# Objektorientiertes Programmieren

Grundlagen

11.04.2020

### OOP vs. Prozedural

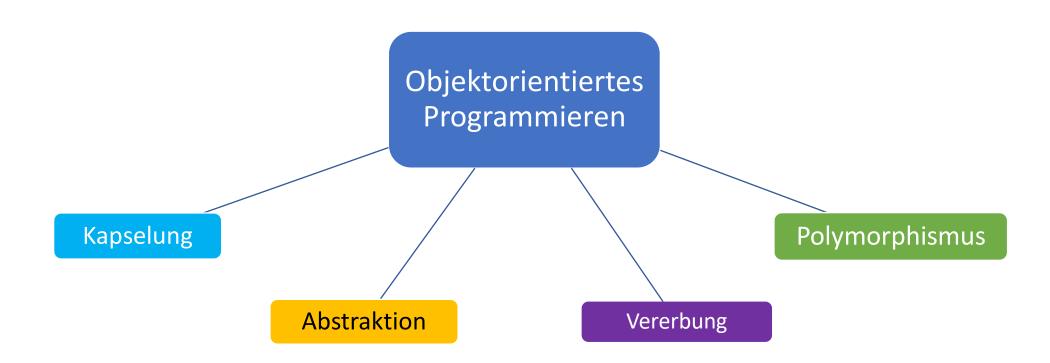
```
let carTank = 50
let consumptionEachKm = 14.1
let pricePerLitre = 1.45
function fillCar(maxTank, pricePerLitre) {
  let cost = maxTank * pricePerLitre
  console.log("You filled your car. Cost: " + cost)
function drive(kilometres, consumptionEachKm) {
  let consumedFuel = kilometres * consumptionEachKm
  console.log("You drove " + kilometres + " and used " +
  consumedFuel + "l of fuel.")
fillCar(carTank, pricePerLitre)
drive(50, consumptionEachKm)
```

Ein Beispiel in JavaScript...

```
let car = {
   carTank = 50,
   consumptionEachKm = 14.1,
   pricePerLitre = 1.45,
   fillCar: function () {
     let cost = this.maxTank * this.pricePerLitre
        console.log("You filled your car. Cost: " + cost)
   },
   drive: function(kilometres) {
     let consumedFuel = kilometres * this.consumptionEachKm
        console.log("You drove " + kilometres + " and used " +
        consumedFuel + "l of fuel.")
     this.carTank -= consumedFuel
   }
}
```

Und dann einmal als Objekt

## Prinzipien des OOP



## Anmerkung

Die objektorientierte Programmierung ist <u>nicht zwingend von einer Programmiersprache abhängig</u>. Sobald man alle Prinzipien versteht, lässt es sich ziemlich leicht auf alle anderen Programmiersprachen übertragen.

Wie sich dieses Paradigma allerdings umsetzen lässt, muss in der jeweiligen Dokumentation der Programmiersprache eingesehen werden

```
let carTank = 50
let consumptionEachKm = 14.1
let pricePerLitre = 1.45
function fillCar(maxTank, pricePerLitre) {
  let cost = maxTank * pricePerLitre
  console.log("You filled your car. Cost: " + cost)
function drive(kilometres, consumptionEachKm)
  let consumedFuel = kilometres * consumptionEachKm
  console.log("You drove " + kilometres + " and used " +
  consumedFuel + "l of fuel.")
drive(50, consumptionEachKm)
```

```
let car = {
  carTank = 50,
                             Properties,
  consumptionEachKm = 14.1.
                             Attribute, Felder
  pricePerLitre = 1.45,
  fillCar: function () {
    let cost = this.maxTank * this.pricePerLitre
    console.log("You filled your car. Cost: " + cost)
  drive: function(kilometres) {
    let consumedFuel = kilometres * this.consumptionEachKm
    console.log("You drove " + kilometres + " and used " +
    consumedFuel + "l of fuel.")
                                    Methoden,
    this.carTank -= consumedFuel
                                   functions
 car.fillCar()
 car.drive(50)
```

access modifier

### "private" Felder

Nur die Klasse selbst kann direkt auf sie zugreifen.

### "public" Methoden

Jede Klasse/jedes Objekt kann auf diese Methode zugreifen.

In diesem Fall: Indirekter Zugriff auf private Felder von außen.

### public void Drive(int kilometres)

Jede Klasse/jedes Objekt kann auf diese Methode zugreifen.

Logik zur Ausführen der Aktion "Fahren".

```
namespace ConsoleApp1
   0 references
   class Car
       /* Nehmen wir an, da sind gerade 501 im Tank
         * und es passen nur 501 rein.
       private double carTank = 50.0;
       private double consumptionEachKm = 0.141;
       0 references
       public double GetConsumptionEachKm()
            return consumptionEachKm;
       0 references
       public double GetCarTank()
            return carTank;
       public void Drive(int kilometres)
            double consumedFuel = kilometres * consumptionEachKm;
           if (consumedFuel > carTank)
               Console.WriteLine($"Not enough fuel available in order to drive {kilometres}km.");
           else
               carTank -= consumedFuel;
               Console.WriteLine($"You drove {kilometres} and used {consumedFuel}l of fuel.");
```

Initialisierung eines Objekts der Klasse "Car"

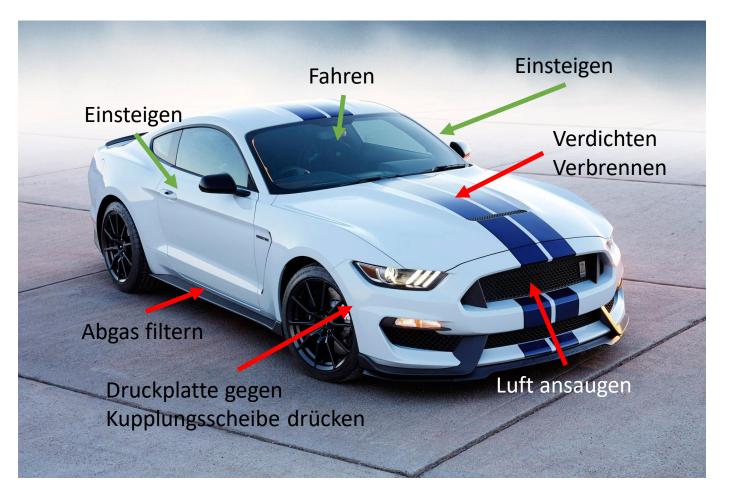
```
Tank: 501
Fuel consumption: 14,099999999999981/100km
You drove 50 and used 7,04999999999999 of fuel.
Not enough fuel available in order to drive 500000km.
```

```
using System;
□namespace ConsoleApp1
     0 references
     class Program
         static void Main(string[] args)
                                                  Zugriff auf klassendefinierte Methoden
             Console.WriteLine("Hello World!");
            Car honda = new Car();
             Console.WriteLine("Car details:");
             Console.WriteLine($"Tank: {honda.GetCarTank()}1");
             Console.WriteLine($"Fuel consumption: {honda.GetConsumptionEachKm() * 100}l/100km");
             honda.Drive(50);
             honda.Drive(500000);
```

### **Fazit**

- Einheits-/Klasseninterne Properties/Felder bleiben "private".
- Zugriff auf Felder erfolgen nur indirekt über sog.
   Getter/Setter-Methoden
- Durch interne Felder kann bspw. die Anzahl an Parameter pro Methode verringert werden → Je weniger, desto besser!

