

Programming with C++ Eğitimi

Süre: 5 Gün

Açıklama

C++ Programlama Eğitimi, C++ sınıfları, nesne yönelimli tasarım ve bellek yönetimi konularını ele alan 5 günlük bir eğitim programıdır.

İşaretçiler, sınıflar, yapıcılar ve yıkıcılar, operatörler, konteynerler ve nesne ilişkileri teorik açıklamalarla uygulamalı oturumlarda pekiştirilir.

Bu eğitim, en iyi uygulamaları zorlayan standart kütüphaneleri ve programlama özellikleri ile açık ve güçlü kod yazmayı teşvik etmeye odaklanarak birçok soruna yol açabilecek kavram yanılgıları ve zayıf programlama uygulamalarının ortadan kaldırılmasına yardımcı olur.

Bu, eğitimde işlenen C++ sözdizimi ve nesneye dayalı programlama tekniklerinin pekiştirilmesi için tasarlanmış, her teknik bölüm için farklı öğretim şekillerini ve uygulamalı oturumları bir araya getiren büyük oranda uygulamaya ağırlık verilmiş bir eğitimidir.

Bu eğitimde neler öğreneceksiniz?

- Nesneye dayalı ana kavramlar ve kelime dağarcığı
- Temel ve birleşik veri türlerini kullanma
- Sınıfları belirleme ve kullanma
- Sınıf üyesi işlevleri yazma
- İşaretçi ve dinamik bellek kullanımı
- Yapıcıları ve yıkıcıları kullanma
- Verimli ve güçlü kod yazma
- Toplama ve ilişkilendirme kullanarak diğer sınıflardan yeni sınıflar oluşturma
- Devralma kullanarak diğer sınıflardan yeni sınıflar oluşturma
- Şablon sınıfları dahil konteyner sınıflarını kullanma
- Operatör aşırı yüklemesini kullanma
- Çok biçimli davranışa sahip kod yazma ve tasarlama

Eğitim İçeriği

Course Introduction

- Course Prerequisites

- Course Objectives
- Course Delivery
- Course Practicals
- Course Structure

C ++ Programs

- Key features of C++
- Identifiers and keywords
- Simple declarations, expressions and statements
- Basic I/O
- Layout
- Guidelines

Fundamental Data Types

- Built-in types
- Integer numbers
- Floating Point numbers
- Characters
- Booleans
- Assignment
- Compound Assignment
- Increment and Decrement
- Defining constants
- Type conversions

Composite Data Types

- Defining and using enumerations
- Built-in arrays and their limitations
- Using the vector class
- Built-in strings as character arrays
- Using the string class
- Defining and using structures

Control Flow

- Simple and compound statements
- Selection with if else and switch statements
- Conditional expressions
- Looping with while and for statements

Functions

- Declaring, calling and defining functions
- Overloading
- Default arguments
- Scope issues
- Pass by copy
- Pass by reference
- Inline functions
- Header files and source files
- Pitfalls and guidelines

Object Concepts

- Object behaviour
- Object state
- Object identity, Object-oriented programming
- Classes
- Encapsulation

Using Classes

- Associating functionality with data
- Class definitions
- Public and private
- Queries functions and modifier functions
- Struct vs class

Pointers

- Concepts and syntax
- Pointers to structured types
- Pointers for encapsulated objects
- Null pointers
- Pointers vs. references

Implementing Classes

- Defining member functions
- Object identity
- The this pointer
- Initialisation
- Constructors
- Default constructors
- Member Initialisation

- Scope issues
- Inlining member functions

Operator Functions

- Operators as functions
- Global operators
- Member operators
- I/O stream operators
- Pitfalls and guidelines

Object Relationships

- Associations and their implementation
- Compositions and their implementation
- Navigation
- Delegation
- Multiplicity

Dynamic Memory

- The need for dynamic memory
- Dynamic objects
- Using new and delete
- Dynamic arrays
- Using new[] and delete[]
- Destructors

More Pointers

- Pointers and arrays
- Pointer arithmetic
- Pointers as array iterators
- Pointers and const
- Pointers vs. references

Containers

- Container concepts and classification
- Template classes
- Standard containers
- Vector
- List
- Iterators

- Template functions
- Algorithms

Copying

- Copy construction
- Copy assignment
- Compiler generated copy behaviour
- Problems
- Solutions

Ön Koşullar

Temel programcılık bilgisine sahip olmak