

Restricting Read/Write Access to Fields

[Data Aquarium Framework](#) provides multiple options to allow precise control of *read* and *write* access to data fields displayed in grid and form views.

Generate an [Aquarium Express](#) application with [Code OnTime Generator](#) and *Northwind* database. Open `~/Controllers/Products.xml` and follow instructions to try various methods of restricting user's ability to change the *UnitPrice* field values.

ReadOnly Attribute

If you want to prevent your users from changing field values then defining *readOnly* attribute on fields in data controller will do the trick.

```
<field name="UnitPrice" type="Decimal" default="(0)"
  label="Unit Price" readOnly="true"/>
```

This is how the field is rendered in the grid view of sample application when you try to edit any row.



Notice that *readOnly* attribute will affect all grid and form views in the data controller.

Duplicate Field With ReadOnly Attribute

Sometimes you may want to prevent users from editing field in a grid view but still want to allow editing in forms. A simple solution is to define an additional field in your SQL query with matching *field* element adorned with *readOnly* attribute.

Here is how you can change the query that retrieves *Products*. The query includes a duplicate field that selects *UnitPrice* with an alias *UnitPriceReadOnly*.

```
select
  "Products"."ProductID" "ProductID"
```

```

, "Products"."ProductName" "ProductName"
, "Products"."SupplierID" "SupplierID"
, "Supplier"."CompanyName" "SupplierCompanyName"
, "Products"."CategoryID" "CategoryID"
, "Category"."CategoryName" "CategoryCategoryName"
, "Products"."QuantityPerUnit" "QuantityPerUnit"
, "Products"."UnitPrice" "UnitPrice"
, "Products"."UnitPrice" "UnitPriceReadOnly"
, "Products"."UnitsInStock" "UnitsInStock"
, "Products"."UnitsOnOrder" "UnitsOnOrder"
, "Products"."ReorderLevel" "ReorderLevel"
, "Products"."Discontinued" "Discontinued"
from "dbo"."Products" "Products"
left join "dbo"."Suppliers" "Supplier"
    on "Products"."SupplierID" = "Supplier"."SupplierID"
left join "dbo"."Categories" "Category"
    on "Products"."CategoryID" = "Category"."CategoryID"

```

Add a matching field *UnitPriceReadOnly* to the *fields* section of the data controller.

```

<field name="UnitPriceReadOnly" type="Decimal" default="(0)"
    label="Unit Price" readOnly="true"/>

```

Next change the definition of *grid1* view to prevent users from editing the field by replacing reference to *UnitPrice* with the reference to *UnitPriceReadOnly*.

```

<view id="grid1" type="Grid" commandId="command1" label="Products">
    <headerText>This is a list of products. </headerText>
    <dataFields>
        <dataField fieldName="ProductName" columns="40" />
        <dataField fieldName="SupplierID" aliasFieldName="SupplierCompanyName" />
        <dataField fieldName="CategoryID" aliasFieldName="CategoryCategoryName"
    />
        <dataField fieldName="QuantityPerUnit" columns="20" />

```

```
<dataField fieldName="UnitPriceReadOnly"
  dataFormatString="c" columns="15" />
<dataField fieldName="UnitsInStock" columns="15" />
<dataField fieldName="UnitsOnOrder" columns="15" />
<dataField fieldName="ReorderLevel" columns="15" />
<dataField fieldName="Discontinued" />
</dataFields>
</view>
```

Form views *editForm1* and *createForm1* will retain references to the original field *UnitPrice*, which allows users to make price adjustments in form mode only.

WriteRoles Attribute

A superior approach is to rely on the security infrastructure of *ASP.NET* and its support in [Data Aquarium Framework](#).

Attribute *roles* defined on a field will limit visibility of the field in views to users with the specified roles. Field *writeRoles* will prevent users from changing the field if the user's role is not on a comma-separated list.

If you add this attribute as shown in the snippet then the result will be a read-only display to all users that do not have *Admin* role.

```
<field name="UnitPrice" type="Decimal" default="(0)"
  label="Unit Price" writeRoles="Admin"/>
```

The sample application generated with [Aquarium Express](#) project is relying on *Windows* authentication by default. Role *Admin* is not defined in a typical *Windows* configuration and this will cause the *UnitPrice* to be displayed as read-only in all grid and form views of *Products* data controller.

Most *Windows* user accounts belong to *Users* group. If you replace *Admin* with *Users* then an editable version of *UnitPrice* field is presented.

The same role level access is available if you switch your application to *forms authentication* or create a custom role and/or user manager.

You can read more about field-level security in [Data Aquarium Framework](#) applications in the post [Using "roles" And "writeRoles" Attributes With Fields And Actions](#).