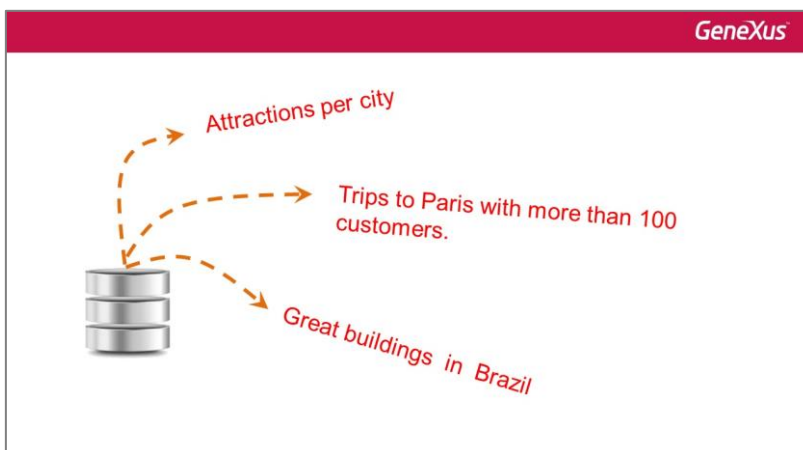


Diseñando consultas dinámicas (objeto Query)



En muchas ocasiones necesitamos realizar consultas a la base de datos para analizar la información y poder tomar decisiones.

Podemos necesitar agrupar la información por uno o varios criterios, realizar cálculos, y visualizar finalmente el resultado en una forma particular

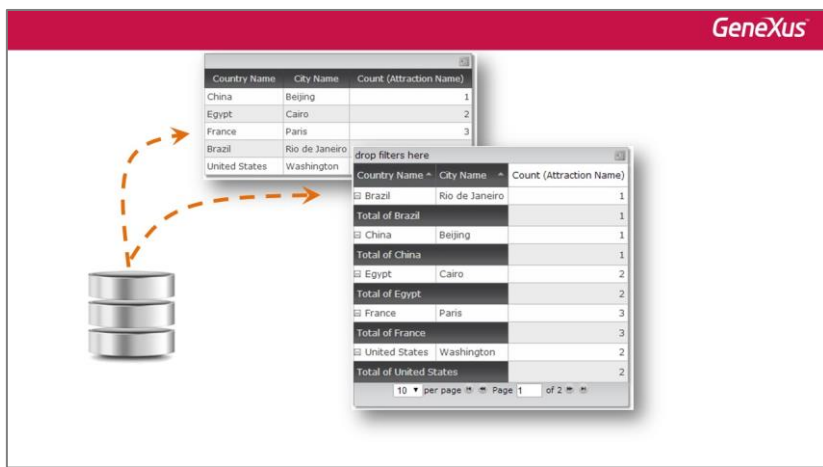


como puede ser

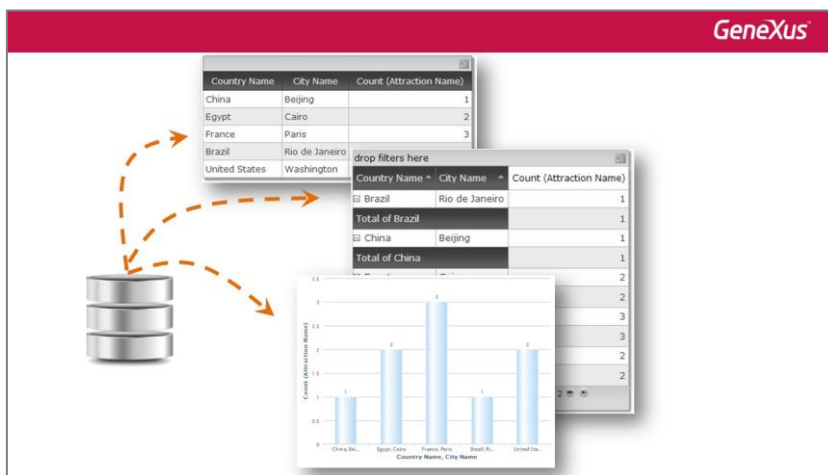
- una tabla estática, o sea filas y columnas fijas:



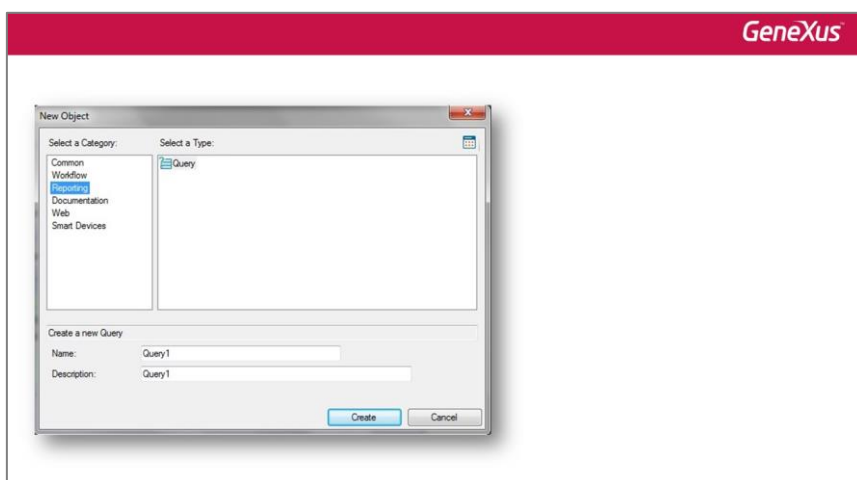
- una tabla dinámica, que permite por ejemplo intercambiar columnas de lugar y agrupar la información:



