

```
using System.Linq;
```

```
InitializerClass i = new InitializerClass();
```

```
Console.WriteLine("Distinct von vm Kilometers");
```

```
IEnumerable<int> query1 = i.vw.Select(one => one.Kilometers).Distinct();
```

```
foreach (var item1 in query1)
```

```
{
```

```
    Console.WriteLine(item1.ToString());
```

```
}
```

```
Distinct von vm Kilometers
```

```
0
```

```
20000
```

```
120000
```

```
12000
```

```
85621
```

```
500
```

```
Console.WriteLine("Distinct von a1 Price");
```

```
IEnumerable<int> query11 = i.a1.Select(one => one.Price).Distinct();
```

```
foreach (var item1 in query11)
```

```
{
```

```
    Console.WriteLine(item1.ToString());
```

```
}
```

```
Distinct von a1 Price
```

```
57279
```

```
179000
```

```
189988
```

```
74814
```

```
50243
```

```
85149
```

```
150346
```

```
105236
```

```
85621
```

```
86292
```

```
Console.WriteLine("Intersect von Seatcover von a und a1");
```

```
IEnumerable<string> query2 = i.a.Select(one =>
```

```
one.SeatCover).Intersect(i.a1.Select(two => two.SeatCover));
```

```
foreach (var item2 in query2)
```

```
{
```

```
    Console.WriteLine(item2.ToString());
```

```
}
```

```
Intersect von Seatcover von a und a1
```

```
Faux leather
```

```
Leather
```

```
Polyester
```

```
Art Velour
```

```
Console.WriteLine("Intersect vw und vw1 Insurance");
```

```
IEnumerable<EInsurance> query22 = i.vw.Select(one =>
```

```
one.Insurance).Intersect(i.vw1.Select(two => two.Insurance));
```

```
foreach (var item2 in query22)
```

```
{
```

```
    Console.WriteLine(item2.ToString());
```

```
}
```

```
Intersect vw und vw1 Insurance
```

```
FULLYCOMPREHENSIVE
```

```
PARTIALCOMPREHENSIVE
```

```
LIABILITY
```

```
Console.WriteLine("Except m und m1 Modell");
```

```
IEnumerable<string> query3 = i.m.Select(one => one.Modell).Except(i.m1.Select(two  
=> two.Modell));
```

```
foreach (var item3 in query3)
{
    Console.WriteLine(item3.ToString());
}
```

```
Except m und m1 Modell
G-Klasse
AMG A 35 4MATIC
AMG CLS 53 4MATIC+
A 250 E Kompaktlimousine
A 180 d Kompaktlimousine
AMG GT C Roaster
C 220 d T-Modell
AMG GT 43 4MATIC+ Coupé
```

```
Console.WriteLine("Except a und a1 FirstRegistration");
IEnumerable<DateTime> query33 = i.m.Select(one =>
one.FirstRegistration).Except(i.m1.Select(two => two.FirstRegistration));
foreach (var item3 in query33)
{
    Console.WriteLine(item3.ToString());
}
```

```
Except a und a1 FirstRegistration
```

```
Console.WriteLine("Concat von a und a1 Modell");
IEnumerable<string> query4 = i.a.Select(one => one.Modell).Concat(i.a1.Select(two
=> two.Modell));
foreach (var name in query4)
{
    Console.WriteLine(name);
}
```

```
Concat von a und a1 Modell
A4 allroad quattro
S8 TFSI quattro
RS Q8
TTS Roadster quattro
Q4 40 e-tron
A5 allroad quattro
S7 TFSI
RS Q2
TTS Roadster
Q5
A4 allroad quattro
S8 TFSI quattro
RS Q8
TTS Roadster quattro
Q4 40 e-tron
A5 allroad quattro
S7 TFSI
Q2
TTS
Q5
```

```
Console.WriteLine("Concat von vw und vw1 TareWeight");
IEnumerable<int> query44 = i.vw.Select(one =>
one.TareWeight).Concat(i.vw1.Select(two => two.TareWeight));
foreach (var name in query44)
{
    Console.WriteLine(name);
}
```

```
Concat von vw und vw1 TareWeight
1220
2242
1070
1005
1934
1524
1715
3010
1423
1828
1220
2242
1070
1005
1934
1524
1715
3010
1423
1828
```

```
Console.WriteLine("Union von vw und vw1 PS");
IEnumerable<int> query5 = i.vw.Select(one => one.PS).Union(i.vw1.Select(two =>
two.PS));
foreach (var item3 in query5)
{
    Console.WriteLine(item3.ToString());
}
```

```
Union von vw und vw1 PS
110
299
115
204
150
462
122
```

```
Console.WriteLine("Union von m und m1 ManualTransmission");
IEnumerable<string> query55 = i.vw.Select(one =>
one.Color).Union(i.vw1.Select(two => two.Color));
foreach (var item3 in query55)
{
    Console.WriteLine(item3.ToString());
}
```

```
Union von m und m1 Color
black
red
yellow
blue
white
cyan
darkgrey
darkred
gold
```