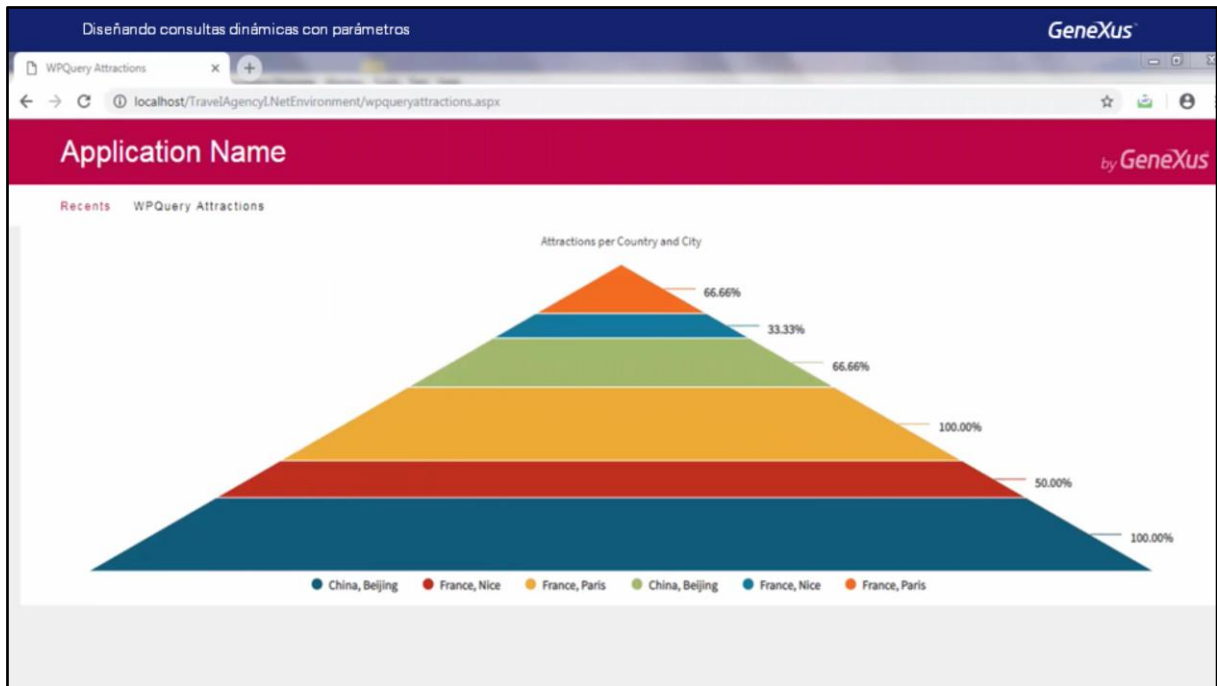


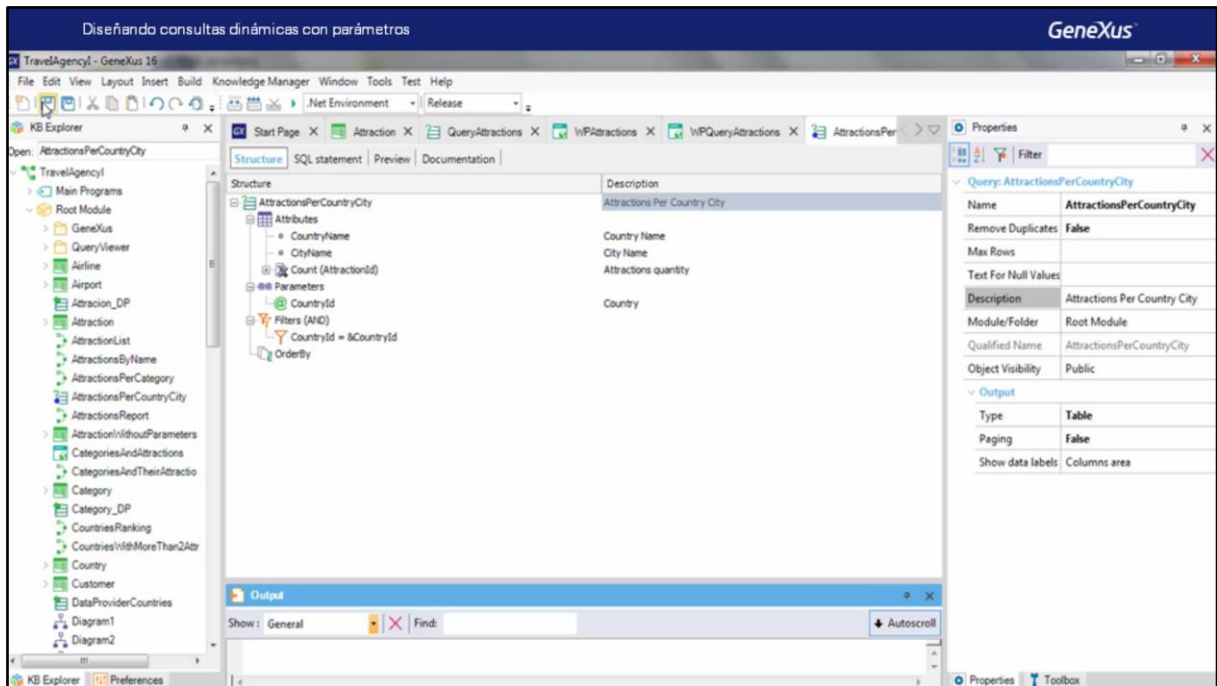
## Diseñando consultas dinámicas con parámetros