Ziemlich offensichtlich, jeder kann das machen (und sollte das wissen?)



Ziemlich unscheinbar Interne Prozessabläufe Schön zu wissen, aber juckt uns theoretisch nicht.

Verbrauchlogik wird als eine neue "private" Methode extrahiert

```
Oreferences
public void Drive(int kilometres)

{
    double consumedFuel = kilometres * consumptionEachKm;

    if (consumedFuel > carTank)
    {
        Console.WriteLine($"Not enough fuel available in order to drive {kilometres}km.");
    }
    else
    {
        carTank -= consumedFuel;
        Console.WriteLine($"You drove {kilometres} and used {consumedFuel} of fuel.");
    }
}
```

```
public void Drive(int kilometres)
   double consumedFuel = kilometres * consumptionEachKm;
   if (ConsumeFuel(consumedFuel))
        Console.WriteLine($"You drove {kilometres} and used {consumedFuel}l of fuel.");
   els
       Console.WriteLine($"Not enough fuel available in order to drive {kilometres}km.");
private bool ConsumeFuel(double fuelToConsume)
  bool canConsume = (fuelToConsume <= carTank);</pre>
   if (canConsume)
       carTank -= fuelToConsume;
       return true;
   else
       return false;
```

```
private bool ConsumeFuel(double fuelToConsume)
{
   bool canConsume = (fuelToConsume <= carTank);

   if (canConsume)
   {
      carTank -= fuelToConsume;
      return true;
   }
   else
   {
      return false;
   }
}</pre>
```

Eine Methode, die nur innerhalb der Klasse Car verwendet wird und auch nur verwendet werden soll!

Dafür gibt's ja die Methode Drive() und als Person selbst kann man dem Fahrzeug nicht direkt sagen, dass er Sprit verbrauchen soll

### **Fazit**

- Mithilfe von <u>Abstraktion</u> möchte man <u>ein sauberes Interface zur</u> <u>Interaktion des jeweiligen Objekts geben</u>
- Klasseninterne Methoden, die der "Endnutzer" nicht benötigt oder nicht verwenden soll, können mit dem *Access Modifier* "private" von außen unzugänglich machen.
- Dient der Übersicht und besseren Gliederung des Codes (Kein Spaghetti-Code!)

- carTank: double
- consumptionEachKm: double
- + GetCarTank(): double
- + GetConsumptionEachKm(): double
- + Drive(int): void
- ConsumeFuel(double): bool Viewer does not support full SVG 1.1

#### Car

- carTank: double
- consumptionEachKm: double
- + GetCarTank(): double
- + GetConsumptionEachKm(): double
- + Drive(int): void
- ConsumeFuel(double): bool Viewer does not support full SVG 1.1

- carTank: double
- consumptionEachKm: double
- + GetCarTank(): double
- + GetConsumptionEachKm(): double
- + Drive(int): void
- ConsumeFuel(double): bool Viewer does not support full SVG 1.1

#### Car

- carTank: double
- consumptionEachKm: double
- + GetCarTank(): double
- + GetConsumptionEachKm(): double
- + Drive(int): void
- ConsumeFuel(double): bool Viewer does not support full SVG 1.1

### Car2

- carTank: double
- consumptionEachKm: double
- + GetCarTank(): double
- + GetConsumptionEachKm(): double
- + Drive(int): void
- ConsumeFuel(double): bool Viewer does not support full SVG 1.1

- carTank: double
- consumptionEachKm: double
- + GetCarTank(): double
- + GetConsumptionEachKm(): double
- + Drive(int): void
- ConsumeFuel(double): bool Viewer does not support full SVG 1.1

#### Car

- carTank: double
- consumptionEachKm: double
- + GetCarTank(): double
- + GetConsumptionEachKm(): double
- + Drive(int): void
- Consume Fuel (double): bool Viewer does not support full SVG 1.1

#### Car

- carTank: double
- consumptionEachKm: double
- + GetCarTank(): double
- + GetConsumptionEachKm(): double
- + Drive(int): void
- ConsumeFuel(double): bool Viewer does not support full SVG 1.1

#### Car

- carTank: double
- consumptionEachKm: double
- + GetCarTank(): double
- + GetConsumptionEachKm(): double
- + Drive(int): void
- ConsumeFuel(double): bool Viewer does not support full SVG 1.1

#### Car

- carTank: double
- consumptionEachKm: double
- + GetCarTank(): double
- + GetConsumptionEachKm(): double
- + Drive(int): void
- ConsumeFuel(double): bool Viewer does not support full SVG 1.1

#### Car

- carTank: double
- consumptionEachKm: double
- + GetCarTank(): double
- + GetConsumptionEachKm(): double
- + Drive(int): void
- ConsumeFuel(double): bool Viewer does not support full SVG 1.1

- carTank: double
- consumptionEachKm: double
- + GetCarTank(): double
- + GetConsumptionEachKm(): double
- + Drive(int): void
- ConsumeFuel(double): bool Viewer does not support full SVG 1.1.

Car
- carTank: double
- consumptionEachKm: double
+ GetCarTank(): double
+ GetConsumptionEachKm(): double
+ Drive(int): void

### - carTank: double - consumptionEachKm: double + GetCarTank(): double

- Consume Fuel (double): bool

+	Drive(int): void
-	ConsumeFuel(double): bool Viewer does not support full SVG 1.1

+ GetConsumptionEachKm(): double

Car
- carTank: double
- consumptionEachKm: double
+ GetCarTank(): double
+ GetConsumptionEachKm(): double

+ GetConsumptionEachKm(): double
+ Drive(int): void
- ConsumeFuel(double): bool Viewer does not support full SVG 1.1

