

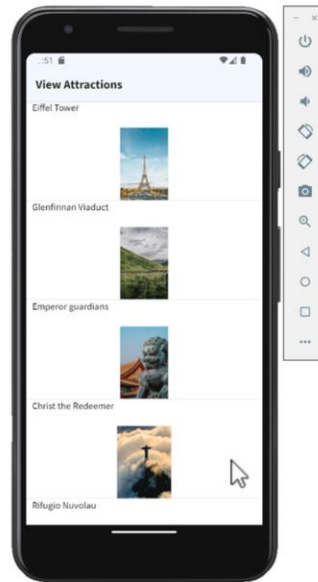
Grids. Types and Functionalities



Vanesa Fernández

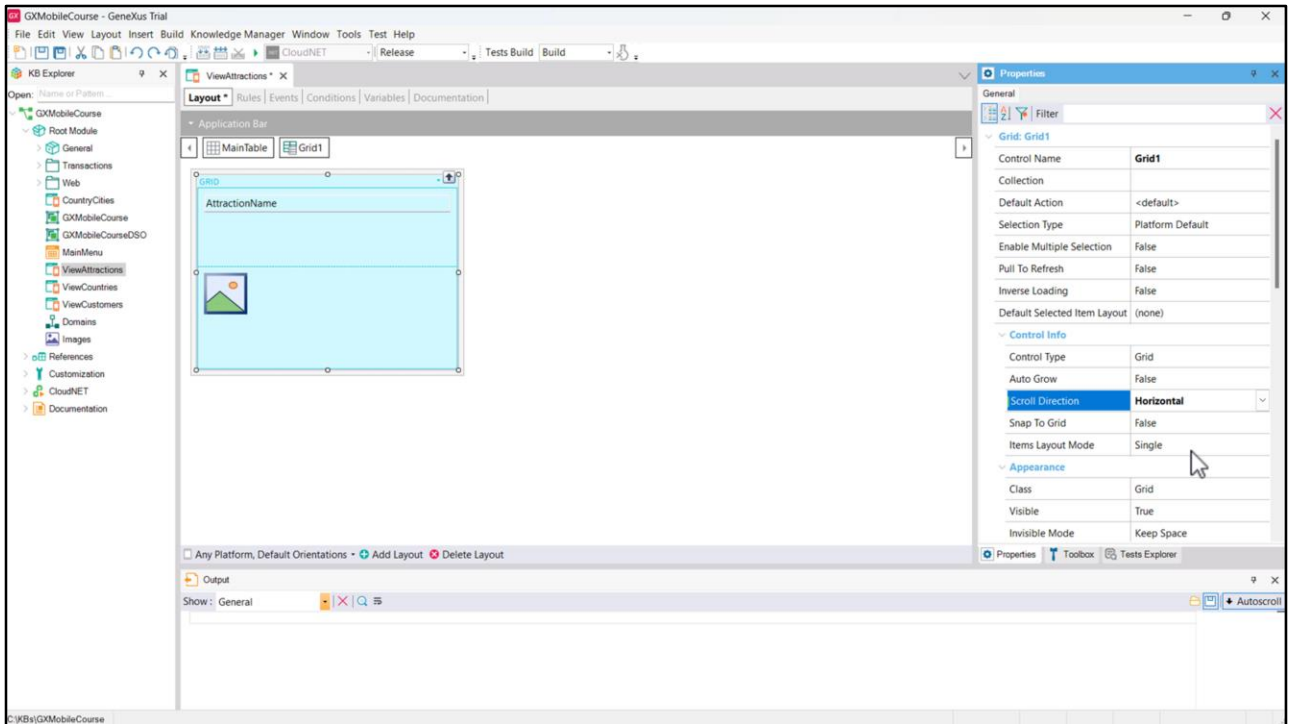
In this video, we will see how to configure a Grid to adopt a different behavior than the default one, as well as other types of Grids available to develop our application.

Standard Grid (Smart Grid)



