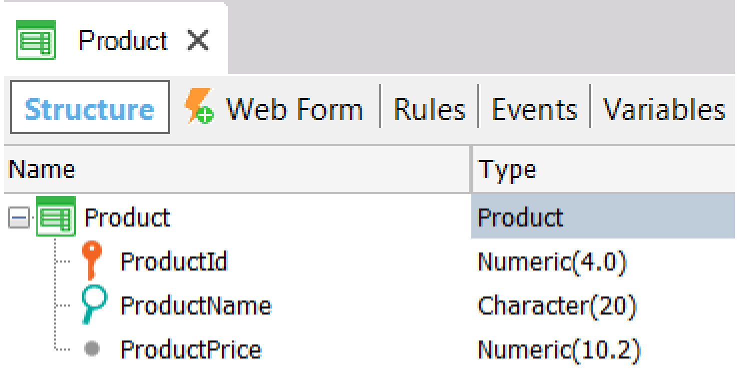
Curso GeneXus

Resumo Conceitual

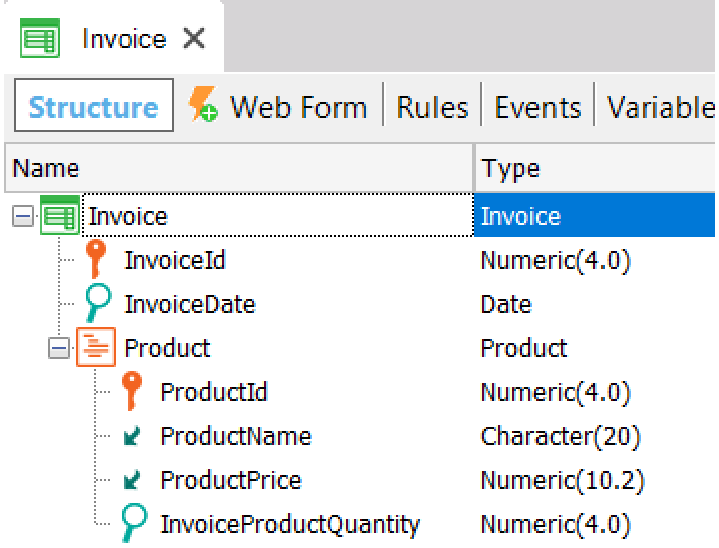
GeneXus™ 16

Transações

Quais nomes de atributos usamos?



Name	Type
Product	Product
ProductId	Numeric(4.0)
ProductName	Character(20)
ProductPrice	Numeric(10.2)



Name	Type
Invoice	Invoice
InvoiceId	Numeric(4.0)
InvoiceDate	Date
Product	Product
ProductId	Numeric(4.0)
ProductName	Character(20)
ProductPrice	Numeric(10.2)
InvoiceProductQuantity	Numeric(4.0)

Nomenclatura GIK

Entidade + Categoria [+ Qualificador]

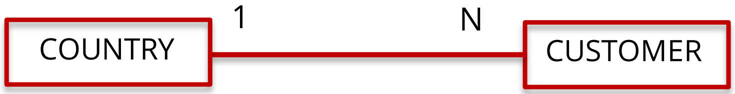
```
Product
{
    ProductId*  (PK)
    ProductName (S)
    ProductPrice (S)
}
```

```
Invoice
{
    InvoiceId* (PK)
    InvoiceDate (S)
    -----
    Product
    {
        ProductId* (PK, FK)
        ProductName  (I)
        ProductPrice  (I)
        InvoiceProductQuantity  (S)
        --------
    }
}
```

Desenho de Transações

1 – N forte

Cada cliente pertence a um pais e um pais tem muitos clientes



COUNTRY 1 — N CUSTOMER

COUNTRY
CountryId*
CountryName

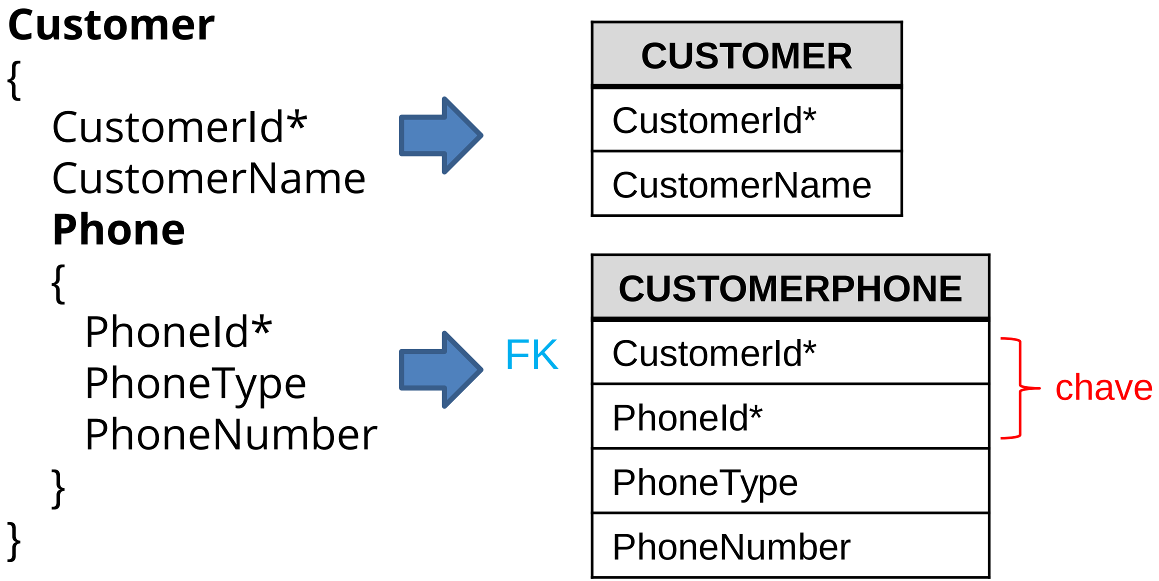
```
Country
{
   CountryId*   (PK)
   CountryName
}
```

```
Customer
{
   CustomerId*
   CustomerName
   CountryId    (FK)
   CountryName
}
```

CUSTOMER
CustomerId*
CustomerName
CountryId

1 – N fraca

Cada cliente tem muitos telefones e cada telefone pertence a um único cliente



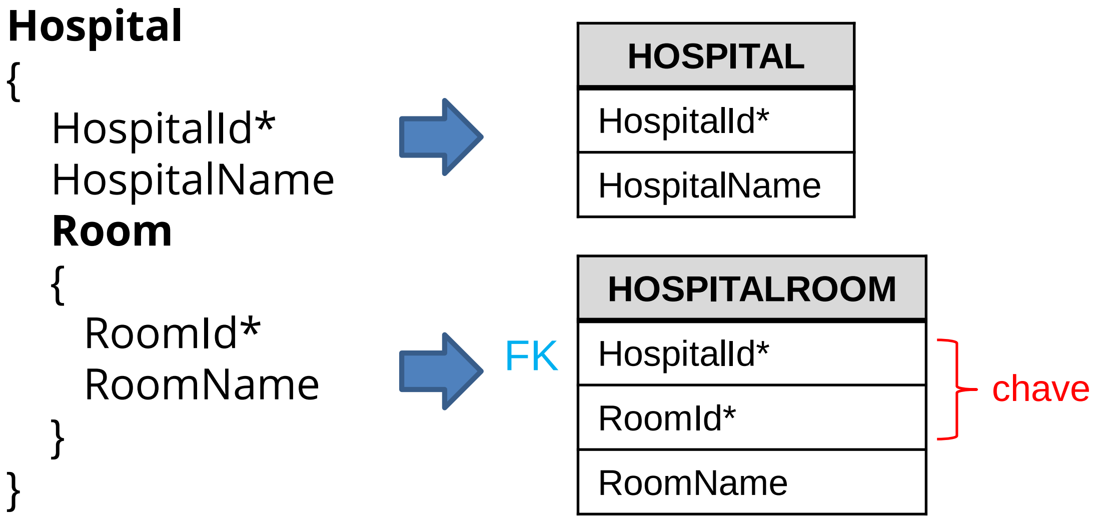
```
Customer
{
   CustomerId*
   CustomerName
   Phone
   {
      PhoneId*
      PhoneType
      PhoneNumber
   }
}
```

CUSTOMER
CustomerId*
CustomerName

CUSTOMERPHONE
CustomerId*
PhoneId*
PhoneType
PhoneNumber

1 – N fraca

Cada hospital tem muitas salas e cada sala pertence a um único hospital



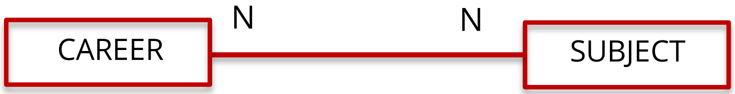
```
Hospital
{
   HospitalId*
   HospitalName
   Room
   {
      RoomId*
      RoomName
   }
}
```

HOSPITAL
HospitalId*
HospitalName

HOSPITALROOM
HospitalId*
RoomId*
RoomName

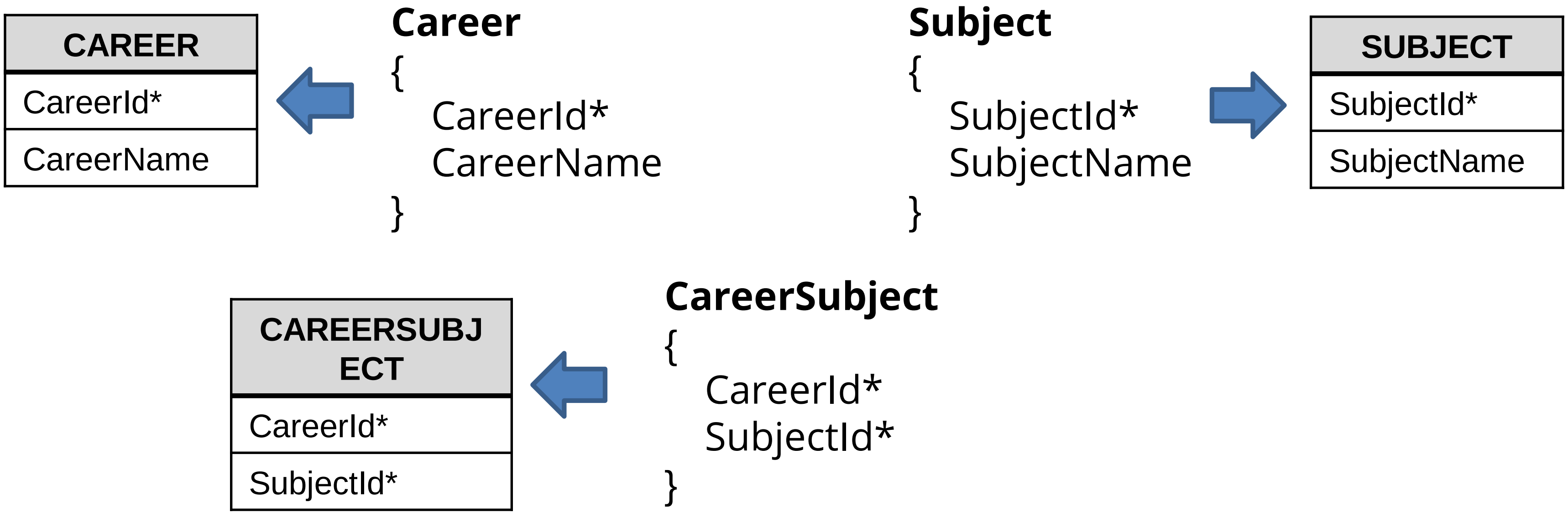
N – N (M)

Cada carreira tem muitas disciplinas e cada disciplina pode estar em muitas carreiras



N – N: Opção 1 de 4

Cada carreira tem muitas disciplinas e cada disciplina pode estar em muitas carreiras



CAREER
CareerId*
CareerName

```
Career
{
    CareerId*
    CareerName
}
```

```
Subject
{
    SubjectId*
    SubjectName
}
```

SUBJECT
SubjectId*
SubjectName

CAREERSUBJECT
CareerId*
SubjectId*

```
CareerSubject
{
    CareerId*
    SubjectId*
}
```

N – N: Opção 1 – Tabelas geradas

CAREER

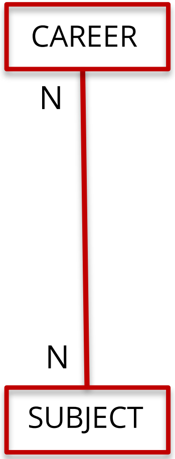
CareerId	CareerName
1	Computer Science
2	Data Science for Health

SUBJECT

SubjectId	SubjectName
1	Computer Logic
2	Programming Fundamentals

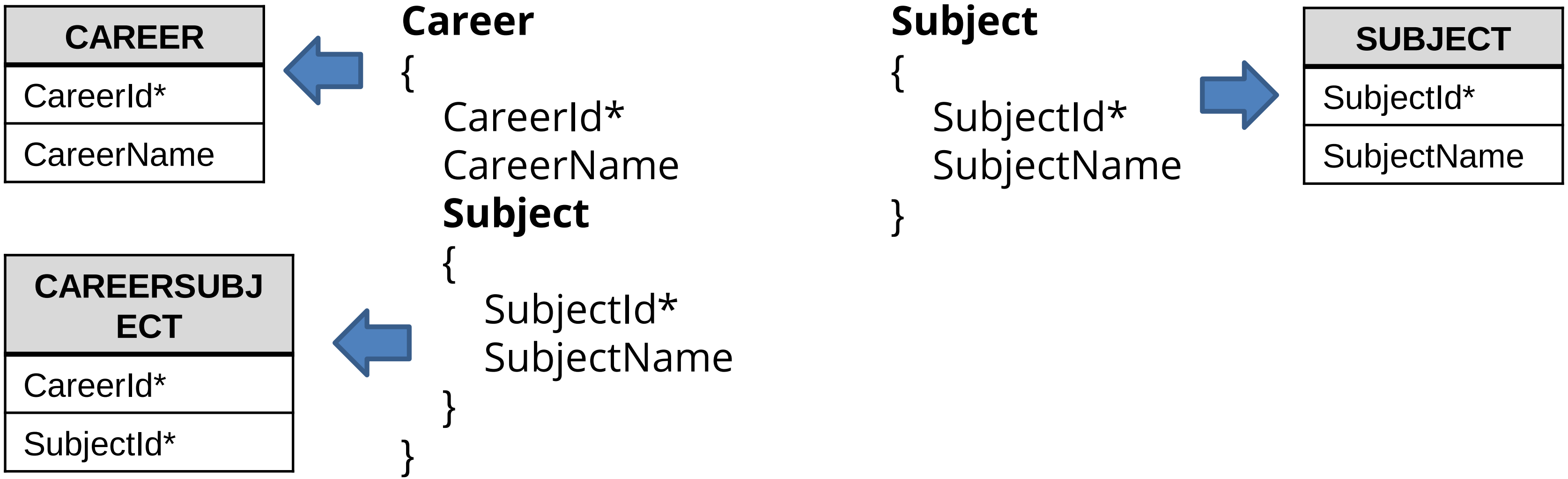
CAREERSUBJECT

CareerId	SubjectId
1	1
1	2
2	2



N – N: Opção 2 de 4

Cada carreira tem muitas disciplinas e cada disciplina pode estar em muitas carreiras



CAREER
CareerId*
CareerName

CAREERSUBJECT
CareerId*
SubjectId*

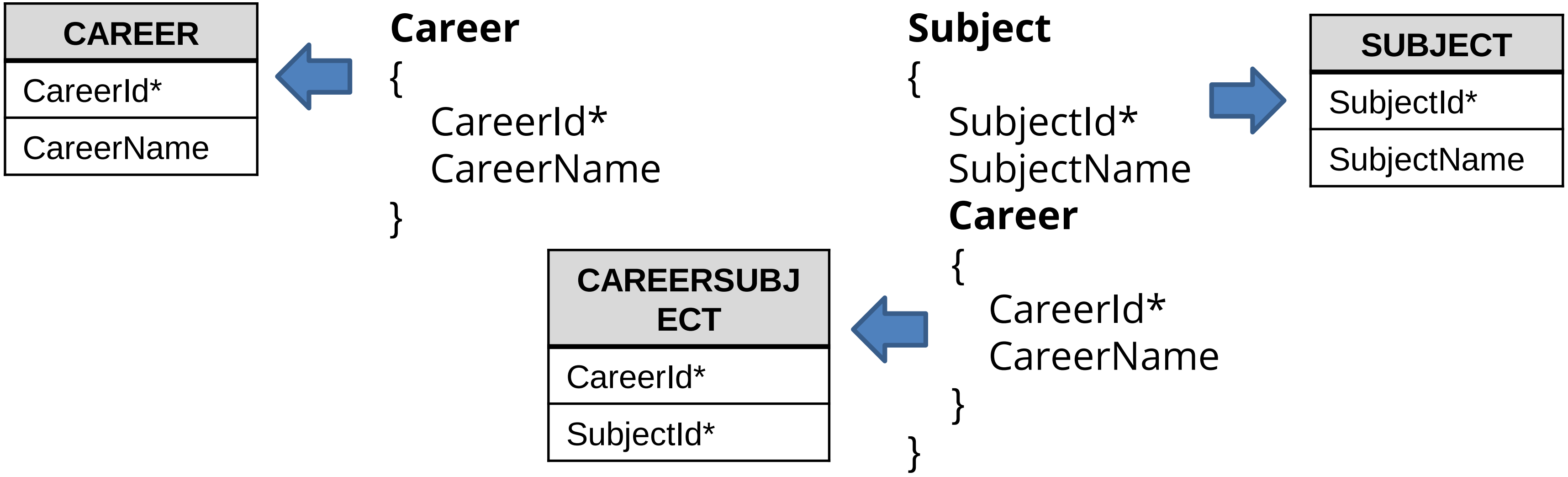
```
Career
{
    CareerId*
    CareerName
    Subject
    {
        SubjectId*
        SubjectName
    }
}
```

```
Subject
{
    SubjectId*
    SubjectName
}
```

SUBJECT
SubjectId*
SubjectName

N – N: Opção 3 de 4

Cada carreira tem muitas disciplinas e cada disciplina pode estar em muitas carreiras



CAREER
CareerId*
CareerName

```
Career
{
    CareerId*
    CareerName
}
```

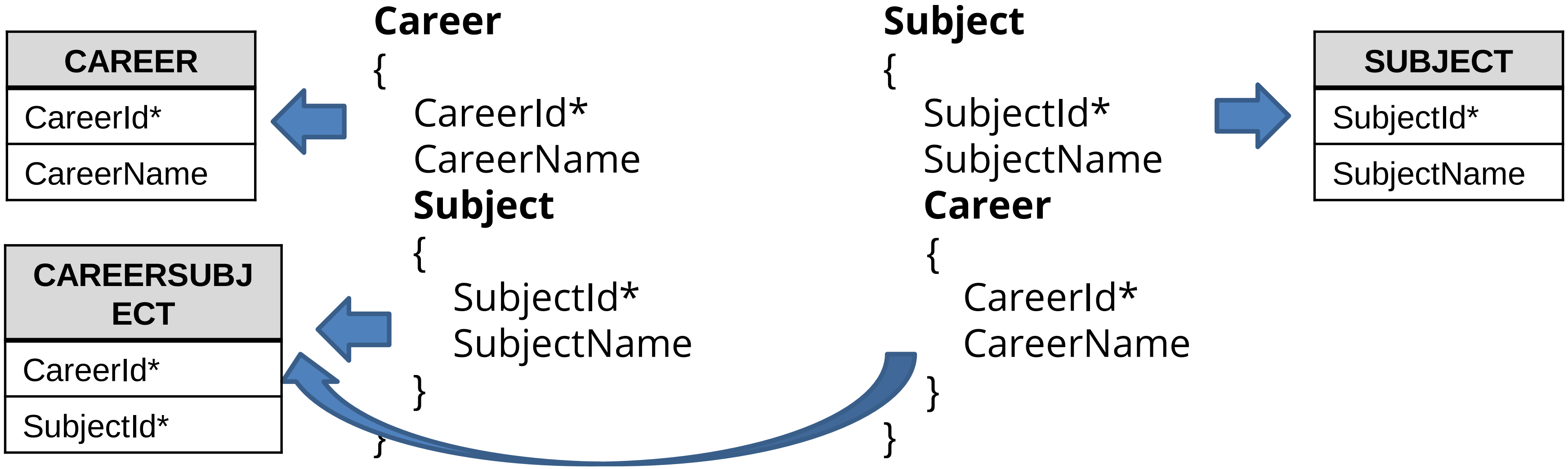
```
Subject
{
    SubjectId*
    SubjectName
    Career
    {
        CareerId*
        CareerName
    }
}
```

SUBJECT
SubjectId*
SubjectName

CAREERSUBJECT
CareerId*
SubjectId*

N – N: Opção 4 de 4

Cada carreira tem muitas disciplinas e cada disciplina pode estar em muitas carreiras



CAREER
CareerId*
CareerName

CAREERSUBJECT
CareerId*
SubjectId*

```
Career
{
   CareerId*
   CareerName
   Subject
   {
      SubjectId*
      SubjectName
   }
}
```

```
Subject
{
   SubjectId*
   SubjectName
   Career
   {
      CareerId*
      CareerName
   }
}
```

SUBJECT
SubjectId*
SubjectName

1 - 1

Cada curso é ministrado em uma sala e esta sala só pode ser usada para ministrar esse curso

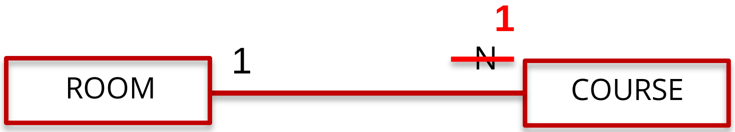
ROOM
RoomId*
RoomName

```
Room
{
    RoomId*
    RoomName
}
```

```
Course
{
    CourseId*
    CourseName
    RoomId
    RoomName
}
```

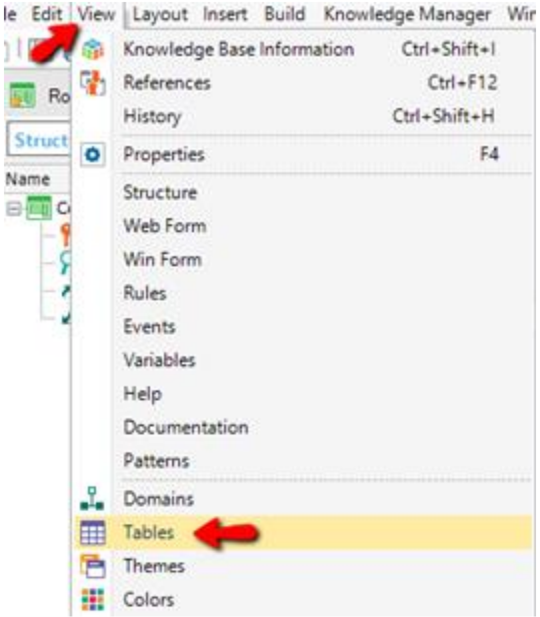
COURSE
CourseId*
CourseName
RoomId

Unique index

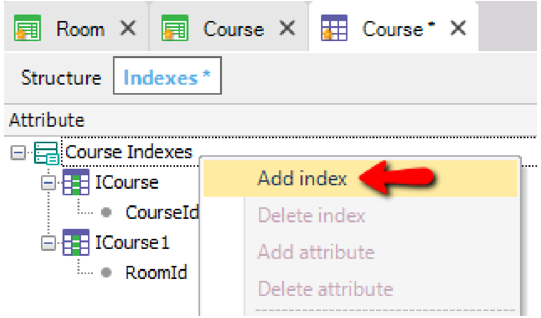


Criação de um índice

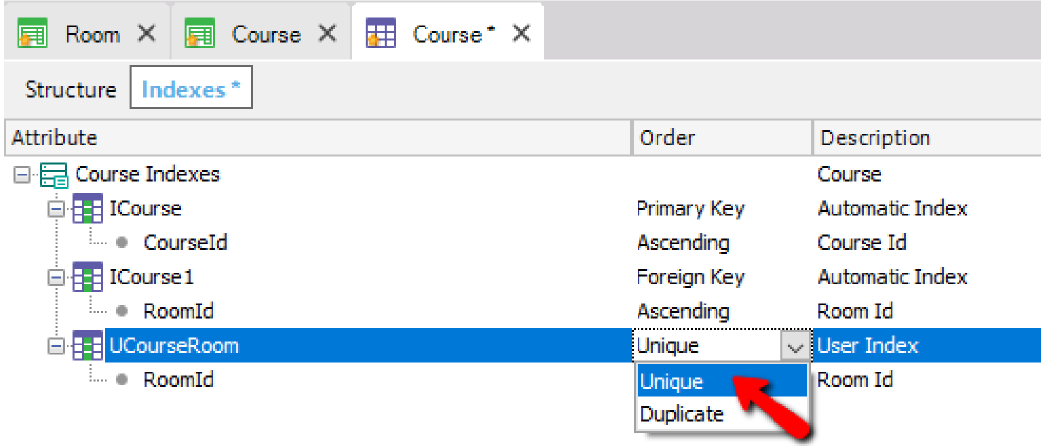
1



2



3

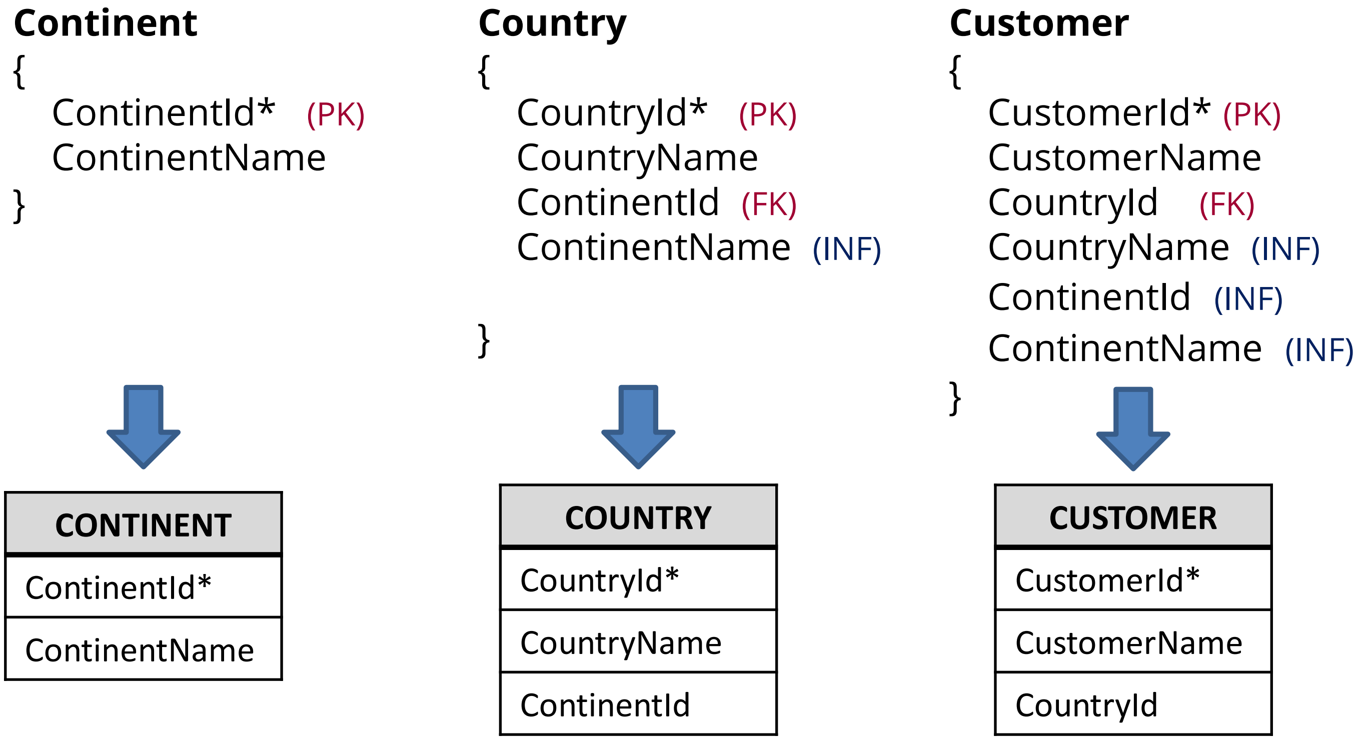


Attribute	Order	Description
Course Indexes		Course
ICourse	Primary Key	Automatic Index
CourseId	Ascending	Course Id
ICourse1	Foreign Key	Automatic Index
RoomId	Ascending	Room Id
UCourseRoom	Unique	User Index
RoomId		Room Id

Normalização

GeneXus normaliza as tabelas na Terceira Forma Normal (3NF)

- Atributos inferidos em uma transação não são incluídos na tabela gerada



Continent
```
{
   ContinentId*   (PK)
   ContinentName
}
```

Country
```
{
   CountryId*   (PK)
   CountryName
   ContinentId  (FK)
   ContinentName  (INF)

}
```

Customer
```
{
   CustomerId* (PK)
   CustomerName
   CountryId    (FK)
   CountryName  (INF)
   ContinentId  (INF)
   ContinentName  (INF)
}
```

CONTINENT
ContinentId*
ContinentName

COUNTRY
CountryId*
CountryName
ContinentId

CUSTOMER
CustomerId*
CustomerName
CountryId

Integridade referencial

Integridade referencial

```
Country
{
   CountryId*   (PK)
   CountryName
}
```

```
Customer
{
   CustomerId*
   CustomerName
   CountryId    (FK)
   CountryName
}
```

CountryId	CountryName
1	URUGUAI
2	ARGENTINA

CustomerId	CustomerName	CountryId	
1	ANA	1	✓
2	PEDRO	2	✓
3	LUIS	2	✓
4	JOSE	3	✗

O registro não é inserido ⇦

Integridade referencial

```
Country
{
    CountryId*   (PK)
    CountryName
}
```

```
Customer
{
    CustomerId*
    CustomerName
    CountryId    (FK)
    CountryName
}
```

CountryId	CountryName
1	URUGUAI ✗
2	ARGENTINA

O registro não é eliminado

CustomerId	CustomerName	CountryId
1	ANA	1
2	PEDRO	2
3	LUIS	2
4	JOSE	3

Tabela Base e Estendida

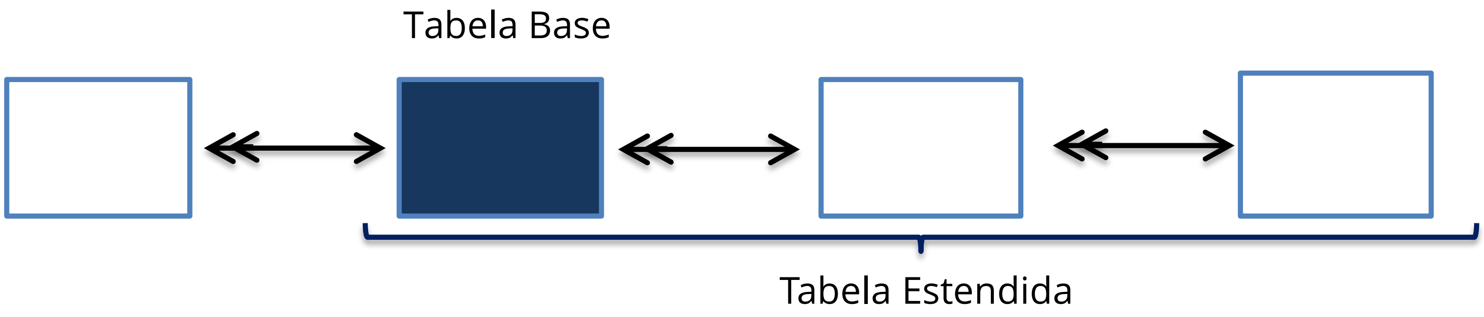
- **Tabela Base**

  É qualquer tabela do banco de dados em que estamos posicionados trabalhando em um determinado momento.

- **Tabela Estendida**

  Dada uma tabela, sua tabela estendida é um conceito que nos permite pensar todas as informações às quais temos acesso, a partir de suas chaves externas.

  É o conjunto de atributos da própria tabela + todos os atributos das tabelas com as quais ela tem um relacionamento N-1 direta ou indiretamente.

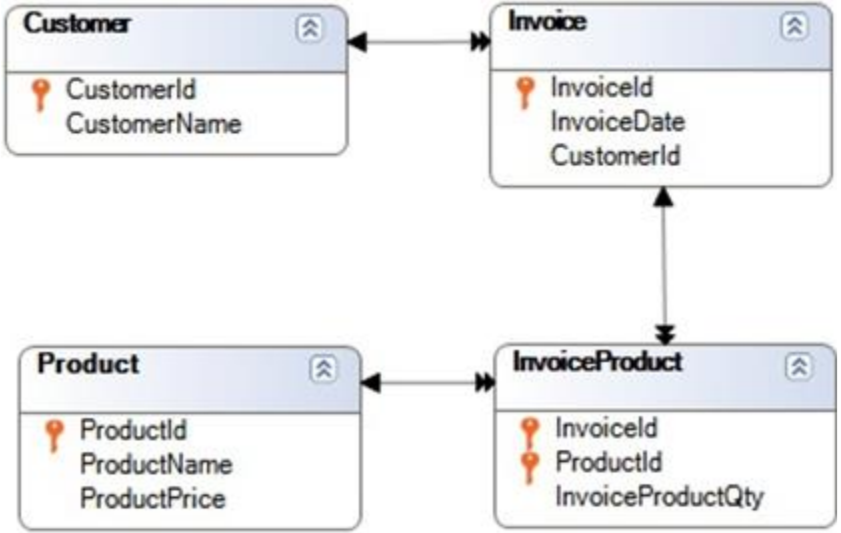


Exemplo

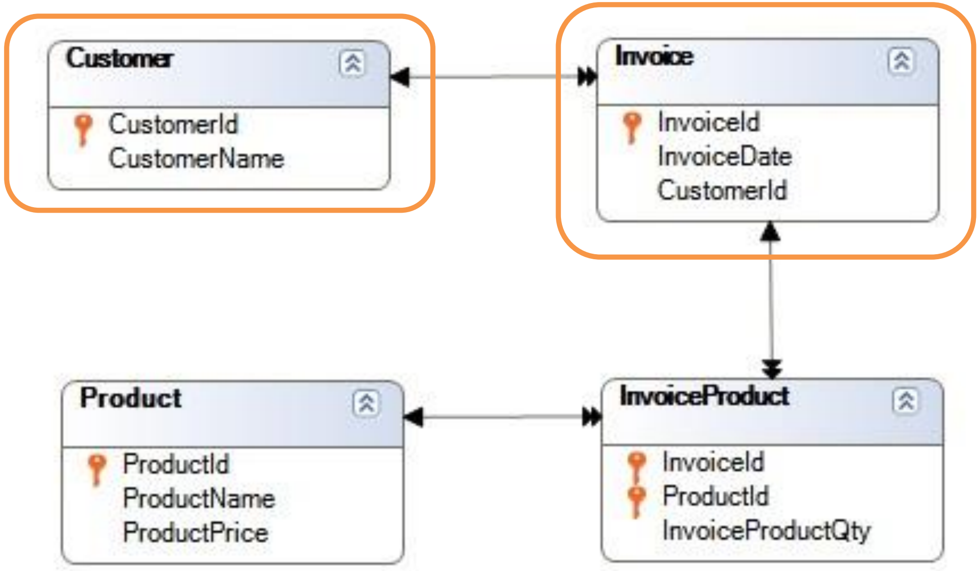
```
Customer
{
   CustomerId*
   CustomerName
}

Product
{
   ProductId*
   ProductName
   ProductPrice
}
```

```
Invoice
{
   InvoiceId*
   InvoiceDate
   CustomerId
   CustomerName
   Product
   {
      ProductId*
      ProductName
      ProductPrice
      InvoiceProductQuantity
      --------
   }
}
```

Diagrama de Tabelas
(diagrama de Bachman)



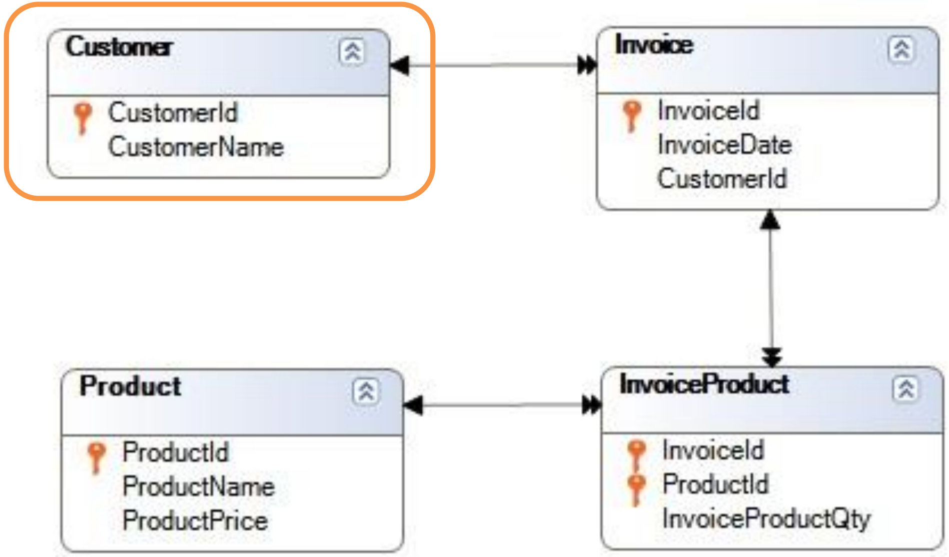
Exemplo: Tabela Estendida de Invoice



BASE TRN

Exemplo: Tabela Estendida de Customer

BASE TRN



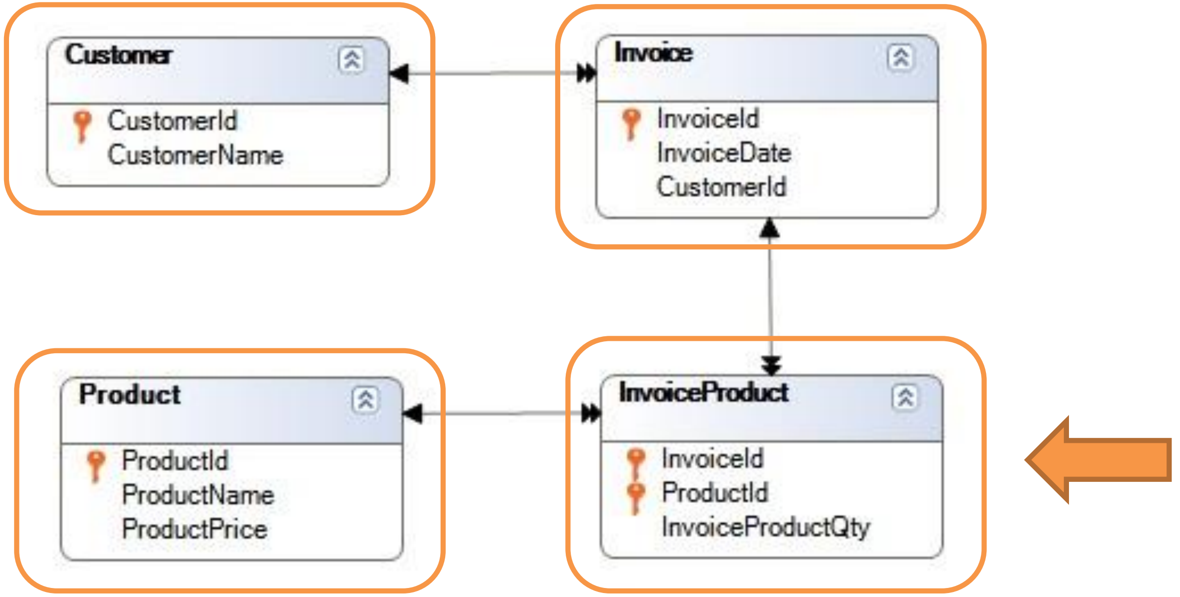
Customer
CustomerId
CustomerName

Invoice
InvoiceId
InvoiceDate
CustomerId

Product
ProductId
ProductName
ProductPrice

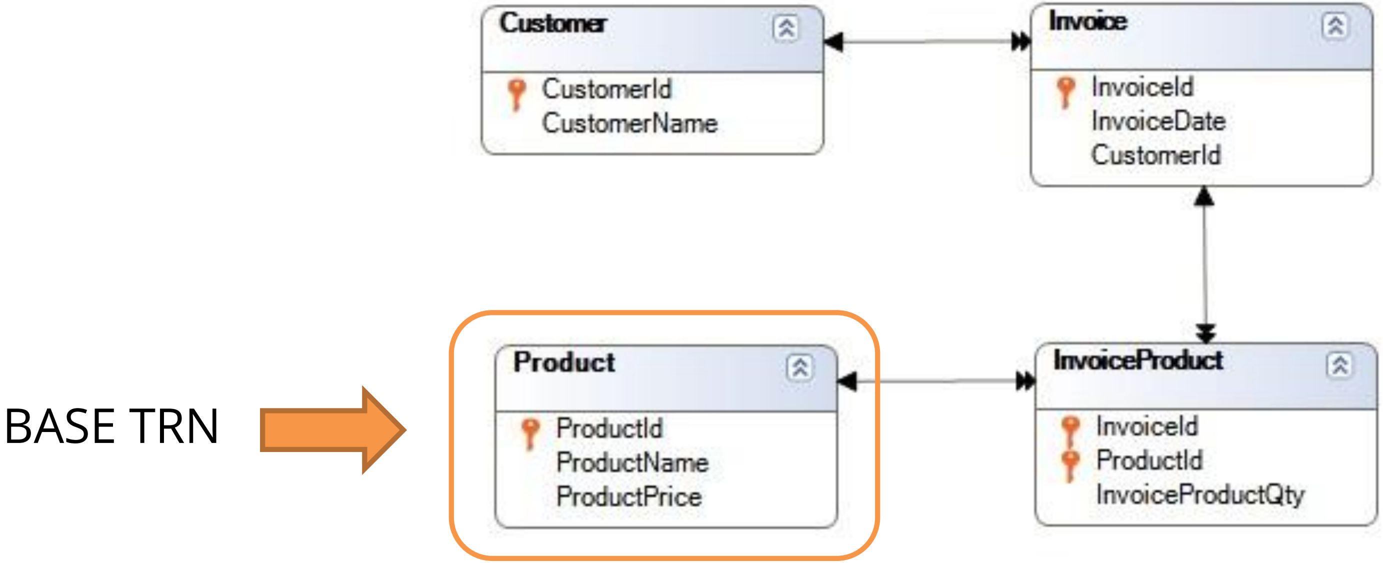
InvoiceProduct
InvoiceId
ProductId
InvoiceProductQty

Exemplo: Tabela Estendida de InvoiceProduct



BASE TRN

Exemplo: Tabela Estendida de Product



BASE TRN

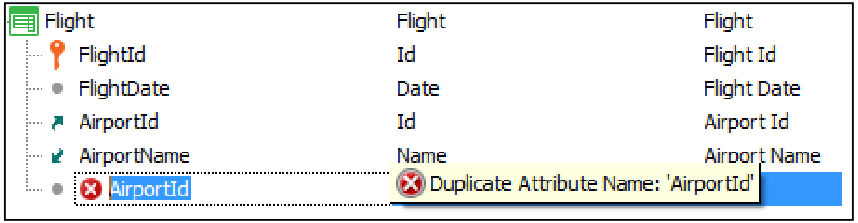
Subtipos

Múltiplas Referências :
É preciso salvar, para cada vôo, o aeroporto de partida e o aeroporto de chegada

```
Airport
{
   AirportId*
   AirportName
}
```

```
Flight
{
   FlightId*
   FlightDate
   AirportId
   AirportName
   AirportId
   AirportName
}
```

Erro por nome de atributos duplicados



Flight	Flight	Flight
FlightId	Id	Flight Id
FlightDate	Date	Flight Date
AirportId	Id	Airport Id
AirportName	Name	Airport Name
AirportId		

Duplicate Attribute Name: 'AirportId'

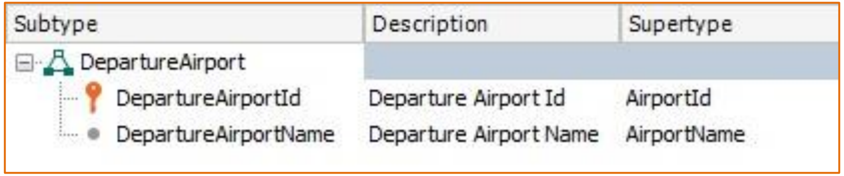
Solução 1 de 3: criar dois grupos de subtipos, um para o aeroporto de partida e outro para o aeroporto de chegada

Airport
{
AirportId*
AirportName
}

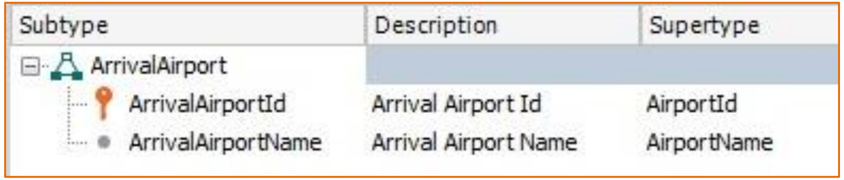
Flight
{
FlightId*
FlightDate
DepartureAirportId
DepartureAirportName
ArrivalAirportId
ArrivalAirportName
}

Subtype group: DepartureAirport

Subtype group: ArrivalAirport



Subtype	Description	Supertype
DepartureAirport		
DepartureAirportId	Departure Airport Id	AirportId
DepartureAirportName	Departure Airport Name	AirportName



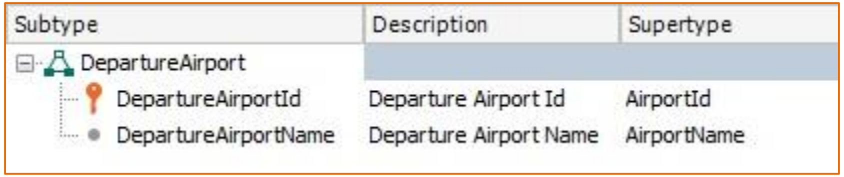
Subtype	Description	Supertype
ArrivalAirport		
ArrivalAirportId	Arrival Airport Id	AirportId
ArrivalAirportName	Arrival Airport Name	AirportName

Solução 2 de 3: criar um grupo de subtipos, somente para o aeroporto de partida

Airport
{
AirportId*
AirportName
}

Flight
{
FlightId*
FlightDate
DepartureAirportId
DepartureAirportName
AirportId
AirportName
}

Subtype group: DepartureAirport



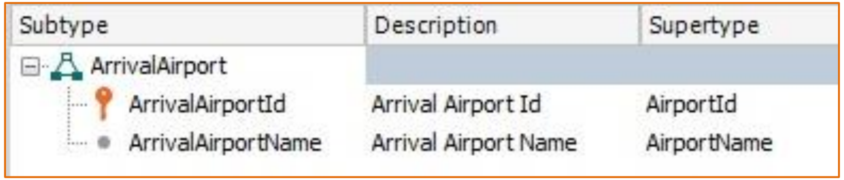
Subtype	Description	Supertype
DepartureAirport		
DepartureAirportId	Departure Airport Id	AirportId
DepartureAirportName	Departure Airport Name	AirportName

Solução 3 de 3: criar um grupo de subtipos, somente para o aeroporto de chegada

```
Airport
{
    AirportId*
    AirportName
}
```

```
Flight
{
    FlightId*
    FlightDate
    AirportId
    AirportName
    ArrivalAirportId
    ArrivalAirportName
}
```

Subtype group: ArrivalAirport



Subtype	Description	Supertype
ArrivalAirport		
ArrivalAirportId	Arrival Airport Id	AirportId
ArrivalAirportName	Arrival Airport Name	AirportName

Múltiplas Referências :
É preciso salvar, além do país do cliente, o país onde foi emitida a Fatura

Country

```
{
    CountryId*
    CountryName
}
```

Customer

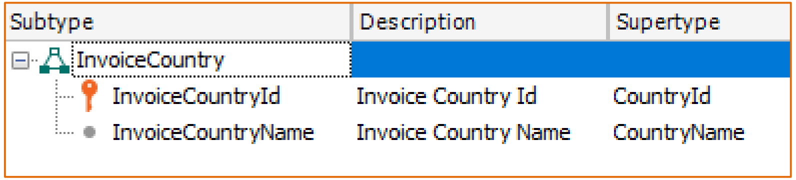
```
{
    CustomerId*
    CustomerName
    CountryId
    CountryName
}
```

Invoice

```
{
    InvoiceId*
    InvoiceDate
    CustomerId
    CustomerName
    CountryId
    CountryName
    InvoiceCountryId
    InvoiceCountryName
}
```

Atributos inferidos

Subtype group: InvoiceCountry



Subtype	Description	Supertype
InvoiceCountry		
InvoiceCountryId	Invoice Country Id	CountryId
InvoiceCountryName	Invoice Country Name	CountryName

Múltiplas Referências: problema

Subject

```
{
   SubjectId*
   SubjectName
   TeacherId
   TeacherName
   TeacherId
   TeacherName
}
```

Professor titular

Professor suplente

Erro por nomes de atributos duplicados

Teacher

```
{
   TeacherId*
   TeacherName
}
```

Múltiplas Referências : solução

Subject

{

SubjectId*

SubjectName

SubjectPermanentTeacherId

SubjectPermanentTeacherName ------- Professor titular

SubjectSubstituteTeacherId

SubjectSubstituteTeacherName ------- Professor suplente

}

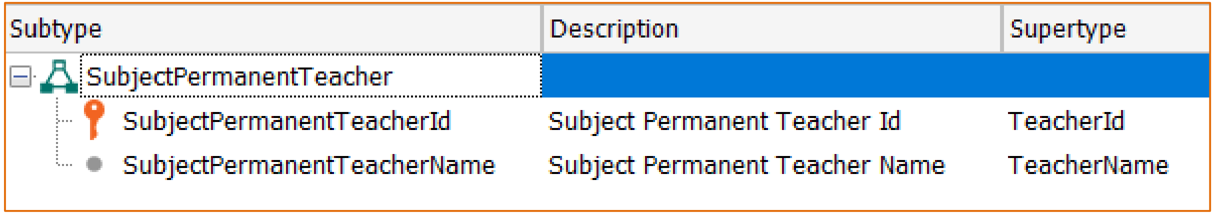
Teacher

{

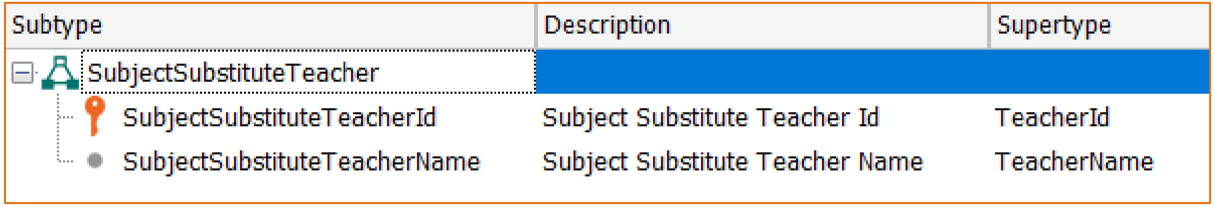
TeacherId*

TeacherName

}



Subtype	Description	Supertype
SubjectPermanentTeacher		
SubjectPermanentTeacherId	Subject Permanent Teacher Id	TeacherId
SubjectPermanentTeacherName	Subject Permanent Teacher Name	TeacherName



Subtype	Description	Supertype
SubjectSubstituteTeacher		
SubjectSubstituteTeacherId	Subject Substitute Teacher Id	TeacherId
SubjectSubstituteTeacherName	Subject Substitute Teacher Name	TeacherName

Múltiplas Referências : problema

```
Subject
{
   SubjectId*
   SubjectName
}

Teacher
{
   TeacherId*
   TeacherName
   SubjectId
   SubjectName
}
```

```
Exam
{
   ExamId*
   ExamDate
   SubjectId
   SubjectName
   Teacher
   {
      TeacherId*
      TeacherName
      SubjectId
      SubjectName
   }
}
```

Atributos inferidos

Erro por nomes de atributos duplicados

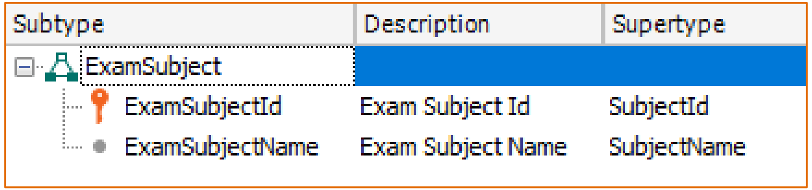
Múltiplas Referências : solução

```
Subject
{
   SubjectId*
   SubjectName
}

Teacher
{
   TeacherId*
   TeacherName
   SubjectId
   SubjectName
 }
```

```
Exam
{
   ExamId*
   ExamDate
   ExamSubjectId
   ExamSubjectName
   Teacher
   {
      TeacherId*
      TeacherName
      SubjectId
      SubjectName
   }
}
```

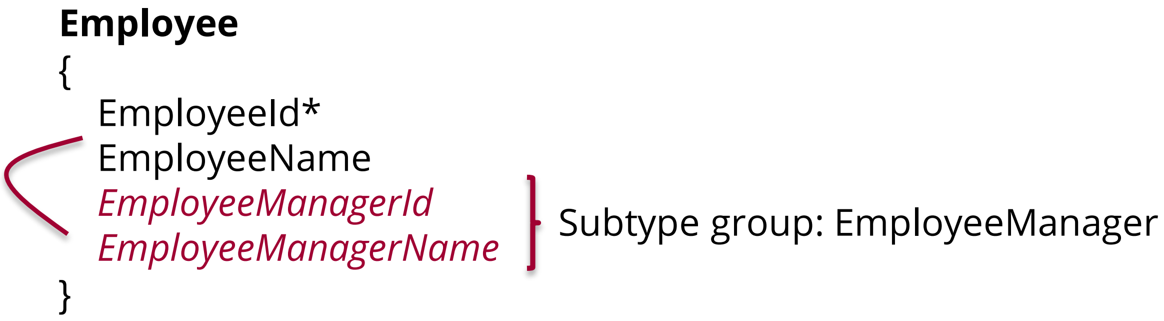
Subtype group: ExamSubject

Atributos inferidos



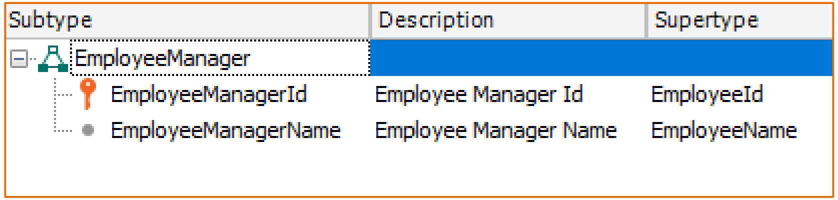
Subtype	Description	Supertype
ExamSubject		
ExamSubjectId	Exam Subject Id	SubjectId
ExamSubjectName	Exam Subject Name	SubjectName

Subtipos recursivos



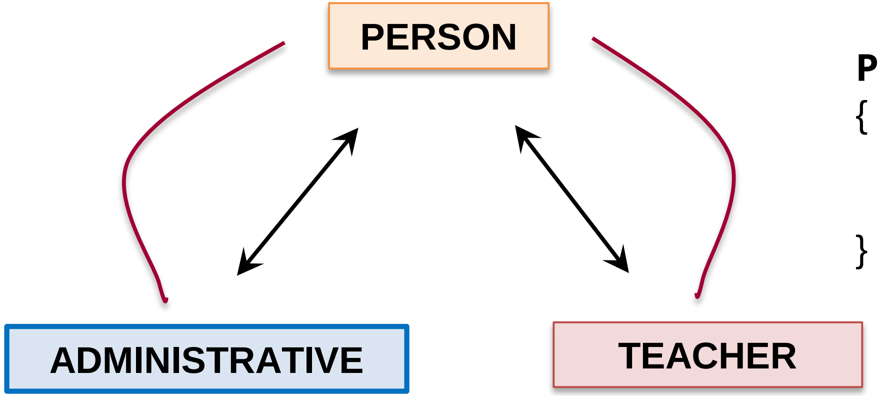
Employee
{
EmployeeId*
EmployeeName
EmployeeManagerId
EmployeeManagerName
}

Subtype group: EmployeeManager



Subtype	Description	Supertype
EmployeeManager		
EmployeeManagerId	Employee Manager Id	EmployeeId
EmployeeManagerName	Employee Manager Name	EmployeeName