Car
- carTank: double
- consumptionEachKm: double
+ GetCarTank(): double

· GetGai fallk(). dodbie	
+ GetConsumptionEachKm(): double	
+ Drive(int): void	

- ConsumeFuel(double): bool

Car
carTank: double
consumptionEachKm: double
GetCarTank(): double

carTank: double

+ Drive(int): void

+ GetCarTank(): double

- ConsumeFuel(double): bool Viewer does not support full SVG 1.1

- consumptionEachKm: double

+ GetConsumptionEachKm(): double

ConsumeFuel(double): bool Viewer does not support full SVG 1.1

Car

- consumptionEachKm: double

GetConsumptionEachKm(): double

ConsumeFuel(double): bool Viewer does not support full SVG 1.1

+ GetCarTank(): double

+ GetConsumptionEachKm(): double	+ GetConsumptionEa	achKm(): double
+ Drive(int): void	+ Drive(int): void	
- ConsumeFuel(double): bool Viewer does not support full SVG 1.1	- ConsumeFuel(double viewer does not st	ole): bool pport full SVG 1.1
Car	Ca	ar
- carTank: double	- carTank: double	
- consumptionEachKm: double	- consumptionEachK	m: double
+ GetCarTank(): double	+ GetCarTank(): doub	le
+ GetConsumptionEachKm(): double	+ GetConsumptionEa	achKm(): double
+ Drive(int): void	+ Drive(int): void	

Car
- carTank: double
- consumptionEachKm: double
+ GetCarTank(): double
+ GetConsumptionEachKm(): doubl
+ Drive(int): void
- ConsumeFuel(double): bool Viewer does not support full SVG 1.

- carTank: double

consumptionEachKm: double

- ConsumeFuel(double): bool

+ GetCarTank(): double

- Consur	neFuel(double): bool wer does not support full SVG 1.1
	Car
- carTank	x: double
- consun	nptionEachKm: double
+ GetCar	Tank(): double
+ GetCor	nsumptionEachKm(): double
+ Drive(ii	nt): void

Com
- ConsumeFuel(double): bool Viewer does not support full SVG 1.1
+ Drive(int): void
+ GetConsumptionEachKm(): double
+ GetCarTank(): double
- consumptionEachKm: double
- carTank: double

Car
- carTank: double
- consumptionEachKm: double
+ GetCarTank(): double
+ GetConsumptionEachKm(): double
+ Drive(int): void
- ConsumeFuel(double): bool Viewer does not support full SVG 1.1
<u> </u>

Car
- carTank: double
- consumptionEachKm: double
+ GetCarTank(): double
+ GetConsumptionEachKm(): double
+ Drive(int): void
- ConsumeFuel(double): bool Viewer does not support full SVG 1.1

· ·
Car
carTank: double
consumptionEachKm: double
+ GetCarTank(): double
+ GetConsumptionEachKm(): double
+ Drive(int): void
ConsumeFuel(double): bool Viewer does not support full SVG 1.1

Car
carTank: double
consumptionEachKm: double
GetCarTank(): double
GetConsumptionEachKm(): double
Drive(int): void
ConsumeFuel(double): bool Viewer does not support full SVG 1.1

- carTank: double

+ Drive(int): void

- carTank: double

+ Drive(int): void

- carTank: double

Car

consumptionEachKm: double

+ GetConsumptionEachKm(): double

- ConsumeFuel(double): bool Viewer does not support full SVG 1.1 Car

- consumptionEachKm: double

+ GetConsumptionEachKm(): double

- ConsumeFuel(double); bool Viewer does not support full SVG 1.1

Car

- consumptionEachKm: double

GetConsumptionEachKm(): double

- ConsumeFuel(double): bool Viewer does not support full SVG 1.1

+ GetCarTank(): double

+ GetCarTank(): double

GetCarTank(): double

Car
- carTank: double
- consumptionEachKm: double
+ GetCarTank(): double
+ GetConsumptionEachKm(): double
+ Drive(int): void
- ConsumeFuel(double): bool

Viewei does not support full 575 1.1
Car
- carTank: double
- consumptionEachKm: double
+ GetCarTank(): double
+ GetConsumptionEachKm(): double
+ Drive(int): void
- Consume Fuel (double): bool

- ConsumeFuel(double): bool Viewer does not support full SVG 1.1
Car
- carTank: double
- consumptionEachKm: double
+ GetCarTank(): double
+ GetConsumptionEachKm(): double
+ Drive(int): void
- ConsumeFuel(double): bool Viewer does not support full SVG 1.1

Car	Car
- carTank: double	- carTank: double
- consumptionEachKm: double	- consumptionEachKm: double
+ GetCarTank(): double	+ GetCarTank(): double
+ GetConsumptionEachKm(): double	+ GetConsumptionEachKm(): double
+ Drive(int): void	+ Drive(int): void
- ConsumeFuel(double); bool Viewer does not support full SVG 1.1	- ConsumeFuel(double): bool Viewer does not support full SVG 1.1

	Car
- carTank:	double
- consum	ptionEachKm: double
+ GetCar	Tank(): double
+ GetCon	sumptionEachKm(): double
+ Drive(in	t): void
- Consum	eFuel(double): bool

	Car
- carTank: do	uble
- consumptio	onEachKm: double
+ GetCarTan	k(): double
+ GetConsur	mptionEachKm(): double
+ Drive(int): v	roid
- ConsumeF	uel(double): bool

	Car
- carTanl	c: double
- consun	nptionEachKm: double
+ GetCa	rTank(): double
+ GetCo	nsumptionEachKm(): double
+ Drive(i	nt): void
- Consu	meFuel(double): bool

