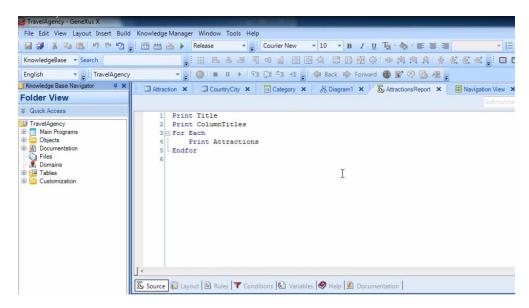
Comando For Each simple y For Each anidado para listar info relacionada

Volvamos ahora a la sección Source del procedimiento



Una cosa que nos había quedado pendiente es que queríamos que las atracciones salgan ordenadas en forma alfabética, por nombre de atracción.

Y esto lo logramos simplemente escribiendo al lado del comando "For Each", la cláusula "order AttractionName" :

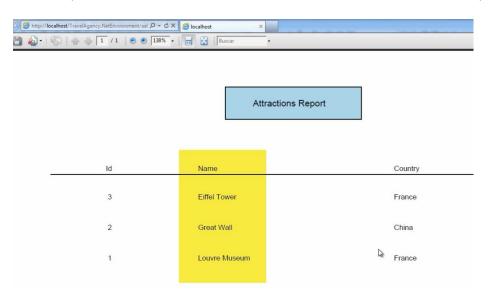
```
AttractionsReport* X

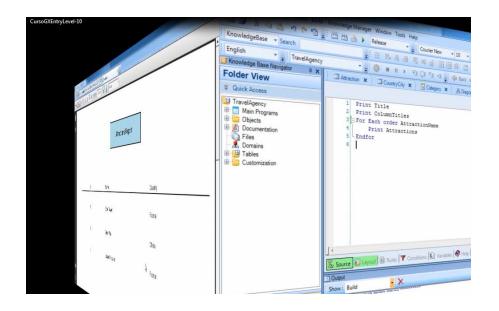
1     print Title
2     print ColumnTitles
3     For each order AttractionName
4     print Attractions
Endfor
6
```

Vamos a ejecutar el procedimiento para verlo ...

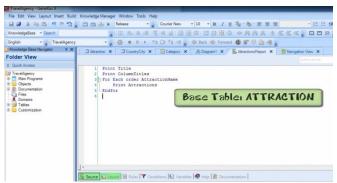
Presionamos el botón derecho del mouse, seleccionamos "Run With this Only"

Y vemos que las atracciones turísticas son listadas ordenadas alfabéticamente por nombre.

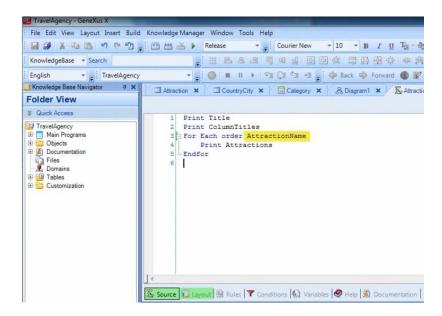




Ahora vamos a observar algo interesante. El For each navega una tabla física determinada



que en este caso es ATTRACTION y habíamos ordenado sus datos usando el atributo AttractionName



que también está en la tabla ATTRACTION.

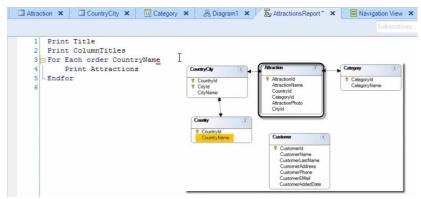
Pero GeneXus nos permite ordenar por el valor de un atributo que no está en la tabla ATTRACTION, sino en su tabla extendida.

Cambiemos el atributo que sucede al order, por CountryName

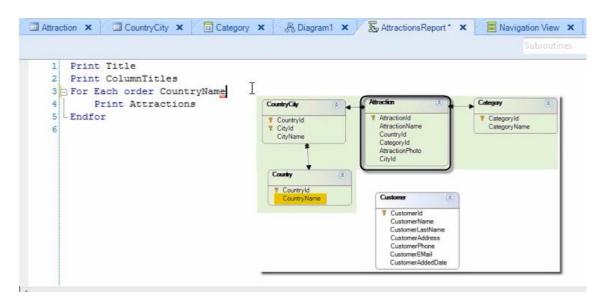
```
AttractionsReport * X

1    print Title
2    print ColumnTitles
3    For each order CountryName
4    print Attractions
5    endfor
6
```

Este atributo no se encuentra físicamente en la tabla base del For each,

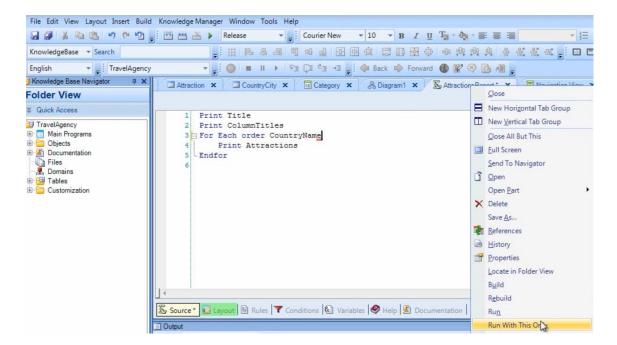


pero se encuentra en la tabla extendida de la tabla base

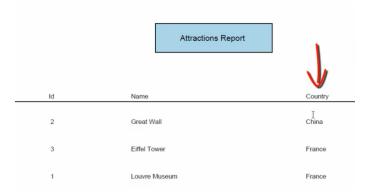


y por lo tanto podemos ordenar por él.

