

Programmeerspellen met Small Basic:

Hoofdstuk 1: Schrijfprogramma's

[Kleine basisboeken](#) > [kleine basisboeken](#) > [programmeerspellen met kleine basis](#) > **Hoofdstuk 1: Schrijfprogramma's**

Dit hoofdstuk is een bewerking van het boek Programming Games with Microsoft Small Basic van Philip Conrod en Lou Tylee.

Om dit boek in zijn geheel te kopen, zie [de Computer Science For Kids website](#).

Programma's schrijven met Small Basic



Voorbeeld

In dit eerste hoofdstuk zullen we een overzicht geven van hoe je een programma schrijft met Small Basic. Je krijgt een korte geschiedenis van Small Basic en kijkt naar het gebruik van de Small Basic ontwikkelomgeving.

Maak kennis met programmeerspellen met Microsoft Small Basic

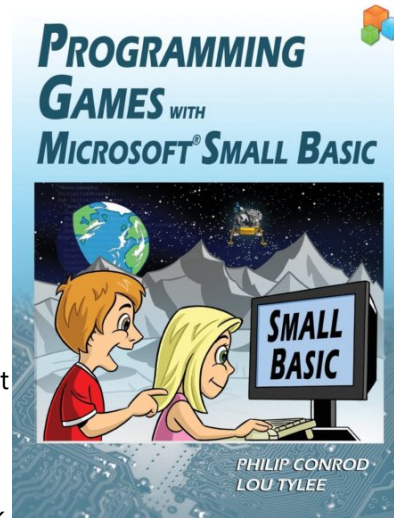
In deze notities zullen we Small Basic gebruiken om veel leuke spelprogramma's te bouwen. De spellen zijn geweldloos en leren logisch denkvermogen. Ze zijn geschikt voor kinderen van alle leeftijden (zelfs volwassenen). De programma's die je gaat bouwen zijn (in toenemende complexiteit):

- **Safecracker** - Ontcijfer een geheime combinatie met behulp van aanwijzingen van de computer.
- **Tic Tac Toe** - Het klassieke spel - een van de eerste geprogrammeerd door Bill Gates!
- **Match Game** - Vind bijpassende paren van verborgen foto's? je gebruikt je eigen foto's!

- **Pizza Delivery** - Een bedrijfssimulatie waarbij u een kleine pizzawinkel voor een nacht beheert.
- **Maanlanding** - Land een module op het oppervlak van de maan.
- **Leap Frog** - Een leuk arcadespel waarbij je een kikker door het verkeer en over een woeste rivier moet krijgen.

Deze programma's zullen veel van de vaardigheden leren die nodig zijn om een succesvolle spelprogrammeur te zijn. Je leert over timing, multi-player games, scores, simulatietechnieken en animatie.

Elk programma wordt behandeld in een enkel hoofdstuk. Volledige stapsgewijze instructies voor elk programmadetail worden verstrekt. Voordat we met de programma's beginnen, zullen we small basic en zijn ontwikkelingsomgeving in de rest van dit hoofdstuk bekijken. Vervolgens geven we een overzicht van de objecten die worden gebruikt in Small Basic (hoofdstuk 2), een overzicht van de programmeertaal Small Basic (hoofdstuk 3)) en instructies voor het gebruik van de small basic-foutopsporing (hoofdstuk 4). De programma's beginnen met hoofdstuk 5.




Vereisten voor het programmeren van games met Small Basic

Voordat we beginnen, laten we eens kijken wat u nodig hebt om de programma's die zijn opgenomen in **Programming Games With Small Basic** met succes te bouwen. Wat computervaardigheden betreft, moet u zich comfortabel voelen bij het werken in de Windows-omgeving. U moet weten hoe u programma's moet uitvoeren, mappen kunt zoeken en maken en vensters moet verplaatsen en vergroten of verkleinen.

Wat programmeervaardigheden betreft, gaan we ervan uit dat je enige blootstelling hebt gehad aan computerprogrammering met behulp van een bepaalde taal. Als die taal **Small Basic** is, geweldig!! We bieden twee tutorials **Small Basic for Kids** en **Beginning Small Basic**, die u kunnen helpen die bekendheid te krijgen (zie onze website voor meer informatie). Maar als je ooit in een taal hebt geprogrammeerd (Visual Basic, C, C++, C#, Java, J#, Ada, zelfs FORTRAN), zou je moeten kunnen volgen wat er aan de hand is. Zelfs als u een ervaren programmeur bent, raden we u aan de eerste drie hoofdstukken te doorlopen voordat u de programma's aanvalt. Deze beoordeling geeft u een idee van de terminologie die we gebruiken bij het verwijzen naar verschillende delen van een Small Basic-programma.