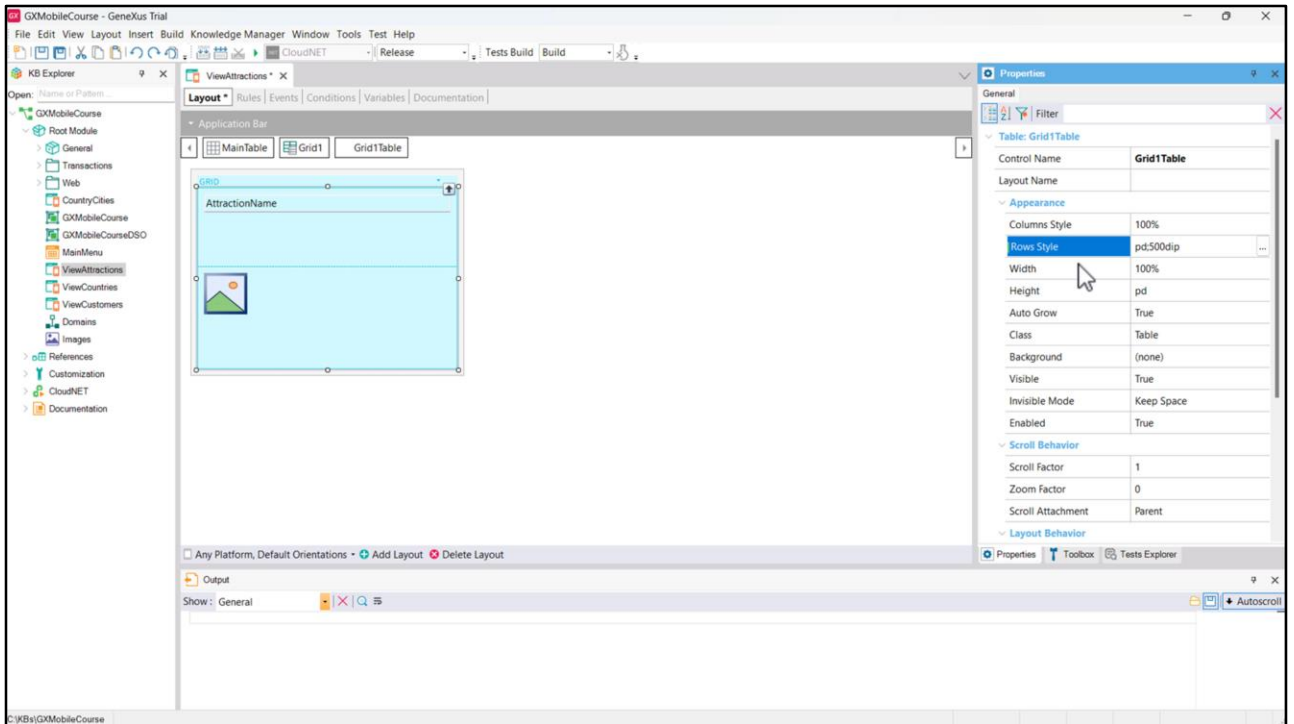
Let's start with the standard grid, which can be configured to scroll horizontally to view record data by scrolling in that direction instead of vertically.

In upgrade 11 of GeneXus 16, the Grid control became a Smart Grid, so there are properties that allow us to change how the information will be displayed, in a more flexible way.