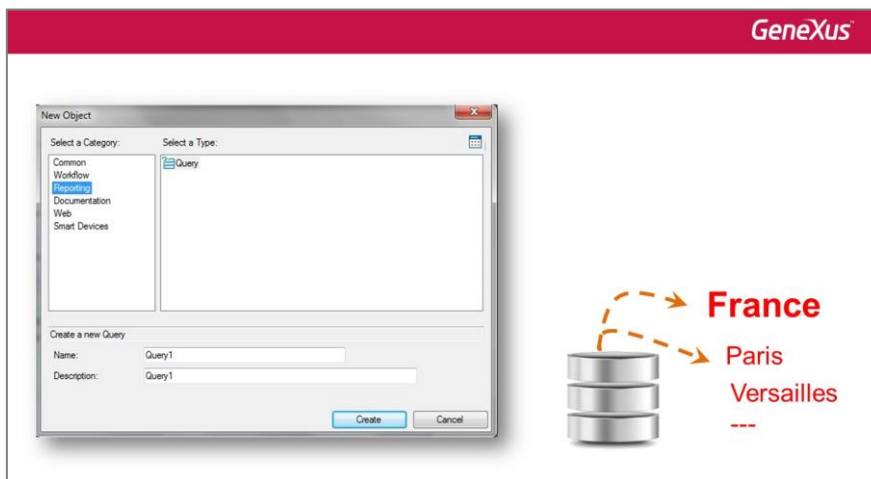
- o un gráfico:



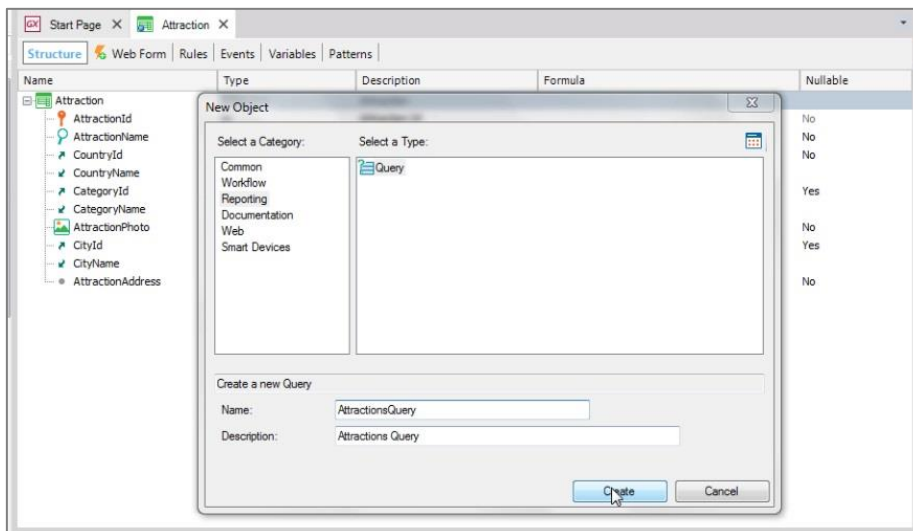
El objeto Query nos permite crear estas consultas de una forma simple e intuitiva, aumentando el valor de la información obtenida de la base de datos.



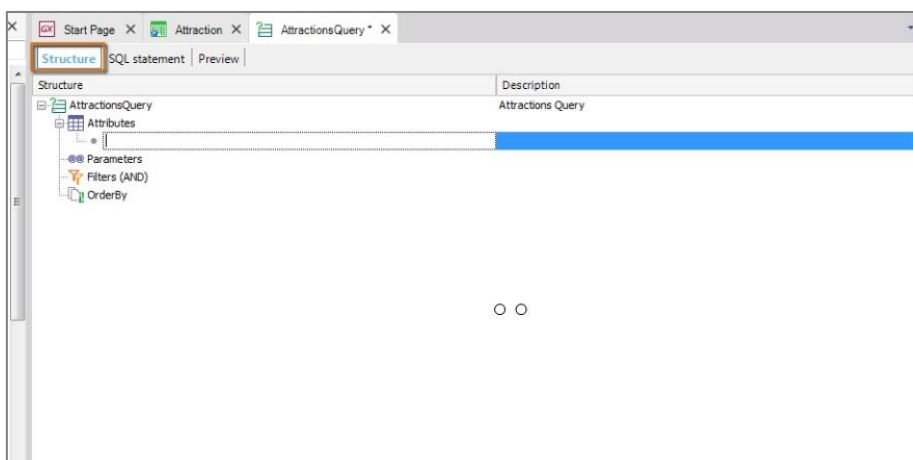
Nuestro objetivo será crear una consulta que nos devuelva solamente las ciudades de Francia, cada una de ellas con su respectiva cantidad de atracciones turísticas.