*GeneXus* 16



Anteriormente ya habíamos definido en nuestra Kb un objeto query para visualizar la cantidad de atracciones turísticas de Francia y China.

Veamos ahora algunos ejemplos más.



Supongamos que ahora la Agencia de viajes nos solicita realizar algunas modificaciones al objeto Query ya definido, ya que desean poder consultar en la gráfica la cantidad de atracciones turísticas de las ciudades de un determinado país que no necesariamente tienen que ser Francia y China.

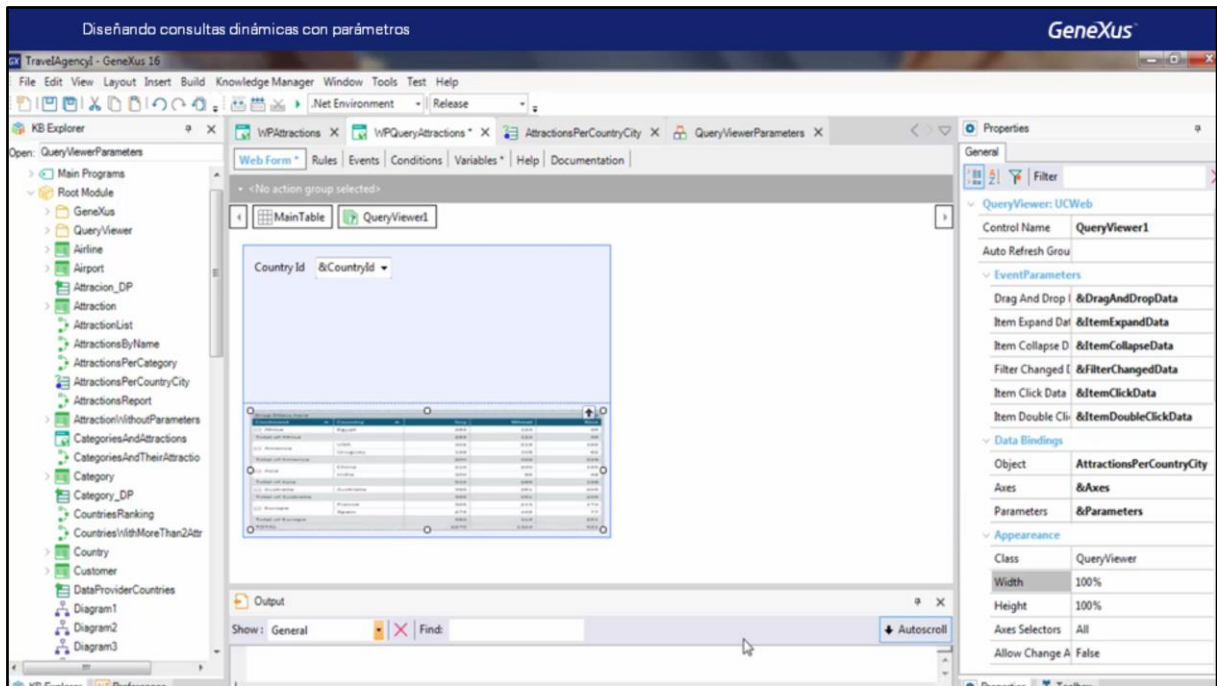
Así que necesitamos hacer algunas modificaciones a la definición de este objeto Query para que reciba el país como parámetro.

Vamos a comenzar borrando el filtro y el orden definidos

En el nodo de los parámetros damos enter, y definimos CountryId.

Ponemos Country como su descripción y definimos también el filtro correspondiente para que se considere el valor de este país en la consulta

Guardamos estos cambios, y vamos ahora a editar el web panel para que el usuario pueda seleccionar el país de una lista y dicho valor se considere como parámetro para la consulta.