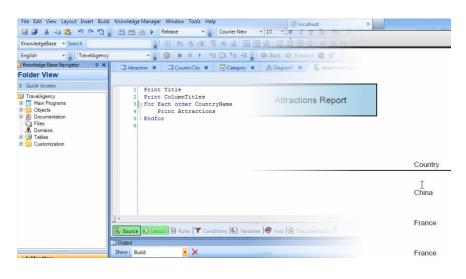
Vamos a ejecutar el procedimiento para ver el resultado



Vemos que salen listadas las atracciones ordenadas alfabéticamente por nombre de país!



Ahora bien, ¿qué pasaría si en la agencia de viajes nos piden que listemos solamente las atracciones turísticas de Francia? Volvamos a GeneXus para resolverlo.



Solamente agregaremos al comando For Each, una claúsula llamada Where, para que filtre y muestre únicamente los datos que cumplan con la condición deseada.

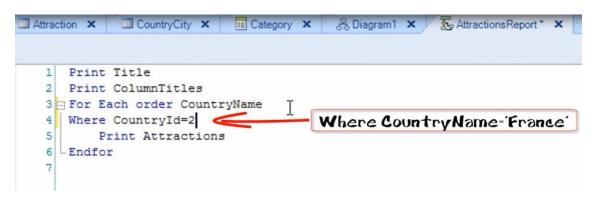
Nos posicionamos en el renglón siguiente al For Each y escribimos Where...CountryId=2

```
Attraction X CountryCity X Category X Diagram1 X AttractionsReport* X

Print Title
Print ColumnTitles
For Each order CountryName
Where CountryId=2
Print Attractions
Endfor
```

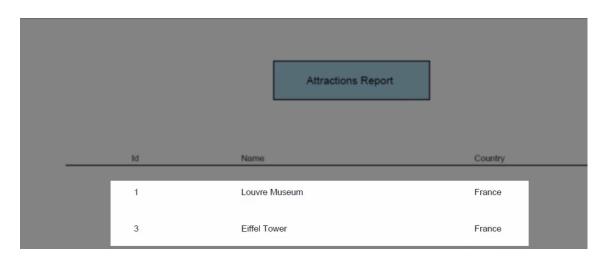
ya que sabemos que el Id de Francia era el 2

También podríamos haber puesto Where CountryName='France'



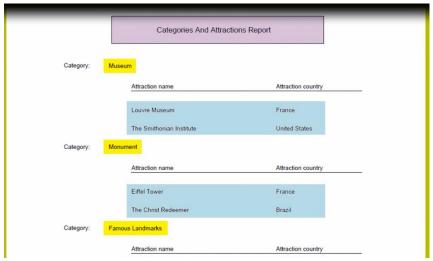
Salvamos y ejecutamos el reporte....

Y efectivamente salen en el listado solamente las atracciones de Francia!

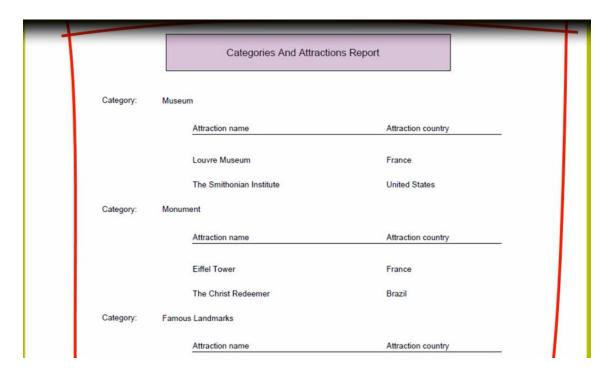


Muy bien.... hasta ahora hemos estado definiendo un procedimiento, que tiene un único comando For each, el cual como vimos accede a los datos de una tabla base y su tabla extendida.

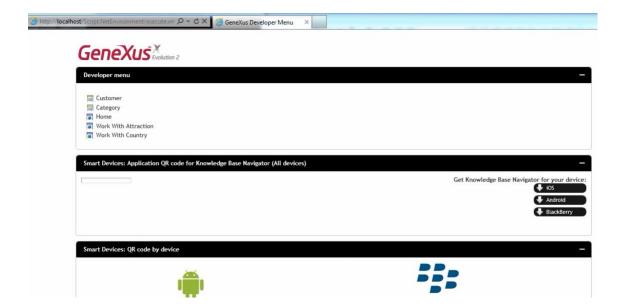
Ahora supongamos que la agencia de viajes nos solicita un listado que muestre a todas las categorías de atracciones y para cada categoría, la lista de atracciones turísticas que hay ingresadas



con este formato:



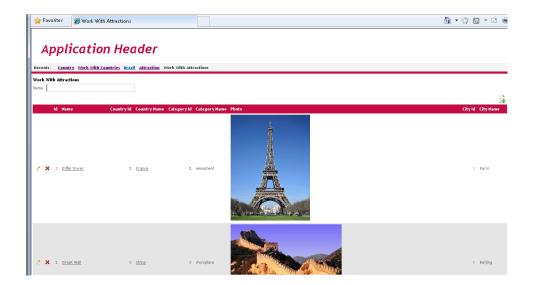
Antes de resolver este pedido, vamos a ejecutar la aplicación para agregar un par de atracciones turísticas. Presionamos F5...



