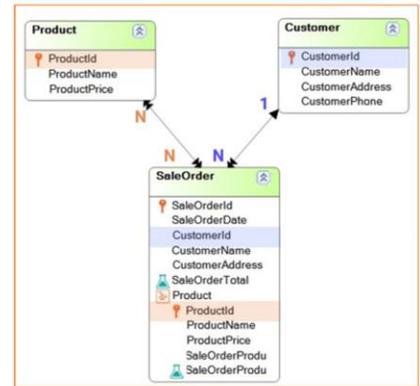
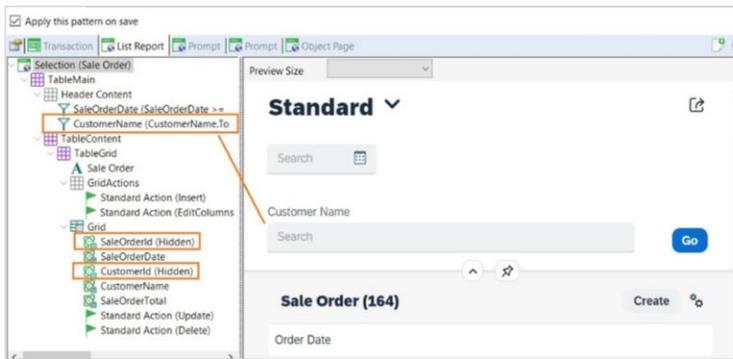


Using patterns: Fiori for Web

Part 3

Moving forward in our development, we are going to apply the Fiori for Web pattern to the missing transaction in our model, which is SalesOrder.

Fiori for Web: Sale order



Also, we will choose the ListReport floorplan again. We already know the functionalities that will be generated so we can customize it directly before running it.

The list of sales orders shows a filter by date, because it is the descriptor attribute, but we would also like to have a filter to search the orders by customer name. Remember that, according to our transaction design, a customer can have many associated orders.

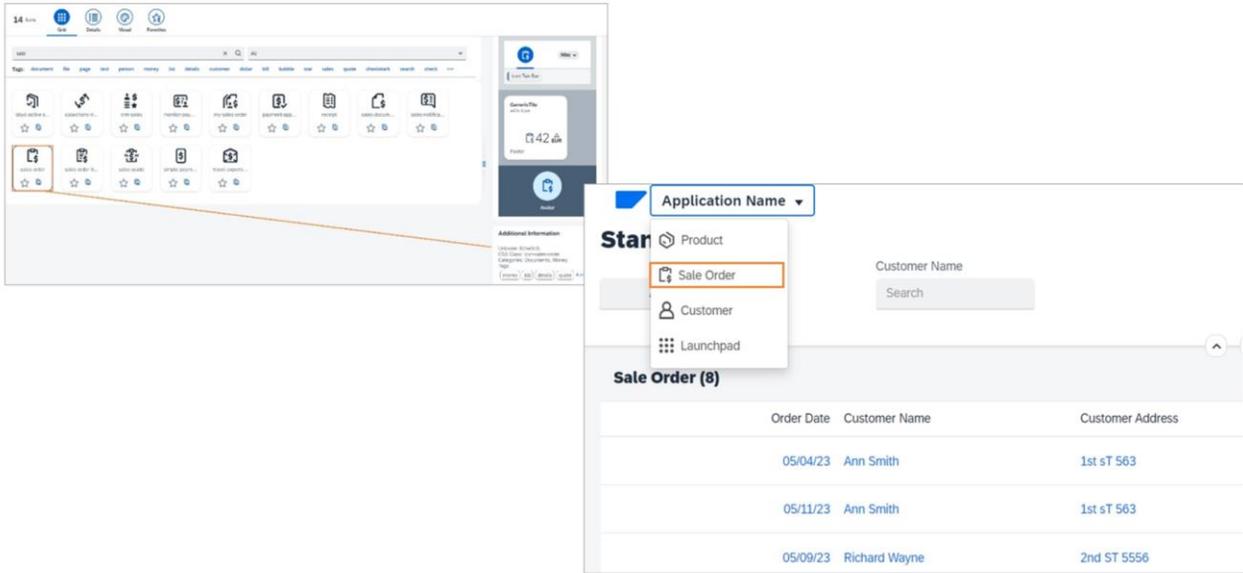
Therefore, we define the new filter in the Header content node of the ListReport tab.

