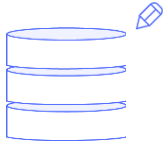


# Actualización de la base de datos

Business Components. Reglas, eventos y chequeos. Repaso parte 2.

GeneXus™

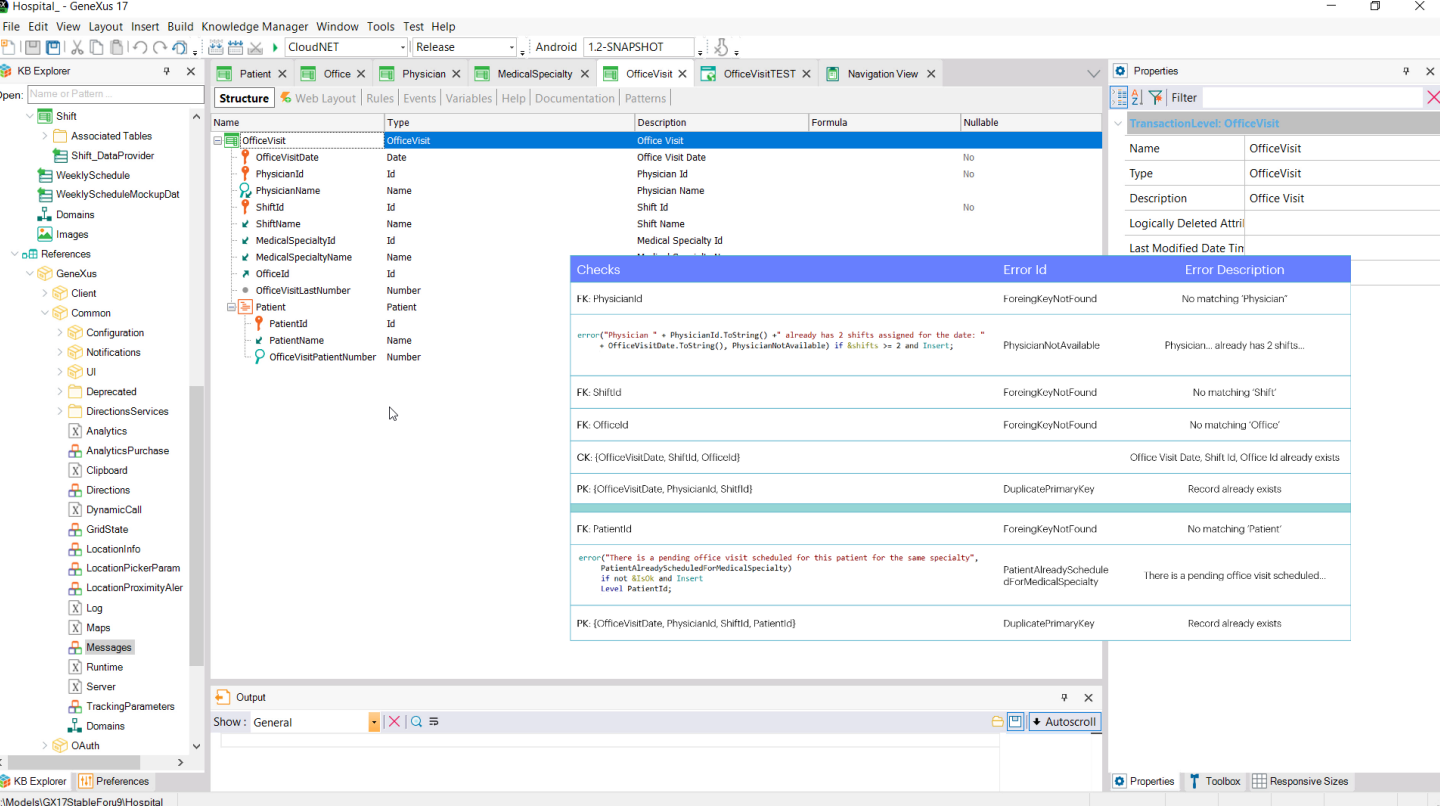


Insert, Update, Delete

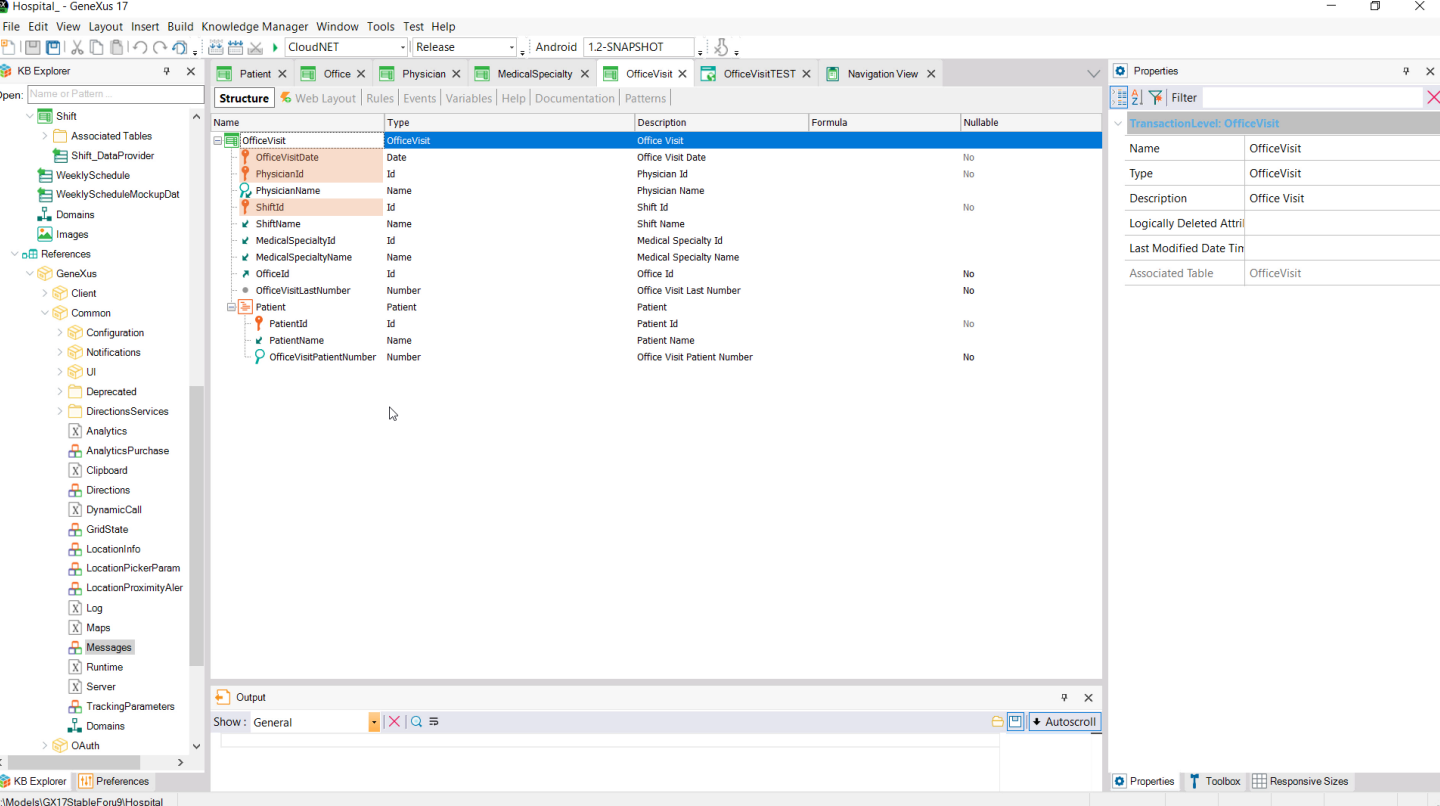
## Business Component

KB Hospital

Centrados en una KB para un servicio de salud, en el video anterior observamos todas las reglas, eventos y chequeos que se realizaban al intentar insertar...

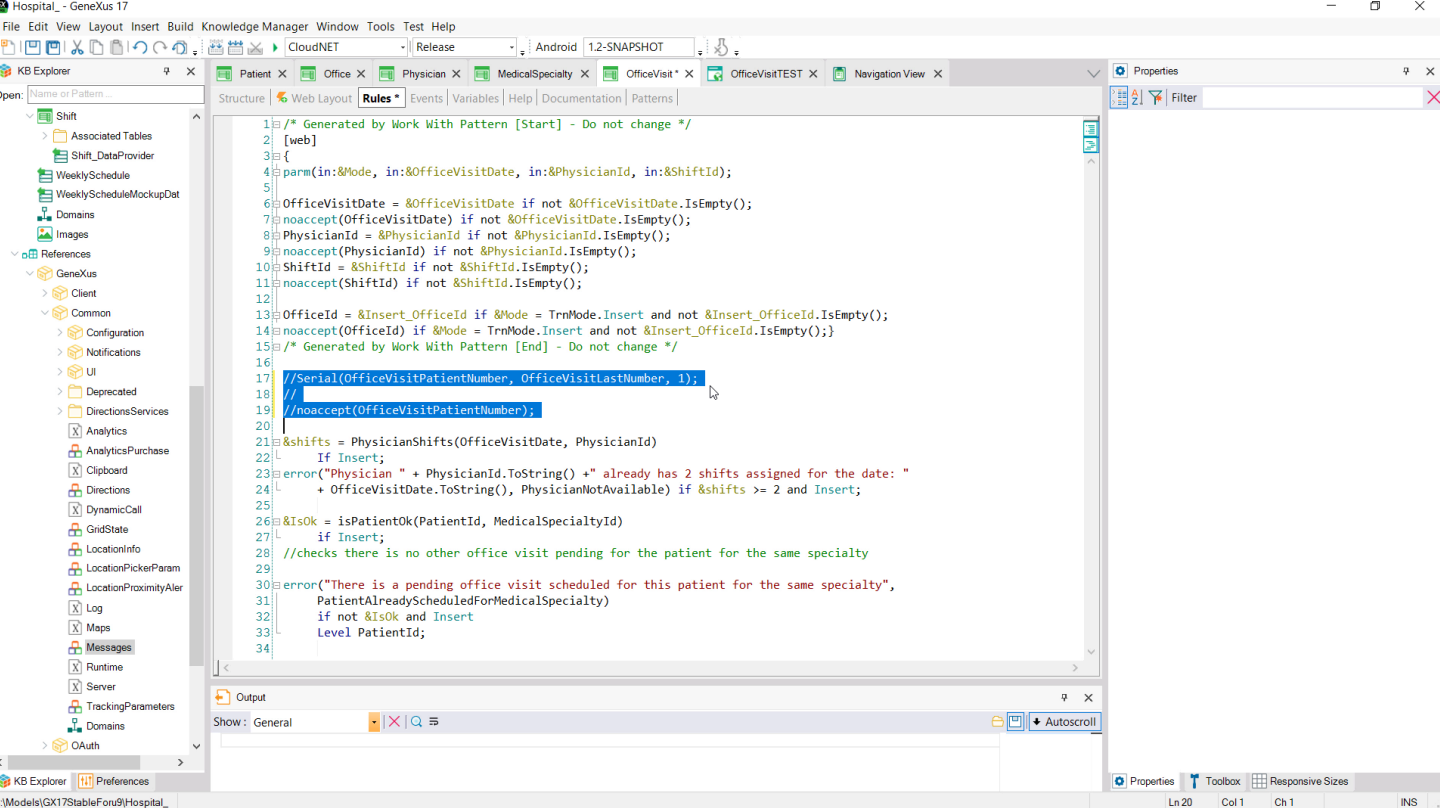


...en una transacción de dos niveles y a través del Business Component.  
Aquí atenderemos la actualización y eliminación.



Querremos modificar una consulta. Como fecha, médico y turno forman parte de la clave primaria ninguno de estos atributos puede ser modificado para un registro existente. Solo podremos modificar el consultorio porque el last number se utiliza para la regla serial, por lo que no puede cambiarse.

El paciente de una línea tampoco puede modificarse por ser parte de la clave primaria. Y solo podría cambiarse el atributo secundario correspondiente al número que se entrega al paciente para ser atendido, salvo por el hecho de que colocamos una regla noaccept para ese atributo.



Podemos comentarla para tener un atributo actualizable para cada línea. Dejemos comentada tanto la regla serial como la noaccept, así podemos modificar también el atributo del cabezal, el del último número de línea dado.

Por supuesto además podemos agregar y eliminar líneas a la consulta.

Bien, queremos entonces poder modificar el consultorio de una consulta dada, cambiarle el número asignado a un paciente agendado, agregar otro paciente con su número respectivo, y eliminar uno.

	Monday	Tuesday	Wednesday	Thursday	Friday
<b>Shift 1</b> morning	<div style="border: 1px solid red; padding: 2px; margin-bottom: 5px;">Physician: 1 Office: 2</div> <div style="border: 1px dashed gray; padding: 2px; margin-bottom: 5px;">Physician: 2 Office: 3</div> <div style="border: 1px dashed gray; padding: 2px;">Patient: 1</div>		<div style="border: 1px solid red; padding: 2px;">Physician: 2 Office: 1</div>	<div style="border: 1px solid red; padding: 2px; margin-bottom: 5px;">Physician: 1 Office: 2 → 3</div> <div style="border: 1px solid gray; padding: 2px; margin-bottom: 5px;">Patient: 2</div> <div style="border: 1px dashed gray; padding: 2px;">Patient: 3</div>	
<b>Shift 2</b> intermediate	<div style="border: 1px solid red; padding: 2px;">Physician: 1 Office: 2</div>				
<b>Shift 3</b> evening		<div style="border: 1px solid red; padding: 2px;">Physician: 1 Office: 2</div>			

Empecemos por esto: actualizar la consulta del jueves, cambiando el consultorio por el 3 y agregándole al paciente 3... si lo hacemos vía la transacción, primero tenemos que ubicar el registro a ser actualizado...

# Hospital Backoffice



Recents Office Visit — Office Visits

SHOW FILTERS

## Office Visits

Q Name

+ INSERT

Visit Date	Physician Id	Physician Name	Shift Name	Medical Specialty ...	Office Id	Last Number		
11/14/22	1	Doctor 1	Shift morning	Family Medicine	2	0	UPDATE	DELETE
11/14/22	1	Doctor 1	Shift intermediate	Family Medicine	2	0	UPDATE	DELETE
11/14/22	2	Doctor 2	Shift morning	Family Medicine	3	1	UPDATE	DELETE
11/15/22	1	Doctor 1	Shift evening	Family Medicine	2	0	UPDATE	DELETE
11/16/22	2	Doctor 2	Shift morning	Family Medicine	1	0	UPDATE	DELETE
11/17/22	1	Doctor 1	Shift morning	Family Medicine	2	1	UPDATE	DELETE
11/21/22	1	Doctor 1	Shift morning	Family Medicine	1	1	UPDATE	DELETE

https://apps5.genexus.com/Id9df182c70350001219afeaceeb30e8e2/officevisit.aspx?Mode=U...

... al presionar aquí se llama a la transacción pasándole por parámetro la clave primaria y el modo Update...

GeneXus Developer Menu x Office Visit x +

apps5.genexus.com/Id9df182c70350001219afeaceeb30e8e2/officevisit.aspx?Mode=UPD&OfficeVisitDate=20221117&PhysicianId=1&ShiftId=1

GeneXus DL Portal

# Hospital Backoffice

by GeneXus

Recents Office Visits — Office Visit

## Office Visit

Visit Date	11/17/22
Physician Id	1
Physician Name	Doctor 1
Shift Id	1
Shift Name	Shift morning
Medical Specialty Id	1
Medical Specialty Name	Family Medicine
Office Id	<input type="text" value="1"/> <input type="button" value="🔍"/> <input type="button" value="🔄"/>
Last Number	<input type="text" value="1"/>

TrnMode.Update

Con lo cuál se va a buscar el registro de esa clave primaria a la tabla del cabezal y se trae su información, y todos los registros relacionados a la tabla del segundo nivel y se traen sus valores.