Vayamos entonces a GeneXus y comencemos por crear un objeto de tipo Query, de nombre "AttractionsQuery"



En la **Estructura**



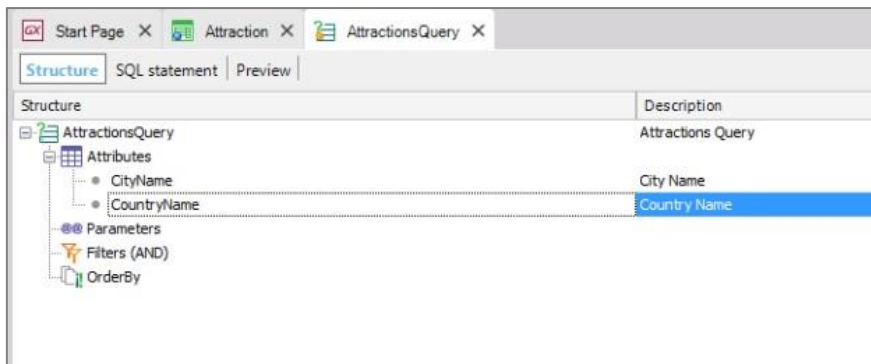
haremos la definición de la consulta.

Analicemos un poquito el significado de estos componentes

Atributos:

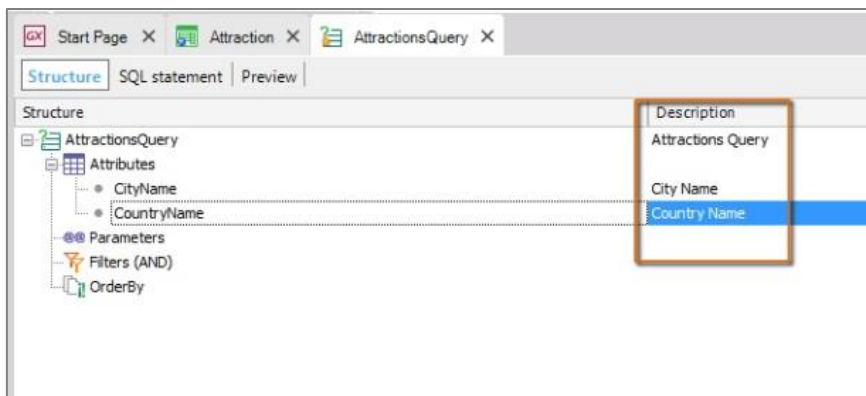
Aquí debemos mencionar todos los atributos que van a participar de la consulta. La declaración es simple y se hace en forma de lista plana.

Así que agreguemos los atributos que queremos ver en nuestra consulta, o sea, CityName y CountryName:



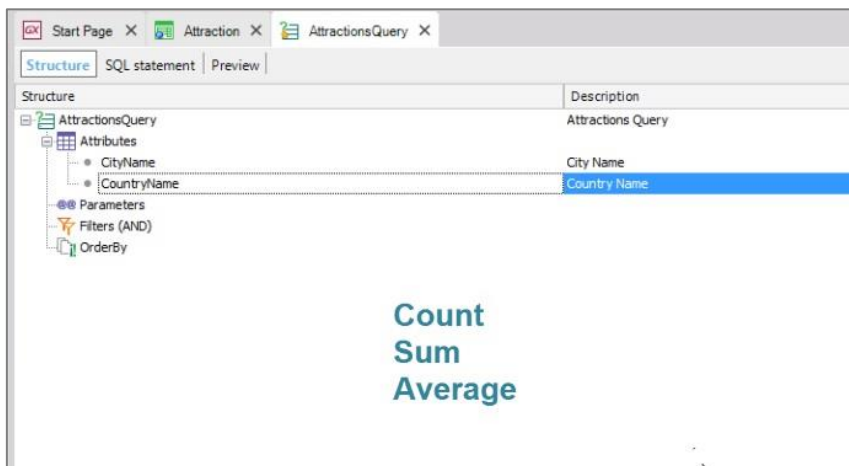
Grabamos.