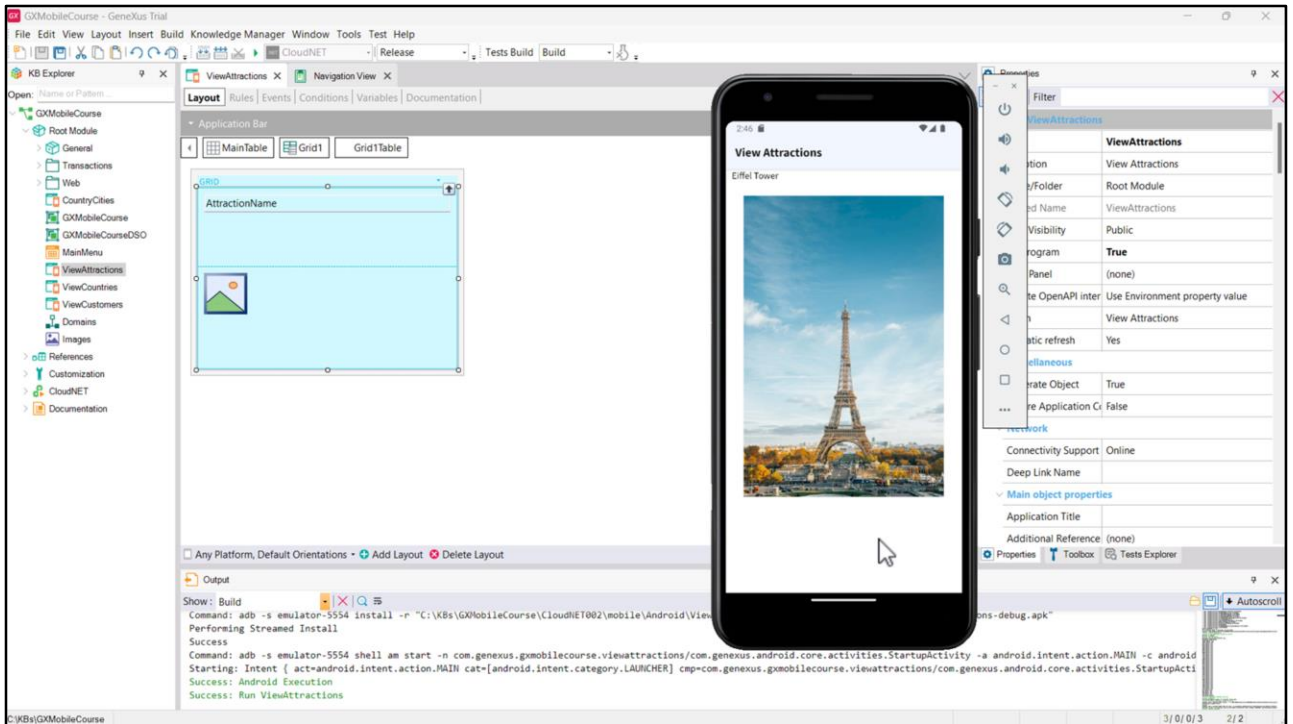


Before we start working on this object, let's set it as a Main program to be able to run it quickly.

If all we want is to scroll the records horizontally, we can set the Scroll Direction property of the Grid to Horizontal.

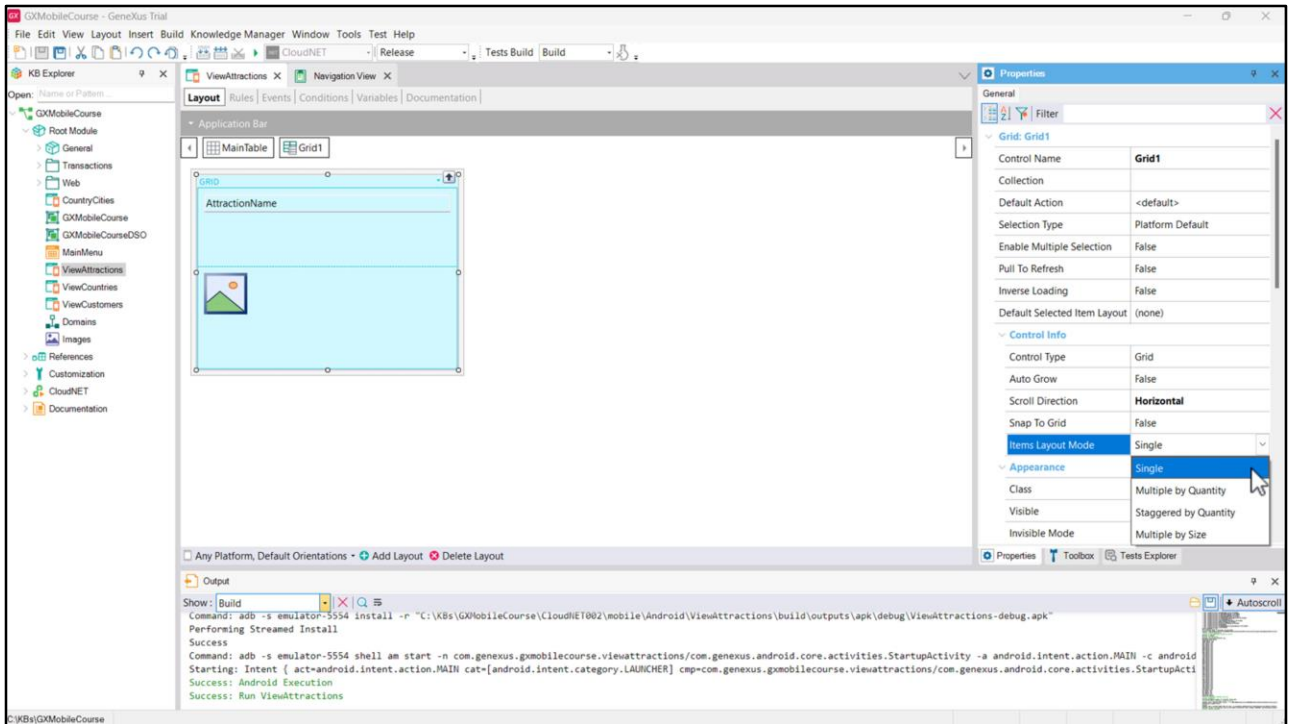


Let's also change the value of the Rows Style property of the table containing the elements inside the Grid so that the second row, which is that of the photo, takes up 500 dips.



We run it...