GeneXus Developer Menu x Office Visit x +

apps5.genexus.com/ld9df162c70350001219afeaceeb30e8e2/officevisit.aspx?Mode=UPD&OfficeVisitDate=20221117&PhysicianId=1&ShiftId=1

GeneXus DL Portal

Visit Date	11/17/22
Physician Id	1
Physician Name	Doctor 1
Shift Id	1
Shift Name	Shift morning
Medical Specialty Id	1
Medical Specialty Name	Family Medicine
Office Id	<input type="text" value="3"/>
Last Number	<input type="text" value=""/>

Checks	Error Id	Error Description
FK: PhysicianId	ForeignKeyNotFound	No matching 'Physician'
error["Physician " + PhysicianId.ToString() + " already has 2 shifts assigned for the date: " + OfficeVisitDate.ToString(), PhysicianId] if Shifts >= 2 and Insert;	PhysicianNotAvailable	Physician... already has 2 shifts...
FK: ShiftId	ForeignKeyNotFound	No matching 'Shift'
FK: OfficeId	ForeignKeyNotFound	No matching 'Office'
CK: (OfficeVisitDate, ShiftId, OfficeId)		Office Visit Date, Shift Id, Office Id already exists
PK: (OfficeVisitDate, PhysicianId, ShiftId)	DuplicatePrimaryKey	Record already exists
FK: PatientId	ForeignKeyNotFound	No matching 'Patient'
error["There is a pending office visit scheduled for this patient for the same specialty", PatientAlreadyScheduledForMedicalSpecialty (if not Edit and Insert Level), PatientId];	PatientAlreadySchedule dFromMedicalSpecialty	There is a pending office visit scheduled...
PK: (OfficeVisitDate, PhysicianId, ShiftId, PatientId)	DuplicatePrimaryKey	Record already exists

Patient Id	Patient Name	Patient Number
x 2	Patient 2	1
0		0

Vemos que el campo OfficeId es el primero editable. Ninguno de los anteriores puede ser modificado. Lo cambiamos por 3. Por supuesto, si hubiera otra consulta para ese día en ese turno en ese consultorio, aquí se nos arrojaría un error de que esa clave (candidata) ya existe. No es el caso.

GeneXus Developer Menu Office Visit

apps5.genexus.com/Id9df182c70350001219afeaceeb30e8e2/officevisit.aspx?Mode=UPD&OfficeVisitDate=20221117&PhysicianId=1&ShiftId=1

GeneXus DL Portal

Medical Specialty Id	1
Medical Specialty Name	Family Medicine
Office Id	<input type="text" value=""/>
Last Number	<input type="text" value="3"/>

**Patient**

	Patient Id	Patient Name	Patient Number
×	2	Patient 2	<input type="text" value="2"/>
×	3	Patient 3	<input type="text" value="3"/>
	<input type="text" value="0"/>	<input type="text" value=""/>	<input type="text" value="0"/>
	<input type="text" value="0"/>	<input type="text" value=""/>	<input type="text" value="0"/>
	<input type="text" value="0"/>	<input type="text" value=""/>	<input type="text" value="0"/>
	<input type="text" value="0"/>	<input type="text" value=""/>	<input type="text" value="0"/>
	<input type="text" value="0"/>	<input type="text" value=""/>	<input type="text" value="0"/>

[New row]

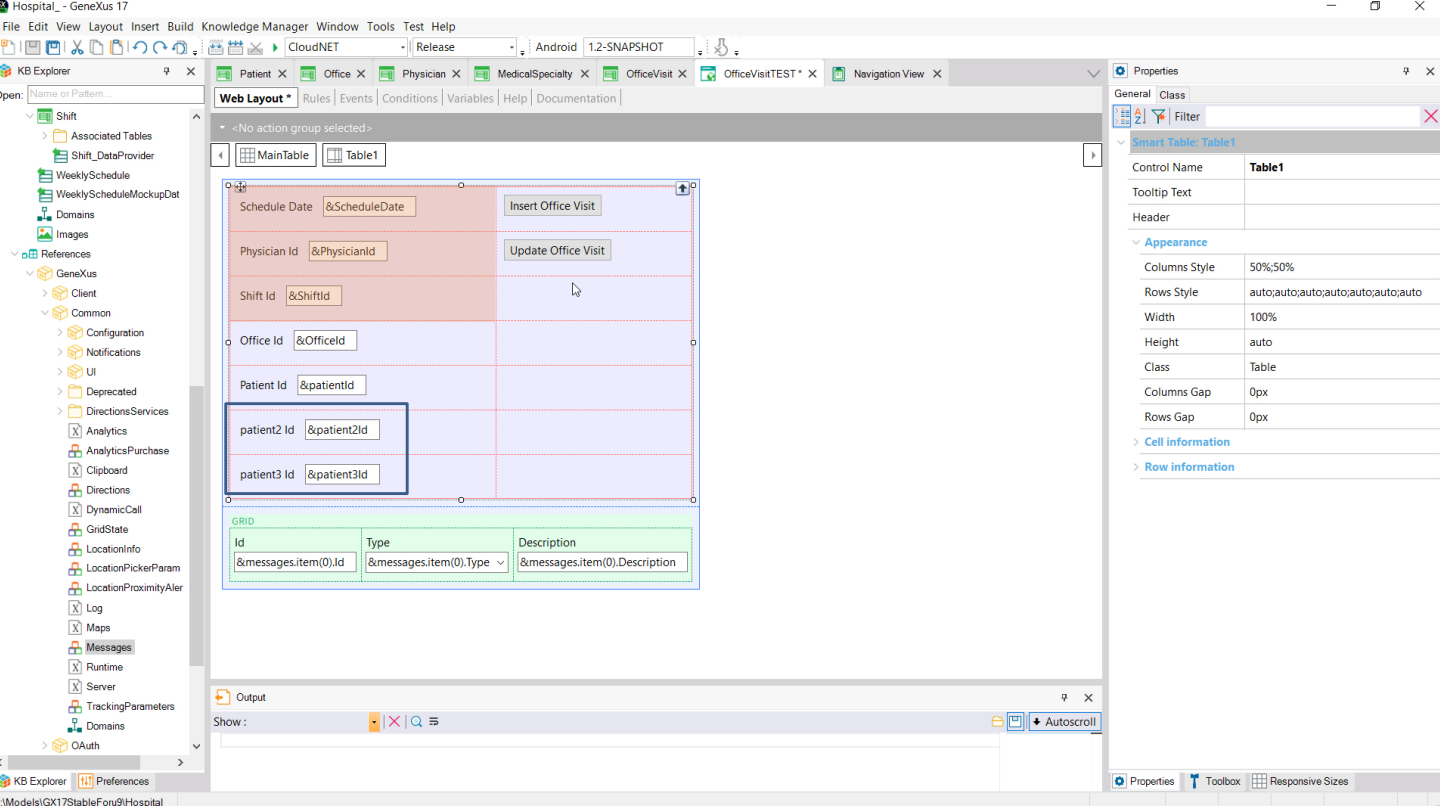
CONFIRM CANCEL

Cambiamos el número dado al primer paciente por 2, agreguemos al paciente 3, con número 3, y cambiamos también el último número, por el 3. Al confirmar la operación que se realizará será la actualización sobre el todo, aunque a nivel de las líneas eso corresponda a modificar una y agregar otra.

	Monday	Tuesday	Wednesday	Thursday	Friday
<b>Shift 1</b> morning	<div style="border: 1px solid red; padding: 2px; margin-bottom: 5px;">Physician: 1 Office: 2</div> <div style="border: 1px dashed gray; padding: 2px; margin-bottom: 5px;">Physician: 2 Office: 3</div> <div style="border: 1px dashed gray; padding: 2px;">Patient: 1</div>		<div style="border: 1px solid red; padding: 2px;">Physician: 2 Office: 1</div>	<div style="border: 1px solid red; padding: 2px; margin-bottom: 5px;">Physician: 1 Office: 3 → 2</div> <div style="border: 1px solid red; padding: 2px; margin-bottom: 5px;">Patient: 2</div> <div style="border: 1px solid red; padding: 2px; margin-bottom: 5px;">Patient: 3 → 4</div> <div style="border: 1px dashed gray; padding: 2px;">Patient: 4</div>	
<b>Shift 2</b> intermediate	<div style="border: 1px solid red; padding: 2px;">Physician: 1 Office: 2</div>				
<b>Shift 3</b> evening		<div style="border: 1px solid red; padding: 2px;">Physician: 1 Office: 2</div>			

Ahora queremos operar así, pero a través del Business Component.

Supongamos que además de cambiar el consultorio, vamos a querer cambiar al paciente 3 por el 4, y como no podemos hacerlo directamente en el registro por ser parte de la clave primaria y esta no se puede actualizar, deberemos eliminar el registro e insertar uno nuevo con el paciente correcto y el mismo número de atención que tenía el 3.



Hemos agregado 2 variables más para el paciente que queremos reemplazar por el otro para la consulta de esa fecha, médico y turno. Agregaremos un evento para actualizar la consulta de esta clave primaria...

File Edit View Layout Insert Build Knowledge Manager Window Tools Test Help

CloudNET Release Android 1.2-SNAPSHOT

KB Explorer

Open: Name or Pattern

- Shift
- Associated Tables
- Shift\_DataProvider
- WeeklySchedule
- WeeklyScheduleMockupDat
- Domains
- Images
- References
- GeneXus
- Client
- Common
- Configuration
- Notifications
- UI
- Deprecated
- DirectionsServices
- Analytics
- AnalyticsPurchase
- Clipboard
- Directions
- DynamicCall
- GridState
- LocationInfo
- LocationPickerParam
- LocationProximityAler
- Log
- Maps
- Messages
- Runtime
- Server
- TrackingParameters
- Domains
- OAuth

Web Layout\* Rules Events\* Conditions Variables Help Documentation

'Update Office Visit'

```

1 Event 'Insert Office Visit'
2   &officeVisit = new()
3   &officeVisit.OfficeVisitDate = &ScheduleDate
4   &officeVisit.PhysicianId = &PhysicianId
5   &officeVisit.ShiftId = &ShiftId
6   &officeVisit.OfficeId = &OfficeId
7   If not &patientId.IsEmpty()
8     &officeVisitPatient = new()
9     &officeVisitPatient.PatientId = &patientId
10    &officeVisit.Patient.Add(&officeVisitPatient)
11  endif
12  if &officeVisit.Insert()
13    Commit
14  endif
15  &messages = &officeVisit.GetMessages()
16 Endevent
17
18
19 Event 'Update Office Visit'
20   &officeVisit.Load(&ScheduleDate, &PhysicianId, &ShiftId)
21
22 Endevent

```

&officeVisit

OfficeVisitDate	#2022-11-17#
PhysicianId	1
PhysicianName	Doctor 1
ShiftId	1
ShiftName	Shift morning
MedicalSpecialtyId	1
MedicalSpecialtyName	Family Medicine
OfficeId	3
OfficeVisitLastNumber	3
Patient	

BC loaded! ←

PatientId	?
Patient Name	Patient 2
OfficeVisitPatientNumber	?

PatientId	3
Patient Name	Patient 3
OfficeVisitPatientNumber	3

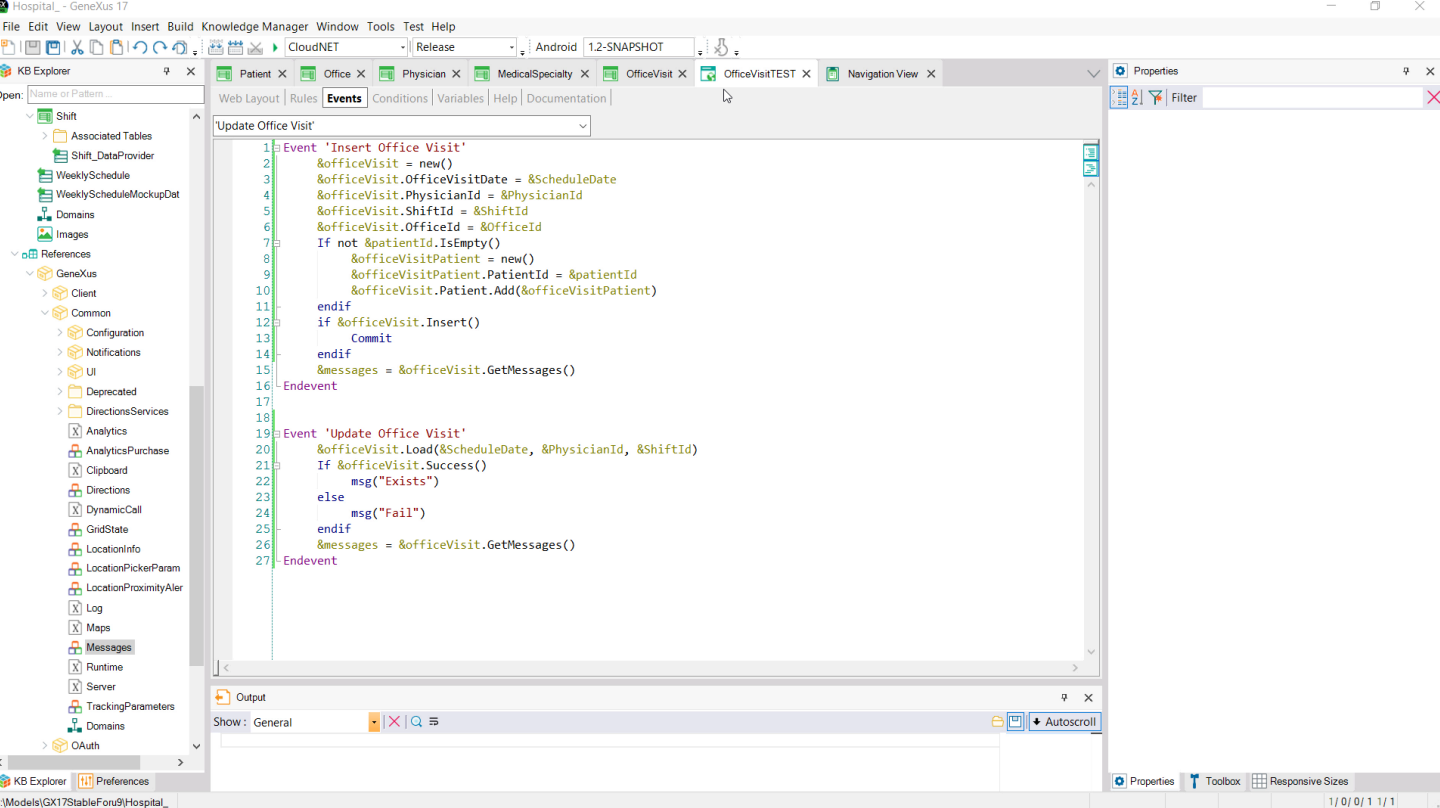
Output

Show: General

Properties

Ln 21 Col 6 Ch 2 INS

Lo primero que debemos hacer es cargar en la variable BC los datos correspondientes a la clave primaria. La operación solo será exitosa si existe un registro en OfficeVisit con esa clave.



Podemos consultarlo con el método Success (o su contrario, el Fail). Mostraremos en pantalla mensajes si existe o no el registro que se pretendió cargar. También recuperemos los mensajes en la colección de mensajes. Probemos rápido.

