**Custom-update:**

<div class="row p-4 c-controls pt-0">

<div class="col-md-2 ps-3">

<label>Product category</label>

</div>

<div class="col-md-4 pe-4">

<**ComboBox** @bind-Value="searchProduct.ProductCategoryId" **TValue**="int" **TEntity**="ProductCategory"

**TextExpression**="t => t.Title" />

</div>

<div class="col-md-2 ps-4">

<label>Status</label>

</div>

<div class="col-md-4 pe-3">

<**DropdownList** @bind-Value="searchProduct.ActiveType" />

</div>

<div class="col-md-12 pt-4">

<**DataGrid** @ref="grid" **TEntity**="Product" **Search**="searchProduct" **TableHeight**="400" **HideInsertIcon**>

<**Columns**>

<**GridColumn** **Field**="t => t.ProductCategory.Title" **Title**="Product category" />

<**GridColumn** **Field**="t => t.Title" />

<**GridColumn** **Field**="t => t.Price.ToString()" style="width:150px;">

<**Template**>

<**NumericTextBox** **Focused**="selectedId == context.Data.Id && colNumber == 1" **TValue**="int"

**Value**="context.Data.Price" **ValueChanged**="value => context.Data.Price = value"

@onkeyup="async e => await UpdatePrice(e, context.Data.Id, context.Data.Price)" />

</**Template**>

</**GridColumn**>

<**GridColumn** **Field**="t => t.TakeoutPrice.ToString()" style="width:150px;">

<**Template**>

<**NumericTextBox** **Focused**="selectedId == context.Data.Id && colNumber == 2" **TValue**="int"

**Value**="context.Data.TakeoutPrice" **ValueChanged**="value => context.Data.TakeoutPrice = value"

@onkeyup="async e => await UpdateTakeoutPrice(e, context.Data.Id, context.Data.TakeoutPrice)" />

</**Template**>

</**GridColumn**>

</**Columns**>

</**DataGrid**>

</div>

</div>

@code

{

int? selectedId;

int? colNumber;

Product searchProduct = new Product();

DataGrid<Product> grid;

async Task UpdatePrice(KeyboardEventArgs e, int productId, int price)

{

if (e.Code == "Enter" || e.Code == "NumpadEnter")

{

using var scope = CreateScope();

var productService = new ProductService(scope.ServiceProvider);

await productService.UpdatePrice(productId, price);

await productService.SaveChangesAsync();

selectedId = productId;

colNumber = 2;

}

}

async Task UpdateTakeoutPrice(KeyboardEventArgs e, int productId, int takeoutPrice)

{

if (e.Code == "Enter" || e.Code == "NumpadEnter")

{

using var scope = CreateScope();

var productService = new ProductService(scope.ServiceProvider);

await productService.UpdateTakeoutPrice(productId, takeoutPrice);

await productService.SaveChangesAsync();

await grid.SelectNextRow();

if (grid.SelectedRowId.HasValue)

{

selectedId = grid.SelectedRowId.Value;

colNumber = 1;

}

}

}

}