Car - carTank: double	Car - carTank: double	Car - carTank: double	Car - carTank: double	Car - parTank: double	Car - carTank: double	Car - cerTenk: double	Car - carTank: double	Car - carTank: double	Car - carTank: double	Car - parTank: double	Car - carTank: double	Car - carTank: double
consumptionEachiúm: double     GetCarTank(): double	consumptionEachKm : double     GetCarTank(): double	consumptionEachKm: double     GetCarTank(): double	consumptionEachKm: double      GetCarTank(): double	consumptionEachKm: double     GetCarTank(): double	consumptionEachKm: double     GetCarTank(): double	consumptionEachkim: double     BetCarTenk(): double	consumptionEachKm : double     GetCarTank() : double	consumptionEachKm: double     GetCarTank(): double	consumptionEachKm: double     GetCarTank(): double	consumptionEachKm: double     GetCarTank(): double	consumptionEachKm: double     GetCarTank(): double	consumptionEachlism: double     GetCarTank(): double
+ GetConsumptionEachKm(): double	+ GetConsumptionEachKm(): double	+ GetConsumptionEachKm(): double	+ GetConsumptionEachKm(): double	+ GetConsumptionEschKm(): double	+ GetConsumptionEachKm(): double	GetConsumptionEachKm(): double	+ GetConsumptionEachKm(): double	+ GetConsumptionEachKm(): double	+ GetConsumptionEachKm(): double	+ GetConsumptionEachKm(): double	+ GetConsumptionEachKm(): double	+ GetConsumptionEachKm@: double
Drive(int): sold     Consume Euclideachie's book	Delve (Int): sold     Description of English and Interest Int	Drive(int): void     Consumo Evolidos bloi: bool	Drive(int): sold     Dressum of publishes been	+ Drive(Int): void	Drive(int): void     Departure of publishments in our	Drive(int): sold     Dressum o Familian which hand	Drive(int): vold     Dressum of unitide which hand	Delive(int): sold     Consum of solid to which insel	Drive(int): void     Consumo Euplidouble: book	Drive(int): void     Consume Euplides blok book	+ Drive(int): void	Drive (int): sold     Consum of solidouble's book
- CongumeEvel(deshie), heplays 1.1	- Congress First daybles hep avairs	- Congume Explide ubbs. Applaya + +	- Congume Eval Mayble Shaplays + +	- Congume Exalida yallah Applaya 1.1	- Company Eyas (Squb)e); her sys ++	- Congress Francisco Aguada Agua sua 11	- Congunative idea by 1 sys 11	- Congrant Everificant big bases 11	- Congung Explidit Uplay Penlava 1.1	- Congume Explide ubje), Penlava 1.1	- Congume Exaltique les Jestiava 1.1	- Consume Evel (dayble), hear siva 1.1
- carTank: double	- carTank; double	- carTank: double	- carTank: double	- parTank; double	Car - carTank: double	- carTank; double	Car - carTank: double	- carTank: double	- parTank: double	- carTank: double	- carTank: double	Car - carTank: double
- consumptionEachKm: double	- consumptionEachKm : double	- consumptionEachKm: double	- consumptionEachKm : double	- consumptionEachKm: double	- consumptionEachKm : double	- consumptionEachKm : double	- consumptionEachKm : double	- consumptionEachKm : double	- consumptionEachKm: double	- consumptionEachKm: doubte	- consumptionEachKm: double	- consumptionEachKin : double
+ GetCarTank(): double	GetCarTanki): double	GetCarTank(): double	GetCarTanki): double	GetCarTank(): double	GetCarTank(): double	GetCarTanki): double	GetCarTank(): double	GetCarTank() double	GetCarTank(): double	GetCarTank(): double	+ GetCarTank(): double	+ GetCarTank(): double
+ GetConsumptionEachKm(): double	+ GetConsumptionEachKm(): double	+ GetConsumptionEachKm(): double	+ GetConsumptionEachKm(): double	+ GetConsumptionEachKm(): double	+ GetConsumptionEachKm(): double	+ GetConsumptionEachKm(): double	+ GetConsumptionEachKm(): double	+ GetConsumptionEachKm(): double	+ GetConsumptionEachKm(): double	+ GetConsumptionEachKm(): double	+ GetConsumptionEachKm(): double	+ GetConsumptionEachKm@ double
+ Drive(int): sold	+ Drilve(int): sold	+ Drive(Int): void	+ Drive(int): sold	+ Drive(Int): void	+ Drilvs(int): vold	+ Drilve(int): sold	+ Drive(Int): vold	Drilve(Int): volid	+ Drive(Int): void	+ Drive(Int): void	+ Drive(int): void	+ Drive (int): sold
- Dengama Euglidenskiek Petri sys 1.1	- Congress See Hauble had sive 11	- Consume Sualida volte, her sya 1.1	- Congress Eval dauble), beal sys 1.1	- Congress Seel day blei, hen sya 1.1	- Congress of the second of th	- Company 5x2, 48 4546, 582 5v2 11	- Company Eval (Anybie) Peols vo 14	- Consume Evalida uble): hep lava 11	- Consume Exalida volle), her lava 1.1	- Consume Exalida vible Deal sys 1.1	- Congume Explide ubjet best sive 1.1	- Congress See (de selections - 1 - 1
Car	Car	Car	Car	Car	Car	Car	Car	Car	Car	Car	Car	Car
- cerTank: double - consumptionEachign: double	- carTank: double - consumptionEachKin: double	- carTank: double - consumptionEachKm: double	- cerTank: double - consumotionEachKm: double	- carTank: double - consumotionEachKm: double	- carTank: double - consumptionEachion: double	- carTank: double - consumptionEachKm: double	- perTank: double - ponsumotionEachKm: double	- carTank: double - consumptionEachKm: double	- carTank: double - consumotionEachKm: double	- carTank: double - consumptionEachKm: double	- carTank: double - consum otionEachKm: double	cerTank: double     consumptionEachion: double
+ GetCarTanki) double	+ BetCarTankii: double	+ GetCarTank/t double	BetCarTankii: double	+ GetCarTankft: double	DetCarTank/it double	GetCarTankii double	GetCarTank/i. double	GetCarTank/i. double	+ BetCarTankft double	+ GetCarTank/I: double	+ GetCarTankft: double	+ GetCarTanki) double