# Hospital Backoffice



Recents Office Visit TEST — Office Visits

Office Visits

Q Name

+ INSERT

X HIDE FILTERS

MEDICAL SPECIALTY NAME

OFFICE VISIT DATE

SHIFT NAME

Visit Date	Physician Id	Physician Name	Shift Name	Medical Specialt...	Office Id	Last Number		
11/14/22	1	Doctor 1	Shift morning	Family Medicine	2	0	UPDATE	DELETE
11/14/22	1	Doctor 1	Shift intermediate	Family Medicine	2	0	UPDATE	DELETE
11/14/22	2	Doctor 2	Shift morning	Family Medicine	3	1	UPDATE	DELETE
11/15/22	1	Doctor 1	Shift evening	Family Medicine	2	0	UPDATE	DELETE
11/16/22	2	Doctor 2	Shift morning	Family Medicine	1	0	UPDATE	DELETE
11/17/22	1	Doctor 1	Shift morning	Family Medicine	3	3	UPDATE	DELETE
11/21/22	1	Doctor 1	Shift morning	Family Medicine	1	1	UPDATE	DELETE

Este registro (el del jueves) sabemos que existe...

# Hospital Backoffice



Recents Office Visit TEST

Exists

Schedule Date

Physician Id

Shift Id

Office Id

Patient Id

patient2 Id

patient3 Id

Id	Type	Description
----	------	-------------

... así que probamos con su clave... ¡se encontró!



# Hospital Backoffice



Recents Office Visit TEST

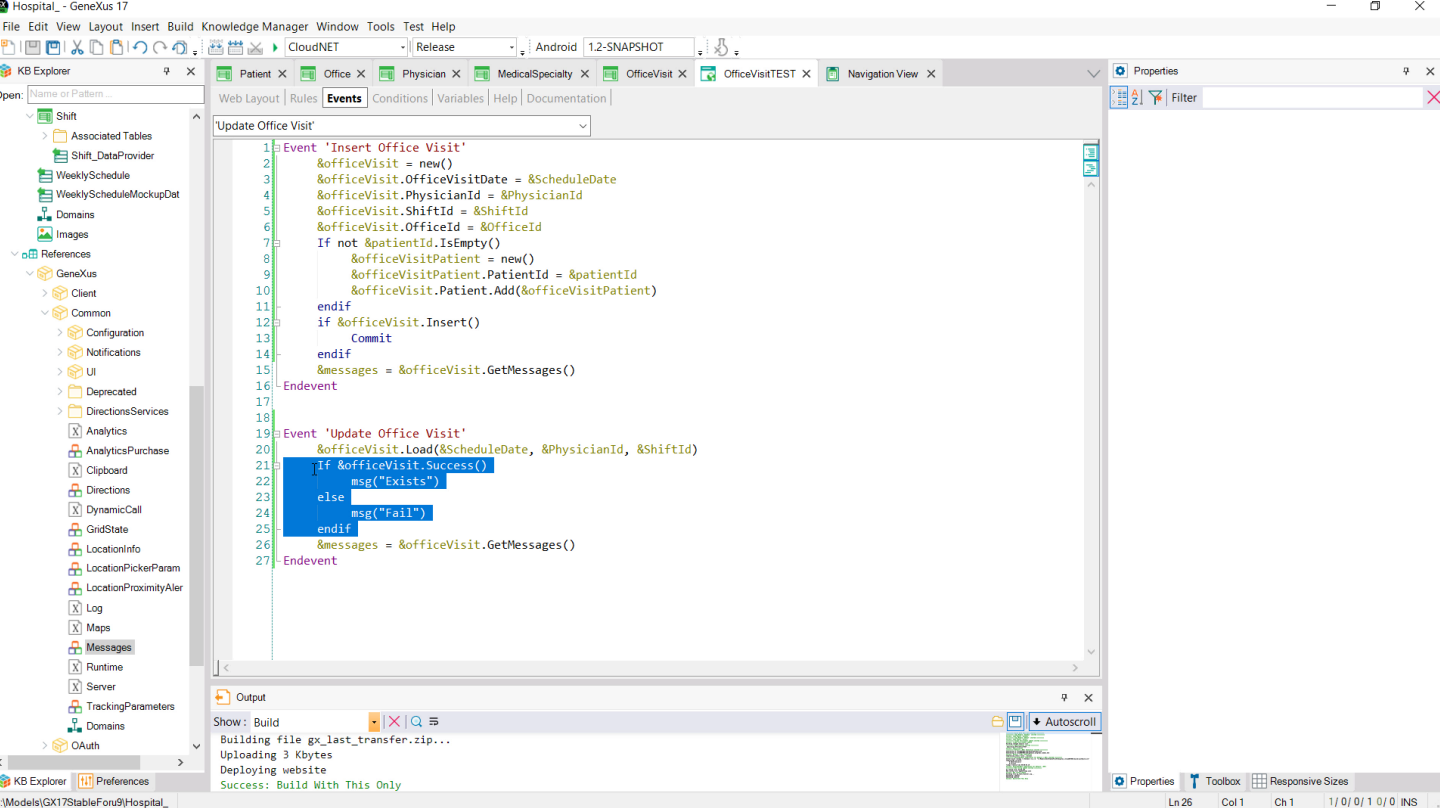
**Fail**

Schedule Date	11/27/22	29	Insert Office Visit
Physician Id	1		Update Office Visit
Shift Id	1		
Office Id	0		
Patient Id	0		
patient2 Id	0		
patient3 Id	0		

Id	Type	Description
PrimaryKeyNotFound	Error	Data with the specified key could not be found.

Y ahora ponemos una fecha para la que no hay consultas... Vemos esto otro.

El método Success resultó False y vemos que en la colección de mensajes nos indica el error de clave primaria no encontrada.



Quitamos este control porque de todos modos se hará cuando demos la orden de actualizar.

File Edit View Layout Insert Build Knowledge Manager Window Tools Test Help

CloudNET | Release | Android | 1.2-SNAPSHOT

KB Explorer | Patient x | Office x | Physician x | MedicalSpecialty x | OfficeVisit x | OfficeVisitTEST x | Navigation View x | Properties

Web Layout | Rules | **Events** | Conditions | Variables | Help | Documentation

'Update Office Visit'

```

7 | If not &patientId.IsEmpty()
8 |     &officeVisitPatient = new()
9 |     &officeVisitPatient.PatientId = &patientId
10 |     &officeVisit.Patient.Add(&officeVisitPatient)
11 | endif
12 | if &officeVisit.Insert()
13 |     Commit
14 | endif
15 | &messages = &officeVisit.GetMessages()
16 | -Endevent
17 |
18 |
19 | Event 'Update Office Visit'
20 |     &officeVisit.Load(&scheduleDate, &physicianId, &shiftId)
21 |     &officeVisit.OfficeId = &officeId
22 |     &officeVisit.OfficeVisitLastNumber +=1
23 |
24 |     //Update line for &patientId
25 |
26 |     &officeVisitPatientNumber = &officeVisit.Patient.GetByKey(&patient2Id).OfficeVisitPatientNumber
27 |     //Delete line for &patient2Id
28 |
29 |     &officeVisitPatient = new()
30 |     &officeVisitPatient.PatientId = &patient3Id
31 |     &officeVisitPatient.OfficeVisitPatientNumber = &officeVisitPatientNumber
32 |     //Insert line for &patient3Id
33 |
34 |     If &officeVisit.Update()
35 |         commit
36 |     endif
37 |     &messages = &officeVisit.GetMessages()
38 | -Endevent

```

OfficeVisitDate	#2022-11-17#
PhysicianId	1
PhysicianName	Doctor 1
ShiftId	1
ShiftName	Shift morning
MedicalSpecialtyId	1
MedicalSpecialtyName	Family Medicine
OfficeId	2
OfficeVisitLastNumber	4
Patient	

PatientId	2
Patient Name	Patient 2
OfficeVisitPatientNumber	2

PatientId	3
Patient Name	Patient 3
OfficeVisitPatientNumber	3

Output

Show: Build | X | Q | S | Autoscroll

Building file gx\_last\_transfer.zip...  
 Uploading 50 Kbytes  
 Deploying website  
 Success: Build With This Only

Properties | Toolbox | Responsive Sizes  
 Ln 22 | Col 44 | Ch 40 | 1 / 0 / 1 / 1 / 1 INS

Entonces, modificamos el consultorio de acuerdo a la variable de pantalla; le sumamos 1 al último número dado...

File Edit View Layout Insert Build Knowledge Manager Window Tools Test Help

CloudNET Release Android 1.2-SNAPSHOT

KB Explorer Patient x Office x Physician x MedicalSpecialty x OfficeVisit x OfficeVisitTEST x Navigation View x Properties

Web Layout | Rules **Events** | Conditions | Variables | Help | Documentation

'Update Office Visit'

```

7 If not &patientId.IsEmpty()
8     &officeVisitPatient = new()
9     &officeVisitPatient.PatientId = &patientId
10    &officeVisit.Patient.Add(&officeVisitPatient)
11 endif
12 if &officeVisit.Insert()
13     Commit
14 endif
15 &messages = &officeVisit.GetMessages()
16 -Endevent
17
18
19 Event 'Update Office Visit'
20     &officeVisit.Load(&scheduleDate, &physicianId, &shiftId)
21     &officeVisit.OfficeId = &officeId
22     &officeVisit.OfficeVisitLastNumber +=1
23
24     //Update line for &patientId
25
26     &officeVisitPatientNumber = &officeVisit.Patient.GetByKey(&patient2Id).OfficeVisitPatientNumber
27     //Delete line for &patient2Id
28
29     &officeVisitPatient = new()
30     &officeVisitPatient.PatientId = &patient3Id
31     &officeVisitPatient.OfficeVisitPatientNumber = &officeVisitPatientNumber
32     //Insert line for &patient3Id
33
34     If &officeVisit.Update()
35         commit
36     endif
37     &messages = &officeVisit.GetMessages()
38 -Endevent

```

OfficeVisitDate	#2022-11-17E
PhysicianId	1
PhysicianName	Doctor 1
ShiftId	1
ShiftName	Shift morning
MedicalSpecialtyId	1
MedicalSpecialtyName	Family Medicine
OfficeId	2
OfficeVisitLastNumber	4
Patient	

PatientId	2
Patient Name	Patient 2
OfficeVisitPatientNumber	2

PatientId	3
Patient Name	Patient 3
OfficeVisitPatientNumber	3

Output

Show: Build | X | Q | Autoscroll

Building file gx\_last\_transfer.zip...  
 Uploading 50 Kbytes  
 Deploying website  
 Success: Build With This Only

Properties | Toolbox | Responsive Sizes  
 Ln 24 Col 32 Ch 28 1/0/0 1/1/1 INS

... porque es por el que vamos a cambiarle el número al primer paciente, el de esta variable.

File Edit View Layout Insert Build Knowledge Manager Window Tools Test Help

CloudNET | Release | Android | 1.2-SNAPSHOT

KB Explorer | Patient x | Office x | Physician x | MedicalSpecialty x | OfficeVisit x | OfficeVisitTEST x | Navigation View x | Properties

Web Layout | Rules | **Events** | Conditions | Variables | Help | Documentation

'Update Office Visit'

```

7 | If not &patientId.IsEmpty()
8 |     &officeVisitPatient = new()
9 |     &officeVisitPatient.PatientId = &patientId
10 |     &officeVisit.Patient.Add(&officeVisitPatient)
11 | endif
12 | if &officeVisit.Insert()
13 |     Commit
14 | endif
15 | &messages = &officeVisit.GetMessages()
16 | -Endevent
17 |
18 |
19 | Event 'Update Office Visit'
20 |     &officeVisit.Load(&ScheduleDate, &PhysicianId, &ShiftId)
21 |     &officeVisit.OfficeId = &OfficeId
22 |     &officeVisit.OfficeVisitLastNumber +=1
23 |
24 | //Update line for &patientId
25 |     &officeVisitPatient = &officeVisit.Patient.GetByKey()
26 |
27 |     &officeVisitPatientNumber = &officeVisit.Patient.GetByKey(&patient2Id).OfficeVisitPatientNumber
28 | //Delete line for &patient2Id
29 |
30 |
31 |     &officeVisitPatient = new()
32 |     &officeVisitPatient.PatientId = &patient3Id
33 |     &officeVisitPatient.OfficeVisitPatientNumber = &officeVisitPatientNumber
34 | //Insert line for &patient3Id
35 |
36 | If &officeVisit.Update()
37 |     commit
38 | endif
39 | &messages = &officeVisit.GetMessages()
40 | -Endevent

```

&officeVisit

OfficeVisitDate	#2022.11.17#
PhysicianId	1
PhysicianName	Doctor 1
ShiftId	1
ShiftName	Shift morning
MedicalSpecialtyId	1
MedicalSpecialtyName	Family Medicine
OfficeId	2
OfficeVisitLastNumber	4
Patient	—

PatientId	2
Patient Name	Patient 2
OfficeVisitPatientNumber	2

PatientId	3
Patient Name	Patient 3
OfficeVisitPatientNumber	3

GetByKey(PrimaryKeyAtt: Variant): Patient

Output

Show: Build | X | Q | Autoscroll

Building file gx\_last\_transfer.zip...  
 Uploading 50 Kbytes  
 Deploying website  
 Success: Build With This Only

Properties | Toolbox | Responsive Sizes  
 Ln 25 | Col 58 | Ch 54 | 1 / 0 / 1 / 1 / 1 INS

Para ello recordemos que tenemos que utilizar una variable de tipo de datos el BC de cada línea; y utilizar el método GetByKey de la colección de pacientes del BC que cargamos aquí.

File Edit View Layout Insert Build Knowledge Manager Window Tools Test Help

CloudNET Release Android 1.2-SNAPSHOT

KB Explorer

GeneXus

- Client
- Common
- Configuration
- Notifications
- UI
- Deprecated
- DirectionsServices
- Analytics
- AnalyticsPurchase
- Clipboard
- Directions
- DynamicCall
- GridState
- LocationInfo
- LocationPickerParam
- LocationProximityAler
- Log
- Maps
- Messages
- Runtime
- Server
- TrackingParameters
- Domains
- OAuth
- SD
- Server
- Social
- Domains
- GXtest
- Customization
- Documentation

Web Layout | Rules | **Events** | Conditions | Variables | Help | Documentation

'Update Office Visit'

```

7 If not &patientId.IsEmpty()
8     &officeVisitPatient = new()
9     &officeVisitPatient.PatientId = &patientId
10    &officeVisit.Patient.Add(&officeVisitPatient)
11 endif
12 if &officeVisit.Insert()
13     Commit
14 endif
15 &messages = &officeVisit.GetMessages()
16 -Endevent
17
18
19 Event 'Update Office Visit'
20     &officeVisit.Load(&scheduleDate, &physicianId, &shiftId)
21     &officeVisit.OfficeId = &officeId
22     &officeVisit.OfficeVisitLastNumber +=1
23
24 //Update line for &patientId
25 &officeVisitPatient = &officeVisit.Patient.GetByKey(
26                                     &patientId)
27
28 &officeVisitPatientNumber = &officeVisit.Patient.GetByKey(&patientId).OfficeVisitPatientNumber
29 //Delete line for &patient2Id
30
31 &officeVisitPatient = new()
32 &officeVisitPatient.PatientId = &patient3Id
33 &officeVisitPatient.OfficeVisitPatientNumber = &officeVisitPatientNumber
34 //Insert line for &patient3Id
35
36 If &officeVisit.Update()
37     commit
38 endif
39 &messages = &officeVisit.GetMessages()
40 -Endevent

```

&officeVisit

OfficeVisitDate	#2022.11.17#
PhysicianId	1
PhysicianName	Doctor 1
ShiftId	1
ShiftName	Shift morning
MedicalSpecialtyId	1
MedicalSpecialtyName	Family Medicine
OfficeId	2
OfficeVisitLastNumber	4
Patient	

&officeVisitPatient

PatientId	2
Patient Name	patient 2
OfficeVisitPatientNumber	2

PatientId	3
Patient Name	patient 3
OfficeVisitPatientNumber	3

Output

Show: Build

Building file gx\_last\_transfer.zip...  
 Uploading 50 Kbytes  
 Deploying website  
 Success: Build With This Only

Properties | Toolbox | Responsive Sizes

Ln 25 Col 58 Ch 54 1/0/0/1/1/1 INS

Tenemos que pasarle al método el id de la línea para que pueda devolver una referencia a ese ítem de la colección. Será una referencia al ítem mismo, a su posición de memoria, no será una copia.

