

WidgetBookingVSTargetTodayComponent

August 17, 2026

WidgetBookingVSTargetTodayComponent

Kind: Class

Source: Pams/Web/Pams.Web/Client/app/components/dashboard/dashboard-library/widgets/widget-booking-vstarget-today/widget-booking-vstarget-today.component.ts (line 21)

Part of: [Pams](#)

`WidgetBookingVSTargetTodayComponent` renders the dashboard widget that compares today's bookings with target values. It loads dashboard data, initializes chart state, handles planned-visit chart interactions, and releases resources when the component is destroyed.

Implements: `OnInit`

Methods

Method	Signature	Returns
<code>getDashboardData</code>	<code>getDashboardData()</code>	<code>void</code>
<code>insializeWidget</code>	<code>insializeWidget(value: any[])</code>	<code>void</code>
<code>sortingByIndex</code>	<code>sortingByIndex(index: number)</code>	<code>void</code>
<code>ngOnInit</code>	<code>ngOnInit()</code>	<code>void</code>
<code>onChartInit</code>	<code>onChartInit(e: any)</code>	<code>void</code>
<code>onPlannedVisitsChartClick</code>	<code>onPlannedVisitsChartClick(e: undefined)</code>	<code>void</code>
<code>ngOnDestroy</code>	<code>ngOnDestroy()</code>	<code>void</code>

Properties

Property	Type
<code>_widget</code>	<code>Widget</code>

Property	Type
todayWidgetsEnum	any
subscription	Subscription
sortingByEnum	any
sortingDirectionEnum	any
requestSortingDirection	number
sortingArray	SortingArray[]
propertyName	string
data	any[]
originalData	any[]
initOpts	any
series	any[]
headerData	any
echartsIntance	any
options	any

Where it refuses work

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

Diagram

```
graph LR
  Init[ngOnInit] --> Setup[insializeWidget]
  Setup --> Data[getDashboardData]
  Data --> Sort[sortingByIndex]
  Sort --> Chart[onChartInit]
  Chart --> Click[onPlannedVisitsChartClick]
  Destroy[ngOnDestroy] --> Cleanup[Release subscriptions and chart resources]
```

Usage

```

import { Component, ViewChild } from '@angular/core';
import { WidgetBookingVSTargetTodayComponent } from './widget-booking-vstarget-today.component'

@Component({
  selector: 'app-dashboard-host',
  template: `
    <app-widget-booking-vstarget-today
      #bookingTargetWidget>
    </app-widget-booking-vstarget-today>

    <button type="button" (click)="refreshWidget()">
      Refresh
    </button>
  `
})
export class DashboardHostComponent {
  @ViewChild('bookingTargetWidget')
  bookingTargetWidget?: WidgetBookingVSTargetTodayComponent;

  refreshWidget(): void {
    this.bookingTargetWidget?.getDashboardData();
  }
}

```

AI Coding Instructions

- Keep data loading inside `getDashboardData()` so widget refresh behavior remains in one place.
- Preserve the initialization flow from `ngOnInit()` through `insializeWidget()` before creating or updating chart state.
- Apply `sortByIndex()` before binding data to the chart when chart ordering depends on source indexes.
- Route planned-visit chart events through `onPlannedVisitsChartClick()` rather than adding click handling in parent dashboard components.
- Clean up subscriptions, chart instances, and event handlers in `ngOnDestroy()`.

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

