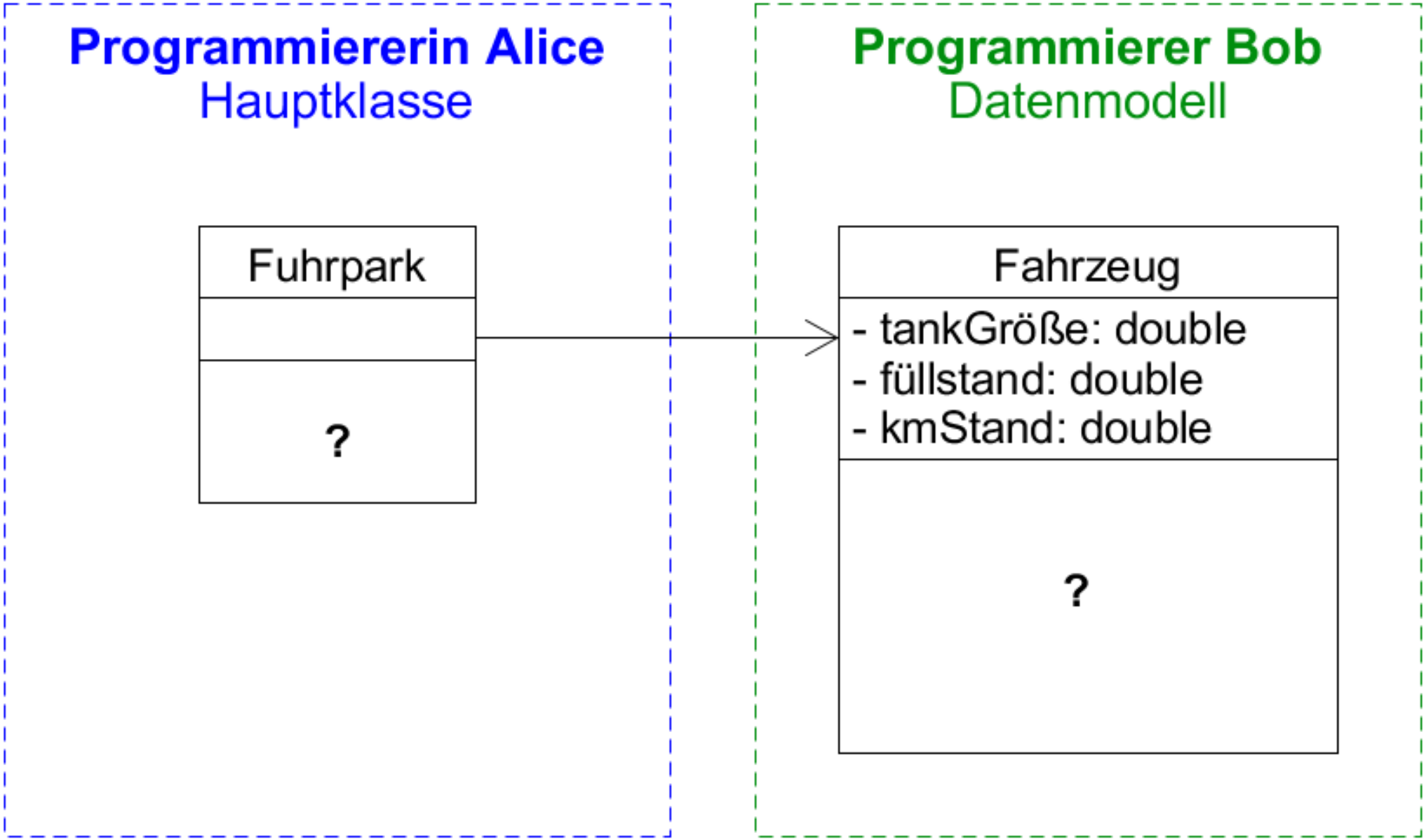
# Sinnvolle Modellierung von Klassen

# Arbeitsteilung



# Bob: Datenmodell

Bob stellt Methoden für Alice zur Verfügung:

- für möglichst viele Anwendungsfälle (ungenaue Vorstellung, was Alice macht)
- mit Sicherheitsmechanismen, um Fehlern vorzubeugen