Wat de softwarevereisten betreft, moet u Windows 8, Windows 7, 2000, Windows XP, Windows NT of Windows Vista gebruiken om Small Basic te gebruiken. Deze notities en alle programma's zijn ontwikkeld met behulp van Windows Vista en versie 1.0 van Small Basic. En natuurlijk moet u het Small Basic-product op uw computer hebben geïnstalleerd. Het is gratis te downloaden van Microsoft.

Volg deze link voor volledige instructies voor het downloaden en installeren van Small Basic op uw computer:

<http://www.smallbasic.com> 

Maak kennis met Small Basic

In de late jaren 1970 en vroege jaren 1980, het lijkt erop dat er overal computers waren met namen als Commodore 64, Texas Instruments 99/4A, Atari 400, Coleco Adam, Timex Sinclair en de IBM PC-Jr. Winkels zoals Sears, JC Penneys en zelfs K Mart verkochten computers.

Een ding dat deze machines gemeen hadden, was dat ze allemaal waren geprogrammeerd in een of andere versie van Microsoft's BASIC. Elke computer had zijn eigen ventilatoren en eigen tijdschriften. Gebruikers zouden elke maand wachten op het volgende nummer van een tijdschrift met BASIC-programma's die u op uw computer kunt typen en thuis kunt proberen.

Dit was een leuke en spannende tijd voor de beginnende programmeur, maar de leuke tijden eindigden met de introductie van de IBM-PC in de vroege jaren 1980. Grotere en snellere computers brachten grotere talen en grotere ontwikkelomgevingen voort. Deze nieuwe talen waren duur om te verwerven en moeilijk te begrijpen voor de beginnende programmeur.

Dat brengt ons bij **Small Basic**, dat ik een familielid van de vroege, originele BASIC-taal zou noemen. De ontwikkeling van Small Basic is een poging om de spannende dagen opnieuw aan te wakkeren waarin zowat iedereen achter een computer kon gaan zitten en een eenvoudig programma kon schrijven met behulp van de BASIC-taal.

Degenen onder u die programma's op die oude "speelgoed" -computers hebben geschreven, zullen de eenvoud van de Small Basic-taal en het gebruiksgemak ervan herkennen. En je zult ook merken dat Small Basic een geweldige omgeving is voor het schrijven en testen van code, iets dat ontbreekt in de vroege jaren 1980. Voor degenen onder jullie die nieuw zijn in programmeren, hoop ik dat je de opwinding kunt voelen die we oldtimers ooit hadden. Voor de oldtimers hoop ik dat je je programmeervaardigheden opnieuw aanwakkert met dit nieuwe product.

Small Basic beschikt over veel functies van krachtigere (en duurdere) programmeertalen:


- Gebruiksvriendelijke, geïntegreerde ontwikkelomgeving (IDE)
- Reactie op muis- en toetsenbordacties
- Volledige reeks wiskundige, tekenreeksverwerkings- en grafische functies
- Kan gemakkelijk werken met arrays
- Ondersteuning voor sequentiële bestanden

Small Basic starten

We gaan ervan uit dat Small Basic op uw computer is geïnstalleerd en operationeel is. Eenmaal geïnstalleerd, om Small Basic te starten:

