



Formation : C# Perfectionnement

Référence : C#2

Durée : 4 jours

Dates : voir calendrier INTER

Niveau : expert

Support de cours : français

Lieu : Paris ou sur site

Tarif : 2200 €HT

Nos engagements pédagogiques !

- Des salles équipées de vidéoprojecteurs et de PCs tout équipés
- 50% de cours théorique
- 50% d'exercices pratiques
- Des formateurs qualifiés et certifiés
- Chefs de projets, ingénieurs dans l'industrie
- Ayant tous plus de 10 ans d'expérience dans l'animation de formations et la réalisation de projets

Renseignement et inscription :

Contactez votre conseiller formation Styrel

Tél. : +33 7 83 07 61 67

formation@styrel.fr

Description

Cette formation sur le langage C# vous permettra de maîtriser tous les concepts fondamentaux liés au développement des applications .NET (WPF, LINQ, Service Windows, Multithread) et à la communication (Socket, WCF, WebServices REST et SOAP).

Public concerné

Cette formation C# avancé s'adresse aux développeurs expérimentés d'applications d'entreprises ainsi qu'aux chefs de projets.

Objectifs de la formation

À l'issue de la formation, le stagiaire sera capable de :

- Comprendre le rôle du Framework .NET et savoir utiliser les classes
- Configurer et utiliser Visual Studio 2017
- Maîtriser la syntaxe du langage C#
- Utiliser les fonctionnalités avancées du C#

Prérequis

Avoir suivi la formation C# est une bonne base pour suivre ce cours avancé.

Programme de la formation

- **Rappels et compléments sur le Framework .Net**
 - Synthèse sur l'architecture du Framework .NET
 - La place de C# dans l'architecture globale de .NET
 - Les nouveautés du Framework .NET
 - Comparaisons de C# avec d'autres langages (Java, C++...)
 - Les options de compilations de CSC (CSC, Roslyn)
 - Créer une bibliothèque réutilisable (dll)
 - Rappels sur le C# (Types et méthodes anonymes, Délégués, Prédicats, Expression lambda, Méthode d'extension...)
- **Performances et mémoire (code non managé) - Optimisation C#**
 - Terminologies systèmes (Processus, Threads, Handles...)
 - Contexte d'exécution d'un processus (Jeton utilisateur, UAC, Virtualisation...)
 - Comprendre les couches basses de .NET (CLR, assembly, module...)
 - Gestion du tas et de la pile par .NET (Garbage collector)
 - Risque et intérêt d'utiliser du code « non managé »
 - Utilisation des pointeurs en C# (unsafe)
 - Exemples de code non managé

Styrel : bien plus qu'une formation !

- Intégrateur en Informatique Industrielle depuis 30 ans
- Une vision terrain indépendante de tout constructeur
- Des solutions complètes, du logiciel au matériel, en passant par le service

Nos certifications

N° d'Agrément Formation Continue :
11 91 02 737 91

DataDock



➤ Créer et déployer un service Windows

- Différences entre un service Windows et une application
- Présentation de la classe ServiceBase (OnStart, OnStop, OnPause, OnRestart)
- Création d'un service avec Visual Studio (template Windows Service)
- Paramétrage du service (CanStop, CanShutdown...)
- Déploiement d'un service Windows
- Vérification du fonctionnement du service dans le Panneau de configuration

➤ Multithreading sous .NET

- Présentation de la programmation parallèle
- Architecture d'un programme multi-threads
- Méthodes de gestion des threads
- Fonctionnement « normal » d'un programme C#
- Cycle de vie d'un thread
- Les threads et les pools de threads
- La Task Parallel Library
- Paralléliser les boucles
- Spécificités techniques de la programmation multi-threads (mémoire, verrou, synchronisation...)
- Les outils dans Visual Studio

➤ Introduction à la communication sous .NET

- Synthèse des différents types d'échanges
- Technologies sous-jacentes dans .NET
- Présentation des différentes API (Socket, Web services, WCF...)
- Pourquoi utiliser les delegate dans la programmation réseau ?

➤ Services Web ASP.NET (SOAP et REST)

- Présentation des types de web services
- Intégration de .NET dans le serveur Web IIS
- Implémentation d'un Webservice SOAP (.asmx)
- Méthode SOAP et le WSDL
- Implémentation d'un Webservice REST (WebAPI)
- Consommer un Webservice REST (AJAX avec jQuery)

➤ Programmation réseau avec les sockets

- Rappels sur le vocabulaire réseau (@IP, n°port, TCP, UDP)
- Quelles différences entre développer « par-dessus » TCP ou UDP ?
- Qu'est-ce qu'un socket pour le développeur ?
- Les méthodes de la classe Sockets (listen, bind, accept, send, receive...)
- Envoi et réception de message (BeginReceive, BeginSend)
- Exemple de code général côté serveur et côté client
- Gestion du pool de connexion (multi clients)
- Implémentation à l'aide de .NET Remoting

➤ WCF

- Architecture orientée Services
- Les contrats de service
- Comprendre et utiliser les points de terminaison
- Configuration et hébergement d'un service

Nos certifications

N° d'Agrément Formation Continue :
11 91 02 737 91

DataDock



- Service bidirectionnel
- Créer un service WCF sans serveur
- **WPF (Windows Presentation Foundation)**
 - Présentation de WPF
 - Rappel sur Windows Forms
 - Implémentation de WPF
 - Scénario de migration et interopérabilité
 - Design et outils (Blend)
 - Implémentation Async/Await Contrôles de positionnement
 - Utilisation des contrôles utilisateurs
 - Présenter et interagir avec les données (Databinding)
- **Créer ses propres contrôles utilisateurs (User Controls)**
 - Création d'un gestionnaire d'événements pour un contrôle
 - Les attributs d'un contrôle utilisateur (Category, Browsable, DefaultEvent...)
 - Utilisation des contrôles avec un formulaire Windows
 - Validation des entrées des utilisateurs
 - Gestion du glisser/déposer de nos propres contrôles
 - Créer des contrôles dynamiquement
 - Ajouter des fonctionnalités aux contrôles
 - Ajout de prise en charge Design-time pour les contrôles
- **LINQ (Language Integrated Query)**
 - Présentation de LINQ
 - Des requêtes/traitements au-delà des simples bases de données ...
 - Comment utiliser LINQ dans un programme C# ?
 - Manipuler des collections d'objets avec LINQ (LINQ to Object)
 - Gérer des documents XML avec LINQ (LINQ to XML)
 - Gérer le mapping entre les types C# et les enregistrements d'une base (LINQ to SQL)
 - Interroger les données d'un dataset (LINQ to Dataset)
 - Ecrire des requêtes sur un modèle conceptuel (LINQ to Entities)
 - LINQ et les performances

Évaluation et suivi de la formation

A l'issue de chaque demi-journée de formation, une feuille d'émargement est signée, à la fois par le stagiaire et le formateur. Des exercices pratiques de programmation en cohérence avec les objectifs de ce programme, permettent de vérifier l'acquisition des compétences tout au long de la session. Une attestation de fin de formation est remise au stagiaire lui permettant de faire valoir l'acquisition de ses nouvelles compétences.