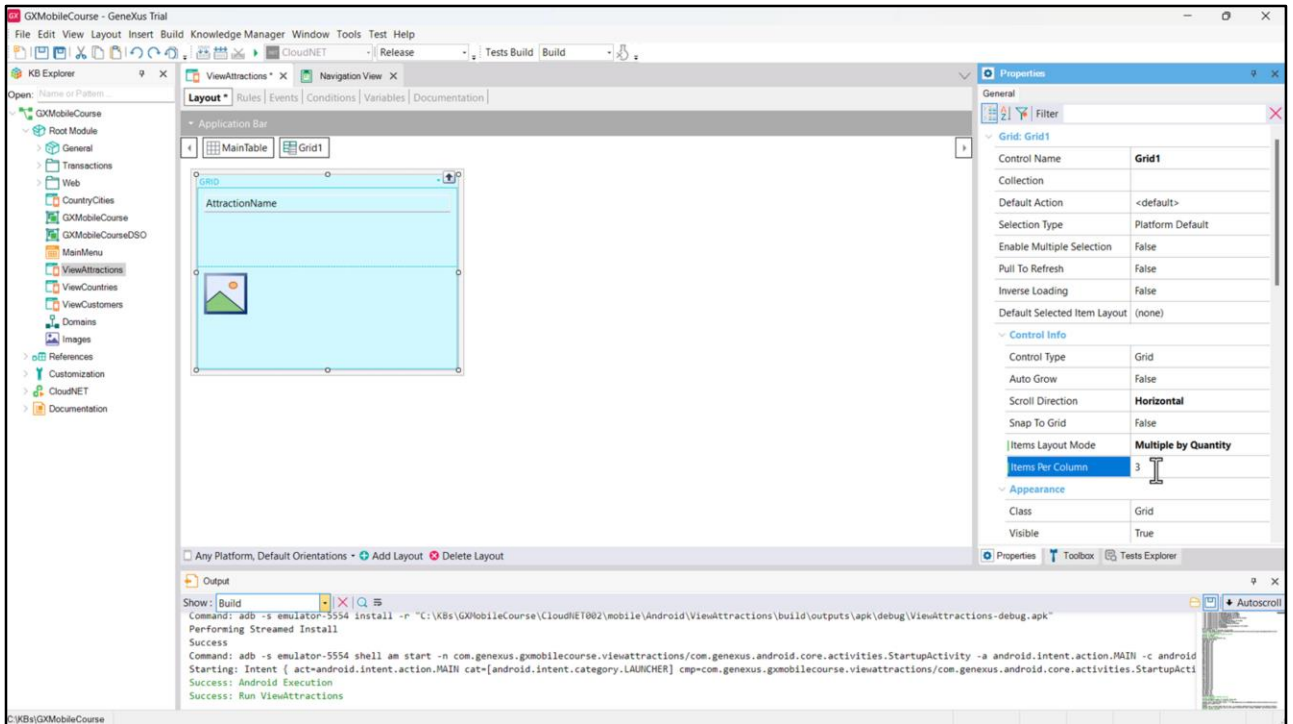
Now the Grid shows a record, and by clicking on it we can move to the right and then back to the left.



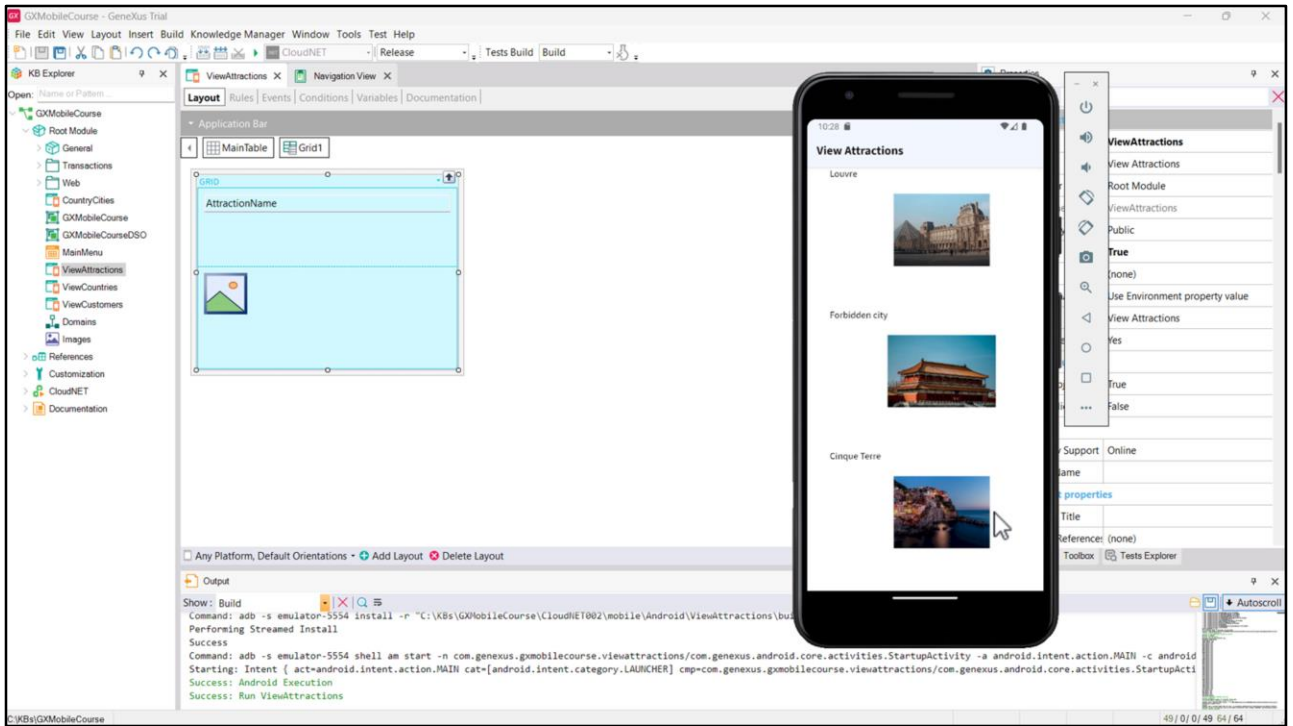
In addition to horizontal scrolling, it would be interesting to view several attractions at the same time to make better use of the space. To do this, the Items Layout Mode property allows changing the way in which the Grid data is displayed.

