

## [CGA-JAVA-WFDA] Web Front-end Development Angular 2.1

[Dashboard](#) / [My courses](#) / [CGA-JAVA-WFDA-2.1](#) / [4. Angular Component & Template](#) / [\[Bài đọc\] Vòng đời của Component](#)

### [Bài đọc] Vòng đời của Component

Angular cung cấp các móc vòng đời (lifecycle hooks), chúng cho phép chèn thêm những tác vụ cần thiết trong những khoảnh khắc quan trọng trong quá trình khởi tạo cho đến khi phá hủy các component. Chúng ta có thể khai thác những khoảnh khắc quan trọng trong vòng đời đó bằng cách triển khai một hoặc nhiều interfaces trong thư viện lõi của Angular.

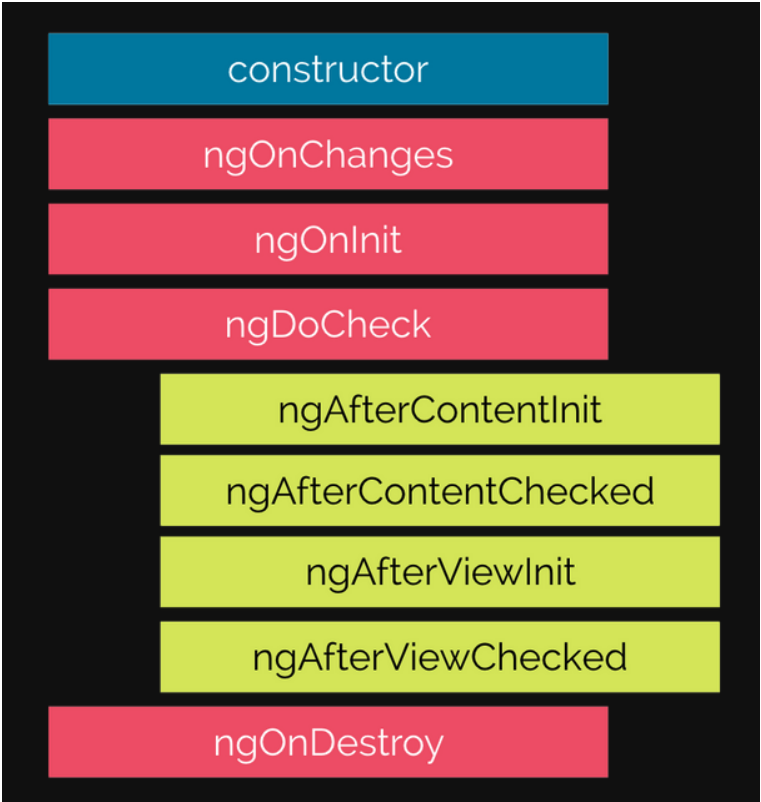
Mỗi giao diện có một phương thức hook duy nhất có tên là tên interface có thêm tiền tố ng. Ví dụ: interface `OnInit` có một phương thức hook có tên `ngOnInit()` mà Angular gọi ngay sau khi tạo component:

```
@Directive()
export class PeekABooDirective implements OnInit {
  constructor(private logger: LoggerService) { }

  // implement OnInit's `ngOnInit` method
  ngOnInit() {
    this.logIt(`OnInit`);
  }

  logIt(msg: string) {
    this.logger.log(`#${nextId++} ${msg}`);
  }
}
```

### Trình tự vòng đời



Sau khi khởi tạo một component / directive bằng cách gọi hàm `constructor` của nó, Angular gọi các phương thức hook theo trình tự sau tại các thời điểm cụ thể:

Hook	Mục đích & thời điểm
<code>ngOnChanges()</code>	Thực thi khi Angular thiết lập các thuộc tính đầu vào ràng buộc dữ liệu. Được gọi trước <code>ngOnInit()</code> và bất cứ khi nào một thuộc tính ràng buộc dữ liệu thay đổi.
<code>ngOnInit()</code>	Khởi tạo directive / component sau khi Angular hiển thị các thuộc tính ràng buộc dữ liệu và đặt các thuộc tính đầu vào của một lần, sau <code>ngOnChanges()</code> đầu tiên.
<code>ngDoCheck()</code>	Phát hiện và hành động theo những thay đổi mà Angular không thể hoặc sẽ không tự mình phát hiện. Được gọi trong mỗi lần chạy sau <code>ngOnChanges()</code> và <code>ngOnInit()</code> .
<code>ngAfterContentInit()</code>	Thực thi sau khi Angular thêm nội dung bên ngoài vào view của component / view mà directive được đưa vào. Được gọi một lần sau <code>ngDoCheck()</code> .
<code>ngAfterContentChecked()</code>	Thực thi sau khi Angular đã kiểm tra nội dung bên ngoài đã được đưa vào view của component. Được gọi sau <code>ngAfterContentInit()</code> theo.
<code>ngAfterViewInit()</code>	Thực thi sau khi Angular khởi tạo các view của component và các view con / view mà directive được đưa vào. Được gọi một lần sau <code>ngAfterContentChecked()</code> đầu tiên.
<code>ngAfterViewChecked()</code>	Thực thi sau khi Angular kiểm tra các view của component và các view con /view mà directive được đưa vào. Được gọi sau mỗi lần chạy tiếp theo <code>ngAfterContentChecked()</code> tiếp theo.

Hook	Mục đích & thời điểm
ngOnDestroy()	Dọn dẹp ngay trước khi Angular phá hủy directive / component. Hủy đăng ký Observables và tách trình xử lý sự kiện để tránh trước khi Angular phá hủy directive / component.

Last modified: Wednesday, 24 February 2021, 8:12 PM

Previous Activity

Jump to...

Next Activity

### CHƯƠNG TRÌNH

- Career
- Premium
- Accelerator

### TÀI NGUYÊN

- Blog
- Tạp chí Lập trình
- AgileBreakfast

### Follow Us