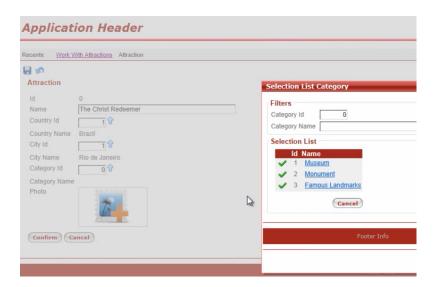
y aquí vemos la nueva pantalla de inicio, que ahora genera la versión Evolution 2 de GeneXus, con la que estamos trabajando.

Vamos a ejecutar a nuestro Trabajar Con Atracciones



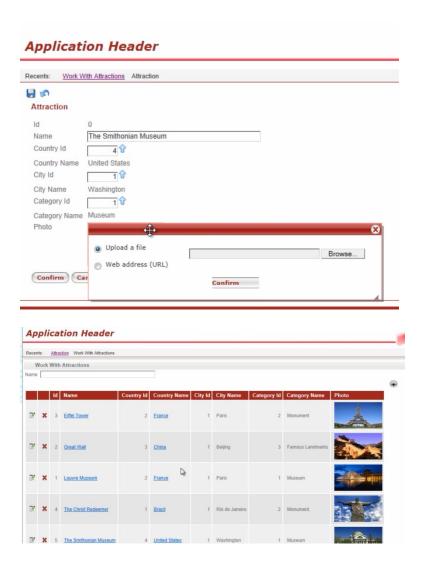


Agregamos al Cristo Redentor, que está en Brasil, en la ciudad de Rio de Janeiro, su categoría es Monument



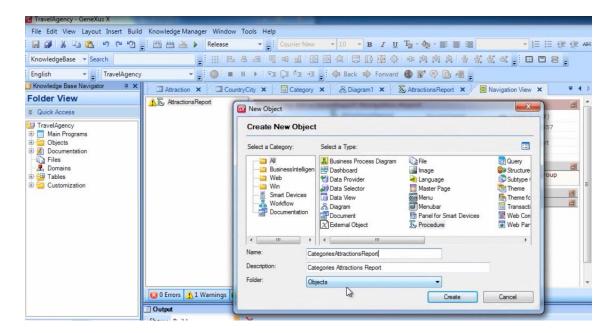
y cargamos su imagen.

Ahora ingresamos al Museo Smithoniano, que está en Estados Unidos, en Washington y elegimos su imagen.

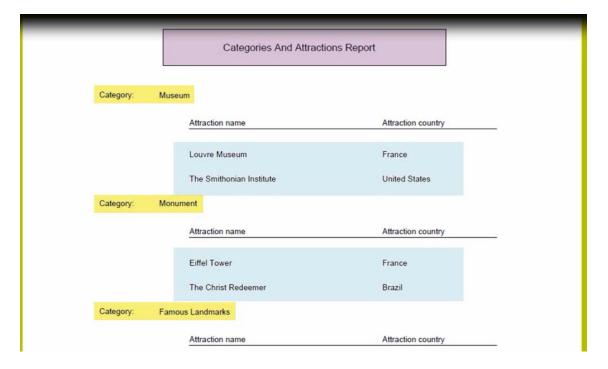


Y ahora sí, pasemos a definir el listado que nos solicitaron.

Vamos a crear un objeto procedimiento. Seleccionamos File/New/Object, elegimos procedure y le damos el nombre "CategoriesAttractionReport"



Ahora volvamos a observar lo que queremos ver en el listado...



Queremos mostrar cada categoría ingresada y para cada categoría, todas las atracciones que pertenecen a esa categoría.

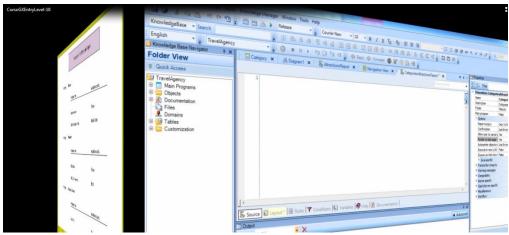
Observemos que tenemos que mostrar 3 contenidos fijos



y 2 contenidos con datos que tenemos que extraer de la base de datos

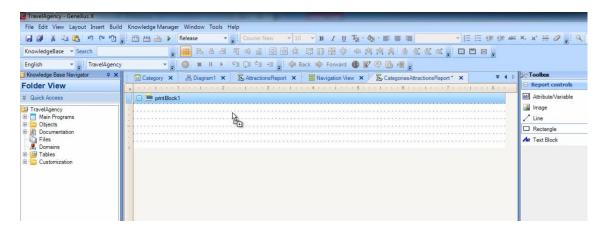


Empecemos por definir el layout del procedimiento basándonos en lo que nos pidieron ver, y luego veremos cómo codificar el source

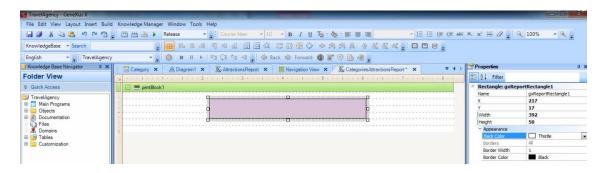


Utilizamos el printblock ya creado y le agregamos el título del listado.