Several values are available:

- Single – This is the default value and causes the grid to display only one item per row.
- Multiple by Quantity – This value indicates that a fixed number of items will be displayed in each row/column of the grid. It will be determined by the properties Items Per Row if the scroll direction is vertical, or Items Per Column if the scroll direction is horizontal.
- Staggered by Quantity – The grid is displayed in a staggered layout, where items can have different sizes. The number of items per row or column is determined by the Items Per Row or Items Per Column properties, depending on the direction of the Grid scrolling.
- Multiple by Size - The number of items displayed in each row/column will be determined by the size of each item. To achieve this, we use the values of the properties Maximum Width and Minimum Width if the scroll direction is vertical, and Maximum Height and Minimum Height if the scroll direction is horizontal.

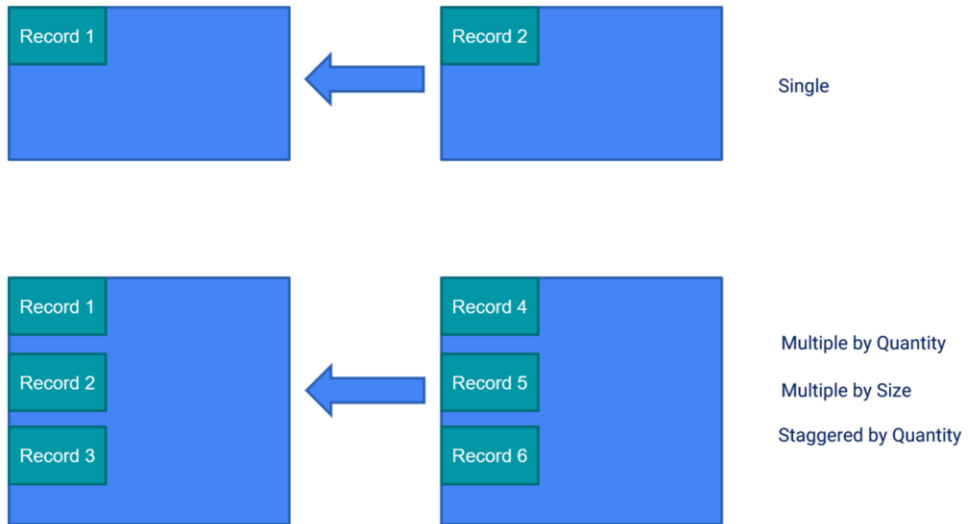


We are going to assign the value Multiple by Quantity, and since we have horizontal scroll direction, the Items Per Column property is displayed. We set it to 3 and run.



Now we see a single column on the page and 3 attractions in it. Paging to the right shows the next 3 attractions, and so on.

