

Dag 07, Opgaver – DataAdapter, DataTable og DataGrid

Exercise 1, Databinding to DataGrid

Make a classlibrary project today's solution. In this project create a Service class and eventually a DAL class. (Delete the class1.cs file which is created.)

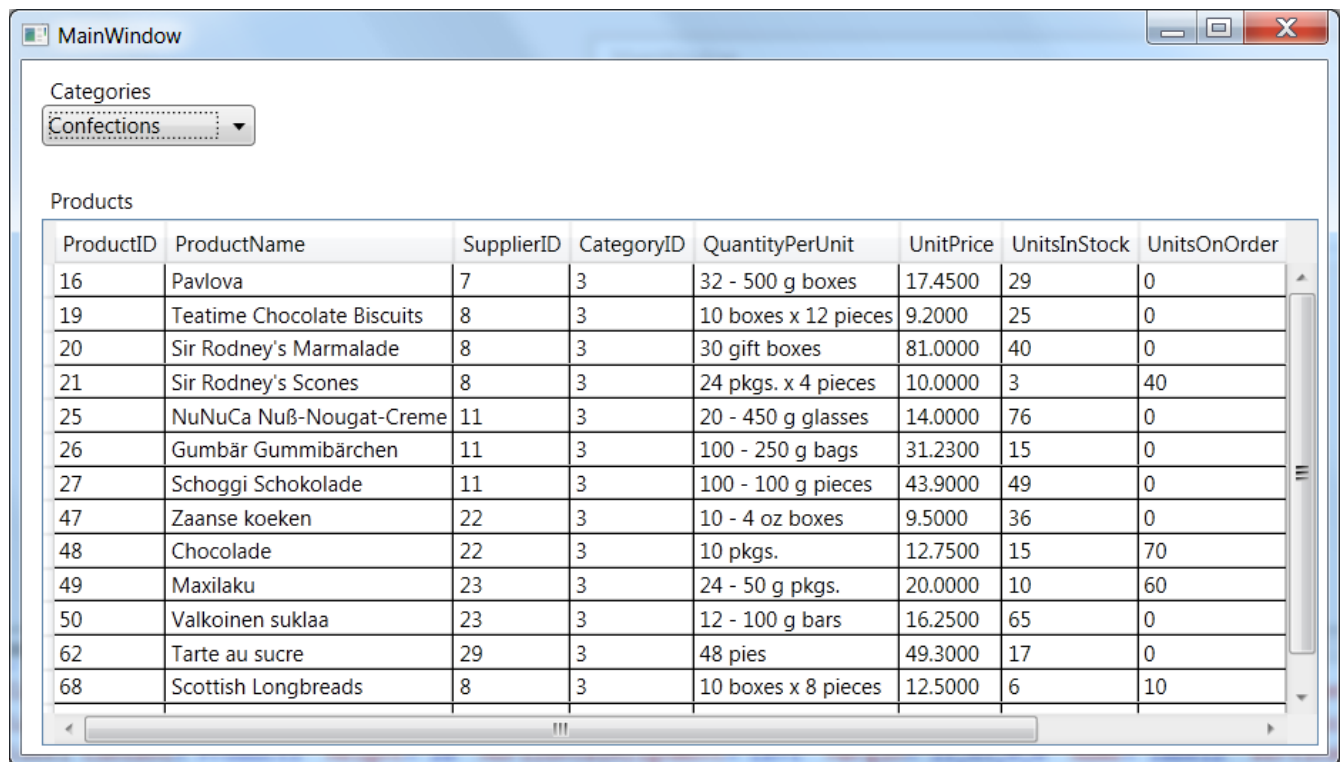
All database access must be in this classlibrary.

Make a project of type WPF application. Add a reference to the classlibrary above.

The WPF application should show Products for a selected Category. Both the ComboBox and the DataGrid must be bound to a datasource by setting the property ItemsSource.

Set *DisplayMemberPath* and *SelectedValuePath* for the ComboBox.

Set *SelectedValuePath* for the DataGrid.



The Category is filled into a DataTable. When the user selects a category another DataTable is filled with products of the corresponding category. The DataGrid is bound to the DataTable's DefaultView.

When the above is implemented make some new facilities for the DataGrid:

- Hide the ProductID and the CategoryId columns
- Changes few of the HeaderTitles.
- Change color in the grid using alternating colors also.

d) Add a Label above the Datagrid. When the user selects a row in the DataGrid, the label should show how many Customers have ordered that product. **REMEMBER: No database access code in MainWindow!!!**

Exercise 2, *Editing*

Make a new WPF application which in a DataGrid shows the Categories table in NorthWind *but ignore the column with an image*.

Copy the Service class and DAL from Exercise 1 to this new Project (Right-click the solution in the Solution Explorer and select: Add → Existing Item).

Extend the Service class with four new method:

```
public DataTable GetCategories();  
public void DeleteRow(int categoryId)  
public void UpdateRow(int categoryId, string categoryName, string categoryDesc)  
public void NewRow(string categoryName, string categoryDesc)
```

Bind the Datagrid to the a DataTable returned by GetCategories().

Set properties such that the user can't edit in the DataGrid i.e. neither add rows nor delete rows, and not been able to edit any cells.

Add three Buttons: New, Edit, Delete.

If the New-button is clicked a new Window should be opened. It has two TextBoxes for CategoryName and Description, a Save button and a Cancel button. The dialog uses the NewRow method above.

If the Edit-button is clicked and a row is selected in the DataGrid the same dialog is open, but now the values from the row is transferred to the dialog window. The dialog uses UpdateRow method above.

Finally implement a delete of the selected using the DeleteRow method above. The user must confirm the delete before it is performed.

In the Main window the DataGrid must be updated after any changes.