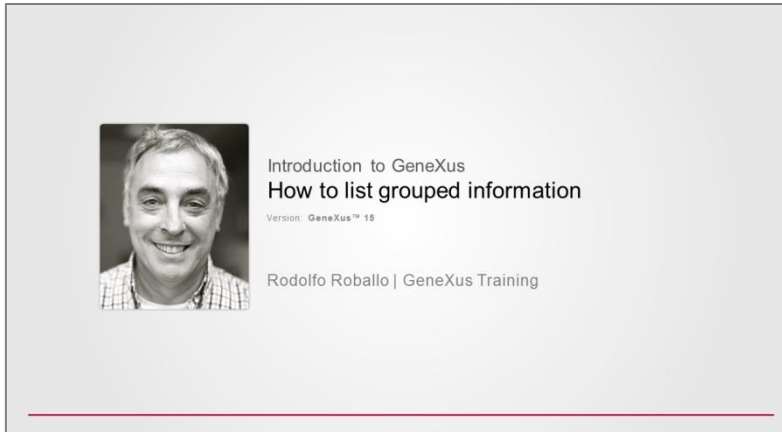


Cómo listar información agrupada



En el video anterior hemos visto un listado solicitado por la agencia de viajes, que mostraba todas las Categorías de atracciones turísticas y para cada categoría, la lista de atracciones que había ingresadas.

Categories and their Attractions

Category: Museum

Id	Name	Country	Photo
1	Louvre Museum	France	
5	Smithsonian Institute	United States	





Category: Monument

Id	Name	Country	Photo
3	Eiffel Tower	France	
4	Christ the Redemmer	Brazil	

Category: Famous Landmark


Id	Name	Country	Photo
2	The Great Wall	China	

Cambiémosle a la atracción “The Great Wall” su categoría, de modo que “Famous Landmark” ya no tenga ninguna atracción asociada.

Id	Name	Country Name	Category Name	Photo	City Name		
4	Christ the Redemmer	Brazil	Monument		Rio de Janeiro	UPDATE	DELETE
3	Eiffel Tower	France	Monument		Paris	UPDATE	DELETE
1	Louvre Museum	France	Museum		Paris	UPDATE	DELETE
5	Smithsonian Institute	United States	Museum		Washington	UPDATE	DELETE
2	The Great Wall	China	Monument		Beijing	UPDATE	DELETE



Ahora ejecutemos nuevamente el listado.

Como podemos observar, este listado nos muestra TODAS las categorías ingresadas, incluso aquellas que no tienen atracciones asociadas.






Categories and their Attractions

Category: Museum

Id	Name	Country	Photo
1	Louvre Museum	France	
5	Smithsonian Institute	United States	

Category: Monument

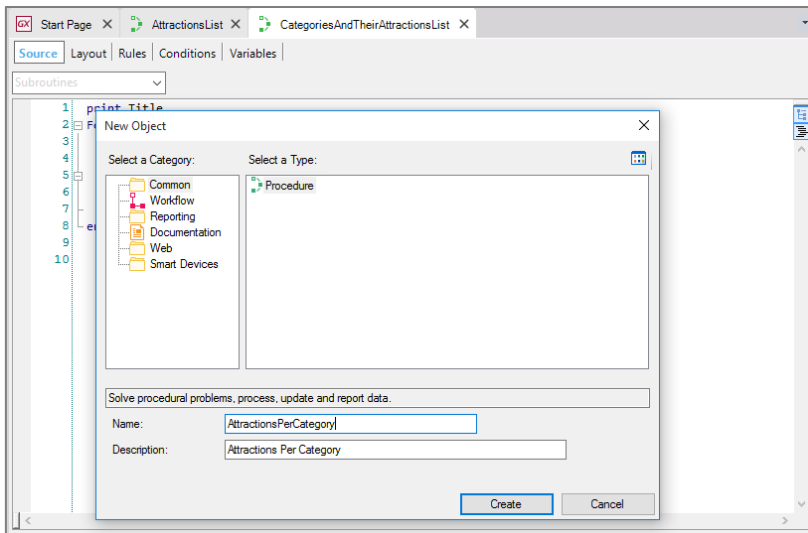
Id	Name	Country	Photo
2	The Great Wall	China	
3	Eiffel Tower	France	
4	Christ the Redemmer	Brazil	