Arrastramos un control Rectangle



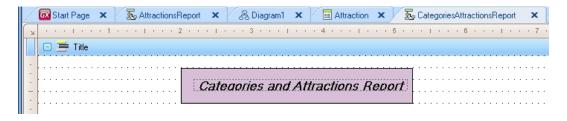
le cambiamos su propiedad "Back color"



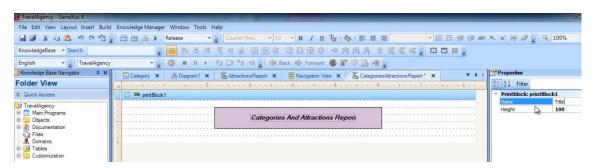
le insertamos dentro un Textblock



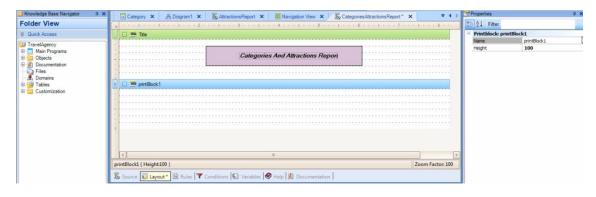
y digitamos el título: Categories and Attractions Report



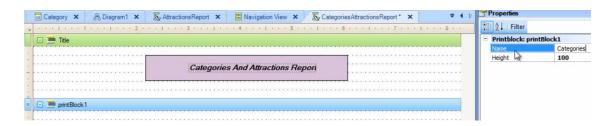
Nombramos a este printblock: Title



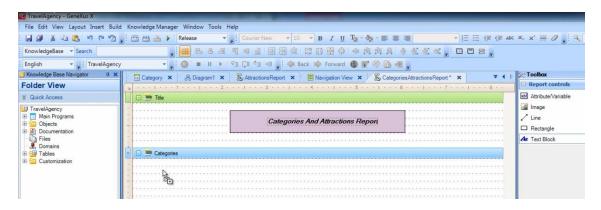
Ahora agregamos otro printblock para mostrar cada nombre de categoría



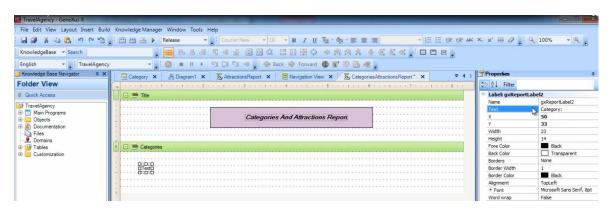
Le damos el nombre: Categories



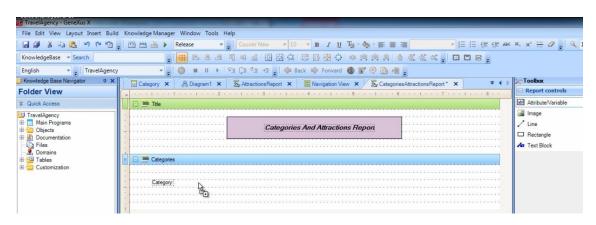
Arrastramos un control de tipo textblock al printblock



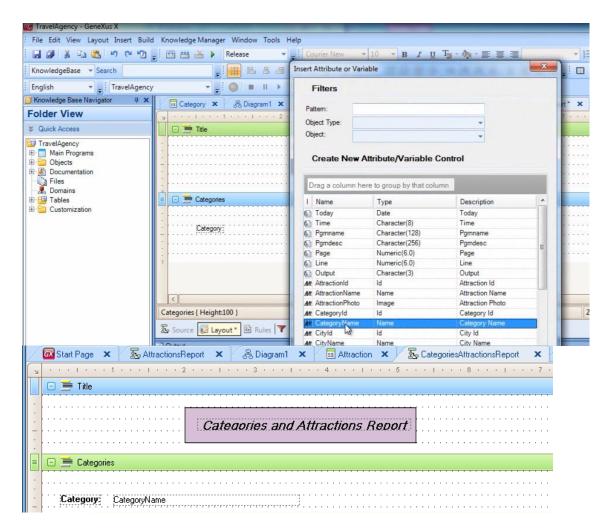
le ponemos el texto "Category"



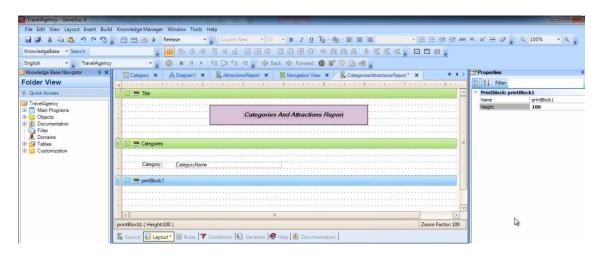
y arrastramos al lado, un control de tipo atributo/variable



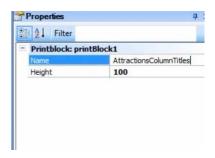
para mostrar el valor del atributo CategoryName



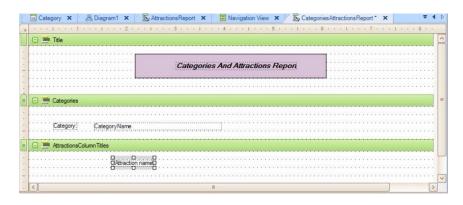
Creamos otro printblock para mostrar los títulos de las columnas.