Aquí en la columna "Description"



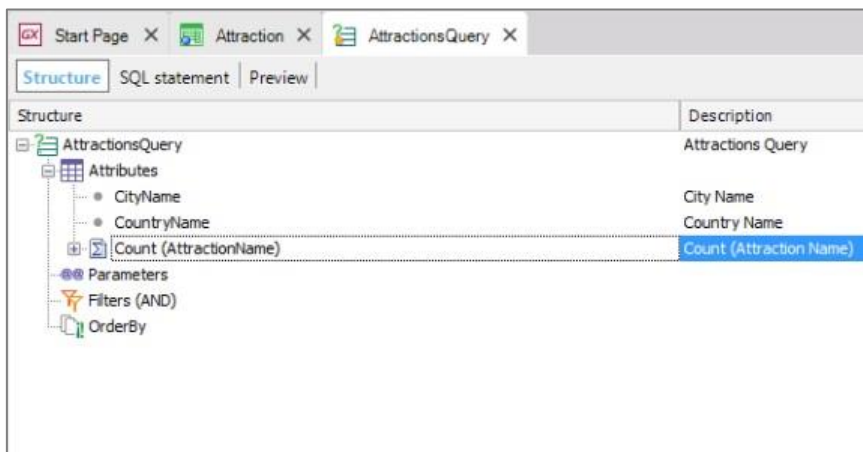
veremos la etiqueta con que se visualizarán los atributos en el resultado de la consulta.

Pero además de atributos aquí también es posible definir funciones (incluso funciones anidadas), como por ejemplo Count, Sum y Average



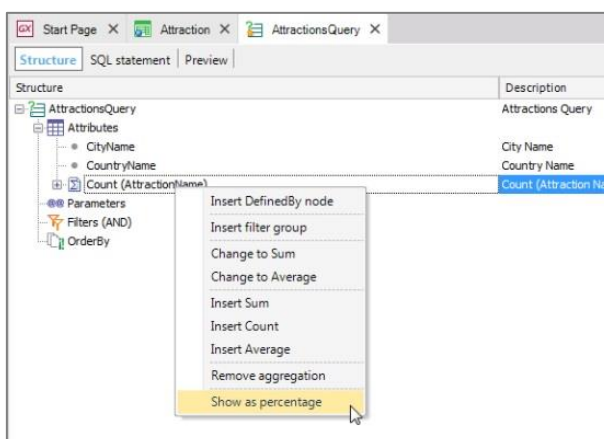
En nuestro ejemplo, nos interesa ver la cantidad de atracciones turísticas de cada ciudad, así que definimos

Count(AttractionName)

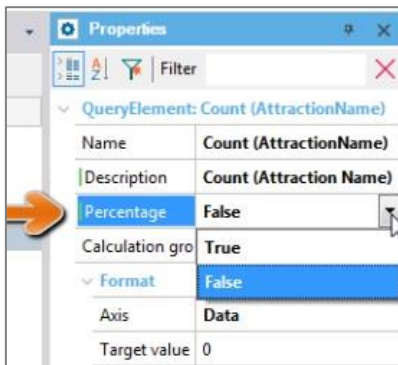


Grabamos.

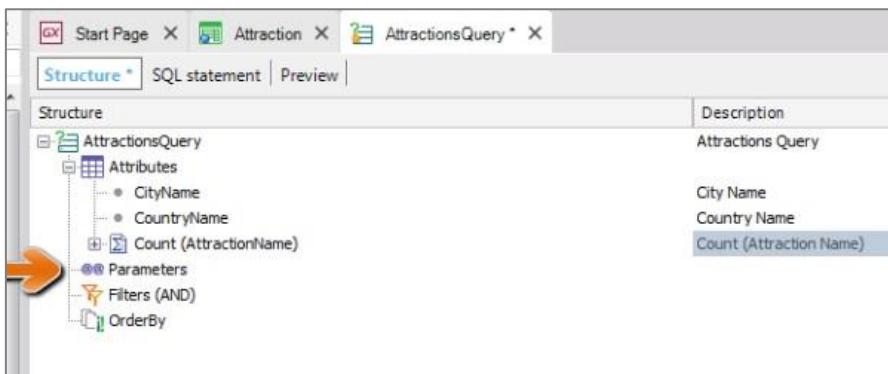
En el caso de haber definido una función sobre un atributo, se podrá indicar que el resultado se visualice como porcentaje. Esto se realiza a través de la opción **Show as percentage** presionando el botón derecho del mouse;



o presionando F4 y editando las correspondientes propiedades:



Pasemos a los Parámetros:



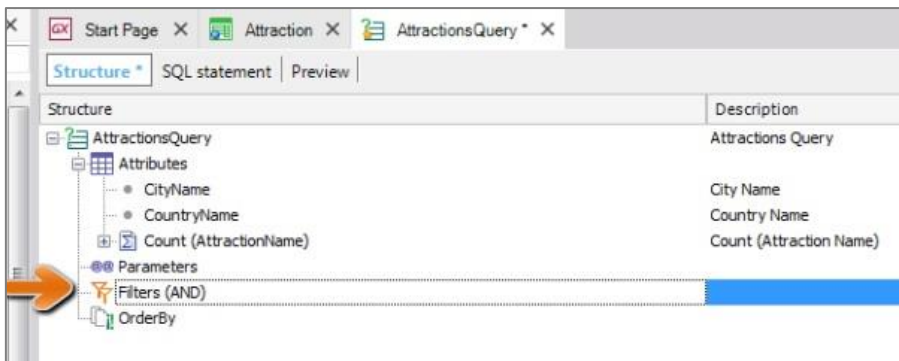
Aquí es posible declarar parámetros que se reciben para ser utilizados en la consulta, si es que corresponde, ya que no toda consulta recibe parámetros.

