

MrmGrossMarginComponent

August 17, 2026

MrmGrossMarginComponent

Kind: Class

Source: Pams/Web/Pams.Web/Client/app/components/dashboard/dashboard-library/widgets/mrm-gross-margin/mrm-gross-margin.component.ts (line 84)

Part of: Pams

MrmGrossMarginComponent is an Angular dashboard widget that retrieves, initializes, displays, and saves gross-margin dashboard data. It responds to component lifecycle events, calculates totals by year or month, and translates month names for dashboard output.

Methods

Method	Signature	Returns
GetAuthorizationAccess	GetAuthorizationAccess(url: string)	void
ngOnInit	ngOnInit()	void
ngOnChanges	ngOnChanges()	void
ngOnDestroy	ngOnDestroy()	void
getDashboardData	getDashboardData()	void
initializingData	initializingData(value: BranchData[])	void
saveDashboardData	saveDashboardData()	void
calculateSumByYear	calculateSumByYear()	void
calculateSumByMonth	calculateSumByMonth(monthIndex: number)	void
translateMonthName	translateMonthName(month: MonthData)	string
getpriceFormatted	getpriceFormatted(x: undefined)	string
onResize	onResize()	void

Method	Signature	Returns
setChaching	setChaching()	void
getChaching	getChaching()	void
switchWithTaxableHandler	switchWithTaxableHandler()	void
switchChartTypeHandler	switchChartTypeHandler()	void
switchBasedOnTendencyHandler	switchBasedOnTendencyHandler()	void
comparingWithPrevMonth	comparingWithPrevMonth(month: MonthData, bIndex: number, monthIndex: number, propertyName: string)	string
comparingBookingWithPrevSumm	comparingBookingWithPrevSumm(month: MonthsSum, index: number, propertyName: string)	string
isCurrentMonth	isCurrentMonth(month: MonthData)	boolean
getpriceForecastAmount	getpriceForecastAmount(month: MonthData)	number
expectedCurrencyChange	expectedCurrencyChange(e: undefined)	void
selectText	selectText(e: undefined)	void
mouseoverRowSpan	mouseoverRowSpan(e: undefined, bIndex: number)	void
mouseoutRowSpan	mouseoutRowSpan(e: undefined, bIndex: number)	void
editSelectedMonthPrepare	editSelectedMonthPrepare(month: MonthData, supplierId: number)	void
sorting	sorting(propertyName: string, propertyType: string, monthIndex: number)	void
syncScroll	syncScroll(e: undefined)	void
openPopup	openPopup(n: string, t: string, i: string, btnTxt: string)	void
savePopup	savePopup()	void
ClosewithDonotSave	ClosewithDonotSave()	void
OnOpenedPopup	OnOpenedPopup()	void
closePopup	closePopup()	void

Properties

Property	Type
sharedPopup	SharedPopupComponent
scrollOne	ElementRef
_widget	Widget
_widgetType	string
withoutOptionBar	boolean
editMode	boolean
bookingOrSales	number
withNominal	boolean
withTaxable	boolean
selectedTeam	number
forecastRuleId	number
allowedtoEdit	boolean
allowedtoEnableEditing	boolean
widgetEditRequest	any
widgetEditEnd	any
forecastRules	any[]
selectedYear	number
numYearsDatasource	any[]
todayWidgetsEnum	any
cachingWidgetData	CachingWidgetData
data	BranchData[]
originalData	BranchData[]
selectedData	BranchData[]
singleMonths	MonthData
branchCurrency	any
basedOnTendency	boolean
subscription	Subscription
SeriseRange	any[]

Property	Type
echartsIntance	any
totalBookingMonths	MonthsSum[]
tempMonthSum	MonthsSum
totalTargets	number
yearBooking	number
totalForecasts	number
tempTotalTarget	TotalTargets
allGroupsTotalSales	number
totalSalesAfterCommission	number
totalSalesSummary	number
totalDirectService	number
totalDirectSalesWithoutScrap	number
totalDirectSalesWithScrap	number
totalTotalDirectSalesWithScrap	number
totalCommission	number
totalCumulativeSales	number
popup	any
searchText	string
fixedColumnLeftNumOne	number
fixedColumnLeftNumTwo	number
fixedColumnLeftNumThree	number
sortingDirection	number

Where it refuses work

- `MrmGrossMarginComponent` stops the work with an early return when `a[propertyName] < b[propertyName]` , in 2 places.
- `MrmGrossMarginComponent` stops the work with an early return when `a[propertyName] > b[propertyName]` , in 2 places.

- `MrmGrossMarginComponent` stops the work with an early return when `month.MonthId == 12` .
- `MrmGrossMarginComponent` stops the work with an early return when `month[propertyName] < this.data[bIndex].MrmCogsGroups[monthIndex - 1][propertyName]` .
- `MrmGrossMarginComponent` stops the work with an early return when `month[propertyName] < this.totalBookingMonths[index - 1][propertyName]` .
- `MrmGrossMarginComponent` stops the work with an early return when `this.allowedtoEnableEditing` .

Diagram

```
graph LR
  A[Angular lifecycle] --> B[MrmGrossMarginComponent]
  B --> C[GetAuthorizationAccess]
  B --> D[getDashboardData]
  D --> E[initializingData]
  E --> F[calculateSumByYear]
  E --> G[calculateSumByMonth]
  G --> H[translateMonthName]
  B --> I[saveDashboardData]
  B --> J[ngOnDestroy]
```

Usage

```
import { Component, ViewChild } from '@angular/core';
import { MrmGrossMarginComponent } from './mrm-gross-margin.component';

@Component({
  selector: 'app-dashboard-host',
  template: `
    <app-mrm-gross-margin></app-mrm-gross-margin>
  `,
})
export class DashboardHostComponent {
  @ViewChild(MrmGrossMarginComponent)
  grossMarginWidget?: MrmGrossMarginComponent;

  refreshGrossMargin(): void {
    this.grossMarginWidget?.GetAuthorizationAccess();
    this.grossMarginWidget?.getDashboardData();
  }

  saveGrossMarginSettings(): void {
    this.grossMarginWidget?.saveDashboardData();
  }
}
```

```
}  
}
```

AI Coding Instructions

- Keep data loading inside `getDashboardData()` and initialization logic inside `initializingData()` .
- Run authorization checks through `GetAuthorizationAccess()` before requesting or saving dashboard data.
- Preserve lifecycle behavior in `ngOnInit()` , `ngOnChanges()` , and `ngOnDestroy()` when adding subscriptions or input-driven updates.
- Use `calculateSumByYear()` and `calculateSumByMonth()` for aggregate values instead of duplicating sum logic in the template.
- Keep month display formatting in `translateMonthName()` so dashboard labels use one translation path.

Used by

1 reference from 1 file. Each is a place in this repository where the symbol is actually used — go read one rather than trusting an example.

Imported by (1)

- `TranslateHandler` — `Pams/Web/Pams.Web/Client/app/app.module.ts` :1