Category: Famous Landmark

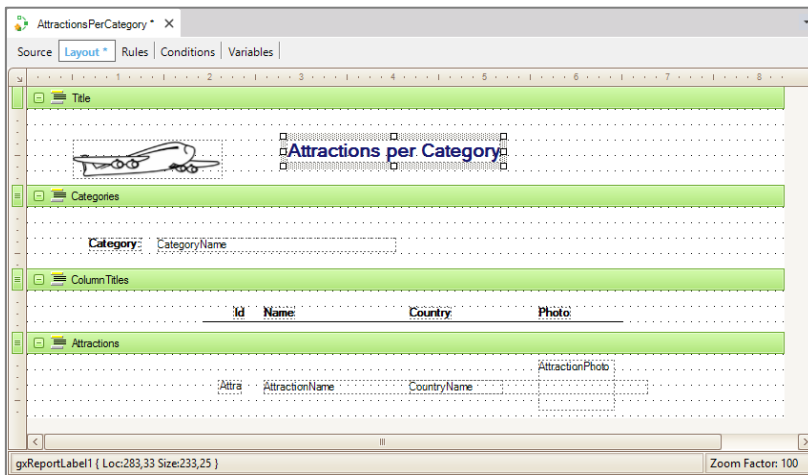
Id	Name	Country	Photo

Si eso no es lo que se desea, es decir, si lo que se quiere es mostrar solamente las categorías con atracciones, ¿cómo hacemos?

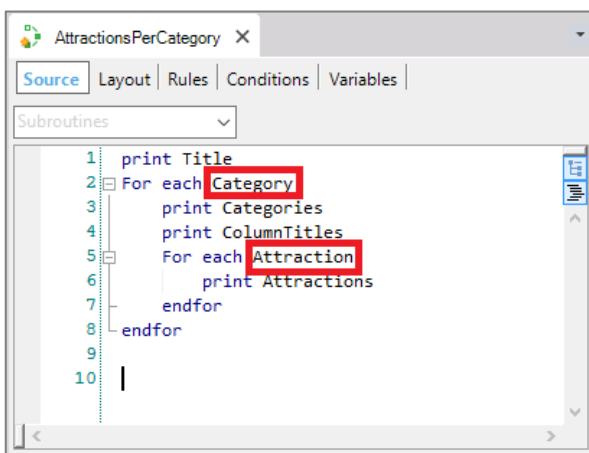
Vamos a implementar esto en otro procedimiento. Para ello salvemos el que teníamos con otro nombre.



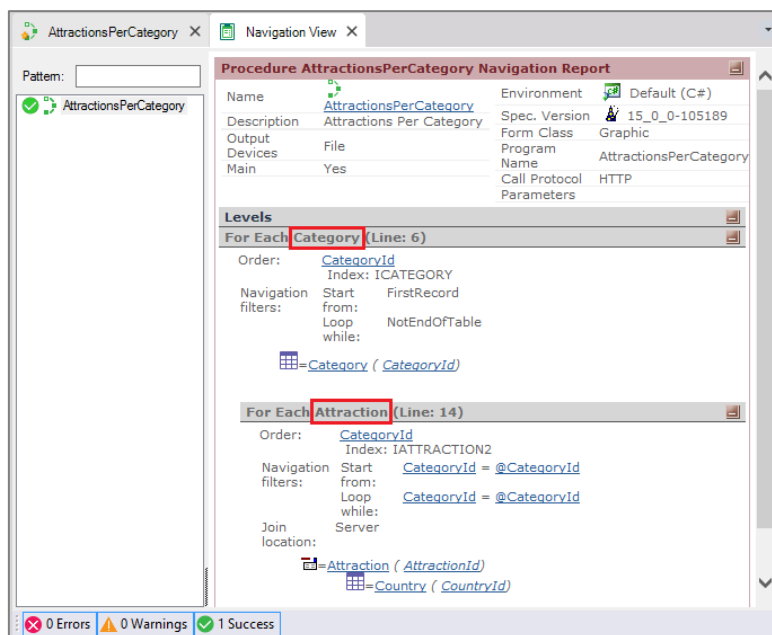
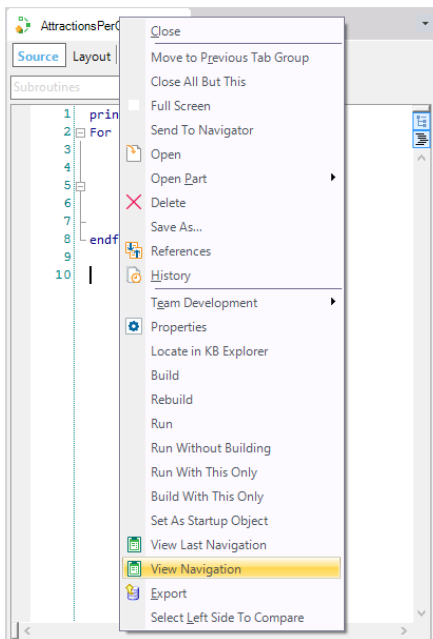
Y cambiémosle el título del text block:



Si analizamos los For Eachs que teníamos implementados, vemos que la tabla base del For Each externo es Category y la tabla base del For Each anidado es Attraction.



Podemos corroborarlo viendo el listado de navegación:



Pero de este modo, como primero se accede a la tabla Category, se imprime la información del registro y recién después se pasa a ejecutar el For each anidado, se estará imprimiendo la categoría antes de saber si tiene atracciones relacionadas o no

Esto no nos sirve. Necesitamos acceder a las categorías de las atracciones, porque es la única manera de estar seguros de que la categoría que se va a imprimir tiene por lo menos una atracción.





Foreign Key					Primary Key	
AttractionId	AttractionName	CountryId	CategoryId	...	CategoryId	CategoryName
1	Louvre Museum	2	1		1	Museum
5	Smithsonian Institute	4	1		2	Monument
2	The Great Wall	3	2		3	Famous Landmark
3	Eiffel Tower	2	2			
4	Christ the Redemmer	1	2			

La idea será agrupar las atracciones de la tabla Attraction por categoría, y luego recorrer esos grupos, imprimiendo para cada uno la categoría, (para la que habrá que acceder a la tabla Category para recuperar su nombre)

Y también imprimiendo cada atracción del grupo

Para luego pasar al siguiente grupo... y así sucesivamente.

Foreign Key					Primary Key	
AttractionId	AttractionName	CountryId	CategoryId	...	CategoryId	CategoryName
1	Louvre Museum	2	1		1	Museum
5	Smithsonian Institute	4	1		2	Monument
2	The Great Wall	3	2		3	Famous Landmark
3	Eiffel Tower	2	2			
4	Christ the Redemmer	1	2			

Category:	Museum		
1	Louvre Museum	France	
5	Smithsonian Institute	United States	
Category:	Monument		
2	The Great Wall	China	
3	Eiffel Tower	France	

En definitiva, necesitamos recorrer únicamente la tabla Attraction, primero agrupándola por categoría e imprimiendo la categoría



Y luego imprimiendo de cada grupo de categoría, (navegando la misma tabla Attractions, claro) sus atracciones.

GeneXus



Foreign Key				
AttractionId	AttractionName	CountryId	CategoryId	...
1	Louvre Museum	2	1	
5	Smithsonian Institute	4	1	
2	The Great Wall	3	2	
3	Eiffel Tower	2	2	
4	Christ the Redemmer	1	2	

Primary Key	
CategoryId	CategoryName
1	Museum
2	Monument
3	Famous Landmark

Category: Museum

1	Louvre Museum	France	
5	Smithsonian Institute	United States	

Category: Monument

2	The Great Wall	China	
3	Eiffel Tower	France	

```

For each Attraction order CategoryId
  print Categories
  For each Attraction
    print Attractions
  endfor
endfor

