

SOFTWARE ONTWIKKELEN

AGENDA VOOR VANDAAG

- Lesdoelen

LESDOELEN

Na de les van vandaag:

- Kun je eenvoudig de lengte van een lijst achterhalen!
- Kun je de volgende functies gebruiken en toepassen op een lijst:
 - Maximale waarde
 - Minimale waarde
 - Gemiddelde waarde

VORIGE LES

- Eenvoudig beheerprogramma.
 - Hoe is dat gegaan?
 - Zijn er nog vragen over?

HOEVEEL ELEMENTEN HEEFT DE LIJST

Via een loop!

```
List<double> gasMeterStanden = [12.5, 13.72, 14.23, 14.75, 15.35, 17.20, 19.35];
int lengte = 0;

foreach (double gasMeterstand in gasMeterStanden)
{
    lengte++;
}

Console.WriteLine(lengte);
```



Dat is absoluut niet nodig en gewenst.
Gebruik de property Count.

```
Console.WriteLine(gasMeterStanden.Count);
```

Geen haakjes. Dit is geen method, maar een property. Je kunt dit zien als een intern bijgehouden lengte!

WELKE DAG HEB IK HET MEESTE VERBRUIKT?

- O.b.v. meterstanden zijn dit eigenlijk 2 vragen ineen!
- Eerst het verbruik per dag!
- De vraag is: hoe kunnen we eenvoudig van meterstand naar verbruik?



VAN METERSTAND NAAR VERBRUIK

```
[11.78, 12.5, 13.72, 14.23, 14.75, 15.35, 17.20, 19.35]
```

12,5 - 11,78

13,72 - 12,5

Etc.

lijst[1] - lijst[0]

```
List<double> gasMeterStanden = [11.78, 12.5, 13.72, 14.23, 14.75, 15.35, 17.20, 19.35];  
Console.WriteLine($"Verbruik dag 1: {gasMeterStanden[1] - gasMeterStanden[0]}");  
Console.WriteLine($"Verbruik dag 2: {gasMeterStanden[2] - gasMeterStanden[1]}");  
Console.WriteLine($"Verbruik dag 3: {gasMeterStanden[3] - gasMeterStanden[2]}");
```

DAT KAN HANDIGER

```
List<double> gasMeterStanden = [11.78, 12.5, 13.72, 14.23, 14.75, 15.35, 17.20, 19.35];  
Console.WriteLine($"Verbruik dag 1: {gasMeterStanden[1] - gasMeterStanden[0]}");  
Console.WriteLine($"Verbruik dag 2: {gasMeterStanden[2] - gasMeterStanden[1]}");  
Console.WriteLine($"Verbruik dag 3: {gasMeterStanden[3] - gasMeterStanden[2]}");
```

- Met een for loop!

```
for (int i = 1; i < gasMeterStanden.Count; i++)  
{  
    Console.WriteLine($"Verbruik dag {i}: {gasMeterStanden[i] - gasMeterStanden[i-1]:f2}");  
}
```


EN TIJDELIJK EVEN IN EEN LIJST!

```
List<double> verbruikPerDag = [];  
for (int i = 1; i < gasMeterStanden.Count; i++)  
{  
    verbruikPerDag.Add(gasMeterStanden[i] - gasMeterStanden[i-1]);  
}
```

Wat is dan het hoogste/ laagste en gemiddelde verbruik?

```
Console.WriteLine(verbruikPerDag.Max()); // Max geeft maximum waarde!  
Console.WriteLine(verbruikPerDag.Min()); // Min geeft minimum waarde!  
Console.WriteLine(verbruikPerDag.Average()); // Gemiddelde waarde!
```

ER IS VEEL MOGELIJK MET LIJSTEN!

<https://learn.microsoft.com/en-us/dotnet/api/system.linq.enumerable.sum?view=net-8.0>

- Zoek in tweetallen een handige methode uit.
- Schrijf een voorbeeld en bedenken een klein opdrachtje die kan worden uitgevoerd met de door jullie uitgezochte methode!
- Klaar: wissel dan met het groepje naast je!

HANDIGHEIDJES DELEN!

- Schrijf de handigste methodes op een whiteboard.

```
Console.WriteLine(verbruikPerDag.Sum());  
Console.WriteLine(verbruikPerDag.Last());  
Console.WriteLine(verbruikPerDag.First());
```


LESDOELEN BEHAALD?

- Kun je eenvoudig de lengte van een lijst achterhalen!
- Kun je de volgende functies gebruiken en toepassen op een lijst:
 - Maximale waarde
 - Minimale waarde
 - Gemiddelde waarde