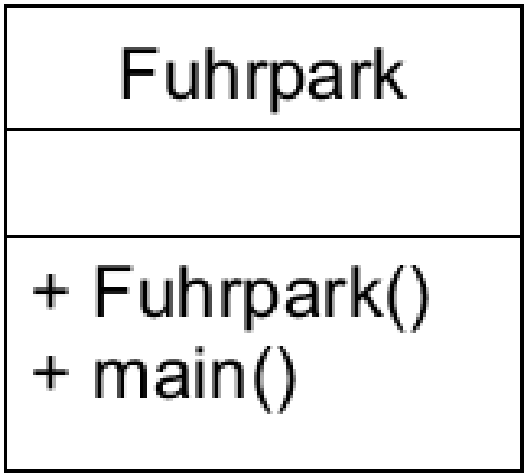
# Alice: Hauptklasse

Alice nutzt Bob’s Methoden:

- erzeugt Objekte mit zugehörigen Daten
- bearbeitet diese Daten mit den Methoden

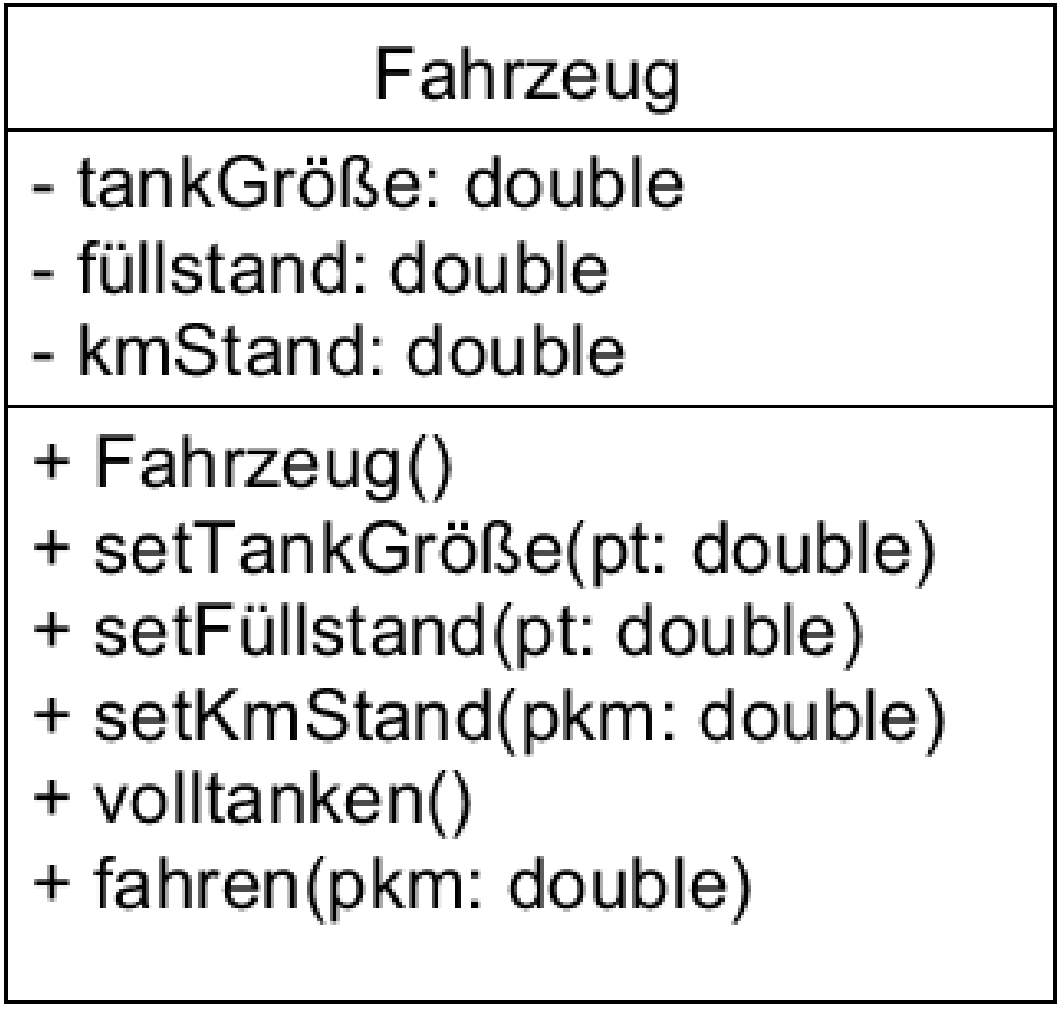
# Alice: Hauptklasse

einfach: Konstruktor erzeugt Objekte,
main-Methode beschreibt Ablauf



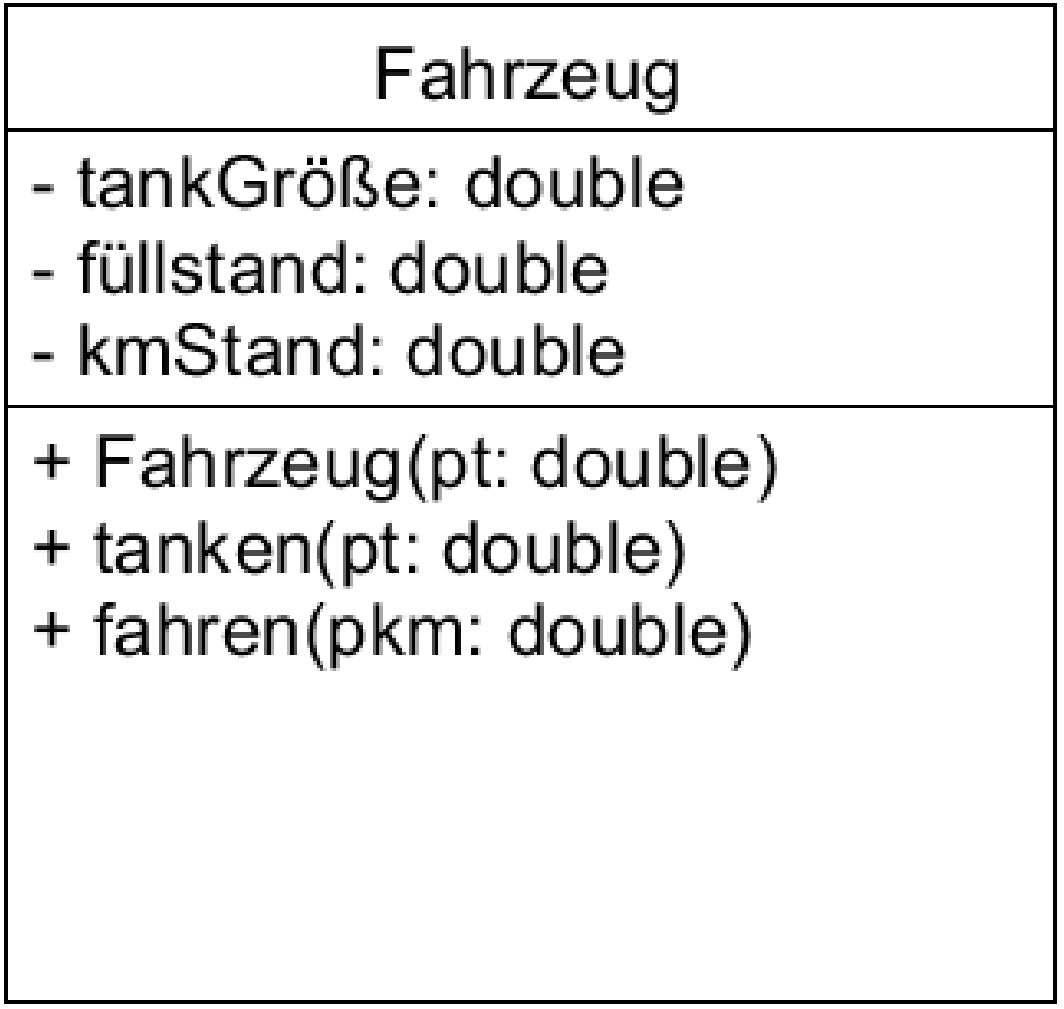
# Bob: Datenmodell

Vorschlag Bob: was ist problematisch?



# Bob: Datenmodell

Vorschlag Bob: Methoden verbessert



# Bob: Datenmodell

1. Parameter überprüfen
2. Attribute ändern – nur, falls Parameter OK

```
public void fahren(double pkm)
{
    if (                                    )
    {




    }
}
```

# Bob: Datenmodell

1. Parameter überprüfen
2. Attribute ändern – nur, falls Parameter OK

```
public void fahren(double pkm)
{
    if (pkm > 0 && pkm/10 < füllstand)
    {
        kmStand = kmStand + pkm;
        füllstand = füllstand – pkm/10;
    }
}
```

# Alice: Hauptklasse

Alice will folgendes implementieren:
„Falls weniger als 10 L im Tank: volltanken“

```
public void main()
{

    if (                              )
    {


        f1.tanken(                          )
    }
```

# Alice: Hauptklasse

Alice will folgendes implementieren:
„Falls weniger als 10 L im Tank: volltanken“

```
public void main()
{
    double tankmenge;
    if (f1.getFüllstand() < 10)
    {
        tankmenge = f1.getTankGröße() -
                    f1.getFüllstand();
        f1.tanken(tankmenge);
    }
```

