

WidgetTopOffersByTeamsLastNyearsComponent

August 18, 2026

WidgetTopOffersByTeamsLastNyearsComponent

Kind: Class

Source: Frontend/src/app/components/dashboard/dashboard-library/widgets/widget-top-offers-by-teams-last-nyears/widget-top-offers-by-teams-last-nyears.component.ts (line 21)

Part of: Frontend

WidgetTopOffersByTeamsLastNyearsComponent is an Angular dashboard widget that retrieves and displays top offers grouped by team across a recent-year range. It checks feature subscription access, manages chart initialization and updates, and handles chart interactions that open related planned-visit data.

Extends: BaseWidgetComponent

Implements: OnInit

Methods

Method	Signature	Returns
checkFeatureSubscription	checkFeatureSubscription()	boolean
viewData	viewData()	void
initializeChart	initializeChart(seriesData: undefined, headerData: undefined)	void
getDashboardData	getDashboardData()	void
afterResponse	afterResponse(Data: any[])	void
ngOnChanges	ngOnChanges()	void
ngOnInit	ngOnInit()	void
setYearColor	setYearColor(year: number)	string
onChartInit	onChartInit(e: any)	void

Method	Signature	Returns
onPlannedVisitsChartClick	onPlannedVisitsChartClick(e: undefined)	void
setChaching	setChaching()	void
getChaching	getChaching()	void
onValueChanged	onValueChanged(e: undefined)	void
switchChartTypeHandler	switchChartTypeHandler()	void
switchChartRefernceCommissionHandler	switchChartRefernceCommissionHandler()	void
toggoleOnlyCurrentOffers	toggoleOnlyCurrentOffers()	void
toggoleBudgetOffers	toggoleBudgetOffers()	void
toggoleFirmOffers	toggoleFirmOffers()	void
switchFirmOffers	switchFirmOffers()	void
switchBudgetOffers	switchBudgetOffers()	void
nextPage	nextPage()	void
prevPage	prevPage()	void

Properties

Property	Type
_widget	Widget
meAsVendor	boolean
isRefernceCommission	boolean
withoutOptionBar	boolean
onlyCurrentOffers	boolean
numYears	number
_FirmOffers	boolean
_BudgetOffers	boolean
numYearsDatasource	any[]
todayWidgetsEnum	any
branchName	string

Property	Type
maxValueForYaxis	number
initOpts	any
series	any[]
header	any[]
echartsIntance	any
cachingWidgetData	CachingWidgetData
originalDatasource	WidgetTopOffers[]
datasource	WidgetTopOffers[]
originalHeader	any[]
options	any
skip	number
currenFeature	number
seriesData	any
headerData	any[]
showDataOnChart	boolean

Where it refuses work

- WidgetTopOffersByTeamsLastNyearsComponent stops the work with an early return when `this.dashboardLibraryService.checkFeatureSubscription(this.widget.chartEnumId)` .
- WidgetTopOffersByTeamsLastNyearsComponent stops the work with an early return when `this.showDataOnChart` .

Diagram

```
graph LR
    Lifecycle[Angular lifecycle] --> Init[ngOnInit]
    Lifecycle --> Changes[ngOnChanges]

    Init --> Subscription[checkFeatureSubscription]
    Subscription --> Data[getDashboardData]
    Changes --> Data

    Data --> Response[afterResponse]
    Response --> Chart[initializeChart]
```

```

Chart --> ChartInit[onChartInit]
Chart --> Color[setYearColor]

ChartClick[Chart click] --> PlannedVisits[onPlannedVisitsChartClick]
PlannedVisits --> ViewData[viewData]

```

Usage

```

import { TestBed } from '@angular/core/testing';
import { WidgetTopOffersByTeamsLastNyearsComponent } from './widget-top-offers-by-teams-last-ny

// Configure the test module with the widget and its required dependencies first.
const fixture = TestBed.createComponent(
  WidgetTopOffersByTeamsLastNyearsComponent
);

fixture.detectChanges();

const widget = fixture.componentInstance;

if (widget.checkFeatureSubscription()) {
  widget.getDashboardData();
}

```

AI Coding Instructions

- Keep data loading behind `checkFeatureSubscription()` so unavailable features do not request dashboard data.
- Let Angular invoke `ngOnInit()` and `ngOnChanges()` during normal rendering; call `getDashboardData()` directly only for explicit refresh flows.
- Initialize or update chart state through `afterResponse()` and `initializeChart()` after dashboard data is available.
- Keep chart event handling in `onChartInit()` and route planned-visit interactions through `onPlannedVisitsChartClick()`.
- Use `setYearColor()` when adding chart series so year-based color selection remains consistent.

Used by

2 references from 2 files. Each is a place in this repository where the symbol is actually used — go read one rather than trusting an example.

Imported by (2)

- `WidgetTopOffersByVariantsComponent` — `Frontend/src/app/components/dashboard/dashboard-library/widgets/widget-top-offers-by-variants/widget-top-offers-by-variants.component.ts :1`
- `DashboardWidgetsModule` — `Frontend/src/app/components/dashboard/dashboard-widgets.module.ts :1`