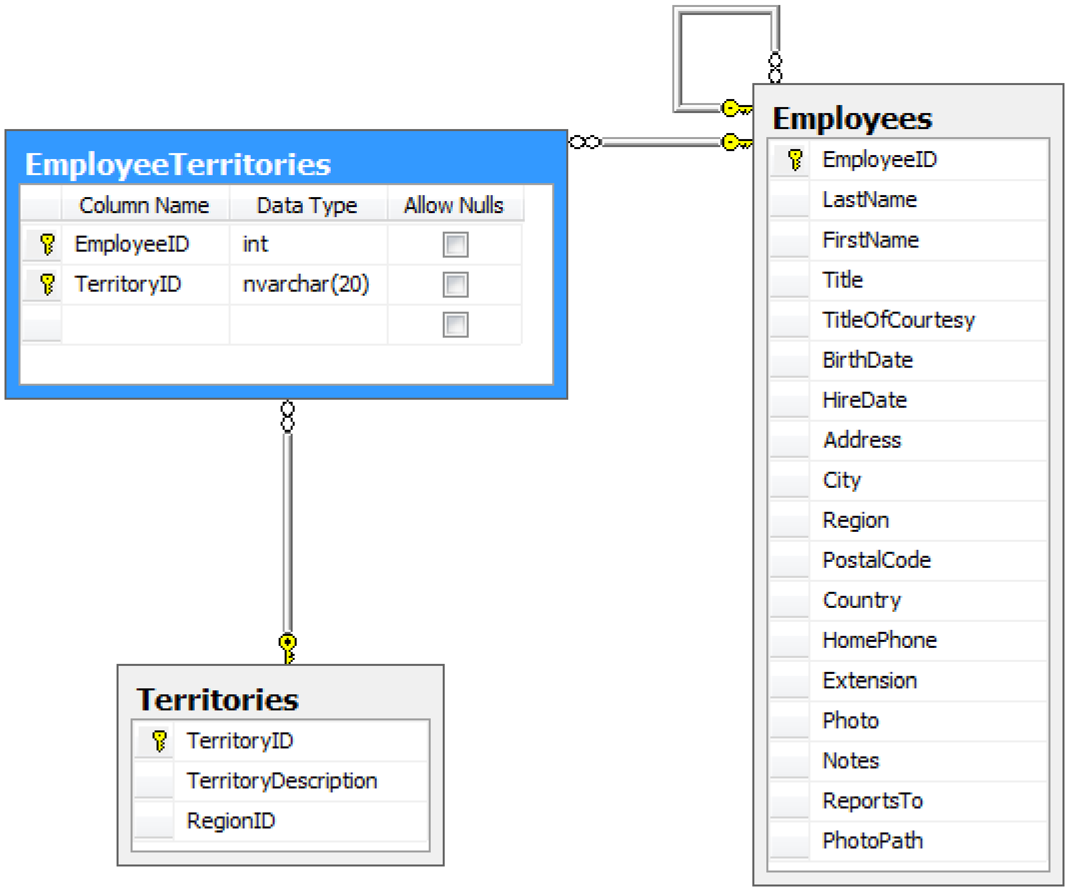
2011

COOKBOOK

Using $external() Function in Filter Expressions

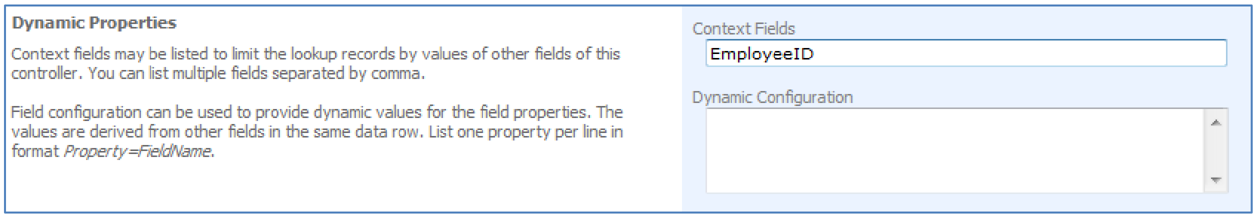
The function *$external* has been introduced in the latest *Code On Time Generator* update. This function can be used to access values in the URLs of the pages and values of fields listed in the *Context Fields* property of lookup fields.

