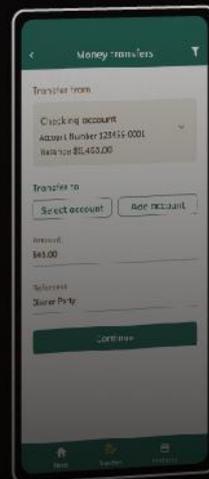
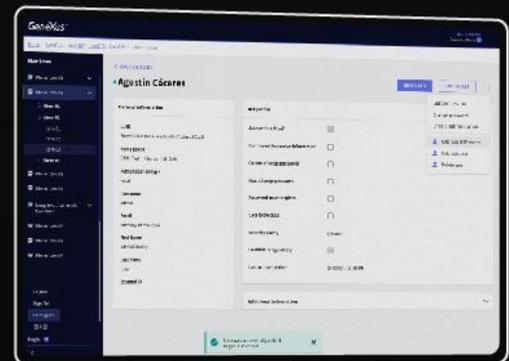


# Primeros pasos con GeneXus™

Crea tu primera aplicación sin saber codificar.



# Tabla de Contenidos

INTRODUCCIÓN	3
COMENZAR A UTILIZAR GENEXUS	3
CREACIÓN DE UNA NUEVA APLICACIÓN CON GENEXUS	4
DEFINICIÓN DE LOS PRIMEROS OBJETOS	6
GENERACIÓN Y EJECUCIÓN DE LA APLICACIÓN POR PRIMERA VEZ.	17
LA APLICACIÓN CRECE	25
AGREGAR REGLAS DE NEGOCIOS	28
DEFINIR CÁLCULOS: FÓRMULAS	31
USO DE PATRONES (PARA WEB Y PARA DISPOSITIVOS MÓVILES)	34
GENERACIÓN DE LO DEFINIDO HASTA AHORA EN OTRO LENGUAJE Y/O PARA UNA BASE DE DATOS DIFERENTE	57
¿QUÉ MÁS OFRECE GENEXUS?	59
PRÓXIMOS PASOS	62

## INTRODUCCIÓN

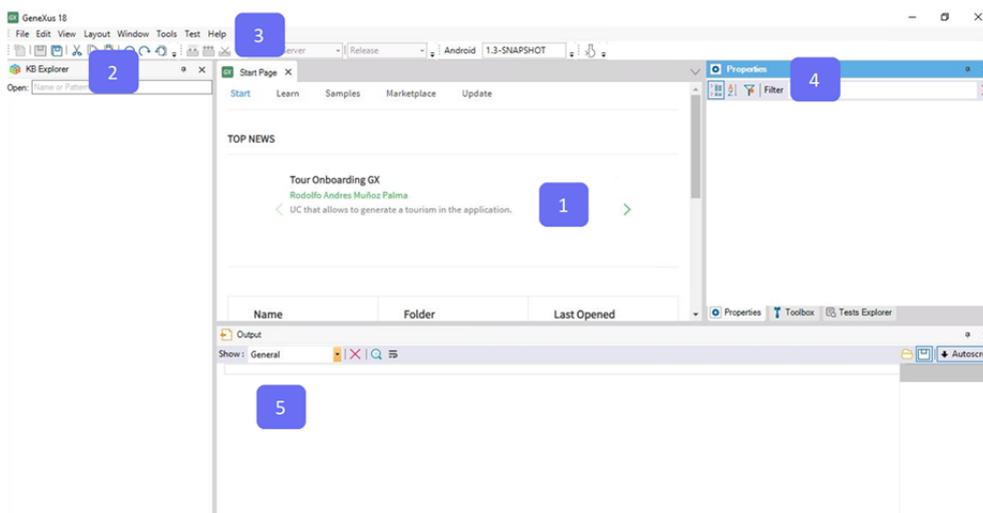
GeneXus es una suite de desarrollo de código de bajo nivel que permite la generación rápida de aplicaciones de software en múltiples lenguajes y plataformas. GeneXus ofrece varias ventajas: es fácil de aprender, altamente productivo, multiplataforma y preparado para el futuro, de forma que protege tus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

Por ejemplo, GeneXus genera aplicaciones para la Web y / o dispositivos móviles (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, verás 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 el desarrollador se encuentra posicionado (como un objeto seleccionado, atributo, variable, control, etc.).

5. **Salida:** muestra la salida de las acciones realizadas.

## CREACIÓN DE UNA NUEVA APLICACIÓN CON GENEXUS

Para comenzar a desarrollar una nueva aplicación con GeneXus, debes 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 te mostrará el siguiente cuadro de diálogo:

The screenshot shows the 'New Knowledge Base' dialog box. The 'Name' field is set to 'Pharmacy' and the 'Location' field is set to 'C:\KBs'. The 'Prototyping Target' is set to 'Local' and the 'User Interface Language' is set to 'English'. Under the 'Back end' section, the 'Prototyping Environment' is set to '.NET' and the 'Data Source' is set to 'SQL Server'. Under the 'Front end' section, the 'Web (.NET)' option is selected. The 'Create' button is highlighted with a blue border.

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, debes ingresar la ruta donde deseas crear esa Base de Conocimiento.

El siguiente paso consiste en seleccionar si deseas que los programas de prototipado a ser generados por GeneXus queden en tu máquina local o en la Nube de GeneXus. Selecciona "GeneXus Prototyping Cloud" en el combo box **Prototyping Target**.

**New Knowledge Base**

Name: Pharmacy

Location: C:\KBs

Basic | Advanced

Prototyping Target: GeneXus Prototyping Cloud

User Interface Language: English

Back end

Prototyping Environment: .NET

Data Source: (Default)

Front end

- Web (.NET)
- Android
- Apple
- Web (Angular)

Create Cancel

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

En el cuadro **Back end** debes indicar:

- Uno de los lenguajes de programación disponibles en el combo llamado Prototyping Environment. GeneXus utilizará el lenguaje que selecciones para generar los programas de aplicación del back end, así como los programas necesarios para crear y mantener la base de datos. Por defecto, el lenguaje seleccionado es .NET. Déjalo así.
- El DBMS sobre el cual se creará, accederá y mantendrá la base de datos. El combo Data Source ofrece SQL Server de forma predeterminada. Déjalo así. Más adelante, se deberán ingresar los detalles de la base de datos.

El cuadro **Front end** te permite indicar los lenguajes de programación deseados para que GeneXus genere el front end de las aplicaciones. Por defecto, el lenguaje seleccionado para generar el front end de las aplicaciones Web es .NET. También se pueden seleccionar otros para generar aplicaciones Web y/o Móviles. En este caso, utiliza .NET.

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

## DEFINICIÓN DE LOS PRIMEROS OBJETOS

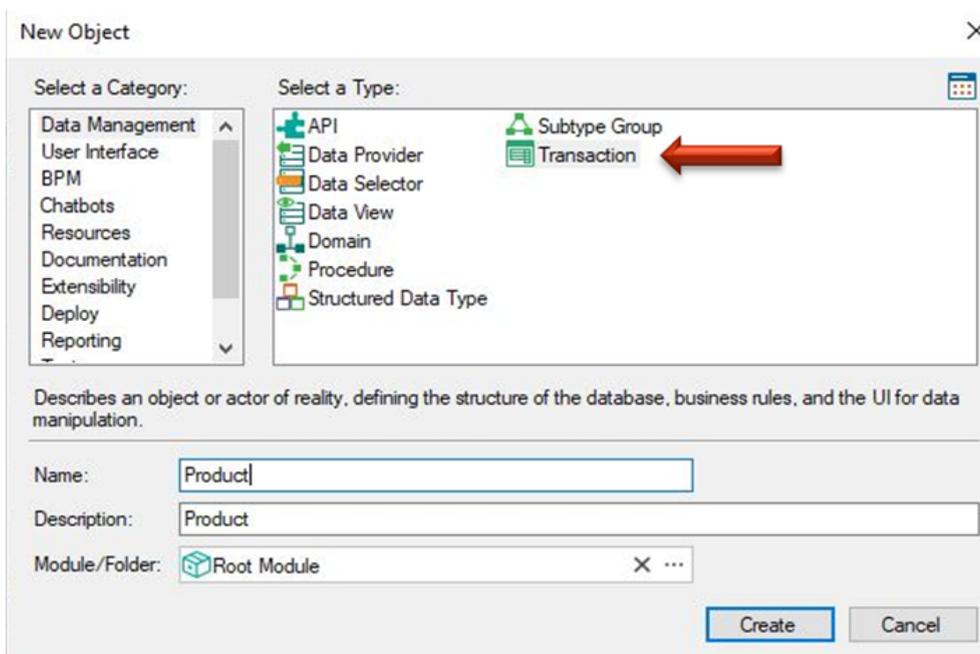
Una vez que hayas creado una nueva Base de Conocimiento, deberás describir las visiones de los usuarios. Para esto, es necesario que identifiques los objetos de la realidad (recomendamos prestar atención a los sustantivos que los usuarios mencionan en sus descripciones, tales como: productos, facturas, 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**.

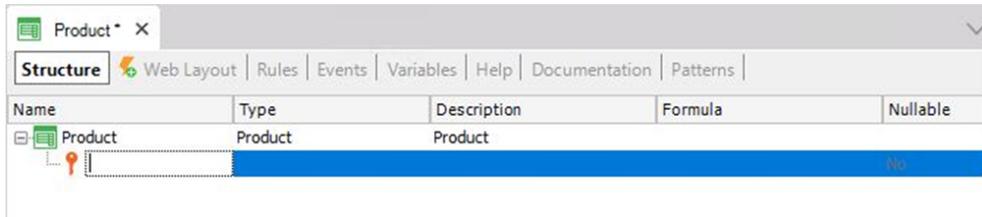
Supón que la farmacia que solicita la aplicación desea poder registrar los Productos que tiene a la venta.

Para describir cada objeto de la realidad identificado, debes crear un objeto GeneXus de tipo Transacción (no está relacionado con el concepto que se refiere a transacciones de la base de datos). Para esto, crea un objeto Transacción que describa los Productos.

Selecciona **File > New > Object** en la barra de herramientas. Se te mostrará el siguiente cuadro de diálogo permitiéndote seleccionar el tipo de objeto que deseas crear (en este caso: Transacción) e ingresar un nombre para él (en este caso: *Product*).



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



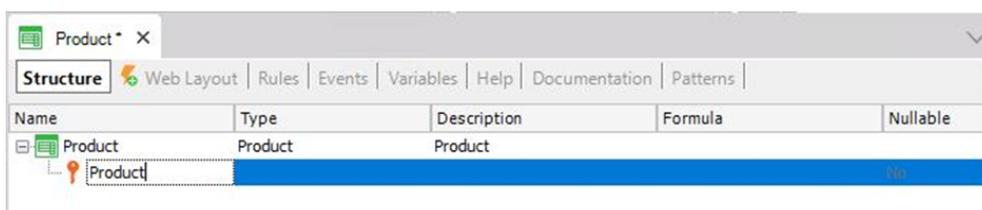
Cada Transacción tiene algunas secciones que se explicarán gradualmente. Concretamente, la estructura (Structure) de una Transacción te permite definir los atributos o campos que describen un objeto de la realidad.

Supón que en la farmacia te transmitieron que para cada producto se necesita 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 debes crear para esta Transacción.

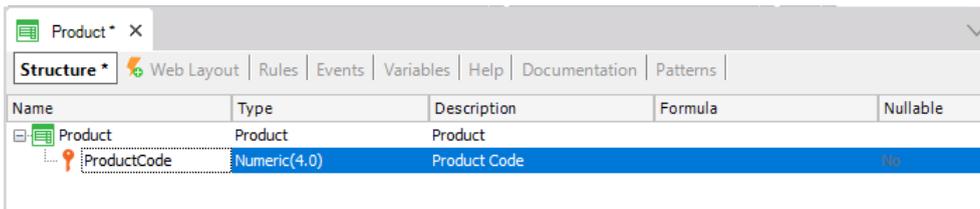
Observa en la imagen de arriba, que la primera línea en la estructura de la Transacción automáticamente queda lista para que definas el primer atributo. Además, observa que un ícono con la imagen de una llave está asociado con esta línea. Esto se debe a que en cada Transacción, un atributo, o conjunto 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 clave primaria de la Transacción *Product* es el código del producto. A continuación, se indica cómo definirlo.

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

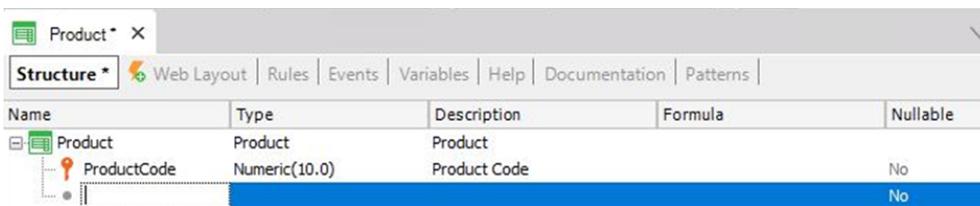


Solo te resta escribir *Code* después del prefijo *Product*:



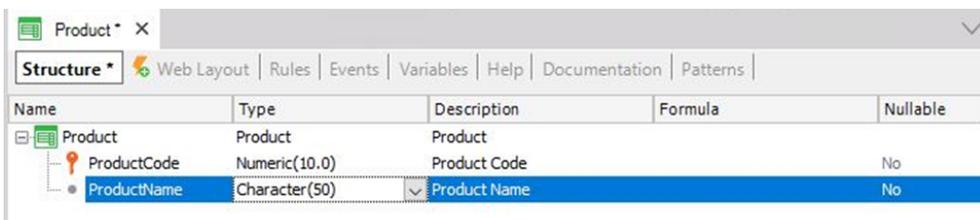
Name	Type	Description	Formula	Nullable
Product	Product	Product		
ProductCode	Numeric(4.0)	Product Code		No

Luego, al presionar la tecla Tab, puedes 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 tienes que cambiar su longitud a 10:



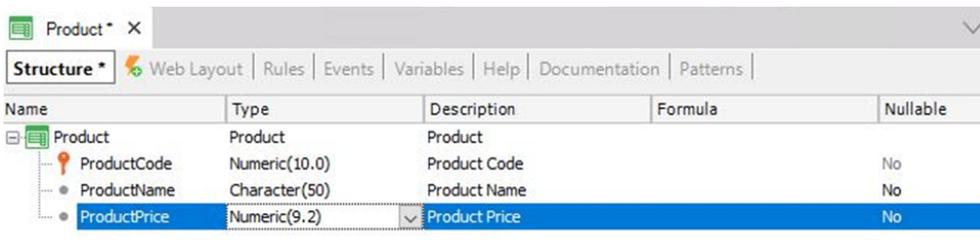
Name	Type	Description	Formula	Nullable
Product	Product	Product		
ProductCode	Numeric(10.0)	Product Code		No
				No

Al presionar Enter, se te abre una nueva línea donde puedes comenzar a definir el segundo atributo. Nuevamente, debes 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):



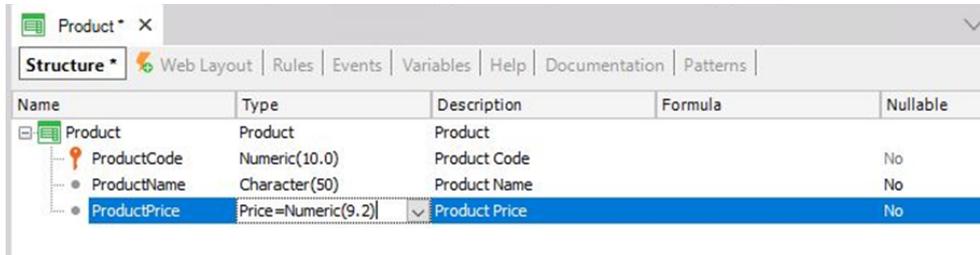
Name	Type	Description	Formula	Nullable
Product	Product	Product		
ProductCode	Numeric(10.0)	Product Code		No
ProductName	Character(50)	Product Name		No

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

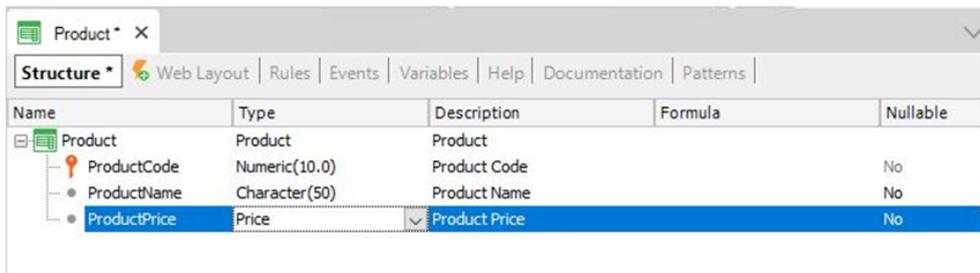


Name	Type	Description	Formula	Nullable
Product	Product	Product		
ProductCode	Numeric(10.0)	Product Code		No
ProductName	Character(50)	Product Name		No
ProductPrice	Numeric(9.2)	Product Price		No

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



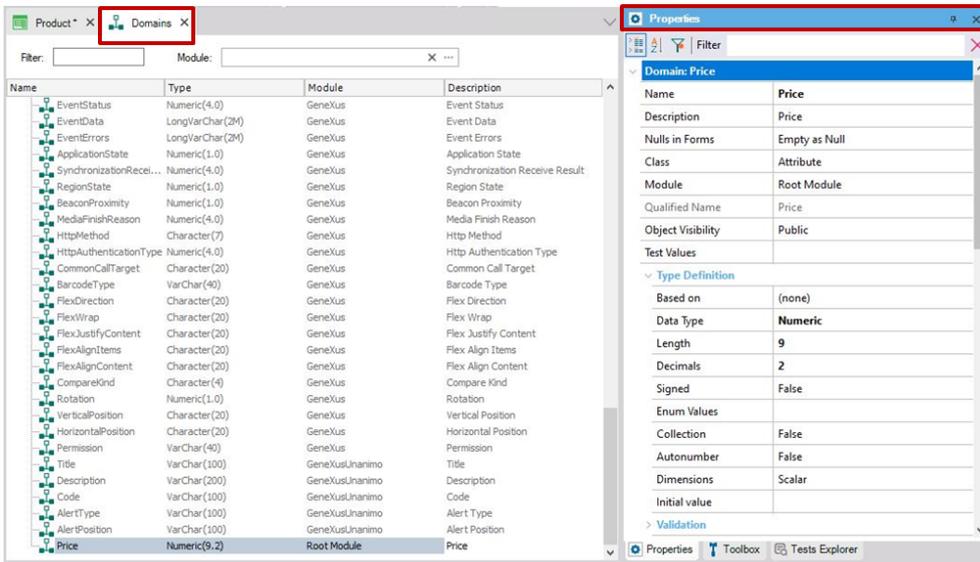
Luego presiona Enter, y verás que el atributo *ProductPrice* se habrá establecido del tipo *Price*:



La definición de *Price* 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 necesitas que los precios sean numéricos de una longitud diferente, solo deberás cambiar la definición del dominio para actualizar todos los atributos basados en ese dominio en un solo paso.