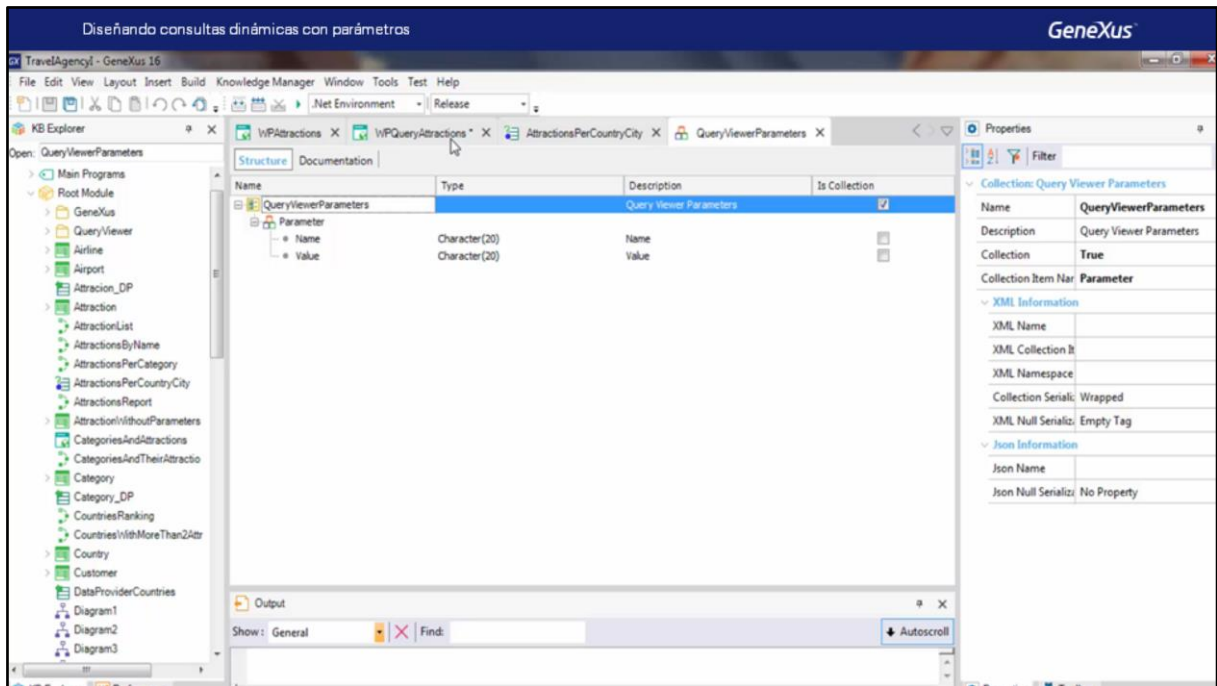


Bien. Lo primero que haremos en el web panel es definir la variable `&CountryId` y declararla como un combo dinámico para que guarde como valor `CountryId` y muestre como descripción `CountryName`.

Para eso vamos a editar la propiedad `Control type` y seleccionamos el valor `Dynamic combo box`. Tenemos indicado que el `Item value`, o sea que el valor que se va a guardar va a ser `CountryId`, y el valor a mostrar será `CountryName`.

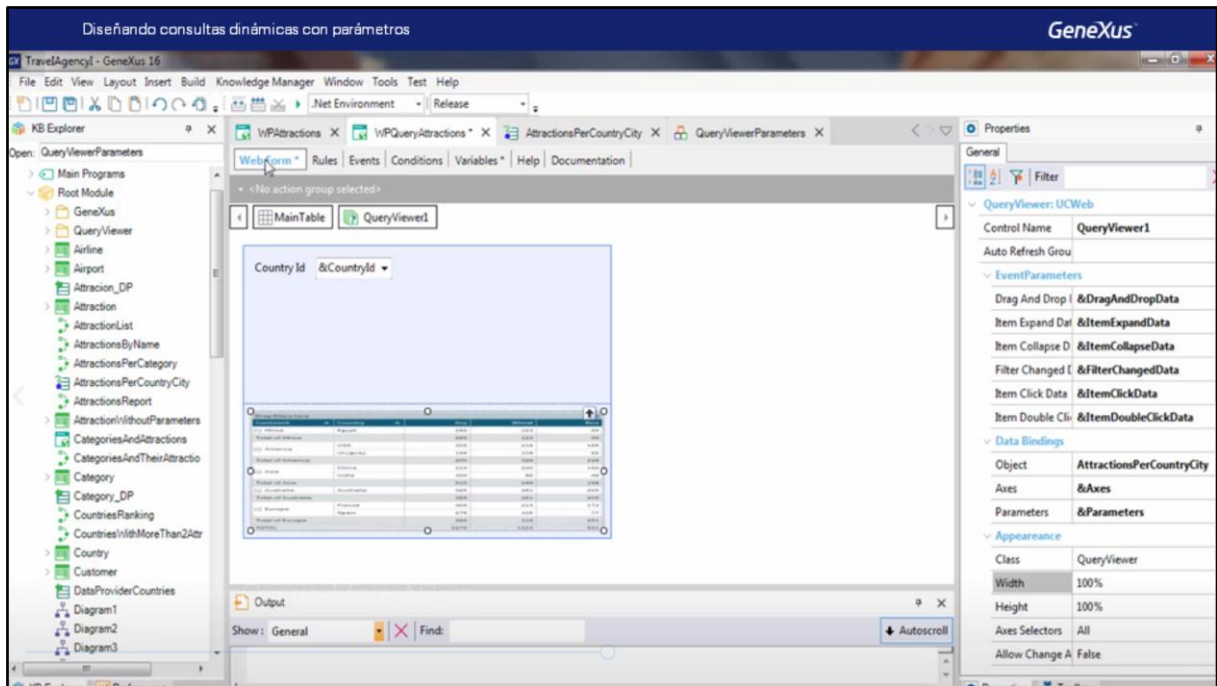
Bien. Nos está faltando que el control `QueryViewer` considere la variable `&CountryId` como parámetro.