```

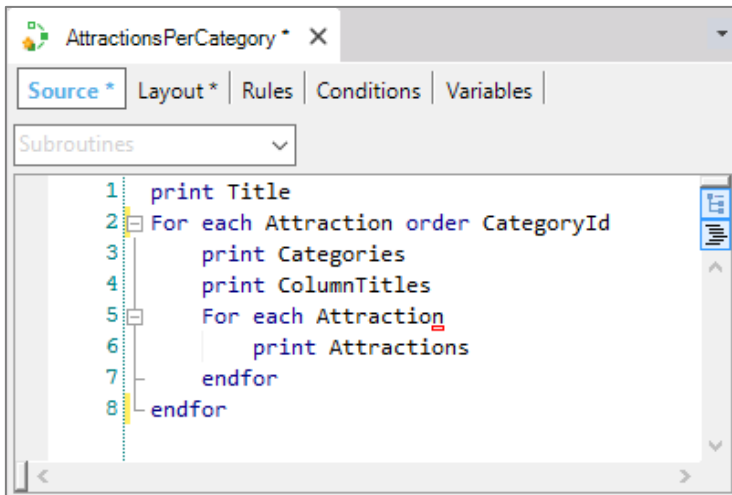
Observemos que la forma de indicarle a GeneXus que se quiere agrupar por CategoryId, es especificando la cláusula Order.

Este caso de for eachs anidados que recorren la misma tabla se conoce como **corte de control**.

Hagamos los cambios en nuestro procedimiento...

Primero cambiamos la transacción base del for each externo, por Attraction...

Y agregamos la cláusula Order, para ordenar por CategoryId, que en el caso del corte de control será además para otra cosa más fuerte: para **agrupar** por ese atributo.

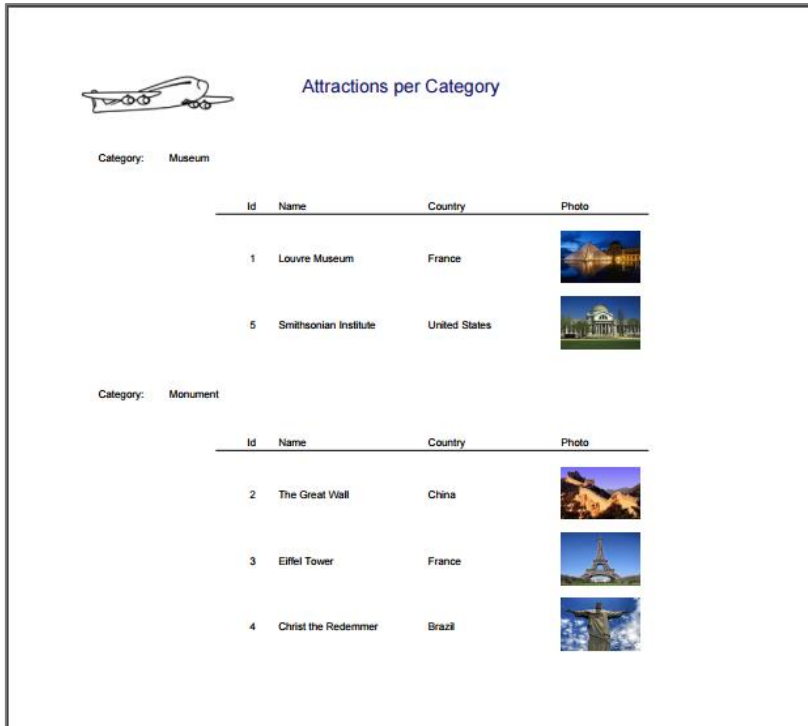


```

1 print Title
2 For each Attraction order CategoryId
3   print Categories
4   print ColumnTitles
5   For each Attraction
6     print Attractions
7   endfor
8 endfor

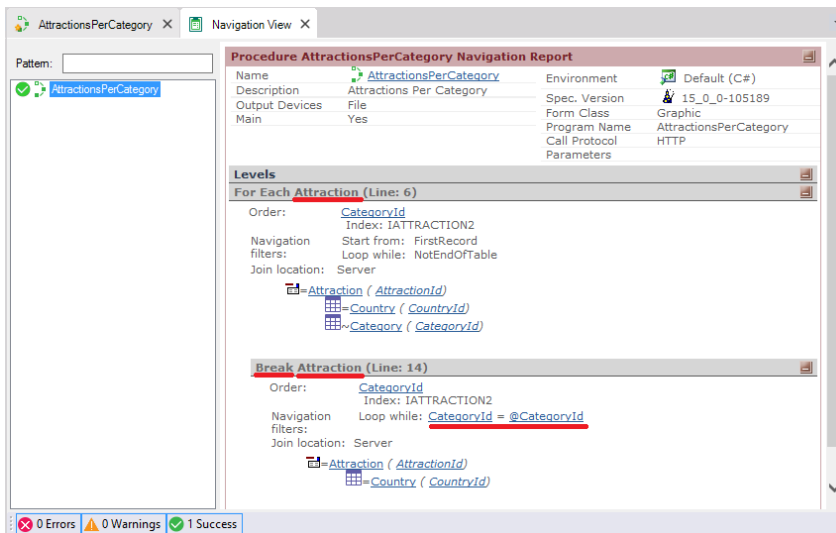
```

Ejecutemos.



