

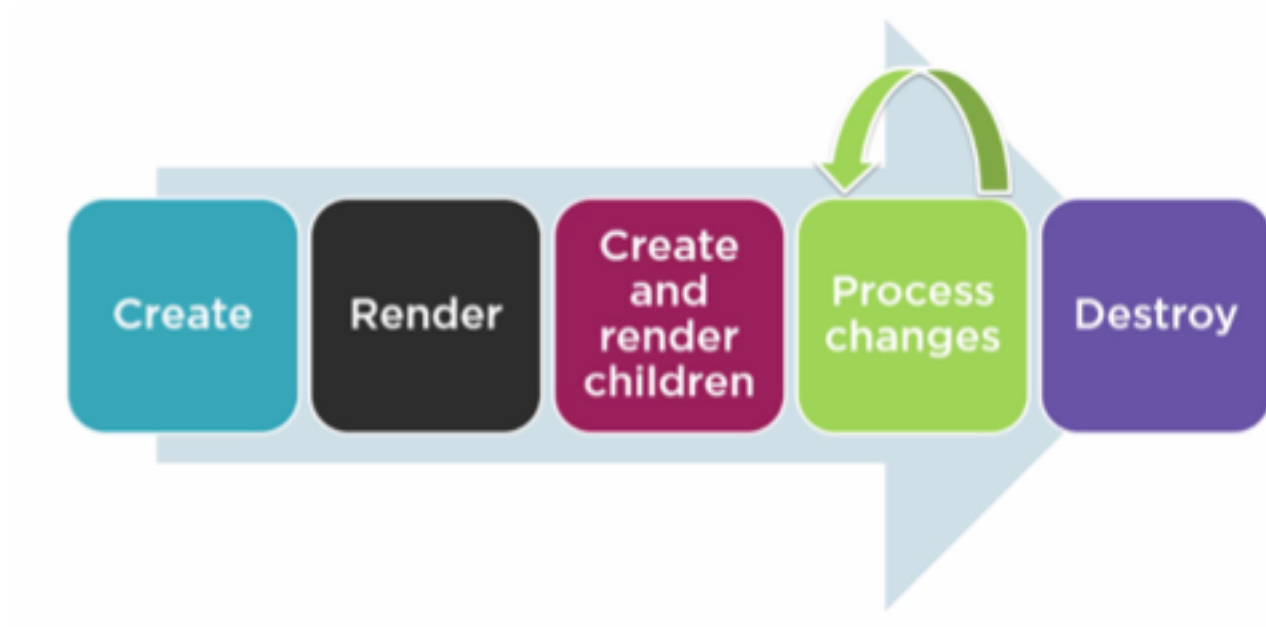


```
function(b, c) {  
  var d = a(c.form.querySelectorAll('input[type=checkbox][name="' + b.el.name + '"']));  
  if (0 === d.index(b.el)) {  
    var e = d.filter(":checked").length;  
    return e >= b.arg || g.minChecked.replace("{count}", b.arg)  
  }  
},  
maxSelected: function(a) {  
  return null !== a.val ? a.val  
},  
minSelected: function(a) {  
  return null !== a.val && a.val  
},  
radio: function(b) {  
  var c = a(this.form.querySelector  
  return 1 === c  
},  
custom: function(a, b) {  
  var c = b.options.custom[a.arg],  
  d = new RegExp(c.pattern);  
  return d.test(a.val) || c.errorMessage  
},  
remote: function(a) {  
  a.remote = a.arg  
}
```