Observemos que la propiedad `Parameters` del control `QueryViewer` tiene automáticamente asociada una variable `&Parameters`. Esa variable está basada en un tipo de dato estructurado creado automáticamente al momento de arrastrar el control `QueryViewer` sobre el form. La variable también fue definida en forma automática.



Si observamos su estructura vemos que se trata de una colección de parámetros, cada uno con un nombre y un valor. Algo importante a considerar es que ambos ítems son de tipo Character. Esto es importante de tener presente porque al momento de cargar esta colección con los parámetros a considerar, deben ser del mismo tipo de dato.

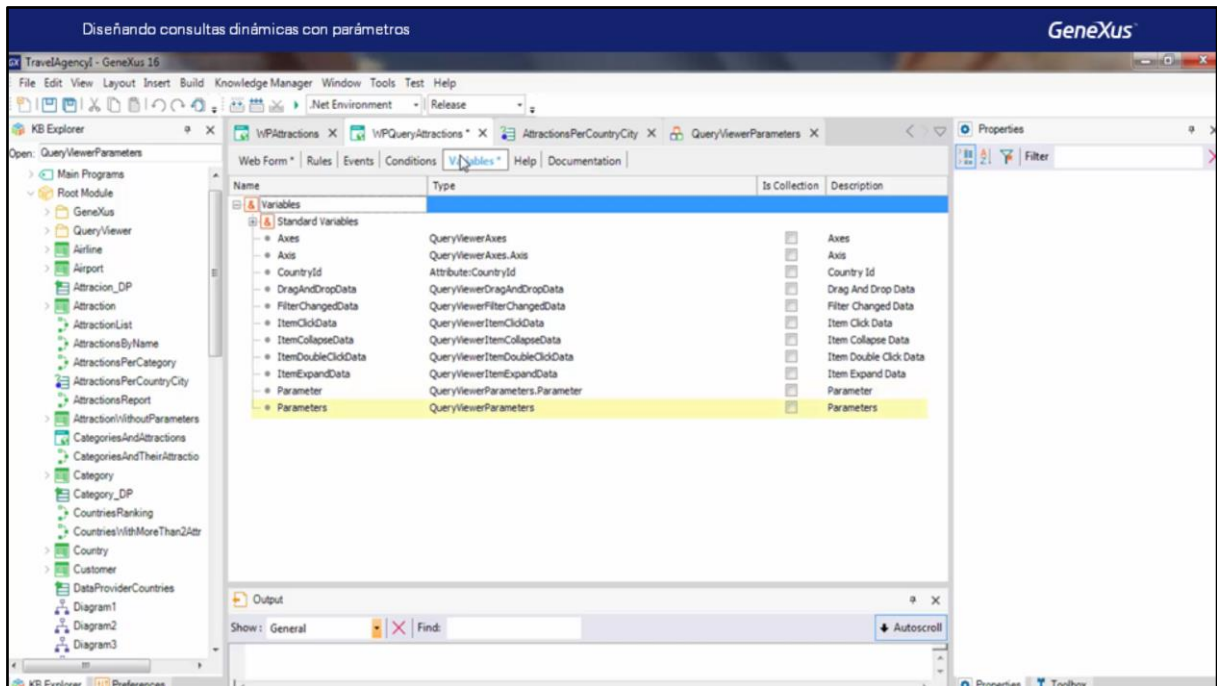
Lo que debemos hacer entonces es cargar la variable de nuestro filtro a la colección de parámetros asociados al control QueryViewer.



