

```
1 using System;
2 using System.Collections.Generic;
3 using System.Linq;
4 using System.Text;
5 using System.Threading.Tasks;
6
7 namespace OOP.Auto
8 {
9     class Auto
10    {
11        #region Eigenschaften
12        // Verbesserungsidee 01: Korrekte Datentypen verwenden!
13        // Verbesserungsidee 02: Keine Umlaute verwenden
14        // Korrekt: private Objektvariablen beginnen mit einem Kleinbuchstaben ↗
15
16        // int autoFahrt;
17        // int motorLäuft;
18        bool motorEin;
19        int aktuelleGeschwindigkeit;
20        #endregion
21
22        #region Konstruktor(en)
23        public Auto()
24        {
25            aktuelleGeschwindigkeit = 0;
26            motorEin = false;
27        }
28        #endregion
29
30        #region Getter und Setter
31        // Verbesserungsidee 03: keine Setter mehr
32        // die Veränderung der Objektvariablen darf nur über die öffentlichen
33        // Methoden stattfinden!
34
35        // Getter sind - wie gefordert - vorhanden
36        public bool GetMotorEin ()
37        {
38            return motorEin;
39        }
40
41        public int GetAktuelleGeschwindigkeit()
42        {
43            return aktuelleGeschwindigkeit;
44        }
45        #endregion
46
47        #region öffentlichen Methoden
48        public bool InFahrt()
49        {
50            return (aktuelleGeschwindigkeit > 0);
51        }
52
53        public bool Stillstand()
54        {
55            return (aktuelleGeschwindigkeit == 0);
56        }
57    }
```

```
56
57     public bool MotorAus()
58     {
59         return (!motorEin);
60     }
61
62     public bool MotorEin()
63     {
64         return motorEin;
65     }
66
67
68     // Verbesserungsidee: public methods beginnen mit einem Großbuchstaben
69     public bool Anstarten()
70     {
71         bool erg = false;
72         // Verbesserungsvorschlag
73         //if (aktuelleGeschwindigkeit == 0 && motorEin == 0)
74         if (MotorAus() && Stillstand())
75         {
76             motorEin = true;
77             erg = true;
78         }
79         return erg;
80     }
81
82     public bool abstellen()
83     {
84         bool erg = false;
85         if (aktuelleGeschwindigkeit == 0 && motorEin == 1)
86         {
87             motorEin = 0;
88             erg = true;
89         }
90         return erg;
91     }
92
93     public bool anfahren()
94     {
95         bool erg = false;
96         if (aktuelleGeschwindigkeit == 0 && motorEin == 1)
97         {
98             aktuelleGeschwindigkeit = 1;
99             erg = true;
100         }
101         return erg;
102     }
103
104     public bool anhalten()
105     {
106         bool erg = false;
107         if (aktuelleGeschwindigkeit == 1 && motorEin == 1)
108         {
109             aktuelleGeschwindigkeit = 0;
110             erg = true;
111         }
112     }
```

```
112         return erg;  
113     }  
114 }  
115 }  
116
```