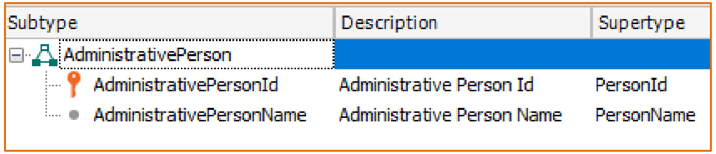
Especialização



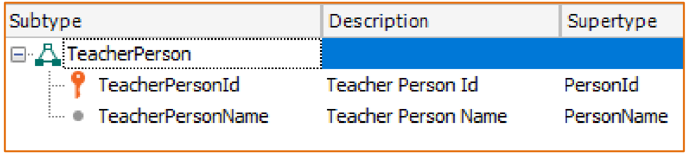
Person
{
    PersonId*
    PersonName
}

Administrative
{
    AdministrativeId*
    AdministrativeName
    AdministrativeLanguage
}

Teacher
{
    TeacherId*
    TeacherName
    TeacherTitle
}



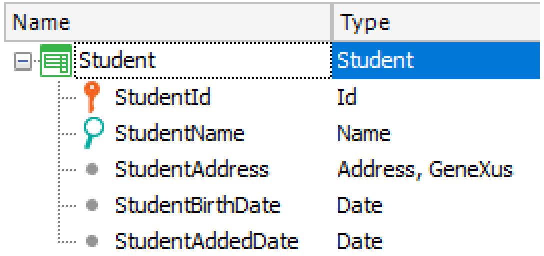
Subtype	Description	Supertype
AdministrativePerson		
AdministrativePersonId	Administrative Person Id	PersonId
AdministrativePersonName	Administrative Person Name	PersonName



Subtype	Description	Supertype
TeacherPerson		
TeacherPersonId	Teacher Person Id	PersonId
TeacherPersonName	Teacher Person Name	PersonName

Regras

Regras



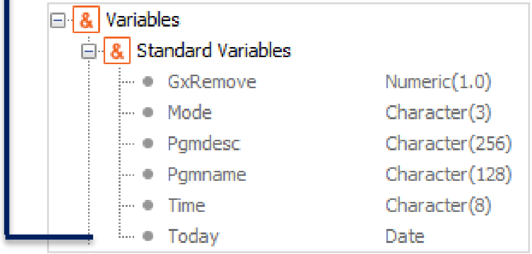
Name	Type
Student	Student
StudentId	Id
StudentName	Name
StudentAddress	Address, GeneXus
StudentBirthDate	Date
StudentAddedDate	Date

Error(“Enter the student name”) if StudentName.isEmpty();

Msg(“The address is empty”) if StudentAddress.isEmpty();

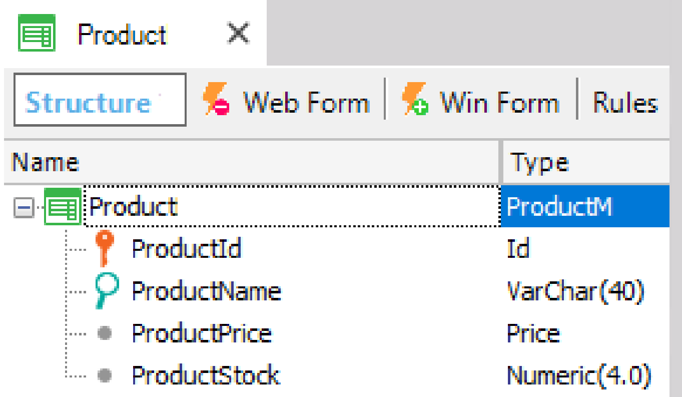
Default(StudentAddedDate, &Today);

Noaccept(StudentAddedDate);



Variables	
Standard Variables	
GxRemove	Numeric(1.0)
Mode	Character(3)
Pgmdesc	Character(256)
Pgmname	Character(128)
Time	Character(8)
Today	Date

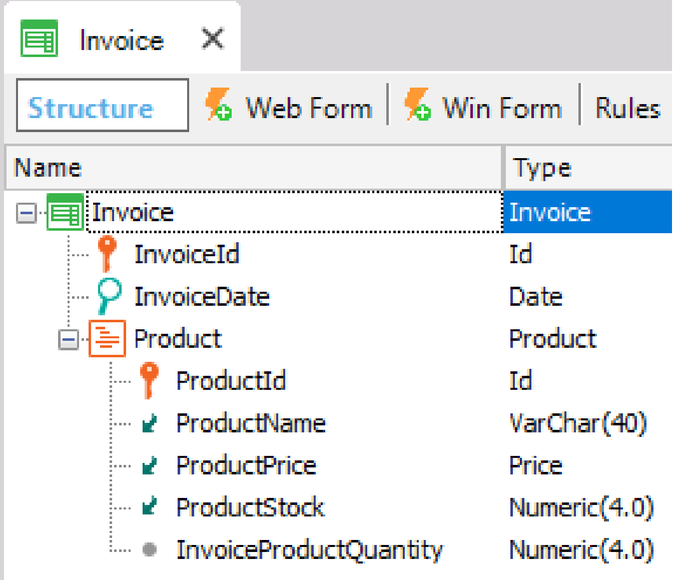
Regras



Product ×

Structure | Web Form | Win Form | Rules

Name	Type
Product	ProductM
ProductId	Id
ProductName	VarChar(40)
ProductPrice	Price
ProductStock	Numeric(4.0)



Invoice ×

Structure | Web Form | Win Form | Rules

Name	Type
Invoice	Invoice
InvoiceId	Id
InvoiceDate	Date
Product	Product
ProductId	Id
ProductName	VarChar(40)
ProductPrice	Price
ProductStock	Numeric(4.0)
InvoiceProductQuantity	Numeric(4.0)

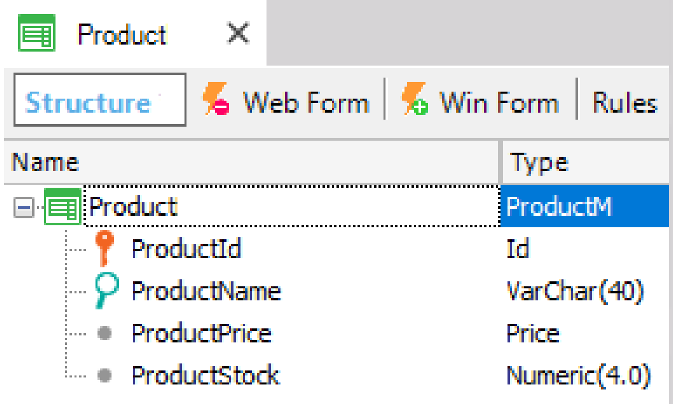
Insert
update
delete
display

Funções booleanas

ProductStock = ProductStock - 100 if insert;

ProductStock = ProductStock + 100 if delete;

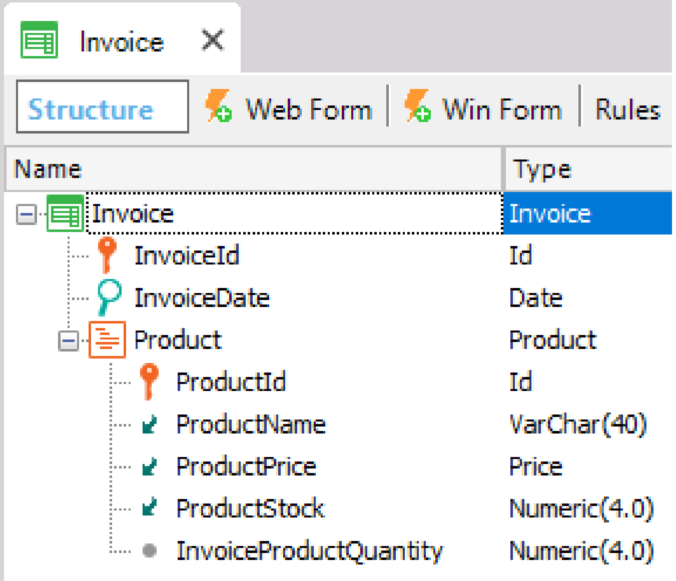
Regras



Product ✕

Structure | Web Form | Win Form | Rules

Name	Type
⊟ Product	ProductM
ProductId	Id
ProductName	VarChar(40)
ProductPrice	Price
ProductStock	Numeric(4.0)



Invoice ✕

Structure | Web Form | Win Form | Rules

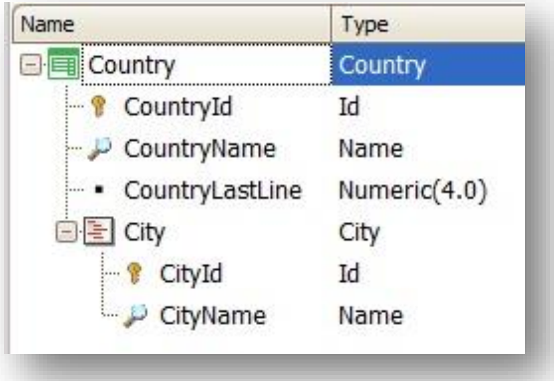
Name	Type
⊟ Invoice	Invoice
InvoiceId	Id
InvoiceDate	Date
⊟ Product	Product
ProductId	Id
ProductName	VarChar(40)
ProductPrice	Price
ProductStock	Numeric(4.0)
InvoiceProductQuantity	Numeric(4.0)

ProductStock = ProductStock - 100;

Subtract(InvoiceProductQuantity, ProductStock);

Add(500, ProductStock);

Regras



Name	Type
Country	Country
CountryId	Id
CountryName	Name
CountryLastLine	Numeric(4.0)
City	City
CityId	Id
CityName	Name

Serial(CityId, CountryLastLine, 1);

Parm(attribute1, &variable1, ….);

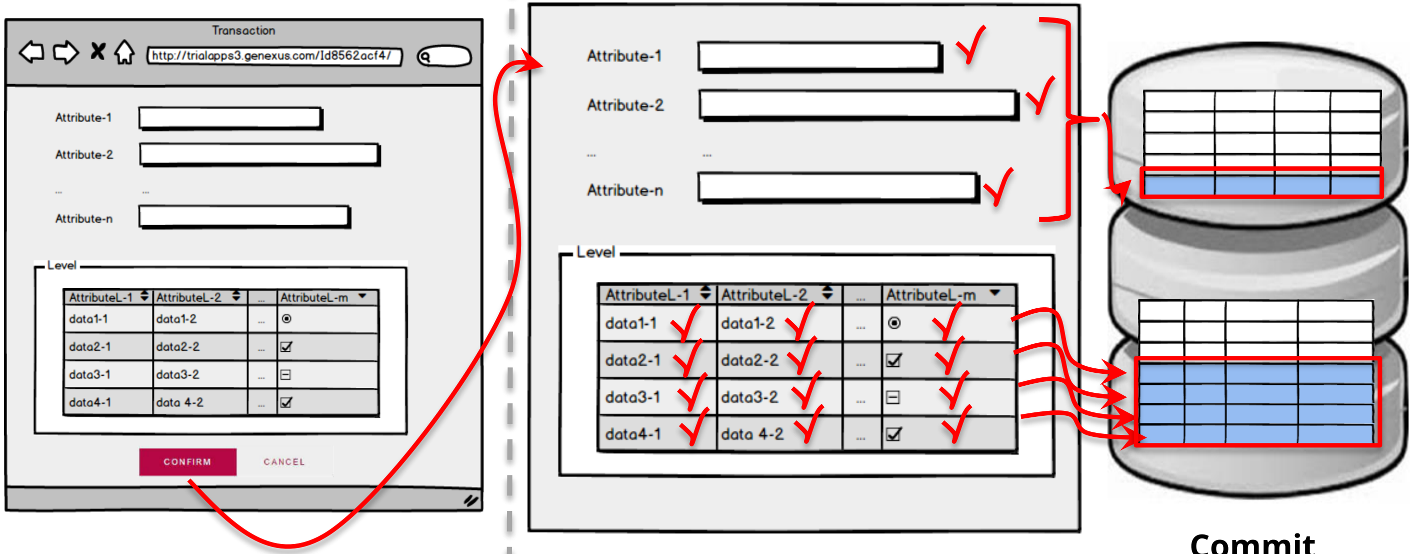
Variável: Espaço em memória que posssui um nome e o tipo de dados que pode guardar. São referenciadas através do “&”.

Momentos de disparo

Browser

Server

Database



Transaction

http://trialapps3.genexus.com/Id8562acf4/

Attribute-1

Attribute-2

... ...

Attribute-n

Level

AttributeL-1	AttributeL-2	...	AttributeL-m
data1-1	data1-2	...	●
data2-1	data2-2	...	☑
data3-1	data3-2	...	⊟
data4-1	data 4-2	...	☑

CONFIRM CANCEL

Attribute-1 ✓

Attribute-2 ✓

... ...

Attribute-n ✓

Level

AttributeL-1	AttributeL-2	...	AttributeL-m
data1-1 ✓	data1-2 ✓	...	● ✓
data2-1 ✓	data2-2 ✓	...	☑ ✓
data3-1 ✓	data3-2 ✓	...	⊟ ✓
data4-1 ✓	data 4-2 ✓	...	☑ ✓

Commit

Momentos de disparo de regras

No servidor, depois de pressionar Confirm

Em transações de um só nivel:

On BeforeValidate

VALIDAÇÃO

On AfterValidate

On BeforeInsert/BeforeUpdate/ BeforeDelete

GRAVAÇÃO

On AfterInsert/AfterUpdate/ AfterDelete

On BeforeComplete

COMMIT

On AfterComplete

Em transações de dois níveis:

On BeforeValidate

VALIDAÇÃO do cabeçalho

On AfterValidate

On BeforeInsert/BeforeUpdate/BeforeDelete

GRAVAÇÃO do cabeçalho

On AfterInsert/AfterUpdate/AfterDelete

Para cada línha

- On BeforeValidate
- **VALIDAÇÃO da línha**
- On AfterValidate
- OnBeforeInsert/BeforeUpdate/BeforeDelete
- **GRAVAÇÃO da línha**
- On AfterInsert/AfterUpdate/AfterDelete

On AferLevel Level Atributo das línhas

On BeforeComplete

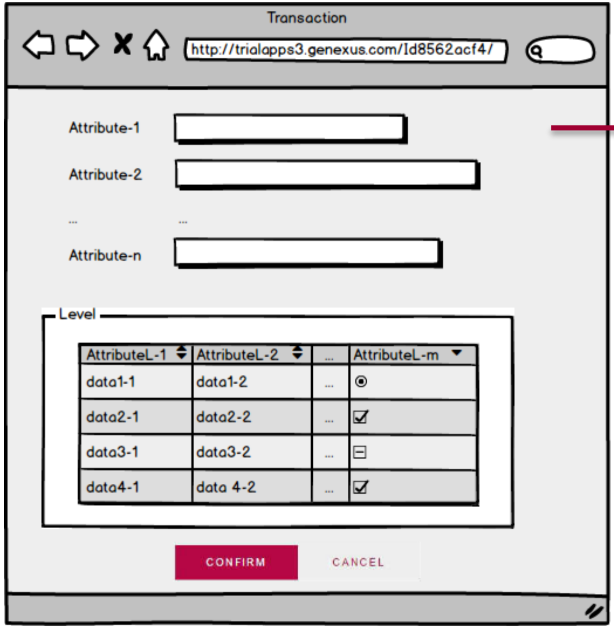
COMMIT

On AfterComplete

Momentos de disparo de regras

PrintCustomer(CustomerId) on AfterValidate; ¿Está certo ou não? ✗

Não está correto porque é chamado ANTES da gravação e a tabela não refletirá as alterações feitas no cliente



Transaction

http://trialapps3.genexus.com/Id8562acf4/

Attribute-1

Attribute-2

...

Attribute-n

Level

AttributeL-1	AttributeL-2	...	AttributeL-m
data1-1	data1-2	...	●
data2-1	data2-2	...	☑
data3-1	data3-2	...	⊟
data4-1	data 4-2	...	☑

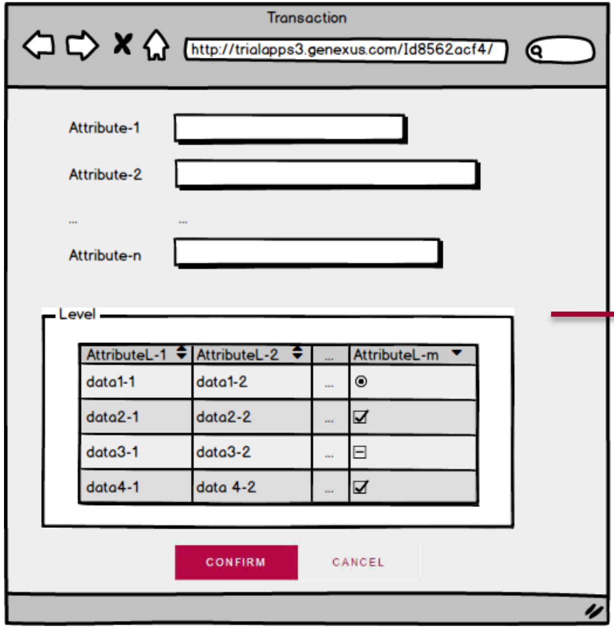
VALIDAÇÃO
On AfterValidate

GRAVAÇÃO
On AfterInsert / On AfterUpdate / On AfterDelete

Momentos de disparo de regras

PrintCustomer(CustomerId) on AfterInsert, AfterUpdate; ¿Está certo ou não?

✓ Está correto!



AttributeL-1	AttributeL-2	...	AttributeL-m
data1-1	data1-2	...	◉
data2-1	data2-2	...	☑
data3-1	data3-2	...	⊟
data4-1	data 4-2	...	☑

VALIDAÇÃO
On Aftervalidate

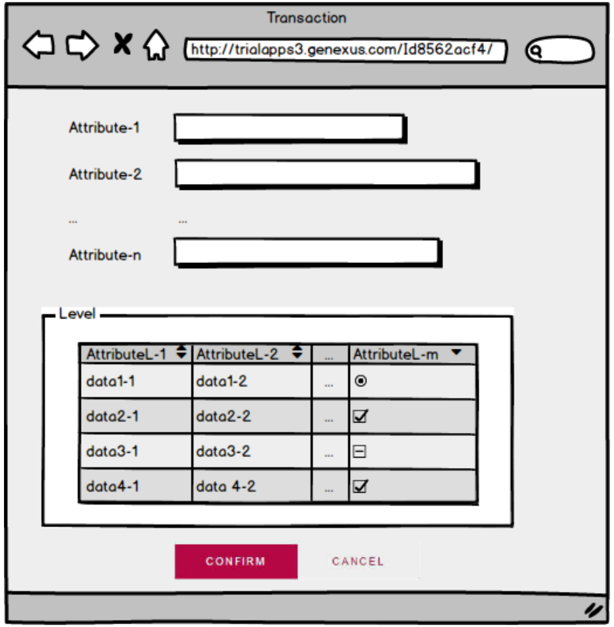
GRAVAÇÃO
On AfterInsert / On AfterUpdate / On AfterDelete

Momentos de disparo de regras

PrintCustomer(CustomerId) on AfterDelete;

¿Está certo ou não? ✗

Não está correto porque é chamado APÓS a exclusão e o cliente não será encontrado com esse ID na tabela



Transaction

http://trialapps3.genexus.com/Id8562acf4/

Attribute-1	
Attribute-2	
...	...
Attribute-n	

Level

AttributeL-1	AttributeL-2	...	AttributeL-m
data1-1	data1-2	...	⦿
data2-1	data2-2	...	☑
data3-1	data3-2	...	⊟
data4-1	data 4-2	...	☑

VALIDAÇÃO

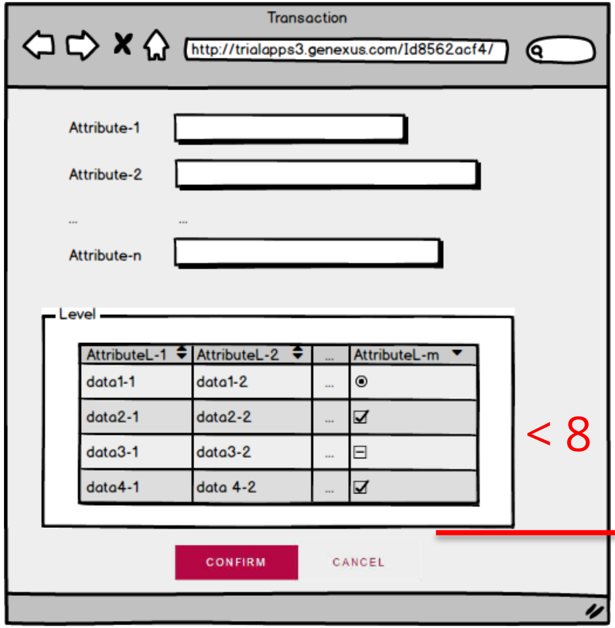
On Aftervalidate

GRAVAÇÃO

On AfterInsert / On AfterUpdate / On AfterDelete

Momentos de disparo de regras

```
Error('The seat quantity should be equal or greather than 8') if FlightCapacity<8
    on AfterLevel
    Level FlightSeatChar;
```



< 8

Em transações de dois niveis:

On BeforeValidate

VALIDAÇÃO do cabeçalho

On AfterValidate

On BeforeInsert/BeforeUpdate/BeforeDelete

GRAVAÇÃO do cabeçalho

On AfterInsert/AfterUpdate/AfterDelete

Para cada linha

- On BeforeValidate
- **VALIDAÇÃO da linha**
- On AfterValidate
- OnBeforeInsert/BeforeUpdate/BeforeDelete
- **GRAVAÇÃO da linha**
- On AfterInsert/AfterUpdate/AfterDelete

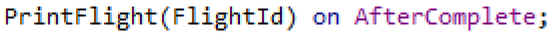
On AferLevel Level Atributo das línhas

On BeforeComplete

COMMIT

On AfterComplete

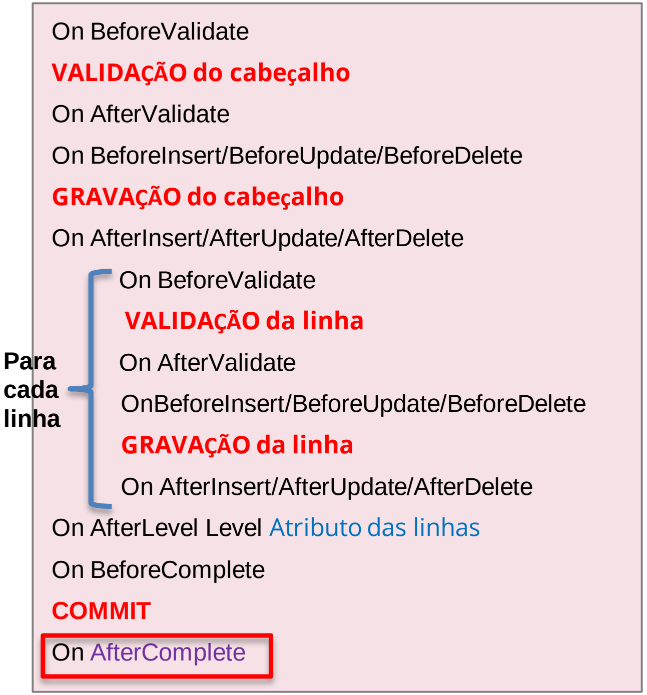
Momentos de disparo de reglas



```
PrintFlight(FlightId) on AfterComplete;
```

✓on AfterComplete: Imediatamente depois de realizar o **Commit** no banco de dados

Em transações de dois níveis:



On BeforeValidate

VALIDAÇÃO do cabeçalho

On AfterValidate

On BeforeInsert/BeforeUpdate/BeforeDelete

GRAVAÇÃO do cabeçalho

On AfterInsert/AfterUpdate/AfterDelete

Para cada linha

- On BeforeValidate
- **VALIDAÇÃO da linha**
- On AfterValidate
- OnBeforeInsert/BeforeUpdate/BeforeDelete
- **GRAVAÇÃO da linha**
- On AfterInsert/AfterUpdate/AfterDelete

On AfterLevel Level Atributo das linhas

On BeforeComplete

COMMIT

On AfterComplete

Exemplos

Determine se está correto ou não:

```
Invoice
{
   InvoiceId*
   InvoiceDate
   -------
   Product
   {
      ProductId*
      ProductName
      ProductPrice
      InvoiceProductQuantity
      --------
   }
}
```

PrintInvoiceDetail(InvoiceId) on **AfterComplete**; ✓

ProductControl(ProductId) **on BeforeInsert**; ✗

ProductControl(ProductId) **on AfterComplete**; ✗

Posso atribuir um valor a um atributo on AfterInsert? ⇨ **NÃO**

Exemplos de atribuição de valores

```
Product
{
   ProductId*
   ProductName
   ProductPrice
}
```

On BeforeValidate

VALIDAÇÃO

On AfterValidate

On BeforeInsert/BeforeUpdate/ BeforeDelete

GRAVAÇÃO

On AfterInsert/AfterUpdate/ AfterDelete

On BeforeComplete

COMMIT

On AfterComplete

ProductPrice = 100 **on BeforeInsert**; ✓

ProductPrice = 100 **on BeforeComplete**; ✗

ProductPrice = 100 **on AfterValidate**; ✓

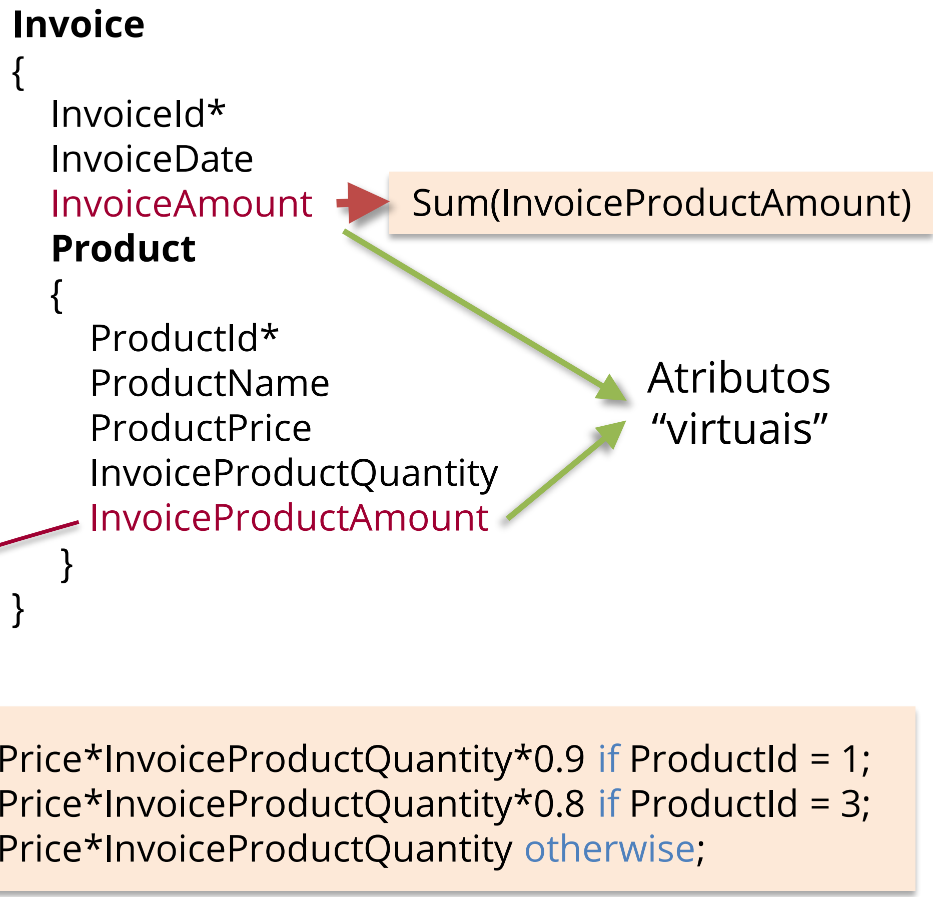
ProductPrice = 100 **on AfterInsert**; ✗

Fórmulas

Fórmulas Globais

- É um cálculo que definimos associado a um atributo
- São acessíveis em toda a KB

```
Product
{
   ProductId*
   ProductName
   ProductPrice
}
```

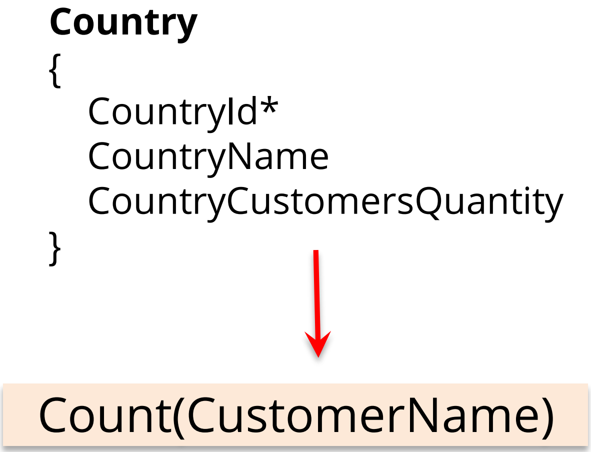


```
Invoice
{
   InvoiceId*
   InvoiceDate
   InvoiceAmount
   Product
   {
      ProductId*
      ProductName
      ProductPrice
      InvoiceProductQuantity
      InvoiceProductAmount
   }
}
```

Sum(InvoiceProductAmount)

Atributos “virtuais”

```
ProductPrice*InvoiceProductQuantity*0.9 if ProductId = 1;
ProductPrice*InvoiceProductQuantity*0.8 if ProductId = 3;
ProductPrice*InvoiceProductQuantity otherwise;
```



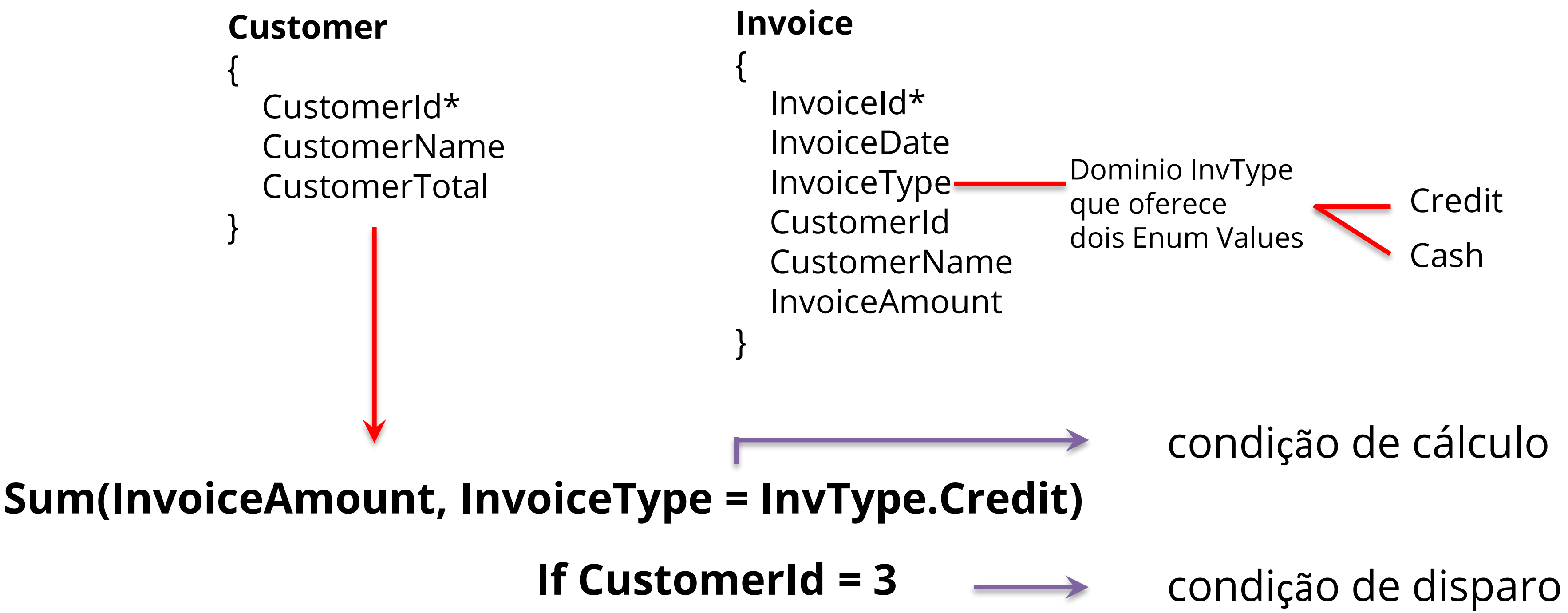
```
Country
{
   CountryId*
   CountryName
   CountryCustomersQuantity
}
```

Count(CustomerName)

```
Customer
{
   CustomerId*
   CustomerName
   CountryId
   CountryName
}
```

Esta fórmula contará clientes por país ou o número total de clientes?

Contará os clientes por cada país já que GeneXus realiza um <u>filtro automático</u> pelo atributo em comum (*CountryId*).



```
Customer
{
    CustomerId*
    CustomerName
    CustomerTotal
}
```

```
Invoice
{
    InvoiceId*
    InvoiceDate
    InvoiceType
    CustomerId
    CustomerName
    InvoiceAmount
}
```

Dominio InvType
que oferece
dois Enum Values

Credit

Cash

Sum(InvoiceAmount, InvoiceType = InvType.Credit)

If CustomerId = 3

condição de cálculo

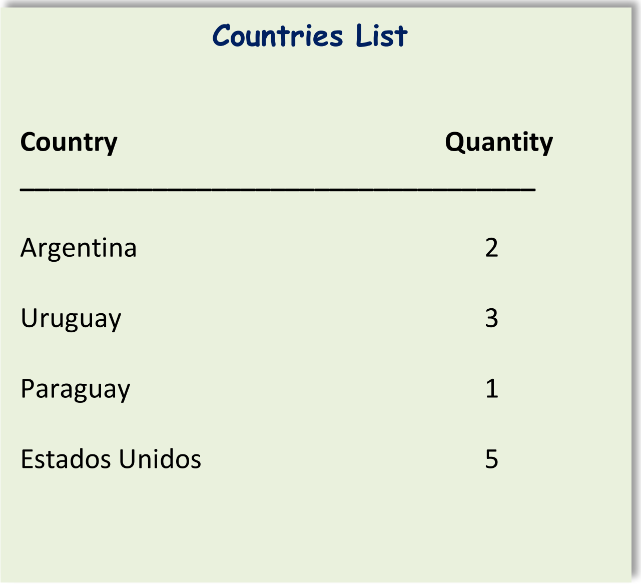
condição de disparo

Fórmulas Inline ⇨

- Estão escritas no código

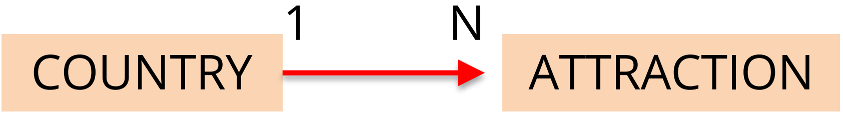
- São acessíveis apenas no objeto em que foram definidas

Requisito: Nos pedem um relatório de países com a quantidade de atrações de cada pais



Countries List

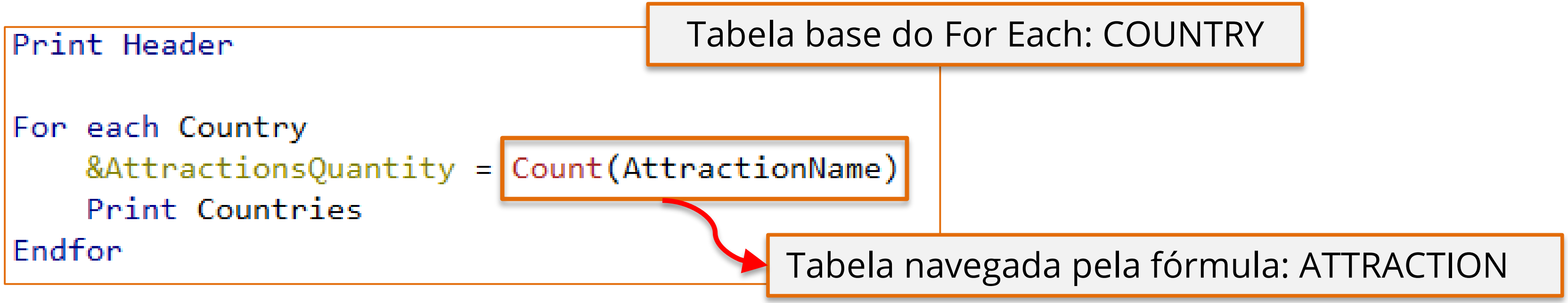
Country	Quantity
Argentina	2
Uruguay	3
Paraguay	1
Estados Unidos	5



```
Country
{
   CountryId*
   CountryName
}
```

```
Attraction
{
   AttractionId*
   AttractionName
   CountryId
   CountryName
}
```

Formula inline no código de um objeto Procedimento



```
Print Header

For each Country
    &AttractionsQuantity = Count(AttractionName)
    Print Countries
Endfor
```

Tabela base do For Each: COUNTRY

Tabela navegada pela fórmula: ATTRACTION

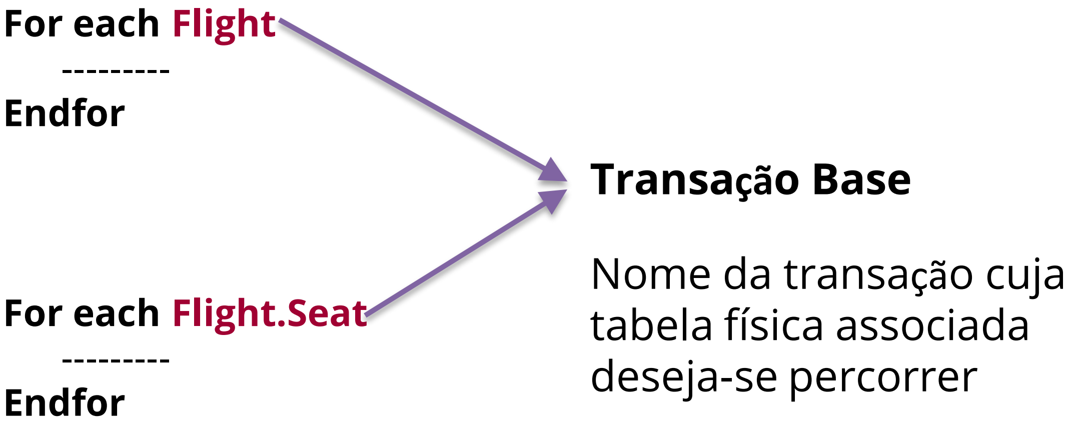
Esta fórmula contará as atrações por país ou o número total de atrações?

Contará as atrações por cada país já que é feito um <u>filtro automático</u> pelo atributo em comum *CountryId* (ambas tabelas estão relacionadas).

For each

Transação Base

```
Flight
{
   FlightId*
   FlightDate
   -----
   Seat
   {
     FlightSeatId*
     FlightSeatChar
   }
}
```



```
For each Flight
     ---------
Endfor
```

```
For each Flight.Seat
     ---------
Endfor
```

Transação Base

Nome da transação cuja tabela física associada deseja-se percorrer

Ordem

```
Customer
{
   CustomerId*
   CustomerName
   ------
}
```

Requisito: nos pedem um relatório de todos os clientes listados em ordem alfabética.

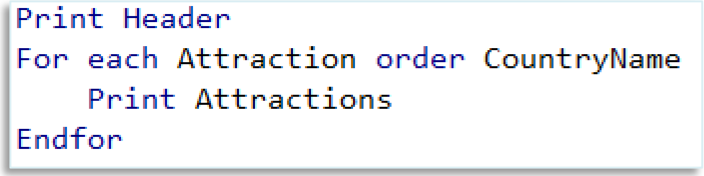
```
For each Customer order CustomerName
   -------
Endfor
```

Requisito : nos pedem um relatório de todos os clientes ordenados de forma decrescente pelo nome.

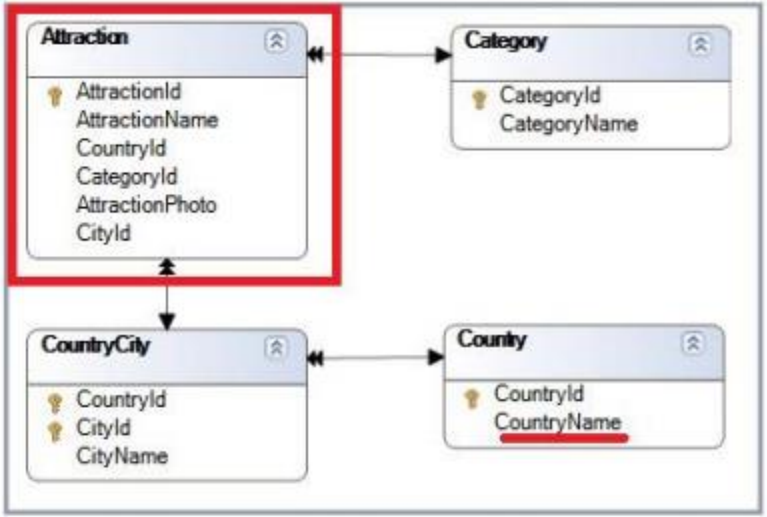
```
For each Customer order (CustomerName)
   -------
Endfor
```

Ordem

GeneXus permite ordenar pelo valor de um atributo que não está na tabela que estamos percorrendo, mas em sua tabela estendida.



```
Print Header
For each Attraction order CountryName
    Print Attractions
Endfor
```



Attraction
AttractionId
AttractionName
CountryId
CategoryId
AttractionPhoto
CityId

Category
CategoryId
CategoryName

CountryCity
CountryId
CityId
CityName

Country
CountryId
CountryName

Filtros com Where

```
Flight
{
   FlightId*
   FlightDate
   -----
   Seat
   {
     FlightSeatId*
     FlightSeatChar
   }
}
```

```
For each Flight
     Where FlightDate = Today()
          ---------
Endfor
```

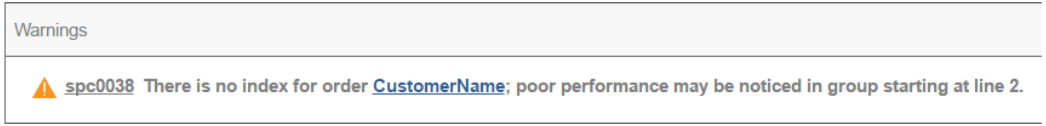
```
For each Flight.Seat
     Where FlightId = 1
          --------
Endfor
```

Filtros + Ordem

```
Customer
{
    CustomerId*
    CustomerName
    CustomerAddress
}
```

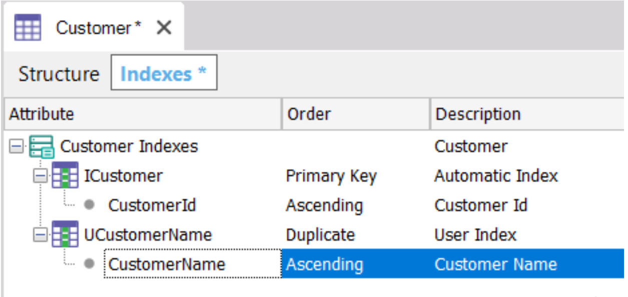
Index

```
For each Customer order CustomerName
    Where CustomerName >= &NameFrom
            ---------
Endfor
```



Warnings

⚠ spc0038 There is no index for order CustomerName; poor performance may be noticed in group starting at line 2.



Attribute	Order	Description
Customer Indexes		Customer
ICustomer	Primary Key	Automatic Index
CustomerId	Ascending	Customer Id
UCustomerName	Duplicate	User Index
CustomerName	Ascending	Customer Name

¡Consulta otimizada! ✓

Sintaxe do For each

```
For each BaseTransaction
     skip expression1 count expression2
     order att_1, att_2, … , att_n [when condition]
     order att_1, att_2, … , att_n [when condition]
     unique att_1, att_2, … , att_n
     using DataSelector( parm_1, parm_2, … , parm_n)
     where condition [when condition]
     where condition [when condition]
     where att IN DataSelector( parm_1, parm_2, … , parm_n)

        main code

When none
        …
Endfor
```

For each Aninhados + Tabela base diferente + Tabelas NÃO relacionadas = PRODUTO CARTESIANO

```
Country
{
   CountryId*
   CountryName
}


Room
{
   RoomId*
   RoomName
}
```

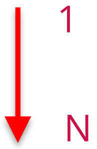
Produto Cartesiano

```
For each Country
        Print Country

    For each Room
          Print Room
    Endfor

Endfor
```

```
1 - Brasil
                    RoomA
                    RoomB
                    RoomC
2 -Uruguay
                    RoomA
                    RoomB
                    RoomC
3 - Argentina
                    RoomA
                    RoomB
                    RoomC
4 - Estados Unidos
                    RoomA
                    RoomB
                    RoomC
```

For each Aninhados + Tabela base diferente + Tabelas relacionadas = JOIN

```
Country
{
   CountryId*
   CountryName
}
```



```
Customer
{
   CustomerId*
   CustomerName
   CountryId
   CountryName
}
```

JOIN

```
For each Country
        Print Country

      For each Customer
          Print Customer
      Endfor

Endfor
```

```
1 - Brasil
                LUIS
                JORGE

2 - Uruguai


3 - Argentina


4 - Estados Unidos
                ANA
```

For each Aninhados + Mesma Tabela base + Tabelas relacionadas = CORTE DE CONTROLE

```
Country
{
   CountryId*
   CountryName
}


Customer
{
   CustomerId*
   CustomerName
   CountryId
   CountryName
}
```

CORTE DE CONTROLE

```
For each Customer order CountryId
      Print Country

   For each Customer
        Print Customer
   Endfor

Endfor
```

```
1 - Brasil
                  LUIS
                  JORGE

4 - Estados Unidos
                  ANA
```

Resumo

Produto Cartesiano

```
For each Country
    Print Country

  For each Room
      Print Room
  Endfor

Endfor
```

Tabelas diferentes, sem relacionamento

Join

```
For each Country
    Print Country

  For each Customer
      Print Customer
  Endfor

Endfor
```

Tabelas diferentes, com relacionamento

Corte de Controle

```
For each Customer order CountryId
    Print Country

  For each Customer
      Print Customer
  Endfor

Endfor
```

Mesma tabela, Agrupando com order

Resumo

Produto Cartesiano

- 1 - Brasil
  - RoomA
  - RoomB
  - RoomC
- 2 -Uruguay
  - RoomA
  - RoomB
  - RoomC
- 3 - Argentina
  - RoomA
  - RoomB
  - RoomC

“Não existe relacionamento entre ambas entidades, mostra todas as possibilidades para cada país”

Join

- 1 - Brasil
  - LUIS
  - JORGE
- 2 - Uruguai
- 3 - Argentina
- 4 - Estados Unidos
  - ANA

“Todos os paises e seus clientes, independentemente que tenha clientes ou não”

Corte de Controle

- 1 - Brasil
  - LUIS
  - JORGE
- 4 - Estados Unidos
  - ANA

“Somente aqueles países que tenhan clientes”

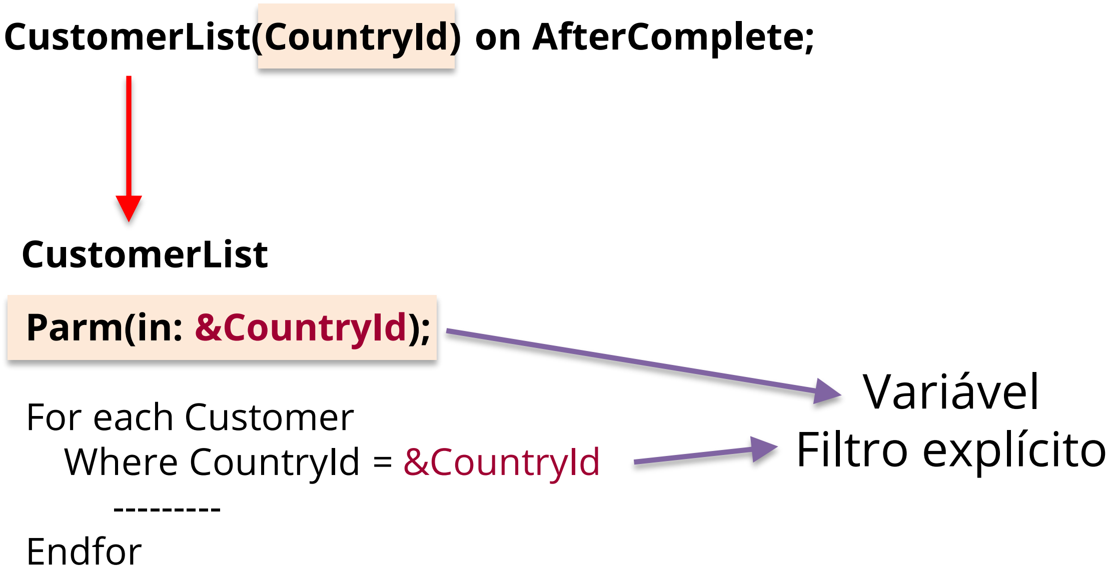
Comunicação entre objetos

Exemplo: passagem de parâmetros

```
Country
{
   CountryId*
   CountryName
}
```

```
Customer
{
   CustomerId*
   CustomerName
   CountryId
   CountryName
}
```

Por exemplo, nas Regras da trn Country:



CustomerList(CountryId) on AfterComplete;

CustomerList

```
Parm(in: &CountryId);

For each Customer
   Where CountryId = &CountryId
      ---------
Endfor
```

Variável

Filtro explícito

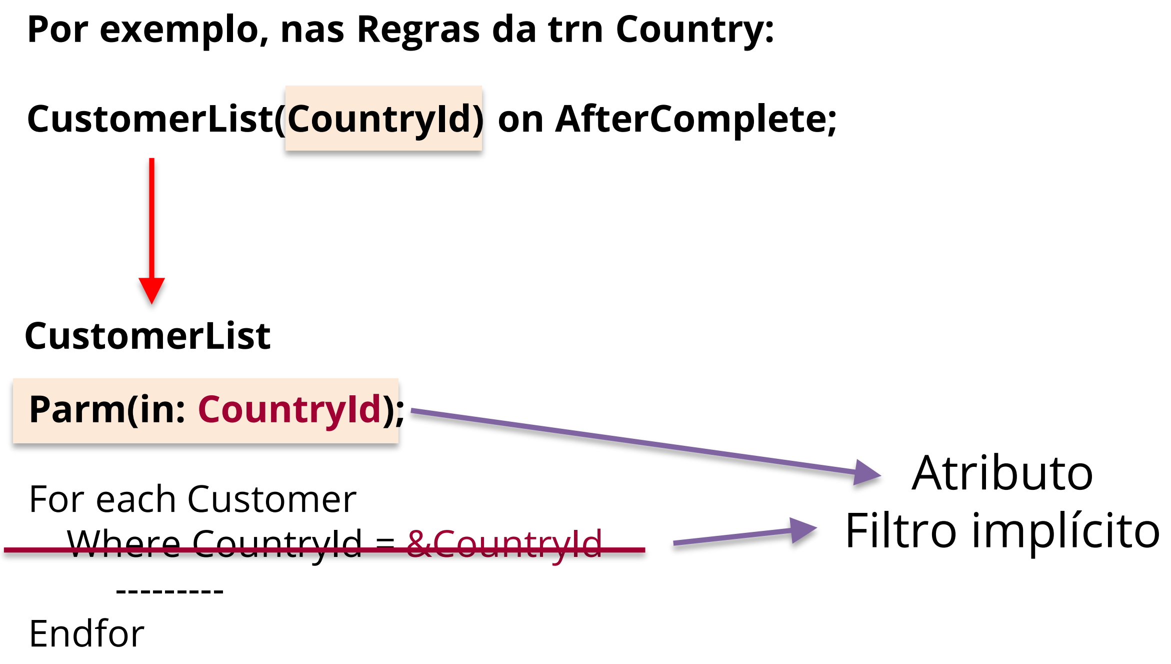
Exemplo: passagem de parâmetros

Country

```
{
   CountryId*
   CountryName
}
```

Customer

```
{
   CustomerId*
   CustomerName
   CountryId
   CountryName
}
```



Por exemplo, nas Regras da trn Country:

CustomerList(CountryId) on AfterComplete;

CustomerList

Parm(in: CountryId);

For each Customer
~~Where CountryId = &CountryId~~

Endfor

Atributo
Filtro implícito

Exemplo: retorno de um valor

```
Country
{
   CountryId*
   CountryName
}
```

```
Customer
{
   CustomerId*
   CustomerName
   CountryId
   CountryName
}
```

Por exemplo, nas Regras da trn Customer:

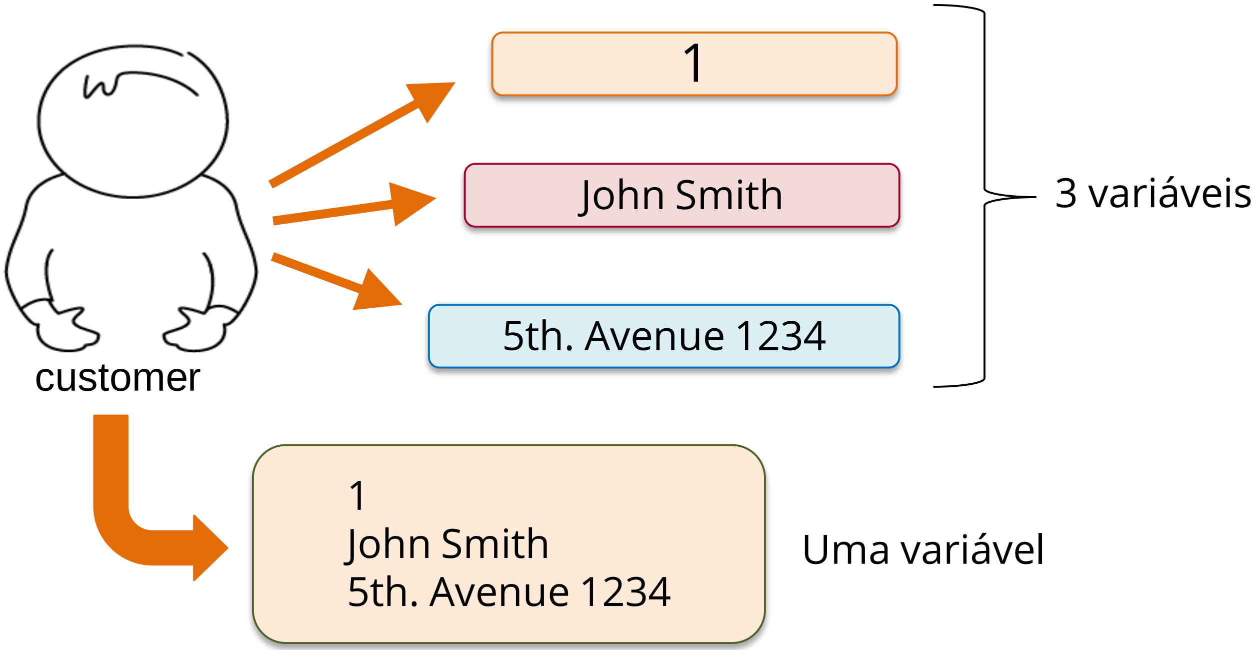
```
&Control = CustomerControl(CustomerId);
```

CustomerControl

```
Parm(in: &CustomerId, out: &Control);

For each Customer
   Where CustomerId = &CustomerId
      ---------
      &Control = True
Endfor
```

Tipos de Dados Compostos

Conceito



customer

1

John Smith

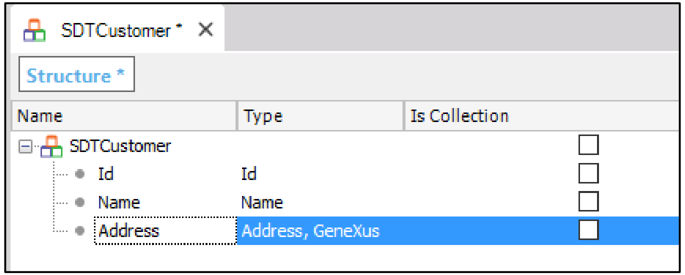
5th. Avenue 1234

3 variáveis

1
John Smith
5th. Avenue 1234

Uma variável

Definição

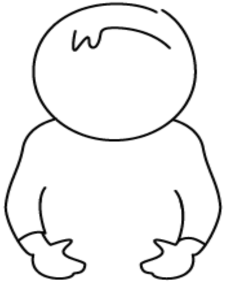


SDTCustomer * ✕

Structure *

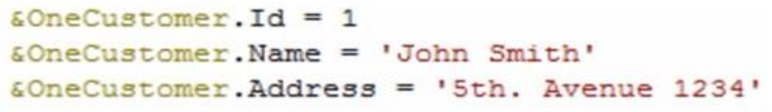
Name	Type	Is Collection
SDTCustomer		□
Id	Id	□
Name	Name	□
Address	Address, GeneXus	□

Objeto Structured Data Type



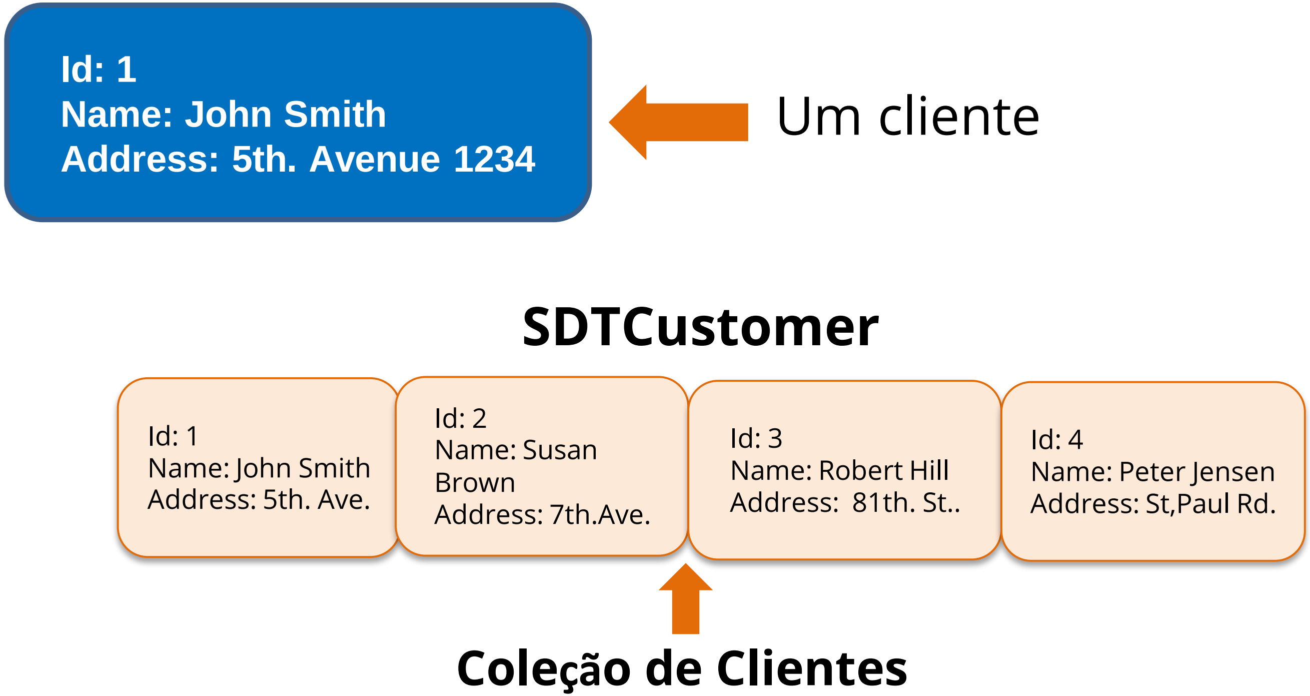
customer

&OneCustomer: SDTCustomer



```
&OneCustomer.Id = 1
&OneCustomer.Name = 'John Smith'
&OneCustomer.Address = '5th. Avenue 1234'
```

Data Providers



Id: 1
Name: John Smith
Address: 5th. Avenue 1234

Um cliente

SDTCustomer

Id: 1
Name: John Smith
Address: 5th. Ave.