Se definirá el nombre, la descripción, el tipo de dato, si está basado en un dominio o atributo, si es colección y el valor predeterminado, si corresponde.

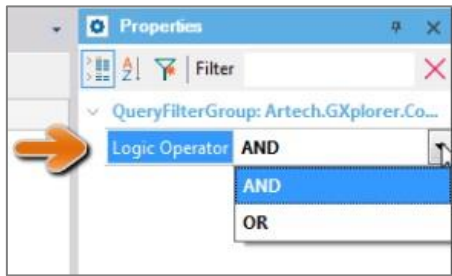
En nuestro ejemplo no necesitamos declarar parámetros.

Veamos ahora los Filtros:

Aquí se especificarán los filtros a aplicarse en la consulta.

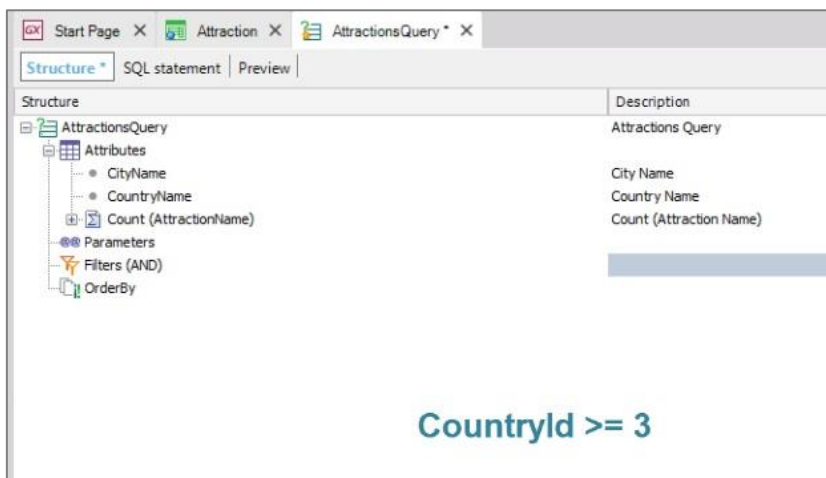


Podemos definir un grupo de filtros, que por defecto serán unidos por AND, pero también podemos usar el conector OR



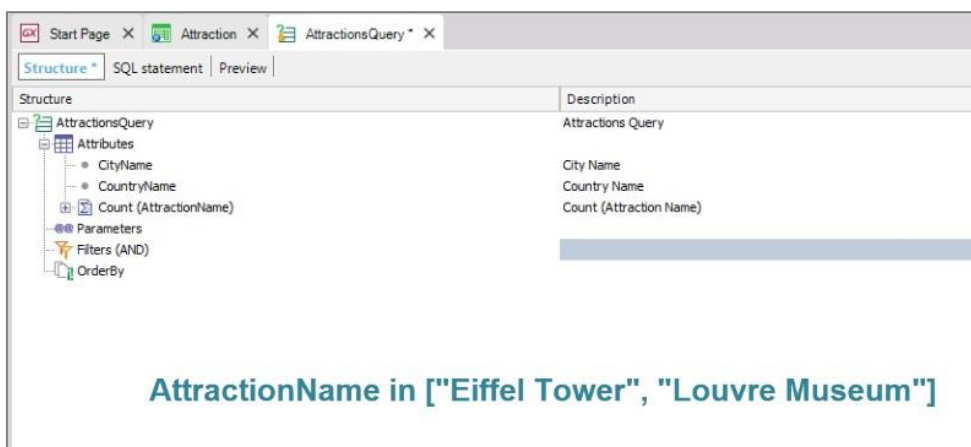
Algunas de las sintaxis posibles para definir filtros son:

- atributo igual a un valor

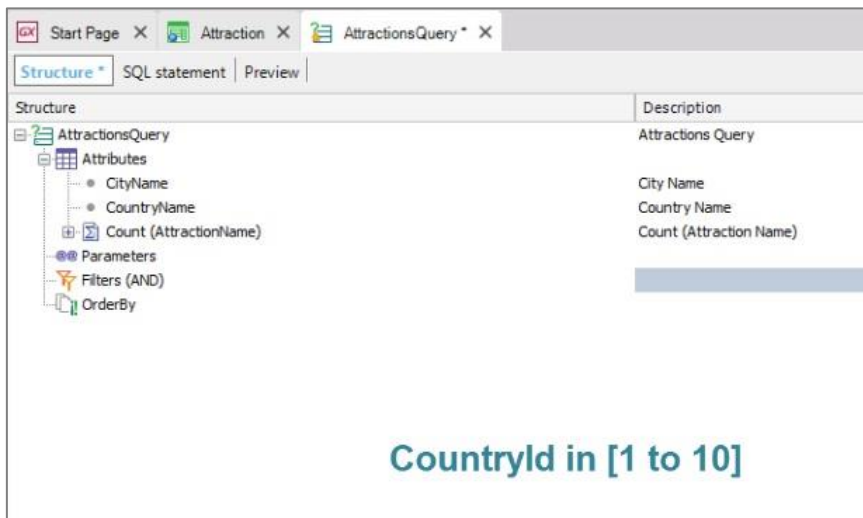


(u otra comparación por mayor, mayor o igual, etc.)

