


```
/**  
 * Injection Token  
 */  
const APP_CONFIG = new InjectionToken<AppConfig>('This is the configuration!');  
const appConfig: AppConfig = { apiUrl: 'api.domain.com', ... };  
  
/**  
 * Declaring the provider  
 */  
@NgModule({  
  ...  
  providers: [{ provide: APP_CONFIG, useValue: appConfig }]  
})  
export class ConfigModule {}  
  
/**  
 * Declaring the dependency  
 */  
@Component({ ... })  
export class MyComponent {  
  
  constructor(@Inject(APP_CONFIG) config: AppConfig) {}  
}
```

infection

Quando una dipendenza non è rapresen-



InjectionToken

Quando una dipendenza non è rappresentabile da una classe

```
/**  
 * Injection Token  
 */  
const APP_CONFIG = new InjectionToken<AppConfig>('This is the configuration!');  
const appConfig: AppConfig = { apiUrl: 'api.domain.com', ... };  
  
/**  
 * Declaring the provider  
 */  
@NgModule({  
  ...  
  providers: [{ provide: APP_CONFIG, useValue: appConfig }]  
})  
export class ConfigModule {}  
  
/**  
 * Declaring the dependency  
 */  
@Component({ ... })  
export class MyComponent {  
  constructor(@Inject(APP_CONFIG) config: AppConfig) {}  
}
```

Utilissimo per rendere dei moduli configurabili dall'esterno (vedi `RouterModule`, `StoreModule` di `ngrx` e tanti altri. La configurazione viene dichiarata nei provider con un `InjectionToken`!

GR ZIE!