**Aufgabenstellungen**

1. Durchschnittspreis von den Autos von VW welche keine Kilometer haben.
2. Durchschnittskilometer von den Autos von Audi die über 100.000€ kosten.
3. Anzahl von den Autos von Audi welche keine Kilometer haben.
4. Anzahl von den Autos von Mercedes welche als Sitzbezug Polyester haben.
5. Maximumpreis von dem Auto von Volkswagen welches mit Erdgas tankt.
6. Maximalen Kilometern von dem Auto von Audi welches ein Automatisches Schaltgetriebe hat.
7. Minimalpreis von dem Auto von Mercedes welches nicht unfallfrei ist.
8. Minimaleigengewicht von dem Auto von Volkswagen welches eine Vollkaskoversicherung hat.
9. Summe des Preises von den Autos von Mercedes welche als Sitzbezug Art Velour haben.
10. Summe des Eigengewichts von den Autos von Mercedes welche ein Leasing haben.
11. Gib das erste Auto von Volkswagen aus, welches AddBlue verwendet. Gib das Modell und den Kilometerstand aus.
12. Gib das erste Auto von Audi aus, welches Kunstleder als Sitzbezug hat. Gib das Modell und den Kilometerstand aus.
13. Gib das letzte Auto von Audi aus, welches eine Haftpflichtversicherung hat. Gib das Modell und den Kilometerstand aus.
14. Gib das letze Auto von Merces aus, welches als PS mehr als 300 hat. Gib das Modell und den Kilometerstand aus.
15. Gib die einzige G-Klasse von Mercedes aus und dessen PS.
16. Gib den einzigen Hybridmotor von Mercedes aus und Sitzbezug.

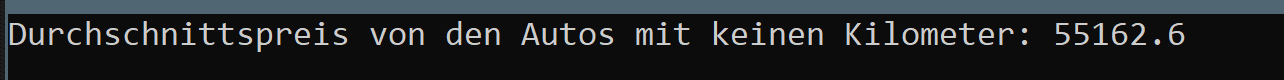
**Lösungen**

**Average:**

Durchschnittspreis von den Autos von VW welche keine Kilometer haben.

var result = i.vw.Where(s => s.Kilometers == 0).Average(s => s.Price);

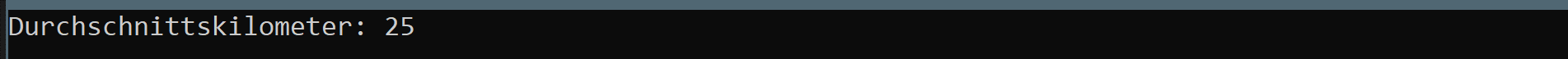
Console.WriteLine("Durchschnittspreis von den Autos mit keinen Kilometer: {0}", result);



Durchschnittskilometer von den Autos von Audi die über 100.000€ kosten.

var result = i.a.Where(s => s.Price >= 100000).Average(s => s.Kilometers);

Console.WriteLine("Durchschnittskilometer: {0}", result);



**Count:**

Anzahl von den Autos von Audi welche keine Kilometer haben.

var result = i.a.Where(s => s.Kilometers == 0).Count();

Console.WriteLine("Anzahl: {0}", result);



Anzahl von den Autos von Mercedes welche als Sitzbezug Polyester haben.

var result = i.m.Where(s => s.SeatCover == "Polyester").Count();

Console.WriteLine("Anzahl: {0}", result);



**Max:**

Maximumpreis von dem Auto von Volkswagen welches mit Erdgas tankt.

var result = i.vw.Where(s => s.Fuel == "Natural Gas").Max(s => s.Price);

Console.WriteLine("Maximun: {0}", result);



Maximalen Kilometern von dem Auto von Audi welches ein Automatisches Schaltgetriebe hat.

var result = i.a.Where(s => s.ManualTransmission == false).Max(s => s.Kilometers);

Console.WriteLine("Maximun: {0}", result);



**Min:**

Minimalpreis von dem Auto von Mercedes welches nicht unfallfrei ist.

var result = i.a.Where(s => s.AccidentFree == true).Min(s => s.Price);

Console.WriteLine("Minimum: {0}", result);



Minimaleigengewicht von dem Auto von Volkswagen welches eine Vollkaskoversicherung hat.

var result = i.vw.Where(s => s.Insurance == EInsurance.FULLYCOMPREHENSIVE).Min(s => s.TareWeight);

Console.WriteLine("Minimum: {0}", result);