- atributo igual a una lista de valores declarados explícitamente, lo cual se especifica así:
-



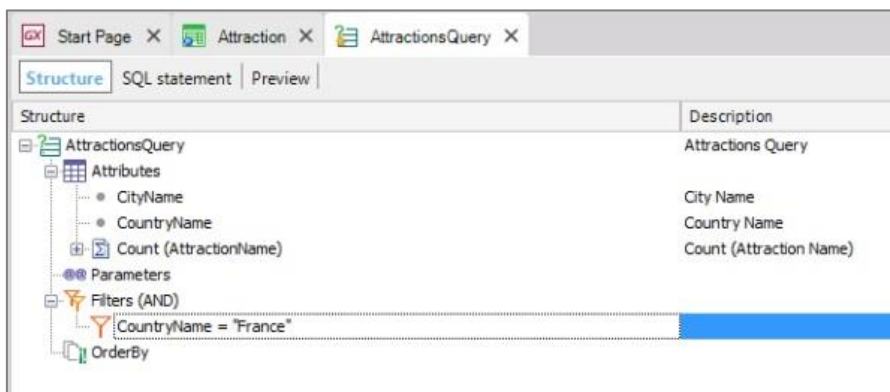
- atributo que esté comprendido dentro de un rango de valores, así:
-



y de esta forma podríamos definir otros grupos con distintas combinaciones de filtros.

Sigamos ahora con nuestro ejemplo. Solo queremos ver la consulta para las ciudades de Francia, así que declaramos el siguiente filtro:

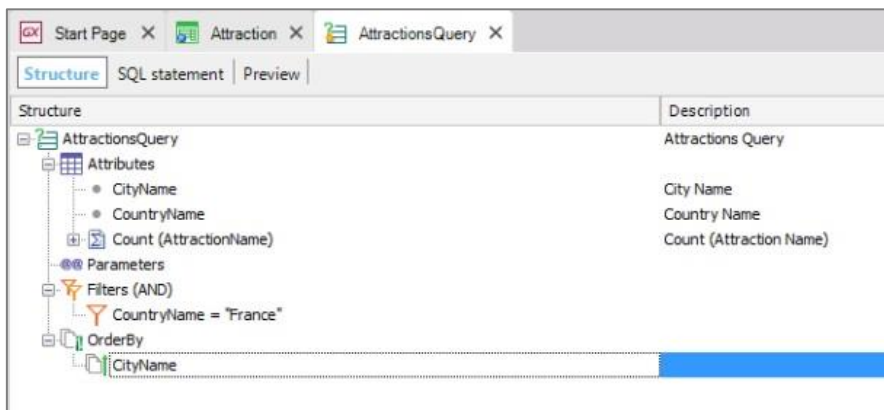
`CountryName = "France"`



Grabamos.

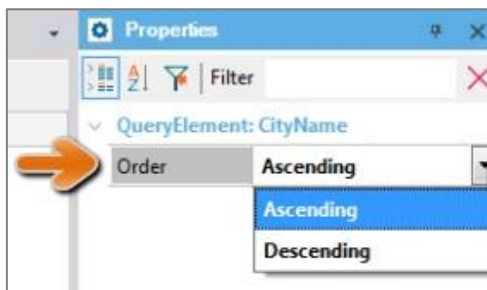
Y finalmente en la **sección Order by** podemos declarar el orden deseado para visualizar la información resultante de la consulta.

En nuestro caso vamos a especificar que la consulta se vea ordenada por CityName, así que declaramos ese orden:



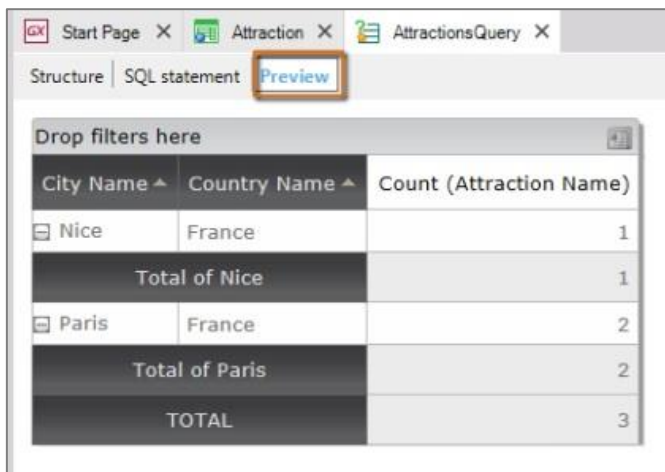
Grabamos.

Si quisiéramos un orden descendente podemos declararlo en las propiedades:



Bien. Ya hemos terminado de definir nuestra consulta. ¿Cómo se verá el resultado?

