

## Carbon Programmeren

### Doelgroep Cursus Carbon Programmeren

De cursus Carbon Programmeren is bestemd voor developers die kennis willen nemen van de mogelijkheden van de Carbon programmeer taal.

### Voorkennis Cursus Carbon Programmeren

Om aan deze cursus te kunnen deelnemen is kennis een programmeertaal zoals C++, Rust, Java of C# wenselijk.

### Uitvoering Training Carbon Programmeren

De cursus is een afwisseling van theorie en praktijk. De theorie wordt behandeld aan de hand van slides en demos en wordt gevolgd door praktijk oefeningen.

### Certificaat Cursus Carbon Programmeren

De deelnemers krijgen na het goed doorlopen van de cursus een certificaat van deelname aan Carbon Programmeren.

**Duur: 2 dagen**

**Prijs: € 1499**

[Open Rooster](#)



**Carbon Programming**



## Inhoud Cursus Carbon Programmeren

In de cursus Carbon Programmeren leren de deelnemers software te ontwikkelen in de volgende generatie programmeertaal Carbon. Carbon is door Google gelanceerd om mogelijk in de toekomst als vervanger te kunnen dienen van C++. Carbon is een open source-project dat bedoeld is om de beperkingen van C++ op te lossen.

### Intro Carbon Language

De cursus Carbon Programming gaat van start met een introductie in de Carbon programmeertaal en behandelt de doelstellingen en design principles ervan. Deelnemers krijgen een fundamenteel inzicht in Carbon als potentiële opvolger van C++ en de bijbehorende moderne tooling.

### Variabelen en Data Types

Vervolgens wordt aandacht besteed aan variabelen en data types in Carbon, waaronder primitive types, compound types en type inference. Aan de orde komt hoe variabelen van verschillende data types in Carbon worden gedeclareerd, geïnitieerd en gemanipuleerd en ook packages in Carbon worden behandeld.

### Control flow en Functies

Vervolgens komen control flow structures zoals conditionals en loops aan bod, evenals functies en hun parameters. Deelnemers leren hoe ze modulaire en herbruikbare code kunnen schrijven met behulp van functies en hoe control flow in Carbon programma's wordt toegepast.

### Data Structures

Ook data structures zoals arrays, lists, maps en sets in Carbon en het gebruik ervan worden besproken. Het omgaan met null values, pointers en generics staan eveneens op het programma van de cursus Carbon programmeren.

### Classes en objects

Vervolgens worden object georiënteerde programmeer concepten in Carbon behandeld, waaronder classes, objects, inheritance, virtual functions en polymorfisme. Deelnemers leren hoe ze classes kunnen definiëren en gebruiken om entiteiten uit de echte wereld te modelleren en object instanties in Carbon te maken.

### Memory Safety

Tenslotte worden de memory safety functies in Carbon uitgelegd waarmee veel voorkomende programmeer fouten zoals memory leaks en buffer overflows worden voorkomen. Carbon's memory management techniques en best practices voor het schrijven van veilige code worden behandeld.

## Modules Cursus Carbon Programmeren

<b>Module 1 : Intro Carbon Language</b>	<b>Module 2 : Variables and Data Types</b>	<b>Module 3 : Control Flow and Functions</b>
Goal of Carbon C++ Limitations Carbon as Successor Interoperability Modern Tooling Package Manager Optimal Learning Curve Build from Scratch Modular Code Generics	Comments Declaration in Carbon var and let Variables Data Types Numerics Strings Type Inference Auto Keyword Packages	Control Flow Conditionals Loops while Loop for Loop match Functions fn Keyword Parameter Passing Return Types
<b>Module 4 : Data Structures</b>	<b>Module 5 : Classes and Objects</b>	<b>Module 6 : Memory Safety</b>
Tuples Multiple Coordinates Pointers Absence of Null Pointers Optionals Arrays Indexed Access Generics Methods Structs	Objects in Carbon State and Behavior Data Members Member Functions Class Methods Alias Declaration Meaning of Self Access Modifiers Inheritance Virtual Functions	Secure Memory Management Dangling Pointers Buffer Overflows Spatial Memory Safety Array Boundary Checks Temporal Memory Safety Use After Free Use After Return Type Safety Data Race Safety