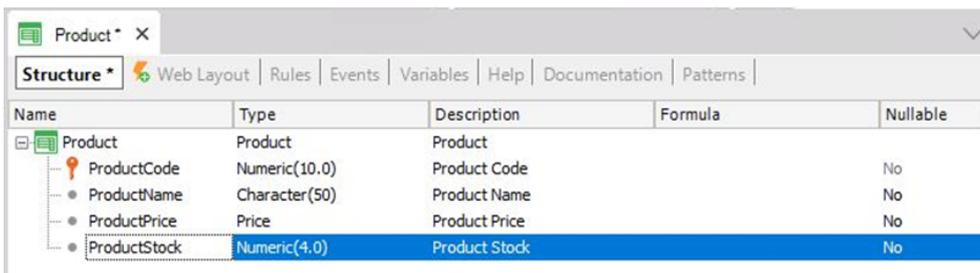
Para ver los dominios creados en la Base de Conocimiento, selecciona **View > Domains** en la Barra de herramientas:



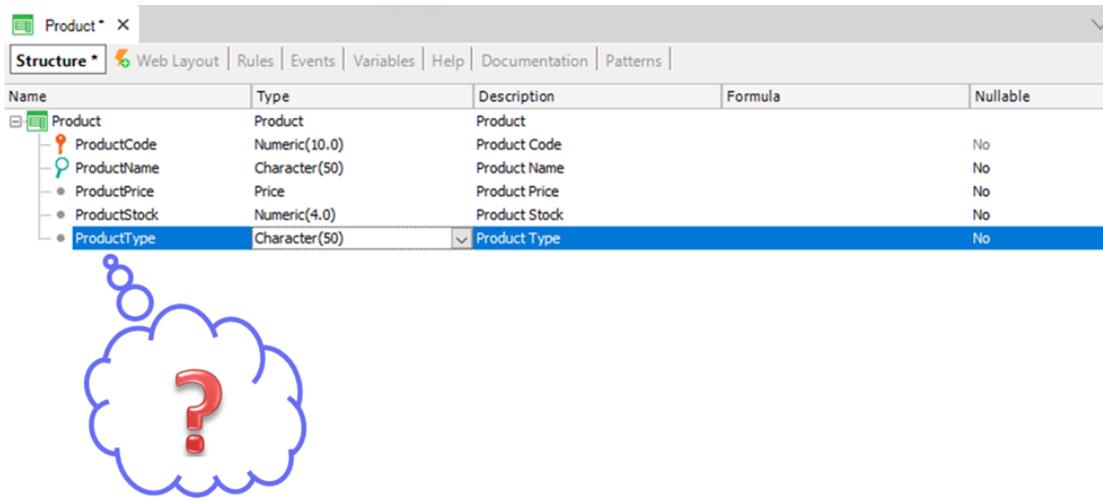
Como puedes observar en la imagen, GeneXus crea algunos dominios automáticamente. Al hacer clic en un determinado dominio, la ventana de propiedades se actualiza mostrando las propiedades del dominio seleccionado. Ten en cuenta que, además de configurar el tipo de datos para un dominio, también puedes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

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

Regresa a la Transacción *Product*, donde el siguiente atributo que debes definir es *ProductStock* de tipo numérico, y longitud 4:

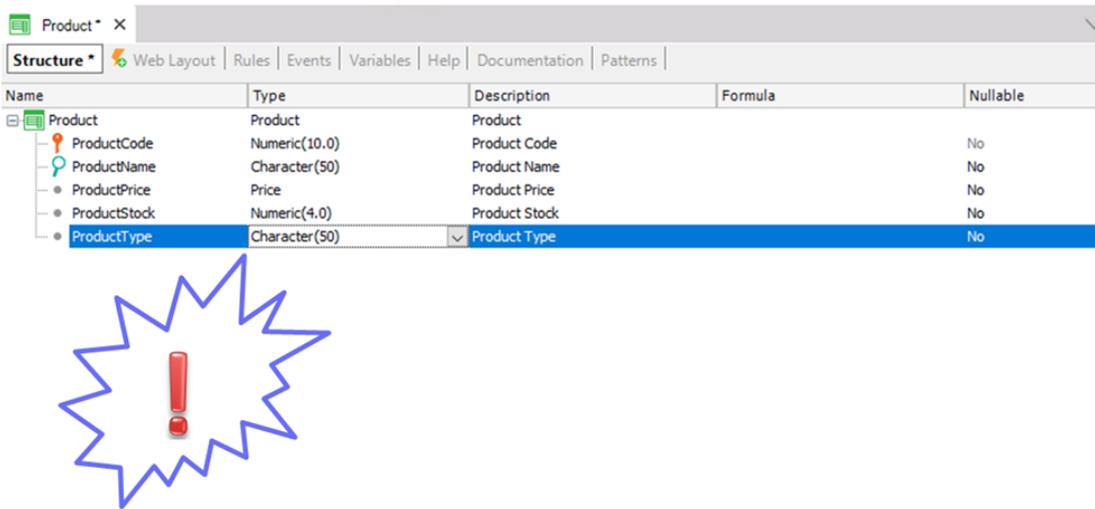


Ahora es necesario registrar el tipo de producto. Podrías crear un atributo llamado *ProductType* como Character(50) ...



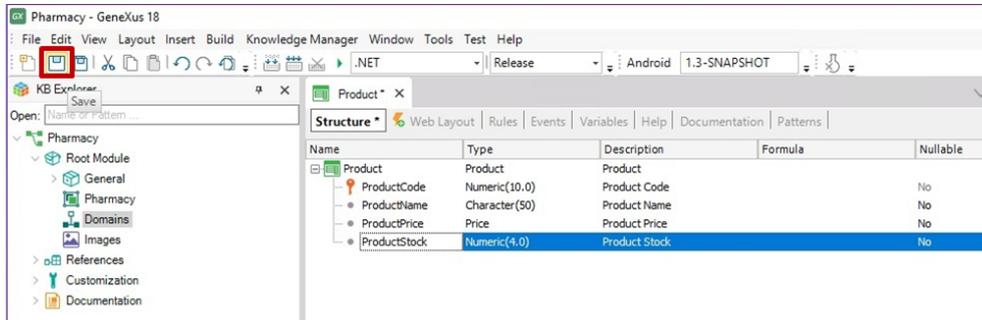
Name	Type	Description	Formula	Nullable
Product	Product	Product		
ProductCode	Numeric(10.0)	Product Code		No
ProductName	Character(50)	Product Name		No
ProductPrice	Price	Product Price		No
ProductStock	Numeric(4.0)	Product Stock		No
ProductType	Character(50)	Product Type		No

Pero, ¿qué pasaría 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.

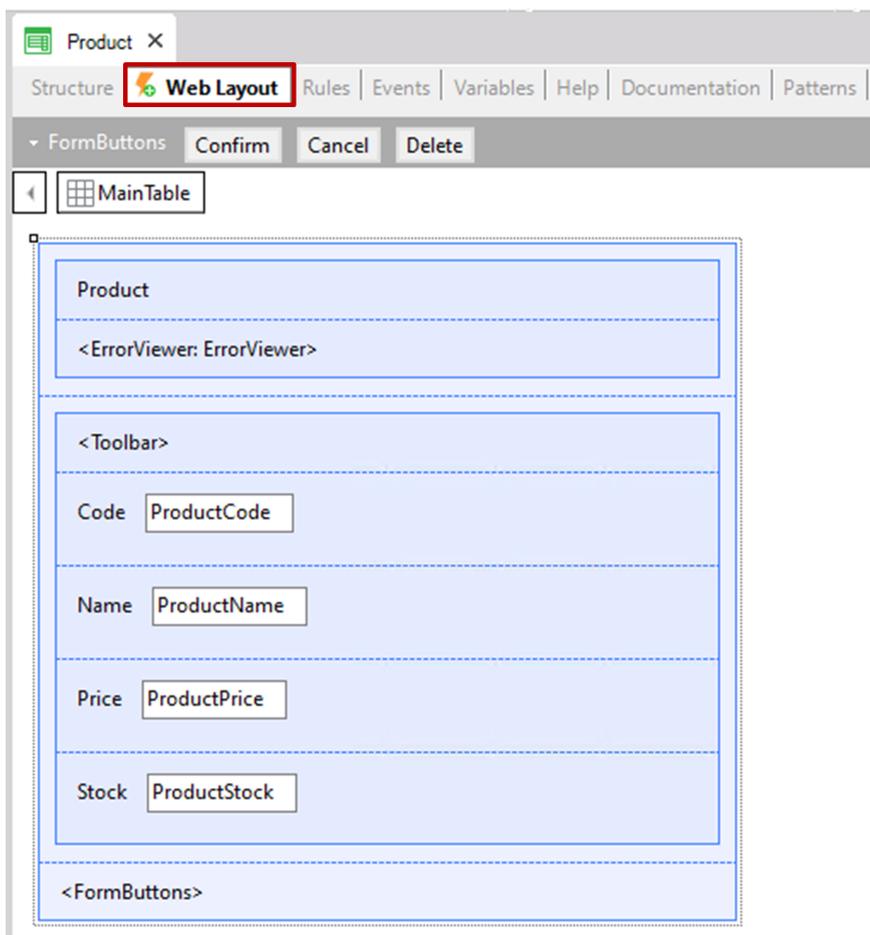


Name	Type	Description	Formula	Nullable
Product	Product	Product		
ProductCode	Numeric(10.0)	Product Code		No
ProductName	Character(50)	Product Name		No
ProductPrice	Price	Product Price		No
ProductStock	Numeric(4.0)	Product Stock		No
ProductType	Character(50)	Product Type		No

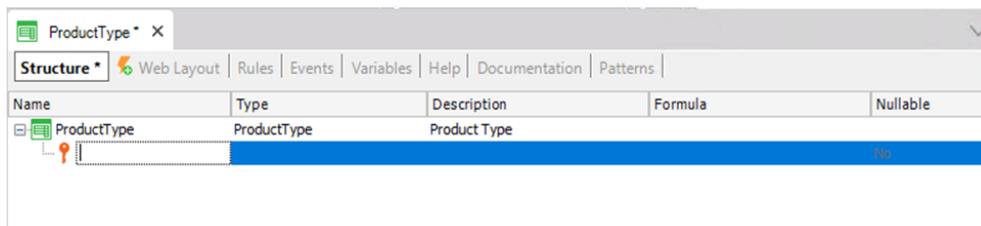
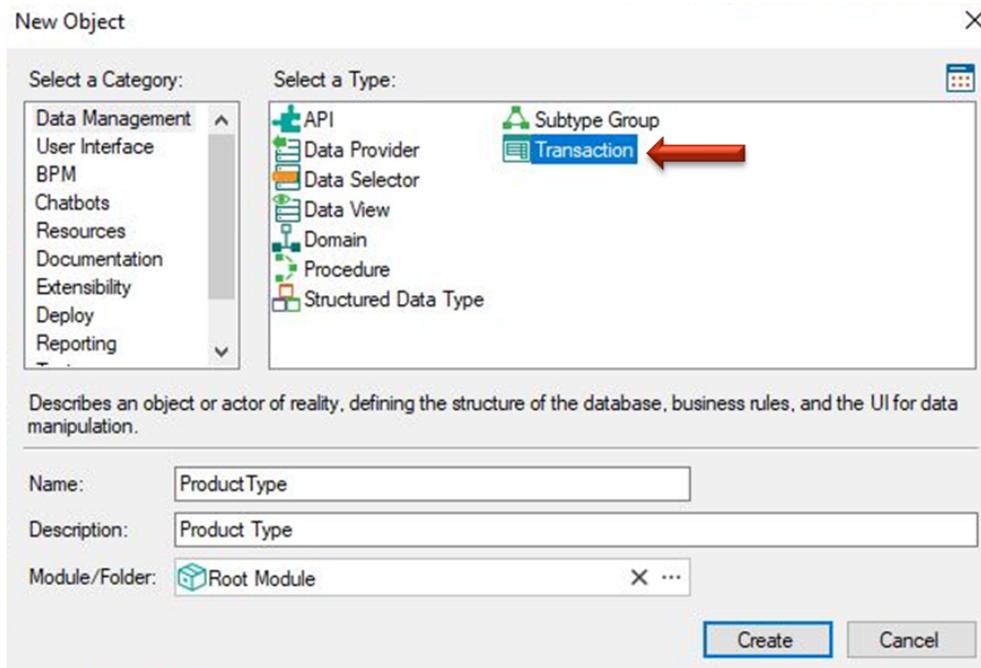
Parece más razonable ingresar el tipo una sola vez, en un solo lugar, y de ahí en adelante, para cada producto, asignar el tipo de producto correspondiente. Por lo tanto, hay que eliminar el atributo ProductType de la estructura y guardar la Transacción Product como se muestra:



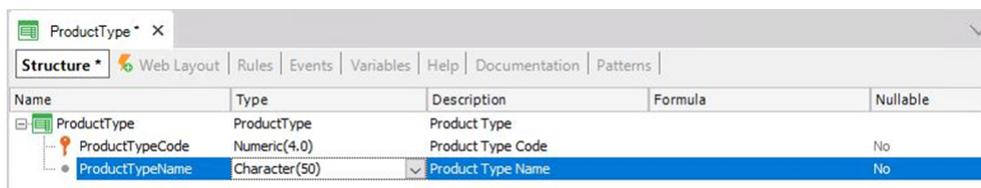
Para cada Transacción, GeneXus diseña automáticamente una interfaz Web (Web Layout) a partir de la estructura (Structure) definida. Esta interfaz Web permitirá a los usuarios de la aplicación agregar, actualizar y eliminar productos en tiempo de ejecución:



Ahora procede a crear otra Transacción para registrar los tipos de productos, para luego asignar un tipo de producto a cada uno de los productos. Para esto, selecciona nuevamente **File > New > Object** y completa lo siguiente:



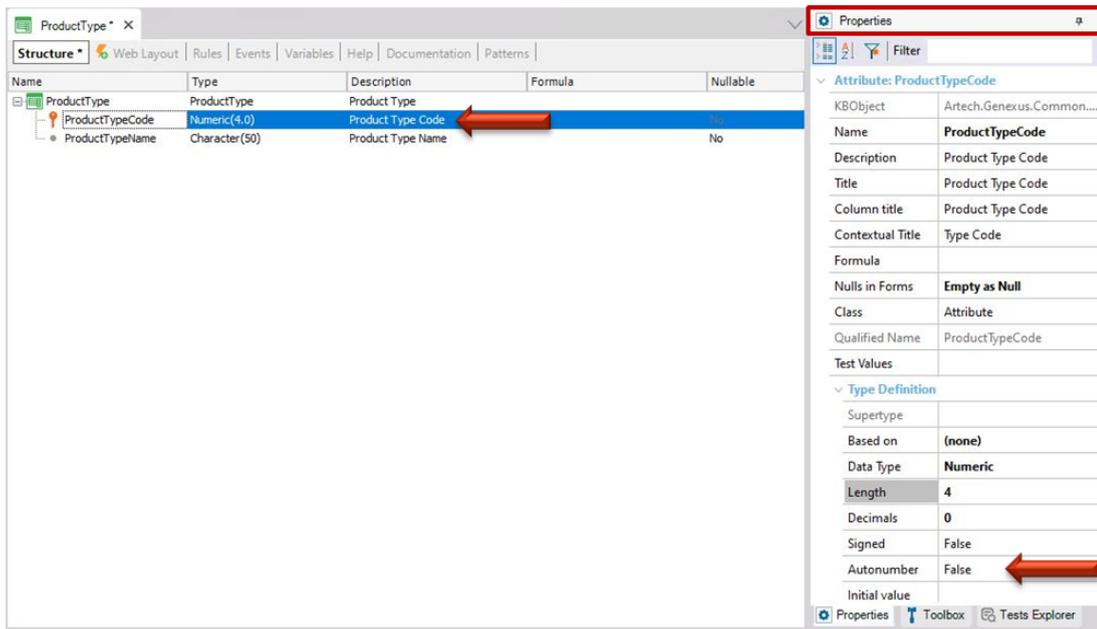
Para cada tipo de producto hay que registrar su código y nombre de tipo de producto, así que debes crear los dos atributos que se muestran:



Recuerda 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.

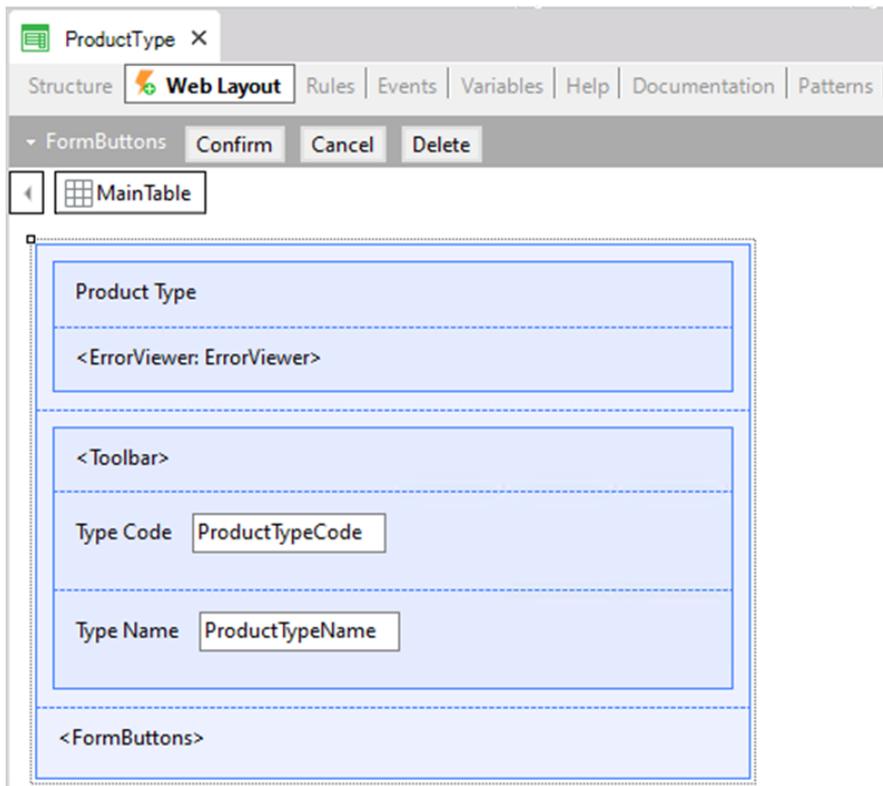
Nombrar atributos con el nombre de Transacción como prefijo no solo permite definir atributos de manera más fácil y rápida. También es una convención de la comunidad GeneXus para facilitar la comprensión al leer un nombre de atributo donde sea que esté, para saber a qué objeto se está describiendo.

