

Named Query Binding

This feature is new in Ignition version **7.9.4**
[Click here](#) to check out the other new features

Binding Properties to a Named Query

The **Named Query** binding is where you can configure a property to call a Named Query that you had previously created in the project. Using Named Query binding instead of a **SQL Query** or **DB Browse binding** helps to make your project more secure due to the built-in **Security Zone** and **User Role** restrictions.

Polling Mode

Each Named Query binding type will use polling to determine when to update the results and run the query again. The Polling Mode dictates how often the query will execute, and works in a similar fashion to [polling on other bindings](#).

Creating Named Queries

In order to use a Named Query, first one has to be created. You might already have some that another developer has created but if not, you will have to make one by going to the **Named Query** section in the Designer or converting a SQL query.

On this page

...

- [Binding Properties to a Named Query](#)
- [Polling Mode](#)
- [Creating Named Queries](#)
- [Using Named Queries on Dataset Properties](#)
- [Using Named Queries on Scalar Properties](#)



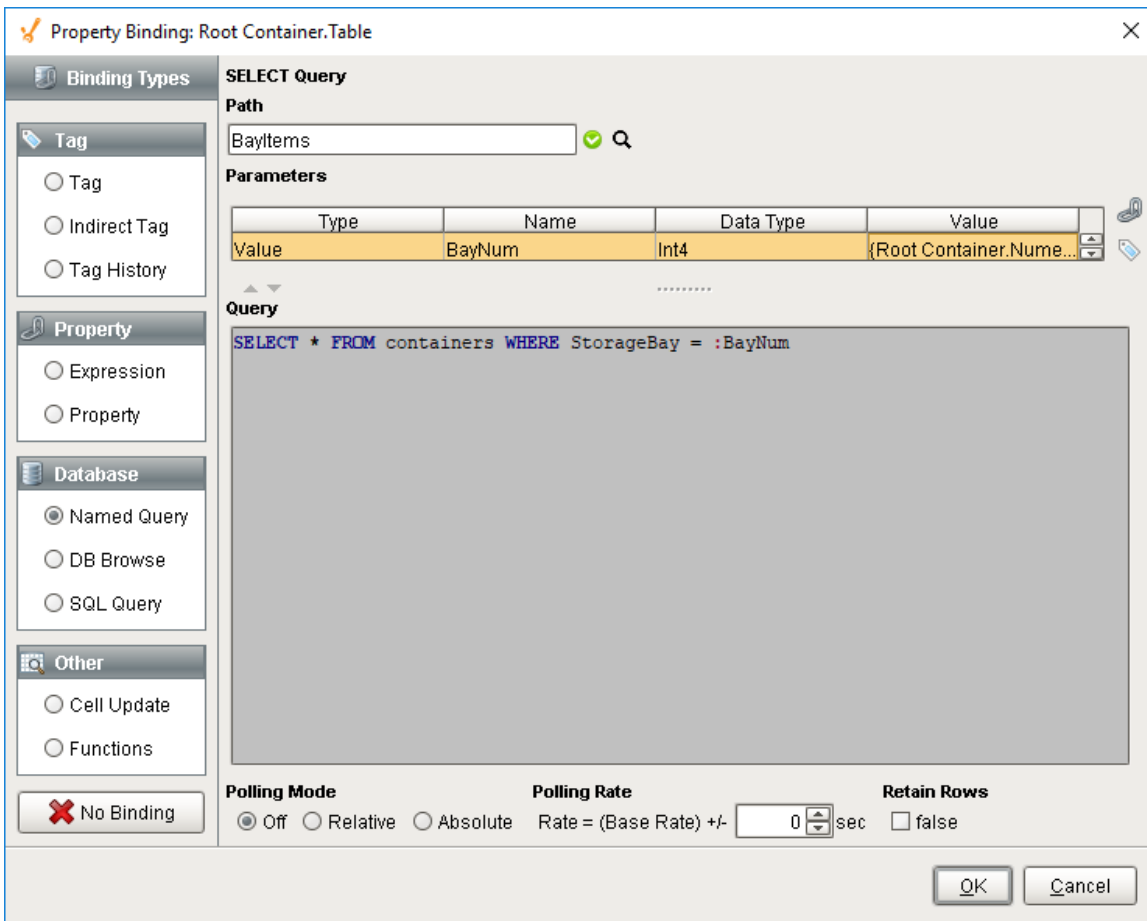
Named Query Binding

[Watch the Video](#)

Using Named Queries on Dataset Properties

The majority of your Named Query bindings will most likely be on a dataset type property. When placed on a Dataset type property, only a single Named Query needs to be specified. An explanation of the various fields on the binding are detailed below:

- **Path:** Here you can enter in the path to the Named Query. Click on the magnifying glass (🔍) button to get a list of all available Named Queries that you can choose from.
- **Parameters:** Here you can see a table of all defined Named Query parameters. You can pass in property or Tag values to the parameters by first highlighting the parameter and then selecting either the property (🏠) button or the Tag (🏷️) button
- **Query:** In the query section, there is nothing to configure, but you can see what the Named Query you have selected looks like.
- **Polling Mode:** Here you can set the Polling Mode of the Named Query binding based on the [Polling Rate](#).
- **Retain Rows:** If true, any rows that you have returned within the Designer will be saved along with the window. This may slow window load times.

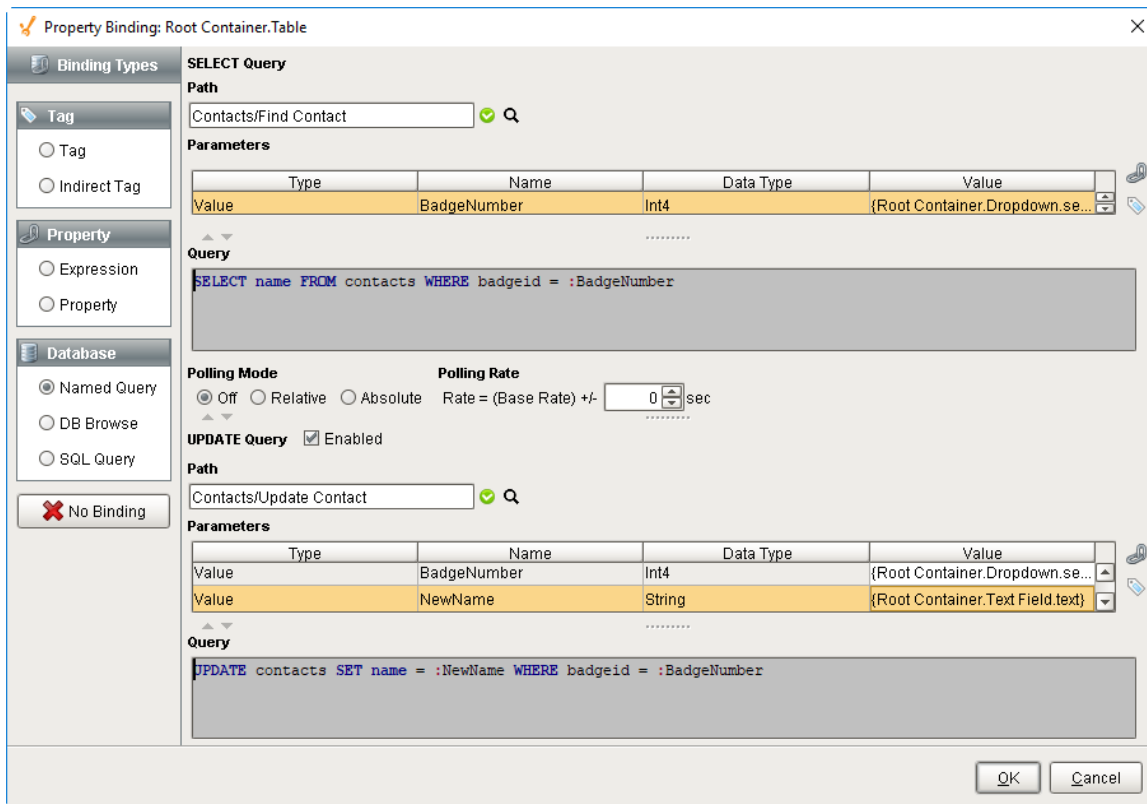


Using Named Queries on Scalar Properties

When placed on a non dataset type property (such as a String or Integer) then the Named Query binding allows for a second Named Query to be specified in the case that the user can update the value on the property. This provides an opportunity to both return and update values in the database from the same component.

The configuration is very similar to a Named Query binding on a dataset property. You need to specify a Named Query path, set up your Parameters, and choose a [Polling Mode](#). You can finish setup at this point, leaving the update query disabled so that the property will simply pull the value from the database.

However, if you want the binding to be bidirectional, you need to specify an UPDATE query. This also works much the same as the SELECT query in that you need to select the path to the Named Query and setup any Parameters. However, it is important to make sure that the Named Query chosen for the update query is in fact setup as an update query by setting the [Query Type property on the Authoring section of the Named Query](#).



Related Topics ...

- Cell Update Binding
- Named Queries
- Named Query Parameters