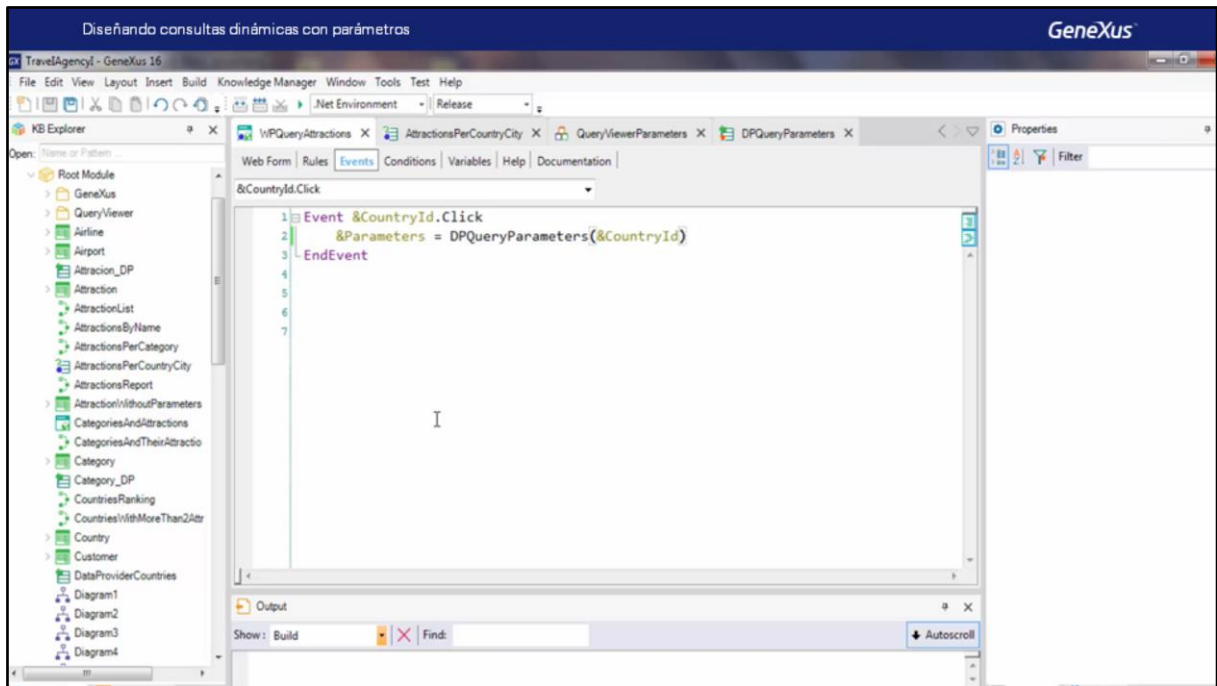
La pregunta que surge ahora es la siguiente:  
¿Dónde vamos a codificar esa carga? ¿En qué evento?  
Analicemos lo siguiente:

Lo que queremos lograr es que al seleccionar un país de la lista se cargue el parámetro y entonces la consulta se actualice mostrando la cantidad de atracciones de las ciudades del país seleccionado.

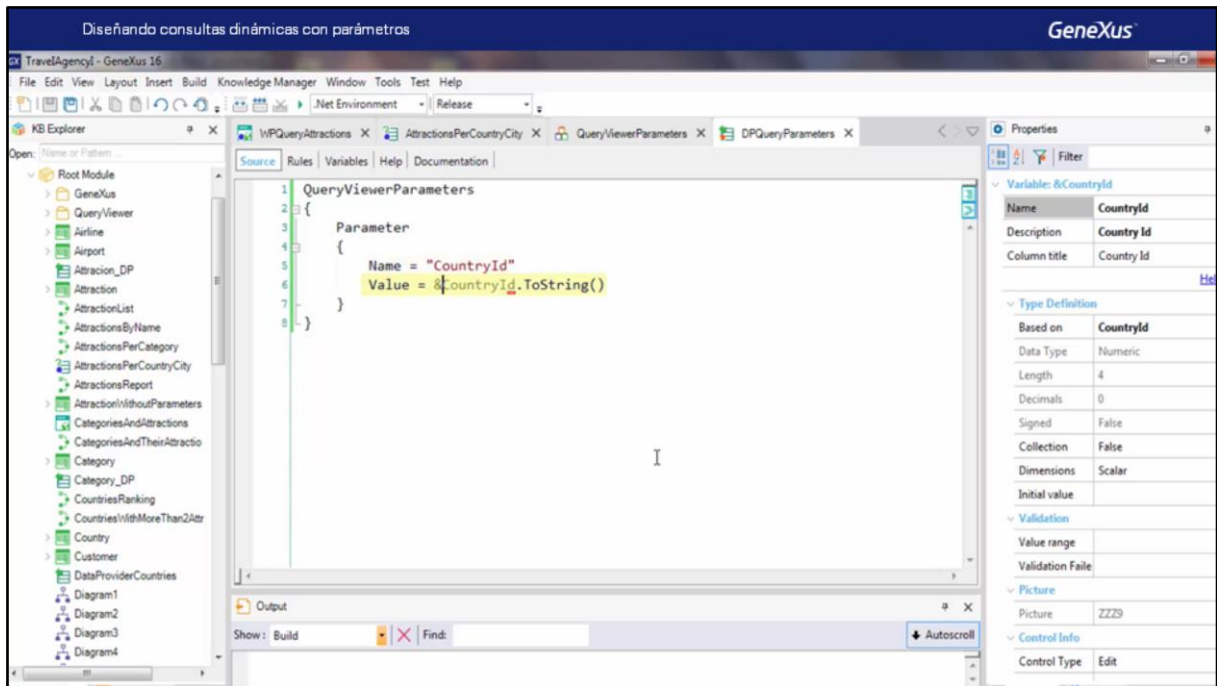
Para lograrlo vamos a codificar el evento Click asociado a la variable &CountryId. Este evento se disparará cada vez que el usuario seleccione un elemento del combo dinámico.