Observa las propiedades del atributo *ProductTypeCode*:

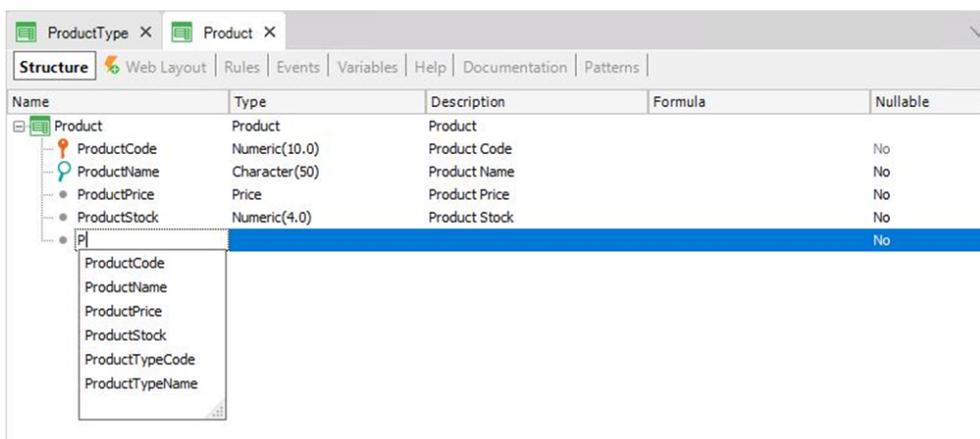


Observa la propiedad Autnumber. Está configurada con valor False por defecto. Al cambiarla a True, todos los nuevos tipos de productos ingresados por el usuario final se numerarán automáticamente en forma consecutiva. Por lo tanto, configura la propiedad Autnumber en True para este atributo identificador y guarda 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*:



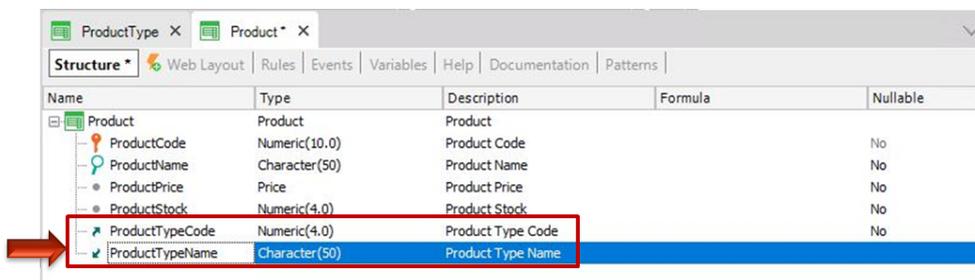
Ahora, asigna un tipo de producto a cada producto. Para ello, vuelve a la Transacción *Product*, agrega una nueva línea en su estructura y escribe la letra "P". Se te mostrará la lista de atributos existente en la Base de Conocimiento que comienza con esa letra:



Selecciona *ProductTypeCode* y se te mostrará la definición completa del atributo.

En esta Transacción también debes incluir 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.

Concéntrate 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 clave primaria de esa Transacción). Entonces, cuando se incluye una clave primaria en otra Transacción, GeneXus entiende que allí el atributo tiene el papel de clave foránea.

Incluir un atributo que es una clave 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. Recuerda que la Transacción es el objeto GeneXus que creas 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.

Durante la ejecución de la interfaz Web de la Transacción *Product*, el usuario deberá ingresar para el atributo *ProductTypeCode* (que allí es un atributo clave foránea) un valor que haya sido registrado previamente a través de la Transacción *ProductType*. De lo contrario, se le mostrará un error. Al ingresar un *ProductTypeCode* válido, su valor *ProductTypeName* se obtendrá y se mostrará en la pantalla.

## GENERACIÓN Y EJECUCIÓN DE LA APLICACIÓN POR PRIMERA VEZ.

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



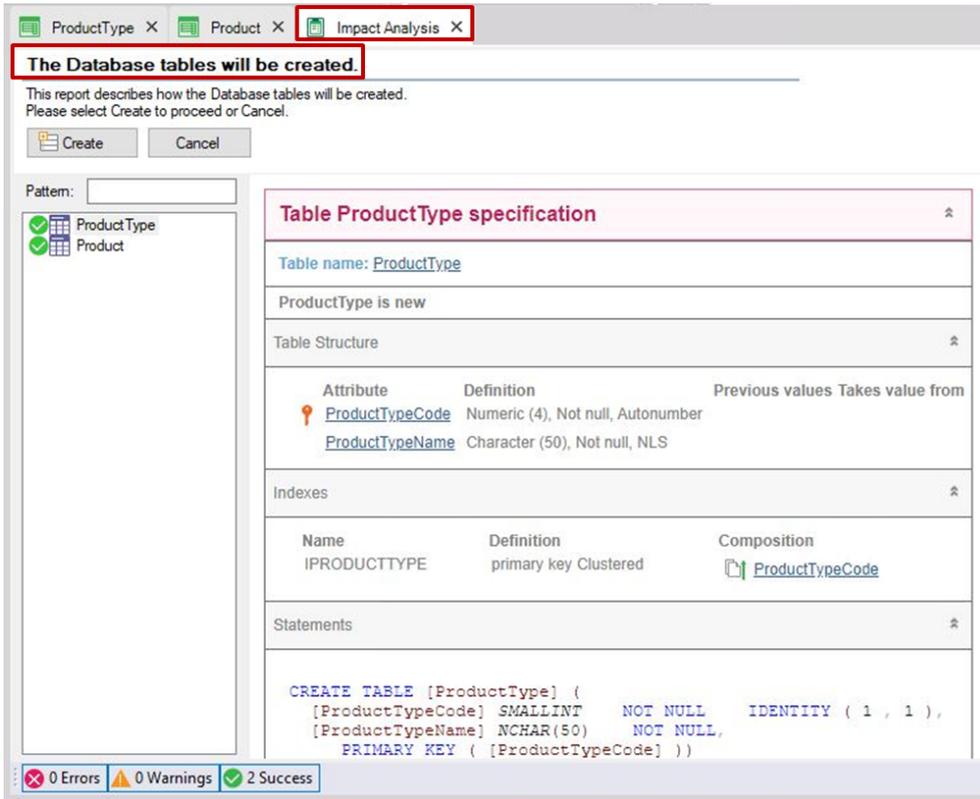
The image shows a Windows-style dialog box titled "GeneXus Account Authentication". It features a blue header bar with a key icon on the left and a question mark and close button on the right. Below the header, the text "Enter your GeneXus Account information." is displayed. There are two input fields: "User name:" with a dropdown menu containing a person icon, and "Password:" with a text box. Below the password field is a checkbox labeled "Remember my password". At the bottom, there are two buttons: "OK" and "Cancel".

Utiliza las credenciales de tu cuenta GeneXus para completar los siguientes campos:

- *User name or email* correspondiente a tu cuenta GeneXus.
- *Password* correspondiente a tu cuenta GeneXus.

Después, presiona el botón OK.

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)**:



Este informe muestra qué cambios estructurales deben realizarse en la base de datos. Observa 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 (*ProductType* y *Product* en la ventana izquierda), verás en la ventana de la derecha los atributos que se incluirán en ellos:

The screenshot shows a software interface with three tabs: ProductType, Product, and Impact Analysis. The main window displays a report titled "The Database tables will be created." with a sub-header "This report describes how the Database tables will be created. Please select Create to proceed or Cancel." Below this are "Create" and "Cancel" buttons. On the left, a "Pattern:" field is empty, and a tree view shows "ProductType" and "Product" with checkmarks. An orange arrow points to "ProductType". The right pane shows the "Table ProductType specification" with the following details:

- Table name: [ProductType](#)
- ProductType is new
- Table Structure:
 

Attribute	Definition	Previous values	Takes value from
<a href="#">ProductTypeCode</a>	Numeric (4), Not null, Autonumber		
<a href="#">ProductTypeName</a>	Character (50), Not null, NLS		
- Indexes:
 

Name	Definition	Composition
IPRODUCTTYPE	primary key Clustered	<a href="#">ProductTypeCode</a>
- Statements:
 

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

At the bottom, a status bar shows "0 Errors", "0 Warnings", and "2 Success".

The screenshot shows the same software interface, but the tree view on the left has "Product" selected with an orange arrow. The right pane shows the "Table Product specification" with the following details:

- Table name: [Product](#)
- Product is new
- Table Structure:
 

Attribute	Definition	Previous values
<a href="#">ProductCode</a>	Numeric (10), Not null	
<a href="#">ProductName</a>	Character (50), Not null, NLS	
<a href="#">ProductPrice</a>	Numeric (9.2), Not null	
<a href="#">ProductStock</a>	Numeric (4), Not null	
<a href="#">ProductTypeCode</a>	Numeric (4), Not null	
- Indexes:
 

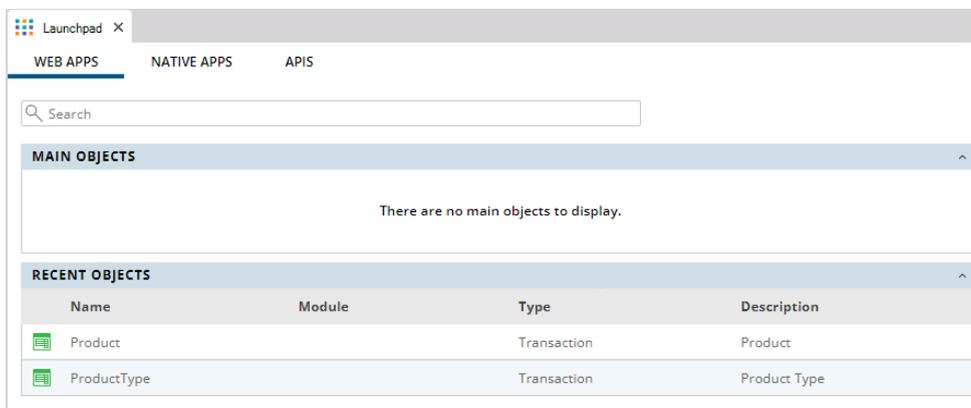
Name	Definition	Composition
IPRODUCT	primary key Clustered	<a href="#">ProductCode</a>
IPRODUCT1	duplicate	<a href="#">ProductTypeCode</a>

At the bottom, a status bar shows "0 Errors", "0 Warnings", and "2 Success".

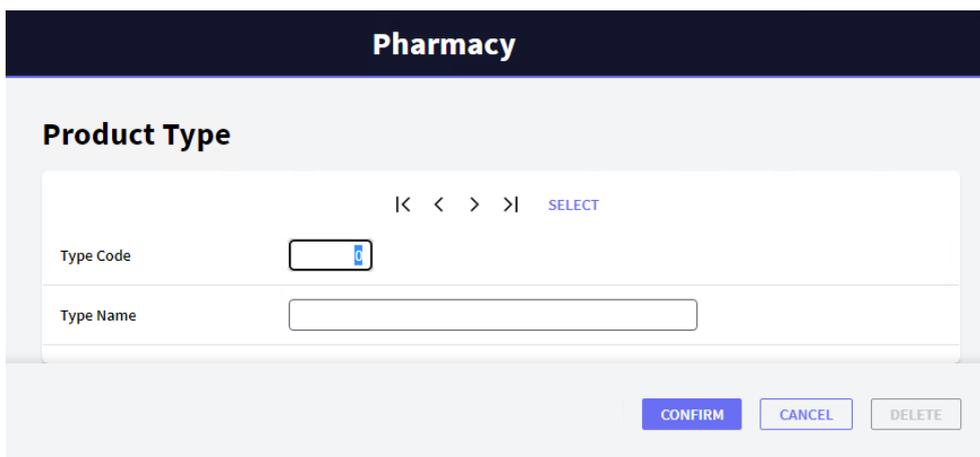
Observa que, como se explicó, el atributo *ProductTypeName* no se incluye en la tabla física *Product* que se creará, a pesar de que la has incluido en la estructura de la Transacción *Product* (con el objetivo de mostrar su valor en la interfaz Web de la Transacción).

Si estás de acuerdo con el Análisis de Impacto, puedes 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 estos 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 te informará si el resultado fue exitoso o si hubo algún error o advertencia. Luego podrás ejecutar y testear la ejecución. Para esto se abrirá la ventana Launchpad, que ofrece una manera rápida de ejecutar los objetos definidos.

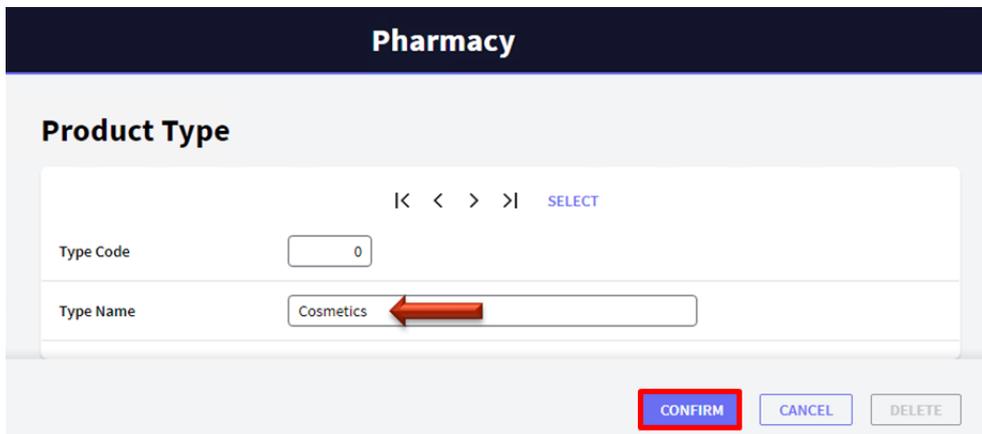


Haz clic sobre el enlace *ProductType*:



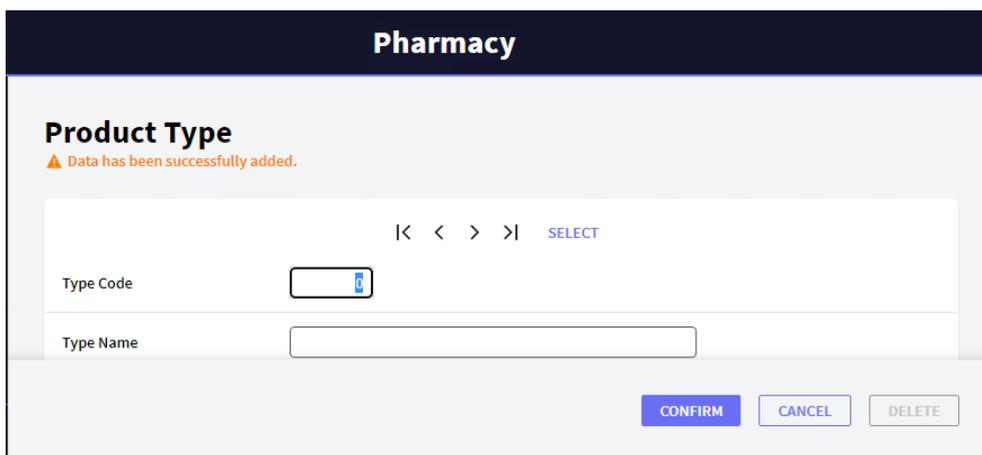
La página que se abre permite al usuario agregar, actualizar y eliminar tipos de productos. Ingresas el primer *tipo de producto*.

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



The screenshot shows a web interface for a 'Pharmacy' with a 'Product Type' form. At the top, there is a dark blue header with the word 'Pharmacy' in white. Below it, the 'Product Type' section has a title and a navigation bar with icons for back, forward, and search, along with a 'SELECT' button. The form contains two input fields: 'Type Code' with the value '0' and 'Type Name' with the value 'Cosmetics'. A red arrow points to the 'Type Name' field. At the bottom right, there are three buttons: 'CONFIRM' (highlighted with a red box), 'CANCEL', and 'DELETE'.

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 lista para ingresar otro tipo de producto:



The screenshot shows the same 'Pharmacy' interface after a successful submission. A message 'Data has been successfully added.' is displayed in orange text above the form. The 'Type Code' field now contains a blue cursor, and the 'Type Name' field is empty. The 'CONFIRM' button is now blue, while 'CANCEL' and 'DELETE' remain grey.

Ingresa el segundo tipo de producto:

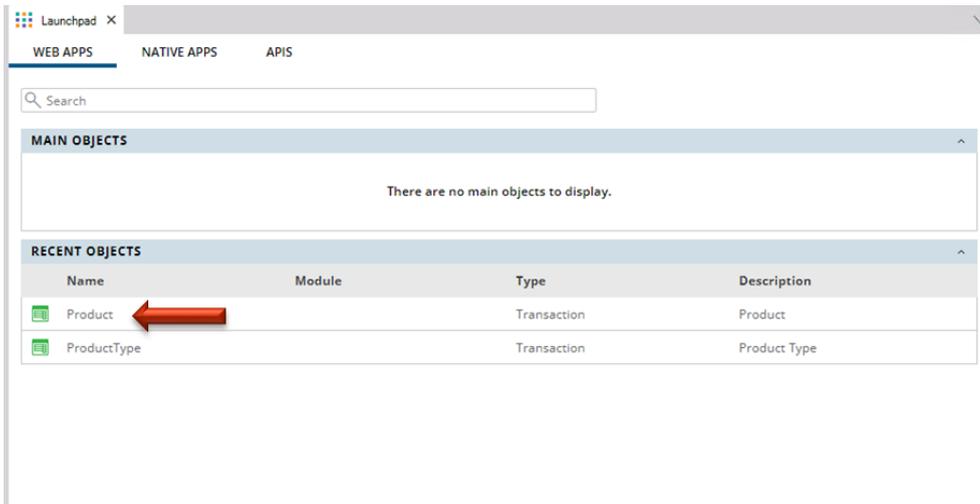
The screenshot shows a web interface for a Pharmacy. At the top, there is a dark blue header with the word "Pharmacy" in white. Below the header, the section is titled "Product Type". There is a navigation bar with icons for back, left, right, and forward, followed by a "SELECT" button. Below this, there are two input fields: "Type Code" with the value "0" and "Type Name" with the value "Medicines". At the bottom right, there are three buttons: "CONFIRM" (highlighted in blue), "CANCEL", and "DELETE".

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

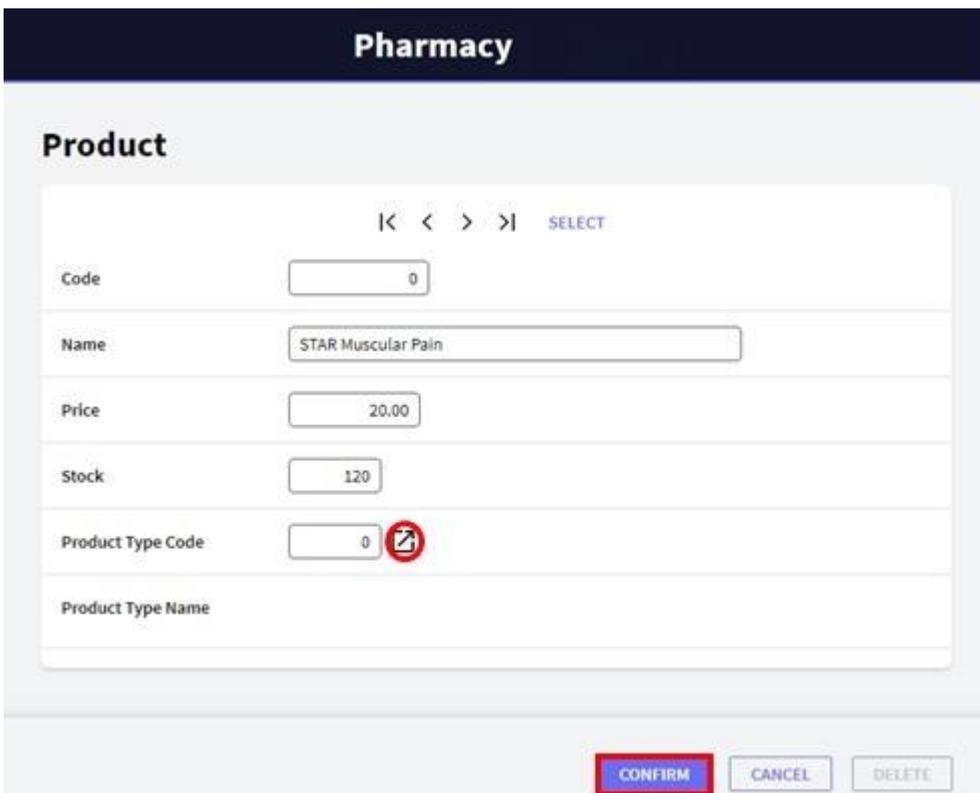
This screenshot is similar to the first one, but the "Type Code" is "1" and the "Type Name" is "Cosmetics". The left arrow button in the navigation bar is circled in red, indicating it is the button to be clicked to navigate to the next item.

