

SELECTED LOGICAL VIEW REPORT

LOGICAL VIEW

FAHRZEUG

Basis-Klasse "Fahrzeug".
Abstrahiert ein generelles Fahrzeug das eine absolute Koordinate (in Kilometer) hat.
Kann zu einer neuen Koordinaten fahren und die dafür zurückgelegten Kilometer berechnen.

Private Attributes:

PosX : double

Absolute X-Koordinate in Kilometer.

PosY : double

Absolute Y-Koordinate in Kilometer.

Public Operations:

Fahrzeug (Xinit : double, Yinit : double) :

Konstruktor mit Initial-Koordinate (X/Y).

fahren (X : double, Y : double) : double

Zur neuen, absoluten Koordinate (X/Y) fahren und die soeben zurückgelegten Kilometer als Return-Wert zurückgeben.

Trace auf Standard-Output:

- Klassenname.Methode():Resultat

- z.B. "Fahrzeug.fahren(): 5.0"

printlnInfos (Prefix : String) : void

Gleich wie printInfos(), aber Zeile wird mit Newline abgeschlossen.

Protected Operations:

printInfos (Prefix : String) : void

Gibt die Koordinaten auf die Konsole aus.

Der Parameter 'Prefix' wird vor die Koordinaten geschrieben.

Beispiel Aufruf:

```
fz.printInfos("fz");
```

Ausgabe auf Konsole (Beispiel):

```
fz: Koord = 3.0/4.0
```

Zeile wird ohne(!) Newline abgeschlossen.

STRASSENFAHRZEUG

Strassenfahrzeug.

Hat alle Eigenschaften von Fahrzeug.

Zusätzlich wird die total zurückgelegte Kilometeranzahl kumuliert.

Derived from Fahrzeug

Private Attributes:

KM : double

Summer der total gefahrenen Kilometer.

Public Operations:

Strassenfahrzeug (Xinit : double, Yinit : double, KMinit : double) :

Konstruktor mit Initial-Koordinate (X/Y) und Initial-Wert der gefahrenen Kilometer.

fahren (X : double, Y : double) : double

Zusätzlich zu Fahrzeug.fahren() werden die neu zurückgelegten Kilometer dem Attribut KM hinzuaddiert.

Gibt die soeben zurückgelegten Kilometer als Return-Wert zurückgeben.

printlnInfos (Prefix : String) : void

Gibt die Koordinaten und die total gefahrenen Kilometer auf die Konsole aus.

Der Parameter 'Prefix' wird vor die Koordinaten geschrieben.

Beispiel Aufruf:

```
sfz.printlnInfos("sfz");
```

Ausgabe auf Konsole (Beispiel):

```
sfz: Koord = 3.0/4.0 km = 5.0
```

Zeile wird mit Newline abgeschlossen.

WASSERFAHRZEUG

Wasserfahrzeug.

Hat alle Eigenschaften von Fahrzeug.

Zusätzlich wird die total zurückgelegte Seemeilenanzahl kumuliert.

Derived from Fahrzeug

Private Attributes:

SM : double

Summer der total gefahrenen Seemeilen.

Public Operations:

Wasserfahrzeug (Xinit : double, Yinit : double, SMinit : double) :

Konstruktor mit Initial-Koordinate (X/Y) und Initial-Wert der gefahrenen Seemeilen.

fahren (X : double, Y : double) : double

Zusätzlich zu Fahrzeug.fahren() werden die neu zurückgelegten Kilometer auf Seemeilen umgerechnet und dem Attribut SM hinzuaddiert.

Eine Seemeile entspricht 1852 Meter.

Gibt die soeben zurückgelegten Kilometer als Return-Wert zurückgeben (nicht Seemeilen!).

printlnInfos (Prefix : String) : void

Gibt die Koordinaten und die total gefahrenen Seemeilen auf die Konsole aus.

Der Parameter 'Prefix' wird vor die Koordinaten geschrieben.

Beispiel Aufruf:

```
wfz.printlnInfos("wfz");
```

Ausgabe auf Konsole (Beispiel):

```
wfz: Koord = 3.0/4.0 sm = 2.6997840172786174
```

Zeile wird mit Newline abgeschlossen.