Klassendiagramm

ℹ️ PlantUML docs: https://plantuml.com/class-diagram

Template (`template.puml`)

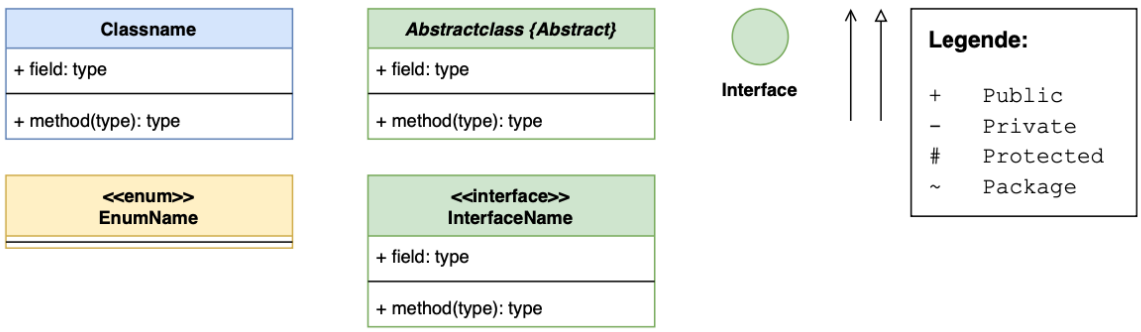
```
@startuml

skinparam classAttributeIconSize 0

' TODO: ...

hide empty members

@enduml
```



Klassen

- Default-Konstruktor wird weggelassen

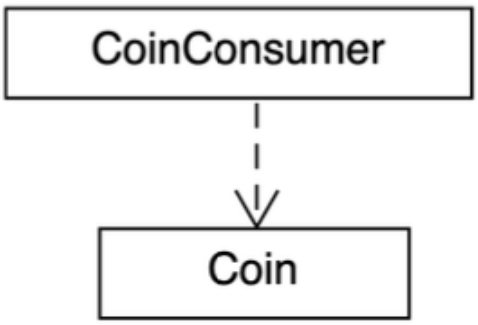
Beziehungen

- https://plantuml.com/class-diagram#9dd2a6eca0c2a0e7

gestrichtelter Pfeil

⇒ einfache Abhängigkeit

⇒ Klasse wird ausschliesslich als Parameter oder Rückgabewert in Methoden / Konstruktor verwendet.



Beispiel: Die Klasse CoinConsumer ist von Coin abhängig, da Coin als Datentyp in einer Methode verwendet wird.

```
A ..> B
```

durchgezogene Linie

⇒ stärkere Abhängigkeit = Assoziation

⇒ Datentyp in den Datenfeldern

```
A --> B
```

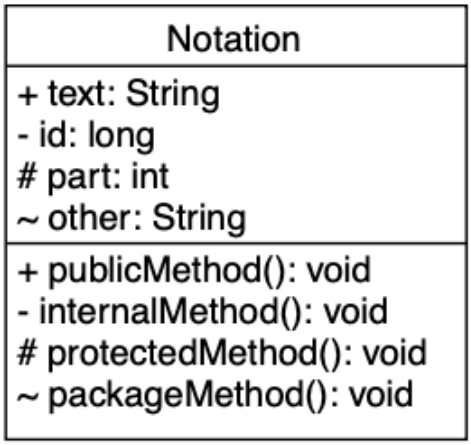
Zugriffsmodifikatoren

`private` -

`protected` #

`public` +

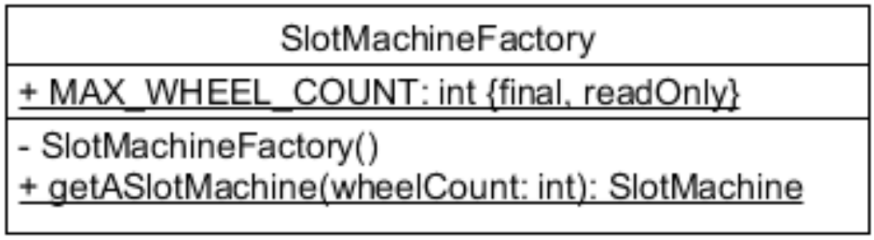
`package` ~



Schlüsselwörter

`static` unterstrichen

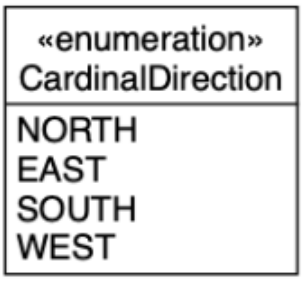
`final` {final} anhängen




```
class A {
    -String id {static} {final}
}
```

Enums

```
enum Test << enumeration >> {
    A
    B
    C
}
```



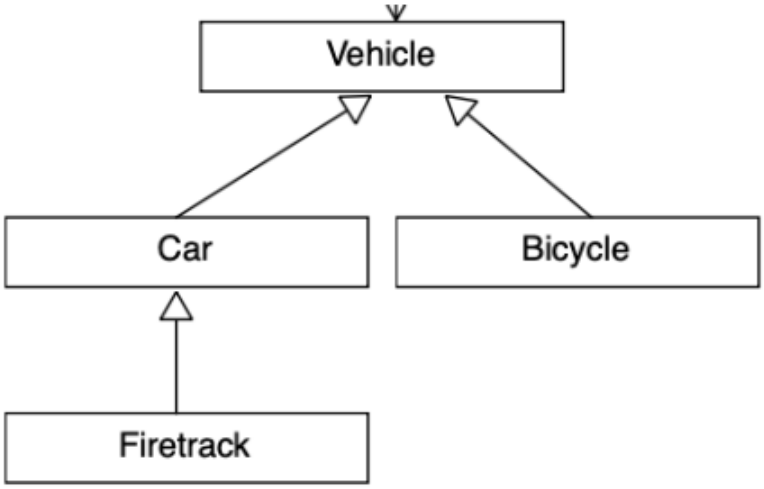
Vererbung

`extends` : durchgezogener Strich `--|>`

`implements`: gestrichelter Strich `..|>`

```
class Vehicle {}
class Bycicle {}
class Car {}
class Firetruck {}

Bycicle --|> Vehicle
Car --|> Vehicle
Firetruck --|> Car
```



Abstrakte Klassen

⇒ kursiv oder {abstract}

```
abstract class Vehicle << abstract >> {
    {abstract} + getVelocity(): int
}
```

Interfaces

⇒ keine Sichtbarkeitsangaben

```
interface Test << interface >> {
    compareTo(String otherString): int
}
```