Vemos que efectivamente no está siendo listada la categoría “Famous Landmark” que no tiene atracciones.

Si observamos el listado de navegación resultante:



Podemos ver que nos informa de un for each a la tabla Attraction, ordenado por CategoryId... y que será cortado, Break, por el for each anidado sobre la misma tabla, Attraction. Observemos que en este “corte” se recorren solamente las atracciones de la categoría **del** grupo.

Conceptualicemos, entonces, cómo se define un corte de control:

Se trata de For eachs anidados...

GeneXus

Conceptualization

- Break the information by some criteria: ✓ **Same base table all of them**

```
For each BaseTransaction1
  order Att1, Att2, ..., Attn
  ...
  For each BaseTransaction2
  ...
  endfor
  ...
endfor
```

Break control

Con la misma tabla base para todos los For Eachs...

GeneXus

Conceptualization

- Break the information by some criteria: ✓ **Same base table all of them**

```
For each BaseTransaction1
  order Att1, Att2, ..., Attn
  ...
  For each BaseTransaction2
  ...
  endfor
  ...
endfor
```

BASE TABLE: TABLE_A

=

BASE TABLE: TABLE_A

Yes → Grouped information

Break control

Y la cláusula Order para establecer el criterio de corte.

GeneXus

Conceptualization

- Break the information by some criteria: ✓ **Break criteria: order clause**

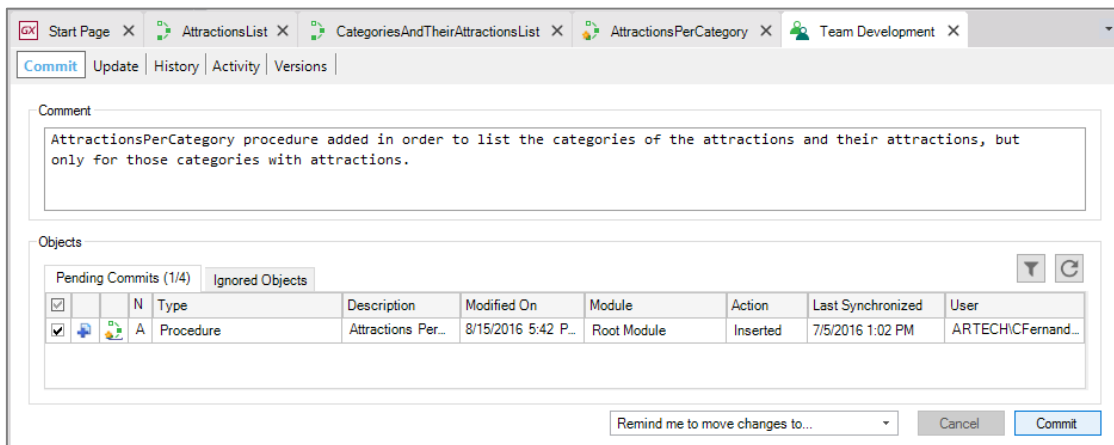
```
For each BaseTransaction1
  order Att1, Att2, ..., Attn
  ...
  For each BaseTransaction2
  ...
  endfor
  ...
endfor
```