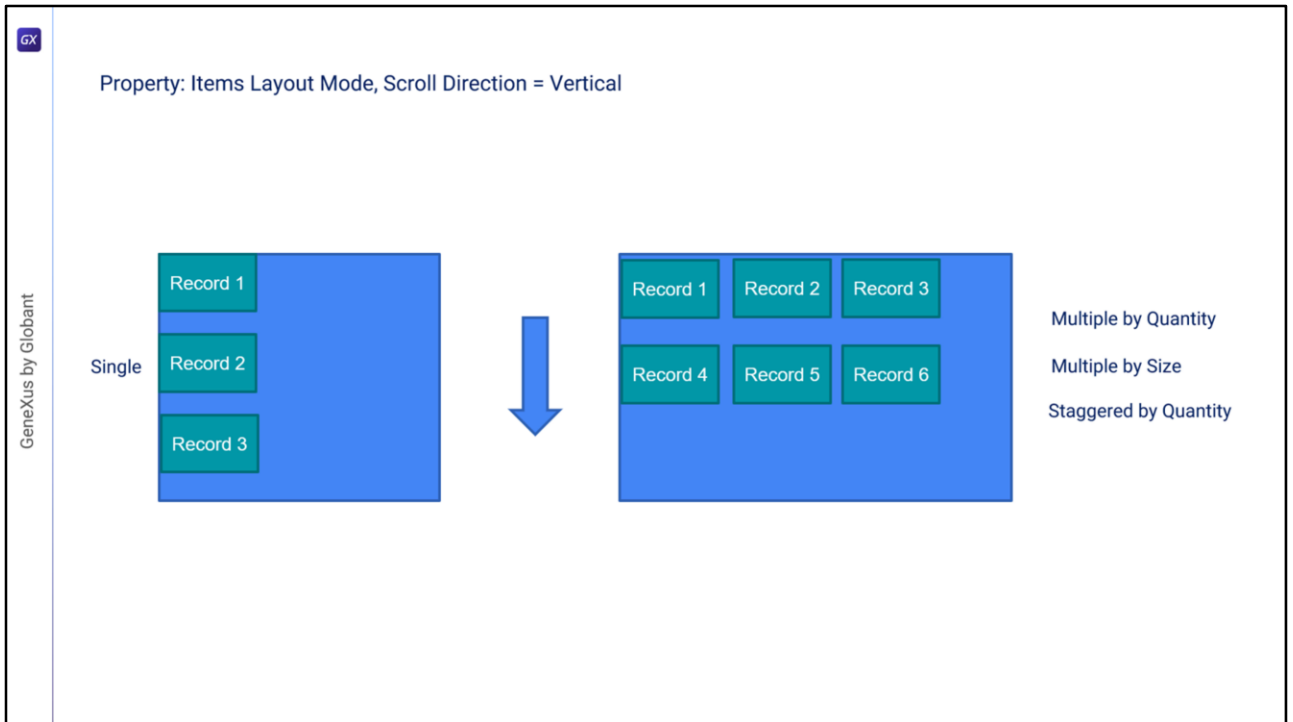
Property: Items Layout Mode, Scroll Direction = Horizontal



In summary, in the standard Grid (Smart Grid), if scrolling is horizontal as in our example, we can define how many records we want to view per column; that is, in the same column downwards.

But we will always view one column at a time.

The number of records per column displayed downwards can be defined as a fixed value, or depending on the height (minimum and maximum) that we want to give to each item. If we do it with a fixed value, we can see it staggered or not, depending on whether we choose Staggered by Quantity or Multiple by Quantity, respectively.

