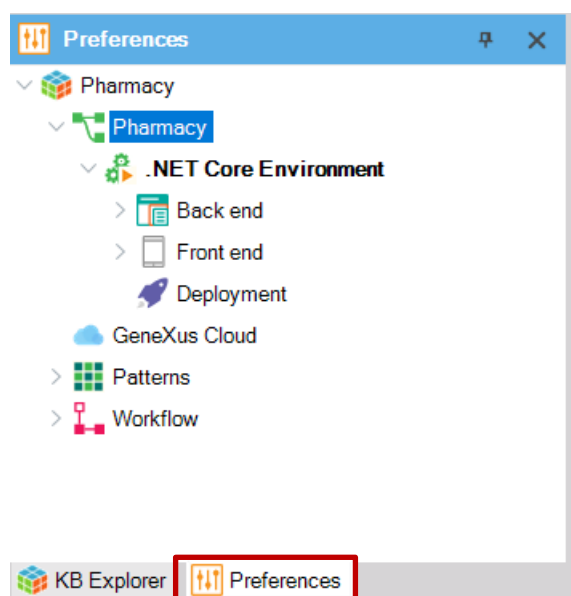
Obviamente, esto es solo una demostración muy simple ya que se pueden lograr aplicaciones mucho más sofisticadas.

¿QUÉ SUCEDE SI QUIERE GENERAR LO QUE HA DEFINIDO HASTA AHORA EN OTRO LENGUAJE Y / O PARA OTRA BASE DE DATOS?

Como se mencionó anteriormente, una de las grandes ventajas de GeneXus es que le permite generar la misma aplicación para diferentes plataformas, generar código en diferentes lenguajes de programación y/o almacenar los datos de la aplicación en diferentes bases de datos. Toda esta información se define en un **Environment**.

Un **Environment** le permite configurar y almacenar toda la información relacionada con una **determinada implementación** de su aplicación (los generadores que desea utilizar para generar el Back-end de su aplicación, los generadores que se utilizarán para generar el Front-end, la información de la base de datos, etc.).

Seleccione la ventana Preferences, haciendo clic en la pestaña que se encuentra al lado de la pestaña KB Explorer:

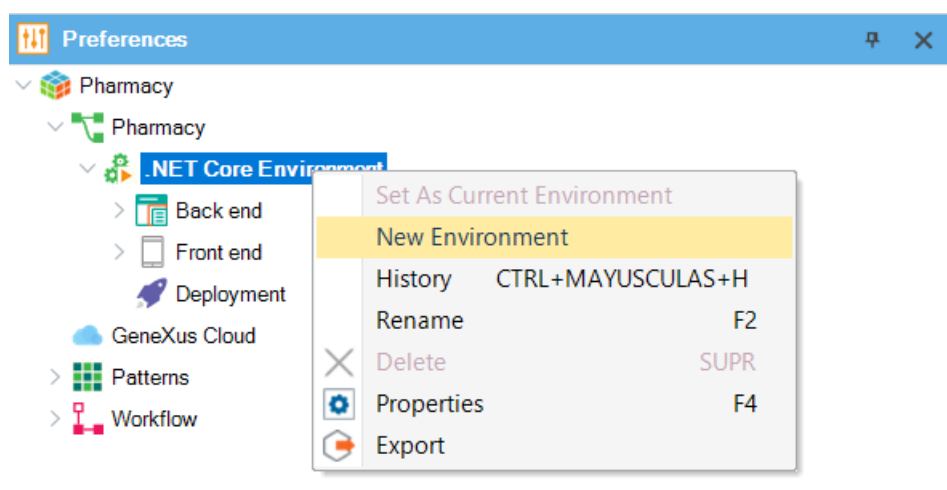


Puede observar que hay un único Environment definido (**.NET Core Environment**). Se creó automáticamente al momento de crear la Base de Conocimiento cuando eligió .NET Core como lenguaje de generación. Luego, se completó con más información cuando presionó F5 por primera vez e indicó la información relativa a la base de datos a crear y el servidor de base datos. Los subnodos del Environment permiten configurar más propiedades.

Es posible crear más de un Environment para la misma Base de Conocimiento. Por ejemplo, es común crear un Environment para desarrollo donde se conecta a una base de datos con datos de prueba y otro Environment para producción, donde se define el servidor y la base de datos que usará para el sistema terminado.

También es posible que desee crear, en la misma Base de Conocimiento, un nuevo Environment para generar todo para una plataforma diferente (por ejemplo, generar el código en Java y utilizar otro DBMS como podría ser Oracle, MySQL, PostgreSQL, DB2 u otros. Y para la generación del Front end podría usar Apple o Angular).

Como se muestra en la siguiente imagen, para crear un nuevo Environment, debe hacer clic con el botón derecho del mouse sobre el nodo de su Environment, seleccionar **New Environment** y luego configurar las propiedades necesarias.



