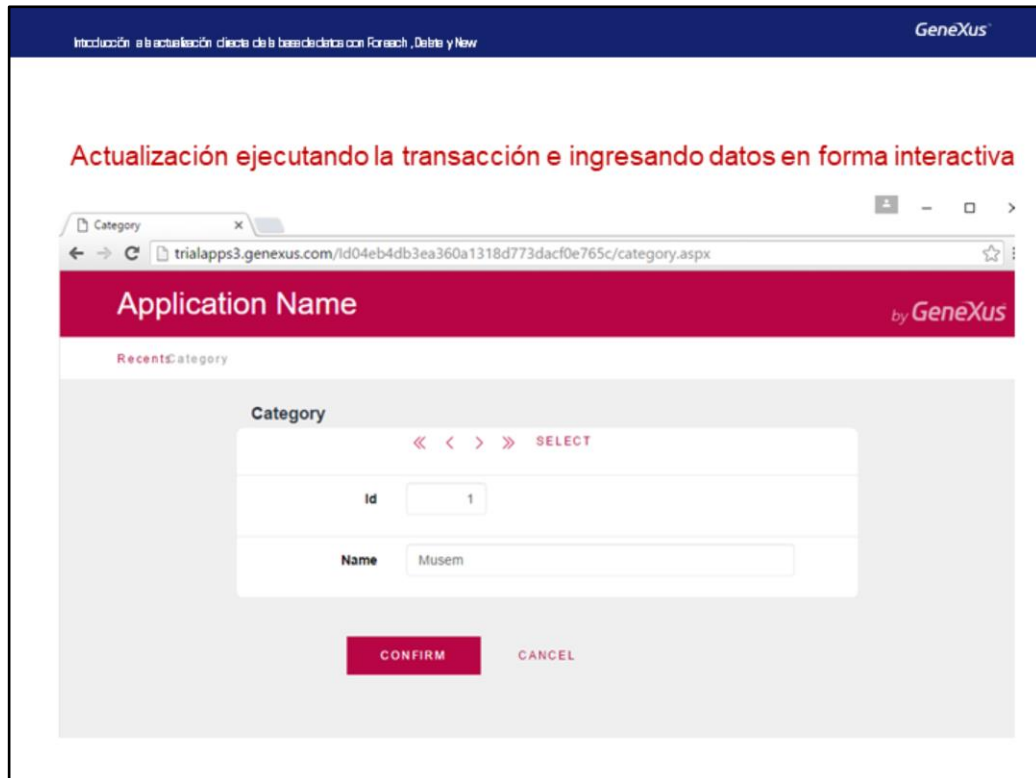


Actualización directa de base de datos con comandos
For each, New y Delete

Introducción

GeneXus 16



Hasta el momento, para actualizar los datos de la base de datos, hemos empleado las transacciones en sus dos formas de uso:

- Ejecutando su pantalla e ingresando datos en forma interactiva

Actualización a través del business component asociado a la transacción

The screenshot displays the GeneXus IDE interface. At the top, the 'Category' business component is selected. The 'Structure' tab shows a table with columns: Name, Type, Description, Formula, and N. The table contains two rows: 'CategoryId' (Type: Id, Description: Category Id, Formula: No, N: No) and 'CategoryName' (Type: Name, Description: Category Name, Formula: No, N: No). The 'Properties' tab on the right shows the 'BusinessComponent: Category' properties. The 'Business Component' property is highlighted with a red box and set to 'True'. Below the main interface, the 'InsertCategoriesUpdateAttractions' transaction is shown. The 'Variables' tab is active, displaying a list of variables: 'Standard Variables' and 'category'. The 'category' variable is highlighted. To the right of the variables list, the following code is shown:

```
&category.CategoryName = "Tourist site"
&category.Save()
```

Y a través de su Business Component asociado, mediante una variable, sin usar la pantalla.

Para la inserción contamos con el comando New

```
New  
    CategoryId = 5  
    CategoryName = "Tourist Site"  
Endnew
```

```
New  
    CategoryName = "Tourist Site"  
Endnew
```

¡Sólo válido en objetos procedimiento!

Pero tenemos otra forma de realizar inserciones, modificaciones y eliminaciones a la base de datos, válida únicamente en objetos Procedimiento.

Para la inserción, contamos con el comando new.

Aquí insertamos una nueva categoría en la tabla CATEGORY. Si CategoryId es autonumerado entonces no le especificamos valor.

Para editar datos utilizamos el comando For Each

```
For each Attraction  
Where CityName = "Beijing" and CategoryName = "Monument"  
    CategoryId = find( CategoryId, CategoryName = "Tourist site")  
Endfor
```

¡Sólo válido en objetos procedimiento!

Para la **actualización** se cuenta con el propio comando for each. Por ejemplo, para cambiar a todas las atracciones de Beijing que tuvieran categoría Monument, a la nueva categoría Tourist Site, escribimos un comando For Each y asignamos los nuevos valores a los atributos que queremos actualizar.

Aquí se está actualizando un atributo de la tabla base, aunque bien podría ser de la tabla extendida.

Para eliminar datos usamos el comando Delete

```
For each Attraction  
Delete  
Endfor
```

¡Sólo válido en objetos procedimiento!

Y para la **eliminación**, una vez posicionados sobre el registro a ser eliminado, contamos con el comando Delete

Ventajas y desventajas de usar estos comandos en los procedimientos

- Desventajas:
 - No se realizan controles de integridad referencial, por lo cual los datos pueden quedar inconsistentes
 - No disparan ninguna regla de la transacción
- Ventajas:
 - Son más rápidos (más "performantes")

Estos comandos tienen la desventaja, frente a la solución que vimos con business components, de que no controlan la integridad referencial, ni disparan ninguna regla de la transacción.

Su ventaja es, justamente, que son más rápidos, lo que también se conoce como "más performantes".

Si usted tiene que eliminar los datos de una tabla entera de millones de registros, hacerlo de esta manera, con el comando Delete, insumirá mucho menos tiempo. Sin embargo, esta eliminación no hará controles y podría quedarle la base de datos en estado inconsistente.

No estudiaremos estos comandos en este curso. Le sugerimos, si desea verlos en detalle, acceder al material correspondiente al curso para Analista GeneXus

GeneXus™

The power of doing.

Videos

Documentation

Certifications

training.genexus.com

wiki.genexus.com

training.genexus.com/certifications