**Load on first*:***

<div class="col-md-6">

<div class="ps-3 pe-4 pb-2">

<**RadioList** @bind-Value="expanded" **OnChange**="ExpandedChanged">

<**RadioItems**>

<**RadioItem** **TValue**="bool" **Value**="false" **Title**="Collapsed" />

<**RadioItem** **TValue**="bool" **Value**="true" **Title**="Expanded" />

</**RadioItems**>

</**RadioList**>

</div>

<**CTreeView** **TEntity**="TreeViewItem" style="height:500px;" **Source**="loadOnFirstSource" />

</div>

@code

{

bool expanded = true;

IList<TreeViewItem> loadOnFirstSource;

protected override async Task OnInitializedAsync()

{

using var scope = CreateScope();

var query = new MenuCategoryService(scope.ServiceProvider).GetAll()

.Where(t => t.SubSystemKind == SubSystemKind.Demo);

/// load menus category and menus together

loadOnFirstSource = await query.Select(t => new TreeViewItem()

{

Text = t.Title,

Expanded = expanded,

Items = t.Menus.Select(u => new TreeViewItem() //Menus load as subtree

{

Text = u.Title,

Collabsable = false

}).ToList()

}).ToListAsync();

await base.OnInitializedAsync();

}

void ExpandedChanged()

{

foreach (var item in loadOnFirstSource)

item.Expanded = expanded;

}

}

**Load on expand*:***

<**CTreeView** **TEntity**="TreeViewItem" style="height:500px;" **Source**="loadOnExpandSource"

**OnExpanded**="async node => await OnExpanded(node)" />

@code

{

IList<TreeViewItem> loadOnExpandSource;

protected override async Task OnInitializedAsync()

{

using var scope = CreateScope();

var query = new MenuCategoryService(scope.ServiceProvider).GetAll()

.Where(t => t.SubSystemKind == SubSystemKind.Demo);

/// load menus category and on node expand load menus

loadOnExpandSource = await query.Select(t => new TreeViewItem()

{

Text = t.Title,

Value = t.Id.ToString()

}).ToListAsync();

await base.OnInitializedAsync();

}

async Task OnExpanded(TreeViewItem node)

{

using var scope = CreateScope();

var categoryId = Convert.ToInt32(node.Value);

node.Items = await new MenuService(scope.ServiceProvider).GetAll()

.Where(t => t.MenuCategoryId == categoryId).Select(t => new TreeViewItem()

{

Text = t.Title,

Collabsable = false

}).ToListAsync();

}

}