Para elegir trabajar con un Environment u otro, debe hacer clic con el botón derecho sobre el Environment que desee activar y elegir **Set as current** (Establecer como actual).

Puede identificar fácilmente el Environment activo por el símbolo PLAY.

[Obtenga más información sobre cómo crear un nuevo entorno.](#)

¿QUÉ MÁS OFRECE GENEXUS?

- Acceso a bases de datos externas:

- Es posible que necesite acceder a bases de datos externas desde aplicaciones GeneXus. Por ejemplo, cargar datos de una base de datos externa a las tablas de la base de datos asociada con la Base de Conocimiento para realizar una carga inicial. A continuación, tal vez no necesite permanecer conectado a esa base de datos externa, o puede que necesite conectarse y estar siempre conectado a una determinada tabla o tablas de una o más bases de datos externas (no solo para leerlas, sino también para acceder y cambiar los datos en ellos). GeneXus ofrece un “proceso de ingeniería inversa” para conectarse a tablas de bases de datos externas con el fin de lograr las necesidades descritas anteriormente.

- Soporte de desarrollo colaborativo:

- **GeneXus Server** ofrece la opción de cargar una Base de Conocimiento en un servidor. A continuación, los nuevos desarrolladores pueden crear una copia local desde la Base de Conocimiento en el servidor, desde cualquier lugar geográfico, cuando surja la necesidad. Pueden trabajar (siempre localmente) y enviar sus cambios al servidor. Por supuesto, existe un mecanismo para la resolución de conflictos. Esta solución ofrece muchas ventajas, incluido el control de versiones de los modelos en la Base de Conocimiento.

- Consumir y definir servicios web:

- Es posible consumir servicios web desarrollados por terceros desde una aplicación GeneXus, así como desarrollar sus propios servicios web con GeneXus.

- Definir actualizaciones masivas de la base de datos y definir libremente otros tipos de procesos.

- Definir paneles interactivos y personalizados para aplicaciones web y aplicaciones para dispositivos móviles.

- Diseñar y ajustar interfaces de usuario (UIs)

- GeneXus ofrece potencia para personalizar la interfaz de usuario y, debido a que la experiencia del usuario es extremadamente importante, ofrece generadores específicos para aplicaciones nativas, aplicaciones con diseño web responsivo, web móvil, etc. También ofrece una función [Live Editing](#) de edición en vivo multiplataforma que simplifica el proceso de diseñar su aplicación y crear un prototipo en vivo (Live Prototyping).

- Implementar su aplicación en producción en servidores locales o proveedores de servicios en la nube

- o Al hacer clic en un botón puede implementar su aplicación en producción.

- Documentar dentro de la Base de Conocimiento

- o GeneXus proporciona un editor de Documentación de estilo Wiki, para que pueda describir fácilmente el propósito de la Base de Conocimiento (en un objeto del tipo de Documentación, llamado Main, que se crea automáticamente en toda Base de Conocimiento).

Además, todos los objetos GeneXus tienen una pestaña de Documentación, donde también puede describir el propósito del objeto.

Al escribir la documentación, puede incluir textos, imágenes, enlaces a atributos, objetos, etc. Los archivos también se pueden almacenar en la Base de Conocimiento como parte de su documentación.

- Inteligencia artificial

- o GeneXus proporciona capacidades para integrar fácilmente la [Inteligencia Artificial \(IA\)](#).

- Generador de chatbot

- o GeneXus incluye un [generador de Chatbot](#) para crear e implementar automáticamente un chatbot en cualquiera de los proveedores de Chatbot compatibles.

- Extensibilidad

- o GeneXus permite crear extensiones específicas, que permiten a los desarrolladores aprovechar diferentes lenguajes de plataforma para crear soluciones específicas y ampliar las capacidades centrales de GeneXus.

- Integración de sistemas externos y fuentes de datos en una aplicación GeneXus.

- o GeneXus ERP Connector para SAP hace posible el desarrollo de aplicaciones integradas a SAP ERP, lo que le permite complementar las funcionalidades que ofrece.

- Gestionar la seguridad

GeneXus ofrece un módulo de seguridad (totalmente integrado en GeneXus), llamado GeneXus Access Manager (GAM). Con solo habilitarlo, ofrece resolver las funcionalidades de autenticación y autorización, tanto para aplicaciones web como para dispositivos móviles y dispositivos inteligentes.