Id: 2
Name: Susan Brown
Address: 7th.Ave.

Id: 3
Name: Robert Hill
Address: 81th. St..

Id: 4
Name: Peter Jensen
Address: St,Paul Rd.

Coleção de Clientes

Exemplo: Ranking de quantidade de atrações por país

País	Qtde atrações
BRASIL	4
ARGENTINA	3
URUGUAI	2
CHILE	1

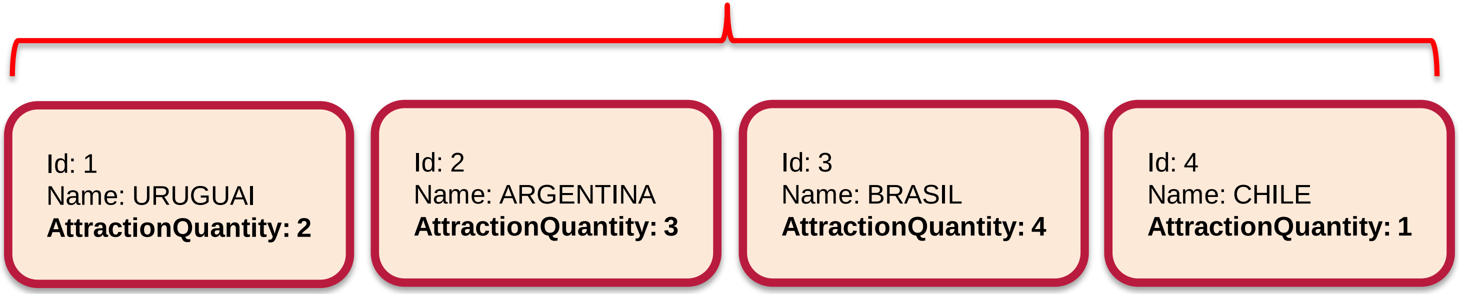
….

…

Exemplo: Ranking de quantidade de atrações por país

SDTCountries

Uma coleção de países



Id: 1
Name: URUGUAI
AttractionQuantity: 2

Id: 2
Name: ARGENTINA
AttractionQuantity: 3

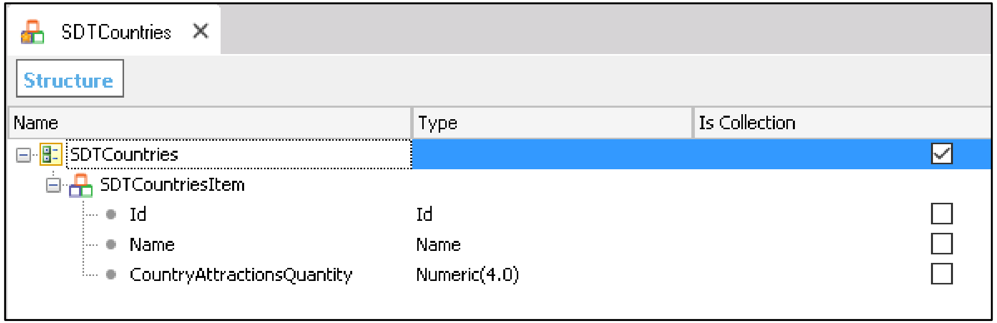
Id: 3
Name: BRASIL
AttractionQuantity: 4

Id: 4
Name: CHILE
AttractionQuantity: 1

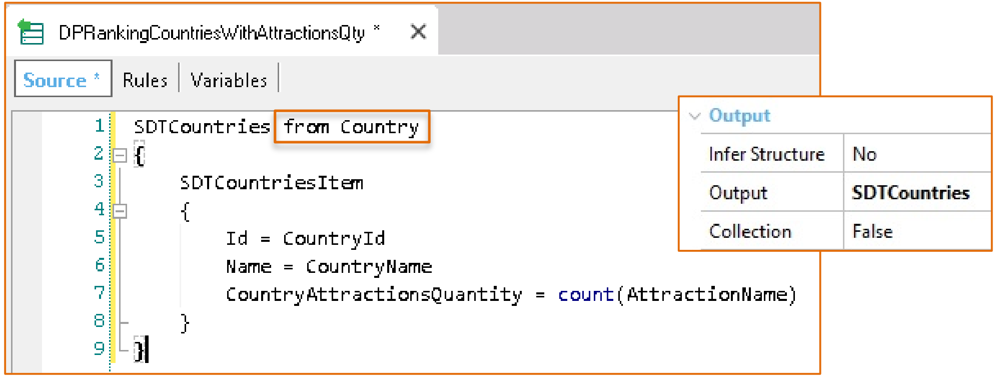
Exemplo: Ranking de quantidade de atrações por país

```
Country
{
   CountryId*
   CountryName
}
```

```
Attraction
{
   AttractionId*
   AttractionName
   CountryId
   CountryName
}
```



Name	Type	Is Collection
SDTCountries		☑
SDTCountriesItem		
Id	Id	□
Name	Name	□
CountryAttractionsQuantity	Numeric(4.0)	□

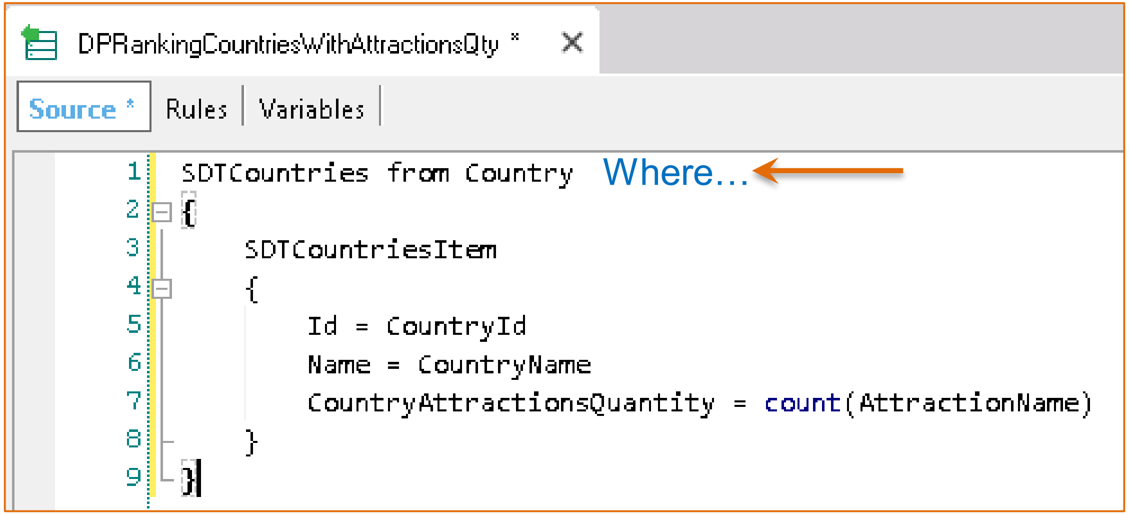


```
SDTCountries from Country
{
    SDTCountriesItem
    {
        Id = CountryId
        Name = CountryName
        CountryAttractionsQuantity = count(AttractionName)
    }
}
```

Output

Infer Structure	No
Output	**SDTCountries**
Collection	False

Exemplo: Ranking de quantidade de atrações por país



```
SDTCountries from Country
{
    SDTCountriesItem
    {
        Id = CountryId
        Name = CountryName
        CountryAttractionsQuantity = count(AttractionName)
    }
}
```

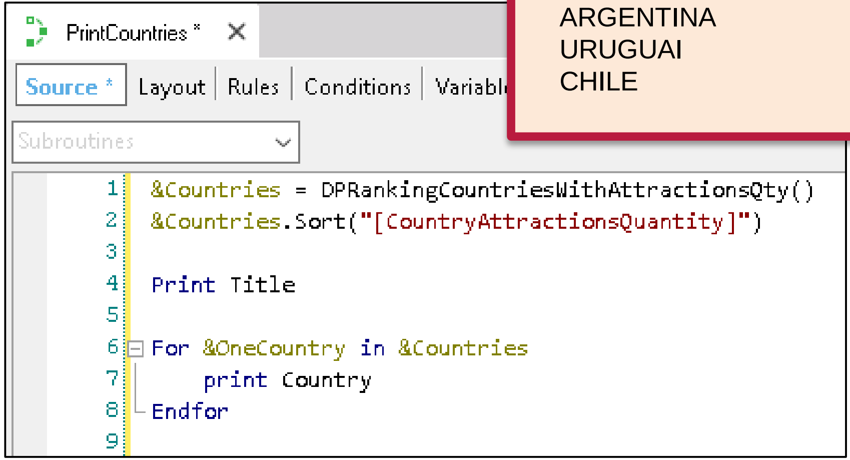
Where…

Exemplo: Ranking de quantidade de atrações por país

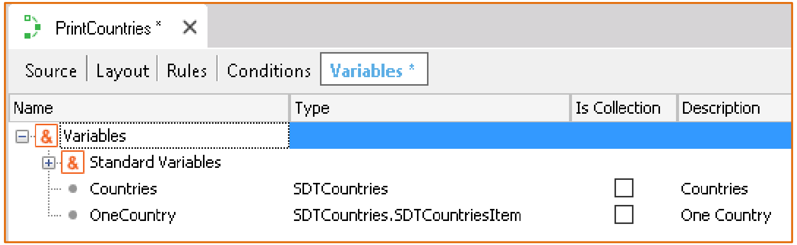
País	Qtde Atrações
BRASIL	4
ARGENTINA	3
URUGUAI	2
CHILE	1

```
Country
{
  CountryId*
  CountryName
}
```

```
Attraction
{
  AttractionId*
  AttractionName
  CountryId
  CountryName
}
```



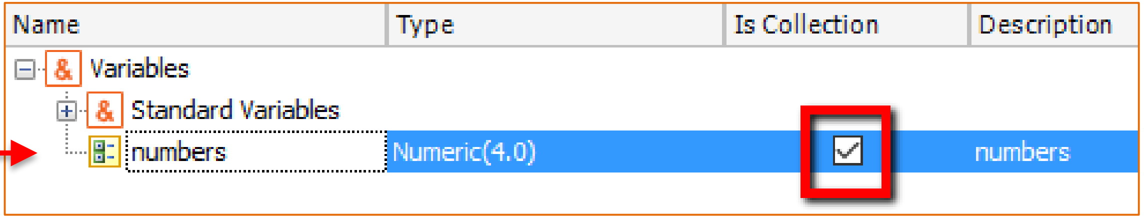
```
&Countries = DPRankingCountriesWithAttractionsQty()
&Countries.Sort("[CountryAttractionsQuantity]")

Print Title

For &OneCountry in &Countries
    print Country
Endfor

```



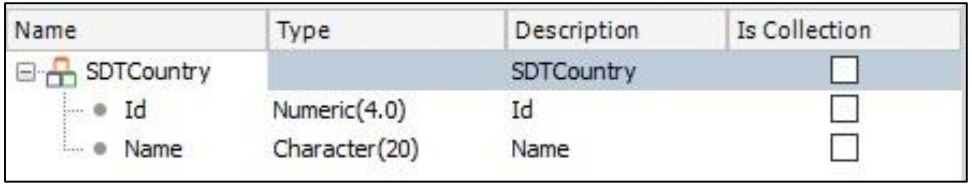
Name	Type	Is Collection	Description
Variables			
Standard Variables			
Countries	SDTCountries	□	Countries
OneCountry	SDTCountries.SDTCountriesItem	□	One Country

Variáveis Coleção

&numbers

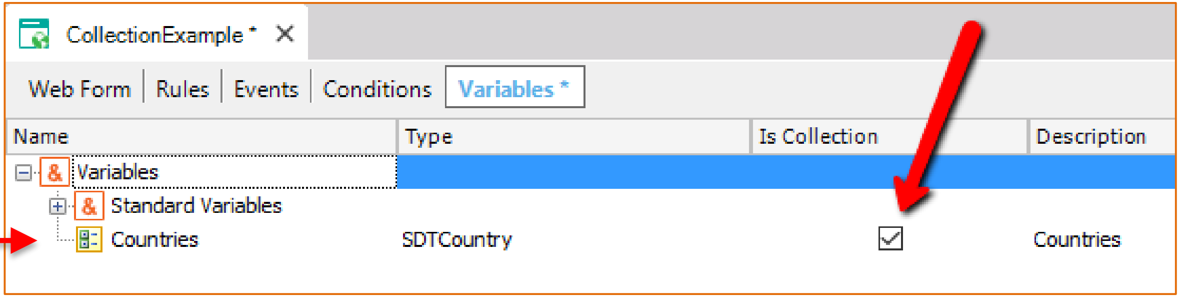


Name	Type	Is Collection	Description
Variables			
Standard Variables			
numbers	Numeric(4.0)	☑	numbers



Name	Type	Description	Is Collection
SDTCountry		SDTCountry	□
Id	Numeric(4.0)	Id	□
Name	Character(20)	Name	□

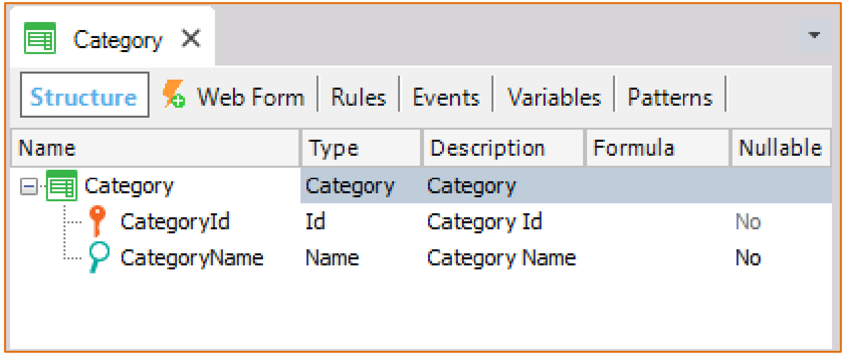
&Countries



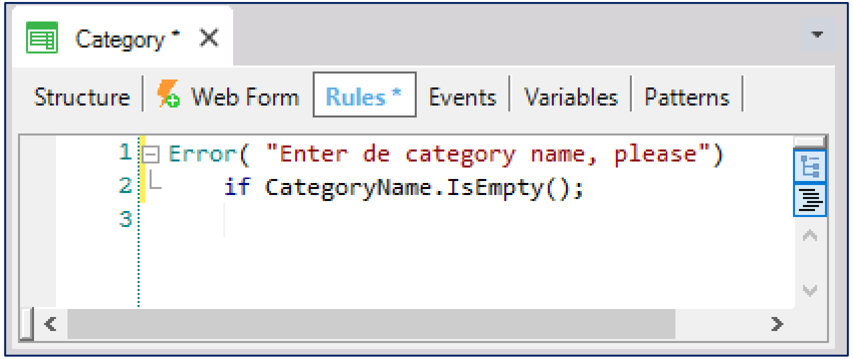
Name	Type	Is Collection	Description
Variables			
Standard Variables			
Countries	SDTCountry	☑	Countries

Business Components

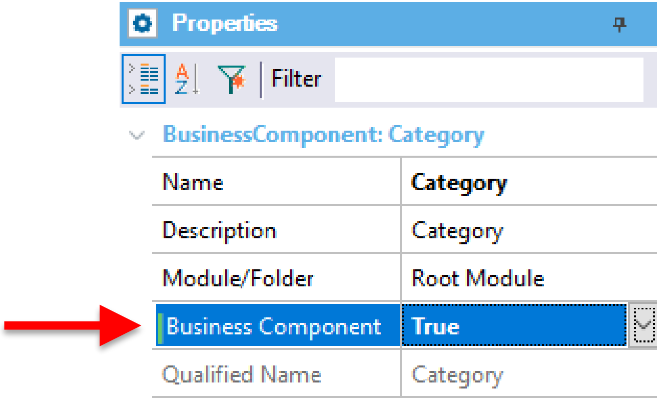
Conceito: tipo de dados especial baseado em uma transação



Name	Type	Description	Formula	Nullable
Category	Category	Category		
CategoryId	Id	Category Id		No
CategoryName	Name	Category Name		No



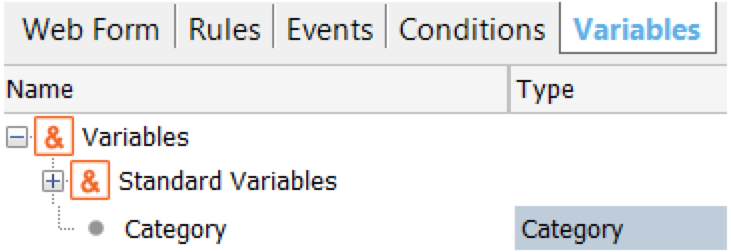
```
Error( "Enter de category name, please")
    if CategoryName.IsEmpty();
```



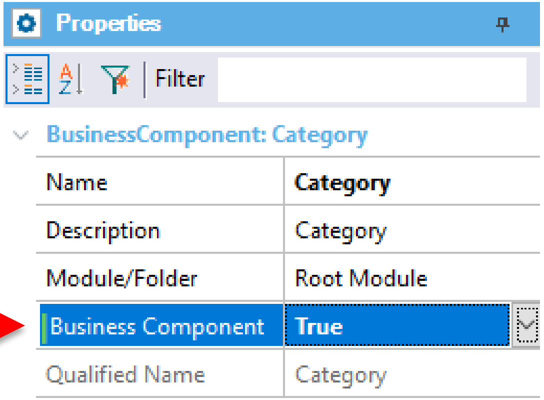
BusinessComponent: Category

Name	**Category**
Description	Category
Module/Folder	Root Module
Business Component	True
Qualified Name	Category

Conceito: tipo de dados especial baseado em uma transação

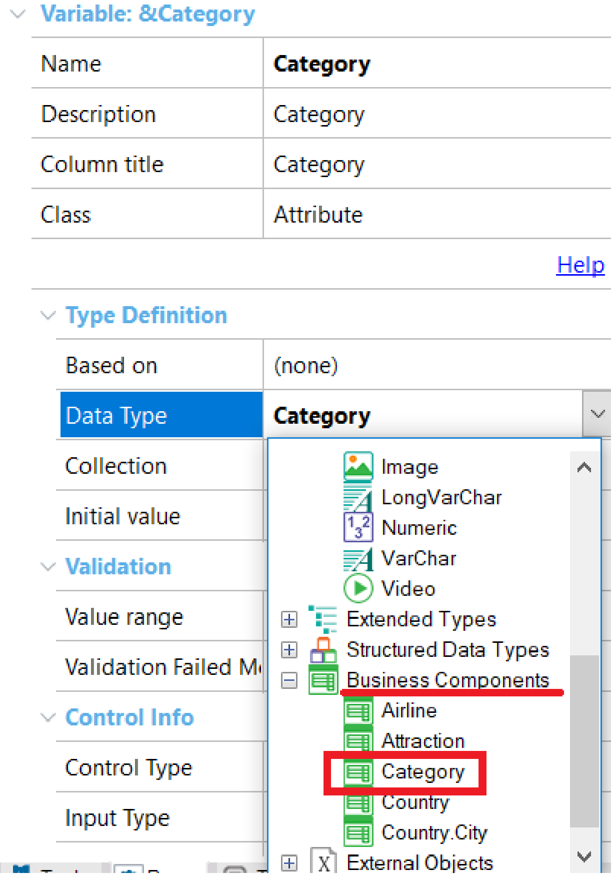


Name	Type
Variables	
Standard Variables	
Category	Category



BusinessComponent: Category

Name	Category
Description	Category
Module/Folder	Root Module
Business Component	True
Qualified Name	Category



Variable: &Category

Name	Category
Description	Category
Column title	Category
Class	Attribute

Type Definition

Based on	(none)
Data Type	Category
Collection	
Initial value	

Validation

Value range	
Validation Failed M	

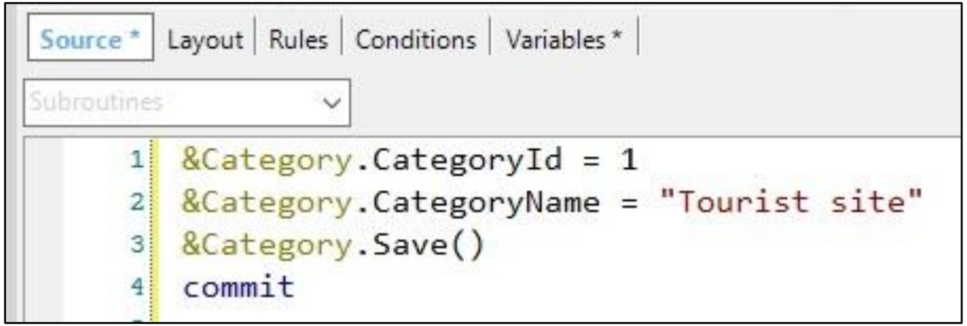
Control Info

Control Type	
Input Type	

Exemplos: inserção e alteração

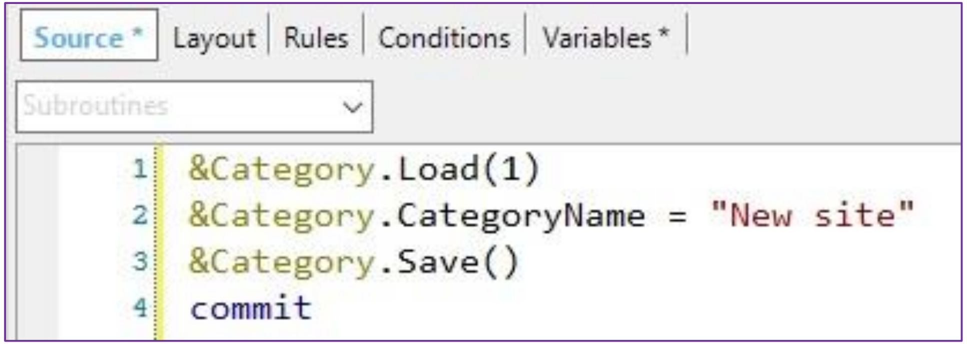
Category
{
    CategoryId*
    CategoryName
}

Insert



```
&Category.CategoryId = 1
&Category.CategoryName = "Tourist site"
&Category.Save()
commit
```

Update

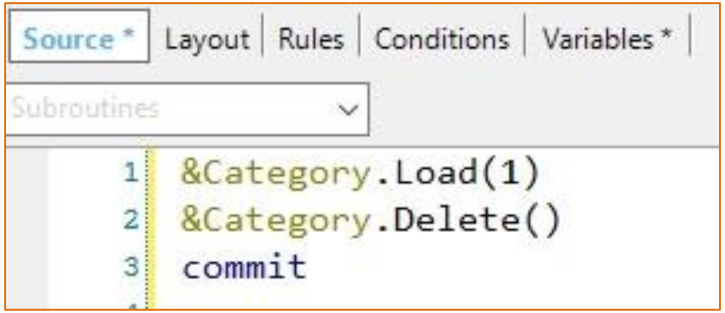


```
&Category.Load(1)
&Category.CategoryName = "New site"
&Category.Save()
commit
```

Exemplo: exclusão

Category
{
  CategoryId*
  CategoryName
}

Delete

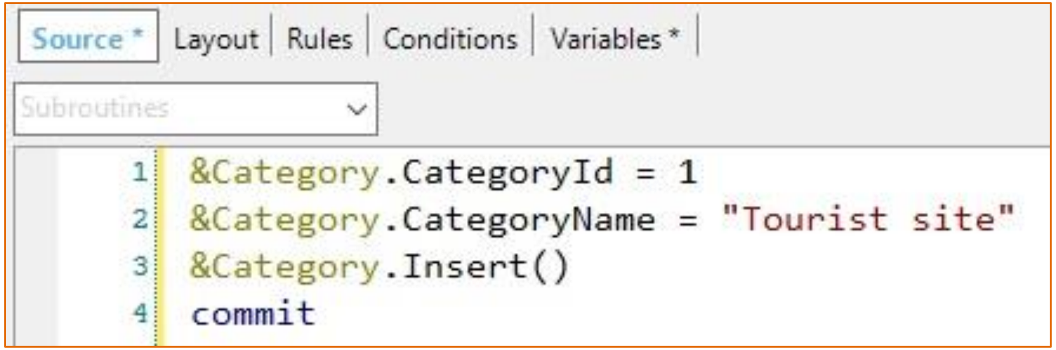


```
&Category.Load(1)
&Category.Delete()
commit
```

Métodos Insert e Update

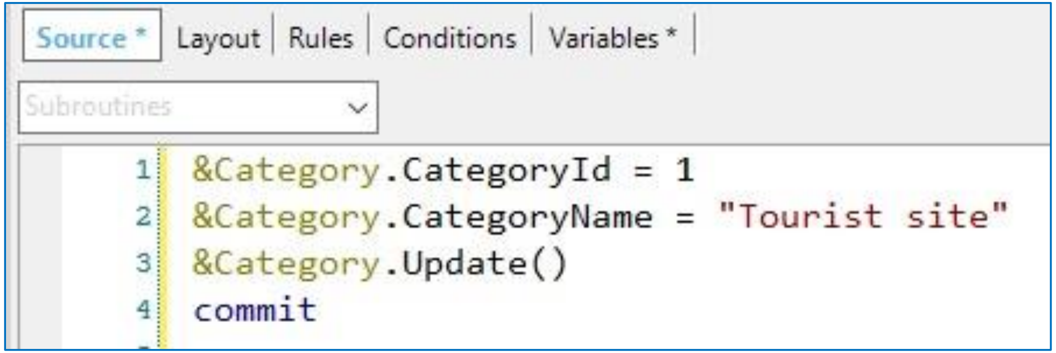
Category
{
  CategoryId*
  CategoryName
}

Insert



```
&Category.CategoryId = 1
&Category.CategoryName = "Tourist site"
&Category.Insert()
commit
```

Update

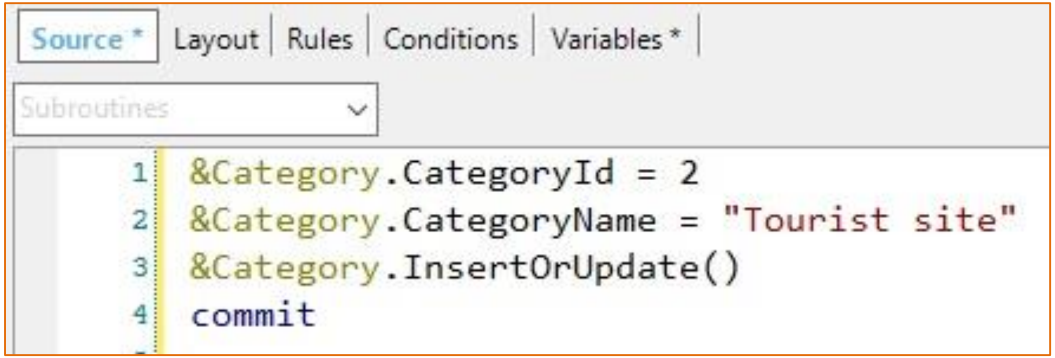


```
&Category.CategoryId = 1
&Category.CategoryName = "Tourist site"
&Category.Update()
commit
```

Método InsertOrUpdate

Category
{
  CategoryId*
  CategoryName
}

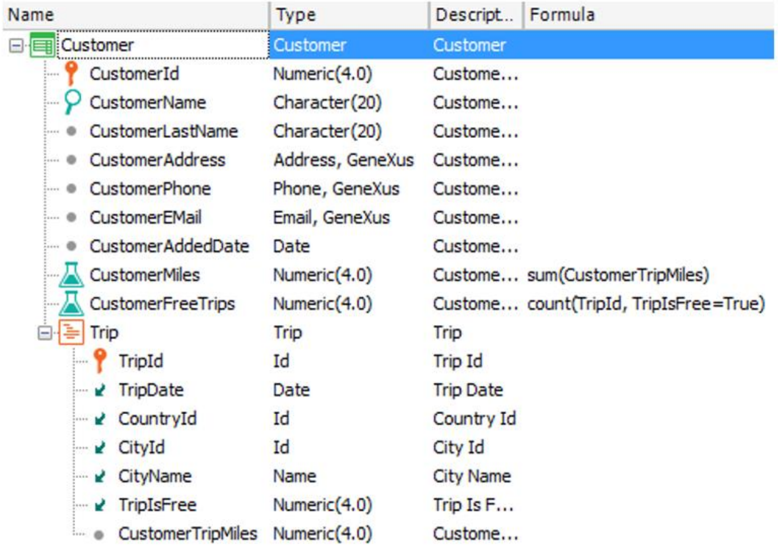
InsertOrUpdate



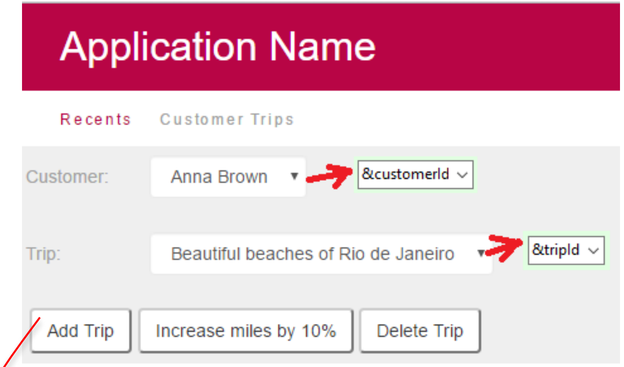
```
&Category.CategoryId = 2
&Category.CategoryName = "Tourist site"
&Category.InsertOrUpdate()
commit
```

Insert / Update em transação de dois níveis

Insert



Name	Type	Descript...	Formula
Customer	Customer	Customer	
CustomerId	Numeric(4.0)	Custome...	
CustomerName	Character(20)	Custome...	
CustomerLastName	Character(20)	Custome...	
CustomerAddress	Address, GeneXus	Custome...	
CustomerPhone	Phone, GeneXus	Custome...	
CustomerEMail	Email, GeneXus	Custome...	
CustomerAddedDate	Date	Custome...	
CustomerMiles	Numeric(4.0)	Custome...	sum(CustomerTripMiles)
CustomerFreeTrips	Numeric(4.0)	Custome...	count(TripId, TripIsFree=True)
Trip	Trip	Trip	
TripId	Id	Trip Id	
TripDate	Date	Trip Date	
CountryId	Id	Country Id	
CityId	Id	City Id	
CityName	Name	City Name	
TripIsFree	Numeric(4.0)	Trip Is F...	
CustomerTripMiles	Numeric(4.0)	Custome...	

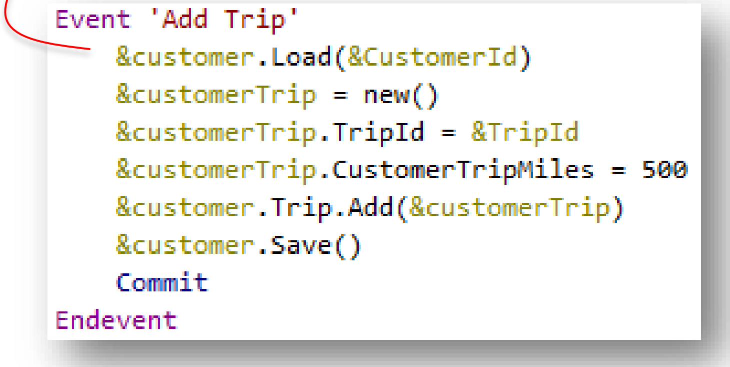


Application Name

Recents   Customer Trips

Customer: Anna Brown ▾ &customerId

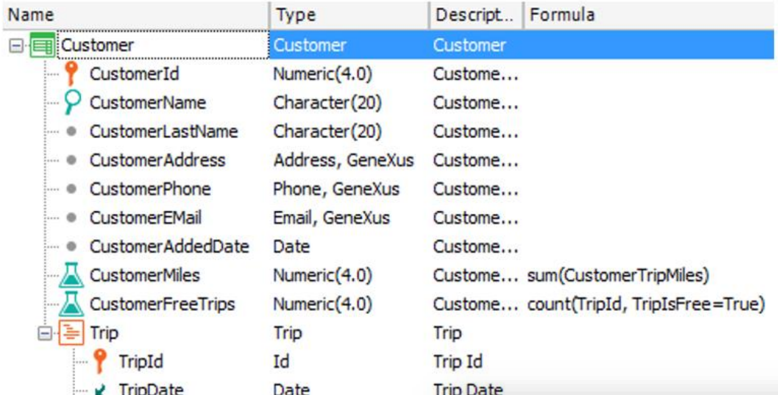
Trip: Beautiful beaches of Rio de Janeiro ▾ &tripId



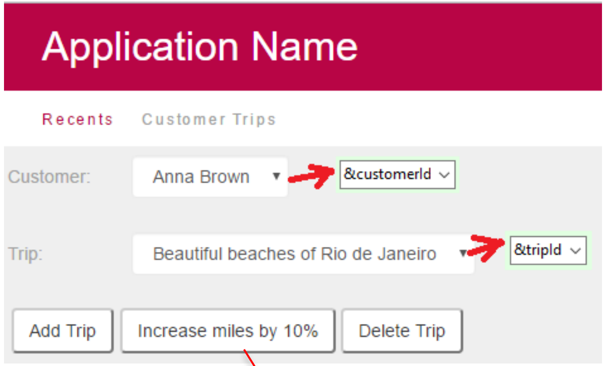
```
Event 'Add Trip'
    &customer.Load(&CustomerId)
    &customerTrip = new()
    &customerTrip.TripId = &TripId
    &customerTrip.CustomerTripMiles = 500
    &customer.Trip.Add(&customerTrip)
    &customer.Save()
    Commit
Endevent
```

Insert / Update em transação de dois níveis

Insert



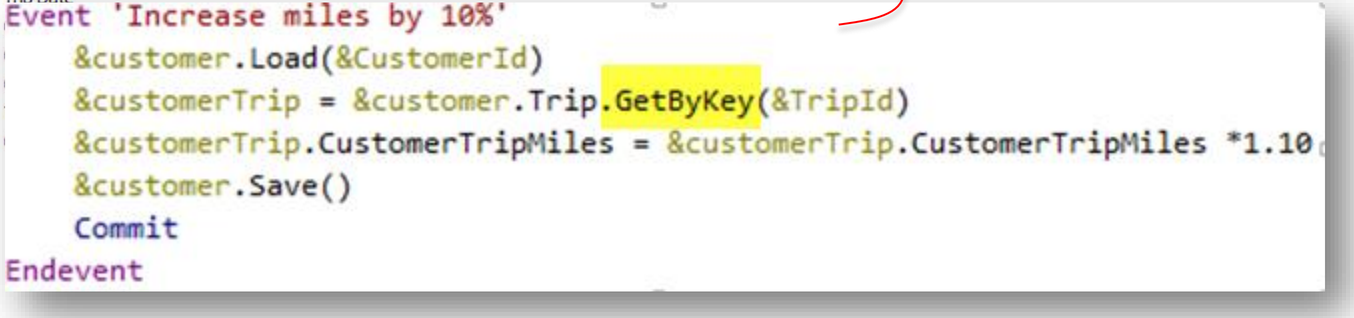
Name	Type	Descript...	Formula
Customer	Customer	Customer	
CustomerId	Numeric(4.0)	Custome...	
CustomerName	Character(20)	Custome...	
CustomerLastName	Character(20)	Custome...	
CustomerAddress	Address, GeneXus	Custome...	
CustomerPhone	Phone, GeneXus	Custome...	
CustomerEMail	Email, GeneXus	Custome...	
CustomerAddedDate	Date	Custome...	
CustomerMiles	Numeric(4.0)	Custome...	sum(CustomerTripMiles)
CustomerFreeTrips	Numeric(4.0)	Custome...	count(TripId, TripIsFree=True)
Trip	Trip	Trip	
TripId	Id	Trip Id	
TripDate	Date	Trip Date	
CountryId	Id		
CityId	Id		
CityName	Name		
TripIsFree	Numeric(4.0)		
CustomerTripMiles	Numeric(4.0)		



Application Name

Customer: Anna Brown &customerId

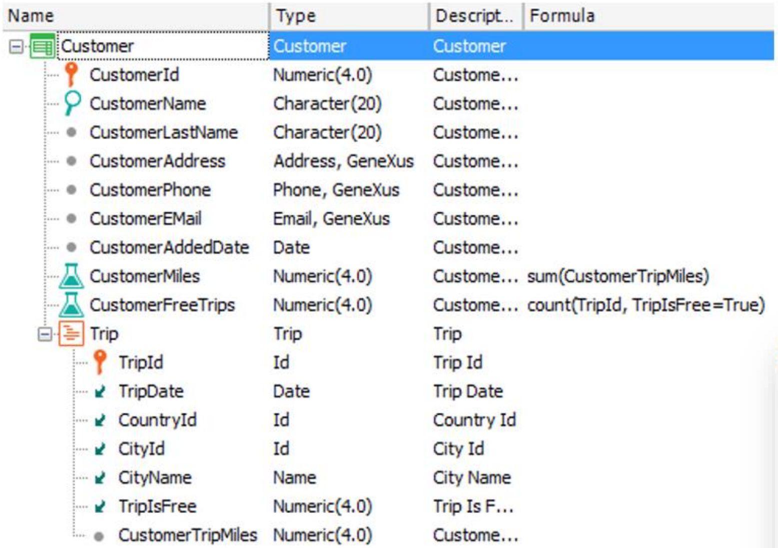
Trip: Beautiful beaches of Rio de Janeiro &tripId



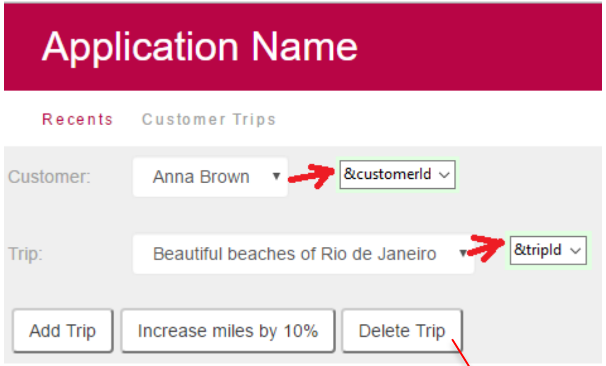
```
Event 'Increase miles by 10%'
    &customer.Load(&CustomerId)
    &customerTrip = &customer.Trip.GetByKey(&TripId)
    &customerTrip.CustomerTripMiles = &customerTrip.CustomerTripMiles *1.10
    &customer.Save()
    Commit
Endevent
```

Insert / Update em transação de dois níveis

Insert



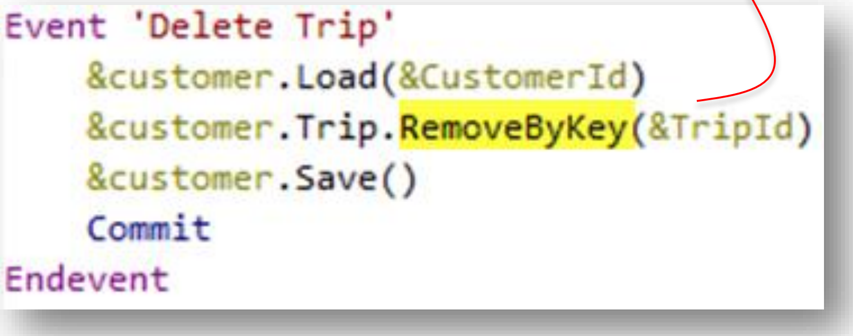
Name	Type	Descript...	Formula
Customer	Customer	Customer	
CustomerId	Numeric(4.0)	Custome…	
CustomerName	Character(20)	Custome…	
CustomerLastName	Character(20)	Custome…	
CustomerAddress	Address, GeneXus	Custome…	
CustomerPhone	Phone, GeneXus	Custome…	
CustomerEMail	Email, GeneXus	Custome…	
CustomerAddedDate	Date	Custome…	
CustomerMiles	Numeric(4.0)	Custome…	sum(CustomerTripMiles)
CustomerFreeTrips	Numeric(4.0)	Custome…	count(TripId, TripIsFree=True)
Trip	Trip	Trip	
TripId	Id	Trip Id	
TripDate	Date	Trip Date	
CountryId	Id	Country Id	
CityId	Id	City Id	
CityName	Name	City Name	
TripIsFree	Numeric(4.0)	Trip Is F…	
CustomerTripMiles	Numeric(4.0)	Custome…	



Application Name

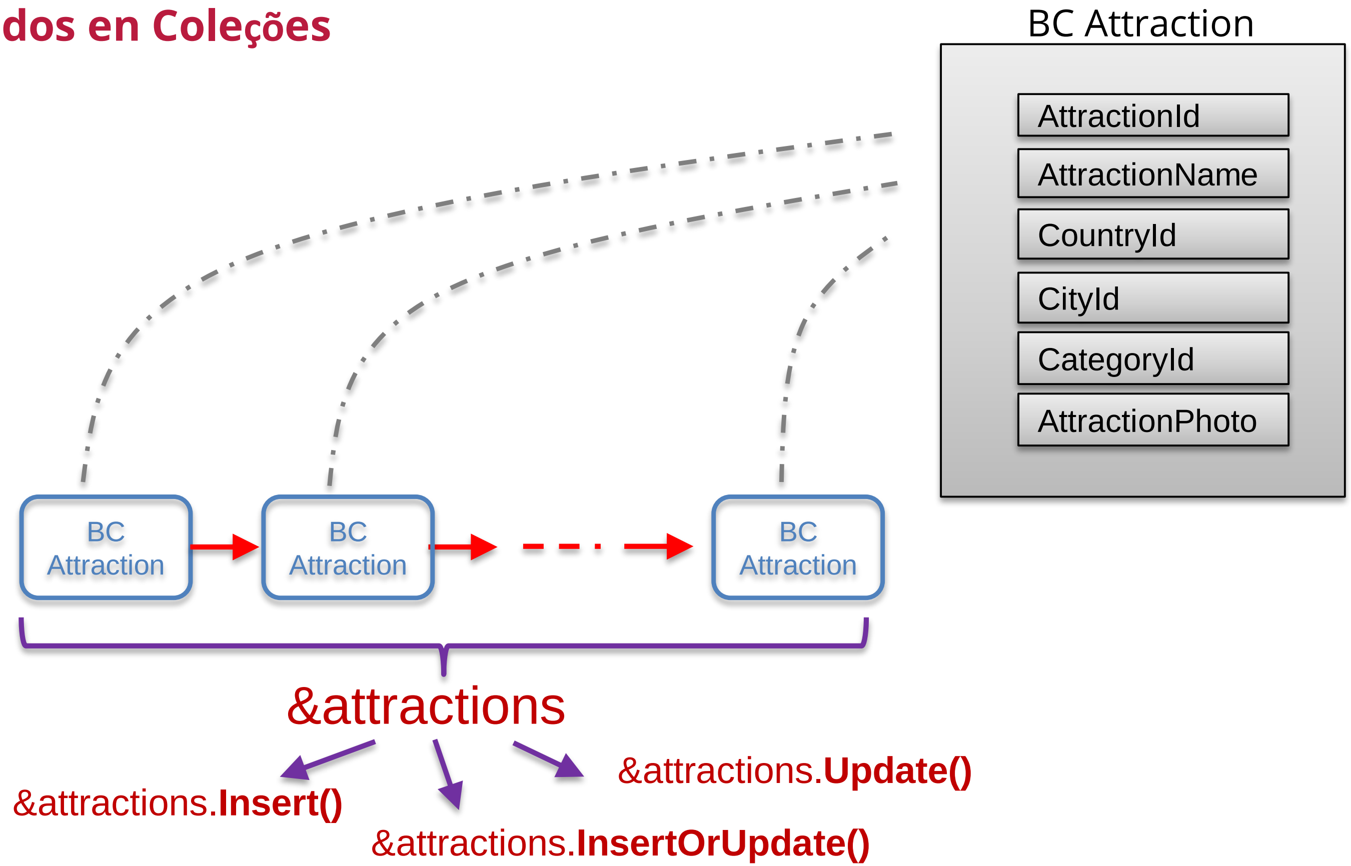
Customer: Anna Brown &customerId

Trip: Beautiful beaches of Rio de Janeiro &tripId



```
Event 'Delete Trip'
    &customer.Load(&CustomerId)
    &customer.Trip.RemoveByKey(&TripId)
    &customer.Save()
    Commit
Endevent
```

Métodos en Coleções



BC Attraction

AttractionId

AttractionName

CountryId

CityId

CategoryId

AttractionPhoto

BC Attraction

BC Attraction

BC Attraction

&attractions

&attractions.**Insert()**

&attractions.**InsertOrUpdate()**

&attractions.**Update()**

Insert / Update / InsertOrUpdate

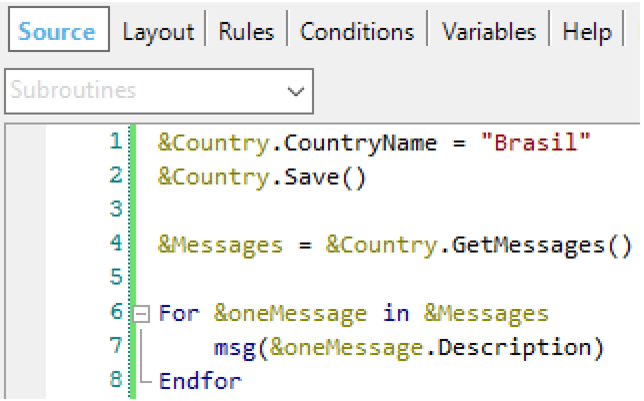
Recomendamos o uso dos métodos Insert, Update e InsertOrUpdate uma vez que:

- O uso dos métodos Load e Save para modificar o banco de dados é acessado duas vezes - o que diminui o desempenho - enquanto no Update ou InsertOrUpdate é acessado apenas uma vez.

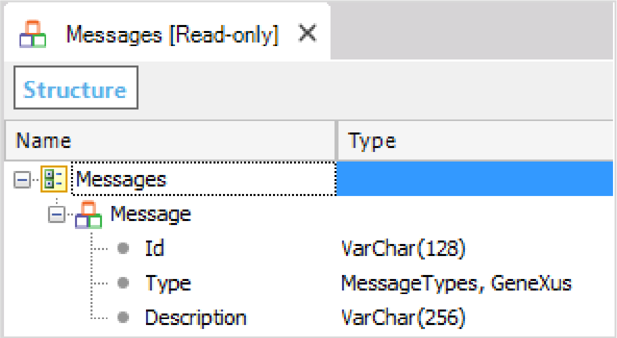
- Os nomes desses novos métodos já são claramente auto-explicativos.

Manuseio de erros trabalhando con BC

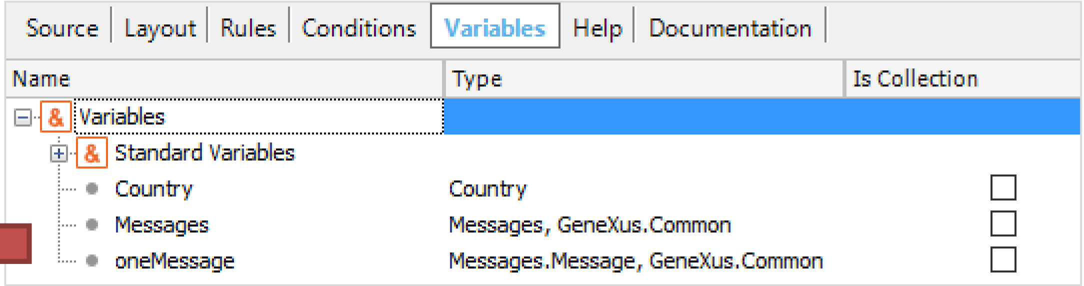
Para cada variável Business Component, uma coleção é carregada na memória com todas as mensagens de aviso ou erro que ocorrem como resultado das operações.



```
&Country.CountryName = "Brasil"
&Country.Save()

&Messages = &Country.GetMessages()

For &oneMessage in &Messages
    msg(&oneMessage.Description)
Endfor
```



Name	Type
Messages	
Message	
Id	VarChar(128)
Type	MessageTypes, GeneXus
Description	VarChar(256)



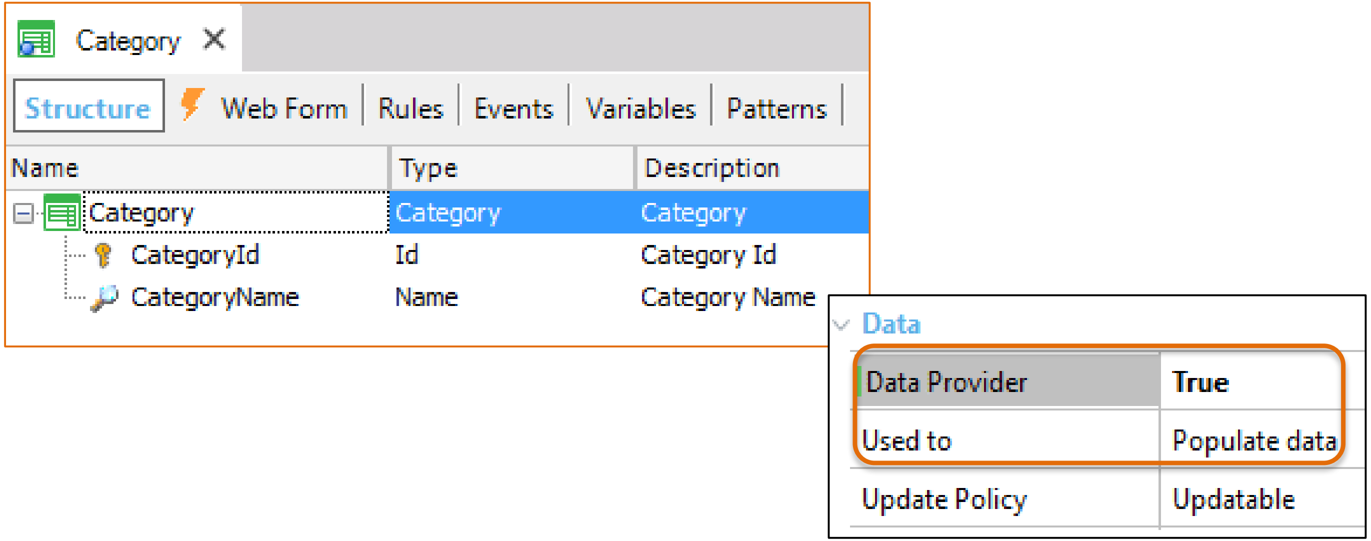
Name	Type	Is Collection
Variables		
Standard Variables		
Country	Country	□
Messages	Messages, GeneXus.Common	□
oneMessage	Messages.Message, GeneXus.Common	□

Carregamento de Dados

Transação

Inicializar dados de forma automática

GeneXus 15 oferece uma alternativa muito fácil para definir os dados com os quais deseja-se popular as tabelas físicas criadas associadas às transações, o que evita ter que criar as cargas de dados de outras maneiras.