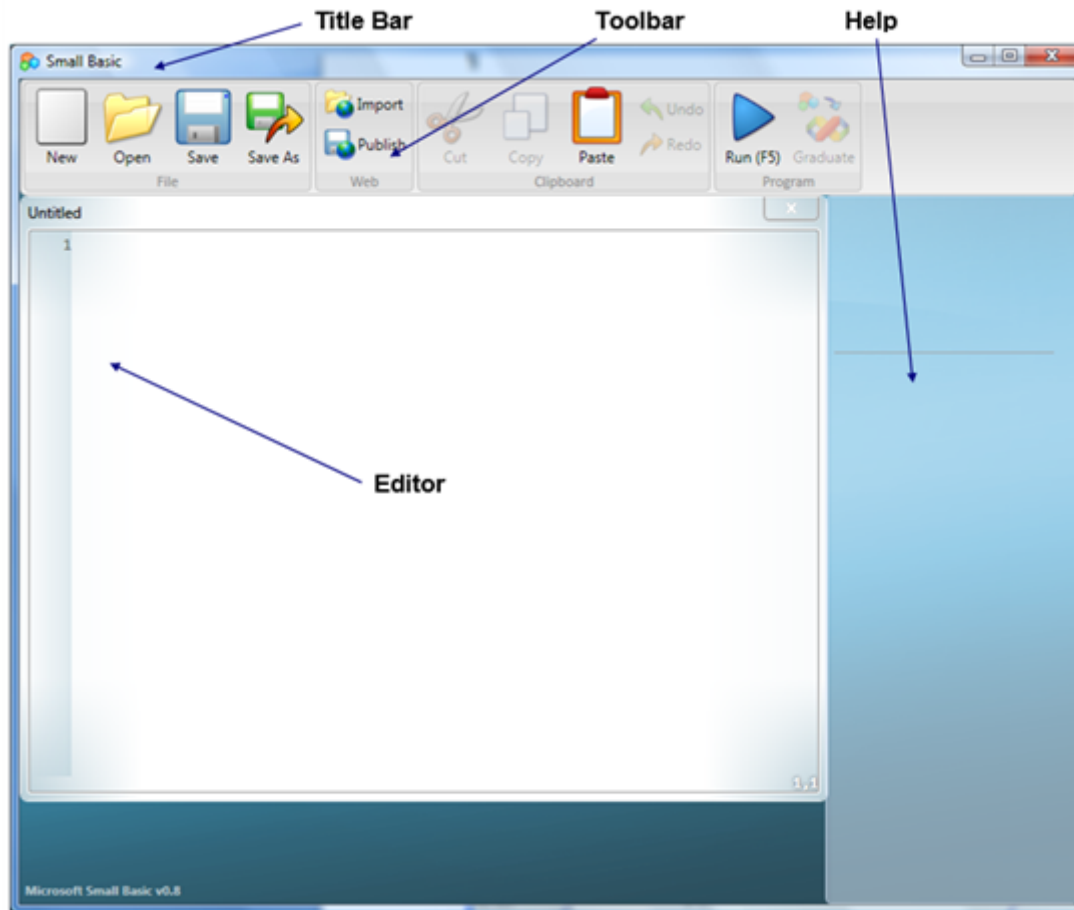
- Klik op de knop **Start** op de taakbalk van Windows
- Selecteer **Programma's** en vervolgens **Small Basic**
- Klik op **Microsoft Small Basic**

(Sommige van de hier gegeven koppen kunnen enigszins verschillen op uw computer, maar u zou geen moeite moeten hebben om de juiste te vinden.) Het Small Basic-programma zou moeten starten.

Na de installatie en het starten, ziet u mogelijk een foutbericht waarin wordt aangekondigd dat Small Basic niet kan worden gestart. Als dit gebeurt, probeert u de nieuwste versie van Microsoft .NET Framework te downloaden en te installeren op: <http://msdn.microsoft.com/en-us/netframework/aa569263.aspx> 

Dit bevat enkele bestanden die Small Basic nodig heeft om te werken en dergelijke bestanden staan mogelijk niet op uw computer.

Bij het starten toont mijn scherm:



In dit venster wordt de **Small Basic Development Environment** weergegeven. Er zijn veel interessegebieden op het scherm. Boven in het venster bevindt zich de **titelbalk**. De titelbalk geeft ons informatie over welk programma we gebruiken en met welk Small Basic-programma we werken.

Onder de titelbalk bevindt zich een **werkbalk**. Hier kunnen we met kleine knoppen met afbeeldingen Small Basic bedienen.

In het midden van het scherm bevindt zich de **Editor**. Dit is waar we onze Small Basic-programma's zullen schrijven. Aan de rechterkant is een **Help-gebied**. Small Basic heeft geweldige hulpsfuncties bij het schrijven van programma's. In dit gebied worden hints en tips weergegeven terwijl we code schrijven.