File Edit View Layout Insert Build Knowledge Manager Window Tools Test Help

CloudNET | Release | Android | 1.2-SNAPSHOT

KB Explorer | Patient x | Office x | Physician x | MedicalSpecialty x | OfficeVisit x | OfficeVisitTEST x | Navigation View x

Web Layout | Rules | **Events** | Conditions | Variables | Help | Documentation

'Update Office Visit'

```

7 | If not &patientId.IsEmpty()
8 |     &officeVisitPatient = new()
9 |     &officeVisitPatient.PatientId = &patientId
10 |     &officeVisit.Patient.Add(&officeVisitPatient)
11 | endif
12 | if &officeVisit.Insert()
13 |     Commit
14 | endif
15 | &messages = &officeVisit.GetMessages()
16 | -Endevent
17 |
18 |
19 | Event 'Update Office Visit'
20 |     &officeVisit.Load(&ScheduleDate, &PhysicianId, &ShiftId)
21 |     &officeVisit.OfficeId = &OfficeId
22 |     &officeVisit.OfficeVisitLastNumber +=1
23 |
24 |     //Update line for &patientId
25 |     &officeVisitPatient = &officeVisit.Patient.GetByKey(&patientId)
26 |     &officeVisitPatient.OfficeVisitPatientNumber = &officeVisit.OfficeVisitLastNumber
27 |
28 |     &officeVisitPatientNumber = &officeVisit.Patient.GetByKey(&patient2Id).OfficeVisitPatientNumber
29 |     //Delete line for &patient2Id
30 |
31 |     &officeVisitPatient = new()
32 |     &officeVisitPatient.PatientId = &patient3Id
33 |     &officeVisitPatient.OfficeVisitPatientNumber = &officeVisitPatientNumber
34 |     //Insert line for &patient3Id
35 |
36 |     If &officeVisit.Update()
37 |         commit
38 |     endif
39 |     &messages = &officeVisit.GetMessages()

```

OfficeVisitDate	#2022.11.17#
PhysicianId	1
PhysicianName	Doctor 1
ShiftId	1
ShiftName	Shift morning
MedicalSpecialtyId	1
MedicalSpecialtyName	Family Medicine
OfficeId	2
OfficeVisitLastNumber	4
Patient	

&officeVisitPatient

PatientId	2
Patient Name	Patient 2
OfficeVisitPatientNumber	4

PatientId	3
Patient Name	Patient 3
OfficeVisitPatientNumber	3

Output

Show: Build | X | Q | S

Building file gx\_last\_transfer.zip...  
 Uploading 50 Kbytes  
 Deploying website  
 Success: Build With This Only

Properties | Filter | Attribute: OfficeVisitLastNumber

Properties | Toolbox | Responsive Sizes

Ln 26 | Col 87 | Ch 83 | 1 / 0 / 1 / 1 / 1 | INS

Por ello alcanzará con modificar para esta variable el valor de este elemento, porque se estará modificando el ítem de la colección directamente. No hay que hacer ningún update. Con esto alcanza.

File Edit View Layout Insert Build Knowledge Manager Window Tools Test Help

CloudNET Android 1.2-SNAPSHOT

KB Explorer

GeneXus Client Common Configuration Notifications UI Deprecated DirectionsServices Analytics AnalyticsPurchase Clipboard DynamicCall GridState LocationInfo LocationPickerParam LocationProximityAler Log Maps Messages Runtime Server TrackingParameters Domains OAuth SD Server Social Social Domains Gxtest Customization Documentation

Patent x Office x Physician x MedicalSpecialty x OfficeVisit x OfficeVisitTEST x Navigation View x Properties

Web Layout | Rules | **Events** | Conditions | Variables | Help | Documentation

'Update Office Visit'

```

7 If not &patientId.IsEmpty()
8     &officeVisitPatient = new()
9     &officeVisitPatient.PatientId = &patientId
10    &officeVisit.Patient.Add(&officeVisitPatient)
11 endif
12 if &officeVisit.Insert()
13     Commit
14 endif
15 &messages = &officeVisit.GetMessages()
16 -Endevent
17
18
19 Event 'Update Office Visit'
20     &officeVisit.Load(&scheduleDate, &physicianId, &shiftId)
21     &officeVisit.Patient = &officeVisit.Patient
22     &officeVisit.OfficeVisitLastNumber += 1
23
24     //Update line for &patientId
25     &officeVisitPatient = &officeVisit.Patient.GetByKey(&patientId)
26     &officeVisitPatient.OfficeVisitPatientNumber = &officeVisit.OfficeVisitLastNumber
27
28     &officeVisitPatientNumber = &officeVisit.Patient.GetByKey(&patient2Id).OfficeVisitPatientNumber
29     //Delete line for &patient2Id
30
31     &officeVisitPatient = new()
32     &officeVisitPatient.PatientId = &patient3Id
33     &officeVisitPatient.OfficeVisitPatientNumber = &officeVisitPatientNumber
34     //Insert line for &patient3Id
35
36 If &officeVisit.Update()
37     commit
38 endif
39 &messages = &officeVisit.GetMessages()

```

OfficeVisitDate	#2022-11-17#
PhysicianId	1
PhysicianName	Doctor 1
ShiftId	1
ShiftName	Shift morning
MedicalSpecialtyId	1
MedicalSpecialtyName	Family Medicine
OfficeId	2
OfficeVisitLastNumber	4
Patient	---

PatientId	2
Patient Name	Patient 2
OfficeVisitPatientNumber	4

PatientId	3
Patient Name	Patient 3
OfficeVisitPatientNumber	3

PatientId	4
Patient Name	Patient 4
OfficeVisitPatientNumber	3

Output

Show: Build | X | Q | S

Building file gx\_last\_transfer.zip...  
 Uploading 50 Kbytes  
 Deploying website  
 Success: Build With This Only

Ln 27 Col 6 Ch 2 1/0/0/1/1/1 INS

Ahora querremos eliminar la línea correspondiente a este paciente, para insertar otra para cambiar el paciente.



File Edit View Layout Insert Build Knowledge Manager Window Tools Test Help

CloudNET Android 1.2-SNAPSHOT

KB Explorer

Patent x Office x Physician x MedicalSpecialty x OfficeVisit x OfficeVisitTEST x Navigation View x Properties

Web Layout | Rules **Events** | Conditions | Variables | Help | Documentation

'Update Office Visit'

```

7   If not &patientId.IsEmpty()
8       &officeVisitPatient = new()
9       &officeVisitPatient.PatientId = &patientId
10      &officeVisit.Patient.Add(&officeVisitPatient)
11   endif
12   if &officeVisit.Insert()
13       Commit
14   endif
15   &messages = &officeVisit.GetMessages()
16 -Endevent
17
18
19 =Event 'Update Office Visit'
20   &officeVisit.Load(&ScheduleDate, &PhysicianId, &ShiftId)
21   &officeVisit.OfficeId = &OfficeId
22   &officeVisit.OfficeVisitLastNumber +=1
23
24   //Update line for &patientId
25   &officeVisitPatient = &officeVisit.Patient.GetByKey(&patientId)
26   &officeVisitPatient.OfficeVisitPatientNumber = &officeVisit.OfficeVisitLastNumber
27
28   &officeVisitPatientNumber = &officeVisit.Patient.GetByKey(&patient2Id).OfficeVisitPatientNumber
29   //Delete line for &patient2Id
30   |
31
32   &officeVisitPatient = new()
33   &officeVisitPatient.PatientId = &patient3Id
34   &officeVisitPatient.OfficeVisitPatientNumber = &officeVisitPatientNumber
35   //Insert line for &patient3Id
36
37   If &officeVisit.Update()
38       commit
39   endif

```

OfficeVisitDate	#2022.11.17#
PhysicianId	1
PhysicianName	Doctor 1
ShiftId	1
ShiftName	Shift morning
MedicalSpecialtyId	1
MedicalSpecialtyName	Family Medicine
OfficeId	2
OfficeVisitLastNumber	4
Patient	—

PatientId	2
Patient Name	Patient 2
OfficeVisitPatientNumber	4

PatientId	3
Patient Name	Patient 3
OfficeVisitPatientNumber	3

Output

Show: Build | X | Q | S

Building file gx\_last\_transfer.zip...  
 Uploading 50 Kbytes  
 Deploying website  
 Success: Build With This Only

Properties | Toolbox | Responsive Sizes

Ln 30 Col 6 Ch 2 1 / 0 / 1 / 1 / 1 INS

Pero antes de eliminarla necesitamos recuperar cuál era el número que se le había asignado. Para ello, a una variable donde guardaremos ese valor le asignamos lo de la derecha. Aquí estamos recuperando una referencia al BC de las líneas que tiene este valor en el identificador. Y de ese BC, estamos pidiendo el Patient Number.

File Edit View Layout Insert Build Knowledge Manager Window Tools Test Help

CloudNET Release Android 1.2-SNAPSHOT

KB Explorer

Patent x Office x Physician x MedicalSpecialty x OfficeVisit x OfficeVisitTEST x Navigation View x Properties

Web Layout | Rules | **Events** | Conditions | Variables | Help | Documentation

Name or Pattern

GeneXus

- Client
- Common
- Configuration
- Notifications
- UI
- Deprecated
- DirectionsServices
- Analytics
- AnalyticsPurchase
- Clipboard
- Directions
- DynamicCall
- GridState
- LocationInfo
- LocationPickerParam
- LocationProximityAler
- Log
- Maps
- Messages
- Runtime
- Server
- TrackingParameters
- Domains
- OAuth
- SD
- Server
- Social
- Domains
- GXtest
- Customization
- Documentation

'Update Office Visit'

```

7 If not &patientId.IsEmpty()
8     &officeVisitPatient = new()
9     &officeVisitPatient.PatientId = &patientId
10    &officeVisit.Patient.Add(&officeVisitPatient)
11 endif
12 if &officeVisit.Insert()
13     Commit
14 endif
15 &messages = &officeVisit.GetMessages()
16 -Endevent
17
18
19 Event 'Update Office Visit'
20     &officeVisit.Load(&ScheduleDate, &PhysicianId, &ShiftId)
21     &officeVisit.PatientId = &OfficeId
22     &officeVisit.OfficeVisitLastNumber +=1
23
24     //Update line for &patientId
25     &officeVisitPatient = &officeVisit.Patient.GetByKey(&patientId)
26     &officeVisitPatient.OfficeVisitPatientNumber = &officeVisit.OfficeVisitLastNumber
27
28     &officeVisitPatientNumber = &officeVisit.Patient.GetByKey(&patient2Id).OfficeVisitPatientNumber
29     //Delete line for &patient2Id
30     &officeVisit.Patient.RemoveByKey(&patient2Id)
31
32     &officeVisitPatient = new()
33     &officeVisitPatient.PatientId = &patient3Id
34     &officeVisitPatient.OfficeVisitPatientNumber = &officeVisitPatientNumber
35     //Insert line for &patient3Id
36
37 If &officeVisit.Update()
38     commit
39 endif

```

OfficeVisitDate	#2022.11.17#
PhysicianId	1
PhysicianName	Doctor 1
ShiftId	1
ShiftName	Shift morning
MedicalSpecialtyId	1
MedicalSpecialtyName	Family Medicine
OfficeId	2
OfficeVisitLastNumber	4
Patient	---

PatientId	2
Patient Name	Patient 2
OfficeVisitPatientNumber	4

PatientId	3
Patient Name	Patient 3
OfficeVisitPatientNumber	3

Output

Show: Build | X | Q | S

Building file gx\_last\_transfer.zip...  
 Uploading 50 Kbytes  
 Deploying website  
 Success: Build With This Only

Properties | Toolbox | Responsive Sizes

Ln 30 Col 51 Ch 47 1/0/0/1/1/1 INS

Ahora sí, podemos borrar la línea. Para ello sobre la lista de pacientes de la consulta ejecutamos el método RemoveByKey, al que le pasamos el identificador de la línea que queremos eliminar, que es el id de este paciente, el de la segunda variable de la pantalla del web panel. Este método por ahora solamente eliminó el ítem de la colección en memoria. Aún no hemos hecho nada en la base de datos.

File Edit View Layout Insert Build Knowledge Manager Window Tools Test Help

CloudNET Release Android 1.2-SNAPSHOT

KB Explorer

GeneXus

- Client
- Common
- Configuration
- Notifications
- UI
- Deprecated
- DirectionsServices
- Analytics
- AnalyticsPurchase
- Clipboard
- Directions
- DynamicCall
- GridState
- LocationInfo
- LocationPickerParam
- LocationProximityAler
- Log
- Maps
- Messages
- Runtime
- Server
- TrackingParameters
- Domains
- OAuth
- SD
- Server
- Social
- Domains
- GXtest
- Customization
- Documentation

Web Layout | Rules | **Events** | Conditions | Variables | Help | Documentation

'Update Office Visit'

```

7   If not &patientId.IsEmpty()
8       &officeVisitPatient = new()
9       &officeVisitPatient.PatientId = &patientId
10      &officeVisit.Patient.Add(&officeVisitPatient)
11  endif
12  if &officeVisit.Insert()
13      Commit
14  endif
15  &messages = &officeVisit.GetMessages()
16  -Endevent
17
18
19  Event 'Update Office Visit'
20      &officeVisit.Load(&ScheduleDate, &PhysicianId, &ShiftId)
21      &officeVisit.OfficeId = &OfficeId
22      &officeVisit.OfficeVisitLastNumber +=1
23
24      //Update line for &patientId
25      &officeVisitPatient = &officeVisit.Patient.GetByKey(&patientId)
26      &officeVisitPatient.OfficeVisitPatientNumber = &officeVisit.OfficeVisitLastNumber
27
28      &officeVisitPatientNumber = &officeVisit.Patient.GetByKey(&patient2Id).OfficeVisitPatientNumber
29      //Delete line for &patient2Id
30      &officeVisit.Patient.RemoveByKey(&patient2Id)
31
32      &officeVisitPatient = new()
33      &officeVisitPatient.PatientId = &patient3Id
34      &officeVisitPatient.OfficeVisitPatientNumber = &officeVisitPatientNumber
35      //Insert line for &patient3Id
36      |
37
38  If &officeVisit.Update()
39      commit

```

&officeVisit

OfficeVisitDate	#2022-11-1#
PhysicianId	3
PhysicianName	Doctor 1
ShiftId	1
ShiftName	Shift morning
MedicalSpecialtyId	3
MedicalSpecialtyName	Family Medicine
OfficeId	2
OfficeVisitLastNumber	4
Patient	

&officeVisitPatient

PatientId	2
Patient Name	Patient 2
OfficeVisitPatientNumber	4

PatientId	4
Patient Name	
OfficeVisitPatientNumber	3

Output

Show: Build

Building file gx\_last\_transfer.zip...  
 Uploading 50 Kbytes  
 Deploying website  
 Success: Build With This Only

Properties | Toolbox | Responsive Sizes

Ln 36 Col 6 Ch 2 1/0/0/1/1/1 INS

Por último, debemos agregar un nuevo ítem a la colección, para el paciente sustituto, con el número dado al reemplazado.

