

# Query Tag Reference

Designed primarily to be used with large SQL databases, Query Tags let you save the results of a query and then later refer to the data set returned by that query without having to re-access the database. This can save time and system resources, because you aren't making unnecessary calls to the database.

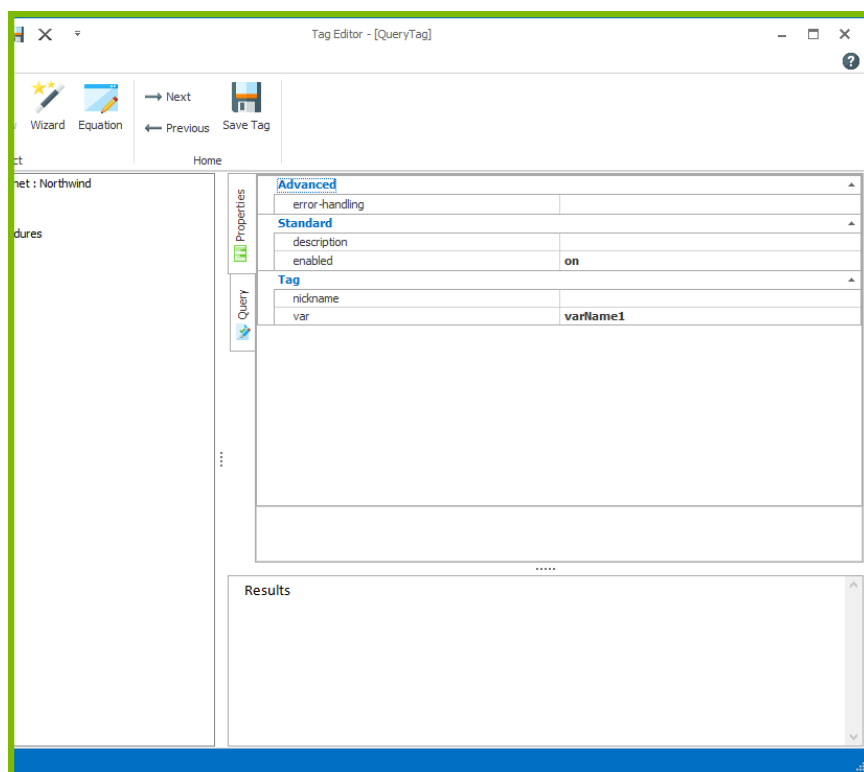
Use the Query Tag to retrieve the first row or node of the data set of a query from your data source, storing the result in a newly created variable. The name of this variable can subsequently be used in other tags as a relative data reference, using "\${VariableName}" syntax.

For an example of using a Query Tag, see [How Do I Use a Query Tag?](#)

**i** A Query Tag is different from a [Set Tag](#) in that the value of a Query Tag variable is the first row or node of a data set returned from a data source, while the value of a Set Tag variable is a string or number.

## Query Tag Properties

Here we see the properties of a Query Tag. Below, each property is described in detail. Unless otherwise noted, each property is required.



The rest of this article describes each section of the Set Tag properties:

[Tag Properties](#)

[Standard Properties](#)

[Advanced Properties](#)

## Tag Properties

**nickname** (optional)

The nickname will appear in the template rather than the generic "Query" label. Square brackets ("[]") will surround any text you enter here when it appears in the template to identify it as a Tag. Descriptive nicknames can be very important in designing complex templates.

**var** (default: *varName1*)

This property allows a user to set a variable name for the value returned to the Query Tag. The variable can be referenced in other Tags after it is set and used for comparisons, additional printing, filtering, etc.

## Standard Properties

**description** (optional)

A brief description of this Tag.

**enabled** (default: *on*)

Controls whether a Tag is executed when generating output. This can be useful when debugging a template.

- *on* - this Tag will be executed when output is generated
- *off* - this Tag will not be executed when output is generated
- *engine-only* - this Tag will be executed only if output is generated using one of the Report Engines
- *autotag-only* - this Tag will be executed only if output is generated using the Report Designer

## Advanced Properties

**error-handling** (optional)

Selects which types of errors produce *warnings* rather than *exceptions*, which allows output to be generated despite the error.

- *Ignore type error* - these errors occur when a Tag's defined data type is different than the data type of the data returned by the Tag's query
- *Ignore formatting error* - these errors occur when a Tag's specified format is not compatible with the format of the data returned by the Tag's query
- *Ignore select error* - these errors occur when a Tag's query fails to find valid data
- *Node must exist* - these errors occur when a row or node is queried which doesn't exist in the data source. These errors typically return an empty value that is output without warning.
- *Node must not return NULL* - these errors occur when a query returns an empty or NULL value. These errors typically return an empty value that is output without warning.
- *Treat warning as error* - forces all of the above error-handling warnings to appear as errors (not exceptions). This is useful when used with Report Designer's Verify feature.