



Mise à niveau pour les développeurs C# et Java

Windows 8 a relancé l'intérêt du langage C++ auprès de développeurs pratiquant le C# et Java. Sur la base de vos connaissances objets, ce cours vous assurera une transition douce vers le C++ 11. Vous apprendrez comment C++ et sa librairie standard permettent d'écrire du code efficace et robuste pour vos applications.

Objectifs

Mise à niveau pour les développeurs C# et Java

Moyens pédagogiques

- Présentation du formateur et du programme.
- Présentation et écoute de chacun de stagiaires.
- Apports didactiques pour apporter des connaissances communes.
- Mises en situation de réflexion sur le thème du stage et des cas concrets.
- Méthodologie d'apprentissage interactive et participative.
- Exercices et études de cas concrets.
- Temps d'échanges.
- Accompagnement pédagogique individualisé.

Formateur

Les formateurs de CROSSTHINK sont des experts de leur domaine, disposant d'une expérience terrain qu'ils enrichissent continuellement. Leurs connaissances techniques et pédagogiques sont rigoureusement validées en amont en interne.

Suivi de l'exécution et évaluation des résultats

- Accueil des stagiaires dans une salle dédiée à la formation / A distance
- Feuilles de présence.
- Documents supports de formation projetés.
- Mise à disposition du stagiaire des documents et supports de formation.
- Tout au long et/ou à l'issue de la formation : Evaluation des acquis des stagiaires via des exercices, des QCM, des QUIZZ, des mises en situation et/ou des cas pratiques.
- Enquête de satisfaction.
- Attestation de fin de formation.

EN BREF

Durée : 2 jours

Tarif : 990€ HT

Public et pré requis

Cette formation ne nécessite pas de prérequis.

FORMATIONS A DISTANCE

Contactez-nous
contact@crossthink.fr

PROCHAINES DATES

Nous consulter



Mise à niveau pour les développeurs C# et Java

Programme

Rappel des concepts de base du développement C++

- Les différents fichiers sources (.h, .cpp...).
 - Linkage et compilation de code écrit en C++.
 - Ecriture de classes et structures : la syntaxe de base.
 - Réutilisation de composants déjà compilés : .lib, .dll, .h etc.
- Travaux pratiques *Création d'une solution multi-projets en C++, avec linkage statique et dynamique.*

Présentation des concepts C++ avancés

- Instanciation « statique » versus instanciation « dynamique » d'objets.
 - Les différents types de constructeurs (par défaut, par copie...).
 - Gestion de la mémoire sans garbage collector (manuellement ou en utilisant les smart pointers de la librairie standard).
 - Héritage multiple : cas pratiques et pièges à éviter.
 - Les Templates.
 - Orientation objet et comparaison avec C#.
- Travaux pratiques *Atelier de manipulation des smart pointers de la librairie standard.*

Librairie standard : la BCL de C++

- Représentations mémoires possibles d'une chaîne de caractère et manipulation avec `std::string` et `std::wstring`.
 - Introduction à la stl (standard template library).
 - Containers et itérateurs, les collections C++.
- Travaux pratiques *Atelier de manipulation de chaînes de caractères et de collections en C++.*

Syntaxe C++ avancée

- Inférence de types et mot-clef automatique.
 - Orientation fonctionnelle du langage : Lambdas, Functors... et utilisation avec la stl.
 - Asynchronisme et parallélisation avec C++ 11.
 - Utilisation d'objets COM Microsoft, avec et sans smart pointers.
- Travaux pratiques *Exercice d'introduction à la parallel platform library, et manipulation d'expressions Lambda. Cas pratique de consommation d'objets COM, illustration avec Direct 2D.*

C++ et l'interopérabilité

- Publication d'objets COM Windows réutilisables.
 - Interopérabilité avec .Net sur le Desktop : C++ CLI.
 - Interopérabilité avec .Net et JavaScript dans les applications Metro : C++ / CX.
- Travaux pratiques *Création d'une application métro hybride C# / Xaml + C++ / Direct 2D.*



CROSSTHINK