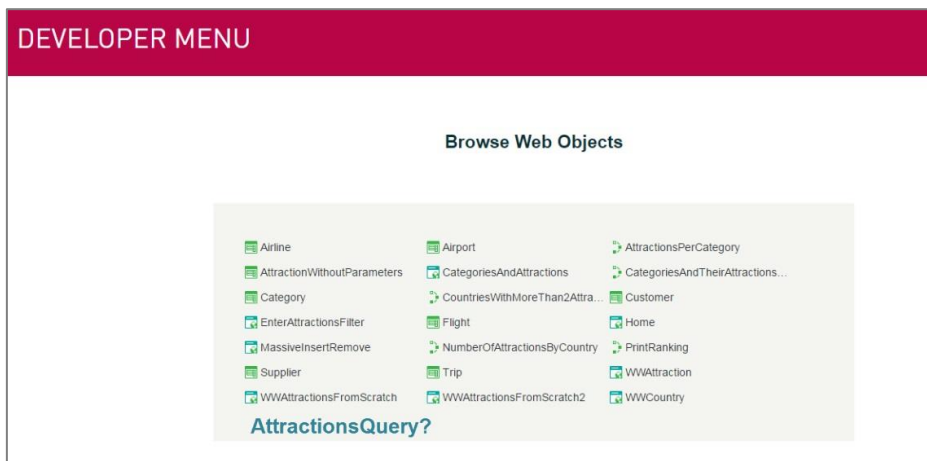
Vayamos a la **solapa Preview**.



Aquí podemos precisamente ver una vista previa del resultado de la consulta,

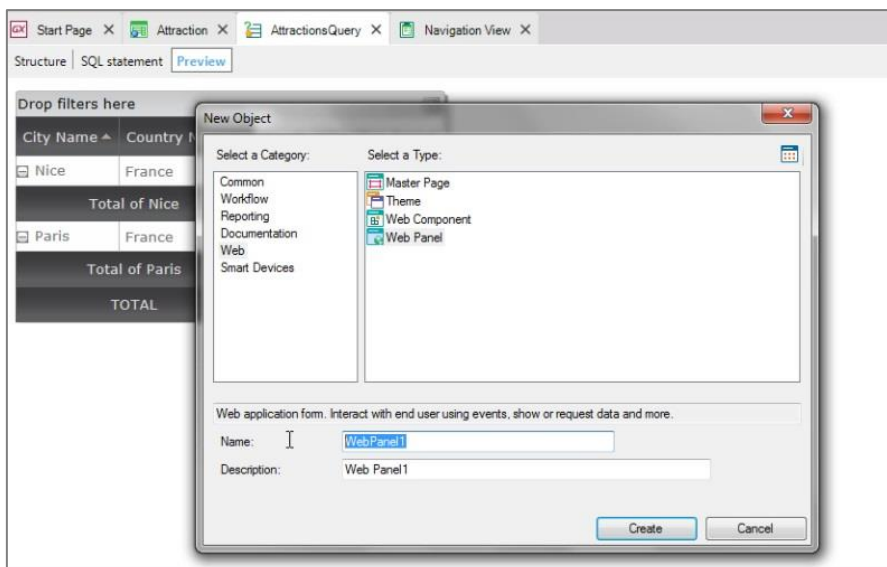
Es importante mencionar que si bien aquí podemos ver el resultado de la consulta en el propio objeto Query definido, esto no sucede cuando ejecutamos la aplicación.

Para poder ver en ejecución

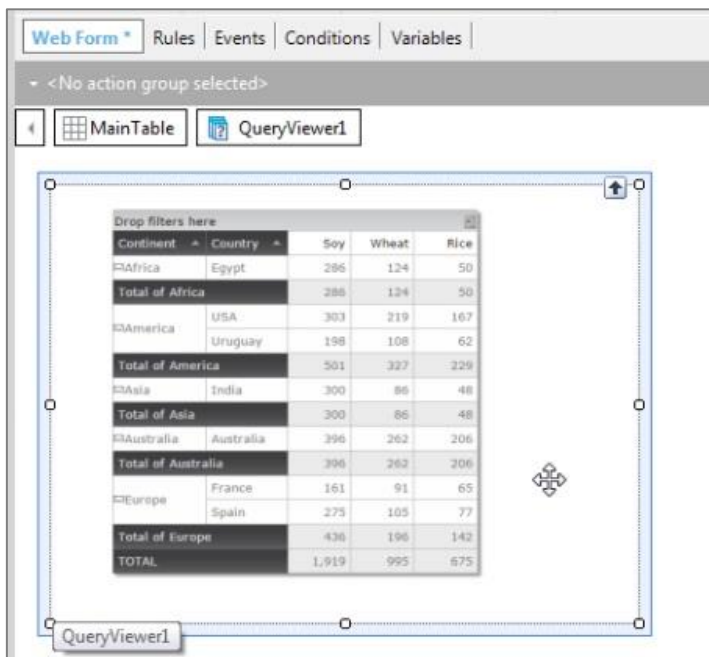


la consulta que definimos utilizando el objeto Query, debemos incluir el user control Query Viewer en un web form.