As for the grid that shows the list of orders, we don't want the identifiers of the order or the customer to be shown.

When creating a new order, we don't want to see its identifier either since, as we said before, the identifier attributes are based on an autonumbered domain.

So, we hide the ID in the Transaction tab screen.

Fiori for Web: Sale order



The screenshot shows the Fiori for Web interface. On the left, a grid of application tiles is visible, with one tile highlighted. An orange arrow points from this tile to a dropdown menu titled 'Application Name'. The dropdown menu lists several options: 'Product', 'Sale Order' (highlighted with an orange border), 'Customer', and 'Launchpad'. Below the dropdown, a search bar for 'Customer Name' is present. The main content area displays a table titled 'Sale Order (8)' with the following data:

Order Date	Customer Name	Customer Address
05/04/23	Ann Smith	1st st 563
05/11/23	Ann Smith	1st st 563
05/09/23	Richard Wayne	2nd ST 5556

OK. We also want to include an access to Sales Order in the Master Page menu. We go back to the root of the List Report tab and enable the Include in Master Page Menu property.

We must not forget to look for the code of the icon that represents this entity...

Now let's see all these changes together at runtime. We press F5.

In the list of orders, we see the filter by customer name and don't see the order and customer identifiers.

If we create a new order... we don't see the ID either...

And we do see the access in Master Page menu. Perfect! We already have the screens as we wanted.

Let's go back to the launchpad... We are going to customize the card.

We want this card to show a chart with the number of sales orders by date.

How can we do it?

Query object

Query object

Structure	Description
SaleOrdersByDate	Sale Orders By Date
Attributes	
SaleOrderDate	Sale Order Date
Count(SaleOrderId)	Quantity of Sale Order Id
Parameters	
Filters	

Web component object

Web Layout: Roles | Events | Conditions | Variables | Help | Documentation | Patterns

<No action group selected>

MainTable QueryViewer1

Properties: UCWeb

Control Name: QueryViewer1

Data bindings

Object	SaleOrdersByDate
Elements	&Elements
Allow Elements On	False
Parameters	&Parameters

Appearance

Fiori launchpad tile

The image illustrates the integration of a Query object into a Fiori launchpad tile. On the left, the 'Structure' pane shows the 'SaleOrdersByDate' query object with attributes like 'SaleOrderDate' and 'Count(SaleOrderId)'. In the center, the 'Web Layout' editor shows a 'QueryViewer1' control. On the right, the 'Properties' pane shows the configuration for 'QueryViewer1', where the 'Object' property is set to 'SaleOrdersByDate'. Below, a 'Fiori launchpad tile' is shown with a 'Customer' profile, a 'Product' image, and a 'Sale orders by Date' chart. The chart displays a donut chart and a bar chart, with a legend for dates: 05/03/23 (blue), 05/10/23 (red), and 05/22/23 (orange).

GeneXus offers the Query object that allows defining dynamic queries in an intuitive and declarative way, and its output can be shown in different ways, for example, as a chart.

To see the Query object running, we must define a Web panel with a QueryViewer type control that will show the result of the dynamic query.

In turn, to view this web panel in the Fiori launchpad card, we must define it as a Web Component... What is a Web component? It is a web object that can be executed independently, like any web object, or that can be part of another web object, as we need here.

More about Query object:

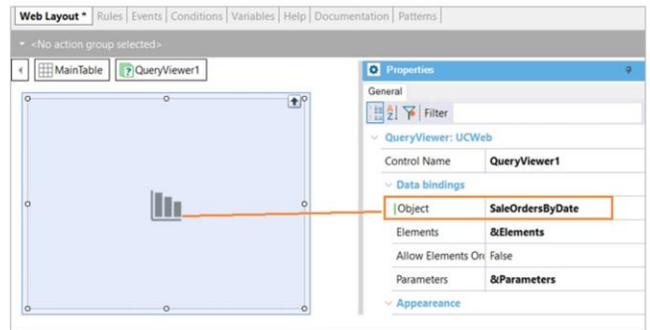
<tps://wiki.genexus.com/commwiki/servlet/wiki?9026,Category%3AQuery+object>

More about Web component object:

<https://wiki.genexus.com/commwiki/servlet/wiki?1864,Category%3AWeb+Component+object>

Fiori for Web: Tile personalization (Web component)

Structure	Description
SaleOrdersByDate	Sale Orders By Date
Attributes	
• SaleOrderDate	Sale Order Date
• Count(SaleOrderId)	Quantity of Sale Order Id
Parameters	
Filters	
OrderBy	



Let's go then to GeneXus to define the dynamic query and the web component. We create a Query object named SalesOrdersByDate, and define its structure. We want to see each date with its number of registered orders.