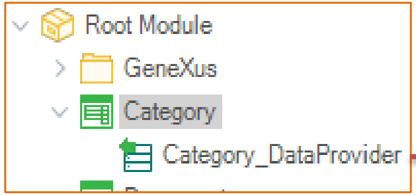


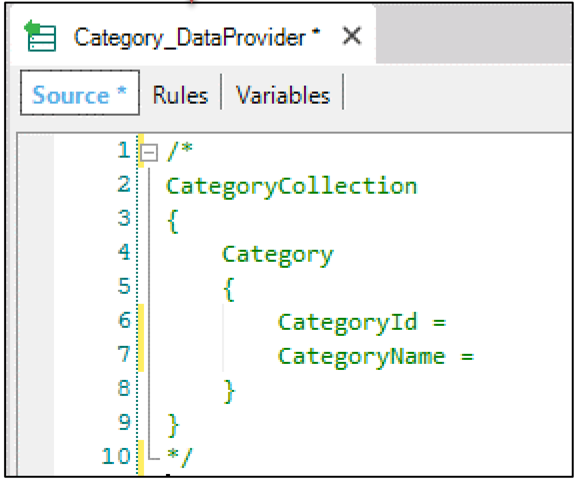
Name	Type	Description
Category	Category	Category
CategoryId	Id	Category Id
CategoryName	Name	Category Name

Data

Data Provider	True
Used to	Populate data
Update Policy	Updatable

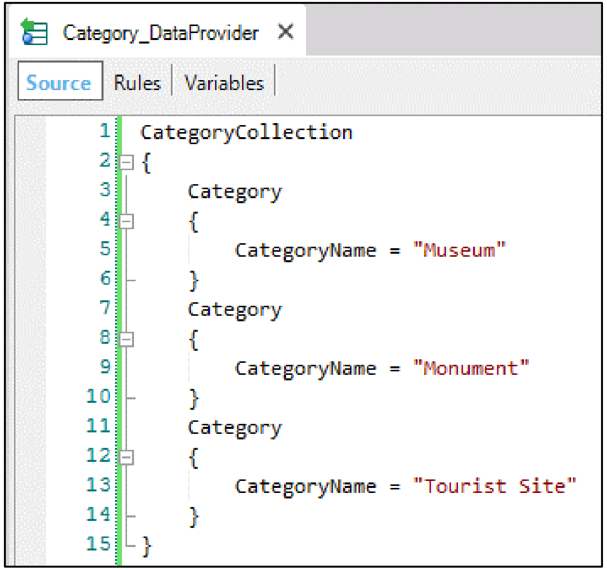
Inicializar dados





```
/*
CategoryCollection
{
    Category
    {
        CategoryId =
        CategoryName =
    }
}
*/
```

Não carregamos o CategoryId
já que foi definido como autonumber

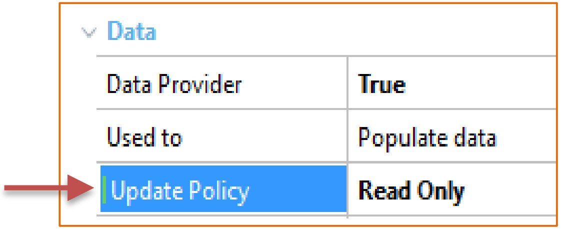


```
CategoryCollection
{
    Category
    {
        CategoryName = "Museum"
    }
    Category
    {
        CategoryName = "Monument"
    }
    Category
    {
        CategoryName = "Tourist Site"
    }
}
```

Inicializar dados: Somente leitura

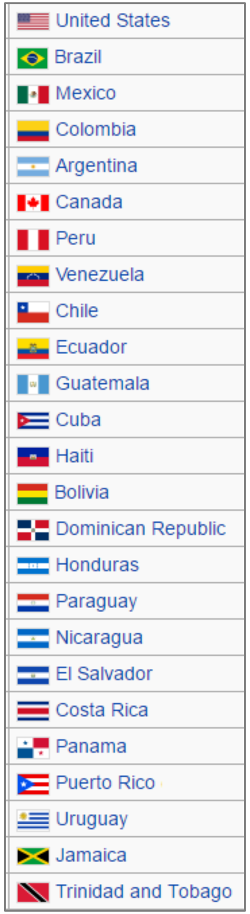
Country

```
{
   CountryId*
   CountryName
}
```



Data

Data Provider	True
Used to	Populate data
Update Policy	Read Only



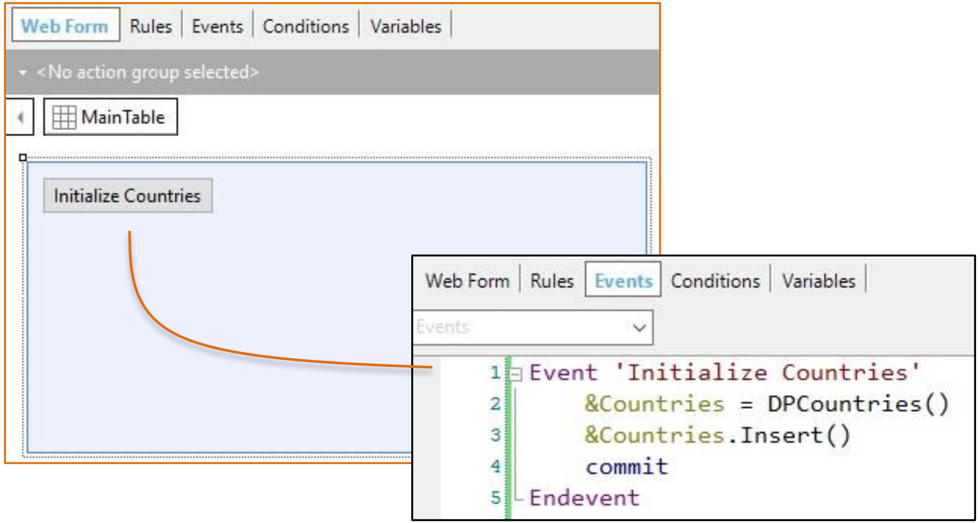
- United States
- Brazil
- Mexico
- Colombia
- Argentina
- Canada
- Peru
- Venezuela
- Chile
- Ecuador
- Guatemala
- Cuba
- Haiti
- Bolivia
- Dominican Republic
- Honduras
- Paraguay
- Nicaragua
- El Salvador
- Costa Rica
- Panama
- Puerto Rico
- Uruguay
- Jamaica
- Trinidad and Tobago

Carregamento de Dados

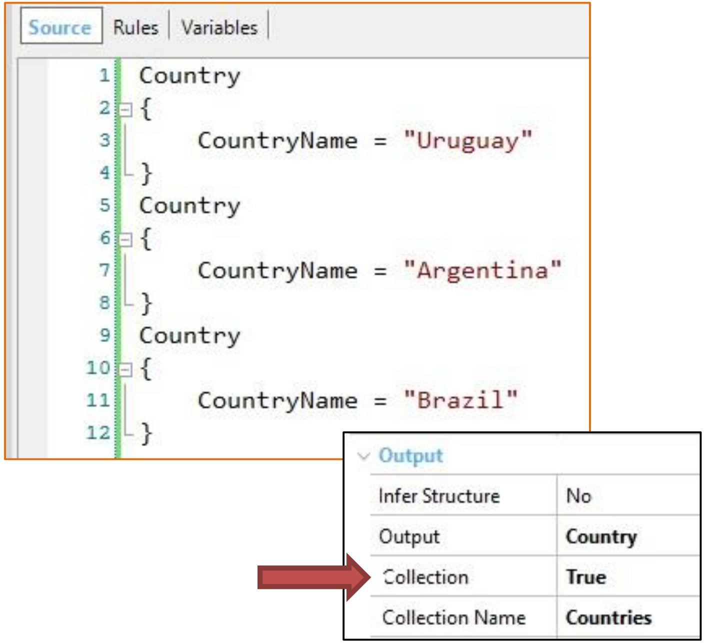
Business Components e Data Providers

Exemplo

Country
{
CountryId* ← Autonumber = True
CountryName
}


Web Form | Rules | Events | Conditions | Variables
<No action group selected>
MainTable
Initialize Countries
Web Form | Rules | Events | Conditions | Variables
Events

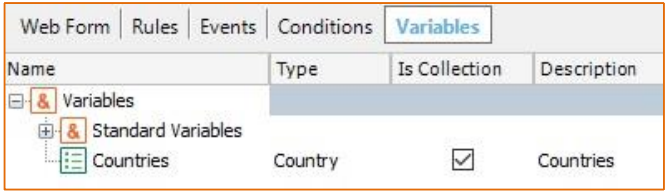
```
Event 'Initialize Countries'
    &Countries = DPCountries()
    &Countries.Insert()
    commit
Endevent
```



```
Country
{
    CountryName = "Uruguay"
}
Country
{
    CountryName = "Argentina"
}
Country
{
    CountryName = "Brazil"
}
```

Output

Infer Structure	No
Output	**Country**
Collection	**True**
Collection Name	**Countries**



Name	Type	Is Collection	Description
Variables			
Standard Variables			
Countries	Country	☑	Countries

Carregamento de Dados com Procedimentos
Comandos New / For each / Delete

Considerações

Os comandos New / For Each / Delete são usados para inserir, alterar e excluir informações do banco de dados, mas eles só podem ser usados em Procedimentos.

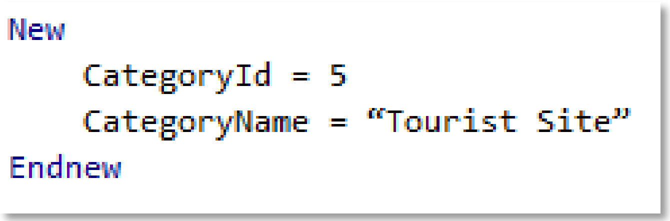
Embora os comandos a seguir sejam usados para inserir, alterar e excluir informações do banco de dados, recomendamos o uso do Business Component, uma vez que eles:

- Controlam a integridade referencial

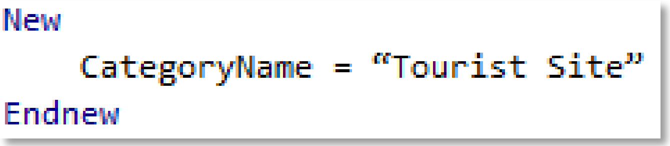
- Disparam as regras definidas na transação

Inserção – Comando NEW

Category
{
  CategoryId*
  CategoryName
}



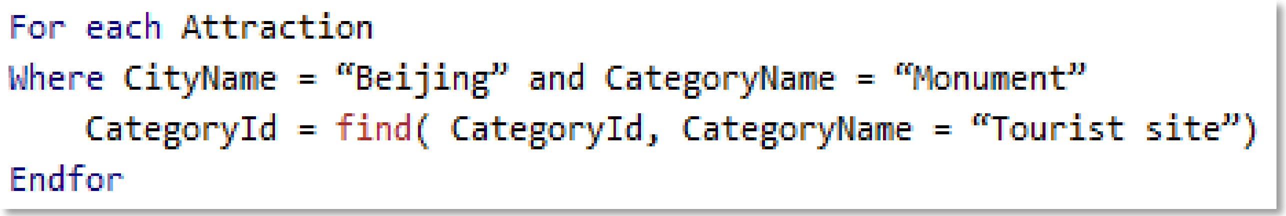
```
New
    CategoryId = 5
    CategoryName = "Tourist Site"
Endnew
```



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

Se o atributo é autonumber
não o informamos

Alteração / Comando FOR EACH



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

```
Category
{
  CategoryId*
  CategoryName
}
```

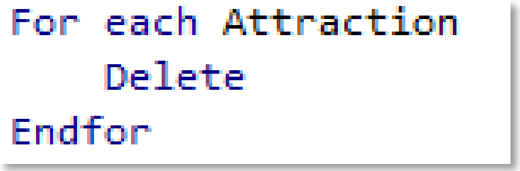
```
Country
{
  CountryId*
  CountryName
  City
  {
     CityId*
     CityName
  }
}
```

```
Attraction
{
  AttractionId*
  AttractionName
  CategoryId
  CategoryName
  CountryId
  CountryName
  CityId
  CitlyName
}
```

Exclusão – Comando DELETE

Attraction

```
{
  AttractionId*
  AttractionName
  CategoryId
  CategoryName
  CountryId
  CountryName
  CityId
  CitlyName
}
```

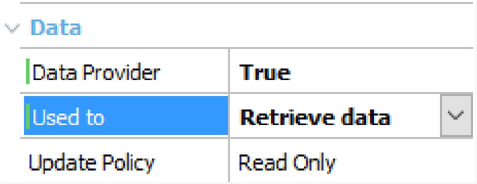


```
For each Attraction
    Delete
Endfor
```

Transações Dinâmicas

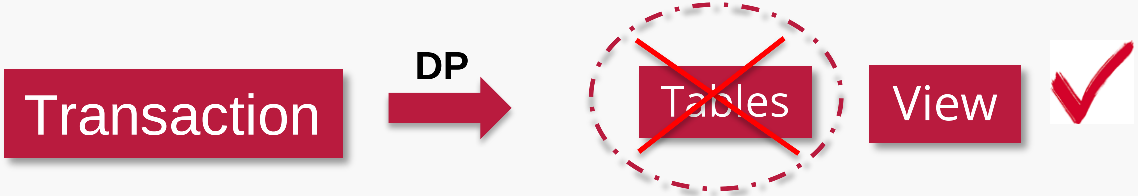
Transações dinâmicas

1. Data Provider: True
2. Used to: Retrieve data
3. Update Policy: - Read Only
   - Updatable



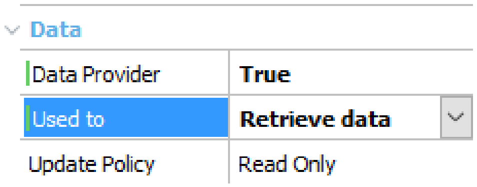
Data

Data Provider	True
Used to	Retrieve data
Update Policy	Read Only



Transações dinâmicas para recuperar dados

- No Data Provider devemos indicar quais os dados que queremos recuperar
- Uma transação dinâmica pode ser referenciada como Base Trn

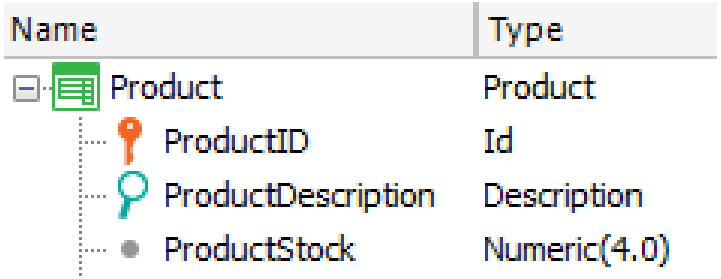


Data

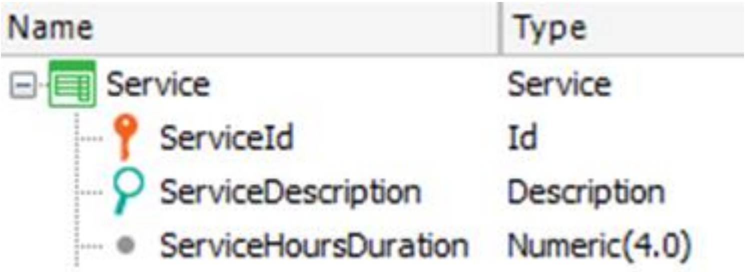
Data Provider	**True**
Used to	**Retrieve data**
Update Policy	Read Only

Exemplo 1: União de dados

Uma empresa vende produtos e oferece serviços.
É necessário emitir um relatório em orden alfabética com tudo o que a empresa oferece.

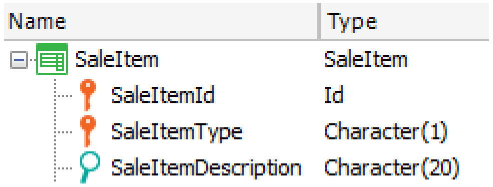


Name	Type
Product	Product
ProductID	Id
ProductDescription	Description
ProductStock	Numeric(4.0)

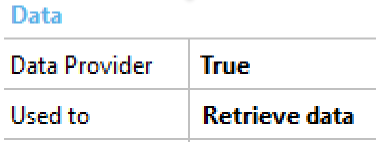


Name	Type
Service	Service
ServiceId	Id
ServiceDescription	Description
ServiceHoursDuration	Numeric(4.0)

Exemplo: DataProvider gerado

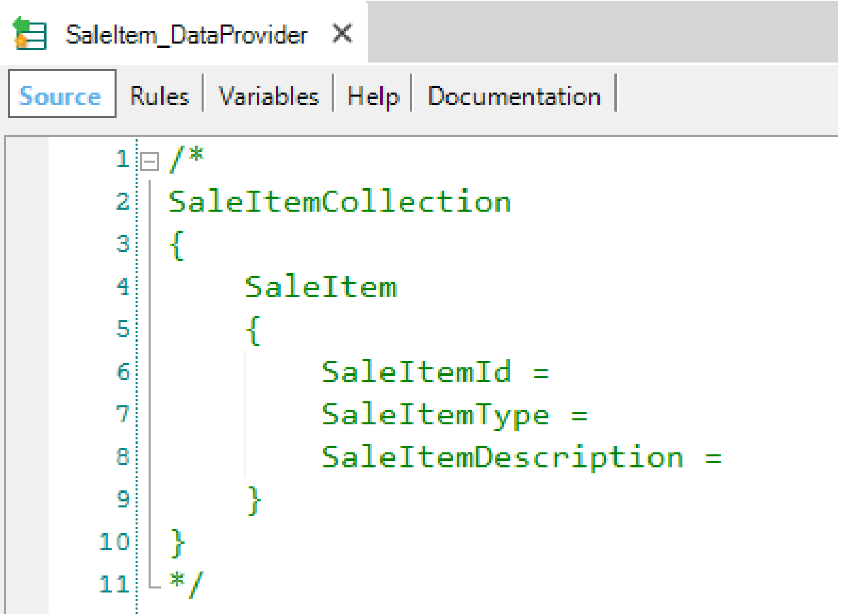


Name	Type
SaleItem	SaleItem
SaleItemId	Id
SaleItemType	Character(1)
SaleItemDescription	Character(20)



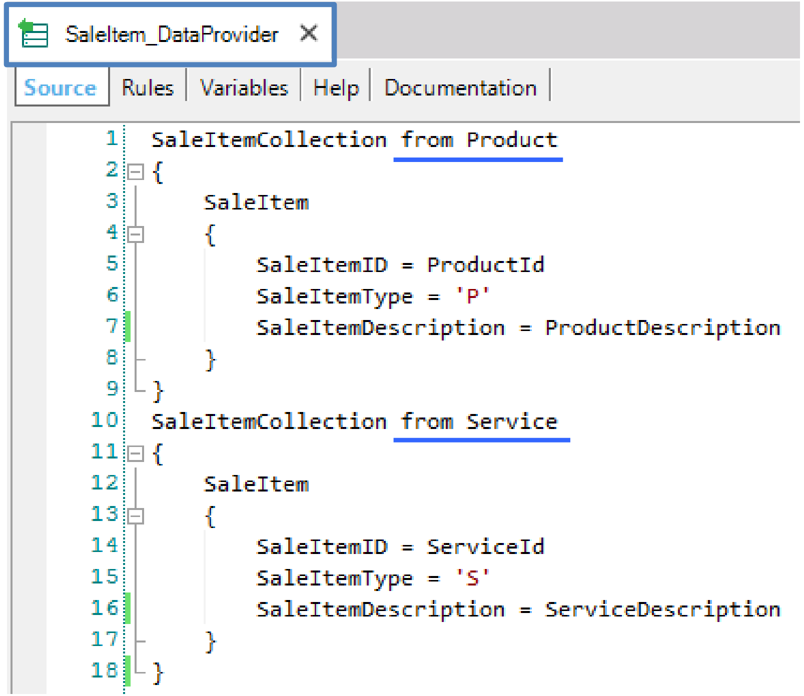
Data

Data Provider	**True**
Used to	**Retrieve data**



```
/*
SaleItemCollection
{
    SaleItem
    {
        SaleItemId =
        SaleItemType =
        SaleItemDescription =
    }
}
*/
```

Exemplo: DataProvider



```
SaleItemCollection from Product
{
    SaleItem
    {
        SaleItemID = ProductId
        SaleItemType = 'P'
        SaleItemDescription = ProductDescription
    }
}
SaleItemCollection from Service
{
    SaleItem
    {
        SaleItemID = ServiceId
        SaleItemType = 'S'
        SaleItemDescription = ServiceDescription
    }
}
```

Exemplo: especificação de SaleItem

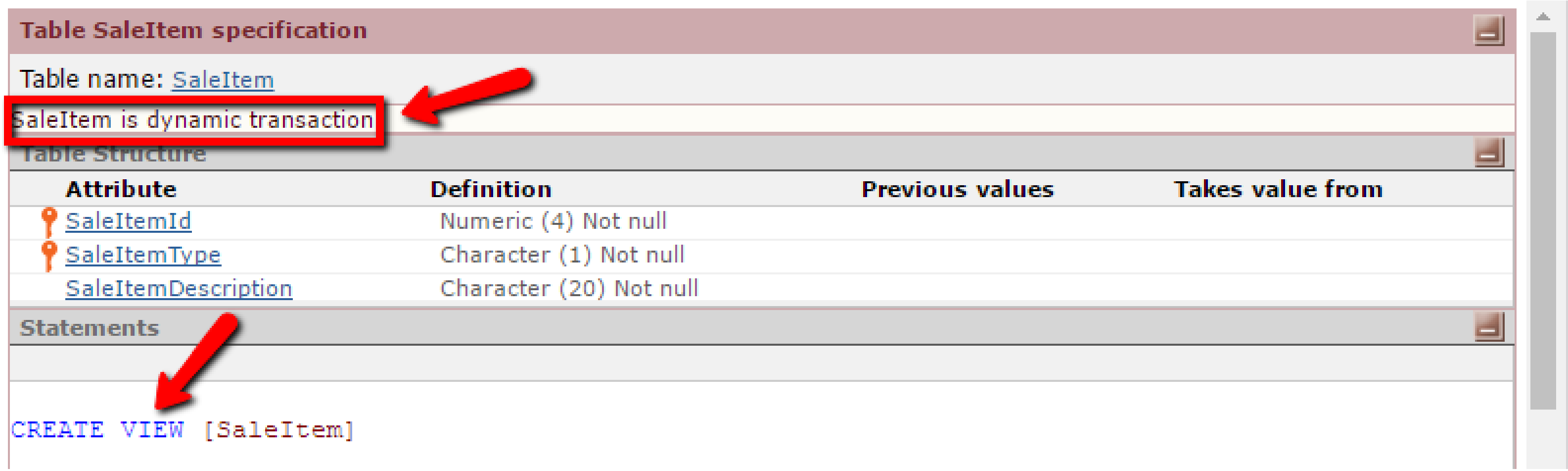


Table SaleItem specification

Table name: SaleItem

SaleItem is dynamic transaction

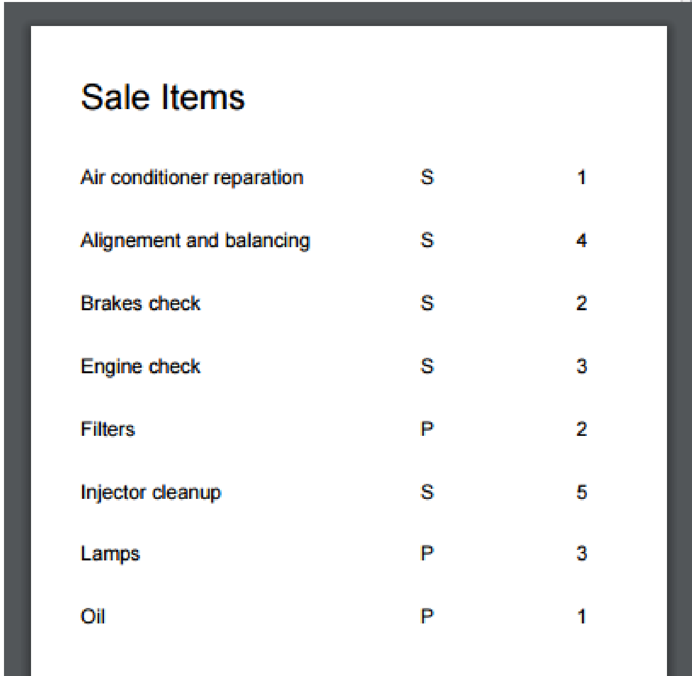
Table Structure

Attribute	Definition	Previous values	Takes value from
SaleItemId	Numeric (4) Not null		
SaleItemType	Character (1) Not null		
SaleItemDescription	Character (20) Not null		

Statements

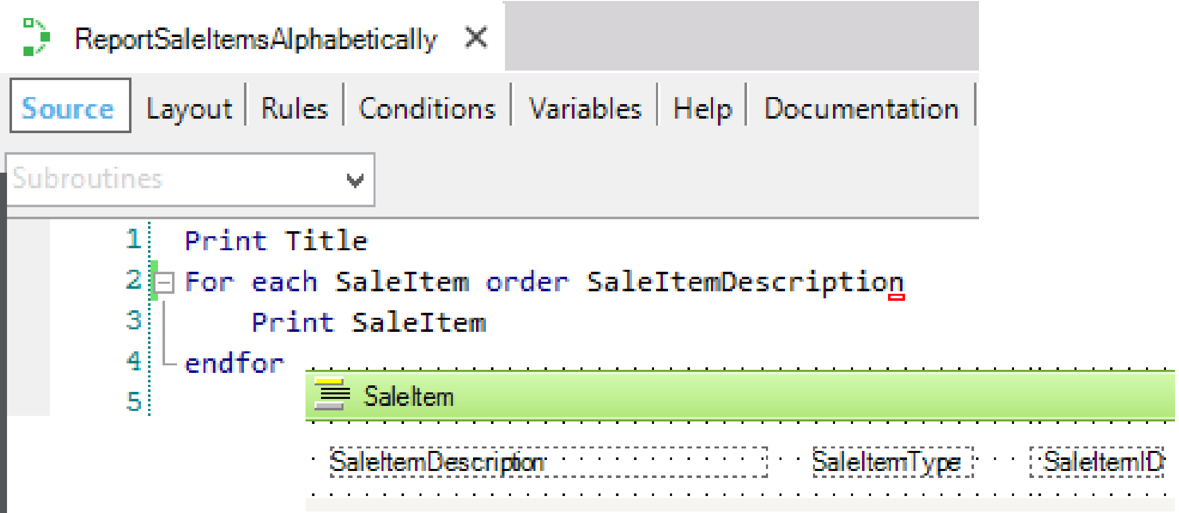
```
CREATE VIEW [SaleItem]
```

Exemplo: relatório PDF



Sale Items

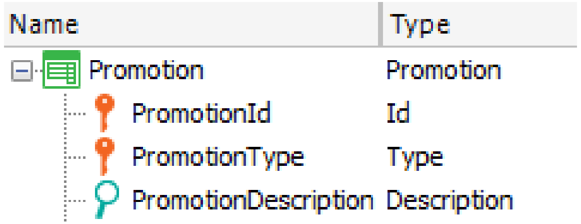
Air conditioner reparation	S	1
Alignement and balancing	S	4
Brakes check	S	2
Engine check	S	3
Filters	P	2
Injector cleanup	S	5
Lamps	P	3
Oil	P	1



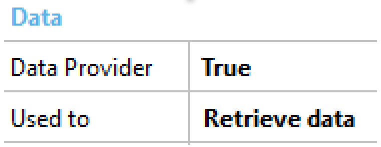
```
Print Title
For each SaleItem order SaleItemDescription
    Print SaleItem
endfor
```

Exemplo 2: Modelar a realidade

- Products → stock > 1000
- Services → < 10 contratações

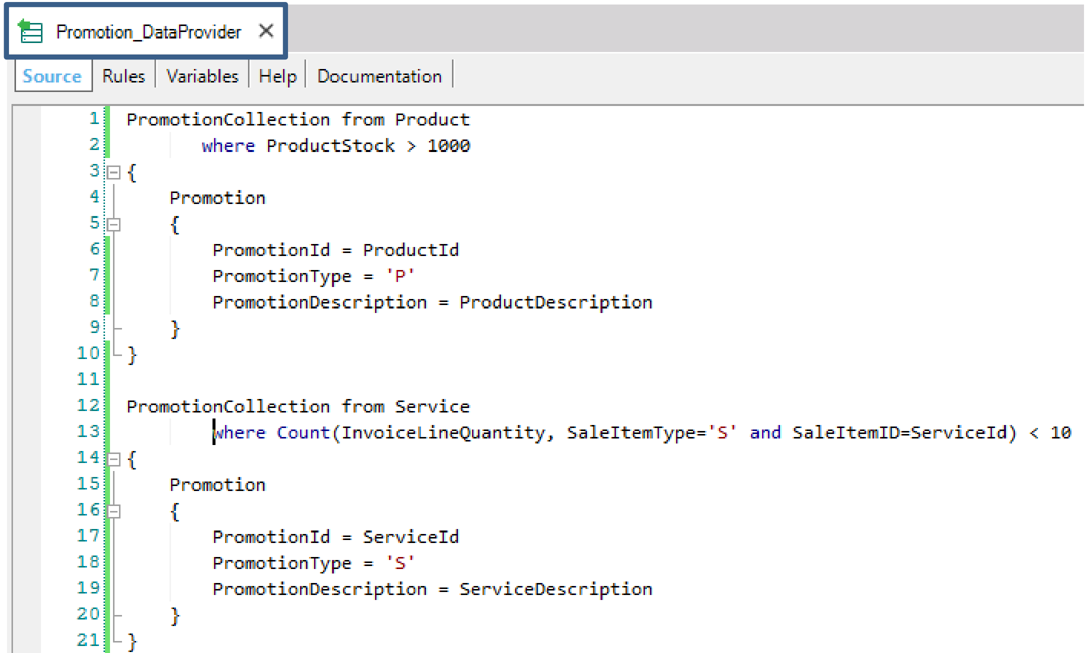


Name	Type
Promotion	Promotion
PromotionId	Id
PromotionType	Type
PromotionDescription	Description

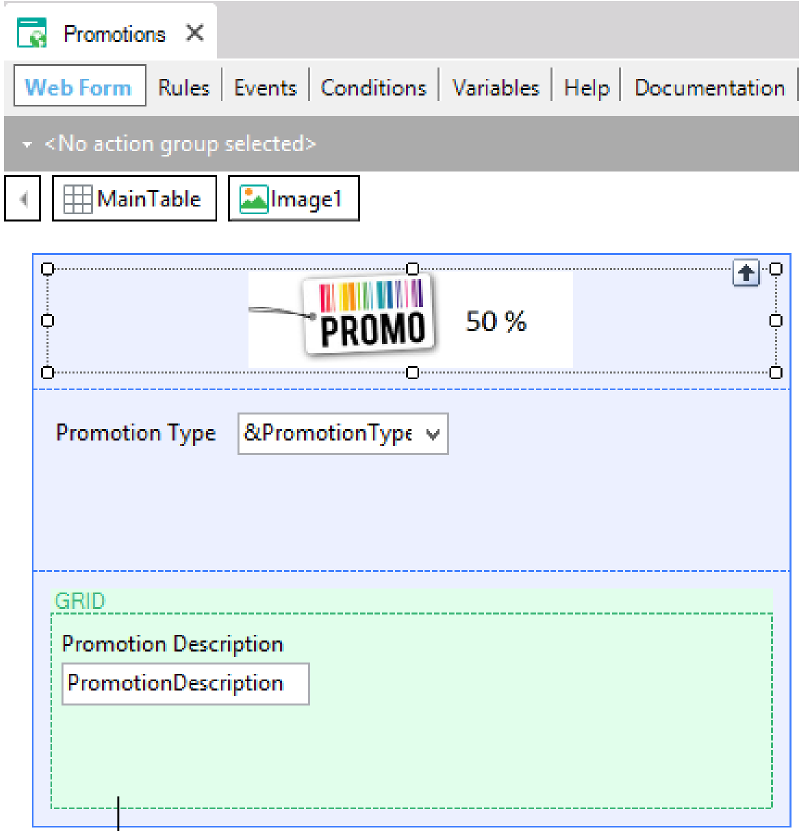


Data

Data Provider	True
Used to	Retrieve data

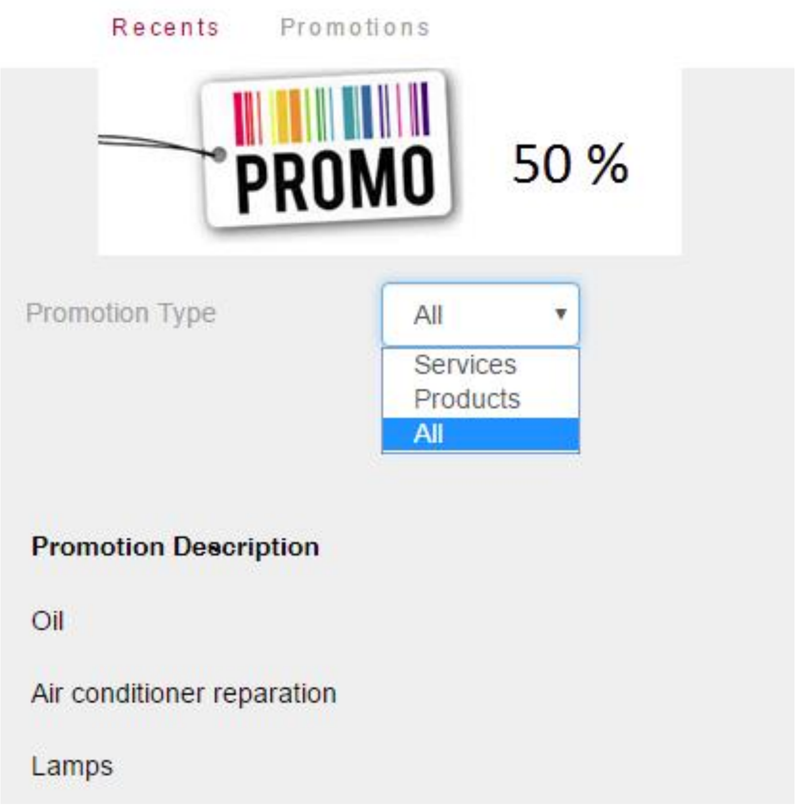


```
PromotionCollection from Product
        where ProductStock > 1000
{
    Promotion
    {
        PromotionId = ProductId
        PromotionType = 'P'
        PromotionDescription = ProductDescription
    }
}

PromotionCollection from Service
        where Count(InvoiceLineQuantity, SaleItemType='S' and SaleItemID=ServiceId) < 10
{
    Promotion
    {
        PromotionId = ServiceId
        PromotionType = 'S'
        PromotionDescription = ServiceDescription
    }
}
```



Promotion Type &PromotionType

Promotion Description PromotionDescription



Promotion Type All

- Services
- Products
- All

Promotion Description

Oil

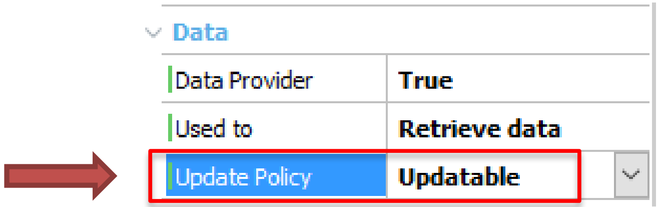
Air conditioner reparation

Lamps

Conditions	**PromotionType=&PromotionType when &PromotionType <> 'A';**

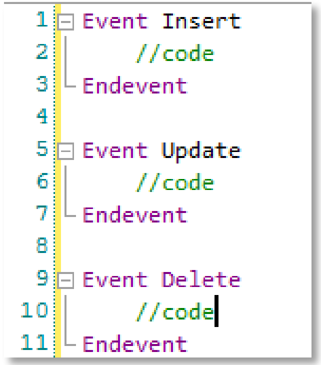
Transações Dinâmicas para atualizar dados

Como conseguimos atualizar os dados, uma vez que não temos uma tabela associada à transação?



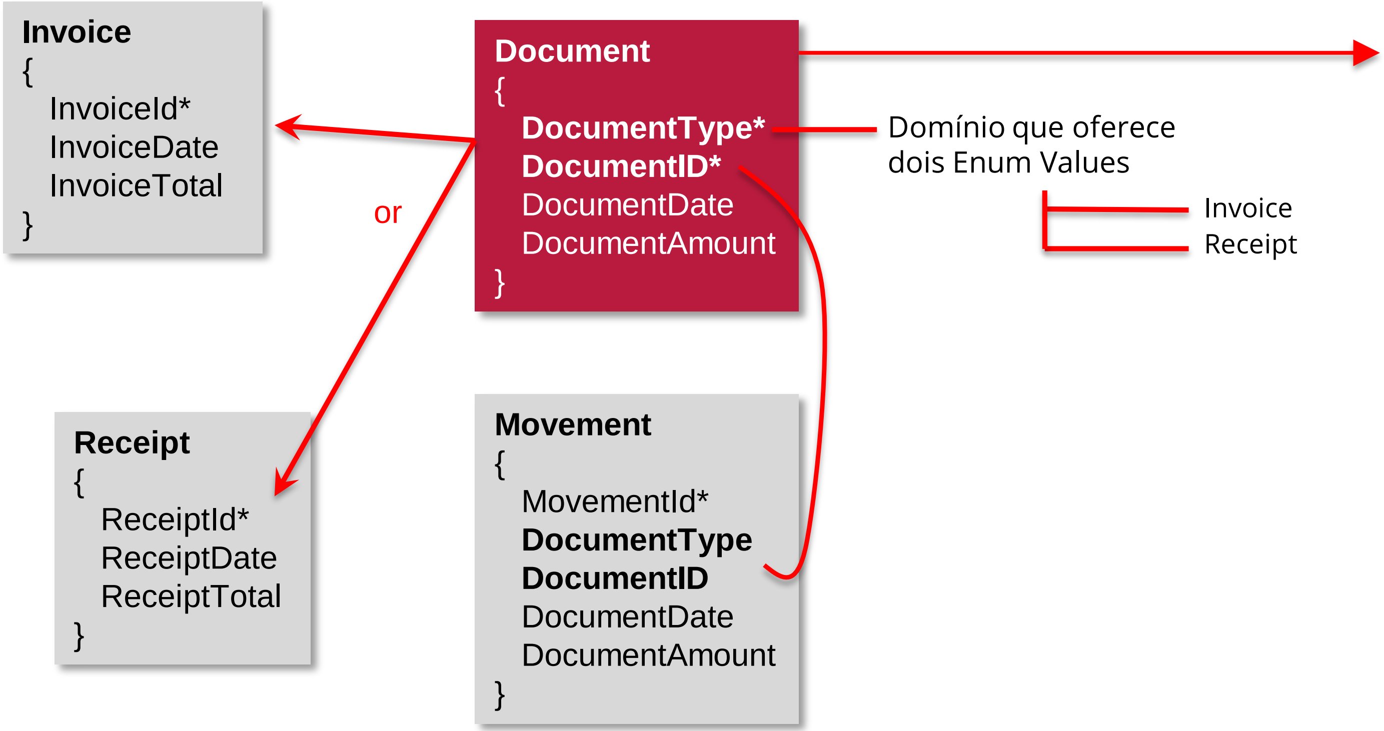
Data

Data Provider	True
Used to	Retrieve data
Update Policy	Updatable



```
Event Insert
	//code
Endevent

Event Update
	//code
Endevent

Event Delete
	//code
Endevent
```

O desenvolvedor deve programar os eventos **Insert**, **Update** e **Delete**.



```
Invoice
{
   InvoiceId*
   InvoiceDate
   InvoiceTotal
}
```

or

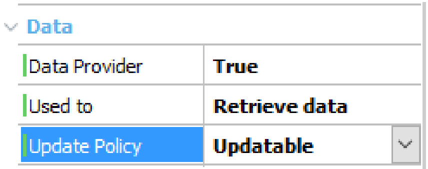
```
Receipt
{
   ReceiptId*
   ReceiptDate
   ReceiptTotal
}
```

```
Document
{
   DocumentType*
   DocumentID*
   DocumentDate
   DocumentAmount
}
```

Domínio que oferece dois Enum Values

- Invoice
- Receipt

```
Movement
{
   MovementId*
   DocumentType
   DocumentID
   DocumentDate
   DocumentAmount
}
```



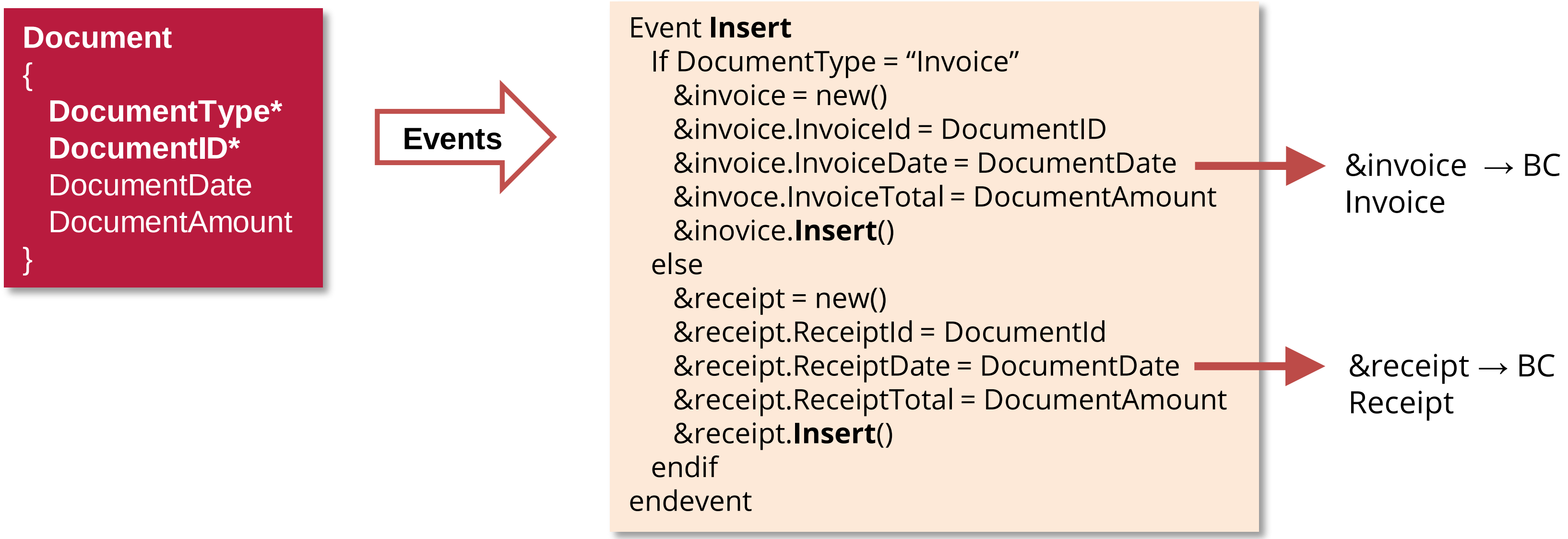
Data

Data Provider	True
Used to	Retrieve data
Update Policy	Updatable

Transação Dinâmica

Não gera tabela

Exemplo 3: uso de Transações Dinâmicas para atualizar dados



```
Document
{
   DocumentType*
   DocumentID*
   DocumentDate
   DocumentAmount
}
```

```
Event Insert
  If DocumentType = “Invoice”
    &invoice = new()
    &invoice.InvoiceId = DocumentID
    &invoice.InvoiceDate = DocumentDate
    &invoce.InvoiceTotal = DocumentAmount
    &inovice.Insert()
  else
    &receipt = new()
    &receipt.ReceiptId = DocumentId
    &receipt.ReceiptDate = DocumentDate
    &receipt.ReceiptTotal = DocumentAmount
    &receipt.Insert()
  endif
endevent
```

&invoice → BC Invoice

&receipt → BC Receipt

Integridade Transacional

Conceitos

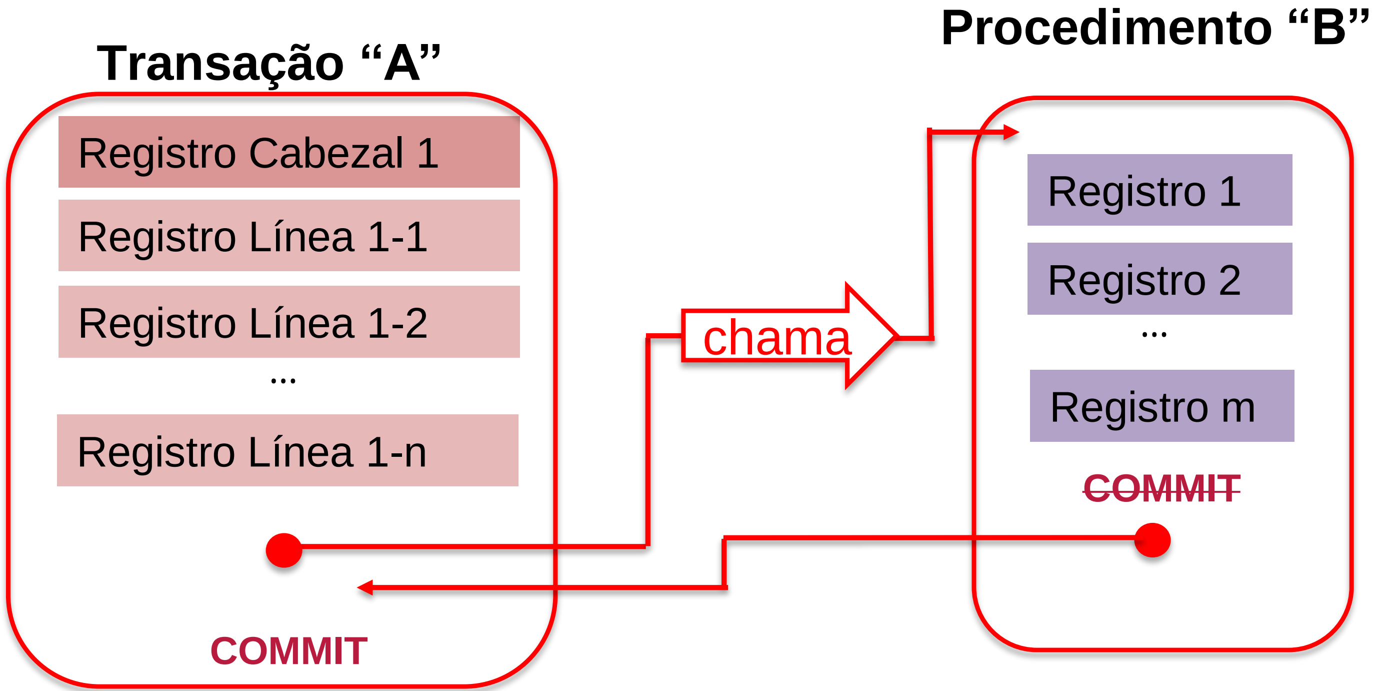
- Um conjunto de atualizações ao banco de dados possui **integridade transacional** quando, caso aconteça uma finalização “anormal”, o banco de dados permanece em **estado consistente.**

- A **consistência**, neste ponto, é determinada pelas Unidades de Trabalho Lógicas (UTL): operações sobre o banco de dados realizadas entre dois **Commit.**

- **Transações** e **Procedimentos** → GeneXus escreve automaticamente no final dos programas gerados, o comando **Commit.** Através da propriedade **Commit on Exit** (“Yes”, “No”) do objeto, pode ser desabilitado.

- **Business Component** → GeneXus não escreve Commit.

Personalizar UTLs



Transação "A"

Registro Cabezal 1

Registro Línea 1-1

Registro Línea 1-2

…

Registro Línea 1-n

COMMIT

chama

Procedimento "B"

Registro 1

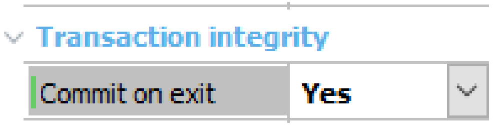
Registro 2

…

Registro m

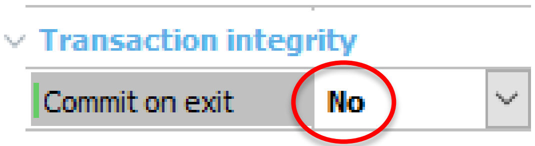
~~COMMIT~~

B (parm$_1$, … , parm$_n$) **on BeforeComplete**;



Transaction integrity

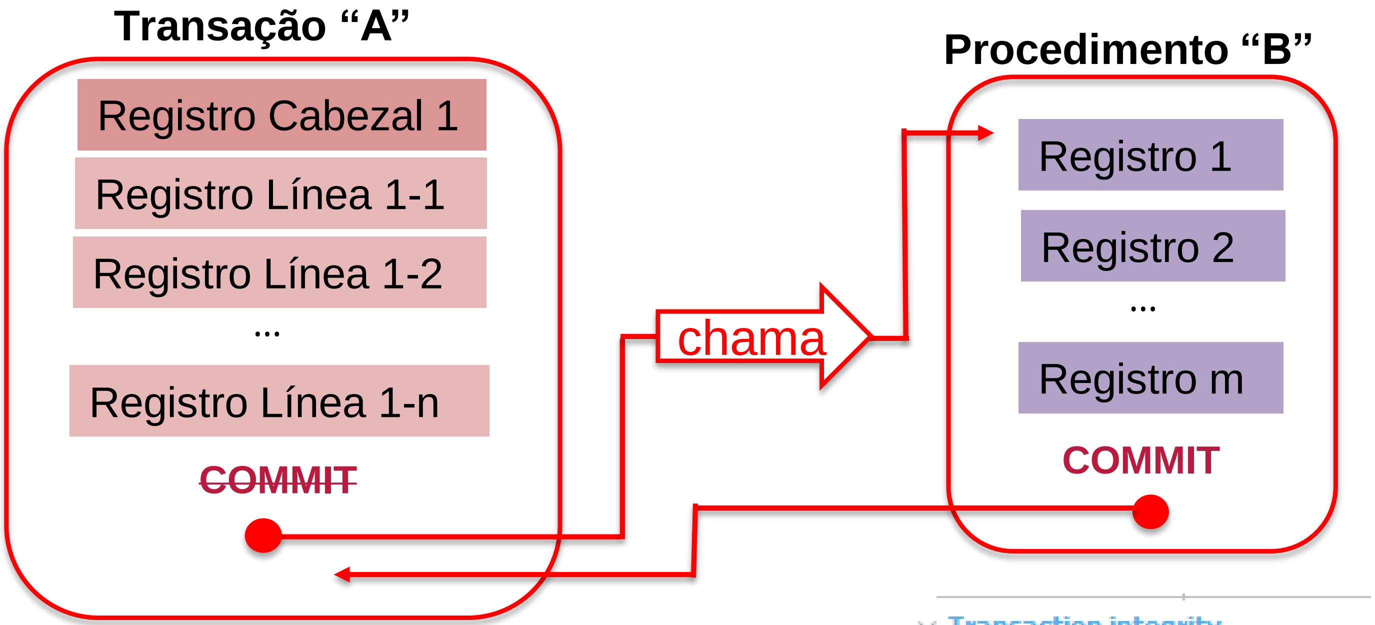
Commit on exit: Yes

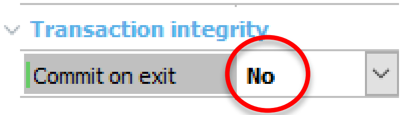


Transaction integrity

Commit on exit: No

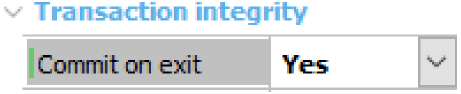
Personalizar UTLs





Transaction integrity

Commit on exit [✓ No]



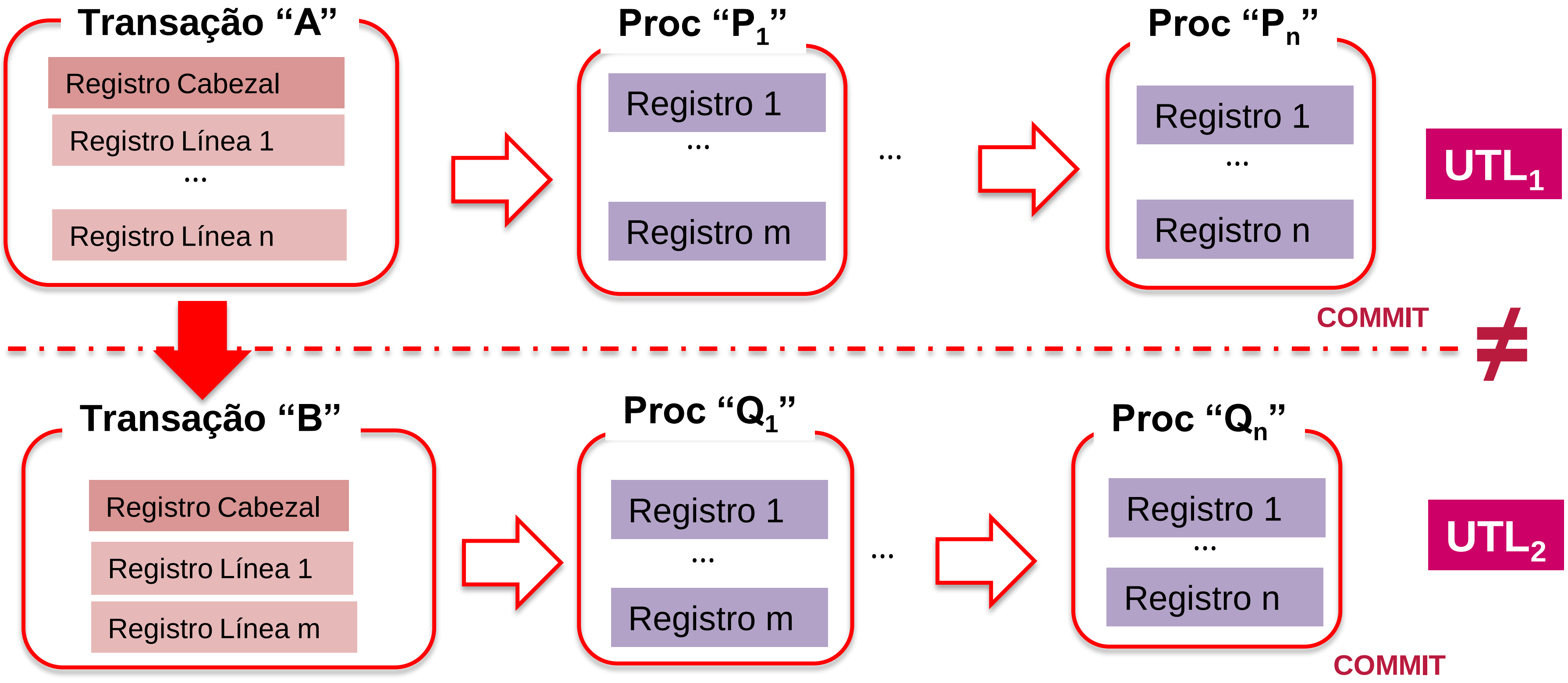
Transaction integrity

Commit on exit Yes

B (parm$_1$, … , parm$_n$) on AfterComplete;

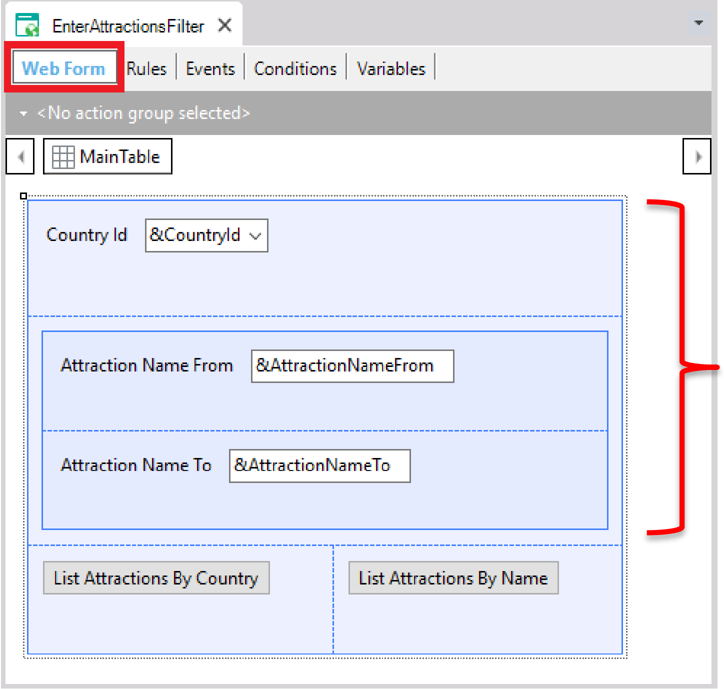
Personalizar UTLs

Transação só pode commitar seus registros e os de procedimentos em cascata, na orden em que são chamados: NÃO os registros de outra transação:



Web Panels

Web Panel sem grid, com variáveis no form



Country Id &CountryId

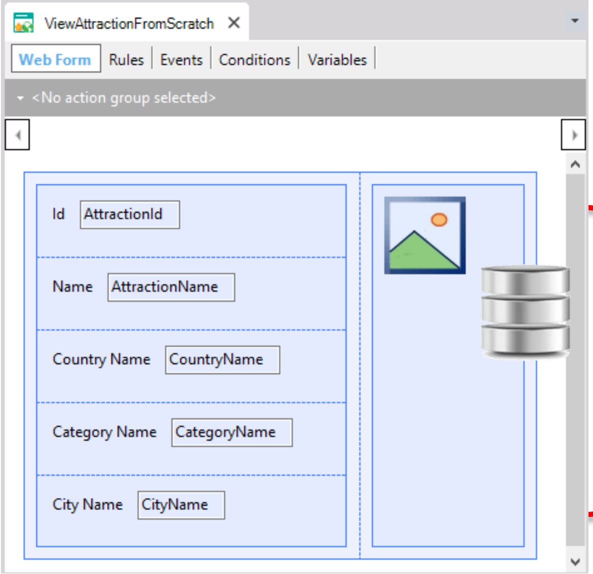
Attraction Name From &AttractionNameFrom

Attraction Name To &AttractionNameTo

Variáveis: **de entrada**
(não readonly)

Web Panel sem grid mas com atributos no form

Parm(in: **AttractionId**);

