

# Hacking dependent values

The following article is subject to change during 2020: The functions **lookupBuilderShowAll** and **lookupBuilderFilter** will probably become a part of the standard TS javascript library.

## Example: Show all sub categories for a distinct main value

In the following the field CAT2 will display all possible categories, if the field CAT has the value "alle"

```
$("#DATA_CAT1").on('change', function() {  
    if( getValue("CAT1") == "alle" )  
        lookupBuilderShowAll("DATA_CAT2");  
});
```

```
/* remove function after next update after 2020-01-08 */  
function lookupBuilderShowAll(dependantField) {  
    var selectTo = document.getElementById(dependantField);  
    var selectToValue = selectTo[selectTo.selectedIndex].value;  
    var optionList = masterOptionContainer[dependantField];  
    var optionListLength = optionList.length;  
    var optionCount = 0;  
    selectTo.options[0] = new Option( "", "" );  
    for( var i=0; i < optionListLength; i++ )  
    {  
        var optionItem = optionList[i];  
        var dependID = optionItem["dependID"];  
        optionCount++;  
        selectTo.options[optionCount] = new Option( optionItem["itemValue"], optionItem["itemID"]);  
        if( selectToValue == optionItem["itemID"] ) {  
            selectTo.options[optionCount].selected = true;  
        }  
    }  
}
```

# Example: Filter categories based on value in another field

In the following the field CAT2 will filter out values containing ":", where the value of the field NAVN cannot be found in the category name.

```
$("#DATA_CAT1").on('change', function() {  
  if( getValue("CAT1") == "andet" )  
    lookupBuilderFilter("DATA_CAT2",getValue("NAVN"),':');  
});
```

```
/* remove function after next update after 2020-01-08 */  
function lookupBuilderFilter(dependantField,lookForValue,lookForMarker) {  
  var selectTo = document.getElementById(dependantField);  
  var selectToOptions = selectTo.options;  
  var optionListLength = selectToOptions.length;  
  for( var i=0; i < selectToOptions.length; i++ )  
  {  
    var itemValue = selectToOptions[i].text;  
    if( itemValue.indexOf(lookForValue) == -1 && itemValue.indexOf(lookForMarker) > -1 ) {  
      console.log("Removing element: " + itemValue );  
      console.log("Removing element: " + selectToOptions[i].value );  
      selectTo.remove(i);  
    }  
  }  
}
```

---

Revision #1

Created 8 April 2025 13:22:06 by Theis Villumsen

Updated 8 April 2025 13:22:37 by Theis Villumsen