- Modelar y automatizar procesos de negocios

- o GeneXus cuenta con un conjunto de herramientas que permiten el modelado y la automatización de los procesos de negocios, así como un entorno de ejecución para administrarlos. La herramienta de modelado GeneXus Business Process Modeler se basa en el estándar BPMN 2.0 y está dirigida a usuarios cuyo objetivo es modelar procesos de negocios. Estos diagramas se pueden integrar o crear en el entorno de desarrollo de GeneXus para implementar la etapa de automatización, donde, mediante GeneXus, asociamos los diferentes objetos en cada tarea modelada en los procesos. GXflow ofrece las herramientas de ejecución, administración y monitoreo para los usuarios finales. De esta manera, GeneXus ofrece lo que conocemos como GeneXus BPM Suite, que es el conjunto de herramientas que permiten el desarrollo de sistemas basados en Business Process Management, es decir, sistemas orientados a procesos de negocio.

- Informes

- o Definición de informes estáticos (informes típicos que pueden imprimirse, guardarse o visualizarse en la pantalla).
- o Definición de consultas visuales y dinámicas.
 - Puede crear consultas a la base de datos, agrupar datos según uno o varios criterios, realizar cálculos y, finalmente, mostrar el resultado en diferentes tipos de gráficos, tablas de pivote y tablas. Para llevar a cabo este tipo de consultas, GeneXus ofrece el objeto Query y el control Query Viewer.
 - Además, el producto GXquery permite a los usuarios finales realizar consultas de forma dinámica, basándose en el mismo modelo de datos de la Base de conocimiento. Esta herramienta se enfoca en habilitar el acceso y análisis de datos en la base de datos operativa real del sistema, y le brinda al usuario una interfaz intuitiva desde la cual puede crear sus propias consultas y luego verlas a través de la interfaz web y la aplicación móvil, o integradas en Microsoft Office Excel.

- Desarrollo compartido y Marketplace

- o [GeneXus Marketplace](#) permite a los desarrolladores compartir sus controles de usuario, extensiones, patrones, herramientas externas y objetos externos creados para y con GeneXus.

- Test de aplicaciones con GXtest

- o Cuando se implementan nuevas funcionalidades o variaciones, es necesario verificar que lo que ya funcionó (antes de los cambios) continúe funcionando correctamente. Este tipo de tarea puede volverse muy tediosa si la aplicación crece mucho, ya que la cantidad de cosas a probar aumentará cada vez, etc. GeneXus ayuda a automatizar estas pruebas a través de su software **GXtest**, que le permite guardar secuencias de operaciones para probar. Luego,

las pruebas se reproducen automáticamente, verificando que el sistema aún funciona correctamente.

PRÓXIMOS PASOS

Ha llegado hasta acá familiarizándose con GeneXus, por lo que la pregunta natural es “¿Cuál es el próximo paso?”.

- **Primero lo primero:**
 - o Acceda al siguiente curso en línea para continuar aprendiendo:
<https://training.genexus.com/es/aprendiendo/cursos/genexus/v17/core>
 - o Si aún no ha probado GeneXus, puede hacerlo gratis en este link: <http://genexus.com/trial>

- **Aprenda más:** GeneXus es una plataforma de desarrollo muy completa, y hay mucho que leer y aprender. Podría comenzar a profundizar en:
 - o Sitio de Capacitación de GeneXus: <http://training.genexus.com/>
 - o GeneXus Wiki: <http://wiki.genexus.com/>

¡Obtenga GeneXus! Nos esforzamos para proporcionar la mejor herramienta con el fin de simplificar el desarrollo de software, por lo que recibimos a cada nuevo cliente como el primero. Póngase en contacto con nosotros a través de info@genexus.com o visite <http://genexus.com/plans> para ver cuál le conviene más.

- **Sea parte de nuestra comunidad:** cuando esté listo, puede unirse a nuestra comunidad en constante crecimiento a través de una gran variedad de opciones.
 - o Publique su trabajo en nuestro Marketplace: <http://marketplace.genexus.com/>
 - o Trabaje con GeneXus: <http://genexus.com/company/work-with-us?en>
 - o Oportunidades ofrecidas por nuestros socios:
<http://genexus.com/jobs/Opportunities?en>
 - o Sea parte de la GeneXus Alliance: <http://genexus.com/partners>
 - o Asista al próximo Encuentro GeneXus:: <http://genexus.com/meetings>

We really hope we hear from you soon!!

The GeneXus Team

GeneXus™
www.genexus.com

MONTEVIDEO - URUGUAY
CIUDAD DE MÉXICO - MÉXICO
MIAMI - USA
SÃO PAULO - BRASIL
TOKYO - JAPAN

Av. Italia 6201- Edif. Los Pinos, P1
Hegel N° 221, Piso 2, Polanco V Secc.
7300 N Kendall Drive, Suite 470
Rua Samuel Morse 120 Conj. 141
2-27-3, Nishi-Gotanda
Shinagawa-ku, Tokyo, 141-0031

(598) 2601 2082
(52) 55 5255 4733
(1) 201 603 2022
(55) 11 4858 0300
(81) 3 6303 9381
(81) 3 6303 9980