

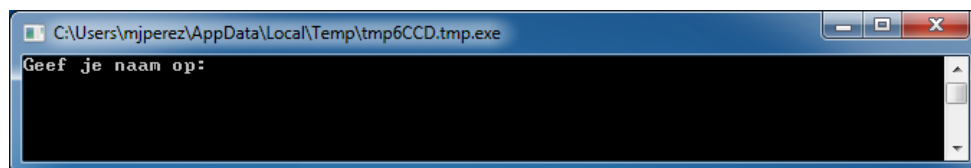
Variabelen introduceren

Variabelen gebruiken in ons programma

Het zou leuk zijn als ons programma “Hallo” kan zeggen met de naam van de gebruiker in plaats van het algemene “Hallo wereld?”. Als we dit willen doen, moeten we eerst de gebruiker om zijn of haar naam vragen, dan deze naam opslaan en vervolgens “Hallo” met de naam van de gebruiker naar het tekstvenster schrijven. Laten we eens kijken hoe we dit kunnen doen:

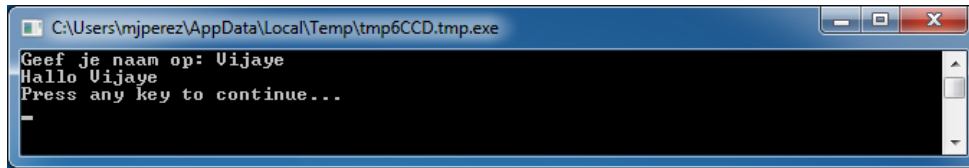
```
TextWindow.Write("Geef je naam op: ")
naam = TextWindow.Read()
TextWindow.WriteLine("Hallo " + naam)
```

Als je dit programma typt en uitvoert, zie je de volgende uitvoer:



Afbeelding 7 – Vraag om de naam van de gebruiker

En wanneer je je naam typt en op ENTER klikt, zie je de volgende uitvoer:



Afbeelding 8 – Een warme groet

Als je het programma opnieuw uitvoert, wordt er opnieuw dezelfde vraag gesteld. Je kunt een andere naam typen en de computer zal Hallo zeggen met die naam.

Analyse van het programma

De volgende regel in het programma dat je zojuist hebt uitgevoerd, heeft mogelijk je aandacht getrokken:

```
naam = TextWindow.Read()
```

Read() ziet er precies zo uit als *WriteLine()*, maar zonder invoer. Het is een bewerking waarmee de computer wordt verteld te wachten totdat de gebruiker iets typt en op de ENTER-toets drukt. Nadat de gebruiker op de ENTER-toets heeft gedrukt, slaat de computer op wat de gebruiker heeft getypt en geeft dit terug aan het programma. Het interessante hier is dat wat de gebruiker heeft getypt, is nu opgeslagen in een *variabele* met de naam **naam**. Een *variabele* wordt gedefinieerd als een plaats waar je waarden tijdelijk kunt opslaan om later te gebruiken. In de regel hierboven is **naam** gebruikt om de naam van de gebruiker op te slaan.

De volgende regel is ook interessant:

```
TextWindow.WriteLine("Hallo " +  
naam)
```

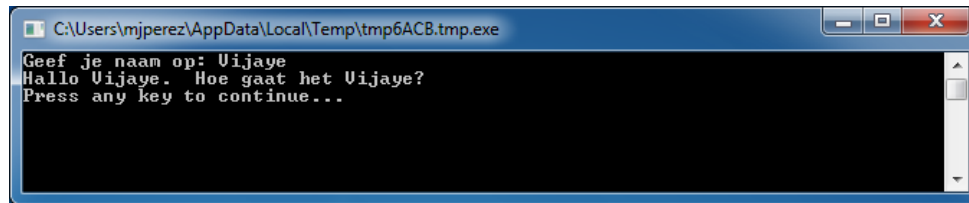
Write is net als WriteLine een andere bewerking in het tekstvenster. Met Write kun je iets naar het tekstvenster schrijven, maar met deze bewerking kun je ook de tekst die volgt op dezelfde regel als de huidige tekst plaatsen.

