SAP Mobility 101

Tutorial 10 – Filters

# Objective of Exercise

## Build an example application

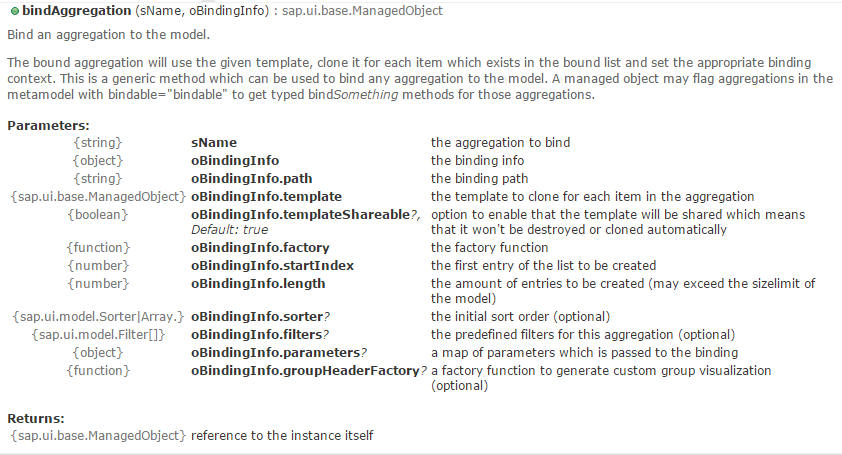
The objective of this exercise is to build an HTML page that uses JavaScript and output values in a filtered table.

## Note

* We recommend that you use a chrome browser for testing
* Eclipse Juno would be needed for this Tutorial.

# Task 1: Add a filter to the table

In the API reference of bindItems, there are a property that can be set: filters.

Add a filter named oFilter to the Products table on the second view:

list.bindAggregation("items", {

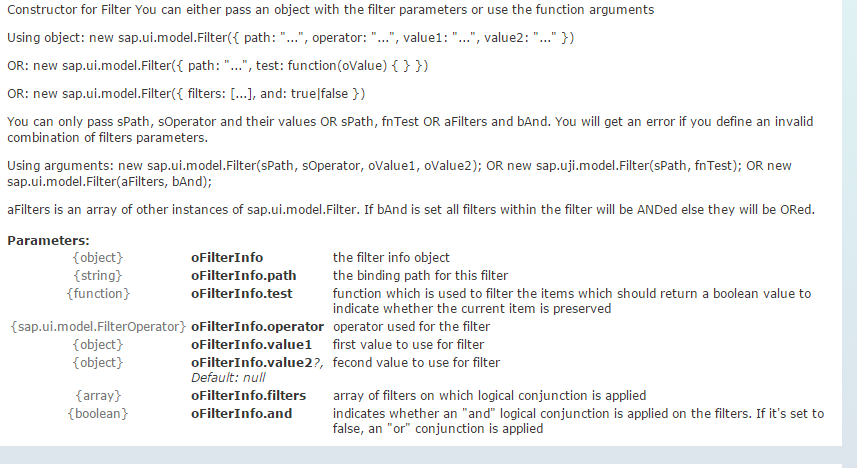
path: "data2>/Products",

filters: oFilters,

# template: otemplate});

# Task 2: Create the filter

Under the API reference for filter, there are different methods of creating a new filter:



Create a filter that only lets the CategoryID 2 through:

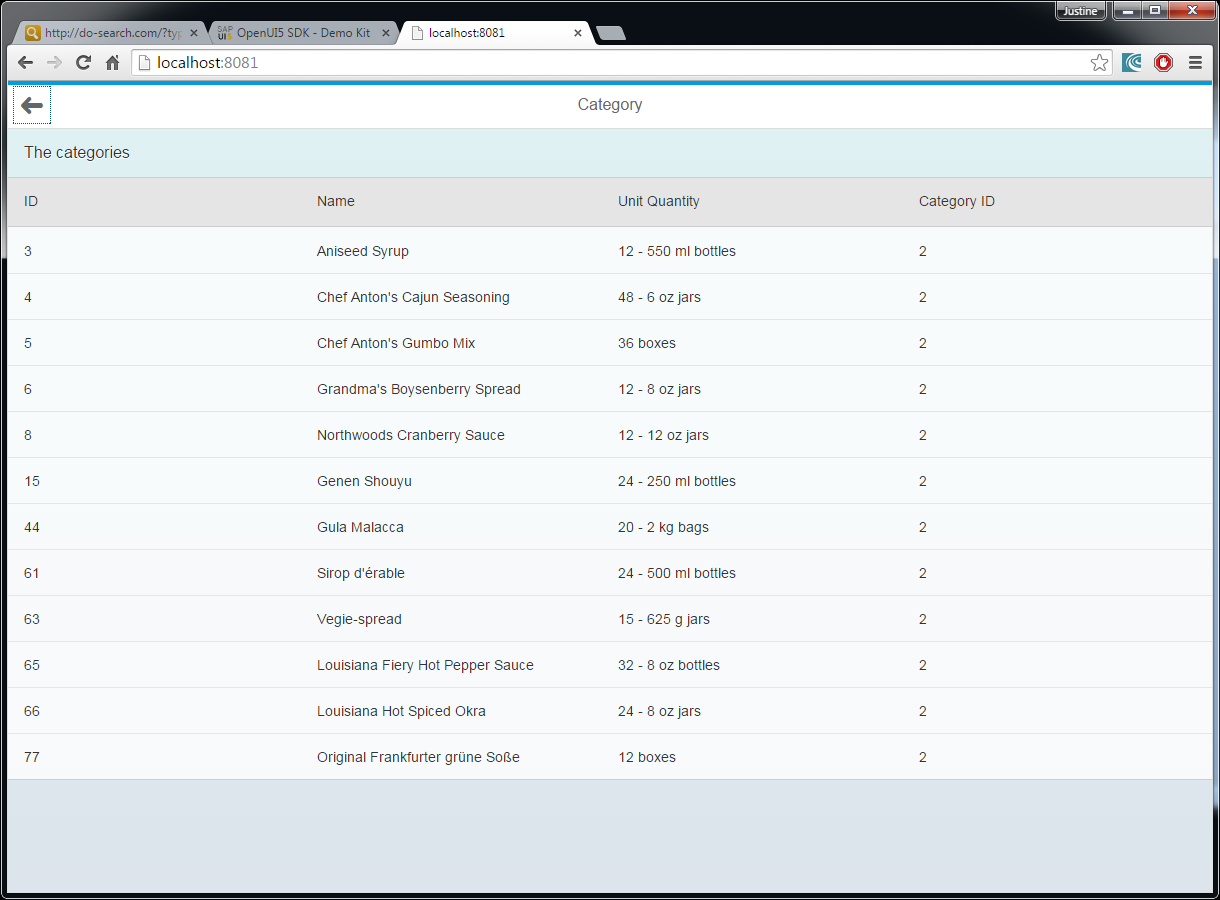
**var** oFilters = [

**new** sap.ui.model.Filter(

"CategoryID", "EQ", "2"

)];

Result:



# 

Note: the sap.m developers are currently working on having the filter value bound to a data model, this will be a very usefull application. The filter can then be used as follow:

**var** oFilters = [

**new** sap.ui.model.Filter(

"CategoryID", "EQ", "{data1>CategoryID}"

)];