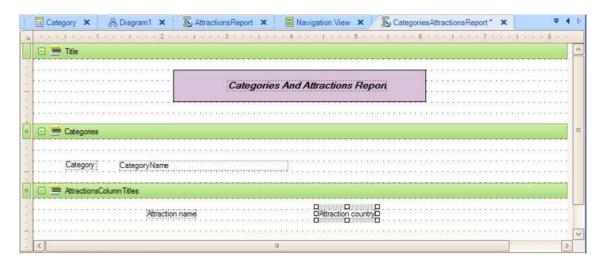
Le damos el nombre AttractionsColumnTitles



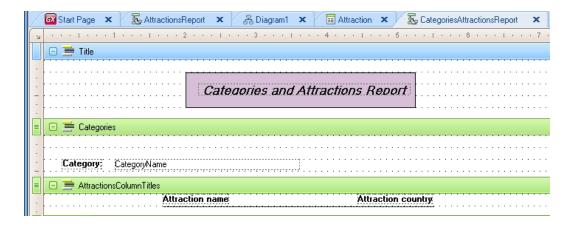
y le agregamos el texto Attraction name



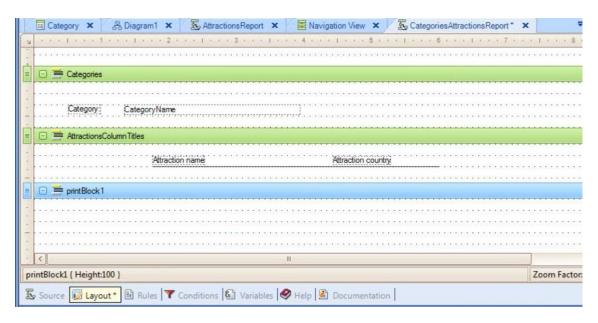
el texto Attraction country



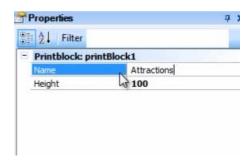
y la línea debajo de estos textos:



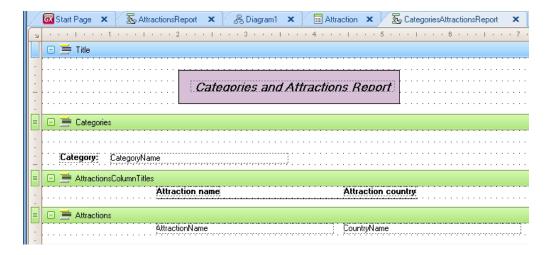
Ahora insertamos un último printblock para mostrar en el mismo los datos de las atracciones.



Le damos el nombre "Attractions"

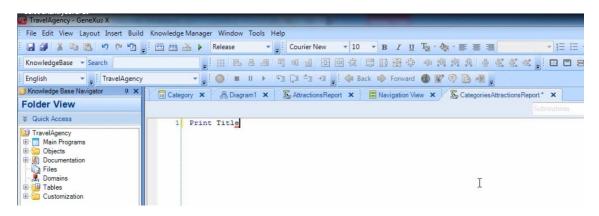


e incluimos en el mismo a los atributos AttractionName y CountryName, debajo de los títulos respectivos:



Ahora pasemos a la sección Source.

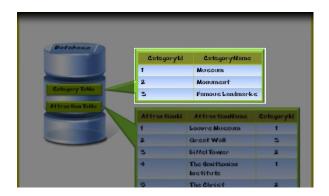
Escribamos la primer instrucción para imprimir el título del listado: Print Title



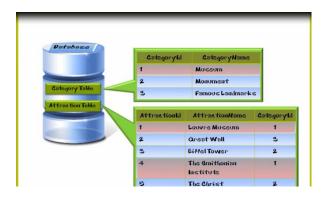
Y ahora detengámonos a ver cómo seguimos



Como tenemos que navegar las categorías

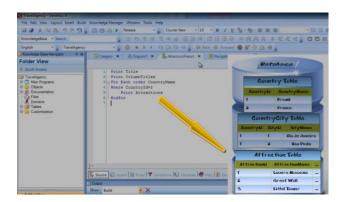


y para cada una de ellas, **navegar <u>varias</u>** atracciones que pertenecen a dicha categoría

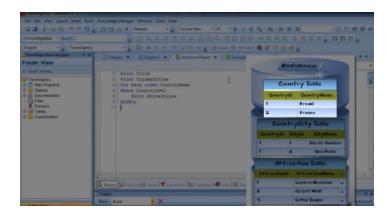


este caso es diferente al del listado anterior que hemos resuelto.

En el listado anterior como hemos visto, navegábamos las atracciones

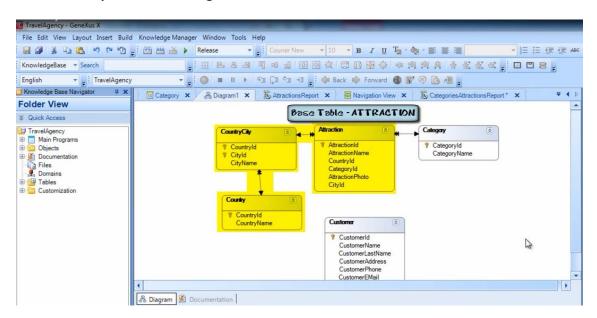


y dado que cada atracción tiene 1 sólo país, podíamos recuperar el nombre del país



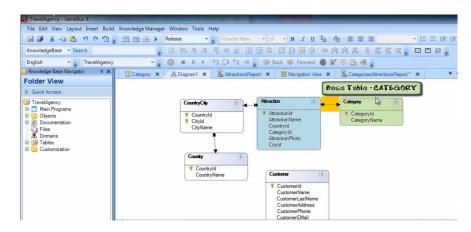
de cada atracción, por estar en la tabla extendida de la tabla base que navegábamos.

Siempre que disponemos de la información que queremos mostrar, en la tabla extendida de la tabla base que estamos navegando



la podemos referenciar directamente en el mismo For each.

En cambio si estamos navegando una tabla y necesitamos para cada registro accedido,



navegar <u>varios</u> <u>registros relacionados</u> que están en otra tabla que no pertenece a la tabla extendida de la tabla que estamos navegando, necesitamos escribir otro For each dentro del primero, para recorrer el conjunto de registros relacionados.

Volvamos al source.