Dit is de waar we de waarden gebruiken die zijn opgeslagen in onze variabele **naam**. We nemen de waarde in **naam** en hechten dit aan "Hallo" en schrijven dit naar het TextWindow.

Als een variabele eenmaal is ingesteld, kun je deze zo vaak gebruiken als je dat wilt. Je kunt bijvoorbeeld het volgende doen:

```
TextWindow.Write("Geef je naam op: ")  
naam = TextWindow.Read()  
TextWindow.Write("Hallo " + naam + ". ")  
TextWindow.WriteLine("Hoe gaat het " + naam + "?")
```

En je zult de volgende uitvoer zien:



```
C:\Users\mjperrez\AppData\Local\Temp\tmp6ACB.tmp.exe
Geef je naam op: Uijaye
Hallo Uijaye. Hoe gaat het Uijaye?
Press any key to continue...
```

Afbeelding 9 – Een variabele opnieuw gebruiken

Regels voor het benoemen van variabelen

Variabelen hebben namen waaraan je ze kunt herkennen. Er zijn bepaalde eenvoudige regels en enkele goede richtlijnen voor het benoemen van deze variabelen. Dit zijn:

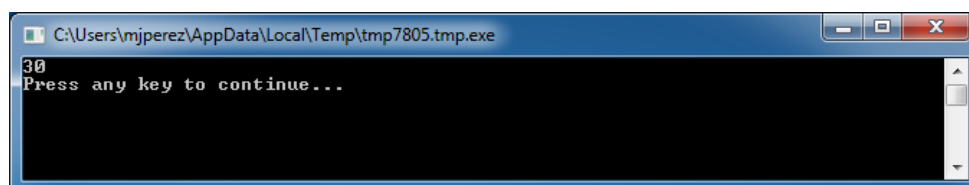
1. De naam moet beginnen met een letter en mag niet conflicteren met een van de sleutelwoorden zoals **if**, **for**, **then** enz.
2. Een naam kan elke combinatie van letters, cijfers en onderstrepingen bevatten.
3. Het is handig om de variabele een zinvolle naam te geven. En aangezien je de variabele zolang kunt maken als je wilt, kun je de bedoeling beschrijven.

Spelen met getallen

We hebben zojuist gezien hoe je variabelen kunt gebruiken om de naam van de gebruiker op te slaan. In de volgende programma's zullen we zien hoe we getallen in variabelen kunnen opslaan en manipuleren. Laten we beginnen met een heel eenvoudig programma:

```
getal1 = 10
getal2 = 20
getal3 = getal1 + getal2
TextWindow.WriteLine(getal3)
```

Als je dit programma uitvoert, zie je de volgende uitvoer:



```
C:\Users\mjperrez\AppData\Local\Temp\tmp7805.tmp.exe
30
Press any key to continue...
```

Afbeelding 10 – Twee getallen optellen

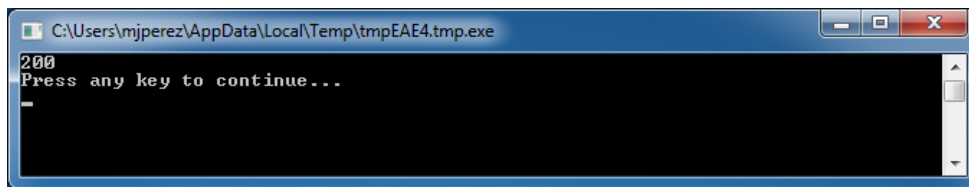
Op de eerste regel van het programma wijs je de variabele **getal1** toe met een waarde van 10. En op de tweede regel wijs je de variabele **getal2** toe met een waarde van 20. Op de derde regel voeg je **getal1** en **getal2** toe en wijs je het resultaat hiervan aan **getal3** toe. In dit geval heeft **getal3** daarom een waarde van 30. En dit is wat we naar het TextWindow hebben geschreven.