File Edit View Layout Insert Build Knowledge Manager Window Tools Test Help

CloudNET Release Android 1.2-SNAPSHOT

KB Explorer

GeneXus

- Client
- Common
- Configuration
- Notifications
- UI
- Deprecated
- DirectionsServices
- Analytics
- AnalyticsPurchase
- Clipboard
- Directions
- DynamicCall
- GridState
- LocationInfo
- LocationPickerParam
- LocationProximityAler
- Log
- Maps
- Messages
- Runtime
- Server
- TrackingParameters
- Domains
- OAuth
- SD
- Server
- Social
- Domains
- GXtest
- Customization
- Documentation

Patent x Office x Physician x MedicalSpecialty x OfficeVisit x OfficeVisitTEST x Navigation View x Properties

Web Layout | Rules | **Events** | Conditions | Variables | Help | Documentation

'Update Office Visit'

```

7   If not &patientId.IsEmpty()
8       &officeVisitPatient = new()
9       &officeVisitPatient.PatientId = &patientId
10      &officeVisit.Patient.Add(&officeVisitPatient)
11  endif
12  if &officeVisit.Insert()
13      Commit
14  endif
15  &messages = &officeVisit.GetMessages()
16 -Endevent
17
18
19 Event 'Update Office Visit'
20     &officeVisit.Load(&ScheduleDate, &PhysicianId, &ShiftId)
21     &officeVisit.OfficeId = &OfficeId
22     &officeVisit.OfficeVisitLastNumber +=1
23
24     //Update line for &patientId
25     &officeVisitPatient = &officeVisit.Patient.GetByKey(&patientId)
26     &officeVisitPatient.OfficeVisitPatientNumber = &officeVisit.OfficeVisitLastNumber
27
28     &officeVisitPatientNumber = &officeVisit.Patient.GetByKey(&patient2Id).OfficeVisitPatientNumber
29     //Delete line for &patient2Id
30     &officeVisit.Patient.RemoveByKey(&patient2Id)
31
32     &officeVisitPatient = new()
33     &officeVisitPatient.PatientId = &patient3Id
34     &officeVisitPatient.OfficeVisitPatientNumber = &officeVisitPatientNumber
35     //Insert line for &patient3Id
36     &officeVisit.Patient.Add(&officeVisitPatient)
37
38     If &officeVisit.Update()
39         commit

```

&officeVisit

OfficeVisitDate	#2022-11-1#
PhysicianId	3
PhysicianName	Doctor 1
ShiftId	1
ShiftName	Shift morning
MedicalSpecialtyId	3
MedicalSpecialtyName	Family Medicine
OfficeId	2
OfficeVisitLastNumber	4
Patient	

&officeVisitPatient

PatientId	2
Patient Name	Patient 2
OfficeVisitPatientNumber	4

PatientId	4
Patient Name	
OfficeVisitPatientNumber	3

Output

Show: Build

Building file gx\_last\_transfer.zip...  
 Uploading 50 Kbytes  
 Deploying website  
 Success: Build With This Only

Properties | Toolbox | Responsive Sizes

Ln 32 Col 22 Ch 18 1/0/0/1/1/1 INS

Nos falta insertar este ítem a la colección. Lo hacemos con el método Add.

File Edit View Layout Insert Build Knowledge Manager Window Tools Test Help

CloudNET Release Android 1.2-SNAPSHOT

KB Explorer

GeneXus

- Client
- Common
- Configuration
- Notifications
- UI
- Deprecated
- DirectionsServices
- Analytics
- AnalyticsPurchase
- Clipboard
- Directions
- DynamicCall
- GridState
- LocationInfo
- LocationPickerParam
- LocationProximityAler
- Log
- Maps
- Messages
- Runtime
- Server
- TrackingParameters
- Domains
- OAuth
- SD
- Server
- Social
- Domains
- GXtest
- Customization
- Documentation

Patent x Office x Physician x MedicalSpecialty x OfficeVisit x OfficeVisitTEST x Navigation View x Properties

Web Layout | Rules | **Events** | Conditions | Variables | Help | Documentation

'Update Office Visit'

```

7   If not &patientId.IsEmpty()
8       &officeVisitPatient = new()
9       &officeVisitPatient.PatientId = &patientId
10      &officeVisit.Patient.Add(&officeVisitPatient)
11  endif
12  if &officeVisit.Insert()
13      Commit
14  endif
15  &messages = &officeVisit.GetMessages()
16  -Endevent
17
18
19  Event 'Update Office Visit'
20      &officeVisit.Load(&ScheduleDate, &PhysicianId, &ShiftId)
21      &officeVisit.OfficeId = &OfficeId
22      &officeVisit.OfficeVisitLastNumber +=1
23
24      //Update line for &patientId
25      &officeVisitPatient = &officeVisit.Patient.GetByKey(&patientId)
26      &officeVisitPatient.OfficeVisitPatientNumber = &officeVisit.OfficeVisitLastNumber
27
28      &officeVisitPatientNumber = &officeVisit.Patient.GetByKey(&patientId).OfficeVisitPatientNumber
29      //Delete line for &patient2Id
30      &officeVisit.Patient.RemoveByKey(&patient2Id)
31
32      &officeVisitPatient = new()
33      &officeVisitPatient.PatientId = &patient3Id
34      &officeVisitPatient.OfficeVisitPatientNumber = &officeVisitPatientNumber
35      //Insert line for &patient3Id
36      &officeVisit.Patient.Add(&officeVisitPatient)
37
38      If &officeVisit.Update()
39          commit

```

&officeVisit

OfficeVisitDate	#2022-11-1#
PhysicianId	3
PhysicianName	Doctor 1
ShiftId	1
ShiftName	Shift morning
MedicalSpecialtyId	3
MedicalSpecialtyName	Family Medicine
OfficeId	2
OfficeVisitLastNumber	4
Patient	

&officeVisitPatient

PatientId	2
Patient Name	Patient 2
OfficeVisitPatientNumber	4

PatientId	4
Patient Name	
OfficeVisitPatientNumber	3

Output

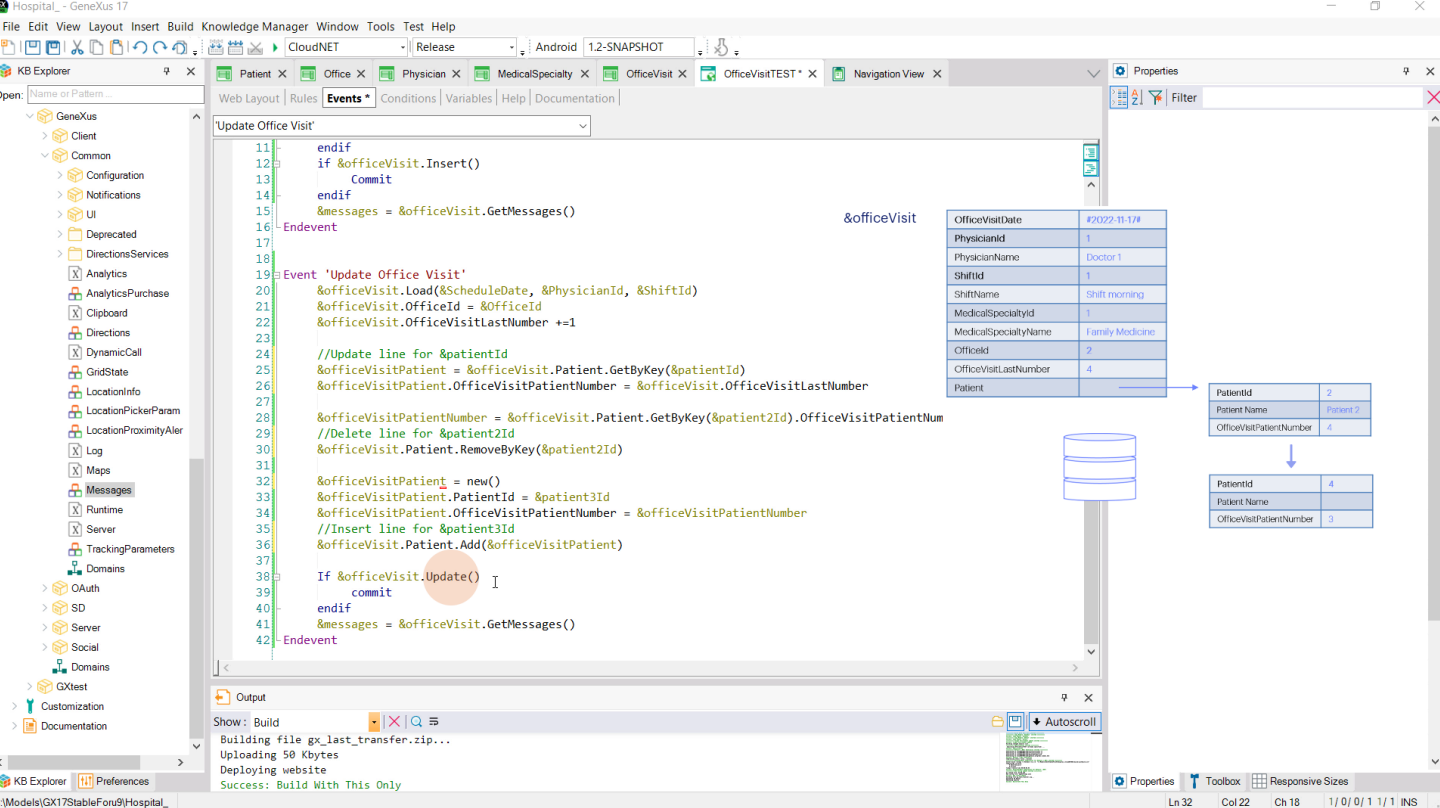
Show: Build | X | Q | Autoscroll

Building file gx\_last\_transfer.zip...  
 Uploading 50 Kbytes  
 Deploying website  
 Success: Build With This Only

Properties | Toolbox | Responsive Sizes

Ln 32 Col 22 Ch 18 1/0/0/1/1/1 INS

En este caso era imprescindible pedir nueva memoria para esta variable BC de la línea, porque ya la habíamos utilizado aquí, y si no pedíamos memoria nueva, entonces estaríamos insertando a la colección la misma posición de memoria que ya estaba, por lo que en lugar de 2 líneas: una para el patientId y otra para el patient3Id, tendríamos una sola, con estos valores que habrán sobrescrito a los primeros.



Llegados aquí ya hicimos sobre la variable BC que cargamos de la base de datos todas las modificaciones que queríamos. Nos falta ahora pedir lo que en la transacción pediríamos presionando el botón Confirm. Es decir, que se actualice la información en la base de datos, ejecutando todos los controles del Business component. Si el resultado del Update es exitoso, commiteamos. Y de cualquier modo recuperamos los mensajes producidos. Probemos esto.

11/17/22

1 Doctor 1

Shift morning

Family Medicine

3

3 UPDATE

DELETE

Shift Id: 1

Shift Name: Shift morning

Medical Specialty Id: 1

Medical Specialty Name: Family Medicine

Office Id: 2

Last Number: 3 4

**Patient**

Patient Id	Patient Name	Patient Number
2	Patient 2	2 4
3	Patient 3	3
4		3
0		0
0		0
0		0
0		0

Para esta consulta, queremos cambiar el consultorio por el 2, al paciente 2 darle el número siguiente al último y actualizar el último; eliminar el paciente 3 y agregar e 4 con su número.

# Hospital Backoffice



Recents Office Visit TEST

Schedule Date	11/17/22	<input type="button" value="29"/>	<input type="button" value="Insert Office Visit"/>
Physician Id	1		<input type="button" value="Update Office Visit"/>
Shift Id	1		
Office Id	2		
Patient Id	2		
patient2 Id	3		
patient3 Id	4		

Id	Type	Description
SuccessfullyUpdated	Warning	▼ Data has been successfully updated.

Entonces aquí elegimos estos valores, los de esa clave primaria... consultorio 2, paciente 2; 3 al que queremos cambiar por el 4. ¡Success!, como esperábamos.



GeneXus Developer Menu x Office Visit x Office Visit TEST x +

apps5.genexus.com/ld9df182c70350001219afeaceeb30e8e2/officevisittest.aspx

GeneXus DL Portal

# Hospital Backoffice

by GeneXus

Recents Office Visits — Office Visit — Office Visit TEST

Schedule Date: 11/17/22, 29

Physician Id: 1

Shift Id: 1

Office Id: 3

Patient Id: 2

patient2 Id: 4

patient3 Id: 1

Id

Insert Office Visit

Update Office Visit

Monday	Tuesday	Wednesday	Thursday	Friday
Physician: 1 Office: 2 Physician: 2 Office: 3 Patient: 1		Physician: 2 Office: 1	Physician: 1 Office: 2 Patient: 2 <del>Patient: 4</del>	
Physician: 1 Office: 2				
	Physician: 1 Office: 2			

Ahora hagamos fallar la actualización para ver que se está chequeando todo lo que esperaríamos que se chequeara.

Por ejemplo, vamos a intentar cambiar la oficina por la 3 (no habrá problema), al paciente 2 darle el siguiente número del último (no habrá problema) y eliminar al paciente 4 (no habrá problema), para insertar una nueva línea para el paciente 1, que sabemos que ya está agendado en una consulta de esta especialidad.

# Hospital Backoffice



Recents Office Visits — Office Visit — Office Visit TEST

Schedule Date: 11/17/22

Physician Id:

Shift Id:

Office Id:

Patient Id:

patient2 Id:

patient3 Id:

Insert Office Visit

Update Office Visit

Checks	Error Id	Error Description
FK: PhysicianId	ForeignKeyNotFound	No matching 'Physician'
<code>error("Physician " + PhysicianId.ToString() + " already has 2 shifts assigned for the date: " + OfficeVisitDate.ToString(), PhysicianNotAvailable) if &amp;shifts &gt;= 2 and Insert;</code>	PhysicianNotAvailable	Physician... already has 2 shifts...
FK: ShiftId	ForeignKeyNotFound	No matching 'Shift'
FK: OfficeId	ForeignKeyNotFound	No matching 'Office'
CK: {OfficeVisitDate, ShiftId, OfficeId}		Office Visit Date, Shift Id, Office Id already exists
PK: {OfficeVisitDate, PhysicianId, ShiftId}	DuplicatePrimaryKey	Record already exists
FK: PatientId	ForeignKeyNotFound	No matching 'Patient'
<code>error("There is a pending office visit scheduled for this patient for the same specialty", PatientAlreadyScheduledForMedicalSpecialty) if not &amp;IsOK and Insert Level PatientId;</code>	PatientAlreadyScheduledForMedicalSpecialty	There is a pending office visit scheduled...
PK: {OfficeVisitDate, PhysicianId, ShiftId, PatientId}	DuplicatePrimaryKey	Record already exists

Como la línea se estará insertando, se deberá ejecutar la regla de error de la transacción que impide la actualización de la base de datos. Probemos...

# Hospital Backoffice



Recents Office Visits — Office Visit — Office Visit TEST

Schedule Date	11/17/22	<input type="button" value="29"/>	<input type="button" value="Insert Office Visit"/>
Physician Id	1		<input type="button" value="Update Office Visit"/>
Shift Id	1		
Office Id	3		
Patient Id	2		
patient2 Id	4		
patient3 Id	1		

Id	Type	Description
PatientAlreadyScheduledForMedicalSpecialty	Error	There is a pending office visit scheduled for this patient for the same specialt

Efectivamente se disparó. Y la consulta quedó tal como estaba, sin cambios. No se hizo ninguna actualización en la base de datos.