+ GetConsumptionEachKm(): double	GetConsumptionEachKm(): double	+ GetConsumptionEachKm(): double	DetConsumptionEachKm(): double	+ GetConsumptionEachKm(): double	GetConsumptionEachKm(): double	+ GetConsumptionEachRm(): double	GetConsumptionEachKm(): double	GetConsumptionEachKm(): double	+ GetConsumptionEschKm(): double	+ GetConsumptionEachKm(): double	+ GetConsumptionEachKm(): double	GetConsumptionEachKm(): double
+ Drive(int): sold	+ Drive(int): void	+ Drive(int): void	+ Drive(int): void	+ Drive(int): void	+ Drive(int): vold	+ Drilve(int): vold	Drive(int): vold	+ Drilve(int): volid	+ Drive(int): void	+ Drive(Int): void	+ Drive(Int): void	+ Drive (int): sold
- Donguma Sugi danahlari hari sus 1.1	- Domay Tue Stall Sayble), Peplays 11	- Consume Exelide uble), beel sys 4.4	- Dongume Eve. (guble): Pgol <sub>3/05-1-1</sub>	- Congressell dauble), her lava 1.1	- Consume Eva (dauble): Peol <sub>siva 11</sub>	- Domarting Englished And 1 1	- Consume Exalida y bigs her laws 1.1	- Conguna Exalida y blas Agail siva 1 1	- Consume Evel dauble), benisva 1.1	- Congume Equil double   beel sys ++	- Conguitte Seel (dayble), her lava 1.1	- Consume Exalidade la Septiana 11
Car	Car	Car	Car	Car	Car	Car	Car	Car	Car	Car	Car	Car
- carTank: double	- carTank: double	- parTank: double	- carTank: double	- parTank; double	- carTank: double	- carTank: double	- carTank: double	- carTank: double	- parTank: double	- parTank: double	- parTank: double	- carTank: double
consumptionEachiúm: double     GetCarTankii: double	consumptionEachKm : double     GetCarTanki: double	- consumptionEachKm: double	consumptionEachKm: double     GetCarTackN: double	consumptionEachKm: double     GetCarTankfit double	consumptionEachKm: double     GetCarTank(): double	- consumptionEachKm : double	consumptionEachKm : double     GetCarTank(): double	consumptionEachKm : double     GetCarTank() : double	- consumptionEachKm: double	consumptionEachKm: double     GetCarTankft: double	consumptionEachKm: double     GetCarTankf: double	consumptionEachtim : double     GetCarTankii : double
GetCarTank(): double     GetConsumptionEachKm(): double	GetContank(): double     GetConsumptionEachKm(): double	GetCarTank(): double     GetConsumptionEachKm(): double	GetCarTank(): double     GetCansumptionEachKm(): double	GetConsumptionEachKm(): double	GetCarTank(): double     GetConsumptionEachK(m(): double	GetCarTank(): double     GetConsumptionEachKm(): double	GetCarTank(): double     GetCangumptionEachFim(): double	GetCarTank(): double     GetConsumptionEachKm(): double	GetCarTank(): double     GetConsumptionEachKm(): double	+ GetCarTank(): double + GetConsumptionEachKm(): double	GetCarTank(): double     GetConsumptionEach((m)): double	GetCarTank(): double     GetConsumptionEachKm(): double
+ Drive(int): sold	+ Drilve(int): solid	+ Drive(Int): vold	+ Drilve(int): solid	+ Drive(int); vold	+ Drive(int): sold	+ Drilve(int): vold	+ Drilve(Int): vold	+ Drilve(int): vold	+ Drive(int); vold	+ Drive(Int): void	+ Drive(Int): vold	+ Drive(int): solid
- Donguma Svalida vible à Peri sus 1.1	- Congulative dayble been swa 11	- Congume Eyelidayble), beel sys 4.4	- Congume Evel (dauble); beol sys ++	- Consume Exelida yelle), bealloys 1.1	- Consume Eval (dauble), heal sive 11	- Congulta Eval dayble beel siva + +	- Company Eval (dayble), beoless 1 4	- Congress See Hauble Applicant	- Consume Exelida uble), beal sva 1.1	- Congume Equilida ubile), bearings 1.1	- Consume Exelida yelle), bealises 1.1	- Denguma Euskida (bija); hepi sus 1.1
Car	Car	Car	Car	Car	Car	Car	Car	Car	Car	Car	Car	Car
carTank: double consumetonEachKm: double	- carTank: double - consumptionEachKm: double	- carTank: double - consumotionEachKin: double	- carTank: double - consumptionEachKm: double	- carTank: double - consumptionEachKm: double	- carTank: double - consumotionEachKm: double	- carTank: double - consumptionExchilin: double	- carTank: double - consumotionEachKm: double	- carTank: double - consumptionEachKm: double	- carTank: double - consumotionEachKm: double	- carTank: double - consum otionEachKin: double	- carTank: double - consumotionEachKm: double	- carTank: double - consumptionEachKm: double
OetCarTank(): double	+ OetCarTank(): double	+ DetCarTank(): double	+ OetCarTank(): double	+ OetCarTank(): double	+ OetCarTank(): double	+ GetCarTack() double	OetCarTank(): double	+ OetCarTank(): double	+ OetCarTank(): double	+ OetCarTanki): double	+ OetCarTank(): double	+ OetCarTank(): double
· GetConsumptionEach(m(): double	+ GetConsumptionEachKm(): double	+ GetConsumptionEschKm(): double	DelConsumptionEachKm(): double	+ GetConsumptionEachKm(): double	+ GetConsumptionEachKm(): double	+ GetConsumptionEachKm(): double	GetConsumptionEschKm(): double	GetConsumptionExchKm(): double	+ DelConsumptionEschKm(): double	+ DetConsumptionEachKm(): double	+ GetConsumptionEachKm(): double	+ SetConsumptionEschKm(): double
Drive(Int): vold	+ Drive(int): void	+ Drive(int): sold	+ Drive(int): vold	+ Drive(int): sold	+ Drive(Int): void	+ Drive(int): sold	+ Drive(Int); void	+ Drive(int): sold	+ Drive(Int): vold	+ Drilve(int): sold	+ Drive(Int): void	+ Drive(Int); void
Consuma Explidavidal, benisus 4.4	- Congress See Haubles had see 1.1	- Congume Eval idauble); heal siva + +	- Consume Ever Hauble's had sys 11	- Congress of the ships her size 1.1	- Consume Spellidguble), benisya 1.1	- Congumative identities her laws 1.1	- Congame Seel dauble), best sys 1.1	- Dong your Eve, dauble) beolesses as	- Congume Explidentile), beel sys 1.1	- Congume Eye, dauble), bear sive 1.1	- Congume Spell dauble, health and 1.1	- Congume Evelidanble), benisya 4.4
Car	Car	Car	Car	Car	Car	Car	Car	Car	Car	Car	Car	Car

