

MrmNetProfitComponent

August 18, 2026

MrmNetProfitComponent

Kind: Class

Source: Frontend/src/app/components/dashboard/dashboard-library/widgets/mrm-net-profit/mrm-net-profit.component.ts (line 94)

Part of: Frontend

MrmNetProfitComponent is a dashboard widget component for presenting net-profit data. It checks feature subscription access, loads and saves dashboard data, calculates month and year totals, and formats month and price display values.

Extends: BaseWidgetComponent

Methods

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

Method	Signature	Returns
setChaching	setChaching()	void
getChaching	getChaching()	void
switchChartTypeHandler	switchChartTypeHandler()	void
switchBasedOnTendencyHandler	switchBasedOnTendencyHandler()	void
comparingWithPrevMonth	comparingWithPrevMonth(month: BranchData, bIndex: 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
editSelectedMonthPrepare	editSelectedMonthPrepare(month: MonthData, supplierId: number)	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
_widget	Widget
_widgetType	string
yearId	number
periodId	number
countOfEditablePrevMonths	number

Property	Type
withoutOptionBar	boolean
editMode	boolean
bookingOrSales	number
withNominal	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
SeriseRange	any[]
echartsIntance	any
totalBookingMonths	MonthsSum[]
tempMonthSum	MonthsSum
totalTargets	number
yearBooking	number
totalForecasts	number

Property	Type
tempTotalTarget	TotalTargets
totalOthers	number
totalDepreciationAmount	number
totalTotalCogsAmount	number
totalGrossMarginAmount	number
totalTotalCogsAchieved	number
totalTotalNetProfitAmount	number
totalTotalNetProfitAchieved	number
totalTotalDepreciationAmount	number
totalTotalDepreciationAchieved	number
totalTotalFORXAmount	number
totalTotalFORXAchieved	number
totalTotalEBITAmount	number
totalIncomeTAXAmount	number
totalProfitAmountBeforeTax	number
totalTotalEBITDAAmount	number

Where it refuses work

- `MrmNetProfitComponent` stops the work with an early return when `this.dashboardLibraryService.checkFeatureSubscription(this.widget.chartEnumId)` .
- `MrmNetProfitComponent` stops the work with an early return when `month.MonthId == 12` .
- `MrmNetProfitComponent` stops the work with an early return when `month[propertyName] < this.data[bIndex - 1][propertyName]` .
- `MrmNetProfitComponent` stops the work with an early return when `month[propertyName] < this.totalBookingMonths[index - 1][propertyName]` .
- `MrmNetProfitComponent` stops the work with an early return when `AppSettings.calculateMonthsBeforeToday(month.YearId, month.MonthId - 1) == 0 || AppSetting...` .

Diagram

```
graph LR
  A[Angular lifecycle] --> B[ngOnInit]
  A --> C[ngOnChanges]
  B --> D[checkFeatureSubscription]
  C --> E[getDashboardData]
  D --> E
  E --> F[initializingData]
  F --> G[calculateSumByYear]
  F --> H[calculateSumByMonth]
  F --> I[translateMonthName]
  F --> J[getpriceFormatted]
  F --> K[saveDashboardData]
```

Usage

```
import { MrmNetProfitComponent } from './mrm-net-profit.component';

// Obtain the component instance through Angular component composition,
// a test fixture, or another supported application integration point.
const netProfitWidget = {} as MrmNetProfitComponent;

if (netProfitWidget.checkFeatureSubscription()) {
  netProfitWidget.getDashboardData();

  netProfitWidget.calculateSumByYear();
  netProfitWidget.calculateSumByMonth();

  const monthLabel = netProfitWidget.translateMonthName();
  const formattedPrice = netProfitWidget.getpriceFormatted();

  console.log({ monthLabel, formattedPrice });
}
```

AI Coding Instructions

- Check `checkFeatureSubscription()` before requesting or displaying net-profit dashboard data.
- Keep data loading in `getDashboardData()` and initialization in `initializingData()` so lifecycle handlers remain focused.
- Preserve the existing `initializingData()` method name unless all callers are updated together.
- Recalculate totals through `calculateSumByYear()` and `calculateSumByMonth()` after changing the underlying dashboard data.

- Use `translateMonthName()` and `getpriceFormatted()` for display values instead of duplicating formatting logic.

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)

- `DashboardWidgetsModule` — `Frontend/src/app/components/dashboard/dashboard-widgets.module.ts :1`