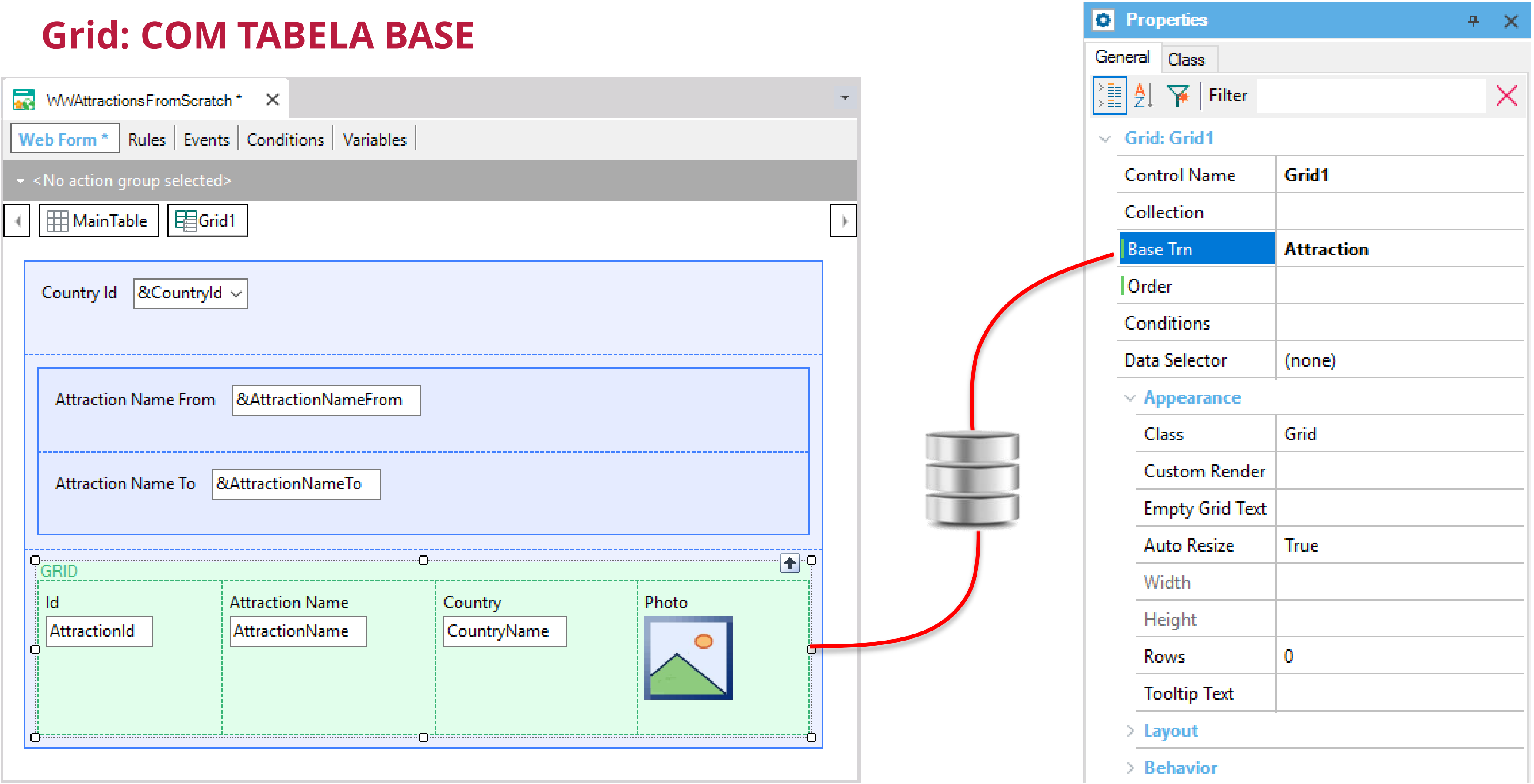


ViewAttractionFromScratch

Id	AttractionId
Name	AttractionName
Country Name	CountryName
Category Name	CategoryName
City Name	CityName

É carregado só **um** registro

Grid: COM TABELA BASE



WWAttractionsFromScratch *

Web Form * | Rules | Events | Conditions | Variables

<No action group selected>

MainTable | Grid1

Country Id: &CountryId

Attraction Name From: &AttractionNameFrom

Attraction Name To: &AttractionNameTo

GRID

Id	Attraction Name	Country	Photo
AttractionId	AttractionName	CountryName	

Properties

General | Class

Filter

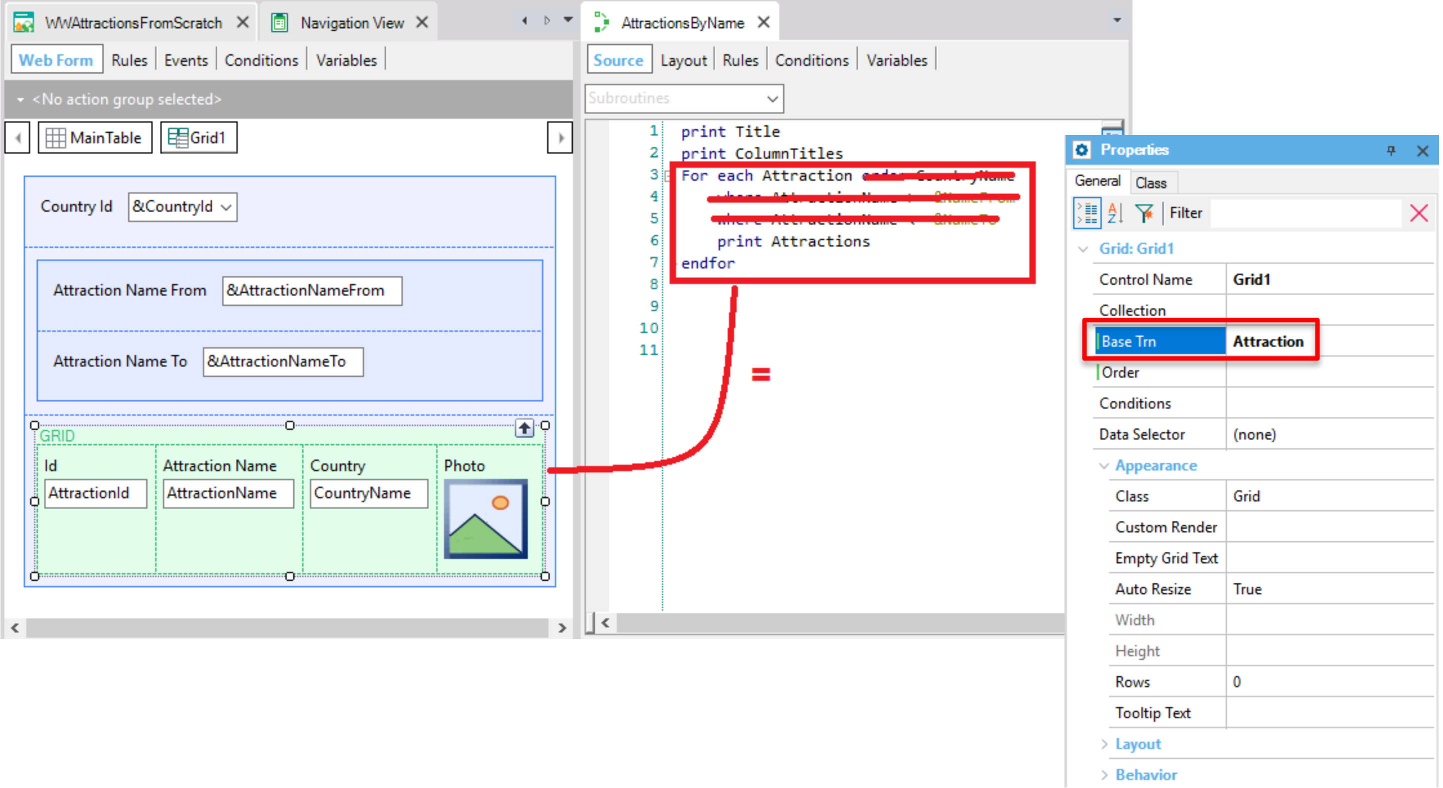
Grid: Grid1

Control Name	Grid1
Collection	
Base Trn	Attraction
Order	
Conditions	
Data Selector	(none)
Appearance	
Class	Grid
Custom Render	
Empty Grid Text	
Auto Resize	True
Width	
Height	
Rows	0
Tooltip Text	

Layout

Behavior

Transação base



Country Id &CountryId

Attraction Name From &AttractionNameFrom

Attraction Name To &AttractionNameTo

GRID

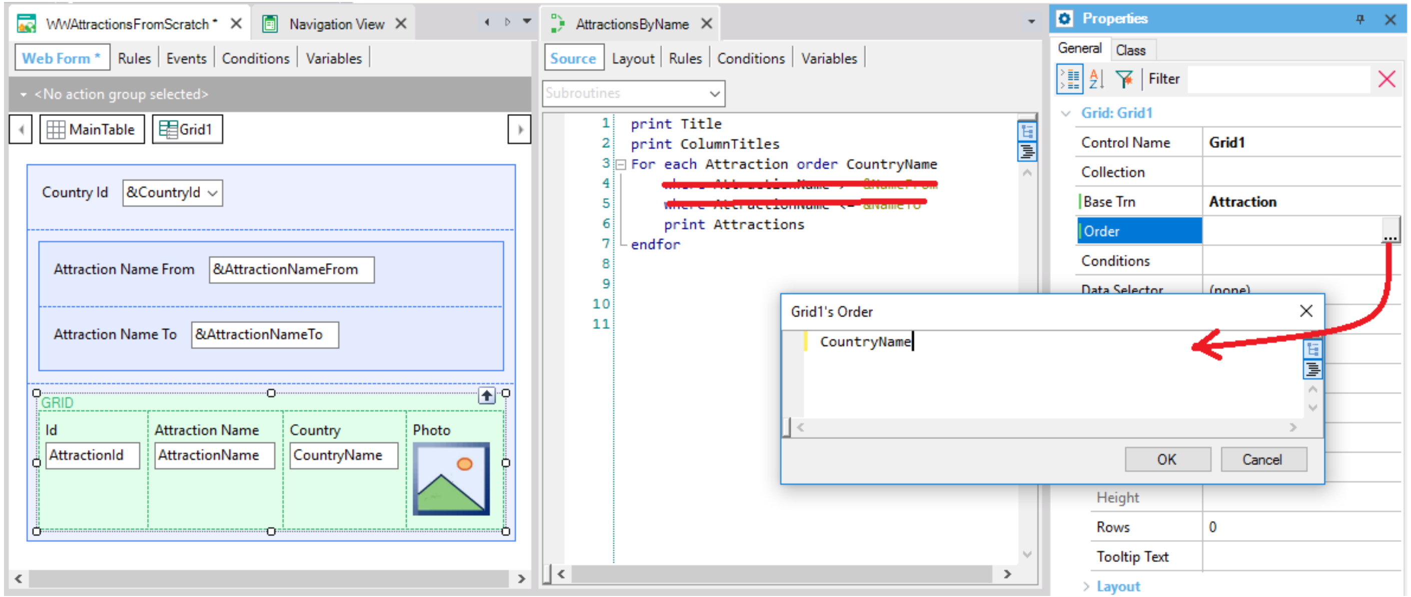
Id	Attraction Name	Country	Photo
AttractionId	AttractionName	CountryName	

```
print Title
print ColumnTitles
For each Attraction ~~order CountryName~~
    ~~where AttractionName >= &NameFrom~~
    ~~where AttractionName <= &NameTo~~
    print Attractions
endfor
```

Grid: Grid1

Control Name	Grid1
Collection	
Base Trn	Attraction
Order	
Conditions	
Data Selector	(none)
Appearance	
Class	Grid
Custom Render	
Empty Grid Text	
Auto Resize	True
Width	
Height	
Rows	0
Tooltip Text	
Layout	
Behavior	

Ordem



```
print Title
print ColumnTitles
For each Attraction order CountryName
    ~~where AttractionName >= &NameFrom~~
    ~~where AttractionName <= &NameTo~~
    print Attractions
endfor
```

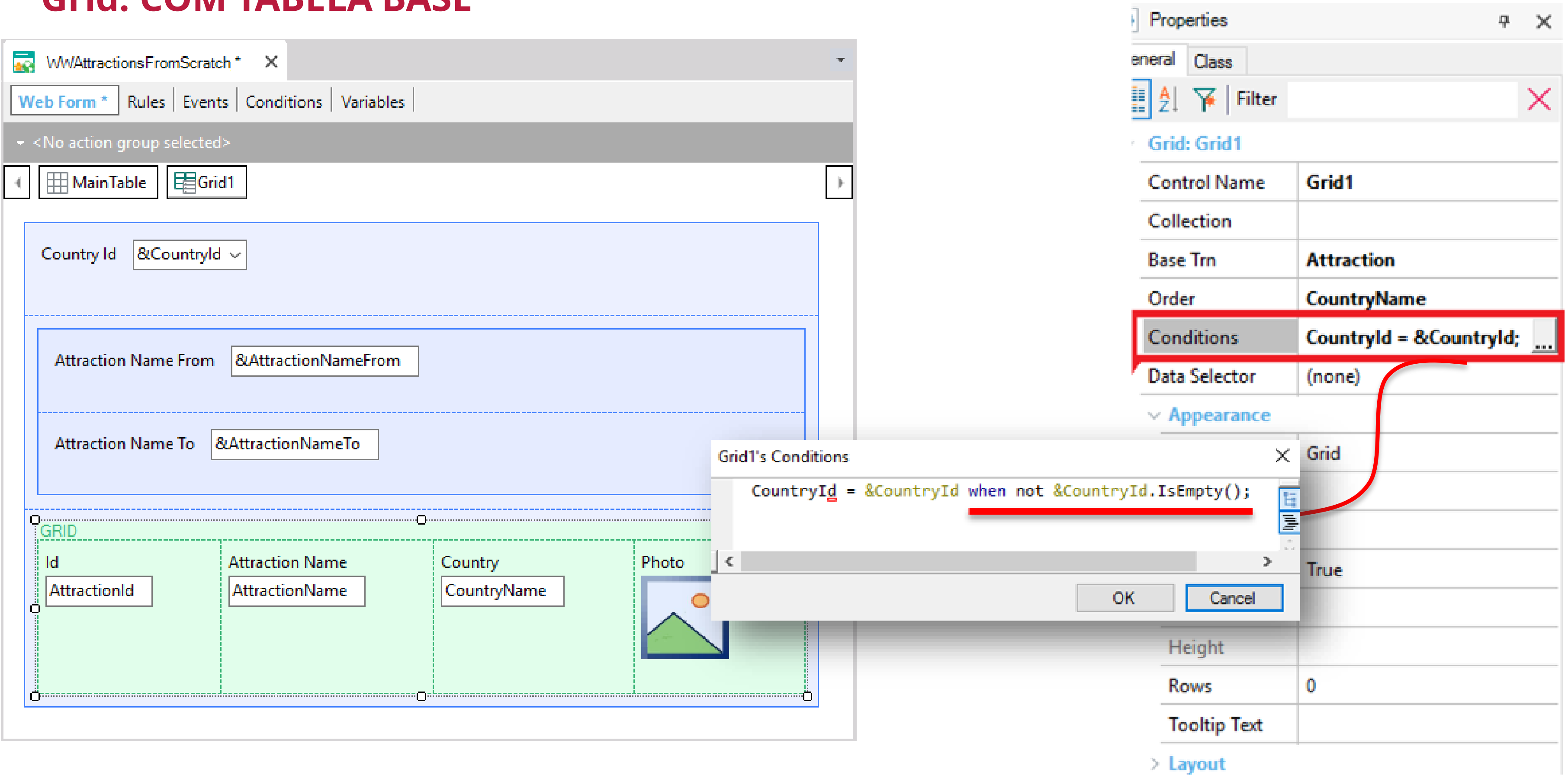
Control Name	Grid1
Collection	
Base Trn	Attraction
Order	
Conditions	
Data Selector	(none)
Height	
Rows	0
Tooltip Text	

Layout

Grid1's Order:

```
CountryName
```

Grid: COM TABELA BASE



Country Id: &CountryId

Attraction Name From: &AttractionNameFrom

Attraction Name To: &AttractionNameTo

GRID

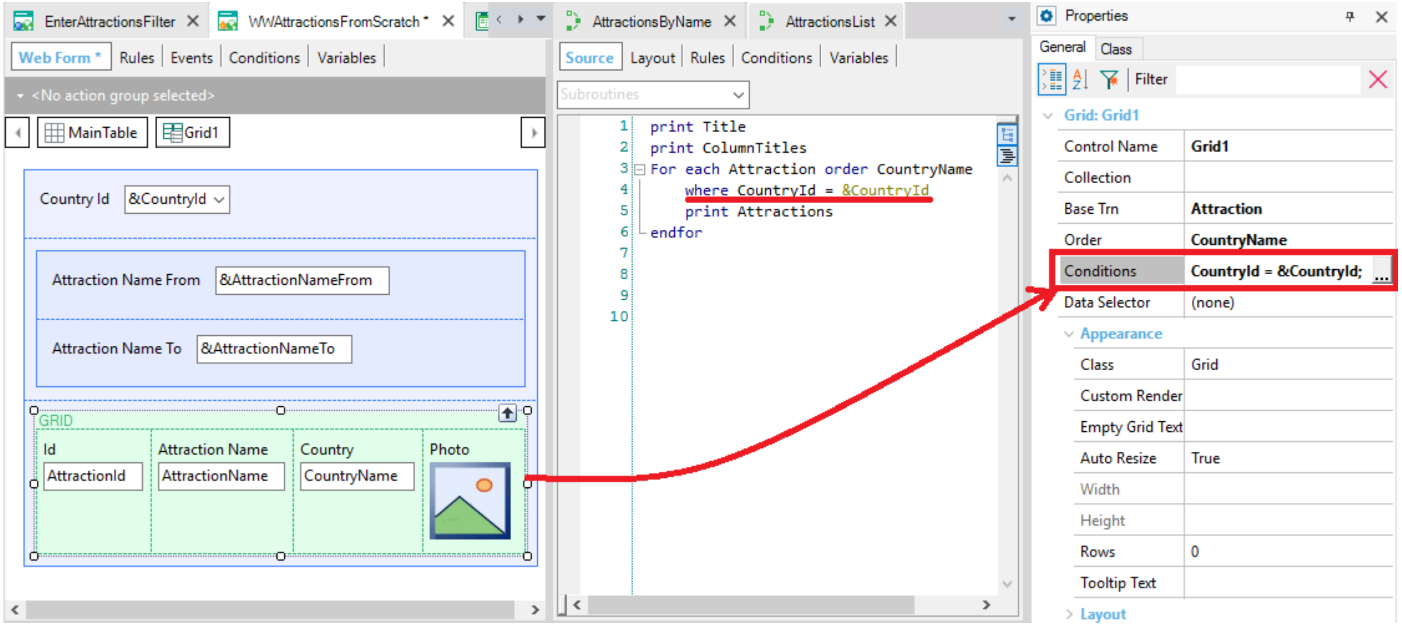
Id	Attraction Name	Country	Photo
AttractionId	AttractionName	CountryName	

Grid: Grid1

Control Name	Grid1
Collection	
Base Trn	Attraction
Order	CountryName
Conditions	CountryId = &CountryId;
Data Selector	(none)
	Grid
	True
Height	
Rows	0
Tooltip Text	

```
CountryId = &CountryId when not &CountryId.IsEmpty();
```

Condições de filtro



Country Id: &CountryId

Attraction Name From: &AttractionNameFrom

Attraction Name To: &AttractionNameTo

GRID

Id	Attraction Name	Country	Photo
AttractionId	AttractionName	CountryName	

```
print Title
print ColumnTitles
For each Attraction order CountryName
    where CountryId = &CountryId
    print Attractions
endfor
```

Grid: Grid1

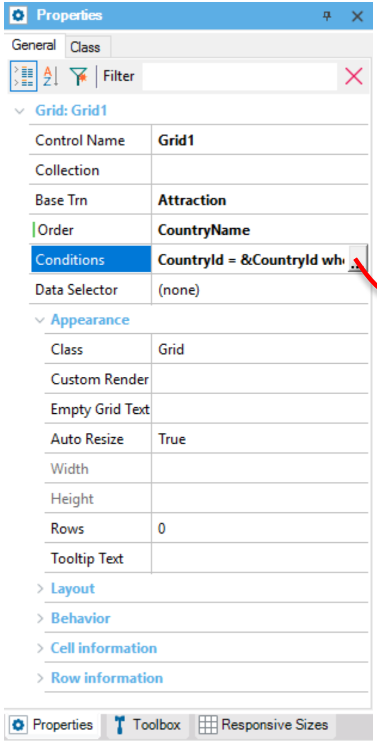
Control Name	Grid1
Collection	
Base Trn	Attraction
Order	CountryName
Conditions	CountryId = &CountryId;
Data Selector	(none)

Appearance

Class	Grid
Custom Render	
Empty Grid Text	
Auto Resize	True
Width	
Height	
Rows	0
Tooltip Text	

Layout

Muitas condições



Grid: Grid1

Control Name	Grid1
Collection	
Base Trn	Attraction
Order	CountryName
Conditions	CountryId = &CountryId whe
Data Selector	(none)

Appearance

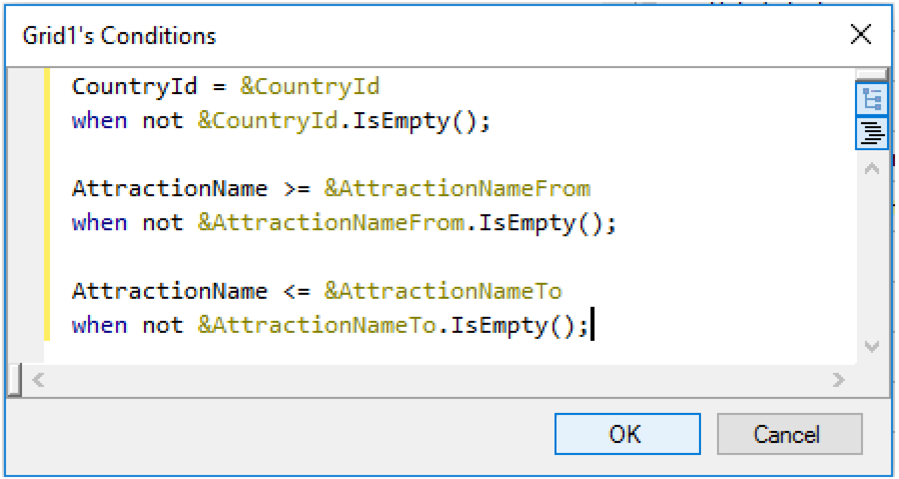
Class	Grid
Custom Render	
Empty Grid Text	
Auto Resize	True
Width	
Height	
Rows	0
Tooltip Text	

Layout

Behavior

Cell information

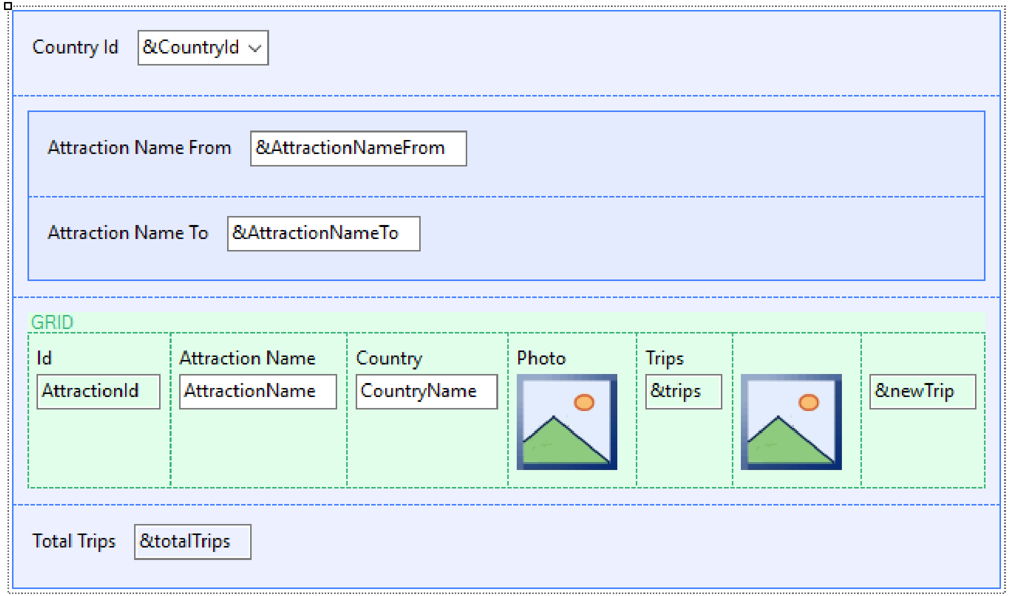
Row information



Grid1's Conditions

```
CountryId = &CountryId
when not &CountryId.IsEmpty();

AttractionName >= &AttractionNameFrom
when not &AttractionNameFrom.IsEmpty();

AttractionName <= &AttractionNameTo
when not &AttractionNameTo.IsEmpty();
```

Eventos



Country Id &CountryId

Attraction Name From &AttractionNameFrom

Attraction Name To &AttractionNameTo

GRID

Id	Attraction Name	Country	Photo	Trips		
AttractionId	AttractionName	CountryName		&trips		&newTrip

Total Trips &totalTrips

First time

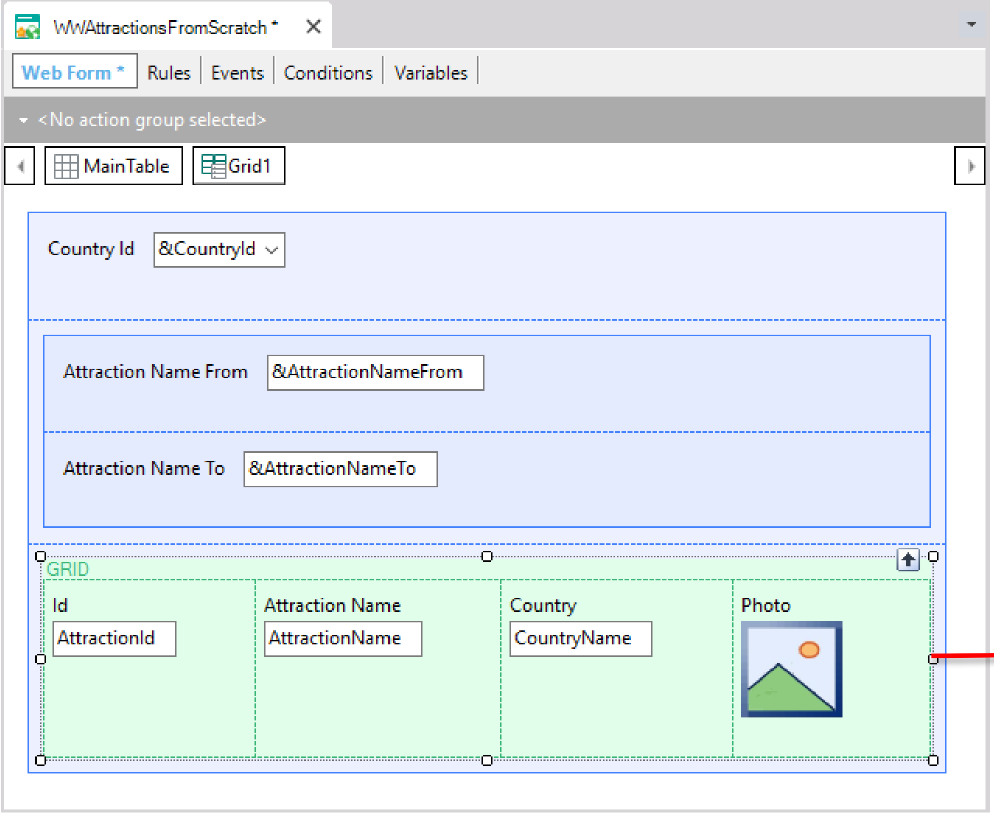
Start

Refresh

Load

User / Control Event

Evento Load em Web Panel COM tabela base



WWAttractionsFromScratch *

Web Form * | Rules | Events | Conditions | Variables

<No action group selected>

MainTable | Grid1

Country Id &CountryId

Attraction Name From &AttractionNameFrom

Attraction Name To &AttractionNameTo

GRID

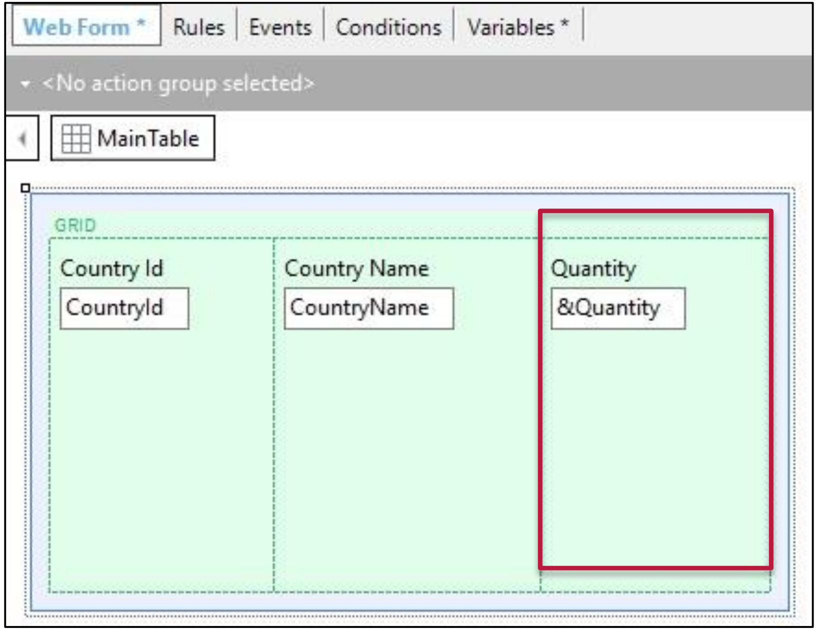
Id	Attraction Name	Country	Photo
AttractionId	AttractionName	CountryName	

Evento LOAD

“N vezes, tantas quantos registros existirem na tabela percorrida”

Outro exemplo

```
Country
{
  CountryId*
  CountryName
}
```

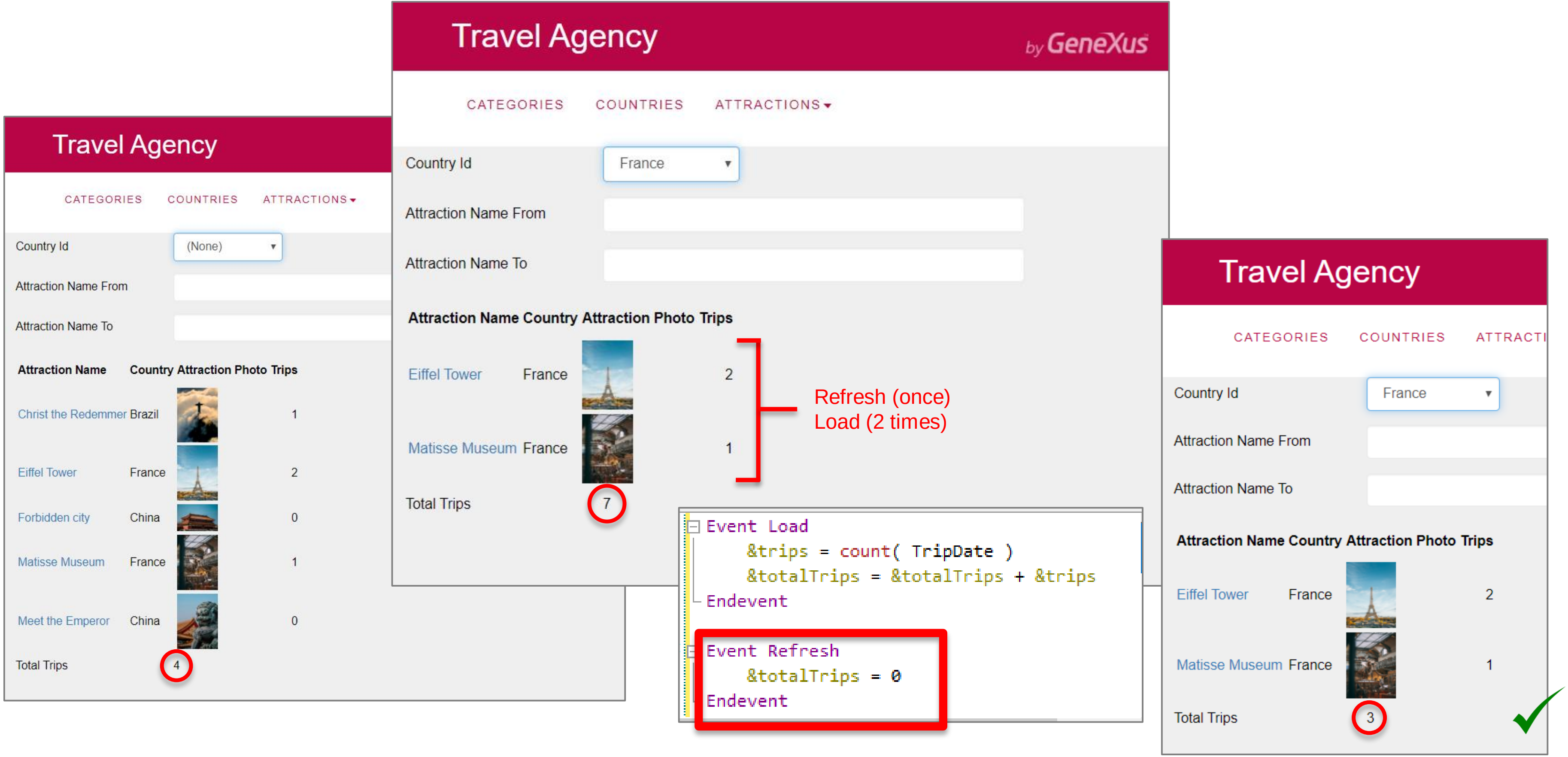


Country Id	Country Name	Quantity
CountryId	CountryName	&Quantity

```
Attraction
{
  AttractionId*
  AttractionName
  CountryId
  CountryName
}
```

```
Event Load
    &Quantity = Count(AttractionName)
endevent
```

Evento Refresh



Travel Agency

CATEGORIES COUNTRIES ATTRACTIONS ▾

Country Id	(None) ▾
Attraction Name From	
Attraction Name To	

Attraction Name	Country	Attraction Photo	Trips
Christ the Redemmer	Brazil		1
Eiffel Tower	France		2
Forbidden city	China		0
Matisse Museum	France		1
Meet the Emperor	China		0

Total Trips 4

Travel Agency by GeneXus

CATEGORIES COUNTRIES ATTRACTIONS ▾

Country Id	France ▾
Attraction Name From	
Attraction Name To	

Attraction Name	Country	Attraction Photo	Trips
Eiffel Tower	France		2
Matisse Museum	France		1

Total Trips 7

Refresh (once)
Load (2 times)

```
Event Load
    &trips = count( TripDate )
    &totalTrips = &totalTrips + &trips
Endevent

Event Refresh
    &totalTrips = 0
Endevent
```

Travel Agency

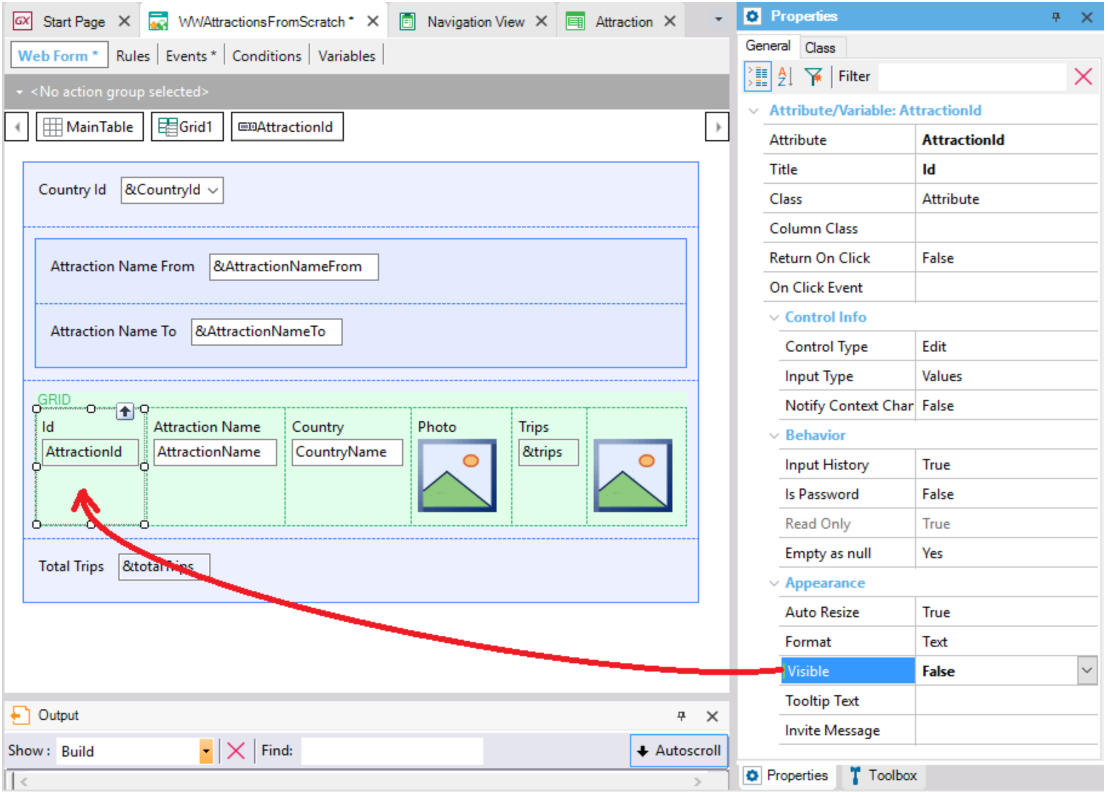
CATEGORIES COUNTRIES ATTRACTI

Country Id	France ▾
Attraction Name From	
Attraction Name To	

Attraction Name	Country	Attraction Photo	Trips
Eiffel Tower	France		2
Matisse Museum	France		1

Total Trips 3 ✓

Atributos na Grid

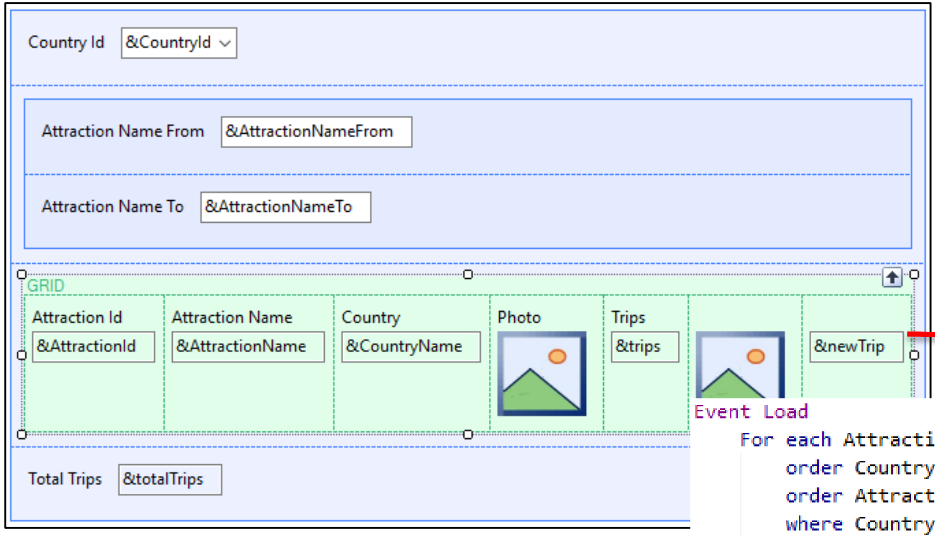


Attribute/Variable: AttractionId

Attribute	AttractionId
Title	Id
Class	Attribute
Column Class	
Return On Click	False
On Click Event	
Control Info	
Control Type	Edit
Input Type	Values
Notify Context Char	False
Behavior	
Input History	True
Is Password	False
Read Only	True
Empty as null	Yes
Appearance	
Auto Resize	True
Format	Text
Visible	False
Tooltip Text	
Invite Message	

Web panels sem tabela base

Web Panels SEM TABELA BASE



Country Id &CountryId

Attraction Name From &AttractionNameFrom

Attraction Name To &AttractionNameTo

Attraction Id	Attraction Name	Country	Photo	Trips		
&AttractionId	&AttractionName	&CountryName		&trips		&newTrip

Total Trips &totalTrips

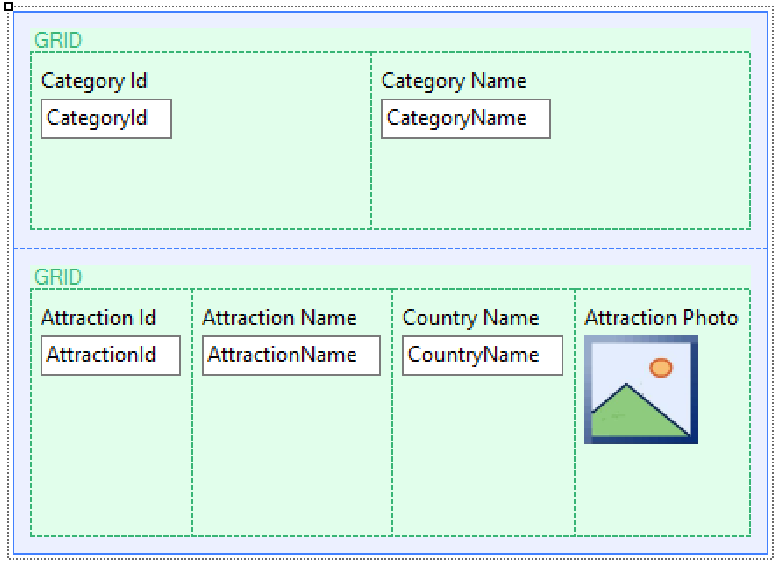
Evento LOAD

"1 vez"

```
Event Load
    For each Attraction
        order CountryId, AttractionName when not &CountryId.IsEmpty()
        order AttractionName
        where CountryId = &CountryId when not &CountryId.IsEmpty()
        where AttractionName >= &AttractionNameFrom when not &AttractionNameFrom.IsEmpty()
        where AttractionName <= &AttractionNameTo when not &AttractionNameTo.IsEmpty()
        &AttractionId = AttractionId
        &AttractionName = AttractionName
        &CountryName = CountryName
        &AttractionPhoto = AttractionPhoto
        &trips = count( TripDate )
        Load
        &totalTrips = &totalTrips + &trips
    endfor
Endevent
```

Web panels
Múltiplas grids

Múltiplas grids



GRID

Category Id	Category Name
CategoryId	CategoryName

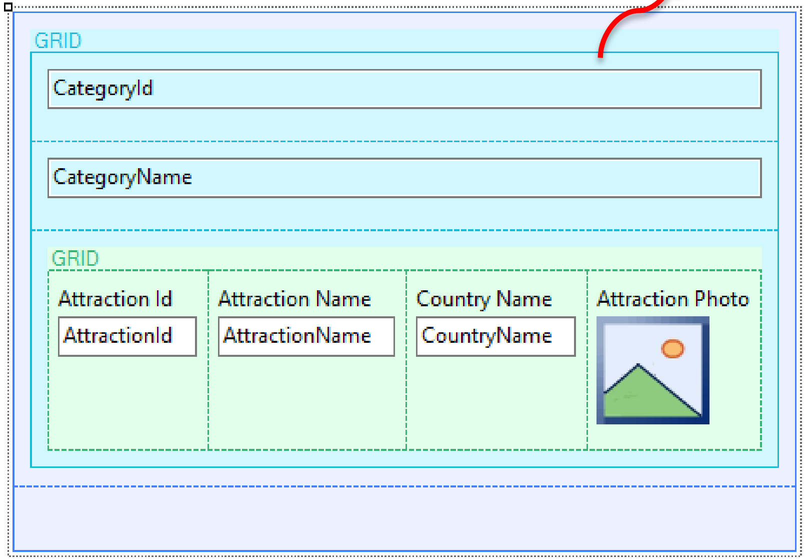
GRID

Attraction Id	Attraction Name	Country Name	Attraction Photo
AttractionId	AttractionName	CountryName	

PARALELO

Navegação independente

FreeStyleGrid



GRID

CategoryId
CategoryName

GRID

Attraction Id	Attraction Name	Country Name	Attraction Photo
AttractionId	AttractionName	CountryName	

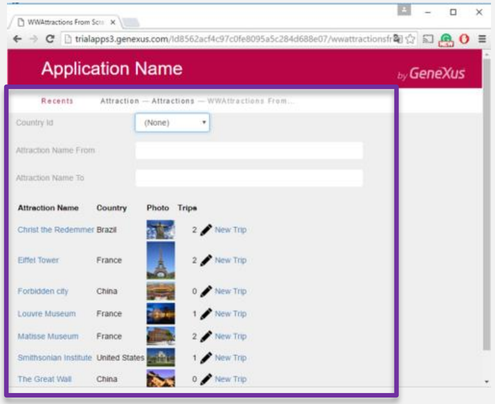
ANINHADO

Navegação com tabelas relacionadas

Tipos de Web Panels

Tipos de Web Panels

Web page (default)



WWAttractions From Scr...

trialapps3.genexus.com/Id8562acf4c97c0fe8095a5c284d688e07/wwattractionsfr...

Application Name by GeneXus

Recents   Attraction — Attractions — WWAttractions From...

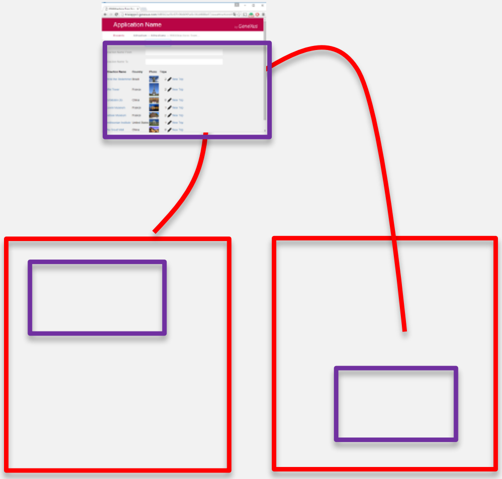
Country Id: (None)

Attraction Name From:

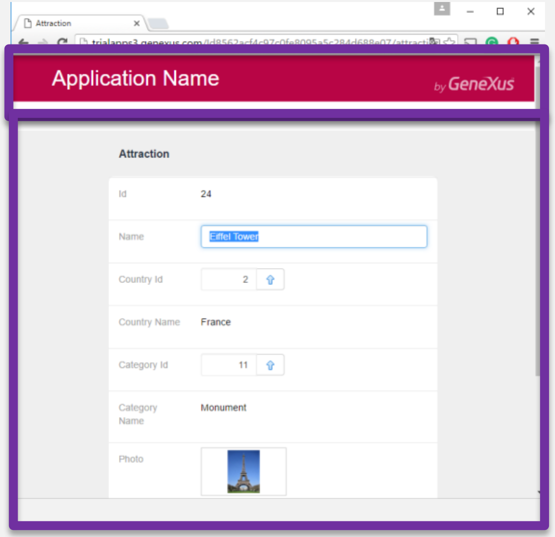
Attraction Name To:

Attraction Name	Country	Photo	Trips	
Christ the Redemmer	Brazil		2	New Trip
Eiffel Tower	France		2	New Trip
Forbidden city	China		0	New Trip
Louvre Museum	France		1	New Trip
Matisse Museum	France		2	New Trip
Smithsonian Institute	United States		1	New Trip
The Great Wall	China		0	New Trip

Component



Master page



Attraction

trialapps3.genexus.com/Id8562acf4c97c0fe8095a5c284d688e07/attracti...

Application Name by GeneXus

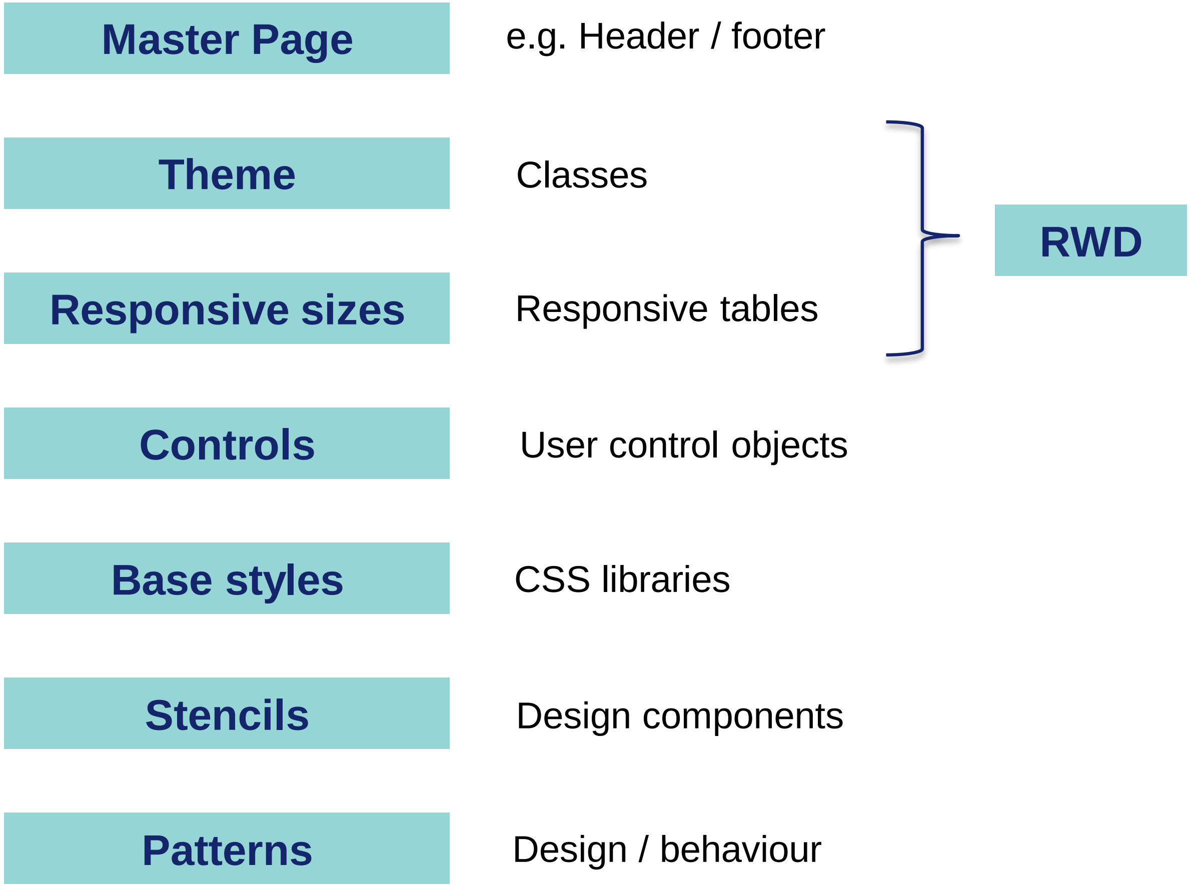
Attraction

Id	24
Name	Eiffel Tower
Country Id	2
Country Name	France
Category Id	11
Category Name	Monument
Photo	

Design Systems

Design Systems

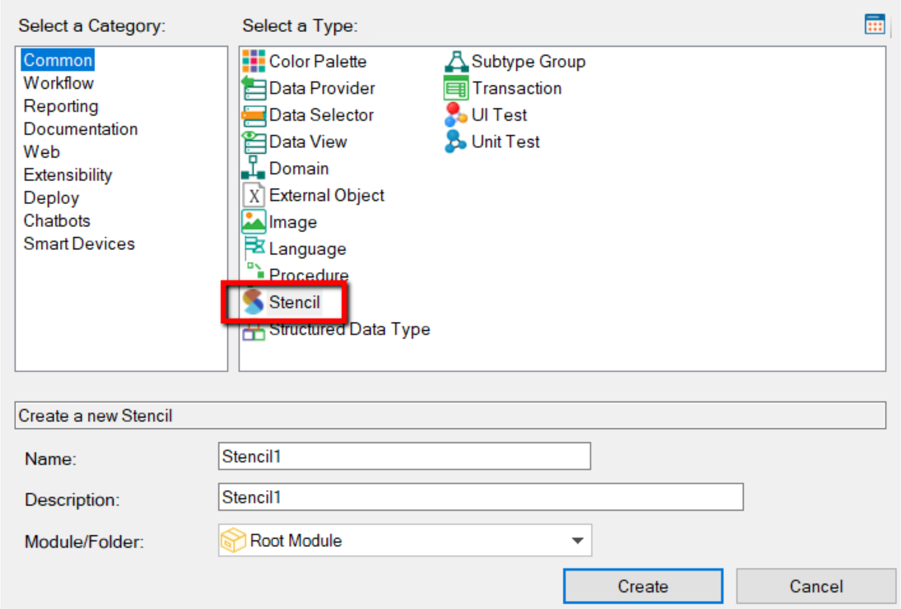
Elementos que dão uniformidade e coerência ao UX



Master Page	e.g. Header / footer
Theme	Classes
Responsive sizes	Responsive tables
Controls	User control objects
Base styles	CSS libraries
Stencils	Design components
Patterns	Design / behaviour

RWD

Stencils



Select a Category:

- Common
- Workflow
- Reporting
- Documentation
- Web
- Extensibility
- Deploy
- Chatbots
- Smart Devices

Select a Type:

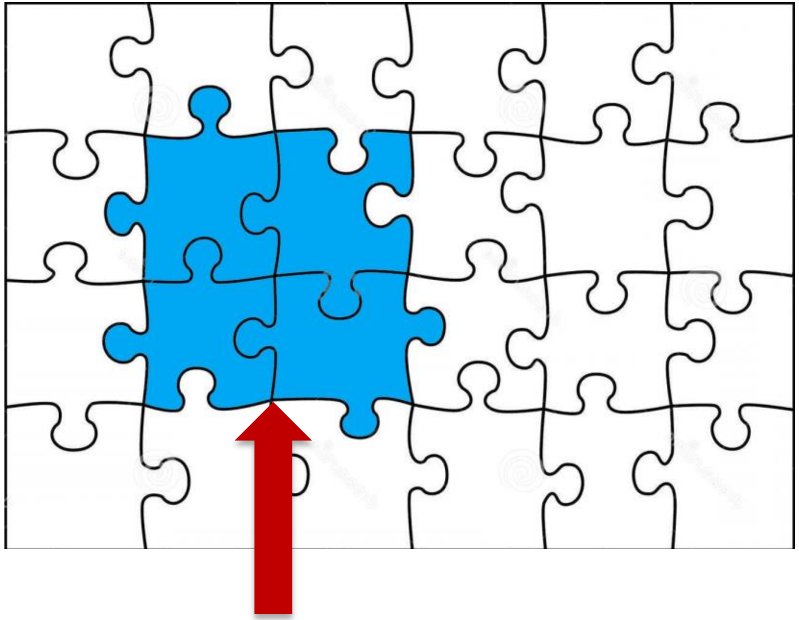
- Color Palette
- Data Provider
- Data Selector
- Data View
- Domain
- External Object
- Image
- Language
- Procedure
- Stencil
- Structured Data Type
- Subtype Group
- Transaction
- UI Test
- Unit Test

Create a new Stencil

Name: Stencil1

Description: Stencil1

Module/Folder: Root Module

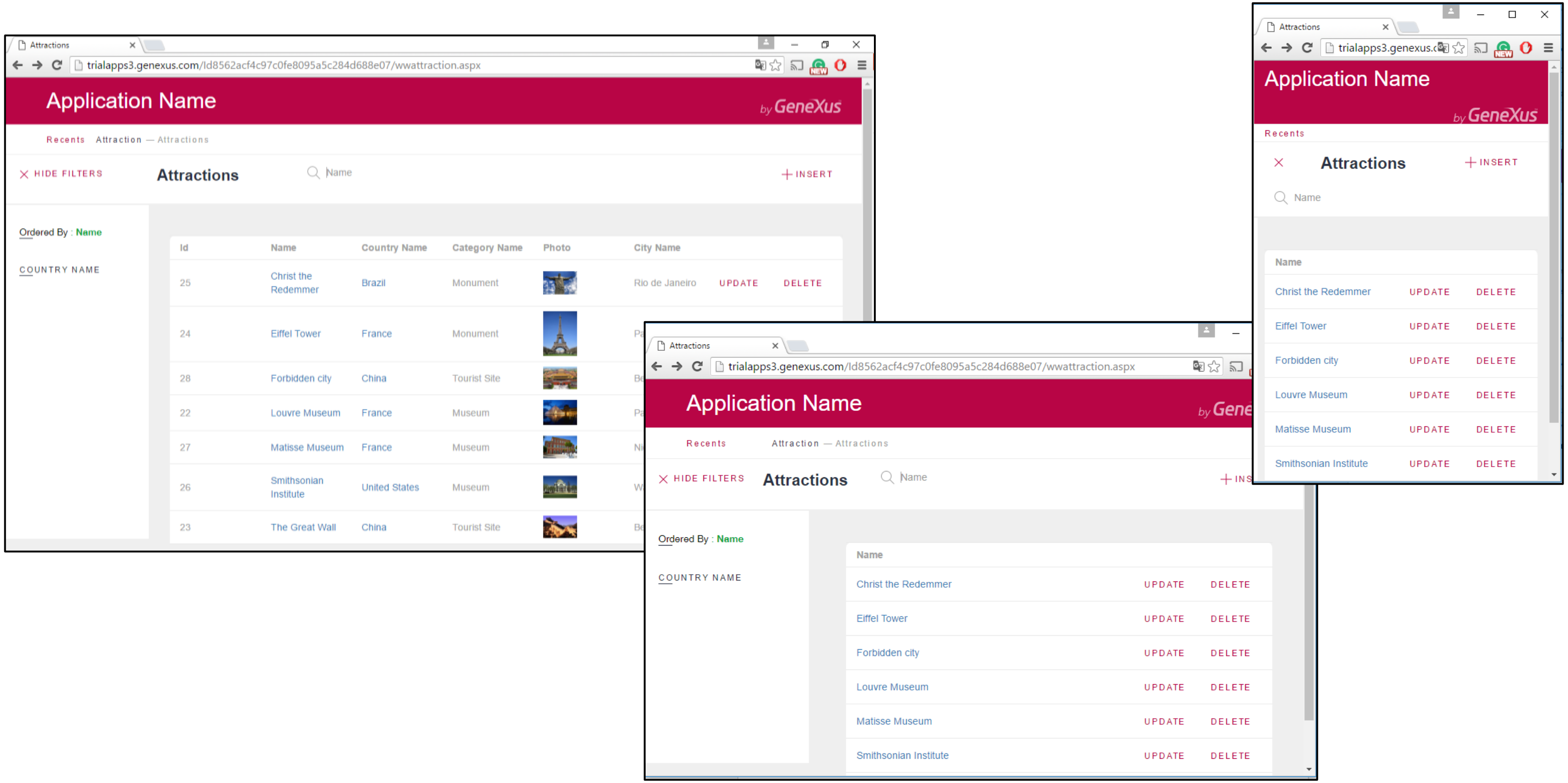


Design component

Objeto que permite repetir o desenho da mesma parte da tela (um conjunto de controles), em várias telas.