Delive (Int): sold     Consymptical dauble ( beel sys. s.s.)	Delive(Int): void     Consume Evalua void: hepisys 1.1	Drive (Int): void     Consume Sual (dauble), benis void	Delive(Int): void     ConsumeEpalidauble): heplays 4.4	Drive (Int): void     Consume Exell (Iguble), beel size 1,1	Develont): void     Consum Exalibrable): Pepisos 1.1	Delive(int): sold     Congugge Eye, dauble): bgolsvs 4.4	Delive(Int): sold     Delive(Int): sold     Delive(Int): sold	Delive (Int): sold     Consume Eval, dauble): Appliance 1.1	Drive(Int): void     ConsumeSeal/dguble); beelsvoid	Drive (Int): void     Congume Euglidguble); beels, value	Drive (Int): void     Congum e Epal/de ubie), beel sva 4,4	Drive (Int): seld     Congrupa Exal/dayble & Peorsons 1.1	Drive(Int): void     Consume Exel(dauble), beel sys 4.4
Car	Car	Car	Car	Car	Car	-	Car	Car	Car	Car	Car	Car	
- carTank: double	- parTank; double	- carTank; double	- carTank: double	- parTank; double	- carTank: double	- carTank: double	- carTank; double	- carTank: double	- carTank: double	- parTank: double	- parTank; double	- carTank: double	- carTank: double
- consumptionExchitim: double	- consumptionEachKm : double	- consumptionEachKm: double	- consumptionEachKm : double	- consumptionEachKm: double	- consumptionEachKm : double	- consumptionEachKm: double	- consumptionEachKm : double	- consumptionEachKm : double	- consumptionEachKm: double	- consumptionEachKm: double	- consumptionEachKm: double	- consumptionExchiGn: double	- consumptionEachKm: double
+ GetCarTank(): double	DetCarTank(): double	+ GetCarTank(): double	GetCarTanki): double	+ GetCarTank(): double	DetCarTank(): double	GetCarTankiir double	+ GetCarTank(): double	+ GetCarTank(): double	+ GetCarTank(): double	+ GetCarTank(): double	+ GetCarTank(): double	+ GetCarTank(): double	GetCarTank/t double
+ GetConsumptionEachKm(): dauble	+ GetConsumptionEachKm(): double	+ GetConsumptionEachKm(): double	+ GetConsumptionEachKm(): double	+ GetConsumptionEachKm(): double	+ GetConsumptionEachKm(): double	+ GetConsumptionEachHm(): double	+ GetConsumptionEachKm(): double	+ GetConsumptionEachRim(): double	+ GetConsumptionEachKm(): double	+ GetConsumptionEachKm(): double	+ GetConsumptionEachKm(): double	+ GetGonsumptionEachKm(): double	+ GetConsumptionEachKm(): double
Drilve (int): sold	Drilve(int): wold	+ Drive(int): void	+ Drive(int): sold	+ Drive(int); void	Drilve(int): sold	+ Drilve(Int): volid	Drilve(int): solid	+ Drilve(int): solid	+ Drive(Int): vold	+ Drive(Int); void	+ Drive(Int): void	+ Drive(int): sold	+ Drive(Int): void
- ConsumeEvel/devalue beefore 1.1	- Congumy Eya, dayble), Agal sys 1.1	- Congress Seel dauble Deslava 11	- Congress Sauble And sva 11	- Congume Explide uble), her lava 4.4	- Congress See dayble See ava 11	- Company Eval dayble shed avair	- Congress Shall day ble), hep-lays 11	- Congregation of the cong	- Congume Explide ubje), bestave 1.1	- Congume Spellidguble), Penlayo 1.1	- Congume Explide ubje), benlava 1.1	- Conguma Euglida (bija), Papilisus 1.1	- Congress Seel (dayble), Penlaya 1.1
- carTank: double	- carTank: double	Car - carTank: double	- carTank: double	- carTank: double	- carTank: double	Car - carTank: double	- carTank: double	- carTank: double	- carTank: double	- carTank: double	- carTank: double	- carTank: double	- carTank: double
- consumptionEachRim: double	- consumptionEachKm: double	- consumptionEachKm : double	- consumptionEachKm : double	- ponsumptionEachKim: double	- consumptionEachKm: double	- consumptionEachKim: double	- consumptionEachKm: double	- consumptionEachKm: double	- consumptionEachKm: double	- consumptionEachKm: double	- consumptionEachKm: double	- consumptionEachKm: double	- consumptionEachKm : double
+ OetCarTank(); double	+ OetCarTanki); double	OetCarTank(): double	OetCarTank(): double	+ OetCarTank(): double	+ OetCarTank(): double	GetCarTack() double	+ OetCarTank(): double	DetCarTank(): double	+ OetCarTank(): double	OetCarTank(): double	+ GetCarTank(): double	+ OetCarTank(): double	GetCarTankii double
+ GetConsumptionEachKm(): double	GetConsumptionEachKm(): double	+ GetConsumptionEachKm(): double	GetConsumptionEachKm(): double	+ GetConsumptionEachKm(): double	+ GetConsumptionEschKm(): double	+ GetConsumptionEachKm(): double	+ GetConsumptionEachKm(): double	+ GetConsumptionEachKm(): double	+ GetConsumptionEschKm(): double	+ GetConsumptionEachKm(): double	+ GetConsumptionEschKm(): double	+ GetConsumptionEschKm(): double	+ GetConsumptionEachKm(): double
+ Drive(Int): vold	+ Drive(int): vold	+ Drive(int): sold	+ Drive(int): sold	+ Drive(int): sold	+ Drive(Int): vold	+ Drive(int): void	+ Drive(Int): vold	+ Drive(int): sold	+ Drive(int): void	+ Drilve(int): vold	+ Drive(int): void	+ Drive(int): void	+ Drive(int): vold
- Consume Suplidential PER sug 4.4	- Congress See Anable See 505 1.1	- Congress sandles had avair	- Congume Seeliday blacked ava 1.1	- Congress Supplied Paril 202 1.1	- Congress Seel Hauble), her lave 4.4	- Congumatival ida vibila i heal sus 1.1	- Congume Seel Hauble), Penlava 4.4	- Congume Exa, Haubleh Papilava 1.1	- Congume Seel dauble), heal swa 1.1	- Consume Eval dauble), baptava 1.1	- Congume Seel (dauble), Penlava 1.1	- Congume Evel signified beet ava 1.1	- Congress of the section of
Car	Car	Car	Car	Car	Car	Car	Car	Car	Car	Car	Car	Car	Car
- carTank: double	- carTank: double	- carTank: double	- parTank; double	- carTank: double	- parTank: double	- carTank: double	- parTank: double	- carTank: double	- parTank; double	- carTank; double	- parTank: double	- carTank: double	- carTank: double
- consumptionEachKm: double	- consumptionEachKm: double	- consumptionEachKm: double	- consumptionEachKm : double	- consumptionExchión: double	- consumptionEachKm: double	- consumptionEachkin: double	- consumptionEachKm: double	- consumptionEachKm : double	- consumptionEachKm: double	- consumptionEachKm : double	- consumptionEachKm: double	- consumptionEachKm: double	- consumptionEachKm: double
+ GetCarTank(): double	+ GetCarTanki): double	GetCarTank(): double	GetCarTank(): double	+ GetCarTank(): double	+ GetCarTank(): double	OetCarTank() double	+ GelCarTank(): double	+ GelCarTank(): double	+ GelCarTank(): double	+ GetCarTank(): double	+ GetCarTank(): double	+ GetCarTank(): double	DetCarTank(): double