Comencemos a escribir un 1er For each, para navegar y mostrar las categorías

Escribimos dentro del For each: Print Categories

```
Cutegory X Diagram1 X AttractionsReport X Navigation View X Categories Attractions Report X Subroutines

1 Print Title
2 For Each
3 Print Categories

1 Categories Categories

Categories Categories

Category Categories
```

y como el printblock Categories, solamente tiene incluído al atributo CategoryName

```
Category X Diagram1 X AttractionsReport X Navigation View X CategoriesAttractionsReport* X

Subroutin

Print Title
Print Categories
Print Categories

Categories

CategoryName

CategoryName

CategoryName
```

GeneXus entenderá que la tabla base de este For each será CATEGORY

Dentro de este For each que navega las categorías, luego de mostrada una categoría, queremos navegar el conjunto de atracciones pertenecientes a dicha categoría... entonces necesitamos escribir el 2do For each que mencionamos, para recorrer el conjunto de atracciones pertenecientes a la categoría que se venía navegando.

Justo antes de navegar el conjunto de atracciones de la categoría, habría que mostrar los títulos que encabezarán a las atracciones que mostraremos, así que escribamos la instrucción **Print AttractionsColumTitles**

```
Start Page X AttractionsReport X B Diagram1 X Attract

1 print Title
2 For each
3 print Categories
4 Print AttractionsColumnTitles
```

Y ahora sí escribamos el 2do For each, dentro del cuerpo del 1ero.

Escribimos For each

```
Category X Diagram1 X AttractionsReport X Navigation View X CategoriesAttractionsReport X

Subroutin

Print Title
Print Categories
Print Categories
Print AttractionsColumnTitles
For Each
```

dentro del mismo Print y el nombre del printblock Attractions

```
Category X Diagram1 X AttractionsReport X Navigation View X Categories Attractions Report* X

Subroutine

1 Print Title
2 For Each
9 Print Attractions Column Titles
For Each
Print Attractions
```

 2 age23

```
Category X Diagram1 X AttractionsReport X Navigation View X CategoriesAttractionsReport* X

Subroutine

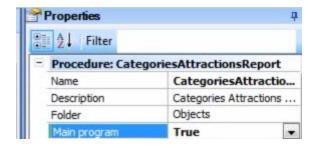
1 Print Title
2 For Each
3 Print Categories
4 Print AttractionsColumnTitles
5 For Each
6 Print Attractions
EndFor

I
```

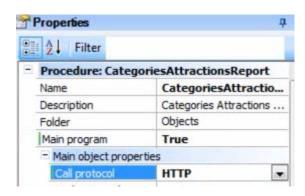
y el último **Endfor** para cerrar la primera navegación.

```
🐼 Start Page 🗶
                AttractionsReport X
                                      옳 Diagram1 🗙
                                                     Attraction X
                                                                     ₹ CategoriesAttractionsReport *
       print Title
    2 For each
    3
            print Categories
            Print AttractionsColumnTitles
    5
            For each
    6
                print Attractions
    7
            endfor
    8
      endfor
    9
```

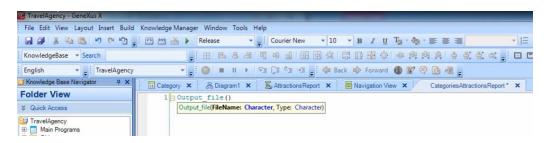
Recordemos que para que salga el listado con formato PDF, tenemos que configurar la propiedades del reporte "Main program" con valor True



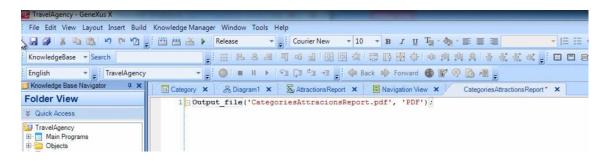
en "Main object properties", configuramos para la propiedad "Call protocol" el valor "HTTP"



 $P_{age}24$

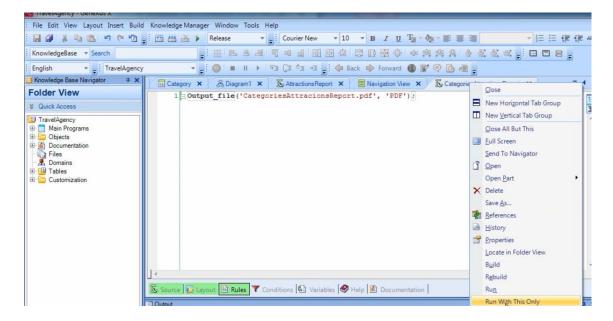


Completamos el nombre del archivo del listado "CategoriesAttractionsReport.PDF" y el formato PDF.

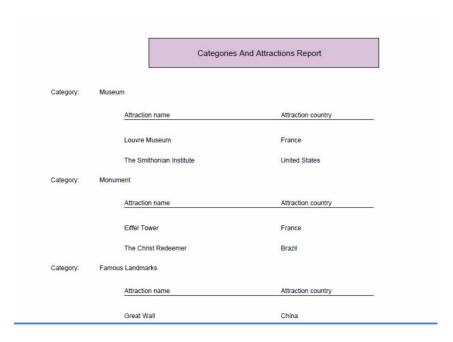


Salvamos...