Responsive Web Design

Responsive Web Design



Attractions
trialapps3.genexus.com/Id8562acf4c97c0fe8095a5c284d688e07/wwattraction.aspx

Application Name

by GeneXus

Recents Attraction — Attractions

✕ HIDE FILTERS

Attractions

Name

+ INSERT

Ordered By : Name

COUNTRY NAME

Id	Name	Country Name	Category Name	Photo	City Name		
25	Christ the Redemmer	Brazil	Monument		Rio de Janeiro	UPDATE	DELETE
24	Eiffel Tower	France	Monument		Pa		
28	Forbidden city	China	Tourist Site		Be		
22	Louvre Museum	France	Museum		Pa		
27	Matisse Museum	France	Museum		Ni		
26	Smithsonian Institute	United States	Museum		W		
23	The Great Wall	China	Tourist Site		Be		

Attractions
trialapps3.genexus.com/Id8562acf4c97c0fe8095a5c284d688e07/wwattraction.aspx

Application Name

by Gene

Recents Attraction — Attractions

✕ HIDE FILTERS

Attractions

Name

+ INS

Ordered By : Name

COUNTRY NAME

Name		
Christ the Redemmer	UPDATE	DELETE
Eiffel Tower	UPDATE	DELETE
Forbidden city	UPDATE	DELETE
Louvre Museum	UPDATE	DELETE
Matisse Museum	UPDATE	DELETE
Smithsonian Institute	UPDATE	DELETE

Application Name

by GeneXus

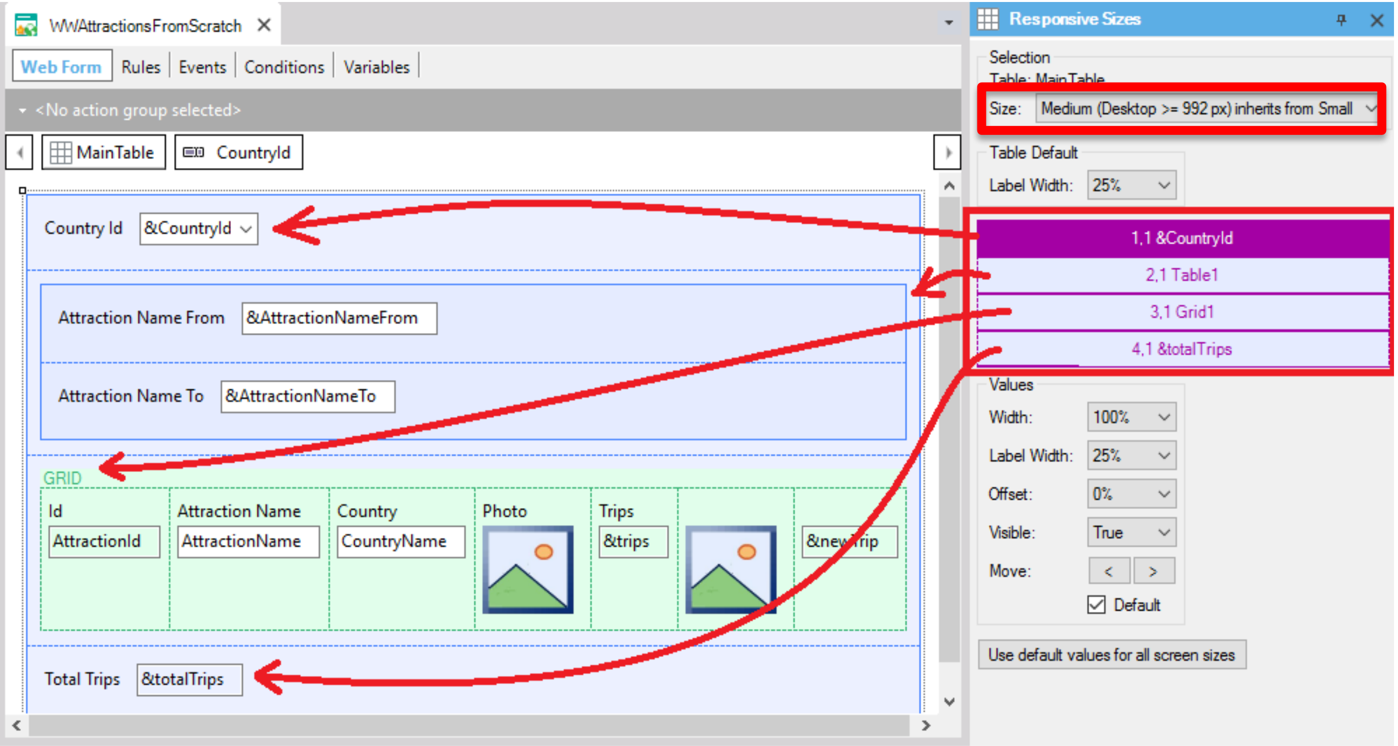
Recents

✕ Attractions + INSERT

Name

Name		
Christ the Redemmer	UPDATE	DELETE
Eiffel Tower	UPDATE	DELETE
Forbidden city	UPDATE	DELETE
Louvre Museum	UPDATE	DELETE
Matisse Museum	UPDATE	DELETE
Smithsonian Institute	UPDATE	DELETE

Responsive Web Design



WWAttractionsFromScratch

Responsive Sizes

Selection

Table: MainTable

Size: Medium (Desktop >= 992 px) inherits from Small

Table Default

Label Width: 25%

1,1 &CountryId

2,1 Table1

3,1 Grid1

4,1 &totalTrips

Values

Width: 100%

Label Width: 25%

Offset: 0%

Visible: True

Move: < >

☑ Default

Use default values for all screen sizes

Country Id: &CountryId

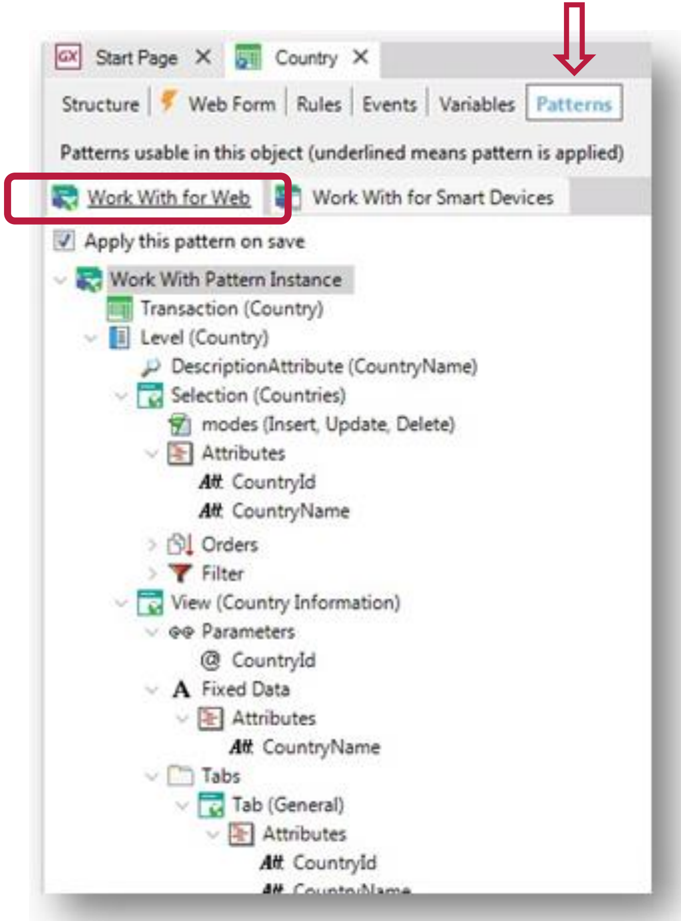
Attraction Name From: &AttractionNameFrom

Attraction Name To: &AttractionNameTo

Id	Attraction Name	Country	Photo	Trips		
AttractionId	AttractionName	CountryName		&trips		&newTrip

Total Trips: &totalTrips

Patterns



Start Page ✕ | Country ✕

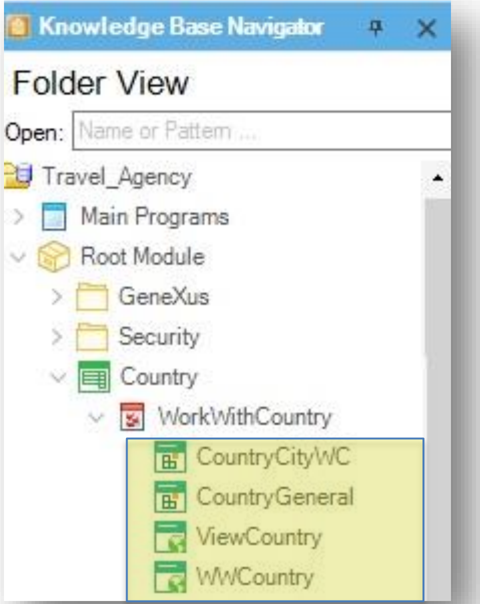
Structure | Web Form | Rules | Events | Variables | Patterns

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

Work With for Web | Work With for Smart Devices

☑ Apply this pattern on save

- Work With Pattern Instance
  - Transaction (Country)
  - Level (Country)
    - DescriptionAttribute (CountryName)
    - Selection (Countries)
      - modes (Insert, Update, Delete)
      - Attributes
        - Att CountryId
        - Att CountryName
      - Orders
      - Filter
    - View (Country Information)
      - Parameters
        - @ CountryId
      - A Fixed Data
        - Attributes
          - Att CountryName
      - Tabs
        - Tab (General)
          - Attributes
            - Att CountryId
            - Att CountryName



Knowledge Base Navigator

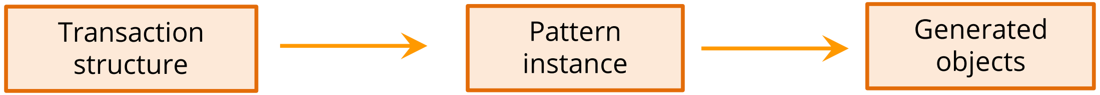
Folder View

Open: Name or Pattern ...

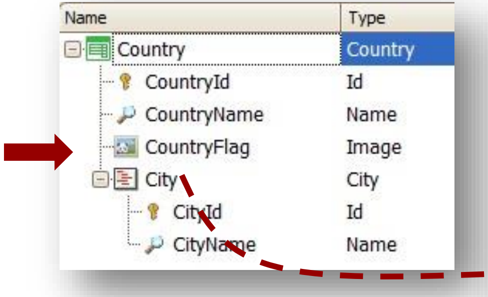
- Travel_Agency
  - Main Programs
  - Root Module
    - GeneXus
    - Security
    - Country
      - WorkWithCountry
        - CountryCityWC
        - CountryGeneral
        - ViewCountry
        - WWCountry

Gerados automaticamente por GeneXus

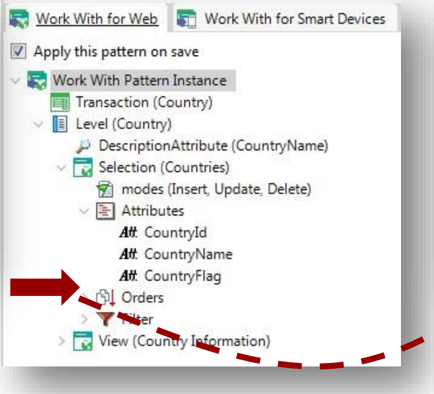
Dinamismo



Example: New attribute CountryFlag



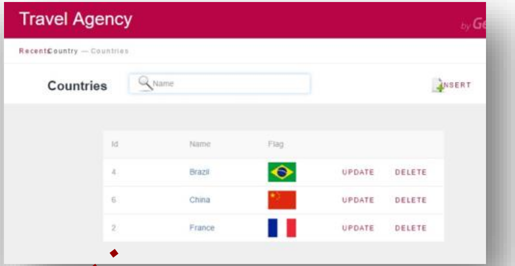
Name	Type
Country	Country
CountryId	Id
CountryName	Name
CountryFlag	Image
City	City
CityId	Id
CityName	Name



Work With for Web | Work With for Smart Devices

☑ Apply this pattern on save

```
Work With Pattern Instance
  Transaction (Country)
  Level (Country)
    DescriptionAttribute (CountryName)
    Selection (Countries)
      modes (Insert, Update, Delete)
      Attributes
        Att CountryId
        Att CountryName
        Att CountryFlag
      Orders
      Filter
    View (Country Information)
```



Travel Agency

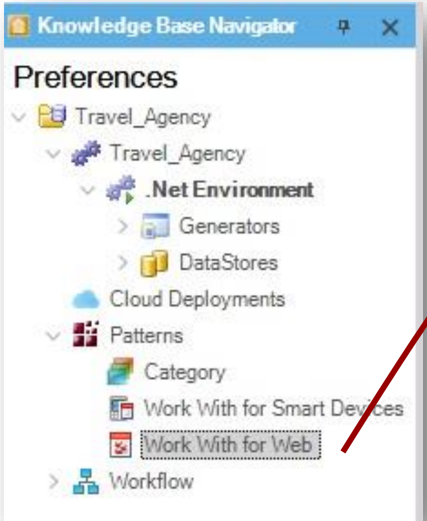
RecentCountry — Countries

Countries [Name] INSERT

Id	Name	Flag		
4	Brazil		UPDATE	DELETE
6	China		UPDATE	DELETE
2	France		UPDATE	DELETE

Configurações Gerais

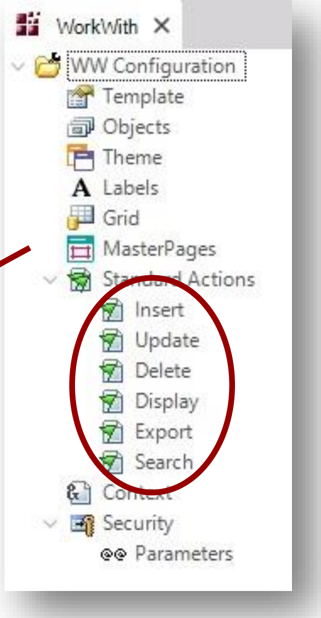
Como são inicializadas todas as instâncias?



Knowledge Base Navigator

Preferences

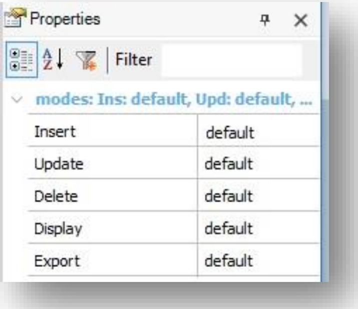
- Travel_Agency
  - Travel_Agency
    - .Net Environment
      - Generators
      - DataStores
    - Cloud Deployments
    - Patterns
      - Category
      - Work With for Smart Devices
      - Work With for Web
    - Workflow



WorkWith

- WW Configuration
  - Template
  - Objects
  - Theme
  - Labels
  - Grid
  - MasterPages
  - Standard Actions
    - Insert
    - Update
    - Delete
    - Display
    - Export
    - Search
  - Context
  - Security
    - Parameters

Em Country...



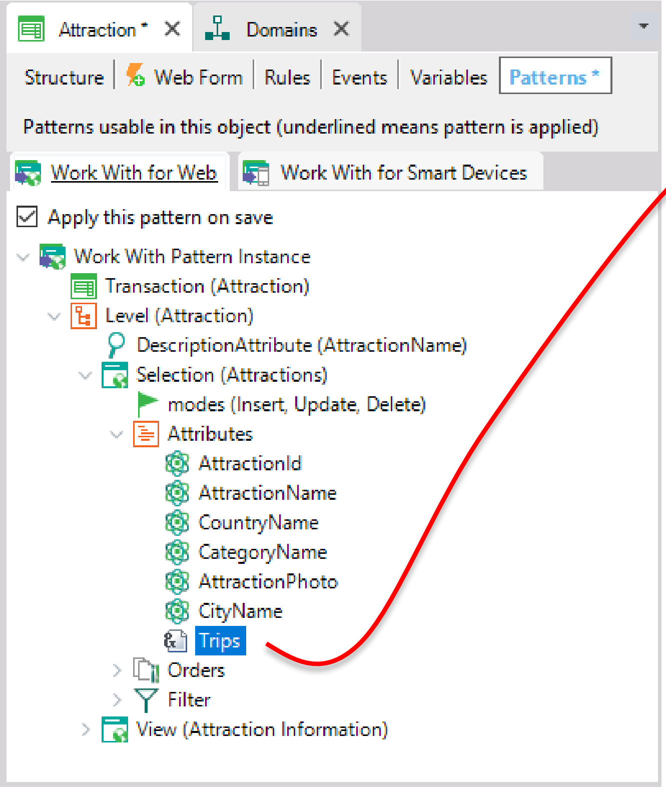
Properties

Filter

modes: Ins: default, Upd: default, ...

Insert	default
Update	default
Delete	default
Display	default
Export	default

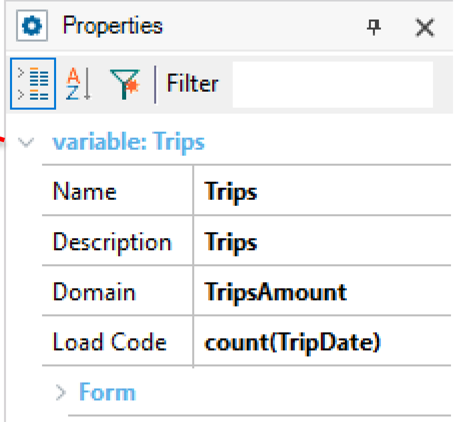
Adicionar Variáveis ou atributos



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

☑ Apply this pattern on save

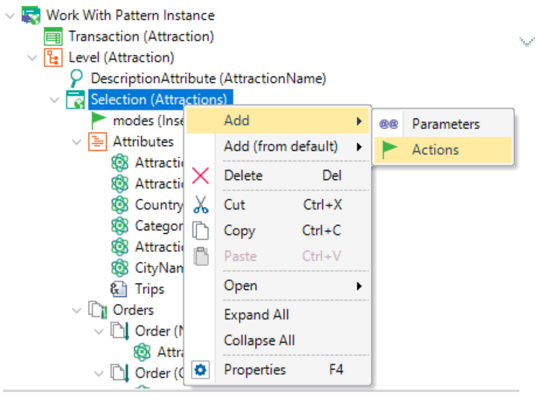
- Work With Pattern Instance
  - Transaction (Attraction)
  - Level (Attraction)
    - DescriptionAttribute (AttractionName)
    - Selection (Attractions)
      - modes (Insert, Update, Delete)
      - Attributes
        - AttractionId
        - AttractionName
        - CountryName
        - CategoryName
        - AttractionPhoto
        - CityName
        - Trips
      - Orders
      - Filter
    - View (Attraction Information)

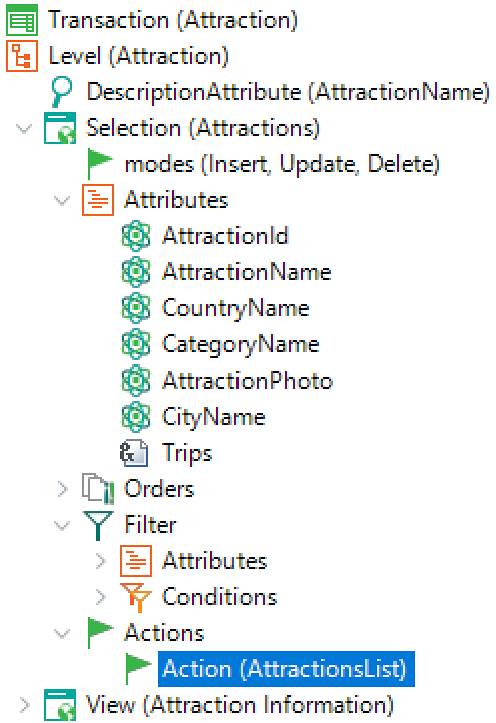


variable: Trips

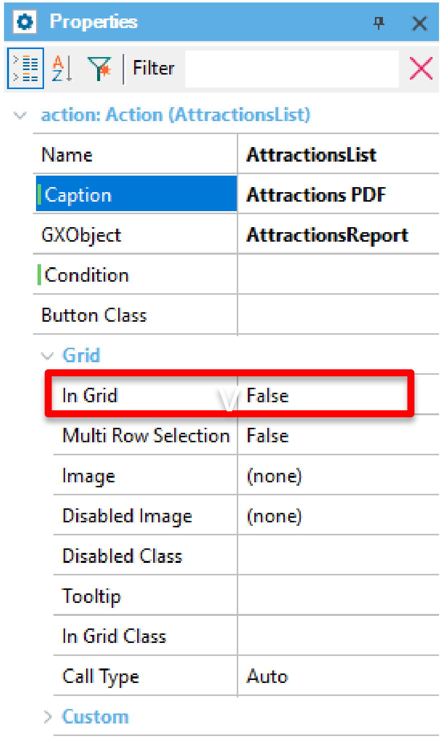
Name	Trips
Description	Trips
Domain	TripsAmount
Load Code	count(TripDate)

Adicionar ações





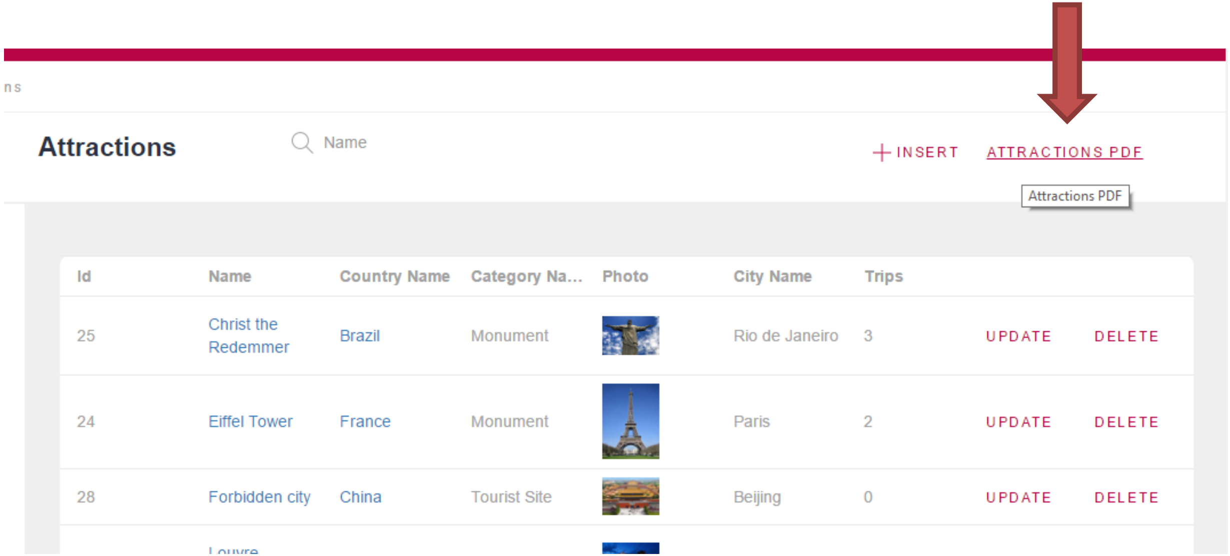
- Transaction (Attraction)
- Level (Attraction)
  - DescriptionAttribute (AttractionName)
  - Selection (Attractions)
    - modes (Insert, Update, Delete)
    - Attributes
      - AttractionId
      - AttractionName
      - CountryName
      - CategoryName
      - AttractionPhoto
      - CityName
      - Trips
    - Orders
    - Filter
      - Attributes
      - Conditions
    - Actions
      - Action (AttractionsList)
  - View (Attraction Information)



action: Action (AttractionsList)

Name	AttractionsList
Caption	Attractions PDF
GXObject	AttractionsReport
Condition	
Button Class	
Grid	
In Grid	False
Multi Row Selection	False
Image	(none)
Disabled Image	(none)
Disabled Class	
Tooltip	
In Grid Class	
Call Type	Auto
Custom	

Adicionar ações



ns

Attractions

Name

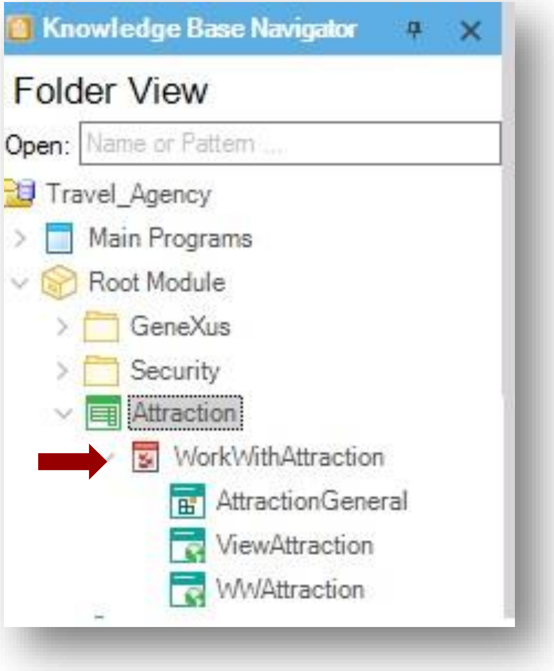
+INSERT

ATTRACTIONS PDF

Attractions PDF

Id	Name	Country Name	Category Na...	Photo	City Name	Trips		
25	Christ the Redemmer	Brazil	Monument		Rio de Janeiro	3	UPDATE	DELETE
24	Eiffel Tower	France	Monument		Paris	2	UPDATE	DELETE
28	Forbidden city	China	Tourist Site		Beijing	0	UPDATE	DELETE
	Louvre							

Exclusão da aplicação do pattern

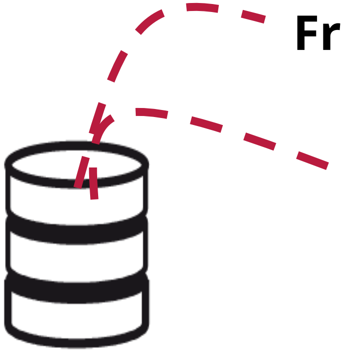


Delete

Consultas Dinâmicas

Objeto Query

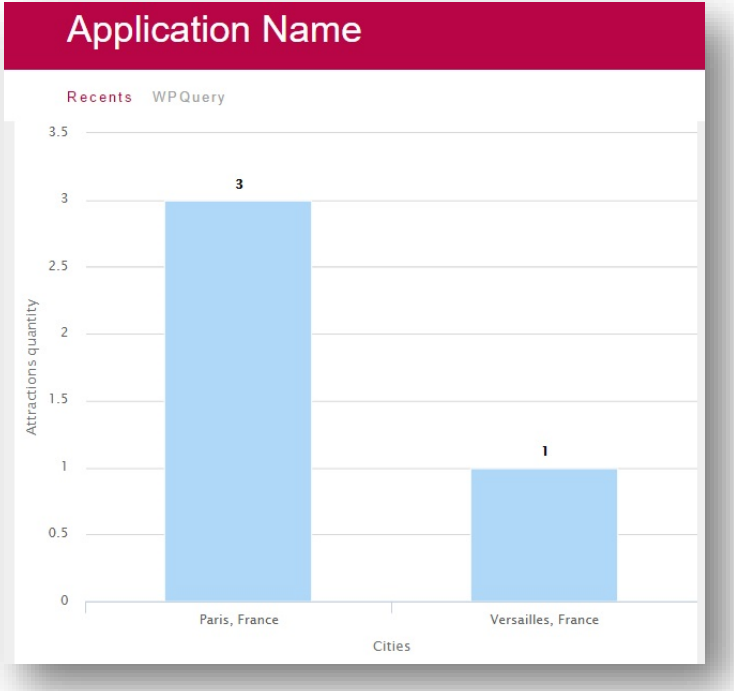
Ver a quantidade de atrações da França, por cidade



France

Paris (3)

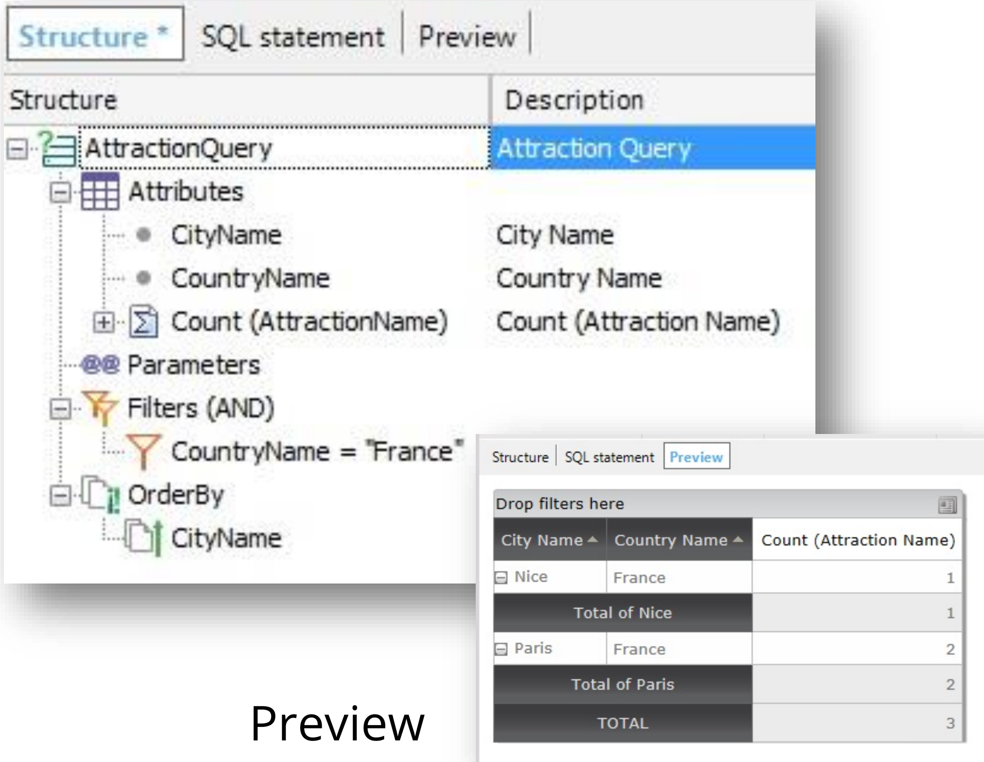
Versailles (1)



Ver a quantidade de atrações da França, por cidade

```
Country
{
  CountryId*
  CountryName
  City
  {
     CityId*
     CityName
   }
}
```

```
Attraction
{
  AttractionId*
  AttractionName
  CountryId
  CountryName
  CityId
  CityName
}
```

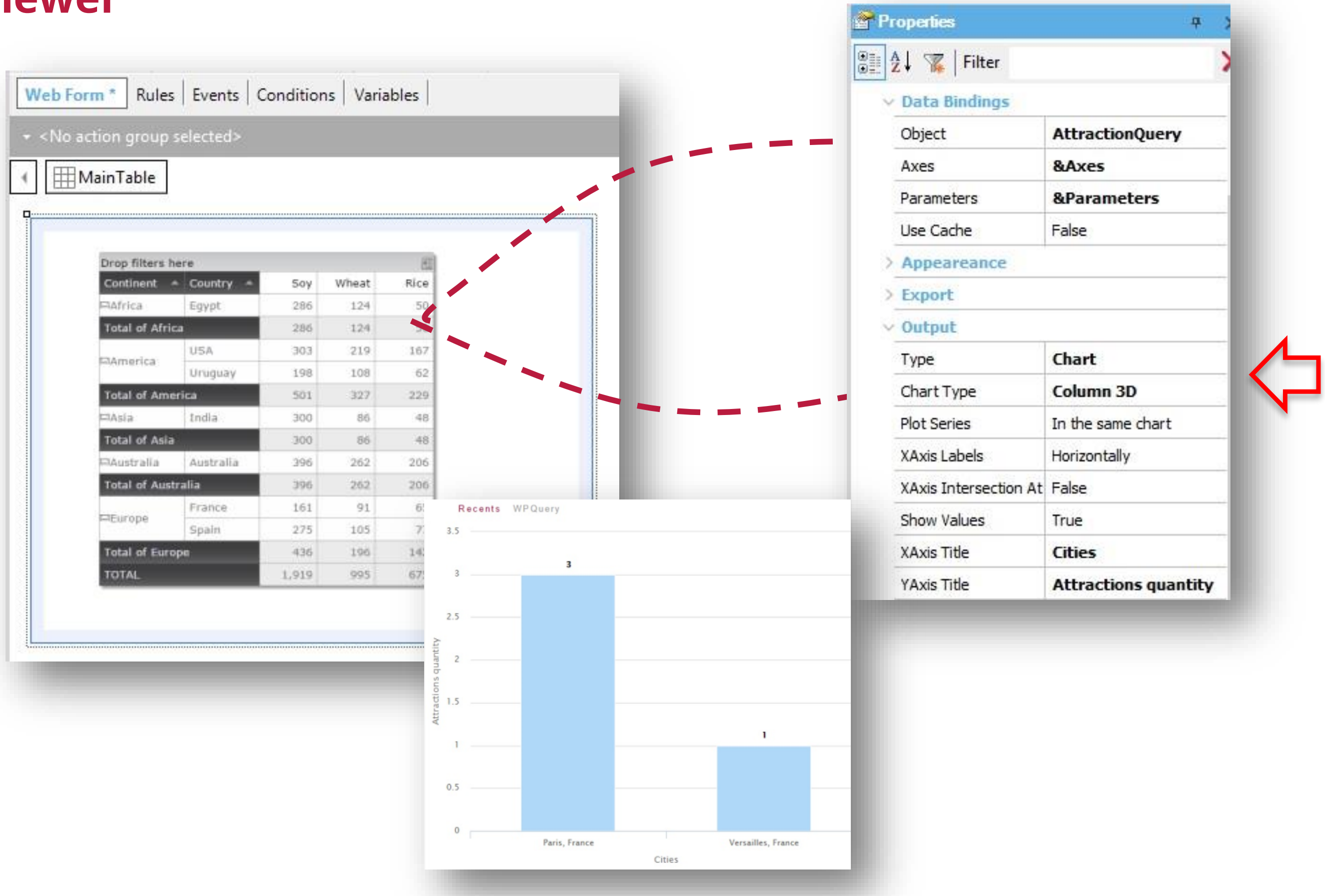


Structure	Description
AttractionQuery	Attraction Query
Attributes	
CityName	City Name
CountryName	Country Name
Count (AttractionName)	Count (Attraction Name)
Parameters	
Filters (AND)	
CountryName = "France"	
OrderBy	
CityName	

City Name ▲	Country Name ▲	Count (Attraction Name)
⊟ Nice	France	1
Total of Nice		1
⊟ Paris	France	2
Total of Paris		2
TOTAL		3

Preview

Query viewer



Drop filters here

Continent	Country	Soy	Wheat	Rice
Africa	Egypt	286	124	50
Total of Africa		286	124	5
America	USA	303	219	167
	Uruguay	198	108	62
Total of America		501	327	229
Asia	India	300	86	48
Total of Asia		300	86	48
Australia	Australia	396	262	206
Total of Australia		396	262	206
Europe	France	161	91	6
	Spain	275	105	7
Total of Europe		436	196	14
TOTAL		1,919	995	67

Data Bindings

Object	AttractionQuery
Axes	&Axes
Parameters	&Parameters
Use Cache	False

Appearance

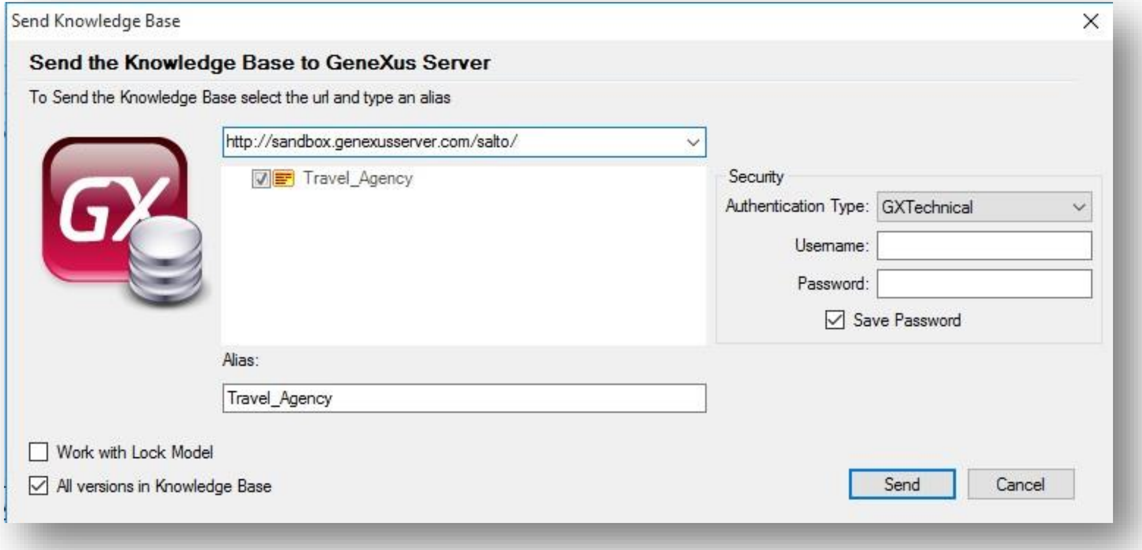
Export

Output

Type	Chart
Chart Type	Column 3D
Plot Series	In the same chart
XAxis Labels	Horizontally
XAxis Intersection At	False
Show Values	True
XAxis Title	Cities
YAxis Title	Attractions quantity

GeneXus Server

Send Knowledge Base to GeneXus Server



Send Knowledge Base

Send the Knowledge Base to GeneXus Server

To Send the Knowledge Base select the url and type an alias

http://sandbox.genexusserver.com/salto/

☑ Travel_Agency

Alias: Travel_Agency

Security

Authentication Type: GXTechnical

Username:

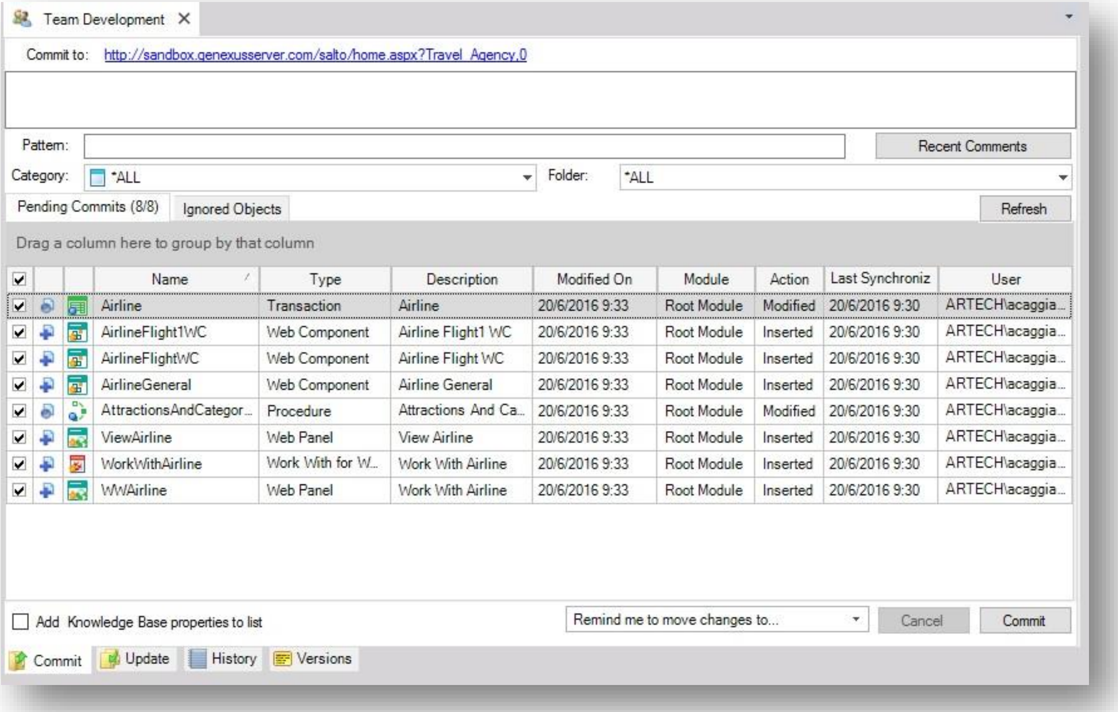
Password:

☑ Save Password

□ Work with Lock Model

☑ All versions in Knowledge Base

Team Development



Commit to: http://sandbox.genexusserver.com/salto/home.aspx?Travel_Agency,0

Pattern:

Category: *ALL

Folder: *ALL

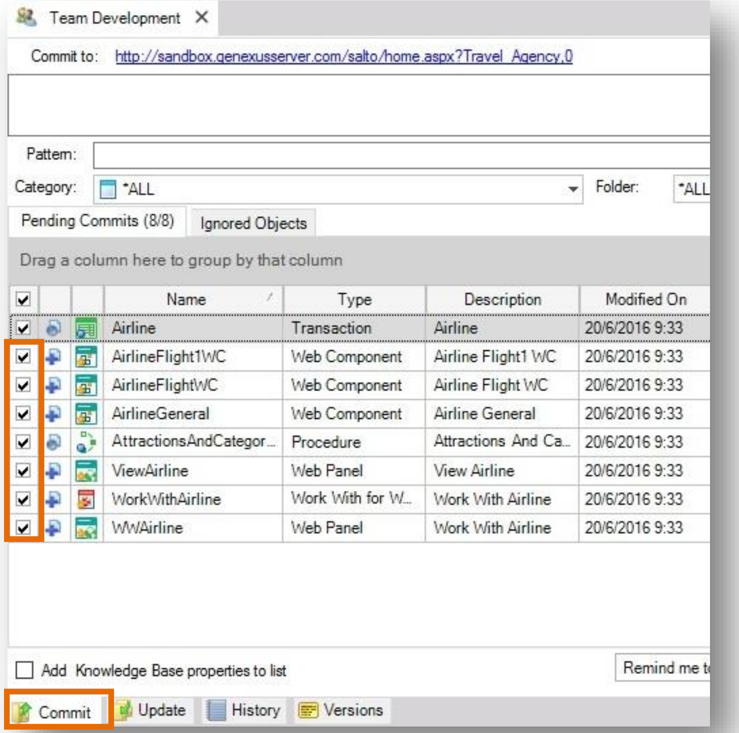
☑			Name	Type	Description	Modified On	Module	Action	Last Synchroniz	User
☑			Airline	Transaction	Airline	20/6/2016 9:33	Root Module	Modified	20/6/2016 9:30	ARTECH\acaggia...
☑			AirlineFlight1WC	Web Component	Airline Flight1 WC	20/6/2016 9:33	Root Module	Inserted	20/6/2016 9:30	ARTECH\acaggia...
☑			AirlineFlightWC	Web Component	Airline Flight WC	20/6/2016 9:33	Root Module	Inserted	20/6/2016 9:30	ARTECH\acaggia...
☑			AirlineGeneral	Web Component	Airline General	20/6/2016 9:33	Root Module	Inserted	20/6/2016 9:30	ARTECH\acaggia...
☑			AttractionsAndCategor...	Procedure	Attractions And Ca...	20/6/2016 9:33	Root Module	Modified	20/6/2016 9:30	ARTECH\acaggia...
☑			ViewAirline	Web Panel	View Airline	20/6/2016 9:33	Root Module	Inserted	20/6/2016 9:30	ARTECH\acaggia...
☑			WorkWithAirline	Work With for W...	Work With Airline	20/6/2016 9:33	Root Module	Inserted	20/6/2016 9:30	ARTECH\acaggia...
☑			WWAirline	Web Panel	Work With Airline	20/6/2016 9:33	Root Module	Inserted	20/6/2016 9:30	ARTECH\acaggia...

☐ Add Knowledge Base properties to list

Remind me to move changes to...

Commit

Knowledge Manager / Team Development



Team Development ✕

Commit to: http://sandbox.genexusserver.com/salto/home.aspx?Travel_Agency,0

Pattern:

Category: *ALL

Folder: *ALL

Pending Commits (8/8)

Ignored Objects

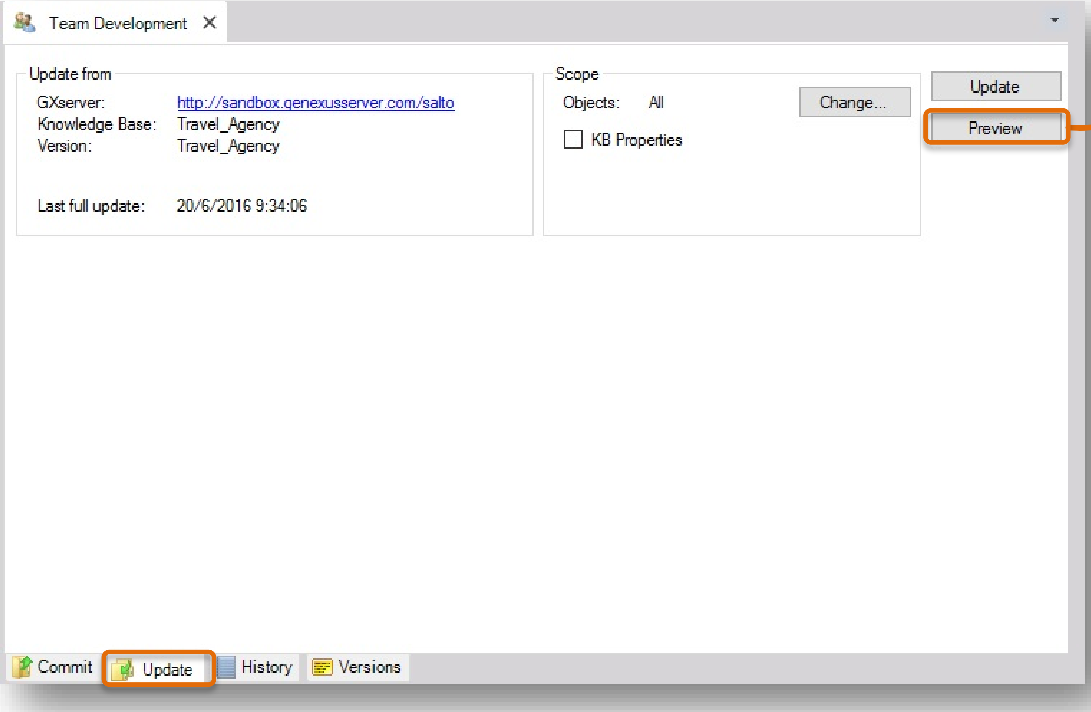
Drag a column here to group by that column

☑	Name	Type	Description	Modified On
☑	Airline	Transaction	Airline	20/6/2016 9:33
☑	AirlineFlight1WC	Web Component	Airline Flight1 WC	20/6/2016 9:33
☑	AirlineFlightWC	Web Component	Airline Flight WC	20/6/2016 9:33
☑	AirlineGeneral	Web Component	Airline General	20/6/2016 9:33
☑	AttractionsAndCategor...	Procedure	Attractions And Ca...	20/6/2016 9:33
☑	ViewAirline	Web Panel	View Airline	20/6/2016 9:33
☑	WorkWithAirline	Work With for W...	Work With Airline	20/6/2016 9:33
☑	WWAirline	Web Panel	Work With Airline	20/6/2016 9:33

□ Add Knowledge Base properties to list

Partial commit

Update



Update from

GXserver: http://sandbox.genexusserver.com/salto
Knowledge Base: Travel_Agency
Version: Travel_Agency

Last full update: 20/6/2016 9:34:06

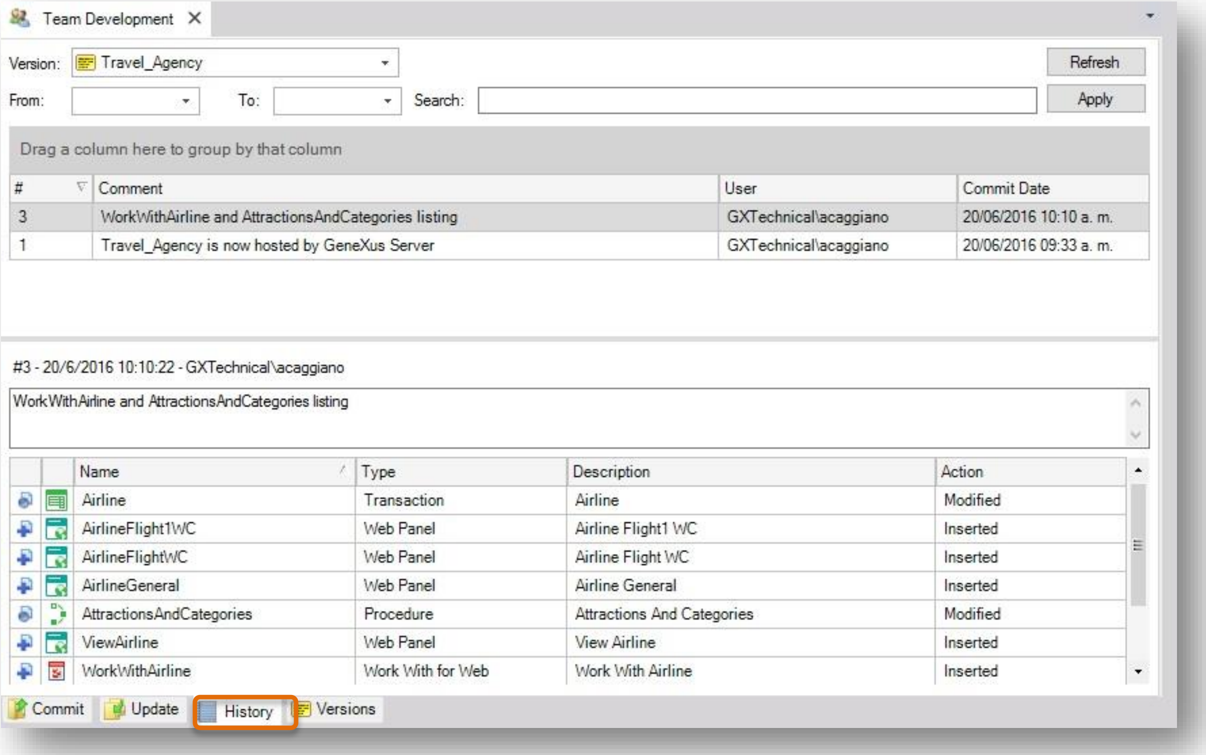
Scope

Objects: All

□ KB Properties

Partial selection

History



Version: Travel_Agency

From: To: Search:

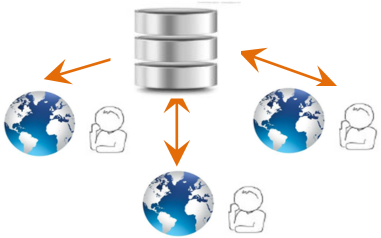
#	Comment	User	Commit Date
3	WorkWithAirline and AttractionsAndCategories listing	GXTechnical\acaggiano	20/06/2016 10:10 a. m.
1	Travel_Agency is now hosted by GeneXus Server	GXTechnical\acaggiano	20/06/2016 09:33 a. m.

#3 - 20/6/2016 10:10:22 - GXTechnical\acaggiano

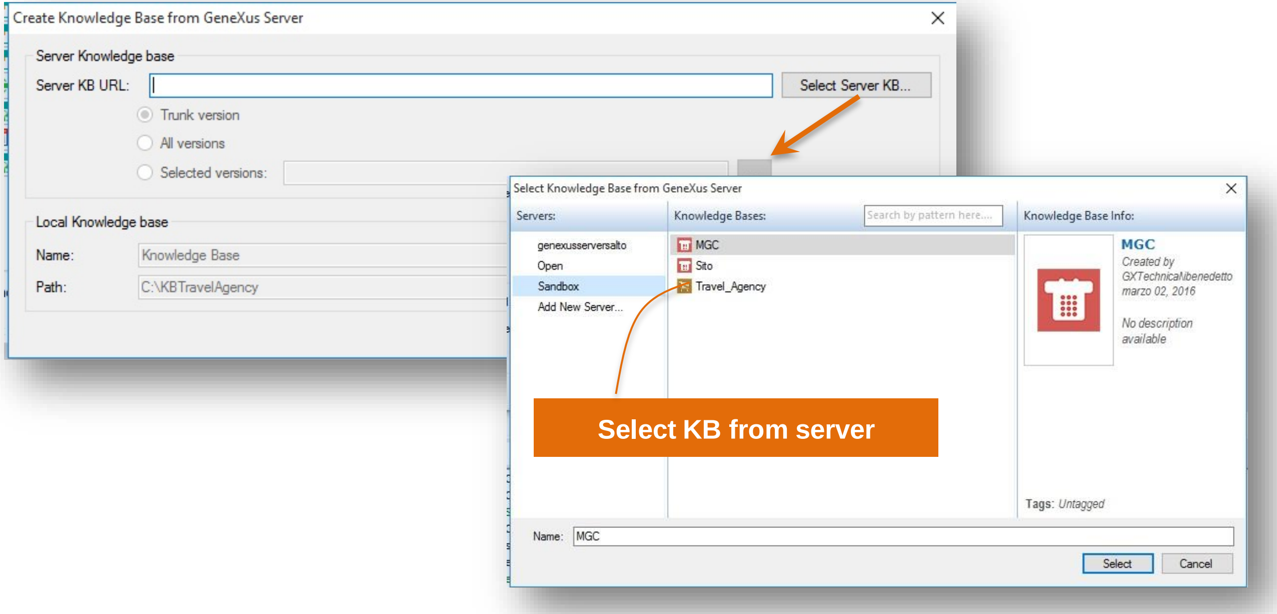
WorkWithAirline and AttractionsAndCategories listing

Name	Type	Description	Action
Airline	Transaction	Airline	Modified
AirlineFlight1WC	Web Panel	Airline Flight1 WC	Inserted
AirlineFlightWC	Web Panel	Airline Flight WC	Inserted
AirlineGeneral	Web Panel	Airline General	Inserted
AttractionsAndCategories	Procedure	Attractions And Categories	Modified
ViewAirline	Web Panel	View Airline	Inserted
WorkWithAirline	Work With for Web	Work With Airline	Inserted

Create KB from GeneXus Server



File / New / Knowledge Base from Server



Create Knowledge Base from GeneXus Server

Server Knowledge base

Server KB URL:

Select Server KB...

● Trunk version

○ All versions

○ Selected versions:

Local Knowledge base

Name: Knowledge Base

Path: C:\KBTravelAgency

Select Knowledge Base from GeneXus Server

Servers:
- genexusserversalto
- Open
- Sandbox
- Add New Server...