Break criteria

BASE TABLE: TABLE_A

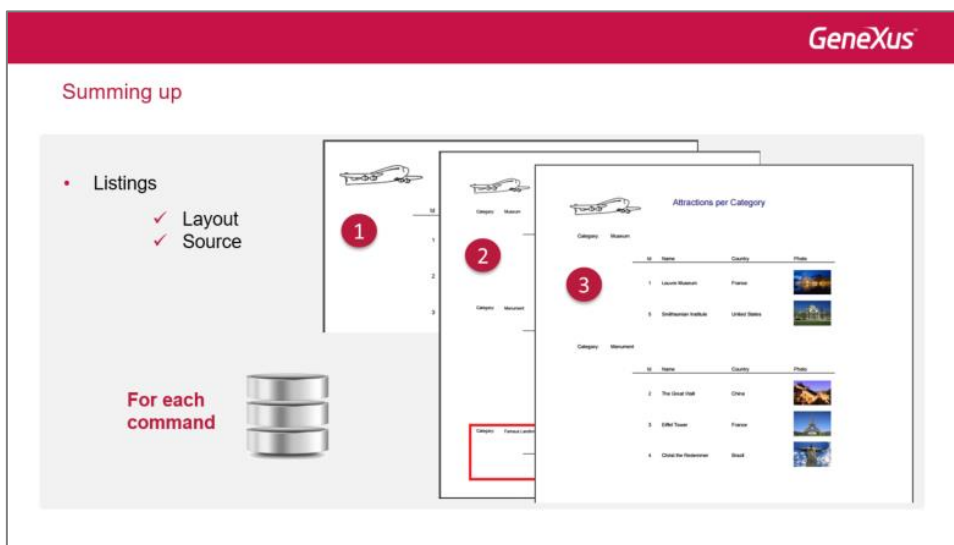
Grouped by...

Para finalizar, actualizamos los cambios en GeneXus Server.



Con estos videos hemos visto las facilidades que nos proporciona GeneXus para hacer listados simples (que navegan una única tabla), o más complejos, navegando información de varias tablas relacionadas (join), o de la misma tabla, pero agrupada por algún criterio (corte de control).

El comando que utilizamos en todos los casos para acceder a la base de datos es el **For each**.




En el primer caso, usamos un for each simple, en el que a través de su Transacción Base se infiere la tabla a ser navegada.

GeneXus




Summing up

- Listings
 - ✓ Layout
 - ✓ Source

For each command



Attractions List

ID	Name	Country	Photo
1	Leaning Tower	France	
2	The Great Wall	China	
3	Eiffel Tower	France	

1

```

For each BaseTransaction
order Att1, Att2, ..., Attn
where condition1
where condition2
...
where conditionn
MainCode
endfor
          
```


En el segundo tenemos un par de for eachs anidados, en el que a partir de transacciones base diferentes, se descubre una relación de uno a muchos entre la información de cada for each.

GeneXus






Summing up

- Listings
 - ✓ Layout
 - ✓ Source

For each command



Categories and their Attractions

Category	Attraction	ID	Name	Country	Photo
Category 1	1	Leaning Tower	France		
	4	Leaning Tower	United States		
Category 2	2	The Great Wall	China		
	3	Eiffel Tower	France		
	5	Christ the Redeemer	Brazil		

2

```

For each BaseTransaction
order Att1, Att2, ..., Attn
where condition1
where condition2
...
where conditionn
...
For each BaseTransaction
order Att1, Att2, ..., Attn
where condition1
where condition2
...
where conditionn
...
endfor
...
endfor
          
```

Mientras que en el tercero también tenemos un par de for eachs anidados, pero cuyas transacciones base **coinciden**. GeneXus entiende que se quiere “cortar” o “agrupar” la información de la tabla a ser recorrida, por el atributo o conjunto de atributos que se especifican en la cláusula order del for each exterior.

Más adelante veremos otros mecanismos para hacer consultas a la base de datos y obtener información de forma flexible y vistosa.

Summing up

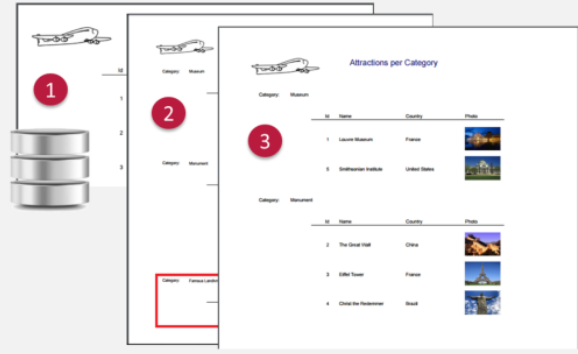
- Listings

- ✓ Layout
- ✓ Source

For each command

- Panels

- Query object



GeneXus™

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