Lists

To setup a list view, simple copy/paste the URL of a view.

Optimally you should strip anything but the parameters from the URL.

```
?SagID=252&command=list&QUERY_FIELD_1=OPFLGNING& ...
```

Filtering the list

To filter the list, add three parameters to the url, pr filter. Make sure that every set of parameters have their own unique number.

```
?SagID=252&command=list&QUERY_FIELD_1=OPFLGNING&QUERY_PARAM_1=1&QUERY_VALUE_1=0 ...
```

Operators:

ID	Command	Note
1	=	Only rows with values equal to
2	<	Only rows with value lower than
3	>	Only rows with values higher than
4	!=	Only rows with values not equal to
5	<=	Only rows with value lower than or equal to
6	>=	Only rows with values higher than or equal to
7	CONTAINS	
8	Starts with	
9	Ends with	
10	IS null	
11	IS NOT NULL	
15	IN DAY	
16	IN MONTH	
17	IN YEAR	
20	HAS	

21	HAS NOT	
22	HAS NONE	
23	HAS ANY	
24	IN	Only rows with values equal to one of the given (comma separated list)

If you are signed in as an admin on a server with a version newer than 7956, you will have access to the "url button".

This button redirects you to the full url, that generates the current filtering, that you have on your list.

This might be an easier way to generate the correct url, when creating dashboard-list-widgets.

Special case: List of values in search parameters

Currently lists of search parameters are encoded

```
... &QUERY_VALUE_3=11344%2C11343%2C11345& ...
```

To make them work in list views, please decode the kommas

```
... &QUERY_VALUE_3=11344,11343,11345& ...
```

(just change "%2C" to ",")

Dynamic selectors

A special case of the SQL query is the selectors. A selector will allow you to select a value or an ID for use as a parameter in another SQL query or a List view.

A selector will have a name defined by setting its column name in the query

```
SELECT DataID, CUSTOMER AS SelectedCustomer FROM data_customer WHERE StatusID = 123
```

In this case the value of DataID is set in a user specific variable named SelectedCustomer. The proper notation is in brackets [].

This can now be used in list and SQL definitions

```
SELECT BILLINGDATE, AMOUNT FROM data_invoice WHERE CUSTOMER = [SelectedCustomer]
```

Revision #2
Created 25 March 2025 15:20:16 by Theis Villumsen

Updated 18 December 2025 08:58:04 by Theis Villumsen