Knowledge Bases: Search by pattern here....
- MGC
- Sito
- Travel_Agency

Knowledge Base Info:

MGC

Created by GXTechnicaNbenedetto marzo 02, 2016

No description available

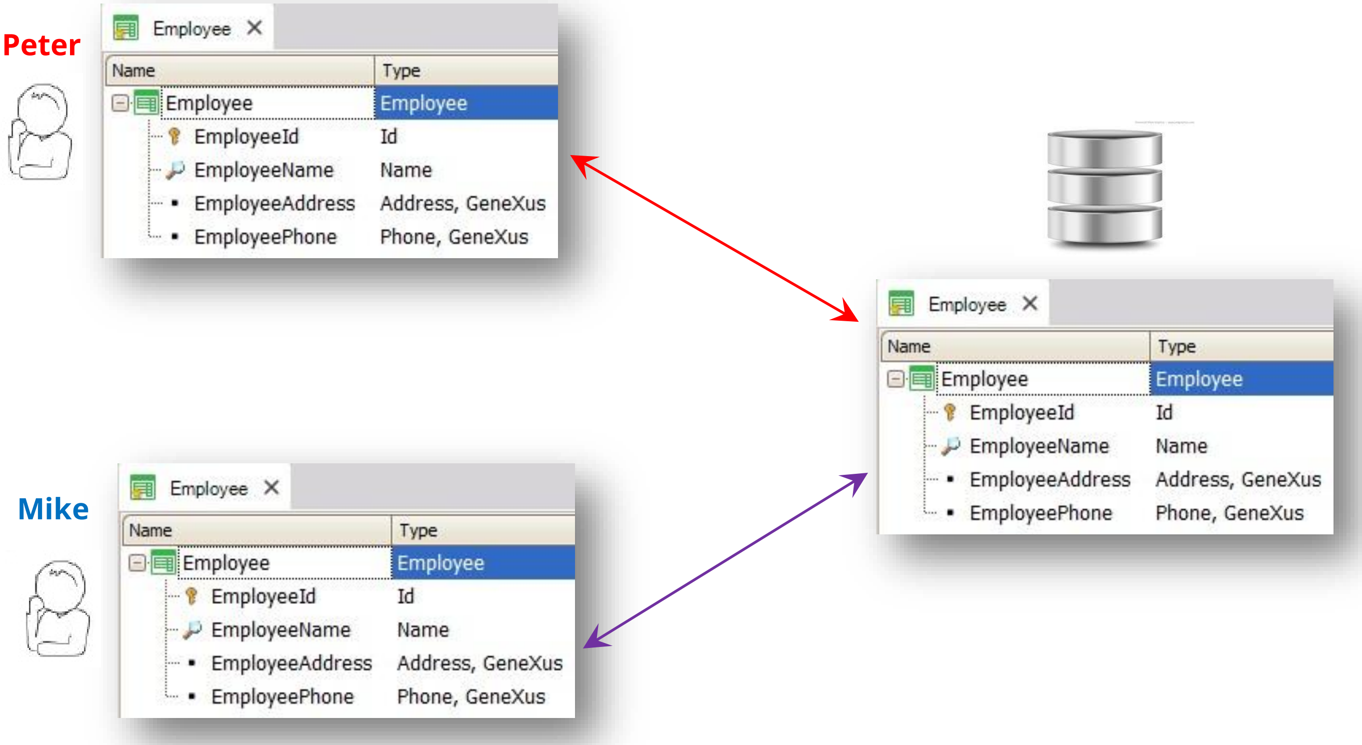
Tags: *Untagged*

Name: MGC

Select | Cancel

Select KB from server

Exemplo



Peter

Employee ×

Name	Type
Employee	Employee
EmployeeId	Id
EmployeeName	Name
EmployeeAddress	Address, GeneXus
EmployeePhone	Phone, GeneXus

Mike

Employee ×

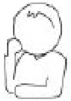
Name	Type
Employee	Employee
EmployeeId	Id
EmployeeName	Name
EmployeeAddress	Address, GeneXus
EmployeePhone	Phone, GeneXus

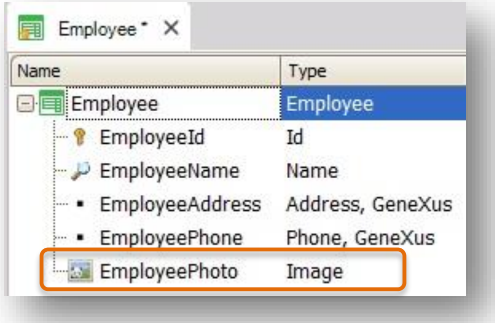
Employee ×

Name	Type
Employee	Employee
EmployeeId	Id
EmployeeName	Name
EmployeeAddress	Address, GeneXus
EmployeePhone	Phone, GeneXus

Exemplo

Peter

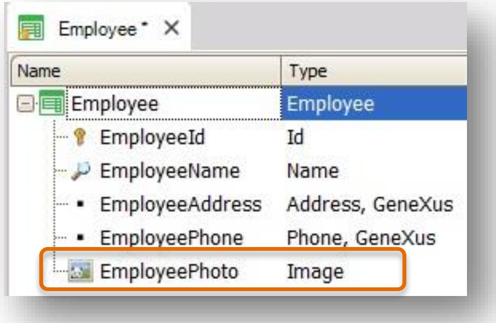




Employee * ✕

Name	Type
Employee	Employee
EmployeeId	Id
EmployeeName	Name
EmployeeAddress	Address, GeneXus
EmployeePhone	Phone, GeneXus
EmployeePhoto	Image

Commit

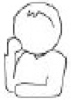


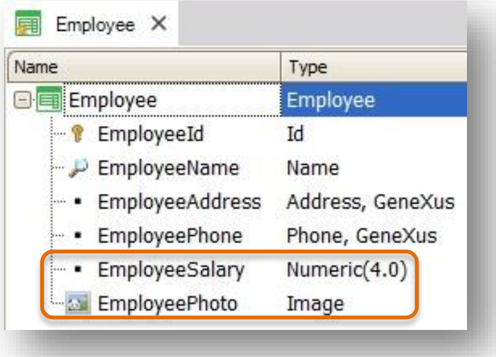
Employee * ✕

Name	Type
Employee	Employee
EmployeeId	Id
EmployeeName	Name
EmployeeAddress	Address, GeneXus
EmployeePhone	Phone, GeneXus
EmployeePhoto	Image

Update

Mike



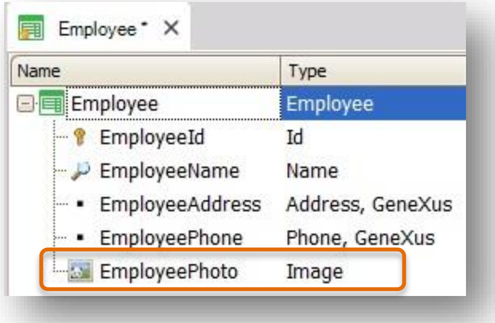


Employee ✕

Name	Type
Employee	Employee
EmployeeId	Id
EmployeeName	Name
EmployeeAddress	Address, GeneXus
EmployeePhone	Phone, GeneXus
EmployeeSalary	Numeric(4.0)
EmployeePhoto	Image

Exemplo

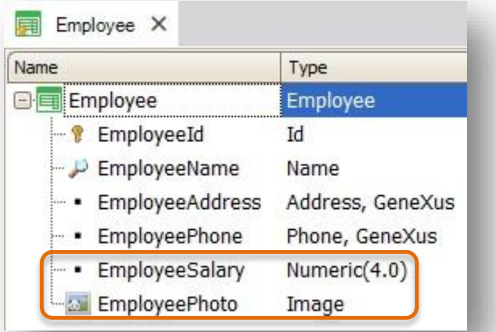
Peter



Employee * ✕

Name	Type
Employee	Employee
EmployeeId	Id
EmployeeName	Name
EmployeeAddress	Address, GeneXus
EmployeePhone	Phone, GeneXus
EmployeePhoto	Image

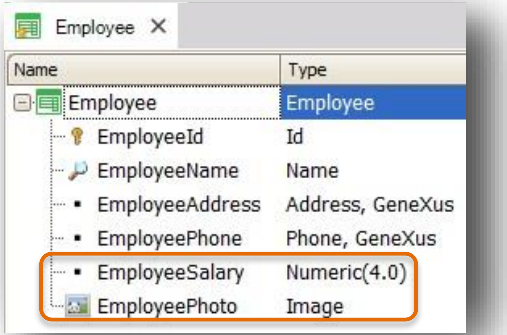
Mike



Employee ✕

Name	Type
Employee	Employee
EmployeeId	Id
EmployeeName	Name
EmployeeAddress	Address, GeneXus
EmployeePhone	Phone, GeneXus
EmployeeSalary	Numeric(4.0)
EmployeePhoto	Image

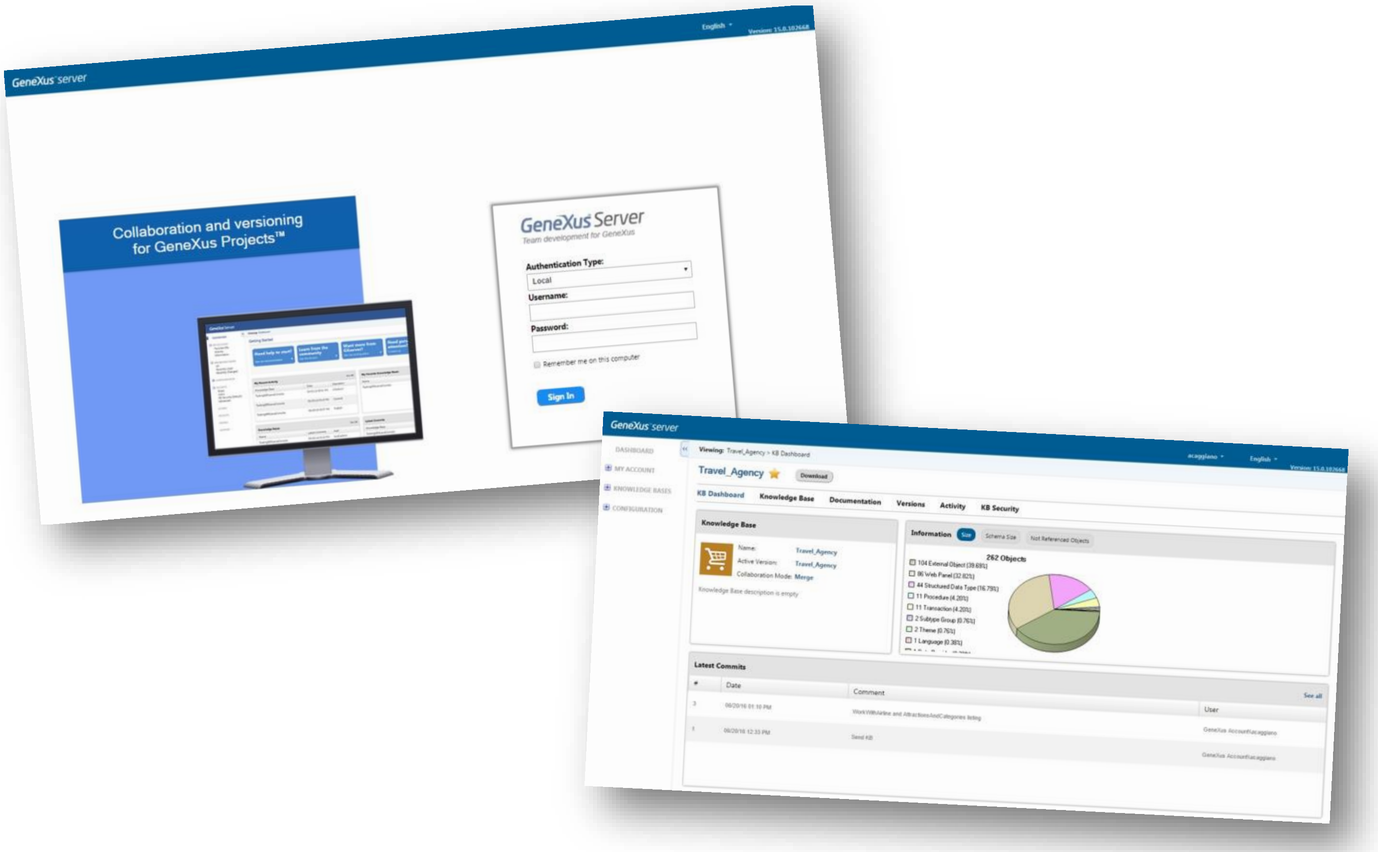
Commit



Employee ✕

Name	Type
Employee	Employee
EmployeeId	Id
EmployeeName	Name
EmployeeAddress	Address, GeneXus
EmployeePhone	Phone, GeneXus
EmployeeSalary	Numeric(4.0)
EmployeePhoto	Image

Web console



GeneXus server

English ▾ Version: 15.0.102668

Collaboration and versioning for GeneXus Projects™

GeneXus Server

Team development for GeneXus

Authentication Type: Local

Username:

Password:

□ Remember me on this computer

Sign In

GeneXus server

DASHBOARD

⊞ MY ACCOUNT

⊞ KNOWLEDGE BASES

⊞ CONFIGURATION

Viewing: Travel_Agency > KB Dashboard

acaggiano ▾ English ▾ Version: 15.0.102668

Travel_Agency ★ Download

KB Dashboard | Knowledge Base | Documentation | Versions | Activity | KB Security

Knowledge Base

Name: Travel_Agency

Active Version: Travel_Agency

Collaboration Mode: Merge

Knowledge Base description is empty

Information Size | Schema Size | Not Referenced Objects

262 Objects

□ 104 External Object (39.69%)
□ 86 Web Panel (32.82%)
□ 44 Structured Data Type (16.79%)
□ 11 Procedure (4.20%)
□ 11 Transaction (4.20%)
□ 2 Subtype Group (0.76%)
□ 2 Theme (0.76%)
□ 1 Language (0.38%)

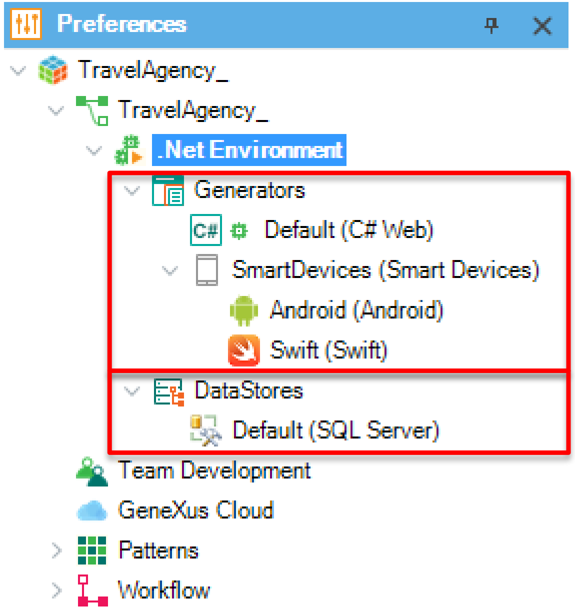
Latest Commits

#	Date	Comment	User
3	06/20/16 01:10 PM	WorkWithAirline and AttractionsAndCategories listing	GeneXus Account\acaggiano
1	06/20/16 12:33 PM	Send KB	GeneXus Account\acaggiano

See all

Ambientes de execução

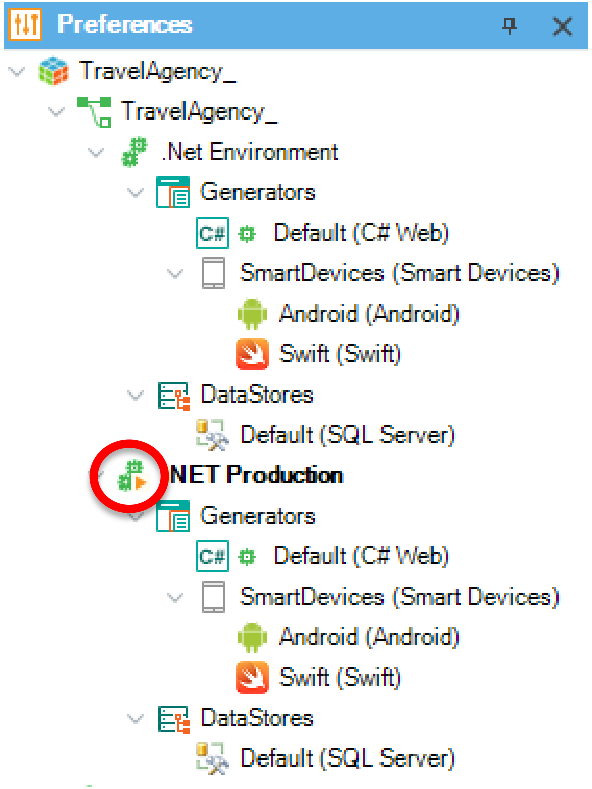
Geradores e Bancos de dados



Preferences

- TravelAgency_
  - TravelAgency_
    - .Net Environment
      - Generators
        - Default (C# Web)
        - SmartDevices (Smart Devices)
          - Android (Android)
          - Swift (Swift)
      - DataStores
        - Default (SQL Server)
  - Team Development
  - GeneXus Cloud
  - Patterns
  - Workflow

Trabalhando com mais de um ambiente

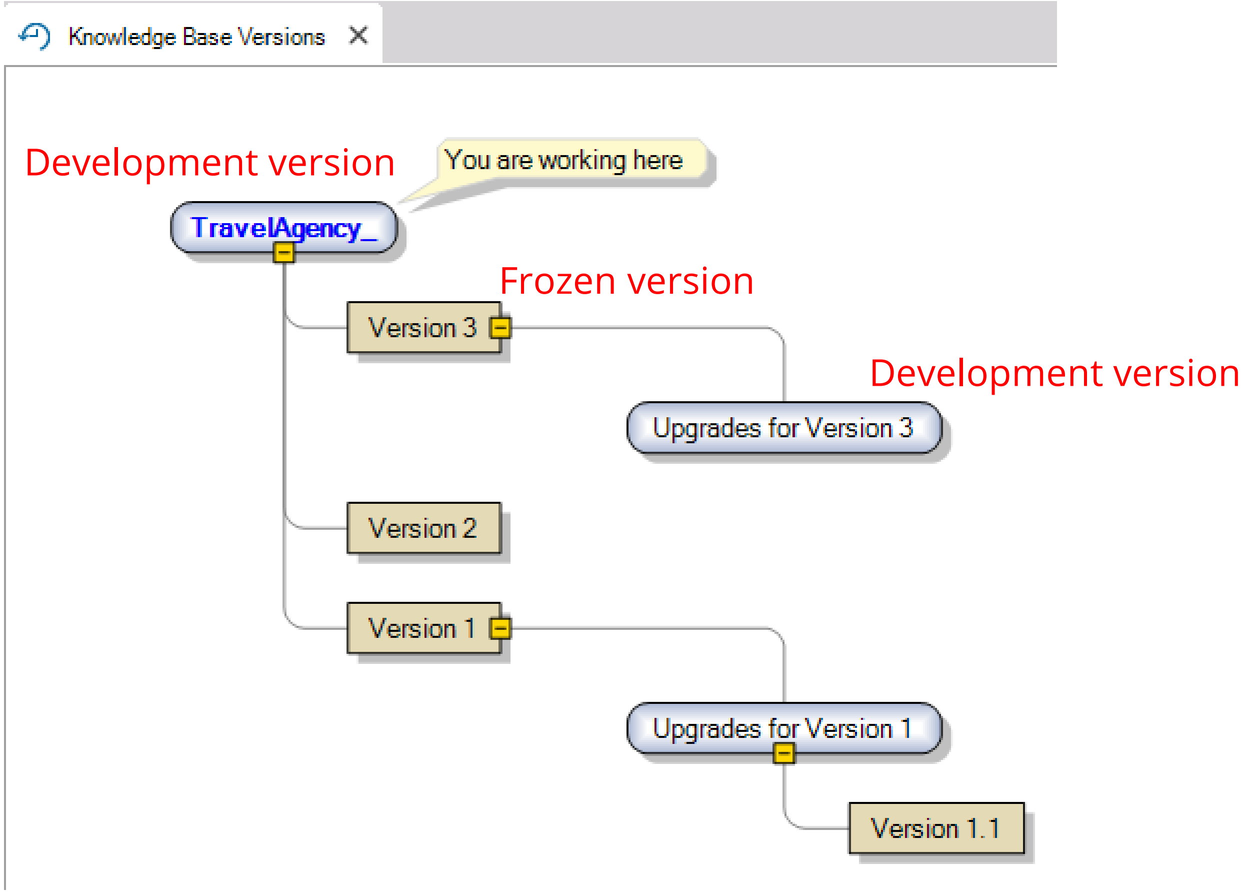


Preferences

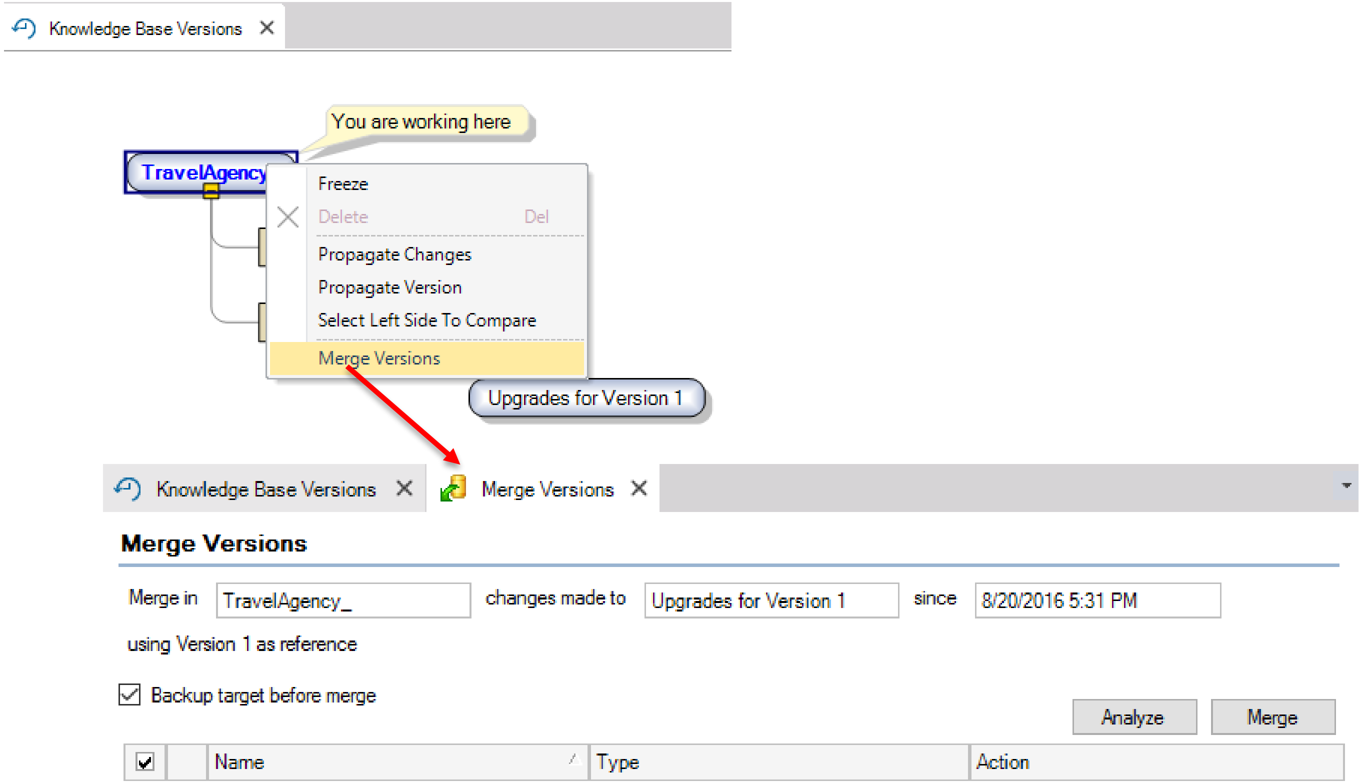
- TravelAgency_
  - TravelAgency_
    - .Net Environment
      - Generators
        - Default (C# Web)
        - SmartDevices (Smart Devices)
          - Android (Android)
          - Swift (Swift)
      - DataStores
        - Default (SQL Server)
    - **.NET Production**
      - Generators
        - Default (C# Web)
        - SmartDevices (Smart Devices)
          - Android (Android)
          - Swift (Swift)
      - DataStores
        - Default (SQL Server)

Versionamento de KBs

Árvore de versões



Para sincronizar (merge) das versões de desenvolvimento:



Merge Versions

Merge in TravelAgency_ changes made to Upgrades for Version 1 since 8/20/2016 5:31 PM

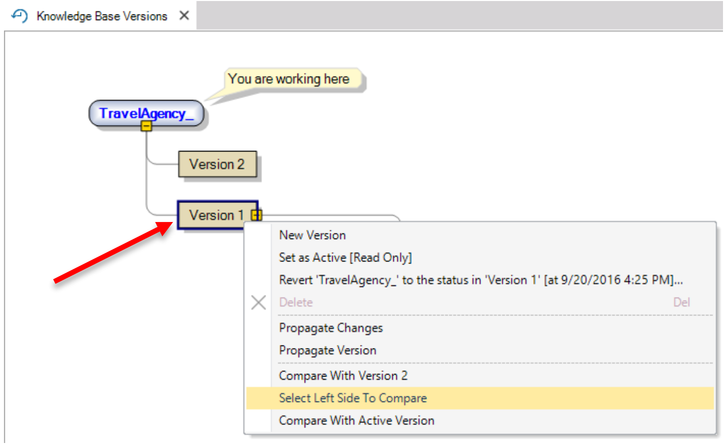
using Version 1 as reference

☑ Backup target before merge

☑		Name	Type	Action

Para comparar versões:

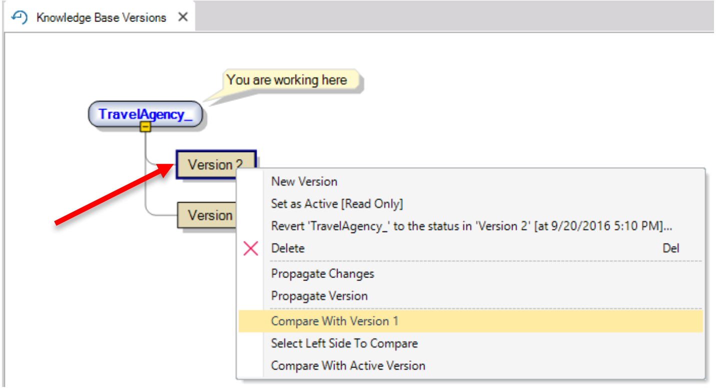
1)



Knowledge Base Versions

New Version
Set as Active [Read Only]
Revert 'TravelAgency_' to the status in 'Version 1' [at 9/20/2016 4:25 PM]...
Delete Del
Propagate Changes
Propagate Version
Compare With Version 2
Select Left Side To Compare
Compare With Active Version

2)



Knowledge Base Versions

New Version
Set as Active [Read Only]
Revert 'TravelAgency_' to the status in 'Version 2' [at 9/20/2016 5:10 PM]...
Delete Del
Propagate Changes
Propagate Version
Compare With Version 1
Select Left Side To Compare
Compare With Active Version

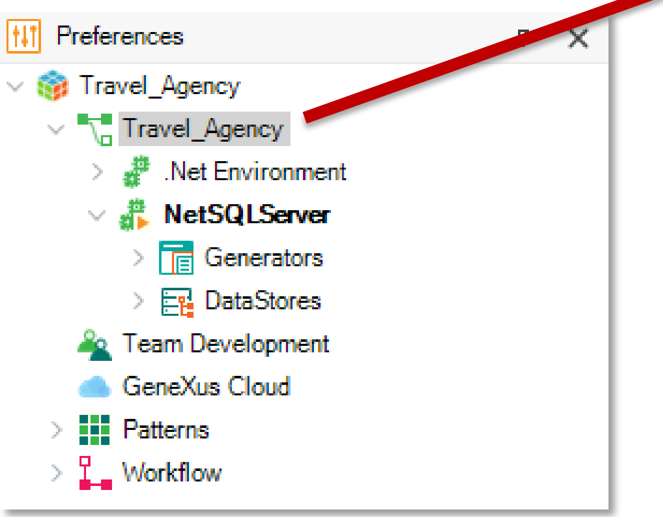
Segurança com GAM

AUTHENTICATION

AUTHORIZATION

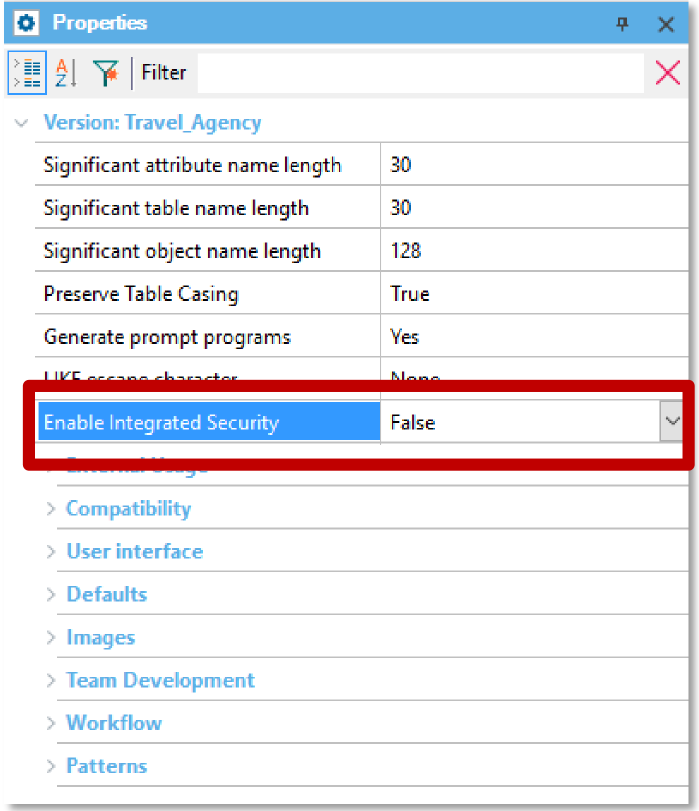
GeneXus™
ACCESS MANAGER

Habilitando o GeneXus Access Manager



Preferences

- Travel_Agency
  - Travel_Agency
    - .Net Environment
    - NetSQLServer
      - Generators
      - DataStores
    - Team Development
    - GeneXus Cloud
    - Patterns
    - Workflow



Properties

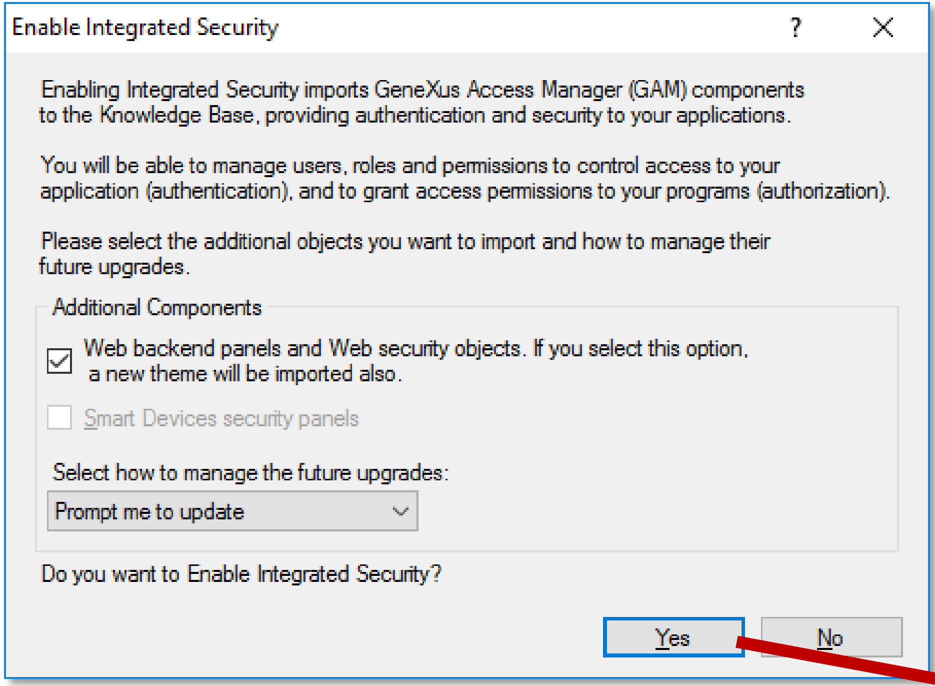
Filter

Version: Travel_Agency

Significant attribute name length	30
Significant table name length	30
Significant object name length	128
Preserve Table Casing	True
Generate prompt programs	Yes
LIKE escape character	None
Enable Integrated Security	False

- External Usage
- Compatibility
- User interface
- Defaults
- Images
- Team Development
- Workflow
- Patterns

Importando os objetos do GAM



Enable Integrated Security

Enabling Integrated Security imports GeneXus Access Manager (GAM) components to the Knowledge Base, providing authentication and security to your applications.

You will be able to manage users, roles and permissions to control access to your application (authentication), and to grant access permissions to your programs (authorization).

Please select the additional objects you want to import and how to manage their future upgrades.

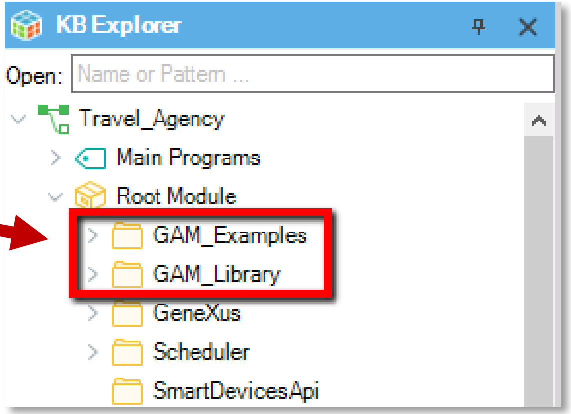
Additional Components

☑ Web backend panels and Web security objects. If you select this option, a new theme will be imported also.

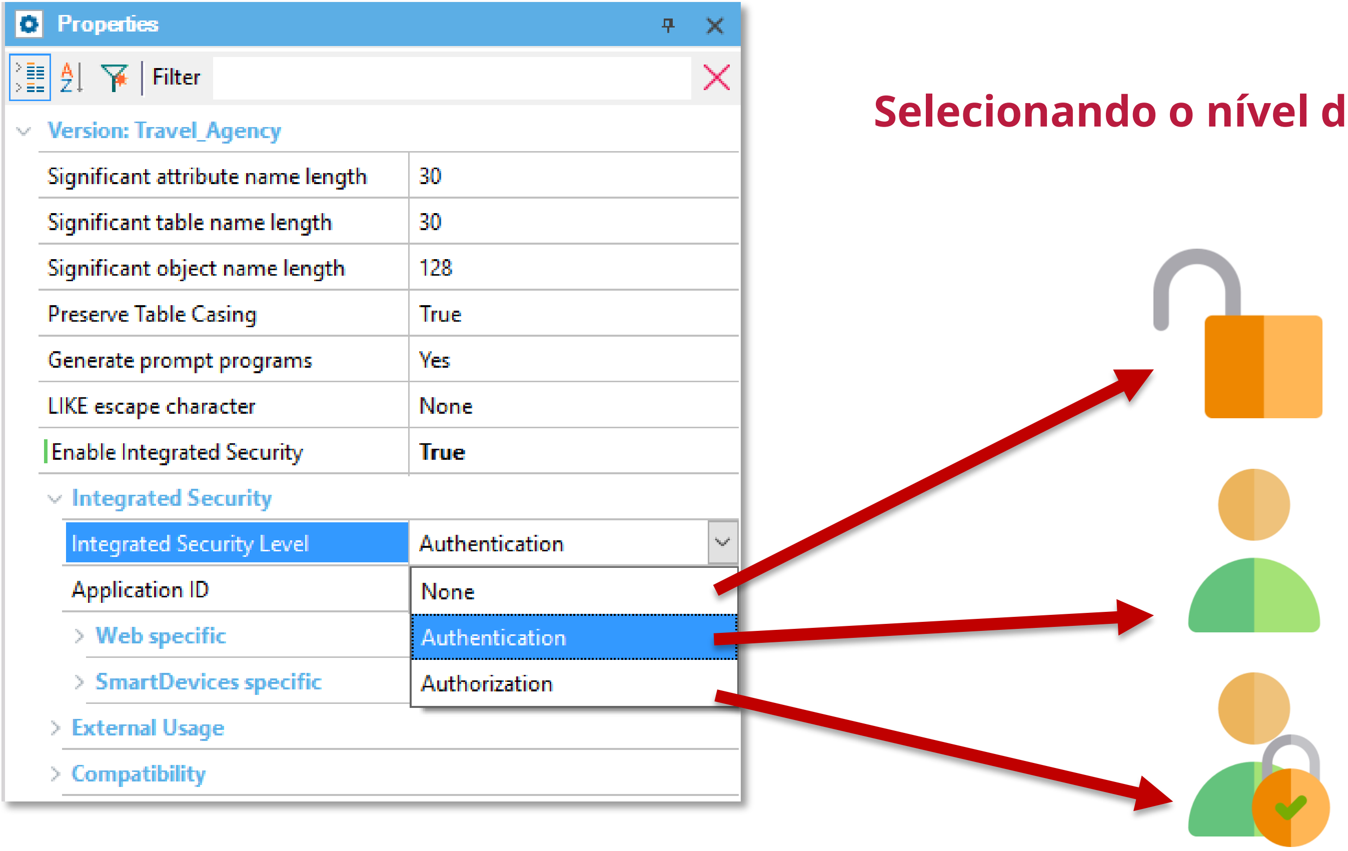
□ Smart Devices security panels

Select how to manage the future upgrades: Prompt me to update

Do you want to Enable Integrated Security?



Selecionando o nível de segurança



Properties

Filter

Version: Travel_Agency

Significant attribute name length	30
Significant table name length	30
Significant object name length	128
Preserve Table Casing	True
Generate prompt programs	Yes
LIKE escape character	None
Enable Integrated Security	**True**

Integrated Security

Integrated Security Level	Authentication
Application ID	

- None
- Authentication
- Authorization

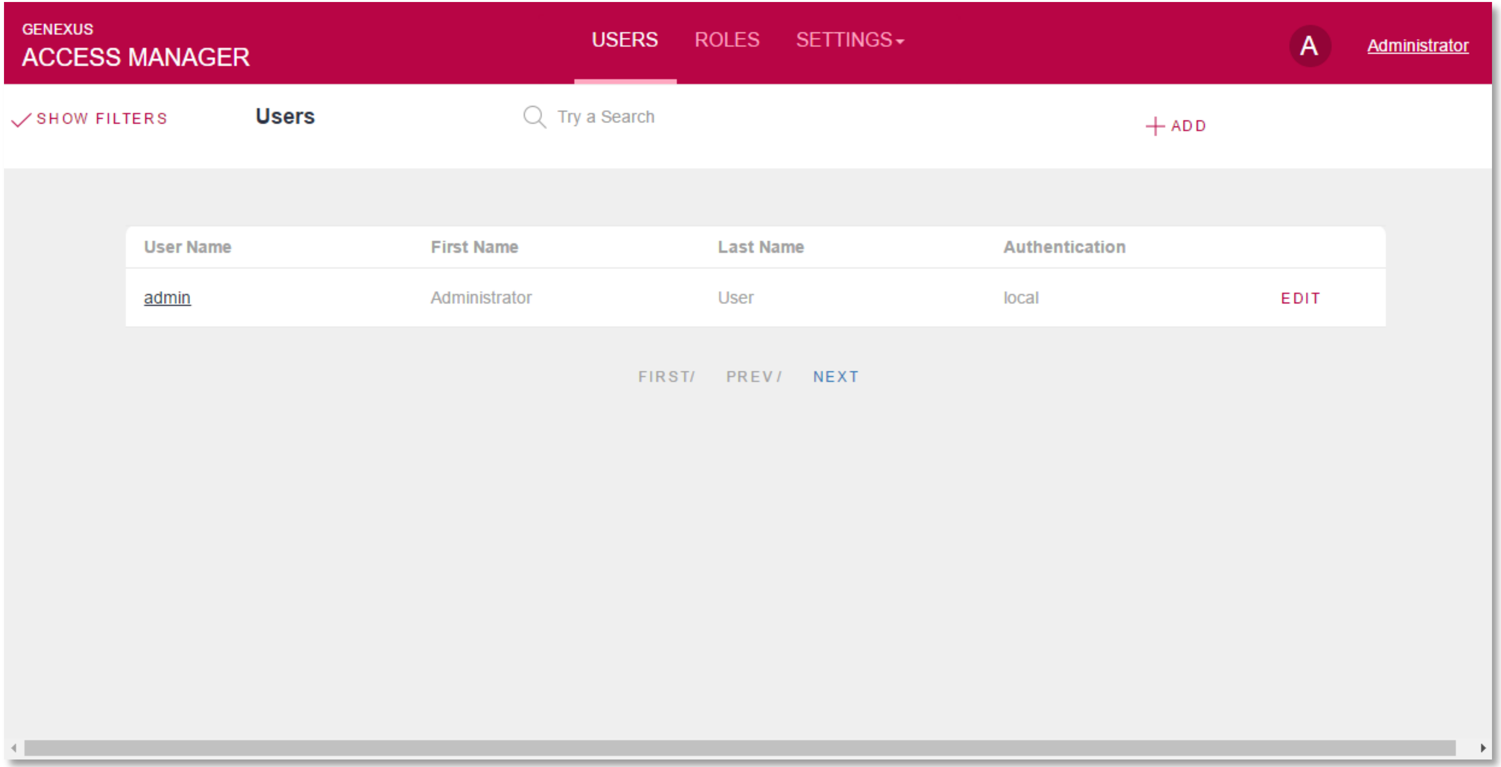
Web specific

SmartDevices specific

External Usage

Compatibility

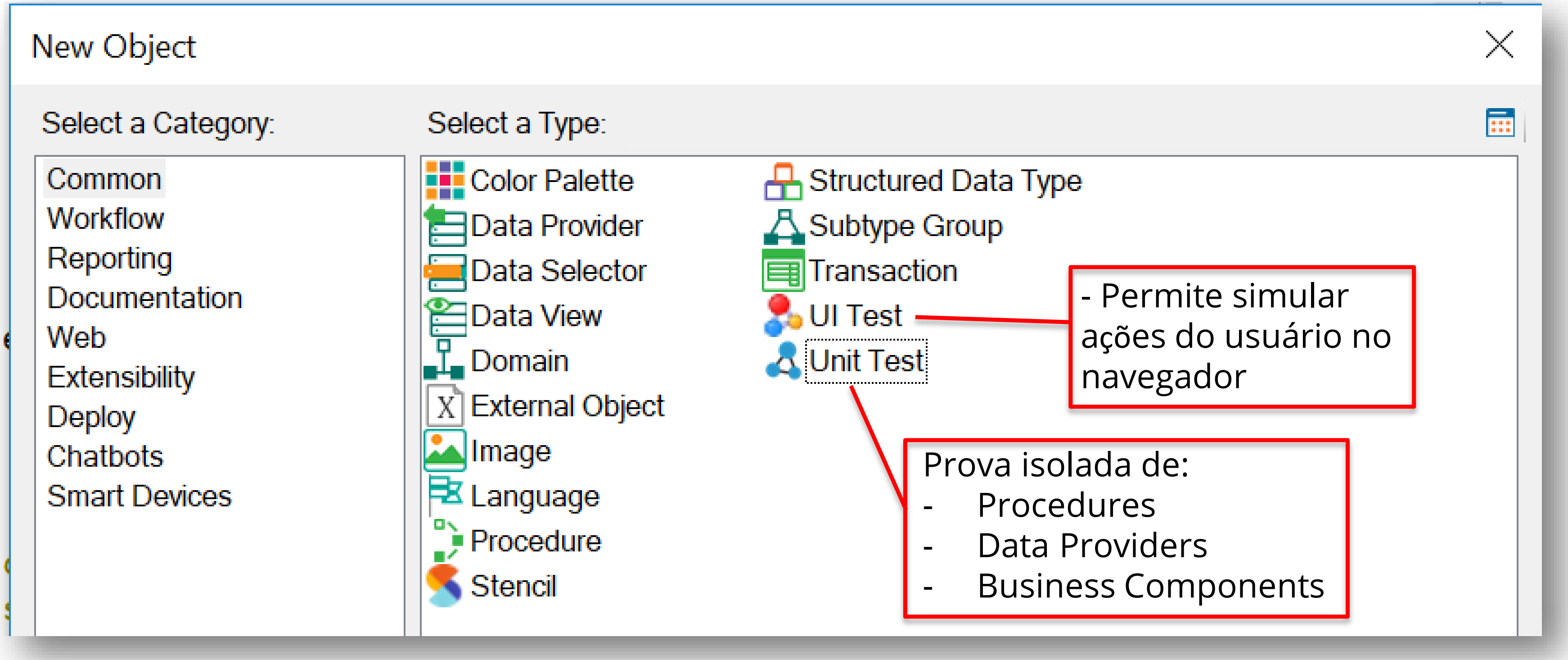
Acesso ao backend do GAM (objeto GAM Home)



User Name	First Name	Last Name	Authentication	
admin	Administrator	User	local	EDIT

Testing

Objetos para gerar testes unitários e testes de interface



New Object

Select a Category:

- Common
- Workflow
- Reporting
- Documentation
- Web
- Extensibility
- Deploy
- Chatbots
- Smart Devices

Select a Type:

- Color Palette
- Data Provider
- Data Selector
- Data View
- Domain
- External Object
- Image
- Language
- Procedure
- Stencil
- Structured Data Type
- Subtype Group
- Transaction
- UI Test
- Unit Test

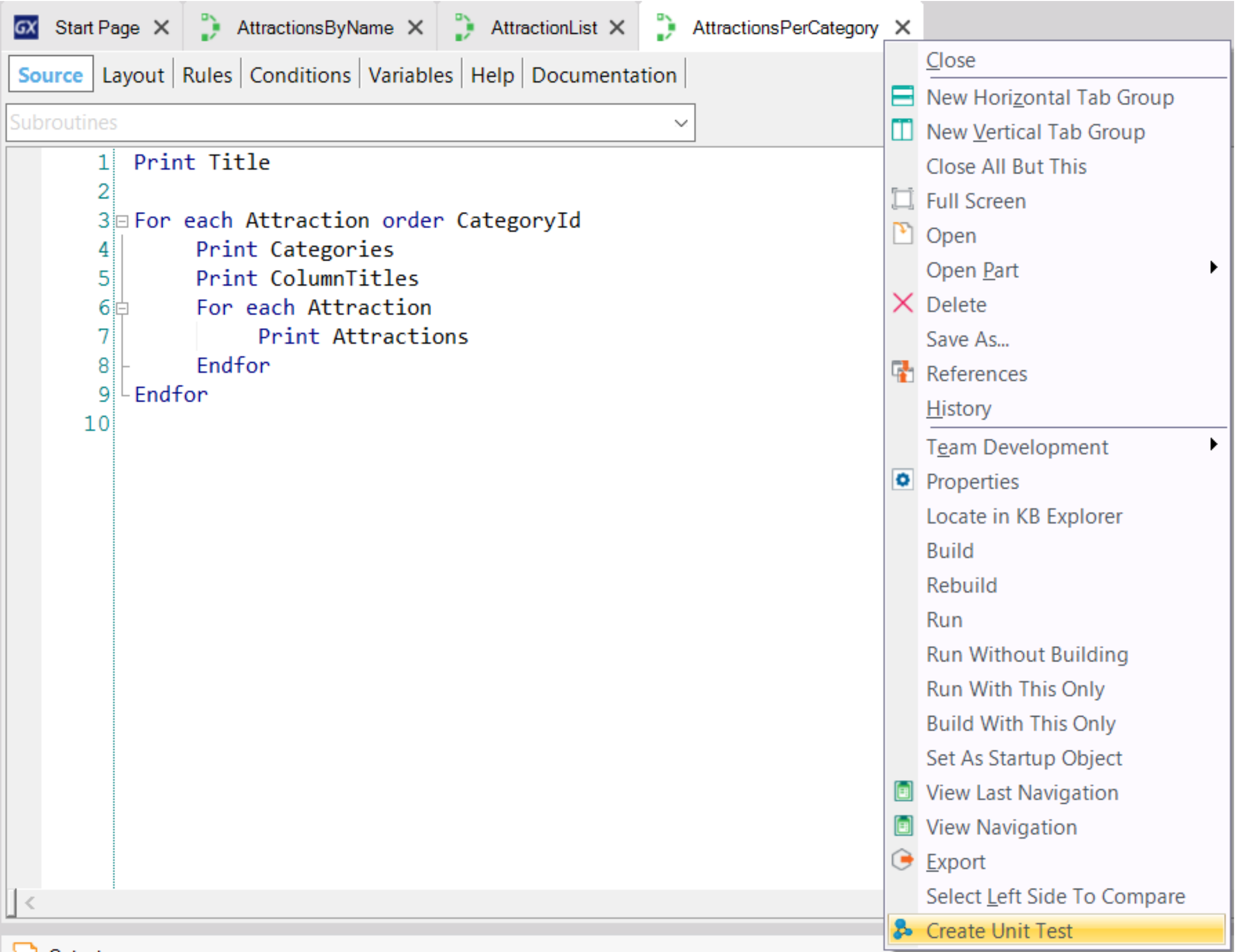
- Permite simular ações do usuário no navegador

Prova isolada de:
- Procedures
- Data Providers
- Business Components

Teste Unitário

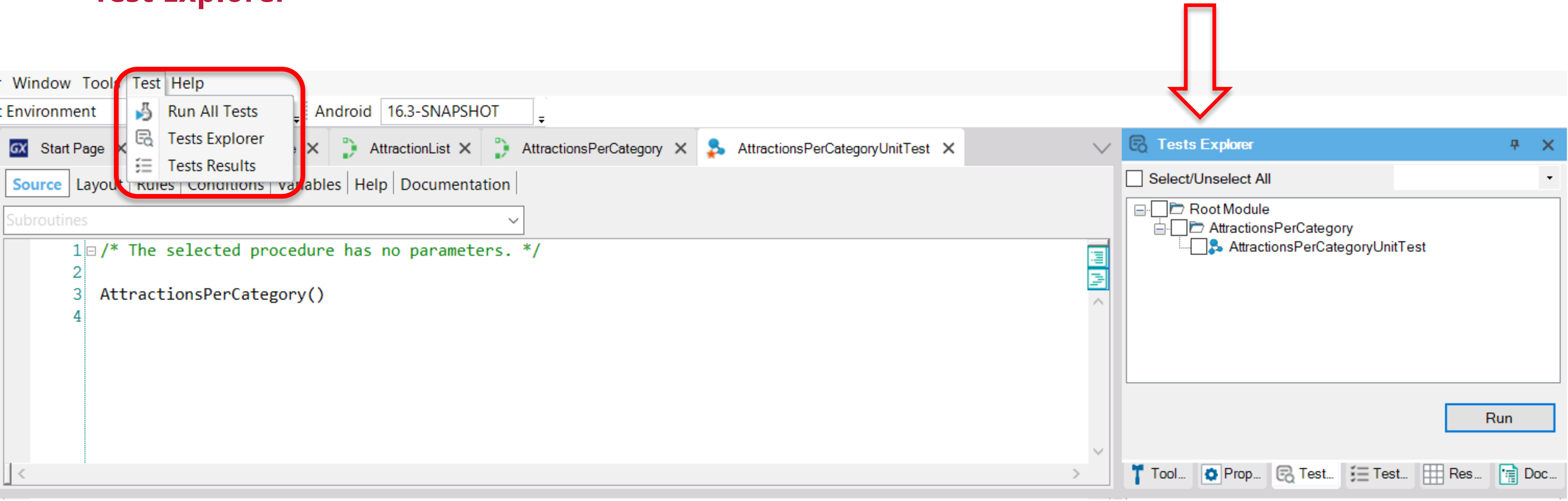
São criados os objetos:

- <ObjectName>UnitTest
- <ObjectName>UnitTestSDT
- <ObjectName>UnitTestData



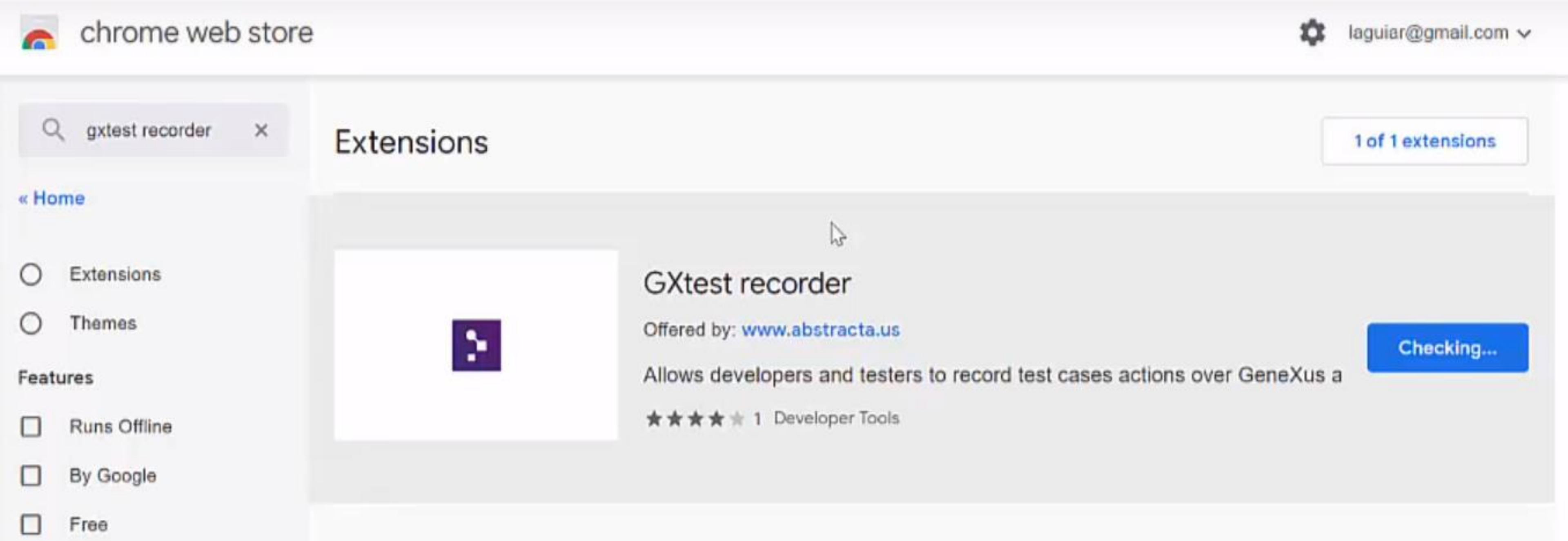
```
Print Title

For each Attraction order CategoryId
      Print Categories
      Print ColumnTitles
      For each Attraction
            Print Attractions
      Endfor
Endfor
```

Test Explorer



```
/* The selected procedure has no parameters. */

AttractionsPerCategory()

```

Teste de Interface: GXTest



chrome web store

laguiar@gmail.com

gxtest recorder ✕

« Home

○ Extensions

○ Themes

Features

□ Runs Offline

□ By Google

□ Free

Extensions

1 of 1 extensions

GXtest recorder

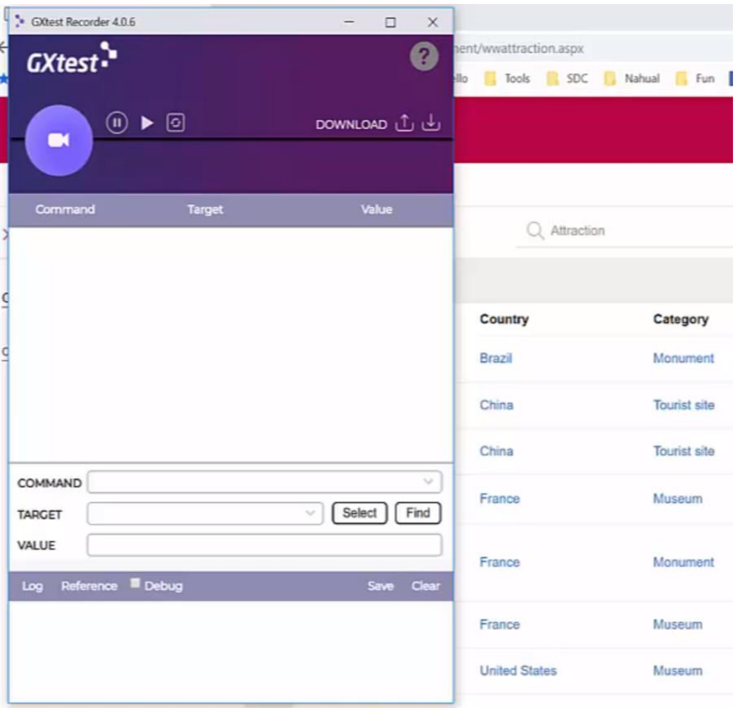
Offered by: www.abstracta.us

Allows developers and testers to record test cases actions over GeneXus a

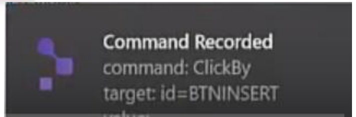
★★★★☆ 1 Developer Tools

Checking...