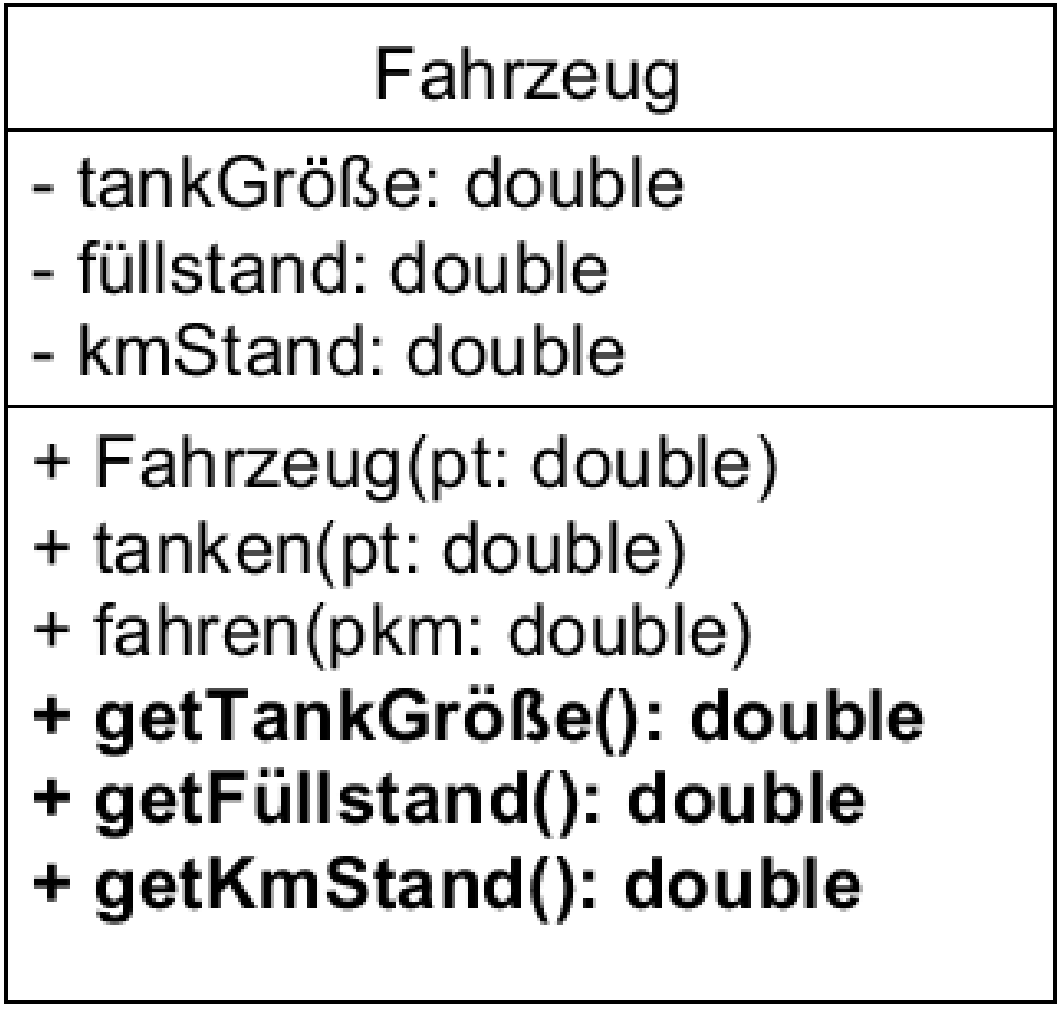
# Alice: Hauptklasse

**Folgerung**:
Bob's Datenmodell braucht get-Methoden!

```
public void main()
{
    double tankmenge;
    if (f1.getFüllstand() < 10)
    {
        tankmenge = f1.getTankGröße() -
                    f1.getFüllstand();
        f1.tanken(tankmenge);
    }
```

# Bob: Datenmodell

Mit get-Methoden:



# Fazit

Das Datenmodell sollte bereitstellen:

- **Methoden** für verschiedene Anwendungen flexibel mit **Parametern**
- **Konstruktor** mit Parametern für Attribute, die zu Beginn gesetzt werden sollen
- **set-Methoden** nur, wenn beliebige Werte möglich sein sollen
- **Methodennamen**, die ausdrücken, was im Anwendungskontext getan wird
- **get-Methoden**, um Attributwerte zu erfragen

# Autor / Quellen

Autor:

- Christian Pothmann (cpothmann.de)
  Freigegeben unter CC BY-NC-SA 4.0, März 2025