We save, and now define an object of Web component type...

From the Toolbox, we drag a control of Query viewer type... and in its Object property we declare the Query object that we have just created. In the set of properties below Output, we will specify how to display the query output. We choose Chart.

We enter a title for this chart...

And choose the type of chart. We have many options to choose from... This time we choose Doughnut.

We can have it show the values, but initially we choose not to show them.

We save.

Fiori for Web: Tile personalization (Web component)

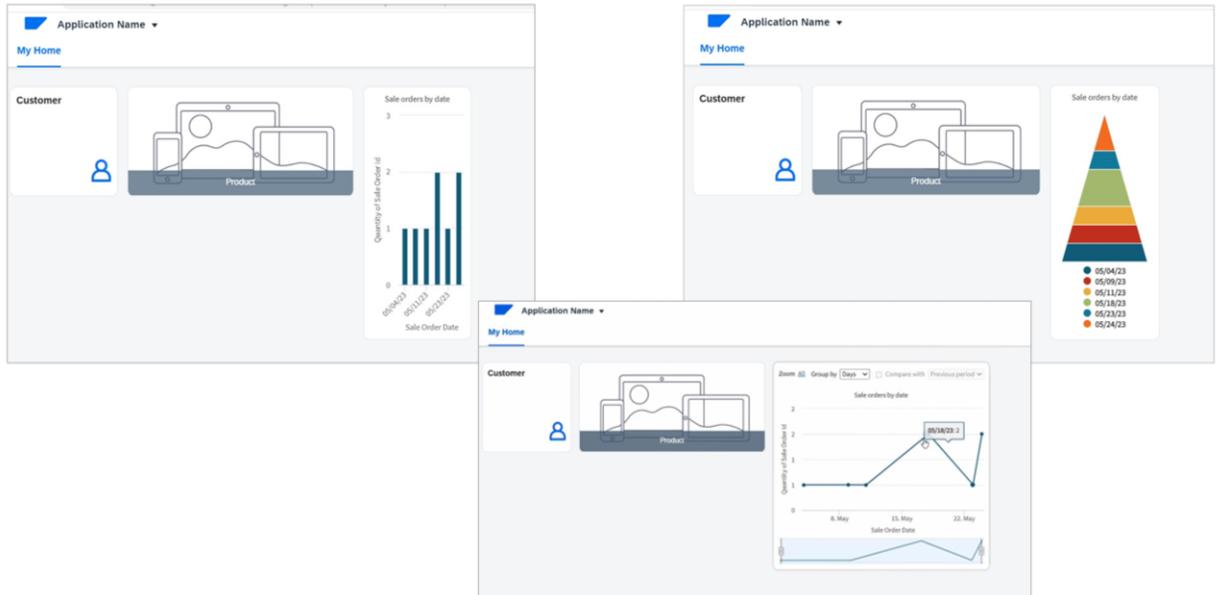
Is Main	False
Master Page	<default>
Form Theme Class	
Is Split Screen Template	False
Generation options	
Update Object	Create default
Web User Experience	Use model property
Base Web Form	Use model property
Launchpad	
Include In Launchpad	True
Tile Category	
Tile Type	Web Component
Tile Web Component	WCSaleOrdersByDate
Tile Size	Automatic
Tile Order Index	0

OK, so what do we have to do now? Integrate this web component to the Sales Order card in the Fiori launchpad.

We go to the transaction. In the root of the ListReport tab, we go to the Tile Type property and choose Web component.

Here we specify the object to show... and set the size of the card to automatic.

Fiori for Web: Tile personalization (more options)



Now we can see everything running. We press F5..

There are more options for you to discover...

GeneXus[™]
by **Globant**

training.genexus.com