



PAMS

MrmDataTableComponent

Kind: Class

VERSION
4

UPDATED
11 August 2026

DOCUMENT
reference-subsystem-pams/class-mrmdatatablecomponent

CONTENTS

1	Methods	2	5	Usage	5
2	Properties	3	6	AI Coding Instructions	6
3	Where it refuses work	4	7	Used by	6
4	Diagram	5			

Kind: Class

Source: `Pams/Web/Pams.Web/Client/app/components/mrm/mrm-booking/mrm-data-table/mrm-data-table.component.ts` (line 45)

Part of: [Pams](#)

`MrmDataTableComponent` manages data displayed in the MRM booking data table. It retrieves dashboard data, initializes table state, calculates performance values, formats month and price output, handles resize events, and stores cached data during its Angular lifecycle.

Implements: `OnInit`, `OnChanges`, `OnDestroy`

1 Methods

Method	Signature	Returns
<code>ngOnInit</code>	<code>ngOnInit()</code>	<code>void</code>
<code>ngOnChanges</code>	<code>ngOnChanges()</code>	<code>void</code>
<code>ngOnDestroy</code>	<code>ngOnDestroy()</code>	<code>void</code>
<code>getDashboardData</code>	<code>getDashboardData()</code>	<code>void</code>
<code>initializingData</code>	<code>initializingData(value: BranchData[])</code>	<code>void</code>
<code>calculatePerformancePer</code>	<code>calculatePerformancePer(month: MonthData)</code>	<code>number</code>
<code>translateMonthName</code>	<code>translateMonthName(month: MonthData)</code>	<code>string</code>
<code>getpriceFormatted</code>	<code>getpriceFormatted(x: undefined)</code>	<code>string</code>
<code>onResize</code>	<code>onResize()</code>	<code>void</code>
<code>setChaching</code>	<code>setChaching()</code>	<code>void</code>
<code>getChaching</code>	<code>getChaching()</code>	<code>void</code>
<code>onValueChanged</code>	<code>onValueChanged(e: undefined)</code>	<code>void</code>
<code>onChartInit</code>	<code>onChartInit(e: any, index: number)</code>	<code>void</code>
<code>switchChartTypeHandler</code>	<code>switchChartTypeHandler()</code>	<code>void</code>
<code>switchBasedOnTendencyHandler</code>	<code>switchBasedOnTendencyHandler()</code>	<code>void</code>

setValue	setValue(value: number)	void
setForecastValue	setForecastValue(value: number)	void
setMarginAvarage	setMarginAvarage(value: number)	void
isForecast	isForecast(month: MonthData)	boolean
isAchievedOrCurrent	isAchievedOrCurrent(month: MonthData)	boolean
getpriceForecastAmount	getpriceForecastAmount(month: MonthData)	number
expectedCurrencyChange	expectedCurrencyChange(e: undefined)	void