# Formación Desarrollo Aplicaciones Web con Angular 5

## 8.- Ciclo de vida de los componentes

## Angular | Ciclo de vida de los componentes

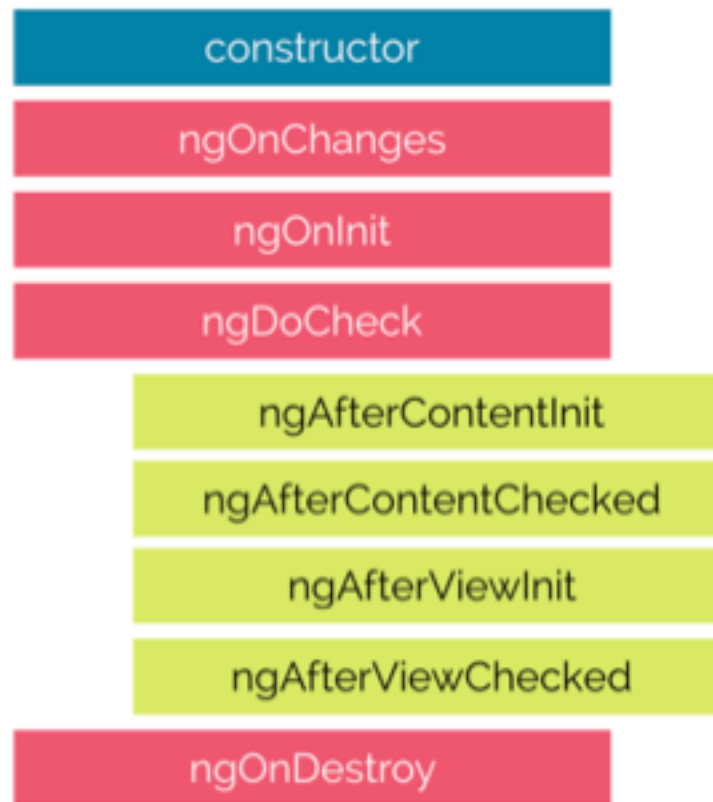


## Angular | Ciclo de vida de los componentes

- Métodos para el ciclo de vida de componentes.  
Angular dispone de una serie de métodos para ejecutar *hooks* en las diferentes fases del ciclo de vida de los componentes. Estos métodos son implementados a través de interfaces en la clase del componente.

## Angular | Ciclo de vida de los componentes

- Métodos para el ciclo de vida de componentes.



## Angular | Ciclo de vida de los componentes

Interfaz	Método	Descripción
OnChanges	ngOnChanges()	Responde cuando Angular restablece propiedades de entrada vinculadas a datos. El método recibe un objeto SimpleChanges de los valores de propiedad actuales y anteriores. Es llamado antes de ngOnInit() y cuando una o más propiedades de entrada enlazadas a datos cambien.
OnInit	ngOnInit()	Inicializa la directiva o el componente una vez que Angular haya mostrado las propiedades vinculadas a datos y establece las propiedades de entrada de la directiva o el componente. Es llamado solo una vez, justo después del primer ngOnChanges().

## Angular | Ciclo de vida de los componentes

Interfaz	Método	Descripción
DoCheck	ngDoCheck()	Es activado durante cada detección de cambios, inmediatamente después de ngOnChanges() y ngOnInit().
AfterContentInit	ngAfterContentInit()	Es activado después de que Angular proyecte contenido externo en la vista del componente. Se llama una vez después del primer NgDoCheck().

## Angular | Ciclo de vida de los componentes

Interfaz	Método	Descripción
AfterContentChecked	ngAfterContentChecked()	Responde después de que Angular comprueba el contenido proyectado en el componente. Es llamado después de ngAfterContentInit() y cada NgDoCheck() posterior.
AfterViewInit	ngAfterViewInit()	Responde después de que Angular inicializa las vistas del componente y las vistas secundarias. Se llama una vez después del primer ngAfterContentChecked().

## Angular | Ciclo de vida de los componentes

Interfaz	Método	Descripción
AfterViewChecked	ngAfterViewChecked()	Responde después de que Angular comprueba las vistas del componente y las vistas secundarias. Se llama después de ngAfterViewInit() y cada ngAfterContentChecked() posterior.
OnDestroy	ngOnDestroy()	Es llamado justo antes de que Angular destruya la directiva o el componente. Anula la suscripción de observables y desconecta los manejadores de eventos para evitar fugas de memoria.