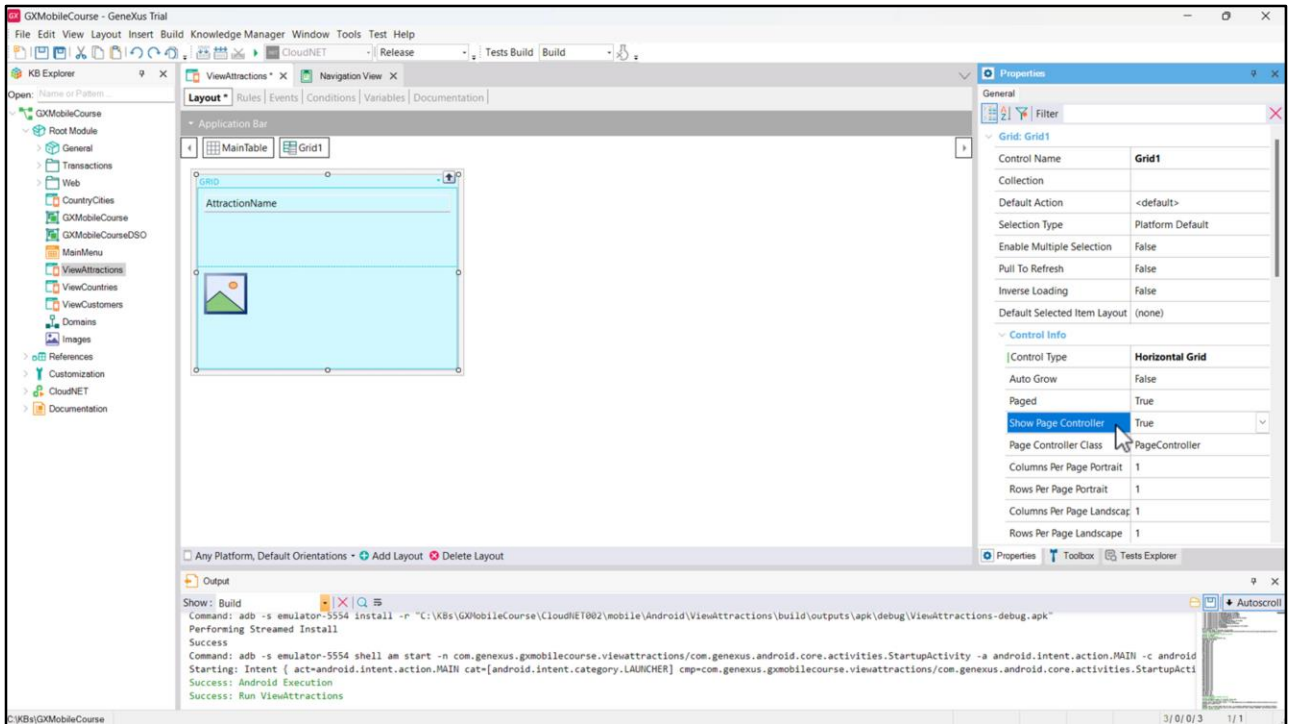


If scrolling is vertical, we can configure how many records we want to see in each row.

That is, the next row will bring back the same group of records. And this can also be defined as a fixed value (staggered or not), or depending on the width (minimum and maximum) that we want to give to each item.

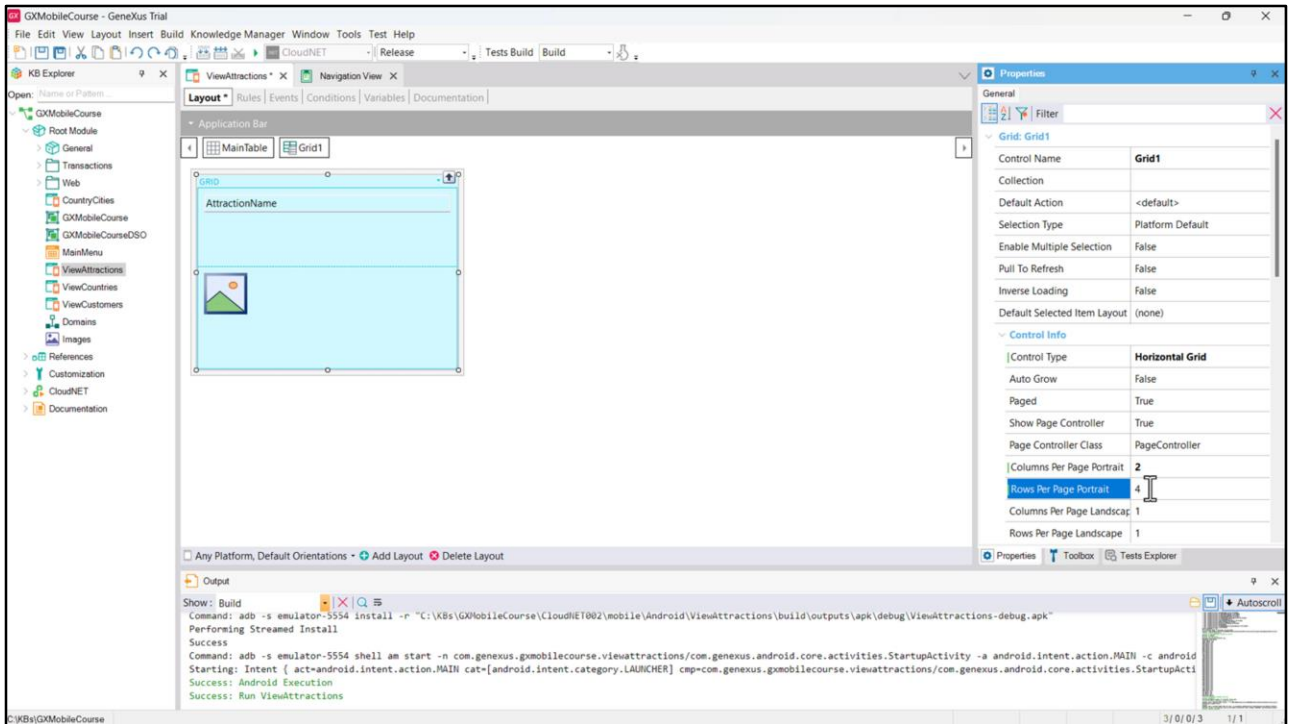


We have already seen several options of the standard Grid to display information. We will now see another control called Horizontal Grid.