**Sum:**

Summe des Preises von den Autos von Mercedes welche als Sitzbezug Art Velour haben.

var result = i.m.Where(s => s.SeatCover == "Art Velour").Sum(s => s.Price);

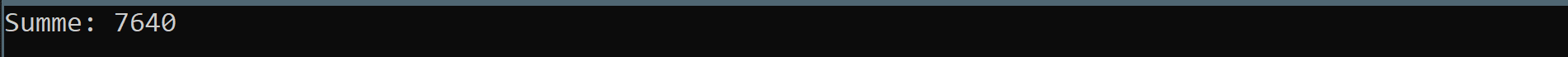
Console.WriteLine("Summe: {0}", result);



Summe des Eigengewichts von den Autos von Mercedes welche ein Leasing haben.

var result = i.m.Where(s => s.Leasing == true).Sum(s => s.TareWeight);

Console.WriteLine("Summe: {0}", result);

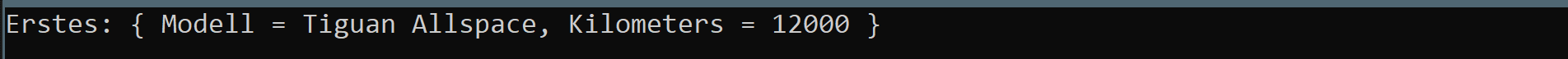


**First(OrDefault):**

Gib das erste Auto von Volkswagen aus, welches AddBlue verwendet. Gib das Modell und den Kilometerstand aus.

var result = i.vw.Where(s => s.AddBlue == true).Select(s => new {Modell = s.Modell, s.Kilometers}).First();

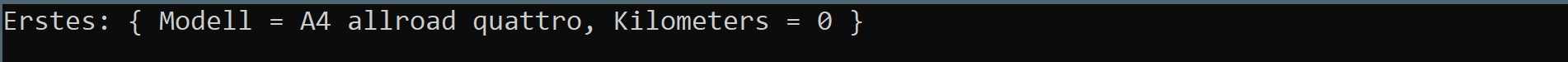
Console.WriteLine("Erstes: {0}", result);



Gib das erste Auto von Audi aus, welches Kunstleder als Sitzbezug hat. Gib das Modell und den Kilometerstand aus.

var result = i.a.Where(s => s.SeatCover == "Faux leather").Select(s => new {Modell = s.Modell, s.Kilometers}).First();

Console.WriteLine("Erstes: {0}", result);

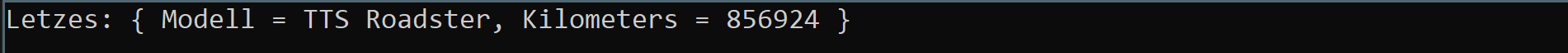


**Last(OrDefault):**

Gib das letzte Auto von Audi aus, welches eine Haftpflichtversicherung hat. Gib das Modell und den Kilometerstand aus.

var result = i.a.Where(s => s.Insurance == EInsurance.LIABILITY).Select(s => new {Modell = s.Modell, s.Kilometers}).Last();

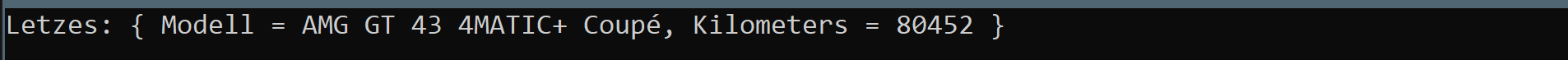
Console.WriteLine("Letzes: {0}", result);



Gib das letze Auto von Merces aus, welches als PS mehr als 300 hat. Gib das Modell und den Kilometerstand aus.

var result = i.m.Where(s => s.PS >= 300).Select(s => new {Modell = s.Modell, s.Kilometers}).Last();

Console.WriteLine("Letzes: {0}", result);



**Single(OrDefault):**

Gib die einzige G-Klasse von Mercedes aus und dessen PS.

var result = i.m.Where(s => s.Modell == "G-Klasse").Select(s => new {Modell = s.Modell, s.Kilometers}).Single();

Console.WriteLine("Einziges: {0}", result);



Gib den einzigen Hybridmotor von Mercedes aus und Sitzbezug.

var result = i.m.Where(s => s.Fuel == "Hybrid").Select(s => new { Modell = s.Modell, Sitzbezug = s.SeatCover }).Single();

Console.WriteLine("Einziges: {0}", result);