# Hospital Backoffice

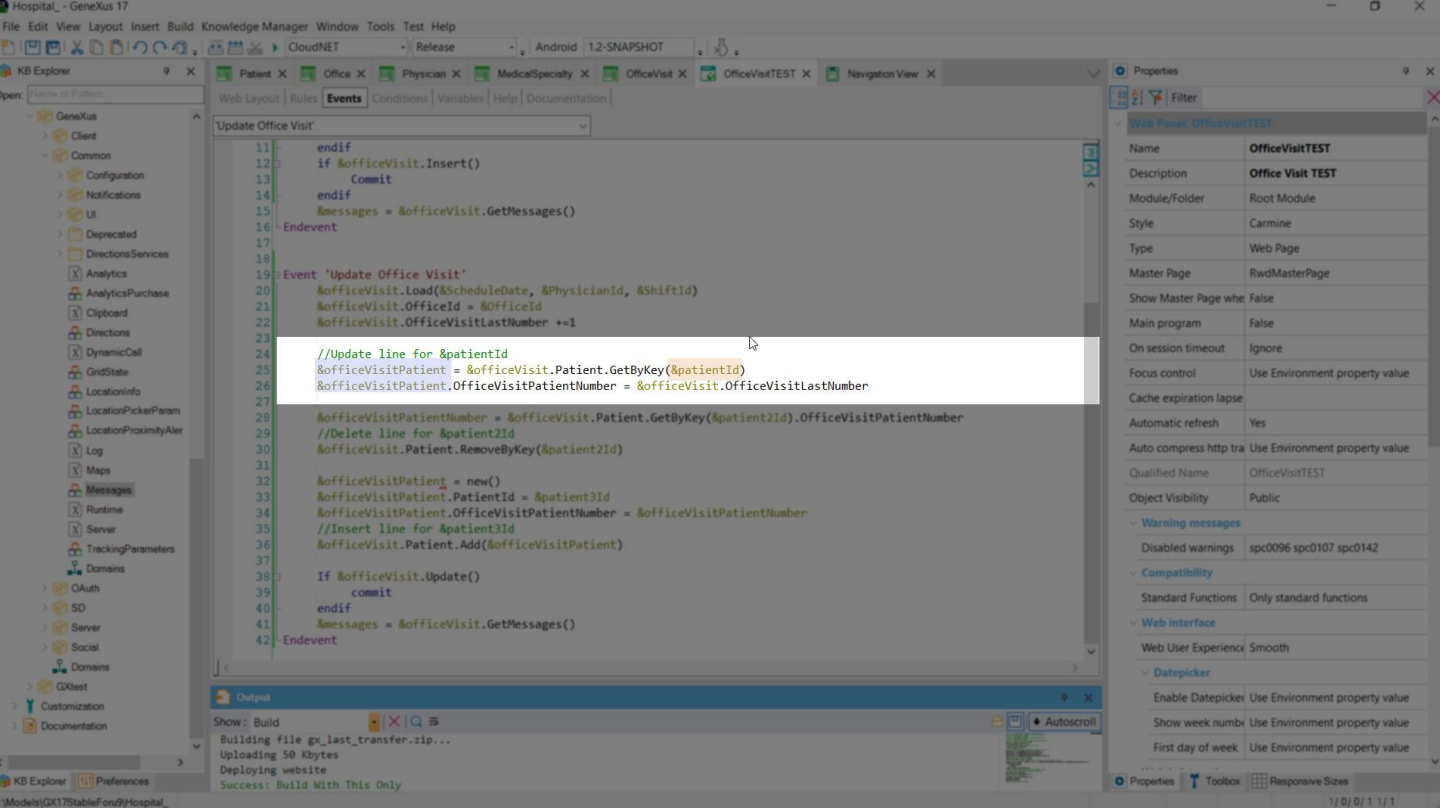


Recents Office Visits — Office Visit — Office Visit TEST

Schedule Date	11/17/22	<input type="text" value="29"/>	<input type="button" value="Insert Office Visit"/>
Physician Id	<input type="text" value="1"/>		<input type="button" value="Update Office Visit"/>
Shift Id	<input type="text" value="1"/>		
Office Id	<input type="text" value="3"/>		
Patient Id	<input type="text" value="1"/>		
patient2 Id	<input type="text" value="4"/>		
patient3 Id	<input type="text" value="3"/>		

Id	Type	Description
SuccessfullyUpdated	Warning	Data has been successfully updated.

Si ahora elegimos darle el último número a un paciente que no está en la consulta (por ejemplo el 1, que no está...y al 4 lo cambiamos por el 3 que no debería dar problemas)... es decir, cambiamos el consultorio por el 3, elegimos el paciente 1 para cambiarle el número, y cambiamos el 4 por el 3. ¿Por qué nos dejó actualizar con éxito si el paciente 1 no existe en la consulta?



Vamos al código. El problema es que el GetByKey no encontró un ítem para el paciente 1 en la colección de líneas, por lo que esta variable no podrá actualizar la línea porque no estará apuntando a ningún lugar. Las otras operaciones no tuvieron inconveniente.

The screenshot shows the GeneXus IDE interface. The main window displays the code for an event named 'Update Office Visit'. The code includes logic for loading an office visit, updating patient information, and adding a new patient. A red box highlights a section of code where a new patient is added, with a red 'X' mark next to it, indicating an error or a point of interest. The output window at the bottom shows the build process, including file transfer and website deployment. The properties panel on the right shows the configuration for the 'OfficeVisitTEST' web page, including its name, description, and various settings.

```
11  endif
12  if &officeVisit.Insert()
13      Commit
14  endif
15  &messages = &officeVisit.GetMessages()
16  -Endevent
17
18
19  Event 'Update Office Visit'
20      &officeVisit.Load(&ScheduleDate, &PhysicianId, &ShiftId)
21      &officeVisit.OfficeId = &OfficeId
22      &officeVisit.OfficeVisitLastNumber +=1
23
24      //Update line for &patientId
25      &officeVisitPatient = &officeVisit.Patient.GetByKey(&patientId)
26      &officeVisitPatient.OfficeVisitPatientNumber = &officeVisit.OfficeVisitLastNumber
27
28      &officeVisitPatientNumber = &officeVisit.Patient.GetByKey(&patient2Id).OfficeVisitPatientNumber
29      //Delete line for &patient2Id
30      &officeVisit.Patient.RemoveByKey(&patient2Id)
31
32      &officeVisitPatient = new()
33      &officeVisitPatient.PatientId = &patient3Id
34      &officeVisitPatient.OfficeVisitPatientNumber = &officeVisitPatientNumber
35      //Insert line for &patient3Id
36      &officeVisit.Patient.Add(&officeVisitPatient)
37
38      If &officeVisit.Update()
39          commit
40      endif
41      &messages = &officeVisit.GetMessages()
42  -Endevent
```

Output

```
Show: Build
Building file gx_last_transfer.zip...
Uploading 50 Kbytes
Deploying website
Success: Build With This Only
```

Name	OfficeVisitTEST
Description	Office Visit TEST
Module/Folder	Root Module
Style	Carmine
Type	Web Page
Master Page	RwdMasterPage
Show Master Page when	False
Main program	False
On session timeout	Ignore
Focus control	Use Environment property value
Cache expiration lapse	
Automatic refresh	Yes
Auto compress http tra	Use Environment property value
Qualified Name	OfficeVisitTEST
Object Visibility	Public
Warning messages	
Disabled warnings	spc0096 spc0107 spc0142
Compatibility	
Standard Functions	Only standard functions
Web interface	
Web User Experience	Smooth
Datepicker	
Enable Datepicker	Use Environment property value
Show week num	Use Environment property value
First day of week	Use Environment property value

Si ahora, en cambio, para la actualización del primer paciente damos el de una línea existente, para el que queremos eliminar también, pero nos equivocamos con el nuevo, y elegimos uno inexistente...

GeneXus Developer Menu | Office Visit | Office Visit TEST | app55.genexus.com/ld9df182c70350001219afeaceeb30e8e2/officevisit.aspx?Mode=UPD&OfficeVisitDate=20221117&PhysicianId=1&ShiftId=1

GeneXus | DL Portal

Shift Name: Shift morning

Medical Specialty Id: 1

Medical Specialty Name: Family Medicine

Office Id:

Last Number:

**Patient**

Patient Id	Patient Name	Patient Number
× 2 <input type="button" value="↑"/>	Patient 2	<input type="text" value="4"/>
× 3 <input type="button" value="↑"/>	Patient 3	<input type="text" value="3"/>
<input type="text" value="0"/> <input type="button" value="↑"/>		<input type="text" value="0"/>
<input type="text" value="0"/> <input type="button" value="↑"/>		<input type="text" value="0"/>
<input type="text" value="0"/> <input type="button" value="↑"/>		<input type="text" value="0"/>
<input type="text" value="0"/> <input type="button" value="↑"/>		<input type="text" value="0"/>
<input type="text" value="0"/> <input type="button" value="↑"/>		<input type="text" value="0"/>

[New row]

OfficeVisitDate	#2022-11-17#
PhysicianId	1
PhysicianName	Doctor 1
ShiftId	1
ShiftName	Shift morning
MedicalSpecialtyId	1
MedicalSpecialtyName	Family Medicine
OfficeId	3
OfficeVisitLastNumber	6
Patient	

↓

<del>PatientId</del>	<del>2</del>
<del>Patient Name</del>	<del>Patient 2</del>
<del>PatientOfficeVisitNumber</del>	<del>4</del>

↓

PatientId	3
Patient Name	Patient 3
PatientOfficeVisitNumber	6

↓

PatientId	25
Patient Name	
PatientOfficeVisitNumber	4

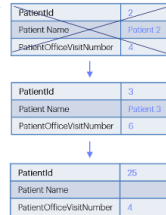
Así, por ejemplo, pedimos dar el último número al paciente 3, eliminar el 2 y sustituirlo por el 25...

# Hospital Backoffice

Recents Office Visits — Office Visit — Office Visit TEST

Schedule Date	11/17/22	29	Insert Office Visit
Physician Id	1		Update Office Visit
Shift Id	1		
Office Id	2		
Patient Id	3		
patient2 Id	2		
patient3 Id	25		

OfficeVisitDate	#2022 11 17#
PhysicianId	1
PhysicianName	Doctor 1
ShiftId	1
ShiftName	Shift morning
MedicalSpecialtyId	1
MedicalSpecialtyName	Family Medicine
OfficeId	2
OfficeVisitLastNumber	6
Patient	



Id	Type	Description
ForeignKeyNotFound	Error	No matching 'Patient'.

...cambiando el consultorio por el 2 por ejemplo... bien, nos arroja el error de integridad referencial del paciente.



# Hospital Backoffice

Recents Office Visits — Office Visit — Office Visit TEST

Schedule Date: 11/17/22 29

Physician Id: 1

Shift Id: 1

Office Id: 2

Patient Id: 3 update

patient2 Id: 25 delete

patient3 Id: 4 insert

Insert Office Visit

Update Office Visit

Id	Type	Description
SuccessfullyUpdated	Warning	Data has been successfully updated.

OfficeVisitDate	#2022-11-17#
PhysicianId	1
PhysicianName	Doctor 1
ShiftId	1
ShiftName	Shift morning
MedicalSpecialtyId	1
MedicalSpecialtyName	Family Medicine
OfficeId	2
OfficeVisitLastNumber	6
Patient	

↓

PatientId	2
Patient Name	Patient 2
PatientOfficeVisitNumber	4

↓

update

PatientId	3
Patient Name	Patient 3
PatientOfficeVisitNumber	6

↓

insert

PatientId	4
Patient Name	
PatientOfficeVisitNumber	0

¿Y si nos equivocamos al revés, con el que queremos eliminar? ¡Nos permitió actualizar! ¿Por qué? Observemos qué hizo.

GeneXus Developer Menu x Office Visit x Office Visit TEST x | +

apps5.genexus.com/ld9df182c70350001219afeaceeb30e8e2/officevisit.aspx?Mode=UPD&OfficeVisitDate=20221117&PhysicianId=1&ShiftId=1

GeneXus DL Portal

Shift Name	Shift morning
Medical Specialty Id	1
Medical Specialty Name	Family Medicine
Office Id	<input type="text" value="2"/> ↑
Last Number	<input type="text" value="6"/>

**Patient**

Patient Id	Patient Name	Patient Number
× 2 ↑	Patient 2	<input type="text" value="4"/>
× 3 ↑	Patient 3	<input type="text" value="6"/>
× 4 ↑	Patient 4	<input type="text" value="0"/>
<input type="text" value="0"/> ↑		<input type="text" value="0"/>
<input type="text" value="0"/> ↑		<input type="text" value="0"/>
<input type="text" value="0"/> ↑		<input type="text" value="0"/>
<input type="text" value="0"/> ↑		<input type="text" value="0"/>
<input type="text" value="0"/> ↑		<input type="text" value="0"/>

[New row]

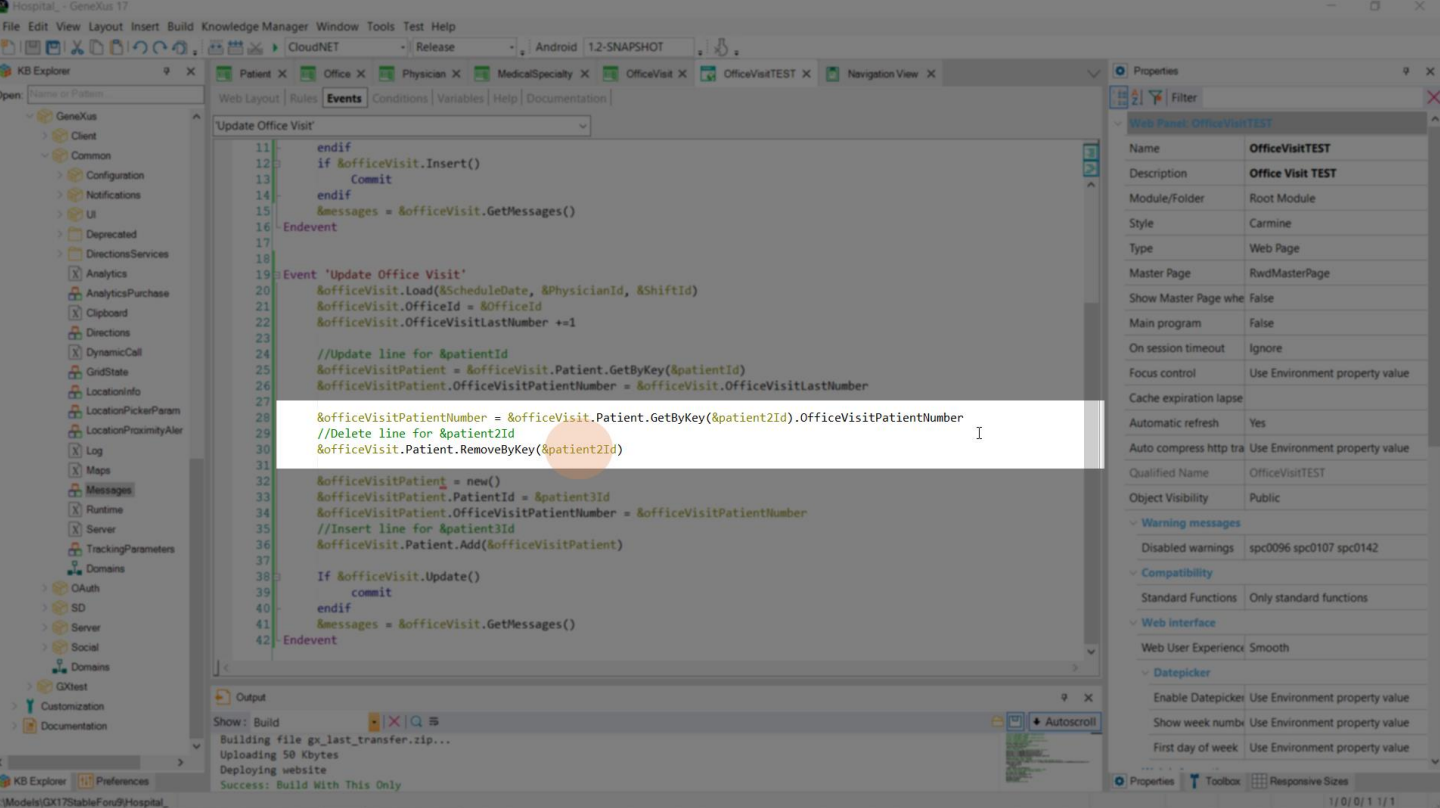
OfficeVisitDate	#2022-11-17#
PhysicianId	1
PhysicianName	Doctor 1
ShiftId	1
ShiftName	Shift morning
MedicalSpecialtyId	1
MedicalSpecialtyName	Family Medicine
OfficeId	2
OfficeVisitLastNumber	6
Patient	