Misschien heb je opgemerkt dat de getallen niet worden omringd door aanhalingstekens. Aanhalingstekens zijn niet nodig voor getallen. Je hoeft alleen aanhalingstekens te gebruiken voor tekst.

Laten we nu het programma enigszins aanpassen en de resultaten bekijken:

```
getal1 = 10
getal2 = 20
getal3 = getal1 * getal2
TextWindow.WriteLine(getal3)
```

In het programma hierboven wordt **getal1** vermenigvuldigd met **getal2** en worden de resultaten opgeslagen in **getal3**. De resultaten van dat programma worden hieronder weergegeven:



Afbeelding 11 – Twee getallen vermenigvuldigen

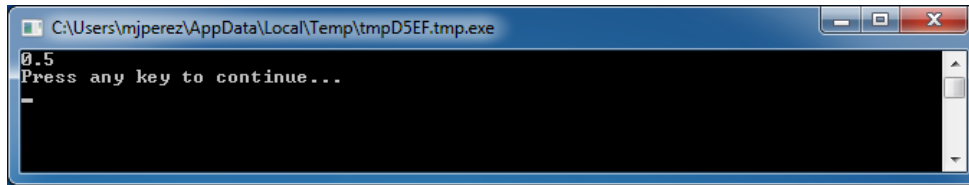
Op gelijke wijze kun je getallen aftrekken of delen. Hier volgt de aftrekking:

```
getal3 = getal1 - getal2
```

En het symbool voor deling is '/'. Het programma ziet er als volgt uit:

```
getal3 = getal1 / getal2
```

En het resultaat van deze deling is:



Afbeelding 12 – Twee getallen delen

Een eenvoudige temperaturomzetter

Voor het volgende programma gebruiken we de formule $^{\circ}\text{C} = \frac{5(^{\circ}\text{F}-32)}{9}$ voor het omrekenen van temperaturen in Fahrenheit naar temperaturen in Celsius.

Eerst moeten we de temperatuur in Fahrenheit, die we van de gebruiker hebben gekregen, opslaan in een variabele. Er is een speciale bewerking waarmee we getallen van de gebruiker kunnen lezen:

TextWindow.ReadNumber.

```
TextWindow.Write("Geef de temperatuur op in Fahrenheit: ")
fahr = TextWindow.ReadNumber()
```

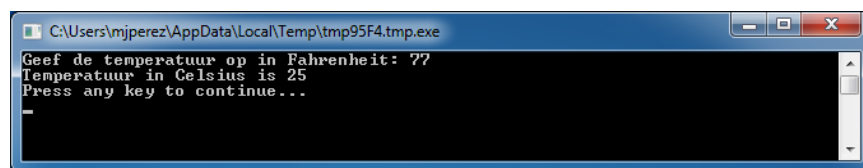
Nadat we de temperatuur in Fahrenheit hebben opgeslagen in een variabele, kunnen we deze als volgt naar Celsius omrekenen:

```
celsius = 5 * (fahr - 32) / 9
```

Met de haakjes wordt de computer de opdracht gegeven het gedeelte **fahr – 32** eerst te berekenen en vervolgens de rest te verwerken. Het enige wat we nu nog moeten doen, is het schrijven van de resultaten naar de gebruiker. Met dit alles krijgen we het volgende programma:

```
TextWindow.Write("Geef de temperatuur op in Fahrenheit: ")
fahr = TextWindow.ReadNumber()
celsius = 5 * (fahr - 32) / 9
TextWindow.WriteLine("Temperatuur in Celsius is " + celsius)
```

En het resultaat van dit programma is:



Afbeelding 13 – Temperatuurconversie