



```
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

## 13.- Creación de servicios

## Angular | Creación de servicios

- Servicios

Los servicios en Angular, centralizan la obtención de datos y los proporcionan a través de inyección de dependencias a uno o varios componentes, dejando a estos la lógica de la presentación y aumentando la modularidad de la aplicación.

Se trata de archivos TypeScript con la convención de nombres:

`nombreservicio.service.ts`

## Angular | Creación de servicios

- Servicios

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

```
@Injectable({  
  providedIn: 'root' //u otro módulo  
})
```

```
export class NombreService {  
  
  //propiedades y métodos  
  
}
```

## Angular | Creación de servicios

- Formas de implementación
  - En el propio servicio  
`providedIn: 'root'` //a partir de Angular 6
  - En el módulo  
`providers=[NombreService]`
  - En el decorador del componente  
`providers=[NombreService]`

## Angular | Creación de servicios

- Implementación en la clase del componente.  
La inyección de dependencias se lleva a cabo mediante una la generación de una instancia de la clase del componente en el constructor, desacoplando el componente del servicio.

```
constructor(private nombreService: NombreService) {}
```

La instancia tendrá los métodos del servicio para acceder a los datos.

```
this.nombreService.metodo()
```