This screenshot shows the "Type Code" as "2" and the "Type Name" as "Medicines". The right arrow button in the navigation bar is circled in red, indicating it is the button to be clicked to navigate to the next item.

Ahora, ejecuta la Transacción *Product*. Para hacerlo, vuelve al Launchpad y luego ejecuta la Transacción *Product* seleccionando su enlace:



Agrega el primer producto. Debes indicar el tipo de producto. Si recuerdas el código del tipo de producto, puedes ingresarlo. Otra opción es seleccionarlo desde una lista haciendo clic en la flecha señalizada.



**Pharmacy**

**Product**

|< < > >| [SELECT](#)

Code

Name

Price

Stock

Product Type Code  

Product Type Name

**Selection List Product Type** ✕

Product Type Code

Product Type Name

Type Code	Type Name
1	<a href="#">Cosmetics</a>
2	<a href="#">Medicines</a>

[CANCEL](#)

[CONFIRM](#) [CANCEL](#) [DELETE](#)

**Pharmacy**

**Product**

|< < > >| [SELECT](#)

Code

Name

Price

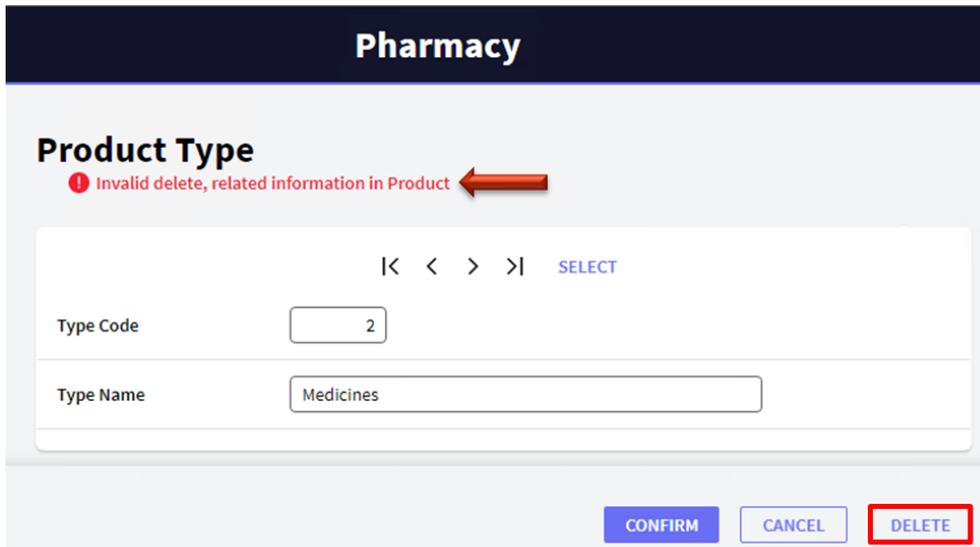
Stock

Product Type Code  

Product Type Name

[CONFIRM](#) [CANCEL](#) [DELETE](#)

Ahora, intenta eliminar el tipo de producto "Medicines". Para esto, selecciona ese tipo de producto y presiona el botón Delete:



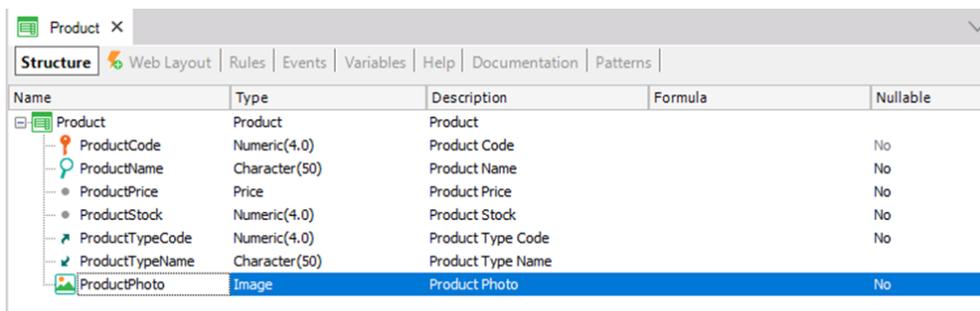
Un mensaje te informará que la eliminación no se puede realizar porque existen datos relacionados en Product (el medicamento para el dolor muscular STAR es un producto que pertenece a este tipo de producto).

## LA APLICACIÓN CRECE

Probablemente hayas notado todo lo que GeneXus generó automáticamente a partir de los dos objetos que has definido.

Ahora, supón que en la farmacia te dicen que, para cada producto, quieren registrar una imagen.

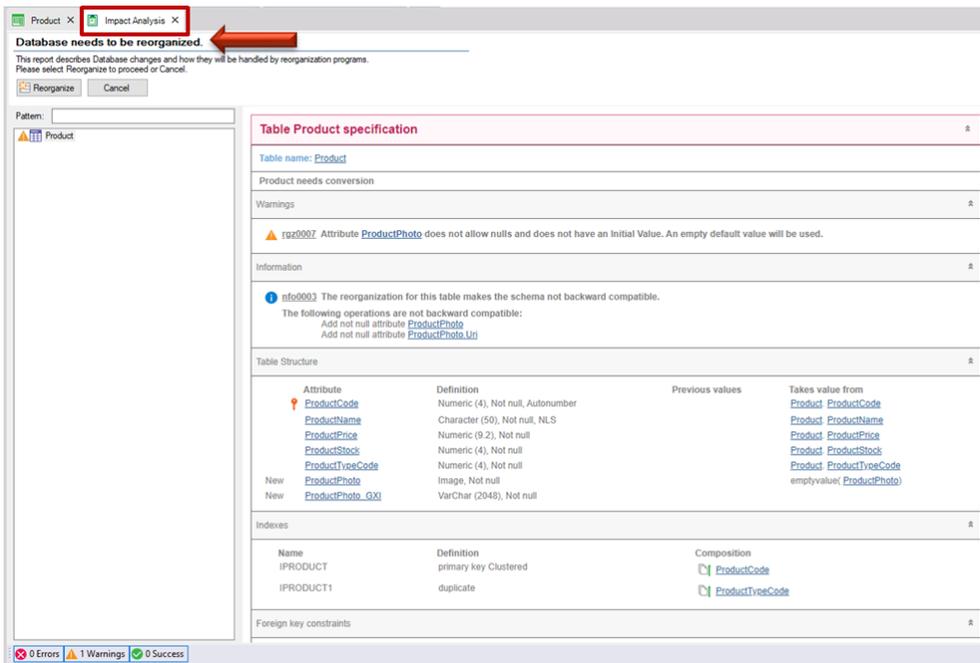
Para hacerlo, vuelve a GeneXus y, en la Transacción *Product*, solo tienes que ingresar un atributo llamado *ProductPhoto*:



El tipo Image te permite almacenar imágenes.

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

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

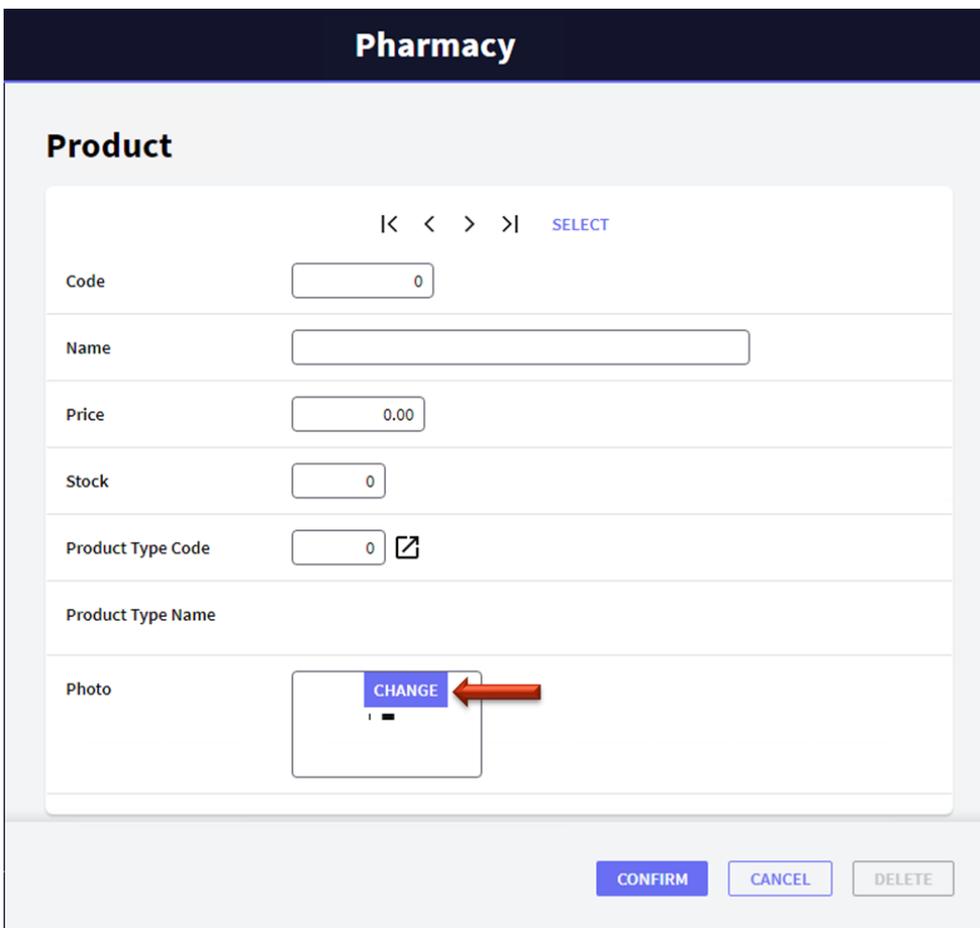
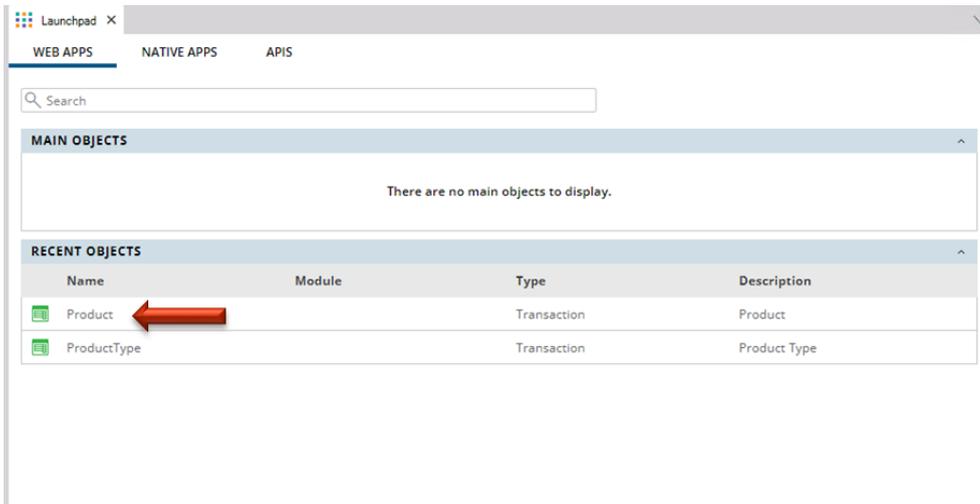


Recuerda: El Informe de Análisis de Impacto indica los cambios estructurales que se necesitan realizar en la base de datos.

Al leer el informe, verás 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 debe actualizarse la tabla *Product*.

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.

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



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

## Pharmacy

### Product

▲ Data has been successfully updated.

|< < > >| [SELECT](#)

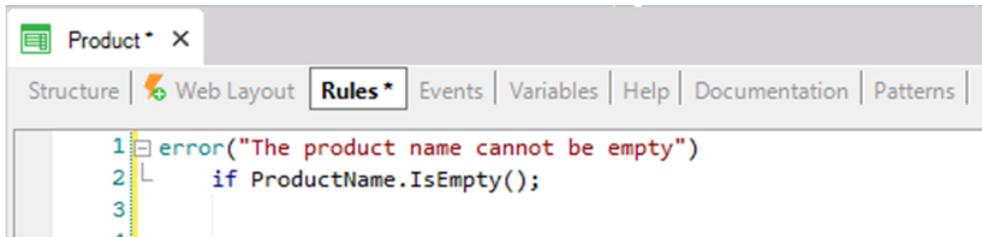
Code	<input type="text" value="1"/>
Name	<input type="text" value="STAR Muscular Pain"/>
Price	<input type="text" value="20.00"/>
Stock	<input type="text" value="120"/>
Product Type Code	<input type="text" value="2"/> 
Product Type Name	Medicines
Photo	<div><a href="#">CHANGE</a></div>

[CONFIRM](#) [CANCEL](#) [DELETE](#)

## AGREGAR REGLAS DE NEGOCIOS

Además de todos los controles automáticos que GeneXus incluye 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**.

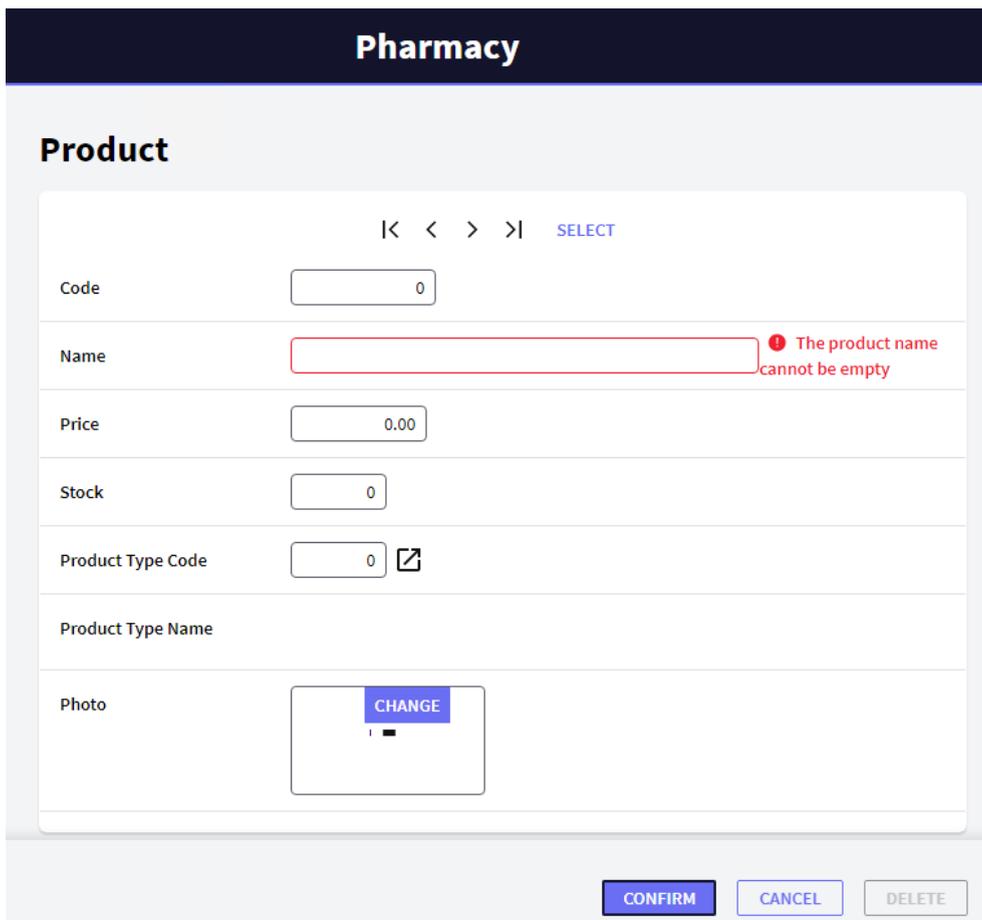
Si, por ejemplo, un requisito es que no se permita guardar productos sin un nombre, GeneXus ofrece una regla llamada **Error** que le permitirá evitar eso:



```
1 error("The product name cannot be empty")
2   if ProductName.IsEmpty();
3
```

Presiona 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.

Ejecuta la Transacción *Product*. Ten en cuenta que si dejas el nombre del producto en blanco, se ejecutará la regla que has definido:



The screenshot shows a web application interface for a pharmacy. At the top, there is a dark blue header with the word "Pharmacy" in white. Below the header, the main content area is titled "Product" in bold black text. The form contains several input fields: "Code" with the value "0", "Name" (which is empty and has a red border and a red error message "The product name cannot be empty"), "Price" with the value "0.00", "Stock" with the value "0", "Product Type Code" with the value "0" and a selection icon, and "Product Type Name". At the bottom of the form, there is a "Photo" field with a "CHANGE" button. At the bottom of the page, there are three buttons: "CONFIRM" (blue), "CANCEL" (light blue), and "DELETE" (grey).

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, deseas informar que el precio del producto se ha dejado en blanco sin obligar al usuario a que lo ingrese, puedes agregar la siguiente regla en la Transacción *Product*:

```
Product * X
Structure | Web Layout | Rules * | Events | Variables | Help | Documentation | Patterns |
1 error("The product name cannot be empty")
2   if ProductName.IsEmpty();
3
4 msg("The product price is empty")
5   if ProductPrice.IsEmpty();
6
```

The screenshot shows a web application interface for a Pharmacy. At the top, there is a dark blue header with the word "Pharmacy" in white. Below the header, the main content area is titled "Product". The form contains several input fields: "Code" (with value 0), "Name" (empty, with a red error message "The product name cannot be empty"), "Price" (with value 0.00 and a yellow warning message "The product price is empty"), "Stock" (with value 0), "Product Type Code" (with value 0 and an edit icon), "Product Type Name", and "Photo" (with a "CHANGE" button). At the bottom of the form, there are three buttons: "CONFIRM", "CANCEL", and "DELETE".

Este conjunto de reglas podría escribirse en cualquier otro orden y el resultado en tiempo de ejecución sería exactamente el mismo, 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.

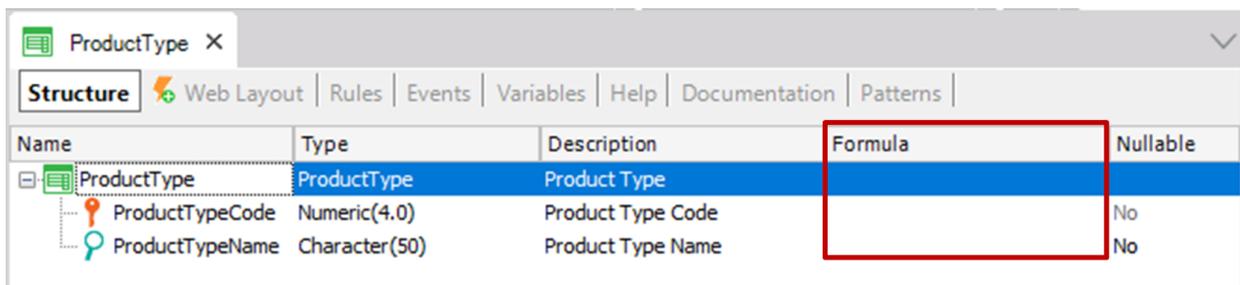
Cada Transacción debe tener sus propias reglas de comportamiento definidas.

