

Lint error: Implement lifecycle hook interfaces

Asked 3 years ago Active 2 years, 11 months ago Viewed 19k times



37

Hey everyone I got a linting error and I am not sure what it is talking about, this is the error: `src/app/particles/particles.component.ts[4, 1]: Implement lifecycle hook interfaces` (<https://angular.io/docs/ts/latest/guide/style-guide.html#!#09-01>) Row nr 4 is the `@Component({` line

I have read the link it gives and I get what it tries to tell me (at least I think so :)) but I can't see how it applies in this case.

Thanks for any assistance.

```
import { Component, ViewChild, ElementRef, HostListener } from '@angular/core';
import { Particle } from './particle';

@Component({
  selector: 'km-particles',
  styles: ['canvas { transition: opacity 0.5s cubic-bezier(0.4, 0.0, 0.2, 1);}'],
  template: ` <canvas #canvas
    [attr.width]='width'
    [attr.height]='height'
    [style.opacity]='opacity'>
    </canvas> `
})
export class ParticlesComponent {

  private ctx: CanvasRenderingContext2D;
  private width: number;
  private height: number;
  private opacity: number;
  private particles: Particle[];
  private particleClearLoop: any;

  public playParticles: boolean;

  // get the element with the #canvas on it
  @ViewChild('canvas') canvas: ElementRef;

  // on window resize, restart the animation with the new boundaries
  @HostListener('window:resize', ['$event'])
```

By using our site, you acknowledge that you have read and understand our [Cookie Policy](#), [Privacy Policy](#), and our [Terms of Service](#).

```
constructor() {
  this.opacity = 0;
}

// wait for the view to init before using the element
ngAfterViewInit() {
  this.ctx = this.canvas.nativeElement.getContext('2d');

  // ok all is ready, start the particle animation
  this.initAnim();
}

createParticles() {
  this.particles = [];
  // let's make us some particles
  for (let i = 0; i < 150; i++) {
    this.particles.push(new Particle());
  }
}

draw() {
  // clear canvas
  this.ctx.clearRect(0, 0, this.width, this.height);

  // draw the particles
  this.particles.forEach((particle) => {

    particle.timestamp = Math.floor(Date.now());
    particle.counter = particle.sign * (particle.timestamp / particle.speed *
Math.PI);

    this.ctx.beginPath();
    // define the circleparticle
    this.ctx.arc( particle.xPos + particle.radius * Math.cos(particle.counter / 100),
      particle.yPos + particle.radius * Math.sin(particle.counter / 100),
      particle.width,
      0,
      Math.PI * 2,
      false);

    this.ctx.globalAlpha = particle.alpha;
    this.ctx.fillStyle = particle.color;
    this.ctx.fill();

  });
}
```

By using our site, you acknowledge that you have read and understand our Cookie Policy, Privacy Policy, and our Terms of Service.

```
    }  
  }  
  
  // init resize and make the particles  
  initAnim() {  
    this.playParticles = false;  
    this.opacity = 0; // so we don't see the flicker  
  
    clearInterval(this.particleClearLoop);  
  
    this.particleClearLoop = setTimeout(() => {  
      this.opacity = 1;  
      this.playParticles = true;  
      this.resize();  
      this.createParticles();  
      this.draw();  
    }, 500);  
  }  
  
  // method that resizes the canvas to the full width/height of the window  
  resize() {  
    this.width = window.innerWidth;  
    this.height = window.innerHeight;  
  }  
}
```



asked Aug 25 '16 at 19:59

[Björn Hjorth](#)

595 2 8 23

1 Answer



you are using `ngAfterViewInit` Lifecycle Hook, you just need to add below to make TSLint happy,

92

```
export class ParticlesComponent implements AfterViewInit
```

By using our site, you acknowledge that you have read and understand our [Cookie Policy](#), [Privacy Policy](#), and our [Terms of Service](#).



answered Aug 25 '16 at 20:06

**Madhu Ranjan****12.5k** 5 42 58

1 Hey Madhu this helped, however I was wondering if you know why It needs it? It works fine without it. – [L1ghtk3ira](#) Sep 6 '18 at 14:45

One option might be to add `/* tslint:disable use-lifecycle-interface */`, just to gag TSLint. A better option is to follow Madhu Ranjan's advice: 1) `import { Component, OnInit } from '@angular/core';`, then 2) `export class MyComponent implements OnInit { ... }`. The app should "work" in any case - it's just a TSLint warning. – [paulsm4](#) Jan 16 at 1:11

Got a question that you can't ask on public Stack Overflow? [Learn more](#) about sharing private information with Stack Overflow for Teams.



By using our site, you acknowledge that you have read and understand our [Cookie Policy](#), [Privacy Policy](#), and our [Terms of Service](#).