



```
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

## 15.- Routing y aplicaciones single page

## Angular | Routing y aplicaciones single page

- Sintaxis implementación en módulo raíz

```
import { Routes, RouterModule } from '@angular/router';
```

```
const rutasModulo: Routes = [  
  {path: 'ruta', component: NombreComponent},  
  ...  
];
```

```
imports: [  
  BrowserModule,  
  RouterModule.forRoot(rutasModulo)  
],
```

## Angular | Routing y aplicaciones single page

- Template de carga

```
<router-outlet></router-outlet>
```

- Navegación desde elementos HTML

```
routerLink='/ruta'
```

- Clases activas para menús

```
[routerLinkActiveOptions]="{exact:true}"
```

```
routerLinkActive="activo"
```

## Angular | Routing y aplicaciones single page

- Rutas con parámetros

En la constante de rutas del módulo

```
{path: 'ruta/:parametro', component: NombreComponent},
```

En el enlace del elemento HTML

```
routerLink= '/ruta/parametro'
```

En el componente

```
import { ActivatedRoute } from '@angular/router';
```

```
private route: ActivatedRoute
```

```
this.route.snapshot.params['parametro'];
```

## Angular | Routing y aplicaciones single page

- Navegación programática

En el componente

```
import { Router } from '@angular/router';
```

```
private router: Router
```

```
this.router.navigate(['/ruta']);
```

## Angular | Routing y aplicaciones single page

- Guards de rutas

En la constante de rutas del módulo

```
{path: 'ruta', component: NombreComponent,  
  canActivate: [NombreGuard]},
```

En el archivo generado TypeScript con ng g guard nombre  
canActivate(

```
  next: ActivatedRouteSnapshot,  
  state: RouterStateSnapshot): boolean {  
    // condición true o false;  
  }
```

## Angular | Routing y aplicaciones single page

- Módulos adicionales

`ng g m nombremodulo`

Para agregar componentes al nuevo módulo:

`ng g c nombremodulo/directorio-opcional/nombrecomponente`

## Angular | Routing y aplicaciones single page

- Lazy Loading

En el módulo raíz

```
{path: 'modulo',  
loadChildren: 'src/app/.../modulo.module#ModuloModule'},
```

En el módulo adicional creado con ng g m nombreModulo

```
{path: "", component: ComponenteComponent},
```

```
RouterModule.forChild(rutas)
```



## Angular | Routing y aplicaciones single page

- Módulos compartidos

En el módulo compartido

```
declarations: [  
    DirectivaDirective, PipePipe, ComponenteComponent  
],  
exports: [  
    DirectivaDirective, PipePipe, ComponenteComponent  
]
```

En el módulo que importe

```
imports: [  
    ModulocompartidoModule  
],
```