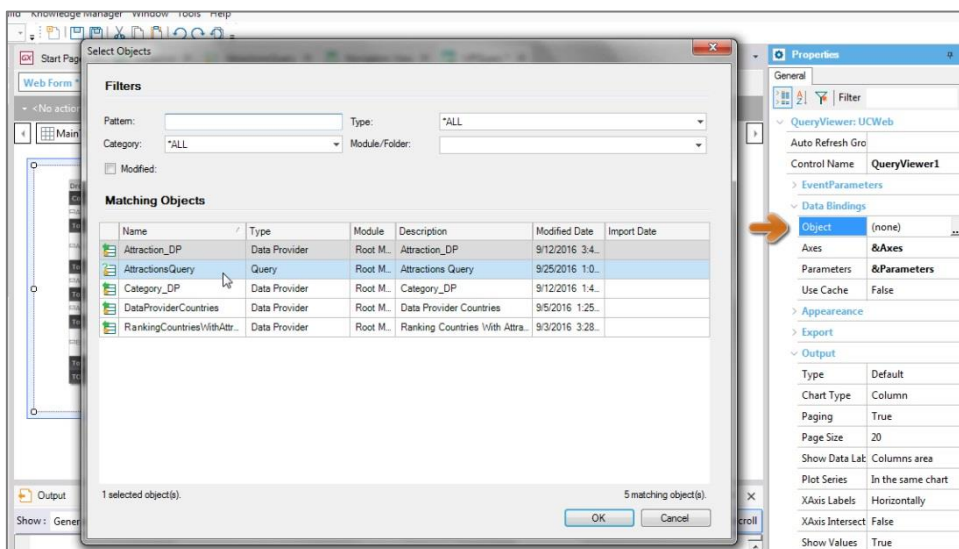
Vamos a crear entonces un web panel de nombre WPQuery



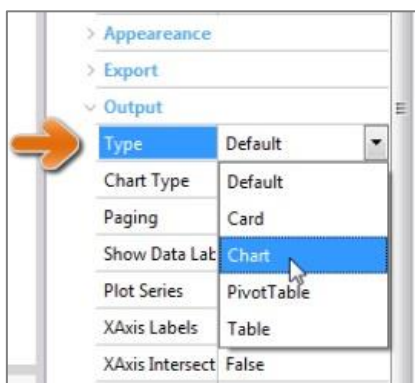
y vamos a arrastrar el user control QueryViewer sobre el form



En la propiedad Object, especificamos el nombre del objeto Query que será ejecutado dentro de este visor:



Editando la propiedad Type, bajo el grupo Output podemos especificar una de las tres formas de ver la salida. Elegimos Chart:

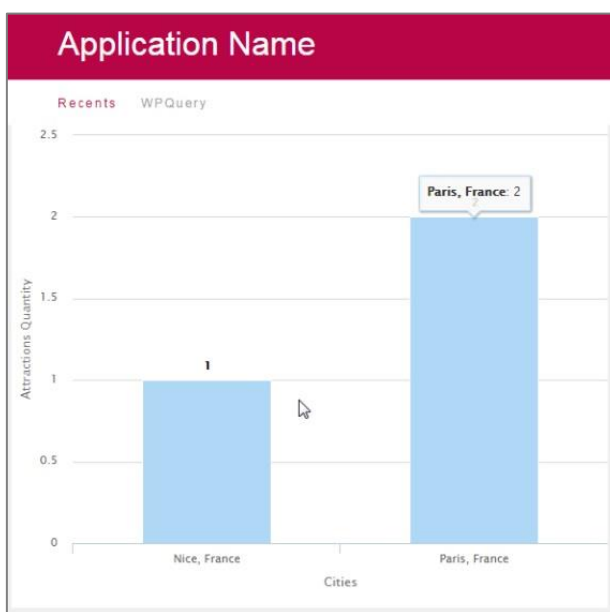


Podemos elegir un tipo de gráfico en particular dejamos Column, y ponemos “Cities” como título del eje X y “Attractions Quantity” como título del eje Y

Type	Chart
Chart Type	Column
Plot Series	In the same chart
XAxis Labels	Horizontally
XAxis Intersect	False
Show Values	True
XAxis Title	Cities
YAxis Title	Attractions Quantity

Grabamos.

Ahora si, estamos en condición de ver nuestra consulta completa en ejecución, así que presionamos F5:



Commit | Update | History | Activity | Versions

Comment:
 New query object AttractionsQuery and new web panel WPQuery in order to execute the dynamic query. Recent Comments...

Pending Commits (130/130) | Ignored Objects

☒	Type	Description	Modified	Module	Action	Last Synchronize	User
☒	Query	Attractions Query	9/25/...	Root Module	Inserted	7/7/2016 4:32 PM	ARTECH/acaggiano
☒	Web Panel	WPQuery	9/25/...	Root Module	Inserted	7/7/2016 4:32 PM	ARTECH/acaggiano

Cancel Commit