PatientId	2
Patient Name	Patient 2
PatientOfficeVisitNumber	4

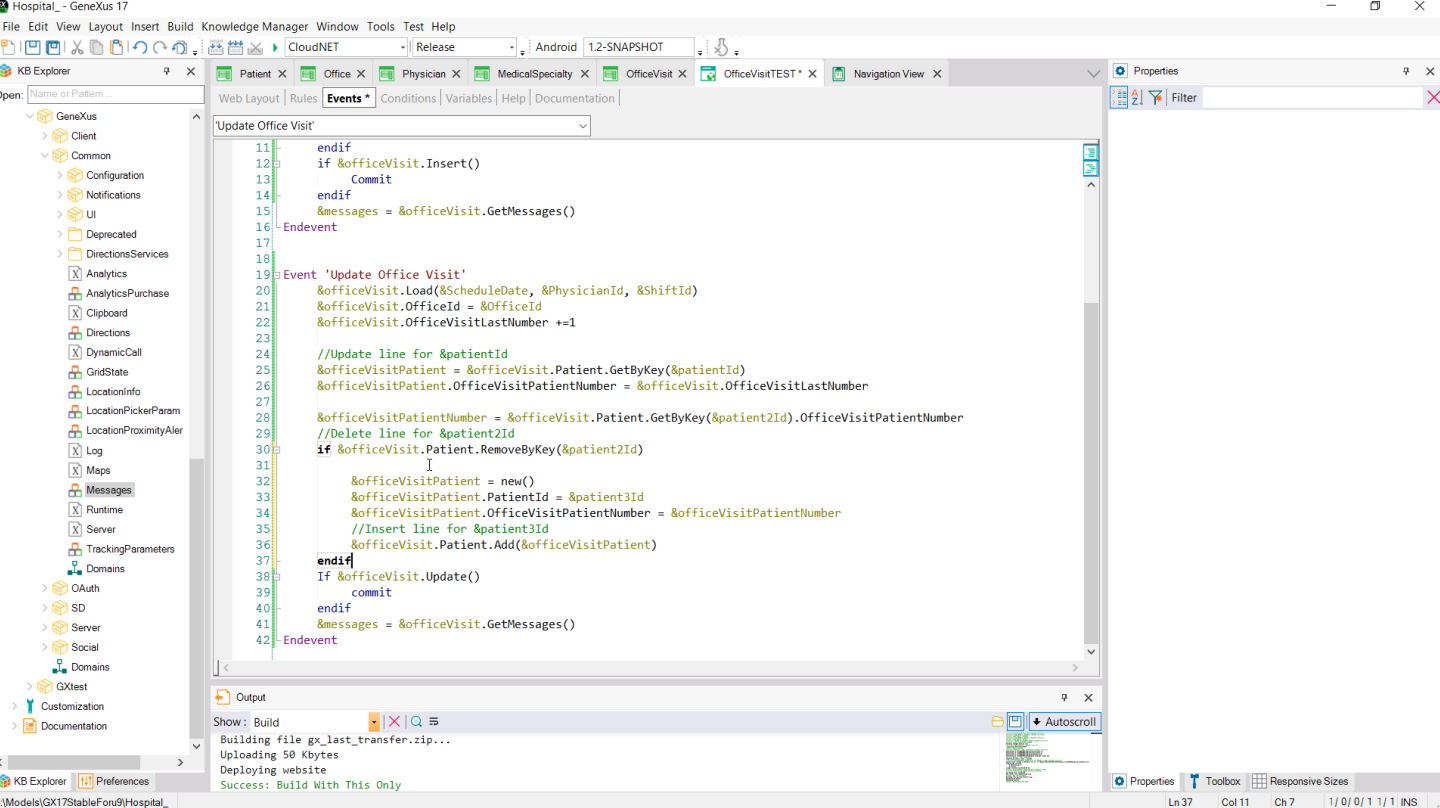
PatientId	3
Patient Name	Patient 3
PatientOfficeVisitNumber	6

PatientId	4
Patient Name	
PatientOfficeVisitNumber	0

Aumentó el Last Number en 1. Bien. Se lo asignó al paciente 3, bien. Pero dejó el paciente 2 como estaba, y agregó el 4, con 0 para el número. Veamos en GeneXus...



Esto nos dará 0, porque no encontrará el ítem de clave 25 en la colección. Y esta operación de Remove no eliminará la línea de esta clave, porque no existe. Por lo que el new se realizará con éxito pero con valor 0.



Podríamos mejorar un poco las cosas, preguntando por el resultado del remove. Y solo si fue exitoso agregar la línea.

GeneXus Developer Menu | Office Visits | Office Visit TEST | +

apps5.genexus.com/ld9df182c70350001219afeaceeb30e8e2/wwofficevisit.aspx

GeneXus | DL Portal

# Hospital Backoffice

by GeneXus

Recents | Office Visit — Office Visits

Office Visits

NAME

+ INSERT

HIDE FILTERS

MEDICAL SPECIALTY NAME

OFFICE VISIT DATE

SHIFT NAME

Visit Date	Physician Id	Physician Name	Shift Name	Medical Specialt...	Office Id	Last Number		
11/14/22	1	Doctor 1	Shift morning	Family Medicine	2	0	UPDATE	DELETE
11/14/22	1	Doctor 1	Shift intermediate	Family Medicine	2	0	UPDATE	DELETE
11/14/22	2	Doctor 2	Shift morning	Family Medicine	3	1	UPDATE	DELETE
11/15/22	1	Doctor 1	Shift evening	Family Medicine	2	0	UPDATE	DELETE
11/16/22	2	Doctor 2	Shift morning	Family Medicine	1	0	UPDATE	DELETE
11/17/22	1	Doctor 1	Shift morning	Family Medicine	2	6	UPDATE	DELETE
11/17/22	2	Doctor 2	Shift morning	Family Medicine	3	0	UPDATE	DELETE
11/21/22	1	Doctor 1	Shift morning	Family Medicine	1	1	UPDATE	DELETE

Últimos apuntes: aquí tenemos la consulta del jueves 17, para el médico 1 en el turno 1 en el consultorio 2. Agreguemos una consulta con otro médico para misma fecha y turno pero en el consultorio 3. Si ahora queremos cambiar la primera consulta, la del médico 1, para que sea en el consultorio 3, que ahora está ocupado...

# Hospital Backoffice



Recents Office Visits — Office Visit — Office Visit TEST

Schedule Date	11/17/22	<input type="button" value="29"/>	<input type="button" value="Insert Office Visit"/>
Physician Id	1		<input type="button" value="Update Office Visit"/>
Shift Id	1		
Office Id	3I		
Patient Id	3		
patient2 Id	25		
patient3 Id	4		

Id	Type	Description
	Error	Office Visit Date,Shift Id,Office Id already exists

...fallará la actualización por clave candidata duplicada.

Schedule Date	11/17/22	<input type="button" value="29"/>	<input type="button" value="Insert Office Visit"/>
Physician Id	1		<input type="button" value="Update Office Visit"/>
Shift Id	1		
Office Id	45		
Patient Id	3		
patient2 Id	25		
patient3 Id	4		

Id	Type	Description
ForeignKeyNotFound	Error	No matching 'Office'.

Y si quisiéramos cambiar el consultorio por uno inexistente fallará el control de integridad referencial.

# Hospital Backoffice



Recents Office Visit TESTC...

Schedule Date: 11/27/23

Physician Id: 1

Shift Id: 1

Office Id: 3

Patient Id: 2

patient2 Id: 3

patient3 Id: 4

Id	Type	Description
PrimaryKeyNotFound	Error	Data with the specified key could not be found.

Y por supuesto, si no existe un registro con esta clave primaria, coloquemos los valores que coloquemos en estas variables, al intentar actualizar nos arrojará este error, porque el Load no fue exitoso.



Office Visit

apps5.genexus.com/ld9df182c70350001219afeaceeb30e8e2/officev... DL Portal

### Office Visit

Visit Date	11/17/22
Physician Id	1
Physician Name	Doctor 1
Shift Id	1
Shift Name	Shift morning
Medical Specialty Id	1
Medical Specialty Name	Family Medicine
Office Id	<input type="text" value=""/>
Last Number	<input type="text" value="6"/>

&officeVisit .Load(#2022-11-17#, 1, 1)

OfficeVisitDate	#2022-11-17#
PhysicianId	1
PhysicianName	Doctor 1
ShiftId	1
ShiftName	Shift morning
MedicalSpecialtyId	1
MedicalSpecialtyName	Family Medicine
Officeld	2
OfficeVisitLastNumber	6
Patient	

PatientId	2
Patient Name	Patient 2
PatientOfficeVisitNumber	4

PatientId	3
Patient Name	Patient 3
PatientOfficeVisitNumber	6

PatientId	4
Patient Name	Patient 4
PatientOfficeVisitNumber	0

&officeVisit.Officeld

En resumen: para actualizar a través de la transacción se le pasa el valor de la clave primaria y allí se traen de la base de datos a la pantalla los valores de los atributos del cabezal y de las líneas.

Para hacerlo a través del BC se aplica el método Load para cargar los elementos de la estructura del BC. Por lo que los atributos en pantalla y las líneas, son como los elementos del BC y los ítems de la colección.

El usuario puede modificar aquellos atributos del cabezal modificables, y lo mismo hace el desarrollador con el BC a través de la variable, accediendo al elemento: variable BC, punto, nombre del elemento (por ejemplo Officeld).

En la transacción, el usuario puede modificar algún dato de una línea directamente posicionándose sobre esa línea y editando el campo que le interese.

Para hacer eso con el BC hay que acceder primero por código a la línea, es decir, al ítem de la colección. Para ello se cuenta con el método GetByKey que solo se puede aplicar al elemento del BC que es una colección de líneas. Aquí es &officeVisit punto Patient. A diferencia de la transacción donde contamos visualmente con la línea y sus valores, aquí tenemos que recuperarla, a partir de su identificador. Si existe un ítem con ese id en la colección, entonces recuperaremos perfectamente la posición de memoria del ítem (y nos convendrá, seguramente, asignársela a una variable del tipo de datos el BC de la línea, así trabajamos más cómodos). Entonces allí modificamos los elementos del ítem/línea que nos interese. Aquí podría ser el PatientOfficeVisitNumber.

¿Pero qué pasa si el id que especificamos para el GetByKey no existe? La

variable `&officeVisitPatient` no estará apuntando a la posición de la línea. Si no se había utilizado antes o se le había asignado un `new`, entonces estará vacía, y el cambio que se le haga al `PatientOfficeVisitNumber` no repercutirá en nada en el BC original.

En la transacción si queremos eliminar una línea, la identificamos y la marcamos así. En el BC tenemos el método `RemoveByKey` de las colección de ítems. Otra vez, si le pasamos un `id` de ítem que no se encuentra en la colección no podrá hacer nada sobre esa línea que queríamos borrar. Quedará todo como estaba. Este método tiene la ventaja de devolver `True` o `False` en base a si encuentra o no el ítem.

Y en la transacción si queremos agregar una nueva línea, lo hacemos directamente utilizando una de las líneas vacías que nos provee el grid. En el BC tendremos que crear el espacio de memoria para el ítem, con una variable del tipo de datos de las líneas, y asignándole el `new`, completar sus elementos, y utilizar el método `Add` de la colección de ítems. Con eso agregamos este espacio de memoria a la colección.

Hasta allí, ya sea manipulando la transacción o manipulando la variable BC hemos trabajado en memoria. Para que todo esto impacte en la base de datos, hay que dar la orden. En la transacción se hace con el botón `Confirm` (que como sabe que está en modo `Update`, intentará actualizar). En el BC lo hacemos con el método `Update` (también podríamos utilizar el método `Save`, pero no lo vamos a ver aquí). Como consecuencia de ello se ejecutarán las reglas, eventos y chequeos de la transacción que correspondan. Es decir, en el BC todos los que no dependan de la interfaz (y no estén calificados para ejecutarse solo en web, a través de los atributos de entorno que habíamos visto en el video anterior).

En definitiva, al trabajar con el BC tenemos que tener más cuidado en el sentido de que no vemos las líneas que existen y en esa programación podemos cometer errores. Luego veremos una forma más sencilla de insertar o modificar líneas (aunque no de eliminar) que es manipulándolas a través de un `Data Provider`.

Office Visit

Medical Specialty Id 1

Medical Specialty Name Family Medicine

Office Id

Last Number

**Patient**

Patient Id	Patient Name	Patient Number
2	Patient 2	4
3	Patient 3	<input type="text" value="3"/>
4	Patient 4	0
0		0
0		0
0		0
0		0
0		0

(New row)

&officeVisit

OfficeVisitDate	#2022-11-17#
PhysicianId	1
PhysicianName	Doctor 1
ShiftId	1
ShiftName	Shift morning
MedicalSpecialtyId	1
MedicalSpecialtyName	Family Medicine
OfficeId	2
OfficeVisitLastNumber	6
Patient	

PatientId	2
Patient Name	Patient 2
PatientOfficeVisitNumber	4

&officeVisitPatient =

&officeVisit.Patient .getByKey(3)

PatientId	3
Patient Name	Patient 3
PatientOfficeVisitNumber	6

PatientId	4
Patient Name	Patient 4
PatientOfficeVisitNumber	0

En la transacción, el usuario puede modificar algún dato de una línea directamente posicionándose sobre esa línea y editando el campo que le interese.

Para hacer eso con el BC hay que acceder primero por código a la línea, es decir, al ítem de la colección. Para ello se cuenta con el método `GetByKey` que solo se puede aplicar al elemento del BC que es una colección de líneas. Aquí es `&officeVisit` punto `Patient`. A diferencia de la transacción donde contamos visualmente con la línea y sus valores, aquí tenemos que recuperarla, a partir de su identificador. Si existe un ítem con ese id en la colección, entonces recuperaremos perfectamente la posición de memoria del ítem (y nos convendrá, seguramente, asignársela a una variable del tipo de datos el BC de la línea, así trabajamos más cómodos). Entonces allí modificamos los elementos del ítem/línea que nos interese. Aquí podría ser el `PatientOfficeVisitNumber`.

