

MrmBookingCityComponent

August 18, 2026

MrmBookingCityComponent

Kind: Class

Source: Pams/Web/Pams.Web/Client/app/components/dashboard/dashboard-library/widgets/mrm-booking-city/mrm-booking-city.component.ts (line 118)

Part of: [Pams](#)

`MrmBookingCityComponent` is an Angular dashboard widget for loading, initializing, and saving booking-city dashboard data. It manages authorization checks, lifecycle updates, period-based totals by year and month, and month-name translation for display.

Methods

Method	Signature	Returns
<code>GetAuthorizationAccess</code>	<code>GetAuthorizationAccess(url: string)</code>	<code>void</code>
<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>saveDashboardData</code>	<code>saveDashboardData()</code>	<code>void</code>
<code>calculateSumByYear</code>	<code>calculateSumByYear()</code>	<code>void</code>
<code>calculateSumByMonth</code>	<code>calculateSumByMonth(monthIndex: number)</code>	<code>void</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>

Method	Signature	Returns
setChaching	setChaching()	void
getChaching	getChaching()	void
switchChartTypeHandler	switchChartTypeHandler()	void
switchBasedOnTendencyHandler	switchBasedOnTendencyHandler()	void
comparingWithPrevMonth	comparingWithPrevMonth(month: BranchData, bIndex: number, index: number, propertyName: string)	string
comparingMonthsWithPrevSumm	comparingMonthsWithPrevSumm(month: TotalBookingMonths, 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
syncScroll	syncScroll(e: undefined)	void
editSelectedMonthPrepare	editSelectedMonthPrepare(month: MonthData, supplierId: number)	void
sorting	sorting(propertyName: string, propertyType: string, monthIndex: 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
scrollOne	ElementRef
scrollTwo	ElementRef
scrollThree	ElementRef
_widget	Widget
_widgetType	string
withoutOptionBar	boolean
editMode	boolean
bookingOrSales	number
withNominal	boolean
selectedTeam	number
forecastRuleId	number
data	BranchData[]
allowedtoEdit	boolean
allowedtoEnableEdting	boolean
widgetEditRequest	any
widgetEditEnd	any
forecastRules	any[]
selectedYear	number
numYearsDatasource	any[]
todayWidgetsEnum	any
cachingWidgetData	CachingWidgetData
originalData	BranchData[]
selectedData	BranchData[]
singleMonths	MonthData
branchCurrency	any
basedOnTendency	boolean
subscription	Subscription

Property	Type
SerializeRange	any[]
echartsIntance	any
totalBookingMonths	TotalBookingMonths[]
tempTotalMonth	TotalBookingMonths
popup	any
searchText	string
sortingDirection	number

Where it refuses work

- `MrmBookingCityComponent` stops the work with an early return when `a[propertyName] < b[propertyName]` , in 2 places.
- `MrmBookingCityComponent` stops the work with an early return when `a[propertyName] > b[propertyName]` , in 2 places.
- `MrmBookingCityComponent` stops the work with an early return when `month.MonthId == 12` .
- `MrmBookingCityComponent` stops the work with an early return when `month[propertyName] < this.data[bIndex].MrmBooking[index - 1][propertyName]` .
- `MrmBookingCityComponent` stops the work with an early return when `month[propertyName] < this.totalBookingMonths[index - 1][propertyName]` .
- `MrmBookingCityComponent` stops the work with an early return when `this.allowedtoEnableEdting` .

Diagram

```
graph LR
  A[Angular Lifecycle] --> B[ngOnInit]
  A --> C[ngOnChanges]
  A --> D[ngOnDestroy]

  B --> E[GetAuthorizationAccess]
  E --> F[getDashboardData]
  F --> G[initializingData]

  C --> F
  G --> H[calculateSumByYear]
  G --> I[calculateSumByMonth]
```

```
I --> J[translateMonthName]
```

```
G --> K[saveDashboardData]
```

Usage

```
import { ComponentFixture, TestBed } from '@angular/core/testing';
import { MrmBookingCityComponent } from './mrm-booking-city.component';

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

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

    fixture = TestBed.createComponent(MrmBookingCityComponent);
    component = fixture.componentInstance;

    fixture.detectChanges();
  });

  it('loads dashboard data during initialization', () => {
    component.ngOnInit();

    component.getDashboardData();
    component.calculateSumByYear();
    component.calculateSumByMonth();
  });
});
```

AI Coding Instructions

- Keep data loading inside the existing authorization and lifecycle flow; call `GetAuthorizationAccess()` before requesting dashboard data when access must be verified.
- Treat `ngOnChanges()` as a refresh path when component inputs change, and avoid duplicating initialization logic outside `initializingData()`.
- Keep yearly and monthly aggregation logic in `calculateSumByYear()` and `calculateSumByMonth()` so displayed totals remain derived from the same data source.

- Use `translateMonthName()` for month labels instead of hardcoding display text, especially when adding localized dashboard output.
- Release subscriptions, timers, or other managed resources in `ngOnDestroy()` when extending the component.

Used by

3 references from 3 files. Each is a place in this repository where the symbol is actually used — go read one rather than trusting an example.

Imported by (3)

- `TranslateHandler` — `Pams/Web/Pams.Web/Client/app/app.module.ts :1`
- `DashboardTabComponent` — `Pams/Web/Pams.Web/Client/app/components/dashboard/dashboard-library/dashboard-tab/dashboard-tab.component.ts :1`
- `ReportsTabComponent` — `Pams/Web/Pams.Web/Client/app/components/dashboard/reports-library/reports-tab/reports-tab.component.ts :1`