


| | | | | |
|---|-------------------------|---------------|---------------------------------------|------|
|  | Klasse | Information | Thema Konstruktoren mit Parametern | |
| | Lehrer Oliver Kreuer | Lernfeld 6 | Datum | Name |

Konstruktor mit Parametern

Man kann Konstruktoren mit beliebigen Parameterlisten versehen. Damit lassen sich beim Erzeugen eines Objekts Werte für die Initialisierung der Daten-Member übergeben.

Beispiel:


```
class Fahrzeug
{
    char marke[20];
    int kilometerstand;
};
```

1. Konstruktor mit Parameterliste

Prototyp: Fahrzeug(char *, int);

Implementation: Fahrzeug::Fahrzeug(char *markenname, int kmStand)
{
 strcpy(marke, markenname);
 kilometerstand=kmStand;
}

Anlegen eines Objekts: Fahrzeug meinAuto(„VW“, „123“);

| | | | | |
|---|-------------------------|---------------|---------------------------------------|------|
|  | Klasse | Information | Thema Konstruktoren mit Parametern | |
| | Lehrer Oliver Kreuer | Lernfeld 6 | Datum | Name |

2. Konstruktor mit Parameterliste und Defaultwerten

Es sind auch Defaultwerte für die zu übergebenden Parameter erlaubt. Damit muss man beim Erzeugen eines Objekts nicht notwendigerweise alle Parameter angeben. Die Defaultwerte werden im **Prototypen** des Konstruktors vereinbart!

```

Prototyp:          Fahrzeug(char *markenname="Audi", int kmStand=0);

Implementation:    Fahrzeug::Fahrzeug(char *markenname, int kmStand)
                    {
                        strcpy(marke, markenname);
                        kilometerstand=kmStand;
                    }

Anlegen eines Objekts  Fahrzeug meinAuto;
                    // Alle Datenmember erhalten vorhandene
                    // Defaultwerte.

                    Fahrzeug meinAuto2("Audi");
                    // Nur der Kilometerstand wird auf den Defaultwert 0
                    // gesetzt. Der Markenname ist „Audi“.

                    Fahrzeug meinAuto3("Polo", "23000");
                    // Alle Datenmember werden auf die übergebenen
                    // Parameterwerte gesetzt.

```

Zu beachten:

1. Parameter mit Defaultwerten **müssen am Ende** der Parameterliste stehen. Es darf danach **kein Parameter ohne Defaultwert mehr angegeben werden!**
2. Man kann keine führenden Parameter beim Anlegen eines Objekts überspringen, wenn nachfolgende Parameter explizite Werte erhalten sollen.

Beispiel:

```

Fahrzeug meinAuto4(17);
// FEHLER: Für den Markennamen gibt es zwar einen
// Defaultwert, aber hier wird versucht den Markennamen
// auf die Zahl 17 zu setzen.

```