For example, consider the *EmployeeTerritories* table from the *Northwind* sample database. Territories can only be associated to an employee one time due to the primary key constraint.



Let’s use the *$external()* function to insure that, when creating a new *Employee Territories* record in the web application, territories previously assigned to the relevant employee are not available.

Start *Code On Time Generator*, click on the project name, and press the *Design* button. Select the *EmployeeTerritories* data controller. Switch to the *Fields* tab and select the *TerritoryID* field. In the *Context Fields* field, under *Dynamic Properties* section, enter “EmployeeID”. Save your changes.



Dynamic Properties

Context fields may be listed to limit the lookup records by values of other fields of this controller. You can list multiple fields separated by comma.

Field configuration can be used to provide dynamic values for the field properties. The values are derived from other fields in the same data row. List one property per line in format *Property=FieldName*.

Context Fields: EmployeeID

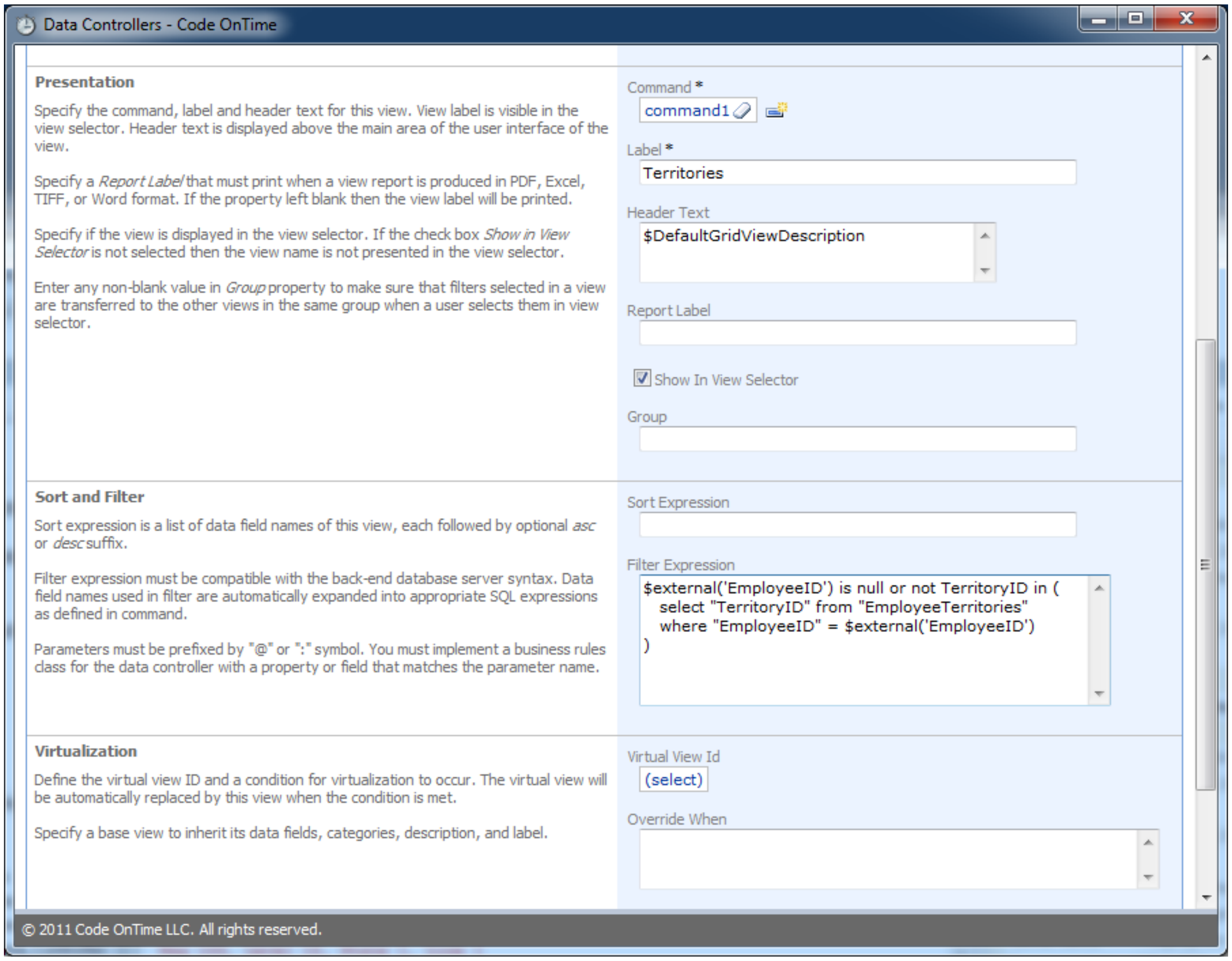
Dynamic Configuration:

The application will now pass the value of the selected *EmployeeID* to the lookup view *Territories.grid1* attached to the field *TerritoryID*. If the lookup view has the field *EmployeeID* then the rows of *Territories* will be automatically filtered to match the value of *EmployeeID* selected in the new record.

However, there is currently no field named *EmployeeID* in the table *Territories*.

Let's create a filter expression. Go back to the *All Controllers* list, and select the *Territories* controller. Switch to the *Views* tab and select *grid1*. Press *Edit*. In the *Filter Expression* property, enter the following:

```
$external('EmployeeID') is null or not TerritoryID in (
   select "TerritoryID" from "EmployeeTerritories"
   where "EmployeeID" = $external('EmployeeID')
)
```


Data Controllers - Code OnTime

Presentation

Specify the command, label and header text for this view. View label is visible in the view selector. Header text is displayed above the main area of the user interface of the view.

Specify a *Report Label* that must print when a view report is produced in PDF, Excel, TIFF, or Word format. If the property left blank then the view label will be printed.

Specify if the view is displayed in the view selector. If the check box *Show in View Selector* is not selected then the view name is not presented in the view selector.

Enter any non-blank value in *Group* property to make sure that filters selected in a view are transferred to the other views in the same group when a user selects them in view selector.

Command * : command1

Label * : Territories

Header Text: $DefaultGridViewDescription

Report Label:

[✓] Show In View Selector

Group:

Sort and Filter

Sort expression is a list of data field names of this view, each followed by optional *asc* or *desc* suffix.

Filter expression must be compatible with the back-end database server syntax. Data field names used in filter are automatically expanded into appropriate SQL expressions as defined in command.

Parameters must be prefixed by "@" or ":" symbol. You must implement a business rules class for the data controller with a property or field that matches the parameter name.

Sort Expression:

Filter Expression:

```
$external('EmployeeID') is null or not TerritoryID in (
   select "TerritoryID" from "EmployeeTerritories"
   where "EmployeeID" = $external('EmployeeID')
)
```

Virtualization

Define the virtual view ID and a condition for virtualization to occur. The virtual view will be automatically replaced by this view when the condition is met.

