

viewPriders

I `viewProvider` rendono **inaccessibili** i servizi a tutti i figli proiettati con *ng-content*!

Utile ad esempio nello sviluppo di una libreria di componenti per evitare che gli sviluppatori vadano a toccare servizi ad uso interno.

```
@Component({
  selector: 'outer',
  template: `
    <div>
      <!-- Projected Content can NOT reach MyService -->
      <ng-content></ng-content>
    </div>
  `,
  providers: [ MyService ]
})
export class OuterComponent {}
```

viewProviders

I viewProvider rendono **inaccessibili** i servizi a tutti i figli proiettati con *ng-content*!

Utile ad esempio nello sviluppo di una libreria di componenti per evitare che gli sviluppatori vadano a toccare servizi ad uso interno.

```
@Component({
  selector: 'outer',
  template: `
    <div>
      <!-- Projected Content can NOT reach MyService -->
      <ng-content></ng-content>
    </div>
  `,
  viewProviders: [ MyService ]
})
export class OuterComponent {}
```

Element Injectors

In un template, ogni elemento ha il suo “mini” Injector, chiamato Element Injector!

Le istanze di direttive applicate ad elementi/componenti vengono gestite dal loro Element Injector.