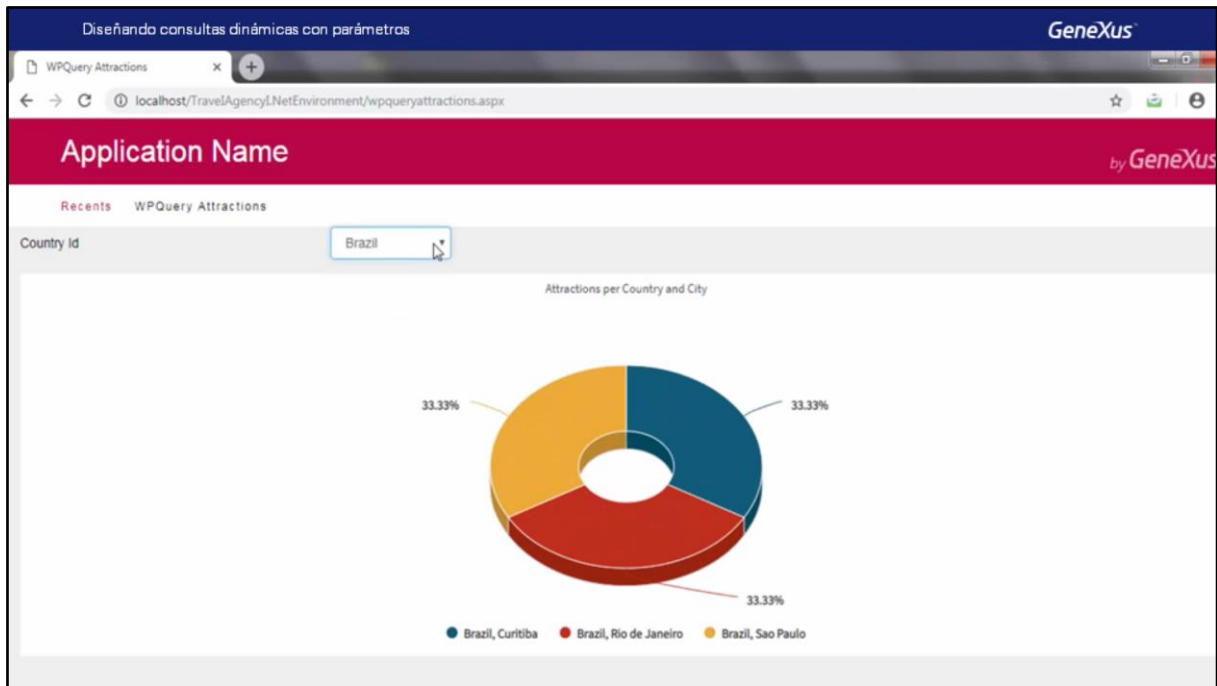
Si vamos al sector de las variables, vemos que ya se han creado las variables &Parameter basada en el tipo de dato que representa a un parámetro, y la variable &Parameters basada en el tipo de dato que representa a la colección de parámetros.



Así que en el sector de los eventos, en el evento Click asociado a la variable &CountryId recibimos la carga de la variable &Parameters a través de la salida de un Data Provider.