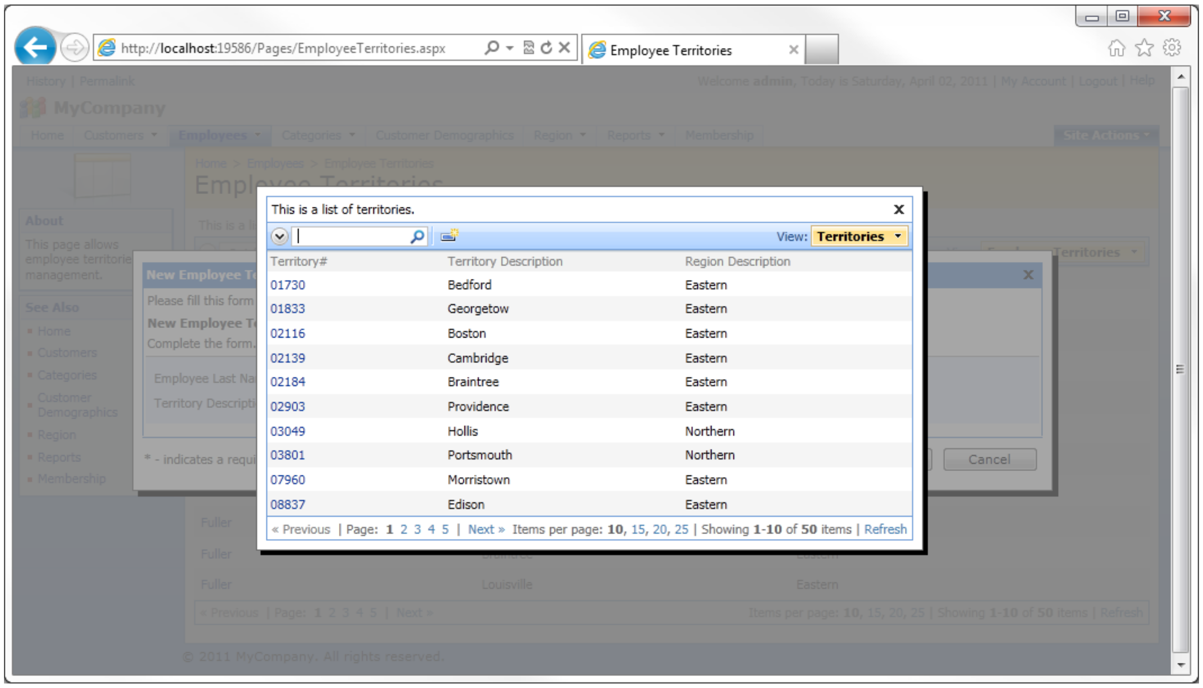
Specify a base view to inherit its data fields, categories, description, and label.

Virtual View Id: (select)

Override When:

© 2011 Code OnTime LLC. All rights reserved.

Save the view, exit the designer, and generate the application. When the web page loads, navigate to the *Employee Territories* page. Create a new employee territory, and select *Davolio* for the *Employee Last Name* field.


http://localhost:19586/Pages/EmployeeTerritories.aspx

This is a list of territories.

View: Territories

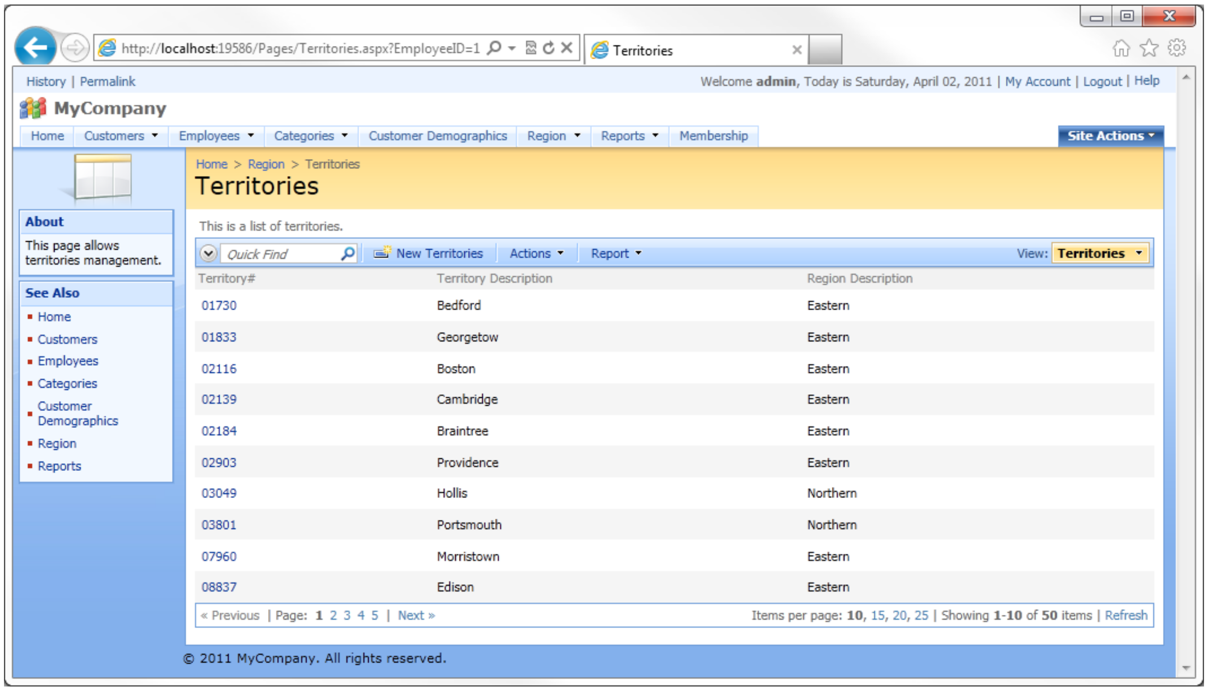
Territory#	Territory Description	Region Description
01730	Bedford	Eastern
01833	Georgetow	Eastern
02116	Boston	Eastern
02139	Cambridge	Eastern
02184	Braintree	Eastern
02903	Providence	Eastern
03049	Hollis	Northern
03801	Portsmouth	Northern
07960	Morristown	Eastern
08837	Edison	Eastern

