Subject: Dropdown list
Posted by htManager on Sun, 22 Jun 2014 08:06:14 GMT
View Forum Message <> Reply to Message

Hi Tony,

I try to use the same option list in the text/<language>/language_array.inc file to serve different colomns in different tables.
The colomn names are different as you discribed in one of your articles.
But only if I name the option list as the colomn name it will be filled. I used the sample in Tutorial 5 (5. Convert STAR_SIGN into a dropdown list).

```
function _cm_getExtraData ($where, $fieldarray)
  // get values for star_sign and insert into lookup array
  $array = $this->getValRep('star_sign');
```

In this line I thought to put the different column name:

```
  $this->lookup_data['different_column_star_sign'] = $array;
```

Is this correct or am I wrong?

Best regards,

Juergen

Subject: Re: Dropdown list
Posted by AJM on Sun, 22 Jun 2014 20:28:03 GMT
View Forum Message <> Reply to Message

While it is usual to name the option list after the column to which it belongs, it is not mandatory. This is why the data dictionary allows you to specify whatver name you want. In this way you can use the same list for multiple fields.

Subject: Re: Dropdown list
Posted by htManager on Mon, 23 Jun 2014 07:45:38 GMT
View Forum Message <> Reply to Message

I also thought so but it doesn't work with me. Do I have to take attention of something else or is something wrong with my code?

Here is my code:

```
function _cm_getExtraData ($where, $fieldarray)
 // Perform custom processing for the getExtraData method.
 // $where = a string in SQL 'where' format.
```

```
 // $fieldarray = the contents of $where as an array.
 {

     // get values for groesse_international and insert into lookup array
        $array = $this->getValRep('groesse_international');
        $this->lookup_data['kontakt_ausruestung_groesse_international'] = $array; // colomn of the
table

        return $fieldarray;

    } // _cm_getExtraData

function _cm_getValRep ($item, $where)
 // get Value/Representation list as an associative array.
 {
     $array = array();
        if ($item == '_groesse_international') {
            $array = getLanguageArray('groesse_international');
            return $array;
 } // if

     return $array;

 } // _cm_getValRep
```

Subject: Re: Dropdown list
Posted by AJM on Mon, 23 Jun 2014 07:58:28 GMT
View Forum Message <> Reply to Message

When you say "it doesn't work" what exactly do you mean? Is your array of values which you insert into $this->lookup_data empty? Or can't you load this list into your dropdown?

Whatever name you use for each list in $this->lookup_data MUST be the same as you assign to 'Option List' in the 'Update Column' screen in the data dictionary. This name need not be the same the column name, but it must point to an entry in $this->lookup_data.

Subject: Re: Dropdown list
Posted by htManager on Mon, 23 Jun 2014 08:06:33 GMT
View Forum Message <> Reply to Message

Thank you for the last response. I now understand how it works. The relation between the option list and the values is made in the data dictionary and not in the class! I thought that also in the class file there must be an entry for the colomn.
I changed the correct option list name in the class file and everything works fine now.

Thank you very much.