

SOFTWARE ONTWIKKELEN

AGENDA VOOR VANDAAG

- Lesdoelen
- Leerstrategie: verwerking
- Methoden schrijven en aanroepen
- Verwerkingsopdracht

LESDOELEN

Na de les van vandaag:

- Kun je de leerstrategie 'retrieval practice' inzetten
- Heb je een aantal handvaten om algoritmes (vraagstukken) in kleinere onderdelen op te knippen.
- Kun je methoden in C# aanroepen en schrijven.

STAP 1 - KENNIS OPHALEN

- 2 minuten
- Laptop dicht en pak een wisbordje
- Student 1: Schrijf alles wat je nog weet van while-lussen op je wisbordje
- Student 2: Schrijf alles wat je nog weet van for-lussen op je wisbordje
- Student 3: Schrijf alles wat je nog weet van foreach-lussen op je wisbordje

STAP 2 – DEEL JE KENNIS

- 5 minuten
- Vind een student die hetzelfde onderwerp had als jij
- Deel met elkaar wat je antwoord was
 - Komt het overeen?
 - Wat zijn de verschillen?
 - Was je iets vergeten?
 - Schrijf wat je nog niet had bij op je wisbordje

STAP 3 – WISSEL UIT

- 10 minuten
- Ga in je groepje van drie zitten
- Iedere student geeft een uitleg van maximaal 2 minuten over het eigen onderwerp
- De anderen stellen vragen en vullen aan (indien nodig)
- Controleer in je groepje van drie of alles wat er gezegd is wel klopt en volledig is
- Pas aan en schrijf bij

OEFEN THUIS

- Maak flashcards met gebruik, algemene syntax en flowchart voor alle herhalingsstructuren
 - Oefen die elke dag 10 minuten
 - Vraag jezelf af: wat zijn de verbanden tussen deze begrippen?
- Doe de Brightspacequizzes van variabelen en expressies nog eens
 - Wat wist je nog niet?
- Maak zelf toetsvragen van alle tot nu toe behandelde onderwerpen
 - Ruil met een medestudent en maak elkaars vragen

METHODEN

- Tot op heden hebben we loops gebruikt om code te herhalen. Code hergebruiken kan ook door methoden te gebruiken.

Gebruiken we die nog niet dan?



```
Console.WriteLine("Hallo wereld");
```

```
int mijnGetal = int.Parse("15");
```

```
int teRadenGetal = Random.Shared.Next(1, 101);
```

```
Console.ResetColor();
```


METHODEN (1)

```
Console.WriteLine("Hallo wereld");
```

Class - waar
komt methode
vandaan?

Naam van de
methode

1 of meer
argumenten

```
double afgerond = Math.Round(12.5d, 0, MidpointRounding.AwayFromZero);
```

METHODEN (2)

- Maar wat roep je nu aan?
 - Voor 'standaard' C# methodes geldt dat je deze meestal beschouwt als 'blackbox', hoewel de code soms wel beschikbaar is.
- Schematisch:



METHODE SCHRIJVEN (1)

Return type

Naam methode

2 parameters

Waarde gaat naar
aanroep

```
int GrootsteGetal(int a, int b)
{
    int grootste = b;
    if (a > b)
    {
        grootste = a;
    }
    return grootste;
}
```

METHODE AANROEPEN

- MethodeNaam(benodigde argumenten);

2 argumenten



```
int getal1 = 15;
int getal2 = 7;
Console.WriteLine(GrootsteGetal(getal1, getal2)); // 15
int x = GrootsteGetal(3, 2);
Console.WriteLine(x); // 3
```


METHODE SCHRIJVEN (2)

Void method geeft niets terug aan aanroep.

```
void VijfKeerWriteLine(string tekst)
{
    for (int i = 0; i < 5; i++)
    {
        Console.WriteLine(tekst);
    }
}

VijfKeerWriteLine("Hallo wereld");
```



```
Hallo wereld
Hallo wereld
Hallo wereld
Hallo wereld
Hallo wereld
```

Hoe zouden we deze methode 'generieker' kunnen maken?

VAN VRAAGSTUK NAAR CODE (1)

Vraag:

Een actueel probleem voor veel elektrische rijders is:
kom ik wel in Zuid-Frankrijk? Oftewel: haal ik mijn bestemming?

Schrijf een methode die uitrekent of een bestemming gehaald kan worden.

VAN VRAAGSTUK NAAR CODE (2)

- Parameters:
 - verbruik in kWh/100km
 - accucapaciteit in kWh
 - gewenste afstand in km
- Wat wordt het return type?
- Schrijf en test de methode.

*De Tesla Model 3 verbruikt:
16,7 kWh/ 100 km*



LESDOELEN BEHAALD?

Ik ga samen met Robin toetsvragen maken. Ik maak ze over Variabelen en Expressies, Robin over Branching. We ruilen op vrijdag de vragen en dan maken we ze van elkaar. We kijken maandag samen de antwoorden na.

- Hoe ga je zorgen dat je de begrippen van Software Ontwikkelen beter begrijpt?
- Hoe schrijf je methodes in C# en wanneer gebruik je deze?