

WidgetBranchTendancyComponent

August 17, 2026

WidgetBranchTendancyComponent

Kind: Class

Source: Frontend/src/app/components/dashboard/dashboard-library/widgets/widget-branch-tendancy/widget-branch-tendancy.component.ts (line 17)

Part of: Frontend

WidgetBranchTendancyComponent is a dashboard widget component that displays branch tendency data in a chart-oriented view. It checks feature subscription access, loads dashboard data, initializes chart state, and derives display values and colors.

Extends: BaseWidgetComponent

Implements: OnInit

Methods

Method	Signature	Returns
getDashboardData	getDashboardData()	void
checkFeatureSubscription	checkFeatureSubscription()	boolean
ngOnInit	ngOnInit()	void
onChartInit	onChartInit(e: any, index: number)	void
setValue	setValue(value: number)	string
setColor	setColor(value: number)	string

Properties

Property	Type
_widget	Widget

Property	Type
todayWidgetsEnum	any
Branches	BranchessTendancy[]
initOpts	any
series	any[]
headerData	any
echartsIntance	any
options	any
TargetBooking	`string
AchievedBooking	`string
currenFeature	number

Where it refuses work

- WidgetBranchTendancyComponent stops the work with an early return when `this.dashboardLibraryService.checkFeatureSubscription(this.widget.chartEnumId)` .
- WidgetBranchTendancyComponent stops the work with an early return when `value > 150` .

Diagram

```
graph LR
    Dashboard[Dashboard View] --> Component[WidgetBranchTendancyComponent]
    Component --> Subscription[checkFeatureSubscription]
    Subscription -->|Allowed| Data[getDashboardData]
    Data --> Chart[onChartInit]
    Chart --> Value[setValue]
    Chart --> Color[setColor]
```

Usage

```
import { Component, ViewChild, AfterViewInit } from '@angular/core';
import { WidgetBranchTendancyComponent } from '../widget-branch-tendancy.component';

@Component({
  selector: 'app-dashboard-page',
  template: `
    <app-widget-branch-tendancy></app-widget-branch-tendancy>
  `,
})
```

```

}))
export class DashboardPageComponent implements AfterViewInit {
  @ViewChild(WidgetBranchTendancyComponent)
  branchTendancyWidget!: WidgetBranchTendancyComponent;

  ngAfterViewInit(): void {
    if (this.branchTendancyWidget.checkFeatureSubscription()) {
      this.branchTendancyWidget.getDashboardData();
    }
  }
}

```

AI Coding Instructions

- Let Angular invoke `ngOnInit()` through the component lifecycle; do not call it directly from application code.
- Check `checkFeatureSubscription()` before triggering dashboard data retrieval or showing feature-specific content.
- Call `onChartInit()` from the chart library initialization callback so chart-dependent state is available.
- Keep `setValue()` and `setColor()` aligned with the data returned by `getDashboardData()`.
- Preserve the existing component integration within the dashboard widget library when changing inputs, chart events, or data loading behavior.

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`