## DEFINIR CÁLCULOS: FÓRMULAS

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

Existen diferentes formas posibles de definir fórmulas.

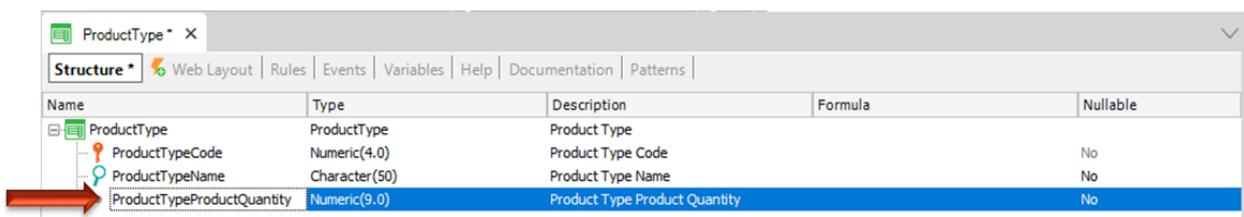
¿Qué es una **fórmula Global**? Una fórmula global es un cálculo que el desarrollador define asociado a un atributo. Observa que las estructuras de las Transacciones contienen una columna llamada **Formula**:



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.

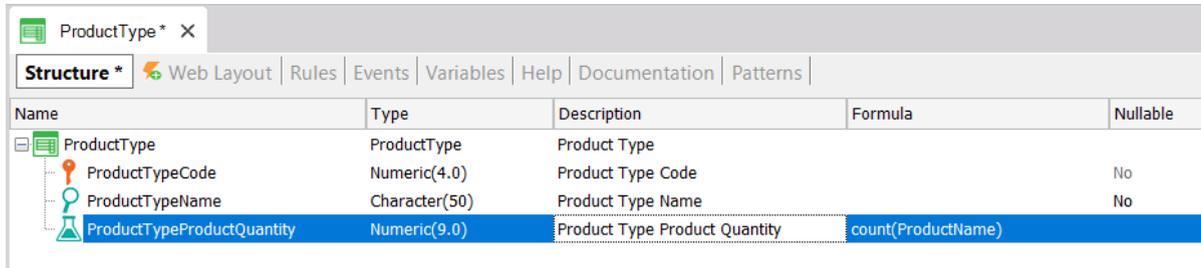
Supón que la farmacia necesita saber en todo momento cuántos productos registrados hay de cada tipo de producto. Para resolver esto, crea un nuevo atributo en la Transacción *ProductType* con el fin de definirlo como una fórmula global:



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, define el cálculo asociado al atributo *ProductTypeProductQuantity*.

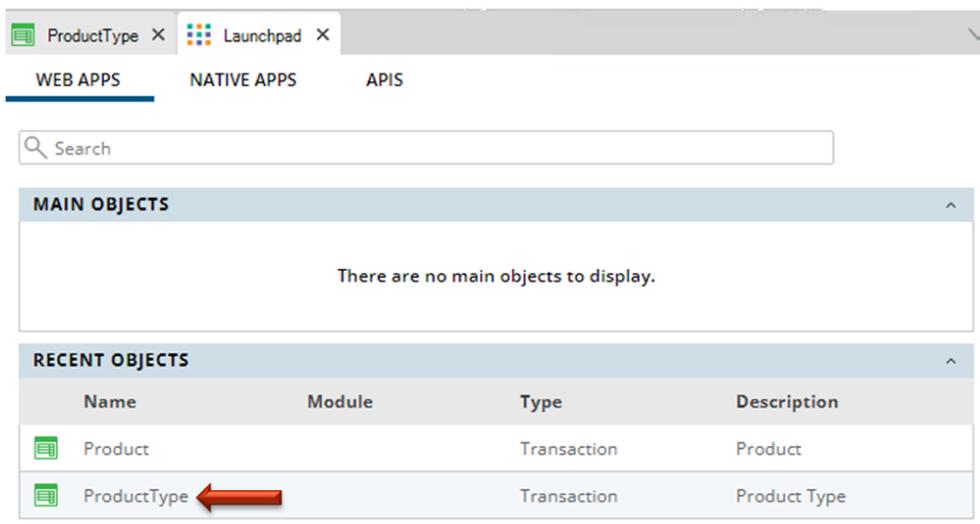
GeneXus ofrece una fórmula llamada Count para resolver la necesidad que solicita la farmacia (hay muchas otras, como Sum, Average, etc.).



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 cuál tabla 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 en la cual debe realizarse la navegación para obtener 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.

Presiona F5. No hay cambios físicos para realizar en la base de datos. GeneXus solo generará algunos programas y el Launchpad se ejecutará nuevamente:



Name	Module	Type	Description
Product		Transaction	Product
ProductType		Transaction	Product Type

Ejecuta la Transacción *ProductType*. Verás cómo la cantidad de productos de cada tipo siempre se calcula para cada tipo de producto, en el momento:

**Pharmacy**

**Product Type**

|< < > >| [SELECT](#)

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

[CONFIRM](#) [CANCEL](#) [DELETE](#)

**Pharmacy**

**Product Type**

|< < > >| [SELECT](#)

Type Code	<input type="text" value="2"/>
Type Name	<input type="text" value="Medicines"/>
Product Quantity	<input type="text" value="1"/>

[CONFIRM](#) [CANCEL](#) [DELETE](#)

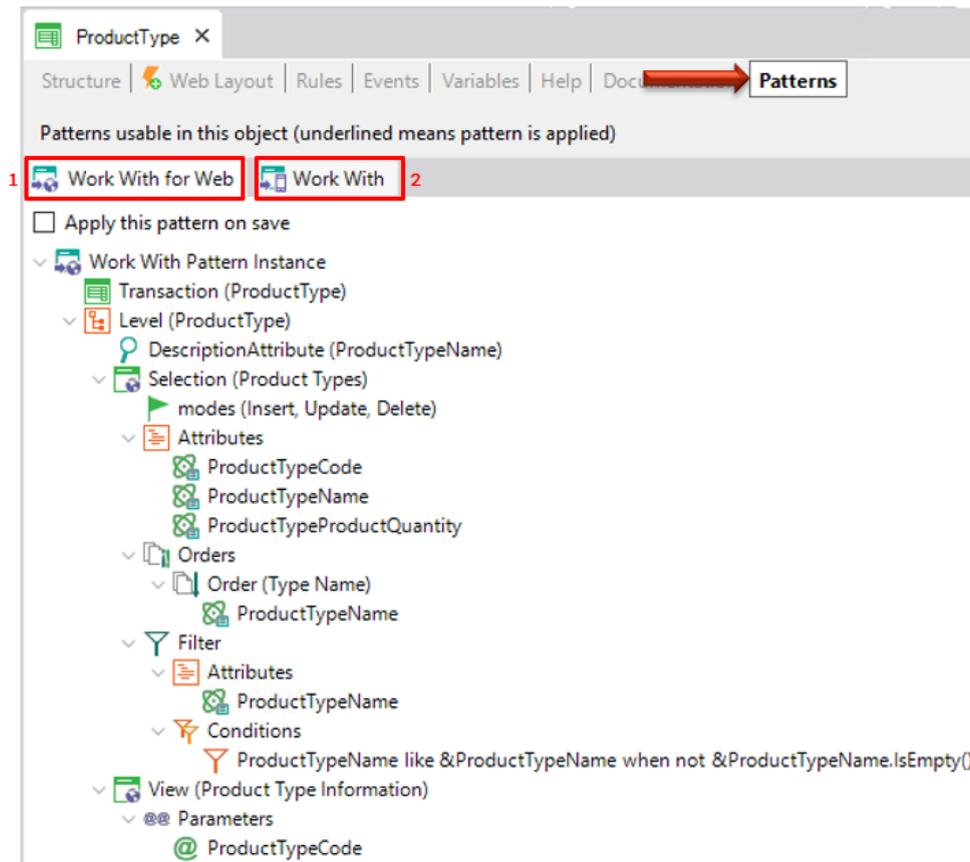
Puedes agregar más productos y verificar que la cantidad de productos de cada tipo siempre se calcula en el momento.

## USO DE PATRONES (PARA WEB Y PARA DISPOSITIVOS MÓVILES)

Los patrones 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 útiles sin necesidad de programarlos.

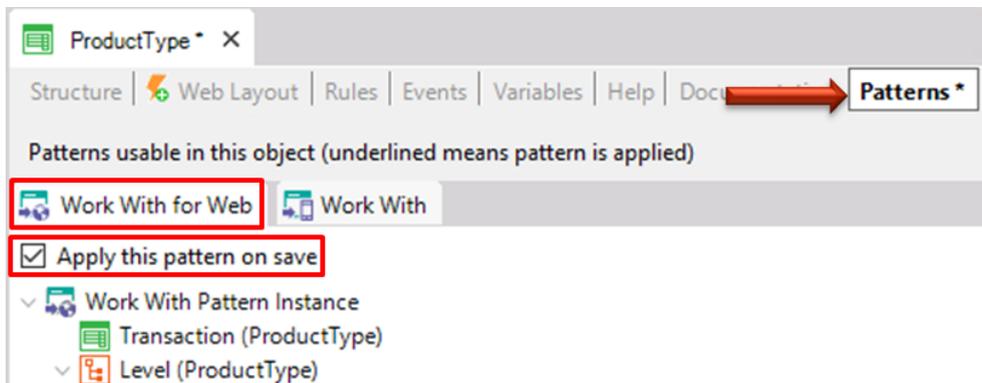
Observa la sección Patterns de una Transacción. Por ejemplo, en la Transacción *ProductType*, selecciona la sección Patterns:



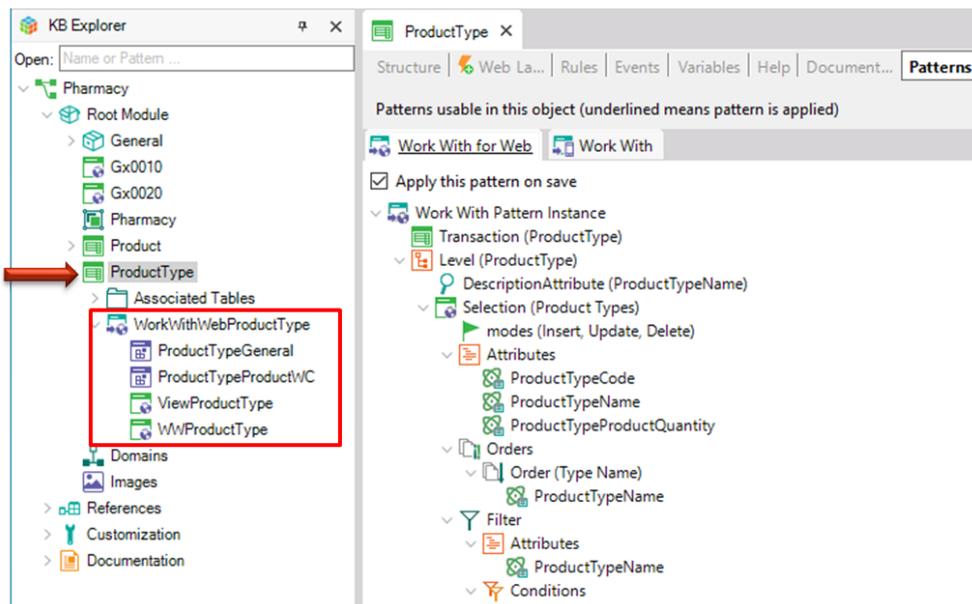
Nota que hay dos pestañas disponibles, y cada una ofrece un patrón diferente para aplicarlo a la misma Transacción.

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

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

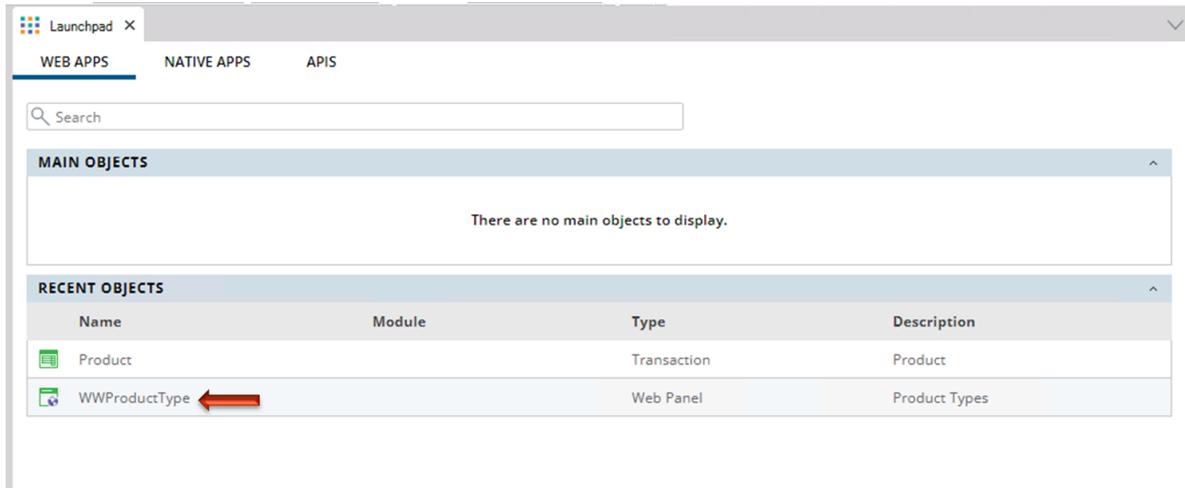


A continuación, si buscas la Transacción *ProductType* en el **KB Explorer**, podrás ver que hay varios objetos nuevos debajo de la Transacción:

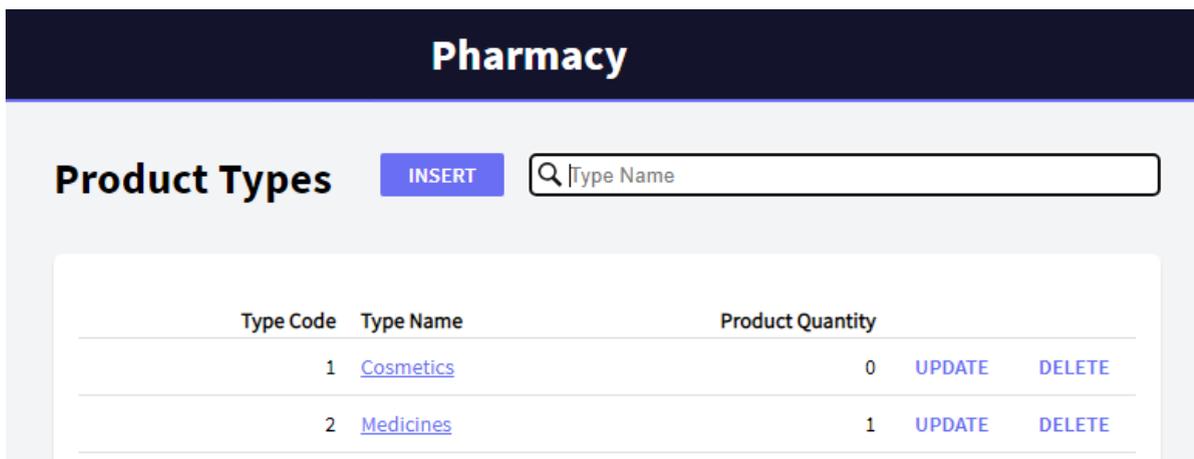


Fueron creados por GeneXus por haberle aplicado el patrón *Work With for Web*.

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



Observa el último enlace que se muestra (WWProductType). Ofrece “trabajar con tipos de productos” (y desde allí hay invocaciones a la Transacción *ProductType*). Haz clic en ese enlace.



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

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

### Pharmacy

**Product Types** INSERT

Type Code	Type Name	Product Quantity		
1	<a href="#">Cosmetics</a>	0	<b>UPDATE</b>	<a href="#">DELETE</a>
2	<a href="#">Medicines</a>	1	<a href="#">UPDATE</a>	<a href="#">DELETE</a>

### Pharmacy

**Product Type**

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

CONFIRM CANCEL

Puedes ver que se abre la Transacción *ProductType* ofreciendo editar los datos del tipo de producto que se encuentra en esa línea. Cambia el nombre del tipo y confirma:

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) y ya se puede ver el cambio:

Type Code	Type Name	Product Quantity		
1	<a href="#">Cosmetics for Teens</a>	0	UPDATE	DELETE
2	<a href="#">Medicines</a>	1	UPDATE	DELETE

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

## Pharmacy

### Product Types

← ame

Type Code	Type Name	Product Quantity		
1	<a href="#">Cosmetics for Teens</a>	0	<a href="#">UPDATE</a>	<a href="#">DELETE</a>
2	<a href="#">Medicines</a>	1	<a href="#">UPDATE</a>	<a href="#">DELETE</a>

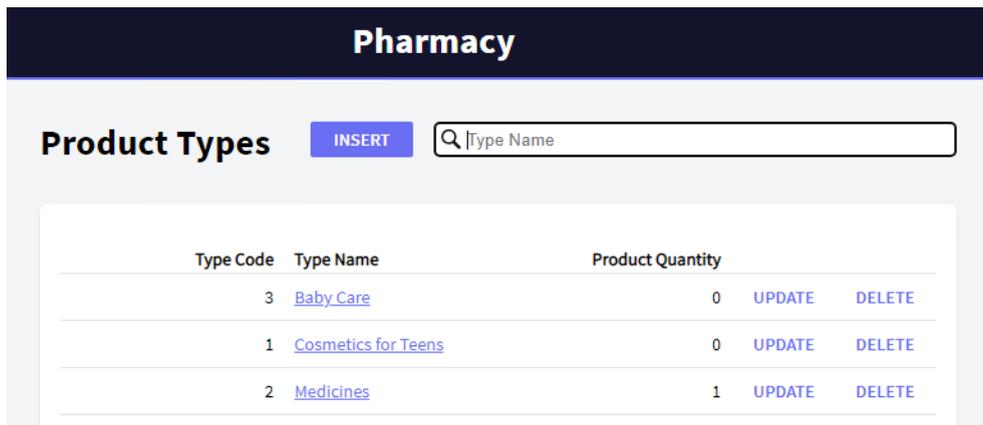
Además, el botón INSERT ubicado fuera de la grilla, permite agregar nuevos tipos de productos. Al hacer clic en el botón INSERT, se abre la Transacción *ProductType*, lista para poder agregar un nuevo tipo de producto. Presiónalo para ingresar un nuevo tipo de producto (recuerda que en este caso solo es necesario ingresar el nombre del tipo de producto porque has configurado la propiedad Autonumber del atributo clave con valor True):