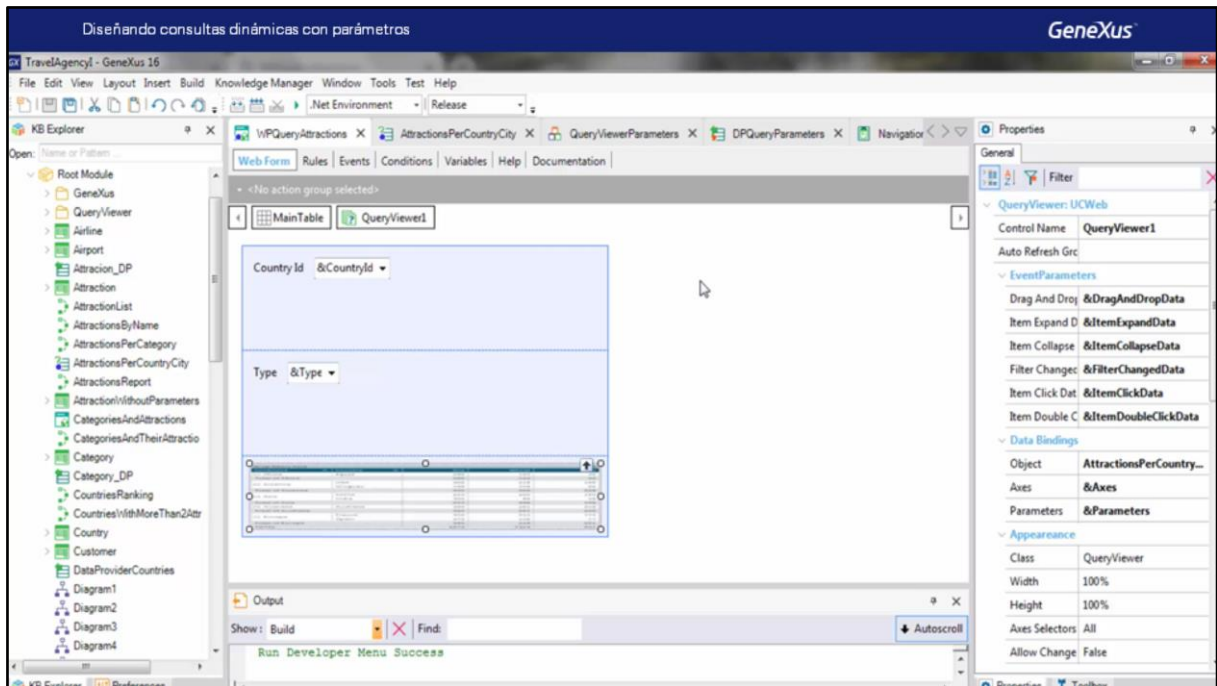


Este Data Provider recibe el valor del país seleccionado y carga ese parámetro a la colección, indicándole el nombre y el valor considerando que al ser de tipo de datos Character debemos convertirlo y por eso utilizamos el método ToString.



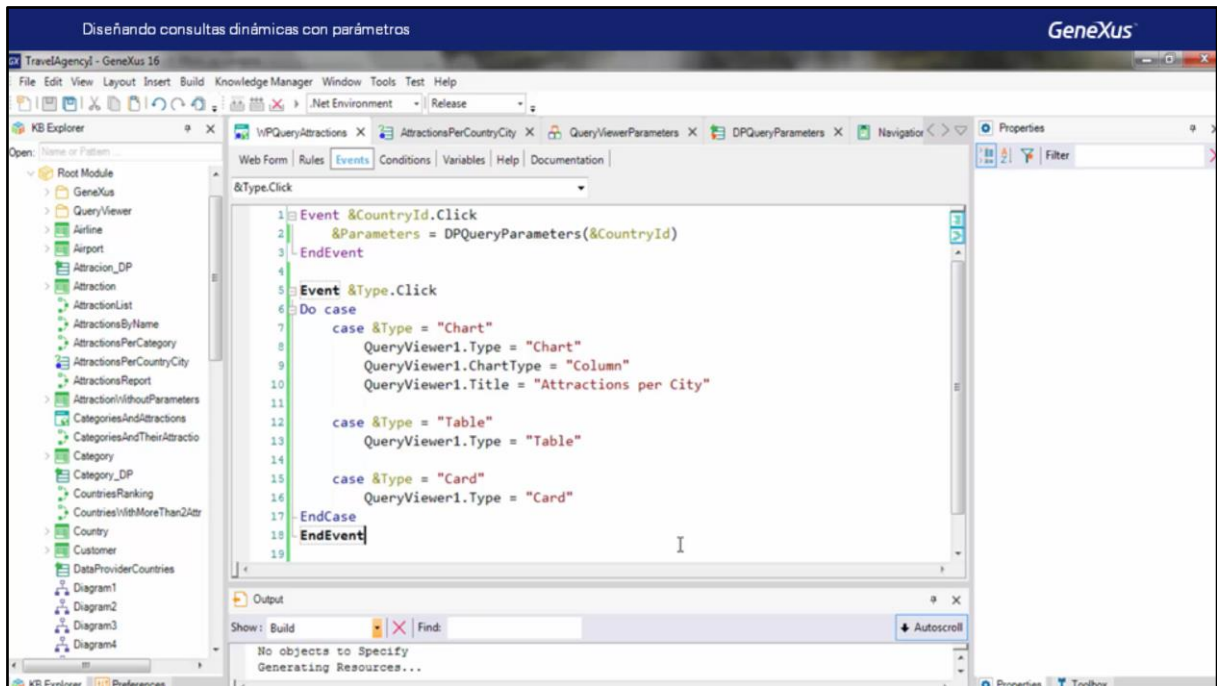
Volvamos al web panel, y para visualizar otro tipo de gráfico en ejecución, vamos a cambiar y vamos a elegir Doughnut 3D.  
Ahora sí, presionamos F5.