¿Pero qué pasa si el id que especificamos para el `GetByKey` no existe? La variable `&officeVisitPatient` no estará apuntando a la posición de la línea. Si no se había utilizado antes o se le había asignado un `new`, entonces estará vacía, y el cambio que se le haga al `PatientOfficeVisitNumber` no repercutirá en nada en el BC original.

Office Visit

apps5.genexus.com/ld9df182c70350001219afeaceeb30e8e2/officev... | DL Portal

Medical Specialty Id: 1

Medical Specialty Name: Family Medicine

Office Id: 2

Last Number: 6

**Patient**

Patient Id	Patient Name	Patient Number
2	Patient 2	4
3	Patient 3	6
4	Patient 4	0
0		0
0		0
0		0
0		0
0		0

[New row]

&amp;officeVisit

OfficeVisitDate	#2022-11-17#
PhysicianId	1
PhysicianName	Doctor 1
ShiftId	1
ShiftName	Shift morning
MedicalSpecialtyId	1
MedicalSpecialtyName	Family Medicine
OfficeId	2
OfficeVisitLastNumber	6
Patient	

PatientId	2
Patient Name	Patient 2
PatientOfficeVisitNumber	4

PatientId	3
Patient Name	Patient 3
PatientOfficeVisitNumber	6

<del>PatientId</del>	<del>4</del>
<del>Patient Name</del>	<del>Patient 4</del>
<del>PatientOfficeVisitNumber</del>	<del>0</del>

&amp;officeVisit.Patient.RemoveByKey(4)

En la transacción si queremos eliminar una línea, la identificamos y la marcamos así. En el BC tenemos el método RemoveByKey de las colección de ítems. Otra vez, si le pasamos un id de ítem que no se encuentra en la colección no podrá hacer nada sobre esa línea que queríamos borrar. Quedará todo como estaba. Este método tiene la ventaja de devolver True o False en base a si encuentra o no el ítem.

Office Visit

Medical Specialty Name: Family Medicine

Office Id: 2

Last Number: 6

**Patient**

Patient Id	Patient Name	Patient Number
2	Patient 2	4
3	Patient 3	6
4	Patient 4	0
0		0
0		0
0		0
0		0

[New row]

CONFIRM CANCEL

&amp;officeVisit

OfficeVisitDate	#2022-11-17#
PhysicianId	1
PhysicianName	Doctor 1
ShiftId	1
ShiftName	Shift morning
MedicalSpecialtyId	1
MedicalSpecialtyName	Family Medicine
OfficeId	2
OfficeVisitLastNumber	6
Patient	

&amp;officeVisit.Patient.Add(&amp;officeVisitPatient)

PatientId	2
Patient Name	Patient 2
PatientOfficeVisitNumber	4

PatientId	3
Patient Name	Patient 3
PatientOfficeVisitNumber	6

&amp;officeVisitPatient

PatientId	
Patient Name	
PatientOfficeVisitNumber	

PatientId	4
Patient Name	Patient 4
PatientOfficeVisitNumber	0

Y en la transacción si queremos agregar una nueva línea, lo hacemos directamente utilizando una de las líneas vacías que nos provee el grid. En el BC tendremos que crear el espacio de memoria para el ítem, con una variable del tipo de datos de las líneas, y asignándole el new, completar sus elementos, y utilizar el método Add de la colección de ítems. Con eso agregamos este espacio de memoria a la colección.

Office Visit

Medical Specialty Name: Family Medicine

Office Id: 2

Last Number: 6

**Patient**

Patient Id	Patient Name	Patient Number
2	Patient 2	4
3	Patient 3	6
4	Patient 4	0
0		0
0		0
0		0
0		0

[New row]

Confirm

CANCEL

&amp;officeVisit

OfficeVisitDate	#2022-11-17#
PhysicianId	1
PhysicianName	Doctor 1
ShiftId	1
ShiftName	Shift morning
MedicalSpecialtyId	1
MedicalSpecialtyName	Family Medicine
OfficeId	2
OfficeVisitLastNumber	6
Patient	

PatientId	2
Patient Name	Patient 2
PatientOfficeVisitNumber	4

PatientId	3
Patient Name	Patient 3
PatientOfficeVisitNumber	6

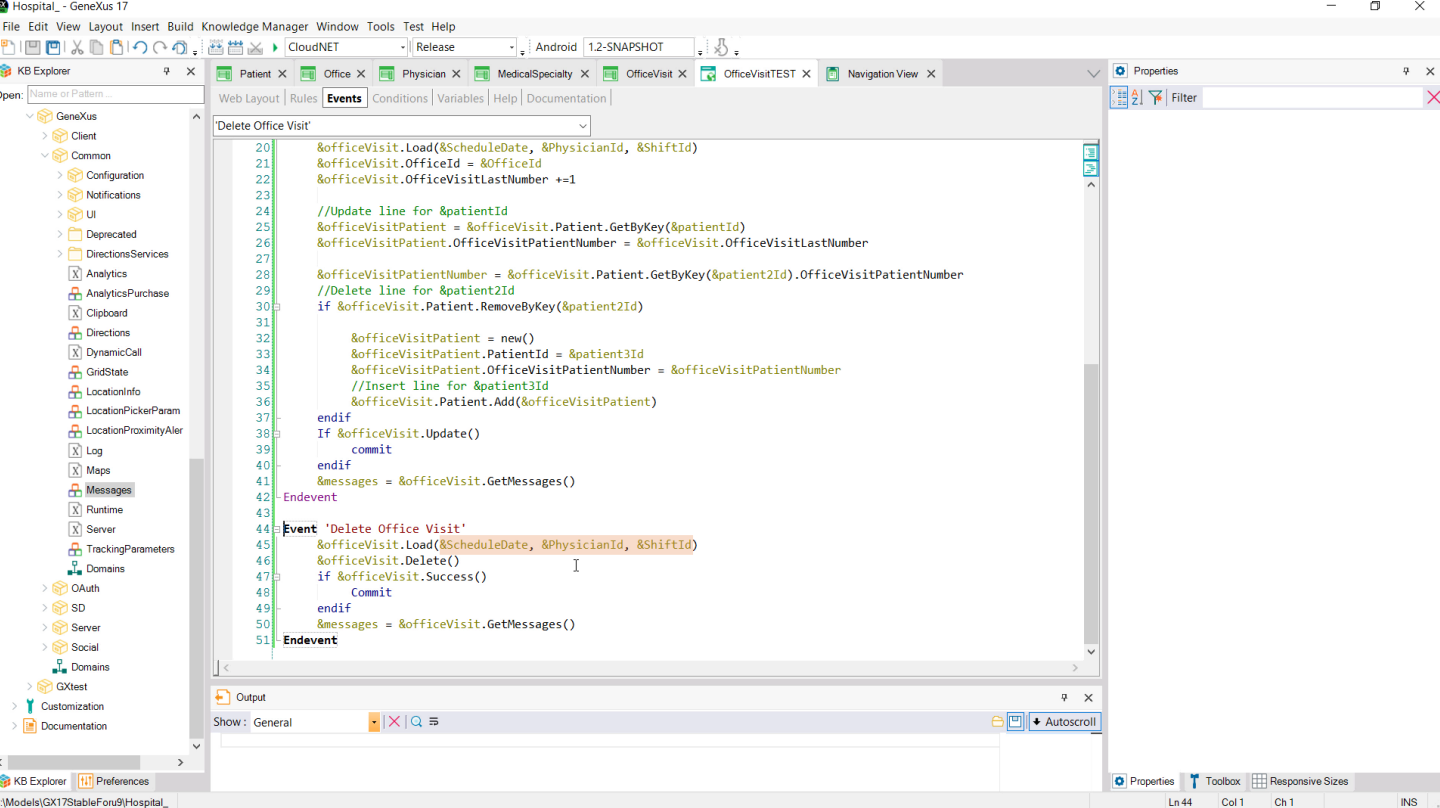
PatientId	4
Patient Name	Patient 4
PatientOfficeVisitNumber	0

&amp;officeVisit.Update()

PatientId	
Patient Name	
PatientOfficeVisitNumber	

Hasta allí, ya sea manipulando la transacción o manipulando la variable BC hemos trabajado en memoria. Para que todo esto impacte en la base de datos, hay que dar la orden. En la transacción se hace con el botón Confirm (que como sabe que está en modo Update, intentará actualizar). En el BC lo hacemos con el método Update (también podríamos utilizar el método Save, pero no lo vamos a ver aquí). Como consecuencia de ello se ejecutarán las reglas, eventos y chequeos de la transacción que correspondan. Es decir, en el BC todos los que no dependan de la interfaz (y no estén calificados para ejecutarse solo en web, a través de los atributos de entorno que habíamos visto en el video anterior).

En definitiva, al trabajar con el BC tenemos que tener más cuidado en el sentido de que no vemos las líneas que existen y en esa programación podemos cometer errores. Luego veremos una forma más sencilla de insertar o modificar líneas (aunque no de eliminar) que es manipulándolas a través de un Data Provider.



Por último, queremos eliminar, para lo que agregamos el botón y en el evento asociado: primero cargamos la variable BC por clave primaria – aquí tenemos la tres variables de pantalla-, y luego aplicamos el método Delete al BC. Por ahora este método no devuelve True o False como sí lo hacen los Insert y Update. Por lo que para saber el resultado de la operación tendremos que ejecutar el método Success (o su contrario, el Fail).

# Hospital Backoffice



Recents Office Visit TEST

Schedule Date	11/17/22	<input type="button" value="29"/>	<input type="button" value="Insert Office Visit"/>
Physician Id	2		<input type="button" value="Update Office Visit"/>
Shift Id	1		<input type="button" value="Delete Office Visit"/>
Office Id	0		
Patient Id	0		
patient2 Id	0		
patient3 Id	0		

Id	Type	Description
SuccessfullyDeleted	Warning	Data has been successfully deleted.

Vamos a probar primero la eliminación de una consulta que sabemos que existe. Esta. Colocamos los valores de la clave. La eliminó sin inconveniente.

Si desde otra tabla hubiera un registro que estuviera referenciando al que quisimos eliminar, entonces la eliminación habría fallado por integridad referencial. Pero no es el caso. En nuestra aplicación por el momento no hay ninguna tabla subordinada de esta, es decir, que la referencie.



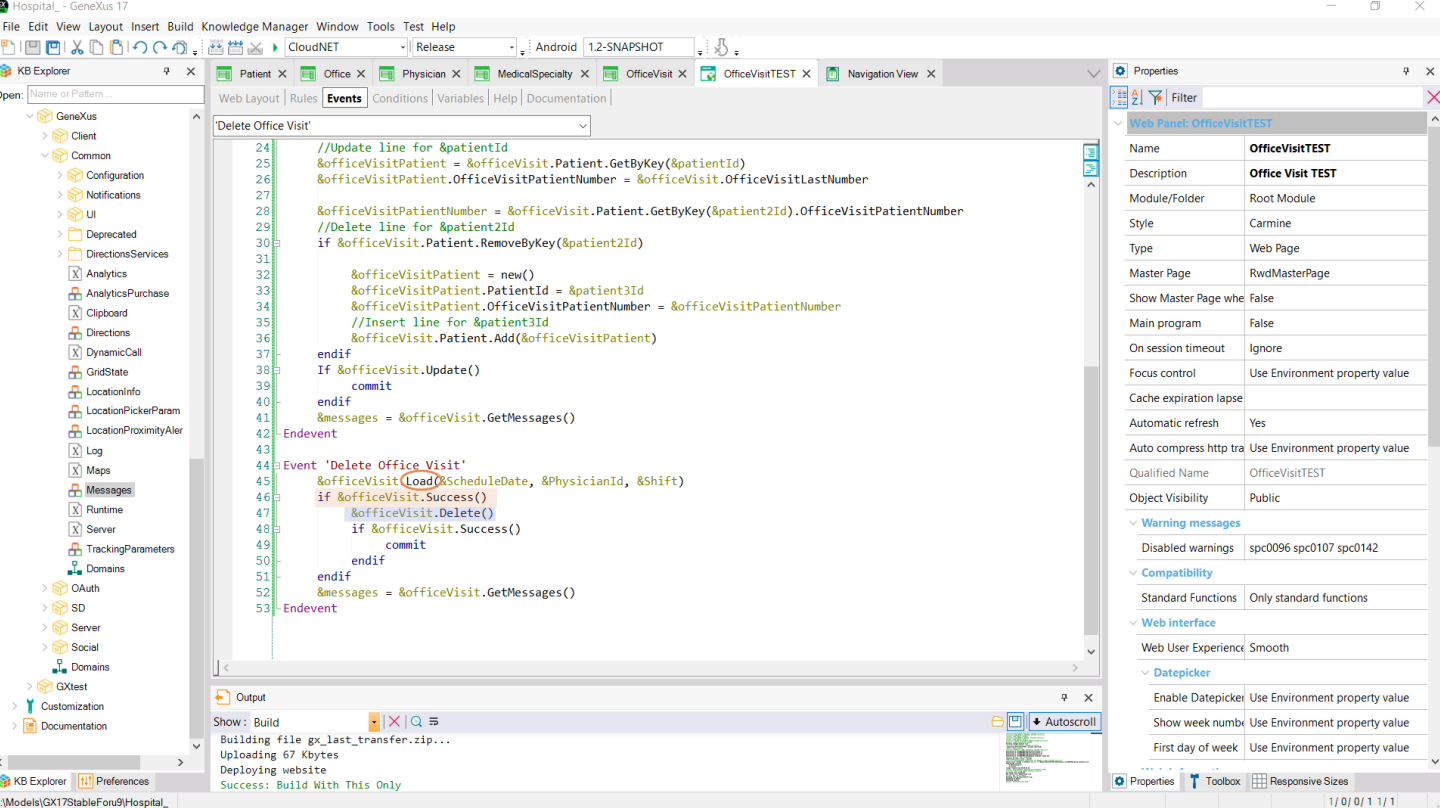
# Hospital Backoffice

Recents Office Visit TEST

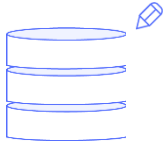
Schedule Date	12/30/22	<input type="button" value="29"/>	<input type="button" value="Insert Office Visit"/>
Physician Id	<input type="text" value="2"/>		<input type="button" value="Update Office Visit"/>
Shift Id	<input type="text" value="1"/>		<input type="button" value="Delete Office Visit"/>
Office Id	<input type="text" value="0"/>		
Patient Id	<input type="text" value="0"/>		
patient2 Id	<input type="text" value="0"/>		
patient3 Id	<input type="text" value="0"/>		

Id	Type	Description
CandidateKeyNotFound	Error	Please 'Get' data before trying to update it.

Y si ahora colocamos una fecha de consulta inexistente... vemos que nos arroja este error que no es muy claro porque menciona un update, pero en verdad el Get nos da la pauta de que falló porque no pudo realizar el Load.



Podríamos intentar el Delete únicamente si fue exitoso el Load. Por supuesto podrá pasar que el Load sea exitoso (porque existe el registro) pero falle el Delete, por ejemplo por integridad referencial, como mencionamos hace un instante. O porque exista una regla de error condicionada a dispararse en Delete y con otra condición que se satisfaga.



Insert, Update, Delete

Transaction

Business Component

Con esto, terminamos de repasar la relación entre la actualización a través de la transacción y la actualización por código a través del Business Component, viendo ejemplos de los tres modos.

# GeneXus™

[training.genexus.com](http://training.genexus.com)  
[wiki.genexus.com](http://wiki.genexus.com)  
[training.genexus.com/certifications](http://training.genexus.com/certifications)