## Pharmacy

### Product Type

Type Code	0
Type Name	Baby Care
Product Quantity	0

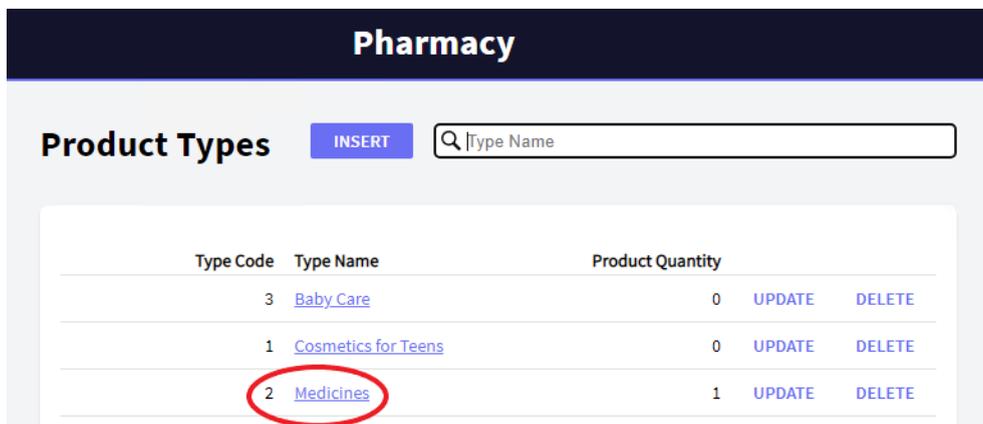
Después de la inserción, la aplicación vuelve de nuevo a la página *Work With Product Types*:



The screenshot shows the 'Pharmacy' section with a 'Product Types' header. Below the header is an 'INSERT' button and a search input field labeled 'Type Name'. A table displays the following data:

Type Code	Type Name	Product Quantity		
3	<a href="#">Baby Care</a>	0	UPDATE	DELETE
1	<a href="#">Cosmetics for Teens</a>	0	UPDATE	DELETE
2	<a href="#">Medicines</a>	1	UPDATE	DELETE

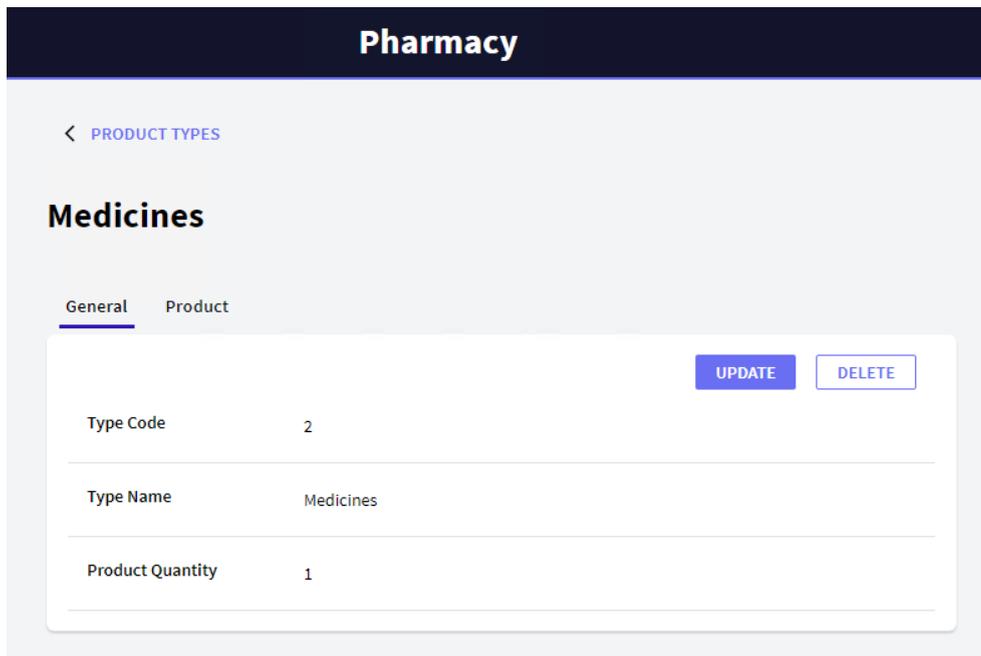
Ahora observa que cada nombre de tipo de producto tiene un enlace. Haz clic en el enlace: *Medicines*.



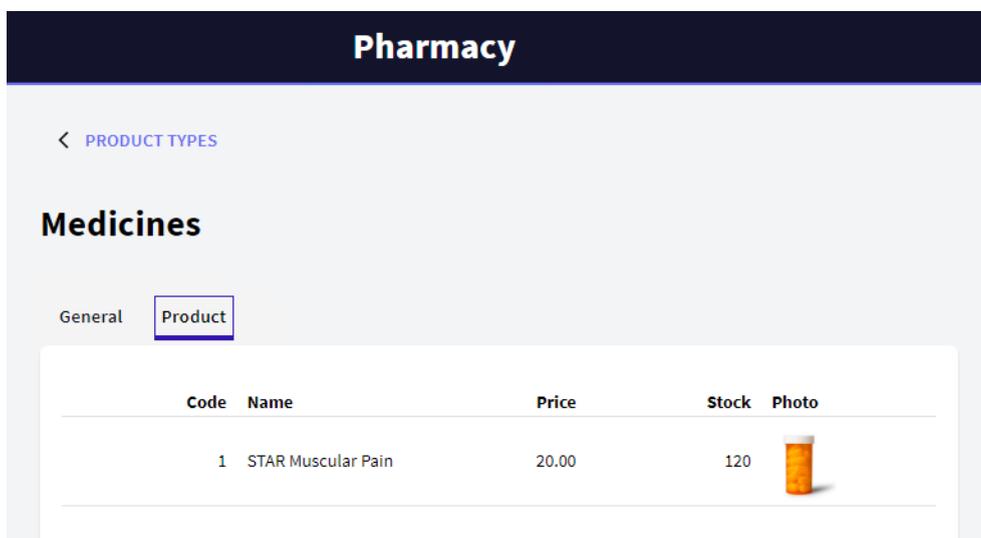
This screenshot is identical to the previous one, but the 'Medicines' link in the table is circled in red to indicate it should be clicked.

Type Code	Type Name	Product Quantity		
3	<a href="#">Baby Care</a>	0	UPDATE	DELETE
1	<a href="#">Cosmetics for Teens</a>	0	UPDATE	DELETE
2	<a href="#">Medicines</a>	1	UPDATE	DELETE

Como puedes observar, en la primera pestaña se muestran todos los detalles del tipo de producto seleccionado (*General Tab*):



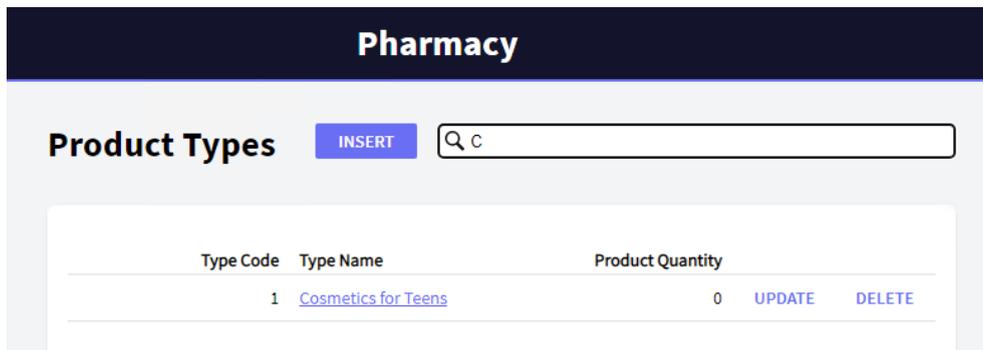
Otra pestaña (*Product Tab*) muestra la lista de productos que pertenecen a ese tipo de producto:



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, vuelve a la página *Work With Product Types*, haciendo clic en el enlace que aparece en la parte superior izquierda de la página.

Observa 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:



The screenshot shows a web application interface for a 'Pharmacy'. At the top, there is a dark blue header with the word 'Pharmacy' in white. Below the header, there is a section titled 'Product Types' with a blue 'INSERT' button and a search bar containing the letter 'C'. Below the search bar, there is a table with the following data:

Type Code	Type Name	Product Quantity		
1	<a href="#">Cosmetics for Teens</a>	0	UPDATE	DELETE

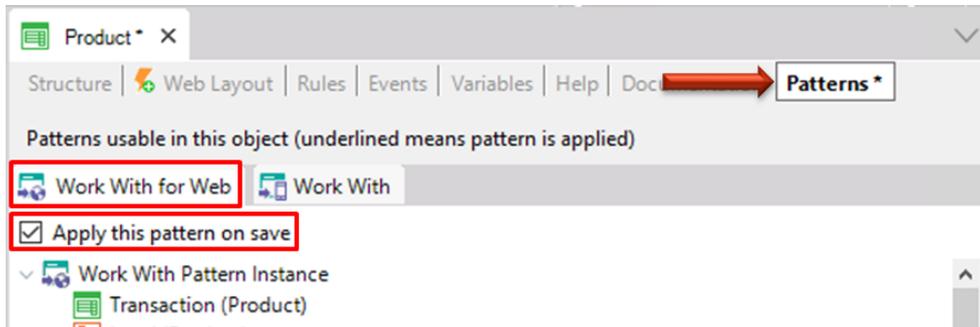
Vuelve a GeneXus. Hasta ahora, solo has seleccionado **Apply this pattern on save** en la pestaña *Work With for Web* de la Transacción *ProductType*, y después de guardar, has visto todas las funciones que se generaron automáticamente.

Lo que quizás no hayas observado es este árbol configurable:

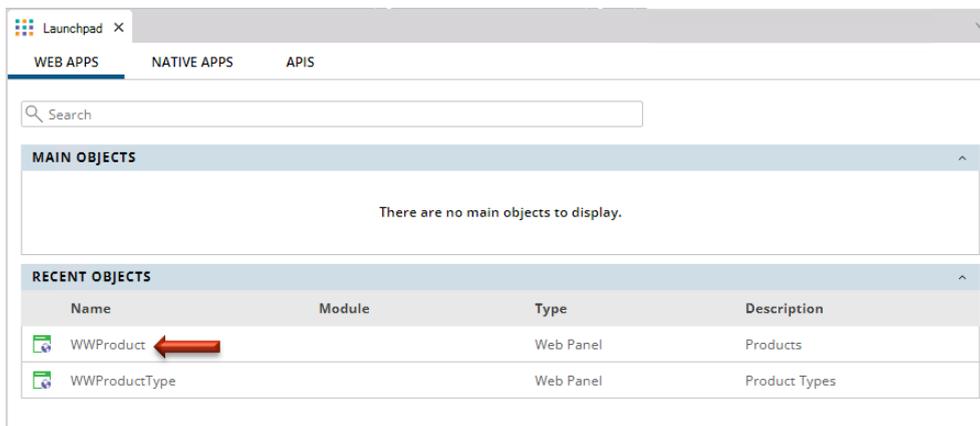


Tiene nodos, subnodos y elementos configurables, de modo que es posible personalizar los comportamientos a generar (es decir, cambiar o agregar criterios de búsqueda, etc.).

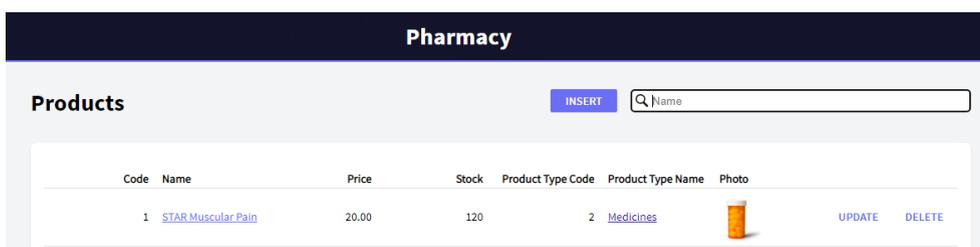
Ahora aplica también el patrón *Work With for Web* a la Transacción *Product*. Como se explicó antes, solo tienes que abrir la Transacción *Product* y en su sección *Patterns*, debes seleccionar la pestaña *Work With for Web*. Allí tienes que marcar la opción **Apply this pattern on save** y guardar:



Presiona F5. GeneXus genera los programas necesarios y ejecuta la aplicación con los cambios. Ejecuta la página *Work With Product*:



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



Inserta un nuevo producto:

### Pharmacy

#### Product

Code	0
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*:

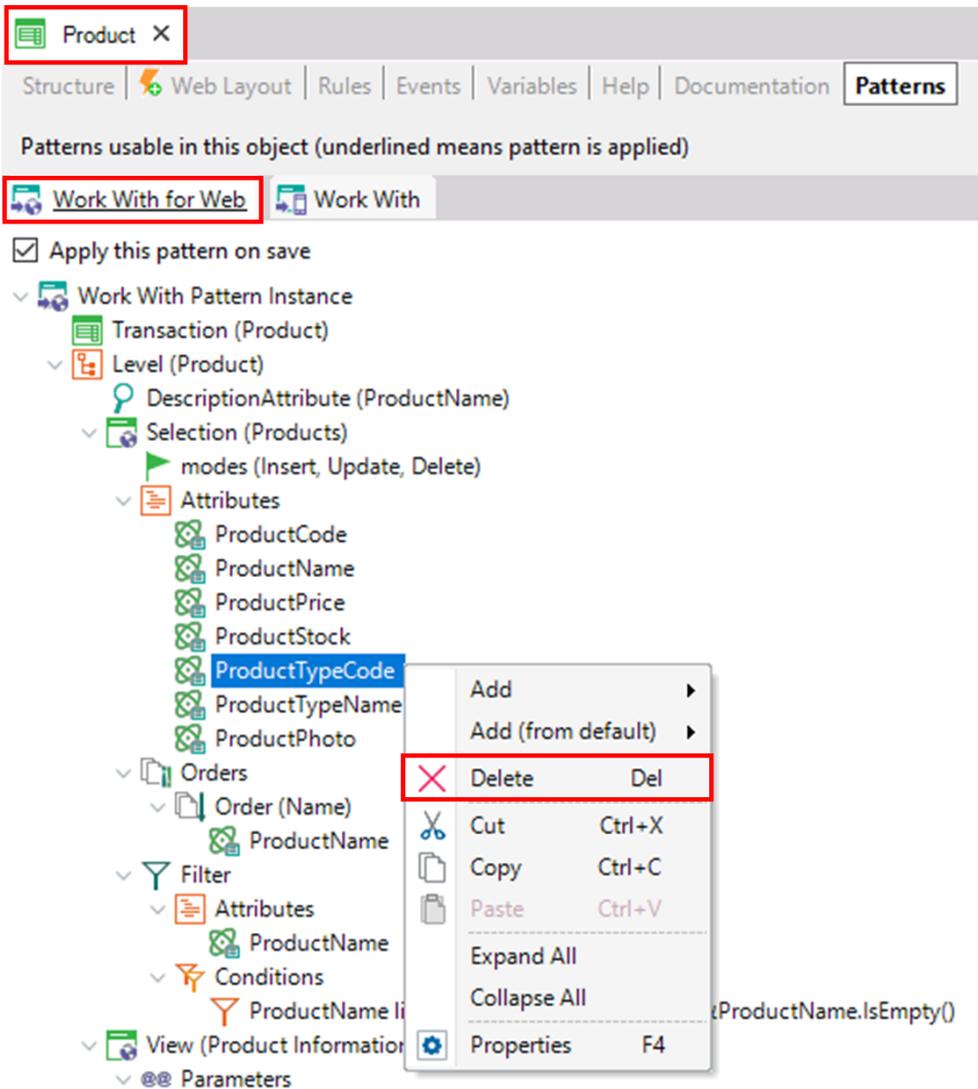
### Pharmacy

#### Products

[INSERT](#)

Code	Name	Price	Stock	Product Type Code	Product Type Name	Photo		
1	<a href="#">STAR Muscular Pain</a>	20.00	120	2	<a href="#">Medicines</a>		<a href="#">UPDATE</a>	<a href="#">DELETE</a>
2	<a href="#">X Eyeshadow Palette</a>	25.00	30	1	<a href="#">Cosmetics for Teens</a>		<a href="#">UPDATE</a>	<a href="#">DELETE</a>

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 *Work With for Web*:



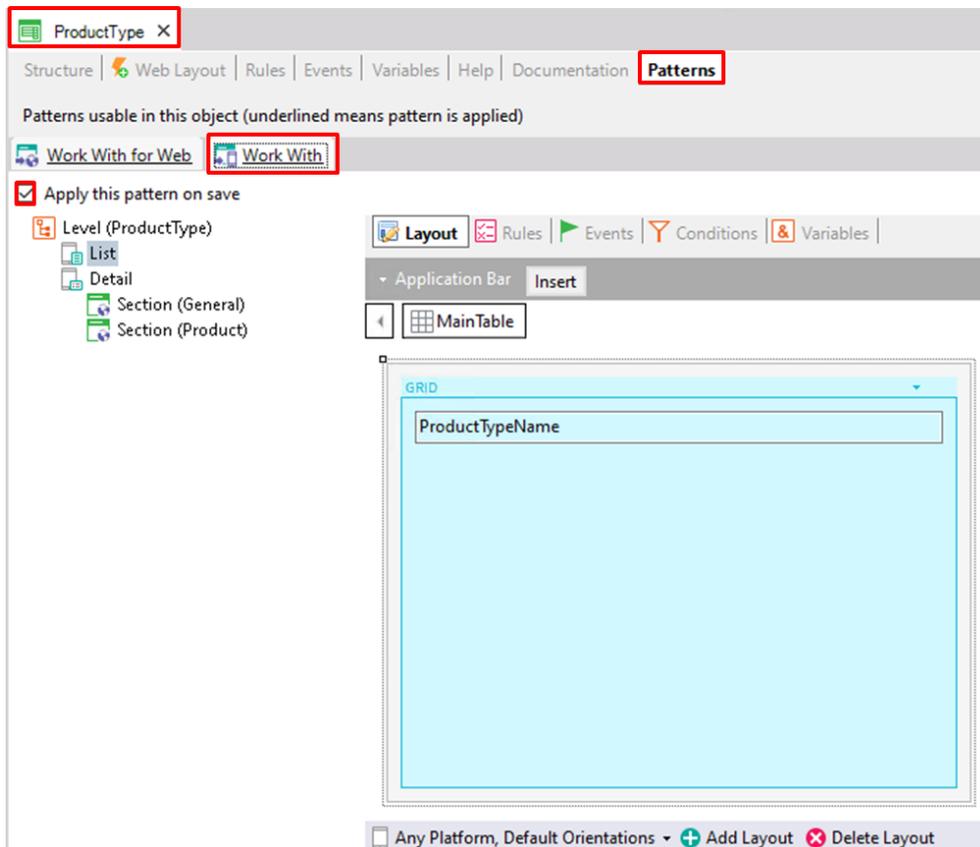
Presiona F5, GeneXus guardará lo realizado, generará solo los programas necesarios y ejecutará la aplicación con los cambios.