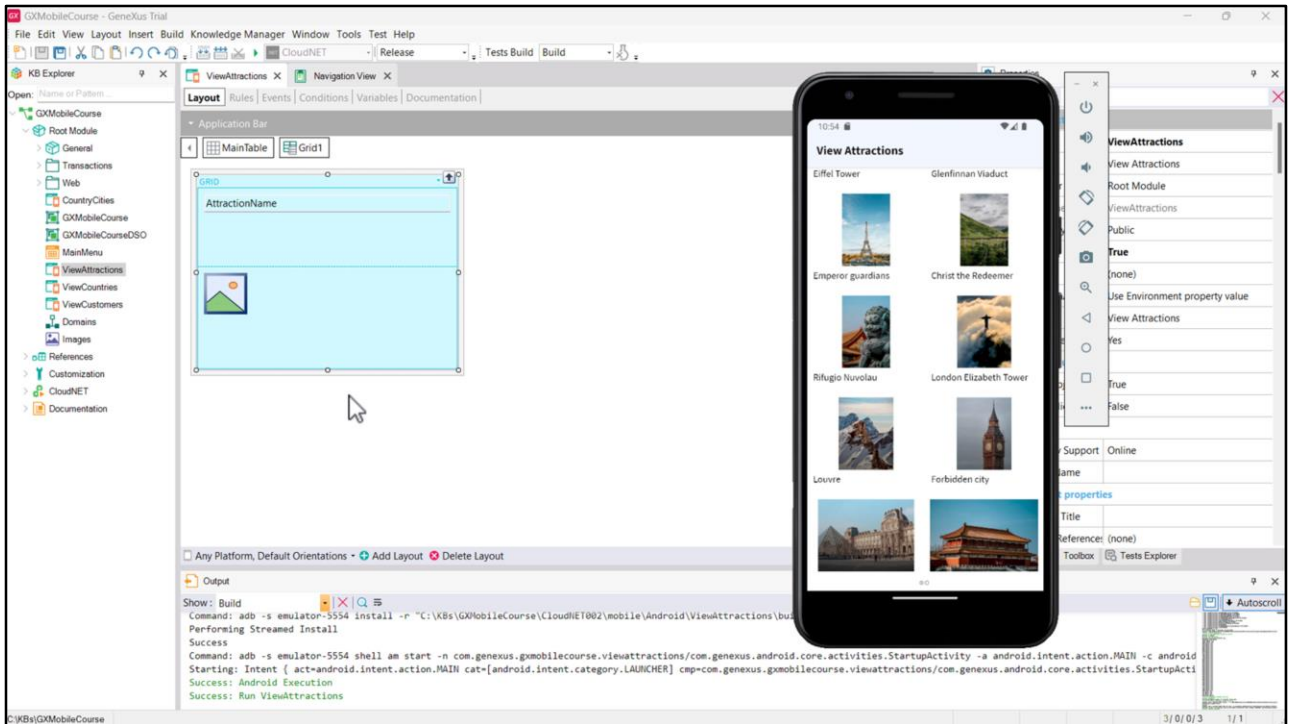


In the Control type property, we can see several controls that we can use to display the information. We choose the Horizontal Grid value.

By assigning this value, a series of properties will be enabled that will allow us to configure the Grid's appearance. For example, the Show Page Controller property determines whether the pager will be shown or not, and the Page Controller Class property will allow us to define the class on which this pager is based.



It is also possible to define how many columns we want to see and at the same time how many rows we want on each page. We are going to set the Columns Per Page Portrait property to 2 and leave the Rows Per Page Portrait property set to 4 because we want to see 4 rows per page.



We run it.

Now we can page to the right and we see the attractions divided into two columns per page. The first page has 4 rows at a time, as we configured the Rows Per Page Portrait property, and 3 on the second page, since we have no more attractions entered.

Below is the paging control, which shows what page we are on.

So far, we have seen configurations and types of Grids that help us to provide a better user experience in terms of information display and management. Note that, if we work with multiple parallel Grids in a Panel, the Load and Refresh events must be programmed for each Grid, as they will not be general to the Panel. We will study these events in detail in a future video.

Some interesting properties

- Enable Multiple Selection
<https://wiki.genexus.com/commwiki/wiki?16149>
- Pull To Refresh
<https://wiki.genexus.com/commwiki/wiki?29993>
- Inverse Loading
<https://wiki.genexus.com/commwiki/wiki?32660>

Finally, it is worth mentioning that we can configure some interesting properties that will allow us to have more control over the data displayed in the grids.

One of them is called *Enable Multiple Selection*, which allows a user to select multiple records of a Grid to perform some action on them, scanning all the selected elements using the *for each selected line* command.

<https://wiki.genexus.com/commwiki/wiki?16149>

Another is called *Pull To Refresh*. By setting it to True, the end user will be able to display a list and, after publishing it, the server will check if there is new data available. If so, it will synchronize and display the data without refreshing the whole Panel containing the Grid.

<https://wiki.genexus.com/commwiki/wiki?29993>

Finally, we will mention the *Inverse Loading* property, which allows reversing the direction of data loading in a Grid. When set to "True", it loads the grid data from the bottom upwards (in a common grid) and from right to left (in a horizontal grid). A typical example of use of this property is in chat messaging systems, where messages are loaded from bottom to top.

<https://wiki.genexus.com/commwiki/wiki?32660>



To learn more about each of these properties and to see practical examples of their use, you can read the documentation in the GeneXus Wiki.