+ GetConsumptionEachKm (): double	+ GetConsumptionEachNm(): double	+ GetConsumptionEachRm(): double	+ GetConsumptionEachRm(): double	+ GetConsumptionEachKm(): double	+ GetConsumptionEachKm(): double	+ GetConsumptionEachKm(): double	+ GetConsumptionEachKm(): double	+ GetConsumptionEachKm(): double	+ GetConsumptionEachKm(): double	+ GetConsumptionEachKm(): double	+ GetConsumptionEachKm(): double	+ GetConsumptionEachKm(): double	GetConsumptionEachKm(): double
Drive(int): void     Consume Exel/dayble(cherless s.s.)	Delive(int): vold     ConsumeExe, dauble); beolstys 1.1	Delve(int): vold     Congutte Seelidayble): Appliance 1.1	Delive(int): vold     Consum Sea, dauble): heptisms 1, 1	Drive(int): sold     ConsumeSee(identitle): heptises 1.1	Drive(Int): vold     Conquite Spalldayble1. Peals vo. 1.1	Drive (int): sold     Denguma Susi (da sola), hepitana 1,1	Drive(Int): void     Congagge Specifique best post sus 4.4	Drive(int): vold     Consume Fuellinushle's hool	Drive(Int): void     CongaggeEggl(Squb)e); begs/syc 1, 1	Drive(int): sold     ConsumeFuelifouble's hool	Drive(Int): vold     CongumeSpell(dauble), Applique 4.1	Drive(Int): void     Conga(I)eSpal(Mayble), heplays 4.1	Drive(int): void     Consuma Exalida uble: Peols vo. 1.1
The service applicable and the	THE WAY STREET SUSSESSED S	The state of the s	TEMPORE STREET SUSSIES	The second section of the second seco	The state of the s	The rest of the re	1500 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1	- Consume Eval (Squb)e); her sva + +	THE WAY ORDER FOR SECURIOR SAVO 1.1	- Congress Spall stauble); hearland 1.1		The second secon	THE WAY STEEL AND RESIDENCE TO SUICE 1 1
Car	Car	Car	Car	Car	Car	Car	Car	Car	Car	Car	Car	Car	Car
- carTank: double - consumptionEachKm: double	- carTank: double - consumptionEachKm: double	- carTank: double - consumptionEachKm: double	- carTank: double - consumptionEachKm: double	- carTank: double - consumptionEachlún: double	- carTank: double - consumptionEachKm: double	- carTank: double - consumptionEschKm: double	- carTank; double - consumptionEachKm; double	- carTank: double - consumptionEachKm: double	- carTank: double - consumptionEachKm: double	carTank: double     consumptionEachKm: double	- carTank: double - consumptionEachKm: double	- parTank: double - consumptionEachKm: double	- carTank: double - consumptionEachKm: double
			_			GetCarTank(): double							
+ GetCarTank(): double + GetConsumptionEachi(m(): double	GetCarTank(): double     GetConsumptionEachKm(): double	GetCarTank(): double     GetConsumptionEachKm(): double	GetCarTank(): double     GetCansumptionEachKm(): double	+ GetCarTank(): double + GetConsumptionEachKm(): double	GetCarTank(): double     GetConsumptionEachKm(): double	GetConsumptionEachKm(): double	GetCarTank(): double     GetConsumptionEachKm(): double	GetCarTank(): double     GetConsumptionEachKm(): double	GetCarTank(): double     GetConsumptionEachKm(): double	GetCarTank(): double     GetConsumptionEachKm(): double	GetCarTank(): double     GetConsumptionEachKm(): double	GetCarTank(): double     GetConsumptionEach((m)): double	GetCarTank(): double     GetCansumptionEachKm(): double
+ Drive(Int): vold	+ Drilve(Int): void	+ Drive(int): sold	+ Drilve(int): sold	+ Drive(int): sold	+ Drive(Int); void	+ Drive(int): sold	+ Drive(Int): void	+ Drive(int): void	+ Drive(Int); void	+ Drive(Int): vold	+ Drive(int); void	+ Drive(Int); void	+ Drive(int): sold
- Consume Exel/double), bool ava 4.4	- Congunativalida (b) a) beal ava 1.1	- Congume Evalida ubjek heal sva 11	- Consume Everlidayble) Applicus 1.1	- DengymyEvalldayble\\ Pg9fava + +	- Congume Explide ubility best sys 1.1	- Consume Evel de vide à Perlans 1.1	- Consume Egglidguble), beel sva 1.1	- Consume Eyel (dauble), heal sys 1.1	- Congume Explidentie), heat avail.	- Congume Eval dauble): heal avairs	- Congume Explide ubje), heel sva 1.1	- Congume Explide the laborator	- Consume Evalidayble), heal sys 1.1
Car	Car	Car	Car	Car	Car	Car	Car	Car	Car	Car	Car	Car	Car
- carTank: double	- carTank: double	- carTank: double	- carTank: double	- carTank: double	- carTank: double	- carTank: double	- carTank: double	- carTank: double	- carTank: double	- carTank: double	- carTank: double	- carTank: double	- carTank; double
- consumptionEachKm: double	- consumptionEachKm : double	- consumptionEachKm : double	- consumptionEachKm : double	- consumptionEachKin : double	- consumptionEachKm: double	- consumptionEachKm: double	- consumptionEachKm: double	- consumptionEachKm : double	- consumptionEachKm: double	- consumptionEachKin : double	- consumptionEachKm: double	- consumptionEachKm: double	- consumptionEachKm: double
+ OetCarTank(): double	OetCarTank(): double	DetCarTank(). double	OetCarTenk(): double	+ OetCarTank(): double	+ OetCarTank(): double	+ GetCarTank(): double	+ GetCarTank(): double	OetCarTenk(): double	GetCarTank(): double	+ OetCarTank(): double	+ DetCarTank(): double	+ OetCarTank(): double	GetCarTank(): double
+ GetConsumptionEachKm(): double	DetConsumptionEachKm(): double     Deterint: sold	GetConsumptionEachKm(): double     Drive(int): void	BetConsumptionEachKm(): double     Drive(int): void	+ BetConsumptionEachKm(): double + Detroint: sold	SetConsumptionEschKm(): double     Drive(Int): void	+ GetConsumptionEachKm@ double	+ GetConsumptionEschKm(): double + Drive(Int): void	GetConsumptionEachKm(): double     Drive(int): void	GetConsumptionEachKm(): double     Drive(Int): vold	GetConsumptionEachKm(): double     Deterint: sold	GetConsumptionEachKm(): double     Drive(Int): vold	DetConsumptionEschKm(): double     Drive(int): void	+ GetConsumptionEachKm(): double
+ Drive(Int): vold - ConsumeRuel/drivible/: hond						Drive (int): sold     Consume Fund (double): book					- Congume Systide ubits   book system		Delve(int): sold     Concurs of sublides block
- Congume Eyel (develor), Per Isus 4.4	- Congress dayble approved in	- Congress sayble and size 11	- Coraymy Eye, dayble, ppo sys 11	- Congress Supil dauble): Peel sus 1.1	- Congume Exelidad black bear 5.va 1.1	- Consume Eval day black hear success	- Congume Eyelidguble), benisya 1.1	- Consume Eval (Squb)e); her sys + +	- Conquite Exalida vol. 9 pen 5 vo. 1.1	- Congress supplies been size 11	THE WAY STREET STATE OF THE STA	- Congume Suelida volle), benisus 1.1	- Consume Eye (Souble), high swa 1.1