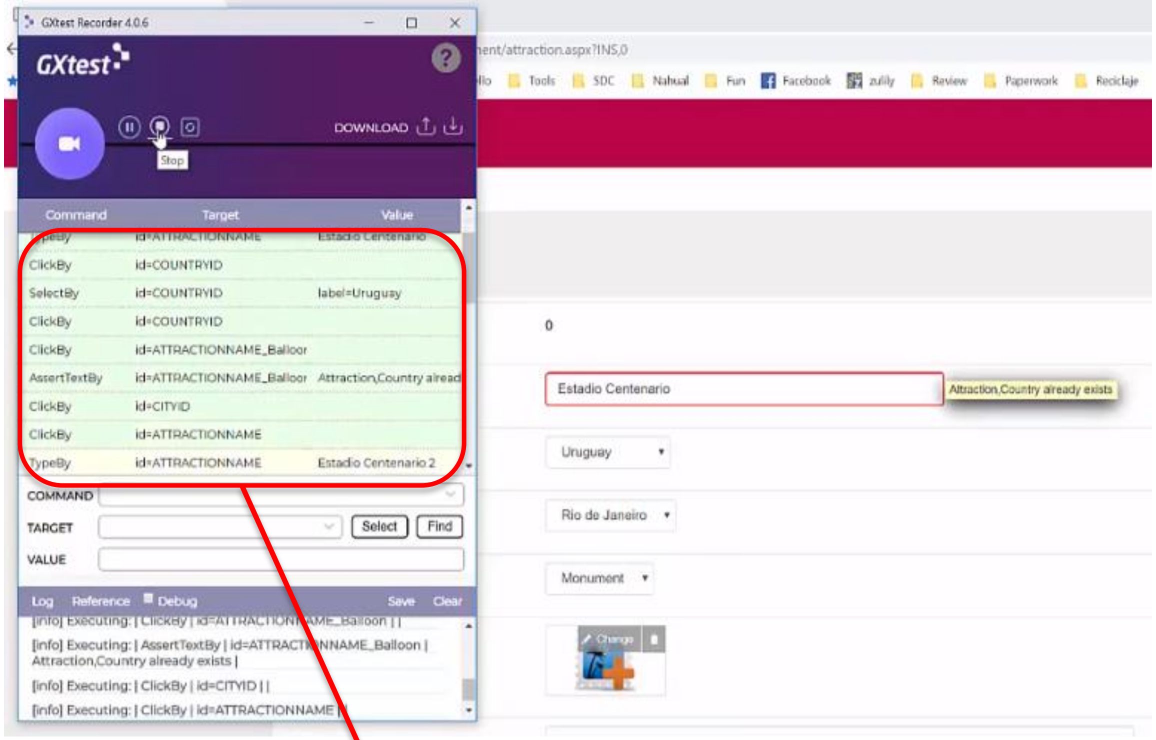
Teste de Interface



Country	Category
Brazil	Monument
China	Tourist site
China	Tourist site
France	Museum
France	Monument
France	Museum
United States	Museum



Command Recorded
command: ClickBy
target: id=BTNINSERT


GXtest Recorder 4.0.6
GXtest
Stop
DOWNLOAD
COMMAND
TARGET
VALUE
Select
Find
Log Reference Debug Save Clear
ent/attraction.aspx?INS,0
llo Tools SDC Nahual Fun Facebook zulily Review Paperwork Reciclaje
0
Estadio Centenario
Attraction,Country already exists
Uruguay
Rio de Janeiro
Monument
Change

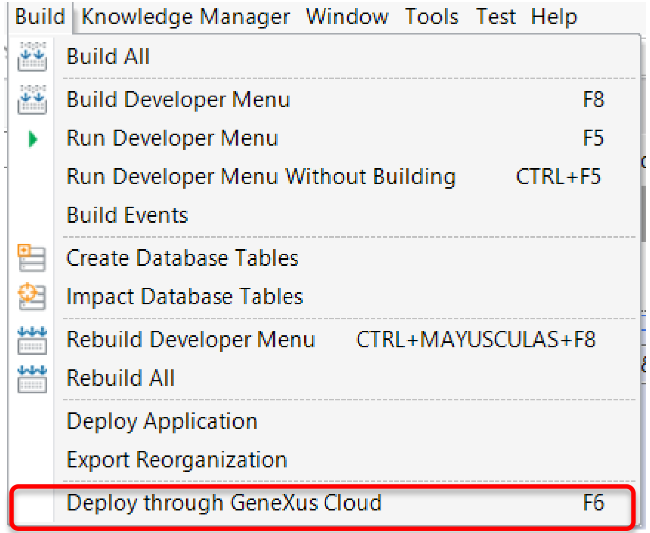
Command	Target	Value
TypeBy	id=ATTRACTIONNAME	Estadio Centenario
ClickBy	id=COUNTRYID	
SelectBy	id=COUNTRYID	label=Uruguay
ClickBy	id=COUNTRYID	
ClickBy	id=ATTRACTIONNAME_Balloor	
AssertTextBy	id=ATTRACTIONNAME_Balloor	Attraction,Country alread
ClickBy	id=CITYID	
ClickBy	id=ATTRACTIONNAME	
TypeBy	id=ATTRACTIONNAME	Estadio Centenario 2

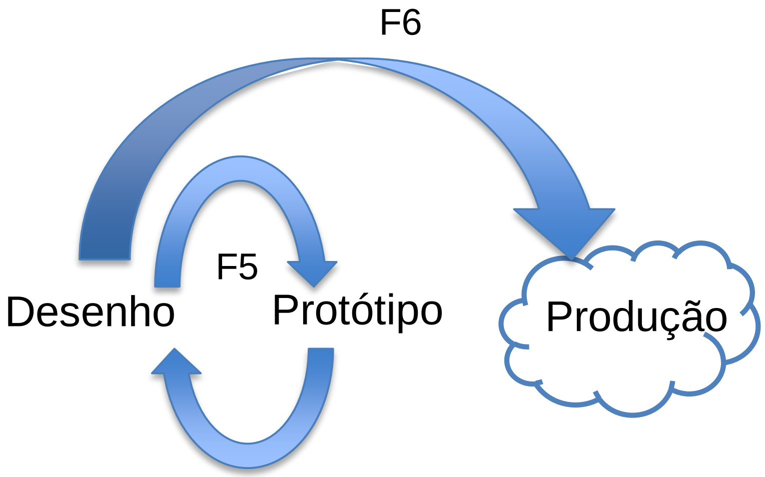
```
[info] Executing: | ClickBy | id=ATTRACTIONNAME_Balloon | |
[info] Executing: | AssertTextBy | id=ATTRACTIONNAME_Balloon |
Attraction,Country already exists |
[info] Executing: | ClickBy | id=CITYID | |
[info] Executing: | ClickBy | id=ATTRACTIONNAME | |
```

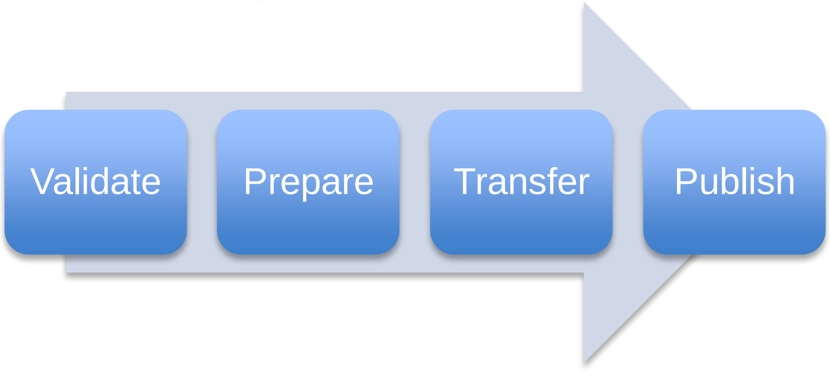
UI Test (Requer licença GXTest)

Deployment

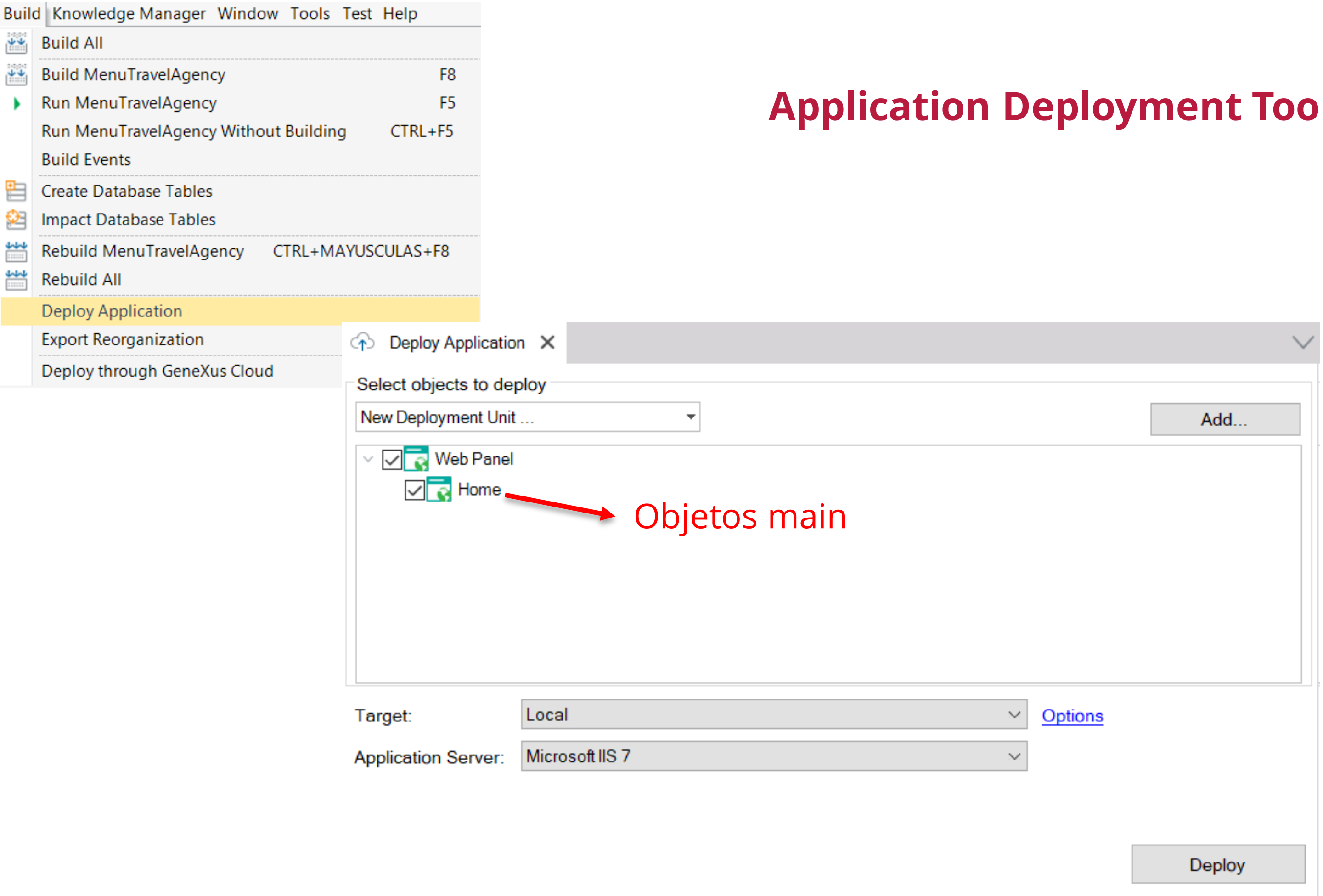
Deploy automático com F6


Build Knowledge Manager Window Tools Test Help
Build All
Build Developer Menu F8
Run Developer Menu F5
Run Developer Menu Without Building CTRL+F5
Build Events
Create Database Tables
Impact Database Tables
Rebuild Developer Menu CTRL+MAYUSCULAS+F8
Rebuild All
Deploy Application
Export Reorganization
Deploy through GeneXus Cloud F6





Application Deployment Tool



Select objects to deploy

New Deployment Unit ...

☑ Web Panel

☑ Home

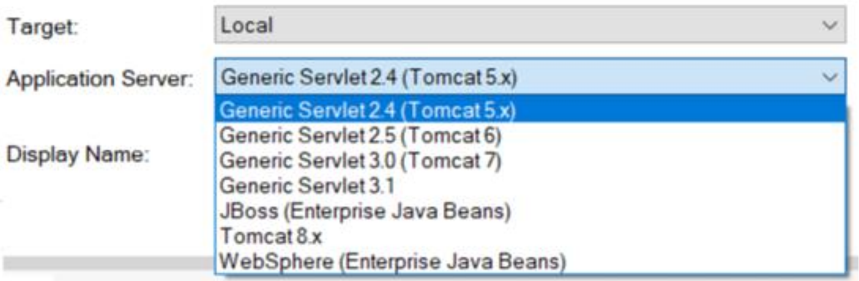
Objetos main

Target: Local

Application Server: Microsoft IIS 7

Application Deployment Tool

Deploy local (criação de pacotes):

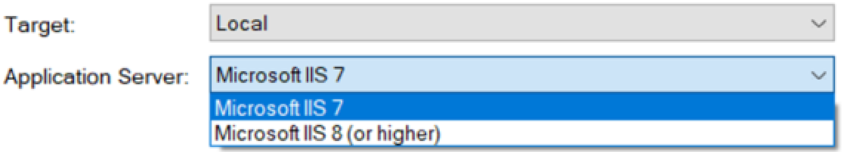


Target: Local

Application Server: Generic Servlet 2.4 (Tomcat 5.x)

Display Name:

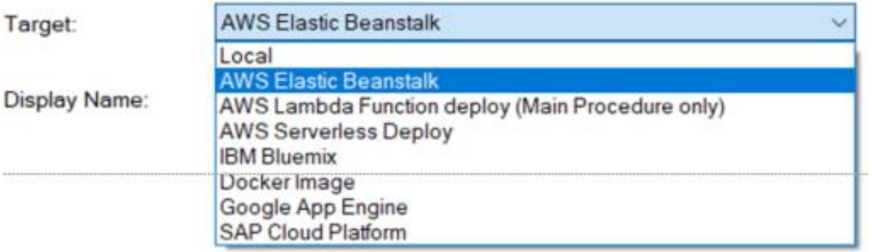
Generic Servlet 2.4 (Tomcat 5.x)
Generic Servlet 2.5 (Tomcat 6)
Generic Servlet 3.0 (Tomcat 7)
Generic Servlet 3.1
JBoss (Enterprise Java Beans)
Tomcat 8.x
WebSphere (Enterprise Java Beans)



Target: Local

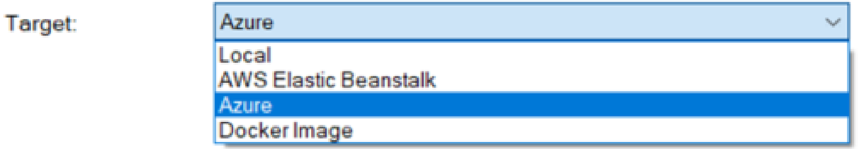
Application Server: Microsoft IIS 7

Deploy para servidores PAAS (Platform As A Service)



Target: AWS Elastic Beanstalk

Display Name:



Target: Azure

GeneXus™ Cloud

Deployment Services

Preparação do ambiente na Nuvem

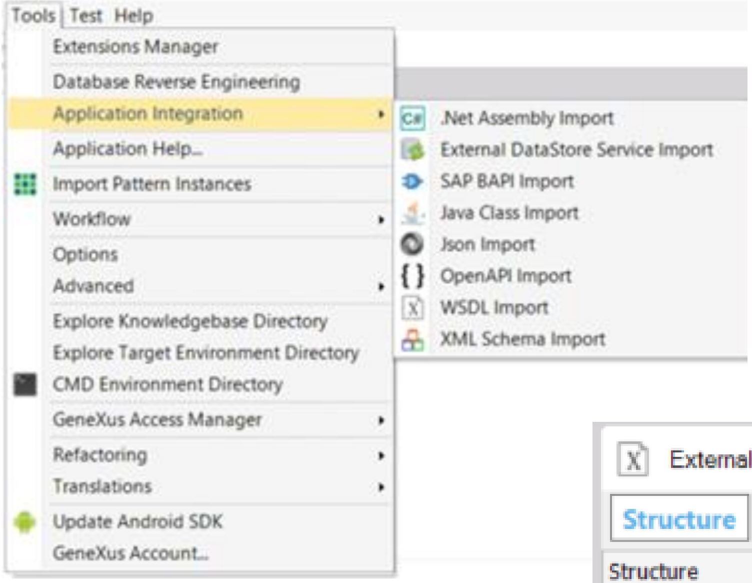
Deploy automático

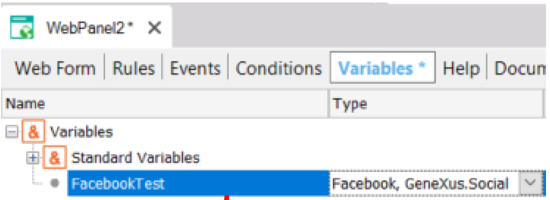
Administração dos deploys

Maximização do tempo para colocar a aplicação em funcionamento

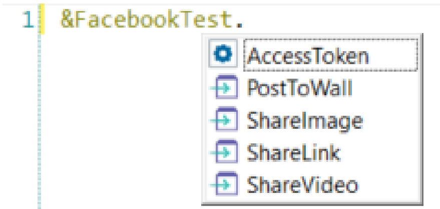
Integração

Objetos externos

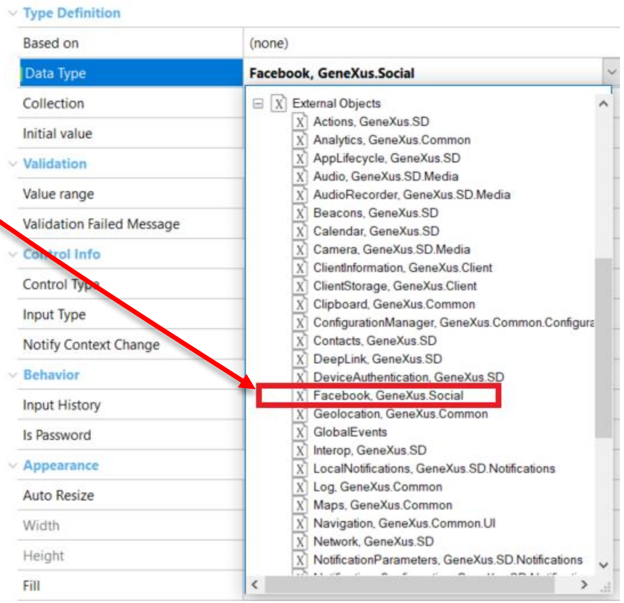



WebPanel2 * ✕
Web Form | Rules | Events | Conditions | Variables * | Help | Docun

Name	Type
⊟ & Variables	
⊞ & Standard Variables	
● FacebookTest	Facebook, GeneXus.Social

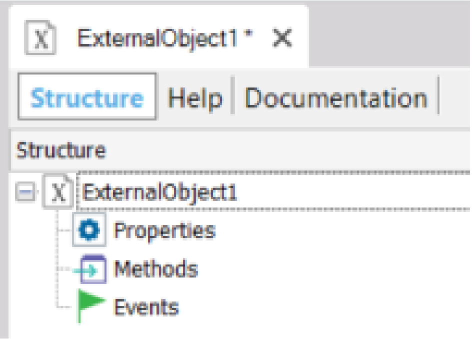


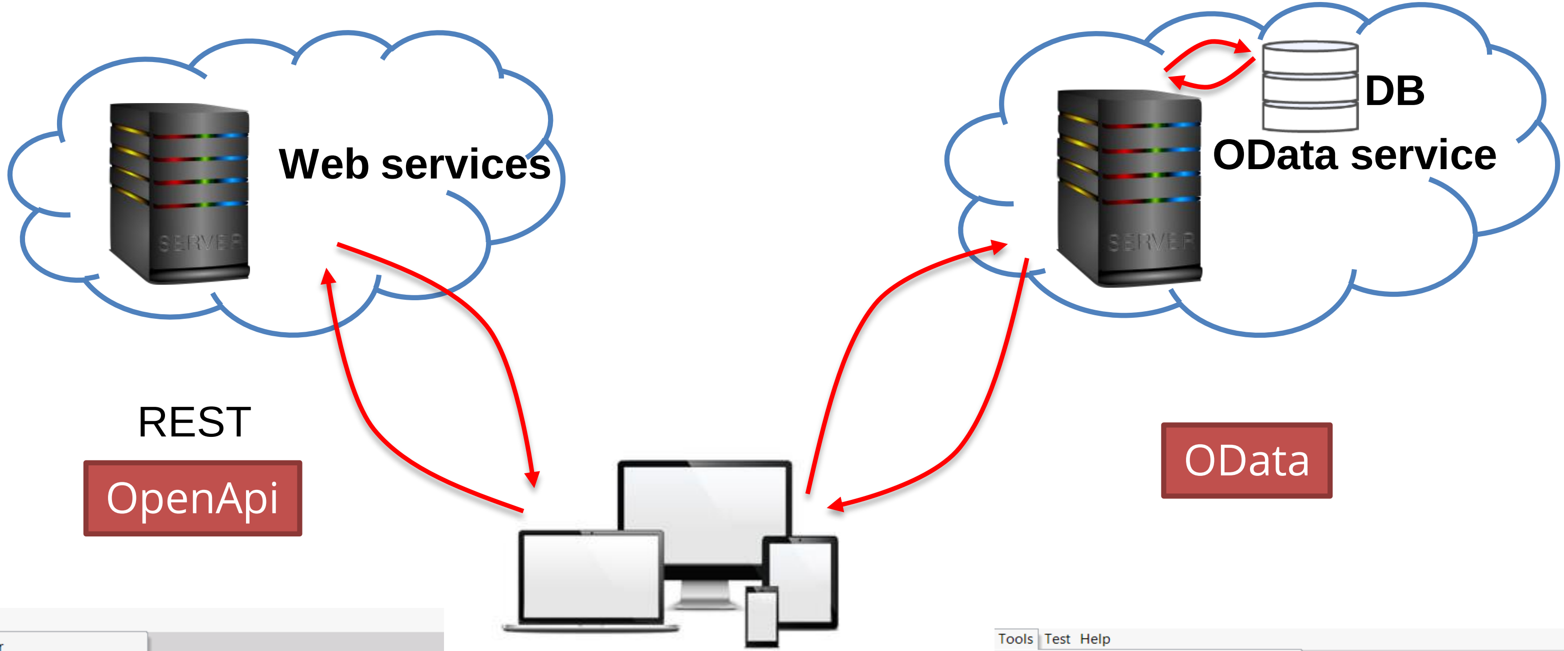
```
1 &FacebookTest.
```



Type Definition	
Based on	(none)
Data Type	Facebook, GeneXus.Social
Collection	
Initial value	
Validation	
Value range	
Validation Failed Message	
Control Info	
Control Type	
Input Type	
Notify Context Change	
Behavior	
Input History	
Is Password	
Appearance	
Auto Resize	
Width	
Height	
Fill	

External Objects
Actions, GeneXus.SD
Analytics, GeneXus.Common
AppLifecycle, GeneXus.SD
Audio, GeneXus.SD.Media
AudioRecorder, GeneXus.SD.Media
Beacons, GeneXus.SD
Calendar, GeneXus.SD
Camera, GeneXus.SD.Media
ClientInformation, GeneXus.Client
ClientStorage, GeneXus.Client
Clipboard, GeneXus.Common
ConfigurationManager, GeneXus.Common.Configura
Contacts, GeneXus.SD
DeepLink, GeneXus.SD
DeviceAuthentication, GeneXus.SD
Facebook, GeneXus.Social
Geolocation, GeneXus.Common
GlobalEvents
Interop, GeneXus.SD
LocalNotifications, GeneXus.SD.Notifications
Log, GeneXus.Common
Maps, GeneXus.Common
Navigation, GeneXus.Common.UI
Network, GeneXus.SD
NotificationParameters, GeneXus.SD.Notifications





Web services

DB

OData service

Aplicações cliente

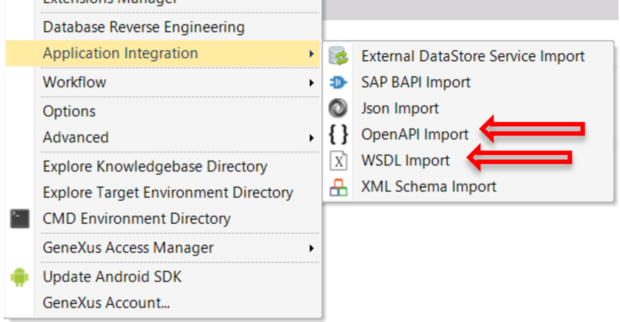
SOAP

WSDL

REST

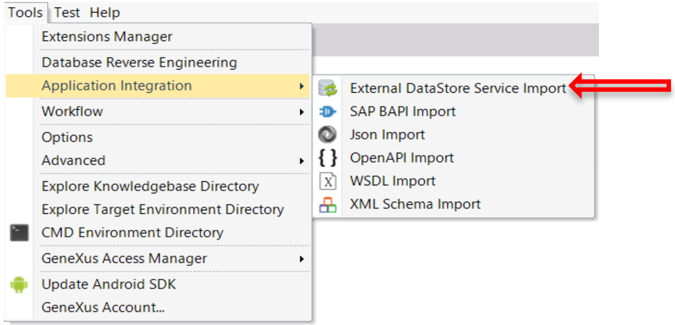
OpenApi

OData



Tools Test Help

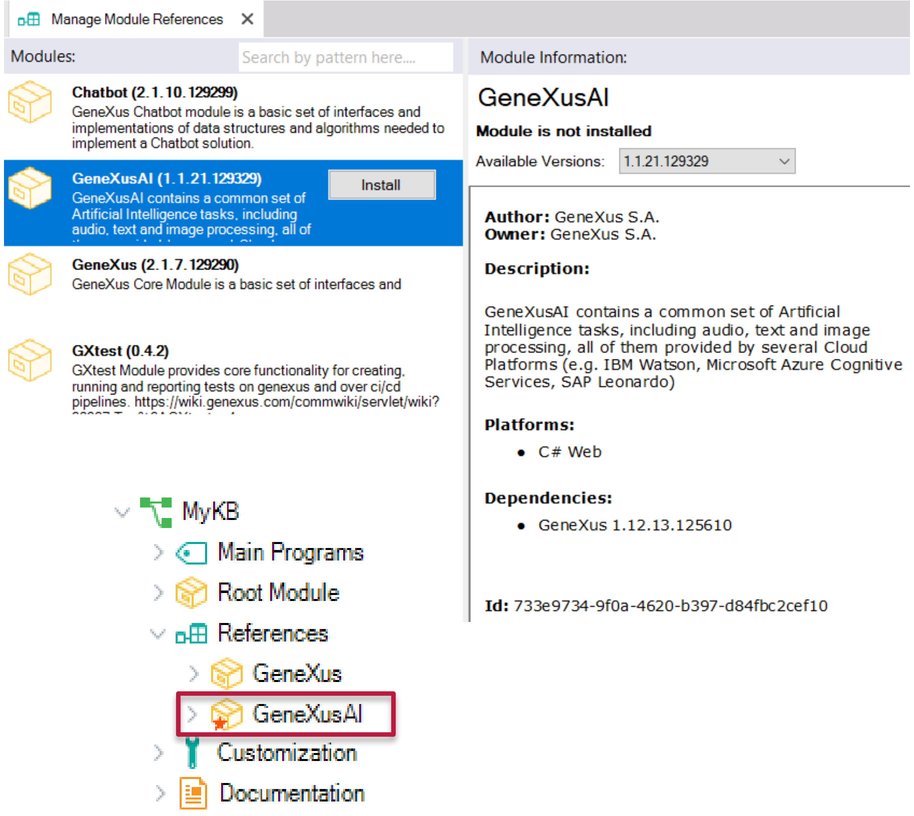
- Extensions Manager
- Database Reverse Engineering
- Application Integration ▸
  - External DataStore Service Import
  - SAP BAPI Import
  - Json Import
  - OpenAPI Import
  - WSDL Import
  - XML Schema Import
- Workflow ▸
- Options
- Advanced ▸
- Explore Knowledgebase Directory
- Explore Target Environment Directory
- CMD Environment Directory
- GeneXus Access Manager ▸
- Update Android SDK
- GeneXus Account...



Tools Test Help

- Extensions Manager
- Database Reverse Engineering
- Application Integration ▸
  - External DataStore Service Import
  - SAP BAPI Import
  - Json Import
  - OpenAPI Import
  - WSDL Import
  - XML Schema Import
- Workflow ▸
- Options
- Advanced ▸
- Explore Knowledgebase Directory
- Explore Target Environment Directory
- CMD Environment Directory
- GeneXus Access Manager ▸
- Update Android SDK
- GeneXus Account...

Módulo de Inteligência Artificial



Manage Module References

Modules: Search by pattern here....

Chatbot (2.1.10.129299)
GeneXus Chatbot module is a basic set of interfaces and implementations of data structures and algorithms needed to implement a Chatbot solution.

GeneXusAI (1.1.21.129329) Install
GeneXusAI contains a common set of Artificial Intelligence tasks, including audio, text and image processing, all of

GeneXus (2.1.7.129290)
GeneXus Core Module is a basic set of interfaces and

GXtest (0.4.2)
GXtest Module provides core functionality for creating, running and reporting tests on genexus and over ci/cd pipelines. https://wiki.genexus.com/commwiki/servlet/wiki?

Module Information:

GeneXusAI

Module is not installed

Available Versions: 1.1.21.129329

Author: GeneXus S.A.
Owner: GeneXus S.A.

Description:

GeneXusAI contains a common set of Artificial Intelligence tasks, including audio, text and image processing, all of them provided by several Cloud Platforms (e.g. IBM Watson, Microsoft Azure Cognitive Services, SAP Leonardo)

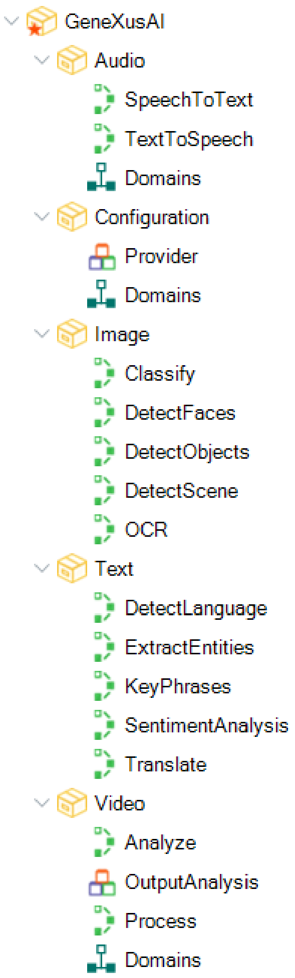
Platforms:

- C# Web

Dependencies:

- GeneXus 1.12.13.125610

Id: 733e9734-9f0a-4620-b397-d84fbc2cef10



IBM Watson

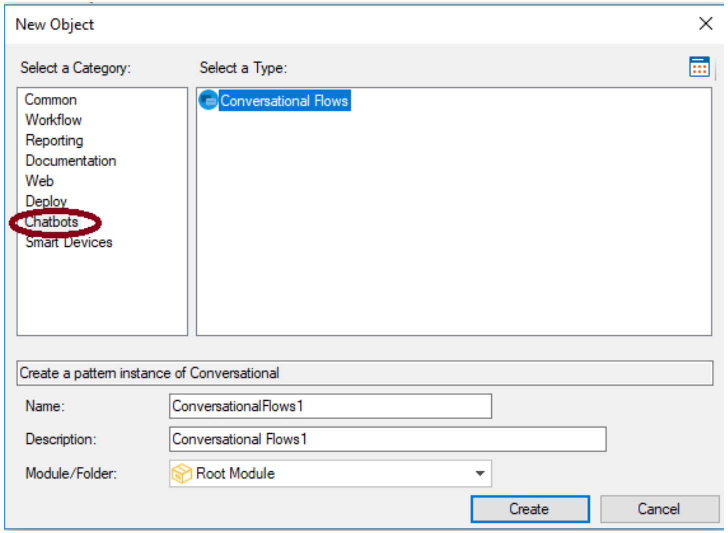
Microsoft Cognitive Services

SAP Leonardo

Google Cloud Services

Amazon Web Services

Chatbots: objeto Conversational Flow



New Object

Select a Category:
- Common
- Workflow
- Reporting
- Documentation
- Web
- Deploy
- Chatbots
- Smart Devices

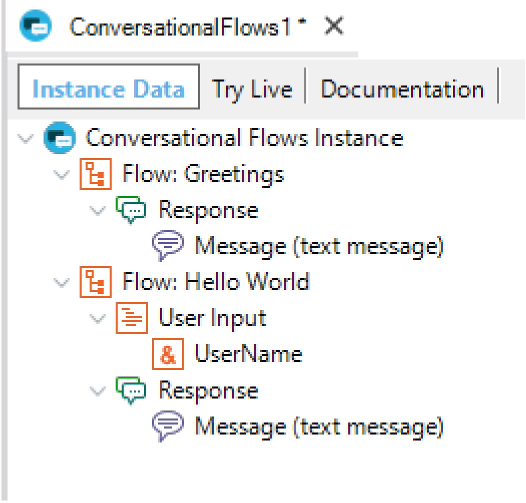
Select a Type:
- Conversational Flows

Create a pattern instance of Conversational

Name: ConversationalFlows1

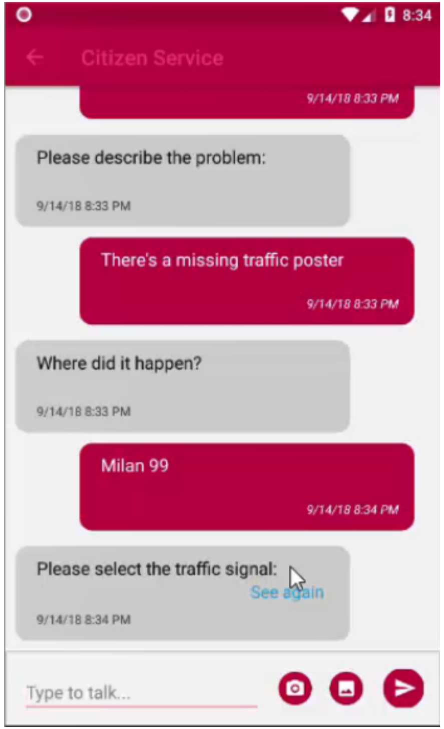
Description: Conversational Flows1

Module/Folder: Root Module



ConversationalFlows1 *

- Conversational Flows Instance
  - Flow: Greetings
    - Response
      - Message (text message)
  - Flow: Hello World
    - User Input
      - UserName
    - Response
      - Message (text message)



9/14/18 8:33 PM

Please describe the problem:
9/14/18 8:33 PM

There's a missing traffic poster
9/14/18 8:33 PM

Where did it happen?
9/14/18 8:33 PM

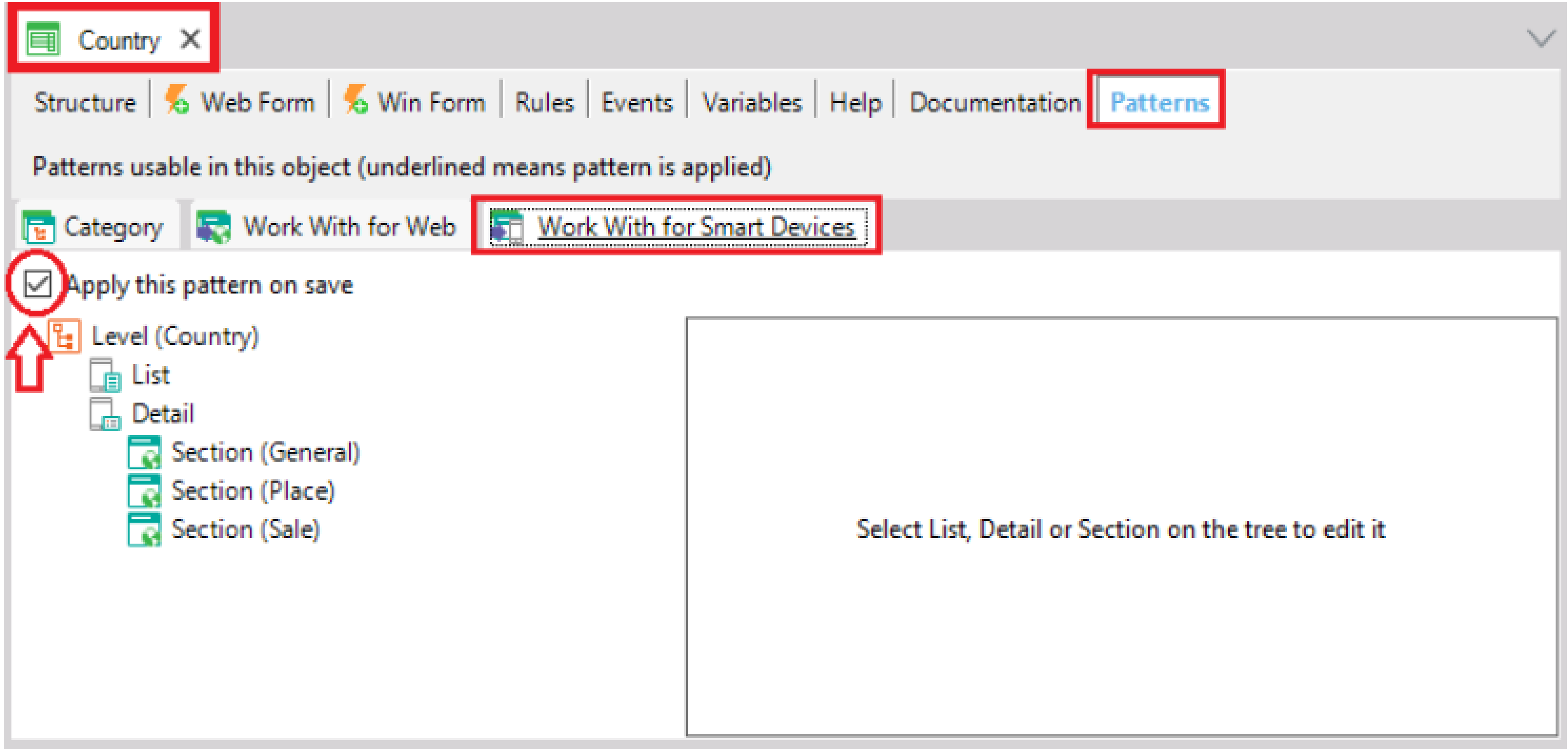
Milan 99
9/14/18 8:34 PM

Please select the traffic signal:
See again
9/14/18 8:34 PM

Type to talk...

Smart Devices

Aplicar Pattern em Transação



Country ✕
Structure | Web Form | Win Form | Rules | Events | Variables | Help | Documentation | Patterns

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

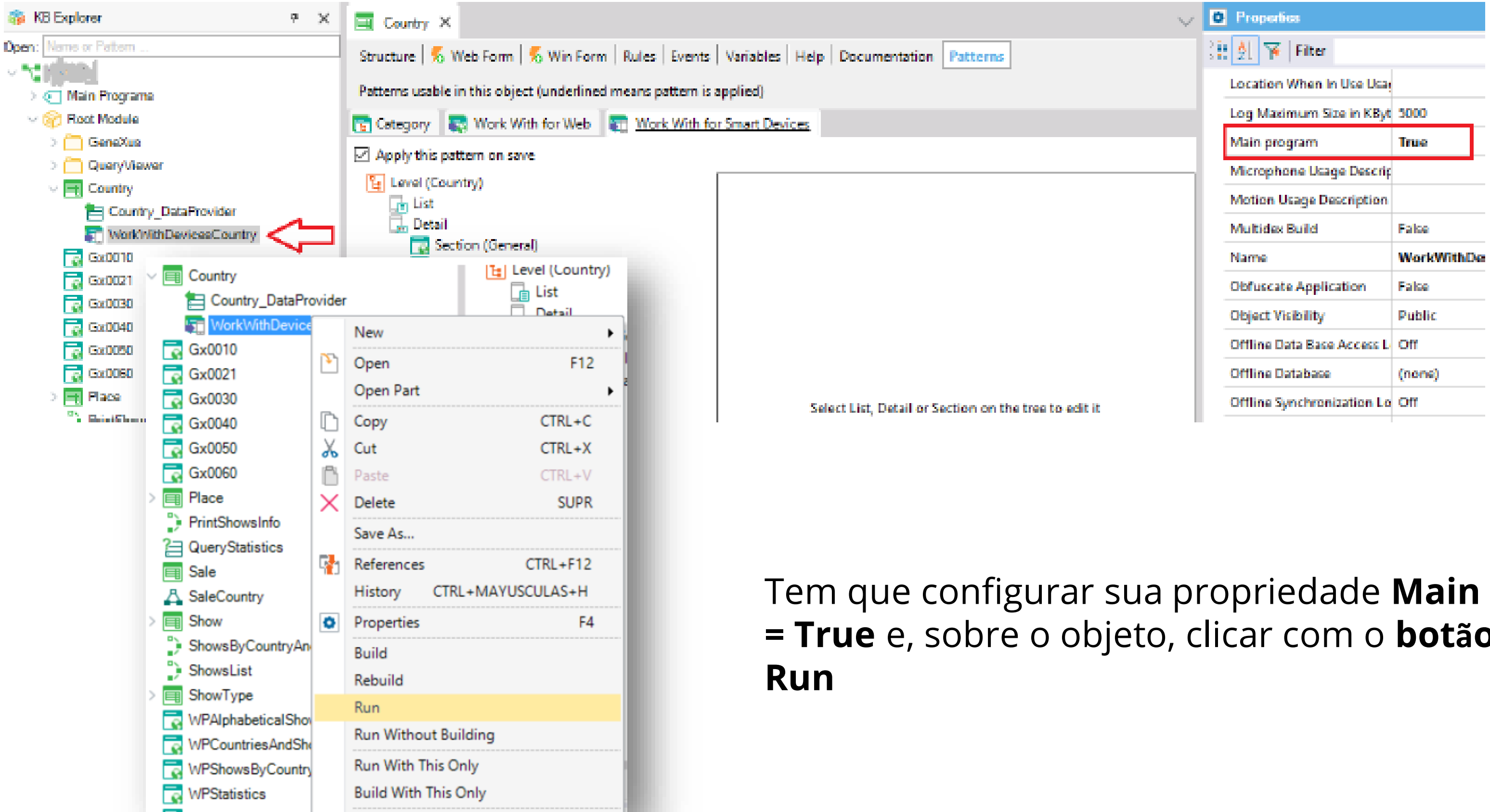
Category | Work With for Web | Work With for Smart Devices

☑ Apply this pattern on save

Level (Country)
List
Detail
Section (General)
Section (Place)
Section (Sale)

Select List, Detail or Section on the tree to edit it

Aplicar Pattern em Transação



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

☑ Apply this pattern on save

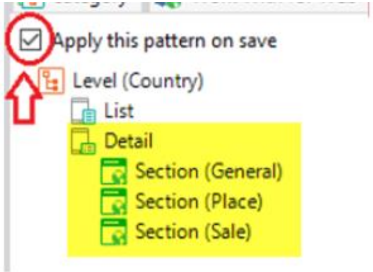
Select List, Detail or Section on the tree to edit it

Location When In Use Usa	
Log Maximum Size in KByt	5000
Main program	True
Microphone Usage Descrip	
Motion Usage Description	
Multidex Build	False
Name	WorkWithDe
Obfuscate Application	False
Object Visibility	Public
Offline Data Base Access L	Off
Offline Database	(none)
Offline Synchronization Lo	Off

Tem que configurar sua propriedade **Main program = True** e, sobre o objeto, clicar com o **botão direito e Run**

Aplicar Pattern em Transação

botão direito e Run:



☑ Apply this pattern on save

Level (Country)

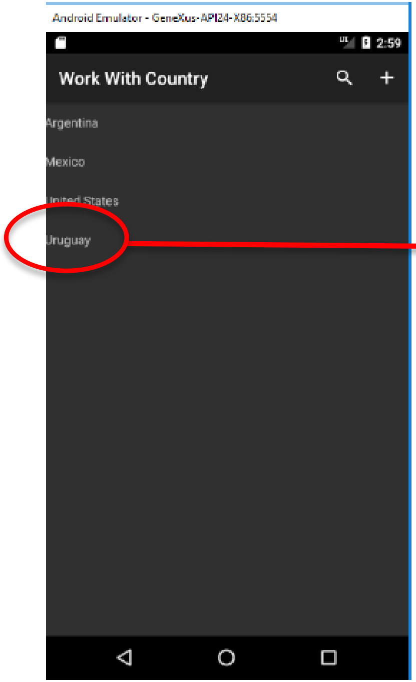
List

Detail

Section (General)

Section (Place)

Section (Sale)



Android Emulator - GeneXus-API24-X86:5554

2:59

Work With Country

Argentina

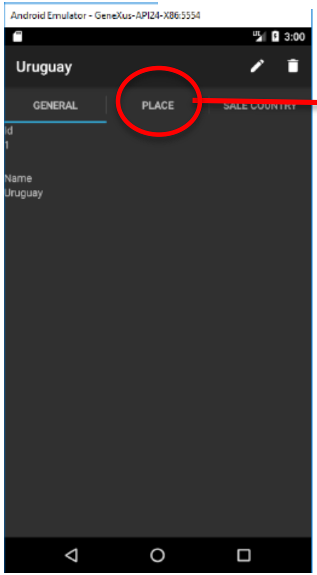
Mexico

United States

Uruguay

GENERAL

Escolho 1 country



Android Emulator - GeneXus-API24-X86:5554

3:00

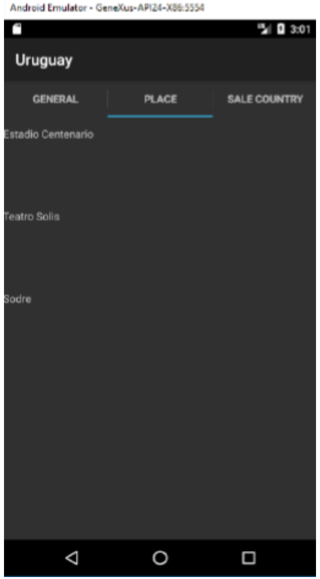
Uruguay

GENERAL | PLACE | SALE COUNTRY

Id
1

Name
Uruguay

PLACE



Android Emulator - GeneXus-API24-X86:5554

3:01

Uruguay

GENERAL | PLACE | SALE COUNTRY

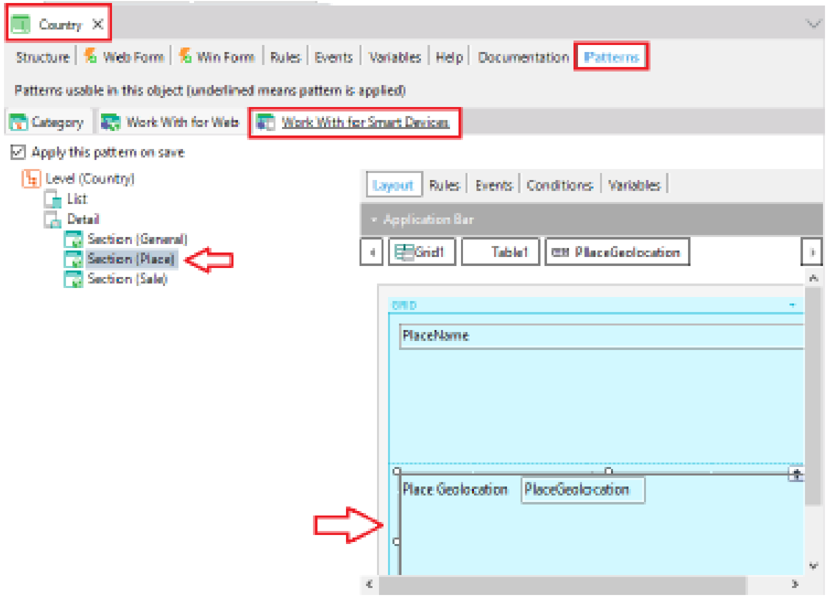
Estadio Centenario

Teatro Solis

Sodre

Aplicar Pattern em Transação

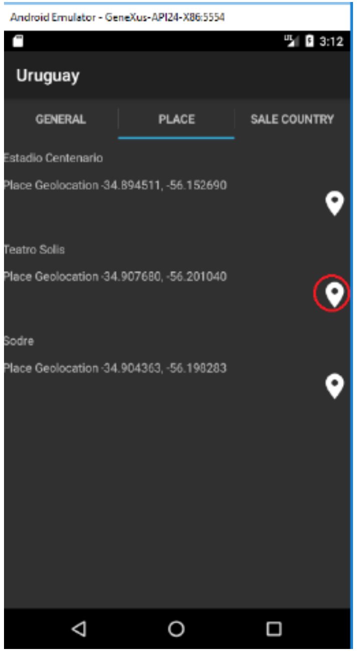
Queremos, para cada lugar, além de ver seu nome, ver sua geolocalização.



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

☑ Apply this pattern on save

- Level (Country)
  - List
  - Detail
    - Section (General)
    - Section (Place)
    - Section (Sale)



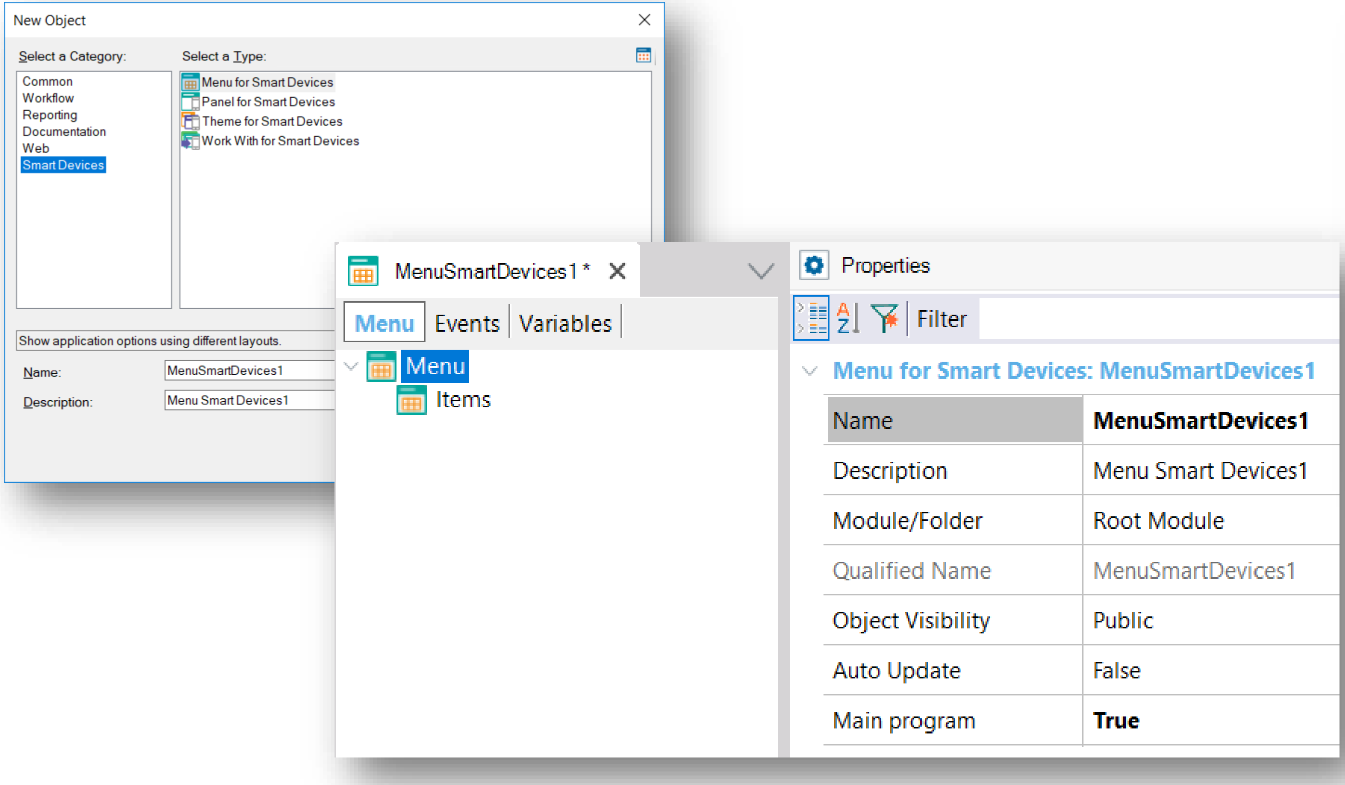
Estadio Centenario
Place Geolocation -34.894511, -56.152690

Teatro Solis
Place Geolocation -34.907680, -56.201040

Sodre
Place Geolocation -34.904363, -56.198283

Menu de acesso: Menu for Smart Devices

Criação:



New Object

Select a Category:
- Common
- Workflow
- Reporting
- Documentation
- Web
- Smart Devices

Select a Type:
- Menu for Smart Devices
- Panel for Smart Devices
- Theme for Smart Devices
- Work With for Smart Devices

Show application options using different layouts.

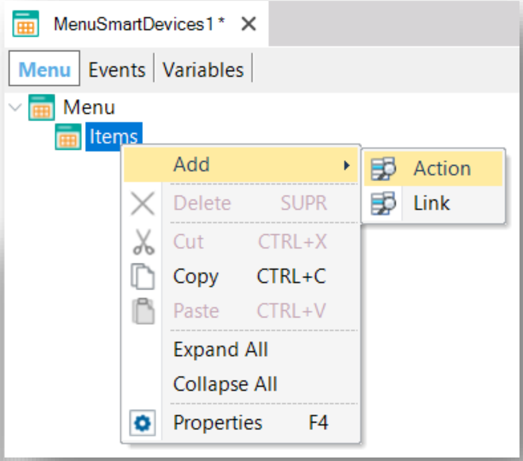
Name: MenuSmartDevices1

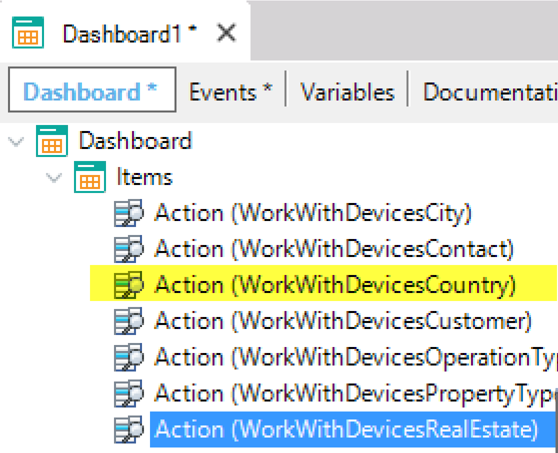
Description: Menu Smart Devices1

Menu for Smart Devices: MenuSmartDevices1

Name	**MenuSmartDevices1**
Description	Menu Smart Devices1
Module/Folder	Root Module
Qualified Name	MenuSmartDevices1
Object Visibility	Public
Auto Update	False
Main program	**True**

Add action…



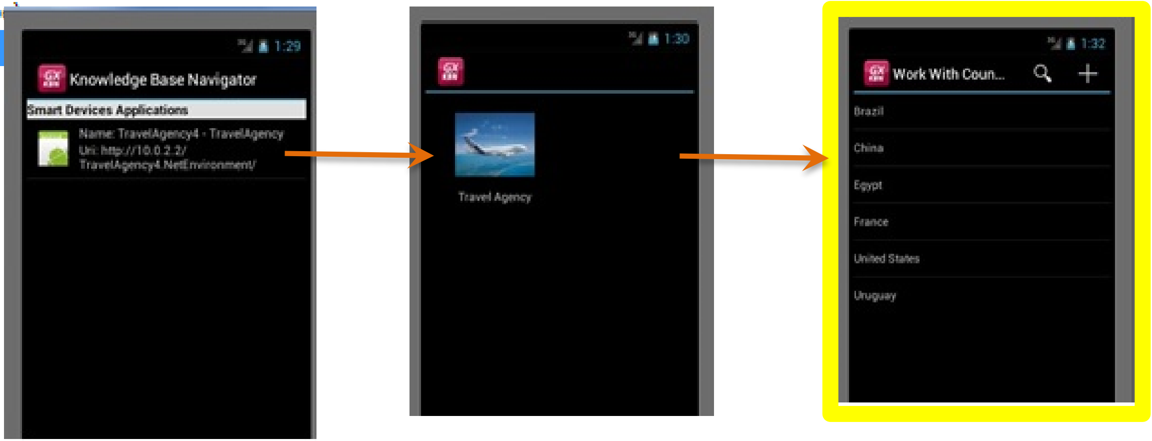


Dashboard1 * ×

Dashboard * | Events * | Variables | Documentation

- Dashboard
  - Items
    - Action (WorkWithDevicesCity)
    - Action (WorkWithDevicesContact)
    - Action (WorkWithDevicesCountry)
    - Action (WorkWithDevicesCustomer)
    - Action (WorkWithDevicesOperationType)
    - Action (WorkWithDevicesPropertyType)
    - Action (WorkWithDevicesRealEstate)

Ao pressionar F5...



1:29
Knowledge Base Navigator
Smart Devices Applications
Name: TravelAgency4 - TravelAgency
Uri: http://10.0.2.2/TravelAgency4.NetEnvironment/
1:30
Travel Agency
1:32
Work With Coun...
Brazil
China
Egypt
France
United States
Uruguay

GeneXus™

Videos	training.genexus.com
Documentation	wiki.genexus.com
Certifications	training.genexus.com/certifications