« Previous | Page: 1 2 3 4 5 | Next » Items per page: 10, 15, 20, 25 | Showing 1-10 of 50 items | Refresh

When the lookup is activated for *Territory Description*, a list will appear with 50 out of 53 records available. The three missing records are territories already assigned to *Davolio*.

If you navigate to the *Territories* page, all 53 records will be present. If you enter the URL parameter of the correct *EmployeeID*, then the territories will be filtered accordingly, with only 50 territories. For Davolio, the URL should look similar to

http://localhost:36745/Pages/Territories.aspx?EmployeeID=1


http://localhost:19586/Pages/Territories.aspx?EmployeeID=1

History | Permalink — Welcome admin, Today is Saturday, April 02, 2011 | My Account | Logout | Help

MyCompany

Home > Region > Territories

Territories

This is a list of territories.

View: Territories

Territory#	Territory Description	Region Description
01730	Bedford	Eastern
01833	Georgetow	Eastern
02116	Boston	Eastern
02139	Cambridge	Eastern
02184	Braintree	Eastern
02903	Providence	Eastern
03049	Hollis	Northern
03801	Portsmouth	Northern
07960	Morristown	Eastern
08837	Edison	Eastern

« Previous | Page: 1 2 3 4 5 | Next » Items per page: 10, 15, 20, 25 | Showing 1-10 of 50 items | Refresh

© 2011 MyCompany. All rights reserved.

The filter expression assigned to the *grid1* view does not work if there is no external filter passed in the page URL or in the *Context Fields* of the lookup view. This is guaranteed by the first comparison in the expression:

```
$external('EmployeeID') is null or . . .
```

The second part of the filter expression tests *TerritoryID* to ensure that it is not matched to any territories not already present in *EmployeeTerritories* and uses *$external('EmployeeID')* to further limit the scope of the test.

Notice the use of double quotations around field and table names. Please use the appropriate symbol that works with your database server. This will ensure that your application will not be trying to resolve the name against the dictionary of fields of the data controller.

Here is the physical SQL statement executed by your application. Note that parameters *@p0* and *@p1* will be replaced with the value of the *EmployeeID* passed in a URL or as a context field.

```
with page_cte__ as (
select
row_number() over (order by "Territories"."TerritoryID") as row_number__
,"Territories"."TerritoryID" "TerritoryID"
,"Territories"."TerritoryDescription" "TerritoryDescription"
,"Territories"."RegionID" "RegionID"
,"Region"."RegionDescription" "RegionRegionDescription"
from
"dbo"."Territories" "Territories"
    left join "dbo"."Region" "Region" on "Territories"."RegionID" =
"Region"."RegionID"

where
(
(((@p0) is null or not "Territories"."TerritoryID" in (
   select "TerritoryID" from "EmployeeTerritories"
   where "EmployeeID" = (@p1)
))))
)
select * from page_cte__ where row_number__ > @PageRangeFirstRowNumber and
row_number__ <= @PageRangeLastRowNumber
```