How to get on scroll events?

Asked 2 years, 7 months ago   Active 8 months ago   Viewed 67k times

I need to get scroll events from a div with overflow: scroll in my Angular 2 app.

25

It seems onscroll event do not works on Angular 2.

How could I achieve that?

★

4

| html | css | 🅰 angular | typescript | scrollbar |

edited Jun 27 '17 at 18:19   asked Dec 23 '16 at 16:40

isherwood   Natanael
38.8k 12   87   114   **775**   4   15   30

1   Quite often you can apply events that as onclick, onsubmit etc. In angular the equivalent is usually just removing the 'on' and adding () from event names. e.g (click), (submit) etc... :) – AJT_82 Dec 23 '16 at 16:45 ✎

4 Answers

```
// @HostListener('scroll', ['$event']) // for scroll events of the current element
@HostListener('window:scroll', ['$event']) // for window scroll events
onScroll(event) {
  ...
}
```

70

or

✔

```
<div (scroll)="onScroll($event)"
```

edited Nov 22 '18 at 12:53   answered Dec 23 '16 at 16:41

Günter Zöchbauer

(scroll)="onScroll($event)" solved my problem well. Do you know if there's a equivalent binding to onresize of a element? – Natanael Dec 23 '16 at 17:40

No, browsers don't provide events for resize of elements, only for window resize (`'window:resize'`). You could implement `ngDoCheck()` and poll the size there. This can be expensive though, therefore if you can rely on less frequent events for polling the size using such an event instead would be preferred. – Günter Zöchbauer Dec 23 '16 at 17:51 ✏️

@GünterZöchbauer Can you tell from the event which direction the scroll is going? That is without examining and comparing the div scrollY value to the previous? – Katana24 Aug 23 '17 at 12:37 ✏️

developer.mozilla.org/en-US/docs/Web/Events/scroll lists the information provided by the event, which doesn't include anything about direction. – Günter Zöchbauer Aug 23 '17 at 12:41

In addition, This one is worked in Angular 6 --> stackoverflow.com/a/42547136/621951 – Günay Gültekin Oct 11 '18 at 17:18

▲

31

▼

for angular 4, the working solution was to do inside the component

```
@HostListener('window:scroll', ['$event']) onScrollEvent($event){
  console.log($event);
  console.log("scrolling");
}
```

answered May 24 '17 at 13:53

Thomas Webber
788 8 9

Hello, i just used your code in my component and last i added a (scroll)="onScrollEvent($event)" but i doesn't work. – gtzinos May 26 '17 at 4:52

1 (scroll)="onScrollEvent($event)" doesn't work, but my code is currently working on angular version 4.0.0, It triggers the scroll event – Thomas Webber May 26 '17 at 10:28

this works fine, no need to add `(scroll)="onScrollEvent($event)` in the HTML with this. – blueprintchris May 3 '18 at 6:57

▲

You could use a @HostListener decorator. Works with Angular 4 and up.

21

```
import { HostListener } from '@angular/core';

@HostListener("window:scroll", []) onWindowScroll() {
    // do some stuff here when the window is scrolled
    const verticalOffset = window.pageYOffset
            || document.documentElement.scrollTop
            || document.body.scrollTop || 0;
}
```

edited Mar 30 '18 at 14:04   answered Sep 26 '17 at 13:38

isherwood   Uliana Pavelko
38.8k 12 87 114   **1,345** 18 19

Listen to `window:scroll` event for window/document level scrolling and element's `scroll` event for element level scrolling.

8

window:scroll

```
@HostListener('window:scroll', ['$event'])
onWindowScroll($event) {

}
```

or

```
<div (window:scroll)="onWindowScroll($event)">
```

scroll

```
@HostListener('scroll', ['$event'])
onElementScroll($event) {

}
```

or

```
<div (scroll)="onElementScroll($event)">
```

`@HostListener('scroll', ['$event'])` won't work if the host element itself is not scroll-able.

Examples

- [Using Event Binding](#)
- [Using HostListener](#)

answered Aug 8 '17 at 14:43

kodebot
870   12   22