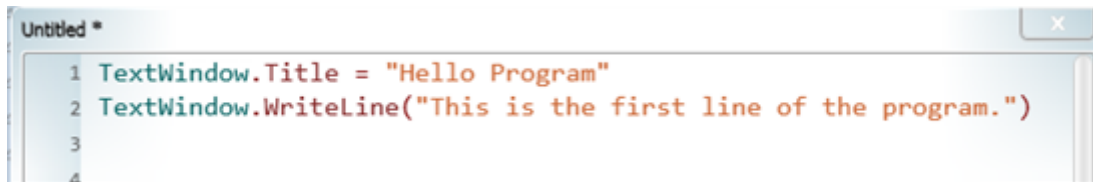
Een klein basisprogramma uitvoeren

Laten we ons eerste Small Basic-programma schrijven. Wanneer u begint, verschijnt er een nieuw editorvenster. U kunt ook een nieuw editorvenster krijgen door op de knop **Nieuwe** werkbalk te klikken. Typ deze twee regels in het editorvenster:

Tekstvenster. Titel= "Hallo Programma"

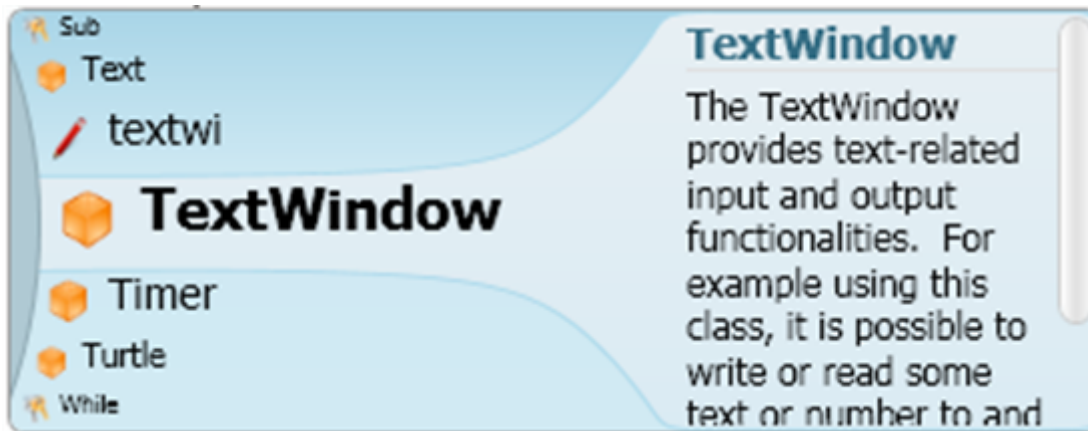
Tekstvenster. WriteLine("Dit is de eerste regel van het programma.")

Het editorvenster zou er als volgt uit moeten zien:



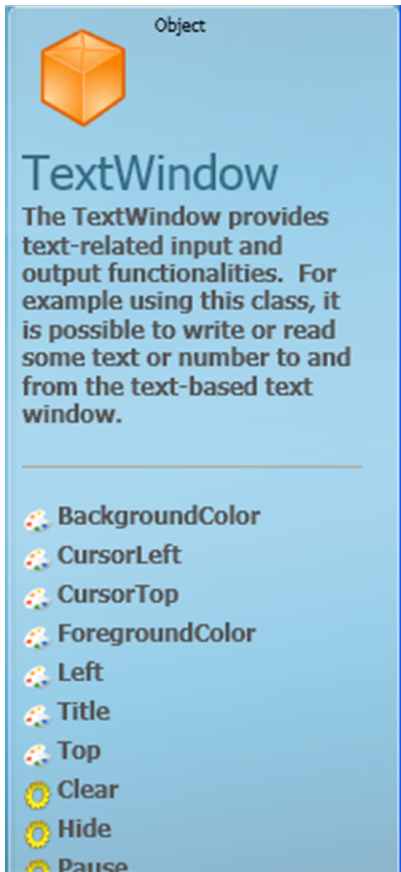
```
1 TextWindow.Title = "Hello Program"
2 TextWindow.WriteLine("This is the first line of the program.")
3
4
```

Merk op toen je de eerste regel begon te typen, dit verscheen:



Small Basic heeft 'IntelliSense' en gebruikt dit om typsuggesties te doen. U kunt gewoon blijven typen of de suggestie accepteren door door de pop-uplijst te bladeren en op de **Enter-toets** te drukken.