2 Properties

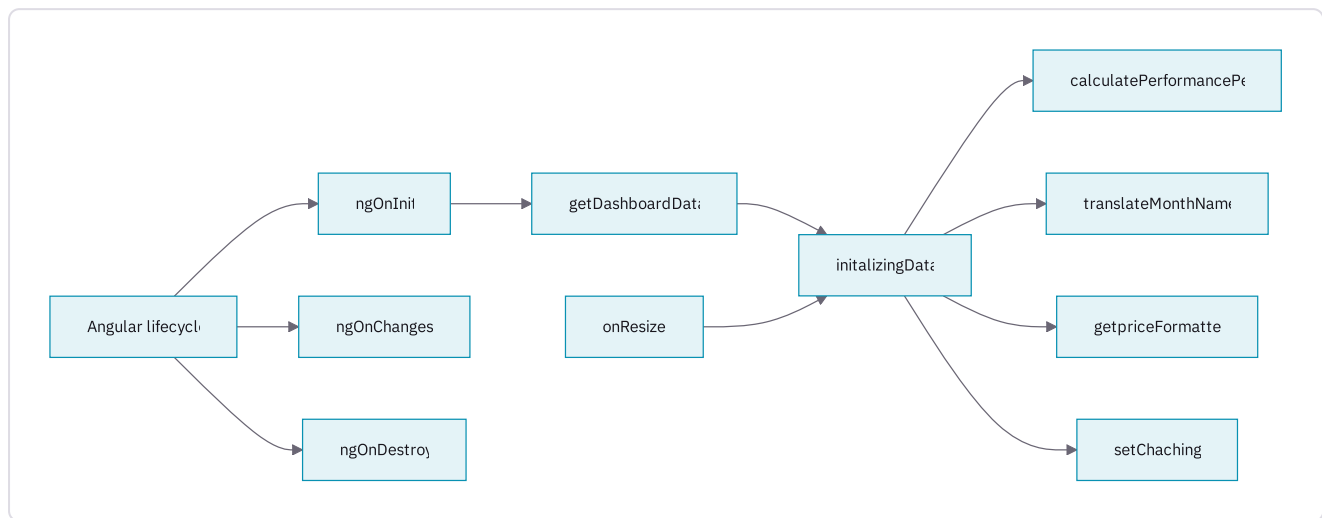
Property	Type
triangleNeedle	DxCircularGaugeComponent
_widget	Widget
_widgetType	string
withoutOptionBar	boolean
bookingOrSales	number
withNominal	boolean
selectedTeam	number
forecastRuleId	number
forecastRules	any[]
selectedYear	number
numYearsDatasource	any[]
todayWidgetsEnum	any
cachingWidgetData	CachingWidgetData
data	BranchData[]

<code>originalData</code>	<code>BranchData[]</code>
<code>singleMonths</code>	<code>MonthData</code>
<code>branchCurrency</code>	<code>any</code>
<code>basedOnTendency</code>	<code>boolean</code>
<code>subscription</code>	<code>Subscription</code>
<code>SeriseRange</code>	<code>any[]</code>
<code>echartsIntance</code>	<code>any</code>
<code>totalBookingMonths</code>	<code>TotalBookingMonths[]</code>
<code>tempTotalMonth</code>	<code>TotalBookingMonths</code>
<code>totalTargets</code>	<code>number</code>
<code>yearBooking</code>	<code>number</code>
<code>totalForecasts</code>	<code>number</code>
<code>tempTotalTarget</code>	<code>TotalTargets</code>

3 Where it refuses work

- `MrmDataTableComponent` stops the work with an early return when `month.Month == 12`.
- `MrmDataTableComponent` stops the work with an early return when `month.Month >= new Date().getMonth() + 1`.
- `MrmDataTableComponent` stops the work with an early return when `month.Month <= new Date().getMonth() + 1`.

4 Diagram



5 Usage

```

import { ComponentFixture, TestBed } from '@angular/core/testing';
import { MrmDataTableComponent } from './mrm-data-table.component';

describe('MrmDataTableComponent', () => {
  let fixture: ComponentFixture<MrmDataTableComponent>;
  let component: MrmDataTableComponent;

  beforeEach(async () => {
    await TestBed.configureTestingModule({
      declarations: [MrmDataTableComponent],
    }).compileComponents();

    fixture = TestBed.createComponent(MrmDataTableComponent);
    component = fixture.componentInstance;
    fixture.detectChanges();
  });

  it('loads and formats dashboard table data', () => {
    component.getDashboardData();

    const performance = component.calculatePerformancePer();
    const monthName = component.translateMonthName();
    const formattedPrice = component.getpriceFormatted();

    expect(performance).toBeDefined();
    expect(monthName).toBeDefined();
    expect(formattedPrice).toBeDefined();
  });
});

```

6 AI Coding Instructions

- Keep data loading and state preparation coordinated through `getDashboardData()` and `initializingData()`.
- Preserve the existing `initializingData` method name when calling or refactoring it, even though its spelling differs from `initializing`.
- Recalculate display values through `calculatePerformancePer()`, `translateMonthName()`, and `getpriceFormatted()` rather than duplicating formatting logic in templates.
- Handle viewport-dependent table updates through `onResize()` and clean up subscriptions or listeners in `ngOnDestroy()`.
- Review cache behavior in `setChaching()` when changing dashboard data sources or table refresh flows.

7 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