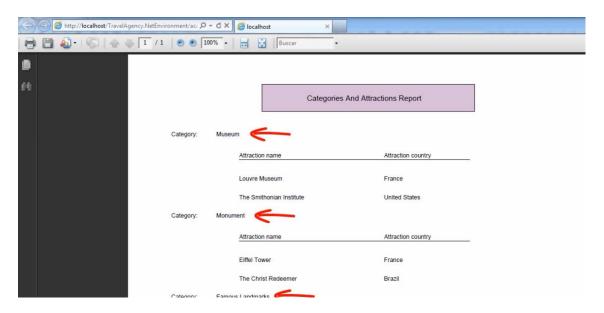
Y vamos a ejecutar a este listado



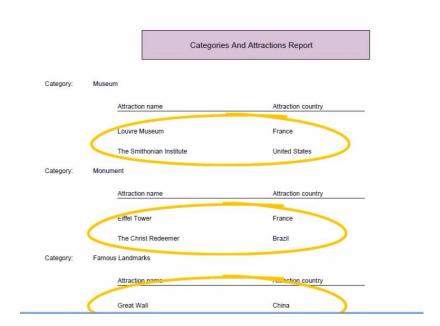




Observamos que se listaron todas las categorías que habíamos ingresado



y para cada una de ellas se listaron las atracciones que pertenecen a dicha categoría!



O sea, exactamente lo que nos solicitaron!

Ahora bien, ¿cómo supo GeneXus cuáles atracciones debía mostrar para cada categoría si nosotros no le agregamos a nuestro source ninguna condición explícita que indique algo al respecto?

```
🐼 Start Page 🗶 🗏 AttractionsReport 🗴 🦂 Diagram1 🗴 🔚 Attraction 🗴 🛴 CategoriesAttractionsReport *
        print Title
     2
       For each
     3
             print Categories
     4
             Print AttractionsColumnTitles
     5
             For each
     6
                  print Attractions
     7
             endfor
     8
        endfor
👼 Source * 🔯 Layout 🖹 Rules 🔻 Conditions 🛍 Variables 🥏 Help 🖺 Documentation 🛭
```

Si bien escribimos un For Each que navega las atracciones

```
Category X & Diagram1 X AttractionsReport X

Print Title
For Each
Print AttractionsColumnTitles
For Each
Print Attractions
EndFor

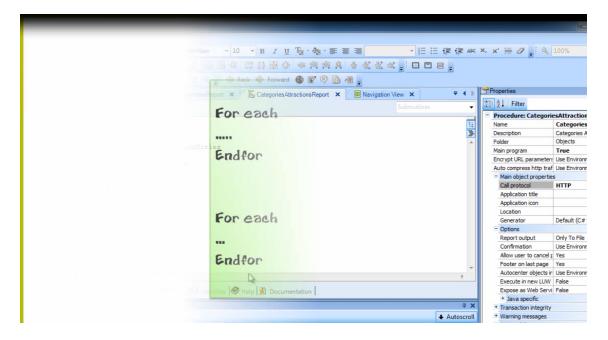
EndFor

Print Print Print Attractions
Print Attractions
Print Attractions
Print Attractions
EndFor
Print Prin
```

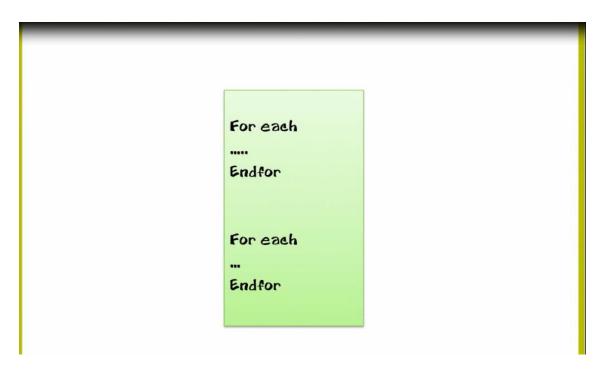
no agregamos en ningún lado una cláusula where, para filtrar aquellas atracciones que cumplan con la condición de que coincida el valor del identificador de la categoría, con el de la categoría que estamos listando.

Entonces ¿cómo hizo GeneXus para que al momento de navegar la tabla ATTRACTION donde están almacenadas todas las atracciones, recuperarnos para cada categoría, solamente aquellas que pertenecían a dicha categoría?

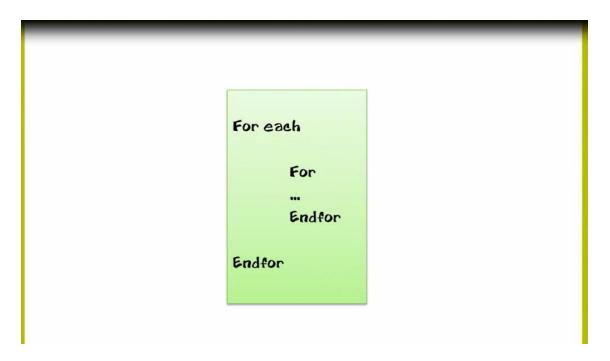
La respuesta está en la forma en que escribimos los For Each



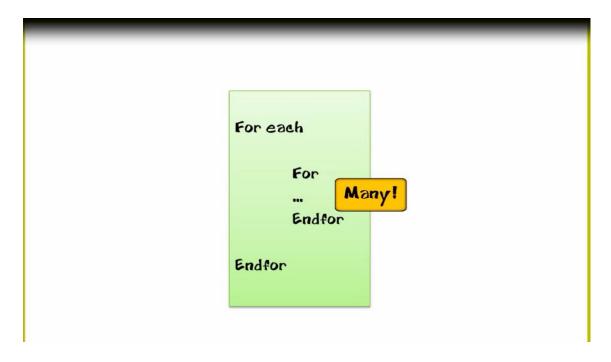
Si 2 comandos For each se escriben uno a continuación del otro, son independientes entre sí



