Functional Guards

ng old

```
export class AuthGuard implements CanActivate {
    constructor(private readonly auth: AuthService) {}

    canActivate() {
        return this.auth.isLoggedIn();
    }
}
```

ng new

```
export const authGuard: CanActivateFn =
    () => inject(AuthService).isLoggedIn();
```

The new Angular API

- Based on **Functions** instead of decorators
- Based on **Signals**

ng old

```
export class BeforeComponent {
  constructor(@Inject(SOME_NUMBER) private num: number) {}
  @Input({required: true}) title!: string;
}
```

ng new

```
export class AfterComponent {
  readonly num = inject(SOME_NUMBER);
  readonly title = input.required<string>();
}
```

Why functions?

- Avoid unnecessary classes just for dependency Injection
- Less decorators
- Less lifecycle hooks
- Cleaner, more direct code