Seleccionamos el web panel, y verificamos que si cambiamos de país, se actualiza la consulta.



Veamos ahora rápidamente que también podemos seleccionar en ejecución el tipo de salida de la consulta. En nuestro ejemplo vamos a considerar que el usuario desea poder elegir entre ver la salida como un gráfico, una tabla o una tarjeta. Definimos entonces una nueva variable `&Type`, de tipo Character, que se mostrará como un combo con las opciones Chart, Table y Card.

En la propiedad Control type elegimos Combo box y asignamos los valores.



Vamos a los eventos, y en el evento click asociado a esta variable vamos a modificar el valor de la propiedad Type del control QueryViewer para que tome en cada caso el valor correspondiente:

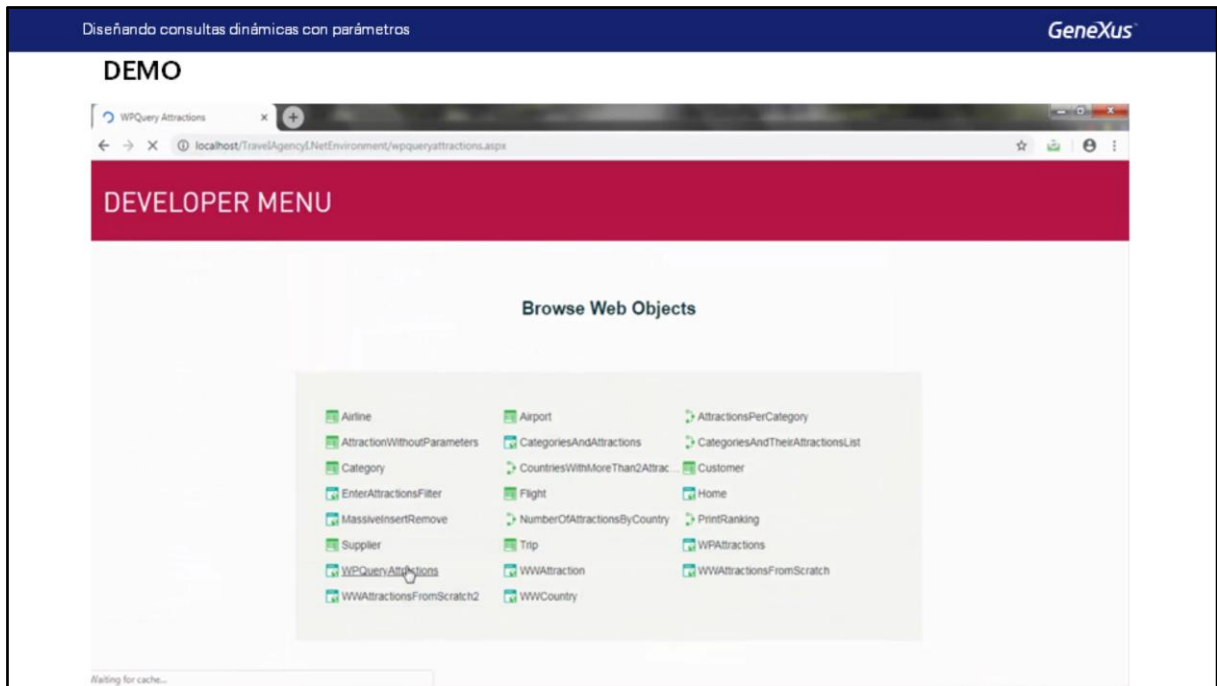
Utilizamos para eso la estructura de control Do case, que permite evaluar qué hacer según el valor, en este caso, que toma la variable &Type:

Para eso indicamos el nombre del control, QueryViewer1, seguido de punto, la propiedad Type e indicamos el valor.

Si el valor seleccionado es "Chart" vamos a construir un gráfico de columnas, asignándole el título de "Atracciones por ciudad".

Si el tipo seleccionado es "Table", entonces lo vamos a visualizar como una tabla, y si no, como una tarjeta.

Presionemos F5.



[ DEMO: [https://youtu.be/sf-H-k\\_JF2M](https://youtu.be/sf-H-k_JF2M)]

Observemos que el formato Table nos permite, por cada columna, ordenar en forma ascendente, descendente, mostrar u ocultar valores.

E incluso, desde aquí, es posible exportar el resultado en distintos formatos.



Videos

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

Documentation

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

Certifications

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