GetCarTank(): double
 GetConsumptionEachKm(): double
 Drive(int): void
 ConguttaSpalidauble; Poolsys 1.1.

- consumptionEachtim double

- OstEx\*Transit; Souble

- OstEx\*Transit; Souble

- Desconding Selection; Souble

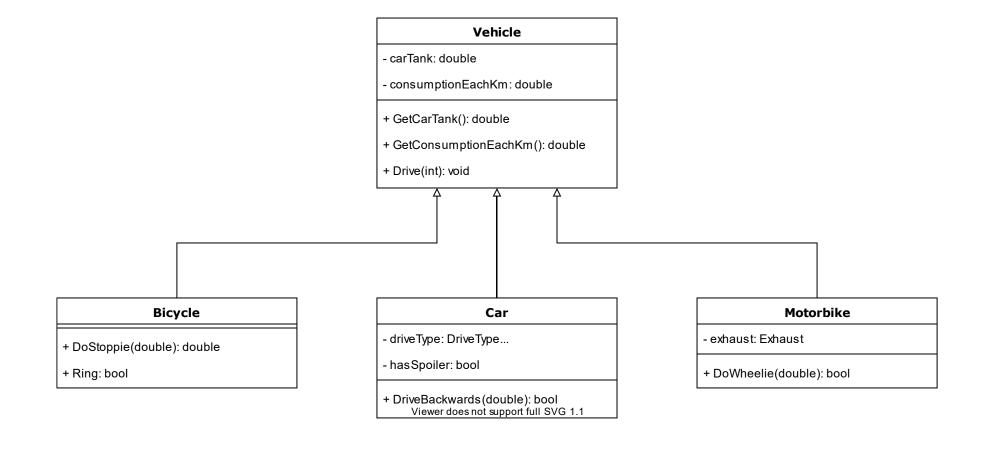
- Conguing Selection; Selection

- Car

- parties; Goodle

- OstEx\*Transit; Selection

Ob du behindert bist?



```
∃namespace ConsoleApp1
                                  ... umbenannt in...
        /* Nehmen wir an, da sind gerade 501 im Tank
         * und es passen nur 501 rein.
                                                   "private" zu "protected"
       private double carTank = 50.0;
       private double consumptionEachKm = 0.141;
       public double GetConsumptionEachKm()
           return consumptionEachKm;
     Oreferences
public double GetCarTank(). entfernt und Beschreibung aus Program.cs
extrahiert...
           if (consumedFuel > carTank)
               Console.WriteLine($"Not enough fuel available in order to drive {kilometres}km.");
               carTank -= consumedFuel:
               Console.WriteLine($"You drove {kilometres} and used {consumedFuel}l of fuel.");
```

```
2 references
class Vehicle
   /* Nehmen wir an, da sind gerade 501 im Tank
     * und es passen nur 501 rein.
                                                       Konstruktor
   protected double carTank;
                                           Dient zur Erzeugung eines Objekts aus der
   protected double consumptionEach100Km; dementsprechenden Klasse
   1 reference
   public Vehicle(double _carTank, double _consumptionEach100Km)
        carTank = carTank;
                                                          ..virtual"
        consumptionEach100Km = consumptionEach100Km;
                                            Schlüsselwort muss verwendet werden,
                                            wenn erbende Klassen diese Methode
   3 references
                                            überschreiben dürfen
   public virtual void Describe()
        Console.WriteLine("Car details:");
        Console.WriteLine($"Tank: {carTank}1");
        Console.WriteLine($"Fuel consumption: {consumptionEach100Km}l/100km");
```

### Konstruktor

Besonderheit bei C#: "super()"-Aufruf in-line mit Konstruktor

```
Java:
public Car(...) {
   super(...)
}
```

### "Neue" Klasse Car

### Vererbung

C#:

class <KLASSE> : <ÜBERKLASSE>

Java:

class <KLASSE>
extends <ÜBERKLASSE>

Python:

class KLASSE(ÜBERKLASSE):

**Attribute, Felder** Klasseneigene Felder

```
Oreferences
class Program
{
    Oreferences
    static void Main(string[] args)
    {
        Console.WriteLine("Hello World!");

        Car honda = new Car(50.0, 14.1, "Honda Civic", false);

        honda.Describe();

        honda.Drive(50);

        honda.Drive(5000000);
    }
}
```

```
Hello World!
Car details:
Tank: 50l
Fuel consumption: 14,11/100km
You drove 50 and used 7,04999999999999 of fuel.
Not enough fuel available in order to drive 500000km.
```

### **Fazit**

- Erstellung von einer generalisierten Klasse möglich
- Vererbung überträgt <u>Konstruktor + in der Überklasse definierte</u> <u>Methoden</u>
- Überklasse kann auch eine <u>abstrakte Klasse</u> sein (Nicht zu verwechseln mit Abstraktion! – eine abstrakte Klasse ist eine Klasse, die als "Schablone" oder Vorschrift an Klassen dient, die davon erben. <u>Man kann davon keine Objekte erzeugen</u>)

```
Oreferences
class Program
{
    Oreferences
    static void Main(string[] args)
    {
        Console.WriteLine("Hello World!");

        Car honda = new Car(50.0, 14.1, "Honda Civic", false);

        honda.Describe();

        honda.Drive(50);

        honda.Drive(500000);
    }
}
```

```
Hello World!
Car details:
Tank: 50l
Fuel consumption: 14,11/100km
You drove 50 and used 7,04999999999999 of fuel.
Not enough fuel available in order to drive 500000km.
```

Beispiel: Describe()

### Klasse Car

```
Hello World!

Car details:

Tank: 50l

Fuel consumption: 14,11/100km

Model name: Honda Civic

Has spoiler?: No

You drove 50 and used 7,0499999999999999 of fuel.

Not enough fuel available in order to drive 500000km.
```

Beispiel: DriveBackwards()

### Klasse Car

```
Oreferences
public bool DriveBackwards(double _distance)
{
    double consumedFuel = _distance * (consumptionEach100Km / 100.0);

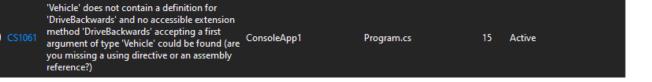
    if (base.ConsumeFuel(consumedFuel))
    {
        Console.WriteLine($"You drove {_distance} backwards and used {consumedFuel}l of fuel.");
        return true;
    }
    else
    {
        Console.WriteLine($"Not enough fuel available in order to drive {_distance}km backwards.");
        return false;
    }
}
```

```
0 references
    class Program
        0 references
        static void Main(string[] args)
           Console.WriteLine("Hello World!");
           Car honda = new Car (50.0, 14.1, "Honda Civic", false);
            honda.Describe();
            honda.DriveBackwards(15.0);
Hello World!
Car details:
Tank: 501
Fuel consumption: 14,11/100km
Model name: Honda Civic
Has spoiler?: No
You drove 15 backwards and used 2,11499999999999981 of fuel.
```

Beispiel: DriveBackwards()

```
Oreferences
class Program
{
    Oreferences
    static void Main(string[] args)
    {
        Console.WriteLine("Hello World!");

        Vehicle honda = new Car(50.0, 14.1, "Honda Civic", false);
        honda.Describe();
        honda DriveBackwards(15.0);
}
```



### **Fazit**

- Durch Überschreiben von Methoden kann die Logik jeder Methode, die von einer Überklasse geerbt wird, verändert werden
- Vor allem: *Don't Repeat Yourself* = Weniger Arbeit