Crude multi entity:

```html
<CrudComponent UpsertData="upsertCategory">
    <div class="row c-controls">
        <div class="col-md-3 ps-4">
            <label>Title</label>
        </div>
        <div class="col-md-6">
            <StringTextBox @bind-Value="SearchCategory.Title" Search />
        </div>
        <div class="d-source-btn col-md-3">
            <button @onclick="() => status = WindowStatus.Open"
                    class="c-btn c-primary">
                Source
                <i class="fa fa-code"></i>
            </button>
        </div>
    </div>
    <DataGrid TEntity="ProductCategory" Search="SearchCategory" TableHeight="206"
              OnRowSelect="ProductCategorySelected">
        <Columns>
            <GridColumn Field="t => t.Title" />
            <GridColumn Field="t => t.ActiveType.FaText()" />
            <GridCommandColumns />
        </Columns>
    </DataGrid>
    <Window Title="Product Category" Style="width:500px">
        <Content>
            <CaspianForm Model="upsertCategory">
                <CaspianValidationValidator
                        ValidatorType="typeof(ProductCategoryService)" />
                <div class="row c-controls">
                    <div class="col-md-4">
                        <label>Title</label>
                    </div>
                    <div class="col-md-8">
                        <StringTextBox @bind-Value="upsertCategory.Title" />
                    </div>
                    <div class="col-md-4">
                        <label>Status</label>
                    </div>
                    <div class="col-md-8">
                        <DropdownList @bind-Value="upsertCategory.ActiveType" />
                    </div>
                    <div class="col-md-4"></div>
                    <div class="col-md-8">
                        <WindowCommandButtons />
                    </div>
                </div>
            </CaspianForm>
        </Content>
    </Window>
</CrudComponent>

@if (productcategoryId.HasValue)
{
    <CrudComponent UpsertData="upsertProduct" Style="width:90%;margin:5px auto"
            BeforUpsert="ConfirmForPrice">
```

```html
<div class="c-continner-head">
    <span>@productTitle</span>
</div>
<div class="row c-controls">
    <div class="col-md-3 ps-4">
        <label>Title</label>
    </div>
    <div class="col-md-9 pe-4">
        <StringTextBox @bind-Value="SearchProduct.Title" Search />
    </div>
</div>
<DataGrid @ref="grid" TEntity="Product" Search="SearchProduct"
TableHeight="206" ConditionExpr="t => t.ProductCategoryId == productcategoryId">
    <Columns>
        <GridColumn Field="t => t.Title" />
        <GridColumn Field="t => t.Price.Seprate3Digit()" />
        <GridColumn Field="t => t.TakeoutPrice.Seprate3Digit()" />
        <GridColumn Field="t => t.ActiveType.FaText()" />
        <GridCommandColumns />
    </Columns>
</DataGrid>
<Window Title="Product" Style="width:500px" OnOpen="SetCategory">
    <Content>
        <CaspianForm Model="upsertProduct">
            <CaspianValidationValidator
                ValidatorType="typeof(ProductService)" />
            <div class="row c-controls">
                <div class="col-md-4">
                    <label>Title</label>
                </div>
                <div class="col-md-8">
                    <StringTextBox @bind-Value="upsertProduct.Title" />
                </div>
                <div class="col-md-4">
                    <label>Price</label>
                </div>
                <div class="col-md-8">
                    <NumericTextBox @bind-Value="upsertProduct.Price"
                        OnChange="SetTakeoutPrice" />
                </div>
                <div class="col-md-4">
                    <label>Take out Price</label>
                </div>
                <div class="col-md-8">
                    <NumericTextBox @bind-Value="upsertProduct.TakeoutPrice"
                        />
                </div>
                <div class="col-md-4">
                    <label>Meal</label>
                </div>
                <div class="col-md-8">
                    <MultiSelectBox @bind-Value="upsertProduct.Meal" />
                </div>
                <div class="col-md-4">
                    <label>Status</label>
                </div>
                <div class="col-md-8">
                    <DropdownList @bind-Value="upsertProduct.ActiveType" />
```

```razor
                        </div>
                        <div class="col-md-4">
                            <label>Discountable</label>
                        </div>
                        <div class="col-md-8">
                            <CheckBox @bind-Value="upsertProduct.Discountable" />
                        </div>
                        <div class="col-md-4"></div>
                        <div class="col-md-8 align-content-center">
                            <WindowCommandButtons />
                        </div>
                    </div>
                </CaspianForm>
            </Content>
        </Window>
    </CrudComponent>
}


@code
{
    string productTitle;
    DataGrid<Product> grid;
    ProductCategory upsertCategory = new ProductCategory();
    ProductCategory SearchCategory = new ProductCategory();

    Product upsertProduct = new Product();
    Product SearchProduct = new Product();

    int? productcategoryId;

    void SetCategory()
    {
        upsertProduct.ProductCategoryId = productcategoryId.Value;
    }

    void SetTakeoutPrice()
    {
        if (upsertProduct.TakeoutPrice == 0)
            upsertProduct.TakeoutPrice = upsertProduct.Price;
    }

    async void ProductCategorySelected(ProductCategory category)
    {
        productcategoryId = category.Id;
        productTitle = category.Title;
        await Task.Delay(1);
        grid.EnableLoading();
    }

    async Task<bool> ConfirmForPrice()
    {
        if (upsertProduct.Price == 0)
            return await Confirm("آیا با ثبت محصول با مبلغ صفر موافقید؟");
        return true;
    }
}
```