

3. Strings samenvoegen

H3: Tekst gebruiken in code



3 manieren om strings samen te voegen

- Manier 1 met +-operator

```
string a= "Dit ben ik: " + myNaam+ ". Knap hoor.";
```

- Manier 2 met string interpolatie met \$-notatie

Prefereerde manier

```
string a = $"Dit ben ik {myNaam}. Knap hoor.";
```

- Manier 3 met String.Format() (klassieke manier)

```
string a = string.Format("Dit ben ik {0}. Knap hoor.",myNaam);
```

Manier 2 : String interpolation met \$

- \$ voor string geeft aan 'alles tussen { } zijn C# expressies

```
string a = $"Dit ben ik {myNaam}. Knap hoor.";
String b= $"Ik kan 1 + 2 optellen dat is {1+2}";
```

Kan ook rechtstreeks in WriteLine

```
Console.WriteLine($"Dit ben ik {myNaam}. Knap hoor.");
```

\$-notatie

Kan je dus ook
rechstreeks
gebruiken in
WriteLine:

```
Console.WriteLine($"Dit ben ik {myNaam}. Knap hoor.");
```

Eender welke
expressie mag je
gebruiken:

```
string tekst=  
    $"Ik ben morgen jarig, dan {myNaam+1} jaar oud";
```

Ook meerdere
interpolaties
uiteraard
toegelaten

```
string tekst= $"Hallo {myNaam}. Ik ben {robotName}";
```

Data type	Voorbeeld	Resultaat
int	<code>int number = 123; Console.WriteLine(\$"{number:D}");</code>	123
int met precisie	<code>int number = 123; Console.WriteLine(\$"{number:5D}");</code>	00123
floating-point met precisie van 2 cijfers na komma	<code>double number = 12.345; Console.WriteLine(\$"{number:F2}");</code>	12,35
floating-point, zonder cijfers na de komma	<code>double number = 12.34; Console.WriteLine(\$"{number:F0}");</code>	12
wetenschappelijk	<code>double number = 12000000; Console.WriteLine(\$"{number:E2}");</code>	1,20E+007
geldbedrag	<code>double money = 12.34; Console.WriteLine(\$"{money:C}");</code>	€ 12,34
Getal met . scheidingen	<code>double number = 1234000; Console.WriteLine(\$"{number:N2}");</code>	1.234.000,00

Strings formatteren

Demo time

- Strings optellen
- Formatteren

