<div class="row p-3">

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

<div class="row">

<div class="col-md-2 text-center">

<a href="/Demo/Orders">

<span class="fa fa-first-order"></span>

</a>

</div>

<div class="col-md-10 text-center">

<h7 class="d-membership">Order</h7>

</div>

</div>

<hr />

<**CaspianForm** @ref="MasterForm" **Model**="UpsertData">

<**CaspianValidationValidator** **ValidatorType**="typeof(OrderService)" />

<div class="row c-controls ps-3 pe-3">

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

<label>Order date</label>

</div>

<div class="col-md-8">

<**DatePicker** @bind-Value="UpsertData.Date" />

</div>

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

<label>Order type</label>

</div>

<div class="col-md-8">

<**DropdownList** @bind-Value="UpsertData.OrderType" />

</div>

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

<label>Customer</label>

</div>

<div class="col-md-8">

<**ComboBox** **TEntity**="Customer" **TValue**="int?"

@bind-Value="UpsertData.CustomerId"

**TextExpression**="@(t => t.CustomerType == CustomerType.Real

? t.FName + " " + t.LName : t.CompanyName)" />

</div>

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

<label>Delivery</label>

</div>

<div class="col-md-8">

<**ComboBox** **TValue**="int?" **TEntity**="Delivery"

@bind-Value="UpsertData.DeliveryId" **TextExpression**="@(t =>

t.FName + " " + t.LName + "-" + t.Code)" />

</div>

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

<label>Description</label>

</div>

<div class="col-md-8">

<**StringTextBox** @bind-Value="UpsertData.Description" **MultiLine**

**Rows**="4" />

</div>

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

<**FormCommandButton** />

</div>

</div>

</**CaspianForm**>

</div>

<div class="col-md-8 ps-3 text-center">

<h7 class="d-membership">Order Details</h7>

<hr />

<div class="col-md-12 ps-3 pe-3">

<**DataGrid** @ref="Grid" **TEntity**="OrderDeatil" **TableHeight**="350"

**Inline** **Batch** **ConditionExpr**="t => t.OrderId == MasterId">

<**Columns** **Context**="con">

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

<**EditTemplate**>

<**AutoComplete** **TValue**="int" **TEntity**="Product"

@bind-Value="con.Data.ProductId" **HideHeader**

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

**Title**="Product" **OnChange**="async () => await

SetPrice(con.Data)">

<**ProductLookupPage** **TValue**="int" />

</**AutoComplete**>

</**EditTemplate**>

</**GridColumn**>

<**GridColumn** **Field**="t => t.Price.Seprate3Digit()">

<**EditTemplate**>

<span>@(con.Data.Price.Seprate3Digit())</span>

</**EditTemplate**>

</**GridColumn**>

<**GridColumn** **Field**="t => t.Count">

<**EditTemplate**>

<**NumericTextBox** @ref="txtPrice"

@bind-Value="con.Data.Count"

@onkeyup="@(async e => await UpsertOnEnter(e,

con.UpsertMode.Value))" />

</**EditTemplate**>

</**GridColumn**>

<**GridCommandColumns**/>

</**Columns**>

</**DataGrid**>

</div>

</div>

</div>

@code

{

NumericTextBox<int> txtPrice;

AutoComplete<Product, int> txtProduct;

protected override async Task OnMasterEntityCreatedAsync()

{

//in insert by default date of day set for order

if (UpsertData.Id == 0)

UpsertData.Date = DateTime.Now.Date;

await base.OnMasterEntityCreatedAsync();

}

async Task UpsertOnEnter(KeyboardEventArgs e, UpsertMode mode)

{

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

await Grid.ValidateUpsert(mode);

}

async Task SetPrice(OrderDeatil detail)

{

if (detail.ProductId > 0)

{

var scope = CreateScope();

var old = await new ProductService(scope.ServiceProvider)

.SingleAsync(detail.ProductId);

//Set price of product

detail.Price = old.Price;

//Set count = 1 by default

if (detail.Count == 0)

detail.Count = 1;

txtPrice.Focus();

}

}

}