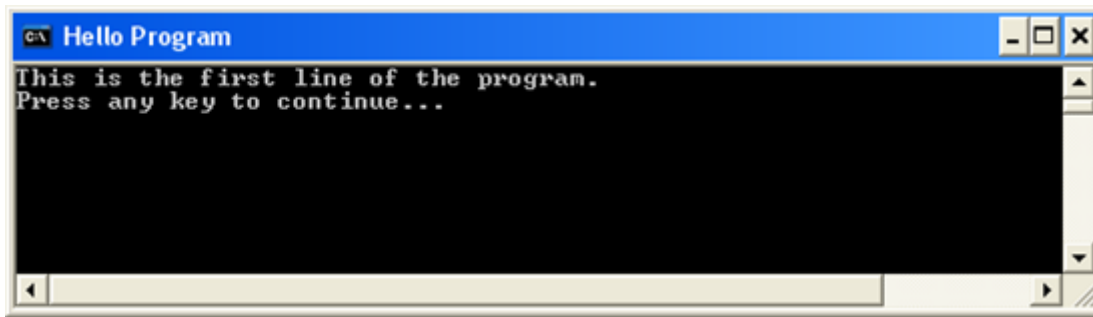
Merk ook op dat dit in het helpgebied verscheen:



Nadat u **TextWindow** had **getypt**, gaf de helpfunctie alles weer wat het weet over het **tekstvenster** om u te helpen bij het programmeren.

Het **tekstvenster** is een klein **basisobject**. Het geeft tekstuitvoer weer. Het object heeft **eigenschappen**, **methoden** en **gebeurtenissen**. Er zijn veel objecten in Small Basic. Zij zullen in hoofdstuk 2 worden besproken.

Laten we het programma uitvoeren. Klik gewoon op de knop **Uitvoeren** op de werkbalk om het volgende te zien:



Dit is het tekstvenster dat de programma-uitvoer weergeeft. Ik heb het formaat van het venster een beetje aangepast. Als u het programma wilt stoppen, klikt u op de **X** in de rechterbovenhoek van het venster.

Dat is alles wat er is om een Small Basic-programma te schrijven. Typ de code in het editorvenster. Klik op de knop **Uitvoeren** om de code uit te voeren. We zullen veel Small Basic-code leren terwijl we de spellen in deze cursus bouwen. Hoofdstuk 3 bespreekt de meeste elementen van de Kleine Basistaal.

Andere handige werkbalkknoppen zijn:

- **Openen** - Open een eerder opgeslagen Small Basic-programma
- **Opslaan** - Een klein basisprogramma opslaan
- **Opslaan als** - Een klein basisprogramma met een andere naam opslaan

We raden u aan elk Small Basic-programma in een eigen map op te slaan.

Er zijn ook de knoppen **Knippen**, **Kopiëren**, **Plakken**, **Ongedaan maken** en **Opnieuw uitvoeren** voor algemene bewerkingstaken. Leer je deze te gebruiken? ze besparen u veel tijd bij het typen van code.

Hoofdstuk Review

Dit completeert ons overzicht van de Small Basic-omgeving en een korte demonstratie van het schrijven van een Small Basic-programma. Als je Small Basic eerder hebt gebruikt, moet dit materiaal bekend zijn. Als je met andere talen hebt geprogrammeerd, moet je een vrij duidelijk begrip hebben van wat er aan de hand is.

Na het voltooien van dit hoofdstuk moet u begrijpen:

- Een beetje van de geschiedenis van Small Basic.
- De verschillende onderdelen van de Small Basic geïntegreerde ontwikkelomgeving.
- Het nut van 'IntelliSense' en het Help-paneel Small Basic.
- Code schrijven met behulp van de code-editor.
- Hoe een Small Basic-programma uit te voeren.

Voordat we de spelprogramma's in hoofdstuk 5 starten, moeten we de objecten bekijken die worden gebruikt om programma's te bouwen (hoofdstuk 2), de taal van Small Basic (hoofdstuk 3) en small basic-foutopsporing (hoofdstuk 4).

Voor een ander online hoofdstuk, zie [Programming Games with Small Basic: Chapter 6: Tic Tac Toe Program](#).

Om dit boek in zijn geheel te kopen, zie [de Computer Science For Kids website](#).

© Uittreksel Copyright 2010-2017 Door Kidware Software LLC Alle rechten voorbehouden. Computer Science For Kids, het Computer Science For Kids-logo en gerelateerde trade dress zijn handelsmerken of geregistreerde handelsmerken van Kidware Software LLC. Philip Conrod & Lou Tylee zijn al meer dan 25 jaar co-auteur van tientallen boeken en tutorials voor beginnende Microsoft Basic-, Small Basic-, Visual Basic- en Visual C #-ontwikkelaars van alle leeftijden.

[Kleine basisboeken](#) > [kleine basisboeken](#) > [_programmeerspellen met kleine basis-](#) > **Hoofdstuk 1:**
Schrijfprogramma's