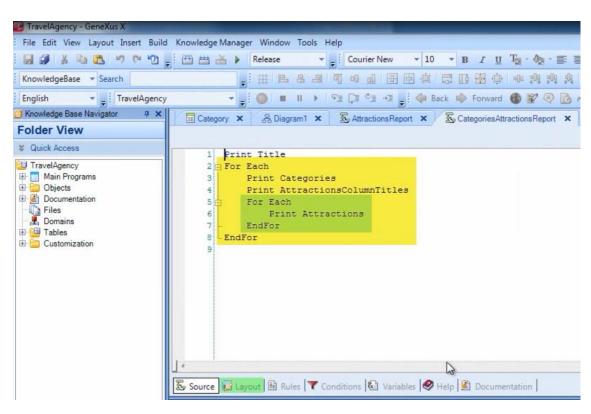
En cambio si escribimos un For each dentro de otro



es porque para cada registro de la primer navegación, queremos recorrer un conjunto de registros

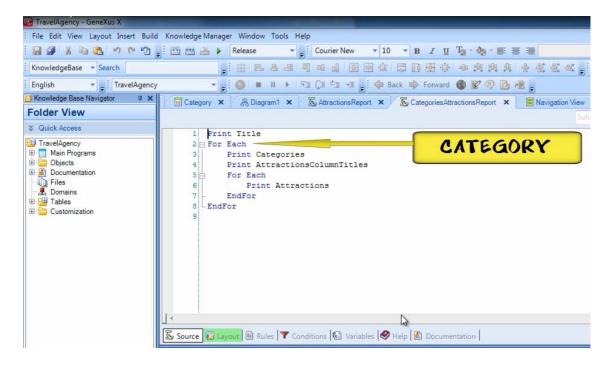


Cuando escribimos comandos For Each anidados

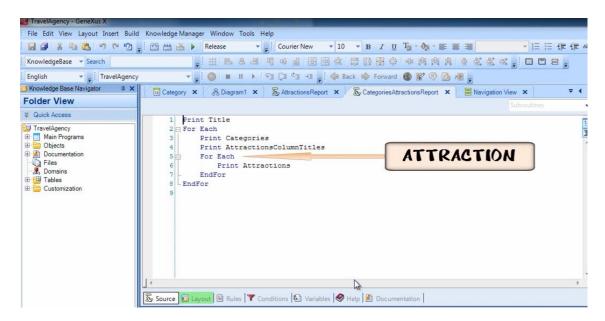


GeneXus determina para cada For each, la tabla base que navegará... y luego busca si existe alguna relación entre dichas tablas base.

En este caso, la tabla base del For each externo es CATEGORY

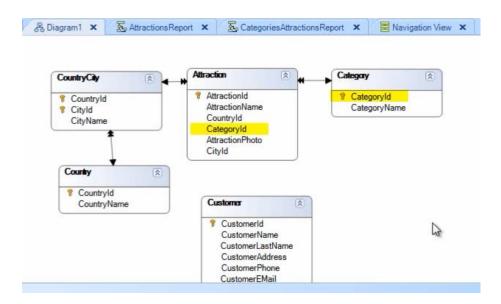


y la tabla base del For each interno es ATTRACTION

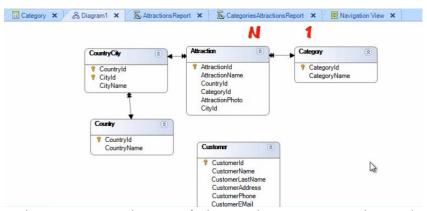


y GeneXus tiene el conocimiento de que hay un atributo en común en ambas tablas: **CategoryId,** que es <u>llave primaria</u> en CATEGORY y <u>llave foránea</u> en ATTRACTION



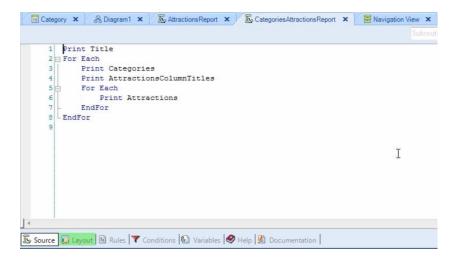


De modo que el atributo Categoryld hace que se relacionen las tablas ATTRACTION y CATEGORY, como vemos aquí, en una relación 1 a N



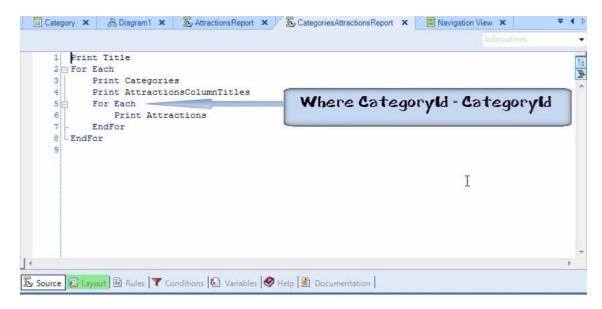
Es decir, que para cada categoría, hay muchas atracciones relacionadas.

Por lo tanto, para cada categoría navegada en el For Each externo, GeneXus ejecuta el For Each que navega la tabla de atracciones



filtrando solamente aquellas atracciones que cumplen que el valor de su Categoryld coincida con el valor del Categoryld de la categoría en la que estamos posicionados.

Es como si en el For each interno, hubiéramos escrito



Where Categoryld=Categoryld.... pero no tenemos que ponerlo, porque GeneXus lo detecta y aplica.

Hemos visto qué fácil es obtener información y desplegarla en un reporte... pero los procedimientos pueden hacer mucho más que eso! Veremos eso más adelante.