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

A continuación se muestra la página *Work With Products* que lista todos los productos que se han agregado:

Pharmacy						
Products						
Code	Name	Price	Stock	Product Type Code	Product Type Name	Photo
4	<a href="#">Asterix Headache Medicine</a>	20.00	100	2	<a href="#">Medicines</a>	
3	<a href="#">Feeding Bottle</a>	20.00	15	3	<a href="#">Baby Care</a>	
5	<a href="#">LOVE Lipstick #18</a>	8.00	60	1	<a href="#">Cosmetics for Teens</a>	
6	<a href="#">Magic Anti-inflammatory painkillers</a>	30.00	10	2	<a href="#">Medicines</a>	
1	<a href="#">STAR Muscular Pain</a>	20.00	120	2	<a href="#">Medicines</a>	
8	<a href="#">WONDER Facial Cream</a>	900.00	20	1	<a href="#">Cosmetics for Teens</a>	
2	<a href="#">X Eyeshadow Palette</a>	25.00	30	1	<a href="#">Cosmetics for Teens</a>	

Ahora presta atención a la pestaña *Work With* que se ofrece para cada Transacción. Aplícalo a la Transacción *ProductType*:



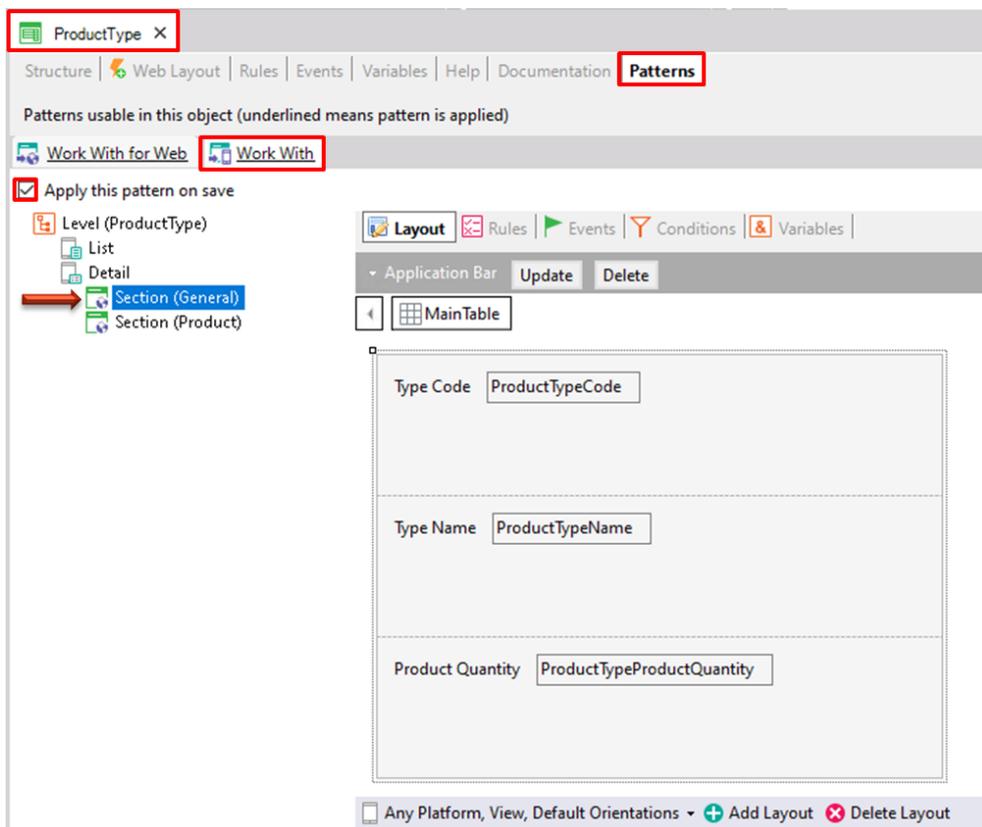
The screenshot shows the Genexus IDE interface with the 'Patterns' tab active for the 'ProductType' object. The 'Work With' pattern is selected and applied to the 'MainTable' grid. The grid contains a 'ProductTypeName' field. The interface also shows the 'Apply this pattern on save' checkbox checked and the 'MainTable' grid selected in the 'Application Bar'.

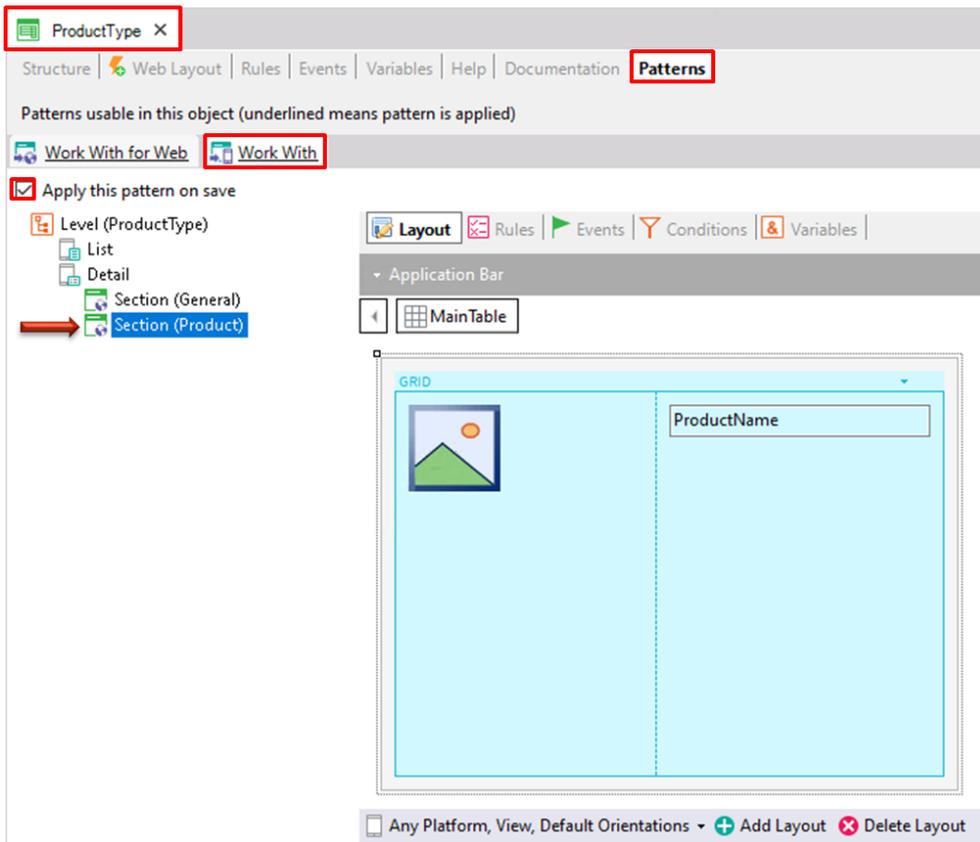
A diferencia del patrón *Work With for Web*, en este caso, ya se muestra el diseño (Layout) con una grilla en vez de una lista de atributos debajo del nodo para ser incluidos en una grilla.

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

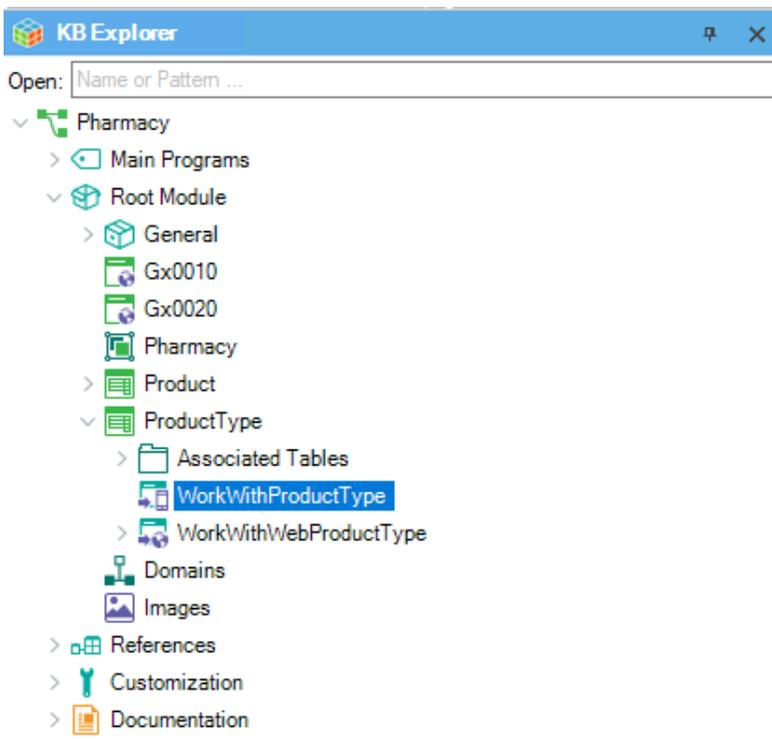
El nodo **Detail** está compuesto por dos secciones: **General** y **Product**.

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.





Después de aplicar este patrón y guardar, si ubicas la Transacción *ProductType* en el **KB Explorer**, ahora puedes ver debajo de ella un nuevo objeto llamado *WorkWithProductType*:



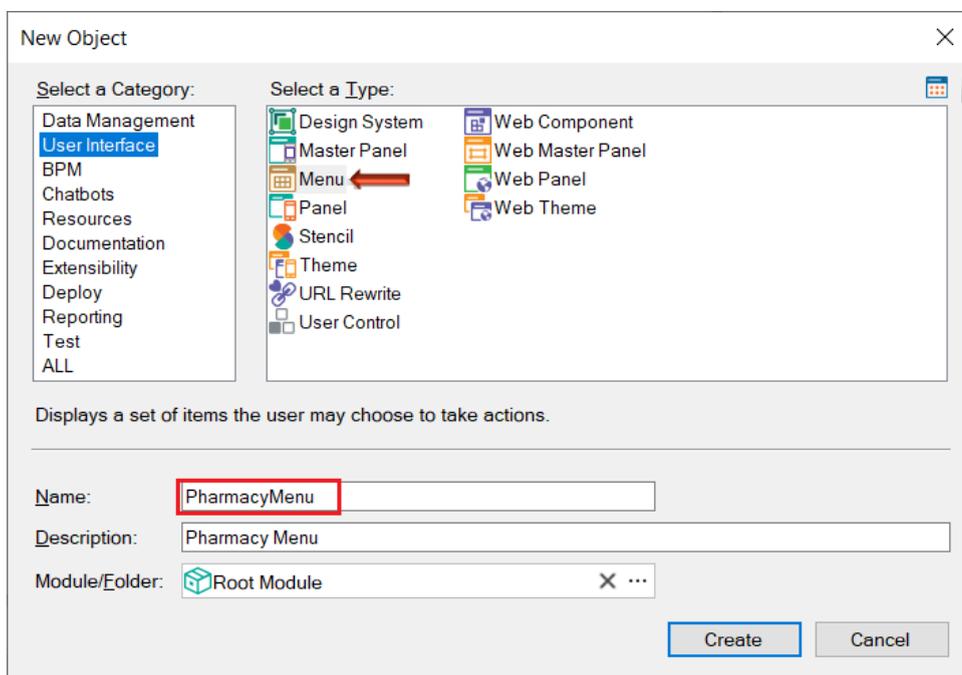
Observa los dos objetos generados bajo la Transacción *ProductType*. Nota que *WorkWithProductType* no tiene otros objetos debajo (porque incluye diferentes secciones para definir la implementación completa en su interior).

En cambio, *WorkWithWebProductType*, es una instancia configurable; y a partir de esa instancia GeneXus genera otros objetos para proporcionar los comportamientos útiles que se han visto.

Pronto verás el componente *WorkWithProductType* en acción.

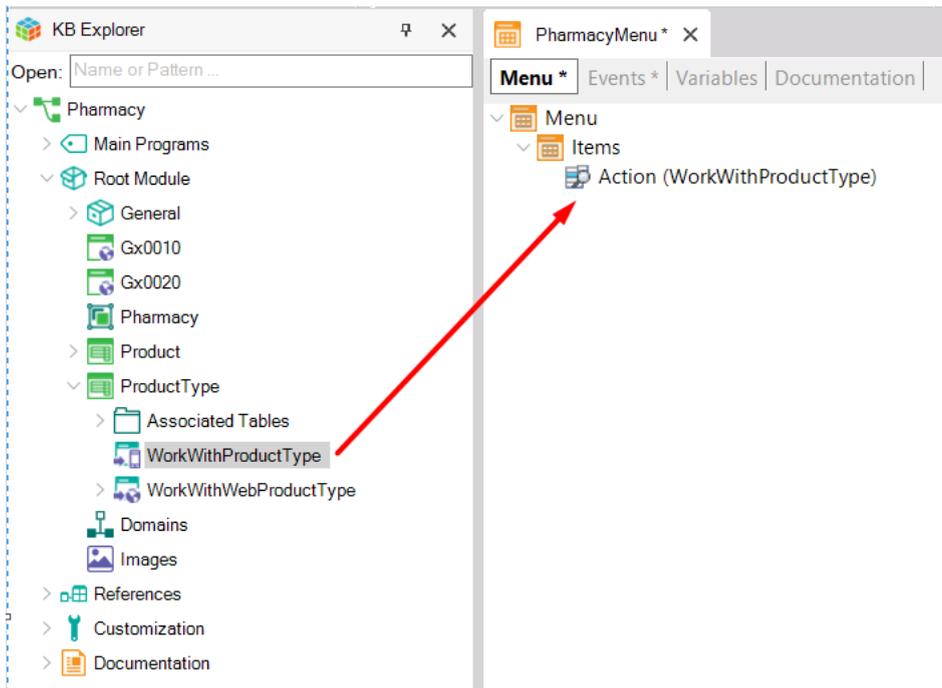
La propuesta ahora es crear un objeto **Menu**, para que sea el primer objeto de la aplicación que se ejecute. Él mostrará un ícono, para que cuando el usuario la pulse, se ejecute el objeto *WorkWithProductType*.

Como se explicó anteriormente, para crear un objeto solo tienes que seleccionar **File > New > Object**:

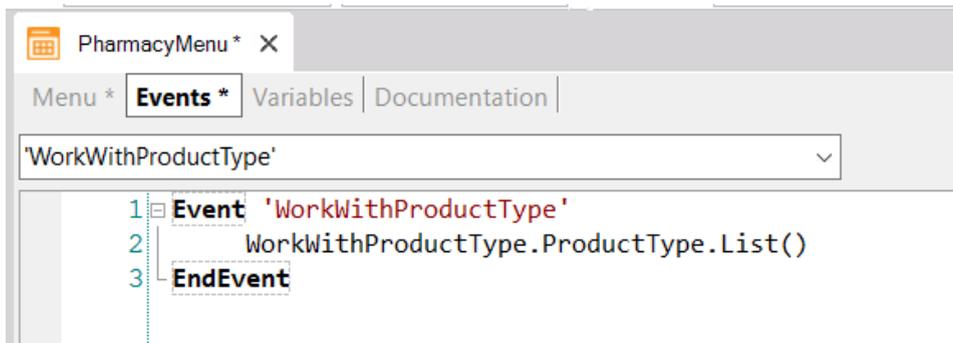


Por defecto, todo **Menu** que se crea tiene su propiedad **Main program = True**, para que este objeto sea un objeto ejecutable independiente (es decir, es compilable y ejecutable por sí mismo).

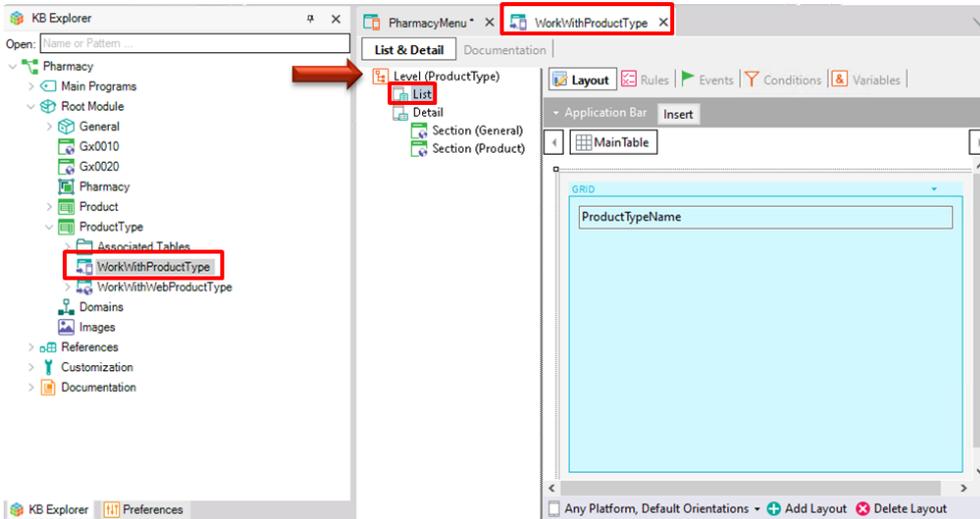
Ahora, arrastra el objeto *WorkWithProductType* desde la ventana KB Explorer hacia el nodo Items del Menu:



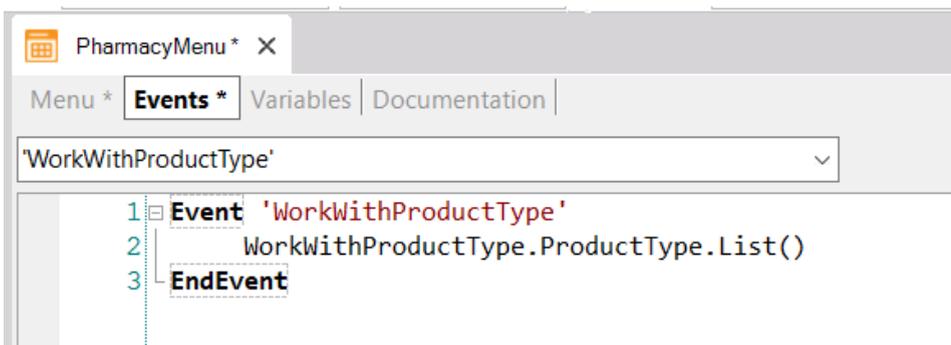
Automáticamente se agrega la Action que se muestra debajo del nodo *Items* y se crea un evento asociado a la acción:



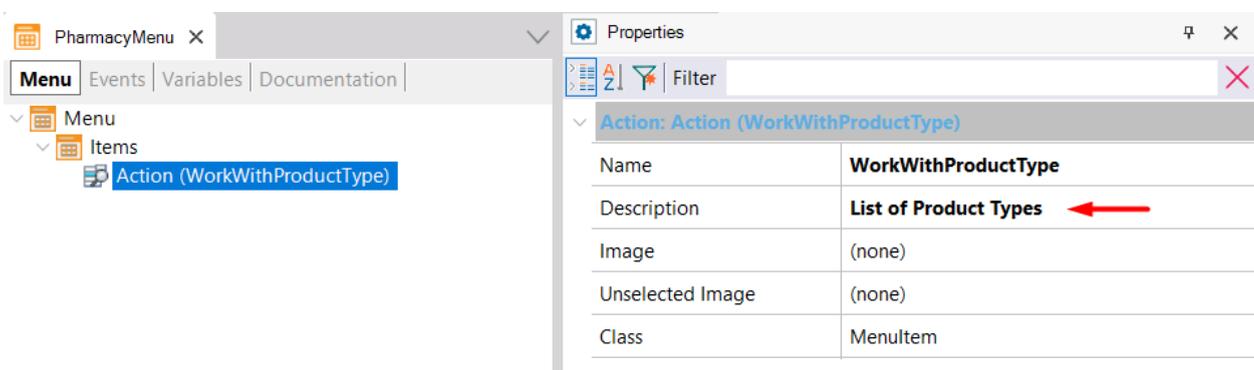
Para comprender la línea de código que se encuentra dentro del evento, vuelva a observar el nodo del árbol configurable contenido en el objeto *WorkWithProductType*:



El objeto tiene un nodo principal (*Level (ProductType)*). Debajo de él puedes ver los nodos *List* y *Detail* respectivamente, así que para invocar al nodo *List* la sintaxis completa es:

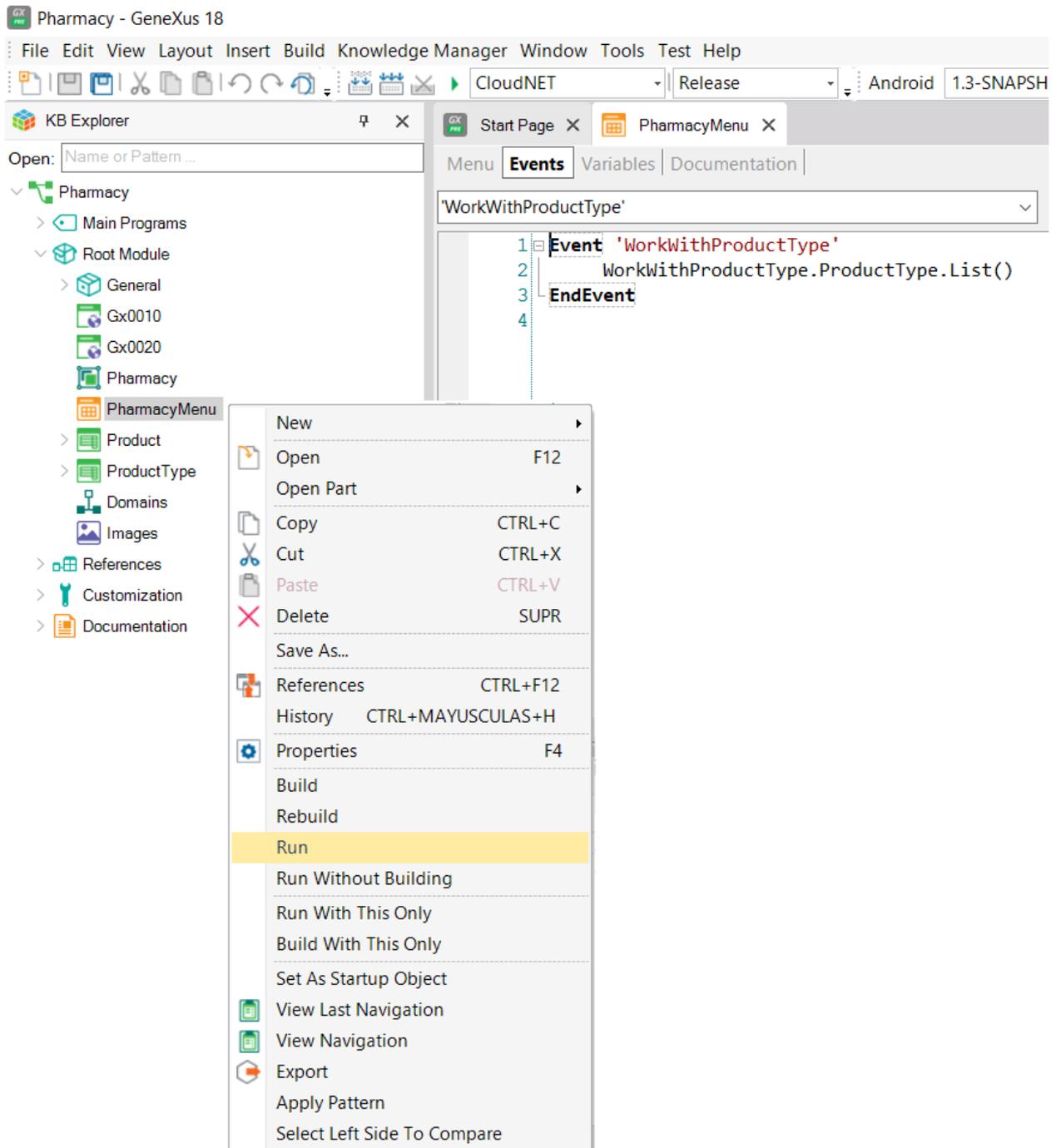


Ahora edita la propiedad *Description* del nodo *Action* para que diga “List of Product Types”:

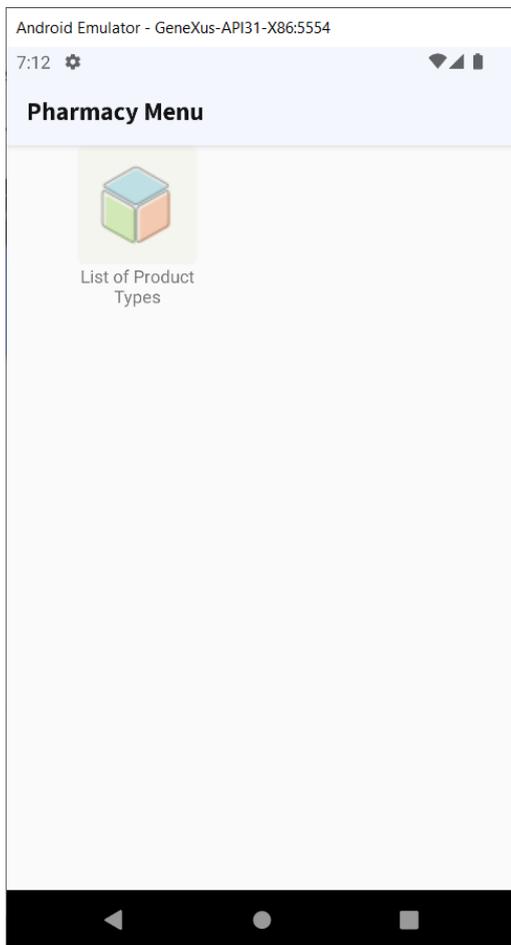


Todo quedó definido y listo para ejecutar la aplicación Mobile.

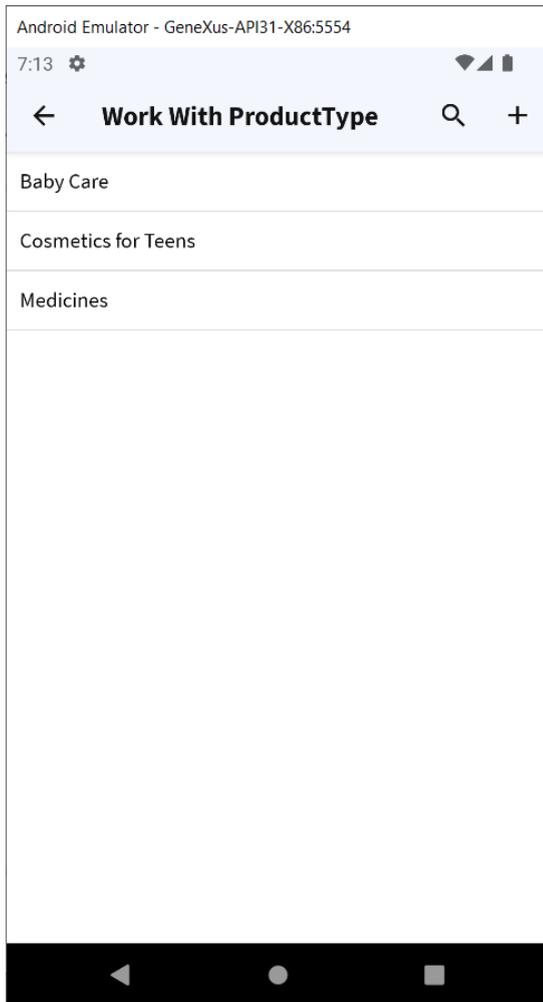
Dado que el objeto *PharmacyMenu* tiene su propiedad Main Program configurada en True, puedes ejecutarlo independientemente desde KB Explorer, haciendo clic con el botón derecho en el objeto y seleccionando **Run**:



La ejecución se puede realizar tanto en un dispositivo móvil que esté conectado a tu computadora o mediante un emulador instalado en tu computadora:

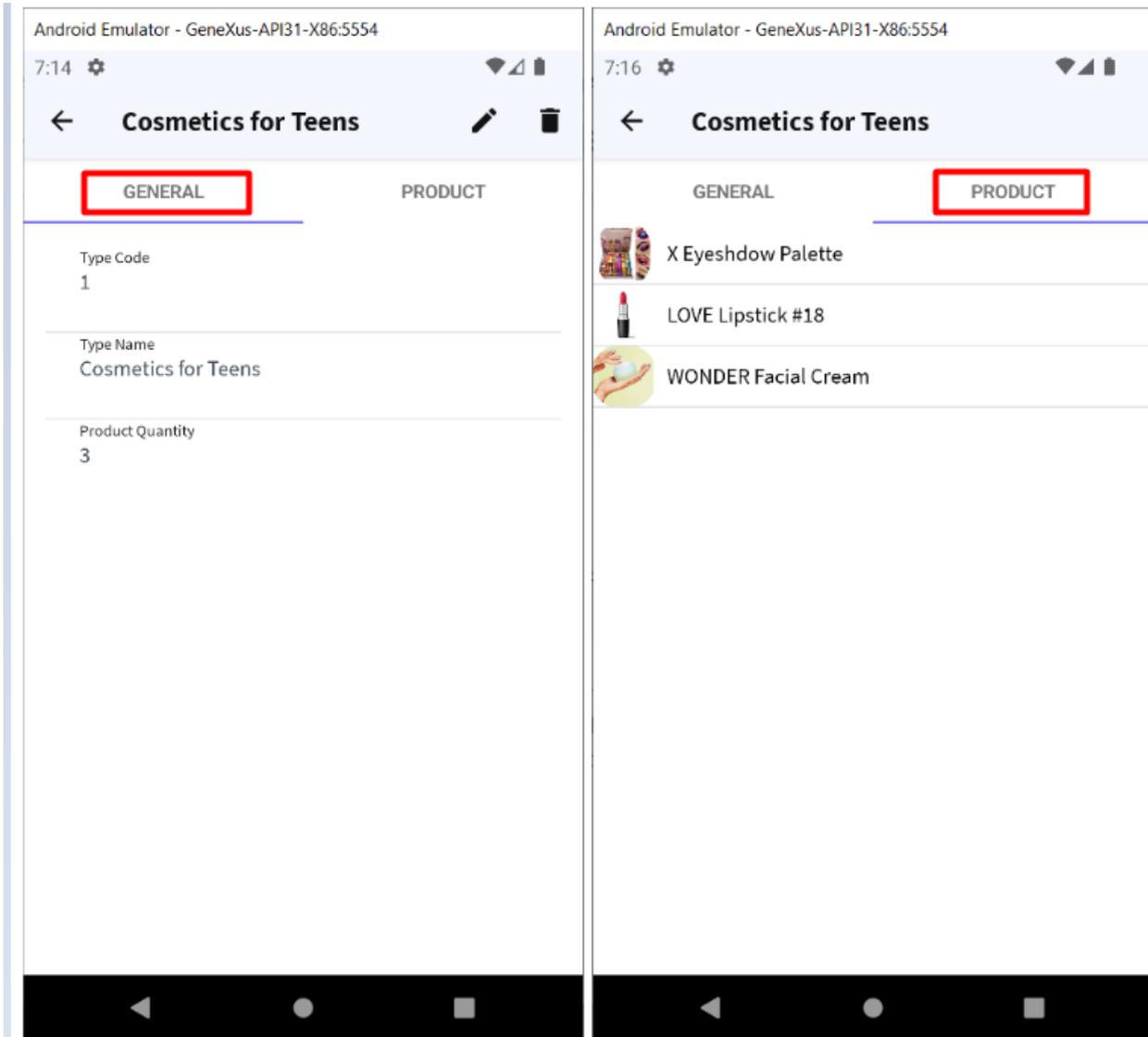


Al hacer tap sobre la imagen, se te muestra la lista de tipos de productos que ofrece la farmacia:



Ten 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 se quiere que ellos puedan insertar nuevos productos; solo deben poder ver los diferentes tipos de productos.

Al hacer tap sobre cada tipo de producto (por ejemplo, "Cosmetics for Teens"), se abre el **Detail** con sus dos secciones GENERAL y PRODUCT:



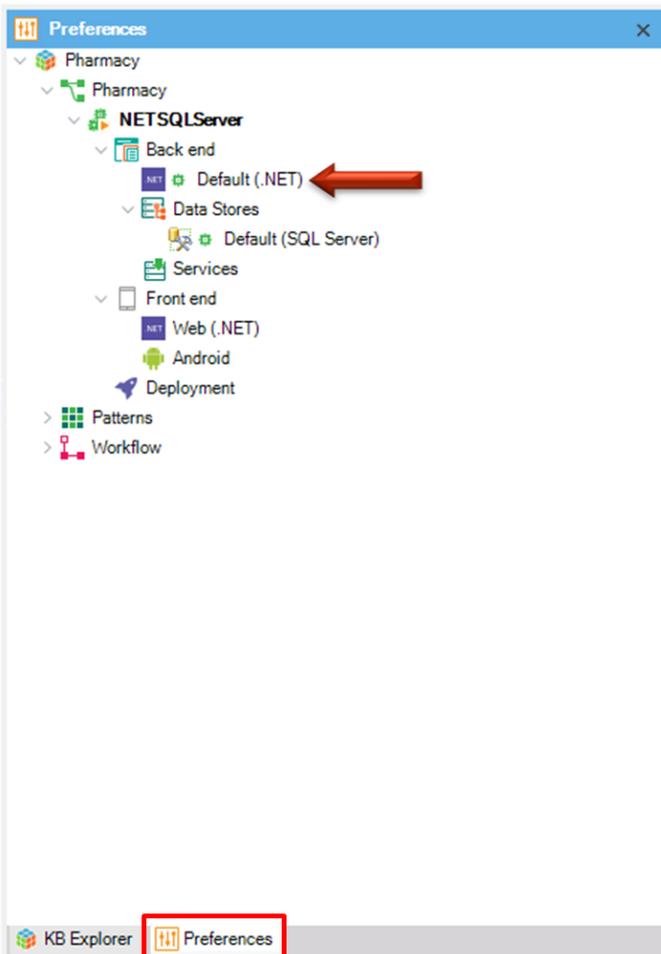
Obviamente, esto es solo una demostración muy simple, ya que se pueden lograr aplicaciones mucho más sofisticadas.

## GENERACIÓN DE LO DEFINIDO HASTA AHORA EN OTRO LENGUAJE Y/O PARA UNA BASE DE DATOS DIFERENTE

Como se mencionó anteriormente, una de las grandes ventajas de GeneXus es que 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** permite configurar y almacenar toda la información **relacionada con una determinada implementación de tu aplicación** (los generadores que deseas utilizar para generar el Back-end de tu aplicación, los generadores que quieres utilizar para generar el Front-end, la información de la base de datos, etc.).

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



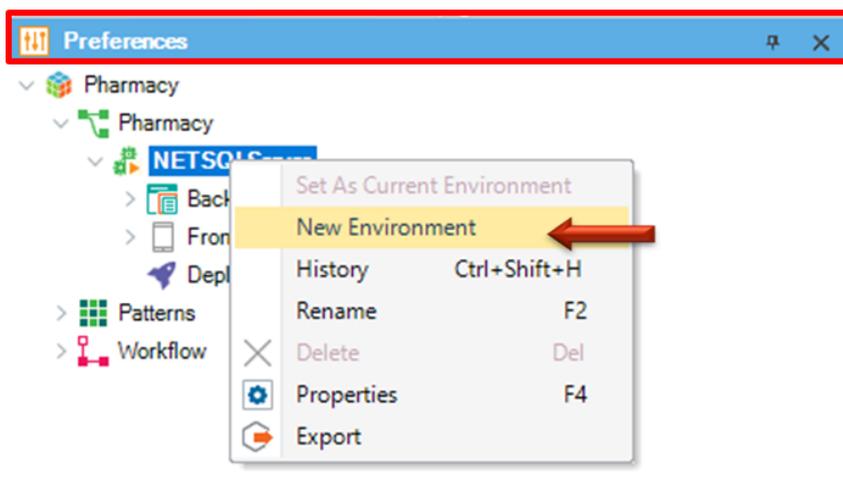
Hay un único Environment definido (**.NET Environment**). Se creó automáticamente al momento de crear la Base de Conocimiento cuando elegiste .NET como lenguaje de generación. Luego, se completó con

más información cuando presionaste F5 por primera vez e indicaste 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 desees 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 podrías usar Apple o Angular).

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



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

Puedes identificar fácilmente el Environment activo por el símbolo PLAY.

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

- **Acceso a Bases de Datos externas**
  - Es posible que necesites 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 necesites permanecer conectado a esa base de datos externa, o puede que necesites conectarte 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 ellas). 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. Luego de hecho esto, nuevos desarrolladores ubicados en cualquier lugar geográfico pueden conectarse a la Base de Conocimiento del servidor y crearse una copia local. 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, incluyendo 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 también puedes desarrollar tus 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**

- Con solo hacer clic en un botón puedes implementar tu aplicación en producción.

- **Documentar dentro de la Base de Conocimiento**

- GeneXus proporciona un editor de Documentación de estilo Wiki, para que puedas 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 puedes describir el propósito del objeto.

Al escribir la documentación, puedes 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**

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

- **Generador de chatbot**

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

- **Extensibilidad**

- 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**

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

## - Gestión de 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

- 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, se asocian 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

- Definición de informes estáticos (listados típicos que pueden imprimirse, guardarse o visualizarse en la pantalla).
- Definición de consultas visuales y dinámicas.
  - Puedes 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**
  - [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**
  - 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**, permitiendo guardar secuencias de operaciones para probar. Luego, las pruebas se reproducen automáticamente, verificando que el sistema aún funciona correctamente.

## PRÓXIMOS PASOS

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

- **Primero lo primero:**
  - Accede al siguiente curso en línea para continuar aprendiendo:
    - <https://training.genexus.com/en/learning/courses/genexus/v18/core>
    - Si aún no has probado GeneXus, puede hacerlo gratis en este link: <http://genexus.com/trial>
  
- **Aprende más:** GeneXus es una plataforma de desarrollo muy completa, y hay mucho que leer y aprender. Podrías comenzar a profundizar en:
  - Sitio de Capacitación de GeneXus: <http://training.genexus.com/>
  - GeneXus Wiki: <http://wiki.genexus.com/>

- ¡**Obtén 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. Pónte en contacto con nosotros a través de [info@genexus.com](mailto:info@genexus.com) o visita <http://genexus.com/plans> para ver cuál te conviene más.
  
- **Sé parte de nuestra comunidad:** cuando estés listo, puedes unirse a nuestra comunidad en constante crecimiento a través de una gran variedad de opciones:
  - Publica tu trabajo en nuestro Marketplace: <http://marketplace.genexus.com/>
  - Trabaja con GeneXus: <http://genexus.com/company/work-with-us?en>
  - Oportunidades ofrecidas por nuestros socios: <http://genexus.com/jobs/Opportunities?en>
  - Sé parte de la GeneXus Alliance: <http://genexus.com/partners>
  - Asiste al próximo Encuentro GeneXus: <http://genexus.com/meetings>

¡Realmente esperamos tener noticias tuyas pronto!

*The GeneXus Team*

Copyright © GeneXus S.A. 1988-2022.

All rights reserved. This document may not be reproduced by any means without the express permission of GeneXus S.A. The information contained herein is intended for personal use only.

**Registered Trademarks:** GeneXus is trademark or registered trademark of GeneXus S.A. All other trademarks mentioned herein are the property of their respective owners.